KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY (KIIT)

Deemed to be University U/S 3 of UGC Act, 1956

M-CLUE: A Case study on Movie based Recommender

Group Members:

I. Priyank Kumar Nigam-22053177
II. Sanskriti Srijan-22053188

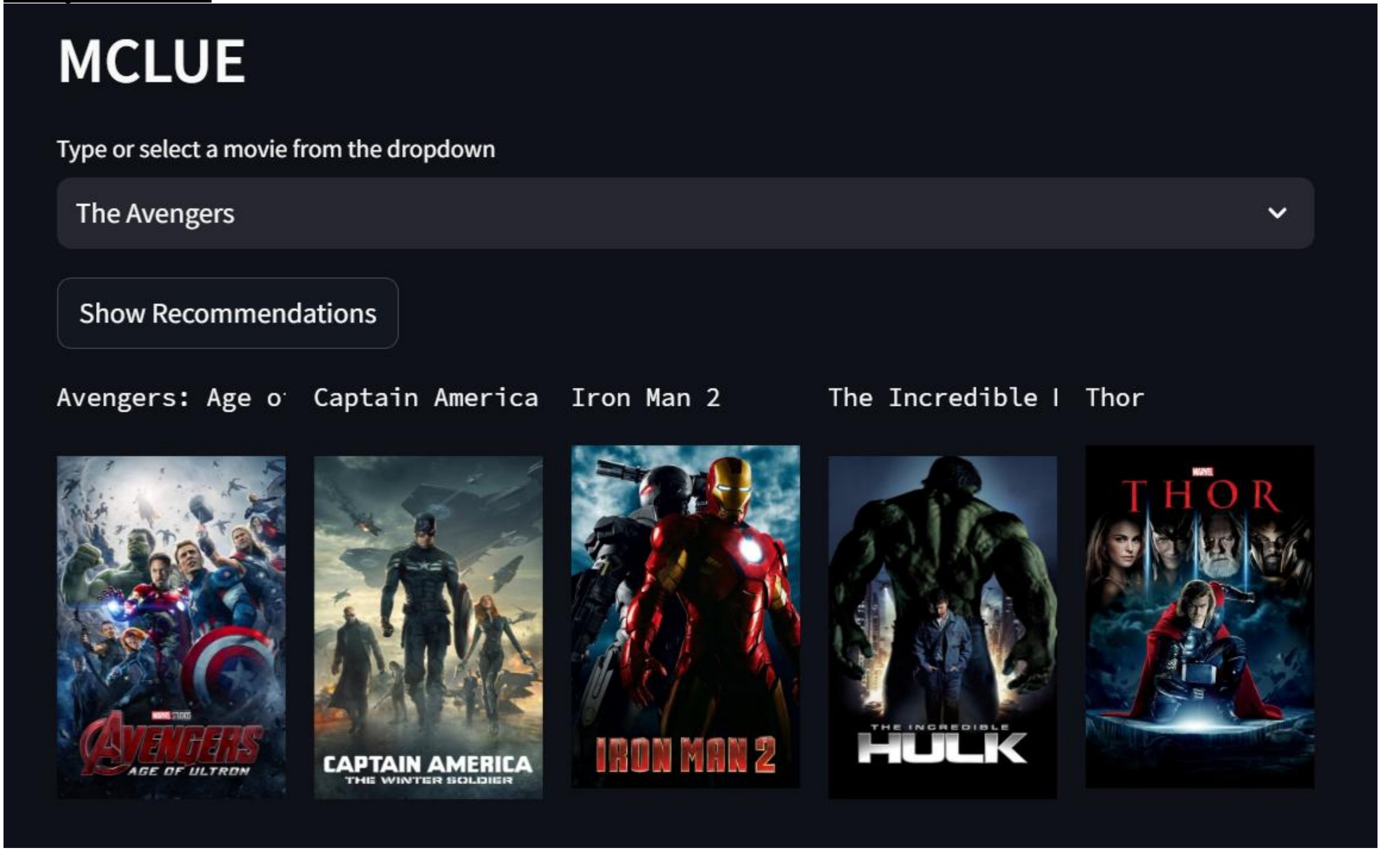
This Case Study Was Done Under The Guidance of :
Sir. Vijay Kumar Meena

MCLUE: A Case study on Movie based Recommender

Abstract:

MCLUE was created for moviegoers who want to find movies of their taste. A moviegoer may have to spend a lot of time on the web reading and watching reviews to reach a conclusion as to whether he/she should watch a movie or not. Usually the information available in various media about the movies are not targeted towards user's taste , or may display the information which may not always stay on the topic and slowly drift away from the viewer's interests. MCLUE involves taking the user's feedback and finding similarity of the user to other users by using collaborative,content filtering and nearest neighbour algorithm. These algorithms, based on the user's feedback, place the user into a particular multidimensional plane using vectors and recommend the user the movies that he/she may like based on the feedback given by users who nearest to him/her using cosine distance method. MCLUE application aims at reducing the time spent in searching for a movie of his/her taste and thereby increasing chances of getting value for their movies.

Keywords: Content-based Recommender system , Expert Users ,Vectorization ,Nearest Neighbour



MCLUE

Type or select a movie from the dropdown

The Avengers

MCLUE: Movie Recommender System

INTRODUCTION :

MCLUE is a web application in which a user gives ratings to movies and builds a custom taste profile. The system then uses collaborative filtering and clustering algorithms to find similar movies based on the custom taste profile and recommends movies to the user. This learns continuously user preferences, and keeps getting better with usage over time. Movie Recommender System : MCLUE targets all people who are interested in movies, but find it difficult to find movies which are to their taste. The application is designed in such a way that the user initially rates a few movies which he has seen and then the application builds a custom taste profile and recommends movies which are closer to his taste profile. The user can then view the list of recommended movies and watch them. The recommendations keep getting better with each rating given by the user.

In the existing system, people had to ask their friends or visit sites like IMDB or YouTube, and he has to read a lot of reviews and then decide on a movie to watch and even then there is no guarantee that the user may find a movie that he may like. People usually end up with watching movies that they are not satisfied with. We overcame these drawbacks in the proposed system.

RELATED WORK:

Recommender systems are a type of information filtering system that gives advice on products, information, or services that a user may be interested in. They assist users with the decision making process when choosing items with multiple alternatives. Recommender systems are popular due to their e-commerce application purposes. Within the e-commerce world, recommendations provide aid to customers and help them find what they may be looking for, thus increasing business. In addition, it can be used as a tool to predict user‟s behavior, but should not be used to select recommendations on their behalf.

There are two basic entities that appear in any recommender system are the user and the item of interest. The user can be a customer in an e-commerce platform or an avid book reader looking for a recommendation for the next book they should read. The users provide their ratings on items and are used to aid other users with their recommendations.

The item is the second piece of a recommender system. Users give items ratings and the algorithm outputs recommended items based on new user queries.

We can classify the recommender systems in three broad categories:

i. Collaborative Filtering Recommender system

ii. Content-Based Recommender system

iii. Hybrid Recommender system

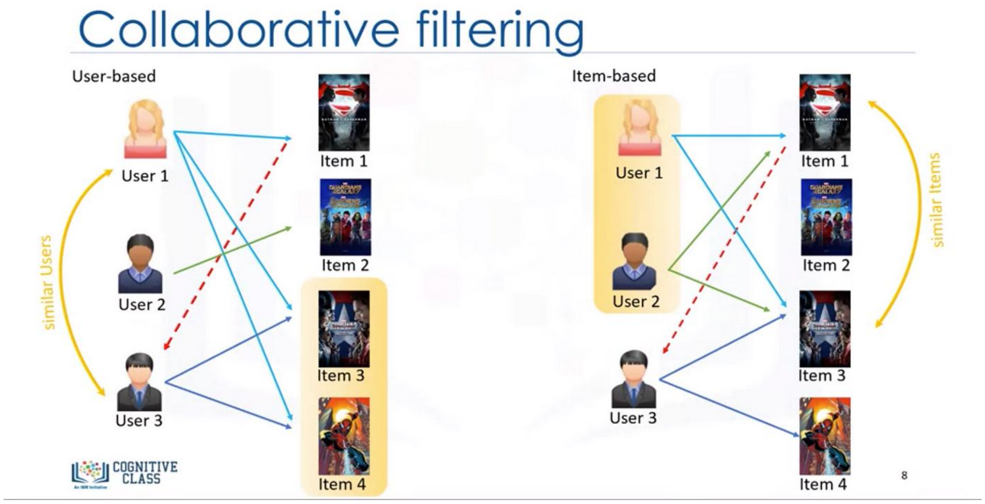
I. Collaborative Filtering :

Collaborative filtering, the traditional Recommender system is based on rating structure usually represented as User-Movie Rating matrix. Each cell value represents the rating of a movie by the user. It predicts the ratings based on similarity measures like Pearson correlation coefficient, cosine similarity, Euclidean distance measure, etc. Collaborative filtering can be classified into two types: Memory based CF and Model based CF. Memory based CF predicts the ratings using the entire user-item database of users who are similar to the active user whereas Model based CF predicts the ratings by using the constructed model. Collaborative filtering often suffers from several issues which include :

Sparsity: Most often users do not rate the movies which results in sparsity of data.

Cold start: To recommend a new item or for new user who has not yet rated any movies, it is very difficult as there exists no user information.

Scalability: To handle millions of users and movies over Internet, CF computations to find similar users grow exponentially and becomes expensive.

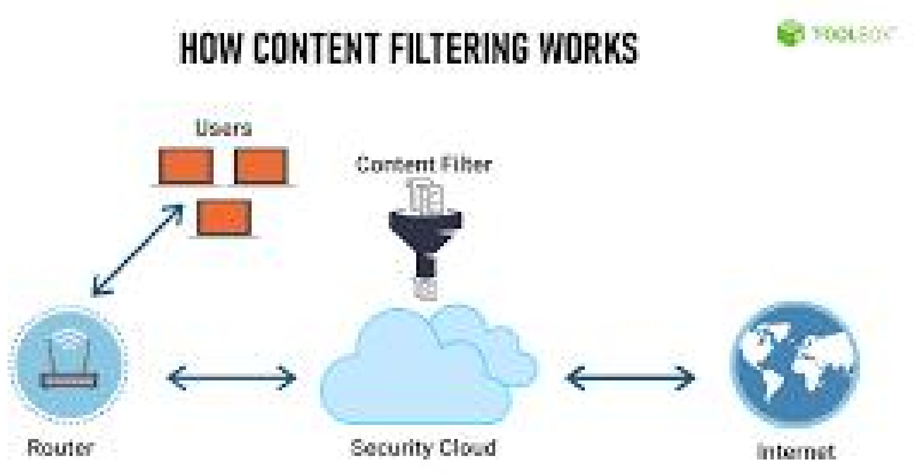


How Collaborative Filtering Works

II. Content Based Recommender System:

(The System Our Application is based on)

Content based recommendations are based on the user individual preferences and tastes.
It recommends the movies preferred by the user in the past.
Content-based Recommender system often suffer from the following issues:
Limited content analysis It is difficult to recommend if there is a limited content about the user profile Overspecialization restricts users to items similar to the ones defined in their respective profiles and thus new items and other options are not discovered.

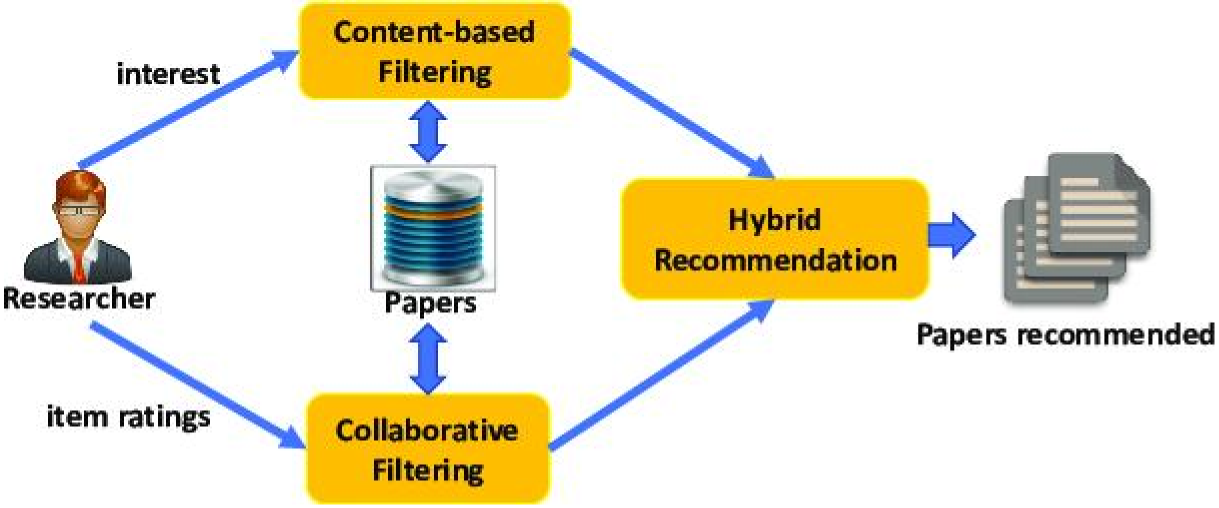


How Content-Based Filtering Works

III. Hybrid Recommender system :

Hybridization of demographic and collaborative filtering approaches had been employed to solve cold start problem. Hybrid model based approach has been applied on Movie

data set to enhance recommendation quality. Additionally Collaborative filtering and Demographic based approach had been used to modify similarity calculation.
In contrast, this paper proposes a novel approach to enhance the Recommendation quality by utilizing user demographic data provided by the users explicitly and Collaborative filtering approach based on user ratings on movies.



How Hybrid Filtering Works

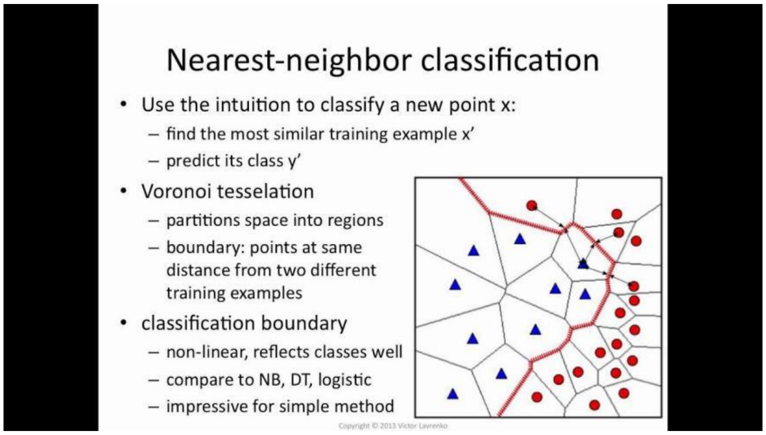
PROPOSED METHODOLOGY :

In this proposed system user overcomes the disadvantages of the existing system by using MCLUE web application. If the user wants to watch a movie and doesn‟t know which movie to watch he registers himself with the application and rates the movies he has already watched and builds a custom taste profile, and depending on the ratings, he is recommended movies by the application.

The methods that we have used in this Proposed system are:
i. Nearest Neighbours
ii. Vectorization(cosine angles)
iii. Recommendation generation

i. **Nearest Neighbours:**

A nearest neighbor algorithm analyzes all the data on every request. Classification, categorization, and everything in between will happen at the time of search (ie: just-in-time results). The search needs to be able to handle an unknown amount of data and an unknown amount of users at any given second. That’s fancy talk for saying it needs to be really really fast. If you’ve ever attempted text comparisons on a large dataset you will know that it’s anything but performant. To overcome this, the text is converted to a collection of numbers called vectors. Computers are very good at comparing numbers.



Nearest-neighbor classification

- Use the intuition to classify a new point x:
  - find the most similar training example x’
  - predict its class y’
- Voronoi tesselation
  - partitions space into regions
  - boundary: points at same distance from two different training examples
- classification boundary
  - non-linear, reflects classes well
  - compare to NB, DT, logistic
  - impressive for simple method

Nearest-Neighbouring Algorithm

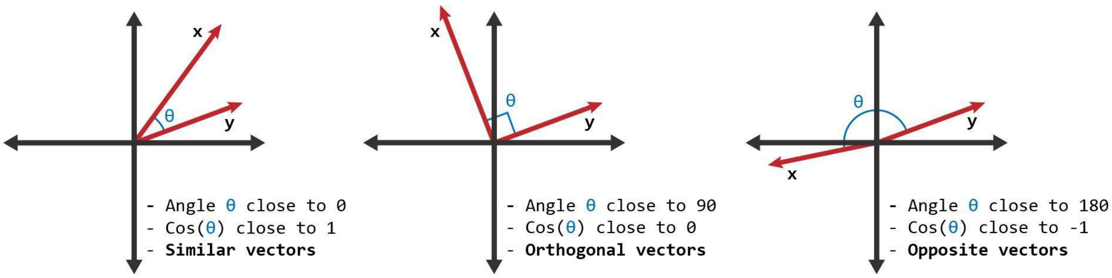
ii. **Vectorization :**

Vectorization is the process of converting data into numerical vectors that represent essential features. This transformation is especially useful in fields like natural language processing (NLP) and computer vision, where unstructured data needs to be represented in a form that algorithms can understand.

Bag of Words (BoW)
TF-IDF (Term Frequency-Inverse Document Frequency)
Word Embeddings
Sentence Embeddings

What is Cosine Similarity?
Cosine similarity is a metric used to measure the cosine of the angle between two vectors. In the context of text similarity, these vectors represent the TF-IDF representations of texts.

How does Cosine Similarity work?
Cosine similarity ranges from -1 to 1, where 1 indicates identical texts, 0 means no similarity, and -1 denotes completely dissimilar texts.
For two TF-IDF vectors, the cosine similarity is calculated by taking the dot product of the vectors and dividing it by the product of their magnitudes.



How Vectors are used to recommend similar contents using cosine angles

iii. Recommendation generation :

Generating a neighbourhood involved calculating the similarity between the given users within the user-item matrix. Similarity will be used to generate a recommendation for a specific user.

Algorithm:

i. Compare the similarity between all users with the active user.
ii. Select n users that have the highest similarity to build a neighbourhood
iii. Compute the prediction based on this similarity matrix.

We compute similarity between users by finding the Euclidian distance between them.

1. Data Description

In this proposed system user overcomes the disadvantages of the existing system by using MCLUE web application. If the user wants to watch a movie and doesn"t know which movie to watch he registers himself with the application and rates the movies he has already watched and builds a custom taste profile, and depending on the ratings, he/she is recommended movies by the application.

In the proposed Model the attributes used to calculate distance of each point from the centroid are

i. Tags

ii. Keywords

iii. Vectors

iv. Ratings

Here, different attributes have different weights and In our Recommended System. Every time when the user gives a rating to a particular movie an Expert for that Cluster in which that user belongs to will change dynamically.

2. Data Preprocessing:

The data preprocessing steps in the provided code are crucial for transforming raw movie data into a format suitable for building the recommendation system. Here's a detailed explanation of the data preprocessing steps involved:

- **Loading and Merging**: Combining movie and credits data.
- **Handling Missing/Duplicate Data**: Dropping rows with missing values and removing duplicates.
- **Parsing JSON Strings**: Extracting relevant fields from JSON-like strings in `genres`, `keywords`, `cast`, and `crew`.
- **Tokenization**: Splitting the overview text into individual words.
- **Replacing Spaces**: Ensuring multi-word names are treated as single tokens.
- **Tag Creation**: Aggregating all textual data into a single `tags` column.

- **Normalization**: Converting text to lowercase and stemming words to their root forms.

CONCLUSION

Movie Recommender web based application MCLUE . This application helps the user to find movies which are similar to his/hers taste. The existing system requires the user to go through several websites and read and watch reviews to come to a conclusion which is not efficient. The intension of this application is to reduce the time spent and increase efficiency by using content filtering techniques.

This project MCLUE showcases how we can use existing data about movies, such as their plots, genres, cast, and crew, to build a system that suggests similar movies to a user. By analyzing and processing this information, the recommendation system can identify patterns and relationships between movies. As a result, when you select a movie you like, the system can recommend other movies with similar features, helping you discover new films that align with your preferences.

The system relies on a process of breaking down and organizing movie details into meaningful data. It then uses this data to compare movies and find those that are most alike. The ultimate goal is to enhance the movie-watching experience by making it easier for users to find movies they will enjoy, without having to search through countless options.

In essence, this project demonstrates the power of data and technology in personalizing entertainment, making movie discovery more efficient and enjoyable for everyone.

REFERENCES

[1] Kantor, P. B., Rokach, L., Ricci, F., & Shapira, B. (2011). Recommender systems handbook. Springer.

[2] J.B.Schafer,D.Frankowski,J.Herlocker,S.Sen, Collaborative filtering recommender systems. The Adaptive Web, 291-324, 2007

[3] G. Adomavicius, A. Tuzhilin, Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Transactions on Knowledge and Data Engineering, 17(6), 734-749, 2005.

[4] M.J. Pazzani, A Framework for Collaborative, Content-based and Demographic Filtering. Artificial Intelligence Review, 13(5-6), 393-408, 1999

[5] Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). Recommender systems survey. KnowledgeBased Systems, 46, 109-132.

[6] Han, J., Kamber, M., & Pei, J. (2011). Data mining: concepts and techniques: concepts and techniques. Elsevier

[7] M. O. Conner and J. Herlocker. Clustering Items for Collaborative Filtering. In Proceedings of the ACM SIGIR Workshop on Recommender Systems, Berkeley, CA, August 1999.

[8] S.H.S. Chee , J Han , K. Wang . Rectree: An efficient collaborative filtering method. Lecture Notes in Computer Science, 2114, 2001

[9] L. H. Ungar and D. P. Foster. Clustering Methods for Collaborative Filtering. In Proc. Workshop on Recommendation Systems at the 15th National Conf. on Artificial Intelligence. Menlo Park, CA: AAAI Press.1998

[10] Yuanyuan Wang, Stephen Chi-fai Chan and Grace Ngai Applicability of Demographic Recommender System to Tourist Attractions : A Case Study on Trip Advisor , 2012 IEEE DOI 10.1109/WI-IAT.2012.133

[11] B. Sarwar, G. Karypis, J. Konstan and J. Riedl, Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering, Proceedings of the Fifth International Conference on Computer and Information Technology, 2002

A. Said, T. Plumbaum, W. E. De Luca, and S. Albayrak, “A comparison of how demographic data affects recommendation,” in Proc. 19th international conference on User modeling, adaption, and personalization, 2011