

Research Article

Interest-Based Content Clustering for Enhancing Searching and Recommendations on Smart TV

Malang Jan,¹ Shah Khusro ,¹ Iftikhar Alam ,² Inayat Khan ,³ and Badam Niazi ⁴

¹Department of Computer Science, University of Peshawar, 25000, Pakistan

²Department of Computer Science, City University of Science and Information Technology, Peshawar 25000, Pakistan

³Department of Computer Science, University of Buner, Buner, 19290, Pakistan

⁴Department of Computer Science, University of Nangarhar, 2600, Afghanistan

Correspondence should be addressed to Badam Niazi; niazi5.48@gmail.com

Received 12 February 2022; Accepted 12 April 2022; Published 5 May 2022

Academic Editor: Muhammad Imran

Copyright © 2022 Malang Jan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Smart TV has become a pervasive device due to its support for numerous entertainment options. These capabilities of smart TV make it attractive for viewers and researcher. Besides, a plethora of multimedia content continues to grow, which makes searching and browsing the desired content a difficult, time-consuming, and contributes to cognitive overload problem. In the case of smart TV, making clusters of the related content based on user's interest is among the best solutions. In this connection, this study proposed a dynamic approach for clustering the TV-related online multimedia content and presenting them in a manageable format on smart TV to mitigate the issue of searching and relevant recommendations. We collected and clustered the content from diverse data sources based on the viewer's interest. This further recommends novel content to the viewers without social metadata, such as rates, tags, which is normally insignificant in for smart TV viewership due to its shared nature. We used bisecting K -means, Lingo, and Suffix Tree Clustering (STC) algorithms. A comparative analysis of these algorithms and suitability in the context of smart TV is also presented. Results show that the proposed approach enhances search results and recommends relevant content based on user's interests.

1. Introduction

In this digital era, the popularity of smart TV is increasing day by day. Smart TV is a device that have traditional flavour of television along with an operating system and support internet connectivity for streaming services. This smart TV have changed the entertainment paradigm in many folds [1]. The statistics show that almost 94.2 million people are using smart TV in the United States (<https://www.statista.com/statistics/718737/number-of-smart-tv-users-in-the-us/>) only. Smart TV is a platform based on the full support of Web2.0 features where a user can read and write the online content [2]. It supports and provides content from diverse data sources like online stored video, video on demand (VoD) services, video clips, social networking sites, and support for online streaming channel [3]. Besides these multiple content facilities, smart TV also offers new features

like Set-Top Boxes (STBs), connection facilities with smart handheld devices like smartphones and tablets [4].

The abundance of multimedia content on smart TV is one of the main causes of attraction for the users. The rapidly growing content on social networking sites, such as YouTube, Dailymotion, and Instagram, creates and generates users' big data sources. These data sources are becoming richer day by day and growing overwhelmingly. However, the users feel difficulty in searching for their desired content, such as searching for channels or programs [5–8]. One such popular social video-sharing website is YouTube, where users upload five hundred hours of videos per minute on YouTube [9]. The growing rate of such content creates trouble for the users to search the desired videos among these rich multimedia data. Searching for the required channels on a smart TV is a major issue as it is based on a linear search (bottom-up) using a traditional remote control [10,

11]. Due to unique features of smart TV and remote control, searching content is a major issue compared to handheld devices like smartphones and tablets. The reason behind least interest in the smart TV compared to other handheld devices is that the handheld devices provide one-touch and are easy to use for accessing the vast collection of online content. At the same time, smart TV is a lean-back device working well with the traditional remote control and other digital devices attached to the smart TV. These features lead a user to tedious task of searching and hurts user feelings [12, 13]. Thus, the searching process on this platform leads to the content overload problem.

Different techniques and methods are presented in the literature to provide and recommend videos based on the different features to the users [11]. The study [14] presented the video based on vector approach and deep learning method of image-based features (objects) extracted from video keyframes. Some works targeted the audio and visual features to classify videos [15, 16]. Besides audio and visual features, the study in [17] presented an approach based on multiple features. Both visual and textual features are selected to cluster the videos. The clustering technique plays a significant role in classifying videos in this domain and is one of the possible solutions for making searching and recommendation more viable and significant [18]. Figure 1 depicts the general approach of content clustering.

The recommendation approach uses user's profile and item's profile data and recommends a relevant data item that is supposed to be relevant to a user or group of users. It has different approaches, such as implicit feedback, explicit feedback, and hybrid approaches [19]. The implicit feedback are navigation and type of sites watching, whereas explicit feedback are likes, dislikes, ratings, and keywords. In hybrid approaches, we combine all these approaches [20].

In this paper, we presented a personalized content-based (CB) recommendation approach based on user previous watching history and presented the results in clusters. Unlike the collaborative filtering (CF) approach, which sometimes provides undesired content based on the neighbour profile (using the rating or number of views information), the presented method in this work offers desired and related content based on the user's interests. The method is providing/suggesting multiple contents to the user from the diverse data sources (YouTube, Dailymotion, etc.) based on their interests (favourite watched programs) by grouping related/similar content into clusters. Clustering or cluster analysis is one of the most commonly used unsupervised machine learning methods, which determines similarities between data points and combines similar data into one group (cluster) without any labelled data [21, 22]. In Figure 1, the data point may be text documents, web pages, or video content. The objectives and contributions of the paper are:

- (i) To present the user with related sets of content from diverse sources to overcome the issues of cognitive overload
- (ii) The paper presents a novel method of content extraction from diverse data sources (YouTube, Dailymotion, etc.) based on watching activity/interest

- (iii) The paper further provides meaningful clusters for searching the desired and related content to a user on smart TV
- (iv) A subjective study has been conducted for evaluating the user's satisfaction, ease of searching, and browsing for both preclustered and postclustered approaches
- (v) Lastly, the paper provides a comparison of clustering algorithms in the domain TV-related content clustering like movies, dramas, and songs and suggests suitable clustering algorithms for these contents

The remainder of the paper is divided into the following sections. Section 2 provides a literature review for this work. Section 3 describes the proposed a solution and discusses its methodology. Sections 4 and 5 describe results and evaluation, and Section 6 concludes the work with future research direction.

2. Related Work

This section provides compact yet comprehensive detail on relevant literature in the context of smart TV domain. This section further provides how different content like stored video, online streaming channels, and programs are clustered and how they are recommended based on clusters and user's preferences. Limitations of the existing work are also discussed in details. A plethora of literature is available on machine learning and clustering techniques [23], which this portion also discusses.

2.1. Clustering Techniques Based on Users Preferences. In context of TV watching, a user preference refers to the interests of the certain programs watched by a user [20, 24] or the user preference to the other content based on their interests watched previously by a user [25]. By examining the preferences of the users, different approaches and techniques are presented [26, 27], which recommends similar content to the users, and reduce the content overload problem. Similar to the user preferences, [28] presents a study on the user's experience and their factors in the domain of smart TV and discussed different studies and their factors, such as ease of use, accessibility, personalization, content diversity, and browsing content. It determined the comprehensive set of factors that affect the user's satisfaction and usage differently. The study [29] clustered the users based on their preferences to provide top-N recommendations for a user in each cluster. The approach offers recommendations based on the other users' preferences within the same clusters. This recommendation technique has limitations, as unwanted content is recommended to the target user based on the neighbour items/users in the same cluster. Similarly, Wu et al. [30] proposed an approach to reduce channel selection paths and provide fast and smooth channel selection based on the user's previous history. The proposed approach presented an efficient navigation approach, which minimizes the channel searching, and selection seeks distance for a

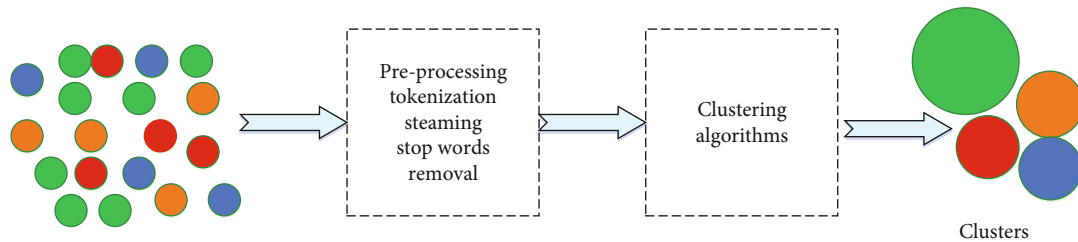


FIGURE 1: General content clustering approach.

user, using the remote-control device based on the next and previous button consequence. For example, a user often switches from news to funny channel, this preference shows the user interests, and it must be closely positioned. The hierarchical clustering schemes were used to construct a sequence of channels and provide close pairs of channels based on users' frequently switching channels.

2.2. Clustering Techniques Based on Visual Features. The videos on the web contain different features (i.e., visual and textual), and based on these features, different techniques and approaches are presented to reduce the information overload problem. Liu et al. [27] proposed an approach for clustering videos return from web search results by using visual features to reduce searching space. The proposed approach clusters similar videos into a cluster to eliminate the duplicate videos returned from the user query results on the web to provide the smooth searching of the relevant video among many videos. Visual content (frames) are matched using the signature-based similarity method to find the similarity between videos. The video frame histogram is calculated in the study [31]. Based on frame histogram, the similarity between videos were calculated, and the affinity propagation (AP) clustering algorithm [32] was used to group similar videos into the cluster. Yang et al. targeted the static video summarization problem for clustering and proposed a novel clustering-based method for static video summarization [32]. They proposed a novel clustering algorithm called Video Representation Based on High-Density Peak Search (VRHDPS). The proposed method includes four steps: (i) presampling, (ii) video frame representation, (iii) clustering, and (iv) static video summarization results. Similarly, [33] also presented a clustering method based on the deep learning technique by extracting the image-based features from the frames. The approaches mentioned above are based on visual features extracted from the videos. The main drawback of the visual features is that it is an expensive process because extracting visual features from lengthy videos is an expensive task.

2.3. Clustering Techniques Based on Textual and Visual Features. The studies [34, 35] presented web video clustering based on multiple features to overcome the limitation of video clustering based on visual features. This proposed system includes the following components: (i) video acquisition and preprocessing, (ii) preprocessing of text information, and (iii) video clustering and results in visualization. In the first step, metadata about videos from YouTube is collected

using the TubeKit open-source YouTube crawler. The information was stored in a local database indexed with a video ID. Second, the authors used their previous work for video processing. With the help of principal component analysis (PCA), the sequence of video frames is transformed into bounded coordinate system (BCS) to form a new coordinate system [25]. BCS uses the bounded principal component (BPC) to remove the noise in frames histogram. In text preprocessing, the text was compared according to their common words in the metadata. The similarity between sentences was calculated according to common words in their sentences. Only tags, titles, and description similarities are calculated. At last, all feature set (visual content, title, tags, and description) was clustered using the clustering algorithm, and the result was visualized for the user. Using the clustering methods AP (affinity propagation) and normalized cut (NC), the experiment results show the best results on higher textual feature weighting than the visual feature.

Another work [36] targeted the multiple model videos to categorize them. The videos on the web are various features and type like home video uploaded by users (low quality), social information of the videos, and professional videos (TV drama, movies, etc.) with low and high quality; size of the videos, nonprofessional videos, and textual information about videos like tags and description title are targeted. The proposed approach consists of three steps, i.e., feature extraction, classification, and fusion. Their visual features, semantic features, surrounding text, and audio features were extracted to represent videos for categorization. Semantic features were extracted from videos using two approaches [37], video annotation (concept) and visual words. Based on the classifiers' results for different features for each category, the results are fused to achieve a final category about videos. The study [25] proposed the playlist-based video clustering method (PV-clustering) by claiming that the proposed method is inexpensive as compared to the existing approaches, which were having some problems, like low-quality text information in the metadata, difficulty in extracting visual content, and noise in the information of users viewing behaviour. The proposed method consists of three steps: playlist information acquisition, video similarity calculation based on the selected features, and video clustering. First, the authors collected the information from YouTube-like Playlist id (Pid), Video id (Vid), video title, and description. This information was expressed in a binary playlist-video incident matrix. Cosine similarity measure was used to compute the similarity between the binary playlist-video incident matrix of the video, and a clustering

algorithm is applied to these features. The limitation of the presented work is that several videos on the web platform or YouTube do not contain playlist information.

2.4. Recommendations Based on Clustering on TV Platform.

Cluster analysis also plays an important role in the recommendation [19, 38]. Once the clusters are created from content or users, the recommendation is carried on these clusters to provide relevant content according to their interests [39–41]. Recommending desired and relevant content to the users reduces the content overload problem. Three approaches are mostly used in the recommendation, i.e., collaborative filtering (CF), content-based filtering (CBF), and hybrid [42]. In the CF approach, the content is recommended to a user based on similar users' preferences. Based on their preferences, the users' similarity is analyzed, and content is recommended to the target user. Different techniques and methods in the domain of TV are presented in the literature to provide CF-based recommendations to the users, i.e., based on similar users' interests [43], based on items rating clustering [36], and recommending movies based on rating information [25]. Content-based (CB) recommendation provides the content to the users based on the item's features and user profile. In this approach, the user's history is examined to determine the user's interest in items. Similar items are recommended to the users based on their feature's similarity (their description, type, genre, etc.). The hybrid recommendation approach combines the characteristics of both CF and CBF approaches to provide more effective and accurate items/content to the users. A detailed review of the recommender systems in the domain of television (TV) is presented in [44, 45].

The study [46] presented an exploratory study about grouping users based on their watching patterns (behaviour) using the clustering technique. The authors presented the user modelling approach to overcome the cold start problem and recommend using the K -means clustering algorithm with Euclidean distance metric for experimental results. The study [47] extended this work and presented Catch-TV Recommendations. The proposed approaches for recommending content similar to the previously watched content and new content where the users are not familiar with them. The recommendation approaches presented by the authors are subscribed series recommendations, new series recommendations, and combining recommendation. The channel recommendation technique for the live streaming platform Twitch is presented in [48]. The proposed approach consists of three steps to recommend relevant channels to a user. In the first step, the users' preference is identified using their time on each channel, game, and language. In the second step, the users are clustered according to similar preferences. The obtained preferences are channel, game, and language. The well-known K -means clustering algorithm was used to cluster the user's preferences. In the last step, after clustering results, based on the users' preferences within the same cluster for each channel, game, and language, the top- n relevant channels were recommended. Similarly, [49] presented a personalized channel real-time recommendation system (PCRS) frame-

work in the IPTV system via deep learning using the users' watching and channel switching sequence history. The work targeted the channel switching history and does not consider the other information like metadata, user profile, and social connection.

The popularity of the channels is considered in the recommender system to provide appropriate recommendations for the users. Artificial neural network (ANN) provides appropriate recommendations for popular and unpopular channels. The popular/hot channels are recommended to the users with the help of ANN trained data from the previous popular channels watching logs called hot artificial neural network (HANN). The unpopular/cold channels are recommended to the users with the help of ANN trained data from the previous unpopular channels watching logs called CANN (cold artificial neural network). The framework produces better results than the author's previous recommendation based on the history of user switching channels [50]. Further, the movie recommendation using Apache-Spark is presented in [14], and multichannel feature vectors by efficient collaborative filtering recommendations are also proposed [15].

The methods and approaches that cluster the users based on the other user preferences may sometimes provide an undesirable recommendation. This recommendation technique has limitations where undesirable content is recommended to the target user based on the neighbour items/users in the same cluster. In this situation, the personalized content-based (CB) recommendation technique provides more relevant and desired content to the users than the collaborative filtering recommendation method. Our proposed approach is based on a content-based recommendation technique based on the user interest, as described in the next section in detail.

2.5. Hard and Soft Clustering. Various clustering algorithms are available in the literature, but selecting suitable clustering algorithms depends on the dataset and domain where applicable [51]. This section only discusses two types of clustering algorithms and the suitability of the clustering algorithms in TV-related content. In the result section that the soft cluster algorithm provides precise results compared to the hard-clustering algorithm, K -means [52], one of the most widely used clustering algorithms, fails to provide precise and accurate results compared to the soft clustering algorithm. Comparisons and discussions are provided in the results section in detail.

The partition-based method is called the flat clustering method. In this method, a set of flat clusters is created. It is a popular and most widely used clustering technique. Unlike hierarchical clustering techniques that generate a dendrogram, partition-based clustering algorithms aim to partition the data into groups of similar data points. The cluster assignments may be hard or soft [53, 54]. In the hard clustering method, the data objects are divided into several unique homogenous datasets called a partition. Each partition represents a cluster. Each data object must belong to only one cluster in the hard-based method. A well-known

example of this type of method is the K -means clustering algorithm. Flat clustering (hard) can be defined as follows:

Given (i) a set of documents $D = (d_1, d_2, \dots, d_n)$, (ii) desired number of clusters, K , and (iii) an objective function that evaluates the quality of clustering, the clustering task is to compute an assignment $\gamma: D \rightarrow \{1, 2, \dots, K\}$ that minimizes (or in certain cases maximizes) the objective function [53].

Hard clustering (classical approach) assigns data point or object to just one cluster. On the other hand, the soft clustering approach assigns data points or object to more than one cluster [55]. The soft flat clustering method is based on the membership value where a document may be assigned to more than one cluster. Soft flat clustering is also called fuzzy clustering because it is based on the membership values belonging to a particular cluster [56]. The membership values are in the interval of (0, 1). The value zero represents the low similarity in the cluster, while one indicates the high similarity of the data points in the cluster. The membership value of the data points or documents within the cluster specifies the closeness of the data points to a cluster [57].

3. Proposed Methodology

This section presents the proposed methodology, as depicted in Figure 2—based on the user interest captured from their favourite watched program. To find the favourite user program, we need to analyse the list of programs watched by a user. Thereupon, we need to choose those channels or programs mostly watched by a user based on the time spent on each program. The longer time spent on the program means that users like that content. The fewer watched program based on the time proportion confirms that the user is not interested in the consumed channels or programs. The steps of the proposed methodology are briefly described below:

- (i) In the first step, the user's browsing and watching activities on a smart TV are captured and analyzed to find their interests. The user's interests are captured from the channel they are watching more often by capturing the content type and the dwell time on each watched content type
- (ii) The metadata related to the user's watched content is extracted in the second step. This metadata, which contains program name, description, and type, will then be searched for and matched with the metadata of content present on other data sources (i.e., YouTube and Twitter). The matched content is then collected
- (iii) The metadata from matched content (from diverse data sources) is extracted in the third step. Based on this extracted metadata, the content is clustered, which will then be recommended to the user

The proposed methodology is shown in Figure 2. We used metadata of the videos to provide the content to a user. Using only the metadata about the content is that textual

features yield better results than those provided by the visual [58, 59]. Apart from providing better results, metadata is less expensive than the visual features because extracting visual features from lengthy videos is an expensive task.

As shown in Figure 3, in the first step, the user's watching activities on smart TV are captured and analyzed to find their preference. User preference refers to the interest in certain programs watched by a user on the television [60] or the user preference to the other content based on their interest watched previously by a user [61]. This interest capturing is done using their watching routine by looking at the channel being watched, the program type on that channel during this specific time, and duration time on each watched program.

For example, if a user watches a horror movie daily, the recommended items should include horror movies or dramas. Table 1 is an example of a user who watched a TV program, assuming that the user has watched four different programs. By analyzing this information, the metadata is captured about the watched programs, which are then used for future recommendations in clusters to the user.

The metadata related to the user's watched content is extracted in the second step. For example, suppose a user has watched a specific content (e.g., horror movie). In that case, according to this extracted metadata, all metadata about this movie will be collected, including its name, type, and description, to look for more content from diverse data sources. The similarity between these features is examined with the help of similarity measures. Different similarity measures are presented in the literature to find the similarity between items based on their features. To find the similarity between these extracted features from user interests and collected features from diverse data sources, the cosine similarity technique is used, one of the most widely used similarity measures [52, 62]. It computes the similarity between two documents. Given the two documents d_i and d_j , then their cosine similarity is

$$\text{Sim}(d_i, d_j) = \cos(\theta) = \frac{(d_i \bullet d_j)}{(\|d_i\| * \|d_j\|)}. \quad (1)$$

The (\bullet) represents in Equation (1) the product of two vectors, where $\|d\|$ represents the document's length. The matched content is then extracted and are clustered. Finally, the collected content from the previous step is clustered using the clustering algorithms. To this end, to provide and create quality clustering results from the collected content, Carrot2 open sources Java API (<http://project.carrot2.org/download.html>) is used. Carrot2 consists of different clustering algorithms, including bisecting K -means, STC (Suffix Tree Clustering) [63], and Lingo [14] to provide the clustering results from the content. STC algorithm in the Carrot2 framework is explained in [64]. The K -means clustering algorithm is also implemented in Carrot2, one of the most widely used in this field [65]. K -means clustering algorithm is also the best choice when clustering similar users' program preferences in the domain of television watching behaviour [49]. In this work, the bisecting K -means (a

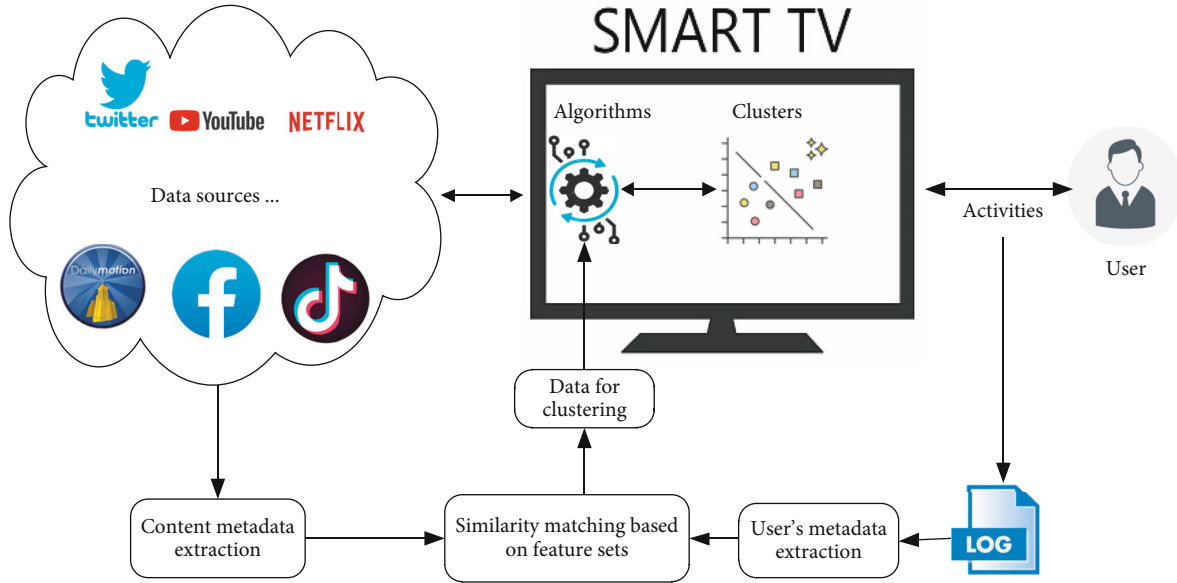


FIGURE 2: Schematic diagram of the proposed methodology.

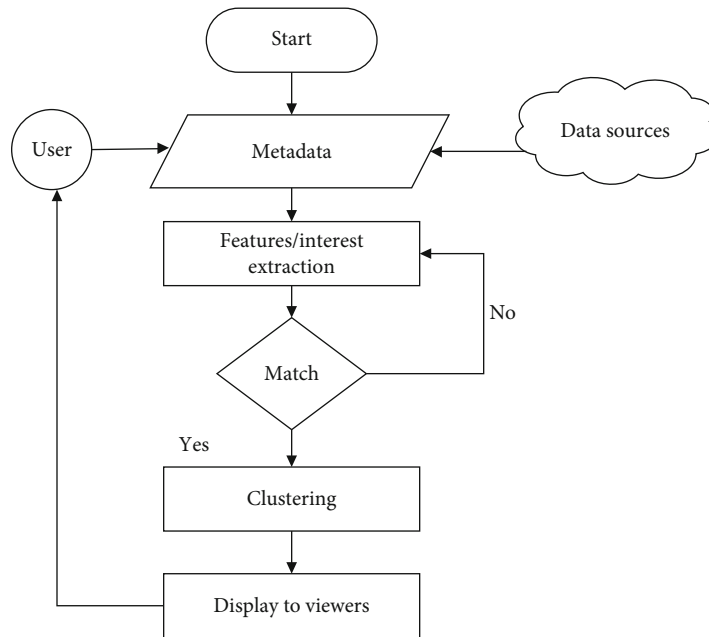


FIGURE 3: Flowchart of the overall process.

variant of K -means) is used, and STC and Lingo clustering algorithms are considered for comparison purposes. The reason behind choosing the bisecting K -means is that the bisecting K -means produce quality clustering results compared to the regular K -means clustering algorithm [50]. The following Algorithm 1 shows the overall process of the system.

The basic steps of the bisecting K -means clustering algorithm start with a single cluster (all documents) and work in the following manner [33]; in the splitting step as shown in Algorithm 1, we pick a cluster to split, then bisecting step

occurs that uses K -means algorithm to split the cluster into two subclusters. The process is repeated until the highest overall similarity (to minimize the sum of overall clusters) is achieved.

4. Results and Analysis

This section will explain the results and the datasets used in experimental results. The subsection elaborated the overall data collection process, metadata extraction, clustering approach, and results.

TABLE 1: Example of a user watching activity (h means hours and m means minutes).

Program	P1	P2	P3	P4
Name	BBC	CNN	Mr. Bean	Speed-Records
Type	News	News	Funny	Songs
Time	10 h	5 h	1 h	30 m

```

{
  Start
  Step i:      Splitting step cluster (C)
               Select (C) to split (SP);
  Step ii:     Bisecting step;
               K-means algorithm to (SP)
               Divide (C) into two sub-clusters.
  Step iii:    Repeat step (ii) and choose the (SP)
  Step iv:    Repeat until best (C) results.
  End
}

```

ALGORITHM 1:

4.1. Data Collection. We created a dataset by recording the user’s watched program history in this work. The purpose of this initial step is to infer some information about user interests. The interests are collected from their log files on smart TV. We have targeted a relatively smaller group of users (i.e., family members in the home). Our dataset consists of 8-week log records of the watched programs. We extracted important metadata from this dataset to search similar content on the diverse data sources. Based on the user watched history, we have collected the metadata of five hundred channels for experimentation, which we believe are enough to fulfil our experimentation needs. For an online collection of data sources, the carrot2 API is used to have a large dataset for further experiments.

4.2. Metadata Extraction Approach. A user’s interests are captured from their watched history by analyzing their watch-log. Apache Tikka API (<https://tika.apache.org>) is used to extract the metadata from a user who watched the program. Apache Lucene (<http://lucene.apache.org>) has been used to index this information for offline clustering. Take an example where a user has watched a news program recorded in his log file.

4.3. Extracting Metadata from Downloaded Content. The extracted metadata from the user interests was used to search and collect the content from the diverse data sources. The search was performed using the metadata of the user interests. The metadata from the obtained results have been extracted using the Apache Tikka API and indexed using the Apache Lucene. This indexed metadata was used in the experimentation. The metadata was extracted from the log record using Apache Tikka. A screenshot of the process is shown in Figure 4.

4.4. Content Clustering. The features extracted from the user interests (metadata) are used to search and match with collected metadata from diverse data sources. The collected content based on the cosine similarity matching scores is clustered using the clustering algorithms. Carrot2 provided algorithms are used for this task to provide the clustering results based on the collected information. Take an example where a user has watched a news program recorded in his log file. Figure 5 shows the results based on a user-watched program using the bisecting K -means clustering algorithm (news in this result).

STC algorithm collected the most news channel in one big cluster, as shown in Figure 6 labelling results. Further, STC provided some balance and small clusters like the latest news cluster. The results presented in Figure 7 presents some novel content (i.e., Johnny English cluster of funny clips) to recommend the related content without considering the ratings and number of views information. Thus, the approach provides related content to the users based on the users watched programs (in our case Mr. Bean funny clips) and presents the results in the form of the cluster to reduce the search space and provide novel related content (i.e. Johnny English Strike Again new movie) to the users effectively without considering the ratings or number of views information. The reason behind ignoring the rating or number of views information is that sometimes users are interested in the content based on their type (i.e., horrors), and recommending other types of content like action or romantic may dishearten the user feel—this our main work to recommend related content to the users based on their previous watched history.

4.5. User’s Satisfaction. The clustering approach yields better user satisfaction and ease in the searching process. A subjective study was conducted to evaluate the approaches used in this paper on real-time and actual watching scenarios. We took a random sample of 31 mature audiences. The browsing/navigation, searching, and user’s satisfaction was measured in both preclustered and postclustered approaches. A questionnaire of 5 rating scale was used for all three parameters, i.e., ease of searching, browsing, and user’s satisfaction. The ease in content browsing in a smart TV has been evaluated and found that browsing for content in the preclustered approach was difficult, as shown in the statistical test (Table 2).

As shown in the following Table 3, the P value is less than the alpha value 0.05, and hence, we can say that the difference between preclustered and postclustered approaches is significant. The user’s satisfaction in the below subsection shows that the difference is due to good results generated by the clustered approaches.

4.6. Ease of Searching and Browsing. The user’s satisfaction has been evaluated using statistical tests. The P value is less than the alpha (0.05), and hence, we can conclude that the difference is significant. The better average (3.74) for the postclustering approach than the preclustering approach (2.38) shows that users feel more satisfied. Similarly, searching for relevant content in the postclustered approach

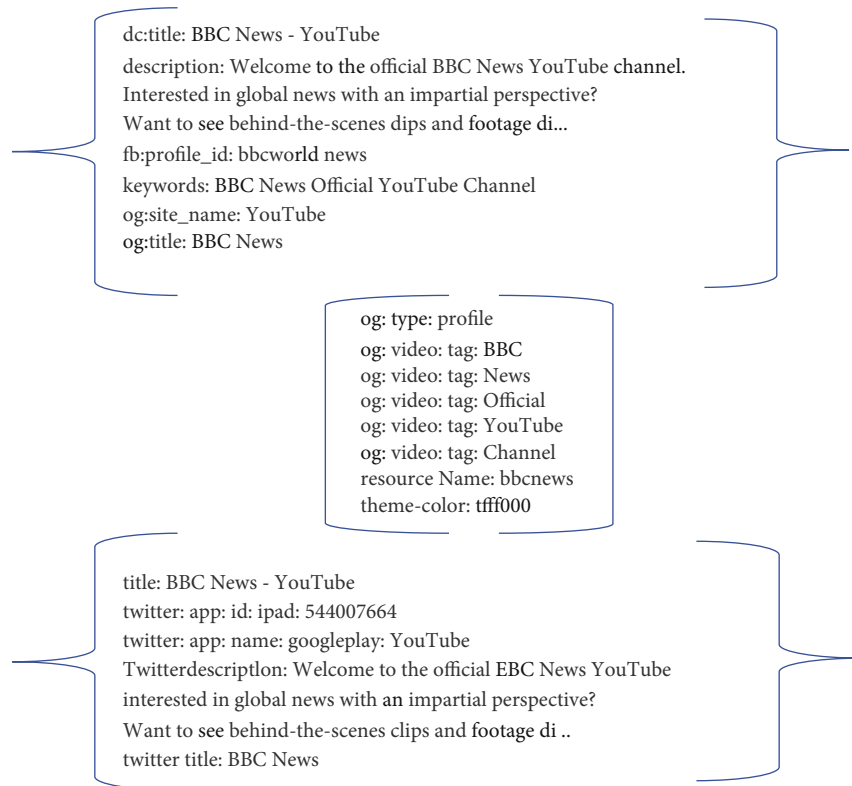


FIGURE 4: Metadata extracted from the BBC news channel.

Channel, ABC, Business (4 docs)

- [0] Watch CNN International Live TV from USA - Online TV channel
<http://www.freeintertv.com/view/id-200>
- [18] Watch France 24 Live
<https://www.france24.com/en/live>
- [42] Channels Television - Breaking Nigerian News, Today's News...
<https://www.channel.stv.com>
- [33] ABC News Live Stream Video ABC News
<https://aLicnews.go.com/Live/video/special-live-l-1447S486>
Live, BBC, UK (4 docs)
- [15] BBC News Channel Live UK - YouTube
<https://www.youtube.com/watch?v=HXeGpCFGu-k>
- [31] News Breakfast
https://en.wikipedia.org/wiki/News_Breakfast
- [45] India Today Live TV: Free Live TV, Live News Streaming and live
<https://www.indiatoday.in/intemational-livetv>
- [57] Watch BBC News UK Live Stream-BBC News Online
<http://www.Iivenews.mg.com/bbc-news-uk-live-stream/>

FIGURE 5: Clustering results using the bisecting K -means.

produced better results. As shown below (Table 3), an ANOVA test depicts that the searching becomes easy compared to the preclustered approach.

5. Comparison and Evaluation

We compared three clustering algorithms on selected features. Table 4 summarizes the results of the three clustering algorithms. The results show that STC and Lingo provide

LINGO Algorithm Cluster Labeling:

- Updated to the Latest News (4 docs, score: 24.99)
- World News (4 docs, score: 14.09)
- ARY News is a Pakistani News Channel (3 docs, score: 14.74)
- Watch Dunya (3 docs, score: 29.97)
- ARY Digital (2 docs, score: 28.94)
- Television Network (2 docs, score: 8.82)g
- Pakistan, News, TV (31 docs, score: 53.32)
- STC Algorithm Labeling:
- Latest News, Breaking News, Live TV (6 docs, score: 12.59)
- BOL News, 92 News HD (5 docs, score: 9.94)
- Channel (18 docs, score: 9)
- Urdu News (7 docs, score: 7.91)
- Watch (15 docs, score: 7.5)
- K-Means Algorithm Cluster Labeling:
- ARY, YouTube, Geo (4 docs)
- Dunya, Aaj, Largest (4 docs)
- News, Updates, Breaking (4 docs)
- HD, PTV, Soaps (3 docs)
- News, Streaming, Video (3 docs)

FIGURE 6: Some cluster labeling results.

better clustering results than bisecting K -means. STC provides different sized clusters containing quality results, while Lingo creates more precise clustering labels and assigns content to these clusters. Figure 7 provides cluster labelling results from Lingo, STC, and bisecting K -means algorithms. Unlike other clustering algorithms, the Lingo algorithm assigns a more precise label to clusters than the other clustering methods because Lingo first creates clusters labels

- Movie (6 Docs, score: 10.77)
- [11] Johnny English Strikes Again (2018)-IMDB
<https://www.imdb.com/title/tt6921996/reviews>
- [12] Johnny English Movie Trailer and Videos|TV-Guide
<https://www.tvguide.com/movies/johnny-english/>
- [17] JOHNNY ENGLISH STEELBOOK - MOV: Amazon.co.uk:
<https://www.amazon.co.uk/JOHNNY-ENGLISH-STEELBOOK->
- [8] Amazon.com.uk: Customer reviews: Johnny English
<https://www.amazon.com.uk/Johnny-English-Tim-Pigott>
- [22] Best Permabanned Johnny Depp GIFs j Find the top
<https://gfycat.com/gifs/search/permabannedtjohnnyt+>
- [24] Mr Bean's Car by ben - Meme Center
<https://www.memecenter.com/fun/19GS493/mr-bean-amp>

FIGURE 7: Novel content clusters based on the user interest.

TABLE 2: ANOVA: single factor.

(a) Summary

Groups	Count	Sum	Average	Variance
Preclustered	31	48	1.548387	0.389247
Postclustered	31	129	4.16129	0.606452

(b) ANOVA

Source of variation	SS	Df	MS	<i>F</i>	<i>P</i> value	<i>F</i> crit
Between groups	105.8226	1	105.8226	212.5594	2.20695E-21	4.001191
Within groups	29.87097	60	0.497849			
Total	135.6935	61				

TABLE 3: ANOVA: single factor.

(a) Summary

Groups	Count	Sum	Average	Variance
Preclustered	31	53	1.709677	0.546237
Postclustered	31	115	3.709677	0.546237

(b) ANOVA

Source of variation	SS	Df	MS	<i>F</i>	<i>P</i> value	<i>F</i> crit
Between groups	62	1	62	113.5039	1.84E-15	4.001191
Within groups	32.77419	60	0.546237			
Total	94.77419	61				

using the vector space model and then assigns content to these clusters [66]. The result present in Figure 7 shows that bisecting *K*-means and STC algorithms created some inappropriate labels for the cluster, i.e., soaps word in label assigned by bisecting *K*-means.

In contrast, STC assigns news, live, channel words in the label due to big clustering results (big cluster compare to others). The bisecting *K*-means algorithm provides good results, but its nonoverlapping behaviour limits it, i.e., it

gives hard clusters where one item cannot be in two clusters at a time, and this very feature of the bisecting *K*-means algorithm limits its applicability in our scenario. Further, labels of clusters are created from single words, and all content (items) in the cluster may not be similar to the label.

Bisecting the *K*-means algorithm is one the most widely used algorithms; however, selecting an appropriate clustering algorithm depends on the dataset and domain where it is better applicable. In our case, both STC and Lingo are

TABLE 4: Comparison of clustering algorithms.

Algorithms	Cluster type	No of clusters	Cluster size	Clusters labels
Bisecting K -means	Hard	10	Small	Less meaningful
Lingo	Soft	17	Small	More meaningful
STC	Soft	15	Big, balance	Meaningful

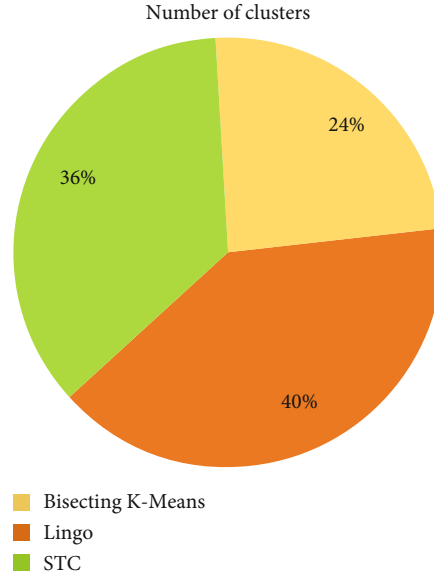


FIGURE 8: Number of clusters produced by different algorithms.

the best choice for clustering TV programs because it provides overlapped clustering results compared to the bisecting K -means algorithm, which provides hard clusters. The hard-clustering results are inappropriate when a user selects any cluster and does not find the desired program. For example, if a movie type is an action and adventure, then this movie must be placed in both action and adventure clusters. Whenever a user selects any cluster (either action or adventure), then the user can watch the movie in both clusters. If a movie (top rated) is placed in the action cluster only (in the case of hard clustering) and a user is interested in the adventure cluster, then they will miss this movie. Due to this situation, we need to present overlapped clustering results to the users on smart TV.

We evaluated the results of clustering algorithms using five evaluation measures, i.e., contamination [65], precision, recall, F -measures, and normalized mutual information (NMI). Figure 8 shows the result of the news topic. As we can see in the results, the Lingo algorithm provides the best result compared to the STC algorithm. Bisecting the K -means algorithm provides some better results on contamination and precision measures. However, bisecting K -means fail to deliver better results on F -measures. If we examine the comparison in Table 4, we can see that bisecting K -means provides hard clustering results, and both Lingo and STC algorithms provide soft clustering results. Due to this reason, bisecting K -means provide some better results from Lingo and STC algorithms. Overall, the Lingo algo-

gorithm provides quality results, meaningful labels for each cluster, and overlapped clustering results. In this situation, the Lingo clustering algorithm is suitable for clustering content to a user in TV-related content.

Figure 9 provides the overall results and comparison (five categories) of the Lingo, STC, and bisecting K -means algorithms. The five categories were news, funny videos, movies, songs, and dramas, as shown in Figure 9. The objective was to select suitable features and clustering algorithms to recommend/suggest the content to the user in clusters without considering ratings or the number of views information.

We have targeted the textual features because they provide better results than the visible results, which are computationally expensive [49, 50]. The limitation of the presented method is that it provides inappropriate results where the textual features are ambiguous. This situation is common in the YouTube video platform, where the user provides ambiguous information to the videos. Similarly, language problems are also associated with this scenario where the user provides textual information other than English. The presented work is suitable for channels or programs providing rich metadata (textual information), and based on this metadata; similar content is presented to the users. We compared three clustering algorithms. Hundreds of clustering algorithms are presented in the literature, and the selection of clustering algorithms depends on the set of features and domains where applicable [66]. We only targeted the hard

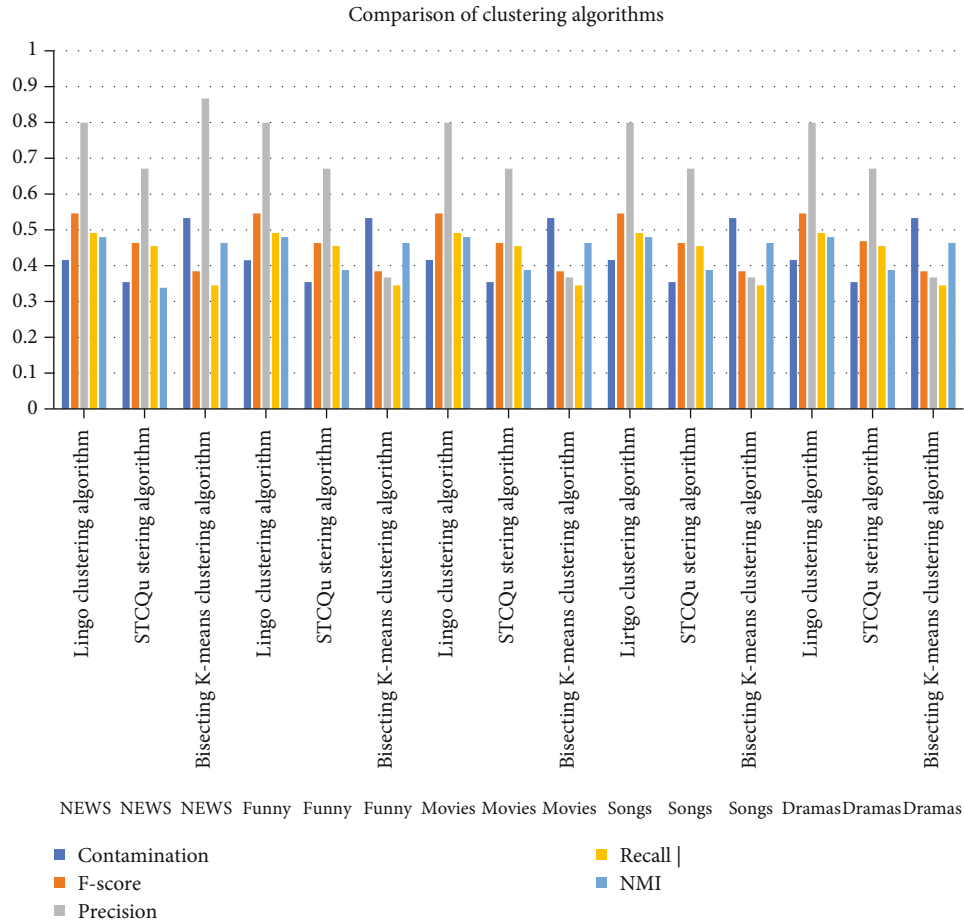


FIGURE 9: Comparison of clustering algorithms on news, funny videos, movies, songs, and dramas.

and soft clustering algorithms and considered these selected algorithms' labelling results. The reason behind the selection of this algorithm is to suggest a suitable clustering algorithm (hard or soft) in the domain of TV-related content and also suggest a suitable algorithm that creates meaningful cluster labels for user.

6. Conclusion and Future Work

Smart TV is changing the way users watch programs. The support of web 2.0 features and the huge amount of content available is the attracting force of the smart TV. The availability of content is always handy, but it sometimes happens that searching in this large collection of content becomes hard and annoying. The same is the case with smart TV users, where it is becoming increasingly difficult to find the desired content in time and with ease. Several solutions have been proposed to solve this content overload problem. In this research work, we have applied the clustering technique to address the problem of content overload on a smart TV. The motivation behind this research work is that a user tends to have some likes and dislikes while watching TV. These likes and dislikes can be monitored to know the behaviour and taste of the smart TV user. We captured the watching activities of the users, and based on these activities, we have collected similar content from multiple diverse data

sources and have presented them to the user in the form of clusters. But to make sure that similar relevant content is retrieved from outside diverse data sources, we extracted features from the user-watched content. Based on those features, we searched for more relevant and similar content. When the user watches a certain program, all the similar and relevant content is clustered and presented. This way, the user only looks for the desired cluster and does not need to search randomly for the desired content. Three algorithms (bisecting K-means, STC, and Lingo) are compared to our collected dataset for comparison purposes. The proposed solution reduces the search time and reduces the content overload problem.

In the future, we are looking to introduce the time factor while clustering the retrieved relevant content on a smart TV. Moreover, the comparative analysis of these algorithms for rich multimedia data can further elaborate the discussion. There are certain specific times when users tend to watch certain specific programs, e.g., news late at night or a bit of music in the morning. Therefore, including the time factor, we believe that the proposed content clusters will improve further. We are planning to look at other factors in the future, e.g., the user's age and language, etc. We believe that the results will improve considerably by further increasing the features on which the clusters are created.

Data Availability

The data that support the findings of this study are available upon request from the first author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] M. Khan, S. Khusro, I. Alam, S. Ali, and I. Khan, "Perspectives on the design, challenges, and evaluation of smart TV user interfaces," *Scientific Programming*, vol. 2022, 14 pages, 2022.
- [2] J. He, "Construction of internet TV industry ecosystem based on data mining technology," *Wireless Communications and Mobile Computing*, vol. 2022, 9 pages, 2022.
- [3] I. Alam, S. Khusro, and M. Khan, "Factors affecting the performance of recommender systems in a Smart TV environment," *Technologies*, vol. 7, no. 2, p. 41, 2019.
- [4] I. Khan and S. Khusro, "Towards the design of context-aware adaptive user interfaces to minimize drivers' distractions," *Mobile Information Systems*, vol. 2020, 23 pages, 2020.
- [5] Y. Elkhatib, R. Killick, M. Mu, and N. Race, "Just browsing? Understanding user journeys in online TV," in *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 965–968, Orlando, Florida, USA, 2014.
- [6] M. Khan, S. Khusro, and I. Alam, "Smart TV-based lifelogging systems: current trends, challenges, and the road ahead," in *Information and Knowledge in Internet of Things*, Springer, 2022.
- [7] I. Khan, S. Khusro, N. Ullah, and S. Ali, "AutoLog: toward the design of a vehicular lifelogging framework for capturing, storing, and visualizing LifeBits," *IEEE Access*, vol. 8, pp. 136546–136559, 2020.
- [8] I. Khan, S. S. Rizvi, S. Khusro, S. Ali, and T.-S. Chung, "Analyzing drivers' distractions due to smartphone usage: evidence from AutoLog dataset," *Mobile Information Systems*, vol. 2021, 14 pages, 2021.
- [9] T. Lian, Z. Chen, Y. Lin, and J. Ma, "Understanding the pulse of the online video viewing behavior on smart TVs," in *Chinese National Conference on Social Media Processing*, pp. 331–342, Beijing, China, 2017.
- [10] K. Watanabe, Y. Miyake, N. Nakamichi, T. Yamada, and T. Ozeki, "Remote touch pointing for smart TV interaction," in *IEEE 3rd Global Conference on Consumer Electronics (GCCE)*, pp. 232–235, Tokyo, Japan, 2014.
- [11] I. Alam and S. Khusro, "Tailoring recommendations to groups of viewers on smart TV: a real-time profile generation approach," *IEEE Access*, vol. 8, pp. 50814–50827, 2020.
- [12] X. Wu, C.-W. Ngo, Y.-M. Zhu, and Q. Peng, "Boosting web video categorization with contextual information from social web," *World Wide Web*, vol. 15, no. 2, pp. 197–212, 2012.
- [13] I. Khan and S. Khusro, "ConTEXT: context-aware adaptive SMS client for drivers to reduce risky driving behaviors," *Soft Computing*, pp. 1–18, 2022.
- [14] H. Zhang, J. Chai, Y. Wang, M. An, B. Li, and Q. Shen, "Application of clustering algorithm on TV programmes preference grouping of subscribers," in *IEEE International Conference on Computer and Communications (ICCC)*, pp. 40–44, Chengdu, China, 2015.
- [15] N. Chang, M. Irvan, and T. Terano, "A TV program recommender framework," *Procedia Computer Science*, vol. 22, pp. 561–570, 2013.
- [16] S. Pyo, E. Kim, and M. Kim, "Automatic and personalized recommendation of TV program contents using sequential pattern mining for smart TV user interaction," *Multimedia Systems*, vol. 19, no. 6, pp. 527–542, 2013.
- [17] M.-W. Kim, E.-J. Kim, W.-M. Song, S.-Y. Song, and A. R. Khil, "Efficient recommendation for smart TV contents," in *International Conference on Big Data Analytics*, pp. 158–167, New Delhi, India, 2012.
- [18] D. Huang, C.-D. Wang, J.-H. Lai, and C.-K. Kwok, "Toward multidiversified ensemble clustering of high-dimensional data: from subspaces to metrics and beyond," *IEEE Transactions on Cybernetics*, pp. 1–14, 2021.
- [19] H. Ko, S. Lee, Y. Park, and A. Choi, "A survey of recommendation systems: recommendation models, techniques, and application fields," *Electronics*, vol. 11, no. 1, p. 141, 2022.
- [20] I. Alam, S. Khusro, and M. Khan, "Personalized content recommendations on smart TV: challenges, opportunities, and future research directions," *Entertainment Computing*, vol. 38, article 100418, 2021.
- [21] S. Chander and P. Vijaya, *Unsupervised learning methods for data clustering*, Elsevier, 2021.
- [22] J. B. Rebecca, V. Ramalingam, V. Sugumaran, and D. Rajkumar, "Predictive analysis of online television videos using machine learning algorithms," in *Fundamentals and Methods of Machine and Deep Learning: Algorithms, Tools and Applications*, John Wiley & Sons, Inc., 2022.
- [23] P. Singh, *Fundamentals and Methods of Machine and Deep Learning: Algorithms, Tools, and Applications*, John Wiley & Sons, 2022.
- [24] J. Jang and Y. Y. Mun, "Determining and validating smart TV UX factors: a multiple-study approach," *International Journal of Human-Computer Studies*, vol. 130, pp. 58–72, 2019.
- [25] C.-Y. Lin and H.-S. Chen, "Personalized channel recommendation on live streaming platforms," *Multimedia Tools and Applications*, vol. 78, no. 2, pp. 1999–2015, 2019.
- [26] H. Kim, J. Crowcroft, and F. M. Ramos, "Efficient channel selection using hierarchical clustering," in *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 1–9, San Francisco, CA, 2012.
- [27] S. Liu, M. Zhu, and Q. Zheng, "Mining similarities for clustering web video clips," in *International Conference on Computer Science and Software Engineering*, pp. 759–762, Hubei, China, 2008.
- [28] X. Wu, A. G. Hauptmann, and C.-W. Ngo, "Practical elimination of near-duplicates from web video search," in *Proceedings of the 15th ACM International Conference on Multimedia*, pp. 218–227, Augsburg, Germany, 2007.
- [29] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.
- [30] J. Wu, S.-H. Zhong, J. Jiang, and Y. Yang, "A novel clustering method for static video summarization," *Multimedia Tools and Applications*, vol. 76, no. 7, pp. 9625–9641, 2017.
- [31] Z. Huang, H. T. Shen, J. Shao, X. Zhou, and B. Cui, "Bounded coordinate system indexing for real-time video clip search," *ACM Transactions on Information Systems (TOIS)*, vol. 27, no. 3, pp. 1–33, 2009.

- [32] L. Yang, J. Liu, X. Yang, and X.-S. Hua, "Multi-modality Web video categorization," in *Proceedings of the International Workshop on Workshop on Multimedia Information Retrieval*, pp. 265–274, Bavaria, Germany, 2007.
- [33] Z. A. A. Ibrahim, M. Saab, and I. Sbeity, "VideoToVecs: a new video representation based on deep learning techniques for video classification and clustering," *SN Applied Sciences*, vol. 1, no. 6, pp. 1–7, 2019.
- [34] M. Kamie, T. Hashimoto, and H. Kitagawa, "Effective Web video clustering using playlist information," in *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, pp. 949–956, Trento, Italy, 2012.
- [35] D. Liu, X. Qu, Y. Wang et al., "Unsupervised temporal video grounding with deep semantic clustering," 2022, <https://arxiv.org/abs/2201.05307>.
- [36] M. Xu, S. Berkovsky, S. Ardon, S. Triukose, A. Mahanti, and I. Koprinska, "Catch-up TV recommendations: show old favourites and find new ones," in *Proceedings of the 7th ACM Conference on Recommender Systems*, pp. 285–294, Hong Kong, China, 2013.
- [37] E. Ntoutsis, K. Stefanidis, K. Nørnvåg, and H.-P. Kriegel, "Fast group recommendations by applying user clustering," in *International Conference on Conceptual Modeling*, pp. 126–140, Florence, Italy, 2012.
- [38] H. H. C. Nguyen, B. T. Khiet, and T. T. Nguyen, "An effective method for clustering-based web service recommendation," *International Journal of Electrical & Computer Engineering*, vol. 12, no. 2, pp. 1571–8708, 2022.
- [39] S. Gong, "An efficient collaborative recommendation algorithm based on item clustering," *Advances in Wireless Networks and Information Systems*, vol. 72, pp. 381–387, 2010.
- [40] M. Ahmed, M. T. Imtiaz, and R. Khan, "Movie recommendation system using clustering and pattern recognition network," in *IEEE 8th Annual Computing and Communication Workshop and cConference (CCWC)*, pp. 143–147, Las Vegas, 2018.
- [41] D. Vêras, T. Prota, A. Bispo, R. Prudêncio, and C. Ferraz, "A literature review of recommender systems in the television domain," *Expert Systems with Applications*, vol. 42, no. 22, pp. 9046–9076, 2015.
- [42] M. Riyahi and M. K. Sohrabi, "Providing effective recommendations in discussion groups using a new hybrid recommender system based on implicit ratings and semantic similarity," *Electronic Commerce Research and Applications*, vol. 40, article 100938, 2020.
- [43] M. Xu, S. Berkovsky, I. Koprinska, S. Ardon, and K. Yacef, "Time dependency in TV viewer clustering," in *UMAP Workshops*, Montreal, Canada, 2012.
- [44] C. Yang, S. Ren, Y. Liu, H. Cao, Q. Yuan, and G. Han, "Personalized channel recommendation deep learning from a switch sequence," *IEEE Access*, vol. 6, pp. 50824–50838, 2018.
- [45] S. Aggarwal, D. Goswami, M. Hooda, A. Chakravarty, and A. Kar, "Recommendation systems for interactive multimedia entertainment," in *Data Visualization and Knowledge Engineering*, Springer, 2020.
- [46] C. Yu, H. Ding, H. Cao, Y. Liu, and C. Yang, "Follow me: personalized IPTV channel switching guide," in *Proceedings of the 8th ACM on Multimedia Systems Conference*, pp. 147–157, Taipei, Taiwan, 2017.
- [47] A. Maheshwari, A. Kumari, A. Kumari, N. Kumar, and B. M. Nandini, "Movie recommendation system using Apache Spark," in *3rd National Conference on Image Processing, Computing, Communication, Networking and Data Analytics*, p. 280, New Delhi, India, 2018.
- [48] H.-R. Zhang, F. Min, Z.-H. Zhang, and S. Wang, "Efficient collaborative filtering recommendations with multi-channel feature vectors," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 5, pp. 1165–1172, 2019.
- [49] A. Hindle, J. Shao, D. Lin, J. Lu, and R. Zhang, "Clustering web video search results based on integration of multiple features," *World Wide Web*, vol. 14, no. 1, pp. 53–73, 2011.
- [50] V. Mekthanavanh, T. Li, J. Hu, and Y. Yang, "Web video clustering based on emotion category," in *Proceedings of the 2018 International Conference on Big Data Engineering and Technology*, pp. 87–91, Chengdu, China, 2018.
- [51] F. U. Siddiqui and A. Yahya, "Quantitative analysis methods of clustering techniques," in *Clustering Techniques for Image Segmentation*, Springer, 2022.
- [52] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.
- [53] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*, vol. 1, Cambridge University Press, Cambridge, 2012.
- [54] A. Kumar, A. Nayyar, S. Upasani, and A. Arora, "Empirical study of soft clustering technique for determining click through rate in online advertising," in *Data Management, Analytics and Innovation*, Springer, 2020.
- [55] M. B. Ferraro and P. Giordani, "Soft clustering," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 12, no. 1, article e1480, 2020.
- [56] M. Li, L. Wen, and F. Chen, "A novel collaborative filtering recommendation approach based on soft co-clustering," *Physica A: Statistical Mechanics and its Applications*, vol. 561, article 125140, 2021.
- [57] V. K. Singh, T. J. Siddiqui, and M. K. Singh, "Evaluating hard and soft flat-clustering algorithms for text documents," in *Proceedings of the Third International Conference on Intelligent Human Computer Interaction (IHCI 2011)*, pp. 63–76, Prague, Czech Republic, 2013.
- [58] A. Huang, "Similarity measures for text document clustering," in *Proceedings of the Sixth New Zealand Computer Science Research Student Conference (NZCSRSC2008)*, pp. 49–56, Christchurch, New Zealand, 2008.
- [59] R. Xu and D. Wunsch, "Survey of clustering algorithms," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005.
- [60] O. E. Zamir and O. Etzioni, *Clustering Web Documents: A Phrase-Based Method for Grouping Search Engine Results*, University of Washington, 1999.
- [61] S. Osiński, J. Stefanowski, and D. Weiss, "Lingo: search results clustering algorithm based on singular value decomposition," in *Intelligent Information Processing and Web Mining*, Springer, 2004.
- [62] J. Stefanowski and D. Weiss, "Carrot 2 and language properties in web search results clustering," in *International Atlantic Web Intelligence Conference*, pp. 240–249, Madrid, Spain, 2003.
- [63] X. He, M.-Y. Kan, P. Xie, and X. Chen, "Comment-based multi-view clustering of web 2.0 items," in *Proceedings of the 23rd International Conference on World Wide Web*, pp. 771–782, Seoul, Korea, 2014.
- [64] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," in *KDD workshop on text mining*, pp. 525–526, Boston, USA, 2000.

- [65] S. Osinski and D. Weiss, "A concept-driven algorithm for clustering search results," *IEEE Intelligent Systems*, vol. 20, no. 3, pp. 48–54, 2005.
- [66] S. Solorio-Fernández, J. A. Carrasco-Ochoa, and J. F. Martínez-Trinidad, "A review of unsupervised feature selection methods," *Artificial Intelligence Review*, vol. 53, 2, pp. 907–948, 2020.