Journal of Information & Optimization Sciences
ISSN 0252-2667 (Print), ISSN 2169-0103 (Online)
Vol. 47 (2026), No. 5-B, pp. 2021–2029
DOI : 10.47974/JIOS-2293

Tag-based recommendation system using bi-clustering

Mahesh Mali *
Payal Mishra †
Hiral Modi §
Department of Computer Engineering
Mukesh Patel School of Technology Management & Engineering
SVKM's NMIMS
Mumbai 400056
Maharashtra
India

Abstract

With the onset of the new age, consumers are overwhelmed by the many choices of movies that can be watched. A solution gives rise to the need for a recommendation system (RecSys) that will suggest a title to the user based on user requirements. The new RecSys must be able to address the issues that exist within the plethora of general RecSys. In this work, we have implemented a novel Tag-based RecSys to improve the quality and usefulness of recommendations based on user information and movie tags. This paper proposes a bi-clustering approach for simultaneously clustering movies and users. This work is implemented using the MovieLens datasets. In trying to gauge the efficacy of the new RecSys, the final RMSE of our Tag-based RecSys is 0.9984, and the MAE is 0.8450. The system also produces more personalised results for users by personalising scores. The system evaluation shows considerably positive results.

Subject Classification: *Primary 93A30, Secondary 49K15.*

Keywords: *Tag-based recommendation system, Movies tag analysis, Tag-based movie recommender.*

1. Introduction

Due to the overwhelming growth in the amount of digital data and consumable content, users have been faced with the risk of being inundated by irrelevant media. It leads to a massive amount of user-generated metadata and a potential challenge of information overload, which further results in the slow

* *E-mail:* **mahesh.mali@nmims.edu (Corresponding Author)**
† *E-mail:* **Payal.Mishra@nmims.edu**
§ *E-mail:* **Hiral.Modi@nmims.edu**

acquisition of resources of interest on the internet [2]. These recommendation systems often help users by suggesting items they are likely to be interested in and estimating how much they might prefer each item, making it easier to discover relevant content. A lot of the user metadata on the internet is available in the form of tags that indicate some degree of preference of a user for that particular resource [1]. The frequent occurrence of a tag indicates the intensity of the user's preference for that resource. For instance, in the case of watching movies online, tagging is often used to add comments or opinions about the content, which expresses the users' interests and preferences. These tags can be used to predict and estimate other similar content that the user might enjoy and provide such recommendations, which is explored further in this paper. In this paper, we submit a tag-based recommender system based on the concept of bi-clustering [4]. Bi-Clustering helps to understand the result of overlapping multiple clusters simultaneously as a co-cluster [3]. The paper is organised into multiple sections for better clarity. Section 3 explains the movie tagging system, and Section 4 explains the bi-clustering method. Section 5 showcases the general architecture of a Tag-based Recommendation System.

2. Background of work

Due to a variety of factors, tags are increasingly being used to produce efficient recommender systems. In a customer service environment that can store or gather numerous types of user data, recommendation systems are information filtering systems that offer personalised product recommendations to users. Information filtering, which is mostly employed in recommendation systems, can be tailored to a user's preferences or limit recommendations to content that has been determined to be helpful to the user [5, 6, 7].

By estimating the user's general preferences at the present from various tags based on various time intervals, [7] develops a content-based recommender system. There have been multiple demonstrations of the advantages of using tag-based profiles for tailored music suggestions on Lastfm [7]. When tailoring recommendations provided by a tag-based collaborative recommender system [6], similar knowledge of the product items as the subject of the recommendations is taken into account as another element in addition to the similar tags [8]. Tags can serve a variety of purposes, and the act of tagging itself can be affected by a variety of conditions [9]. There are solutions available where social bookmarking tagging gives recommender systems context [10] in the form of context cues from tags and connections among users to enhance the collaborative recommender system [11].

Collaborative filtering is the most widely known algorithm in this domain [12, 13]. This algorithm uses user activity and its subsequent history to simulate user choices and preferences. The first paper that paved the way for future studies on collaborative filtering was released by GroupLens [2], and it based its findings on user-based collaborative filtering and pioneered a path for future studies.

Content-based filtering is a simple one, that is, to suggest goods that hold meaning or relevance to previous favourites of the user [14]. There are many ways to model an item, but the most widely known and famous one is the vector space model [15].

A hybrid recommendation system combines the collaborative filtering capabilities with Content-based filtering methods [1].

3. Movie Tagging System

A tag is defined as a word freely added to a movie by a user. Usually, there is no limitation on the number of tags that can be added to a movie. Adding tags provides descriptions of movies and facilitates searching among movies [15]. A tag recommender system generally uses a database of user-assigned tags and movie experts' assigned tags to generate new recommendations [16]. With the rise of Artificial Intelligence and Machine Learning, recommendation systems have become an integral part of modern digital platforms [3].

Figure 1
Concept of Biclustering

There is a recommendation system operating in the background to provide suggestions to the user based on the tags associated with a specific movie, the tags (more specifically pertaining to movies) being genre, actors, and director of a movie, as well as other keywords that describe the content or themes that exist within the movie. A movie tag-based recommendation system leverages descriptive keywords or tags linked to movies to suggest relevant titles to users based on their interests [17].

These tags can include information such as the genre, actors, and director of a movie, which may deal with more sensitive and in-depth information, such as keywords that describe the movie's content or themes. Figure 1 shows the tags for the Jumanji Movie, including the number of times it was repeated. Research in this area typically involves studying the effectiveness of these tags for generating recommendations [7].

4. Bi-Clustering Methods for RecSys

Bi-clustering is a data mining method that allows clustering of the rows and columns of a matrix [4]. We have users in rows and movies in columns, and the intersection denotes the ratings for each movie by each user. A bi-clustering will cluster users as well as movies to find small Bi-clusters [18]. The constraint on each movie is that it must have a minimum of 10 tags and a minimum of 10 ratings from users.

5. Tag-Based RecSys Architecture

The tag-based RecSys first performs bi-clustering based on a set of movies and a set of users. Bi-clustering can help predict the group ratings of various co-clusters. The following steps will help us produce the recommendations and test the results of tag-based recommendations.

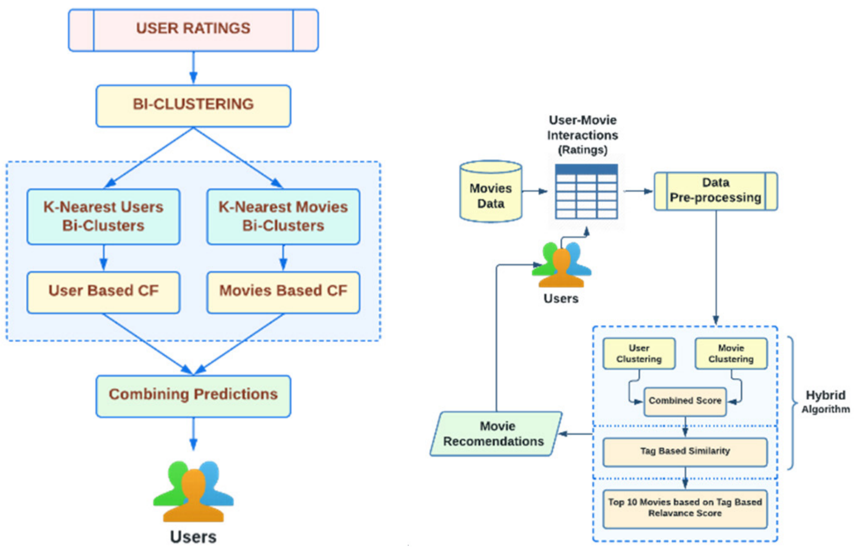


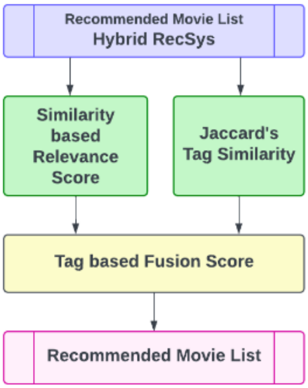
Figure 2
Tag-Based Recommendation Architecture using Bi-clustering

6.1 *Data Processing*

The first step in creating a model for a tag-based recommendation system consists of collecting vast amounts of movie data, which includes multiple parameters of online movie viewing, user details, movie details, ratings, tags, comments, etc [19]. It includes the interaction that occurs between users and the movie platform, which involves ratings, tags, and comments that indicate the user's interest in that particular movie.

6.2 *Bi-Clustering*

The next step in this process is applying the hybrid and then the bi-clustering algorithms to create a recommendation system [20] as shown in Figure 2. The user clusters contain relevant user data, and the movie clusters contain relevant information about the movie, its genre, the ratings received, the popularity ratings, the run time (which may be a subjective preference based on the genre), etc.



	movieId	Title	tag	relevance	relevance_rank
1156	2	Jumanji	adventure	0.98100	1
1711	2	Jumanji	jungle	0.96700	2
1331	2	Jumanji	children	0.96425	3
1715	2	Jumanji	kids	0.95850	4
2078	2	Jumanji	special effects	0.93475	5
1504	2	Jumanji	fantasy	0.93400	6
1189	2	Jumanji	animals	0.91450	7
1542	2	Jumanji	fun movie	0.87050	8
1330	2	Jumanji	childhood	0.87025	9
1501	2	Jumanji	family	0.86925	10

Figure 3
Tag-based Fusion similarity and Relevance Score calculation

Bi-Clustering allows simultaneous clustering of the rows (users) and columns (movies). The clustering of these leads to the eventual production of the cluster ratings for each user cluster based on their relevance.

6.3 *Find Tag-Based Fusion Similarity*

In this segment, we find the relevance of multiple tags pertaining to a particular movie. Subsequently, in Figure 3, we have computed a ranking table according to the relevance associated with a specific movie, which in this case is a popular cult fantasy movie called Jumanji [6].

6.4 *Compute Tag Relevance Scores*

Based on the relevance scores recorded above, a ranking is computed. Figure 3 is computed [21]. Ranks are allotted in accordance with the value of relevance scores; the highest relevance score gets rank 1, and other subsequent tags with an inferior level of relevance allotted by the model will be delegated lower ranks. These ranks show the order of relevance of the tags associated with the movie [21].

6.5 *Rank and Present Recommendations*

After finding the high-ranking tags of a particular movie, we can infer their relevance to other similar movies and use their metadata to find other seemingly similar movies. This list of movies is shortlisted by finding movies in which the relevance of those same tags is high as well. The Jaccard index is used for this in order to compute the degree of association or similarity between 2 movies that have the same tags. The movies are ranked based on Jaccard's similarity.

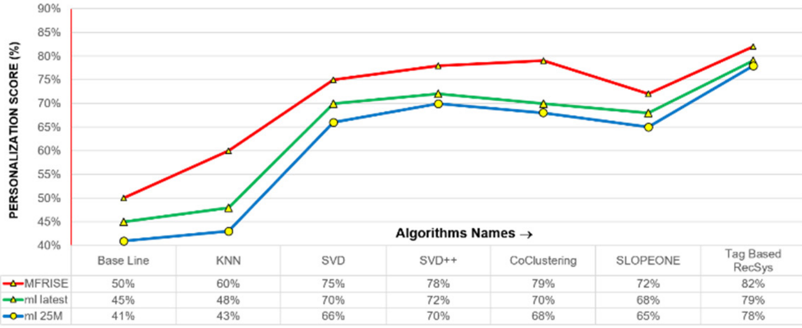
7. Experimentation and Evaluation Metrics

The work is performed and tested on an HP desktop all-in-one PC with an Intel i5 CPU having a speed of 2.33 GHz and 8GB of RAM. It has 1TB of SSD memory to place the dataset for faster access and optimum performance. We used three datasets, MFRISE [21], ML_latest [5], and ML_25M [5], of different sizes for experimentation. The MFRISE dataset is mainly used to validate the recommendations, which has 2100 users, 10000 movies, and 855600 user ratings. The dataset contains data on actors, directors, countries, and tag assignments. The evaluation of the recommendation system algorithm is not as simple as other traditional machine learning algorithms, because the result of the recommendation system algorithm is different for each user [2]. The most popular error metrics, Mean Absolute Error (MAE), Mean Square Error (MSE) and Root Mean Square Error (RMSE) [2] are used in our experimentation. The time required for processing and fitting the model is also used as a performance evaluation metric. A recommendation algorithm implemented with huge data sets may consume considerable time. The time taken to fit the algorithm with the given model of data is called the Fit Time, and the time required to execute the algorithm on test data is the Test Time.

6. Results and Discussion

The section will present an analysis of experimental results obtained after the implementation of a novel Tag-Based recommender system. The bi-clustering method helped the system to predict the group rating for each bi-cluster (combination of a movie cluster and a user cluster). The bi-cluster score is computed using the ANOVA method, which helps us to address the problem of cold start. The tag-based RecSys produces more personalised results for movies with many tags associated with them. The results for new movies or movies with fewer tags depend on the bi-clustering ratings. The bi-clustering method handles the cold start issue better, as the RMSE values for cold movies are almost similar to those of other movies.

The Tag-based recommender system was evaluated using three different datasets of different sizes, as shown in Figure 4. The personalisation scores are computed for all benchmarking algorithms and compared with the Tag-based recommender system. The personalisation score was computed by computing dissimilar results given to multiple users. The score is computed by calculating the cosine similarity of results and subtracting one to get dissimilarity. This score gives the Personalisation introduced by our new tag-based algorithm to each user by offering personalised recommendations. The ideal recommender system must offer different movie recommendations to different users. A higher Personalisation score indicates that each user got recommendations for different movies.



	Base Line	KNN	SVD	SVD++	CoClustering	SLOPEONE	Tag Based RecSys
MFRISE	50%	60%	75%	78%	79%	72%	82%
ml latest	45%	48%	70%	72%	70%	68%	79%
ml 25M	41%	43%	66%	70%	68%	65%	78%

Figure 4
Personalisation Scores of all Benchmarking Algorithms

The Tag-based recommender system was also evaluated using three different datasets of different sizes, as shown in Figure 5. The RMSE and MAE scores are computed for all benchmarking algorithms and compared with the Tag-based recommender system. The various error metrics are computed for our tag-based recommender, like root mean square error(RMSE) and mean absolute error (MAE), which are found as 0.9984 and 0.845. So, a tag-based recommender is performing at par with the benchmarking recommender methods.

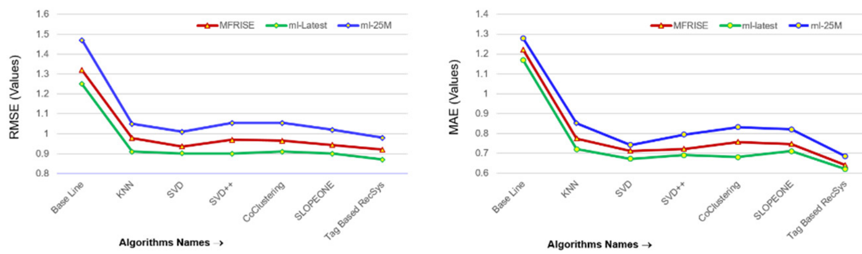


Figure 5
MAE and RMSE Performance Metrics for Benchmarking Algorithms

5. Conclusion

Recommender systems are a necessity of the information age; they help to suggest a set of movies to a group of users from the available movies. In this paper, we presented a novel tag-based RecSys using biclustering. After much experimentation, it's observed that the efficiency of the recommender algorithm is mainly hampered by the type of dataset used for experimentation. We used the hybrid tag-based recommender algorithm to offer the benefits of content-based and collaborative filtering. Our recommender's error metric values are on par with other benchmarking methods. Root mean square error (RMSE), mean absolute error (MAE), Fit Time, and Fit Time are 0.9984, 0.845, 0.067, and 0.05, respectively. Our proposed work with respect to processing time is at par with the existing methods. The Personalisation score of 0.82 in Figure 4 shows that the recommendation of the top 10 movies for two different users has almost eight movies that are different from the other user's recommendation list. It concludes that a tag-based recommender system suggests very personalised recommendations that are in the user's interest. Finally, we can conclude that this work has given a novel Tag-based recommendation system with lower RMSE and MAE errors compared to many recommender systems. This new recommender system also offers more personalised results based on users' interests. The bi-clustering feature can help to generate an implicit rating for a group of users. In the future, we will incorporate some more features in the dataset to improve the accuracy of recommender systems using the SVD algorithm.

References

[1] S. Wei, X. Zheng, D. Chen, and C. Chen, "A hybrid approach for movie recommendation via tags and ratings," *Electronic Commerce Research and Applications*, vol. 18, pp. 83–94 (2016).

[2] T. Bogers, "Tag-based recommendation," in *Social Information Access: Systems and Technologies*, pp. 441–479 (2018).

[3] H. Liang, Y. Xu, Y. Li, and R. Nayak, "Tag-based collaborative filtering for recommender systems," in *Proc. 4th Int. Conf. Rough Sets and Knowledge Technology* (RSKT), Gold Coast, Australia, pp. 666–673 (Jul. 2009).

[4] M. Mali, D. S. Mishra, and M. Vijayalaxmi, "Multifaceted recommender systems methods: A review," *Journal of Statistics and Management Systems,* vol. 23, no. 2, pp. 349–361 (2020).

[5] F. Ricci, L. Rokach, and B. Shapira, "Recommender systems: Introduction and challenges," in *Recommender Systems Handbook*, pp. 1–34 (2015).

[6] Y. Mehta, A. Singhania, A. Tyagi, P. Shrivastava, and M. Mali, "A comparative study of recommender systems," in *Proc. ICDSMLA 2019*, pp. 1021–1029 (2020).

[7] C. S. Firan, W. Nejdl, and R. Paiu, "The benefit of using tag-based profiles," in *Proc. Latin American Web Conf. (LA-WEB)*, p. 341 (2007).

[8] P. K. Vatturi, W. Geyer, C. Dugan, M. Muller, and B. Brownholtz, "Tag-based filtering for personalised bookmark recommendations," in *Proc. 17th ACM Conf. Information and Knowledge Management*, pp. 1395–1396 (2008).

[9] C. S. Firan, W. Nejdl, and R. Paiu, "The benefit of using tag-based profiles," in *Proc. Latin American Web Conf. (LA-WEB)*, p. 341 (2007).

[10] M. Mali, D. Mishra, and M. Vijayalaxmi, "Bi-clustering based recommendation system," *Journal of Information and Optimization Sciences*, vol. 45, no. 4, pp. 1029–1039 (2024).

[11] R. Nakamoto, S. Nakajima, J. Miyazaki, and S. Uemura, "Tag-based contextual collaborative filtering," *IAENG International Journal of Computer Science*, vol. 34, no. 2 (2007).

[12] R. Sharma, D. Gopalani, and Y. Meena, "Collaborative filtering-based recommender system: Approaches and research challenges," in *Proc. 3rd Int. Conf. Computational Intelligence & Communication Technology (CICT)*, pp. 1–6 (2017).

[13] L. Feng, Q. Zhao, and C. Zhou, "Improving performances of top-n recommendations with a co-clustering method," *Expert Systems with Applications*, vol. 143, p. 113078 (2020).

[14] P. Lops, M. De Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in *Recommender Systems Handbook*, pp. 73–105 (2011).

[15] R. Singh, K. Alam, and N. Bhardwaj, "Graph-based recommendation algorithm for personalized suggestions," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 28, no. 3, pp. 931–944 (2025).

[16] M. Chhikara and S. K. Malik, "A recommendation system for online social semantic network using knowledge based, content based and collaborative filtering," *Journal of Information and Optimization Sciences,* vol. 44, no. 4, pp. 795–806 (2023).

[17] R. Ahuja, A. Solanki, and A. Nayyar, "Movie recommender system using k-means clustering and k-nearest neighbour," in *Proc. 9th Int. Conf. Cloud Computing, Data Science & Engineering (Confluence)*, pp. 263–268 (2019).

[18] A. Tanay, R. Sharan, and R. Shamir, "Biclustering algorithms: A survey," in *Handbook of Computational Molecular Biology*, vol. 9 (2005).

[19] N. Hug, "Surprise: A Python library for recommender systems," *Journal of Open Source Software,* vol. 5, no. 52, p. 2174 (2020).

[20] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian Informatics Journal,* vol. 16, no. 3, pp. 261–273 (2015).

[21] M. Mali, D. Mishra, and M. Vijayalaxmi, "Benchmarking for recommender systems (MFRISE)," *3C TIC: Cuadernos de desarrollo aplicados a las TIC*, p. 146 (2022).

Received April, 2025