# Movie Recommendation System

Design:

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 backend is IBM Watson Assistant used for smooth processing of NLP solutions and handling components like intents, dialogues and entities. Dialogs are back and forth communication between bot and user.
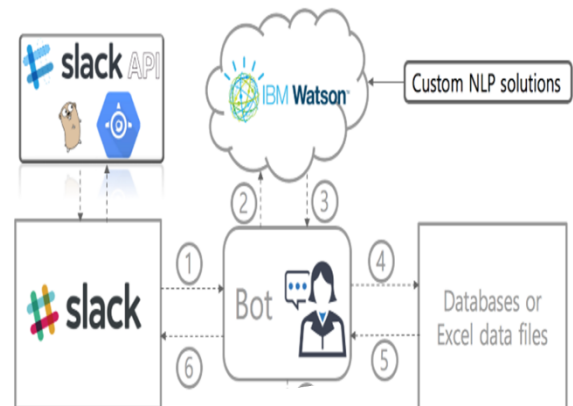
In my project of retrieval-based approach my bot takes the input as the movie name and recommends similar movies based on the user input. The user selects one of the options from recommend movies. The user can ask any info about the movie and this info is fetched from the dataset. The dataset is preprocessed by removing stop words, tf-idf results and cosine similarity to find the relevance between the available terms and the searched term. A category that defines users goal or purpose is intent. In this project I have created intents and given multiple examples for each of the intents and in similar way I have created entities and given entity values and synonyms for each of the entity. A dialog is a branching conversation flow that defines responses to the defined intents and entities. I have used dialog builder in the tool to create conversations with users to provide responses based on intents and entity.

My project works on single intent and not multiple intents. For example, "I want to check movie rating and movie genre," the assistant detects the two intents, but it can only trigger one node using the intent that has the highest confidence score and my chatbot responds to one intent. Hence it supports one intent. MovieBot is unique because it provides all the information about the particular movie to the user and it also links to IMDB website where the movie can be found, and it doesn't deviate from the main idea of the chatbot and it asks to rephrase if it doesn't understand anything but doesn't give wrong response. It is easy extend this to different applications. My chatbot responds on time and I have used read delay to be 1.

A movie recommendation system has been implemented based on hybrid approach of collaborative filtering engine and context-based engine. It improves the performance by overcoming the drawbacks of traditional recommendation systems. This approach can be extended to various domains to recommend books, music, etc by changing the dataset, intents and entities of interest.

Acknowledgement:

IBM Watson
https://www.ibm.com/watson/

Slack
https://slack.com

Kaggle- Rounak Banik
https://www.kaggle.com/rounakbanik/movie-recommender-systems/data -

Watson Assistant-
https://developer.ibm.com/technologies/artificialintelligence/articles/introduction-watson-assistant/

Kaushik
https://github.com/kaushkjadhav01/Movie-Recommendation-Chatbot/tree/master/data

Germanattanasio
https://github.com/watson-developer-cloud/botkit-middleware