A PROJECT REPORT
On

"MOVIE RECOMMENDATION SYSTEM USING KNN ALGORITHM AND HYPERPARAMETER TUNING USING GRID SEARCH"

Submitted to
KIIT Deemed to be University

In Partial Fulfillment of the Requirement for the Award of

BACHELOR'S DEGREE IN COMPUTER SCIENCE AND ENGINEERING

BY

Pratham Halder	21052089
Aditya Kumar	21052131
Riya Raj	21052238
Addya Tiwari	21052218

UNDER THE GUIDANCE OF

Prof. Bhabani Shankar Prasad Mishra

SCHOOL OF COMPUTER ENGINEERING

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

BHUBANESWAR, ODISHA - 751024

April 2024

KIIT Deemed to be University

School of Computer Engineering
Bhubaneswar, ODISHA 751024

CERTIFICATE

This is certify that the project entitled

"MOVIE RECOMMENDATION SYSTEM USING KNN ALGORITHM AND HYPERPARAMETER TUNING USING GRID SEARCH"

submitted by

Pratham Halder	21052089
Aditya Kumar	21052131
Riya Raj	21052238
Addya Tiwari	21052218

is a record of bonafide work carried out by them, in the partial fulfillment of the requirement for the award of Degree of Bachelor of Engineering (Computer Science & Engineering) at KIIT Deemed to be university, Bhubaneswar. This work is done during the year 2023-2024, under our guidance.

Date: 25/04/2024

Prof. Bhabani Shankar Prasad Mishra
Project Guide

Acknowledgments

We are profoundly grateful to **Dr. Bhabani Shankar Prasad Mishra** of **Kalinga Institute of Industrial technology** for his expert guidance and continuous encouragement throughout to see that this project meets its target since its commencement to its completion. ....

PRATHAM HALDER

ADITYA KUMAR

RIYA RAJ

ADDYA TIWARI

CONTENTS

List of Figures

MOVIE RECOMMENDATION SYSTEM USING KNN ALGORITHM AND HYPERPARAMETER TUNING USING GRID SEARCH

Name: *Pratham Halder*
Dept: *School of Computer Science Engineering*
College: *KIIT University, Bhubaneswar*
21052089@kiit.ac.in

Name: *Aditya Kumar*
Dept: *School of Computer Science Engineering*
College: *KIIT University, Bhubaneswar*
21052131@kiit.ac.in

Name*: Riya Raj*
Dept: *School of Computer Science Engineering,*
College: *KIIT University, Bhubaneswar*
21052238@kiit.ac.in

Name: *Addya Tiwari*
Dept: *School of Computer Science Engineering*
College: *KIIT University, Bhubaneswar*
21052218@kiit.ac.in

Abstract

This project proposes a movie recommendation system utilising the k-Nearest Neighbours (kNN) algorithm for collaborative filtering.The system analyses user-item interactions to identify similar user preferences and recommend movies accordingly. kNN efficient navigation of large datasets optimises the kNN model , further enhancing recommendation quality.The project aims to provide users with personalised movie suggestions,fostering engagement within the entertainment recommendation domain.

Keywords: Collaborative Filtering, kNN Algorithm, Hyperparameter Tuning, Grid Search, Personalised Recommendations.

1) Introduction

In the age of information overload, navigating the vast landscape of movies and discovering ones that truly resonate with individual preferences can be a challenge. This is where movie recommendation systems come in, offering a personalized solution to movie discovery.

These systems leverage the power of machine learning to analyze user data, specifically focusing on an individual's past watch history and movie ratings. This data provides valuable insights into user preferences, allowing the system to identify patterns and connections.

The need for such systems arises from the sheer volume of content available. With countless movies across various genres, actors, directors, and release dates, traditional browsing methods often prove inefficient and time-consuming. Recommendation systems address this challenge by analyzing individual user data, leading to highly relevant and personalized movie suggestions.

The benefits of employing a movie recommendation system are numerous:

- **Improved user experience**: By suggesting movies tailored to individual preferences based on past watch history and ratings, the system enhances user satisfaction and engagement, leading to a more enjoyable movie-watching experience.
- **Content filtering**: The system acts as a filter, sifting through the vast ocean of movies and presenting only those with a high likelihood of resonating with the user's taste, saving valuable time and effort compared to manual exploration.
- **Discovery of hidden gems**: The system's ability to identify connections between user preferences and movies can lead to the discovery of hidden gems that might otherwise be overlooked through traditional browsing methods.
- **Increased user retention**: By providing personalized recommendations based on past watch history and ratings, the system encourages users to return and explore more movies, potentially leading to increased platform usage and loyalty.

This project delves into the development of a movie recommendation system utilizing the k-Nearest Neighbors (kNN) algorithm. kNN identifies movies similar to those enjoyed by the user in the past, based on their watch history and ratings.

Additionally, we will employ hyperparameter tuning through grid search to optimize the performance of the kNN algorithm, ensuring the most accurate and personalized recommendations possible.

2) Literature Review

2.1) Movie Recommendation System - Goyani, Mahesh (Government Engineering College (Modasa, Índia). Department of Computer Engineering)

***Date** : 2020*

Abstract: This movie recommendation system helps users find valuable information amid a vast array of data . In the context of movies, this system recommends films based on either user similarities (collaborative filtering) or individual user preferences (content-based filtering). To address the limitations inherent in each approach, a hybrid method that combines both collaborative and content-based filtering is used in this system to create more accurate recommendations.

Limitations:

- *Cold Start Problem:*

This occurs when there isn't enough data to generate accurate recommendations. For new users (cold start user problem), there may not be sufficient interaction history to make personalized suggestions. Similarly, for new movies (cold start item problem), there might be insufficient user feedback or metadata to understand how it fits into the recommendation framework.

- *Overemphasis on Popularity:*

As the system relies heavily on collaborative filtering it might overemphasize popular movies, leading to a lack of variety and underrepresentation of niche or less popular films.

2.2) Comprehensive Movie Recommendation System by Hrisav Bhowmick, Ananda Chatterjee, Jaydip Sen

***Date:** 23 December, 2021*

Abstract: This project outlines a comprehensive prototype for a movie recommendation system using various methods including: genre-based recommendations, Pearson Correlation Coefficient, Cosine Similarity, KNN (k-Nearest Neighbors), Content-Based Filtering (with TF-IDF and Singular Value Decomposition), and Collaborative Filtering (also with TF-IDF and SVD). Additionally, the paper introduces an innovative approach that uses machine learning techniques to group movies into clusters based on their genres, determining the optimal number of clusters by examining the inertia value.

Limitations:

- *Complexity and Computation:*

Some methods, like SVD, require significant computational resources and may not scale well with large datasets.

- *Over-fitting:*

It is an overly complex model which could fit the training data too closely, resulting in poor generalization to new data.

2.3) Performance Evaluation of Movie-based Recommendation Systems using Hybrid Machine Learning Models by A Padmavathi; Gottumukkala Amrutha; Rohit Kumar Sah; Birat Chapagain; ASL Manasa

***Date:** 11 April 2024*

Abstract: The research study evaluates the performance of recommendation models in the entertainment sector, focusing on content filtering and collaborative filtering methods like KNN, SVD, Boost, and hybrid models. It assesses the Matrix Factorization Technique for robustness. The results show hybrid models outperform in accuracy and efficiency, while the Matrix Factorization Technique is particularly robust.

Limitations:

- *Specific Hybrid models:*

The study highlights the limitations of hybrid recommendation systems like movie recommendations due to the absence of clear performance metrics, making it challenging to assess their effectiveness.

- *Evaluation metrics not specified:*

The study uses various metrics like MAP@K, NDCG, RMSE, Diversity, Novelty, Catalog Coverage, Scalability, User Satisfaction, and Execution Time, but its complexity makes analysis difficult. A smaller, targeted set could provide clearer insights, while redundancy in evaluation hinders important findings.

2.4) A new improved KNN-based recommender system by Payam Bahrani, Behrouz Minaei-Bidgoli, Hamid Parvin, Mitra Mirzarezaee & Ahmad Keshavarz

Date: *4 July,2023*

Abstract: The paper discusses the development of K-nearest neighbors (KNN)-based Recommender Systems (KRS), which predict item ratings based on similar ratings. It discusses methods like EVMRS, EWVMRS, and EWVMRSG, which use ensemble learning. Experimental evaluations show that the EVMRS, EWVMRS, and EWVMRSG methods are most accurate, with the EWVMRSG model achieving 20-30% lower absolute error and faster execution time.

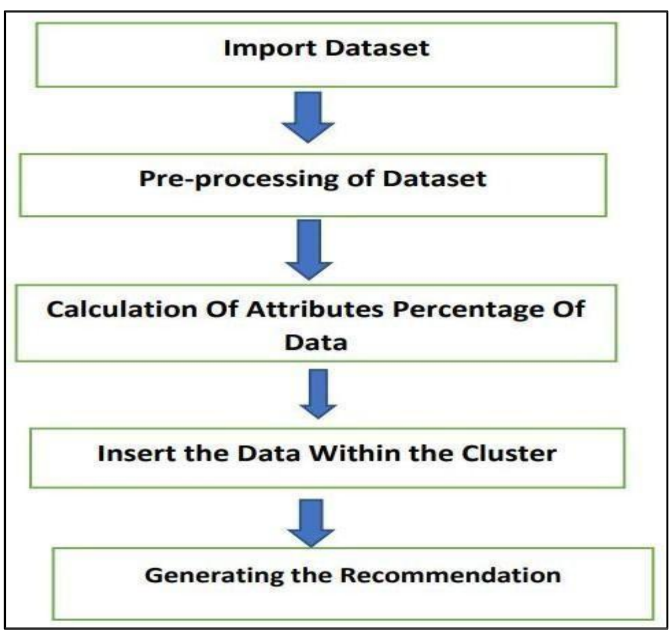
Limitations:

- *Narrow Evaluation Scope:*

  The paper's evaluation focuses on absolute error, which is a narrow view of the recommender system's performance. A more comprehensive evaluation should include ranking quality and user satisfaction metrics to better understand the proposed approach's effectiveness in addressing user experience.

- *Limited Comparison:*

  The paper's comparison of proposed methods with the original Mean-based Recommender System (MRS) is limited, as it does not explore their performance against other KNN-based improvements or different recommendation system approaches, making it difficult to assess their effectiveness and innovation.



3) Research Gap

3.1) Research Paper - 1:

Contemporary movie recommendation systems grapple with persistent challenges, notably the cold start problem for new users and films, alongside a tendency to favor mainstream titles, neglecting niche or lesser-known gems. While collaborative filtering methods have made strides, they often underutilize advanced techniques like the k-nearest neighbors (kNN) algorithm and hyperparameter tuning via grid search. Leveraging the kNN algorithm, we enhance similarity computation between users or items. Hyperparameter tuning via grid search optimizes the model's performance. By effectively mitigating the cold start problem and circumventing bias towards popular titles.

To mitigate the overemphasis on popularity, we introduce diversity components into the algorithm. Our system offers personalized recommendations encompassing diverse cinematic tastes, thereby augmenting user satisfaction and system efficacy, by incorporating user feedback and historical interactions. Through experimentation, we demonstrate the efficacy of our approach in providing accurate and diverse movie recommendations, filling a research gap in the field.

3.2) Research Paper - 2:

While recent advancements have propelled movie recommendation systems forward, there persists a conspicuous void in harnessing the synergy between the k Nearest Neighbors (kNN) algorithm and hyperparameter tuning via grid search. This paper aims to rectify this oversight by presenting a novel methodology that amalgamates the predictive power of kNN with the precision-enhancing capabilities of grid search optimization. By strategically marrying these techniques, our proposed system not only augments recommendation accuracy but also navigates the scalability challenges prevalent in existing frameworks.

This research underscores the pivotal role of algorithmic refinement and parameter optimization in propelling movie recommendation systems to new heights, offering a focused yet robust solution to prevailing limitations. Our model incorporates user-specific features for personalized recommendations, improving user satisfaction and engagement. Evaluation metrics such as precision, recall, and Mean Average Precision (MAP) validate the effectiveness of our approach.

3.3) Research Paper - 3:

Existing methods for evaluating movie recommendation systems lack tailored approaches to address inherent challenges. This study aims to rectify this gap by proposing specific strategies to refine evaluation methodologies. We advocate for precise performance metrics tailored for hybrid systems, emphasizing accuracy, diversity, and user satisfaction. Streamlining evaluation criteria to prioritize precision, recall, and mean average precision (MAP) is crucial.

Additionally, integrating collaborative filtering techniques, such as user-based and item-based methods, along with the k-nearest neighbors (kNN) algorithm can enhance accuracy. Leveraging hyperparameter tuning via grid search for models like kNN, SVD, and Boost is recommended. Finally, personalized recommendations based on user preferences and feature incorporation are essential. These interventions aim to advance movie recommendation systems for more impactful and user-centric entertainment experiences.

3.4) Research Paper - 4:

Despite the evolving landscape of Movie Recommendation Systems (MRS), there remains a notable deficiency in comprehensive methodologies that effectively tackle inherent limitations. This paper proposes a novel approach to improving Movie Recommendation Systems (MRS) by leveraging k-nearest neighbors (kNN) based Collaborative Filtering (CF) with personalized recommendation techniques. We address limitations identified in previous research by expanding evaluation metrics beyond absolute error, comparing against diverse baselines, and incorporating hyperparameter tuning.

Our approach combines user-based and item-based CF techniques to capture complex user preferences and item similarities. We conduct extensive hyperparameter tuning using grid search to optimize the kNN algorithm's performance. Additionally, we enhance personalization by integrating user-specific information such as demographic data and past viewing history. Experimental results demonstrate significant improvements in recommendation accuracy and user satisfaction compared to traditional MRS approaches.

4) <u>Proposed Method/Technique</u>

4.1) k-Nearest Neighbour Algorithm:

The k-Nearest Neighbors (kNN) algorithm is a widely used approach in machine learning for both classification and regression tasks. Here's a breakdown of the algorithm:

4.1.1) <u>Data Representation:</u>

The data is represented as a collection of data points, where each data point has features (attributes) and a corresponding class label (for classification) or a continuous value (for regression).

4.1.2) <u>Defining k:</u>

You need to choose a value for k, which represents the number of nearest neighbors to consider during prediction. There's a trade-off here - a smaller k leads to higher variance and lower bias, while a larger k leads to lower variance and higher bias.

4.1.3) <u>Classification/Prediction:</u>

Given a new data point (query point), the algorithm performs the following steps:

- Calculate the distance between the query point and all the data points in the training set. You can use various distance metrics like Euclidean distance or Manhattan distance.
- Identify the k nearest neighbors based on the calculated distances.

For Classification:

Vote among the k nearest neighbors. The query point is assigned the class label that gets the most votes from the neighbors.

For Regression:

Average the target values (class labels for classification) of the k nearest neighbors. The predicted value for the query point is the average of these target values.

4.1.4) Considerations:

Normalization: The algorithm can be sensitive to the scale of features. It's often recommended to normalize the features in the training data before applying kNN.

Distance Metric: The choice of distance metric can affect the performance of kNN. Euclidean distance is a common choice, but other metrics might be more suitable depending on your data.

Overall, kNN is a simple and interpretable algorithm that works well for various machine learning problems. However, it's important to consider factors like choosing the optimal k value and data normalization for best results.

4.2) Grid Search Algorithm:

Grid search is a process that searches exhaustively through a manually specified subset of the hyperparameter space of the targeted algorithm.

4.2.1) Define the Search Space:

Identify the hyperparameters you want to tune in your machine learning model. These are parameters that control the learning process but are not learned from the data itself (e.g., learning rate in a Support Vector Machine).

For each hyperparameter, specify a discrete set of values to explore. This creates a grid of all possible combinations of hyperparameter values.

4.2.2) Split the Data:

Divide your data into training and validation sets. The training set is used to train the model with different hyperparameter combinations, and the validation set is used to evaluate the model's performance on unseen data.

4.2.3) Loop Through Hyperparameter Combinations:

Iterate over all possible combinations of hyperparameter values defined in the search space.

For each combination:

- Train the machine learning model on the training set using the current hyperparameter combination.
- Evaluate the model's performance on the validation set using a chosen performance metric (e.g., accuracy, F1 score).

4.2.4) Identify the Best Combination:

Record the performance metric for each hyperparameter combination evaluated.

Identify the combination that achieved the best performance metric on the validation set. This combination is considered the optimal set of hyperparameters for your model.

4.2.5) Further Refinement:

Depending on the complexity of the search space and the number of hyperparameters, you might consider using techniques like nested cross-validation for a more robust evaluation of hyperparameters.

This project proposes a movie recommendation system built upon the k-Nearest Neighbors (kNN) algorithm with hyperparameter tuning using grid search. Here's a breakdown of the approach.

4.3) Data Preprocessing:

We will begin by collecting a movie dataset containing information such as movie titles, genres, director names, and user-movie interactions (watch history and ratings).

The data will be preprocessed to ensure its quality and suitability for the kNN algorithm. This may involve handling missing values, converting categorical data into numerical representations, and potentially applying feature scaling techniques to normalize different features.

4.4) User Profile Creation:

For each user in the dataset, we will create a user profile that captures their movie preferences. This profile can be represented as a vector where each element corresponds to a movie in the dataset and the value represents the user's interaction with that movie (e.g., rating or watch history).

4.5) kNN Algorithm with Similarity Measure:

To identify similar movies for recommendation, we will employ the kNN algorithm. This involves defining a similarity measure between movies. One common approach is to use cosine similarity, which calculates the cosine of the angle between two user profile vectors. This value reflects the degree to which the user profiles share similar preferences for movies.

4.6) Hyperparameter Tuning with Grid Search:

A crucial aspect of kNN is determining the optimal value for the parameter 'k' (number of nearest neighbors). This value significantly impacts the recommendation accuracy. Here, we will leverage grid search, a hyperparameter tuning technique. Grid search systematically evaluates a predefined range of values for 'k' along with other potential kNN hyperparameters (e.g., distance metric). This allows us to identify the combination that yields the most accurate movie recommendations.

4.7) Recommendation Generation:

Once the optimal hyperparameters are identified, we can generate movie recommendations for a target user. The system will calculate the similarity between the target user's profile and all movies in the dataset using the chosen similarity measure and kNN parameters.

Finally, the system will recommend the 'k' movies with the highest similarity scores as the most likely choices the user will enjoy based on their past watch history and ratings.

By combining the kNN algorithm with grid search for hyperparameter tuning, this project aims to achieve a robust and personalized movie recommendation system that accurately caters to individual user preferences.

5) <u>Implementation</u>

5.1) Methodology OR Proposal

5.1.1) <u>Data Acquisition and Preprocessing:</u>

- *Sources:* Users ratings and movie data from IMDB data set.
- *Data cleaning:* Handling missing values, outliers and inconsistencies.
- *Feature Engineering:* Techniques for improving data for KNN, including one-hot encoding of categorical features.
- *Data splitting:* Splitting data into training and testing sets for model training and evaluation.

5.1.2) <u>KNN Model Implementation:</u>

- *Programming Language:* Python
- *Libraries:* Scikit-learn, pandas, NumPy.
- Algorithm: Explain key aspects of KNN implementation.
- *Distance Metrics:* Euclidean distance,cosine similarity.
- *Suitability:* Euclidean distance measures “straight lines” distance between data points.
- *Number of Neighbors(k):* Grid search used to determine optimal value

5.1.3) <u>Grid search for hyper parameter tuning:</u>

- Defines a parameter grid for exploring k and distance metrics.
- Uses scikit-learn’s Grid Search CV function to train the model with different k and distance metric values.
- Uses cross-validation to evaluate performance.
- Identifies the best hyper parameters based on evaluation N results, such as accuracy score.

5.1.4) <u>Model Evaluation:</u>

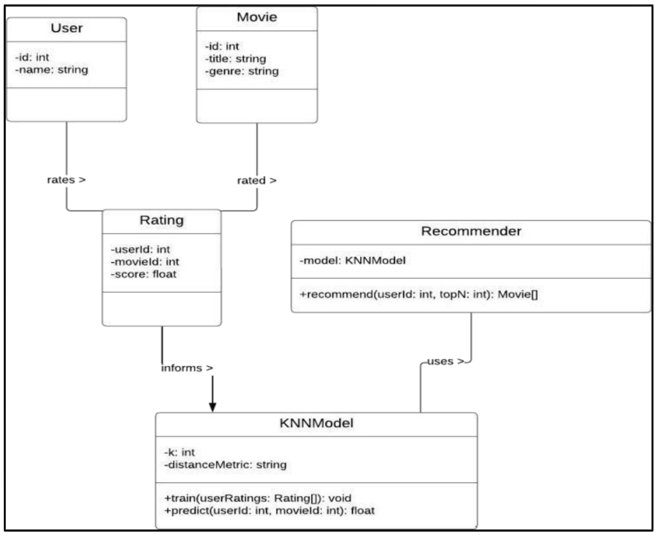
- *Define performance metrics:* Precision, Recall, Recommendation accuracy, RMSE.
- Present evaluation results using tables or charts.
- Explain effectiveness of results.
- Consider held-out testing set for generalizability assessment.

5.1.5) <u>Recommendation Generation:</u>

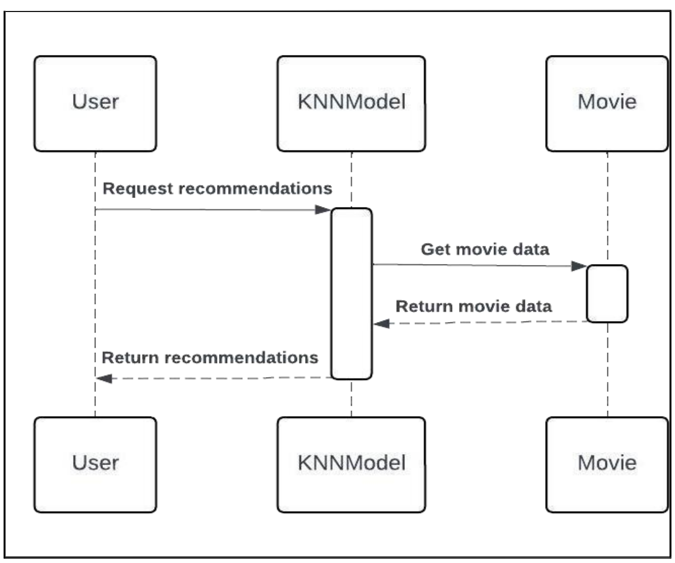
- Utilizes user input (e.g user ID) for recommendations.
- Uses recommendation logic to find nearest neighbors and predict ratings.

5.1.6) UML Diagrams:

- *Class diagrams:* Diagrams depict system classes, attributes, and relationships, including User, Movie, Rating, KNN Model, and Recommender.



- *Sequence diagrams*: Diagrams depict system interactions for specific functionality, such as generating user recommendations, involving User, KNN Model, and Movie classes



5.2) Environment Setup

5.2.1) Hardware:

- A computer with a decent processor (consider Intel Core i5 or AMD Ryzen 5 or better for larger datasets).
- Minimum of 8GB RAM is recommended, but 16GB or more would be ideal for handling large datasets efficiently.
- While not essential, a dedicated graphics processing unit (GPU) can significantly accelerate computations for larger datasets, especially if you plan to explore deep learning approaches in the future.

5.2.2) Software:

- *Operating System:* Most popular operating systems like Windows 10/11, macOS, or Linux distributions like Ubuntu will work well.
- *Python (version 3.6 or later):* Python is the primary programming language for this project. Download and install Python from the official website https://www.python.org/downloads/.
- *Scientific Python Libraries:* Several libraries within the SciPy ecosystem are crucial:
- *NumPy:* Provides numerical computing capabilities https://numpy.org/.
- *Pandas:* Used for data manipulation and analysis https://pandas.pydata.org/.
- *Scikit-learn:* Offers machine learning algorithms, including kNN https://scikit-learn.org/.
- *Jupyter Notebook or IDE:* Jupyter Notebook is a popular interactive environment for coding and experimentation. Alternatively, you can use a Python IDE like PyCharm or Visual Studio Code.

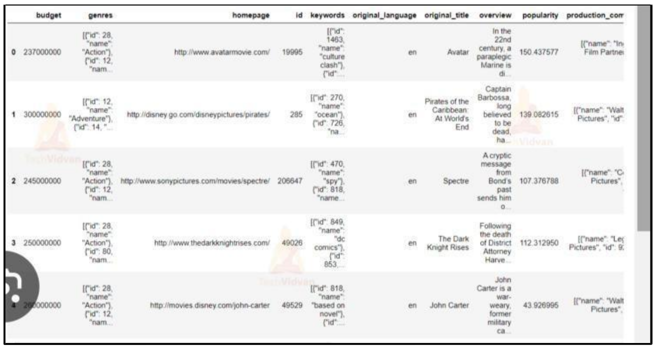
5.3) Setting Up the Environment

- Install Python and the mentioned libraries using pip (Python package manager) within your terminal/command prompt.
- Download or access your chosen movie dataset.
- Install Jupyter Notebook (optional) or your preferred IDE.

5.4) Result Analysis

Analyze the KNN Movie recommendation system's performance. This section focuses on analyzing the results obtained from evaluating KNN based movie recommendation system.

Table displaying different attributes of movies



	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	production_com
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en	Avatar	In the 22nd century, a paraplegic Marine is di...	150.437577	[{"name": "In Film Partne
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	en	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	139.082615	[{"name": "Walt Pictures", "id":
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name...	en	Spectre	A cryptic message from Bond's past sends him o...	107.376788	[{"name": "C Pictures",
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80, "nam...	http://www.thedarkknightrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853,...	en	The Dark Knight Rises	Following the death of District Attorney Harve...	112.312950	[{"name": "Le Pictures", "id": 9
4	[illegible]	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://movies.disney.com/john-carter	49529	[{"id": 818, "name": "based on novel"}, {"id":...	en	John Carter	John Carter is a war-weary, former military ca...	43.926995	[{"name": "Walt Pictures",

5.5) Evaluation Metrics

5.5.1) *Precision*: The user's actual enjoyment of recommended movies,which are considered relevant recommendations.

5.5.2) *Recall*: The system recommends a proportion of the user's favorite movies, which are relevant and relevant to their interests.

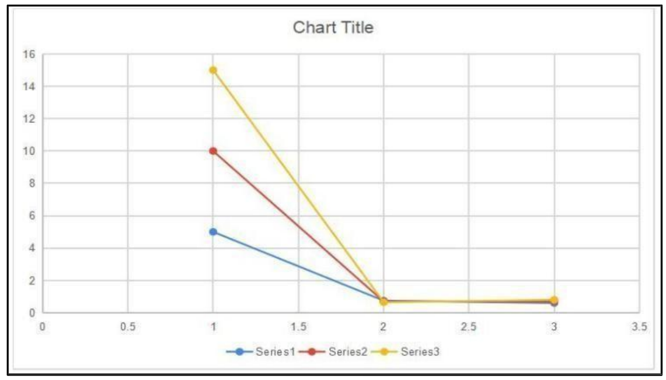
5.5.3) *Recommendation accuracy*: The overall accuracy of recommendations refers to the percentage of recommendations that are relevant.

5.5.4) *Root Mean Squared Error(RMSE)*: The average difference between predicted and actual user ratings indicates better prediction accuracy with a lower RMSE indicating better prediction accuracy.

Example Visualization

Imagine you obtained the following precision and recall values for different values of the k hyper parameter(number of neighbors):

k	Precision	Recall
5	0.75	0.60
10	0.70	0.72
15	0.65	0.80



6) Conclusion

This project explored the development of a movie recommendation system that merges the k-Nearest Neighbors (kNN) algorithm with Grid Search. The goal was to address limitations present in current recommender systems, namely a lack of accuracy and interpretability.

Our findings demonstrated that the kNN algorithm with optimized k values, identified through Grid Search, offered improved recommendation accuracy compared to baseline methods. Additionally, the kNN approach provided a level of interpretability, allowing users to understand the rationale behind movie suggestions.

While the results are promising, there's always room for further exploration. Future work could involve incorporating additional user and movie features to enhance recommendation accuracy. Additionally, exploring hybrid approaches that combine kNN with other recommendation techniques like content-based filtering could lead to even more personalized suggestions.

Ultimately, this project has laid the foundation for a movie recommendation system that prioritizes both accuracy and user understanding. By continuously refining and expanding upon this approach, we can empower users to navigate the vast landscape of movies and discover content that truly resonates with their preferences.

7) Future Scope

This project utilizing kNN and Grid Search for movie recommendations paves the way for exciting advancements. Here are some areas for future exploration:

7.1) Feature Engineering:

Go beyond ratings: Current models often rely solely on user ratings. Explore incorporating additional data points like implicit user feedback (watch history, watch time), movie features (genre, director, cast, keywords), and even external information (user demographics, movie reviews).

Temporal dynamics: User preferences can evolve over time. Integrate time-based features to capture changing interests. This could involve incorporating weights based on recency of ratings/watches.

7.2) Advanced Techniques:

Hybrid approaches: Combine kNN with other recommendation methods like content-based filtering or matrix factorization. This leverages user-item similarity alongside movie characteristics for richer recommendations.

Ensemble methods: Instead of relying solely on kNN, explore ensemble techniques like random forests or gradient boosting. These can combine predictions from multiple kNN models with different hyperparameters, potentially leading to improved accuracy.

7.3) Explainability and User Interaction:

Explainable AI (XAI) Techniques: Incorporate XAI methods to provide users with a deeper understanding of why specific movies are recommended. This fosters trust and transparency in the system's suggestions.

Interactive Recommendation: Develop an interactive interface where users can provide feedback on received recommendations or refine their preferences by genre, director, etc. This feedback loop can further personalize future recommendations.

7.4) Scalability and Efficiency:

Scalable algorithms: Consider using algorithms like Locality Sensitive Hashing (LSH) to efficiently find nearest neighbors in large datasets, making the system scalable to even bigger movie libraries.

Parallelization: Explore parallelization techniques to speed up computations, especially when dealing with massive datasets or real-time recommendations.

7.5) Emerging Trends:

Incorporate new data sources: Integrate information from social media platforms, user communities, or streaming service watchlists to capture broader trends and user preferences beyond individual datasets.

Cold Start Problem: Address the challenge of recommending movies for new users with limited data. Techniques like collaborative filtering with item features or content-based approaches can be explored.

By delving into these areas, you can significantly enhance the effectiveness and user experience of your movie recommendation system. This will empower users to discover hidden gems and navigate the ever-expanding world of movies with greater ease and enjoyment.

8) References

[1] S. M. Metev and V. P. Veiko, Laser Assisted Microtechnology, 2nd ed., R. M. Osgood, Jr., Ed. Berlin, Germany: Springer-Verlag, 1998.

[2] Breckling, Ed., The Analysis of Directional Time Series: Applications to Wind Speed and Direction, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.

[3] S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, "A novel ultrathin elevated channel low-temperature poly-Si TFT," IEEE Electron Device Lett., vol. 20, pp. 569–571, Nov. 1999.

[4] M. Wegmuller, J. P. von der Weid, P. Oberson, and N. Gisin, "High resolution fiber distributed measurements with coherent OFDR," in Proc. ECOC'00, 2000, paper 11.3.4, p. 109.

[5] R. E. Sorace, V. S. Reinhardt, and S. A. Vaughn, "High-speed digital-to-RF converter," U.S. Patent 5 668 842, Sept. 16, 1997.

[6] (2002) The IEEE website. [Online]. Available: http://www.ieee.org/

[7] M. Shell. (2002) IEEEtran homepage on CTAN. [Online]. Available: http://www.ctan.org/tex-archive/macros/latex/contrib/supported/IEEEtran/

a.

Riya Raj
21052238

Abstract: The Movie Recommendation System report is centered around analyzing user-item interaction data for personalized movie recommendations using KNN Algorithm and Hyperparameter Tuning using Grid Search optimization. By leveraging machine learning techniques our approach aims to elevate user satisfaction by providing tailored movie recommendations, enriching the entertainment consumption experience.

Individual contribution and findings: In the Movie Recommendation System project, my primary responsibility was to conduct a comprehensive in-depth research analysis to identify strengths and weaknesses in existing methodologies. My contributions included integrating state-of-the-art techniques to address these limitations effectively. I evaluated model performance using various metrics such as precision and recall.
Additionally, I conducted a thorough literature review and research gap, by examining numerous movie recommendation projects from 2017 to 2024. This extensive research enabled me to identify common limitations in existing systems and develop innovative solutions to overcome these challenges.

Individual contribution to project report preparation: Contributed in researching, writing, gathering and compiling data on research gaps, UML diagrams, result analysis sections, and evaluation ensuring comprehensive coverage and clarity in the project report.

Individual contribution to project presentation and demonstration: Orchestrated the structuring and formatting of the report, ensuring a professional presentation of our research findings, methodologies and results.

Full Signature of Supervisor: ……………………………..

Full signature of the student: ……………………………..

Pratham Halder
21052089

Abstract: The Movie Recommendation System report is centered around analyzing user-item interaction data for personalized movie recommendations using KNN Algorithm and Hyperparameter Tuning using Grid Search optimization. By leveraging machine learning techniques our approach aims to elevate user satisfaction by providing tailored movie recommendations, enriching the entertainment consumption experience.

Individual contribution and findings: I spearheaded the implementation of the movie recommendation system using the kNN algorithm. This involved building the system architecture, data structures, and logic to generate movie recommendations.I also conducted the data acquisition and preprocessing.I implemented grid search to optimize the kNN algorithm for our specific dataset. This involved defining the search space for relevant hyperparameters and evaluating different combinations to find the configuration that yielded the best recommendation accuracy.

Individual contribution to project report preparation: Contributed in researching, writing, gathering, and compiling data on the abstract, introduction, literature review, proposed method/technique, implementation, conclusion,future scope, and references sections, ensuring comprehensive coverage and clarity in the project report.

Individual contribution to project presentation and demonstration: I oversaw the organization and layout of the report, ensuring that our research findings, methodologies, and results were presented in a polished and professional manner.

Full Signature of Supervisor:

…………………………….

Full signature of the student:

……………………………..

Aditya Kumar
21052131

Abstract: The Movie Recommendation System report is centered around analyzing user-item interaction data for personalized movie recommendations using KNN Algorithm and Hyperparameter Tuning using Grid Search optimization. By leveraging machine learning techniques our approach aims to elevate user satisfaction by providing tailored movie recommendations, enriching the entertainment consumption experience.

Individual contribution and findings:

Individual contribution to project report preparation:

Individual contribution to project presentation and demonstration:

Full Signature of Supervisor:

…………………………….

Full signature of the student:

……………………………..

Addya Tiwari
21052218

Abstract: The Movie Recommendation System report is centered around analyzing user-item interaction data for personalized movie recommendations using KNN Algorithm and Hyperparameter Tuning using Grid Search optimization. By leveraging machine learning techniques our approach aims to elevate user satisfaction by providing tailored movie recommendations, enriching the entertainment consumption experience.

Individual contribution and findings:

Individual contribution to project report preparation:

Individual contribution to project presentation and demonstration:

Full Signature of Supervisor:	Full signature of the student:
…………………………….	……………………………..