

# CONTEXTUAL MODEL OF RECOMMENDING RESOURCES ON AN ACADEMIC NETWORKING PORTAL

Anoop Kumar Pandey, Amit Kumar and Balaji Rajendran

Centre for Development of Advanced Computing,  
Electronics City, Bangalore  
{anoop, amit, balaji}@cdac.in

## **ABSTRACT**

*Artificial Intelligence techniques have been instrumental in helping users to handle the large amount of information on the Internet. The idea of recommendation systems, custom search engines, and intelligent software has been widely accepted among users who seek assistance in searching, sorting, classifying, filtering and sharing this vast quantity of information. In this paper, we present a contextual model of recommendation engine which keeping in mind the context and activities of a user, recommends resources in an academic networking portal. The proposed method uses the implicit method of feedback and the concepts relationship hierarchy to determine the similarity between a user and the resources in the portal. The proposed algorithm has been tested on an academic networking portal and the results are convincing.*

## **KEYWORDS**

*Contextual Model, Recommendation Engine, Academic Networking*

## **1. INTRODUCTION**

The development of World Wide Web has contributed to a ever growing global society of internet users across the world. With exponential increase in users and thereby formation of avalanche of information, It has been quite difficult to segregate the information we actually need from amongst the mountain of web information. With need of a system for searching, sorting, filtering and sharing relevant information out of the information overload, several personalized search engines, recommender systems and intelligent software have been developed. A combination of modeling the preferences of particular users, building content models, and modeling social patterns in intelligent agents (Maes 1994) would provide users with a means for managing information in a rational way, thus helping them to overcome the overwhelming information content.

Recommender Systems or Recommendation Systems are quite common and used in several applications. They primarily identify interesting information for a given user such as their likely taste for a type of music, book, or movie or the most likely kind of people or groups whom they would want to be associated with. , The common approaches adopted by these systems are: Content-based filtering that use characteristics of an item or Collaborative filtering that rely on a use's profile and their social network [4]. Some examples are:

- When viewing a product on Amazon, the store will recommend additional items based on a matrix of what other shoppers bought along with the currently selected item.[8]
- Friends suggestion on Facebook.[6]
- People, group, job suggestion on LinkedIn.[7]
- Follow suggestions on Twitter.[9]



Figure 1. Examples of various recommendations from social media

In a relatively short time, several recommender agents have been developed and there is a wide variety of such systems, all of which exploit a particular set of AI techniques. Recommender systems typically produce a list of recommendations in one of the three main information filtering methods: demographic, content based and collaborative [1]. Demographic filtering approaches use descriptions of people to learn the relationship between a single item and the type of people who like it. Content-based filtering approaches recommend items for the user based on the descriptions of previously evaluated items. The collaborative filtering technique matches people with similar interests and then makes recommendations on this basis. Another approach to overcome the shortcomings of collaborative and content based filtering is Hybrid systems that exploit features of content-based and collaborative filtering, since they will almost certainly prove to be complementary. Each type of system has its own strengths and weaknesses.

The proposed algorithm of recommender system exploits user profile, implicit feedback and explicitly cited interests. The algorithm uses a concept hierarchy, that holds all the concepts used in the domain, which will be used to determine the context of the user and the resources to be recommended, having similarity with user's interest.

The rest of this paper is organized as follows. Section 2 discusses some of the recommendation techniques used in recommender agents in general and in academic environments. Section 3 details the proposed recommendation system for academic portals. Section 4 discusses the impact of our proposed algorithm and results and section 5 concludes the paper.

## 2. RELATED WORKS

In this section we briefly present some of the research literature review related to recommender systems in general, academic recommendation system, and evaluation of recommender systems.

Recommender systems typically produce a list of recommendations using one of the four main information filtering methods [10]: demographic filtering, collaborative filtering, content-based filtering, and hybrid methods. First, demographic filtering approaches use descriptions of people to learn the relationship between a single item and the type of people who like it. Collaborative filtering uses user-item rating matrix for predicting unseen preferences [11, 12]. It can be categorized into memory-based CF, which contains the whole matrix on memory, and model-based CF, that involves building a model for estimation [13]. The most effective memory-based algorithms known so far is item-based CF [19]. Matrix factorization, a kind of model-based approach, known to be most efficient and accurate, especially after this approach won the Netflix prize in 2009, is being used in many recent works [14, 16, 18, 17, 20]. Content-based filtering, on the other hand, recommends items based on their characteristics as well as based on previous evaluations of a user [10]. Pazzani et al. [15] studied this approach in depth, including how to build user and item profiles. Hybrid approach, tries to combine both collaborative and content-based recommendation. Koren [21] suggested effectively combining rating information and user, item profiles for more accurate recommendation.

In [2], we used the contextual method of detecting relevant content for a user using a concept hierarchy and concept extraction method. In [22] the authors had defined similarity between two research papers based on the text similarity between them and then proposed their recommender system developed using collaborative filtering methods.

AACORN [23], a course recommendation system that uses the course histories as the basis for course advising. By reusing the experience embodied in historical student's transcripts, AACORN can make reasonable suggestions with a limited amount of domain knowledge. The system uses the edit distance between two course histories, as well as other heuristics to determine the similarity between course histories.

## 3. CONTEXTUAL METHOD OF RECOMMENDATION

The Contextual Model of Recommendation System is built into a system that recommends different resources viz. institutions, courses, events, projects, topic, and persons to 'Follow' and get updates, to the various users of our academic networking portal. The users could be. students, faculties, researchers, representatives of institutions etc, whose sole aim is to discover and share information about their subjects of interest, information about institutions, courses, events, projects, questionnaires, and such related activities for learning and research. In this pursuit, they would like to receive regular updates regarding their favourite courses, persons, topics and important notifications regarding events like seminars, conference and workshop etc. For this purpose they "Follow" particular resource in which they are interested in. While there are thousands of resources on the academic portal, it is a tedious job to search out resources of own interest and check the latest updates. The criticality of recommendation system will guide the user to sift through such large information by recommending them the appropriate resources based on their profile, their explicit and implicit feedback.

We start by defining the common terms that are used in our system, following it with the recommendation algorithm.

### 3.1. Definitions

**Resource:** All components of our academic networking website are treated as resources. Examples may include institutes, faculties, courses, events, projects, web links, topics etc. Our recommendation system recommends these resources to the user to ‘Follow’.

**Concepts & Relationships:** Concept in general, could refer to all the terminologies and vocabulary of a particular domain which is used to describe it. The definition of concepts and relationship between the concepts is typically captured in the form of ontology for a domain. We use a simple structure called concept tree that captures two kinds of relationships: is-a and is-in. For instance in the statement, “Java is an Object-oriented language”, the concepts ‘Java’ and ‘object-oriented language’ are captured using ‘is-a’ relationship. The is-in relationship is captured as a composition tree. For instance, “Object-oriented Languages” is contained within the concept of “Programming Languages”.

N.B.: All resources are concepts.

**Concept Extraction:** It is the process of extracting the concepts in a given piece of text, by comparing the main terms with the concept tree.

**Co-occurrence Pattern:** It defines the pattern and frequency of a concept occurring with another concept within a defined window. E.g. Sachin and Cricket co-occurs with each other often.

**Follow List:** Resources, a user is already watching.

**Posts:** Informative content posted on a resource’s wall/timeline.

**Cosine Similarity:** Cosine similarity is a measure of similarity between two vectors by measuring the cosine of the angle between them. The cosine of the angle between two vectors thus determines whether two vectors are pointing in roughly the same direction.

### 3.2. Steps towards Algorithm

In the proposed recommendation algorithm, we use three parameters – User’s Profile, User’s Follow List, and User’s Navigation History as input to our algorithm for determining the likely list of resources that a user would be interested in. A factor of weight has been given to each of these parameters so that the results are not skewed to a single parameter. They are assigned as high for explicit and low for implicit parameters.

The user profile will explicitly contain the user’s skill-set, experience and areas of interest. The user’s follow list indicates the resources that the user is interested in getting the recent updates of that particular resource. The user’s navigation history is an implicit parameter that reveals the interests of the user.

When we have a set of resources, the user is interested in, concepts are extracted from them through contextual method and then we try to find out if some other resource also belong to or matches with the same pool of resources. If we find resources in that pool which the user is currently unaware of, then they are recommended to the user. The recommendations are generated dynamically – i.e., on the live and changing data.

### 3.3. Algorithm for recommendation generation

- ❖ Assumptions: a) Presence of concept extraction algorithm.
  - b) Presence of concept and containment trees.
  - c) Resource R has been properly modeled.
  - d) Presence of Co-occurrence table based on descriptions of concepts. Input: User's Profile, User's Follow List, User's Navigation Pattern. Output: Set of recommendation comprising 3 resources.
- ❖ Algorithm:
  - Extract the concepts from User Profile Columns: "Skill-Set", "Experience", "Area of Interest". Store it in an associative array  $C_{UP}$  with key as concept-id and value as term frequency of occurrence.
  - Extract the concepts from user's Follow list and store it in an array  $C_{FL}$  with corresponding term frequency.
  - Extract the concepts from user's navigation list and store it in an array  $C_{NL}$  with corresponding term frequency.
  - Combine the array in one array with different weightages.  

$$C = 3 * C_{UP} + 2 * C_{FL} + 0.25 * C_{NL}$$
  - Expand the concept-list C based on co-occurrence table and update the frequency as well.  

$$C_{EX} = C \cup (\text{Concepts co-occurring with } C)$$
  - Create Weight vector of length 'n', where n is the total no of concepts in  $C_{EX}$ .  

$$\vec{w} = [w_i]$$
 where  $w_i$  is the weightage or term frequency of concepts within  $C_{EX}$
  - Represent each resource as a vector  $\vec{R}$ , which the user is not watching, in terms of the concepts in  $C_{EX}$  and their corresponding term frequency in the resource description and the posts on the resource's wall in last one year.
  - For each resource calculate cosine similarity (Cosim) with concept vector  $\vec{w}$  representing the user interest as:  

$$\text{Cosim}(\vec{R}_i) = \frac{\vec{w} \cdot \vec{R}_i}{|\vec{w}| |\vec{R}_i|}$$
  - Arrange the resources in decreasing order of cosine similarity value.
  - Filter out top 'u' resources as per following rule:  

$$\sum_{i=0}^u \text{Cosim}(\vec{R}_i) = 0.8 * \sum_{i=0}^m \text{Cosim}(\vec{R}_i)$$
  - Find out top u resources and recommend any 3 out of them.

## 4. IMPLEMENTATION AND RESULTS

An academic networking portal [27], with interactive features for its users was used for validating our approach. The website features the recommendation system with facility to hover over recommendation to watch information, click to navigate to the resource page and Follow button to accept recommendation. Figure 2 and 3 illustrate the recommendation and hover function. We captured each action (hover, click and Follow) corresponding to a particular recommended resource while user was presented with three recommendations on a page. These actions were treated as metrics for evaluation in addition to manual inspection, where the recommended resources were checked to whether having the similarity with user's interest and trend.



Figure 2. Screenshot of recommendation of resources (institutions and person)

**Welcome Mr. Anoop Kumar Pandey**  
 Profile Views: 96 Followers: 19  
 Last Login at: 08-11-2012 07:39:20 From: 180.149.48.142  
 Profile Completeness: 78 %  
 Edit Profile View Profile as others see it

**Professional Summary**

**Summary of Experience:** Staff Scientist at CDAC, E-City, Bangalore RedHat Certified Engineer EMC Proven Information Storage Associate  
**Domain Expertise:** Database, Theoretical Computer Science, Storage Technologies  
**Skill Set:** Data Structures And Algorithms

**Education Qualification**

Qualification:	PhD
Subject:	CSE
Specialization:	Cognition, Psychology, IT
Year:	2016
Qualification:	DTech

**Indian Institute of Technology Madras Chennai**  
 20 Followers  
 1 Seekha member

**You May Like to Follow**

- CDAC Follow +
- PROJECT Follow +
- INDIAN INSTITUTE OF TECHNOLOGY MADRAS CHENNAI Follow +

**recently visited**

- IIT Madras
- MIT-B

Figure 3. Examples of various recommendations with hover effect.

In manual inspection, the results were pretty impressive with recommended resources amounting to 85-90% similarity with user's interest. The test was conducted with 70 user's profile matching with recommended resources.

The recommendation system was also tested from user's perspective and in terms of effectiveness/usefulness of the recommendation. Corresponding to a user, for each set of recommendations, user's actions viz. 'Click', 'Hover', and 'Watch' were captured. The data collected was analysed to check how recommended resources had been considered useful. Table 1 describes the total recommendations received by each user and the summary of actions taken by the users:

Table 1 Statistics of various actions to check effectiveness of the proposed solution.

User/Action	Total Recommendations	Hover	Click	Watch
U1	15	14(93.3%)	12(80%)	9(60%)
U2	24	21(87.5%)	19(79.2%)	17(70.8%)
U3	18	16(88.9%)	13(72.2%)	11(61.1%)
U4	12	12(100%)	11(91.7%)	9(75%)
U5	33	31(93.9%)	28(84.8%)	24(72.7%)
U6	9	9(100%)	9(100%)	6(66.7%)
-				
-				
Total	966	907(93.9%)	788(81.6%)	693(71.7%)

The results were quite impressive as majority of the presented recommendations 94% were hovered for information display, 82% were clicked to view full information and 72% were accepted for following. Considering the action taken by the user in terms of their „Following“ a resource, the results are quite impressive.

## 5. CONCLUSIONS

In this paper, we have presented a Contextual Model of Resource Recommendation System for academic portals, which recommends different resources viz. Institution, course, project, topic, event and academician to the users of the portal. The proposed system determines the concepts and cosine similarity between them with understanding of context and identifies potential resources to be recommended to the user based on user's profile and their explicit as well as implicit feedbacks. Evaluation results show the utility of our proposed system. We made the assumption that the resources in the system have been well defined and modeled. We also assumed the existence of ontology in terms of concept tree with simple relationships that defines the domain in which the academic portal operates. This methodology could be easily extended to other domain-specific vertical portals, though the challenge would be in extending it to a generic and interactive information system.

## REFERENCES

- [1] Miquel Montaner, Beatriz Lopez and Josep Lluís De La Rosa: A Taxonomy of Recommender Agents on the Internet Artificial Intelligence Review 19: 285–330, 2003.
- [2] Balaji Rajendran and Anoop Kumar Pandey, “Contextual Strategies for Detecting Spam in Academic Portals”, The 2nd International Conference on Computer Science and Information Technology (CCSIT-2012), Royal Orchid Central, Bangalore, January 2-4, 2012.
- [3] Jennings, A. & Higuchi, H. (1993). A User Model Neural Network for a Personal News Service. User Modeling and User-Adapted Interaction 3: 1–25.
- [4] Recommender Systems, [http://en.wikipedia.org/wiki/Recommender\\_system](http://en.wikipedia.org/wiki/Recommender_system)
- [5] Morita, M. & Shinoda, Y. (1994). Information Filtering Based on User Behaviour Analysis and Best Match Text Retrieval. In Proceedings of SIGIR'94, 272–81. Dublin, Ireland: Springer-Verlag.
- [6] Facebook, <http://www.facebook.com>
- [7] LinkedIn, <http://www.linkedin.com>
- [8] Amazon, <http://www.amazon.com>
- [9] Twitter, <http://www.twitter.com>

- [10] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich. Recommender Systems. An Introduction. Cambridge, 2011.
- [11] X. Su and T. M. Khoshgoftaar. A survey of collaborative filtering techniques. *Adv. in Artif. Intell.*, 2009:4:2.4:2, January 2009.
- [12] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734-749, 2005.
- [13] J. S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. pages 43-52. Morgan Kaufmann, 1998.
- [14] A. Paterek. Improving regularized singular value decomposition for collaborative filtering. *Statistics*, 2007:2.5.
- [15] M. Pazzani and D. Billsus. Content-based recommendation systems. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, *The Adaptive Web*, volume 4321 of *Lecture Notes in Computer Science*, pages 325-341. Springer Berlin, Heidelberg, 2007.
- [16] J. D. M. Rennie and N. Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd international conference on Machine learning, ICML '05*, pages 713-719, New York, NY, USA, 2005. ACM.
- [17] L. D. D. Algorithms for nonnegative matrix factorization. *Advances in Neural Information Processing Systems*, 13:556-562, 2001.
- [18] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. 2008.
- [19] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. Pages 285-295, 2001.
- [20] K. Yu, S. Zhu, J. Lafferty, and Y. Gong. Fast nonparametric matrix factorization for large-scale collaborative filtering. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, SIGIR '09*, pages 211-218, New York, NY, USA, 2009. ACM.
- [21] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '08*, pages 426-434, New York, NY, USA, 2008. ACM.
- [22] Joonseok Lee, Kisung Lee, Jennifer G. Kim: Personalized Academic Research Paper Recommendation System, <http://www.cc.gatech.edu/~jkim693/projects/recommendation.pdf>, Georgia Institute of Technology, Atlanta, GA.
- [23] JJ Sandvig, Robin Burke: AACORN: A CBR Recommender for Academic Advising, <http://facweb.cti.depaul.edu/research/techreports/TR05-015.doc>
- [24] K. Sugiyama and M.-Y. Kan. Scholarly paper recommendation via user's recent research interests. In *Proceedings of the 10th annual joint conference on Digital libraries, JCDL '10*, pages 29-38, New York, NY, USA, 2010. ACM.
- [25] Tang and J. Zhang. A discriminative approach to topic-based citation recommendation. In T. Theeramunkong, B. Kijssirikul, N. Cercone, and T.-B. Ho, editors, *Advances in Knowledge Discovery and Data Mining*, volume 5476 of *Lecture Notes in Computer Science*, pages 572-579. Springer Berlin / Heidelberg, 2009.
- [26] K. Yu, S. Zhu, J. Lafferty, and Y. Gong. Fast nonparametric matrix factorization for large-scale collaborative filtering. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, SIGIR '09*, pages 211-218, New York, NY, USA, 2009. ACM.
- [27] Seekha, <http://www.seekha.in>

**Authors****Anoop Kumar Pandey**

Mr Anoop Kumar Pandey holds a B. Tech degree in CSE graduate (2010) from Shri Mata Vaishno Devi University, Jammu. He was a Chancellor Gold Medalist. He joined C-DAC after the college and currently working here as a Technical Officer. He is also pursuing MS by Research from IIIT Bangalore, India.

**Amit Kumar**

Mr Amit Kumar completed his B.Tech in CSE, from Biju Patnaik University of Technology (BPUT), Orissa. He currently works as a Project Engineer at CDAC since 2011. His current work of area is web application developmentt and Disaster Recovery site management and maintenance. His interest area includes web-design, database management system, and information security.

**Dr Balaji Rajendran**

Dr Balaji Rajendran holds a Ph.D degree from Madurai Kamaraj University and is currently working as a Senior Technical officer at CDAC Bangalore. He has 11+ years of experience in teaching and R&D. His interest area and domain expertise include information security, web information systems, social media and intelligence.

