**Recommender System for Projects**

Jateen Mittal Kunal Parnami Neerav Jain Nypunya Devraj

Dept. of Information Science Dept. of Information Science Dept. of Information Science Dept. of Information Science

PESIT PESIT PESIT PESIT

Bangalore, India Bangalore, India Bangalore, India Bangalore, India [jateenmittal0994@gmail.com](mailto:jateenmittal0994@gmail.com) [kparnami1@gmail.com](mailto:kparnami1@gmail.com) [ostwalneerav@gmail.com](mailto:ostwalneerav@gmail.com) [nypunya.devraj@gmail.com](mailto:nypunya.devraj@gmail.com)

*Abstract*— **With the increased Internet penetration over the years, students find it difficult to choose which project to work on. Choosing the project is difficult not only because of indefinite options available through the Internet but also involves looking for user’s area of interest. Interest area leads to a decision for working on a project with a team. Involvement of Risk i.e. Risk Analysis and Risk Management becomes crucial based on the decision taken by the user involving the interest area.**

**Recommender systems have the effect of guiding users in a personalized way to interesting objects in a large space of possible options. These systems have become extremely common in recent years, and are applied in a variety of applications. Mostly the recommenders are meant for movies, music, news, books etc. But this paper talk is the recommender developed for suggesting projects to the User.**

Keywords— Recommender, Content Based Filterring, Collabrative Based Filterring, Vector Space Modelling, TF, IDF

# Introduction

With the development of the Internet, people are more likely to express their views and opinions on the Web. They especially students, like, dislike or comment on the articles they see on Internet, articles being related to interest area of the students. So, based on the user-data i.e. Student’s historical data on the educational websites, open source forums and on the blogs helps to generate user’s interest-data. Recommending projects based on interest data becomes much easier rather than ideally recommending things to any user.

Recommender Systems are a useful alternative to search algorithms since they help users discover items they might not have found by themselves. Recommender Systems typically produce a list of recommendations in one of the two ways – Collaborative or Content-based filtering.

Collaborative filtering approach builds a model from a user’s past behaviour as well as the similar decisions made by other users. This model is used to predict items that the user may have an interest in. Content – Based filtering approach utilize a series of discrete characteristics of an item in order to recommend additional items with the similar properties. In today’s world the two approaches doesn’t provide efficient results: namely because of the non – traditional data, fake user profiles and perspective on different items or insufficiency of the attributes of items for a better recommendation. Hybrid Recommender System, a combination of Content based and Collaborative based filtering techniques comes out to be much more efficient in most cases as produced by earlier studies. So, this work works on a Hybrid approach as using Content-based filtering technique for user-project recommendation and Collaborative-based filtering for user-user recommendation.

# methodology

## Data Set Extraction

The data set consisted of 2 phases:

Firstly, data-set was collected for the different projects. The data set for this project comes from the archive of past CS229 projects and also from the blogs in the open-source communities namely the engpapers.net.

Secondly, user-data, as a measure of historical data was collected through Google Forms presented by us to all students of our Campus fetching their detailed area of interest for recommending the students with projects and also other students.

## Processing

The project papers extracted were dealt through a series of pre-processed steps. The project papers were first converted to text file and then to word–frequency vectors. Conversion to word-frequency vectors involved removal of stop-words, lower–casing words and word-stemming. A Filter list was developed to remove all the “Content less words” from each paper for creation of a better and relevant word-frequency vector.

The problem of content-less words was exacerbated by the fact that the texts were project papers with different vocabularies than Standard English texts. For example, words such as “algorithm” and “learning” would be relevant for ordinary topic modeling. But in this case, every project applies some algorithm that attempts to learn something. So these words do not indicate anything particular about the project. Words like these are considered “content-less” in this context and were subsequently added to the stop–words list. As the pre-processing was done word-frequency vectors were created for every project paper extracted and stored for the further usage as explained in Section III.

## Analysis

After the processing was done and all the techniques as discussed in Section III were implemented a user was presented with two different set of results: user recommended to a set of projects based on the different area of interest of the user and user getting a list of user’s for each area of interest to work with in the projects provided. Results were analyzed as if the recommendations had a better probability rate according to the respective user produced by us. This analysis was not computed but only done manually and has been talked in section V.

# techniques and implementation

Multiple Techniques were applied in serial on the two data-sets and recommendations were done. These techniques involved calculation of weights on words, creation of feature vectors and mapping of vectors based on user-interest data.

In these techniques, item is referred to as content whose attributes are used in the recommender model and attribute refers to the characteristic of an item. So, item in the given context are the project papers and words in the document are the attributes.

## Content – Based Filterring

Content-based recommender works with the data that the user provides either explicitly or implicitly. Based on that data, a user profile is generated which in turn is used to make suggestions to the user. Providing more data to the engine reliability, accuracy and affectability of the engine is improved. Concepts of Term Frequency (TF) and Inverse Document Frequency (IDF) are used for a content based recommender. These help us to determine relative importance of the item.

TF is simply the frequency of a word in a document. IDF is the inverse of the document frequency among the whole corpus of documents. TF-IDF is used because TF-IDF weight negates the effect of high frequency words in determining the importance of an item. Logarithmic way is used to calculate the weights so as to dampen the effect of high-frequency words.

Using the above notation for calculation of weights values become more comparable as opposed to the original raw term frequency. Fig1 depicts a sample of calculated TF values.

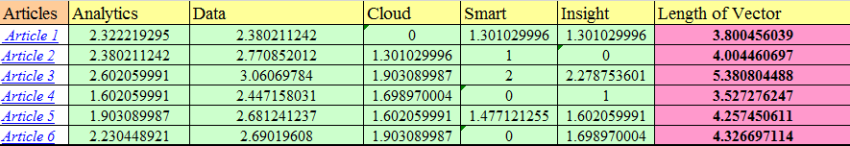


Fig 1.A set of data containing dampened TF values for 5 attributes

Inverse Document Frequency (IDF) is calculated by taking the logarithmic inverse of the document frequency among the whole corpus of documents.

Length of these project vectors, user vectors and domain vectors is calculated as the square root of the sum of the squared values. After the above processes, each term vector is normalized as by dividing the vector by the document vector length to get the normalized length.

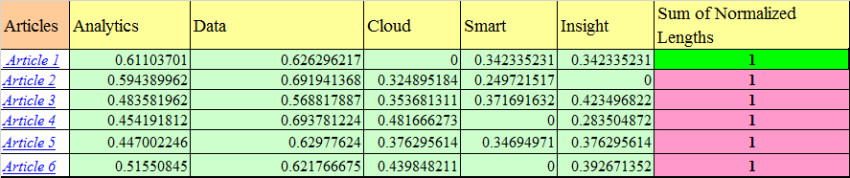


Fig 2.A set of data containing normalized vector length.

## Vector – Space Modelling

In this model, each item is stored as a vector of its attributes (which are also vectors) in a n-dimensional space and the angles between the vectors are calculated to determine the similarity between the vectors.

For our work, every item an i.e. project paper has a unique project vector consisting of attributes of the paper i.e. words appearing in the project paper. Now, the domain vectors were constructed in the similar fashion consisting of all projects belonging to a separate domain ex: Machine Learning.

Next, the user profile vectors were also created based on his/her actions on previous attributes of items and similarity between an item and a user is determined in a similar way.

Fig3 shows a 2-D representation of two users and interest fields being 2 attributes of an item so as to analyze the vector space modeling.

* Attributes being Cloud and Analytics
* Documents being M1 and M2
* User’s being U1 and U2

From the figure it is clear that document M2 is more into Analytics whereas document M1 is more into Cloud. Method of calculation of User’s likes/ dislikes/ measures is calculated by taking Cosine of the angle between the user-profile-vector (u[i]) and the document vector (d[i]). Reason for cosine values is that value of cosine will increase with the decreasing value of the angle between which signifies more similarity.

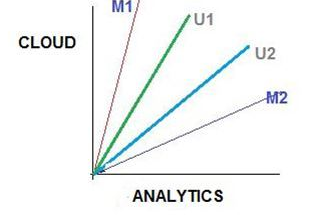


Fig 3.Example for Vector Space Modeling

## Binary Representation of User-Data

Data input by the user is taken through Google – Form. The data is in the form of only 3 values being: 1, -1, and 0. Data is interpreted as if the user likes a particular field as his/her interest area or dislikes the field or cannot decide with the provided information respectively. Fig4 represents the data taken from the user. Once the input is taken then, the vectors are generated and are normalized as depicted in equation 1:

------- (1)

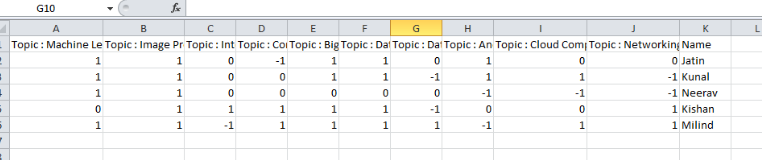


Fig 4. Snap-shot for User – data (Domain - data)

## Collbrative - Based Filterring

Collaborative-based recommender work on automatic predictions or filtering about the interests of a user by collecting preferences from many users as by collaborating. The underlining principle of collaborative filtering approach is that if a person A has the same opinion as a person B on an issue, A is more likely to have B’s opinion on a different issue x than to have the opinion on x of a person chosen randomly.

These techniques require:

* User’s active participation.
* Easy way to represent user’s interest to system.
* Algorithms to match people with similar interests.

Data clustering is a very popular way for approaching towards this filtering. So, in the given work collaborative filtering has been used in the following way: profile vectors are generated with the help of Vector Space Modeling as explained above. Now user-user recommendation is done in two ways:

* Vectors were mapped to domain-vectors as explained in Section III A. So, every user vector was linked to some domain-vectors. This way users belonging to same domain-vector were recommended to each other and their interests were shared. All user profile vectors are generated as shown in Fig5. (Domain data) .
* Euclidean Distance was calculated for each user vector for each domain. This was calculated for single user with every user (for all domains) as depicted in equation2 :

------- (2)

Results are shown in Section IV.

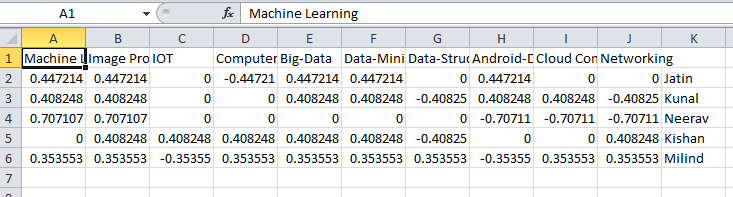


Fig 5.Snap-shots of user-profile vectors (Domain Data)

## Clusterring

Cluster Analysis or Clustering is the task of grouping a set of objects in such a way that objects in the same group, known as a cluster are more familiar to each other than to those in other groups. Cluster analysis is a iterative process and not an automatic task. Appropriate clustering algorithm and parameter setting like the distance function etc. depend on the individual data-set and intended use of the results. In general, the work done defines two types of clustering techniques: Hierarchical Agglomerative Clustering and k-means clustering.

K-means clustering being a method of vector quantization, aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype for the cluster. Some pros and cons for k-means clustering are:

* Time Complexity: Takes linear time in the number of objects i.e. O (n) where n is the number of data objects, whereas the Hierarchical Clustering is quadratic i.e. O (n^2).
* Shaping of clusters: Favors hyper-spherical clusters or globular. Non-spherical clusters are not advised to work with this clustering.
* Number of Clusters: Needs the number of clusters to be defined whereas the hierarchical clustering doesn’t require number being specified.
* Random initial clustering generally results in inconsistent final clusters as the final results would always vary.

Based on our work, we chose hierarchical clustering namely, HAC (Hierarchical Agglomerative Clustering) as work concentrated more on non-spherical clusters and not on the time complexity.

HAC starts by initializing each sample in its own cluster. On every iteration process, clusters that are most similar are merged together. This is a “bottom-up” approach: each observation starts in its own cluster and pair of clusters are merged as one moves up the hierarchy. Upon successive iterations, threshold for merging the clusters is relaxed. So, more similar clusters are merged first, less similar clusters are merged later and dissimilar clusters are left unmerged. Similarity of the clusters is found using the Cosine Similarity Rule depicted by (3):

----- (3)

where A and B are the two word frequency vectors as discussed in Section II. The analysis on the word-frequency vectors dot-product was only done on the top 20 words of every project paper as those tend to be the most frequent and relevant words of the paper. This was also done for having a low word-vocabulary for each project paper and also consists of only the most relevant words of the paper. Fig6 depicts the clusters made during the work process. Clustering results also depict the most common projects and unique projects in the respective fields.

Fig 6.A snap-shot of the Clusters formed using HAC

# RESULTS AND DISCUSSION

The result of our project was to present a recommender system of projects to user i.e. student group and also present a user-user recommendation based on the domains mentioned in our overall work namely: Machine Learning, Internet of Things, Big–Data, Cloud Computing, Data-Structures and Algorithms, Image Processing, Computer Security and Computer Networking. Fig7 and Fig8 describe the user-input. Both ways require a value in [1, -1, 0] referring to liking, disliking or neutral respectively. Around 150 student’s data-set was collected and were processed on 1000 project papers based on different domains.

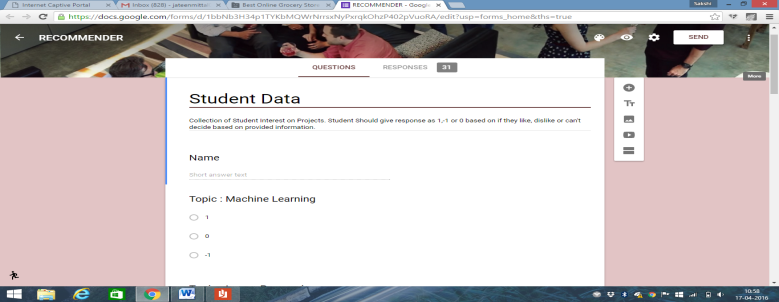


Fig 7.Online User-data Google Form

Fig 8.Offline User-data Graphical User Interface

Recommendation includes domain-wise project recommendation and the users specifically compatible with the concerned user. Fig9 represents the recommended projects and the users. Every project-paper name being a hyper-link to the project-paper content. Fig10 represents the corresponding page.

Fig 9.Recommended users and domain-specific project papers

Fig 10.Recommended project-paper content

Recommendations were provided to a set of students. Measurement of accuracy was a tougher task as recommendations could be useful and not too. One limitation of the clustering algorithm used was the time complexity as on 1000 project papers to be separated in clusters a lot of time was needed. Content-based and Collaborative-based filtering techniques perform well with huge historical data content for a specified user which was in a limit for this work. Results on only domain-data were not that accurate and we provided work for user-user recommendation using sub-domains as explained below.

Fig11 suggests the user data taken as the input from the user based on sub-domains and same strategy as explained in Section II is applied here. The user sub-domain specific vectors are generated. Fig 12 depicts the sub-domain vectors.

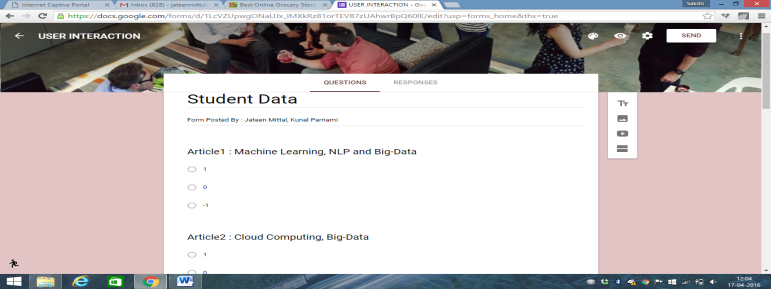
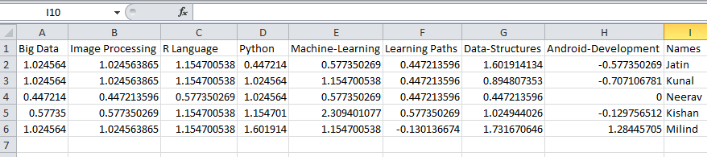


Fig 11.Sub-domain user data Google form



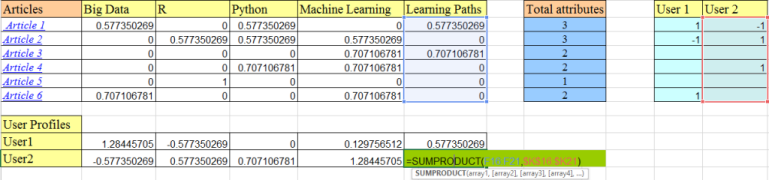


Fig 12.Snap-shots of user-profile vectors (Sub – Domain Data)

Now, every user had differed interests in sub-domains so the recommendation was done domain-specific as specific user was given a set of users sub-domain wise and not on an overall base. Fig13 depicts the differed area of interest of the user. Fig14 depicts the users recommended.

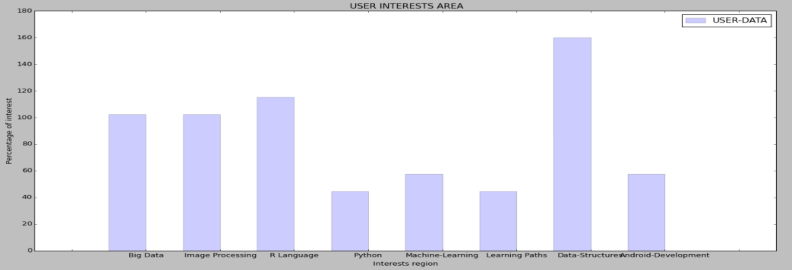


Fig 13.User Interest areas

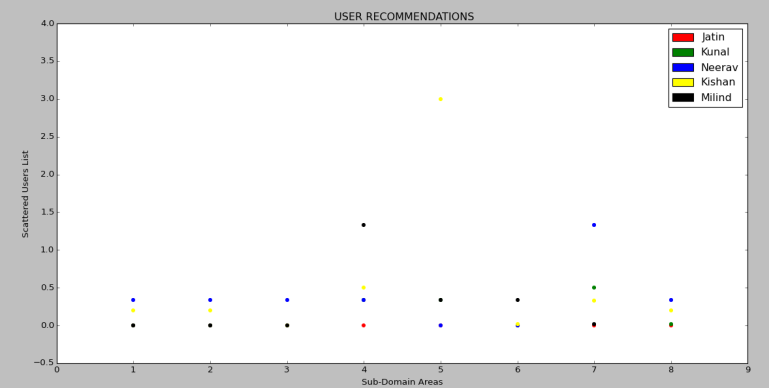


Fig 14.Sub-Domain specific recommended User

# V. CONCLUSION AND FUTURE WORKS

Recommenders are a bit separate when compared to the search engines. Recommenders work on the users preferences through the historical data, suggesting items based on the mapped attributes. Our work confined to be recommending project papers which were domain-specific and also recommending user-list from the data-set which were compatible to work with the corresponding user. Our work also focused on sub-domain area as explained in Section IV but that could only provide the user-user recommendations and not the projects for the users.

Our future work would be to concentrate upon recommending the user with sub-domain specific projects also. We would also like to extend our work in the following areas:

* Recommending appropriate courses to opt for in the future.
* Recommending appropriate teachers to work with and vice-versa.
* Recommending foreign universities for Post-graduation.

# ACKNOWLEDGMENT

We would like to thank Dr. Shylaja S.S. Professor & Head of Department ISE, PES Institute of Technology for her immense encouragement and support throughout the project. We would also like to thank our Information Science Dept. for their kind support.

# REFERENCES

[1] Lops, Pasquale, Marco De Gemmis, and Giovanni Semeraro. "Content-based recommender systems: State of the art and trends." *Recommender systems handbook*. Springer US, 2011. 73-105.

[2] Poon, Andrew. "What Project Should I Choose?."

[3] Resnick, Paul, and Hal R. Varian. "Recommender systems."*Communications of the ACM* 40.3 (1997): 56-58.

[4] Hosken, Benjamin E. "Automated content and collaboration-based system and methods for determining and providing content recommendations." U.S. Patent No. 6,438,579. 20 Aug. 2002.