iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** X **Month of publication:** October 2021

DOI: https://doi.org/10.22214/ijraset.2021.38591

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue X Oct 2021- Available at www.ijraset.com

Voice Assistant based Movie Recommendation System utilizing User Filtered Correlation Factor

Karthik Babu Vadloori[1], Shriya Madhavi Sanghishetty[2], Ram Teja Chittuluri[3], Krushitha Mattewada[4]
[1, 3]*Dept of Computer Science Engineering, Sreenidhi Institute of Science and Technology*
[2, 4]*Dept of Computer Science Engineering, Malla Reddy Institute of Engineering and Technology*

Abstract: ***The project deals with the idea of a Movie Recommendation System, which itself is one of the key tools being deployed across various OTT and Social Media platforms that serve the purpose of engaging a customer of a particular service to stick to their platform service by consuming more relevant content. Primarily the crucial factor to boost the momentum of the Recommendation Systems date back to the emergence of Search Engines (as a part of everyday lives). These search engines make use of multiple algorithms to page and derive the useful results by indexing every page of a hosted/ live site. Due to these technological advancements, there has been a burgeoning of data, which is being generated at every mill-second from numerous streams and multitudes across various fields. While the utility of platforms - like the afore mentioned - give rise to Historical Data, where the behavioural traits of multiple users give scope to Clustering upon a set of distinct characteristics. One such method to suggest (or) recommend movies to online users is to utilize the already available historical data. Within its entirety the Recommendation Systems have become one of the most prominent tools within the Machine Learning framework, serving various industries and sectors. Our project uses the already available Historical Data downloaded from Kaggle and develops the Voice Assisted Approach to feed input to our Movie Recommendation System.***

Keywords: Movie Recommendation System, User-based Collaborative Filtering, Voice Assistant, Pearson's Correlation, Historical Data, Speech Recognition, Tkinter, Data Visualization

I. INTRODUCTION

Given the fact that there are huge number of movies available in various parts of the world, it can be quite challenging for a user to find the appropriate movies suitable for his/her tastes. Every individual may have affinity towards different genres and movies that as a person would be their choice of preference. There is clearly a distinction between every individual. There is merely any chance that there is a 100% similarity between any set of users, but still there is a correlation that exists to some extent which can be lucrative for smart decision oriented predictive outcomes. Different users like different movies or actors. Likewise, it is important to find a method of filtering irrelevant movies and/or find a set of relevant movies.

The answer to the above tasks is Movie Recommendation System. A system that has the capability to perform the above tasks effectively has a scope for immense implications in the present world and market. To grasp a clear understanding, the average human attention span around the globe has been decreasing greatly, simultaneously the level of entertainment related tools and platforms are growing rapidly, in such a manner that makes it tough to spend some time to organize and quantitatively analyse and filter out our needs from the already developed massive pool of choices. Such a system i.e., Recommendation System are being endorsed in every single field around the world. Adding with the success of these systems in different sectors and domains, made them a prominent tool against hitherto existing ones. Recommendation systems are being utilized in different domains such as online OTT, bookstores, TV programs, jokes, news articles. It is one of the most important research in the digital television domain [1] currently and have the scope as the rise of decentralized form of internet is posing a huge potential of growth as the need to tackle the next Big Data will require smart and decentralized working algorithms, although there could be intrinsic cons to this statement*.

The most well-known recommendation systems are mainly based on User-based Collaborative Filtering (UbCF) [2] and Content-based Filtering (CF) [3]. UbCF first tries to find out the groups of similar users automatically from a set of active users. The similarities between users are computed using correlation measure. It then recommends items to a user based on the opinions of the user groups [1]. In this paper, we propose User-based Collaborative Filtering method for the movie recommendation using Pearson's correlation technique. We have implemented the above methodology by inputting our system using a Voice Assistant – Speech recognition for input and Pyttsx3 for making the virtual assistant speak out – mode wherein we will utilize the concerned modules for a virtually handsfree experience.

The effectiveness of our system demonstrated using Kaggle originated Movies and Ratings Data Sets (MovieLens – University of Minnesota) [4]. The rest of this paper is organized as follows. In Section II we discuss the Related Work. In Section III, we define the process of Implementation & Architecture. In Section IV, we discuss Subjective Understanding by considering a practical example. In Section V, we brief the experimental results we have obtained and discuss various issues related to the system. In Section VI, we conclude the paper with a direction of future work.

II. RECOMMENDATION SYSTEMS

Recommender/ recommendation systems are the systems that are particularly developed and designed to suggest (or) recommend items to a particular user based on various factors. The purpose of these systems is to recommend the probable item that a user would be mostly interested to consume (or) purchase next. Multiple MNCs have been utilizing these systems to a vast measure that contribute to a huge share of revenue as they serve the purpose of retaining the customer. Companies like Netflix, Amazon, etc., are famously known for their advance recommendation systems that have been historically accurate to predict the user's next likely purchase or item for consumption i.e., a relevant content – in our case the movies on OTTs. The recommender systems in layman terms, are bound to deal with large amounts of data, which is then filtered based on multiple intrinsic factors and other core information that serves the purpose. Our paper is focused on User-based Collaborative Recommendation System; hence in our next section we will focus on the theoretical aspects behind it.

III.RELATED WORK

Collaborative filtering recommender system has been inculcated into the science and research spectrum very much in today's world. Historically, the need for efficient systems proposing the items in recommended form garnered momentum as the advent of technology grew rapidly.

The first ever recommendation system was introduced as a proposal in the late 1990s [6]. The spark which gave rise to these systems was the result of the elevated Tapestry Project in the year 1992 [5]. Since then, the field evolved into an active area of focus where immense contributions by various researchers and scientists were seen. It became a hub of research advent for Data Scientists, as the growth of data had become a booming incentive-oriented choice for explorations within the multitude of these systems. It primarily roots from the extensive application and development in the fields of cognitive science, approximation theory, information retrieval, forecasting theories and even market analysis and predictions [6].

In today's modern world, Recommender systems have been developed and evolved, and have primarily adopted the below three widely utilized approaches:

A. Clustering based Algorithms
B. Filtering Algorithms
C. Content/ feature-oriented Filtering

Clustering based approach functions on the basis of segregation of the given set of elements in a particular sample space into clusters. The key here is to key characteristics that uniquely identify and bind a certain set of elements together. The aim is to minimize the variance between elements and categorically aligning them to a cluster for further processing and fitting. These algorithms are used in scalability related problems that need some assured level of accuracy bounded to the outcome [7].

Filtering algorithms utilize an N-dimensional vector space, where N being the number of unique entities. The Positive values in the space synchronise with the positive user input, and vice versa for negative inputs also. Whenever there is a correlation established between a group of entities, we obtain a pattern that enables indicators that might be of interest to the end user in understanding the output's relation with the input [8].

Thus, the users (present in the historical data) who correlate with the end user are known as neighbours and concluded to have 'similar tastes' between them.

Content/ feature-oriented algorithms are known to have a much more intrinsic insight as the later focuses of the history of a particular user's engagement in a platform. It takes input as a form of query and suggests items based on the historical data only of that end user. Characteristics like ratings, engagement (in hours/ minutes), frequency of items on the basis of actors, directors (or features that are applicable to delude and derive a unique proposition factor that appeals the end user) [3].

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue X Oct 2021- Available at www.ijraset.com

IV. IMPLEMENTATION AND ARCHITECTURE

The architecture of our recommendation system consists of three main segments.

1) Speech Recognition Module for Input.
2) Movie Recommendation System Segment.
3) Pyttsx3 (text-to-speech) module for Output.

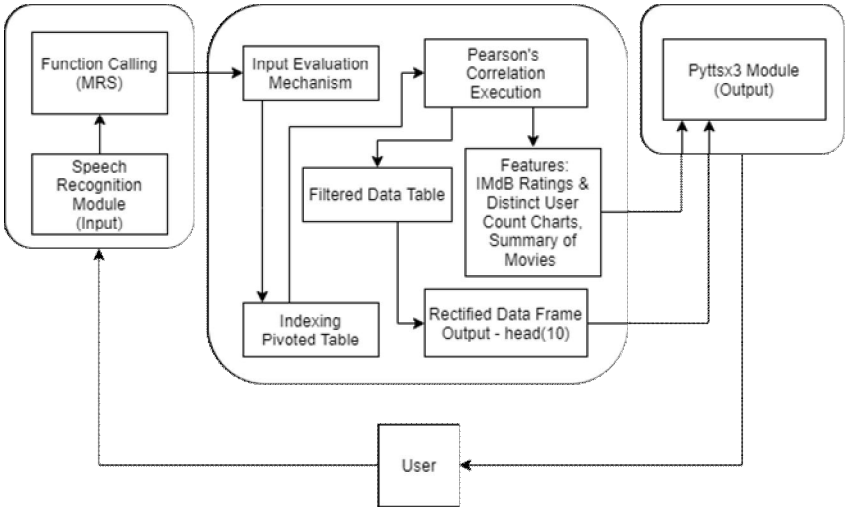


Figure: 3.1 Underlying Architecture of the proposed system.

The implementation of the above architecture is carried out by undergoing a series of essential steps ranging from obtaining the data, collecting, rectifying, and merging them upon our interests for the project.

A. Data Collection:

This is a crucial step where the datasets are obtained from various potential sources. In our case, we obtained it from Kaggle. However, the datasets were originally developed by University of Minnesota and called as MovieLens Datasets.

1) Movies Data Set

	movieId	title	genres
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	2	Jumanji (1995)	Adventure|Children|Fantasy
2	3	Grumpier Old Men (1995)	Comedy|Romance
3	4	Waiting to Exhale (1995)	Comedy|Drama|Romance
4	5	Father of the Bride Part II (1995)	Comedy

Table: 1 Movies Data Set consisting of three main features

2) Ratings Data Set

	userId	movieId	rating	timestamp
0	1	1	4.0	964982703
1	1	3	4.0	964981247
2	1	6	4.0	964982224
3	1	47	5.0	964983815
4	1	50	5.0	964982931

Table: 2 Ratings Data Set consisting of four main features

The feature movieId is the common amongst both the data sets and it is the key for our further study.

B. Data Pre-processing

	userId	movieId	rating	timestamp	title	genres
0	1	1	4.0	964982703	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	5	1	4.0	847434962	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
2	7	1	4.5	1106635946	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
3	15	1	2.5	1510577970	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy

Table: 3 Merged Data Frame – Two data sets

It involves pre-processing the datasets for overlook and analysing the descriptive features and removing unnecessary features for better utility. The target in this step is to transform our datasets into a human understandable and clear-cut format for better comprehension such that it could be relatively easier to work on the data sets. We have imported the datasets and merged them on 'movieId' column/feature and made them ready for visualizing and further performance.

C. Data Analysis

1) Heatmap

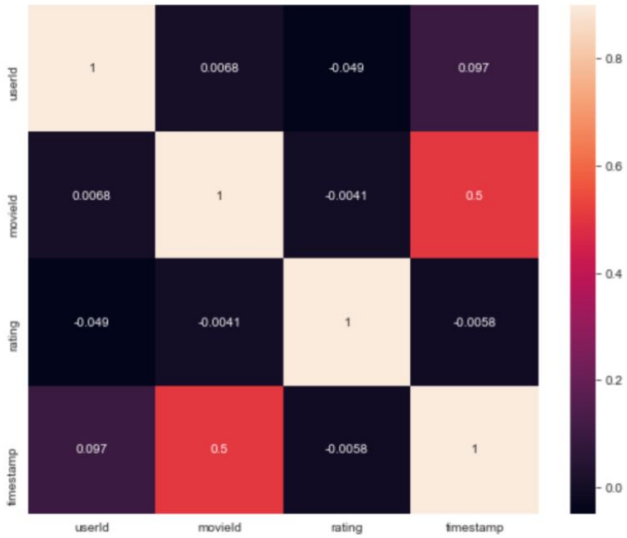


Figure: 3.2 Correlation Heatmap on features of merged dataframe.

It is a correlation based map that indicates the relationship between two features in a particular data frame, thereby helping us with our further analysis. In this case, we can see there is higher possibility of correlation between movieId and timestamp.

2) Average Movie Ratings Plot

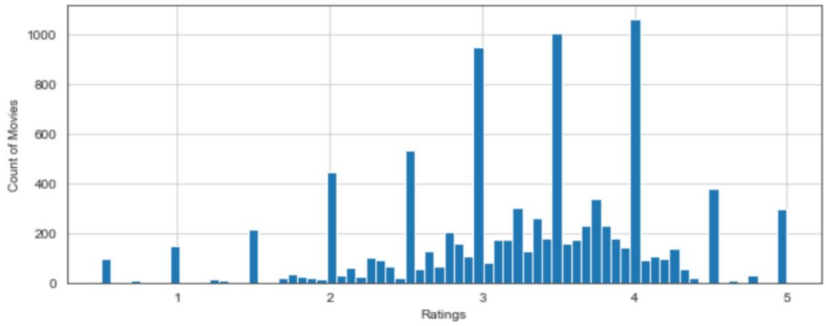


Figure: 3.3 Histogram of Average Ratings of movies – with ratings ranging from 0 to 5. Total No. of Movies: 9742

It is to be noted that integer ratings are the highest. Reason: less users watching certain movies etc.

3) Number of Users V/s Average Movie Ratings – Seaborn Joint Plot

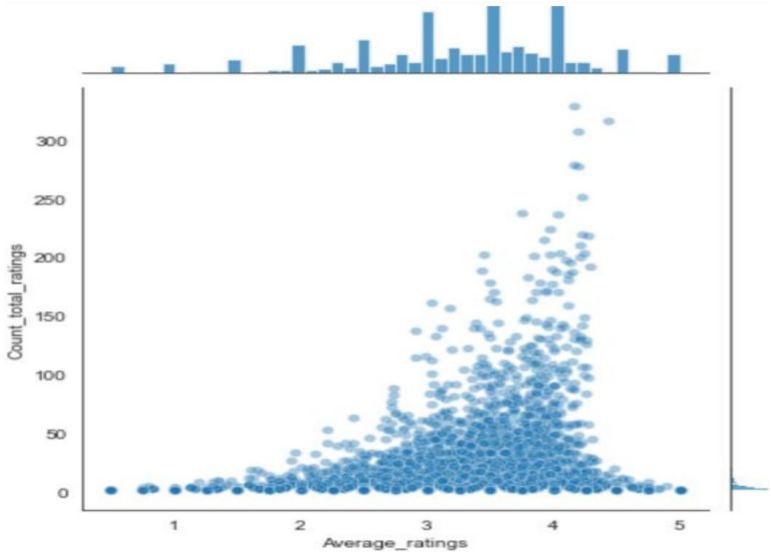


Figure: 3.4 Visualizing the Average movie ratings from 0 to 5 and the count of users who have rated these. It briefs us the demographic view for an overall understanding of the data.

After performing the Data Analysis and Visualizing them for our convenience we are capable to proceed our study further and develop the system.

D. Generating Pivot Table

title	'71 (2014)	'Hellboy': The Seeds of Creation (2004)	'Round Midnight (1986)	'Salem's Lot (2004)	'Til There Was You (1997)	'Tis the Season for Love (2015)	'burbs, The (1989)	'night Mother (1986)	(500) Days of Summer (2009)	*batteries not included (1987)	...
userId											
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...
5	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...

Table: 4 The above Pivot Table consists of occasional values of ratings in place of NaN.
It states the number of ratings a particular user has given to those number of movies.
It helps us indicating the number of movies a particular user has watched.

E. Systematic Study of the Architecture

1) Speech Recognition Module for Input

```
Welcome to Movie Recommendation System. Please tell me the index or name of the movie that you have watched.

listening...
223
<class 'int'>
```

Figure: 3.5 listener() method implementation.

Speech: 'Welcome to Movie….'
Input: 223
It is developed using the 'speech_recognition' package available open source and deployed for recognizing the user input. For it, we need to create listener object and use it with microphone as our source to input. Upon input, the function present in the code segment is called and processed into the MRS segment.

2) *Movie Recommendation System Segment:* After the processed input is sent to the function, it is evaluated for its type – either integer or character i.e., Movie's Index Number or its Full Name – and then indexed in the pivoted table. Now it is correlated using the 'N.corrwith(target)' method, where N is the already existent pivoted table and 'target' is the inputted table which is evaluated for performing the recommendation operation. The data obtained is now filtered collaboratively and rectified for a proper data frame.
3) *Pyttsx3 (text-to-speech) module for Output*

title	Correlation
Coneheads (1993)	1.0
Waterworld (1995)	1.0
Three Musketeers, The (1993)	1.0
Losing Isaiah (1995)	1.0
Father of the Bride Part II (1995)	1.0
RoboCop 3 (1993)	1.0
Kiss of Death (1995)	1.0
Pretty Woman (1990)	1.0
Casper (1995)	1.0
Cliffhanger (1993)	1.0

Figure: 3.6 Output that is read out by voice assistant module. Index No. 223.

The above module is also known as text-to-speech module which basically converts the content generated by the MRS i.e., the data frame given above into speech output. Every single record of the columns is read out in our proposed system, and by adding more features on the need to-do basis we can expand the project's outcome criteria further.

V. SUBJECTIVE UNDERSTANDING

A. *Foundation for User-based Collaborative Filtering:*

1) *Pearson's Correlation Formula*

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

r = correlation coefficient
x_i = values of the x-variable in a sample
$\bar{x}$ = mean of the values of the x-variable
y_i = values of the y-variable in a sample
$\bar{y}$ = mean of the values of the y-variable

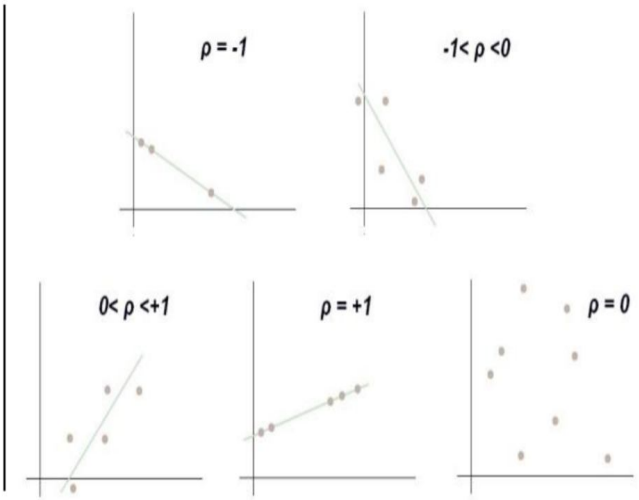


Figure: 4.1&4.2 Pearson's Formula/ Linear Impact of Pearson's Coefficient ('r' or ρ) in a 2-D plane.

a) Using the above formula our hypothesis to find the similarities can be engendered further by scaling the outcome on a range of -1 to 1. With 1 showing the highest similarity correlation.
b) It is one of the most widely used methodologies in statistics and it is used to colloquially measure the strength and direction of a linear relationship between two variables. Values always range between -1 (strong negative relationship) and +1 (strong positive relationship), and the symbol to demonstrate it is 'ρ', or 'r' or Sim(X, Y).
c) In the above formula 'r' is defined as the correlation coefficient and also can be categorized as Similarity Coefficient – Sim(a, b) – between two elements or users. On the right side, we see the impact of **ρ** in xy-plane.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue X Oct 2021- Available at www.ijraset.com

B. Practical Implementation using an Example

1) Let us consider a set of 4 Users –A1, A2, A3 and Joe. And each of these users have watched a union of 4 movies. Wherein each person might have watched only 1 movie to all 4. '?' indicates that the user hasn't watched the particular movie.

	Avengers	Hulk	The Girl Next Door	Forrest Gump
A1	Y	Y	N	N
A2	Y	N	Y	Y
A3	Y	Y	N	Y
Joe	?	Y	?	?

Table: 5 Users and their ratings for movies

2) Now our target user is Joe, and he wants us to recommended him the next movie by utilizing the collaborative filtering. From the above-mentioned users, we will follow a voting-based approach to be implemented in the formula. We shall assign 1 to 'Y' and 0 to 'N', and mark the Similarity Coefficient between each and every user in every particular movie.

The steps we follow are:

1) Detecting the 1's in movie column we just watched. This is known as Knowledge-Filtering so that we can perform our operation to find the similarities between these users.

2) From the above example, we have found A1 and A3 have 'Y' in the movie 'Hulk' and hence we will work on it only.

3) Then we will assign votes to each movie within the scope of A1 and A3.

4) Finally, the movie with highest number of votes will be recommended first.

	Avengers	Hulk	The Girl Next Door	Forrest Gump
A1	Y	Y	N	N
A2	Y	N	Y	Y
A3	Y	Y	N	Y
Joe	?	Y	?	?
	2 votes		0 votes	1 vote

Table: 6 Voting between the movies within the scope of filtered users.

5) After evaluating the votes, the movie with highest number of votes → in our case 'Avengers' (with 2 votes) is going to be recommended first, followed by 'Forrest Gump' and 'The Girl Next Door'.

6) These are calculated with the Similarity Coefficient i.e.; the Pearson's Formula is rectified in the form of Similarity Relationship. It is represented as follows,

$$Sim(a,b) = \frac{\sum_p (r_{ap} - \bar{r}_a)(r_{ab} - \bar{r}_b)}{\sqrt{\sum (r_{ap} - \bar{r}_a)^2} \sqrt{\sum (r_{bp} - \bar{r}_b)^2}}$$

$$r_{up} : rating\ of\ user\ u\ against\ item\ p$$

$$p : items$$

Therefore, we obtain the output of recommended list along with the coefficient,

title	Correlation
Avengers	1.0
Forrest Gump	0.5
The Girl Next Door	0.0

Table: 7 Correlation Table stating the Recommended Movies as output. Type=SeriesObject.

C. Examining Voice Modules:

1) Pyttsx3 (text-to-speech) Module

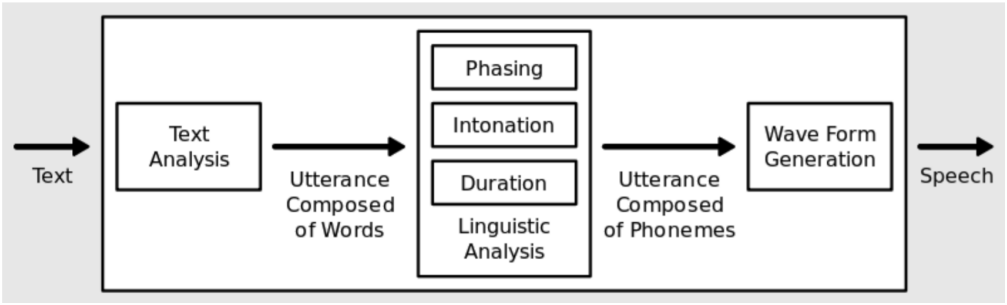


Figure: 4.3 In-built workflow architecture of a text-to-speech module – Pyttsx3

2) Speech Recognition Module

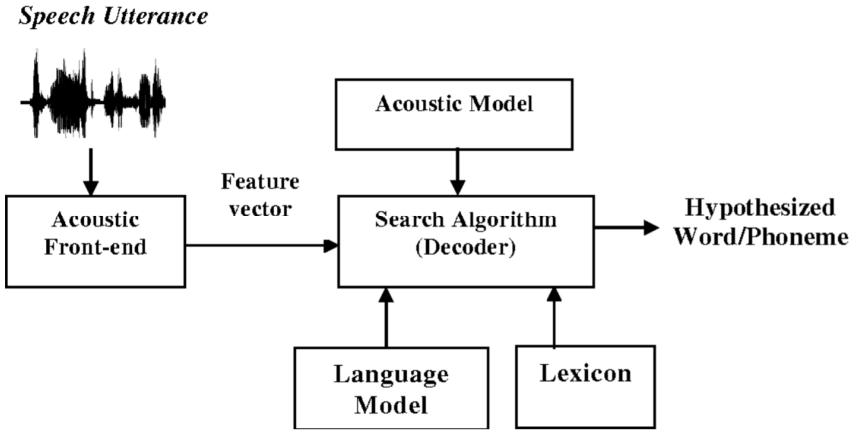


Figure: 4.4 In-built workflow architecture of the Speech Recognition module

VI. RESULTS AND DISCUSSIONS

A. Resultant Crux Points

In the given Movie Recommendation System, we have developed the technique where the MovieLens Data Sets have been optimally utilized for building the tool. The overall project has been an amalgamation of the Movie Recommendation System based process and Voice Assistant based modules. It has been built in such a way that there is no requirement for a user to input other than the voice input. Below is the list of executive steps that have been carried out to build our project successfully under certain systematic constraints:

1) Collective action of Data being gathered. Useful data with key features have been obtained.

2) Performed the Data Pre-processing step with feature evaluation, converting them into data frames and manipulating existing features according to our convenience and traversing them for user appropriacy.

3) Merging the data frames based on a primary key between the two of the previously exercised datasets.

4) Carried out a Data Analysis based on the ratings feature and plotting the obtained results on a graph. We have briefly made the Heatmaps based correlation visible and utilized Joint plotting technique to further excel our understanding of the data.

5) Displayed a maximum list of movies along with its index by using '.set_maximum()' method in the Pandas library for choosing a movie of our choice i.e., the movie we watched.

6) Built a Pivot table made using the distinction of User Watched Movies, and aired to an understanding that converting ratings as a measure to understand that a user has watched a movie. Finally, used the Pivot table as the base for our further correlation of the system in various instances of inputs as movies, either as its name or its index.

7) Implemented the Speech Recognition module using its listener class object and gave the input to the Movie Recommendation System - that we have developed as a part of a function – using the Microphone as a source. It is however suggestable to use a proper microphone to avoid infiltration of unnecessary noise sounds into the system.

8) The data that was inputted to the system is converted into a pivot table having only that one particular movie feature and its corresponding user data as records, and then it is correlated with the existing major pivot table.
9) The correlated data is streamlined to form a data frame and sent in to the instance of the Pyttsx3 module as the input which then gives us the output in the form of speech.
10) The Pyttsx3 module has 'say()' method which reads out the human understandable data that is given to it.
11) After these evaluation-based exchanges and implications, we conclude that our project of Movie Recommendation System based on a Voice Assistant approach has been successfully completed.

B. Discussions on Future Enhancements

Beginning with the scale of these systems being utilized in today's world we can realise the prominence that they have engraved into the fields of Entertainment, IT, Health Industry and e-commerce etc. This is solely because of the increasing and growing markets around the globe. The momentum that we are witnessing the potential boom of continuous generation of data has become a lucrative yet competitive challenge for many stakeholders. The rightful and efficient entity that makes effective utility of these systems has more chances to become a major player in the market. If we consider the scope of these systems in terms of technological perspective, we have to grab an onsite look over the amount of research happening around the various laboratories. There is a tremendous scope to these systems as the advent of decentralized inter-networking has brought a paradigm shift in the way we have to implement any system. The need for resources has grown multiple fold and there is a rapid need for computational capacities as the corresponding data being generated is very vast in quantity. As new business entities tend to adopt these systems into their Customer Reach and Relationship Management Divisions the more challenging and competitive, they tend to become. The need to suit their type of customer base makes it reasonable to invest into making the system precise in terms of the type of inputs we provide. We can provide more demographical information such as nationality, location, race, mother-tongue etc., to make the system more intrinsic to the platform provided by the business entity [4].

VII. CONCLUSION

In this paper, we have successfully developed a Voice Assistant based Movie Recommendation System that utilizes Pearson's Correlation Methodology to derive a User-based Collaborative Filtering which is our faction to perpetuate the systemic implementation of this system. Using this system, we have built a framework approach to systematically get a recommendation of movies that would be of our interests based on the working that we have executed over the Historical User Data from the MovieLens. We have made a tool that takes in voice-based input from the user, wherein the input can be of index value of any particular movie that he/she has watched or would like to be recommended based on it, or directly the name of the movie itself. The user can give voice input consisting of only few words of the movie and the system will automatically present the output by tallying those keyword inputs in the existing list of movie titles. Apart from the core operations as mentioned above, the project also performed an in-depth EDA over the data after pre-processing it, and displayed the charts consisting of distinct number of movies with their corresponding ratings. It was implemented using Histogram and Joint plotting techniques. Heatmaps and describe methods for better understanding of the data during the pre-processing stage. Finally, it can be understood that our study over the Movie Recommendation System Using Automated Voice Assistant feature is viable for multiple future enhancements and can be accessed as a utility tool for various fields and sectors as per the requirements generated by the concerned party. We have assessed these points in the discussions part of the previous section

VIII. ACKNOWLEDGMENT

This research was possible by the guidance, support and motivation provided by our faculties who have boosted our esteem to pursue our interests in the field of Recommender Systems. We are indebted to Mr. Hassan Mohammed, *Assistant Professor, Dept of CSE, MRIET* and Dr. Sundaragiri Dheeraj, *Assistant Professor, Dept of CSE, SNIST*.

REFERENCES

[1] S. Halder, A. M. J. Sarkar and Y. Lee, "Movie Recommendation System Based on Movie Swarm," 2012 Second International Conference on Cloud and Green Computing, 2012, pp. 804-809, doi: 10.1109/CGC.2012.121.
[2] J.S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence, pages 43–52. Morgan Kaufmann Publishers Inc., 1998.
[3] M. Balabanovic and Y. Shoham. Fab: content-based, collaborative recommendation. Communications of the ACM, 40(3):66–72, 1997.

[4] Subramaniyaswamy, V., Logesh, R., Chandrashekhar, M., Challa, A., & Vijayakumar, V. (2017). A personalised movie recommendation system based on collaborative filtering. International Journal of High Performance Computing and Networking, 10(1/2), 54. https://doi.org/10.1504/ijhpcn.2017.083199

[5] Colombo-Mendoza, L.O., Garcia, R.V., Rodriguez-Gonzalez, A., Alor-Hernandez, G. and Samper-Zapater, J.J. (2015) 'RecomMetz: a context-aware knowledge-based mobile recommender system for movie show times', Expert Systems with Applications, Vol. 42, No. 3, pp.1202–1222.

[6] Adomavicius, G. and Tuzhilin, A. (2005) '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, June, Vol. 17, No. 6, pp.734–749.

[7] McNee, S.M., Riedl, J. and Konstan, J.A. (2006) 'Being accurate is not enough: how accuracy metrics have hurt recommender systems', ACM CHI '06 Extended Abstracts, ACM, pp.1103–1108.

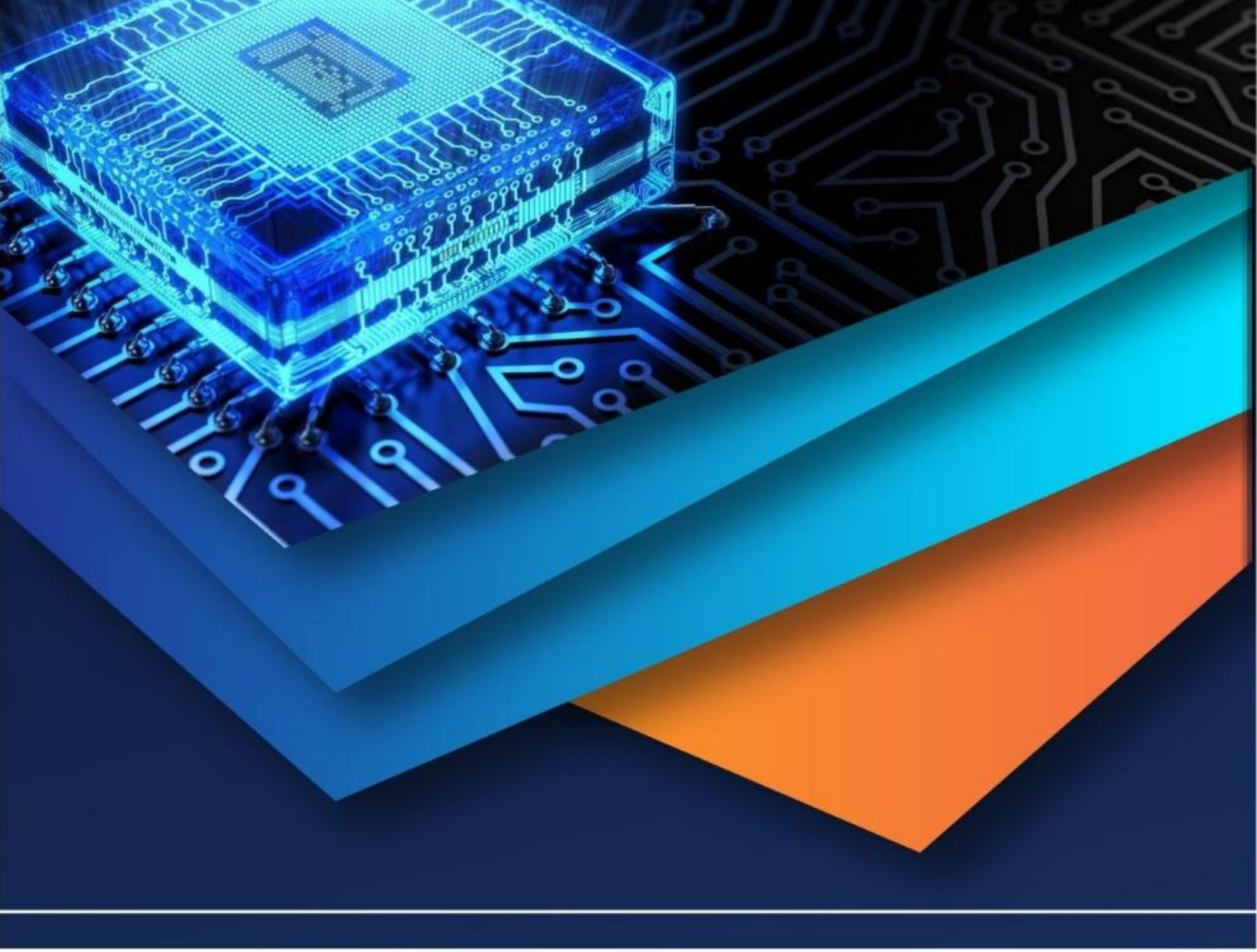
[8] Linden, G., Smith, B. and York, J. (2003) 'Amazon.com recommendations: item to item collaborative filtering', IEEE Internet Computing, Vol. 7, No. 1, pp.76–80.

[9] J. Padhye, V. Firoiu, and D. Towsley, "A stochastic model of TCP Reno congestion avoidance and control," Univ. of Massachusetts, Amherst, MA, CMPSCI Tech. Rep. 99-02, 1999.

[10] Primkulov S., Urolov J., Singh M. (2021) Voice Assistant for Covid-19. In: Singh M., Kang DK., Lee JH., Tiwary U.S., Singh D., Chung WY. (eds) Intelligent Human Computer Interaction. IHCI 2020. Lecture Notes in Computer Science, vol 12615. Springer, Cham. https://doi.org/10.1007/978-3-030-68449-5_30

[11] Pant T. (2016) Understanding and Building an Application with STT and TTS. In: Building a Virtual Assistant for Raspberry Pi. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-2167-9_2

[12] Rallabhandy S., Rodda S. (2020) KeyBoard-Less Online Shopping for the Visually Impaired Using Natural Language Processing and Face Recognition Mechanism. In: Satapathy S., Bhateja V., Mohanty J., Udgata S. (eds) Smart Intelligent Computing and Applications. Smart Innovation, Systems and Technologies, vol 160. Springer, Singapore. https://doi.org/10.1007/978-981-32-9690-9_25



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY