Grenze International Journal of Computer Theory and Engineering , Oct 2016

Real Time Personalized News Recommender using OSN’s

S. Khan[1], Ch. Krishna Keerthi[2] and S. Punna[3]
[1]Muffakham Jah College of Engineering and Technology, Hyderabad, India.
[2]Assistant Professor, Muffakham Jah College of Engineering and Technology, Hyderabad, India.
[3]Nagarjuna Institute Of Technology and Sciences, India

***Abstract*—Virtual newspapers and magazines have become a popular means to read news stories from an enormous collection of news articles from round the world. To help users manage this flood of information, we develop customized news recommender system using an OSN, “Twitter”. Tweets from Twitter’s timeline are used to rank the news articles based on popularity of the article. Additionally, users create a profile of their interests and the news articles are ranked based on how well they match the profile. These two techniques are combined to create a hybrid news recommender system that recommends news articles to the user which are popular as well as in relevance with their user profile.**

***Index Terms*— Recommender system, OSN, Personalized, News, Jaccard, Cosine Similarity.**

I. INTRODUCTION

Due to the ever-increasing volume of information on the web, we can access an enormous amount of information from round the world. The main challenge nowadays is to find relevant information based from a never-ending data source. This drawback has led to the evolution of the recommender systems that present users with information related to their interests. Many organizations use recommendation systems to recommend various types of things to the user. For example, EBay recommends products for shopping to its users, IMDB recommends movies, and YouTube recommends videos based on users’ past history and preferences. Plus, there are online news sources also like NDTV, IBNLive etc. However, the main drawback is to filter and recommend the foremost interesting articles to every user in order that they're not presented with a flood of information to plow through. These articles ought to be associated with every user’s interests and also embrace those news stories that are generating lots of interest round the world.

In this paper, we develop a hybrid personalized news recommender system that recommends interesting news articles to the user using a micro-blogging service “Twitter”. Our hybrid recommender system ranks the news stories in different ways: We consider (1) the user’s profile to recommend articles to the user; and (2) the article's popularity with the help of tweets from Twitter’s public timeline. We present a novel approach to help users find interesting articles by merging the above two methods of ranking articles. The remainder of this paper is organized as follows. Section II describes Recommender Systems, Content Based Recommendation and Collaborative Recommendation. Section III discusses some related work on schemes for Popularity and Personalized recommendation. Section V explains the problems faced in earlier systems and in Section VII, our proposed Jaccard Index and Cosine Similarity Functions are introduced to find similarity between articles and present recommendations. Section IX explains the methods used for solution and Section X presents the results and analysis. Section XI concludes our work.

Grenze ID: 01.GIJCTE.2.3.552

II. LITERATURE STUDIES

A. Recommender Systems

Recommender systems are widely used to help readers filter through an ever-growing flood of information. Basically, these systems implement a method of Information Filtering. Recommender systems collect data from the users directly or indirectly and create user profiles based on this information. The user profiles are then used to generate recommendations. In direct information collection, the user typically rates items along with his regular tasks. For example, when purchasing an item, the user is asked to rate it with one or more stars. However, for indirect information collection, the recommender system analyzes the user's behavior with items during their normal activities. For example, when purchasing an item what are the other items user has shown interest in or purchased along with it. No extra user effort is required.

Traditional recommendation systems are classified based on what information they use and on how they use that information. Recommender systems are usually classified into three types, based on how the recommendations are made Content-based and Collaborative recommender systems.

Content-based Recommender Systems

These systems recommend an item to the user similar to the items the user has shown interest in the past. The contents of item are compared to items user has purchased or viewed earlier. These recommender systems are used in various areas such as websites, movies, blogs, restaurants, news articles etc. The user profile is constructed taking users interests into account. Many traditional content-based recommender systems depend on the attributes of the items themselves. The keywords associated with the items are used to match against the user profile to make recommendations. Hence, this approach only works for those applications whose items have contents in the form of text. Recommendations are done based on the comparison between the contents of the items and the keywords associated with the user profile that indicate his interests. Content-based recommenders use a weighting mechanism to rank the items by associating weights to the keywords and to differentiate between the items. There are several methods to calculate the weights of the keywords in the content. The most commonly used method is the TF-IDF (Term Frequency – Inverse Document Frequency) method.

Collaborative recommender systems

These systems recommend an item to the user based on people's tastes who have shown interest in the same items in the past. Their likes are considered while recommending items to the user. These systems determine similarity of items by analyzing User-item interactions. These systems are very popular and usually are used to recommend books, movies, music etc. Advantage of recommender systems is that they can recommend items for which little semantic is meaning is available (movies, books, etc.). A profile is created for each user (item) according to the similarity of other users (item) in the system. According to the profiles, items are recommended to target users according to the preferences of their similar users. There are two major types of algorithms for collaborative filtering (CF): user-based and the item-based. User-based algorithms find out the most similar neighbor users to a target user based on the similarity of ratings. For item-based algorithms, when user is interested in an item, similar items are also recommended to the user. Item similarity is based on items that are commonly purchased/liked together. If in the past, people who like Vampire diaries also like Twilight, then a new user who has watched Vampire Diaries should have Twilight recommended to them. Traditional collaborative recommender systems use of a similar method to make recommendations to the user. User-item ratings information is collected and each user is provided with a collection of items for them to rate according to their interests. Each user is represented by item-rating pairs, which contains the ratings provided by the user to several items. Next, vectors are created that represent users or items and a similarity measure is chosen. There are several possible measures to calculate the similarity between two vectors. Pearson correlation, cosine vector similarity, mean-squared difference and Spearman correlation are some of the popularly used metrics to measure the similarity between two vectors. Then, the next task is to identify the neighbors who will serve as recommenders. There are two techniques to identify the neighbors selected to make recommendations. With threshold-based selection, vectors whose similarity exceeds a certain threshold value are considered as neighbors of the target user. In contrast, with the top-n technique, the n-closest neighbors are selected for a given value of n. Finally, predictions are produced by calculating the weighted average of the neighbors' ratings, weighted by their similarity to the target user. Similar to content-based algorithms, collaborative filtering algorithms can be classified into two main categories, memory-based collaborative filtering algorithms and model-based collaborative filtering algorithms.

Hybrid recommender systems

These systems combine both content based and collaborative recommendation methods in order to improve the accuracy of the recommendations. The information gathered from either content-based or collaborative filtering approaches can be used for either memory-based or model-based algorithms. Memory-based systems calculate recommendations on-the-go based on the previous user behavior. Model based algorithms require user's feedback to make recommendations.

III. Related Work

A. News Recommender Systems

News recommender systems are widely used and are a promising research direction. With so many information sources, the Internet provides fast access to the millions of news articles around the globe. However, users need recommendations to help them find the most interesting articles from this flood of information. News recommender systems can be broadly classified into two types based on the type of recommendations made to the user. Some recommender systems take advantage of online social networking sites to provide interesting news articles to the user. Such recommendations are called popularity-based news recommendations since the articles are ranked based on their popularity identified from the social networking websites. Other recommender systems recommend interesting news articles to the user solely based on user interests. Such recommendations are called profile based news recommendations since they rank the news articles based on the user's interests. The following two sections explore the applications based on the popularity based recommendation and profile based recommendation techniques.

Popularity Based News Recommender Systems

News recommender systems are widely used to help readers filter through an ever-growing flood of information. Many researchers focus on using real-time social networking sites such as Facebook, Google Plus, and Twitter to identify the most popular and most current news stories. Many news recommender systems make use of activity on micro-blogging services, such as Twitter, to identify news items that generate a lot of interest. Often, news stories break on informal micro-blogs before they appear on the traditional news service websites. Because they are instant, and widely available, they provide a massive source of information on current events. However, because they are unmoderated, the quality of the information is variable. Alan et al. discuss a method to determine which Twitter users are posting reliable information and which posts are interesting. They identified future events based on the tweets from reliable Twitter users with the help of tense identification of their tweets. Micro-blog posts can also be used as a way of identifying the popularity of certain events. Smyth et al. represent users and items based on micro-blogging review of movies and used this technique with various movie recommendation strategies on live-user data. Their strategies combine the frequency of the tweets about a given movie with analysis of the tweets to identify positive and negative reactions. Phelan et al. focus on using micro-blogging activity to recommend news stories. Their recommender system, Buzzer, is applied to RSS feeds to which the users have subscribed. Buzzer mines the content terms from RSS and Twitter feeds and uses them to rank articles. Buzzer recommends news stories based on three different retrieval strategies namely, Public Rank, for mining tweets from Twitter's public timeline, Friend's-Rank, for mining tweets from people the user follows and, Content-Rank, for ranking articles based on term frequency alone and scoring the articles based on the frequency of occurrence of the top RSS terms. It matches the mined terms with the index of RSS items. The overall score of each article is calculated by accumulating the TF-IDF (term frequency - inverse document frequency) scores across all the terms related within each article. They used two new recommendation strategies where the tweets are mined from Twitter's public timeline and from people the user follows and these mined terms are matched with the RSS terms that are gathered from all users' subscriptions in the Buzzer community. Also each user is allowed to sign up to a daily digest of email stories to store each user's response. In, they extended their work by considering two additional strategies. They considered the public-rank and the friend's-rank strategy over all the news articles in the Buzzer system rather than just considering the articles from the users' index. A user trial was conducted on 35 active Buzzer users for over a month. During this period, they collected a total of 56 million tweets from the public timeline and 537,307 tweets from the social graphs of the 35 registered Buzzer users. They observed the user click behavior for all the strategies. The results indicated that the public-rank and friend's-rank strategy outperformed the other strategies and is considered better when compared to the traditional keyword (TF-IDF) based strategy.

Profile-Based News Recommender Systems

Profile based, or personalized, news recommender systems recommend articles to the user based solely on his/her interests. A user profile is built based on the preferences or interests of the user. In one of the earliest news recommendation systems, Pazzani et al. used, News Dude, a personal news recommending agent that uses TF-IDF in combination with Nearest Neighbor algorithm in order to recommend news stories to users. They built their user profile implicitly and divided the user interests into short-term interests and long-term interests. They developed a hybrid user model that considers both long-term and short-term interests and found out that this model outperformed the models that consider either of these interests. Similarly, Michael et al describe a content-based recommendation system that recommends a story to a user based upon a description of the item and a profile of user's interests. More recently, Cantador et al. present a system, News@Hand, that has similar goals but which incorporates semantic web technologies News@Hand uses semantic annotation based on ontologies to group news items and then recommends news stories to users that match their semantically-based profile. Wouter et al. described ontology based methods to recommend news articles to the users depending on their interests. The user profile is based on the news articles browsed recently and the ontology is provided by, Athena, a framework built for news personalization service which is an extension of their framework, Hermes, mentioned in their earlier work. The ontology is developed to store the concepts and their relations to the news items. Athena uses the traditional keyword-based recommendation technique along with the semantic-based recommendation algorithms to compare the unread news articles with the user profile. The news items that most match with the user profile are recommended to the user. The semantic-based recommendation algorithms consider the concepts and the semantics of the text to determine the relation between various keywords. Various techniques are used to identify the semantic relatedness between the keywords in the articles and explained in detail below. The first technique is a simple mechanism called concept equivalence. The user profile is considered as a set of concepts based on the interests of the user. The news article is also considered as a set of concepts pertaining to the domain of the article. For a new article, the interestingness of the articles is calculated by the intersection of the above two sets. The other techniques binary cosine and the Jaccard similarity coefficient compute the cardinality of intersection of above two sets relative to the cross product and union of the above two sets respectively. The semantic relatedness of two keywords is calculated by creating a vector that represents the keywords and calculating the cosine similarity between the two vectors. Second technique is to use jaccard coefficient to calculate similarity between the frequencies of words. The advantage of this approach over the other three techniques is that it takes the related concepts of a concept into account that occurs in a text. For the evaluation of this approach, they considered 5 users with different news interests and each user is provided with 300 different news articles and is asked to rate all the articles by skimming through the summaries. The articles rated by the user are divided randomly into two sets, training set and validation set. The training set is used to create the user profile. The validation set is used to determine the similarity of the news article to the user profile. To determine the performance of this system, measures like accuracy, precision, recall and specificity are used. The results indicate that the semantic-based news recommender performs better than the traditional recommender systems based on TF-IDF. Our news recommender system incorporates both the strategies explained above, popularity and profiles, to present a novel hybrid approach to recommend interesting news articles to the user.

IV. MOTIVATION

In the real world people are always in a hurry to finish their tasks. Newspapers and magazines have given way to virtual news reading. However, people are swamped by a huge amount of news stories from these news channels. This is where news recommender systems come to our rescue. Our research on content based information filtering and retrieval has led to the study of recommendation systems. In the work done by Nirmal Jonalagedda, a very interesting approach to recommendations has been done.

V. Problem Domain

There are many websites like Google News, Yahoo News or news channel websites like NDTV, IBNLive which offer news to the audience. But due to the ever-increasing volume of news on the net, we are able to access a huge amount of information around the globe. The main problem is that users are flooded by news articles of little interest to them. The key challenge these days is for the users is to seek current and relevant news based on their interests. This drawback has resulted in the evolution of the recommender systems that facilitate users to seek information they need based on their interests rather. Many people prefer first-hand information from real people from the place of event. Hence, micro-blogging services must be taken into account.

VI. Problem Statement

Our work aims to (1) present an analysis between Jaccard coefficient and cosine function to find similarity between articles and (2) use the best one in the algorithms for making Hybrid recommendations to user.

VII. Innovative Content

This research uses Jaccard index to give a comparative analysis of the performance of system, with cosine similarity function. Along with news categorized into Topics like Business, Sports etc. categories of location-mainly continents have been considered.

VIII. Problem Formulation

KNN algorithm uses the cosine similarity function for calculating the nearest neighbors of an item, i.e. news articles. Performance of the function needs to be assessed as earlier works have not proven the effectiveness of cosine similarity against any other for recommending news articles to users. Performance is assessed by the following parameter: Size of input data and Relevancy of results. Size of data needs to be analyzed to check performance of system when lesser data is input and when large dataset is input. Relevancy of results of both functions must be evaluated when a query is input to the system. Presenting hybrid recommendations to user based on Hot news from twitter and news divided into categories and next, with news divided with respect to locations.

IX. Solution Methodologies

The basic concept of our work is to recommend articles to the users which are most interesting to them based on a combination of results of the two modules, i.e. popularity and profile based recommendation. The user profile collects past interests of user and interest of the whole public is collected from tweets using twitter's public timeline which is in done two parts. The first step is to collect an assortment of news articles to use as the input of the system. We collect current news stories from RSS feeds of news sources online and create an inverted index. Now, tweets from twitters public feeds are collected and used as search keywords to query against the inverted index and find the news articles most similar to the tweets. By finding how many tweets are relative to the news article we can evaluate the “Trendiest” news articles. In the second part, news articles associated with high level categories like Politics, Health, Business, etc. are collected from RSS feeds of news websites and queried against the first collection of news articles. Each of the current articles are categorized into the different topics automatically.

The users construct their profiles by entering their preferences of topics into a web form. The preferences are matched with the topics of news articles and recommendations are made automatically. The third part of our system is to build a hybrid recommender system by combining the results of the two methods, to recommend news to users which is based on their preferences and also trending on the internet among netizens.

The system consists of three modules:

1. Popularity-Based News Recommendation system
2. Profile-Based News Recommendation system

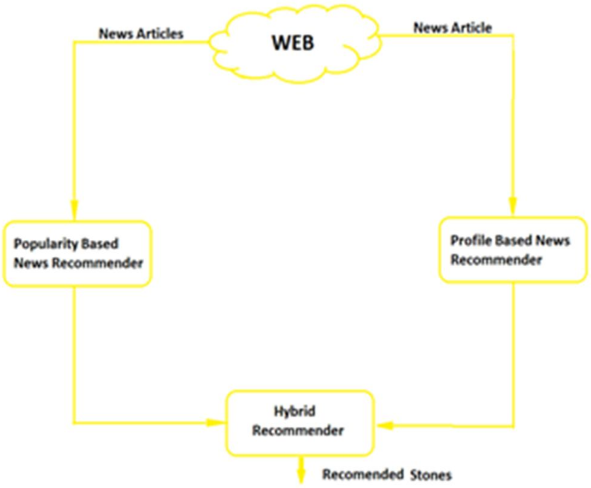

News Articles
WEB
News Article
Popularity Based News Recommender
Profile Based News Recommender
Hybrid Recommender
Recomended Stones

Figure 1. Architecture of the Hybrid News Recommender System

1. Hybrid News Recommendation system

Figure 1 shows an architectural diagram of our Hybrid News Recommender system.

A. Popularity-Based News Recommendations

Figure 2 shows an architectural diagram of the Popularity-based News Recommender system. First, the RSS articles are collected from a news source such as CNN or the BBC that organize their stories by category, e.g. Sports, Business, Politics, and Entertainment, etc. The article, and their associated categories are stored locally. The RSS articles are pre-processed to remove unnecessary content (html tags, numbers, etc.) while preserving the textual content. The pre-processed articles are then indexed.

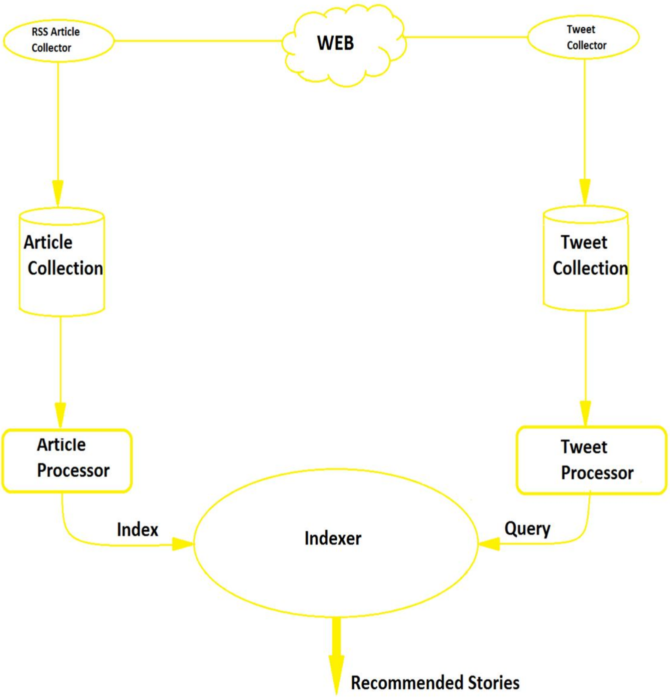


Figure2. Architecture of popularity based system

In order to identify which news stories are most popular, we collect tweets from Twitter. The tweets are collected from Twitter’s public timeline by using Twitter’s streaming API. The collected tweets are stored in files in JSON format. The Tweet Processor eliminates unwanted noise and preserves the actual tweet content. Each processed tweet is queried against the inverted index. The weights for each article are accumulated across all tweets to produce a popularity weight for the article. Thus, the Popularity_Wt for article i is:

$$Popularity\_Wt_i = \sum_{t=1 \text{ to } T} CosineSimilarity(Article_i, Tweet_t)$$

where T is the number of tweets in the collection.

B. Profile-based News Recommendations

Figure 3 shows the architectural diagram of the Profile-Based News Recommender system. The profile based recommender system uses the same article collection as the popularity-based recommender system. Although articles are placed in only one category by the website editor, they may actually partially belong to more than one category. To allow for this, each article is classified into all 5 potential categories using a k-nearest neighbour classifier, the classification module of the KeyConcept project. We store the top 3 most similar categories (and their similarity scores) for use in profile matching. In order to do fast lookup by category, we build an inverted index.

Next, each user creates a profile by manually scoring the topic categories presented on the Web form. This user profile is used to identify articles that best match their profile. The profiles and the articles can be viewed as feature vectors where each category is a feature. The similarity of each news article to user's profile is calculated using dot product between the user profile's category vector and the article's category vector.

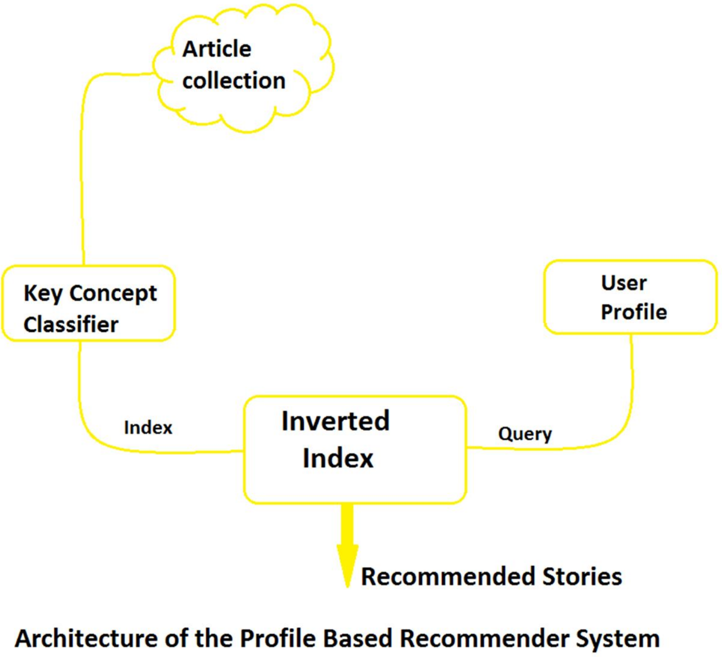


Figure 3. Architecture of the Profile-based Recommender System

Thus, for a given user j, the Personal_Wt for article i is: **Personal_Wtij =CosineSimilarity(Article Profilei, User Profilej)**

We use the same formula to find out the Personal_weight using jaccard instead cosine.

We again use the indexer to query against the index that stores the category vectors with the user's profile.

C. Hybrid News Recommendations

The hybrid recommender module combines the weights provided by each of the previous two modules to produce a recommendation based on both the articles popularity to users everywhere and the article's likely interest to the particular user. We first experimented with multiplying the two factors together:
Hybrid_Wt1ij = Popularity_Wti * Personal_Wtij
This module calculates a Hybrid_Wt by combining the two scores.

X. Results and Analysis

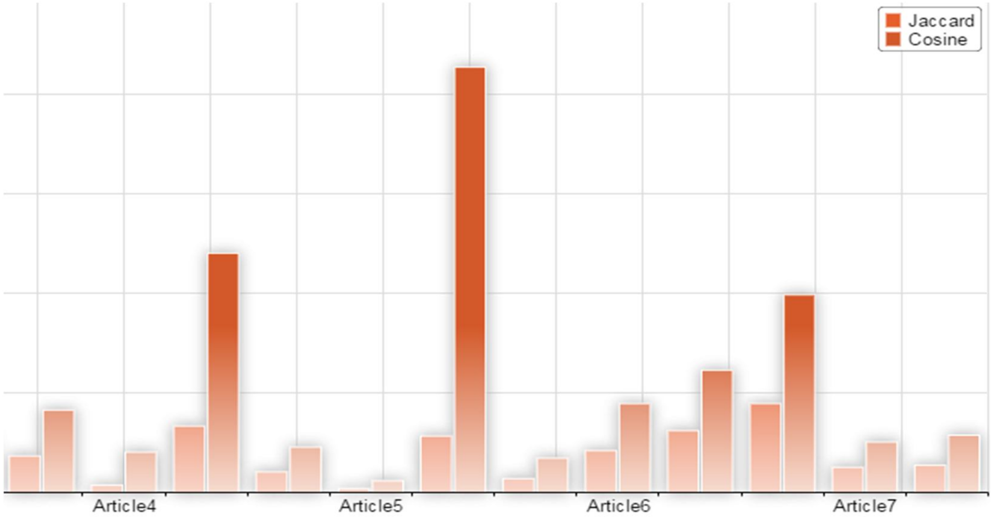
All the experiments were conducted on the same set of news articles 150 news articles collected from IBN and BBC and 10,000 tweets collected from Twitter on the same day. We collected 20 news articles for each of 5 topics (Sports, Business, Politics, Tech, and Health). We used 22 volunteer test subjects to produce the results presented here.

A. Evaluating the Jaccard coefficient against the cosine similarity function

The results of popularity and profile based recommender systems were analysed by using jaccard coefficient and the cosine function by comparing results of both and plotting a graph. We found that 83% of the time cosine function recommended articles that were most interesting to users. However, Jaccard coefficient produced results that were less accurate but not totally different. Recommendations were positive 67% of the time.

B. Evaluating the Hybrid Recommender System

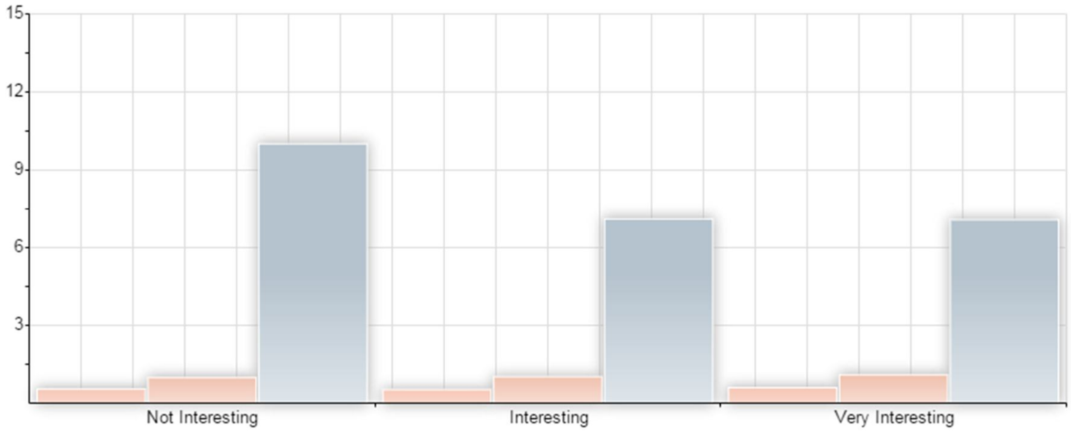
Each volunteer was presented with a web form on which they entered scores from 0 – 5 indicating their interest in each of the five categories. These category/score pairs form their user profile. We identified the top 10 articles recommended by the hybrid system and presented them to the user to be rated as very interesting, interesting, or not interesting. We presented 35 articles in a random order to the volunteers. We identified the top 20 articles recommended by the Popularity, Profile-Based and Hybrid models and presented them to the user to be rated as very interesting, interesting, or not interesting. We presented 25 articles in a random order to the volunteers.



Graph 1. Relevancy of Results compared using Jaccard & Cosine functions

C. Analyzing HybridWt

Since users only really look at the top documents, we used metrics based on the ratings of articles in the top 20 recommendations. We analyzed our results using Average rating metric. Table 1 shows the average rating of news articles in the top 20 for all three approaches. The average rating fluctuates and there is no clear trend, other than the hybrid system presents, on average, a relevant document at every rank, the personal



Graph 2. Evaluate Hybrid System by Rating Articles

system does best at showing a relevant document as the top result, and the popularity-based system is not as effective as the other two. The values can be normalized and further analysis metrics may be applied.

TABLE I. EVALUATE HYBRID SYSTEM BY RATING ARTICLES

	Not Interesting	Interesting	Very Interesting
Popularity	0.545	0.981	3.451
Personal	0.524	1.015	2.106
Hybrid	0.601	1.087	2.075

To take the rank order of relevant documents within the set of 10 recommended documents into account, we employ a cumulative rating metric that calculates, at each rank position in the top 10, the total relevance of all documents at or above that rank. We took the average of those ratings across all users.

XI. CONCLUSION

Our Proposed system implements four different techniques and draw the following conclusions from the results obtained from the experiments: Our hybrid approach outperforms the popularity and profile-based recommendations. In other words, the news articles recommended by the hybrid approach are more relevant to the user as compared to the other two strategies. The profile-based strategy provides slightly better recommendations as compared to the popularity-based strategy. Thus, we demonstrated that our hybrid approach, identifying popular articles in areas of interest to the user, is more effective than either of the two approaches alone.

FUTURE WORK

Naïve Bayes Algorithm may be used to compare the performance of system against the algorithms used. Facebook’s Trending API may be used to enhance the input of the system. Accuracy of our system maybe increased by considering temporal activity.

REFERENCES

[1] N. Jonalagedda, S. Bauch; (2013); *Personalized News Recommender System*, International Joint Conference on Web Intelligence and Intelligent Technologies, USA.

[2] M. Vanetti, E. Binaghi, E. Ferrari, B. Carminati, and M. Carullo; (2013); *A System to Filter Unwanted Messages from OSN User Walls;* IEEE Transactions On Knowledge And Data Engineering; 25.

[3] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, M. Demirbas; *Short Text Classification in Twitter to Improve Information Filtering*; USA.

[4] N. Agarwal, M. Rawat, V. Maheshwari; (2014); *Comparative analysis of jaccard coefficient and cosine similarity For web document Similarity Measure*; IJARET; 2.

[5] O. Phelan, K. McCarthy and B. Smyth; (2009); *Using Twitter to Recommend real-time Topical News*; Proceedings of the Third ACM Conference; New York, USA.

[6] A. Tuzhilin and G. Adomavicius; (2005); *Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions*; IEEE Transactions on Knowledge and Data Engineering; **734-749**; New Jersey, USA.

[7] L. Baoli, L. Qin and Y. Shiwen; (2004); *An adaptive k-nearest neighbour text categorization strategy*; Journal of ACM Transactions on Asian Language Information Processing; 215–226.