make appropriate suggestions. If a music streaming app is not able to predict and play the music that the user likes, then the user will simply stop using it. This has led to a high emphasis by tech companies on improving their recommendation systems. However, the problem is more complex than it seems. Every user has different preferences like in addition even the taste of a single user can vary depending on a large number of factors, such as good mood, season, or type of activity the user is doing. Since each user is different, this approach is considered to be too simple.

1.2 PROBLEM DEFINITION

A chatbot is a tool to retrieve information and generate human-like conversation. It is mainly a dialog system aimed to solve/serve a specific purpose. Chatbots have gained increasing importance for research and practice with a lot of applications available today including Amazon's Alexa or Apple's Siri. In this project, we present the underlying methods and technologies behind a Chatbot for movie recommendation that allows people to textually communicate with the purpose of movie recommendation, ratings, reviews, cast, and other interesting insights. It is a simple bot that answers questions about movies.

The user can ask about ratings, people voted for the movie, genre, movie overview, similar movies, IMDb links, budget, revenue, and adult content. The data and processing are all handled in the local system. Even though we use IBM, it is used as an API service and none of the internal data is sent to IBM. This way the entire design can be implemented in your workplace without having to worry about data transfers. Besides the underlying foundations, we provide a use case from the intended movie domain to show how such a model chatbot effectively can be used in practice.

1.4 FEATURES

The basic idea behind this system is that movies that are more popular and critically acclaimed will have a higher probability of being liked by the average audience. Second is content-based filtering, where we try to profile the user's interests using information collected and recommend items based on that profile.

- Create a user account.
- Record his/her history.
- Based on the history, I recommend more movies.
- Based on his previous rating, I recommend movies.

- Also recommends movies based on a similar genre.
- Can track the preferred IMDB-rated movies based on his history.
- Can track the most preferred movie genre among n users.

LITERATURE REVIEW

These chapters summarize some of those research papers. We studied various papers describing the research done in this domain and also the comparison of various models.

Based on a collaborative filtering approach. The system also has a provision for the user to select attributes on which he wants the movie to be recommended. Collaborative filtering makes use of information provided by the user. MovREC [10] is a movie recommendation system presented by D.K. That information is analyzed and a movie is recommended to the users which are arranged with the movie with the highest rating first. Yadav et al.

Luis M Campos et al. [5] have analyzed two traditional recommender systems i.e. content-based filtering and collaborative filtering. As both of them have their

drawbacks he proposed a new system which is a combination of Bayesian network and collaborative filtering. The proposed system is optimized for the given problem and provides probability distributions to make useful inferences.

A hybrid system has been presented by Harpreet Kaur et al. [9]. The system uses a mix of content as well as a collaborative filtering algorithm. The context of the movies is also considered while recommending. The user-user relationship, as well as the user-item relationship, plays a role in the recommendation.

The user-specific information or item-specific information is clubbed to form a cluster by Utkarsh Gupta et al. [12] using chameleon. This is an efficient technique based on Hierarchical clustering for recommender systems. To predict the rating of an item voting system is used. The proposed system has a lower error and has better clustering of similar items.

Urszula Kużelewska et al. [6] proposed clustering as a way to deal with recommender systems. Two methods of computing cluster representatives were presented and evaluated. Centroid-based solution and memory-based collaborative filtering methods were used as a basis for comparing the

effectiveness of the proposed two methods. The result was a significant increase in the accuracy of the generated recommendations when compared to just the centroid-based method.

Costin-Gabriel Chiru et al. [3] proposed Movie Recommender, a system that uses the information known about the user to provide movie recommendations. This system attempts to solve the problem of unique recommendations which results from ignoring the data specific to the user. The psychological profile of the user, their watching history, and the data involving movie scores from other websites are collected. They are based on aggregate similarity calculation. The system is a hybrid model which uses both content-based filtering and collaborative filtering.

To predict the difficulty level of each case for each trainee Hongli Lin et al. proposed a method called content boosted collaborative filtering (CBCF). The algorithm is divided into two stages, the First being the content-based filtering that improves the existing trainee case rating data and the second being collaborative filtering that provides the final predictions.

PROPOSED WORK

The system uses advanced java technology along with machine learning concepts. This system uses three-tier architecture. The system uses all the existing algorithms i.e. This system has added the positive features of existing systems and has overcome the drawbacks of existing systems. content-based, context-based, and collaborative-based algorithms. The proposed system is a better system than any other existing system. The web service layer provides the android user to rate movies, view similar recommendations given by the system, and comment on them. The system is built on the Windows 2007 operating system. All these algorithms are combined to give more precise results. MySQL is used for storing data.

1. ADMIN

The system admin will be able to add a movie in a database, view movies, and update it. Simultaneously, the admin will be able to create chat rooms with the chatbot and add or remove members to it.

2. RECOMMENDATION ENGINE

This recommendation engine will calculate the similarities between the different users as well as between different movies. Based on the similarities calculated, this engine will recommend the movie to a user.

3. MOVIE WEB SERVICE

This will allow users to view movies on their official website and rate movie or comments on movies. This service will also show the movie recommendation to the users.

4. END-USER

The end-user can rate a movie, can comment on any movie, and can see similar movies recommended by other users who are similar to this user. The user can also ask questions to the chatbot such as movie genre, overview, budget, revenue, rating, voters, IMDB or TMDB links, adult content, etc.

SYSTEM ARCHITECT

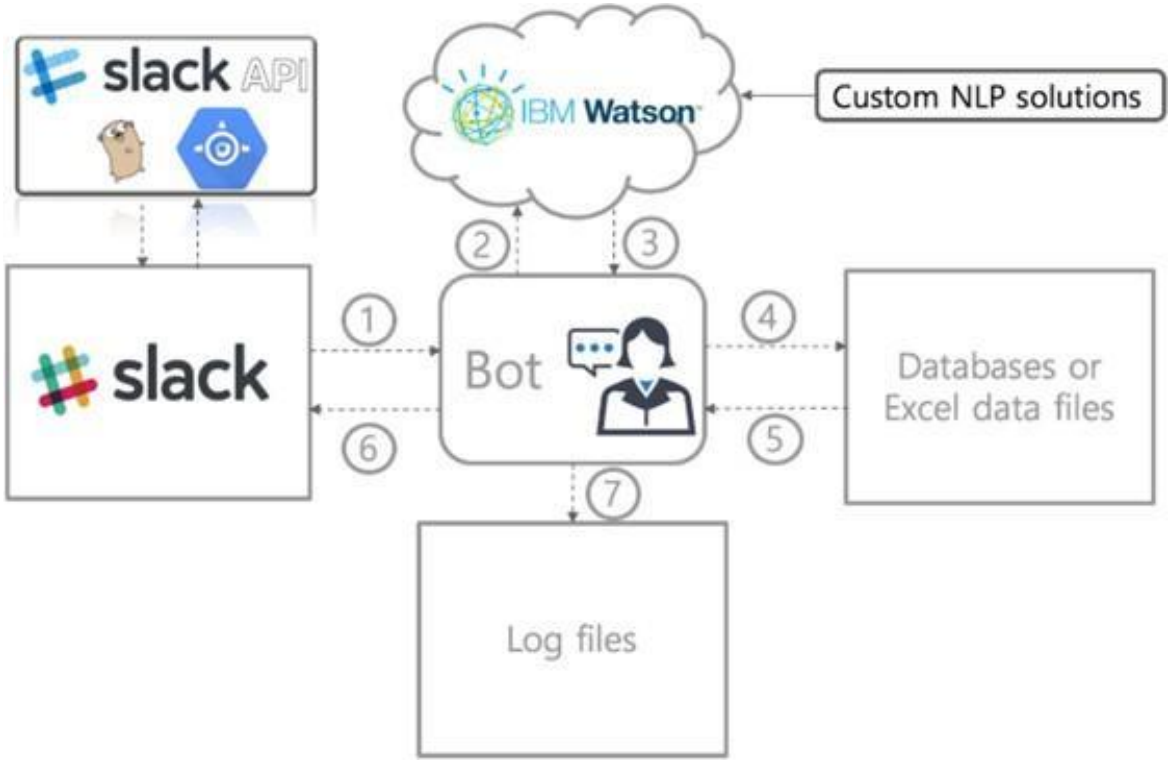

Fig 3.1: Block Diagram

IBM Watson Assistant is used for the smooth processing of NLP solutions and handling components like intents, dialogues, and entities. You can directly talk to the Slack Chabot along with your chatbot name, ask about the movies, genre, etc. illustrates the Block Diagram, which represents the design of our system. The movie recommending chatbot has two parts the frontend and the backend the frontend is using the Application Program Interface provided by the Slack

technologies, the slack is a collaborative platform where groups can collaborate and discuss various projects they are working on, the best feature of the Slack application program interface is that it provides a domain which is hosted on the slack platform servers and create a separate workspace for each, with this excellent feature you can access your slack platform from anywhere and anytime because your workspace is hosted and you are individually assigned a domain which can be accessed on the slack website or any personal computer application or a mobile application.

RESULT AND CONCLUSION

A recommendation system has been implemented based on a hybrid approach of collaborative filtering engine and context-based engine. We have tried to combine the existing algorithms for recommendation to come up with a hybrid one. It improves performance by overcoming the drawbacks of traditional recommendation systems. It describes the conventional Content, Collaborative Filtering, and Context Filtering recommendation approaches along with their precision, recall, and accuracy parameters. This paper has presented a number of utilized evaluation metrics, from which some were used to measure quality,

while others to measure performance. Recommender systems make the selection process easier for the users. The hybrid recommendation engine is a competent system to recommend Movies for e-users, whereas the other recommender algorithms are quite slow with inaccuracies. This recommender system will assuredly be a great web application, which can be clubbed with today's high demanding online purchasing websites. Our approach can be extended to various domains to recommend books, music, etc.

REFERENCE

[1] James Bennett, Stan Lanning; "The Netflix Prize", In KDD Cup and Workshop in conjunction with KDD, 2007.

[2] Mohammad Yahya H. Al-Shamir, Kamal K. Bharadwaj; "A Compact User Model for Hybrid Movie Recommender System " in International Conference on Computational

[3] Costin-Gabriel Chiru, Vladimir-Nicolae Dinu, Ctlina Preda, Matei Macri; "Movie Recommender System Using the User's Psychological Profile" in IEEE International

Conference on ICCP, 2015.

[4] Christina Christakou, Leonidas Lefakis, Spyros Vrettos, and Andreas Stafylopatis; “A Movie Recommender System Based on Semi-supervised Clustering ”, IEEE Computer Society Washington,

DC, USA 2015.

[5] Luis M. de Campos, Juan M. Fernández-Luna *, Juan F. Huete, Miguel A. Rueda-Morales;

“Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian

networks”, International Journal of Approximate Reasoning, revised 2010.

[6] Urszula Kużelewska; “Clustering Algorithms in Hybrid Recommender System on MovieLens Data”,

Studies in Logic, Grammar, and Rhetoric, 2014.

[7] Dietmar Jannach, Gerhard Friedrich; “Tutorial: Recommender Systems”, International Joint

Conference on Artificial Intelligence, Beijing, August 4, 2013.

[8] Gaurangi, Eyrun, Nan; “MovieGEN: A Movie Recommendation System”, UCSB.