DEVELOPING A DYNAMIC RECOMMENDATION SYSTEM FOR PERSONALIZING EDUCATIONAL CONTENT WITHIN AN E-LEARNING NETWORK

A Thesis

Submitted to the Faculty

of

Purdue University

by

Marzieh Mirzaeibonehkhater

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Electrical and Computer Engineering

August 2018

Purdue University

Indianapolis, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Brian King, Co-Chair

Department of Electrical and Computer Engineering

Dr. Ali Jafari, Co-Chair

Department of Computer Information Technology

Dr. Hongbo Liu

Department of Computer Information Technology

Approved by:

Dr. Brian S. King

Head of Graduate Program

I would like to express my thanks and appreciation to my Mom and my Dad and my caring brother and sisters whose love has supported me through my travels abroad and my academic endeavors. I want to thank Ms. Sherrie Tucker and also my colleagues at CyberLab, Menguan Zhao (Alice) and Ms. Allison Wigginton and of course my classmates, Katuta, Saurab, and Homero and everyone who supported me during all these challenging years with their kindness and their guidance to pursue my career goals and accomplish this project during these years.

ACKNOWLEDGMENTS

My thanks and appreciation for this learning opportunity also go to my committee: Dr. Brian King, Dr. Ali Jafari, and Dr. Hongbo Liu. I would like to express my appreciation to Dr. Ali Jafari and his family for providing this opportunity for me to pursue my master degree. This work would not have been accomplished without the financial support of the Jafaris Fellowship award. I am especially indebted to Dr. Brian King, Chair of Department of Electrical and Computer Engineering as my teacher and my mentor, he has taught me more than I could ever give credit here. I am grateful to all of those with whom I have had the pleasure to work during these years.

TABLE OF CONTENTS

Page

LIST OF TABLES vii

LIST OF FIGURES viii

ABSTRACT x

1 INTRODUCTION 1

1.1 Motivation 1

1.2 Problem Statement 2

2 EXISTING METHODS 5

2.1 Introduction to Recommender Systems (RS) 5

2.1.1 Goals of Recommender Systems 5

2.2 Collaborative Filtering 7

2.3 Memory-based 8

2.4 Model-based 9

2.5 Cold-start Problem 9

2.6 Sparsity 10

2.7 Matrix Factorization 10

3 COURSE NETWORKING (CN) 13

3.1 Overview 13

4 OUR METHODOLOGY 25

4.1 Introduction 25

4.2 Design Requirement 26

4.3 Design Decision 27

4.3.1 Feature Selection 28

4.3.2 Classification as a Feature Selection 33

4.3.3 Content Analysis 38

Page

4.4 Ground Truth Matrix 59
4.5 Matrix Factorization 60
4.6 Evaluation of our Recommender System 63
5 CONCLUSION AND FUTURE WORK 71
REFERENCES 73

LIST OF TABLES

Table Page

4.1 User Features 28

4.2 Post Features 32

LIST OF FIGURES

Figure Page

3.1 A New Course or Network Creation 16

3.2 Users Tagged in the Post 18

3.3 Rating Star, Scale to Show Level of Interest on a Post 19

3.4 Anar Seeds Represent of User's General Activity 20

3.5 Badges Icon Which is in the Left Side Bar in the CN 21

3.6 Some of the Predefined Badges and their Description by Clicking on the Badges Icon 22

3.7 One of the Posts with the Highlighted Music Keyword Searched in the Taskbar 23

3.8 Anar Icon to View Anar Table of a Course 23

3.9 Anar Tool Bar Provided by the Course Instructor 24

4.1 Outline of our Proposed Method 26

4.2 Bag-Of-Words Model on a Post Sample 37

4.3 Bag-Of-Words Model on all the Global Posts 37

4.4 Using 80% of the Output of Bag-Of-Words to Train the Classifier 39

4.5 Logistic Regression 43

4.6 SVM (Large Margin) Classifier 44

4.7 Non-linear SVM with Polynomial Kernel 50

4.8 Non-linear SVM with RBF Kernel 50

4.9 Non-linear SVM with Linear Kernel 51

4.10 Polynomial Kernel with Degree1 51

4.11 Polynomial Kernel with Degree2 52

4.12 Polynomial Kernel with Degree3 52

4.13 Polynomial Kernel with Degree4 53

Figure Page

4.14 Linear SVM 53
4.15 Logistic Regression 54
4.16 Accuracy of Different Classifiers 55
4.17 Confusion Matrix Showing the Accuracy of the SVM Classifier 55
4.18 Classifications 57
4.19 Accuracy of Logistic Regression Classifier for Different Learning Rates . . 57
4.20 Overfitting and Underfitting in Logistic Regression 58
4.21 Output of Using PCA for Data Visualization 59
4.22 Ground Truth Matrix 60
4.23 Hybrid Function Considering Explicit and Implicit Features 61
4.24 Frequency of the Number of Entries in GTM Matrix 64
4.25 Outline of Evaluation of our Recommender System 65
4.26 Neutral Global-posts 67
5.1 Post-Course Rating Values 72

ABSTRACT

Mirzaeibonehkhater, Marzieh. M.S.E.C.E., Purdue University, August 2018. Developing a Dynamic Recommendation System for Personalizing Educational Content within an E-Learning Network. Major Professors: Brian King and Ali Jafari.

This research proposed a dynamic recommendation system for a social learning environment entitled CourseNetworking (CN). The CN provides an opportunity for the users to satisfy their academic requirement in which they receive the most relevant and updated content. In our research, we extracted some implicit and explicit features from the system, which are the most relevant user feature and posts features. The selected features are used to make a rating scale between users and posts so that represent the link between user and post in this learning management system (LMS). We developed an algorithm which measure the link between each user and post for the individual. To achieve our goal in our system design, we applied natural language processing technique (NLP) for text analysis and applied various classification technique with the aim of feature selection. We believe that considering content of the posts in learning environments as an impactful feature will greatly affect to the performance of our system. Our experimental results demonstrated that our recommender system predict the most informative and relevant posts to the users. Our system design addressed the sparsity and cold-start problems, which are the two main challenging issues in recommender systems.

1. INTRODUCTION

1.1 Motivation

One of the principal goals of recommendation systems is to recommend relevant items to users for different purposes such as learning, increasing sales by recommending products or recommending the user's favorite movie. Users interact with each other through social media in web 2.0. Furthermore, their activities to the system affects their viewing items in the virtual community, compared with Web 1.0 where users are limited to view specific fixed items. Recommendation systems (RS) provide content preferences to the users based on the user preferences and users necessities. The Web 2.0 provides an opportunity for the people to interact with each other, share their information and having collaboration with each other throughout the world conveniently. In Web 2.0 developments, an intelligent agent is used to recommend individual users various items based on the user preferences in terms of online information and various resources among a very large of web data is a challenging issue [1].

In order to build different models of recommendation systems to make an automatic prediction for individual users via using user profiles and also user-generated contents, one must face some challenging issues. This requires systems to ameliorate the problems. The goal of all the recommendation systems is to address the issues by recommending items to the users via considering users preferences and developing a dynamic system with the most accuracy. The current recommendation systems face some challenging issues to provide content preferences for the targeted users. Indeed, in such networks, there are limited information and attributes about users and items which makes it difficult for the models to build a robust item prediction. This thesis

proposes a new hybrid recommendation system (HRS) for social media network with limited object features for predicting personalized news feeds to cover the individual users' preferences with the most accuracy.

1.2 Problem Statement

In Web 2.0 applications like Amazon, Facebook, and etc, the user's feedback to the system is very important, especially in business transactions. Indeed, all of them are trying to satisfy the customer's desires in terms of user's requirements. Developing a *Recommendation System (RS)* with the most accuracy to automatically serve the users preferences not only helps to increase their sales but also allows the users to access the items that they would prefer to see.

One of the principal goals of recommendation systems is to recommend relevant items to the users for different purposes such as learning, increasing sales by selling their product or recommending the user's favorite movie. The user interests and their preferences are different from each other and it depends on many cases. It could be based on the mood of the users, his or her personality and also the purpose of developing the application for that network. For example, we should not expect a social network like Netflix to recommend clothes, as it is used for recommending movies. Moreover, different recommendation models are developed for recommending items to the users via using implicit and explicit user's ratings in order to provide the best match items to the users. As a consequence, if a user isolated, which means he/she is not active in the social network, then predicting items for him/her with low activity frequency would be very difficult. So, the main purpose of the recommendation systems is to provide the most relevant items to all the users based on their preferences. The current recommendation systems face some challenging issues to provide content preferences for the targeted users. In some social environments, we have lack of information and attributes about users. We proposed a new *Hybrid*

and Dynamic Recommendation System (HDRS) for a social and learning network with limited object features to cover the individual users preferences with the most accuracy.

We tested our HDRS on the CourseNetworking (CN). The CN provides one of the social learning networks shares similar aspects to social networks like Facebook and Twitter, however an academic purposes.

In this social-learning network, the CN users submit their contents or resources in the form of posts, and the posts are being shown to the user classmates or also different classmates belonging to the same category via a tool that is called Global posts. Now, this environment is static in order to show the posts to the users. As a consequence, we constructed a smart recommendation engine, in order to help the users to access to the posts of their global classmates corresponding to their academic necessities and their relationship. Our goal is to make an intelligent engine that recommends the best posts on the top of each user's home page by considering the following concepts:

- Considering the content of the posts that the target user and all his/her neighbors (those who have the same interest with the target user) had most trend to see, and also the posts that the users would have some reaction to, considering user-post features which represents the link between users and posts.
- Target user; a specific user that we want to recommend him/her the best top k sets of post on the top.
- Considering the correlation between the users and posts via analyzing the posts that they shared with their global classmates and considering the users reaction to the posts.
- Considering the user's selected skills.

The top k number of recommended posts based on our methodology are the posts that the users would probably have the most interest to see on the top when they click on "Global posts". For example, consider you are in the course category Business, and

you click on the global posts, and there are more than $10,000$ posts. Our recommender system would show the best-matched posts on the top among $10,000$ posts to the users based on his/her educational necessities and some of his/her social activities in the system.

In our hybrid model, we combined the machine learning classifier and memory-based method as a regression-based model by using the concept of neighborhood-based models and considering the implicit features in the system. In this case study, the users and posts in the business course category are analyzed. The proposed method overcomes the sparsity problem exist in the memory-based model plus the cold start issue in our experimental results applying dynamic algorithm and the alternative least square (ALS) factorization method. ALS is a robust and optimized method over the known entries.

The chapters of this thesis study are as follows. The first Chapter discussed the introduction of our case study and explained the statement of the problem. The traditional collaborative filtering methods and the recommendation systems challenging issues are discussed in the Chapter 2. The existing methods applied for different dataset by the aim of developing a recommendation system is discussed in Chapter 3. Chapter 4 consideres course networking as a social and learning network. Our proposed hybrid and dynamic recommendation system is explained in chapter 5, and the experimental results and future work are discussed in Chapter 6.

2. EXISTING METHODS

2.1 Introduction to Recommender Systems (RS)

The principle of using a recommender system in social networks and e-commerce is to help users access their item preferences in a data-driven manner. There are various models of recommender systems (RS) that have been proposed, and generally in all of the RS models are utilized the robust predictions based on user behavior and users' interaction to the system. There are various models of recommendation systems and all of them infer and suggest user principle requirements. However, various challenging issues such as the cold-start issue and data sparsity exist in recommender systems [2] which will be explained in details as follows. Section 3.1 discusses the goal of recommendation systems. Section 3.2 explained about some basic models of recommender systems, and in Section 3.4 hybrid recommender systems are discussed. In Section 3.5, a variety of evaluating recommender systems are studied. In Section 3.5, the domain of challenged in recommendation system is discussed.

2.1.1 Goals of Recommender Systems

The goal of using recommendation systems is to show the most relevant items to the users by recommending relevance selected items. In e-commerce, recommendation systems are vitally important to predict and recommend selected items in order to increase the profit [3]. Base on the goal of the system, different type of items are recommended in the system. For example, in social networks like Facebook, they do not directly recommend products or services, rather they are recommending posts to the users, or they may recommend social connections [3].

The Simultaneous Co-clustering and Learning (SCOAL) algorithm applied in [4] can address the cold start issue since it can recommend items to user, regardless of size of the information about user. This algorithm takes a matrix consisting of users attributes as rows and item attributes as the columns. This prediction model is a matrix of cluster of users and items to which the new user belongs to, with high probability. This method built a classification model to predict the best classifier for a new user (cold start issue). In the case of pure cold start problem, the users' attributes in the input matrix are confined to demographical features. In this research, they compared four methods and compared their performance to address the cold start issues. They applied the models on two different datasets, Movielens and jester datasets [5].

The *Social Network based Recommender Systems (SNRS)* is a model that attempts to estimate the prediction model based on a group of users by whom the target user is influenced most. This model is what the [6] prediction model is based on. In [6], the similarity measure is based on users' rating values to the different items, restaurants in this case, by considering only immediate links in the social network.

In [7] Chaney et al. proposed SPF, which incorporates social information in Poisson factorization method [8] for making recommendations. Poisson factorization is a probabilistic tool which based on users preferences estimates the probability that a user might like an item. In addition, preferences and rating of the users friends were taken into account to figure out level of influence that affect the user.

The [9] model-based approach is employed on Epinions.com and Flixter.com datasets. In this model, they used matrix factorization-based technique merged with Trust propagation. The experimental results show that using the matrix factorization method increased the accuracy of the system in predicting the relevant items to the users and also addressed a cold-start problem for a new user with a better accuracy in comparison with the STE.

The item-based method is estimated unknown items for a target user by considering known similar items. This method has a better accuracy in comparison with the user-based models [10] [11]. Regarding the item-based model, Deshpande and Karypis discussed this model in detail. The proposed model in [12] applies implicit feedback datasets and used the concept of confidence level. Considering the implicit features used in the online television services and leveraged the indirect relationship between users and TV-programs. In our thesis, we have applied the idea of implicit feedback in our proposed hybrid recommendation system to predict special posts based on the users preferences.

As reported in this case of social network analysis, the *Enhanced Content-based Algorithm (ECSN)* model was used in [13] to recommend relevant information to the users of a social network. In the ECSN model, the target users preferences and his/her users friends leveraged in the system. The model applied to 920 users and 1398 academic items in 14 weeks. The output of the proposed ECNS model indicates that the prediction model worked very well with a higher level of accuracy. The Markov Cluster Algorithm (MCL) clustering method applied in paper [14] to improve the accuracy of the collaborative filtering method. In this paper, they used the last.fm dataset to test their proposed model which contain listening records of users and user's relationship in the network. The experimental results of this method show that considering the time factor would increase the accuracy of the collaborative filtering.

2.2 Collaborative Filtering

In the basic collaborative filtering method, we are using the rating scale specified for the items via using a matrix in such a way that the rows in the matrix corresponding to the users and columns corresponding to the items. The challenging issue

is the sparsity of the matrix since usually the number of elements corresponding to the rating value that the user gave to the item is very low in comparison with the whole items exist in the large dataset.

The idea behind the collaborative filtering method is that the future rating value of unobserved items can be predicted using the observed items rated by the users. In the unobserved item prediction process using collaborative filtering model, the memory-based and model-based approach leveraging on the observed items to impute rating value for the unseen items. The memory-based and model-based are briefly explained in this chapter.

2.3 Memory-based

These models predict unobserved items on the basis of neighborhoods. The item based, and user-based methods are the approach used in neighborhood based for prediction. In user-based method the k most similar users with the target user are determined and then the rating value for the unseen item for the target user is computed using the average rating values given by similar users to that specific item. In this method, we are considering that the target user and his/her similar users shares the same interest. To find similar users, the similarity functions like cosine similarity can be used. The other model, item-based collaborative filtering leveraged on the set of similar items to predict the rating value for the target item. Therefore, the similarity function computes the similarity between columns in the defined matrix.

$$\hat{r}_{uj} = \mu_u + \frac{\sum_{v \in P_u(j)} Sim(u, v) \cdot (r_{vj} - \mu_v)}{\sum_{v \in P_u(j)} |Sim(u, v)|}. \tag{2.1}$$

Equation (2.1) estimated the unknown rating of item i from user u. In the memory-based method, both user-based and item-based approach predict the rating value of the unknown elements using the linear function of the known rating values. In item-based, we estimate the rating value of an unobserved item using the rating values of the sets of similar items to the target item given by the target user. In user-based method, set of k similar users who rated item j are considered to predict

the rating value of user u to the item j. The Equation (2.1) is used to predict rating value of specific user u to the item j. Also, in Equation (2.1), if we consider all the ratings of the similar users instead of k nearest user then the Equation (2.1) operates as a linear regression models. As we know, the linear regression model considers all the previous behavior in the system to predict future.

The challenging issue of the memory-based approach is that this method is not working very well when we are facing with the sparsity issue in the matrix. A sparse matrix makes it difficult to compute the rating value for the target item using the similar items or similar user's behavior in the system. However, when we are not facing the sparsity issue, this method is a robust method to find and predict the sets of items for the users.

2.4 Model-based

In this model, the machine learning methods and data mining approaches both are used in prediction process by learning the hyper-parameters using the training sets.

2.5 Cold-start Problem

One of the main issue in recommendation systems is about new users coming to the system and we don't have much information about them and their behavior in to the system. The issue of item prediction for the new users so-called cold start problem. The traditional collaborative filtering method which was a basic method to recommend items to the users is notable to tackle this issue. However, there are some methods to ameliorate this issue.

2.6 Sparsity

One of the main issue in a recommender system is the sparsity problem. The rating matrix which is created using the user's activity in the network may be very sparse. Therefore, accessing to the large dataset and more features to strongly link users and posts would address the sparsity issue.

2.7 Matrix Factorization

Matrix factorization method is an effective method to predict unobserved items using the observed rated items, so if the matrix is very sparse then this method is not a reliable approach to estimate the rating value for the unseen data since the matrix do not have access to the sufficient observed value. The singular value decomposition is one of the matrix factorization methods to do prediction analysis. The singular value decomposition method is explained as follows.

Assuming that we have a $m \times n$ matrix R. The rows corresponding to the users and columns corresponding to the items. The elements in this matrix are the rating values given to the items by the users. Mathematically as it discussed in [2] the matrix R can be factorized to the three matrices Q and Σ and P.

$$R = Q\Sigma P^T. \tag{2.2}$$

In Equation 2.2, Q is a $m \times m$ matrix. The m columns are all the eigenvectors of RR^T. P in (2.2) is $n \times n$ and the n is the eigenvectors of RR^T. Σ is an $m \times n$ diagonal matrix. The entries which called singular values are the eigenvalues of $R^T R$

In SVD method, the d number of eigenvectors are used to predict unobserved entries in the matrix. The parameter d is smaller than dim m and n and is the largest number of eigen values in our matrix. This method provides a robust estimated rating matrix using the Equation (2.3).

$$R = (Q_d \Sigma_d P_d^T). \tag{2.3}$$

Recommendation system can be used for different purposes. There exists lots of paper representing different models for item predictions. In [15], the authors compare the trust-based model with frequency-based model. The idea of utilizing trust-based model is to consider the social network and trust interactions to filter the valuable and relative information. The intention of applying this model comes from the importance of social networks, which is crucially important and impactful in "linking people, organizations, and knowledge [16].

According to the research in paper [17], the information shared daily with the people increased highly with the internet growth which make it difficult for the people to select the truthful information among all the options they have based on the researches on [18], [19], [20], [21], [22], [23]. The performance of the system based on the experimental results in [15] is very good. Based on the research of this paper, a network would have a good performance in comparison with frequency-based model when the system is dense to predict useful information.

Recommendation systems are applied across different websites such as YouTube, Facebook, Instagram, and so on. One of the main issues of a recommendation system is cold-start problem. Beside, cold-start issue, it is difficult for the agent to track the user preferences and recommend the user sets of items based on changes in his/her preferences [24]. In this paper, the authors proposed the SNetRS method which is based on the collaborative filtering approach, and the dataset that they considered belongs to the Facebook considering the user preferences in Facebook. In the SNetRS, they considered both user-based and item-based methods in their hybrid filtering algorithm that are both take the usage user-based and item-based methods proposed in [25]. The experimental results shows that considering the social networks information in recommendation system is very effective. Especially considering the time-factor model and cross-domain filtering [24].

In [26], they considered the social network information. They believe that the social network information had a valuable data which increased the performance of the RS system. So, they proposed the SoCo model which considered the contexts

and social media information. They also applied the matrix factorization method to predict user preferences. Their experimental results shows that will improve the performance of recommendation systems.

It is very important to measure the truth relationship between users before recommending items from a user to another [27]. So, in [27], the authors considered the direct and indirect trust degree between pair user and also considering the trust degree between a group of users on Epinions which is a public dataset in their proposed hybrid model [27]. The experimental result shows that their system has a better performance in comparison with MoleTrust model. The trust degree-based model needs a strong relationship between users. However, in our cased study most of the users are still isolated.

3. COURSE NETWORKING (CN)

3.1 Overview

Course Networking is a social learning environment including tools for learning managment systems (LMS), social networks and eportfolio. The Course Networking model enable users to interact with each other through posts. This proposed learning management system (LMS) provide a strong and secure foundation for the users to interact with each other.

The Course Networking (CN) provides users a forum to speak on diverse topics which can be historical, philosophical, scientifically or social topics. Notably, the Course Networking models underlying on educational goal. The CN model allows the users from diverse cultures to collaborate with each other throughout the world.

There are different features and characteristics in the CN environment that provide valuable services for the users with different roles in this social and learning management system. Some of these features are briefly explained as follows:

- **Posts**: Users use this tool to share various context related to their interests. The shared context can be a link, images, videos, and/ or pictures that uploaded from phones or desktop. Posts are one of the main and powerful ways that enable users to share content about different topics. Two types of posts exist in this environment: public posts and global posts. Users can share their posts about a specific topic with their global classmates via global posts icon. Public posts allow the users to share the content of their posts with all the CN members, also before sharing their posts users can restrict their posts to be visible for all the CN members, or their followers or his/her courses network or just only share the posts with themselves for different purposes. Users also are able to choose a topic in their posts.

- **Polls**: The polls tool allows the users to share one or multiple questions with the CN members through Yes/No questions, True/False questions, short answer questions or etc. Moreover, users can share files, images, or videos or links throughout the polls.

- **Event**: For making an announcement or event, the CN members can use this tool for different purposes like set up a group discussion, or group study and remind their personal network about the team meeting. An event also like Polls and Posts can include different types of attachments like YouTube links or images or files etc.

- **Social Engagements**: There is a graphical illustration in the CN that illustrate user's social interactions in the CourseNetworking with the other users. User's activity on the CN is measured by considering the number of reflections that they are making on each other posts and their rating star. In this graphical design, the larger the user's value is the more Anar seeds the person will accumulate.

- **Followers**: The Follow icon allows the CN users to follow different people through this icon.

- **Anar seeds**: Based on the users activities in CN, users earn Anar seeds.

- **Rating stars**: There is a star icon in the CN which allow the users to indicate the level of their interests to the posts. In this social and learning environment the three-point ratings are used which has different meaning based on the post that is public or global post scaled.

- **Reflections**: There is an option for the users to display their feedback to a post via making comment on a post which is so-called reflection.

- **Shown times**: The number of times a post was seen is shown through a shown attribute for each post is displayed.

- **Best icon on reflections**: This occurs when there are three or more reflections on a post. The Best function which is an effective feature in social and learning environment will become available to allow the users to select the best reflection among the other posts.

- **Attachments**: Users can attach files to their posts when they are creating a post. The attached file can be in different format such as a link, image, YouTube video or link from Google Drive.

- **Hashtags**: Users can use some specific, predefined hashtags when they are creating a Global post, or they can hashtags their own words in their post. The hashtag would be added in the list of Global Discussion topic besides all the other hashtags which are already used in that specific topic.

- **Repost**: The repost icon is accessible for the public posts. So, when the users log in to their account in their profile they would allow to repost a public post if they are interested in sharing that post again with the other users.

- **RememberIt**: This tool allows the users to mark a post to revisit it later.

In the next section we will discuss in greater details some of the above explained feature sets. In addition, we will discuss some other features which are used in this learning management system, in order to show how enhanced CN is compare to previous traditional learning management systems.

Course Networking is a next generation of online social and learning environment that allows the users to collaborate with each other throughout the world. In CN, users play a variety of roles. They could have student role in some courses or instructor role in some other courses. CN environment offers a possibility for the users to create and manage their personal course or network. Users can create a course to teach in an institution needs to define a very short discussion topics for their course which represent the topic that they supposed to teach. Then the students in the class would be prompted to select some specific hashtags that the instructor already defined when

they want to create a post in that course. Using hashtags would help the users to easily and quickly search for the posts related to the specific topic they are looking for. Moreover, there is an option for the CN instructors to choose whether they want to keep their course a private course which is applicable just for the users registered in that course or a public course o be available for all the CN members in that course category or a private course for themselves. Also, CN enables the course creator to choose the course level whether it is for graduate school or high school or etc. The following figures represent these features. The option for the users to create their course is in the right-hand side of the user profile, is shown in the Figure (3.1).

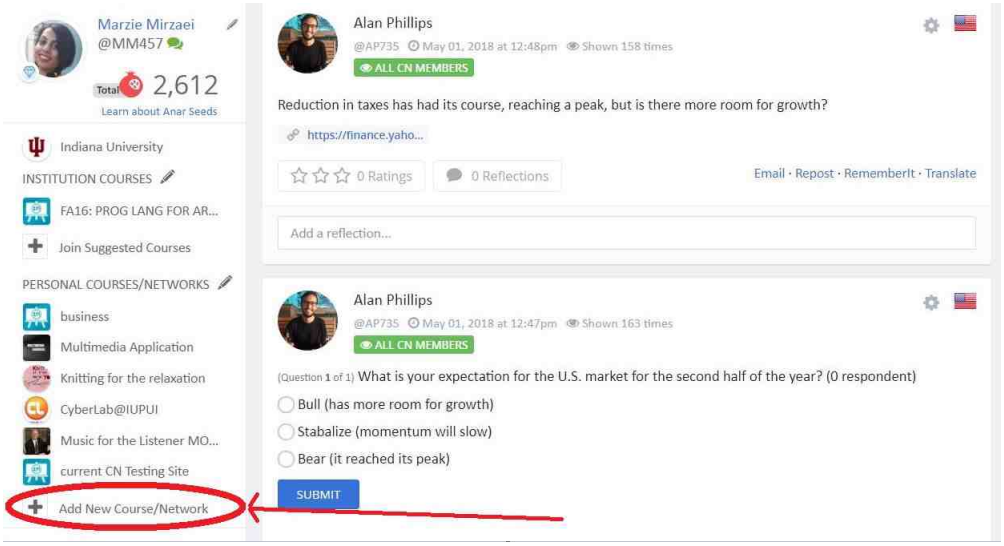

Fig. 3.1. A New Course or Network Creation

The other main option that the users from different countries throughout the world mainly use to collaborate with each other is posts. Post is a tool in the CN, which was designed for enormous capabilities. There are two types of posts in the CN: Public posts and Global posts. Public posts are all the posts that we are able to access in our profile page. The post tool is used to generate text with or without any attachment such as picture, link or etc. CN user can choose an option to share their posts with different group of users. The visibility options allow the users to share the content of their posts with different groups of users:

- **Group(s):** The users can share their posts with the CN members who they are a member of. However, the instructors can create a post and share it with any group in that course.

- **This class:** By selecting this option, the users decide to share their posts only with their classmates.

- **This course and Global Class:** Using this option allow the users to share their posts with all the users in that specific course category that their course is.

- **This course and my other courses:** This option allows the members of this course and the other courses view the post.

- **Only Me:** No one will see the post except the post creator. This option allows the user would be able to edit his/her post anytime in the future before sharing with the other users.

Global post icon is a tool in the CN which makes it unique in comparison with the other learning management systems. Global post tool is in the right-side bar after clicking on a course that we already created or registered through an instructor in the CN. The Global post option allows the users to view all the posts that shared from the global classmates throughout the world. Global classmates are all the users who are in the same course category.

Furthermore, CN as an educational and networking environment, offers an option for the users to mention specific users in which the users are tagged in the post that they supposed to create. This tagging option sends the CN members who are tagged in a post, a notification on their profile page to see that post.

- **Hashtags:** There are a set of keywords in each course that are created by CN members regardless of their roles whether they are students or instructors. Using a hashtag provide an option for the users to define their post topics which

can be predefined by the course instructor in the beginning of the course creation time or it can be a new word that highlighted by the post creator. Both of these types of hashtags are entered into the "Discussion Topic which is labeled in a left side bar in the specific course. Click a hashtag would filter out all the posts in the specific course that contains this hashtag and show them on the top. The Figure 3.2 illustrates this feature.

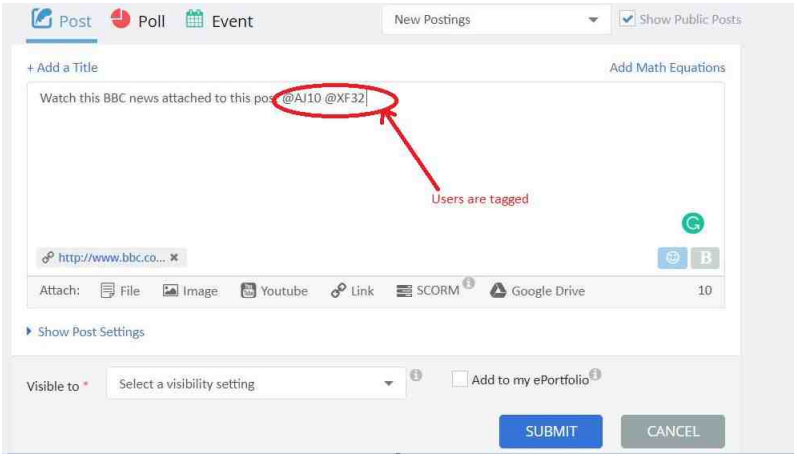

Fig. 3.2. Users Tagged in the Post

- **Rating star:** Rating star is a scale to give a rating value to the posts. The concept of the rating values are different in Public posts and Global posts. The rating values are Ok rating point, Good rating point and Great rating point. Users evaluate the posts through rating starts. Ok means they like a post but with a very low scale. In the global post, the rating scale is considered as a quality of the posts academically whether it is ok, good, or great post or not. Figure 3.3 represents this feature.

- **Anar seeds:** Anar seeds is a unique and motivation feature in the Course Networking that is designed for different academia purposes. Based on the users activity in the CN the users would receive Anar seeds.

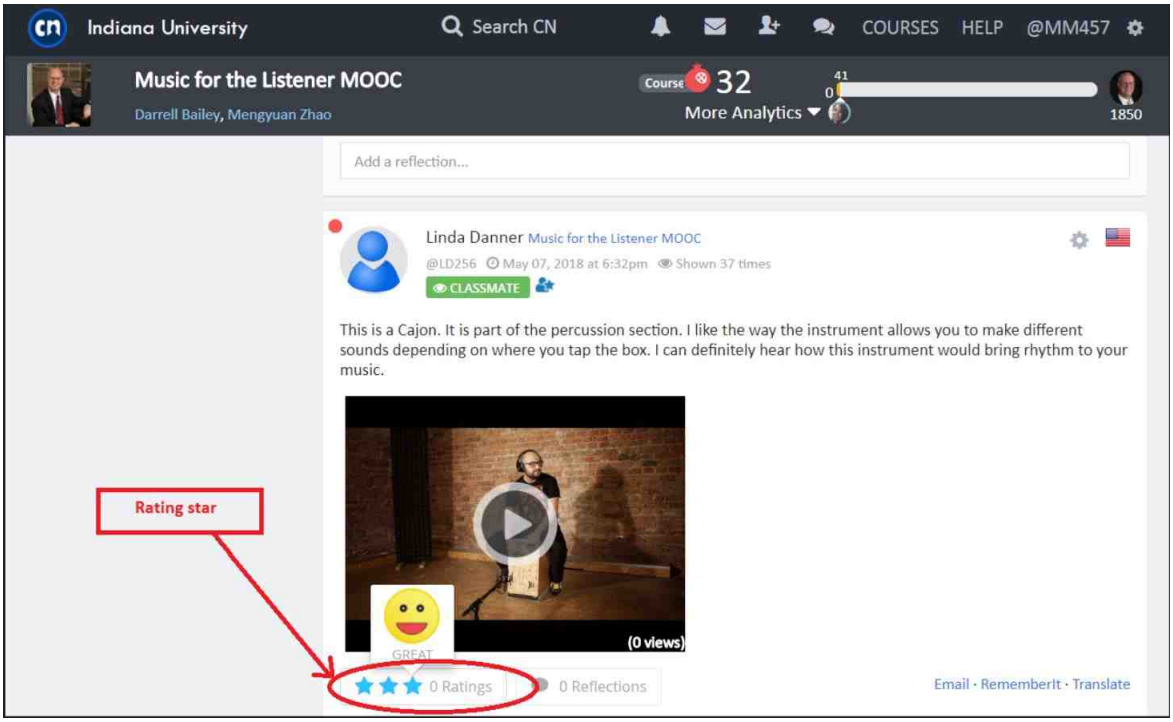

Fig. 3.3. Rating Star, Scale to Show Level of Interest on a Post

Anar seeds as a major bonus in this learning management system which are given to the students based on their activity in the CN. This feature plays an important role in the CN to represent the users general activity. This feature refers to the word Anar which means pomegranate in Persian. The total number of Anar seeds in the users homepage represents the total user activity in the CN. The Figure (3.4) illustrate the total activity of a user in this learning environment.

- **Badges:** There are some predefined badges for the users of the courses in this learning environment. The badges are available for the users to award in the specific course. The badges icon defined in the left side bar to represent an option for the users to access very quickly to the badges. When the CN users clicked on the badges from the left side bar, they would see the predefined badges, name of the badges, shape of the badges and recipients of the badges.

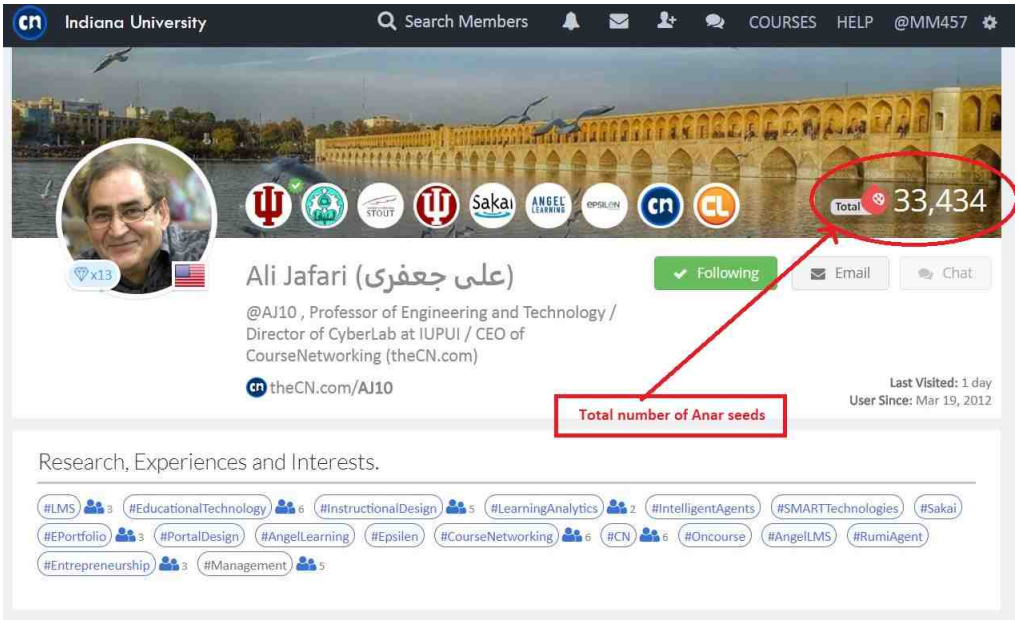

Fig. 3.4. Anar Seeds Represent of User's General Activity

The guideline badges explained all the requirement that the students in the CN needs to have in order to get the badges which would be appeared in their home page. The badges feature is illustrated in the Figure 3.5 and Figure 3.6.

- **Search Task bar:** A search bar is in the left sidebar of the CN. This option provides an option for the users to search for the content of the posts that include the specific keywords written in the search bar if the keyword entered in the search bar exist among all the posts. If the keyword which is searched found, then it would be highlighted and allow the users to see all the posts with this searched keyword shown on the top for the users. This explained task bar is shown and its feature is illustrated in the Figure 3.7.

- **Anar progress bar:** Anar seeds as explained previously is the representation of users activity in the CN. Anar progress bar is in the right-side bar of the CN posts and represent the user activity in a course. Clicking on the notification

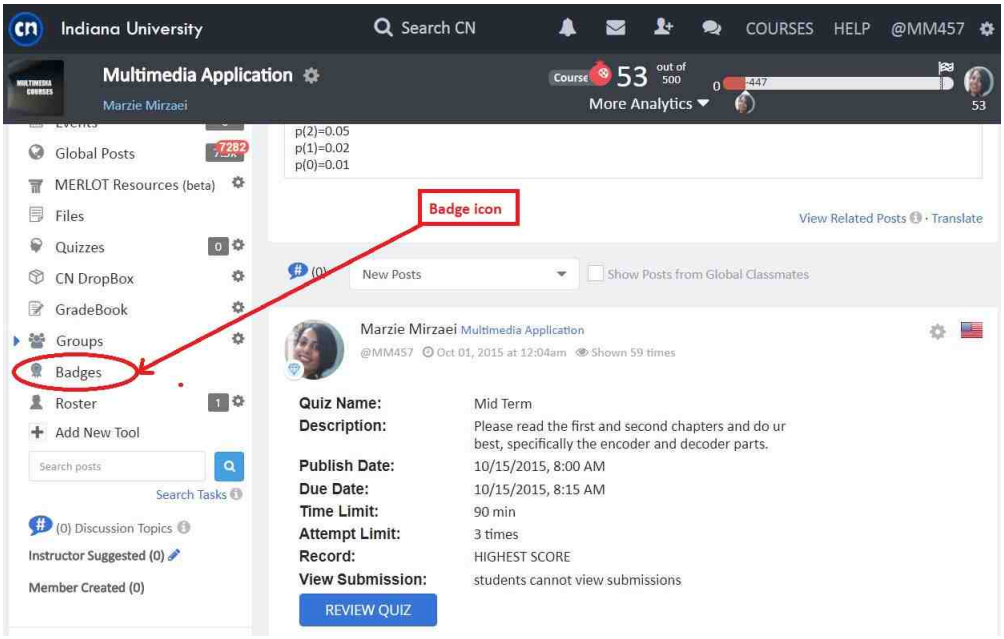

Fig. 3.5. Badges Icon Which is in the Left Side Bar in the CN

which is beside the Anar progress bar will show all the details about the course defined by the course instructor. Moreover, clicking on the pomegranate or Anar symbol beside the side bar illustrate a table of all the activities and the number of Anar seeds that the user would earn if they meet the minimum requirements adjusted by their instructor to receive Anar seeds or points. The Anar seeds role in the CN is very important since it represents the user's activity in the course. The users can earn a limited number of seeds per day. During the semester, the course instructor is able to change the number of bonus assigned to the users and shows in the Anar task bar. Figure 3.8 and Figure 3.9 clearly illustrate the explained feature of the CN.

These are the most important features defined in this social and learning network as a next generation of learning management system tries to help the users around the world interact with each other virtually and conveniently. In the following sections we explained our novel approach which help the users to see the best match posts through a dynamic and hybrid algorithm.

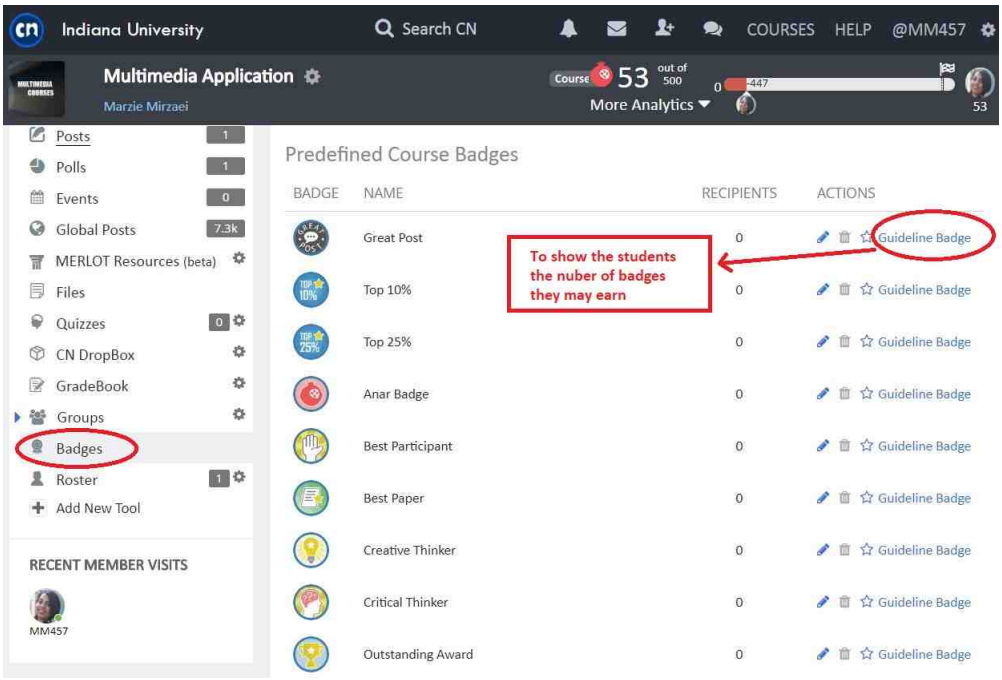

Fig. 3.6. Some of the Predefined Badges and their Description by Clicking on the Badges Icon

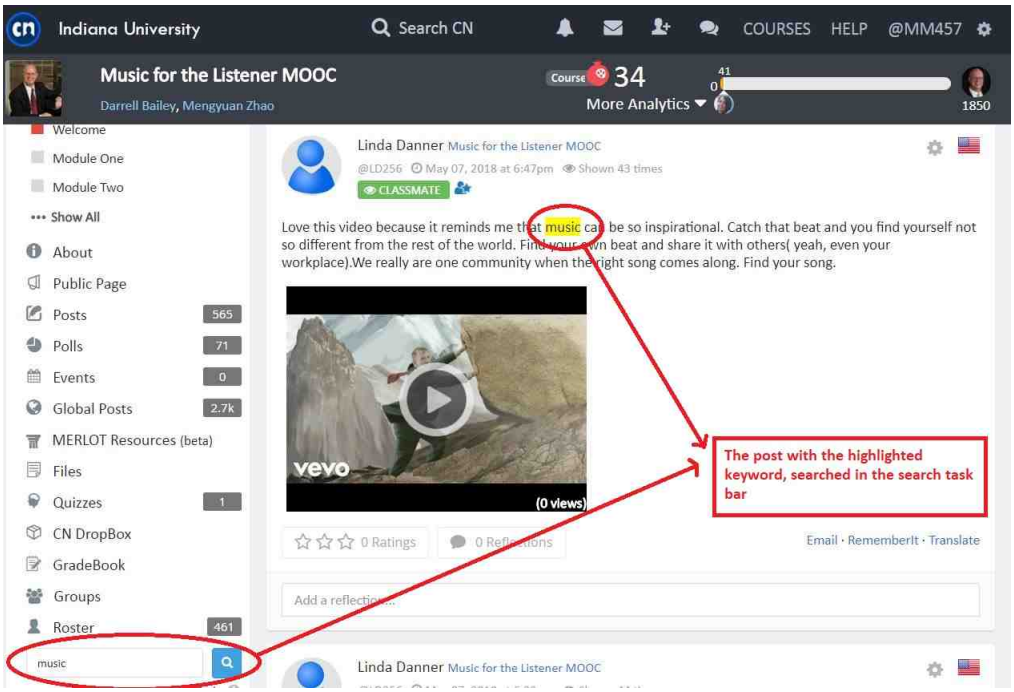

Fig. 3.7. One of the Posts with the Highlighted Music Keyword Searched in the Taskbar

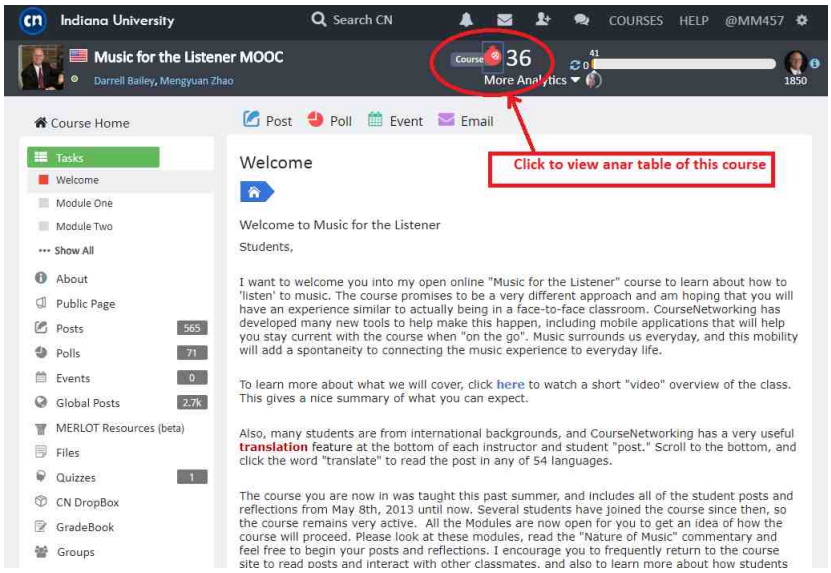

Fig. 3.8. Anar Icon to View Anar Table of a Course

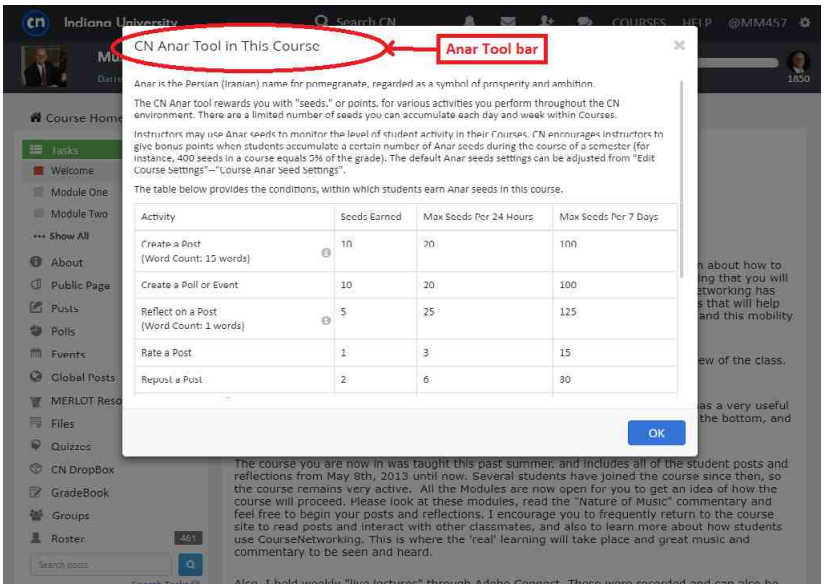

Fig. 3.9. Anar Tool Bar Provided by the Course Instructor

4. OUR METHODOLOGY

4.1 Introduction

The main purpose of this research is to develop a reliable and effective recommendation system for which each individual is provided the most relevant posts on the top through an enormous amount of educational content. These sets of recommended contents are related to the topic belongs to the specific course category that connected a user to his/her global classmates through user's course.

In developing a course-networking agent as a learning management system, a fundamental principle is to present global posts to the users considering the topics that the users have the highest desire to see. To achieve this goal, we developed a dynamic recommendation system for personalizing educational content. The necessity of building this autonomous recommendation system is to serve the CN users sets-of-posts immediately that best match their preferences. The main purpose of this research is representing a reliable and effective recommendation system by which providing each individual the most relevant posts on the top through an enormous amount of information. These sets of recommended contents are related to the topic belongs to the specific course category that connected a user to his/her global classmates through user's course.

To achieve the goal of developing our system design, there are some steps that we do not need to repeat in the whole prfSocess of our system design. In the first section of our thesis, we discussed about our system design requirements. Then, in the second section, we discussed about the design decision list of our recommendation system. The third section of developing our recommendation system is about the development

of the Ground Truth Matrix (GTM). Section four is described the matrix factorization method with the aim of building the whole recommendation system to recommend posts to the individual. The outline of our system design is illustrated in Figure 4.1.

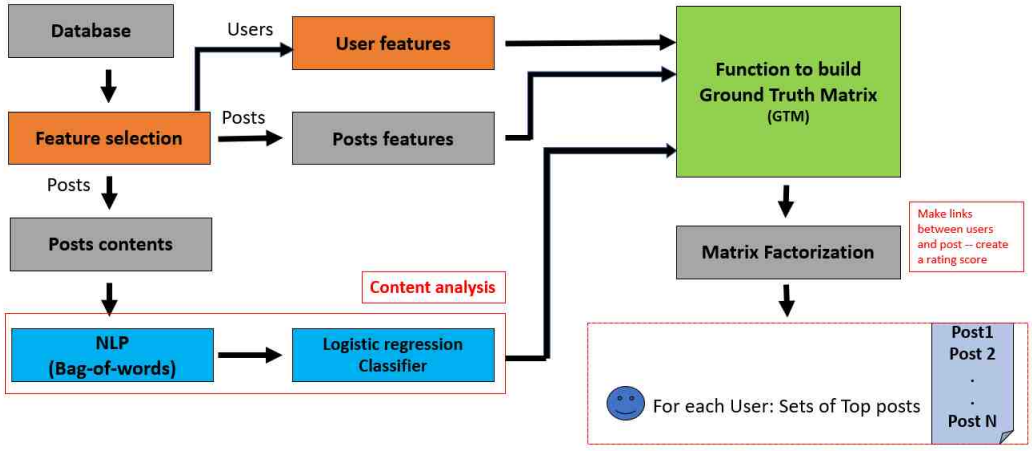

Fig. 4.1. Outline of our Proposed Method

4.2 Design Requirement

There are some important factors that we need to consider in our recommendation system:

- System accuracy.
  - We need an accurate recommendation system despite of limited feature objects in our network to show the users the most important content on the top.
- Accuracy for new users.
  - The system should be efficient and accurate for new users. The new users will not use the system unless they see valuable content in our e-network, so we need to have an efficient system for new users.

- Efficient up-to-date results.
  - Our designed autonomous system should provide real-time post to be impactful.
- Update itself.
  - The system should be able to update itself automatically based on the user activity in the system.
- Result existing tools/solutions whenever they exist.
  - In our design decision we examined at the existing tools to see if they can be adapted for our purpose. So, we applied natural language processing (NLP) and machine learning (ML) techniques in our system design.

4.3 Design Decision

The main purpose of this research concerns presenting a reliable and effective system by which providing everyone the most relevant, educational and up-to-date information on the top through an enormous amount of information.

In the second generation of world-wide-web (Web 2.0), recommender systems (RS) provide content preferences to the users based on the user preferences, in comparison with Web 1.0, where users were limited to view the specific fixed items without having to search for their desired content. In order to provide relevant content to the users via (RS) models, user-item interaction, object behaviors and/or information which are extracted from the other social networks are leveraged in the system in order to infer relevant content.

Recommender systems are developed for certain goals, and feature limitation is one of the main setbacks in most recommendation systems in new social or learning networks in order to achieve their goals. As a consequence, developing a robust recommendation system is crucially important for these networks to address this

issue. To deal with this issue, we applied unique pre-processing techniques to select some specific features that are able to make a link between user and posts and also extract more features in the system. In the following section, the feature extraction technique is discussed.

4.3.1 Feature Selection

To obtain the informative features from the CN network first we analyzed our CN dataset. The data is stored in the JavaScript Notation or JSON format. We extracted the user features and posts features from the system in order to make an explicit link between users and posts. Some of the users and posts features which exist in the CN network beside some other features that obtained in the system using some techniques applied in our proposed model to make a link between users and posts. All these features explained in the next paragraphs. Some of the defined features are used in the next steps of our dynamic recommendation system. First, we extract unstructured data stored in JSON format and then among all the features exist in the system, the relevant features fed to the central recommender system.

Table 4.1. User Features

User Features
Gender of the user (Female/Male)
User's Anar seeds
Role of the user (Instructor/students)
User's rating star
User's reflection
Creation time of a post

User features

- User share the posts.
  - Whether a user shared a post that was already created in the system or not.
- Gender of the user.
  - The gender of a user is an optional item in the CN. We analyzed the gender of a user and his/her interaction with the posts in the CN. We determined whether user is a female, male, or not defined by the user in the system. These features are optional, so for some of the users are accessible and for some other are not.
- User action to a post.
  - How a user interacted with a post. Users can be a creator of a global post or can rate a post or reflect on a post.
- Role of the user in the system.
  - The role of the user in the CN could be a student, instructor, or professional. In the CN, users can have two different roles; instructor or student or both in different courses. It is important for the users to see the posts that are created by their instructor or student who shared the same topic that they are interested in following in the specific period of time. Sometimes posts that are created by the instructor are more important than the posts that are created by a user as a student. So, the users do care more about the contents of their instructors.
- Grade of a user in a specific course.
- Anar-seed of the user in the course.

  - Represents points given to the users based on their activities in the CN.

- Rating star of a user for a post.

  - If a user rated a post or not and if he/she rated a post how much was his/her rating star to the post.

- User re-post the post or not.

- Rating stars of the users.

  - It is important to consider the rating stars of the users to a post as an explicit way to measure users's academic feedback to a post.

- Reflection on a post.

  - Reflecting a post is an option for the user which is giving them a chance to put comment under a post. So, extracting all the information of the users who reflected on a post and analyzing their comments on the posts as user's behavior in the system would directly impact future posts that would be provided to them.

The Table 4.1 represent the user features, and the bold features in this table represents the user features that were used in our recommender system.

Post features

To make a link between per user and post, it is important to access the post features. There are some features related to the posts, and among all of them, we extracted those relevant features that enable us to make a link between user and post. In the following, some of the posts' features are described.

- Creation time of a post.

  - Time that a post was created.

- Shown-times.
  - Number of times that posts are shown in the CN.
- Number of times that posts are shared.
  - Number of times that a post reposted which means that specific post which shared was important for a user.
- Content of a post.
  - We analyzed the content of the posts as a main feature in our proposed system in order to divide them to two separate categories: educational content (positive), non-educational content or totally irrelevant posts (negative).
- Rating star of each person to the posts.
- Average rating number of post.
  - The CN is using a function to assess the average rating of a post using the rating star of a user on a post.
- Number of reflection.
  - Number of comments that the users put on a post.
- Post Of the Week (POW).
  - Every Saturday, CN selects the most popular post from the previous week. We call this selection Post of the Week (POW). POWs are selected based on the total rating a post received. For example, if a Post received 5 one-star ratings, 3 two-star ratings, and 6 three-start ratings, it is POW score is $1X_5 + 2X_3 + 3X_6 = 29$. If more than one post received the same number of points, the post contains more words will be the winner. If the number of words is the same, the post created latest will be the winner.

Table 4.2. Post Features

Post Features
Creation time of a post
Content of a post
Number of comments (reflection) on a post
Average rating star of a post
Shown-time (number of time post visited or shown in the CN
Post Of the Week(POW)
Hashtags
Attachments

- Hashtags.
  - Hashtag can be defined by the instructor for that course known as pre-defined hashtags or the users can create them. Check whether the posts have hashtags that defined in the network.
- Attachments.
  - Posts can have attachments such as video, image and links. Some attachments like images, videos or links are ignored in our analysis because they would not transfer any meaningful textual information, and using the information of the videos, pictures, or any other attachments to the posts is outside the scope of our. This is a future work research focus that requires some other methods for image and video processing to extract useful information from them.

Table 4.2 represent the post features, and the bold features in this table represents the post features that fed to the central recommender system. The first step for our recommender systems is to extract the useful item-features and user-features from

the CN database. All the user and item information are stored in JSON and excel format. In our recommender system, items are global posts that are shared in the business course category. Recommender systems face some challenging issues such as cold-start problem (new users) and data sparsity problem which make it very difficult to do missing entry analysis since we do not have enough available and social information. So, to address these issues, we applied natural language processing plus machine learning techniques for textual analysis in the interest of feature extraction. Among all the features explained, there are limited number of features that are able to make a link between users and posts. These limited features are selected to improve the accuracy of our proposed dynamic recommendation system. In order to address the feature limitation with the aim of increasing the accuracy of the system, we applied classification technique as a feature identification technique to be used in our proposed prediction model.

Regarding the existing features in the CN, there are some optional features like gender of a user that is specifically explained users features but not making a link between users and posts to be used in our dynamic algorithm to build a hybrid link between users and posts. Indeed, some of the pre-defined features like the gender of the users are optional and for some users we access to these features and for some others we do not.

These selected features enable us to give a rating value representing a link between users and posts using a hybrid function. Classification as the next step of our design decision is explained in Section 4.3.2.

4.3.2 Classification as a Feature Selection

There are many courses that users hold the instructor role or student role which are already registered for. And, each of those courses belongs to the specific topic e.g computer engineering, business, art and so on. In our case, all the data that we are studying belong to the business course category. The main purpose of this research

is representing a reliable and effective system by which providing each individual the most relevant information on the top through an enormous amount of information. This information is related to the topic belongs to the specific course category that user selected. So, for the sake of recommending relevant educational contents, we filtered out the meaningless posts shared in the business course category among all the existing posts and we kept those features that belong to the target topic; business.

The next step in our system design is to analyze the content of the post which is very important to be considered in our e-learning network CN. So, in order to analyze the content of the posts, the first step is to vectorize the content of the posts and for this aim, we are using bag-of-word model as a Natural Language Processing (NLP) technique.

Natural language processing / Bag-Of-Words

The advantage of using Natural language processing (NLP) method is for text analysis to map the content of the global posts to the numerical vectors to be fed to the classifiers and to improve the performance of our recommendation model. Bag of words is a model in the NLP. The principle of using bag of words model is for text vectorization, in order to convert the posts into the vectors to be readable by the machines. So, we used Bag-Of-Words model to map the content of the posts to the vectors of the binary numbers and then the output of the NLP model fed to the classifier models. This model is applied for feature generation. The output of using this model provides the term frequency of the words in the documents as our global posts in the business course category, we apply these frequencies to feed into the classifiers, as they are considered as principle implicit post feature and then by using our hybrid recommendation system function to make best recommendation.

In this model, the content of the global posts in business course category represents as a bag of all the words regardless the position of each word in the posts and the grammar. This model is used for vectorising the posts. It is utilized in the posts

that a user shared with his/her global classmates disregarding the stop words like "in", "the", "at" that are commonly used in the users posts which should not affect text classification. In the CN, most of the users utilize the same hashtags in their posts. So, considering the specific hashtags would not help us in recommending unique news feeds to the users and understanding the content of the posts, but using Bag-of-word model helps us address this issue. In the matrix, the rows corresponding to the documents which are our global posts, and the columns corresponding to our dictionary of words. We built our dictionary after filtering out some words and characters. The process of filtering some specific words and characters from our global posts is explained in the following steps.

- Removing the stop words like "the", "at", "between", "do", "doing", "by" and so on.
- Convert @username to username.
- Remove additional white spaces; for some of the posts the inserted additional space in between content of their global posts, so we removed these additional white spaces before analyzing type of posts.
- Replace #word with word. Generally, hashtag is used to place the content of the posts in a category according to the topic being shared. However, in the CN network, after analyzing the global posts in business course category, it looks that most of the users used the same hashtag in their posts regardless of the post is related to business topic or not. So, there is no difference between the words with hashtags and the words without hashtag.
- Remove special character like $ and % in content of the posts.
- Remove repetitions of characters.
- Remove the posts written in the other languages but English.

The number of textual data, which used only the global posts extracted from business course category, was 6395 and among these posts, some of them are in foreign languages like Chinese. In this model, all the global posts that are generated by the users in another language, but English are filtered out. Also, some of the posts did not shared any textual information. Indeed, they are included just some attachments like images, videos or links. These kinds of posts are also ignored in our analysis because they would not transfer any meaningful textual information, and using the information of the videos, pictures, or any other attachments to the posts is another research and requires some other methods for image and video processing to extract meaningful information from them.

After filtering the global posts and extracting the posts that explained in the above paragraph, around $3k$ of the posts fed to the Bag-Of-Words model. The output of using this model is a matrix with $3k$ rows and $16k$ columns. Rows in this matrix correspond to all the global posts shared in business course category as our documents, and columns correspond to the words that used in the shared posts after filtering out the special characters in the documents. The output of using this model would construct the term frequency of words in the documents which used to feed to the classifiers to be considered as principle implicit post feature that was fed to the central algorithm. Keywords are all the words that are defined already and all the stop words like “the”, “in”, “at” are excluded.

The entries of a matrix are binary. We fed the 80% of the document in our binary matrix to the classifiers to train the model and set the value of the hyper parameters to classify the samples as our posts or documents. The input of the classifiers are the global posts that belongs to the business course category and the output of the classifier is represent in the form of binary or probability number based on the classifier which is used. So, the dimension of the input is equal to the 80% of the global posts. In the next step, a different type of classifier which used for feature extraction after

vectorizing the content of the posts using NLP is explained. In Figure 4.2, the process of vectorising a global post is shown. The whole global posts fed to the bag of words model to be vectorized to numeric vectors which is shown in Figure 4.3.

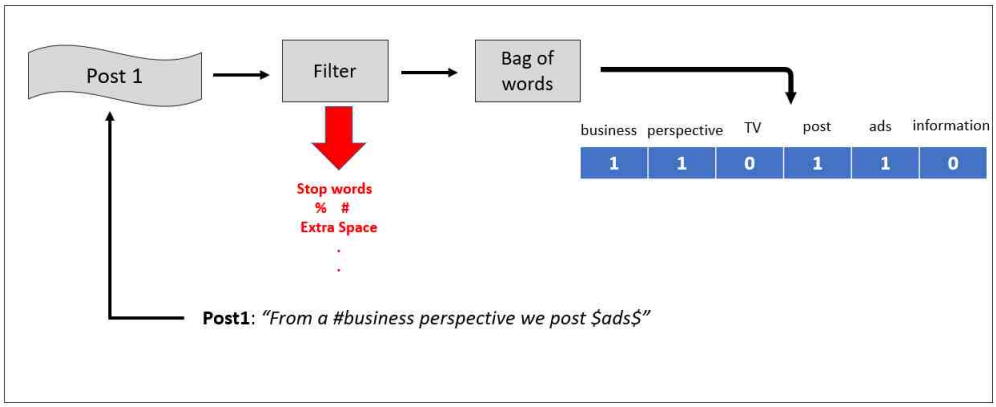

Fig. 4.2. Bag-Of-Words Model on a Post Sample

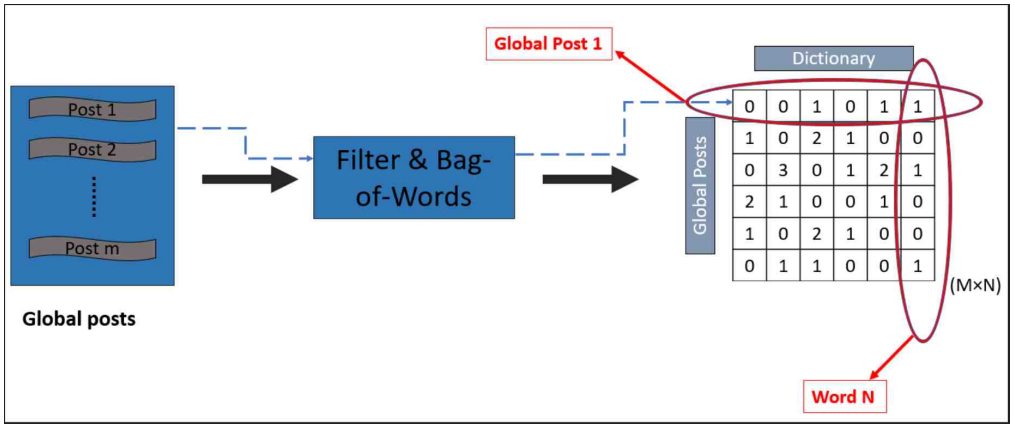

Fig. 4.3. Bag-Of-Words Model on all the Global Posts

4.3.3 Content Analysis

Supervised and Unsupervised Machine learning

In the data analysis field, machine learning allows us to make prediction based on our data sets. In machine learning, there are two types of systems:

- Unsupervised.
  - Unsupervised learning is used for the data sets that are not already labeled and there is no labeled output for the sets of data that we have. In unsupervised models, based on some features of the data, one is going to map the data in different clusters. Then based on the predefined clusters we would map the new samples and then update the model.
- Supervised.
  - In supervised learning, one is try to train the model with some inputs and outputs that already labeled (training set) and then we would test and predict the output of the model for new samples (test set). The principle is to build the system for increasing the accuracy of the models for new samples feeding to the model.
  - In the supervised machine learning, the specific algorithm would use the mapping function to map the input variable to the output variable. In the following paragraphs some of the supervised learning algorithms are explained. We applied these classifiers to check the accuracy of different models.

In order to choose the best classifier, first we split the output of our Bag-Of-Words model to two parts, train and test. We used the 80% to train our model and via feeding the test sets which is 20% of the output of the bag of word model to test the accuracy of the system. The output of the NLP is the binary vectors representative of the

global posts related to the business course category. So, in the following whenever we used the global posts, we mean the posts related to the business course category that we studied and built our model on the basis of the information extracted from NoSQL data based and belongs to the business topic.

First, we filter out the posts written in the other languages except English and all the symbols used in the shared global posts. Then, after removing the non-English global posts, we read around 70% of the global posts which is around 3000 posts and labeled them either 0 or 1 to build our training set. 0 represents the posts which are not related to the business topic and 1 means the posts are related to the business topic. After labeling the posts we fed them to the different type of classifiers to train the model in order to tune the hyperparameters of the model and to predict automatically the future posts coming to the system to check whether they are related to the specific topic or not. The Figure4.4 illustrates the splitting process.

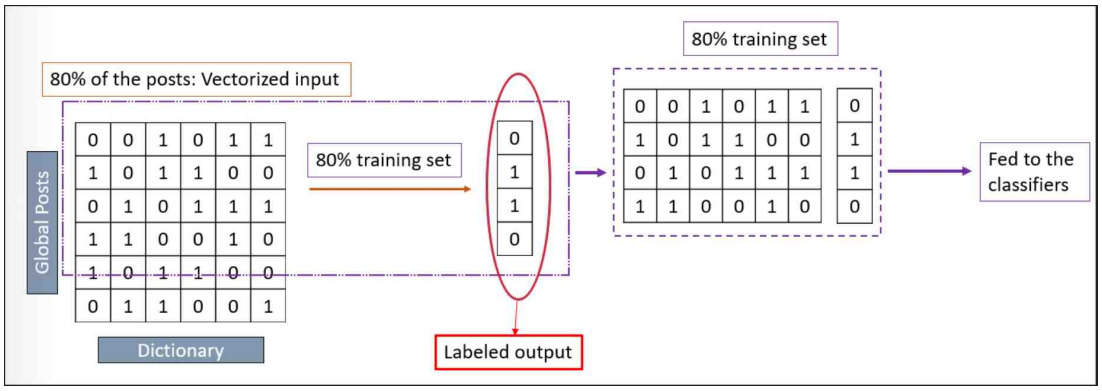

Fig. 4.4. Using 80% of the Output of Bag-Of-Words to Train the Classifier

Different supervised models have been developed for the posts classification purposes. Some of the supervised machine learning models task are like classification, recommendation, and feature extraction and so on. In this study, we developed the supervised machine learning classifier for feature extraction so as to be used as an implicit feature in our recommendation system to overcome the limit object features and also developing a reliable and effective system by which providing each individual

the most relevant information through an enormous amount of information. In the following sections different type of supervised machine learning classifiers and their output are discussed and we applied them to choose the best classifier to be used in our recommendation system.

Before explaining about different type of classifier, in our analysis, we first tried to classify the content of the posts into three different classes: positive and negative and neutral. Here positive means the content of the post is related to the business course category, negative means that content of the post is not related to the business course category and neutral means that it is meaningless or related to the specific topic in the other course categories.

To classify the content of the posts, we trained and tested the accuracy of different classifiers, the process is explained in the next sections. The output of a specific classifier with the most accuracy, is considered as a feature of the content of a post. The output of different classifiers evaluate that with which probability a post is related to the positive or neutral or negative category. Content of reflections of the posts can also be analyzed as positive, negative, or neutral; positive means this reflection has educational content which is related to the business course category, negative means this reflection has educational content but related to different course category(class negative), and neutral means that it does not have any meaningful content (class negative). However, after training different classifiers with the existing textual data, the classifiers had a very low accuracy to classify the posts. Therefore, we developed a supervised machine learning classifier to classify the content of the posts to two different course categories positive and negative. The positive class includes all the posts belonging to the business topic and negative class contains all the posts belong to the other topic except business and/or meaningless posts such as the posts that the users just introduced themselves to the others or the posts which shared just some video or image without any contents. As a result, after trying different classifiers and measure the accuracy of these classifiers, we chose a classifier with the most accuracy.

Logistic regression

To start the supervised machine learning classifiers, we would start from logistic regression. Logistic regression is a non-linear and binary classification model to classify the dataset to two positive and negative classes. Logistic regression is a non-linear model. In logistic regression, the output is shown with a probability to predict with which probability the input belongs to positive class or negative class.

Mathematical definition of logistic regression is:

In logistic regression, there is an activation function $\hat{h}_\theta(x)$ or $\hat{y}$ to predict the output for unseen samples that is given in Equation (4.1).

$$\hat{y} = \hat{h}_\theta(x) = \sigma(\omega^T \cdot x + b) \tag{4.1}$$

where,

$$\sigma(z) = 1/(1 + e^{(-z)}) \tag{4.2}$$

and,

$$z = \omega^T x. \tag{4.3}$$

In Equation (4.1), the ω is a set of all the coefficient parameters or feature parameters θ, x is a set of all the samples used to train the model and then test the model for new unseen sample x and predict $\hat{y}$. The output depends on the linear combination of ω and the bias b. Our goal is to minimize the cost function by tuning Ω or the parameters θ and the bias b to reduce the difference between actual output y and the predicted output $\hat{y}$. The cost function in Equation (4.4) of logistic regression is displayed as follows.

To learn hyperparameters for our model, we applied the $m - sized$ training set of $(x^{(i)}, y^{(i)})$ to train our model. We want to determine w and b to have a minimum error or minimum cost on our test set. The cost function that applied for logistic regression is:

$$J(\theta) = \min_{\theta} \Bigg\{ (1/m) * (\sum_{i=1}^{m} (y^{(i)} \cdot (-\log(h_\theta(x^{(i)})))) + \\ (1 - y^{(i)}) * (1 - \log(1 - h_\theta(x^{(i)})))) + (\lambda/(2 * m)) * \sum_{j=1}^{m} (\theta_j^2) \Bigg\} \quad (4.4)$$

In the Equation (4.4), m is the number of samples used in our training set and λ is the generalization parameter. To minimize the cost function, a gradient descent method used to update ω and b. Gradient descent is an algorithm applied to optimize the ω and b which is used to minimize the cost function. The Gradient descent is illustrated in Equation (4.5). The first iteration in Gradient descent will start with random coefficients then we evaluate the cost function with the used coefficients and again updating the coefficients ω and b using the gradient descent algorithm, then evaluate cost function; The process of updating the coefficients will continue till convergence. Then, we used the optimized ω and b to estimate $\hat{y}$ with the minimum error for the training set. The goal is to minimize cost function with the best updated coefficients which means the best prediction. The regularization term λ, in the cost function, is used to make an accurate classifier for unseen datasets to address overfitting.

$$\theta_j := \theta_j - \alpha \cdot \frac{\partial J(\theta)}{\partial(\theta_j)} \quad (4.5)$$

The result of using the logistic regression for classification as a feature extraction purpose is shown in the Figure 4.5. In Figure 4.5, the samples in our training set split to two parts training and cross validation set. The size of the training set was gradually increased to verify the accuracy of the model with the training sets to tune the hyperparameters. As we see in the Figure 4.5 by increasing the data set, the accuracy of the cross validation and training sets, increased and converted to each other, which shows that the logistic regression model has a very good performance.

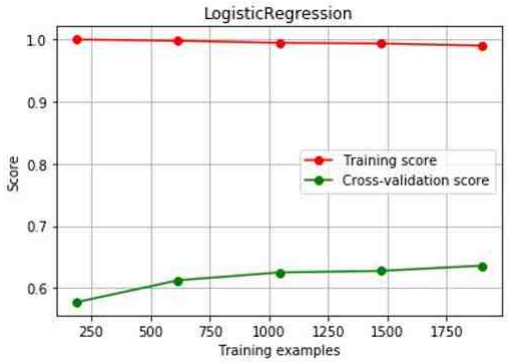

Fig. 4.5. Logistic Regression

Support Vector Machine(SVM)

SVM models are generally used for data classification or regression analysis. SVM is known as a large margin classifier. In SVM, the decision boundary for separating the positive and negative samples is selected in such a way to make the largest distance between the classes. So, it tries to separate the data with large margin, which is a consequence of the optimization problem. Regarding the SVM classifiers, linear and non-linear SVM both have a good performance because it is ignoring a few outliers. The output of SVM make a prediction $y = 1$ or $y = 0$ directly.

The cost function used in SVM is similar to the logistic regression cost function with a very slight different and is shown as follows.

$$J(\theta) = \min_{\theta} C \sum_{i=1}^{m} y^{(i)}.Cost_1(\theta^T.x^{(i)}) + (1 - y^{(i)}Cost_0(\theta^T).x^{(i)} + 1/2 \sum_{j=1}^{m} \theta_j^2 \qquad (4.6)$$

where,

$$Cost_1(\theta^T.x^{(i)}) = -\log_{10} h_\theta(x^{(i)}) \qquad (4.7)$$

and,

$$Cost_0(\theta^T.x^{(i)}) = -\log_{10}(1 - h_\theta(x^{(i)})) \qquad (4.8)$$

In Equation (4.6), $C = 1/\lambda$. The hypothesis function in SVM is illustrated in Equation (4.9). The hypothesis function for support vector machine is:

$$h_\theta(x) = \begin{cases} 1 & \text{if } \theta^T \cdot x \geq 1 \\ 0 & \text{otherwise} \end{cases} \tag{4.9}$$

SVM Decision boundary

In supervised learning, we try to train the model with some known inputs and outputs that already labeled and then we would test and predict the output of the model for new samples. The principle is increasing the accuracy of the models for new and unseen samples. Regarding the linear separable boundary, there exist a straight line that separates the positive and negative examples and the SVM choose a decision boundary which is more robust to separate the positive and negative examples. Mathematically, this means the SVM decision boundary has a larger minimum distance from the training samples and this distance is called margin of SVM.

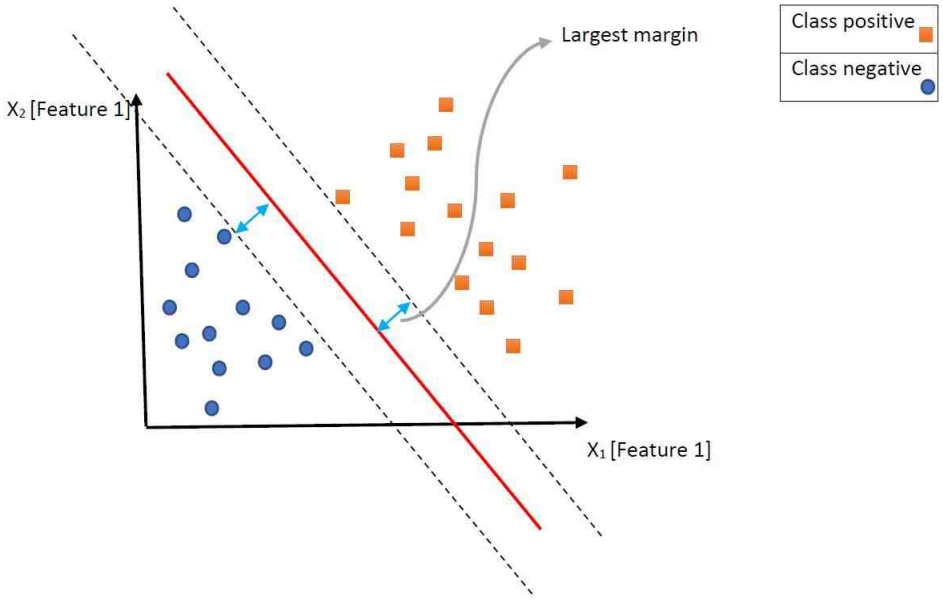

Fig. 4.6. SVM (Large Margin) Classifier

The red decision boundary or hyperplane in the Figure 4.6 has the largest distance from our classes which contain both orange and blue points. This boundary might also ignore some outliers which is because of the regularization term used in the cost function. The regularization term is trying to optimize the model for the unseen data with the aim of reducing the overfitting.

The other thing about SVM classifier besides having the largest margin is that it still might be sensitive to outlayers. It is not a good idea to change the decision boundary just because of some outliers, so if a regularization parameter C where $C = 1/\lambda$ is not too large then we end with a good decision boundary regardless of outliers.

In the case that the data is not linearly separable, we need a non-linear classifier. The reason to have a non-linear classifier is to construct kernels that make non-linear decision boundaries. The main purpose of using classifier is to build a hyperplane to distinguish between a positive and negative classes in our model. Regarding non-linear classifiers, we come up with a set of complex features that looks like the Equation (4.10). In Equation (4.10), f_i is a combination of features like $f_1 = x_1$, $f_2 = x_2$, $f_3 = x_1 \cdot x_2$ and so on. So, in Equation (4.9), our hypothesis is going to predict 1 when $\theta_i \cdot f_i \geq 1$. We define the extra features to learn more complex nonlinear classifier in order to build our hyperplane.

The non-linear decision boundaries with different kernels comes up with set of complex polynomial features θ in such a way that the $h_\theta(x)$ will predict 1 if all the polynomial features are greater than 0 and otherwise predict 0. It computing the decision boundaries using Equation (4.10).

$$\theta^T \cdot f = \theta_0 + \theta_1.f_1 + \theta_2.f_2 + \theta_3.f_3 + ... + \theta_m.f_m \tag{4.10}$$

The higher order polynomial comes up with more features. Indeed, the goal of using kernels is to find more features for SVM via using f in function 4.10 which is a combination of different features x_i. In non-linear SVM the cost function illustrated in Equation (4.6)

Cross-validation.

The cross validation is a validation technique used to assess the accuracy of the estimated prediction model for the system using the known training dataset. In our case study, the goal of classifier is to classify the global posts in our system to two positive and negative classes in such a way having the efficient classifier with the most accuracy. In this model, we are partitioning the result of our statistical analysis which is the vectors of numerical numbers (training data set). In this method, all the training set is partitioned into n subsets and for this reason this method is called n-fold cross validation. In the n-fold cross validation, all the training data set will be divided to n parts and each time we use one part for testing and all the other partitions except the part used for testing set are used to train our model. As a consequence, the model would train and run by considering all the elements into the Matrix factorization, so the predicting model will generalize to different dataset.

Indeed, the cross validation is a method to assess the output of statistical models used for predicting new posts to the users. The numbers used in cross validation, in our case, are the numerical numbers defined in the Matrix factorization. In this method we are partitioning the training set to k fold to reduce the over fitting problem. Overfitting is a property where with the training samples the model perform well yet when we apply new data, we achieve poor results. So, in order to have a good performance for our prediction model for new samples we are selecting the proper hyperparameters and considering the regularization parameter λ.

The cross-validation is a validation technique to estimate an accurate prediction generalized model for the system by rotating on different subsets as a training sets to reduce the over fitting problem.

K-fold Cross-validation

For training the classifier models the training samples would be split into two parts training sets and validation sets. However, splitting the training data into two parts will reduce the number of training samples to tune the hyperparameters and the result or accuracy of the model will be affected by the specific selected part of training sets. So, to address the issues mentioned via using just specific parts of the training sets to build the model, the cross validation technique is applied. In the k-fold cross validation, $k-1$ subsamples used for training and a single subsample used for testing the model. This process would be repeated over k subsamples. So, all the sub samples would be used in training process. In this method usually using 10-fold cross validation is a commonly used method. After experimenting the cross-validation method and calculating the accuracy of each fold, we would make an average on the accuracy of all the folds.

Currently, the posts that are shared in the user's profile page are displayed based on a static function which is not transferring the meaningful information to the users, so we are trying to filter the redundant posts to not being shown to the users on the top as our priority is to show the posts on the top that are transferring meaningful information to the users and related to the topic of the course category and also satisfying the user's interaction in the CN.

As a consequence, we require to use a classifier that could extract the relevant information belonging to the specific course category. Different types of machine learning classifiers are tried and compared. The goal of using classifiers, already mentioned, is for feature extraction to help us addressing the limited features that exist in this e-learning network.

Different classification models, as already mentioned, have been applied. The input of the classifiers are the vectors coming from the output of the bag of word model which gained from the previous step. Indeed, the numeric vectors are the word frequencies which are fed to the classifiers. For this aim, we used 80% of the global

posts to train the model, and the 20% of the data used for testing the accuracy of the classifier so as to determine the most appropriate classifiers. We implemented the classifiers in two different ways:

Firstly, we tried to classify the textual content in three different categories

1. **Negative**: all the posts that are totally irrelevant to the business course category.
2. **Positive**: all the posts that the users are shared and are relevant to the business course category.
3. **Neutral**: all the posts that are not about the business nor other topics belonging to the other course categories. Indeed, they are not transferring any meaningful and academic information to the other users.

Secondly, we tested different classifiers to classify the posts to just two categories

1. **Negative**: all the posts that are related to any topic in any course categories but business.
2. **Positive**: all the posts that are related to the business course category.

We trained our classifiers using 80% of the data in our system, on the basis that the data set has output is already known manually, and the 20% of the data set used for testing the classifiers.

In the first experiments, we tried to categories the data to three different categories; positive, negative and neutral posts. The positive posts, as we mentioned already, means those posts that contain relevant information which means their content are related to the business course category. And, the negative posts are those that are not related to the business course category but they are related to the other course categories, and the neutral posts are those that does not have any meaningful information like the posts that the users specifically the new users who just join the system and want to socialize with the others and make a link with the other users.

However, the small number of posts that we have right now in our system restricted us to train our system with great accuracy. Our classifiers output demonstrated that the accuracy of the system was not good enough and the system do not have high accuracy to categorize data to these three categories.

The experimental results demonstrate that the system is not working accurately and using these kind of classifiers would reduce the accuracy of the whole system. The principle of using different classifiers is to address the limited features in the system by increasing the number of the features of the system and using the meaningful textual information to feed that to the central system.

For increasing the accuracy of the machine learning classifiers require to access the large data. So, basically, the more the data we have the more accurate classifiers model we would have. As a result, gradually increasing the number of the users and the number of the posts in the network would increase the accuracy of the system.

As a consequence, to address the low accuracy and increase the accuracy of the system via textual analysis, we applied different classifiers for categorizing the posts into two categories, positive and negative posts. Positive means the posts that are related to the business course category and negative are the posts that are irrelevant to the business course category. The irrelevant posts defined as all the posts that are meaningless like just the posts that the users created to say "hi" to the others or introducing themselves or the posts that has some meaningful information but are not related to the business course category.

In order to evaluate the accuracy of the different classifiers, learning curves repeated on a series of training samples. Learning curve show us the accuracy of the learning system.

In the following Figures, the output of testing different classifiers for classifying posts to relevant and irrelevant posts are illustrated.

The Figures 4.7, 4.8, 4.9, 4.10, 4.11, 4.12, 4.13, 4.14, 4.15 illustrate the accuracy of the classifiers for different training samples. As we see in Figure 4.9 and 4.15 with increasing the sampling data, the cross-validation score and the training score

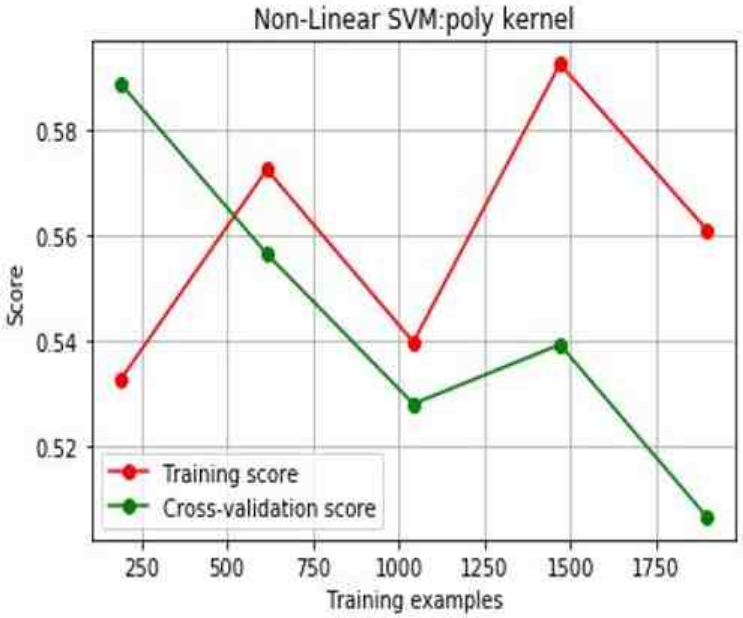

Fig. 4.7. Non-linear SVM with Polynomial Kernel

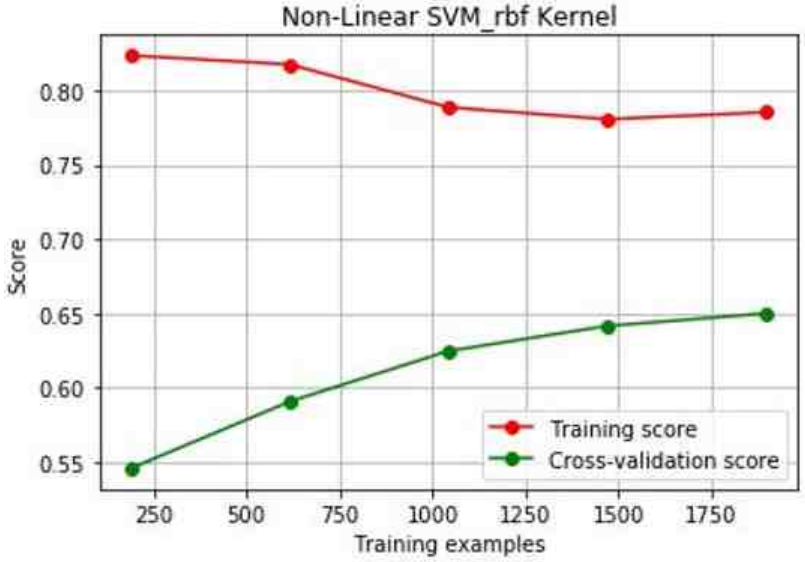

Fig. 4.8. Non-linear SVM with RBF Kernel

coverage percent accuracy, shows that by increasing in the number of training data and the samples the accuracy of the classifiers would increase. On the basis of the

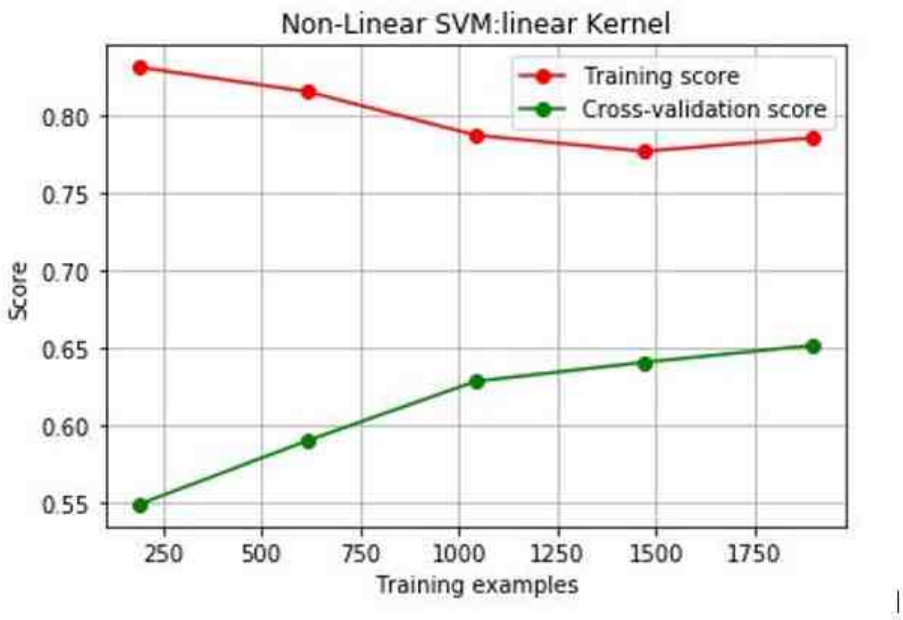

Fig. 4.9. Non-linear SVM with Linear Kernel

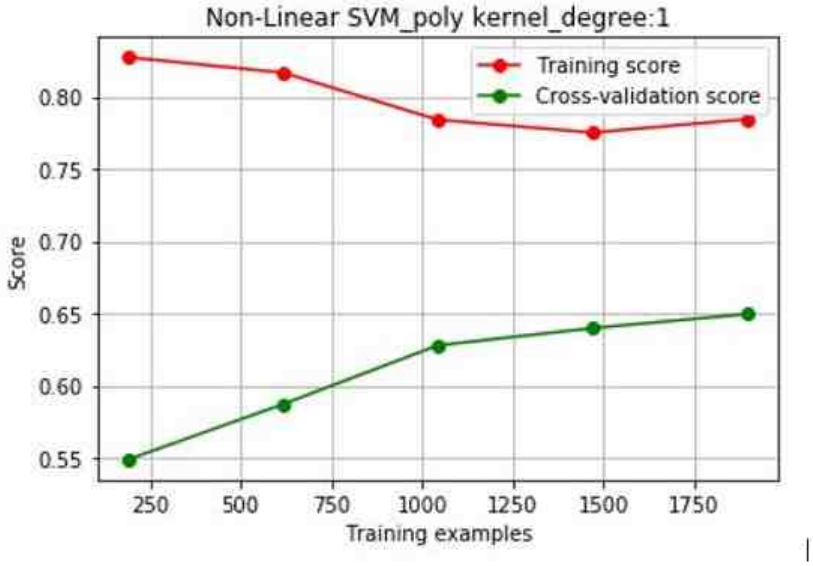

Fig. 4.10. Polynomial Kernel with Degree1

results, we used the logistic regression model with the better accuracy in comparison with the other classifiers and also because of its probability output which would be used in our dynamic recommendation system to make a better prediction.

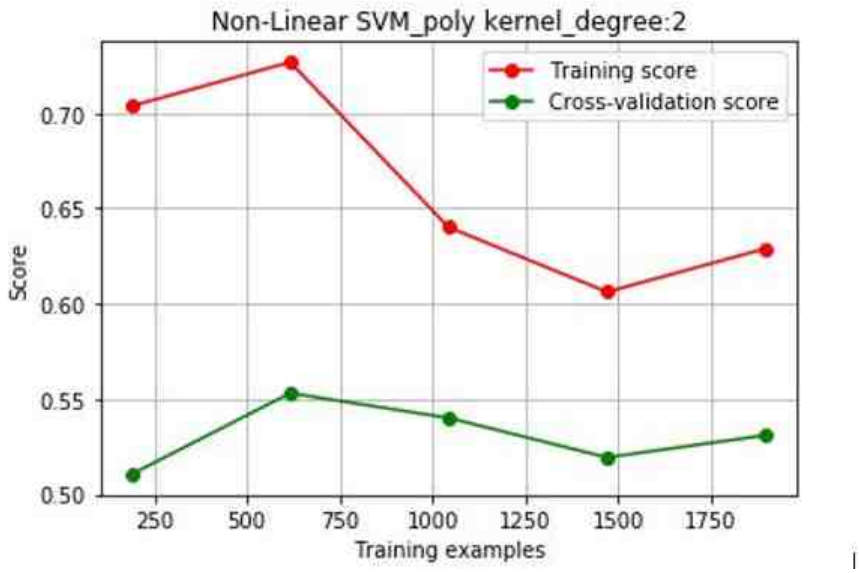

Fig. 4.11. Polynomial Kernel with Degree2

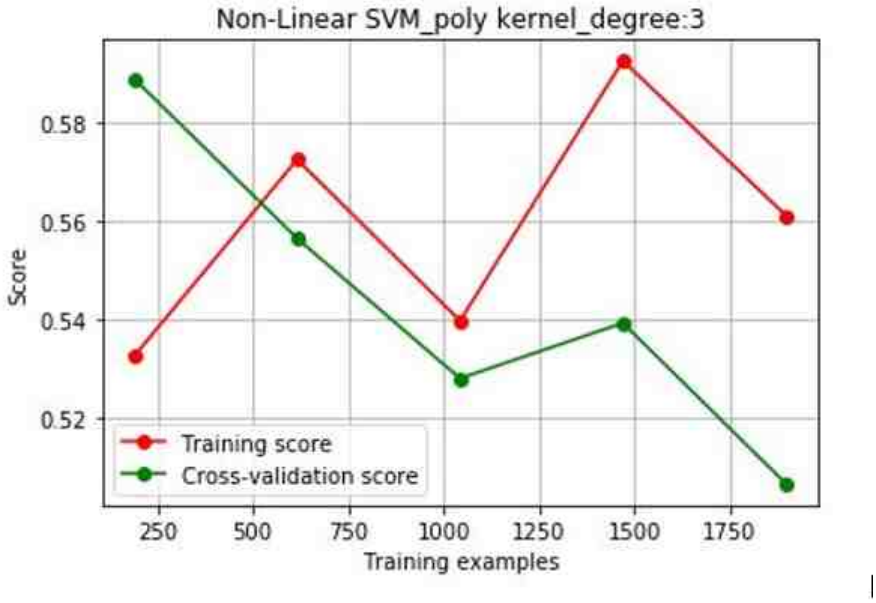

Fig. 4.12. Polynomial Kernel with Degree3

The accuracy of the classifiers for classifying content of the global posts in business course category is displayed in the Figure 4.16. As we see, the logistic regression and the support vector machine have the most accuracy in comparison with the

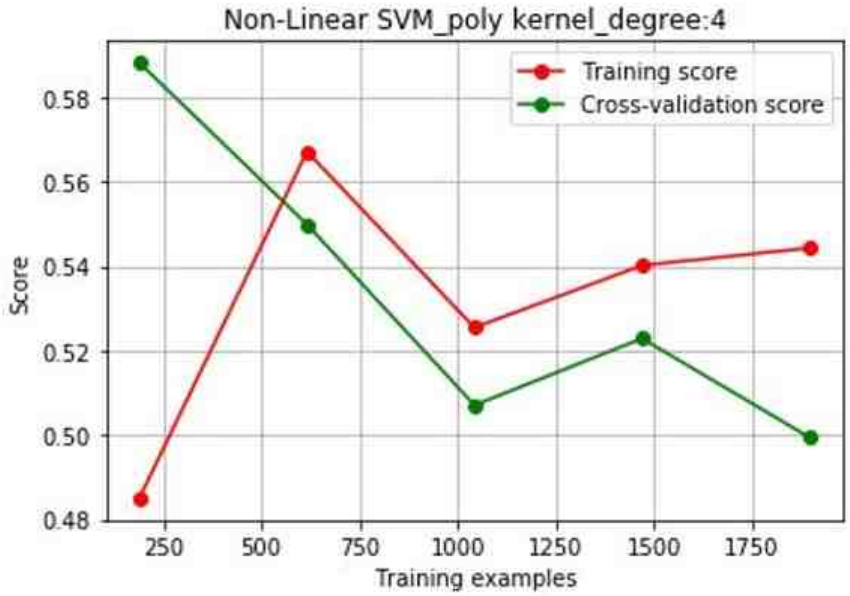

Fig. 4.13. Polynomial Kernel with Degree4

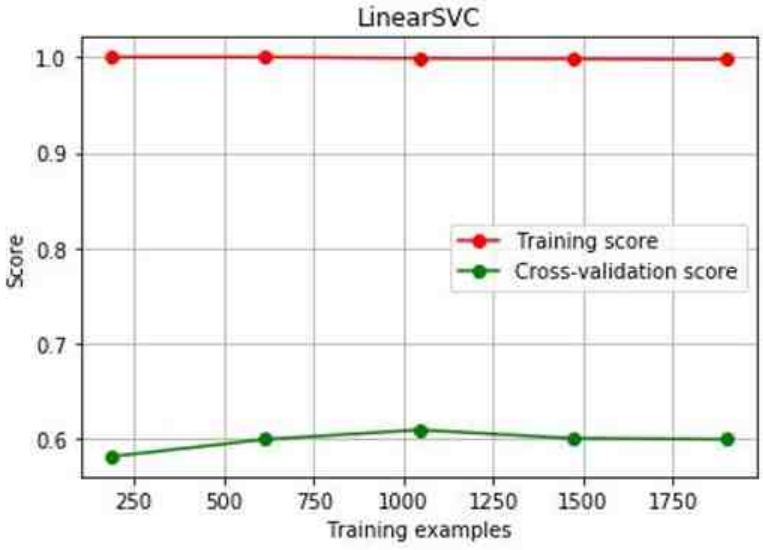

Fig. 4.14. Linear SVM

other models. The output of using logistic regression is probability, which is the probability that the post belongs to the class positive (observes that the complement of this probability is the probability that it belongs to the negative class). However,

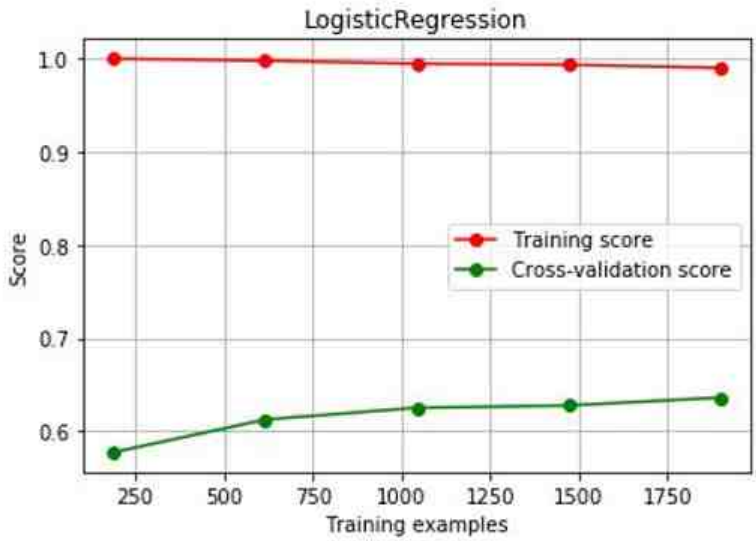

Fig. 4.15. Logistic Regression

the support vector machine has a binary output. Support vector machine as a large margin classifier is a very good model to classify the dataset, however its binary output has a very sharp impact to reduce the effect of other features applied in our hybrid recommendation system to make a link between users and posts. If there is some link between a user and post when considering some features, but the post is not completely related to the target topic, then the SVM will make a link between user and post very low or even 0. So, we used the logistic regression model and its output effect to make a link between users and posts.

Furthermore, the output of using SVM is shown in Figure 4.17. We see that 1434 of the posts predicted correctly in the class positive which means the class of the posts that are related to the business topic and 167 number of the posts predicted as a class negative. However, they belong to the business topic or class positive. All these posts are in our test dataset. Among all the test data set, 156 of the posts that belong to the class negative are predicted correctly to the class negative and 95

posts that are related to the other topics, business/meaningless posts, are predicted indirectly as relevant posts. As we expected, the SVM classifier classify the most number of the posts with the most accuracy.

Classifier	Accuracy
Non-Linear SVM – Poly kernel – degree 1	0.80
Non-Linear SVM – Poly kernel – degree 2	0.17
Non-Linear SVM – Poly kernel – degree 3	0.82
Non-Linear SVM – Poly kernel – degree 4	0.82
Non-Linear SVM – RBF kernel	0.82
Non-Linear SVM – linear kernel	0.85
Logistic regression	0.85

Fig. 4.16. Accuracy of Different Classifiers

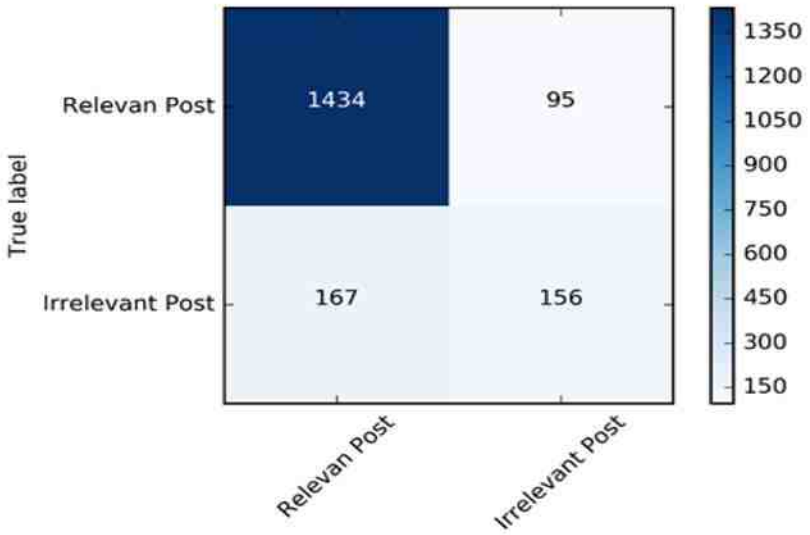

Fig. 4.17. Confusion Matrix Showing the Accuracy of the SVM Classifier

We tried different classifiers and the output of testing different classifiers which also is shown in the Figure 4.18 is displayed. So, among different features, we used logistic regression. In order to tune the hyperparameters of the logistic regression model, we trained the model with our train dataset and test the accuracy of the model applying different learning rate in order to use a specific learning rate parameter for our model. Learning rate is one of the hyperparameters which will affect to the accuracy of the logistic regression model and the running time of this model. The output of using different learning rate parameter is shown in the Figure (4.19). As we see in Figure 4.19, we implemented a variety of learning rates ranging from 10^{-6} to 100000 and determined the error with each of the learning rates. As we see in Figure 4.19, the learning rates below the 0.01 we see underfitting and for the learning rates above 1, we have overfitting. To be more accurate in our selected hyperparameter, we analyzed our model considering the concept of overfitting and under fitting. Overfitting happens when our train model can classify the train data set with a very high accuracy, but the trained model is not working properly to classify the test data set. Underfitting happens when the model is not able to fit the train and test data set well, and the accuracy of the system is very low. In the Figure 4.20, the error of the train and test model using different learning rate shows that the model have the appropriate fitting when the hyperparameter is 0.01, so in our model we used the learning rate 0.01 to train the classifier.

Dimensionality reduction

Dimensionality reduction is the process for reducing the number of variables in the process which can be used for different purposes. In this case study, sets of similar features that probably would transfer the similar attributes or the features which are irrelevant to our specific topic (business) are selected and filter out from our datasets. Then, we will continue our process with the remaining subsets that we have. There are different techniques for dimensionality reduction purpose such

Fig. 4.18. Classifications

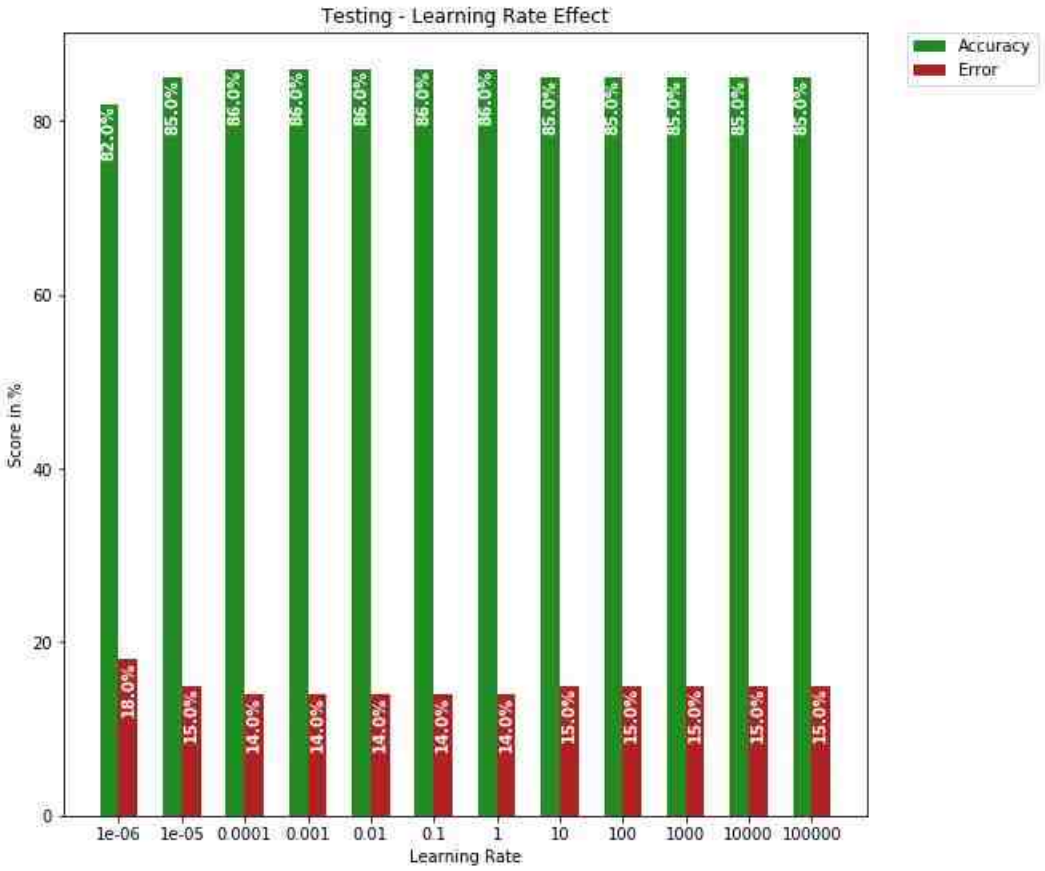

Fig. 4.19. Accuracy of Logistic Regression Classifier for Different Learning Rates

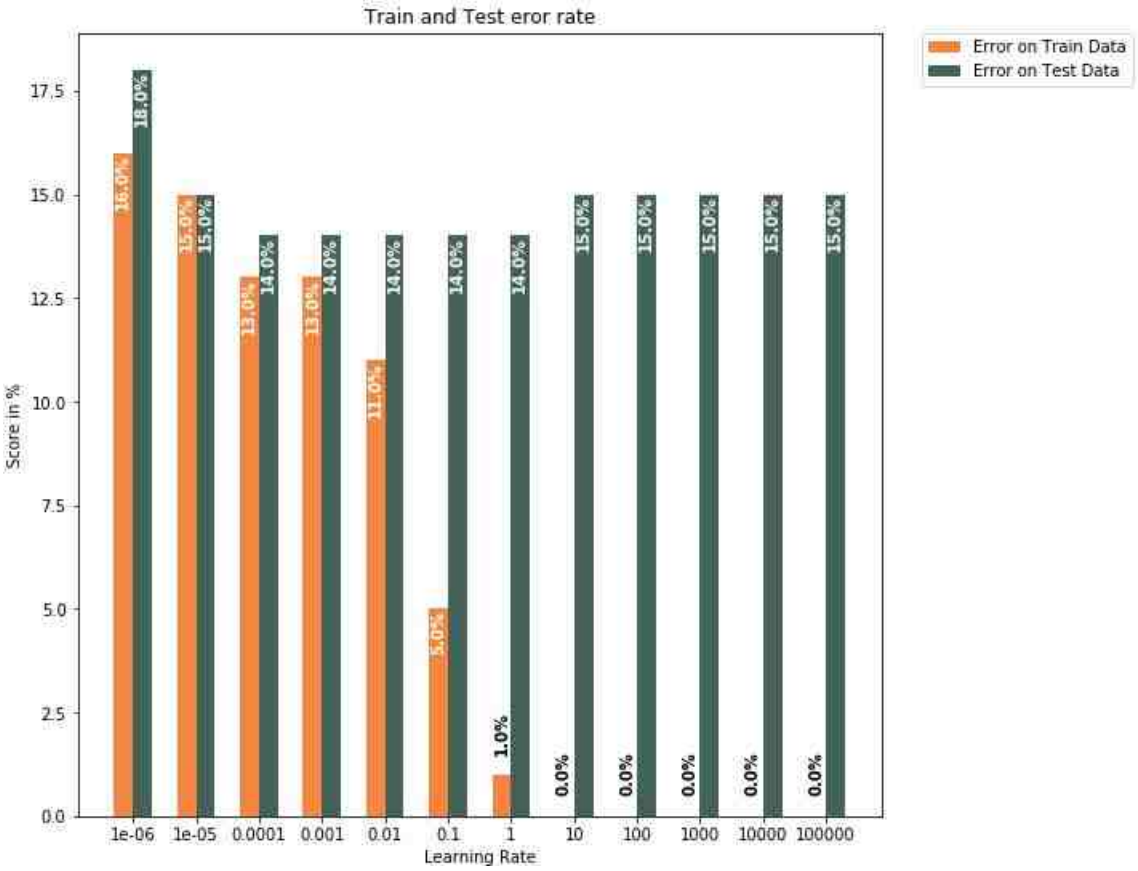

Fig. 4.20. Overfitting and Underfitting in Logistic Regression

as Principle Component Analysis (PCA) and linear discriminant analysis (LDA). In this case study, we used the dimensionality reduction technique for dimensionality reduction with the aim of data visualization. The PCA method is used to map the data linearly to lower space by removing some of the features exist in our binary matrix. The binary matrix is the output of the Bag of word. PCA method calculate the eigenvalues and eigenvectors of the matrix and then would keep only the largest eigenvalues and eigenvectors which represents the most variance in our matrix. In this approach we keep only those columns features of the matrix that transfer the most variance of the textual information. The output of using the PCA is illustrated in the Figure 4.21. In Figure 4.21 the red points correspond to the global posts that are related to the business topic. The red points in our figure belongs to the class

positive and the green points corresponds to the global posts that are related to the other topics except business or meaningless posts. The output of using PCA method shows the accuracy of our classification model in classifying the textual information. However, the logistic regression model is classifying the global posts with the very good accuracy, the dimensionality reduction is not able to show the output of logistic regression model as well.

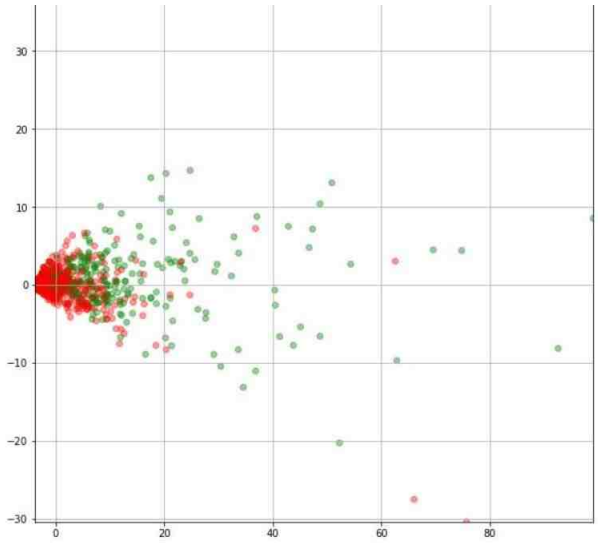

Fig. 4.21. Output of Using PCA for Data Visualization

4.4 Ground Truth Matrix

Ground Truth Matrix (GTM) used to represent the information in a matrix. The entries of this matrix refer to the link between users and items (global posts). If the entry has a low value means the link between users and post is weak, otherwise it is high. Our goal is to build the GTM matrix considering user and post explicit and implicit features. The Figure 4.22 illustrates the GTM matrix.

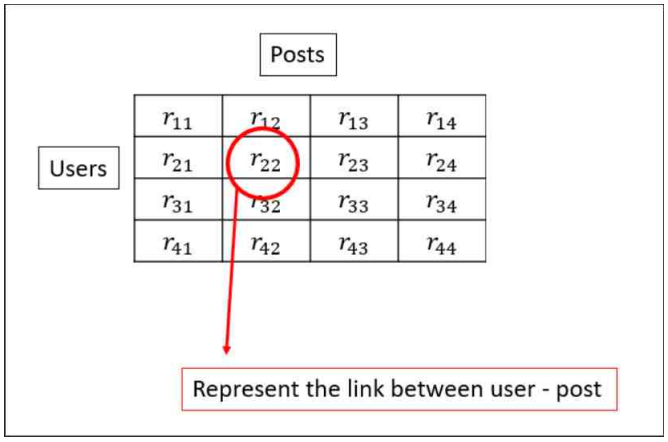

Fig. 4.22. Ground Truth Matrix

4.5 Matrix Factorization

Our recommendation system

By using the proposed novel hybrid recommendation system, we built a matrix to indicate the users preference to all observed items. In this matrix, the rows correspond to the users and the columns corresponds to the posts all belonging to the business course category. If there is no action between user u and item i, a zero value would assigned to the numerical entry in the matrix. There are no negative entries in the matrix. As we mentioned, the entries in the matrix indicates a user preference to an items i through r_{ui}. And, in this non-negative matrix the high entry indicates the stronger preference and the lower entry indicate the slight preference of user u to item i. As can be seen in the Figure 4.23, some user features and some posts features of each global post fed to our hybrid recommendation system to make the implicit entry in our Ground Truth Matrix (GTM). The GTM is a non-negative matrix, in this matrix the rows correspond to the users and columns correspond to the posts and the entries in this matrix represent the link between user and post.

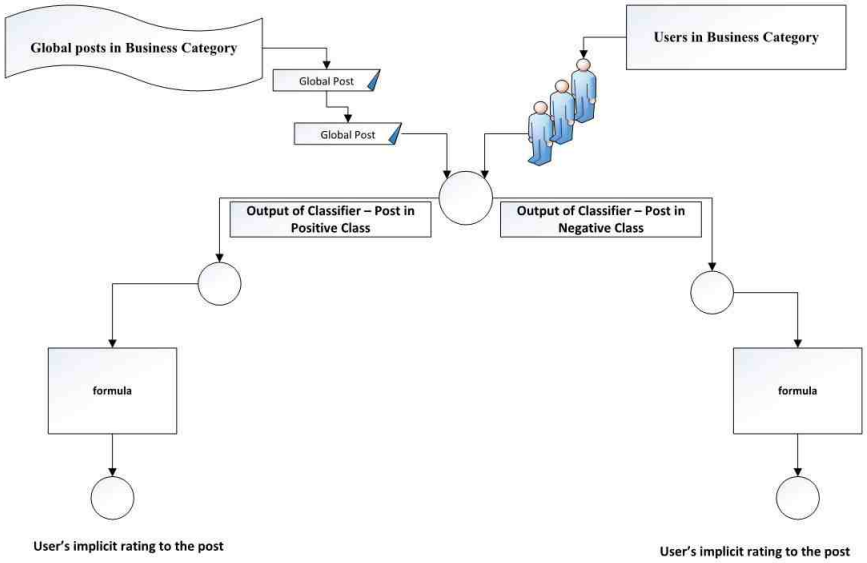

Fig. 4.23. Hybrid Function Considering Explicit and Implicit Features

The implicit entry of using hybrid recommender system developed using the following function:

$$User - Post = e^{x} \cdot z \cdot \alpha \tag{4.11}$$

In Equation (4.11) the hypothesis function defines the indirect link between the users and the posts. If a user is creator of a post, then to compute the link between user and post, we are using Equation (4.12). In Equation (4.12), the $X_i = 1$ and $CP = 1$, y depends on the role of the user, can be either 1 or 2. It is a 2 when a user is instructor and 1 if the user is student. α is computed using Equation (4.14) when parameter t is the number of the weeks from the time post created until present to consider how old the post is, and c_i is number of comments written under that post i.

$$z \cdot e^{R_i} \cdot CP \cdot \alpha \tag{4.12}$$

where,

$$z = x_i + \log(1 + \sum c_i) \cdot y \tag{4.13}$$

where,

$$\alpha = 1/\log(1+t). \tag{4.14}$$

If a user has rated a post then to compute the link between user and post in GTM, we are using Equation (4.15) where x is computed using Equation (4.16), and α and z are computed using Equations (4.13) and (4.14).

$$GTM_{i,j} = z \cdot e^{x} \cdot CP \cdot \alpha \tag{4.15}$$

where,

$$x = R_i(R_{avg-rating-star} + R_{user-rating-star}). \tag{4.16}$$

Regarding the posts that user does not rate nor create, to compute the entry in GTM matrix, we are using the Equation (4.17), where α and y are computed in similar way as before, but for computing z, Equation (4.18) is used.

$$GTM_{i,j} = z \cdot e^{(R_i)} \cdot (\alpha) \tag{4.17}$$

where,

$$z = \log(1 + \log(1 + \sum c_i) * y. \tag{4.18}$$

In Equation (4.12), the CP or Post-Course parameter has a binary value. Each post created in one specific course and the users who had reaction to the post via creating the post, rating or reflecting on the post are also belonging to a specific course. The PC parameter represents the relation between a user and a post through a course. Users can belong to more than one course in the specific course category, but each post comes from one specific course. In our proposed method, we tried to make a link between users and posts through the course concept by considering the fact that the users interested in having some reaction to the posts that belongs to one of the courses that the user is already registered in. Moreover, the priority of the course instructor is to share some important posts with everyone in the business course. So, we tried to analyze the user-post relationship through course. However, we did not access to the data to compute the PC parameter since these data was not shared

with me. So, in the Equation (4.15), if user and post are from the same course then the CP parameter is 1, otherwise it is 0. In all our equations, the parameter R_i is the probability that the content of the posts belongs to class positive or negative using logistic regression model. Parameter z in the Equations (4.11) could have several formulas based on the user's action in the CN.

Based on our recommendation system when the user makes any interaction with the post, an implicit entry would assign in the matrix which is corresponding to the user-post action. If there is no action then it is considered that the user already has seen the post but did not like to make any reaction to the post, so the rating value would have a very low value. However, in the future if the user does not have any reaction to the post created recently then the rating value would be 0 which means that he/she may want to see the post in the near future. The link between user-post for 0 rating values will be calculated using the hybrid function.

The values of our hybrid recommendation model analyzed. The frequency of the values in this matrix is illustrated in the Figure 4.24. Based on the following histogram, most of the elements in this matrix are between zero and one.

At the end, using matrix factorization, we extract more user and post features to make a link between users and posts. Each row in the matrix correspond to a user. For each row, all the entries would be sorted and then the posts with the highest values would be recommended to the users on the top. Algorithm 1 demonstrates the whole process of our system to recommend posts to the individual.

4.6 Evaluation of our Recommender System

Alternating Least Square method

In order to measure the accuracy of the system to predict the best match posts to each individual user based on their history to the system, the least square method is used. All the values of GTM matrix are positive, so before using the Alternating Least Square (ALS) method to predict the highest rating posts to the individual users

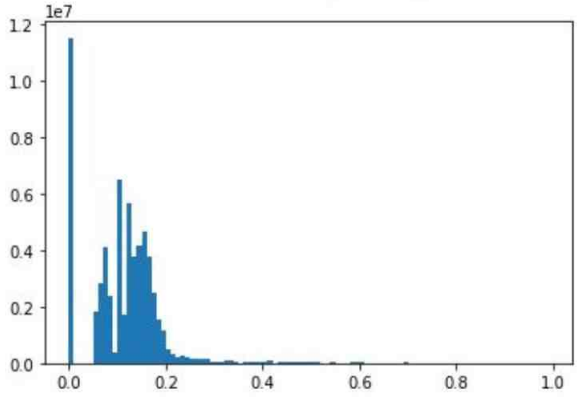

Fig. 4.24. Frequency of the Number of Entries in GTM Matrix

we need to define a threshold to assign zero to some of the elements in this matrix and make them zero then the most informative post that would be recommended to the users could be a series of the posts included the new informative and relevant posts coming to the system recently and/or posts that already posted in the CN and user showed some reaction to them via comments or give some rating star to the posts.

In our previous steps, we built a matrix to demonstrate the link between users and posts. Rows of this matrix correspond to the users and columns correspond to the global posts. Each element in this matrix represents the link between users and posts considering the users' behavior in the business course category. As a consequence, we recommended the k personalized educational posts which are the top k selected global posts to the users considering the users implicit feedback to the global post and users behavior in the system. The top k post selected to be recommended to the users considering the users' behavior in the whole system. Moreover, this method is a reliable and effective method that ameliorate the challenging issues such as cold-start problem (new users) and data sparsity. The outline of evaluation of our Recommender System is illustrated in Figure 4.25.

In our ground truth matrix, each element defines as a confidence variable r_{ui} that measures level of interest of a user to the specific post considering all his/her reaction into the network such as his/her rating and indirect reactions like output of classifier.

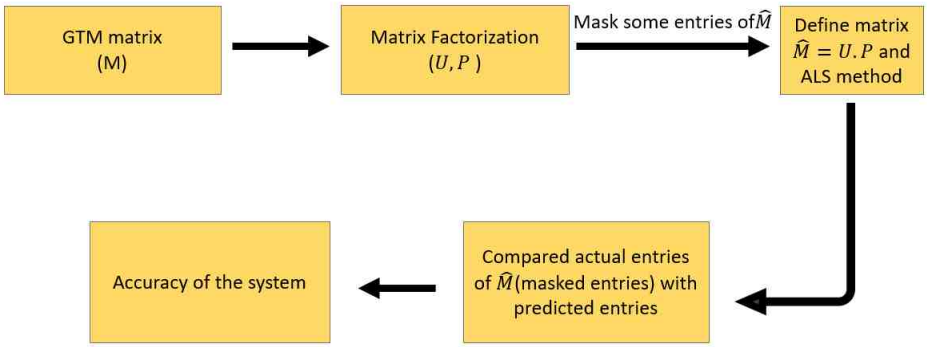

Fig. 4.25. Outline of Evaluation of our Recommender System

We define a binary matrix P. In binary matrix P whenever the r_{ui} in our ground truth matrix is greater than zero, the P_{ui} entry is 1, otherwise, P_{ui} assigned 0 (when user u take no action to the global post i). A user's may give 3 star to a post for some reason but the quality of a post. He/she may give 3 star to a post because the post shared with his/her close friend or because it is not about a topic that is totally unrelated to a specific course topic but his favorite food! Or he randomly just wrote down some comment under the post to get more Anarseeds while ignoring the content of the post. As a consequence, we consider a parameter in our hybrid recommendation system which is so-called confidence level. The confidence level measures the preference of a user who reacted to a post. Therefore, measuring the confidence level of rating value to have a more accurate indication is important. As a consequence, a parameter C_{ui}, which is defined in the Equation (4.19) would measure the confidence level of rating value for user-post:

$$C_{ui} = 1 + \alpha.r_{ui}. \tag{4.19}$$

In Equation (4.19), parameter α is set to 40 the same as the one used in the paper [12]. At the end, our goal is to factorize the GTM matrix to user feature matrix and post feature matrix that defines a feature vector x_u for each user u and y_i feature vector for each post i respectively. The P_{ui} entry of GTM matrix is computed in the Equation (4.20):

$$P_{ui} = x_u^T . y_i. \tag{4.20}$$

For measuring P_{ui}, first we need to calculate the feature for all users and posts. These user-feature and post-feature values are computed by minimizing J in the Equation (4.21) as our cost function (going back and forth between user features (x_u) and post features (y_i) and define a regularization term λ to prevent overfitting):

$$J = min \sum_{ui} C_{ui} . (P_{ui} - x_u^T . y_i)^2 + \lambda . ||x^2|| + ||y^2|| \tag{4.21}$$

C_{ui} is defined in (4.21) measures the level of interest of a user to a post since the user may have some reaction to a post by giving a 3 star to the post, but the reason of his/her reaction can be the result of different issues such as the post shared by her/his best friends. So, in the process of predicting the factorize matrices, the C_{ui} computed using the equation 4.22. The parameter α is 40 the same value as the value used in paper [12]. In each iteration, the values are updated in the Equation (4.21) in order to minimize the cost function and to increase the accuracy of the system for the new data that coming later to the system, we need to update our factorized matrices and P_{ui} to recommend different sets of posts to users during a time.

$$C_{ui} = 1 + \alpha P_{ui} \tag{4.22}$$

After computing the user-feature and item feature matrices, we calculate the predicted $\widehat{M_{ui}} = x_u^T \cdot y_i$ to estimate the $\widehat{GTM}$. The $\widehat{P_{ui}}$ is a binary matrix that is defined from $\widehat{GTM}$. The least square implemented technique uses a new approach to predict $\hat{M}_{ui}$.

The $\hat{M}_{ui}$ is computed by multiplying the factorized matrices. Then from $\hat{M}_{ui}$ and applying a specific cutoff (threshold), we computed $\hat{P}_{ui}$.

We predict the confidence level of a user u to the post i by considering the similarity of item i to all the items that user u had reaction to them and considering his/her action to all those posts. This prediction is unique for each individual considering the

powerful pre-processing in this model and also considering the fact that from each users perspective different number of posts could be in the same category and similar to each other.

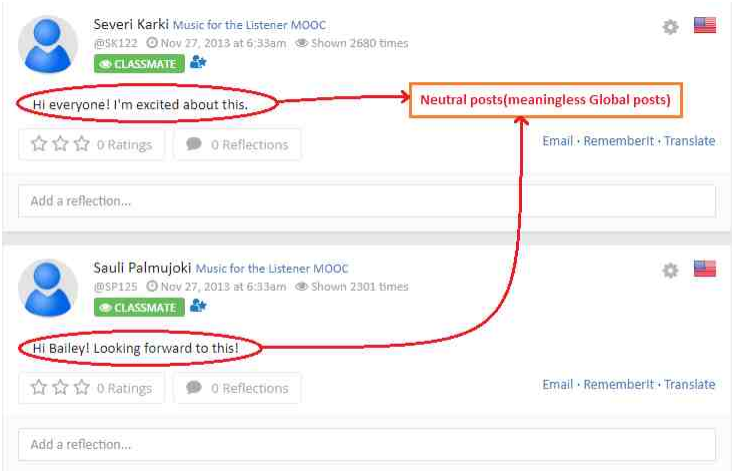

Fig. 4.26. Neutral Global-posts

In our case study, we assessed the accuracy of our model on our datasets. The experimental results show that our proposed model has a very good performance to recommend educational contents to the users considering implicit features and explicit features in this e-learning network. Our proposed model recommends posts to the users based on users past behavior into the system and considering the other users behavior in the system. Our evaluation result shows that the accuracy of our whole model is 98% percent. Our proposed model addressed the cold start issue by recommending the posts with the highest rating value to these new users considering the other users behavior to the system then based on these new users feedback the system update itself and recommend them new sets of posts. Regarding the sparsity issue that almost all the traditional collaborative filtering is faced with, our machine learning classifier technique helps us to address the sparsity issue. Algorithm 2 shows the experimental results step by step.

The Algorithm 1 represents our proposed recommender system (RS) step by step. In our proposed RS, first, we are taking user and post features as our input, then for content analysis, we applied Bag-Of-Words model on all the global posts. After that, for content analysis, we applied Bag-Of-Words model on all the global posts. Then, the output of Bag-Of-Words model feed to the logistic regression model with parameter R_j. In the next step, for each user i and each post j the entries of GTM matrix is computed (GTM_{ij}) considering all the implicit and explicit features in the system. Afterward, we developed the matrix factorization $\hat{M}$ out of GTM matrix with the aim of extracting more user and post features. At the end, for each row i of matrix $\hat{M}$ corresponding to a user, all the entries sorted and the posts will be recommended to the users based on the sorted entries.

In Algorithm 2, in order to measure the accuracy of our recommendation system, the average of ROC curve over all the users computed. In order to measure the system accuracy, first some of the entries of $\hat{M}$ masked to be predicted and compared with their actual values. So, we considered the users with at least one masked value. In the next step, we defined different threshold T_c. For each treshold T_c, the binary matrix $\hat{P}$ computed out of $\hat{M}$ and compared it with actual matrix P. P is a binary matrix from our actual GTM matrix wherever entries of GTM is positive is 1 otherwise is 0. Afterward, for each T_c, the TPR and FPR calculated. Then, for each user and all the T_c, the (TPR, FPR) plotted and the AUC curve computed. After computing all the AUC curves, in order to measure the accuracy of the system, the average over all the measured AUC curves computed.

Algorithm 1 Recommender system (RS)

Input $(user, posts)$ **Output** (sets of posts to the users)

for each post (P_j): **do**
    y=Implement Bag-of-Words model on post j
    R_j=Implement logistic regression classifier on y
    Fed R_j and use-post selected features to the central RS system
end for
for each user (u_i): **do**
    for each post (p_j): **do**
        Compute GTM_{ij}
    end for
end for
Recommender system
Make a matrix factorization $\hat{M}$ from GTM matrix
sort row i of $\hat{M}$ and recommend posts to the user i based on sort

Algorithm 2 Calculate Average of ROC Curve over all user(ui)

Input $(user, posts)$ **Output** $AUC = Avg(\Sigma(AUC(ui))$

$Sum = 0$
$Count = 0$
for each user (ui): **do**
  if Use u_i has some masked posts P **then**
    for each threshold Tc: **do**
      Calculate $\hat{P}$ from the output of feature matrices
      Compare $\hat{P}$ with actual P
      Calculate recall(sensitivity) $TPR = \frac{TP}{TP+FN}$
      Calculate $FPR = \frac{FP}{FP+TN}$
    end for
    plot (TPR,FPR)
    Compute $AUC(ui)$
    Compute $Count = Count + 1$
    $Sum = Sum + AUC(u_i)$
  end if
end for
Accuracy= $Sum/Count$
return Accuracy

5. CONCLUSION AND FUTURE WORK

In our proposed method, we developed a hybrid and dynamic recommendation system (HDRS) to analyze Business datasets in CourseNetworking environment. Our proposed method recommends the personalized contents to the users based on the user activity in the online and real network CN. Contents in this social and learning environment are global posts which are shared by the users (global classmates) who belong to the business category. We developed a personalized recommendation system using machine learning classifier and NLP for feature extraction to recommend global posts to the users in such a way that the model will display different sets of global posts to the users in their profile page. Our hybrid recommendation system considers the behavior of per user to the system and users correlation with all the global posts as well. It also considers the other users behavior in the system and their link with the global posts which shared from their global classmates to predict the rating value representing the link between a user and an unseen post. In our proposed model, we observed that the system addressed the cold-start issue since we are considering the other user-post interaction in the system for unseen post predictions. Regarding the sparsity issue, our proposed model addressed the sparsity issue which almost all the traditional recommender systems are challenged with. Our proposed model evaluated with the novel approach (ALS). The experimental results show that accuracy of the system to predict a correct rating value in our user-post matrix is 98 % which confirm that our proposed model has a great accuracy considering users behavior into the system. The system performance of the proposed model is very good in comparison with the current model applied in the CN which is static.

Developing a reliable and accurate recommendation system to provide relevant and attractive contents (global posts in this case study) will bring advantage to the CN network at the end. Consequently, increasing the number of attributes in the

system would increase the accuracy of our system. The more information we have, the more accurate a system we would have. In Equations (4.12) and (4.15), the CP value could be an important parameter to increase the accuracy of the system. In the future when the number of followers and followees of the users increased, considering the users interaction with the other users would be very impactful factor in parameter CP. However, now most of the users isolated and do not have much followers and followees. Besides, the amount of time that the users spend on a post and/or the amount of time they spend to have a reaction on a post could be an impactful factor to increase the accuracy of the RS system.

Moreover, recommendation systems can be developed for the other purposes beside post recommendations such as job or skill recommendations. Also, we can provide an option in the system for announcement in order to share an important news to the users like the LinkedIn. Developing an automatic skill endorsement system for the CN environment help the users to expand their network and their knowledge base which is very helpful for them in their career path.

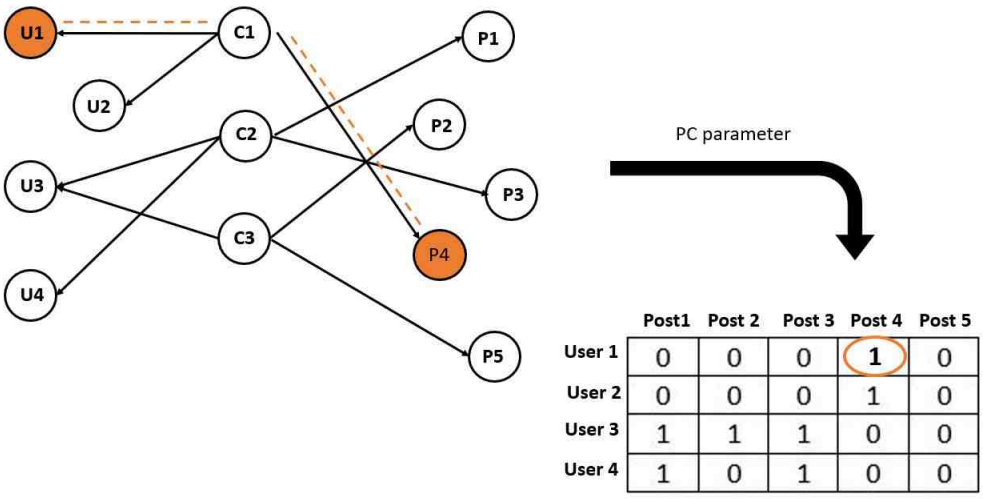

Fig. 5.1. Post-Course Rating Values

REFERENCES

REFERENCES

[1] X. Zhou, Y. Xu, Y. Li, A. Josang, and C. Cox, "The state-of-the-art in personalized recommender systems for social networking," *Artificial Intelligence Review*, vol. 37, no. 2, pp. 119–132, 2012.

[2] G. Strang, *Introduction to linear algebra*. Wellesley-Cambridge Press Wellesley, MA, 1993, vol. 3.

[3] C. C. Aggarwal *et al.*, *Recommender systems*. Springer, 2016.

[4] A. L. V. Pereira and E. R. Hruschka, "Simultaneous co-clustering and learning to address the cold start problem in recommender systems," *Knowledge-Based Systems*, vol. 82, pp. 11–19, 2015.

[5] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins, "Eigentaste: A constant time collaborative filtering algorithm," *information retrieval*, vol. 4, no. 2, pp. 133–151, 2001.

[6] M. Maniktala, S. Sachdev, N. Bansal, and S. Susan, "Finding the most informational friends in a social network based recommender system," in *Annual IEEE India Conference (INDICON), 2015*. IEEE, 2015, pp. 1–6.

[7] A. J. Chaney, D. M. Blei, and T. Eliassi-Rad, "A probabilistic model for using social networks in personalized item recommendation," in *Proceedings of the 9th ACM Conference on Recommender Systems*. ACM, 2015, pp. 43–50.

[8] P. Gopalan, J. M. Hofman, and D. M. Blei, "Scalable recommendation with hierarchical poisson factorization." in *UAI*, 2015, pp. 326–335.

[9] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proceedings of the fourth ACM conference on Recommender systems*. ACM, 2010, pp. 135–142.

[10] G. Linden, B. Smith, and J. York, "Amazon. com recommendations: Item-to-item collaborative filtering," *IEEE Internet computing*, vol. 7, no. 1, pp. 76–80, 2003.

[11] D. W. Oard, J. Kim *et al.*, "Implicit feedback for recommender systems," in *Proceedings of the AAAI workshop on recommender systems*, vol. 83. WoUongong, 1998.

[12] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Eighth IEEE International Conference in Data Mining,' ICDM'08.* IEEE, 2008, pp. 263–272.

[13] V. A. Rohani, Z. M. Kasirun, and K. Ratnavelu, "An enhanced content-based recommender system for academic social networks," in *Fourth International IEEE Conference on Big Data and Cloud Computing (BdCloud), 2014*. IEEE, 2014, pp. 424–431.

[14] E. Cogo and D. Donko, "Clustering approach to collaborative filtering using social networks," in *IEEE 4th International Conference on Electronics Information and Emergency Communication (ICEIEC)*. IEEE, 2013, pp. 289–292.

[15] F. E. Walter, S. Battiston, and F. Schweitzer, "A model of a trust-based recommendation system on a social network," *Autonomous Agents and Multi-Agent Systems*, vol. 16, no. 1, pp. 57–74, 2008.

[16] B. Wellman, "Computer networks as social networks," *Science*, vol. 293, no. 5537, pp. 2031–2034, 2001.

[17] B. A. Huberman and L. A. Adamic, "Internet: growth dynamics of the world-wide web," *Nature*, vol. 401, no. 6749, p. 131, 1999.

[18] A. Abdul-Rahman and S. Hailes, "Supporting trust in virtual communities," in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, 2000.* IEEE, 2000, pp. 9–pp.

[19] R. Falcone and C. Castelfranchi, "Social trust: A cognitive approach," in *Trust and deception in virtual societies*. Springer, 2001, pp. 55–90.

[20] T. Grandison and M. Sloman, "A survey of trust in internet applications," *IEEE Communications Surveys & Tutorials*, vol. 3, no. 4, pp. 2–16, 2000.

[21] S. P. Marsh, "Formalising trust as a computational concept," accessed 07-16-18. [Online]. Available: https://www.nr.no/~abie/Papers/TR133.pdf

[22] L. Mui, M. Mohtashemi, and A. Halberstadt, "A computational model of trust and reputation," in *HICSS. Proceedings of the 35th Annual Hawaii International Conference on System Sciences, 2002.* IEEE, 2002, pp. 2431–2439.

[23] J. Sabater and C. Sierra, "Review on computational trust and reputation models," *Artificial intelligence review*, vol. 24, no. 1, pp. 33–60, 2005.

[24] J. Pareek, M. M. Jhaveri, M. A. Kapasi, and M. M. Trivedi, "Recommendation system using social networking," accessed 07-16-18. [Online]. Available: https://www.nr.no/~abie/Papers/TR133.pdf

[25] J. Kim, H. Kim, and J.-h. Ryu, "Triptip: a trip planning service with tag-based recommendation," in *Extended Abstracts on Human Factors in Computing Systems CHI'09*. ACM, 2009, pp. 3467–3472.

[26] X. Liu and K. Aberer, "Soco: a social network aided context-aware recommender system," in *Proceedings of the 22nd international conference on World Wide Web*. ACM, 2013, pp. 781–802.

[27] J. Zeng, M. Gao, J. Wen, and S. Hirokawa, "A hybrid trust degree model in social network for recommender system," in *IIAI 3rd International Conference on Advanced Applied Informatics (IIAIAAI), 2014*. IEEE, 2014, pp. 37–41.