Introduction to AI

Movie Watch Pattern Clustering

A PROJECT REPORT

Submitted by

SALONI SINGH

202401100300211

In partial fulfillment for the award of the degree of

Bachelor of Technology

CSE(AI)

KIET GROUP OF INSTITUTIONS

INTRODUCTION:

In the era of digital streaming and personalized content delivery, understanding user behavior has become paramount. Users interact with movie platforms in diverse ways—some prefer action films in the afternoon, others enjoy thrillers late at night, while some are particularly generous or critical with their ratings. Identifying these patterns can offer streaming platforms a competitive edge through personalized recommendations, targeted marketing, and optimized user engagement strategies.

This project focuses on clustering users based on three critical aspects of their movie-watching behavior:

- The time of day they typically watch movies
- Their preferred movie genres
- Their rating tendencies

Using unsupervised machine learning, specifically K-Means clustering, this project segments users into distinct groups with similar behavior patterns. By analyzing these clusters, we aim to uncover actionable insights that can improve the user experience and guide business decisions in media and entertainment platforms.

METHODOLOGY:

This project follows a structured data science workflow, involving data preprocessing, feature engineering, clustering using K-Means, and result interpretation. The steps are detailed below:

---

1. Data Collection

The dataset, `movie_watch.csv`, contains 100 user records with the following features:

- `watch_time_hour`: Integer value representing the hour of the day (0–23) the user watches movies
- `genre_preference`: Categorical variable indicating the user's most watched genre
- `avg_rating_given`: Float representing the user's average rating for watched movies

---

2. Data Preprocessing

a. Data Cleaning

- Verified the dataset for missing values or anomalies; no missing data was found.

b. Encoding Categorical Variables

- The `genre_preference` column was converted to numeric form using **One-Hot Encoding**, resulting in separate binary columns for each genre (e.g., `genre_action`, `genre_comedy`, `genre_thriller`).

c. Feature Scaling

- Standardized numerical columns (`watch_time_hour` and `avg_rating_given`) using **StandardScaler** to normalize feature values for effective distance-based clustering.

---

3. Feature Engineering

The final feature matrix used for clustering contained:

- Scaled `watch_time_hour`
- Scaled `avg_rating_given`
- One-hot encoded genre features

This matrix ensures all features are on a comparable scale, making the clustering algorithm more effective.

---

4. Clustering Algorithm: K-Means

a. Algorithm Selection

- **K-Means Clustering** was chosen for its simplicity, scalability, and effectiveness with continuous numerical features.

b. Number of Clusters (k)

- The optimal number of clusters was determined using the **Elbow Method**, analyzing the Within-Cluster-Sum-of-Squares (WCSS).
- Based on this method, **k = 3** was selected.

c. Model Training

- The K-Means model was trained on the feature matrix with `n_clusters=3` and `random_state=42` for reproducibility.

d. Label Assignment

- Each user was assigned a cluster label (0, 1, or 2) representing their behavioral group.

---

5. Dimensionality Reduction for Visualization

To visualize high-dimensional cluster data, **Principal Component Analysis (PCA)** was used:

- Reduced the 5-dimensional feature space to 2 principal components (`PC1`, `PC2`)
- Plotted users in 2D space, colored by cluster labels, to observe separability and cluster compactness

---

6. Cluster Analysis

Post-clustering, each cluster was analyzed based on the original features:

- Mean and distribution of `watch_time_hour`, `avg_rating_given`
- Dominant genre preferences
- Cluster size and spread

---

7. Insights and Interpretation

Each cluster was profiled to understand user behavior patterns:

- Time-based viewing habits
- Genre preferences
- Rating tendencies

These profiles provided actionable insights for content recommendation, scheduling, and targeted engagement.

CODE:

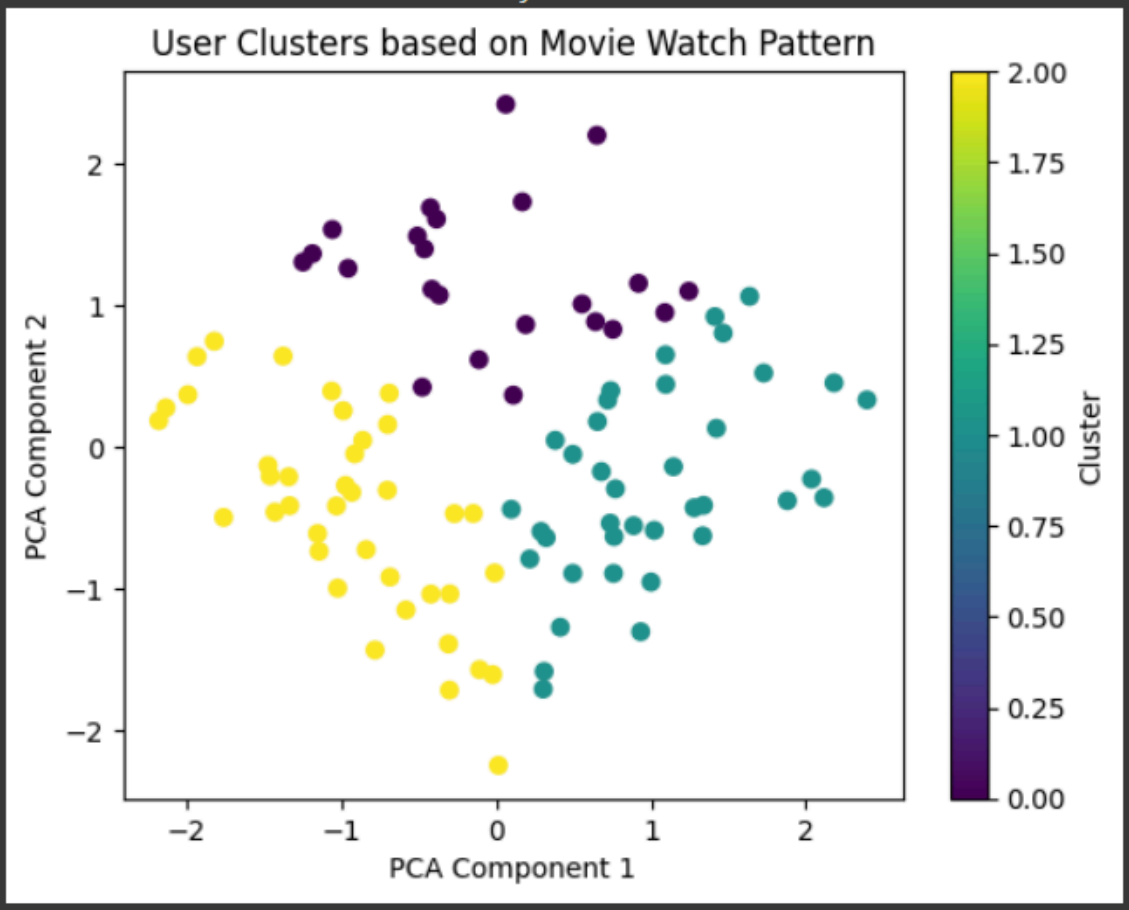
```
import pandas as pd

from sklearn.preprocessing import StandardScaler, OneHotEncoder

from sklearn.compose import ColumnTransformer

from sklearn.pipeline import Pipeline

from sklearn.cluster import KMeans

import matplotlib.pyplot as plt



# Load the data

df = pd.read_csv("/content/movie_watch.csv")



# Define preprocessing for numerical and categorical columns

numeric_features = ['watch_time_hour', 'avg_rating_given']

categorical_features = ['genre_preference']



# Create a column transformer

preprocessor = ColumnTransformer(

    transformers=[

        ('num', StandardScaler(), numeric_features),

        ('cat', OneHotEncoder(), categorical_features)

    ]

)
```

```
# Create a pipeline that first transforms the data then applies KMeans
pipeline = Pipeline([
    ('preprocessor', preprocessor),
    ('clusterer', KMeans(n_clusters=3, random_state=42))
])

# Fit the pipeline
pipeline.fit(df)

# Predict clusters
df['cluster'] = pipeline.predict(df)

# Display cluster assignment
print(df.head())

# Optional: Plotting (only 2D example using PCA)
from sklearn.decomposition import PCA

# Reduce dimensions for visualization
X_transformed = pipeline.named_steps['preprocessor'].transform(df)
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_transformed)
```

```
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=df['cluster'], cmap='viridis')

plt.title("User Clusters based on Movie Watch Pattern")

plt.xlabel("PCA Component 1")

plt.ylabel("PCA Component 2")

plt.colorbar(label='Cluster')

plt.show()
```

OUTPUT/RESULT:

	watch_time_hour	genre_preference	avg_rating_given	cluster
0	13	action	2.037554	2
1	4	comedy	1.350365	2
2	15	thriller	1.359665	2
3	14	thriller	1.772998	2
4	14	comedy	1.202237	2



REFERENCES/CREDITS:

The kiet logo is used from the official kiet portal.
The following data set is used:
Kaggle:https://drive.google.com/file/d/1RknKkLNFSrpFfZpYRozOBaoRMn4RYqn7/view?usp=sharing

The Movies Dataset
Comprehensive metadata on over 45,000 movies, including genres, release dates, and user ratings. Useful for analyzing genre preferences and rating behaviors.
🔗 Access Dataset

Netflix Movies and TV Shows Clustering
Dataset focusing on clustering Netflix content based on various features. Can provide insights into genre preferences.
🔗 Access Dataset

Movie Dataset
Contains information on 1,682 Hollywood movies, including genres and ratings. Useful for analyzing user preferences.
🔗 Access DatasetKaggle+11Kaggle+11Kaggle+11

Movie Rating Data Set
Includes movie ratings and genres, allowing for analysis of rating behaviors across different genres.
🔗 Access DatasetKaggle

Top Rated Movie Dataset
Dataset of top-rated movies with ratings, genres, and release years for analysis.
🔗 Access Dataset