An Efficient and Adaptive pattern for Online-Social Recommendation Systems: Implementation through hybrid intelligence metaphor

Gaurav Agarwal[1], Dr. Himanshu Bahuguna[2] and Dr. Ajay Agarwal[3]

[1]*Research Scholar, Uttarakhand Technical University, Dehradun, (U.K.), INDIA*
[2]*Professor, Shivalik College of Engineering, Dehradun, (U.K.), INDIA*
[3]*Professor, Krishna Institute of Engineering & Technology, Ghaziabad, (U.P.), INDIA*

Abstract

With the explosive growth of online social network, people tend to express opinions and obtain information in social network. Due to the overwhelming amount of data in social network, users resort to recommendation system to find appropriate services or items. Users may not have the time or knowledge to personally evaluate the options which are available on social network platform. Recommendation systems or recommender systems present themselves as a practical answer to endless options available online. Recommendation systems changed the way inanimate websites communicate with their users, rather than providing a static experience in which users search for and potentially buy products, recommender systems increase interaction to provide a richer experience. Recommender systems identify recommendations autonomously for individual users based on past purchases and searches, and on other users' behaviour. The concept of Recommender systems is basically related to information retrieval and filtering. There are various types of filtering techniques associated with recommender systems such as: Collaborative filtering, Content based filtering, Hybridization Strategies. The proposed work discusses the problems faced by these filtering techniques and evaluation criteria on the basis of which various algorithms made for recommendation purpose can be compared. Finally a composite news recommendation system model is proposed.

Keywords: Recommender Systems, Collaborative filtering, Content based filtering, Hybridization Strategies, Evaluation Criteria, News Recommendation, Decision Support System.

Introduction

With the rapid development of online social network (OSN) over the decade, more and more people become active in interacting with each other in OSN and therefore enormous data are being generated every day. The overwhelming amount of data makes it difficult for users to find appropriate services or items. **Meanwhile, users in OSN do not know exactly what they want or need in most cases.** As a result, recommendation plays an increasingly important role in helping people find what they are interested in. There exist three kinds of classical recommendation methods: collaborative filtering, content based filtering and hybrid approach [1], [2]. **Collaborative filtering** arrives at a recommendation that's based on a model of prior user behaviour. For example, suppose you're building a website to recommend blogs. By using the information from many users who subscribe to and read blogs, you can group those users based on their preferences. In the **Table 1**, a set of blogs forms the rows, and the columns define the users. The intersection of blog and user contains the number of articles read by that user of that blog. By clustering the users based on their reading habits (for example, by using a *nearest-neighbour* algorithm), and Similarities, Recommendation System recommend the blog such as:

Blogs	Marc	Megan	Elise	Jill
Linux	13	3	11	-
OpenSource	10	-	-	3
Cloud Computing	6	1	9	-
Java Technology	-	6	-	9
Agile	-	7	1	8
	Articles read per user			
Cluster	1	2	1	2

Table 1: A set of Blogs and Users

As per Table 1, now one can identify some differences within each cluster and make meaningful recommendations. In Cluster 1, Marc reads 10 open source blog articles, and Elise read none; Elise reads one agile blog, and Marc reads none. One recommendation for Elise is the open source blog. No recommendations can be made for Marc because the small difference between him and Elise in agile blog reads would likely be filtered away. In Cluster 2, Jill reads three open source blogs, and Elise reads none; Elise reads 11 Linux blogs, and Jill reads none. Cluster 2, then carries a pair of recommendations: the Linux blog for Jill and the open source blog for Elise.

Content-based filtering constructs a recommendation on the basis of a user's behaviour. If a user commonly reads articles about Linux or is likely to leave comments on blogs about software engineering, content-based filtering can use this history to identify and recommend similar content [1], [2]. Returning to **Table 1**, focus on the user Elise. If you use a blog ranking that specifies that users who read about Linux might also enjoy reading about open source and cloud computing, you can easily recommend — on the basis of her current reading habits — that Elise read about open source. This approach, illustrated in **Table** 2,

Blogs	Elise	Similar content
Linux	11	Linux
OpenSource	-	OpenSource
Cloud Computing	9	Cloud Computing
Java Technology	-	Ranked blogs
Agile	1	
	Articles read per user	

Table 2: Recommendation table basis of User's behaviour

Hybrid approaches that combine collaborative and content-based filtering are also increasing the efficiency (and complexity) of recommender systems. A simple example of a hybrid system could use the approaches shown in Table 1 and Table 2. Incorporating the results of collaborative and content-based filtering creates the potential for a more accurate recommendation. The hybrid approach could also be used to address collaborative filtering that starts with sparse data - known as cold start - by enabling the results to be weighted initially toward content-based filtering, then shifting the weight toward collaborative filtering as the available user data set matures [1], [2], [3]. Several studies empirically compare the performance of the hybrid with the pure collaborative and content-based methods and demonstrate that the hybrid methods can provide more accurate recommendations than pure approaches. These methods can also be used to overcome some of the common problems in recommender systems such as cold start and the sparsity problem**.** Collaborative filtering approaches often suffer from three problems: cold start, scalability, and sparsity [2].

Cold start: These systems often require a large amount of existing data on a user in order to make accurate recommendations.
Scalability: In many of the environments in which these systems make recommendations, there are millions of users and products. **Thus, a large amount of computation power is often necessary to calculate recommendations.**
Sparsity: The number of items sold on major e-commerce sites is extremely large. The most active users will only have rated a small subset of the overall database. Thus, even the most popular items have very few ratings.

Proposed Model

Before discussing deeply the concept of Recommendation model let us see how recommendations are generated for any naive recommender system. First and foremost part is Data Collection. The most common method for Data collection in Collaborative Filtering is explicit feedback. Explicit feedback comes from the ratings i.e. the user rates the item on a scale of one to five or as specified by the designers. The two basic entities in the system are represented in terms of users and items. The below **Table 3** gives rating for user **U1** who has provided ratings on items I1 and I4.

Items	I1	I2	I3	I4
Rating	2	-	-	4

Table 3: Ratings given by User **U1** for various items

Similarly the below **Table 4** gives rating for user **U2** who has given ratings on I2 and I4.

Items	I1	I2	I3	I4
Rating	-	1	-	5

Table 4: Ratings given by User **U2** for various items

On the basis of given user's preferences in Table 3 and Table 4, the user – item ratings matrix is calculated in the **Table 5** :

	Items			
Users	I1	I2	I3	I4
U1	2	-	-	4
U2	-	1	-	5

TABLE 5: User-Items Ratings Matrix

After data normalization, ranking is finally calculated. Items having higher ratings are given higher ranks and finally recommended to the user.

The Hypothesis which works behind this proposed Online Social Recommendation System is "if news is shared across all the 'Social' vectors (like Twitter, Facebook, and RSS Feeds) of news channels, it will have a higher rating." The idea is shown in Figure 1.

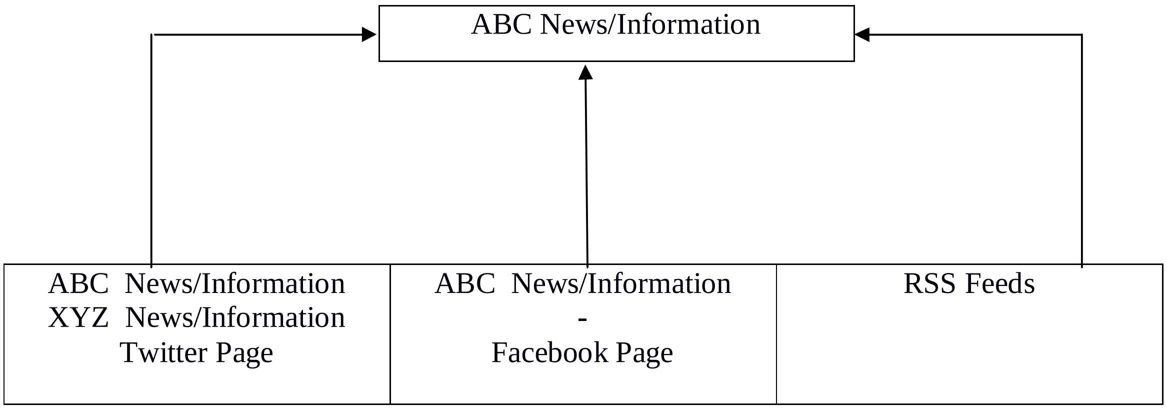

Figure 1: Composite Source News Recommendation

For example, take 2 News channels namely ABC News and PQR News, each of them have their social 'handle' pages i.e. Twitter, Facebook, and RSS Feeds. If news is available on all the social vectors of ABC news then it will get a rating of 3 as shown in the Figure 2.

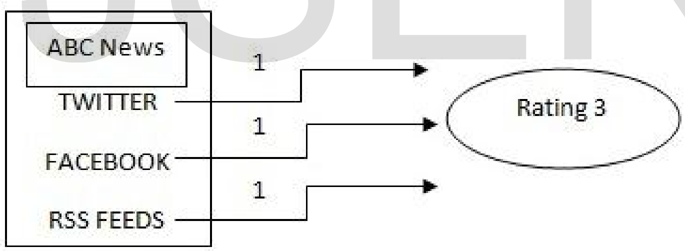

Figure 2: Calculation of Rating for ABC News

For the PQR channel news is shared across only twitter it will have a rating of value 1. The idea shown in the Figure 3.

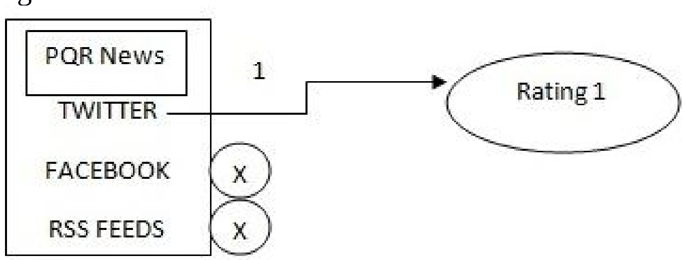

Figure 3: Calculation of Rating for PQR News

Our methodology is to gather data from all the major social vectors of news channels and generate the rankings as per the hypothesis with the recommendation engine. User preferences and keywords set are provided as feedback to the system. This results in recommendation on highly ranked news on the dashboard. The idea is shown in the Figure 4.

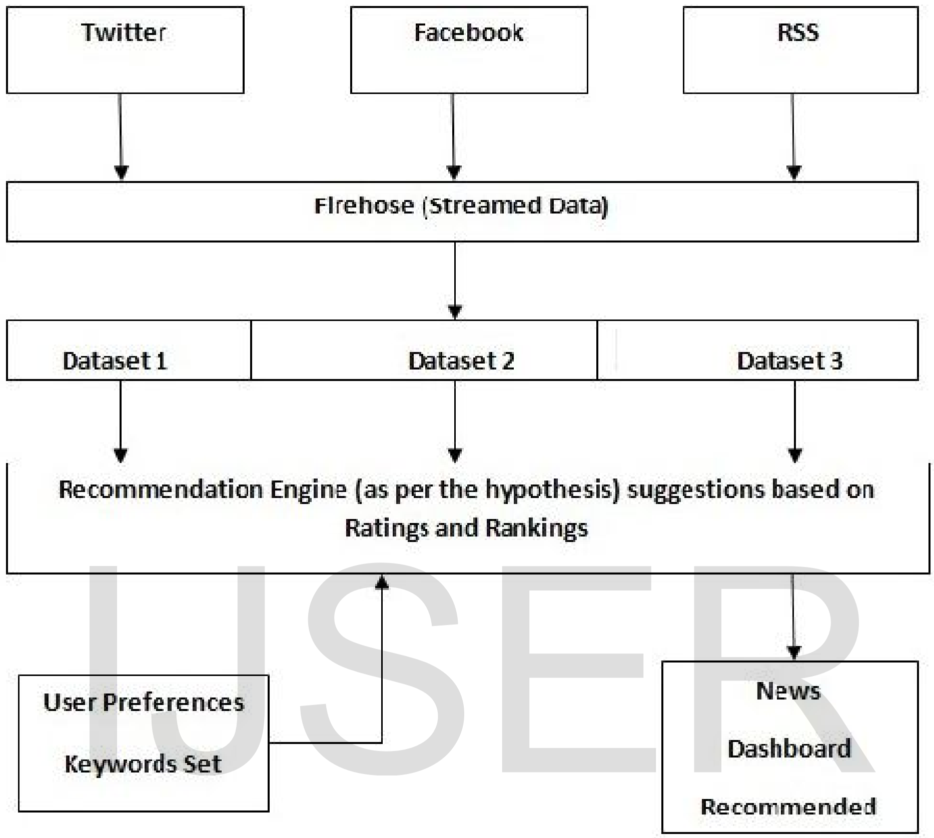

Figure 4: Workflow of Proposed News Recommendation System Based on Composite Data Source

The workflow of proposed News Recommendation System works on the diversity criteria. We looked for more diversified approach while generating recommendations for our system.

Conclusion

A number of application areas based on recommender systems require further investigation. We have proposed one system for news recommendation which is closely related to Social Network Recommender Systems. The ultimate aim is to come up with solutions to use huge amount of information available on the web for the betterment of recommender systems while keeping in mind issues like quality, privacy and security.

References

[1].Wikipedia, "Recommender System" [online] available at: https://en.wikipedia.org/wiki/Recommender_system Hybrid_Recommender_Systems.

[2]. Ruchika, Singh Ajay Vikram, Sharma Dolly, "Evaluation Criteria for Measuring the Performance of Recommender Systems, IEEE 2015.

[3]. Xiao-Lin Zheng, Senior, Chao-Chao Chen, Jui-Long Hung, Wu He, Fu-Xing Hong, and Zhen Lin, "A Hybrid Trust-Based Recommender System for Online Communities of Practice", IEEE TRANSACTIONS ON LEARNING TECHNOLOGIES, VOL. 8, NO. 4, OCTOBER-DECEMBER 2015.

[4]. Yuan Cheng, Jaehong Park, and Ravi Sandhu, "An Access Control Model for Online Social Networks Using User-to-User Relationships", IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 13, NO. 4, JULY/AUGUST 2016, pp. 424-425

[5]. M. Jones, "Introduction to approaches and algorithms for Recommendation systems" December 12, 2013 [online] available: https://www.ibm.com/developerworks/library/os-recommender.

[6]. Linke Guo, Chi Zhang, and Yuguang Fang, "A Trust-Based Privacy-Preserving Friend Recommendation Scheme for Online Social Networks", IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, JULY/AUGUST 2015, pp. 413-415.

[7]. Pan Zhou, Yingxue Zhou, Dapeng Wu, Hai Jin, *Seni,* "Differentially Private Online Learning for Cloud-Based Video Recommendation With Multimedia Big Data in Social Networks", IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 18, NO. 6, JUNE 2016, pp. 1217-1219.

[8]. Hanhua Chen, Hai Jin, and Shaoliang Wu, "Minimizing Inter-Server Communications by Exploiting Self-Similarity in Online Social Networks", IEEE TRANSACTIONS PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 27, NO. 4, APRIL 2016, pp. 1116 -1125.

[9]. Hanhua Chen, Member, IEEE, and Hai Jin, Senior Member, IEEE and Fan Zhang, "CBL: Exploiting Community based Locality for Efficient Content Search Service in Online Social Networks", IEEE TRANSACTIONS ON SERVICES COMPUTING, 2015, pp. 1 -12.

[10]. Esther Palomar, Lorena González-Manzano, Almudena Alcaide, Álvaro Galán, "Implementing a privacy-enhanced attribute based credential system for online social networks with co-ownership management", The Institution of Engineering and Technology 2016, pp. – 60- 68.

[11]. Yifeng Zeng, Xuefeng Chen, Yew-Soon Ong, Jing Tang and Yanping Xiang," Structured Memetic Automation for Online Human-like Social Behavior Learning", IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION [Online]. Available: http://dx.: doi.org/ 10.1109/TEVC.2016.2577593

[12]. Amazon.com [Online]. Available: http://en.wikipedia.org/wiki/Amazon.com

[13]. R. Chen, E. K. Lua, and Z. Cai, "Bring order to online social networks," in Proc. IEEE INFOCOM, 2011, pp. 541–545.

[14]. R. Chen, E. K. Lua, and Z. Cai, "Bring order to online social networks," in Proc. IEEE INFOCOM, 2011, pp. 541–545.

[15]. M. Srivatsa, L. Xiong, and L. Liu, "TrustGuard: Countering vulnerabilities in reputation management for decentralized overlay networks," in Proc. 14th Int. Conf.WorldWideWeb, 2005, pp. 422–431.

[16]. D. N. Kalofonos, Z. Antonious, F. D. Reynolds, M. Van-Kleek, J. Strauss, and P. Wisner, "MyNet: A platform for secure p2p personal and social networking services," in Proc. IEEE 6th Annu. Int. Conf. Pervasive Comput. Commun., 2008, pp. 135–146.

[17]. S. Wen, W. Zhou, J. Zhang, Y. Xiang, W. Zhou, and W. Jia , "Modeling propagation dynamics of social network worms," IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 8, pp. 1633–1643, Aug. 2013.

[18]. A. Sridharan, Y. Gao, K. Wu, and J. Nastos, "Statistical behavior of embeddedness and communities of overlapping cliques in online social networks," in Proc. INFOCOM, 2011, pp. 546–550.