# Design and Implementation of Movie Recommendation System based on NLP And Content-based Filtering algorithm

**Mr. Omprakash Yadav #1, Krishna Mishra #2, Dhananjay Patil #3, Elvis Braganza #4,**

**Charles Finny #5**

*[1]Omprakash Yadav: Professor, Dept. of Computer Engineering, Mumbai University*
*[2]Krishna Mishra: Student, Dept. of Computer Engineering, Mumbai University*
*[3]Dhananjay Patil: Student, Dept. of Computer Engineering, Mumbai University*
*[4]Elvis Braganza: Student, Dept. of Computer Engineering, Mumbai University*
*[5]Charles Finny: Student, Dept. of Computer Engineering, Mumbai University*

***

**Abstract***: This is the era of information; Large amount of data is available. The availability of data is Only valid when it is beneficial to human in their work/ daily/labour. This system is a Personalized movie Recommendation which suggests the user which movie the individual should watch based on, the individual's previous interest, ratings & interaction with the system. The system will play an important role especially when the user has no clear view/idea of which movie, he/she should watch. The system is designed and implemented with the help of Content-based filtering algorithm the system is built and tested on the available data set and the test results showed that the system has good recommendation* effect.

**Keywords:** content-based filtering, Movie Recommendation, NLP Algorithm

- **Introduction**

In the era of emerging technology, we have been living in the world where data is everywhere available about almost everything, the data only needs to be used properly and correctly in order and use the data making the users work more effortless. Recommender System is a system that seeks to predict or filter preferences according to the user's choices. Recommender systems are utilized in a variety of areas including movies, music, news, books, research articles, search queries, social tags, and products in general Recommender systems produce a list of recommendations in any of the two ways – Collaborative filtering: Collaborative filtering approaches build a model from user's past behaviour (i.e. items purchased or searched by the user) as well as similar decisions made by other users. This model is then used to predict items (or ratings for items) that user may have an interest in. Content-based filtering: Content-based filtering approaches uses a series of discrete characteristics of an item in order to recommend additional items with similar properties. Content-based filtering methods are totally based on a description of the item and a profile of the user's preferences. It recommends items based on user's past preferences.

We are making Recommendation System That uses Content based filtering for making the recommendation decision. We focus on providing a basic recommendation system by suggesting items that are most similar to a particular item, in this case, movies. It just tells what movies/items are most similar to user's movie choice.

Content based filtering uses item features to recommend other items similar to what the user likes, based on their previous action or explicit feedback.

In this paper, the key research contents are to help users to obtain user-interested movie automatically in the massive movie information data using content-based filtering algorithm, and to develop a prototype of movie recommendation system based on content-based filtering algorithm.

## 1. Literature survey

**1.1 NLP algorithm** In Machine Learning Natural language processing (NLP) is a field in which computers understand, analyse, and derive meaning from the provided language or human language in a smart and useful and effective way. By using NLP, developers can learn, organize and structure knowledge which will help them to perform tasks such as automatic summarization, translation, named entity recognition, relationship extraction, speech recognition, and topic segmentation and many their task can be performed using NLP.

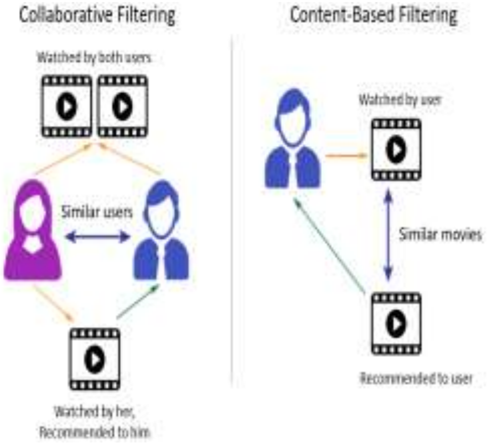
"Apart from common word processor operations that treat text like a mere sequence of symbols, Where NLP considers the hierarchical structure of language: several words make a phrase, and those several phrases make a sentence and, ultimately, the sentences is formed which convey the ideas. "By analysing language for its meaning, NLP systems have wide range useful roles in many fields, such as correcting grammar, converting speech to text and automatically translation between languages." NLP is used to analyse the text, allowing the machines to have a better understanding how human's provided data. This leads to human-computer interaction which enables real-world applications like, topic extraction, name and entity recognition, part of speech tagging, relationship extraction and many more. NLP is generally used for text mining, machine transaction. NLP is characterized as a difficult problem in computer science. Human language is rarely precise, or plainly spoken there are certain meaning of the word or sentences which may lead machine to confusion in making decision, if not handled properly. To understand the human language is not only to understand the words, but the concepts behind them and how they're linked together to create meaning. Despite of language being one of the easiest things for the human mind to learn, the ambiguity of language is what makes natural language processing a difficult problem for computers to master.

I. NLP Examples

- Use Summarizer to automatically summarize a block of text, exacting topic sentences, and ignoring the rest.

### 1.2 Content-based filtering

**Content-based** filtering —Content based Filtering is to make recommendations based on similar products/and services according to their attributes



Content based recommendation engines, which is the movie recommendation system that we developed, takes content or attributes of a product/movies you like, for example a movies genre, cast, director, keywords etc. , and then similarity matrix is calculated, based on the ranks other products/movies, on how similar they are to the liked product/movies, in this case we rank different movies based on how similar the recommended movies are to the liked movie using something called similarity scores.

The Recommendation system can also be built using collaborative filtering algorithm using k-mean algorithm[1], but the problem in using collaborative filtering algorithm is that, when there is an introduction of the new users or new items, it can cause the cold start problem, as there will be no data or insufficient data on these new entries for the collaborative filtering to work more precisely.

When new items are added to the system, they need to be rated by a substantial number of users before they could be recommended to users who have similar tastes to the ones who rated them. The new item problem does not affect content-based recommendation, because the recommendation of an item is based on its discrete set of descriptive qualities rather than its ratings.

And in collaborative filtering system, the system requires a substantial number of users to rate a new item before that item can be recommended. i.e. the prediction is often not accurate, if the user is newly created or/and haven't watched any movies yet.

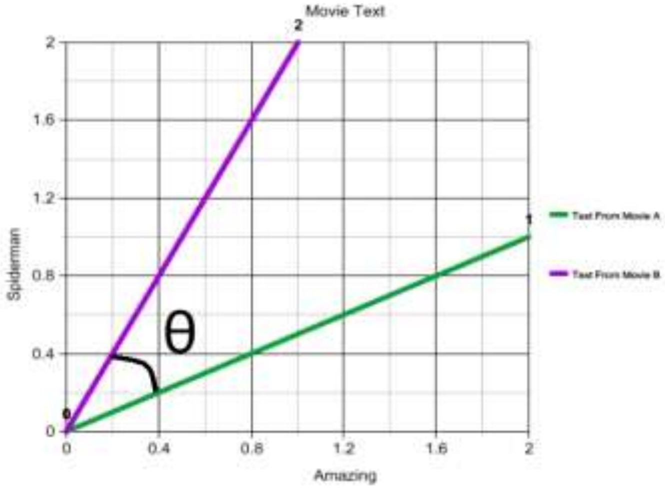
## 2 Finding Similarity

### 2.1 How similar are the text from each movie and how do we find the similarity between them?

First let us analyse the text,

**Text From Movie A:** The word "Amazing" is present 2 times and the word "Spiderman" is present 1 time.

**Text From Movie B:** The word "Amazing" is present 1 time and the word "Spiderman" is present 2 times. Now, let's go and plot this on a 2-Dimensional graph. Text from Movie A will have the point (1,2) and The Text from Movie B will have the point (2,1) where the X-axis on the graph indicates the number of times the word "Spiderman" appears and the Y-axis indicates the number of times word "Amazing" appears. The origin point for both vectors is (0,0). We can change text to a similar vector of word counts by using a **Count Vectorizer function** or just by doing what we did above.



Text graphed as vectors

Now, the two texts are represented as vectors and the closer the vectors angular distance are, the more similar they are. So, we can simply get the angular distance which is called theta and represented by the symbol θ to find the similarity between the two vectors. When thinking in terms of probability, machine learning, and likelihood it makes even more sense to use cos θ to get the similarity of the two vectors, this ensures that the value returned is between 0 and 1 since cos 90° = 0 and cos 0° = 1. Now we understand how to get similarities in 2-Dimensions for text represented as vectors and this method can be used for N-Dimensions as well where N is an arbitrary positive integer. So, in summary, we can get the similarity of text by changing the text into vectors and getting the angular distance (θ) between values 0 and 1 using cos θ and ultimately getting a similarity value between 0 and 1.

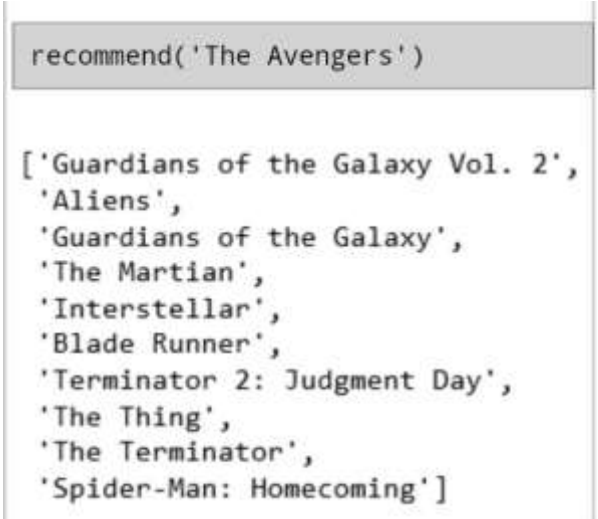
**2.2 Create vector representation for Bag_of_words, and create the similarity matrix**
The recommender model can only read and compare a vector (matrix) with another, so we need to convert the 'Bag_of_words' into vector representation using **Count Vectorizer**, which is a simple frequency counter for each word in the 'Bag_of_words' column. Once I have the matrix containing the count for all words, I can apply the cosine similarity function to compare similarities between movies.

$$similarity = \cos(\theta) = \frac{u \cdot v}{\|u\| \|v\|} = \frac{\sum_{i=1}^{n} u_i v_i}{\sqrt{\sum_{i=1}^{n} u_i^2}\sqrt{\sum_{i=1}^{n} v_i^2}}$$

$$u \cdot v = [u_1\ u_2\ \dots\ u_n] \cdot \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = u_1 v_1 + u_2 v_2 + \dots + u_n v_n = \sum_{i=1}^{n} u_i v_i$$

Cosine Similarity formula to calculate values in Similarity Matrix

**3. Result**



```
recommend('The Avengers')

['Guardians of the Galaxy Vol. 2',
 'Aliens',
 'Guardians of the Galaxy',
 'The Martian',
 'Interstellar',
 'Blade Runner',
 'Terminator 2: Judgment Day',
 'The Thing',
 'The Terminator',
 'Spider-Man: Homecoming']
```

The Top 10 Recommended Movies B the System

The movie recommendation system that we created provides good predictions based on the data set used for training and testing the model. The input given to the recommendation system from the data set and in the output, it provides the most similar movies that the user should watch next.

The system can only compare vector so the" bag of words" are converted in to vector using Count Vectorizer. Which is a simple frequency counter for each word in "bag of words" columns which is the used to find the cosine_ similarity Matrix. Here, the similarity between the Movies are found via natural language processing using cosine similarity Matrix.

- **Future scope of NLP**

There is a vast amount of data available in the digital era, in which we are living now. Most of the data i.e. about 79% of the data is in the form of text data. NLP being sub branch of data science, which helps to extract meaningful data and insight from, the raw available data. Here, NLP play's an important role in Data Science, especially in the field of text data which provides insight from text data. Experts have predicted that the demand for the NLP experts will grow exponentially in the near feature.

Future scope of NLP explains that, In NLP machines are thought to process and interpret text as it is done by the humans. NLP is considered as the "text analysis enabler and speech recognition applications. " NLP having the capability of interpreting text makes the task of analysing of Large amount of data simple and productive."

With the increment of data from various channels like Social and Mobile data, businesses need to have a solid technology which can access and evaluate customers sentiments. Up till now business have been only analysing customers actions, but in the current competitive environment, only analysing the customers action is outdated. Businesses needs to analyse and understand the customers Preferences, attitude and also their moods, this is all possible via sentiment analytics using NLP. Without using NLP business owners would not be able to do most basics sentimental analytics.

Here are some business areas that implemented NLP for text analysis to increase their productivity.

• Use of NLP in business, to exchange market intelligence with all stakeholders.

•Now a days Cat-bots Solves most of the common customer problems for customer call centres. Chat-bots provides human-like assistance to customers, minimizing the call loads and customer frustration.

•As mentioned before, businesses operators are increasingly relying on social data to monitor customer sentiments. Much of this data is text and requires NLP for sentiment analysis.

•NLP has substituted several customer-service functions with reliable service.

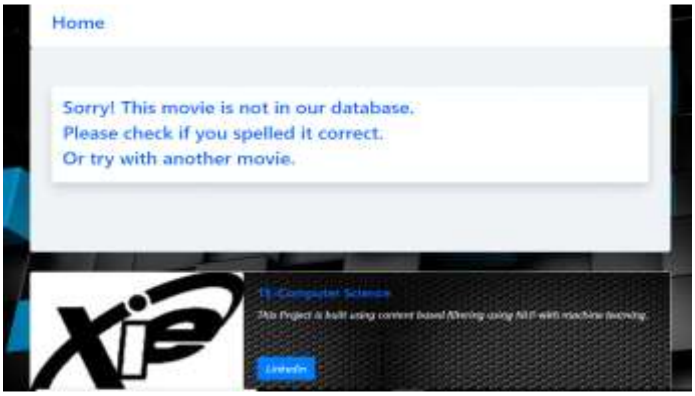
•NLP has also helped target advertising funnels targeted at segmented customers

- **PROJECT DESIGN AND CREATION**

After Pre-processing, testing and training the module, we moved on to design the Front-end for the developing the frontend we used Flask frame work. In the front-end design, the user is asked to enter the Hollywood movie on which they would like to have the recommendation. If the movie is present in to the database then based on the similarity matrix top 10 most similar movies are recommended by the system. /

If the entered movie is not present the system shows notification as "The Movie You Entered is not Present in the Database Please Enter another Hollywood Movie".

i. This figure shows that the Movie iOS not in the data base or it's spelled wrong.

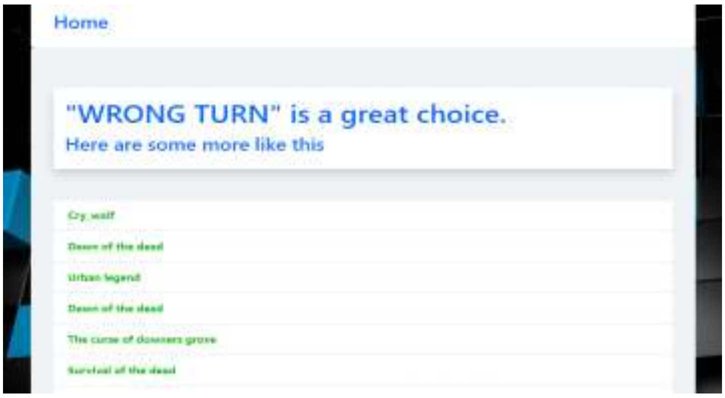

Home

Sorry! This movie is not in our database.
Please check if you spelled it correct.
Or try with another movie.

TE-Computer Science

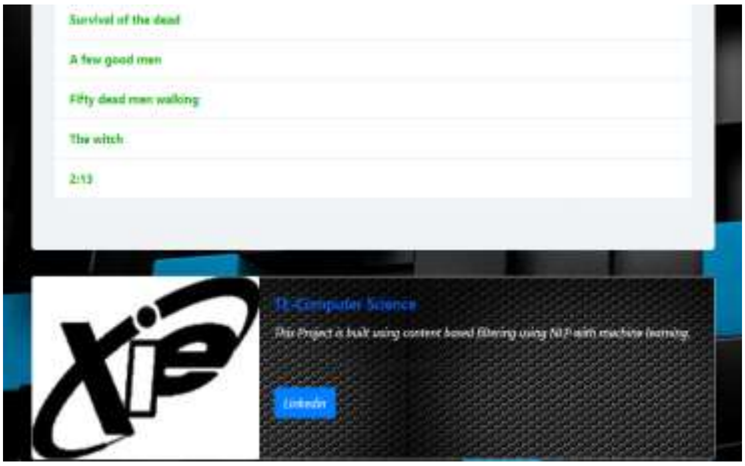
This Project is built using content based filtering using NLP with machine learning.

ii. This figure shows top 10 movie's which are similar to the users entered movie.


Home

"WRONG TURN" is a great choice.
Here are some more like this

Cry_wolf
Dawn of the dead
Urban legend
Dawn of the dead
The curse of downers grove
Survival of the dead


Survival of the dead
A few good men
Fifty dead men walking
The witch
2:13

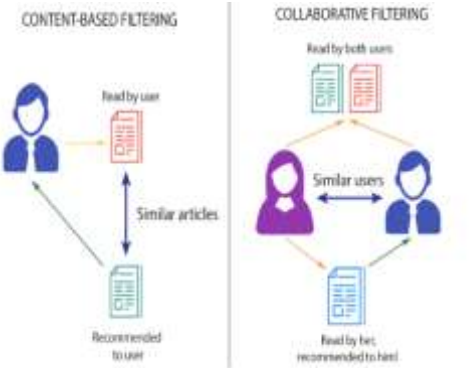
TE-Computer Science

This Project is built using content based filtering using NLP with machine learning.

- **Conclusion**

The model has recommended vary similar movies. From my "domain knowledge", We can see some similarities mainly based on directors, actor's and other plot's. We trained and tested the recommendation system. The accuracy rate is 80%. The movie recommender system provides very good prediction rate and is more reliable then the recommendation system based on collaborative filtering algorithm. Which is explained in content-based filtering (2.2) why we have used collaborative filtering?

As per image:



Content-based and collaborative-based filtering comparison

We have built the Recommendation system using content-based filtering with the of natural language processing .and conclude that the movie recommendation system which uses content-based filtering is more reliable and provides more accurate prediction and doesn't shows a new users/item problem when new one is added.

**References**:

[1] G. N. Yannakakis, ‘‘Game AI revisited,’’ in Proc. 9th Conf. Comput. Frontiers (CF), Cagliari, Italy, May 2012, pp. 285–292

[2] IBM, ‘‘Virtual worlds, real leaders: Online games put the future of businessleadershipondisplay,’’AGlobalInnovationOutlook2.0Report, 2007.

[3] V. M. Petrović, "Artificial Intelligence and Virtual Worlds – Toward Human-Level AI Agents," in IEEE Access, vol. 6, pp. 39976-39988, 2018.

[4] M.McPartlandandM.Gallagher,‘‘Reinforcementlearninginfirstperson shooter games,’’ IEEE Trans. Comput. Intell. AI in Games, vol. 3, no. 1, pp. 43–56, Mar. 2011.

[5] H. Wang, Y. Gao, and X. Chen, ‘‘RL-DOT: A reinforcement learning NPC team for playing domination games,’’ IEEE Trans. Comput. Intell. AI Games, vol. 2, no. 1, pp. 17–26, Mar. 2010.

[6] M. Cavazza, ‘‘Al in computer games: Survey and perspectives,’’ Virtual Reality, vol. 5, no. 4, pp. 223–235, 2000.

[7] D.FuandR.Houlette,‘‘PuttingAIinentertainment:AnAIauthoringtool for simulation and games,’’ IEEE Intell. Syst., vol. 17, no. 4, pp. 81–84, Jul. 2002.

[8] D. Isla, ‘‘Handling complexity in the halo 2 AI,’’ in Proc. Game Developers Conf., vol. 12, 2005.

[9] I. Millington and J. Funge, Artificial Intelligence for Games, 2nd ed. San Mateo, CA, USA: Morgan Kaufmann, 2009.

[10] E. F. Anderson, ‘‘Playing smart—Artificial intelligence in computer games,’’ in Proc. zfxCON Conf. Game Develop., Hankensbüttel, Germany, Oct. 2003.