



# A proficient video recommendation framework using hybrid fuzzy C means clustering and Kullback-Leibler divergence algorithms

H. Anwar Basha<sup>1</sup> · S. K. B Sangeetha<sup>2</sup> · S. Sasikumar<sup>3</sup> · J. Arunnehr<sup>2</sup> · M. Subramaniam<sup>4</sup>

Received: 14 April 2021 / Revised: 9 February 2022 / Accepted: 31 January 2023 /  
Published online: 7 February 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

## Abstract

A video recommendation framework for e-commerce clients is proposed using the collaborative filtering (CF) process. One of the most important features of the CF algorithm is its scalability. To avoid the issue, a hybrid model-based collaborative filtering approach is proposed. KL Divergence was developed to address the CF technique's scalability problem. The clustering with enhanced sqrt-cosine similarity Recommender scheme is proposed. For successful clustering, Kullback–Leibler Divergence-based Fuzzy C-Means clustering is suggested, with the aim of focusing on greater accuracy during movie recommendation. The proposed scheme is viewed as a trustworthy contribution that significantly improves the ability of movie recommendation by virtue of the KL divergence-based Fuzzy C-Means clustering mechanism and enhanced sqrt-

---

✉ H. Anwar Basha  
anwar.mtech@gmail.com

S. K. B Sangeetha  
sangeets8@srmist.edu.in

S. Sasikumar  
sasikumar@saveetha.ac.in

J. Arunnehr  
arunnehr.aucse@gmail.com

M. Subramaniam  
subbu.21074@gmail.com

<sup>1</sup> School of Computer Science and Engineering, REVA University, Bengaluru, India

<sup>2</sup> Department of Computer Science and Engineering, College of Engineering and Technology, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Vadapalani Campus, Chennai, India

<sup>3</sup> Department of Computer Science and Engineering, Saveetha Engineering College, Chennai, India

<sup>4</sup> Department of CSE, Chaitanya Bharathi Institute of Technology, Hyderabad, India

cosine similarity. The proposed scheme highlighted and addressed the critical role of the KL divergence-based cluster ensemble factor in improving clustering stability and robustness. For prediction, the enhanced sqrt-cosine similarity was used to calculate successful related neighbor users. The performance of Recommendation is improved when KLD-FCM is combined with improved sqrt-cosine similarity. The proposed scheme's empirical work on the Movielens dataset in terms of MAE, RMSE, SD, and Recall were found to be superior in recommendation accuracy compared to traditional approaches and some non-clustering based methods recommended for study. With the specified number of clusters, it is capable of providing accurate and customized movie recommendation systems.

**Keywords** Recommendation systems · Content-based filtering · Knowledge base · Fuzzy C-means algorithm

## 1 Introduction

A Recommendation System (RS) is a filtering program that enables consumers to evaluate product recommendations for online purchases and provide information into products that they are interested in. In recent years, the extensive advancement of science and technology has resulted in vast amounts of digital knowledge being accessible via the internet. As a result of the overabundance of data, users are unable to obtain accurate taste information. This problem of information overload can be solved with RS, which filters out irrelevant information and only recommends related things to users. RS's knowledge filtering system assists users in making decisions in difficult situations by scanning vast data for items of interest. It also offers a personalized proposal based on the user's desires and preferences. RS is widely used in e-learning, e-shopping, e-tourism, e-business, e-government, and social networking sites like Facebook. Articles from research, books, music, news, films, DVDs/CDs, and other e-shopping products recommended by RS for their clients [1, 2].

To create a working recommendation system, a large amount of data must be collected. RS accepts a variety of inputs, both explicit and implicit. The explicit ratings of 1 to 5 given by users for their preferences in the items they purchase make up the covert reviews. User actions such as accessing and navigating past websites, click and search logs provide implicit input. Demographic data is another addition to RS. This information index was developed for each client who visits the site. Following this point, the data is filtered to obtain sufficient data for customer/user suggestions. Content-based filtering, collaborative filtering, and hybrid filtering technology are examples of filtering algorithms that would be more suitable for the recommending engine [3, 4]. The collection of data and application of recommendation filtering methods yield a set of recommendations that comply with the procedures to be considered in a recommendation system's calculation. In general, two types of performance are predicted and suggested. The ranking items for which the target consumer has not been rated are forecasted by prediction. On the basis of the forecasted ratings, the recommendation recommends the top-n recommendations to the target consumer, where each item does not include an evaluation value for these top-n recommendations. The consumer should be ecstatic with the performance of the recommender [5].

A framework that offers good and helpful recommendations for its own users requires the use of appropriate and reliable recommendation techniques. The content-based technology

employs a domain-based algorithm that focuses on analyzing the characteristics of predictive posts. When documents such as blogs, magazines, and news are recommended, it is the most effective material filtration technique. The user profile recommendation in the Content-Based Filtering (CBF) approach is based on characteristics extracted from the content of items checked by the user. Objects that are mostly associated with positive items are recommended to the customer. CBF employs a variety of models to identify similarities between documents in order to generate useful recommendations. To form the relationship between different documents within the Corpus, it could use a Vector Space Model or a probabilistic model like the Naive Bayes Classification method, Decision Trees, or Neural Networks. It's also possible to use the Vector Classification method. These approaches make recommendations based on the underlying model's statistical analysis or machine learning techniques [6].

Other users' profiles aren't needed for content-based filtering because they don't affect the recommendation. Furthermore, as the user profile changes, CBF's approach adjusts its recommendations in a very short time. The key drawback of this method is that it necessitates thoroughly informing and explaining the characteristics of the objects in the profile. CF is a domain-agnostic content prediction technique that can't be easily or reliably classified by metadata like film or music. Filtering by working together creates a database of user preferences (user-item matrix). It then matches individuals with relevant interests and preferences to make recommendations. This group of people is creating a social network. A consumer receives recommendations for items that he hasn't yet rated but that other users in his field have given high marks to. CF may make recommendations in the form of forecasts or recommendations. Collaborative filtering has a range of benefits over CBF, including the ability to be used in fields where object content is scarce and computer system content is difficult to assess (such as opinions and ideals). CF technology should provide persuasion recommendations so that things that are useful to the user can be recommended, even though the user profile does not include the content [9].

Hybrid filtration strategies combine various recommendation approaches to boost device optimization and prevent some of the drawbacks and issues that come with pure recommending systems. Since one algorithm can overcome its drawbacks with another algorithm, a combination of algorithms is built to make recommendations more accurately and efficiently than a single algorithm. Using various recommendation models, a combined model will eradicate the flaws of a single process. Separate algorithms can be applied and the results merged, content-based co-filters can be used, content-based collaborative filters can be used, and a single recommendation system can be developed to bring all methods together [10].

The most promising products from which consumers can choose are included in the recommendation question. Some well-known approaches for solving the problem of scalability under model-based collaboration filters are clustering-based approaches. Predominantly, the majority of CF-based clustering strategies have relied on K-means and Fuzzy C-means clustering, which lack the ability to pick a relevant clustering core, lowering the predictive efficiency. As a result, there are trade-offs between scalability and predictive efficiency. Improved clustering techniques were suggested in the study to recognize agreed problems due to scalability. Many previous studies have shown that clustering-based CF systems (CF combined with clustering algorithms) are a promising schema for providing accurate personal recommendations and solving large-scale problems [8]. Fuzzy C-Means is a soft clustering approach that allows each individual data to be allocated to multiple clusters based on different membership degrees. They also concluded that good clustering-based CF performance is dependent on appropriate clustering techniques as well as the dataset's design. The analysis shows that the Clustering algorithm's stability and robustness need to be

improved in order to achieve critical accuracy in the process of movie recommendation to target users. The following are the limitations of the current method:

- Fuzzy-C refers to a lack of ability to choose the initial cluster Center point, which can lead to a local optimum solution and affect clustering accuracy. In certain cases, the obtained clusters can be impractical, affecting the Recommendation outcomes.
- To get a good grouping of data, most current clustering algorithms require the configuration of some parameters.
- The disadvantage of FCM is that it has a higher error rate and needs further iterations to obtain well-framed clusters.
- Because of the prejudices and assumptions that each clustering algorithm contains, applying a single clustering method generally results in inconsistent results.

The Fuzzy C Means clustering with KL divergence is suggested to solve the aforementioned limitations. The research work's contributions have the following features.

- To prevent the drawbacks of a bad initialization, Ensemble FCM clustering is used to divide users into separate groups.
- The Ensemble Fuzzy C-Means clustering methods use a KL divergence-based cluster ensemble factor to improve the stability and accuracy of the clustering process, resulting in successful clustering with the goal of focusing on better performance results during movie recommendation.
- A better approach is to treat the membership vector as a discrete probability function, with the statistical distance, such as KL divergence, serving as the similarity metric.
- For active users, the enhanced sqrt-cosine similarity is often used to find the most powerful nearest neighbors.
- Reduce the problem of scalability.

The latest analysis of RS methods is summarized in Section 2. In addition, work on various RS is discussed in this chapter. The proposed framework model for the hybrid video recommender is defined in Section 3. The quality of predictions as measured by the assessment metrics is also stated. Section 4 brings the analysis to a close by highlighting the algorithms that aided in the achievement of the objectives and promoted the desired outcome. The study's limitations have been established, and potential research directions have been summarized.

## 2 Related study

Recommend systems use data mining and predictive algorithms to predict user preferences among the vast array of images, goods, and services available. The rapid growth of knowledge on the Internet, as well as the number of visits to websites, are posing significant challenges to system recommendations. The development of precise recommendations, the successful management of a large number of recommendations, and the large number of system members are all examples of these challenges. As a result, new system recommendation technologies are needed that can produce high-quality recommendations quickly, even for large data sets. There are numerous methods and algorithms for data filtering and recommendation. This section provides a brief overview of recent system-related studies in the literature.

Collaborative filtering is a widely used and relevant technology that makes predictions and suggestions based on the ratings and actions of other system users. The key premise of this strategy is that the user can pick and consolidate views from other users in order to understand his choice for the active user. Memory-based CF algorithms generate a forecast for the entire database or a subset of the user-item database. Each consumer is a member of a group of people who share similar interests. A prediction of a new user's tastes for new objects can be rendered by identifying related neighbors (or active users). Memory-based collective filtering has a number of drawbacks, the most significant of which is that it must use the entire database every time it predicts something, making it extremely sluggish in memory. If the rating matrix is so broad that many people use it, the issue becomes serious. Computing resources are depleted, and device performance suffers as a result, making it impossible for the system to respond quickly to user requests [11].

The model-based approach learns a model to boost collaborative filtering technology efficiency using prior scores. The model could be built using machine learning or data mining techniques. Since these approaches rely on pre-computed models, they can quickly suggest a large number of items and have been shown to yield results that are comparable to neighborhood-based recommendation techniques. Dimension reduction, for example, includes techniques including singular decomposition, matrix completion, latent semantic approaches, regression, and clustering. Content-based filters recommend items on a user's item profile and user profile. When an account is created and the framework is first used, these types of profiles are created. As a result of the user's interaction with the system, a better user profile is developed. If a user likes an object in the past, the CBF scheme assumes that the user would like similar things in the future [7]. The most powerful filtering technique is used in information documents such as web sites, journals, and news. To produce meaningful recommendations, CBF employs a variety of models to detect correlations between documents. Model-based vector space models can be used to represent the relationships between various documents within a corpus, such as reverse frequency or probabilistic models like the Naive Bayes Classification, Decision Trees, or Networks. Object metadata is used in the filtration mechanisms. Before users can get a recommendation, they'll need a large collection of items and a well-organized user profile. As a result, the effectiveness of CBF is dependent on the availability of descriptive data. Over-specialization is another major problem with the CBF methodology. Users can only get suggestions that are close to their own [12].

In certain applications, hybrids of various types outperformed individual algorithms. When the algorithms in question cover a wide range of use cases or aspects of the data set, hybrids can be particularly useful. The suggestion has been suggested to be implemented using a range of approaches, including material-, collaborative-, knowledge-based, and other techniques. Every form of recommendation has its own set of strengths and weaknesses. In order to improve efficiency, these strategies were often combined into hybrid recommenders. The hybrid recommender method has a higher level of complexity and implementation costs [14].

The Knowledge Base (KB) suggestion suggests things based on user experience, artifacts, and/or user relationships. In most cases, KB recommendations maintain a knowledge base describing how a particular item serves the needs of a specific person, which can be carried out based on inferences about the relationship between a user's need and a potential recommendation. The semantic similarity between objects can be calculated using the domain ontology. Social recommendation services are an integral part of everyday life on social media. Every minute, users on social media exchange details. Social advisor programs are designed to help people understand what they really want by reducing the amount of information available on

social media. They want to help people on social networking sites like Facebook, Twitter, YouTube, Flickr, and Weibo by providing them with tweets and profiles that meet their needs [13].

Systems that are recommended are those that can analyze past user habits and make suggestions for current issues. Simply put, information on similar behavior, remarks, and users will be used in the RSs to try to define the user's thought style in order to assess and suggest the user's taste as the most appropriate and near object. Many of the methods in the RSs are used to make recommendations that are as accurate as the users need. As a result, several models for RSs clustering existing data have been presented in order to efficiently process data with large volumes. Given the goals, special characteristics, and relationships between data in the RSs, using an effective data cluster approach to efficiently process data in subsequent steps and produce more reliable suggestions has always been seen as an important research area for developing these systems [15].

Collaborative filtering recommendation systems are extremely useful for a wide range of online activities, including e-commerce. However, there are significant issues, especially in scalable and dynamic scenario implementations where new users, objects, and ratings are added frequently. Scalability refers to a system's, network's, or process's ability to control or expand its capacity to accommodate development. For example, a device is called scalable if it can increase its total power under increased load as resources (typically hardware) are added. Dimension reduction-based approaches address scalability problems such as SVD, MF, clustering, and so on. In short, cluster-based approaches retain the advantages of low computation cost (for searching candidates) over memory-based approaches as models for dimension reduction [16, 17].

To improve the performance of the recommendation, related users are grouped together based on their interests. Clustering was used to quickly locate a user's neighbors. By reducing the size of the original data to more manageable partitions, clustering systems can react quickly. Clustering, in particular, boosts the scalability and accuracy of recommendation systems. Despite the advantages of low machine costs (for searching candidates) over memory-based and SVD methods, MF methods remain cluster-based methods. One of the most serious problems with a recommending method is scarring, and data sparsing has an effect on the accuracy of the recommendation. In general, machine data like Movie Lentsils is interpreted as a user-item matrix made up of films, which increases matrix dimensions and sparsity since the user and items are no longer used. Most users do not rate most items, and there are few available ratings. The key explanation for this is a lack of knowledge. In order to condense the user's object matrix, the reduction of dimension addresses the issue of scarcity by excluding non-representative or insignificant users or objects. However, during the reduction process, potentially valuable information is lost. However, some potentially valuable information can be lost during the reduction process [18, 19].

Collaborative filters create this problem because they depend on the rating matrix in most cases. Many researchers have attempted to address this problem, but more research is still required in this area. Most recommendation systems on the major electronic commerce platforms have been influenced by the long tail effect in some way. Since accuracy-focused recommender systems tend to recommend common goods, recommending items with few ratings is critical (long tail items). Popular products that are likely to be less helpful to users can be easily recommended using detailed recommendation algorithms. To assess the ability of systems to recommend unpopular goods, the assessment metrics diversity and innovation have been added. Recommending long tail artifacts can result in the recommendation's precise

results being lost. As a result, a recommendation process must be developed that recommends controversial products while minimizing accuracy loss. Several guidelines have recently been proposed to strike a balance between precision, diversity, and novelty [20–22].

Existing CF clustering method algorithms are ineffective at improving RS efficiency and addressing scalability issues. The recommended performance has an effect on the efficiency of the clustering procedures [23, 24]. As a result, there is a lack of precision and coverage, which makes clustering-based approaches in recommender systems difficult to use in practice. To improve the recommendation's performance, better methods for optimizing the above problem are needed.

### 3 Proposed system model

#### 3.1 KL divergence based ensemble fuzzy C means clustering

Cluster-based CF has been shown to address scalability issues while also improving the consistency of recommendation outcomes in recent years. The aim of clustering algorithms is to group objects into clusters with the shortest distance between them in order to find objects that are identical. Clustering strategies will typically group a large number of users into various clusters based on their rating similarity in order to locate “like-minded” neighbors. One of the most widely used clustering methods is fuzzy clustering. To get a decent grouping of data, most current FCM algorithms require the specification of some parameters. As a result, the Fuzzy cluster ensemble solution usually prevents the drawbacks of a bad initialization.

##### 3.1.1 Fuzzy C means clustering algorithm

1. Initialize Membership matrix  $M$  with random data points.
2. Fuzzy cluster center is calculated  $C$
3. Calculate the objective function  $F = \sum M^k(X_i - C_j)$
4. For every iteration fuzzy Membership is updated by using  $M = \sum C^k$  Where  $k$  is the number of clusters
5. The iteration will stop when  $(k+1) - (k) < \text{termination criterion}$

Ensemble clustering blends a dataset's various simple partitions into a more stable and robust one. The basic concept behind the Ensemble Fuzzy C Means cluster method is to apply the clustering method to the data several times (rather than only once) and then merge the results into a single partition. Ensemble clustering takes a collection of data partitions as input. The cluster ensembles are divided into two parts. The ensemble clustering generator is one, and the consensus function is another. The first section focuses on generating more diverse clustering results, while the second section focuses on seeking a good consensus feature to increase the results' accuracy. Homogeneous ensemble FCM clustering is used in the first component of cluster ensembles. The term “homogeneous ensemble” refers to the use of multiple runs of a single clustering algorithm (fuzzy c-means algorithm) with different initializations and fuzzy parameter values. Several soft partitions of the data are obtained at the end of the first stage of the ensemble method as a result of several runs of the algorithm(s). The aim of this is to improve the accuracy and consistency of fuzzy cluster analysis procedures. Soft ensembles are

characterized by the concatenation of membership probability distributions in the second part of cluster ensembles. The obtained partition will be combined in the second stage using a KL divergence-based objective function to produce a single final partition. The Kullback–Leibler (KL) divergence was then used to describe a distance measure between two instances. The similarity of a membership vector to a cluster center is measured by FCM using squared Euclidean distance. This is ineffective in situations where a data's membership in all clusters normally equals one. A better approach is to treat the membership vector as a discrete probability function, with the statistical distance, such as KL divergence, serving as the similarity metric. This algorithm is identical to fuzzy c-means, with the exception that it employs the KL divergence to treat memberships as discrete probabilities.

The data from the User Item is first categorized using homogeneous fuzzy clustering methods. After that, a fuzzy KL divergence-based objective function aggregates the soft clustering effects.

### 3.2 Improved Sqrt-cosine similarity

Improved Sqrt-Cosine Similarity (ISC) is a modern similarity measure that uses Hellinger distance and is based on sqrt cosine similarity. Hellinger distance (L1 norm) is a much better metric for high-dimensional data mining applications than Euclidean distance (L2 norm). In terms of implementation, the ISC is very similar to cosine similarity, and it outperforms other similarity measures in high-dimensional results. The enhanced sqrt-cosine similarity determines how close two users are. For High Dimensional data, the Hellinger distance-based Similarity is more accurate. The KLD-FCM with enhanced sqrt-cosine similarity outperforms current systems in terms of recommendation performance.

### 3.3 Proposed KLD-FCM based movie recommendation scheme

For improving movie recommendation methods (KLD FCM-RS), a kullback–leibler divergence-based fuzzy c-means clustering is proposed, and the steps involved in KLD FCM-RS are discussed in detail. The aim of the proposed method is to develop a Collaborative Movie Recommender framework that can solve scalability problems while also improving prediction accuracy in terms of MAE, Precision, Recall, and Speed. The proposed KLD-FCM based Movie Recommendation Scheme is architecturally depicted in Fig. 1 as follows.

It is divided into two phases: offline and online. The User Item Rating Matrix derived from the used Movie Lens Dataset is used as a possible input during the offline process. Then, over the extracted user Item scores, a method of different homogeneous Fuzzy C means clustering is applied to divide the users into different classes. Furthermore, KL Divergence-based cluster ensemble FCM is used to combine the various FCM clustering findings in order to generate efficient single User clusters. The nearest cluster estimation for Active consumer is computed using the Euclidean distance method in the online process. The active user's nearest neighbors in his or her nearest cluster are then found using an enhanced sqrt-cosine similarity tool. Finally, the top list of recommended movie items is calculated in an online mode by determining the movie items that are most frequently recommended by the context's neighborhood users. Leibler–Kullback For the Recommender method, divergence-based Fuzzy C-Means clustering with enhanced sqrt-cosine similarity worked well. Three possible steps are included in this proposed KLD-FCM:



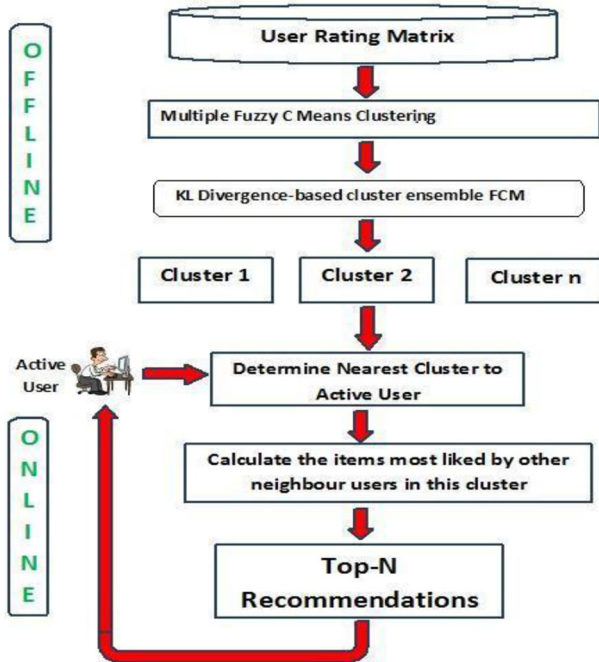


Fig. 1 Architectural view of the proposed KLD-FCM Scheme

### 3.3.1 Procedure

**Step 1 (FCM-KLD)** FCM-KLD is divided into two phases. Multiple Fuzzy C Means Clustering is the first step. The user data ratings from the Movie Lens data set are used as input in this phase of the clustering process. Over the User Item scores, apply three different homogeneous Fuzzy C Means clustering methods with different initializations. By executing the FCM several times for each initialization with different fuzzy parameter values, homogeneous FCM clustering is used to create several partitions with different random initializations (here 1.5, 2 and 2.5 are used). **Input:** User Rating Matrix **Output:** Three Clustering results.

Table 1 shows a snapshot of the User object rating matrix from the Movie Lens dataset for 5 users on 5 movies, where U1-U5 are users and M1-M5 are objects (movies). The value 1–5 represents the user’s likelihood rating for a specific film. The value ‘0’ denotes that the consumer has not rated (or seen) the film. The recommender framework identifies unrated values and suggests the top N films to the consumer Tables 2, 3, 4, and 5.

**Step 2 (determine the nearest cluster to active user)** After clustering the users into various clusters, the Euclidean distance approach is used to determine the nearest cluster to Active User.

$$sim_i(Cent_i, U) = \sum_{j=1}^d (Cent_{i,j} - U_j)^2 \tag{1}$$

- $Cent_i$  is the centroid of ‘i’ th cluster,  $U$  is the Active User Profile.
- $d$  is the dimension of data(Number of Attribute).
- $Cent_{i,j}$  is the jth attribute of centroid profile in cluster i.

**Table 1** Example of rating matrix from Movie Lens dataset

	M1	M2	M3	M4	.....	$M_m$
U1	5	3	4	3	.....	3
U2	4	?	?	?	.....	1
U3	?	?	?	?	.....	0
U4	5	?	?	2	.....	?
....	.....	.....	.....	.....	.....	.....
$U_n$	4	3	?	?	.....	1

**Table 2** MAE comparison analysis

Cluster size	5	10	15	20	25
Proposed	1.2	1.2	1.15	1.15	1.14
FCM	1.3	1.29	1.28	1.28	1.27
FCMBAT	1.4	1.39	1.38	1.38	1.37
COA	1.5	1.49	1.48	1.48	1.49

**Step 3 (using improved sqrt-cosine similarity, determine the top N recommended movies)** Improved sqrt-cosine similarity is used to calculate Active User’s nearest neighbors. The following formula is used to determine how close users  $u1$  and  $u2$  are.

$$sim(u1, u2) = \frac{\sum_{i=1}^m \sqrt{R_{u1,i} R_{u2,i}}}{\sqrt{(\sum_{i=1}^m R_{u1,i})} \sqrt{(\sum_{i=1}^m R_{u2,i})}} \tag{2}$$

$m$  Set of common items rated by user  $u1$  and user  $u2$ .

$R_{u1,i}$  is the rating given to item ‘i’ by user  $u1$ .

$R_{u2,i}$  is the rating given to item ‘i’ by user  $u2$ .

**Table 3** RMSE comparison analysis

Cluster Size	5	10	15	20	25
Proposed	0.78	0.77	0.76	0.76	0.75
FCM	0.81	0.8	0.79	0.78	0.77
FCMBAT	0.84	0.84	0.84	0.84	0.81
COA	0.87	0.86	0.86	0.85	0.83

**Table 4** Recall analysis

Cluster size	5	10	15	20	25
Proposed	25.28	25.68	27.24	27.8	28.56
FCM	23.12	24.68	25.45	25.12	25.89
FCMBAT	20.12	22.68	23.45	23.12	24.89
COA	18.9	19.89	20.35	20.02	21.67

**Table 5** Accuracy analysis

Cluster size	5	10	15	20	25
Proposed	89.3	88.2	88.2	87.1	87
FCM	86.3	86.6	86.3	85.4	85
FCMBAT	84.8	84	83.7	83.5	84.3
COA	82.5	82.4	82.3	82.1	81.67

The Hellinger distance is used to compute the similarity between two vectors in the enhanced sqrt-cosine similarity. This phase is essential for comparing each individual user’s rating to the ratings of other users in the clusters. Finally, the top list of recommended movie items that could be suggested to an active user at any time is calculated based on the movie items that are most often recommended by the context’s neighborhood users. The movies are recommended to target users who are most likely used by other neighbor users and are not used by him/her during the Recommendation process. The weighted average of the ratings of items in the same cluster neighbor’s is used to predict the ranking of unrated items for active users, and then the top-N suggestions list is sent to the active user. The rating of unrated movie (item) ‘i’ for an active user ‘a’ is predicted by  $P_a(i)$

$$P_a(i) = \underline{R}_a + \frac{\sum_{N \in C_x} Sim(a, N) \times (R_N(i) - \underline{R}_N)}{\sum_{N \in C_x} (|Sim(a, N)|)} \tag{3}$$

Where,

- $a$  Active User.
- $\underline{R}_a$  Average of active user a.
- $C_x$  set of nearest neighbors of active user a belonging to one common cluster.
- $\underline{R}_N$  Average rating score given by active user’s neighbor N.
- $Sim(a, N)$  similarity between active user a and Neighbor.

### 4 Experimental design

Experiments using the publicly accessible Movielens dataset are used to equate the performance of the proposed KLD-FCM with that of baseline recommendation system The Movielens data set used to equate the proposed KLD-FCM scheme to the compared COA, FCM-BAT, FCM, and ICF contains 10,00,000 ratings, with 850 users theoretically rating them. This Movielens data set contains reviews ranging from approximately 1000 to 1513 movies, each scored on a scale of 1 to 5. By partitioning the entire Movielens data set using the k-cross validation process, the performance of the proposed KLD-FCM approach is investigated. The findings were evaluated using 5-fold cross validation. The original dataset is divided into five equal subsets. One is used as a test set (20%), while the other is used as a training set (80%). The procedure is repeated five times, with a different test set selected each time, and the average results recorded. In terms of MAE and RMSE, the proposed method was compared to non-clustering methods such as Basic CF (BCF), User-Based CF (UBCF), SVDM (a variant of Single Value Decomposition (SVD) that uses batch learning with a learning momentum), and RSVD (Regularized SVD model). Various collaborative approaches are selected from the literature to verify the proposed method’s role in comparison to other cluster-based techniques.

**ICF(Integrated Collaborative Framework)** The merits of the item k-NN algorithm were used to propose an Integrated Collaborative Framework (ICF). This ICF also provided classification restrictions, ensuring that only potential rules are used during the collaborative filter-based categorization of user ratings.

**UPCC (user based CF Pearson correlation coefficient based CF)** For the Collaborative filtering recommender scheme, the Pearson Correlation Coefficient test is used to determine how closely two users are related.

**FCM (fuzzy C means)** The performance of User-Based Collaborative Filtering with Fuzzy C Means is compared to that of other clustering methods such as K-means and Self-Organizing Maps (SOM).

**FCMBAT** The Fuzzy C-Means and BAT-based Movie Recommendation Scheme (FCM-BAT) is an integrated Fuzzy C-Means and BAT-based Movie Recommendation Scheme for promoting efficient and collaborative recommendation to the target users. This FCM-BAT was proposed to address scalability issues and improve the clustering process, with the aim of improving the consistency of the recommendation process.

**COA(cuckoo Optimization Algorithms)** Furthermore, a possible movie recommendation system based on k-means and COA is proposed in order to improve the rate of recommendation accuracy when using the Movielens dataset.

#### 4.1 Mean absolute error

The suggested method used to measure MAE can be seen in Fig. 2 for different numbers of neighbors.

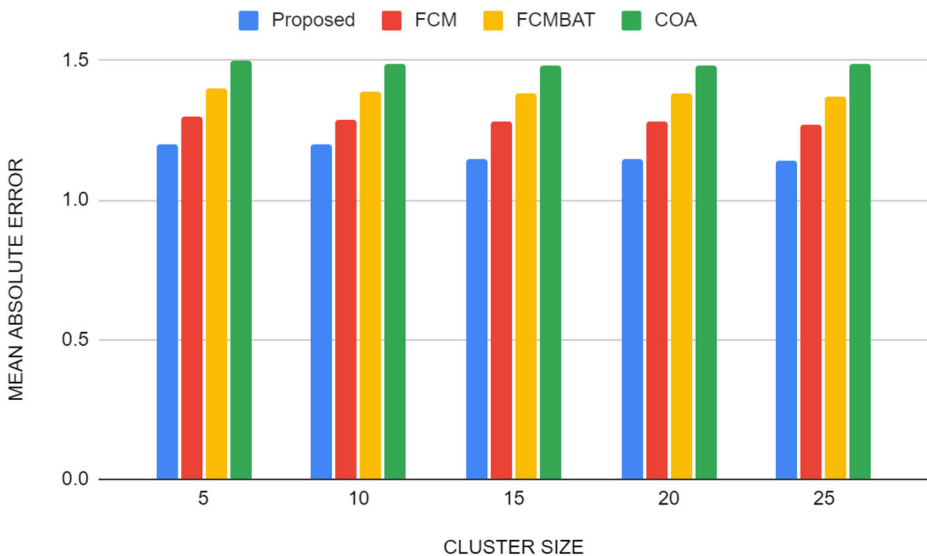


Fig. 2 MAE comparison analysis

Figures 2 and 3 show the contrast of the proposed FCM KLD with current Collaborative Recommender framework methods in terms of MAE and RMSE for different numbers of clusters. The proposed scheme’s MAE and RMSE are proving to be significantly lower than those of current schemes. As a result, the proposed scheme’s MAE is lower than the COA, FCM-BAT, FCM, and ICF approaches. The proposed scheme’s RMSE is also tested to be substantially lower than the baseline schemes under consideration.

Figure 4 graphically depicts the contrast of the proposed KLD-FCM with current Collaborative Recommender framework approaches in terms of recall. It highlights the efficiency of the proposed KLD-FCM scheme as measured by Recall for a variety of cluster sizes. The proposed KLD-FCM scheme’s Recall value is determined to be excellent as compared to baseline methods, since the KL divergence factor’s guidance in the clustering phase is responsible for the majority of progress.

Figure 5 depicts a graphic comparison of the proposed KLD-FCM with current Collaborative Recommender framework approaches in terms of accuracy. The performance of the proposed method is significantly better than that of current methods. Since it takes advantage of the advantages of KL Divergence Fuzzy C, Clustering with enhanced sqrt-cosine similarity is possible.

### 5 Conclusion

The KL Divergence Fuzzy C means Clustering with improved sqrt-cosine similarity Recommender framework (KLD-FCM) is proposed to solve the CF technique’s scalability problem. For successful clustering, Kullback–Leibler Divergence-based Fuzzy C-Means clustering is suggested, with the aim of focusing on greater accuracy during movie recommendation. The proposed KLD-FCM scheme is described as a trustworthy contribution that significantly improves the ability of movie recommendation by virtue of the KL divergence dependent Fuzzy C-Means clustering mechanism and enhanced sqrt-cosine similarity. The proposed

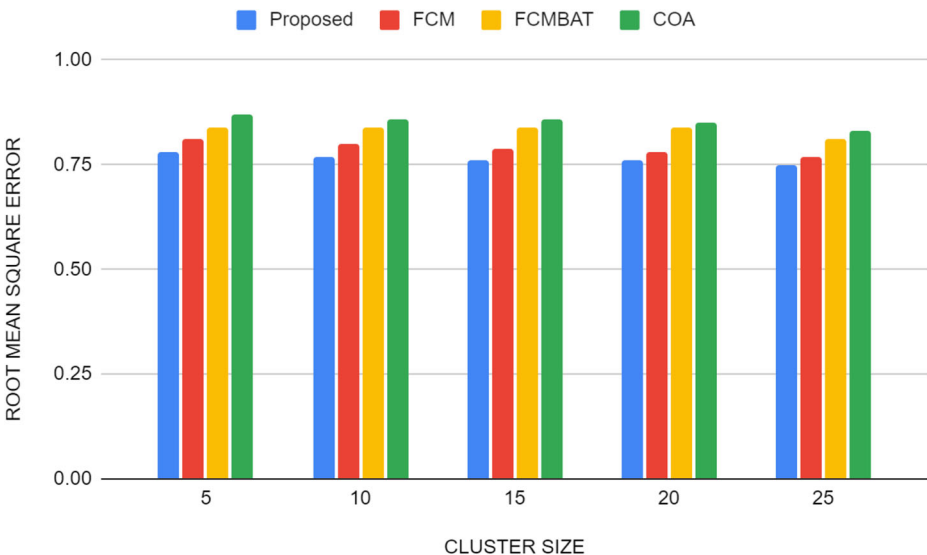
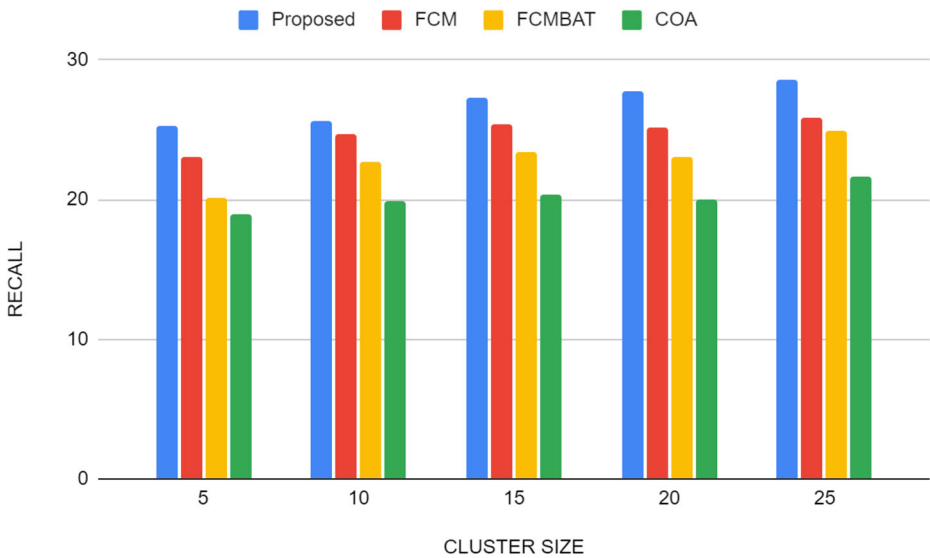
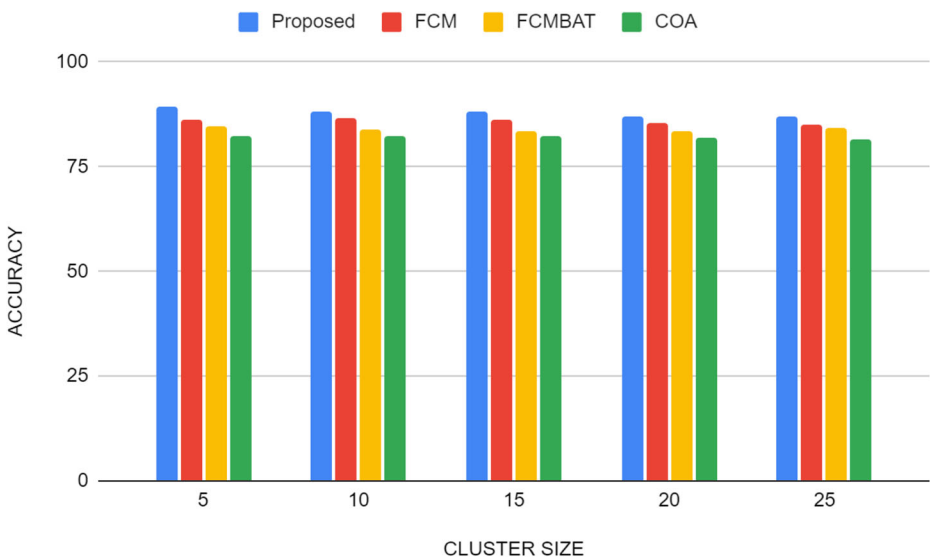


Fig. 3 RMSE comparison analysis



**Fig. 4** Recall comparison analysis

scheme emphasized and presented the critical role of the KL divergence-based cluster ensemble factor in improving clustering stability and robustness. For prediction, the enhanced sqrt-cosine similarity was used to calculate successful related neighbor users. The performance of Recommendation is improved when KLD-FCM is combined with improved sqrt-cosine similarity. The proposed KLD-FCM scheme was found to be superior in recommendation Accuracy compared to the COA, FCM-BAT, FCM and ICF approaches, as well as some non-clustering based methods considered for study, when tested on the Movielens dataset in terms of MAE, RMSE, SD, and Recall. With the specified number of clusters, it is capable of



**Fig. 5** Accuracy analysis

providing accurate and customized movie recommendation systems. In future work, the proposed design has to be tested with different datasets.

## Declarations

**Conflict of interest** The author(s) propose a clear no conflict of interest involved in this research work in form of publication in this Journal Multimedia Tools and Applications.

## References

1. Antony Vijay J, Anwar Basha H, Arun Nehru J (2021) A dynamic approach for detecting the fake news using random forest classifier and NLP. In: In computational methods and data engineering. Springer, Singapore, pp 331–341
2. Asadi E & Charkari N (2012). Video summarization using fuzzy c-means clustering. ICEE 2012 - 20th Iranian Conference on Electrical Engineering. 690–694. <https://doi.org/10.1109/IranianCEE.2012.6292442>
3. Basha SM, Rajput DS (2019). Survey on evaluating the performance of machine learning algorithms: past contributions and future roadmap. In deep learning and parallel computing environment for bioengineering systems. Academic Press, Cambridge, pp 153–164
4. Clement J. (2020). Impact of recommendation engine on video-sharing platform -YouTube. <https://doi.org/10.13140/RG.2.2.15746.50882>
5. Cui L & Dong L & Fu X & Wen Z & Lu N & Zhang G. (2016). A video recommendation algorithm based on the combination of video content and social network: CONTENT AND SOCIAL NETWORK BASED VIDEO RECOMMENDATION. Concurrency and Computation: Practice and Experience. 29: <https://doi.org/10.1002/cpe.3900>.
6. Davidson J, Liebold B, Liu J, Nandy P, Vleet T, Gargi U, Gupta S, He Y, Lambert M, Livingston B, Sampath D (2010). The YouTube video recommendation system. 293–296. <https://doi.org/10.1145/1864708.1864770>
7. De Vriendt J, Degrande N, Verhoeven M (2011) Video Content Recommendation: An Overview and Discussion on Technologies and Business Models. Bell Labs Tech J 16:235–250. <https://doi.org/10.1002/blj.20513>
8. Deldjoo Y. (2019). Enhancing video recommendation using multimedia content. [https://doi.org/10.1007/978-3-030-32094-2\\_6](https://doi.org/10.1007/978-3-030-32094-2_6)
9. Deldjoo Y, Elahi M, Quadrana M, Cremonesi P (2015). Toward Building a Content-Based Video Recommendation System Based on Low-Level Features. <https://doi.org/10.1007/978-3-319-27729-5>
10. Deldjoo Y, Elahi M, Cremonesi P, Garzotto F, Piazzolla P, Quadrana M (2016) Content-Based Video Recommendation System Based on Stylistic Visual Features. J Data Semant 5:1–15. <https://doi.org/10.1007/s13740-016-0060-9>
11. Deldjoo Y, Schedl M, Cremonesi P, Pasi G (2020) Recommender Systems Leveraging Multimedia Content. Comput Surv 53:1–38. <https://doi.org/10.1145/3407190>
12. Gupta M, Thakkar A, Gupta V, Rathore DP (2021). Movie Recommender System Using Collaborative Filtering. 978–979
13. Homann L, Martins D, Vossen G, Kraume K (2018) Enhancing traditional recommender systems via social communities. Vietnam J Comput Sci 6. <https://doi.org/10.1142/S2196888819500040>
14. Huang Y, Cui B, Jiang J, Hong K, Zhang W, Xie Y (2016). Real-time Video Recommendation Exploration. 35–46. <https://doi.org/10.1145/2882903.2903743>.
15. Kamran M, Shah SS, Baig MN, Khan RH (2020). A movie recommender system by combining both content based and collaborative filtering algorithms
16. Khadse VP, Basha SM, Iyengar N, Caytiles R (2018) Recommendation engine for predicting best rated movies. Int J Adv Sci Technol 110:65–76
17. Lu W & Chung FL (2016). Computational Creativity Based Video Recommendation. 793–796. <https://doi.org/10.1145/2911451.2914707>.
18. Mercanoglu O & Yildirim Z (2017). Video Recommendation System Using Collaborative Filtering
19. Mohamed A, Sherif A, Osama F, Roshdy Y, Hassan MA, El Ashmawi WH (2020). A new challenge on video recommendation by content. <https://doi.org/10.1109/ICCES48960.2019.9068169>.
20. Patil, Lalit. (2016). Fuzzy C means clustering MATLAB code. <https://doi.org/10.13140/RG.2.1.3924.9046>.

21. Ramezani M, Yaghmaee F (2016) A novel video recommendation system based on efficient retrieval of human actions. *Physica A: Statistical Mechanics and its Applications* 457. <https://doi.org/10.1016/j.physa.2016.03.101>
22. Shah P, Sanghvi S (2020) Video Recommender System
23. Tohidi N, Dadkhah C (2020) Improving the performance of video Collaborative Filtering Recommender Systems using Optimization Algorithm. *Int J Nonlinear Anal Appl (IJNAA)* 11:283–295. <https://doi.org/10.22075/IJNAA.2020.19127.2058>
24. Zhou X, Chen L, Zhang Y, Cao L, Huang G, Wang C (2015). Online Video Recommendation in Sharing Community. 1645–1656. <https://doi.org/10.1145/2723372.2749444>.

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.