Bulletin of Pure and Applied Sciences *Print version ISSN 0970-0765*
Zoology (Animal Science), Vol.43B, No.1s, *Online version ISSN 2320-3188*
January-June 2024: P.1035-1058

Original Research Article *Content Available online at: www.bpasjournals.com*

E-LEARNING RECOMMENDATION SYSTEM USING ENHANCED FIREFLY AND FINE-TUNED K-NEAREST NEIGHBOR ALGORITHM

D.Poornima[1], Dr. D. Karthika[2]
[1]Research Scholar, P.K.R Arts College for Women, Gobichettipalayam.
[2]Associate Professor in Computer Science, VET Institute of Arts and Science College, Erode.
Email ID: [1]poornima.snsk@gmail.com, [2]karthikad@vetias.ac.in

ABSTRACT

Nowadays, the rapid growth of e-learning platforms has led to an overwhelming amount of educational content available online. While this abundance of resources is beneficial, it can also create challenges for learners in identifying the most relevant content tailored to their needs and preferences. To resolve this problem, Recommendation Systems (RS) have become essential tools for e-learning platforms, helping to personalize the learning experience by suggesting courses, articles, and other educational materials that align with users' interests and learning goals. With the application of an improved Firefly Algorithm (FA) and a refined K-Nearest Neighbor (KNN) algorithm, this research aims to create an advanced e-learning Recommendation System (RS). The goal is to use the advantages of both classification and optimization methods to give learners individualized, high-quality recommendations. The methodology begins with pre-processing using K-Means Clustering (KMC) to effectively handle noise and identify dense data regions. Subsequently, an Enhanced Firefly Algorithm optimizes system parameters, achieving optimal fitness values (FV) for recommendation accuracy. The integration of Content-Based and Collaborative Filtering techniques by the RS to extract features and insights from data. Content-Based Filtering (CBF) by Cosine similarity (CS) focuses on item similarities and user preferences derived from item attributes, while Collaborative Filtering by Improved genetic algorithm (IGA) leverages user interactions to predict preferences based on similar users or items. These filtering methods are complemented by a classification approach using a fine-tuned K-Nearest Neighbour (KNN) algorithm adjusted by Artificial Neural Networks (ANN). The prediction accuracy, precision, recall, and reduces mean absolute error (MAE) was enhanced by this hybrid approach.

Key Words: E-learning, Recommendation System (RS), clustering, Enhanced Firefly Algorithm (FA) , Cosine similarity (CS), Artificial Neural Networks (ANN), K-Nearest Neighbour (KNN) algorithm**,** Improved Genetic Algorithm (IGA).

INTRODUCTION

The world is shifting toward an e-world, where the majority of things are digital and accessible with a single mouse click, due to the introduction of new technology and the Internet's explosive expansion. Millions of products are available for sale on the biggest e-learning platforms. For customers, it can be difficult to choose from so many possibilities. To address this issue, Recommender Systems (RS) have been developed. A customer's product interests are sent into an e-learning site's RS, which then suggests products based on the possibility of fit [1]. RS are used now to serve millions of users across hundreds of websites.

Content-Based (CB) techniques and collaborative approaches are the 2 main categories of underlying strategies used in modern RS. Typically, a CB system analyzes an extensive set of papers that each user has rated. Based on the content of these papers and the ratings they have received, the system deduces a profile from which it can suggest more relevant materials [2]. Nevertheless, collaborative strategies do not rely on the availability of textual descriptions; instead, they make recommendations for items based on the cumulative consumer reviews of those items. Finding needles in an exponentially rising haystack is one of the major research challenges of the information era, and both approaches aim to provide assistance to users in their search for items of interest. Still the E-learning system having many challenges.

Enhancing the content's scalability and Collaborative Filtering (CF)methods is initially challenging. These algorithms can examine tens of thousands of possible neighbours in actual time, however tens of millions of prospective neighbours must be searched by contemporary e-learning systems. Furthermore, individual users for whom the website has a lot of content have performance issues with the current algorithms.

Enhancing the quality of consumer recommendations is another challenge. To help them identify things they would like, consumers need suggestions they can rely on. RS users are unlikely to use the service again if they trust it, buy a product, and then find they don't like it. False Positives (FP) and False Negatives (FN) are two common forms of typical errors in RS, similar to other search systems: things that are recommended but the user finds it objectionable (FP), and products that are not recommended but the user finds it appealing (FN).

There are a lot of items on an e-learning site that a customer would intend to buy, so there's no motive to take the chance of recommending one customer won't like. FP are the most critical errors to avoid in the e-learning area because these errors will result in disappointed consumers. These two difficulties are somewhat at odds with one another since an algorithm becomes more scalable at the expense of quality the less time it spends looking for neighbours. It is crucial to address the two problems at the same time in order to ensure that any solutions found are applicable and workable.

Machine learning (ML) algorithms play a central role in RS by learning from historical data and adapting to evolving user preferences. By identifying intricate connections among consumer choices and product attributes, supervised learning techniques such as (DT) Decision Trees, Support Vector Machines (SVMs), and ensemble approaches like Random Forests (RF) optimize recommendation models. These algorithms enable systems to predict user preferences with high accuracy and recommend items that maximize user satisfaction and engagement. Existing E-learning recommendation systems often struggle with scalability and recommendation accuracy due to the inherent challenges of handling sparse data and ensuring personalized recommendations for diverse user bases. Traditional approaches may also exhibit limitations in adapting to new users or items and fail to effectively integrate contextual information to enhance recommendation relevance [3].

In many of the real-world applications, recommendation of E-learning data-sets is the point of attraction. This paper explores the pivotal role of methods and techniques in recommendation systems, highlighting their significance in overcoming challenges, improving accuracy, and shaping the future of personalized online recommendation experiences. This project focuses on developing an advanced e-learning recommendation system using an

Enhanced Firefly Algorithm (FA) combined with a fine-tuned KNN procedure. The goal is to use the advantages of both classification and optimization methods to give learners individualized, high-quality recommendations. The methodology begins with pre-processing using K-Means Clustering (KMC) to handle noise and identify dense data regions. Subsequently, the Enhanced Firefly Algorithm optimizes system parameters for recommendation accuracy. The recommendation process integrates Content-Based Filtering by Cosine Similarity (CS) and Collaborative Filtering by Improved Genetic Algorithm (IGA) to extract insights from data. Additionally, a fine-tuned KNN algorithm adjusted by Artificial Neural Networks (ANN) enhances prediction accuracy.

The research's remaining sections are organized in the following manner: Section 2 offers a concise summary of a few of the existing study in the recommendation system for e-learning. Section 3 provides specifics on the suggested technique for the proposed system. Section 4 provides the findings and a discussion of the performance analysis. Also, Section 5 summarizes the findings.

1. RELATED WORK

CF strategies are the main topic of this research. Numerous algorithms have been already described in the literature, and empirical evaluations have been conducted to assess their prospective performance.

A KMC-basedRS was presented by Zahra et al. [2015] [4] to address the scalability problems with conventional RS. Choosing the first k centroid at random in conventional KMC techniques causes erroneous recommendations and raises the expense of offline cluster training. This research demonstrates the manner in which centroid selection in k-means based RS can reduce expenses while simultaneously enhancing performance. When compared to standard centroid selection methods that select centroids at random, the suggested centroid selection technique has been shown to offer improved accuracy and performance due to its ability to utilize the basic correlation patterns. An extensive set of studies depends on 5 distinct datasets (from the movie, book, and music domain) have validated the suggested strategy. The outcomes of these studies indicates that the recommended technique converges more quickly and produces a higher-quality cluster than the previous methods, which enhances the recommendation's accuracy.

The enhanced FA is the foundation of the collaborative RS that Sharma et al. [2022] [5] suggested. Optimal clusters with useful recommendations are created using the FA method. Phase I of the suggested approach uses the FA to create clusters, while Phase II provides recommendations in real time. Apache Spark is equipped to handle Big Data (BD) due to the implementation of the FA. It is far quicker and more effective than the most advanced recommendation models because due to the combination of Apache Spark and enhanced firefly-based clustering. Various evaluation measures were employed for performance analysis and the movie-lens dataset was used for the experiments. The findings demonstrate that, in comparison to current techniques, the suggested approach produces better outcomes.

The CB Journals & Conferences RS on computer science was presented by Wang et al. [2018] [6]. Based on a document's abstract, this technique highlights and recommends appropriate publications or conferences. To maintain updated regarding the quick advances in computer science and technology, a web crawler is employed to update the learning model and the training set frequently. The efficient hybrid approach, which combines softmax regression (SR) and chi-square (FS) Feature Selection, enables interactive online response. According to the research outcomes, the algorithm can recommend the top journals or conferences in an average time of 5s, with an accuracy rate of 61.37%.

A revolutionary Deep Learning (DL) strategy that mimics a successful intelligent suggestion by predicting the users and objects was suggested by Fu et al. [2018] [7]. Initially, the semantic information describing the correlation between consumers and items is embedded in matching lowdimensional vectors of the consumers and things, which are learned separately. When a Feed-Forward NN (FFNN) is used to model the interaction among the consumers and the object The related pretrained representational vectors are used as the (NN) Neural Networks' inputs throughout the prediction step. The efficiency of the suggested approach is verified through a series of tests depends on (MovieLens 1M and MovieLens 10M) benchmark datasets. The outcomes demonstrate that the suggested model executes similar to (SOTA) State-Of-The-Art approaches on MovieLens 1M and MovieLens 10M and significantly outperforms previous approaches that utilized FFNN.

Using a Genetic Algorithm (GA), Neysiani et al. [2019] [8] presented an effective way to generate new associations rules with enhanced performance. On the MovieLens data set, evaluations were conducted. Run time, the average of quality rules, recall, precision, accuracy, and F1-measurement are the metrics of the evaluation. In the end, runtime has decreased by around 10% in the test validation of a system built using this method, which performs better than the multi objective particle swarm optimization (PSO) Association Rule Mining Algorithm (ARMA).

A method that provides generalized recommendations to each user based on the popularity and/or genre of movies was disclosed by Singh et al. [2020] [9]. KNN techniques are used in the implementation of the CB RS. Researchers have attempted to address the issues raised by the CB RS, which are also discussed in this work.

By using the KNN method and ARMA to attain higher performance, Bhagirathi and Kiran [2019] [10] assisted in resolving the issue of data sparcity. The matrix factorization procedure is another method the system employs to locate missing data. The KNN algorithm then runs on the factorized values.

2. PROPOSED METHODOLOGY

Fine-tuned KNN procedure is suggested to enhance the recommendation accuracy for the given datasets in this study. Here, Enhanced Firefly Algorithm (FA) and a fine-tuned KNN is suggested to enhance the recommendation system for e-learning more effectively. It contains main steps are such as pre-processing, clustering, optimization and e- learning recommendation. Fig. 2 displays the suggested method's general block diagram.

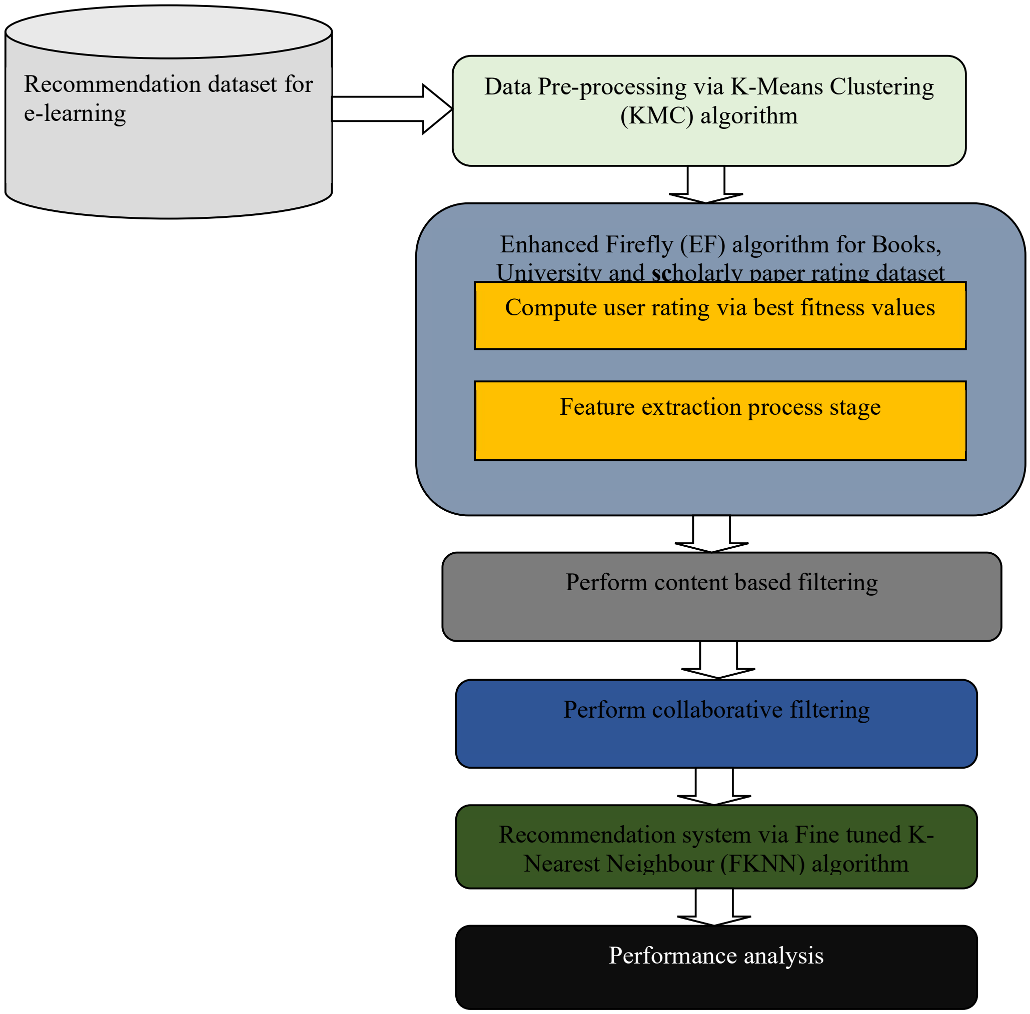

FIG 1 OVERALL BLOCK DIAGRAM OF THE SUGGESTED TECHNIQUE

a. PRE-PROCESSING VIA K-MEANS CLUSTERING (KMC) ALGORITHM

This work uses KMC algorithm in pre-processing to enhance recommendations of accuracy in datasets. Input datasets consist of books, **articles** and open university learning recommendations where KMC clustering groups comparable data from initial centroids of clusters identified by Euclidean distances. Beginning with random partitioning iterative computations are executed for (i) centers of current clusters (i.e., average vectors in data spaces for clusters), and (ii) sends information to clusters whose centers are closest. When there are no more assignments, the course is ended. This method reduces intra-cluster variances locally, which are total squares of variances among data attributes and corresponding cluster centers. Figure 3 provides an illustration of the KMC method.

Its efficiency and linear runtimes are the main advantages of KMC [11,12] implementation. One cluster per class is the maximum number that can exist. To determine the Euclidean Distance (ED), Apply the following formula and obtain the cluster centroids.

$$d(i,j) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad (1)$$

In the Euclidean n-space, the 2 points are represented as x_i and y_i.

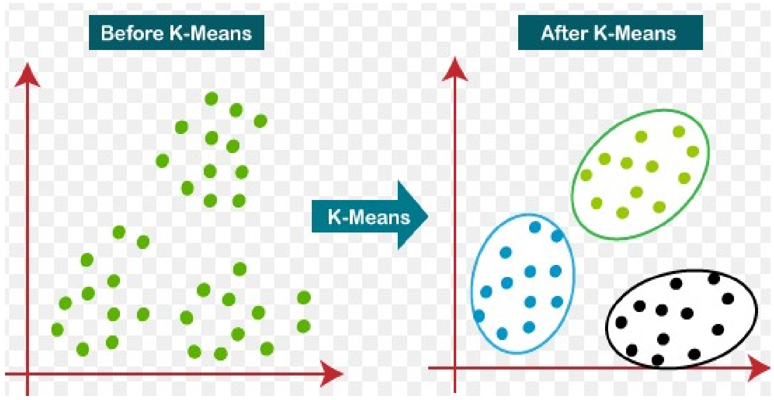

FIG 2 INSTANCE OF KMC PROCEDUREAlgorithm 1: KMC algorithm

Input: Recommendation dataset for e-learners

Output: Pre-processed data

1. From the dataset(D), choose k clusters
2. Create cluster centers µ1,…µk
3. extract k data points for alignments with cluster centers.
4. Compute cluster means and arbitrarily allocate cluster points.
5. Each data point's closest cluster center should be identified, and the noise values should be calculated using that distance.
6. Add clusters to data points.
7. Recalculate the cluster centers (cluster mean of data points).
8. Find and eliminate errors and missing values
9. On lack of fresh reassignments, stop.

The dataset does not include the cases from the original dataset that had missing characteristics. The dataset has been divided into two groups: fully complete examples with no missing values and partially complete examples with missing values from the other group. KMC is used to collect complete instances in order to form clusters of complete instances. As a result, each instance is examined separately, and any missing attributes are substituted in with potential values. The newly included instance is verified to determine whether it has been clustered in the right class after KMC has been used to the dataset from the resulting clusters. The allocated value becomes permanent in the appropriate cluster, and the procedure continues for the next instance. The next value, which puts the instance in the improper cluster, will be analysed until the right cluster, is located. Consequently, the KMC technique is applied in the pre-processing strategy may improve the recommendation classification accuracy.

b. ENHANCED FIRE FLY OPTIMIZATION ALGORITHM FOR FEATURE EXTRACTION (FE)

The AFO method is utilized in this work to efficiently for FE. FE and the AFO Fitness Function (FF) are implemented for a classification problem in this stage using AFO. AFO is utilized in this study to maximize classification accuracy by extracting just crucial features and minimizing the amount of features.

The physiological and sociological features of real fireflies served as stimulus for the FA [13]. A brief, rhythmic flash is produced by real fireflies to aid in attracting and communicating with potential mates as well as acting as a warning system for protection. This flashing behaviour is formulated using the objective function (OF) of the problem to be optimized by the FA. The flashing lights of fireflies serve as the basis for its operation. The light's intensity encourages a group of fireflies to move to bright, attracted locations, with the objective of producing the greatest resolution possible over the search area.

A few firefly features are normalized by this technique, which is demonstrated as follows:

(i) All fireflies are unisex. Therefore, sex is not a matter of importance in attracting other fireflies.

(ii) The attractiveness of the firefly is proportional to its brightness intensity, as 2 fireflies are present, the more brightness firefly attracted to one with less brightness. Then, it will travel at random because it can’t find a brighter firefly nearer.

The firefly's brightness is calculated statistically by the OF. Because it has the ability to provide the most effective solutions for MO problems, the FA is extracted. One can simply relate the brightness to the OF in a maximizing problem. For the sake of simplicity, it is believed that a firefly's brightness or (LI) Light Intensity, which is linked to the encoded OF, determines the attractiveness.

a) LI and Attractiveness at the source: According to the Inverse Square Law (ISL), the LI varies as.

$$I(r) = \frac{I_0}{r^2} \tag{2}$$

Here, $I(r)$ represents the LI at attractiveness r^2. r^2 was formed by the random assignments of pixels.

b) The light intensity while the intermediate is provided:

$$I(r) = I_0 \exp(-\gamma r) \tag{3}$$

I_0denotes the absorption coefficient of the medium.

c) To prevent the singularity, the following Gaussian form of the approximation is as

$$I(r) = I_0 \exp(-\gamma r^2) \tag{4}$$

The light intensity that nearby fireflies perceive determines how attractive a firefly is. By taking variance into account and adjusting the allocation at random, a new solution is produced. Attractive features are those that need to be ascribed to a batch of pixels. Thus, a firefly's attraction coefficient (β) is given below.

$$\beta = \beta_0 \exp(-\gamma r^m) \tag{5}$$

The attractiveness at r=0 is denoted as β_0.

The following formula is employed for computing the distance among any 2 firefly (pixels), i and j.

$$r_{i,j} = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2} \tag{6}$$

Here, the k[th] factor of the spatial match x_i of the k[th] firefly is denoted as $x_{i,k}$. A technique known as Adaptive Firefly Optimization (AFO) is produced by introducing an adaptation fuzzy parameter for both the absorption and random parameters [14]. the number of dimensions can be denoted as d. By altering the parameter linearly over the course of the iterations, these modifications enhance the capacity for both local search (LS) and global search (GS) [16]. To create the best image features for the specified database, AFO is used.

The following equation can be used to calculate the parameter α:

$$\alpha\ (x1{:}a,b,c)=\max(\min(\{\frac{x-a}{b-a},\frac{c-x}{c-b}\}),0) \tag{7}$$

Here, x1 value is used to determine the membership value, and fuzzy parameters a, b, and c. α adjusts the value based on the optimization's degree of distance divergence to improve the solution's accuracy and rate of convergence. Simultaneously, it is rewritten in order to increase population flexibility:

$$\alpha = \alpha_{\min} + (\alpha_{\max} - \alpha_{\min}) \times \frac{||x_i - x_{best}||}{L_{\max}} \tag{8}$$

$$\text{Where } L_{\max} = (x_{worst} - x_{best}) \tag{9}$$

The maximum features are represented as $\alpha_{\max}$ and minimum features are denoted by $\alpha_{\min}$. The location of the worst individual at the generation t firefly is represented by x_{worst} in Eq. (9), and the distance among the x_{worst} and the global optimal individual x_{best} is denoted by $L_{\max}$. Most of the firefly individuals are far from the globally ideal individuals in the early stages of the algorithm, and they are dispersed over the entire space. At this stage, $L_{\max}$and $(\alpha_{\max} - \alpha_{\min})$ are fixed values, and the value of $\|x_i - x_{best}\|$ is greater. Consequently, Eq. (8) demonstrated that an early stage with a greater value of α had a superior global optimization effect.

Firefly individual is attracted to fireflies that are brighter than itself and have features that are similar to the global best ones, according to the procedure's application. The value of $\|x_i - x_{best}\|$ is now reduced, which is favorable to improve the finding optimum features. Later on, firefly people would cluster around the global ideal individuals. The procedure's convergence rate is increased by varying the value of α in each iteration based on the optimal position. The aforementioned research indicates that the step size factor α varies dynamically and adaptively based on the distance among firefly individuals, hence balancing the search and algorithm development capabilities [15].

$$x_i = x_i + \beta_0 e^{-\gamma r^2}(x_j - x_i) + \alpha\left(\text{rand} - \frac{1}{2}\right) \tag{10}$$

The distance among two firefly features are x_i and x_j.

In the population, the FV of every feature is computed. In the primary generation, the FV of every firefly is determined, and the amount of pixels in a collection is assigned at arbitrary. Next, two extracted fireflies are chosen using the extraction process. The firefly, which has a brighter brightness and the best FV, is used to extract features for the following generation.

Algorithm 2: AFO

Input: Input dataset

Output: Optimal features

1. Start
2. OF (x), $x = (x1,\ldots,)T$ take greater accuracy as OF

3. Produce the fireflies for the AFO procedure. Create initial population of fireflies xi (i = 1, 2, . . . , n), light intensity Ii at xi is determined via $f(xi)$, label light absorption coefficient γ
4. Calculate the OF. The AFO procedure computes the OF to conclude the best value.
5. while (t < MaxGeneration)
6. for i=1:n all n fireflies (features)
7. for j=1:i all n fireflies (features)
8. if (Ij > Ii), change position of firefly i towards j in d-dimension;
9. end if
10. Attractiveness varies together with fuzzy function α through exp[$-\gamma r$]
11. Calculate FF by (10)
12. Calculate objective model by (6)
13. Calculate novel solutions and update LI by (3)
14. Update the optimal features by (8)
15. end for j
16. end for i
17. extract the features and determine the current best features
18. end while
19. return optimal features

c. COSINE SIMILARITY (CS) BASED CONTENT-BASED FILTERING

In RS, CBF with the CS technique is a common method. On the basis of the similarity between objects, recommendations are sent to the user [16, 17]. CBF makes recommendations for things based on the user's preferences and their features. It looks for similarities by depending on the features of the objects. The resemblance between a target set of data and a provided set of data can be ascertained using the CS technique. It computes the cosine angle among the two vectors in a multidimensional space and compares the similarity of two documents based on how different they're in size. In this work, utilize CS as a computational metric to ascertain if a given term appears in an item or not.

The degree of similarity among the relevant item and the active item linked to a particular user is then taken into consideration when formulating recommendations. Higher degrees of similarity exhibited to the active item improve the possibility of recommending it to a user.

By utilizing CS, one can perform extensive vector-based analysis and comparison by considering the dataset's data items as vectors.

The cosine of the angle formed by two non-zero vectors in a multidimensional space is measured by CS. The similarity among two text documents or item Feature Vectors (FV) is frequently determined using this method. The formula for calculating the CS among 2 vectors, like A and B, is expressed as:

$$\text{Cosine Similarity}=\frac{A\cdot B}{\|A\|\|B\|} \tag{11}$$

Here:

- The 2 vectors as A and B, thus the dot product of those vectors can be denoted as $A \cdot B$.

- The magnitudes (or lengths) of vectors A and B is denoted as $||A||\ ||B||$.

Dot Product (A·B**)**: This represents the total of the products between the matching items in the two numerical sequences. Given 2 vectors, A and B, the dot product can be computed as follows:

$$A\,.\,B=\sum_{i=1}^{n} A_i B_i \tag{12}$$

The vector component of those 2 vectors: (A and B) are represented as A_i and B_i.

Magnitude (|| A || and || B ||**)**: A vector's length can be determined by its magnitude. It is computed as follows:

$$||A|| = \sqrt{\sum_{i=1}^{n} A_i^2} \tag{13}$$

Similarly,

$$||B|| = \sqrt{\sum_{i=1}^{n} B_i^2} \tag{14}$$

The CS is employed to compute the similarity among products or user profiles represented as feature vectors. Here’s a step-by-step guide in content-based filtering,:

1. **Vector Representation**: Represent each item and user profile as a vector of features.
2. **Calculate Dot Product**: Compute the dot product of the item vectors(eq.12).
3. **Compute Magnitudes**: Compute the magnitudes of the vectors (eq.13,14).
4. **Compute Cosine Similarity**: Apply the CS formula to get the similarity score.

Use the final data generated by the CB method as input for CF. This can be done by considering the items recommended by content-based filtering as additional interactions in the user-item matrix used for collaborative filtering.

d. COLLABORATIVE FILTERING BY IMPROVED GENETIC ALGORITHM:

The real-world (DM) Decision-Making process, which involves getting advice on unfamiliar circumstances or objects from friends and acquaintances and making an informed decision, served as the model for CF algorithms [18, 19]. There are several steps involved in the CF prediction process. The users are first evaluated based on how similar they are to one another. Then, by combining the rates of other comparable customers for that same thing, the predicted score for an item by an active user is calculated.

Using the data information, a unique GA-based strategy is developed to locate customers who match characteristics to the target customer. The relevance weights of the neighbors are provided by the suggested technique known as the IGA. The recommendation algorithm will then make use of the users and the related weight. To determine the important values of the chosen users, an evolutionary approach is employed. In order to do this, a group of comparable users and the corresponding effects they have on the target user represent each solution. After initializing a set of solutions at random, the genetic operators are applied to evolve the solutions.

Until a specific level of fitness is attained or a predetermined number of offspring are produced, this process is repeated. Algorithm 1 provides the IGA pseudo-code. Each step's specifics are given in the section that corresponds to it.

3.3.1. Fundamentals of GA

The strongest stochastic algorithm based on the principles of natural selection and natural genetics by John Holland first published GA [20]. It is implemented to ML and optimization applications with great efficacy. For the purpose of determining an effective method for an issue, a GA probabilistically modifies a population of individuals by utilizing some genetic procedures, such as crossover (CO), mutation, and selection. Finding a solution that approaches the optimal solution is the objective.

a) Initial population

The initial population consists of randomly generated chromosomes. Each chromosome's feature subset, or collection of candidate features, indicates a single solution in the (SS) Search Space. Each chromosome ha n binary vectors, the selected feature is indicated by a bit with a value 1 and a bit with the value "0" denotes the feature that has not been selected.

b) Fitness function

When selecting the feature subsets, the FF, which evaluates the performance obtained with the candidate feature, is important, which have to be added in the upcoming generation (new offspring). The subgroups that perform the best will be more probable to be selected for and reproduce in order to create future generations.For determining the fitness function and evaluating the candidate feature according to this research, the size of the feature subset is chosen; the approach below uses the fitness function:

$$\mathrm{Fit} - \mathrm{Fun}(\mathrm{S_i}) = \mathrm{Z} * \mathrm{C}(\mathrm{S_i}) + (1 - \mathrm{Z})(1/\mathrm{size}(\mathrm{S_i})) \tag{15}$$

Here, the selected feature subset is denoted by $(\mathrm{S_i})$, and the macro-average F-measure of its size $(\mathrm{S_i}))$ is indicated by , $\mathrm{C}(\mathrm{S_i})$.Z is a parameter with a range of 0 to 1, and N is the quantity of features included in this subset. while classification ability improves, the appropriateness of the feature set Si grows, and while $\mathrm{S_i}$ size increases, it decreases.The higher Z value suggests that the classification performance is highly significant when compared to the size of the $(\mathrm{S_i})$. In the current method, the Z value is set to 80%.

c) Selection

Following the evaluation of the subsets, the selection process is carried out, using the applied selection procedure to choose the subgroups based on fitness. The strategy that is commonly employed with the GA is called RWS (Roulette Wheel Selection). The reproducible subset is created by this technique based on the subset-proportional probability selection FF (Fitness Function), which is inversely related to the fitness of other subsets in a verified population. The probability of choosing a subset Si for reproduction can be found using the following formula.

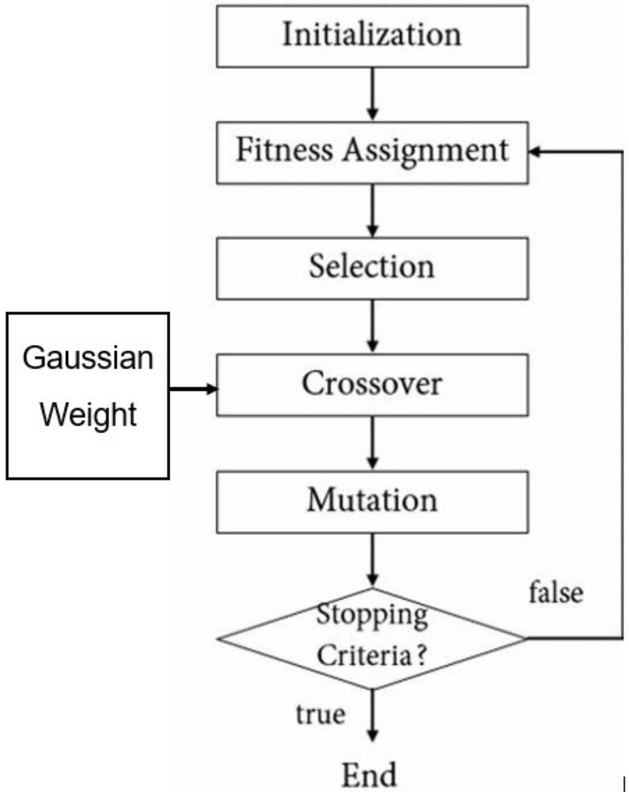

FIG.3. IMPROVED GENETIC ALGORITHMCrossover

A new generation of subsets (children) from the chosen subsets (parents). For the population to remain diverse, these surgeries are essential was generated by CO and mutation operators are reproduction operations. By switching the properties of the two parent subsets, two new subsets are created using the commonly utilized crossover strategy. During the crossover procedure, a crossover rate or probability is used to determine the likelihood that two parents' features will change. It comprises exchanging all features that are next to an arbitrary point I that is chosen anywhere along the parent subsets. So, one subset is used to choose the first I attributes, while another subset is used for the remaining attributes.

The suggested modification to this CO method divides each parent of the chosen parents into two equal parts.

Summing the weight values of the features within for determining the weight of each segments, through utilizing this procedure that considers document frequency and feature frequency. The computation of the weight assigned to the attributes has already been finished during the data preprocessing step. Next, compute the cumulative feature weight based on the weight. Determine the weight by sorting the features of every class in every subset of chromosomes according to that weight. Next, compute the cumulative feature weight based on the weight. Determine the weight by sorting the features of every class in every subset of chromosomes according to that weight.

Following this, the top two segments from each of the two parents are combined to generate a new feature subset (a new child), and the remaining two segments are used to build a second subset (second child). Consequently, in this method, the IGA search is directed towards the best of the subgroups using feature weight information.

d) Mutation

In this mutation procedure, a single feature subset is employed. Then, the mutation rate determines how many more feature subsets that needs to be altered. The subset that has to be modified allows for the modification of any number of feature values. In this procedure, only some of the features in the modified subset are used, with features selected from the best subsets collected in the preceding generation. A subset of features with low weight is chosen in order to produce a novel set of results and those features are replaced with the most significant features from the best obtained feature subset that are not included in the subset. Additionally, the source of the feature subset (child) that requires mutation is identified, and the cumulative accuracy of the real parents is calculated.

Starting with a set of randomly chosen optimal responses for the given problem, by the genetic algorithm (GA) and then let them evolve through a sequence of stochastic operators, namely crossover, mutation, and selection. The method depends on making an unbiased fit (function) assessment. The response with the highest FV is the more effective one. To find the FV, only theOF needs to be assessed and the thresholds need to be satisfied. It is not necessary for gradient information. The application of this type of derivative-freeness technique increases the generality and capability of the GA to deal with problems involving complex objective functions where obtaining derivatives is either impossible or very difficult.

The drawback of the optimization depends on gradients strategy of sliding into local optima is minimized due to the stochastic and random nature of the GA.

3.3.2. Gaussian weight function (GWF) based GA

The user can eliminate the effort of selecting the sensitive parameter(s) contained in the GA unlike simple genetic algorithms (SGA), and the performance of the GA can be increased to enhance the possibility of reaching the global optimum, we propose new GWF techniques in the GA for the crossover operators. Calculate the probability P(i) of selecting individual iii based on its Gaussian weight.

$$P(i) = \frac{\omega(i)}{\sum_{j=1}^{N} w(j)} \tag{16}$$

During the evolutionary process, the GWF technique can adapt themselves. Thus, any pre-established parameters for the algorithm are required. Furthermore, because of its GWF features, the crossover operator may execute the scalable point crossover.

A more accurate and personalized RS can be achieved by using CF to enhance the initial recommendations made by CBF [21] based on item features. This refined data to identify the nearest neighbours and generate the final recommendations step.

a. Finetuned KNN:

The extracted features are used as an input for recommendations. The dataset is effectively classified by the basic KNN Classifier technique based on the similarity with neighbors. K stands for the quantity of data that are collected for the recommendation process. The class that uses highly representative features for separation is identified, and the K closest attribute is chosen from the classified training data in the dataset that is provided. In high-dimensional data, the distances between points become increasingly similar, making it difficult for KNN to distinguish between nearby and distant points. This can lead to poor performance. Artificial Neural Networks (ANNs) can help overcome these issues in several ways. ANNs can learn and extract relevant features from high-dimensional data. Through multiple layers of neurons, they can transform the input data into a more manageable representation where important features are highlighted, and irrelevant ones are suppressed.

ANN algorithm

Utilizing learning, ANNs are used to acquire knowledge. The Input Layer (IL), Hidden Layer (HL), and Output Layer (OL) are the 3 stages of an ANN. 'n' number of inputs are created after processing the IL's collection of input data features. A set of weights is used to direct these procedures. In NN, weights are the information that helps in problem solving. It transfers information from the IL to the OL in the HL after some advantageous hidden extraction. Here, the balanced dataset categorization is done using ANN. Using ANN to train the balanced dataset, classify the features, and evaluate the results.

The ANN is improved with Multilayer Perceptron (MLP) via sigmoid function which is called as ANN [22]. Fig 3 shows the ANN architecture

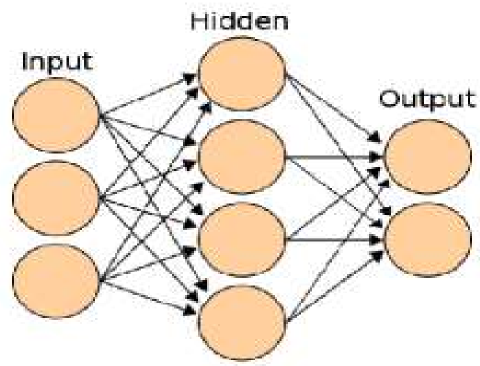

FIG 4. ARCHITECTURE OF ANN Input Layer - Information from books, open universities, and scholarly papers that is fed into the network is carried via the IL. At first, this knowledge is slightly raw.

Hidden Layer – The primary function of the HL is to transform the input layer's raw dataset data into a format that the output layer is able to utilize. There can be one or more HL in an ANN architecture.

Output Layer: Higher classifier accuracy and faster execution times are the outcomes of processing the data provided to the OL from the HL.

MLP, a FNN framework with a cascade-wise neuronal arrangement, is the most widely used model. MLP comprises 2 layers minimum. The ith layer's outputs are the inputs to the i+1th layer's neurons in MLPs. And no DT among neurons inside a layer. The total count of output classes is indicated by the amount of nodes in the OL, the IL's node count reflects the number of features present in the input vector.

The following eqn presents the nodes m in the HL and n in the OL are represented by wnm, the connective weight among them is v_{ml}, the connective weight between them is l in the IL and m in the HL, and the bias terms or threshold of the transfer function f of the nodes m in the HL and n in the OL are represented by θ_{vm} and θ_{wm}.

$$Y_n = f(\sum_{m=1}^{h} (w_{nm}, f(\sum_{l=1}^{i} v_{ml} X_l + \theta_{vm}) + \theta_{wm} \tag{17}$$

$n = 1, \ldots\ldots, o$

One of the main benefits of utilizing ANN is that it doesn't make any assumptions about the class distribution. When the weighted total of the inputs exceeds a threshold value that may be adjusted, also known as an (AF) Activation Function, an ANN's perceptron model outputs 1. Every neuron produces a weighted total of its inputs, including bias. "W" and "x" stand for the input neuron and weights.

$$\sum_{i=1}^{m} \text{bias} + (w^i x^i) \tag{18}$$

The Sigmoid function (SF) is one of the functions that the AF employs.

$$f(x) = \text{sigmoid} = \frac{1}{1 + \exp(-x)} \tag{19}$$

The network weights are made up of each neuron's bias term and connection weights. Updates to the network weights and the precise calculation of weights and biases are called NN training. It is claimed to be the most efficient way to extract the desired output from the input.

Algorithm 3: ANN

Input: Selected features

Output: Better classification results for balanced dataset

1. Procedure ANN (input, neurons, repeat)
2. Create input database
3. Input←database with all possible combinations
4. Train ANN
5. For input = 1 to end of input do
6. For neurons =1 to n do
7. For repeat = 1 to n do
8. Train ANN
9. ANN-storage ←save value with highest accuracy features
10. End for
11. End for
12. ANN-storage←save best prediction of ANN based on inputs
13. End for
14. Return ANN-storage → Result with best classification of ANN for every feature combinations

Combining ANN with K-Nearest Neighbors (KNN) can create a powerful hybrid model that leverages the strengths of both algorithms. In this approach, an ANN is used to learn complex feature representations from the feature data. These learned features are then fed into a KNN classifier for the final recommendation.

KNN algorithm

Using training tuples that are similar to a particular test set and comparing them with each other is the basis of (NN) Nearest Neighbor classifiers, based on learning by analogy. As a non-parametric method, KNN has been applied to statistical estimation and (PR) Pattern Recognition. In SL, KNN is considered to be a technique where training sets are used to help classify points for particular groups [23]. Consider the set of data points (Xi, Ci) here i = 1, 2..., n. Xi represents feature values, while Ci shows the labels associated with each i in Xi. n features explain the training tuples. In an n-dimensional space, each tuple represents a point.

A KNN classifier identifies the k training tuples that are most similar to the unknown tuple when it is given an unknown tuple by searching the pattern space [24]. The k "nearest neighbors" of the unknown tuples are these k training tuples. A distance metric, like ED, is used to define close proximity. It is defined as follows:

$$(X_1, X_2) = \text{sqrt (sum ((xj-xij) ^2)} \quad (20)$$

Here, new point can be denoted as X, existing point across among all input features j can be denoted as xi

KNN Steps

1. Start
2. {
3. Read output of ANN
4. Using ED, determine the K training instances are closer to the unknown class instance.
5. Find top FKNN values
6. The instances are sorted to the NN based on distance
7. Choose the most commonly occurring K instance values
8. }
9. End

After obtaining the output from the fine tuned K-Nearest Neighbors (FKNN) model, the following stage is to assess the efficiency of the model. Performance evaluation helps in understanding how well the model is performing on the given task. Here are the typical steps and metrics used for evaluating the performance of a KNN model.

3. EXPERIMENTAL RESULT

The experiment used 3 E-learning datasets, namely books, open university, and academic paper learning suggestions. The book dataset is taken from the following link https://www.kaggle.com/datasets/dilaaslan/biggest-ecommerce-bestselling-books-dataset. The open university dataset is taken from the following link: https://www.kaggle.com/datasets/rocki37/open-university-learning-analytics-dataset?select=studentAssessment.csv. **Sc**holarly paper is taken from the following link: https://www.kaggle.com/datasets/tayorm/arxiv-papers-metadata. In this work, these three datasets are evaluated using existing CF and TSVM and ICSO-TSVM and proposed FKNN algorithms. Accuracy, precision, recall, MAE, and time complexity are the performance parameters that are taken into consideration.

The book dataset describes in figure 5, Academic paper dataset in excel sheet denoted in figure 6 and Open university dataset denoted in figure 7. Each dataset contains unique and relevant information that is crucial for the study, aiding in comprehensive data analysis and interpretation.

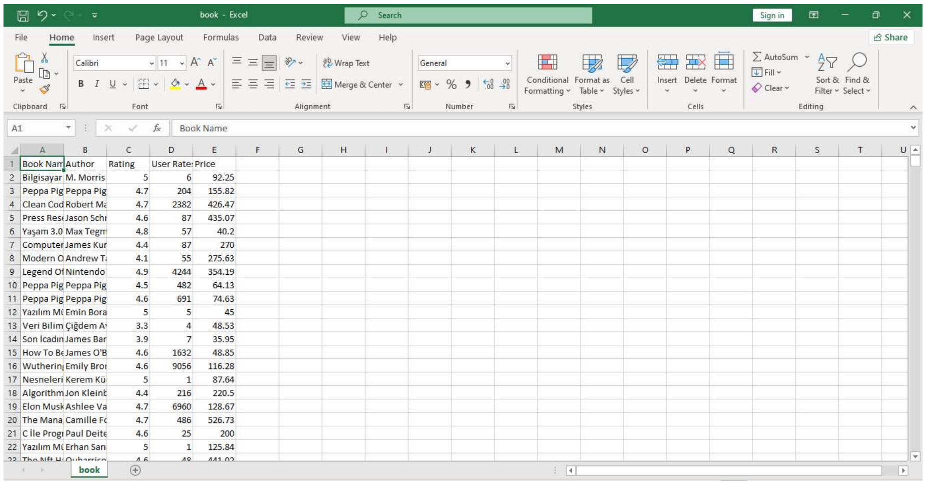

FIG 5.BOOK DATASET MODEL

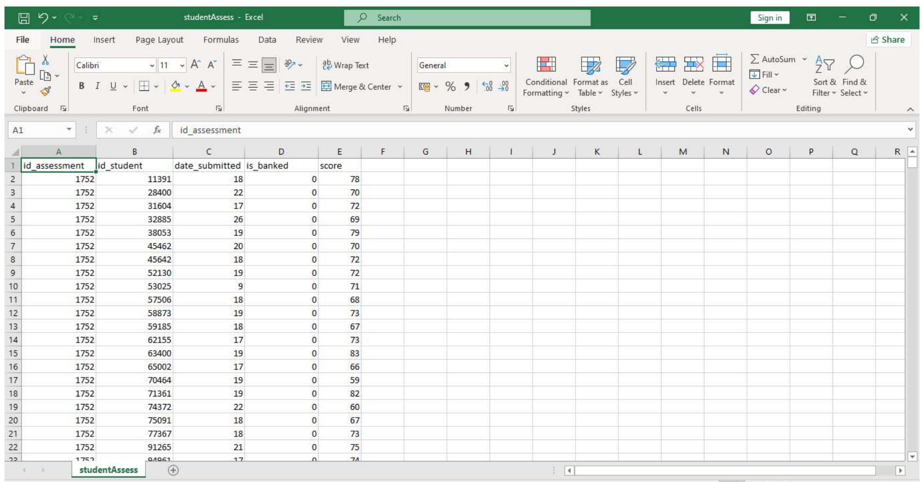

FIG 6. ACADEMIC PAPER DATASET

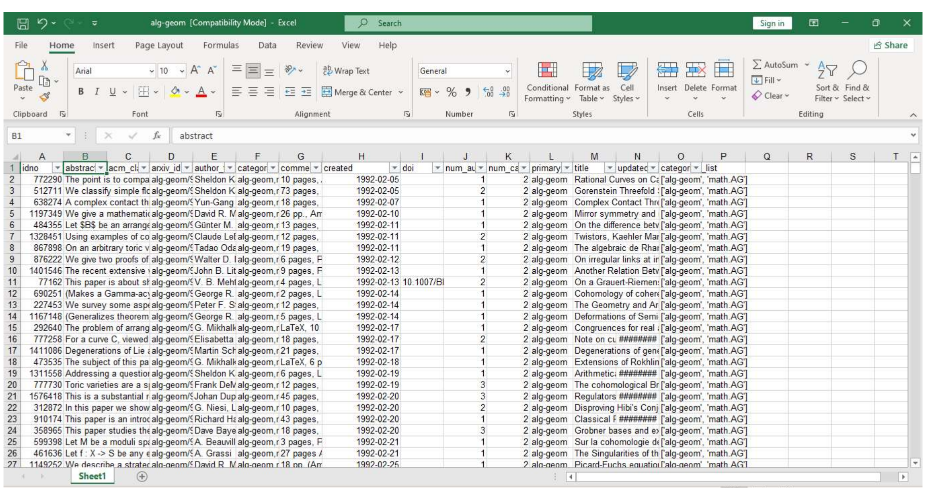

FIG 7. OPEN UNIVERSITY DATASETAccuracy

By dividing the total true classification parameters $(T_p + T_n)$ by the sum of the classification parameters $(T_p + T_n + F_p + F_n)$, the overall accuracy of the framework is determined The calculation of accuracy is done as follows:

$$\text{Accuracy} = \frac{T_p + T_n}{(T_p + T_n + F_p + F_n)} \quad (20)$$

Here, False Negative can be denoted as F_n , False Positive is denoted as F_p, True Positive is denoted as T_p , True Negative is denoted as T_n.

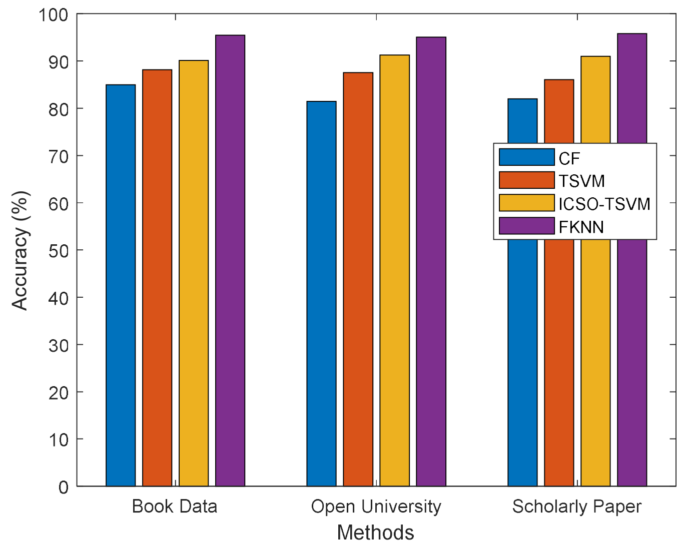

Fig 8 Accuracy

It is evident from Fig. 8. The comparative evaluation of both current and suggested techniques in terms of accuracy are presented. The datasets and approaches are taken for the x-axis, and y-axis represents the accuracy . For the supplied books, open university, and academic article datasets, the suggested FKNN approach gives more accuracy than the current methods, such as centralized CF, TSVM, and ICSO-TSVM algorithms, which provide lesser accuracy. By using a pre-RS, content and CF techniques are applied to improve classification accuracy. The outcome thus indicates that the optimal recommendation generated by the suggested FKNN technique improves the dataset accuracy.

Precision

Equation 21 can be employed to determine precision,

$$\text{Precision} = \frac{T_p}{T_p + F_p} \quad (21)$$

Precision is a computation of correctness or quality, whereas recall is a measure of accuracy or the sum. When an algorithm achieves high precision, it usually indicates that a bigger percentage of pertinent outcomes than irrelevant ones were returned by the system.

A classification problem in a class is found by dividing the total amount of elements classified as belongs to the positive class by the number of TP.

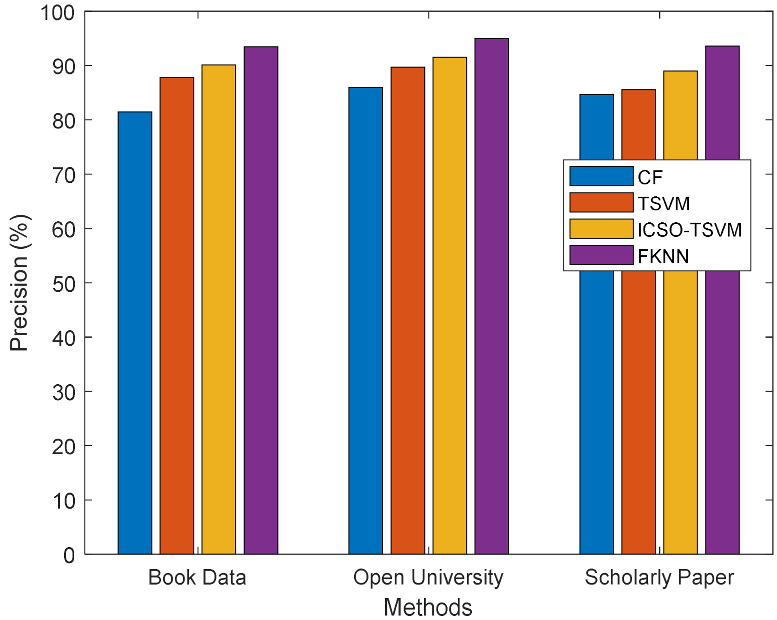

Fig 9 Precision

It is evident from Fig. 9. The comparative evaluation of both current and suggested techniques in terms of precision are presented. The methods are represented on the x-axis. Y-axis represents the precision value. For the three datasets provided, the suggested FKNN method yields more precision than the current approaches, including CF and TSVM and the ICSO-TSVM algorithm, which provide lower precision. The suggestion model in the suggested approach increases precision. The suggested FKNN technique improves classification performance by using the best recommendation step, according to the results.

Recall

The following equation is used to calculate recall,

$$\text{Recall} = \frac{T_p}{T_p+F_n} \tag{22}$$

The quantity of pertinent papers found by a analysis divided by the total amount of relevant papers already in existence is known as recall, Conversely, precision is defined as the ratio of the quantity of pertinent papers identified during a analysis to the total number of papers identified during that analysis.

Following shows a representation of the comparison graph:

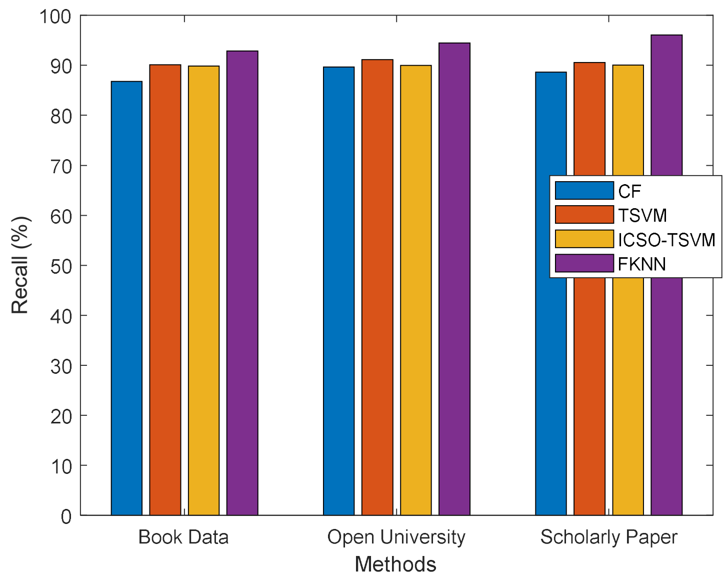

Fig 10 Recall

It is evident from Fig. 10. The comparative analysis of both the suggested and the current techniques in terms of recall are presented. The methods are represented on the x-axis, while the recall value is plotted on the y-axis. For the supplied datasets, the recommended FKNN procedure offers higher recall than the existing approaches, including CF, TSVM, and ICSO-TSVM techniques. It improves the stability of training performance. Therefore, by balancing the E-learning dataset, FKNN enhances performance.

MAE Analysis

The MAE is calculated by summing together all the N's errors, which are proportional to evaluating forecast matches and then determining normal value. The MAE measure is calculated as follows for each combination *pi* and *qi* of anticipated assessments *pi* and actual evaluations *qi*:

$$MAE = \frac{\sum_{i=1}^{N} |p_i - q_i|}{N} \tag{23}$$

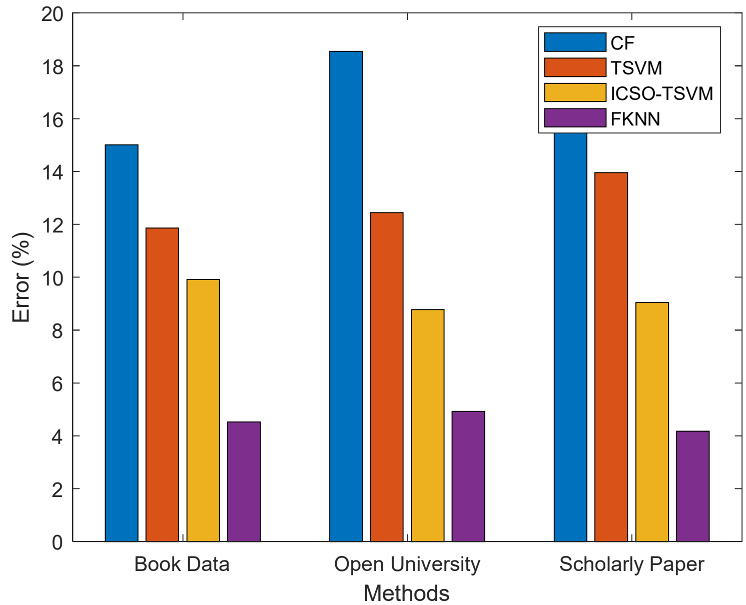

Fig 11 MAE

The contrast metric is evaluated using the most recent methods, as shown in Fig. 11. Then, the x-axis represents the Methods, the y-axis represents the MAE values. While the current CF, TSVM and ICSO-TSVM approach offers greater MAE, the suggested FKNN approach offers a lower MAE. The precision will be greater with a low MAE. The planned technique outperformed the current approaches in terms of output. It has been discovered that the MAE lowers as the number of students rises. The findings indicate that the provided FKNN technique greatly boosts the effectiveness of the recommendation system.

Computation time

The algorithm is better when it provides lower time complexity

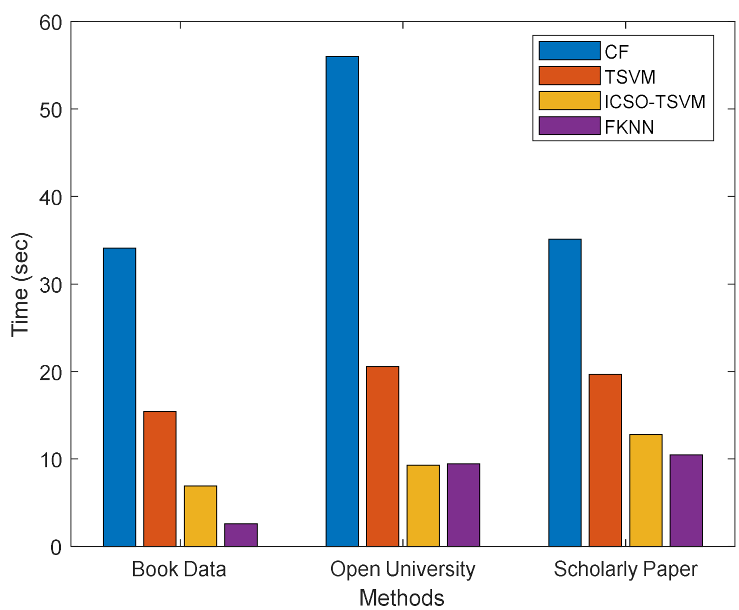

Fig 12 Execution time

Then the comparison of analysis based on execution time, as represented in Fig. 12, utilizing both the recommended and current approaches. The execution time is plotted on the y-axis, while the approaches are plotted on the x-axis.For the provided recommendation dataset, the suggested FKNN algorithm executes more quickly than the current methods, such as CF , TSVM and ICSO-TSVM algorithms. The findings suggest that the FKNN decreases recommendation execution time by recommendation learner behaviours.

4. CONCLUSION

This study proposes an advanced e-learning recommendation system using a Fine-Tuned K-Nearest Neighbour (FKNN) algorithm, demonstrating significant improvements in recommendation accuracy, precision, recall, and time complexity. The system's architecture integrates essential modules: pre-processing to enhance dataset quality, K-Means clustering to group similar learners, the Enhanced Firefly Algorithm (EFA) for feature selection and parameter tuning, and a combination of CBF and CF for personalized recommendations. The incorporation of FKNN and Artificial Neural Networks (ANN) in the classification module has shown superior performance over existing methods. The outcomes of the experiment confirm that the suggested FKNN model is capable of offering e-learners precise and effective recommendations. The study also outlines future research directions, including the implementation of DL algorithms, real-time data processing, and hybrid optimization techniques to further enhance recommendation performance. With the help of these developments, e-learning systems could become much more effective and personalized, offering students individualized and pertinent course material.

REFERENCES:

1. Gupta, S. and Dave, M., 2020. An overview of recommendation system: Methods and techniques. *Advances in Computing and Intelligent Systems: Proceedings of ICACM 2019*, pp.231-237.
2. Hwangbo, H., Kim, Y.S. and Cha, K.J., 2018. Recommendation system development for fashion retail e-commerce. *Electronic Commerce Research and Applications*, *28*, pp.94-101.
3. Singh, P.K., Pramanik, P.K.D., Dey, A.K. and Choudhury, P., 2021. Recommender systems: an overview, research trends, and future directions. *International Journal of Business and Systems Research*, *15*(1), pp.14-52.
4. Zahra, S., Ghazanfar, M.A., Khalid, A., Azam, M.A., Naeem, U. and Prugel-Bennett, A., 2015. Novel centroid selection approaches for KMeans-clustering based recommender systems. *Information sciences*, *320*, pp.156-189.
5. Sharma, B., Hashmi, A., Gupta, C. and Jain, A., 2022. Collaborative recommender system based on improved firefly algorithm. *Computación y Sistemas*, *26*(2), pp.537-549.
6. Wang, D., Liang, Y., Xu, D., Feng, X. and Guan, R., 2018. A content-based recommender system for computer science publications. *Knowledge-based systems*, *157*, pp.1-9.
7. Fu, M., Qu, H., Yi, Z., Lu, L. and Liu, Y., 2018. A novel deep learning-based collaborative filtering model for recommendation system. *IEEE transactions on cybernetics*, *49*(3), pp.1084-1096.

8. Neysiani, B.S., Soltani, N., Mofidi, R. and Nadimi-Shahraki, M.H., 2019. Improve performance of association rule-based collaborative filtering recommendation systems using genetic algorithm. *International Journal of Information Technology and Computer Science*, *11*(2), pp.48-55.
9. Singh, R.H., Maurya, S., Tripathi, T., Narula, T. and Srivastav, G., 2020. Movie recommendation system using cosine similarity and KNN. *International Journal of Engineering and Advanced Technology*, *9*(5), pp.556-559.
10. Bhagirathi, Y.N. and Kiran, P., 2019. Book Recommendation System using KNN Algorithm. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *2*(6), pp.336-339.
11. Andra, D., 2022. E-commerce Recommender System Using PCA and K-Means Clustering. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, *6*(1), pp.57-63.
12. Bandyopadhyay, S., Thakur, S.S. and Mandal, J.K., 2021. Product recommendation for e-commerce business by applying principal component analysis (PCA) and K-means clustering: benefit for the society. *Innovations in Systems and Software Engineering*, *17*(1), pp.45-52.
13. Koosha, H.R., Ghorbani, Z. and Nikfetrat, R., 2022. A clustering-classification recommender system based on firefly algorithm. *Journal of AI and Data Mining*, *10*(1), pp.103-116.
14. Perumal, S.P., Sannasi, G. and Arputharaj, K., 2020. REFERS: refined and effective fuzzy e-commerce recommendation system. *International Journal of Business Intelligence and Data Mining*, *17*(1), pp.117-137.
15. Madhavan, S.S., Babu, M.R., Venkatesh, S. and Mohan, M.M., 2022. Enactment of Firefly Algorithm and Fuzzy C-Means Clustering For Consumer Request and Demand Prediction.
16. Shahbazi, Z. and Byun, Y.C., 2019. Product recommendation based on content-based filtering using XGBoost classifier. *Int. J. Adv. Sci. Technol*, *29*, pp.6979-6988.Liao, M. and Sundar, S.S., 2022. When e-commerce personalization systems show and tell: Investigating the relative persuasive appeal of content-based versus collaborative filtering. *Journal of Advertising*, *51*(2), pp.256-267.
17. Nurcahya, A. and Supriyanto, S., 2020. Content-based recommender system architecture for similar e-commerce products. *Jurnal Informatika Ahmad Dahlan*, *14*(3), pp.90-101.
18. Jena, K.K., Bhoi, S.K., Malik, T.K., Sahoo, K.S., Jhanjhi, N.Z., Bhatia, S. and Amsaad, F., 2022. E-learning course recommender system using collaborative filtering models. *Electronics*, *12*(1), p.157.
19. Cui, Z., Xu, X., Fei, X.U.E., Cai, X., Cao, Y., Zhang, W. and Chen, J., 2020. Personalized recommendation system based on collaborative filtering for IoT scenarios. *IEEE Transactions on Services Computing*, *13*(4), pp.685-695.
20. Stitini, O., Kaloun, S. and Bencharef, O., 2022. An improved recommender system solution to mitigate the over-specialization problem using genetic algorithms. *Electronics*, *11*(2), p.242.
21. Singh, M.K. and Rishi, O.P., 2020. Event driven recommendation system for E-commerce using knowledge based collaborative filtering technique. *Scalable Computing: Practice and Experience*, *21*(3), pp.369-378.

22. Madhuri, J. and Indiramma, M., 2021. Artificial neural networks based integrated crop recommendation system using soil and climatic parameters. *Indian Journal of Science and Technology*, *14*(19), pp.1587-1597.
23. Amer, A.A., Abdalla, H.I. and Nguyen, L., 2021. Enhancing recommendation systems performance using highly-effective similarity measures. *Knowledge-Based Systems*, *217*, p.106842.
24. Patro, S.G.K., Mishra, B.K., Panda, S.K., Kumar, R., Long, H.V., Taniar, D. and Priyadarshini, I., 2020. A hybrid action-related K-nearest neighbour (HAR-KNN) approach for recommendation systems. *IEEE Access*, *8*, pp.90978-90991.