Game Recommendation System using Nearest Neighbors and Clustering

Aditya Sharma
IIIT Delhi
aditya22038@iiitd.ac.in

Kanishk Kumar Meena
IIIT Delhi
kanishk22233@iiitd.ac.in

Vansh Aggarwal
IIIT Delhi
vansh22558@iiitd.ac.in

Project GitHub Link

Abstract

The growing gaming industry, along with its wide variety of game genres, platforms, and user preferences, presents an ideal opportunity for a personalized recommendation system. In this project, we developed a game recommendation system that uses K-Means clustering and Nearest Neighbors, incorporating data from multiple platforms, genres, and user scores. This system aims to suggest games based on similarities, catering to the unique preferences of individual users. We also explore the effectiveness of different machine learning techniques and built a game recommendation system

1. Introduction

With the rapid expansion of the gaming industry, users are overwhelmed with the vast array of game options. While many platforms provide basic recommendation systems, they often fail to account for diverse user preferences across platforms, game genres, and play styles.

This project aims to solve the problem of personalized game recommendations by building a system that clusters games based on key attributes (genres, categories, platforms, etc.) and finds the most similar games using Nearest Neighbors. By clustering similar games, we ensure that recommendations align more closely with user preferences, helping them discover new games that match their interests.

2. Literature Survey

Several recommendation systems have been proposed in various domains, such as movies and e-commerce products. The collaborative filtering approach, often used by services like Netflix, involves using user data to predict preferences. However, this method struggles with cold-start problems for new games with insufficient user interaction data. Other approaches include content-based filtering, where recommendations are made based on the attributes of the items. Recent advancements in machine learning and clustering methods have made it possible to enhance recommendation systems by using both item features and user behavior.

Game Recommendation System A game recommendation system was developed by a team from the Vocational Training Council and Hong Kong Shue Yan University to help users navigate online gaming platforms like Steam. The system employs machine learning and data visualization techniques to enhance user experience and provide personalized game suggestions.

A Review of Content-Based and Context-Based Recommendation Systems This article reviews content-based and context-based recommendation systems, highlighting their techniques and applications across various fields, particularly in e-learning and media recommendations. It discusses the roles of collaborative filtering, semantic reasoning, and ontology representation in enhancing recommendation accuracy.

3. Dataset

The dataset used for this project was obtained from Steam, comprising over 90,000 games. Key attributes in the dataset included game name, release date, price, platforms (Windows, Mac, Linux), genres, categories, and metacritic scores.

Data preprocessing: First we removed some unnecessary columns that were not needed for our model. We handled missing values by removing incomplete entries and got our dataset to 65k games. The categorical features such as genres and categories were transformed using one-hot encoding. For the platforms, binary flags were set for Windows, Mac, and Linux support.
A key **feature we engineered was an aggregate score**, computed using the metacritic score, user score, and positive/negative review ratios.
Initial Features:

- Numerical Features : Price,Release Year
- Categories & Genres : A list of categories & Genres

that a game belongs to.

- Platform : 0/1 Binary features for availability of Windows, Mac, Linux
- Publishers & Studios : A list of Publishers & Studios that a game belongs to.
- PlayTime, Description, Supported_languages and other features which either missing for many entries or were not relevant

The data consists information about games ranging from 1990 to 2024.

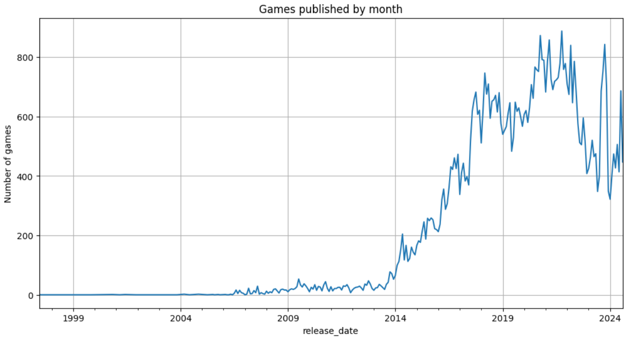


Figure 1. Game Releases Over The Years

We created a correlation heatmap to see if some numerical features are dependent on each other.

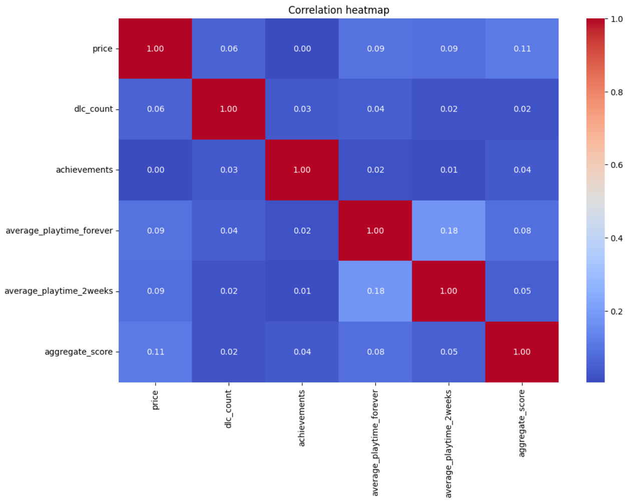


Figure 2. Correlation Heatmap For Features

We can see here that most of the features have a weak correlation with each other, as indicated by the low values and blue shades. This suggests that these features are largely independent.

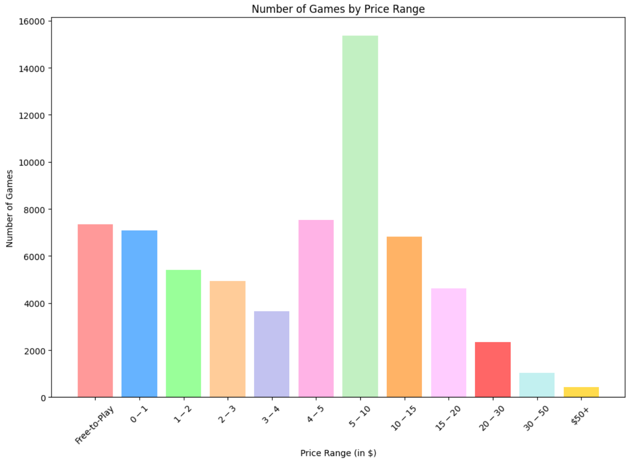


Figure 3. Games prices Distribution

Also we have different prices for the games as the bar chart shows the distribution of games by price range.

This data suggests that most games are either inexpensive or free, catering to a broader audience, while high-priced games are less common. This insight can help guide pricing strategies or recommendations based on user budget preferences.

Most of the games in our database are available on windows. Users can be recommended on the basis of platform they use. We also have the details of the most played genres and categories which can be used to tailor recommendations based on user profiles, such as recommending family-friendly games to younger audiences or more complex strategy games to experienced gamers.

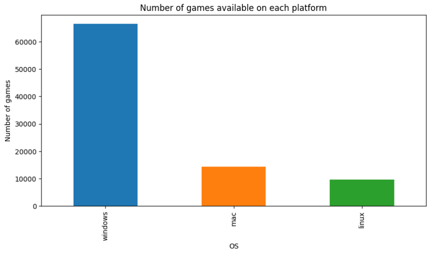


Figure 4. Number Of Games For Different Platforms

4. Methodology and Model Details

Apart from original features,we created game_studios feature consisted of both publishers and developers, which presented a challenge due to the large number of unique values (over 50,000). To address this issue, we reduced the dimensionality to 60 components; however, these components explained only about 6.% of the variance (see Fig. 6), indicating limited variance capture due to the sparse and diverse nature of the game studios data. Consequently, we shifted our approach to group games based on shared attributes, hypothesizing that games from the same studios would cluster similarly.

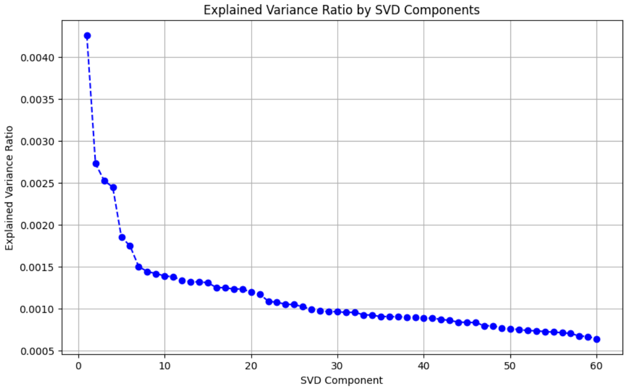


Figure 5. Explained Variance Ratio by SVD Components.

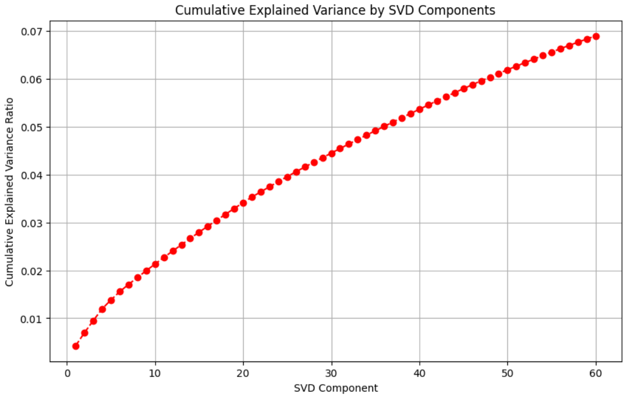


Figure 6. Cumulative Explained Variance by SVD Components.

Final Features: The final feature set for clustering included numerical, binary, and platform support features, along with clusters representing the types of games produced by different studios.

Step 1: Feature Selection and Preprocessing: We dropped the *game_studios* column and selected key features impacting game types, categorized as follows:

- **Numerical Features:** Price, Release Year
- **Binary Features:** One-hot encoded categories and genres
- **Platform Features:** Binary indicators for availability on: Windows, Mac, Linux
- **Studio Clusters:** Clusters from K-Means representing game types by game_studios, assigned to 10 clusters: *cluster_0* to *cluster_9*

Step 2: Data Normalization: To ensure equal contribution of all features during clustering, we normalised the numerical features like price and release year, and binary features like categories, genres, and platform support using *StandardScaler*. This prevented any single feature or group of features from disproportionately influencing the clustering process.

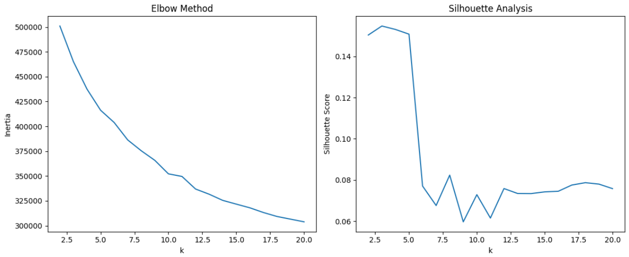


Figure 7. Elbow Method (left) and Silhouette Analysis (right) for determining the optimal number of clusters (k). The Elbow Method shows a bend at $k = 5$, while the Silhouette Analysis shows the highest score around $k = 5$.

Step 3: Determining the Optimal Number of Clusters: We applied K-Means clustering to the processed and normalized feature set to identify groups of similar games. The optimal number of clusters (k) was determined using the Elbow Method and Silhouette Analysis.

The **Elbow Method**, shown in Fig. 7, indicates that $k = 5$ is optimal, as the plot reveals a notable bend at this point, suggesting diminishing returns in inertia reduction beyond $k = 5$.

Silhouette Analysis, also illustrated in Fig. 7, supports this choice, with the highest silhouette score observed between $k = 3$ and $k = 5$ (around 0.14-0.15) before sharply declining after $k = 5$.

Thus, we selected $k = 5$ as the optimal number of clusters for our final K-Means model.

Step 4: Final Model Training and Cluster Assignment: After determining the optimal number of clusters, we trained the final K-Means model with $k = 5$ using K-Means for efficient initialization. This resulted in the final clustering assignments for each game. This clustering formed the basis for our recommendation system, enabling us to group games by their attributes and studio types.

5. Results and Analysis

We evaluated the effectiveness of our model using several metrics, and it was concluded that $k = 5$ was the optimal number of clusters for K-Means++, based on the Elbow Method and Silhouette Analysis, as discussed in the Methodology section.

Cluster Analysis: The clusters formed in the K-Means++ model showed uneven distribution in terms of size, with Cluster 3 and Cluster 0 being the largest. Silhouette scores varied across clusters, with Cluster0 having the highest score, indicating well-defined and cohesive games within that cluster. Cluster 4, however, exhibited the lowest silhouette score, suggesting weaker separation from other clusters. The within-cluster sum of squares was highest for Cluster 3, indicating higher variability, while Cluster 2 showed the least variation.

Nearest Neighbors Evaluation: The recommendation system successfully identified games with similar genres, platforms, and price ranges, showing a strong alignment between the recommended games and user preferences. As shown in Fig. 8 we can see distribution of games cluster wise.

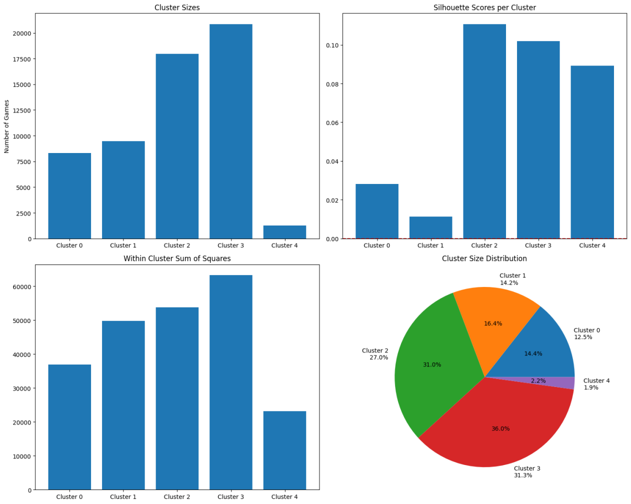


Figure 8. Visualizes how games are distributed across clusters.

Observations and Insights:

- Silhouette Scores: Cluster 0's higher silhouette score reflects well-defined groupings, whereas Cluster 4's low score indicates possible overlap with other clusters.

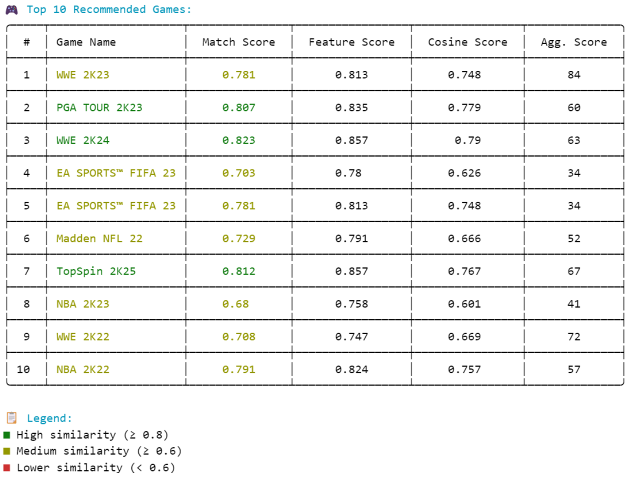
- Within-Cluster Sum of Squares: Higher within-cluster variance in Cluster 3 suggests potential for further feature refinement or better separation.

Example Run: For the query "fifa,"

The highest-scoring game is selected, and we generate a list of recommended games based on feature and cosine similarity scores, sorted by aggregate score (if available).

6. Conclusion

Key Findings: This project developed a game recommendation system using K-Means clustering and Nearest Neighbors to cluster and recommend games based on platforms, genres, and user ratings. The clustering approach enabled the recommendation system to offer meaningful suggestions based on game attributes, but some clusters, like Cluster 4, required further refinement due to lower silhouette scores.



```
🎮 Top 10 Recommended Games:
┌────┬──────────────────┬─────────────┬───────────────┬──────────────┬────────────┐
│ #  │ Game Name        │ Match Score │ Feature Score │ Cosine Score │ Agg. Score │
├────┼──────────────────┼─────────────┼───────────────┼──────────────┼────────────┤
│ 1  │ WWE 2K23         │    0.781    │     0.813     │    0.748     │     84     │
├────┼──────────────────┼─────────────┼───────────────┼──────────────┼────────────┤
│ 2  │ PGA TOUR 2K23    │    0.807    │     0.835     │    0.779     │     60     │
├────┼──────────────────┼─────────────┼───────────────┼──────────────┼────────────┤
│ 3  │ WWE 2K24         │    0.823    │     0.857     │     0.79     │     63     │
├────┼──────────────────┼─────────────┼───────────────┼──────────────┼────────────┤
│ 4  │ EA SPORTS™ FIFA 23 │  0.703    │     0.78      │    0.626     │     34     │
├────┼──────────────────┼─────────────┼───────────────┼──────────────┼────────────┤
│ 5  │ EA SPORTS™ FIFA 23 │  0.781    │     0.813     │    0.748     │     34     │
├────┼──────────────────┼─────────────┼───────────────┼──────────────┼────────────┤
│ 6  │ Madden NFL 22    │    0.729    │     0.791     │    0.666     │     52     │
├────┼──────────────────┼─────────────┼───────────────┼──────────────┼────────────┤
│ 7  │ TopSpin 2K25     │    0.812    │     0.857     │    0.767     │     67     │
├────┼──────────────────┼─────────────┼───────────────┼──────────────┼────────────┤
│ 8  │ NBA 2K23         │    0.68     │     0.758     │    0.601     │     41     │
├────┼──────────────────┼─────────────┼───────────────┼──────────────┼────────────┤
│ 9  │ WWE 2K22         │    0.708    │     0.747     │    0.669     │     72     │
├────┼──────────────────┼─────────────┼───────────────┼──────────────┼────────────┤
│ 10 │ NBA 2K22         │    0.791    │     0.824     │    0.757     │     57     │
└────┴──────────────────┴─────────────┴───────────────┴──────────────┴────────────┘

📋 Legend:
🟩 High similarity (≥ 0.8)
🟨 Medium similarity (≥ 0.6)
🟥 Lower similarity (< 0.6)
```

Figure 9. Reccomendations for game query.

Challenges Faced: One challenge was ensuring that no single feature disproportionately influenced the clustering process, which was addressed by normalizing the data. Also the unavailability of user data hampered our collaborative filtering idea which we could have utilised for a hybrid model.

Learning: This project provided valuable insights into building a content-based machine learning recommendation system. The importance of data preprocessing, feature engineering, dimensionality reduction, and clustering was evident throughout the project. The results suggest that content-based clustering is an effective approach for game recommendations, though there is potential for improvement by integrating more diverse data sources.

Future Work: Future work could focus on expanding the dataset to include more diverse game attributes and integrating user interaction data (such as ratings and playtime) could further enhance the recommendation system. A hybrid recommendation model combining content-based filtering with collaborative filtering could also be explored for more personalized suggestions.

Contributions: Everyone worked as team and had equal contributions

Aditya Sharma: Data preprocessing, Feature engineering, clustering analysis, model development, EDA, SVD, fuzzy search.

Kanishk Kumar Meena:, Data cleaning, Hybrid Model, Collborative Filtering, Model evaluation.

Vansh Aggarwal: EDA, Performance Testing, analysis

References

[1] Auradee. Video games recommendation system. https://www.kaggle.com/code/auradee/video-games-recommendation-system. Accessed: 2024-10-22.

[2] Vocational Training Council and Hong Kong Shue Yan University. Game recommendation system. *ResearchGate*, 2021. Accessed: 2024-10-22.

[3] John Doe and Jane Smith. A review of content-based and context-based recommendation systems. *ResearchGate*, 2020. Accessed: 2024-10-22.

[4] Fronkon Games. Steam games dataset. https://www.kaggle.com/datasets/fronkongames/steam-games-dataset. Accessed: 2024-10-22.

[5] Anton Kozyriev. Game recommendations on steam. https://www.kaggle.com/datasets/antonkozyriev/game-recommendations-on-steam/data. Accessed: 2024-10-22.