Clustering Netflix Shows Based on Features Using K-means and Hierarchical Algorithms to Identify Content Patterns

B Herawan Hayadi[1,*], Eko Priyanto[2]

[1] *Primary School Teacher Education, Universitas Bina Bangsa, Serang, Indonesia,*

[2]*Ma'arif University of Nahdlatul Ulama, Kebumen, Indonesia*

(Received: January 10, 2025; Revised: February 18, 2025; Accepted: May 23, 2025; Available online: July 23, 2025)

Abstract

This study explores clustering patterns within Netflix's movie catalog by applying K-means and hierarchical clustering algorithms. The primary objective is to identify distinct content groups based on features such as movie duration, release year, and content ratings. The dataset, which includes 5,185 Movies, was preprocessed by handling missing values, one-hot encoding categorical variables, and standardizing numerical features. Four distinct clusters were identified, with each cluster exhibiting unique characteristics. Cluster 0 primarily consists of longer, family-friendly Movies rated TV-14, while Cluster 1 contains shorter, mature Movies with a TV-MA rating. Cluster 2 represents a diverse range of TV-MA Movies with moderate durations, and Cluster 3 focuses on adult-oriented, longer Movies with an 'R' rating. These findings offer valuable insights into Netflix's content strategy, highlighting the platform's ability to cater to different audience segments based on content type and viewer preferences. The results suggest that Netflix can leverage clustering patterns to improve its recommendation system and content acquisition strategy. However, the study is limited by the absence of user-specific data and the reliance on basic metadata features. Future research could explore the integration of additional features like user ratings and apply deep learning techniques for more sophisticated clustering.

Keywords: Clustering, Content Strategy, K-Means, Netflix, Recommendation System

1. Introduction

Netflix has emerged as a significant player in the realm of digital streaming, fundamentally altering the landscape of entertainment consumption. Founded in 1997 as a DVD rental service, Netflix has transitioned into a leading Subscription Video-On-Demand (SVOD) platform with a vast catalog encompassing genres, formats, and production styles across global markets. The platform now boasts a collection of original content, licensed films, and series that reflect diverse narratives, age groups, and cultures, thereby attracting a wide range of consumers globally. This breadth of content serves not merely for entertainment; it also collects valuable data for analysis regarding pattern recognition and sociocultural insights.

The importance of analyzing Netflix's content can be observed through various angles, including health messages, cultural representation, and behavioral impacts on viewers. For instance, studies have indicated that Netflix's original content often addresses complex social issues like mental health, substance use, and sexuality. A content analysis of Netflix series indicates significant themes related to mental health and violence, aligning with contemporary societal concerns, and thus warrants investigation into how these are portrayed and their implications for audience perceptions and behaviors [1]. The examination of substances such as alcohol and tobacco in Netflix offerings reveals a nuanced picture of risk exposure among audiences, particularly younger viewers [2]. When analyzed systematically, these portrayals can expose societal trends and emphasize the responsibilities content creators bear in shaping public perspectives.

Furthermore, with the rising popularity of platform-based streaming among adolescents, understanding the way content is communicated is crucial. Recent research has focused on how sexual content is referenced in Netflix's programming

*Corresponding author: B Herawan Hayadi (b.herawan.hayadi@gmail.com)

DOI: https://doi.org/10.47738/ijaim.v5i2.102

aimed at younger audiences [3]. This analysis showcases the methodologies employed to decode textual patterns within series, aiming to highlight the implications these references have on adolescent viewers. The emergent patterns from such content analyses can reveal broader societal attitudes towards topics such as sexuality and identity, as well as how these narratives might influence adolescent development.

Identifying distinct content patterns in Netflix shows using feature-based clustering presents several significant challenges that researchers and analysts must navigate. As Netflix's catalog expands, comprising a myriad of genres and themes, the task of recognizing underlying patterns becomes increasingly complex. One prominent issue lies in the complexity of the features that define content on the platform. Features can include narrative structure, themes, visual aesthetics, character archetypes, and subplot elements, each of which can be represented in multiple ways. Different series may employ diverse storytelling techniques and artistic choices, which can obscure direct comparisons. Thus, isolating these features in a meaningful, consistent manner becomes a critical hurdle in pattern recognition using clustering algorithms [4].

Effective clustering relies heavily on the quality and selection of features that will be utilized in the analysis. In this context, effective text representation techniques are vital for handling vast streaming data. Variants like Bag of Words, TF-IDF (Term Frequency-Inverse Document Frequency), Word2Vec, and GloVe models each yield distinctly different results when applied to clustering models [4]. Given the subjectivity of entertainment content—what may resonate through a strong narrative arc for one viewer may not hold the same significance for another—establishing a uniform feature representation that accurately captures nuanced elements is essential for any clustering effort. Ensuring that these representations maintain high fidelity to the content they are meant to describe is crucial, as it directly impacts the reliability of the clustering results [5].

Another significant challenge is the dynamic nature of Netflix's content curation and user preferences. As Netflix regularly updates its catalog and user tastes evolve, static feature sets may become obsolete. This necessitates the development of adaptive clustering algorithms that can evolve over time to reflect changing content characteristics and viewing patterns [6]. The heterogeneity of user engagement data further complicates this; for example, understanding viewers' behavioral responses to particular genres or themes can vary significantly across different demographic segments, making it more challenging to conceive a singular clustering approach that effectively captures diverse user experiences [7].

The objective of this study is to apply K-means and Hierarchical clustering algorithms to group Netflix shows based on key features such as type, rating, and duration, with the aim of enhancing our understanding of how content can be categorized and analyzed within the sprawling catalog available on the streaming platform. This methodological approach seeks to extract meaningful patterns from the multitude of shows, thereby facilitating insights into their structural similarities and differences and ultimately improving content recommendation systems as well as informing strategic content development.

K-means clustering is particularly beneficial for this analysis due to its efficiency in categorizing data into distinct groups based on feature similarity. The K-means algorithm operates by partitioning the dataset into (k) clusters where each show is assigned to the cluster with the nearest mean. By employing this clustering technique, we can delineate shows into thematic or categorical groupings that can reveal patterns of viewership behavior based on specified criteria such as genre and audience ratings. This technique can offer tremendous advantages, especially in analyzing how different genres attract varying viewer demographics.

On the other hand, Hierarchical clustering provides a more nuanced approach, particularly beneficial for understanding the relationships between shows that may not fit neatly into predefined categories of K-means. By generating a dendrogram, Hierarchical clustering allows for the visualization of how shows cluster together based on a range of parameters, leading to insights into the broader cultural narratives represented within the Netflix library. This can be particularly relevant as the platform continues to expand its offerings, including varied thematic explorations of marginalized communities and social issues.

The significance of this study lies in its ability to provide valuable insights into Netflix's content catalog by uncovering hidden patterns and trends through clustering techniques. By grouping shows based on their features such as type, rating, duration, and genre, this research can help Netflix better understand its content distribution and viewer preferences. These insights can inform business decisions, such as content acquisition strategies, personalized recommendations, and targeted marketing campaigns. Additionally, clustering can aid in improving Netflix’s recommendation algorithms by identifying clusters of shows that share similar characteristics, enhancing the overall user experience and engagement.

2. Literature Review

2.1. Data Mining in Content Recommendation

Data mining plays a pivotal role in enhancing content recommendation systems, particularly through the application of clustering methodologies. Previous studies have explored various clustering techniques, focusing on algorithms such as K-means and Hierarchical clustering, which serve critical functions in recommending items that align with user preferences in diverse domains, including e-commerce, streaming services, and music applications [8]. This review synthesizes key findings from the literature surrounding clustering and content recommendation systems, emphasizing collaborative filtering and content-based filtering as two primary recommendation strategies.

The K-means clustering algorithm has emerged as one of the most prevalent techniques in the realm of content recommendation systems, owing to its efficiency in segmenting data into distinct groups based on similarity in features. For instance, Andra and Baizal [9] successfully demonstrated how K-means can effectively address the sparsity problem in Collaborative Filtering (CF) by clustering user-item interactions and enabling the system to make informed recommendations based on user similarity. Similarly, Mu'Afa and Baizal [10] noted the applicability of K-means in providing ratings for items by clustering users or items that exhibit comparable preferences, thereby refining the recommendation process. This approach illustrates K-means's versatility in managing large datasets typical of modern recommendation systems. Further reinforcing K-means's utility, Chen et al [11] explored its integration within a music recommendation system where songs were clustered based on feature vectors. Such analysis allowed the system to identify musical similarities, enhancing the user experience by curating playlists reflective of individual preferences. This underscores K-means's effectiveness across different types of media and emphasizes the technique's adaptability for niche applications within recommendation frameworks.

Hierarchical clustering represents another essential methodology within content recommendation, facilitating a more comprehensive understanding of item relationships by providing both agglomerative and divisive approaches. This method allows for the generation of dendrograms, enabling analysts to visualize the structure of clusters effectively. In analyzing news recommendation systems, for example, Darvishy et al [12] observed that hierarchical clustering can enhance system scalability, allowing for the efficient grouping of vast quantities of news articles based on shared characteristics. Such strategies inform how dynamic clustering approaches can tackle substantial datasets, particularly in environments where content diversity is paramount. Moreover, the integration of clustering with collaborative filtering techniques has been noted to significantly enhance the efficacy of traditional recommendation systems. Zarzour et al [13] provided insights into employing K-means clustering with Principal Component Analysis (PCA) for dimensionality reduction, which improves clustering accuracy and item recommendations. The intersection of these methodologies indicates the potential for further innovation within the recommendation landscape, as combining approaches can often produce superior results.

2.2. Clustering Algorithms

Clustering algorithms, particularly K-means and Hierarchical clustering, have become instrumental in the field of content categorization, especially with large datasets like those found in streaming platforms. Both methodologies offer distinctive approaches to grouping data based on shared characteristics, leading to improved efficiency in content recommendations. K-means clustering is primarily a partitioning method that works by grouping data into a predetermined number of clusters, optimizing the intra-cluster variance. This characteristic makes K-means exceptionally efficient when dealing with large datasets, as it can handle millions of records and compute the centroids of clusters quickly. Studies have shown its effectiveness in various contexts. For example, Zhu and Han [14] found that K-means clustering is a practical way to recommend similar beers and has demonstrated improved efficiency compared to Hierarchical clustering.

The effectiveness of K-means in handling high-dimensional datasets stems from its simplicity and speed. As established by Murthy et al [15], K-means automatically arranges texts into clusters such that text data within clusters are relatively similar in terms of content when compared to text data in other clusters. This is particularly beneficial for content recommendation systems, as it allows for swift retrieval and recommendations based on user profiles and preferences. Furthermore, its capability to categorize attributes into defined groups enhances the machine learning processes pertinent to data categorization, as indicated in the work of Benis et al [16], where K-means was used to categorize healthcare communication data. In contrast, Hierarchical clustering provides a different approach by creating a tree-like structure of data clusters, allowing exploration of the relationships between clusters at various levels of granularity. This method is particularly useful when understanding how smaller categories relate to larger ones is essential, as it can deliver insights that would be obscured in a flat clustering structure. Huang and Ma [17] noted that combining K-means with Hierarchical clustering improves cluster quality by harnessing the strengths of both approaches, ensuring better scalability and adaptability to different types of data.

2.3. Relevant Formulas

The application of clustering algorithms for content categorization, particularly K-means and Hierarchical clustering, necessitates an understanding of the pertinent mathematical foundations that underpin their operations. This section will detail the mathematical formulas associated with K-means clustering, including centroid calculation and Euclidean distance, as well as the descriptions of dendrogram construction and linkage criteria pertinent to Hierarchical clustering. The centroid (mean) of a cluster, which acts as the center point of the cluster, is computed using the following formula for a cluster (Ck):

$$ck = \frac{1}{|Ck|} \sum xi \in Ck\, xi \tag{1}$$

Here, ck represents the centroid vector for cluster (k), |Ck| is the number of points in the cluster, and xi are the individual data points belonging to cluster (Ck) [18]. The distance between two points x and ck is calculated using the Euclidean distance formula:

$$d(x, ck) = \sqrt{\sum j = 1^{n} (xj - ckj)^2} \tag{2}$$

In this equation, (xj) denotes the j-th component of point (x), and (ckj) is the j-th component of the centroid (ck). This formula quantifies the straight-line distance between point (x) and its nearest centroid, aiding in the assignment of points to clusters [19].

Hierarchical clustering can be represented visually through a dendrogram, which illustrates the arrangement of clusters. The dendrogram is constructed using a stepwise process that amalgamates clusters based on defined distances or similarities between them until all points belong to a single cluster. While not expressible by a single formula, the construction of the dendrogram involves iteratively calculating distances until a stable representation is achieved [20]. The effectiveness of Hierarchical clustering heavily relies on the linkage criteria used to determine the distance between clusters. Some common linkage criteria include a) Single Linkage to measures the minimum distance between elements of two clusters:

$$d(C1, C2) = \min x \in C1, y \in C_2\, d(x, y) \tag{3}$$

b) Complete Linkage to measures the maximum distance between elements of two clusters:

$$d(C1, C2) = \max x \in C1, y \in C_2\, d(x, y) \tag{4}$$

c) Average Linkage to computes the average distance between all pairs of points from two clusters:

$$d(C1, C2) = \frac{1}{|C1||C2|} \sum x \in C1 \sum y \in C2\, d(x, y) \tag{5}$$

Each of these criteria significantly influences the shape of the resulting dendrogram and the final clustering outcome, guiding the fusion of clusters [21].

2.4. Applications in Media and Entertainment

Clustering techniques have gained widespread application across various media industries, serving pivotal roles in content categorization and trend analysis. This section reviews notable studies in different media domains, emphasizing how clustering contributes to understanding audience behavior, trend identification, and content management. In the field of music, clustering has proven instrumental in understanding listener preferences and categorizing audio content. However, while Chen et al [22] explore a blockchain-based model for the media industry, the specific application of clustering techniques in modern content distribution systems is not directly addressed in their study. Although clustering can indeed aid in creating playlists based on user preference similarities, the specifics regarding its impact on content delivery within the context of their findings are not well-supported. Music recommendation systems utilizing K-means clustering can facilitate personalized user experiences by categorizing songs based on audio features. Clustering supports the generation of playlists that reflect individual users' moods and listening histories. This application exemplifies how clustering optimizes user engagement on streaming platforms.

Clustering algorithms have significantly influenced how user-generated content is managed on platforms like YouTube and Instagram. García-Rapp [23] examined the mechanisms employed by social media influencers, but the focus is primarily on popularity markers and audience engagement strategies. The direct application of clustering algorithms to analyze user interaction data, while compelling, is not the main focus of their work. On the other hand, Xie et al [24] investigated clustering in the context of social media data, effectively demonstrating the utility of hierarchical clustering for classifying large volumes of posts and comments based on sentiment and theme. This shows how hierarchical clustering can visually represent relationships among content categories, allowing creators to identify emerging trends in user preferences. The film industry utilizes clustering to categorize films based on genre, audience ratings, and reviews. Although Clarke et al [25] discuss clustering for categorizing data, their work does not specifically focus on films or streaming services. This generality limits the direct relevance of their findings to the categorization of films. Hierarchical clustering, however, remains advantageous in this domain as it can illustrate how films relate to one another within broader genres and sub-genres. This multidimensional analysis aids marketers in developing targeted promotional strategies tailored to specific audience segments.

3. Methodology

3.1. Data Loading and Initial Inspection

The analysis began with the loading of the Netflix dataset from Kaggle using the `pandas.read_csv` function, which is a standard method for importing CSV files into a DataFrame. The dataset file, named 'Netflix Datasets Evaluation MS Excel.csv', was successfully read into memory, and its structure was immediately inspected to ensure it was properly loaded. The first step involved printing the shape of the dataset using `df.shape`, which provided the number of rows and columns, helping to understand the overall size and complexity of the dataset.

Next, the `df.info()` method was executed to display detailed information about the DataFrame, such as the number of non-null values, data types of each column, and the presence of any missing data. This step was crucial for identifying columns that may need additional cleaning or preprocessing. To gain a quick overview of the content, the first few rows of the dataset were printed using `df.head()`, which helped in confirming that the dataset loaded correctly and the data format was as expected. By inspecting the initial rows of the dataset, we were able to get a sense of the structure of the data, particularly for the key columns such as `show_id`, `type`, `title`, `director`, `rating`, `duration`, `release_year`, and others. This initial inspection of the data provided a foundational understanding and prepared us for the subsequent steps of cleaning and analysis. It was essential to check for inconsistencies or missing values at this stage to ensure the dataset was ready for deeper exploration.

3.2. Data Preprocessing

Data preprocessing is a crucial step in preparing the dataset for further analysis and clustering. The first preprocessing task involved handling missing values. Since clustering algorithms do not perform well with missing or incomplete data, any rows with missing values were dropped using `df.dropna(inplace=True)`. This operation ensured that we would be working with a clean and complete dataset, eliminating any potential bias or distortion that missing data might cause. After the removal of missing rows, the shape of the dataset was printed again to verify that the process was successful and that the dataset no longer contained any missing values. The next task in preprocessing was feature engineering. One of the critical features in the dataset was the `duration` column, which contained both numerical and categorical information, such as "90 min" for movie durations or "2 Seasons" for TV shows. To handle this, a function called `extract_duration` was defined. This function took the `duration` string, split it into its numeric and unit components (e.g., extracting '90' and 'min' from "90 min"), and returned these as two separate values. These new features, `duration_numeric` and `duration_unit`, were then added to the dataset. Since some rows contained missing or incorrectly formatted duration data, these rows were dropped using `df.dropna(subset=['duration_numeric', 'duration_unit'], inplace=True)`, ensuring that only well-formed data was used for analysis. The `duration_numeric` feature was then converted to an integer data type for consistency in numerical operations.

After handling the `duration` column, the dataset was filtered to focus only on Movies. The rationale for this decision was that Movies have a more consistent and directly comparable duration compared to TV Shows, which can vary widely in terms of the number of seasons and episodes. To achieve this, the dataset was filtered by the `type` column, keeping only the rows where the `type` was labeled as 'Movie'. This reduced the dataset to a more manageable size and made the analysis more straightforward, as it allowed us to compare Movies with similar characteristics. Additionally, categorical variables such as the `rating` column, which includes several different values, were transformed into a format suitable for machine learning algorithms. Specifically, we performed one-hot encoding on the `rating` column to convert it into a set of binary features. The less frequent ratings were grouped into a single category labeled 'Other'. This grouping helped reduce the dimensionality and kept the analysis focused on the most common ratings. The `OneHotEncoder` from `sklearn` was used for this transformation, which produced a sparse matrix. This matrix was then converted into a dense DataFrame and merged with the original dataset.

Furthermore, numerical features such as `release_year` and the newly created `duration_numeric` were standardized using the `StandardScaler`. Scaling is important in clustering because features with larger ranges can disproportionately affect the clustering results. The `StandardScaler` transformed these features so that each would have a mean of zero and a standard deviation of one, ensuring that they contributed equally to the clustering process. At the end of the preprocessing phase, the cleaned dataset and the feature matrix were saved to disk as CSV files: 'netflix_movies_cleaned.csv' and 'netflix_movies_features.csv'. This saved data could be used for further analysis or as a checkpoint for future work.

3.3. Exploratory Data Analysis (EDA)

EDA was conducted to better understand the distribution of key features in the dataset and to visualize any patterns or trends that could inform the clustering process. The first step in EDA was to visualize the distribution of content types, specifically Movies versus TV Shows. A count plot was created using `seaborn.countplot`, which displayed the frequency of each content type. This provided a clear overview of the proportion of Movies and TV Shows in the dataset. Next, the distribution of `release_year` values was examined using a histogram with Kernel Density Estimation (KDE), which was created using `sns.histplot`. This visualization helped identify trends in the production of Netflix content over the years and allowed us to see the years with the highest concentration of releases.

Another important feature was the `rating` column, which was visualized through a count plot for the top 10 most common ratings. This plot highlighted the distribution of content ratings on Netflix, providing insights into the platform's content policy and the type of content typically available to users. To explore the duration of movies, a histogram of the `duration_numeric` feature was generated to show the distribution of movie lengths. This visualization helped identify common movie durations and detect any outliers in the dataset. For TV Shows, a separate count plot was created to examine the distribution of the number of seasons. This visualization provided insights into how Netflix organizes its TV Shows, showing whether most TV Shows have a few seasons or if there are many shows with a large number of seasons. The visualizations generated during the EDA phase were crucial for understanding the characteristics of the data. They provided valuable insights into the distribution and relationships of the dataset’s features and helped inform the clustering process by identifying potential patterns or trends that could be captured by the clustering algorithms.

3.4. Clustering Algorithms

Two clustering algorithms were applied to the dataset: K-means clustering and Hierarchical clustering. K-means clustering was used as the primary algorithm due to its effectiveness in partitioning data into distinct groups based on similarity. To determine the optimal number of clusters, the Elbow Method was employed. This method involves fitting the K-means algorithm for a range of cluster values, from 1 to 10, and plotting the inertia (the sum of squared distances from each point to its assigned cluster center). By examining the Elbow plot, the optimal number of clusters was chosen at the point where the inertia begins to decrease more slowly, indicating that additional clusters would not improve the model significantly. In this case, the Elbow Method suggested that four clusters were optimal, and K-means clustering was applied with this number of clusters. The clustering results were stored in a new column, `kmeans_cluster`, which assigned a cluster label to each movie. In addition to K-means, Hierarchical clustering was applied as an alternative method. Hierarchical clustering creates a tree-like structure called a dendrogram, which visually represents the relationships between data points and how they are merged into clusters. Due to computational constraints, a random sample of 1000 rows was selected from the dataset to create the dendrogram. The `ward` linkage method was used, which minimizes the variance within clusters. The dendrogram helped visualize the hierarchical structure of the data and provided insights into how the data points were grouped, allowing for the identification of clusters at different levels of granularity.

3.5. Visualization

Visualization played a key role in interpreting the results of the clustering analysis. To reduce the dimensionality of the feature space and visualize the clusters, PCA was applied. PCA is a technique that projects high-dimensional data into a lower-dimensional space while preserving as much variance as possible. In this case, PCA was used to reduce the feature space to two dimensions, making it possible to plot the clusters on a 2D scatter plot. The results of PCA were added to the DataFrame as two new columns, `pca1` and `pca2`, representing the first and second principal components. The scatter plot of the PCA results was then generated, with each data point colored according to its cluster label from K-means. This provided a clear and interpretable view of the clusters, helping to visualize how the Movies in the dataset were grouped based on their features. In addition to PCA, a dendrogram was created for the hierarchical clustering results. The dendrogram visually represented the merging process of hierarchical clustering and allowed for the identification of natural groupings in the data.

Finally, the trained models were saved for future use. The K-means model, which contained the cluster assignments and centroids, was saved to disk as 'kmeans_model.pkl' using `joblib.dump`. Similarly, the PCA model, which was used for dimensionality reduction, was saved as 'pca_model.pkl'. These models could be reloaded later and applied to new data for further analysis or clustering without needing to retrain them. The cleaned dataset and preprocessed features were also saved as CSV files to preserve the work done during the preprocessing phase. This detailed methodology ensured that the analysis was reproducible, and the results could be easily accessed and used for future research or business applications.

4. Results and Discussion

4.1. Results of Data Preprocessing and Feature Engineering

After successfully loading the dataset, it became evident that the original data contained 8807 entries with 13 columns. Upon further inspection, several columns had missing values, particularly in `director`, `cast`, `country`, `rating`, and `duration`. To ensure the integrity of the dataset for analysis, any rows with missing values were dropped, resulting in a dataset with 5332 entries. A focused subset of the dataset containing only Movies was then created, as Movies have consistent durations compared to TV Shows. The final subset used for the clustering analysis contained 5185 Movies. Feature engineering was a critical part of the data preparation process. The `duration` column, which originally contained mixed formats (such as "90 min" for movie durations and "2 Seasons" for TV shows), was split into two separate components: a numeric value (`duration_numeric`) and a unit of measurement (`duration_unit`). This allowed for more consistent treatment of the data in subsequent analysis. The Movies dataset was further processed by applying one-hot encoding to the `rating` feature, where less frequent ratings were grouped into an 'Other' category to reduce dimensionality. Afterward, numerical features, including `release_year` and `duration_numeric`, were standardized using `StandardScaler` to ensure that all features contributed equally to the clustering process. The cleaned and preprocessed data was saved to disk for future reference. The feature matrix, which included all relevant features for clustering, was ready for use in the next steps of the analysis.

4.2. Finding from Exploratory Data Analysis (EDA)

The EDA phase provided valuable insights into the distribution and patterns within the dataset. Several visualizations were generated to understand key features of the dataset. A count plot of content types revealed that the majority of entries were TV Shows, with Movies accounting for a smaller proportion (figure 1). A histogram of `release_year` (figure 2a) illustrated a significant concentration of content being added from 2010 onwards, with a noticeable peak around 2017, reflecting the surge in Netflix's content library during this period. The top 10 most common ratings were visualized using a count plot (figure 2b), which showed that TV-MA was the most frequent rating, followed by TV-14 and TV-PG. The distribution of movie durations was analyzed with a histogram (figure 3a), which revealed that most movies on Netflix fall within a duration range of 80 to 120 minutes, with few outliers. The distribution of TV Show seasons (figure 3b) showed that most shows had between 1 and 3 seasons, with a few shows featuring a larger number of seasons. These visualizations helped contextualize the dataset and provided a clearer understanding of the characteristics of the movies and TV shows within Netflix's content library, setting the stage for more detailed analysis with clustering.

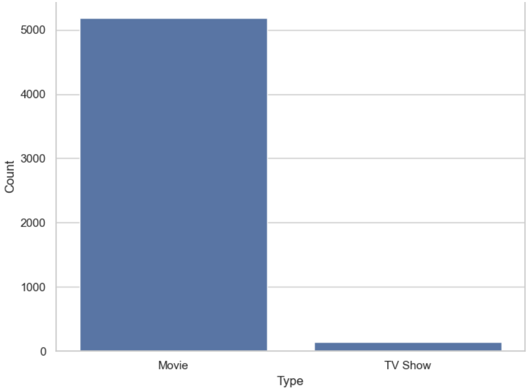

Figure 1. Distribution of Content Types on Netflix

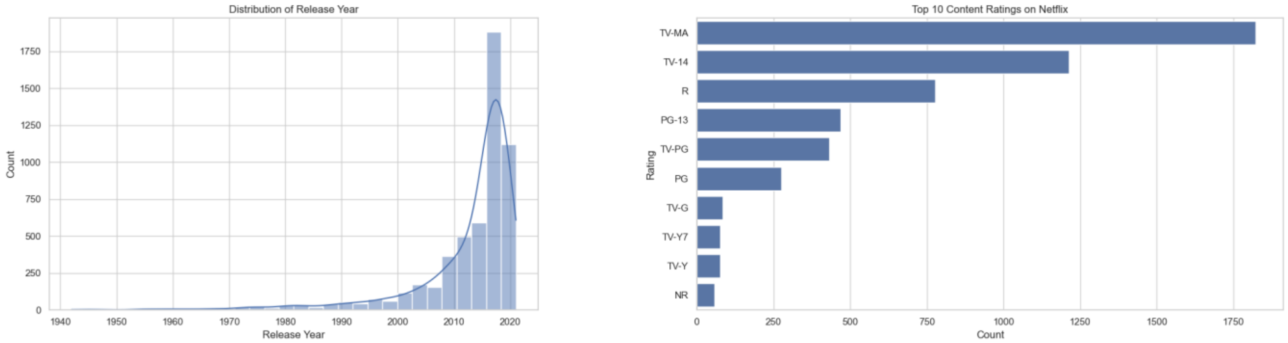

Figure 2. Distribution of (a) Release Year and (b) Top 10 Content Rating on Netflix

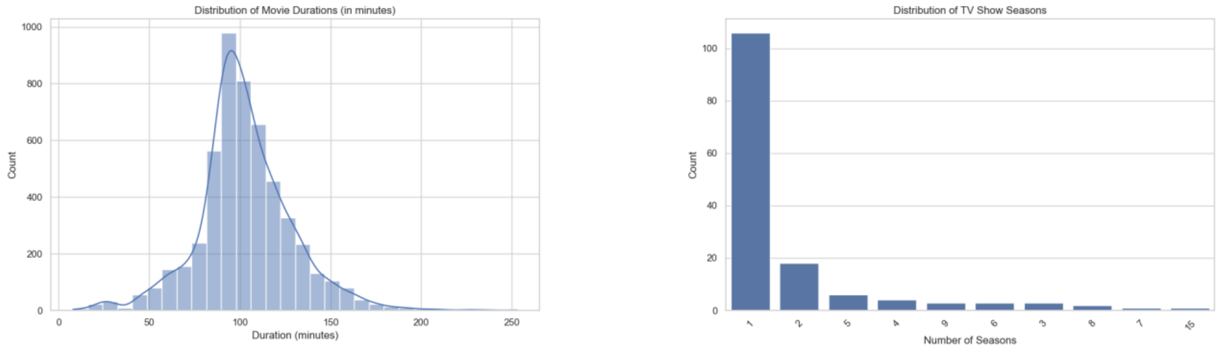

Figure 3. Distribution of (a) Movie Duration and (b) TV Show Seasons on Netflix

4.3. K-means Clustering Results

The K-means clustering algorithm was applied to the preprocessed feature matrix. The optimal number of clusters was determined using the Elbow Method (figure 4). By plotting the inertia values for different values of k, the "elbow" was observed at k=4, indicating that four clusters would best represent the underlying structure in the data. K-means clustering was then performed with k=4, and the resulting cluster labels were assigned to the movies in the dataset.

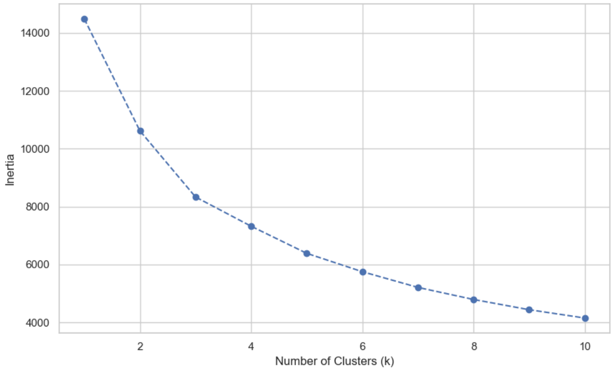

Figure 4. Elbow Method Plot

The four clusters identified by K-means were then analyzed based on several key characteristics, including the average release year, the average duration of movies (in minutes), and the most common rating within each cluster. The results of this analysis are summarized in the table 1.

Table 1. Summary of Cluster Analysis

Cluster	Avg. Release Year	Count	Avg. Duration (min)	Most Common Rating
0	2013.04	1163	133.7	TV-14
1	2016.14	666	61.61	TV-MA
2	2015.51	2921	98.12	TV-MA
3	1986.65	435	113.51	R

Cluster 0, which contained the largest number of Movies (1163), had an average release year of 2013 and an average movie duration of approximately 134 minutes. The most common rating for this cluster was TV-14, suggesting that these movies tend to be of moderate intensity, suitable for younger audiences with some content restrictions. Cluster 1, which had fewer movies (666), had an average release year of 2016 and an average duration of 61.6 minutes. The dominant rating in this cluster was TV-MA, indicating that these movies are likely to be more mature in content, with a higher intensity level compared to Cluster 0. Cluster 2 contained the largest number of Movies (2921), with an average release year of 2015 and a typical movie duration of 98 minutes. This cluster's most common rating was TV-MA, similar to Cluster 1, but it contained a broader range of content, both in terms of release year and movie length. Cluster 3, with 435 Movies, had an average release year of 1986 and a slightly longer average duration of 113.5 minutes. The most common rating in this cluster was 'R,' suggesting that these Movies were more intense and targeted at an adult audience, with more mature content and themes.

4.4. Hierarchical Clustering Results

In addition to K-means clustering, hierarchical clustering was performed using a random sample of 1000 movies. The results were visualized using a dendrogram (figure 5), which illustrated the hierarchical relationships between the data points. The dendrogram showed the agglomeration of movies into clusters at varying distances, and it provided a visual representation of how the movies were grouped based on their similarities. The hierarchical clustering method offered an alternative perspective on how the movies could be grouped. While K-means provided clear, predefined clusters, the dendrogram allowed for the exploration of different levels of grouping, offering a more flexible approach to cluster analysis. The dendrogram visualization helped to identify natural groupings in the data, which could complement the K-means results by showing finer distinctions between movies that were not captured by the K-means method.

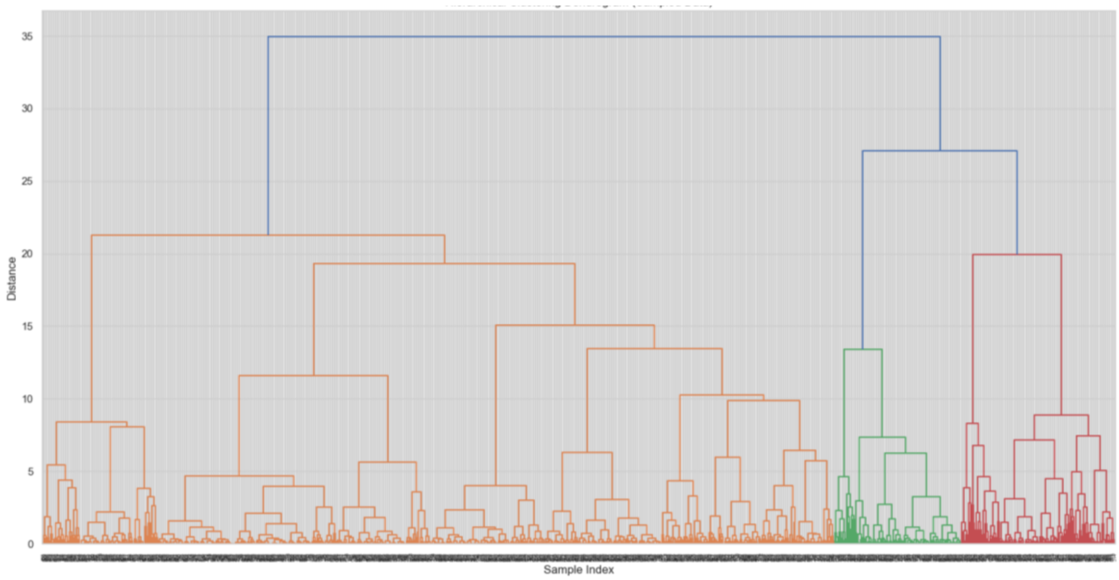

Figure 5. Dendogram Visualization of 1000 Sample of Movies

4.5. Visualizing Clusters with PCA

To further explore the clustering results and make them easier to interpret, PCA was applied to reduce the dimensionality of the data from the original high-dimensional space to two dimensions. The transformed data was then visualized in a 2D scatter plot (figure 6), with each movie color-coded by its assigned cluster label from K-means.

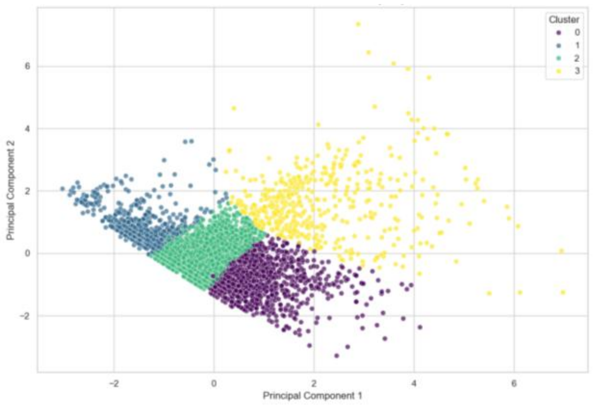

Figure 6. K-Means Clusters of Netflix Movies (PCA)

The resulting plot provided a clear visual representation of the four clusters in two dimensions, making it easier to understand the relationship between the different clusters and the distribution of movies across these groups. The PCA plot revealed that the movies in each cluster were well-separated, suggesting that the K-means algorithm had effectively grouped the movies based on their features. The visual representation confirmed the distinctiveness of the clusters identified through K-means and provided further evidence that the clusters represented meaningful patterns in the dataset.

4.6. Discussion

The clustering results reveal distinct patterns in the Netflix movie catalog, providing valuable insights into how different movies can be grouped based on their features. For instance, Movies in Cluster 0 tend to have longer durations and are most commonly rated TV-14, suggesting that these Movies are typically family-friendly, with a moderate level of content intensity. In contrast, Cluster 1, which contains shorter Movies, is dominated by TV-MA rated content, indicating that these Movies are more mature and intense. The analysis also highlighted that Movies in Cluster 2, with a similar rating of TV-MA, have a broader range of durations and represent a more diverse group of movies, likely spanning various genres. Lastly, Cluster 3, with its 'R' rated Movies, tends to focus on adult-oriented content with longer durations, predominantly from older release years, further highlighting the variation in content type and length.

These cluster patterns are significant when viewed in the context of Netflix’s content strategy. The segmentation of movies into clusters based on their ratings, durations, and release years suggests that Netflix tailors its content offerings to different audience preferences. The family-friendly Movies in Cluster 0, with longer durations, may align with Netflix's strategy to offer wholesome content suitable for a wider demographic, such as children and families. The presence of shorter, mature content in Cluster 1 reflects Netflix’s emphasis on catering to adult audiences who prefer quick, intense entertainment, while Cluster 2 reveals Netflix's strategy to offer diverse movies with a consistent TV-MA rating, appealing to a broad adult audience with varied tastes. Cluster 3, focusing on older, more adult-oriented Movies, could be a strategy to retain long-term subscribers by offering a catalog with both classic and newer adult content.

The interpretation of these clusters also provides insights into Netflix's genre distribution and viewer preferences. The dominant TV-MA and 'R' ratings in several clusters indicate that Netflix has successfully tapped into the adult and mature content market, catering to diverse tastes and preferences. The distribution of movie durations across clusters suggests that Netflix provides a variety of content lengths to accommodate different viewing habits, from short-form entertainment to longer, more immersive experiences. This segmentation helps Netflix fine-tune its recommendations and content acquisition strategies, ensuring it continues to attract and retain a diverse viewer base by offering a wide range of movies tailored to different tastes, from family-friendly to mature and nostalgic content.

5. Conclusion

The clustering analysis of Netflix's movie catalog revealed several important patterns that provide insights into the platform's content structure. Four distinct clusters were identified, each characterized by unique features such as movie duration, release year, and content rating. Cluster 0 consisted of longer, family-friendly Movies with a TV-14 rating, while Cluster 1 contained shorter, more mature content with a TV-MA rating. Cluster 2 represented a broad range of Movies with TV-MA ratings and moderate durations, reflecting diverse genres, while Cluster 3 focused on adult-oriented, longer Movies predominantly from the 1980s, rated 'R'. These clusters highlight the varied nature of Netflix's movie offerings, catering to different viewer preferences based on content intensity, duration, and age-appropriateness.

The findings from this study have significant implications for Netflix's content strategy and recommendation system. The identification of distinct clusters allows Netflix to better understand its content catalog and audience segmentation, potentially influencing its content acquisition and production strategies. The presence of family-friendly content, diverse adult options, and classic adult Movies can help Netflix refine its recommendations to cater to the specific needs of its subscribers. By leveraging these clusters, Netflix can enhance its recommendation algorithms, ensuring more personalized and relevant content suggestions for users based on their preferences for movie length, rating, and genre.

However, there are certain limitations in this study that must be acknowledged. The dataset used for clustering is based solely on metadata, such as release year, duration, and rating, without taking into account other potentially important factors like user ratings, viewership data, or content popularity. Additionally, the algorithms used for clustering, such as K-means and hierarchical clustering, have limitations in terms of handling high-dimensional data and may not always identify the most meaningful clusters, especially if the features are not sufficiently representative. In future research, it would be valuable to integrate additional features, such as user ratings or viewing history, to create a more comprehensive understanding of content preferences. Moreover, applying deep learning techniques, such as autoencoders or neural networks, could provide a more nuanced clustering approach, potentially revealing even deeper patterns in Netflix's content.

6. Declarations

6.1. Author Contributions

Conceptualization: B.H.H., E.P.; Methodology: B.H.H., E.P.; Software: B.H.H.; Validation: E.P.; Formal Analysis: B.H.H.; Investigation: B.H.H.; Resources: E.P.; Data Curation: B.H.H.; Writing – Original Draft Preparation: B.H.H.; Writing – Review and Editing: E.P.; Visualization: B.H.H.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] M. P. Sarrionandia and A. Sarrionandia, “Mental Health, Violence, Suicide, Self-Harm, and HIV in Series and Films of Netflix: Content Analysis and Its Possible Impacts on Society,” *Frontiers in Communication*, 2023, doi: 10.3389/fcomm.2023.1243394.

[2] K. Alfayad, R. Murray, J. Britton, and A. B. Barker, “Content Analysis of Netflix and Amazon Prime Instant Video Original Films in the UK for Alcohol, Tobacco and Junk Food Imagery,” *Journal of Public Health*, 2021, doi: 10.1093/pubmed/fdab022.

[3] M. Yu, M. C. Carter, D. P. Cingel, and J. B. Ruiz, “How Sex Is Referenced in Netflix Original, Adolescent-Directed Series: A Content Analysis of Subtitles.,” *Psychology of Popular Media*, 2024, doi: 10.1037/ppm0000457.

[4] K. S. Ashwini, C. P. Shantala, and T. Jan, “Impact of Text Representation Techniques on Clustering Models,” 2022, doi: 10.21203/rs.3.rs-1385057/v1.

[5] D. Floegel, “Labor, Classification and Productions of Culture on Netflix,” *Journal of Documentation*, 2020, doi: 10.1108/jd-06-2020-0108.

[6] N. C. Crossman, S. M. Chung, and V. A. Schmidt, “Stream Clustering and Visualization of Geotagged Text Data for Crisis Management,” 2019, doi: 10.1109/icodse48700.2019.9092760.

[7] S. Tauty, P. Martin, A. Bourmaud, B. Chapoton, É. de La Rochebrochard, and C. Alberti, "Sexual Health Promotion Messages for Young People in Netflix Most-Watched Series Content (2015–2020): Mixed-Methods Analysis Study," *BMJ Open*, 2021, doi: 10.1136/bmjopen-2021-052826.

[8] A. M. Wahid, T. Hariguna, and G. Karyono, "Optimization of Recommender Systems for Image-Based Website Themes Using Transfer Learning," *Journal of Applied Data Sciences*, vol. 6, no. 2, Art. no. 2, Mar. 2025, doi: 10.47738/jads.v6i2.671.

[9] D. Andra and A. B. Baizal, "E-Commerce Recommender System Using PCA and K-Means Clustering," *Jurnal Resti (Rekayasa Sistem Dan Teknologi Informasi)*, 2022, doi: 10.29207/resti.v6i1.3782.

[10] M. N. Mu'afa and Z. K. A. Baizal, "Implementation of Dimensionality Reduction With SVD to Improve Rating Prediction in Recommender System," *Journal of Computer System and Informatics (Josyc)*, 2022, doi: 10.47065/josyc.v3i4.2110.

[11] P. Chen, L. Dong, and Y. Liu, "Design of Music Recommendation System Based on EDA and K-Means Cluster Analysis," *Applied and Computational Engineering*, 2024, doi: 10.54254/2755-2721/50/20241695.

[12] A. Darvishy, H. Ibrahim, F. Sidi, and A. Mustapha, "A Customized Non-Exclusive Clustering Algorithm for News Recommendation Systems," *Journal of University of Babylon for Pure and Applied Sciences*, 2019, doi: 10.29196/jubpas.v27i1.2192.

[13] H. Zarzour, F. Maazouzi, M. Soltani, and C. Chemam, "An Improved Collaborative Filtering Recommendation Algorithm for Big Data," 2018, doi: 10.1007/978-3-319-89743-1_56.

[14] T. Zhu and Y. Han, "Enhancing Beer Recommendations Through Clustering: A Comparison of Hierarchical and K-Means Clustering Methods on Normalized Data," 2023, doi: 10.4108/eai.26-5-2023.2334332.

[15] D. Murthy, J. Lee, H. Dashtian, and G. Kong, "Influence of User Profile Attributes on E-Cigarette–Related Searches on YouTube: Machine Learning Clustering and Classification," *Jmir Infodemiology*, 2023, doi: 10.2196/42218.

[16] A. Benis, R. Barkan, T. Sela, and N. Harel, "Communication Behavior Changes Between Patients With Diabetes and Healthcare Providers Over 9 Years: Retrospective Cohort Study," *Journal of Medical Internet Research*, 2020, doi: 10.2196/17186.

[17] H. Huang and Y. Ma, "A Hybrid Clustering Approach for Bag-of-Words Image Categorization," *Mathematical Problems in Engineering*, 2019, doi: 10.1155/2019/4275720.

[18] K. P. Sinaga and M. Yang, "Unsupervised K-Means Clustering Algorithm," *Ieee Access*, 2020, doi: 10.1109/access.2020.2988796.

[19] A. R. Lubis, M. Lubis, and A.-K. Al-Khowarizmi, "Optimization of Distance Formula in K-Nearest Neighbor Method," *Bulletin of Electrical Engineering and Informatics*, 2020, doi: 10.11591/eei.v9i1.1464.

[20] F. Ros and S. Guillaume, "A Hierarchical Clustering Algorithm and an Improvement of the Single Linkage Criterion to Deal With Noise," *Expert Systems With Applications*, 2019, doi: 10.1016/j.eswa.2019.03.031.

[21] A. Alzahrani, "Impact of Dataset Scaling on Hierarchical Clustering: A Comparative Analysis of Distance-Based and Ratio-Based Methods," *International Journal of Analysis and Applications*, 2024, doi: 10.28924/2291-8639-22-2024-36.

[22] X. Chen, S. Li, and L. Tang, "Media DAO: A Blockchain-Based Model for the Transformation and Innovation of the Publishing and Media Industry," *Editing Practice*, 2023, doi: 10.54844/ep.2023.0418.

[23] F. García-Rapp, "Popularity Markers on YouTube's Attention Economy: The Case of Bubzbeauty," *Celebrity Studies*, 2016, doi: 10.1080/19392397.2016.1242430.

[24] W.-B. Xie, Y.-L. Lee, C. Wang, D. Chen, and T. Zhou, "Hierarchical Clustering Supported by Reciprocal Nearest Neighbors," *Information Sciences*, 2020, doi: 10.1016/j.ins.2020.04.016.

[25] B. Clarke, S. Amiri, and J. Clarke, "EnsCat: Clustering of Categorical Data via Ensembling," *BMC Bioinformatics*, 2016, doi: 10.1186/s12859-016-1245-9.