

MOVIE TICKETING BOT

1. Introduction

1.1. Overview

1.2 Purpose

2. Literature Survey

2.1 Existing Problem

2.2 Proposed Solution

3. Theoretical Analysis

3.1 Block Diagram

3.2 Hardware / Software Desingning

4. Experimental Investigations

5. Flowchart

6. Result

7. Advantages & Disadvantages

8. Applications

9. Conclusion

10. Future Scope

11. Bibilography

Appendix

A. Source Code

1. INTRODUCTION

Dr. Ashok Goel teaches a class called Knowledge-Based Artificial Intelligence every semester at Georgia Tech. It's a massive class of around 300 registered students. Understandably, being a teaching assistant (TA) for this course can be quite the challenge as 300 students post roughly 10,000 messages in the online forums. But during the spring semester of 2017, one particular TA was especially good at her job. She answered the students questions with excellent efficiency. She would even make herself available during late hours the night before a deadline. Naturally, the students loved her and gave her reviews.

You can probably guess how the story ends. At the end of the semester, most students were surprised to find out that the TA, named Jill Watson, was actually a chatbot. Dr. Goel and his team built Jill by tracking down all the questions that had ever been asked on the course's online forum (about 40,000 postings in all). They then trained Jill to answer these questions using IBM Watson. Jill wasn't very good at first, but she learned from her mistakes and was eventually giving answers with 97% certainty.

It's stories like this that sparked the interest of developers and entrepreneurs and made 2016 the year of chatbots. In this project, we will explore a few aspects of this topic. We'll first talk about what a chatbot is and why now is a good time for you to build one. We'll then share a few important considerations for creating a successful chatbot.

1.1 OVERVIEW

What is a chatbot?

A chatbot is a computer program that allows human to interact with technology using a variety of input methods such as voice, text, gesture and touch, 26/7 365.

A chatbot is a way to expose a business's service or data via a natural language interface. It's important to understand that as an interface, the chatbot is only as good as the underlying service or data. So if you're thinking about building a chatbot, first make sure your services and data are solid. How do you know if your services and data are in good shape? Imagine the service you're providing has a traditional web or mobile interface. Would that still be a useful service? If the answer is "no", then your service is not ready for a chatbot interface either.

If built well, chatbots can help your business cut costs and establish additional revenue streams. A virtual customer support agent can reduce headcount and exponentially scale your real-time customer support capabilities. A conversational commerce chatbot gives your business a whole new channel to engage with your customers via messaging platforms.

1.2 PURPOSE

Smartphones, wearables and the Internet of Things(IoT) have changed the technology landscape in recent years. As digital artefacts got smaller, the computing power inside has become greater.

But mobile apps and data-heavy activities don't go hand in hand. Wading through complicated menus isn't the fast and seamless user experience businesses need to deliver today.

With the increased innovations and use of technology in different domains, chatbots are a new trend for the companies. Chatbot are a piece of technology that replace the need for human to converse.

The report of a study suggests that in future customers will prefer using chatbots and virtual agents than talking to call center agents.

In addition, consumers are no longer content to be restricted by the communication methods chosen by an organization. They want to interface with technology across a wide number of channels.

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM

We always have to go out to buy movie tickets and also stand in line for so much time to buy movie tickets and even if we stand in line there is no guarantee we will get tickets, instead of that if we can use chatbots so that it will save our time and energy and we no need to go before the schedule time and also we can choose our own seat and show time .

2.2 PROPOSED SOLUTION

The movie ticketing bot will provide detailed information so that a customer can know about the movie and based on the information customer will book the ticket. The user can search for the movies which will be going to release in the future, so they will have an option to book the ticket in advance. The movie ticketing bot will make the user experience much better than the present system. The movie ticketing bot will provide a much better experience for booking of movie tickets. If seats are available, the customer can change the position of the seats while they will get an option to cancel of tickets. If we have any queries, or any problems during booking tickets the bot can support to the user and it will look into it and also it saves our time.

3. THEORITICAL ANALYSIS

There are a few reasons especially in the enterprise:

- The availability of NLP capabilities as discussed in the previous chapter, and particularly those in the cloud
- The proliferation of popular messaging platforms such as Slack and Facebook Messenger
- The push for natural language interfaces

The next several sections will elaborate more on each.

Natural Language Processing in the Cloud

The availability of natural language processing capabilities in the cloud has been the most potent force behind the rise of chatbots. NLP, specifically text classifiers and entity extractors, powers a few of the core functionalities inside a chatbot. The problem has been the difficulty in utilizing them for people outside the research community. Open source solutions have made these methods more accessible, but the arrival of cloud APIs, with a much superior user experience, has enabled many more enterprises to use this technology.

Proliferation of Messaging Platforms

Messaging apps have come to dominate our mobile app usage. Recent data shows that these messaging apps have surpassed social networks in monthly active users.

As more users spend time on messaging apps, companies are looking at ways to reach users through these channels. It turns out there is a large amount of contextual data buried in these messages. We make dinner plans, inquire about stores, and look to purchase goods. Companies are now looking to help users by embedding chatbots inside these message channels to answer questions or assist with various tasks.

Natural Language Interface

In most human-machine interactions, users translate their intentions into a series of keystrokes and button clicks. The machine then responds via pixels on a screen. Wouldn't it be nice to talk to computers the way we talk to each other? This desire for natural language interfaces has always been around. In the early days of search engines, people liked AskJeeves because it allowed its users to search the web using natural language. Now, the proliferation of devices such as the Amazon Echo has drawn developers toward the idea of a voice-controlled home. After all, home appliances are notorious for their clunky user interfaces, and to replace them with smart agents that we could talk to seems like a much better user experience.

Types of Bots:

Open / Closed Domain, Generative / Retrieval-based

Let's take a look at a typology of bots with the regard of their purpose and responses.



Figure1: Types of Chatbot

Retrieval-based model

As the name says it retrieves the answers/responses from a set of predefined responses and some kind of heuristic to pick an appropriate response based on the input and context. The heuristic could be as simple as a rule-based expression match,

or as complex as an ensemble of Machine Learning classifiers.

Pro's

- No grammatical or meaning less errors as we store the answers
- Works 100% well for the business problems and customer satisfaction and attention can be gained
- Super easy to build these models as we don't require huge data.

Con's

- These systems don't generate any new text, they just pick a response from a fixed set.
- A lot of hard coded rules have to be written so not much intelligent.

Generative models

These models don't rely on pre-defined responses. They generate new responses from scratch. Generative models are typically based on Machine Translation techniques, but instead of translating from one language to another, we "translate" from an input to an output (response). It uses sequence to sequence models for generating the text (we will implement these also in the next stories) (anyone could not explain better than this for the Generative Retrieved based models I took the exact to just to give the idea)

Pro's

- No need to worry about the predefined responses and the rules.

Con's

- Super difficult to implement these and the output may not be accurate (grammatical / meaning less errors may occur) Not applicable for the business problem (unless you are providing a service which may require text summarization techniques) #will explain Huge data is required to train these models.

Open Domain

Open domain is the place where the chat conversation can go anywhere, users can type/ask anything. There isn't necessarily have a well-defined goal or intention. The chatbot Mitsuku is the example for this. The convo can go into all kinds of directions. The infinite number of topics and the fact that a certain amount of world knowledge is

required to create reasonable responses makes this a hard problem.

Closed Domain

Closed domain is the place where you are solving a particular business problem (The business could be in any sector/industry) ex : Pizza bot, Bankingbot, Medical bot, CricketScore bot etc... closed domain bots focus on one particular sector or industry. so you can't ask questions like "how is the weather now??", "what is the score for IndVsPak match today?" when you dealing with a banking bot or pizza bot. similarly you can't ask pizza bot a banking query. if you ask , you will get a decent answer "I am sorry I don't understand". The closed domain bots have the limited functionalities/ services based on the business problem. Note: In this story I only focus on the closed domains bots and I hope you get a picture about the chatbot architecture. Now lets get continued with the chatbots.

A chat bot typically has 3 things in it

1. Intent (Intention of the query asked by the user).
2. Entities (Named entities in Query like , Location names, People names, date and etc...) #NamedEntityRecognition.
3. Action or Response (the result to throw back to the user).

Ex: what's the weather in Bengaluru tomorrow??

Here **Intent** → Weather check

Entities → Bengaluru (Location), tomorrow (Date)

Response → "The weather in {Location} {Date} is so and so"

The chat bot has always canned responses depending upon the problem/service you provide.

Note : NLP is hard at this moment. Computers started generating text with the help of deep learning recently so it can't produce a meaning full response so Chat bots always have canned responses (For User/Customer services) if you are not a programmer, there are a lot of chatbots frameworks where you can build a bot very easily without coding. Chatfuel, Manychat, FlowXO, Octane, Recime and etc....

There are dozes of frameworks out there you can use any to build a bot for your business.If you are a programmer,have a little experience with Machine learning and

wanna build a chat bot for your services, you can use the following tools.

3.1 BLOCK DIAGRAM

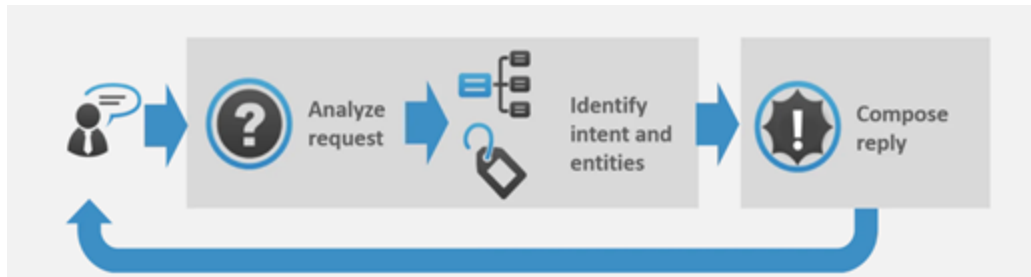


Figure2: High level Architecture of a Chatbot

There are two different tasks at the core of a chatbot:

- User request analysis
- Returning the response

User request analysis: This is the first task that a chatbot performs. It analyzes the user's request to identify the user intent and to extract relevant entities.



Figure3: Chatbot User request analysis.

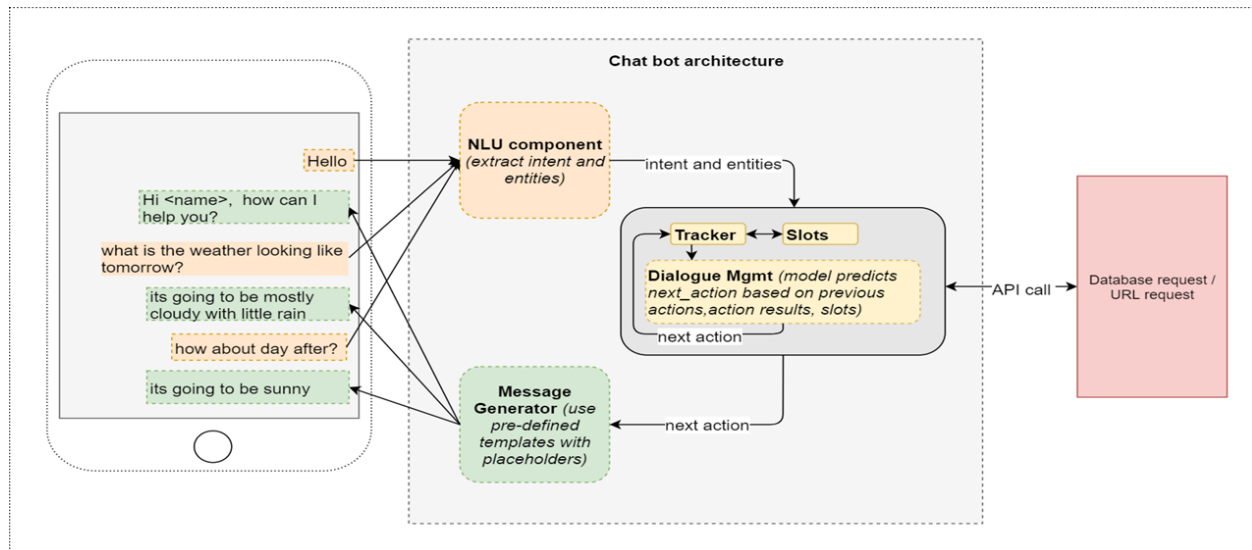
The ability to identify the user's intent and extract data and relevant entities contained in the user's request is the first condition and the most relevant step at the core of a chatbot: If you are not able to correctly understand the user's request, you won't be able to provide the correct answer.

Returning the response: Once the user's intent has been identified, the chatbot must provide the most appropriate response for the user's request.

The answer may be:

- A generic and predefined text
- A text retrieved from a knowledge base that contains different answers

- A contextualized piece of information based on data the user has provided
- Data stored in enterprise systems
- The result of an action that the chatbot performed by interacting with one or more backend application
- A disambiguating question that helps the chatbot to correctly understand the user's request



The above diagram shows the block diagram of chatbot. The explanation is following steps:

1. Know the user's intent – We will call these as intents.
Exs: 'request_weather', 'request_restaurant' etc.,
The intent in the above example is 'request_weather'.
2. Know the specific intents in the request (we will call them as entities) Ex: The answers to the questions like when?, where?, how many? etc., that correspond to extracting the information from the user request about date, time, location, number respectively. Here datetime, location, number are the entities. From the above weather example, the entities can be: 'datetime' (user provided information) and 'location' (note – location need not be an explicit input provided by the user and will be determined from the user location as default, if nothing is specified).
3. The intent and the entities together will help to make a corresponding API call to a weather service and retrieve the results.
4. The NLU component (Natural Language Understanding) helps in extracting the intent and entities from the user request.

5. Dialogue Management: Our model must be context aware and look back into the conversational history to predict the next_action.
6. The predicted value of the next_action can be something like –
 - Respond to the user with an appropriate message
 - Retrieve some data from a database (if we have any)
 - Make an API call and get some results matching the intent.
7. Message generator component consists of several user defined templates (templates are nothing but sentences with some placeholders, as appropriate) that map to the action names.
 - So depending on the action predicted by the dialogue manager, the respective template message is invoked.
 - If the template requires some placeholder values to be filled up, those values are also passed by the dialogue manager to the generator.
 - Then the appropriate message is displayed to the user and the bot goes into a wait mode listening for the user input.

3.2 HARDWARE / SOFTWARE DESIGNING

- Hardware Required: A computer or Laptop or Smart Phone
- Software Required: A Chatbot Software(IBM Watson Assistant(service))

4. EXPERIMENTAL INVESTIGATIONS

We need to research based on what people search like words or keywords , this can help us improve our bot and also the main purpose is to thoroughly investigate the character of human and no-human interaction process.

This chatbot will be used to avoid the difficulty users face for searching their favourite movie and will be user friendly and time saving.

To create a chatbot that gives the list of movie names and displays different show timings and to book tickets.

The main objective of movie ticketing bot is to develop an user friendly ,automated, accurate chatbot. We will create a chatbot that gives the list of movie names and

displays different show timings and to book tickets.

A chatbot is a software application used to conduct an on-line chat conversation via text or text-to-speech, in live of providing direct contact with a live human agent. It is a system implemented by many researchers to support various types of platforms. A chatterbot or chatbot aims to make a conversation between both human and machine. The machine has been embedded knowledge to identify the sentences and making a decision itself as a response to answer a question. The response principle is matching the input sentence from the user.

How to Build a Chatbot

A chatbot has a frontend and a backend. The frontend is the messaging channel where the chatbot interacts with the user. The backend is the application logic, the persistence stores, and the supporting services.

The Messaging Channel

There are a lot of messaging channels out there. You can leverage an existing one such as Slack, Facebook Messenger, or Kik. You can also build your own messaging layer such as a custom website or mobile app. Choosing the right channel depends on how you plan to engage your users. If you're a bank with a popular mobile app, you should expose your chatbot there. If you're a small business that has an active Facebook page, integrating your chatbot with Facebook Messenger is a good idea.

The Backend

Text classifiers

Let's first discuss and expand upon one NLP technology in particular—text classifiers. It was not until recently that text classifiers became easy to use and available on the cloud. They're an important part of chatbots. If you were to approach building a customer support chatbot from scratch, the task probably seems overwhelming. After all, even in a narrow domain such as customer support, there can still be an intractable number of different requests. How can you have an answer to every single one of them? The key insight is that though customer requests can be expressed in an almost infinite number of ways, the solution space is magnitudes smaller.

Here's a more specific example. When talking to a customer service agent, a customer might say any of the following:

- "How come I can't log into my account anymore?"
- "I forgot my password."
- "It says my password is incorrect."
- "I'm locked out of my account."

Luckily, all these requests have the same solution, which is to reset the customer password. So the core functionality of a chatbot is to map all possible user inputs into a much smaller set of responses. This type of pattern recognition is what text classifiers do best.

Dr. Goel reached the same conclusion when building Jill Watson: "One of the secrets of online classes is that the number of questions increases if you have more students, but the number of different questions doesn't really go up. Students tend to ask the same questions over and over again."

Using a framework versus building your own

The first significant decision in building a chatbot backend is to decide whether or not to leverage an existing framework. There are pros and cons to using one, and if chosen correctly, a framework can provide a large part of the solution with little effort. But this also means giving up control to the framework itself. It's straightforward to build a chatbot within it, but impossible to customize or integrate the chatbot in a way that's outside the design of the framework.

There are many chatbot frameworks available, including API.ai, Wit.ai, Microsoft Bot Framework, and IBM Watson Conversation. The next few sections will use IBM Watson Conversation as an example to help illustrate the concepts in leveraging an existing framework. Though it may appear product-specific, the ideas are translatable to other chatbot frameworks as well as building chatbots in general.

Anatomy of a chatbot backend

If you do decide to choose a framework, the backend of your chatbot will most likely consist of three main parts: intents, entities, and dialog. These can then be integrated with one or more messaging channels. Extra features such as sentiment analysis,

human intervention, or personality can be added as well.

Intent: You usually start building a chatbot with intents. An intent is the purpose of a user's input. This can be a question about your business hours or a complaint about the registration process. The response of your chatbot to this intent is entirely up to you. It might be a paragraph that answers the question or an action such as starting the password reset process. Another way to think about intents is that they're the verbs for your chatbots to action. They dictate what your chatbots will do next.

To train your chatbot to recognize an intent, first determine the action you'd like to map to this intent—for example, provide information on business hours. Then supply the framework with examples of user inputs that would require this action. The business hours scenario would include example inputs such as the following:

"What time are you open?" "Are you open on weekends?" "I can't find your hours of operation." Usually, a minimum of five inputs is needed, but more is better.

Remember, it might be tempting to make up these examples, but it is always better to draw these from past data. The more similar these examples are to real-world user requests, the better your chatbot will perform. With Watson Conversation, an intent (really a text classifier) is built from these examples so the service will recognize similar inputs in the future, even if they're not exact matches.

Let's see this in action. The most common intent for a chatbot is to greet the user. When a user first comes in contact with your chatbot, it needs to introduce itself. This is an excellent opportunity to establish the context of the chatbot, state its purpose, and set boundaries. **Figure 4** shows an example of some common greetings that a developer might deploy using Watson Conversation's Intents tab.

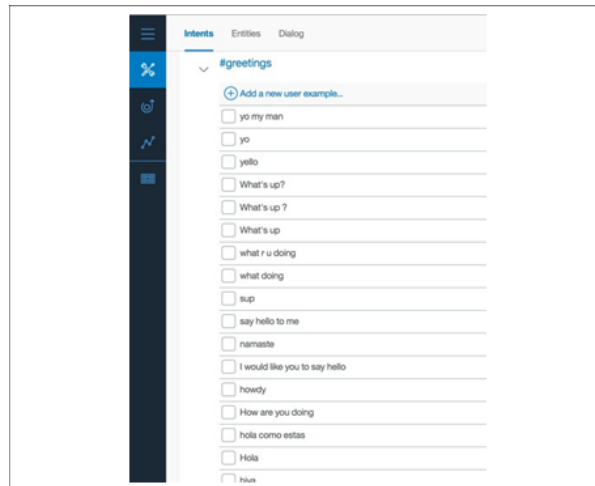


Figure 4: Some example greetings your chatbot might use

Entities: If intents are the verbs for a chatbot to act on, then entities are the nouns. They're the keywords in a user's input. If a particular word differentiates one input from another, it probably should be an entity.

For example, if a user wants to find the available movies of a Movie ticketing bot, the available of the movies would be one of the entities. Specifically, if a user asks, "What are the movies are available?" provide Movie Availability would be the intent and Movie would be the entity. Figure 5 shows an example of a Movie entity in Watson Conversation's Entities tab.

Entity (9) ↑	Values	Modified ↑
<input type="checkbox"/> @Book_Tickets	Movie, Seats, Time	8 days ago
<input type="checkbox"/> @City	kakinada, visakhapatnam, Vizayanagaram	8 days ago
<input type="checkbox"/> @Greetings	Good Afternoon, Good Evening, hello, hi, Good Morning	8 days ago
<input type="checkbox"/> @Inquiry	Offers, Movie_list	8 days ago
<input type="checkbox"/> @Movies	Sarileru Nekevaru Sarileru Neekevaru, Bheeshma, Ala Vaikunthapurramuloo, V ...	8 days ago
<input type="checkbox"/> @Movies-availability	City, Suggestions, Streaming	8 days ago
<input type="checkbox"/> @Payment	Credit card, UPI, Debit card	8 days ago
<input type="checkbox"/> @Thanks	Thank u, Thank you	a day ago
<input type="checkbox"/> @Time	9AM, 2PM, 6PM	8 days ago

Figure 5. An example of a Movie entity, which encompasses a list of different movies that you'd like your chatbot to recognize

Dialog: Dialog is the conversation flow. It's usually represented as a directed graph where each node represents one exchange in the conversation. Together, it is the combined structure of all your possible conversations. Since this can become quite complex, most chatbot platforms provide a UI to help you visualize the process. Figure 6 shows an example in Watson Conversation's Dialog tab.

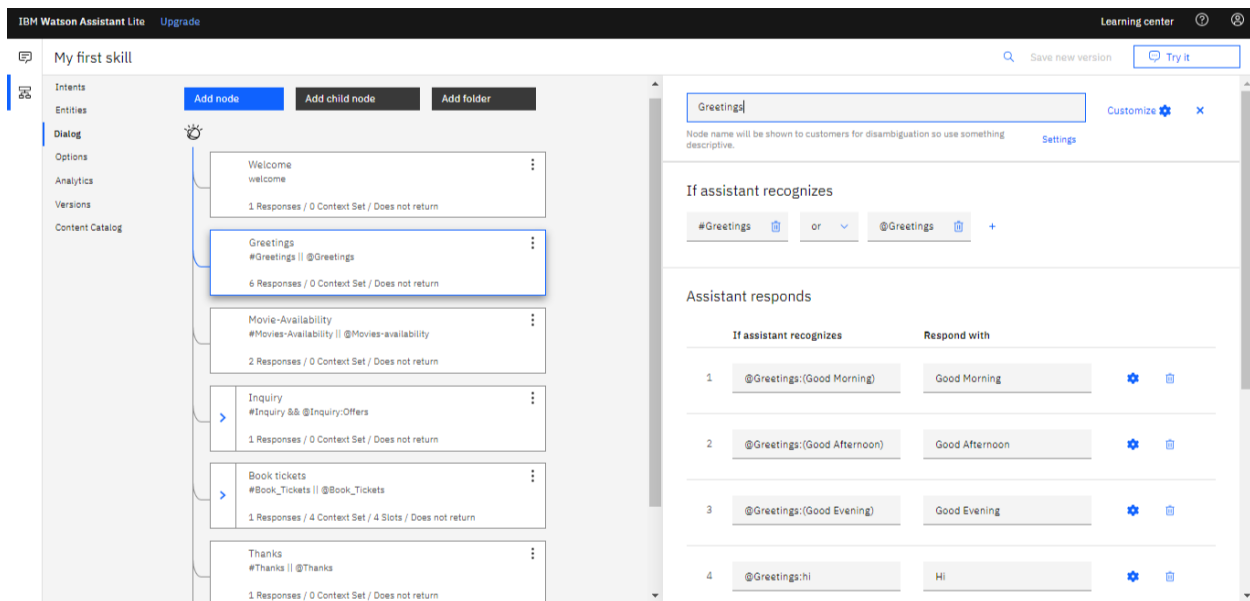


Figure 6. An example of a dialog tree in Watson Conversation

The screenshot in Figure 6 shows an example of a dialog tree, starting with the welcome node after that blue Greeting node at the 2nd top left. The screen on the right shows the detailed view of this node when it's selected. In this case, the condition that triggers this node is the welcome condition, which the system understands to be the start of the conversation. You can then dictate the response of the chatbot via the "Respond with" section. Here, the chatbot introduces itself as the Watson Movie ticketing bot.

Context variable: A context variable contains the information shared between the framework and your application. It's the way to exchange information between your business logic and the framework. Watson Conversation provides a context object that lets you store any key/value pair as context variables. In Figure 7, the context object delete all the variables related to the user's book movie ticket order. The variables are updated as the user progresses through the ordering process. If the chatbot recognizes

the Deleting context intent, it will set all Book ticket parameters to empty so the user details deleted after the book movie ticket.

Figure 7. Arriving at the Book tickets node leads the chatbot to tell the user “Movie ticket details” and deletes all context variables.

Human in the loop: Embedding a human into your chatbot has two significant benefits.

First, humans are an excellent way to bootstrap your chatbot before it gathers enough data to operate autonomously. If the chatbot is not confident about its response, it can ask the human for approvals or edits before sending it to the end user. Another setup is to have the chatbot provide multiple responses, and have the human choose the most appropriate one.

The second benefit is that humans are the best way to prevent your chatbot from utterly failing your users. There are a few ways to detect if the conversation needs to be routed to a human:

- The chatbot doesn’t understand the user’s input—this usually means the user input doesn’t match any of your established intents.
- The conversation is taking too long, or a circular pattern is detected.
- Negative sentiment is caught in the user’s input.
- The user directly asks to talk to a real person.

Analytics and metrics

One of the best ways to improve chatbot performance is to monitor user interactions through chat logs. This is especially important as you scale up your operation to simultaneously serve multiple customers. The more users you have, the larger the chat logs. Being able to examine the logs effectively will help you monitor your chatbot performance, and deliver a better user experience.

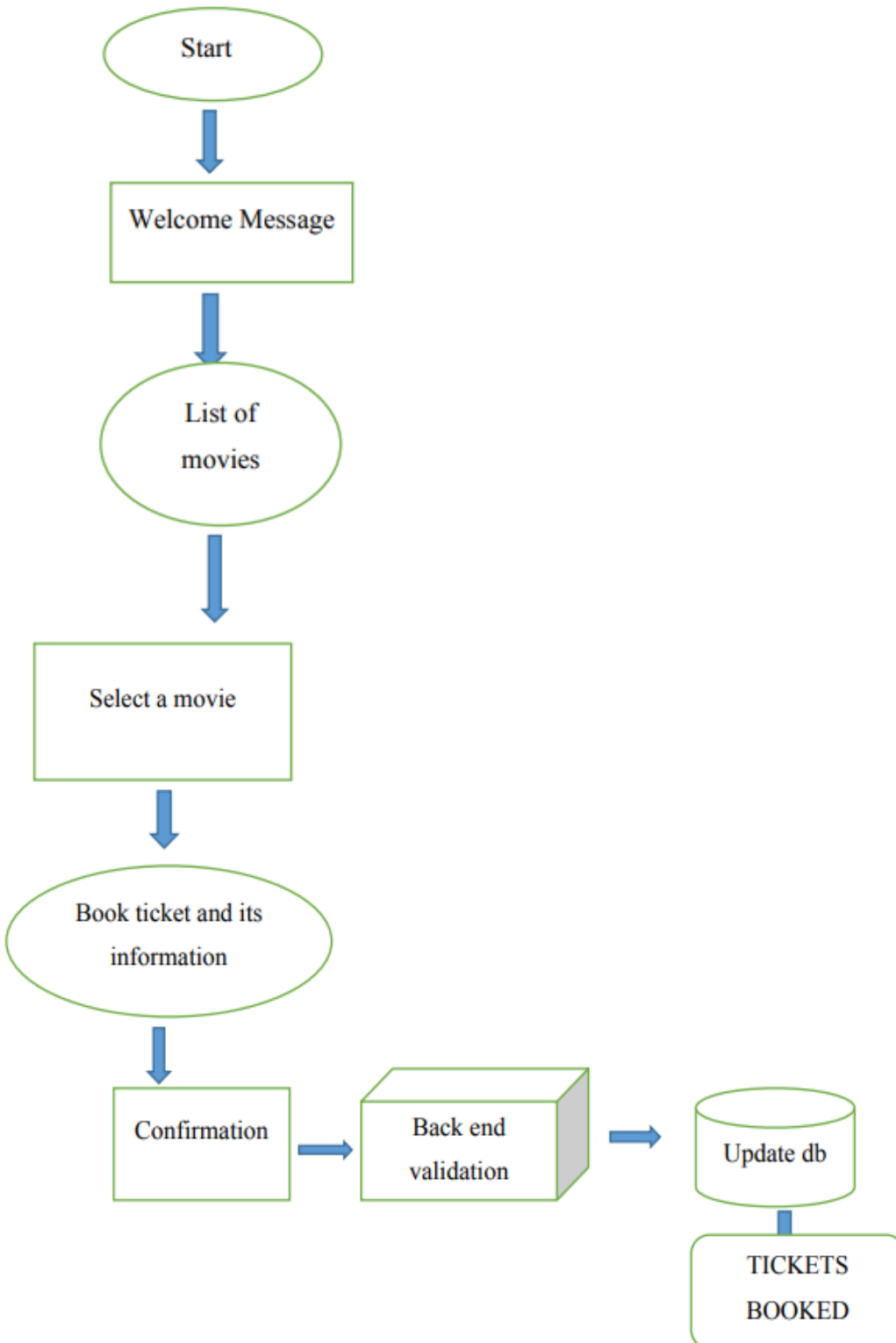
Typically, analytics for chatbots include general statistics such as average length of conversation as well as a complete history of all messages. If you're building your own analytics backend, make sure you provide these along with the ability to open a specific conversation to go back to the thread to better understand the original context.

Challenges of Building a Successful Chatbot

There are many things to get right for a chatbot to be useful. One of the most important is to define the project scope correctly. It needs to be broad enough for the chatbot to be helpful, yet narrow enough so that you're not wasting time building artificial general intelligence. Specifically, this means capturing as many user requests as possible, yet still being able to reconcile the nuanced differences between each one.

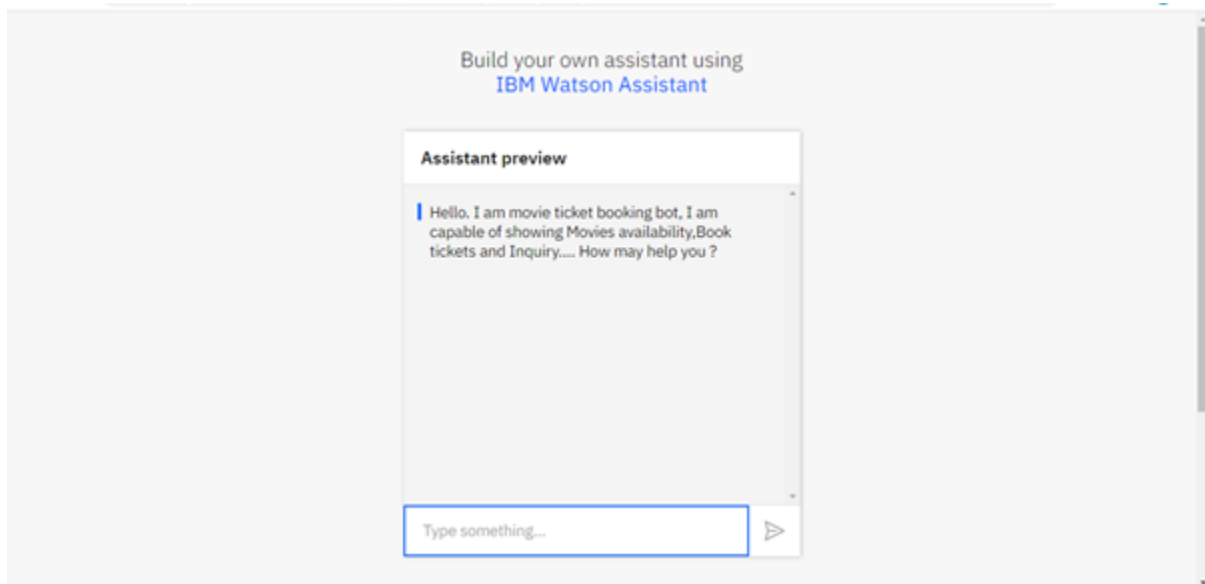
This is not an easy problem. For example, one travel agency tried to deploy a vacation planning chatbot. A critical component was a vocabulary base large enough to recognize all the destinations, along with its colloquial variations. It turns out there were over 10 ways people could refer to the Cayman Islands, even assuming all spellings were correct. It took the company months to build a list that could confidently capture all the variations for this one destination.

5. FLOW CHART



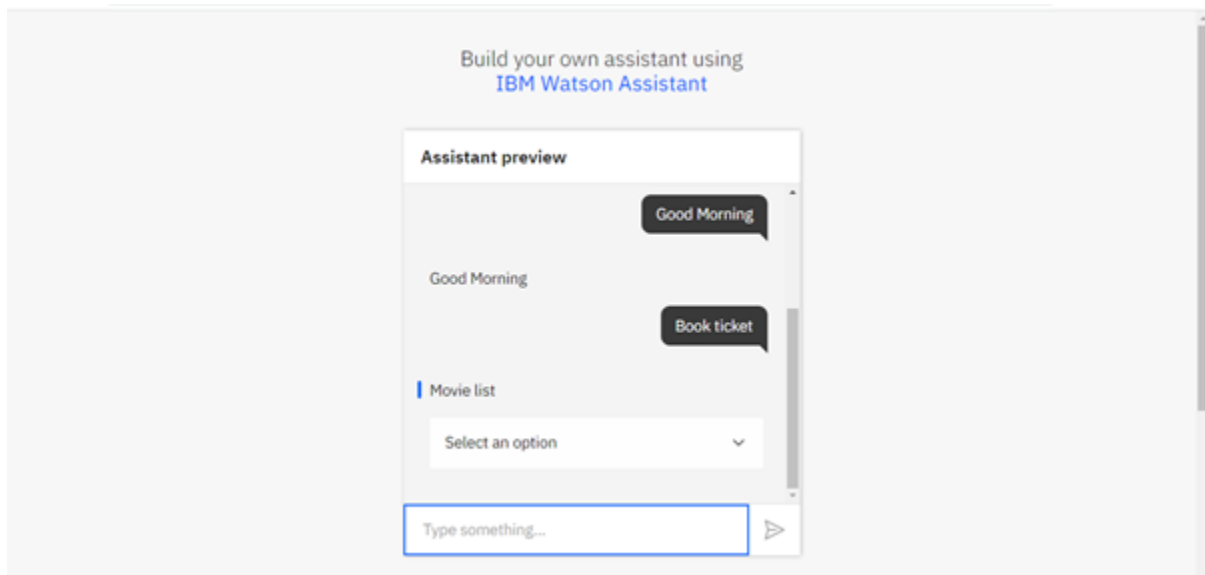
6. RESULT

The below screens are results of Movie ticketing bot
Step1: **welcome** message of Movie ticketing bot.



Step2: Greetings-User give good morning to bot then bot responses Good Morning to user.

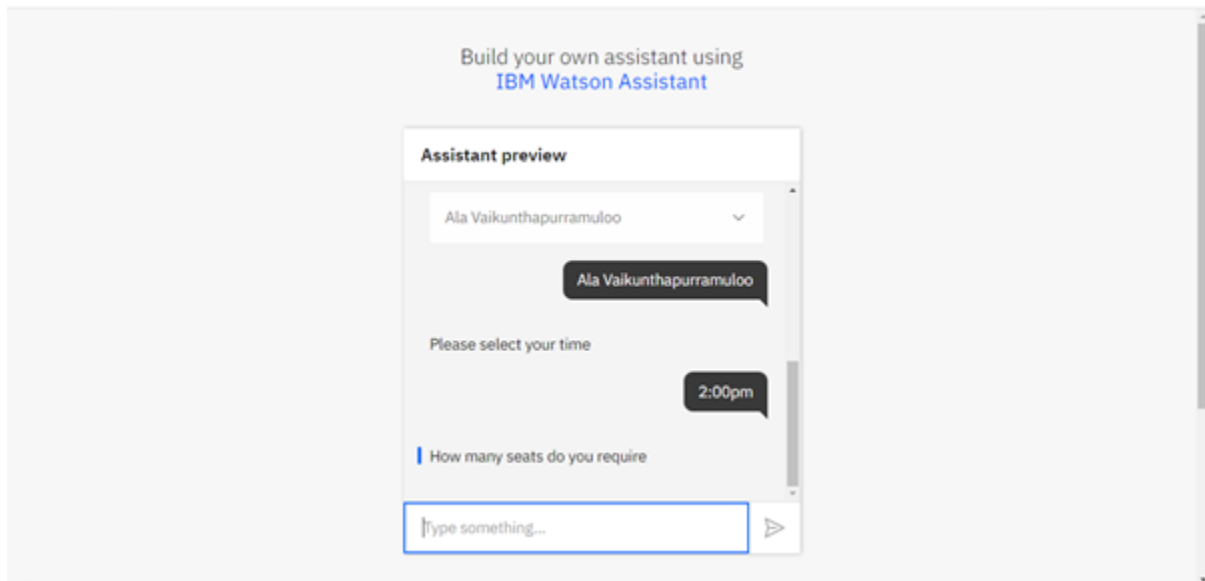
Step3: Book ticket-User ask to bot book a ticket then bot display the list of movies options



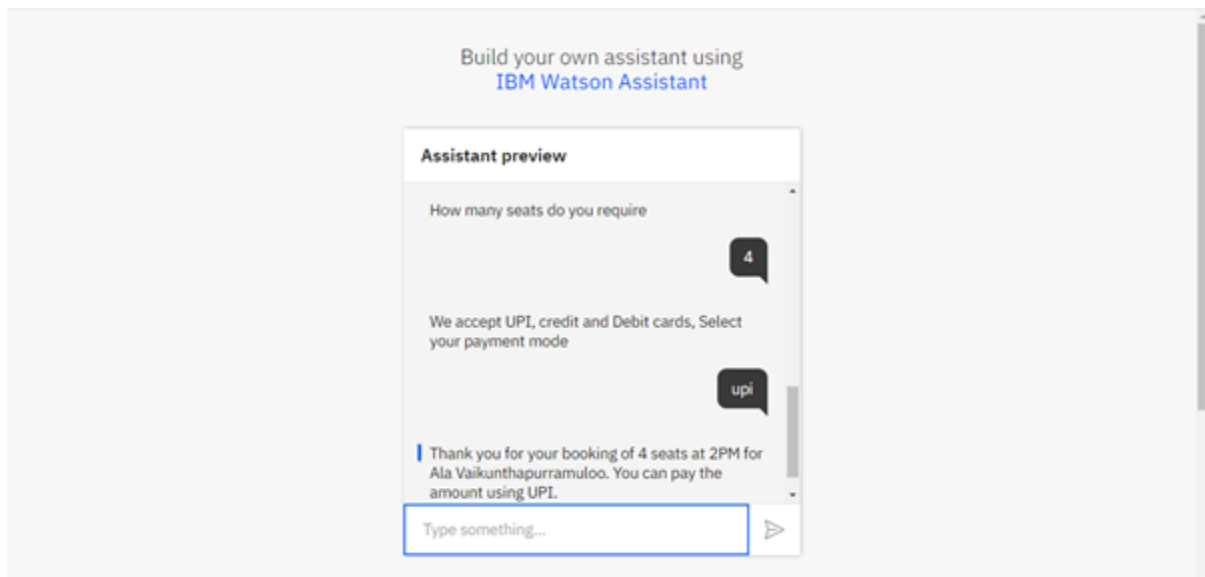
Step4: Time slot-After user selecting the movie then bot asks "please select your time".

Step 5: Number of seats- Bot ask to user "How many seats do you requier " then user

selects the number of seats.



Step 6: Payment - Bot ask to user types of payment modes then user selects the payment mode. Bot give the details of movie ticket .



7. ADVANTAGES & DISADVANTAGES

Advantages:

- One of the most important is to define the project scope correctly.
- The bot helps the user to simply choose some options and book movie tickets.
- The bot does not display ads due to which the user does not get annoyed.

- The bot is always up-to-date so that users do not get confused or have to try again.
- Available for 24 hours a day, seven days a week.
- We can book easily and in less time according to our choice.
- Multi customer handling.

Disadvantages:

Typically, analytics for chatbots include general statistics such as average length of conversation as well as a complete history of all messages. If you're building your own analytics backend, make sure you provide these along with the ability to open a specific conversation to go back to the thread to better understand the original context.

- Time consuming.
- Poor memory.
- Inability to understand.
- Lack emotions
- Made to handle first level questions
- Require maintenance

8. APPLICATIONS

Banking, Financial Services & Insurance Chatbot

Simplify Operations

Guide customers into performing a variety of financial operations in a conversational way and with complete safety. From checking an account, reporting lost cards or making payments, to renewing a policy or managing a refund, the customer can manage simple tasks autonomously.

Guaranteed Customer Support

Provide immediate support to existing customers and prospects through a chatbot capable of addressing all queries in real time. With each conversation the chatbot learns more about customers, delivering a proactive and personalized service.

Internal Training and Support

Help provide adequate support to employees by facilitating the most complex and

time-consuming back-office operations, such as managing internal documentation or reviewing agreements, as well as providing the necessary training to new staff members.

Automotive Chatbot

Increase Customer Engagement

Engage prospects with fast, humanlike interactions to significantly increase conversion rates and provide a solid pipeline of highly qualified leads to dealerships.

Car Discovery

Guide customers into choosing the vehicle that best fits both needs and budget, in a conversational style. Using the information gleaned from talking to the customer, the chatbot can help configure a car, and even schedule a test drive at the nearest dealer.

Connected Vehicles

Create a conversation that goes beyond the boundaries of the vehicle to interact with other services, such as charging stations or road-side assisting. Customers can talk to their in-car systems over any channel available.

Retail & Ecommerce Chatbot

Improve Customer Experience

Address all clients' queries and requests, whether it's pre-purchase information or updates on shipping, over any channel they choose, in a conversational and humanlike way.

Enhance Shopping Journey

Boost conversion and revenue by assisting the customers' journey in an online store by offering personalized shopping advice. For example, a chatbot can help navigate through different categories, find specific products, make suggestions about the right size and even place the order.

Personalize Marketing Communication

Collect and analyze information generated by the conversations the chatbot has every day to better understand the customers' needs and preferences. This conversational data can be used to anticipate users' behavior and place customized offers or

marketing messages at the right time.

Telecom Chatbot

Resolve Technical Issues

For customers searching through self-help FAQs and knowledge forums to find an answer to a question, the frustration is palpable. With a conversational chatbot, customers can resolve technical issues, find out the latest upgrade deal and even change their address at a simple request.

Increase Sales and Acquisition

Use a chatbot to boost cross-selling among existing customers, offering personalized plans and services based on purchase history or user profile. At the same time, chatbots can assist potential customers in choosing the right product for their needs.

Improve Workforce Productivity

Allow employees to focus on more complex tasks while a chatbot handles repetitive or time-consuming activities, like retrieving information about plans and additional services available to come up with the best fit for an interested user.

Energy & Utilities Chatbot

Streamline Customer Support

Give customers the effortless experience they want by removing the frustration caused by call center queues, endless online menus or outdated FAQs. A chatbot can fill out forms, deliver technical advice, process billing queries, and even recommend better tariffs.

Customer Retention

Ensure customer retention and strengthen relationships by offering proactive information about users plans, usage or habits, and include suggestions on how to save on consumption.

Manage Field Operations

Manage appointments between customers and technical staff in order to simplify field operations and optimize installation and maintenance processes.

Media & Entertainment Chatbot

Transform the Gaming Experience

Combine a conversational chatbot with other forms of AR or VR technology to offer an immersive experience that will transform any gaming experience, whether it's an online gambling site that delivers the whole casino experience, or a role-playing game that allows the player to converse with non-playing characters in a totally natural way.

Unique Targeted Content

By analyzing a user's past behavior, chatbots can learn about preferences and suggest new and targeted pieces of content users would love to consume – and in a conversational way, taking the entertainment experience to a new level.

Boost Conversion

Increase the amount of monetization opportunities, like subscriptions, plan upgrades and other content promotions, with the support of an intelligent chatbot that can handle the whole sales process, from discovery to final purchase.

Smart Homes & IOT Chatbot

Connected Home Experience

Enable customers to interact and control any smart- home connected device and appliance (like thermostats, switches or smart fridges), using the power of everyday speech and language.

Interact with Smart Vehicles

Improve the driving experience, from the moment a customer accesses the vehicle until he reaches the final destination. From unlocking the car, setting the desired temperature, to planning routes that avoid busy roads and ensuring the safety of the drivers and passengers alike.

Travel & Hospitality Chatbot

Make Recommendations

By asking simple questions, the chatbot can figure out what the user is looking for and

make recommendations based on preferences, like budget restrictions and destination types. The chatbot can also include suggestions on other related services, like car rentals or travel insurance.

Enhance Loyalty

Take advantage of the customer data gathered during endless interactions to deliver personalized offers, upgrades or add-on extras, that will help increase engagement and drive brand loyalty.

Provide Customer Care

Provide immediate support to customers during crucial situations, for example if they need to re-book a missed flight or change a hotel reservation, wherever they are and on whatever device or service they choose to communicate on.

9. CONCLUSION

Chatbots or smart assistants with artificial intelligence are dramatically changing businesses. There is a wide range of chatbot building platforms that are available for various enterprises, such as e-commerce, retail, banking, leisure, travel, healthcare, and so on.

Remember, it's easy to get started with chatbots, but it takes patience and hard work to create a truly successful one. Define the correct scope for your chatbot and keep in mind that context is vital, as the medium has much less bandwidth for communication. Building a chatbot is much like training a new hire—they probably only know a little on their first day, but through coaching and supervision they'll eventually become a productive employee. It's essential for your users to see your chatbot as a human, but it's equally crucial for you to do the same.

Chatbots have been around for a long time. Twenty years ago, they would only look for a couple of words from a very highly restricted set of commands: "Is this correct? Type yes or no." But as you can see from the Georgia Tech story, the chatbots of today look entirely different.

10. FUTURE SCOPE

As bots become more useful, they will become more like virtual assistants representing the company, becoming the first port of call. As they can handle, sales, shipping queries,

customer support and marketing or promotions like upselling and seasonal reminders, the bot will play key role in any business. As we know, when a user faces any problem or wants to contact the customer care or finds a bug or is properly not able to use a website or app, the bot is the only thing that resolves all the problems faced by a user. Apart from solving problems, the bot can easily handle many more things like marketing, business, promotions, etc. Bots act as virtual assistance for a website or for an app and also for a company. In the future, bots will be able to play an important role in almost every digital platform and the usage of boats will be more than compared to today.

11. BIBILOGRAPHY

1. Tom Markiewicz& Josh Zheng, Getting started with Artificial Intelligence, Published by O'Reilly Media,2017
2. Stuart J. Russell and Peter Norvig, Artificial Intelligence A Modern Approach.
3. AurélienGéron, Hands on Machine Learning with Scikit-Learn and TensorFlow [Concepts, Tools, and Techniques to Build Intelligent Systems], Published by O'Reilly Media,2017
4. Build an AI Assistant with Wolfram Alpha and Wikipedia in Python.
<https://medium.com/@salisuwy/build-an-ai-assistant-with-wolfram-alpha-and-wikipedia-in-python-d9bc8ac838fe>
5. Chatbot Architecture Tutorial
<https://towardsdatascience.com/chatbot-tutorial-choosing-the-right-chatbot-architecture-5539c8489def>
6. How do Chatbots work? A Guide to the Chatbot Architecture
<https://marutitech.com/chatbots-work-guide-chatbot-architecture/>
7. Conversational AI chat-bot – Architecture overview
[https://towardsdatascience.com/architecture-overview-of-a-conversational-ai-chat-bot-4ef3dfefd52e\)\`wewssw](https://towardsdatascience.com/architecture-overview-of-a-conversational-ai-chat-bot-4ef3dfefd52e)\`wewssw)

APPENDIX

A. SOURCE CODE

```
{  
  "intents": [  
    {  
      "intent": "Book_Tickets",
```

```
"examples": [  
  {  
    "text": "I prefer to reserve two seats"  
  },  
  {  
    "text": "I would like to book a ticket at 2:00 PM"  
  }  
],  
"description": "",  
},  
{  
  "intent": "Greetings",  
  "examples": [  
    {  
      "text": "Good Afternoon"  
    },  
    {  
      "text": "Good Evening"  
    },  
    {  
      "text": "Good Morning"  
    },  
    {  
      "text": "Hello"  
    },  
    {  
      "text": "Hi"  
    }  
  ],  
  "description": "Normal Greetings "  
},  
{  
  "intent": "Inquiry",  
  "examples": [  
    {  
      "text": "Any Available offers"  
    }  
  ]  
}
```

```
],
  "description": ""
},
{
  "intent": "Movies-Availability",
  "examples": [
    {
      "text": "can you provide the Available Movies ?"
    },
    {
      "text": "Please suggest some movies"
    },
    {
      "text": "Which movies are available"
    }
  ],
  "description": ""
},
{
  "intent": "Thanks",
  "examples": [
    {
      "text": "Ok thank you"
    },
    {
      "text": "Thank u"
    },
    {
      "text": "Thank You"
    }
  ],
  "description": ""
},
{
  "entities": [
    {
      "entity": "Book_Tickets",
```

```
"values": [  
  {  
    "type": "synonyms",  
    "value": "Movie",  
    "synonyms": []  
  },  
  {  
    "type": "synonyms",  
    "value": "Seats",  
    "synonyms": [  
      "bearths",  
      "seat",  
      "ticket"  
    ]  
  },  
  {  
    "type": "synonyms",  
    "value": "Time",  
    "synonyms": []  
  }  
],  
"fuzzy_match": true  
},  
{  
  "entity": "City",  
  "values": [  
    {  
      "type": "synonyms",  
      "value": "kakinada",  
      "synonyms": []  
    },  
    {  
      "type": "synonyms",  
      "value": "visakhapatnam",  
      "synonyms": []  
    },  
  ]  
}
```

```
    "type": "synonyms",
    "value": "Vizayanagaram",
    "synonyms": []
  }
],
"fuzzy_match": true
},
{
  "entity": "Greetings",
  "values": [
    {
      "type": "synonyms",
      "value": "Good Afternoon",
      "synonyms": [
        "ga",
        "good aftrn",
        "gud aftn"
      ]
    },
    {
      "type": "synonyms",
      "value": "Good Evening",
      "synonyms": [
        "ge",
        "good Even",
        "Gud Even"
      ]
    },
    {
      "type": "synonyms",
      "value": "Good Morning",
      "synonyms": [
        "gm",
        "good morng",
        "gud mrg"
      ]
    }
  ],
}
```

```
{
  "type": "synonyms",
  "value": "hello",
  "synonyms": [
    "hellooo"
  ]
},
{
  "type": "synonyms",
  "value": "hi",
  "synonyms": [
    "hiii"
  ]
}
],
"fuzzy_match": true
},
{
  "entity": "Inquiry",
  "values": [
    {
      "type": "synonyms",
      "value": "Movie_list",
      "synonyms": [
        "List"
      ]
    },
    {
      "type": "synonyms",
      "value": "Offers",
      "synonyms": [
        "Offer"
      ]
    }
  ]
},
"fuzzy_match": true
},
```

```
{
  "entity": "Movies",
  "values": [
    {
      "type": "synonyms",
      "value": "Ala Vaikunthapurramuloo",
      "synonyms": []
    },
    {
      "type": "synonyms",
      "value": "Bheeshma",
      "synonyms": []
    },
    {
      "type": "synonyms",
      "value": "Jaanu",
      "synonyms": []
    },
    {
      "type": "synonyms",
      "value": "Sarileru Nekevvuru Sarileru Neekevvuru",
      "synonyms": []
    },
    {
      "type": "synonyms",
      "value": "V (film)",
      "synonyms": []
    }
  ],
  "fuzzy_match": true
},
{
  "entity": "Movies-availability",
  "values": [
    {
      "type": "synonyms",
      "value": "City",
```



```
"synonyms": [
  "area",
  "areas",
  "location",
  "region",
  "town"
]
},
{
  "type": "synonyms",
  "value": "Streaming",
  "synonyms": [
    "accessible",
    "broadcast",
    "streamed",
    "streaming",
    "viewable"
  ]
},
{
  "type": "synonyms",
  "value": "Suggestions",
  "synonyms": [
    "choice",
    "choices",
    "options",
    "preferences"
  ]
}
],
"fuzzy_match": true
},
{
  "entity": "Payment",
  "values": [
    {
      "type": "synonyms",
```

```
    "value": "Credit card",
    "synonyms": []
  },
  {
    "type": "synonyms",
    "value": "Debit card",
    "synonyms": []
  },
  {
    "type": "synonyms",
    "value": "UPI",
    "synonyms": []
  }
],
"fuzzy_match": true
},
{
  "entity": "sys-date",
  "values": [],
  "fuzzy_match": true
},
{
  "entity": "sys-number",
  "values": [],
  "fuzzy_match": true
},
{
  "entity": "sys-time",
  "values": [],
  "fuzzy_match": true
},
{
  "entity": "Thanks",
  "values": [
    {
      "type": "synonyms",
      "value": "Thank you",
```

```
    "synonyms": [
      "ok Thank you",
      "tq"
    ]
  }
],
"fuzzy_match": true
},
{
  "entity": "Time",
  "values": [
    {
      "type": "synonyms",
      "value": "2PM",
      "synonyms": [
        "2-00PM",
        "2:00 PM"
      ]
    },
    {
      "type": "synonyms",
      "value": "6PM",
      "synonyms": [
        "6-00PM",
        "6:00 PM"
      ]
    },
    {
      "type": "synonyms",
      "value": "9AM",
      "synonyms": [
        "9:00AM",
        "9-AM"
      ]
    }
  ],
  "fuzzy_match": true
}
```

```
}
],
"metadata": {
  "api_version": {
    "major_version": "v2",
    "minor_version": "2018-11-08"
  }
},
"dialog_nodes": [
  {
    "type": "standard",
    "title": "Anything else",
    "output": {
      "generic": [
        {
          "values": [
            {
              "text": "I didn't understand. You can try rephrasing."
            },
            {
              "text": "Can you reword your statement? I'm not understanding."
            },
            {
              "text": "I didn't get your meaning."
            }
          ]
        },
        "response_type": "text",
        "selection_policy": "sequential"
      ]
    }
  },
  {
    "conditions": "anything_else",
    "dialog_node": "Anything else",
    "previous_sibling": "node_10_1603127279080",
    "disambiguation_opt_out": true
  }
]
```

```
"type": "event_handler",
"output": {
  "text": {
    "values": [
      "We accept UPI, credit and Debit cards, Select your payment mode"
    ],
    "selection_policy": "sequential"
  }
},
"parent": "slot_5_1603125270357",
"event_name": "focus",
"dialog_node": "handler_10_1603125270377",
"previous_sibling": "handler_1_1603125270377"
},
{
  "type": "event_handler",
  "output": {
    "generic": [
      {
        "title": "Movie list",
        "options": [
          {
            "label": "Ala Vaikunthapurramuloo",
            "value": {
              "input": {
                "text": "Ala Vaikunthapurramuloo"
              }
            }
          }
        ],
      },
      {
        "label": "Bheeshma",
        "value": {
          "input": {
            "text": "Bheeshma"
          }
        }
      }
    ],
  },
}
```

```
{
  "label": "Sarileru Nekevaru Sarileru Neekevaru",
  "value": {
    "input": {
      "text": "Sarileru Nekevaru Sarileru Neekevaru"
    }
  }
},
{
  "label": "V (film)",
  "value": {
    "input": {
      "text": "V (film)"
    }
  }
},
{
  "label": "Jaanu",
  "value": {
    "input": {
      "text": ""
    }
  }
}
],
"response_type": "option"
}
],
},
"parent": "slot_4_1603124612331",
"event_name": "focus",
"dialog_node": "handler_1_1603124612400",
"previous_sibling": "handler_7_1603124612400"
},
{
  "type": "event_handler",
  "output": {},
}
```

```
"parent": "slot_5_1603125270357",
"context": {
  "Payment": "@Payment"
},
"conditions": "@Payment",
"event_name": "input",
"dialog_node": "handler_1_1603125270377"
},
{
  "type": "event_handler",
  "parent": "node_8_1603124489383",
  "event_name": "focus",
  "dialog_node": "handler_2_1603124610462",
  "previous_sibling": "node_3_1603125739441"
},
{
  "type": "event_handler",
  "output": {
    "text": {
      "values": [
        "How many seats do you require"
      ],
      "selection_policy": "sequential"
    }
  },
  "parent": "slot_9_1603125198263",
  "event_name": "focus",
  "dialog_node": "handler_3_1603125198281",
  "previous_sibling": "handler_8_1603125198281"
},
{
  "type": "event_handler",
  "output": {
    "text": {
      "values": [
        "Please select your time"
      ],
```

```
    "selection_policy": "sequential"
  }
},
"parent": "slot_10_1603124898138",
"event_name": "focus",
"dialog_node": "handler_6_1603124898155",
"previous_sibling": "handler_9_1603124898154"
},
{
  "type": "event_handler",
  "output": {},
  "parent": "slot_4_1603124612331",
  "context": {
    "Movies": "@Movies"
  },
  "conditions": "@Movies",
  "event_name": "input",
  "dialog_node": "handler_7_1603124612400"
},
{
  "type": "event_handler",
  "output": {},
  "parent": "slot_9_1603125198263",
  "context": {
    "number": "@sys-number"
  },
  "conditions": "@sys-number",
  "event_name": "input",
  "dialog_node": "handler_8_1603125198281"
},
{
  "type": "event_handler",
  "output": {},
  "parent": "slot_10_1603124898138",
  "context": {
    "Time": "@Time"
  },
}
```



```
"conditions": "@Time",
"event_name": "input",
"dialog_node": "handler_9_1603124898154"
},
{
  "type": "standard",
  "title": "Greetings",
  "metadata": {
    "_customization": {
      "mcr": true
    }
  },
  "conditions": "#Greetings || @Greetings",
  "dialog_node": "node_10_1603114659514",
  "previous_sibling": "Welcome"
},
{
  "type": "standard",
  "title": "Thanks",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "Thank you for using our service visit again."
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  },
  "metadata": {
    "_customization": {
      "mcr": false
    }
  }
},
```

```
"conditions": "#Thanks || @Thanks",
"dialog_node": "node_10_1603127279080",
"previous_sibling": "node_8_1603124489383"
},
{
  "type": "standard",
  "title": "Movies",
  "output": {
    "generic": [
      {
        "title": "Select a movie",
        "options": [
          {
            "label": "V (film)",
            "value": {
              "input": {
                "text": ""
              }
            }
          }
        ],
        {
          "label": "Ala Vaikunthapurramuloo",
          "value": {
            "input": {
              "text": ""
            }
          }
        }
      ]
    },
    "response_type": "option"
  }
],
"parent": "node_4_1603120010446",
"conditions": "@Movies",
"dialog_node": "node_3_1603123198112"
},
```

```

{
  "type": "standard",
  "title": "Deleting context",
  "output": {
    "deleted":
"<?context.remove('time')?><?context.remove('number')?><?context.remove('payment')?
>",
    "generic": [
      {
        "values": [],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  },
  "parent": "node_8_1603124489383",
  "conditions": "true",
  "dialog_node": "node_3_1603125739441"
},
{
  "type": "standard",
  "title": "Inquiry",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "10% off on booking 3 tickets at a time"
          },
          {
            "text": "10% off on UPI Payments"
          }
        ]
      },
      {
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  }
}

```

```

    },
    "metadata": {
      "_customization": {
        "mcr": false
      }
    },
    "conditions": "#Inquiry && @Inquiry:Offers",
    "dialog_node": "node_4_1603120010446",
    "previous_sibling": "node_9_1603118785262"
  },
  {
    "type": "frame",
    "title": "Book tickets",
    "output": {
      "generic": [
        {
          "values": [
            {
              "text": "Thank you for your booking of $number seats at $Time for $Movies.  
You can pay the amount using $Payment."
            }
          ],
          "response_type": "text",
          "selection_policy": "sequential"
        }
      ]
    },
    "conditions": "#Book_Tickets || @Book_Tickets",
    "dialog_node": "node_8_1603124489383",
    "previous_sibling": "node_4_1603120010446"
  },
  {
    "type": "standard",
    "title": "Movie-Availability",
    "metadata": {
      "_customization": {
        "mcr": true
      }
    }
  }

```

```
    }
  },
  "conditions": "#Movies-Availability || @Movies-availability",
  "dialog_node": "node_9_1603118785262",
  "previous_sibling": "node_10_1603114659514"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "source":
https://2.bp.blogspot.com/-sUvp5mU5YG0/XFWlv2A\_XRI/AAAAAAAAABs/\_CJLM6RbXRk5kkiTfWbMHllgBsd6NSyeQCLcBGAs/s1600/two.jpg,
        "response_type": "image"
      }
    ]
  },
  "parent": "node_9_1603118785262",
  "conditions": "anything_else",
  "dialog_node": "response_1_1603119753344",
  "previous_sibling": "response_2_1603118878844"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "title": "Visakhapatnam",
        "options": [
          {
            "label": "Ala Vaikunthapurramuloo",
            "value": {
              "input": {
                "text": "Ala Vaikunthapurramuloo"
              }
            }
          }
        ]
      }
    ]
  }
}
```

```
    },
    {
      "label": "Bheeshma",
      "value": {
        "input": {
          "text": "Bheeshma"
        }
      }
    }
  ],
  "response_type": "option"
},
{
  "title": "Vizaganaram",
  "options": [
    {
      "label": "V (film)",
      "value": {
        "input": {
          "text": "V (film)"
        }
      }
    }
  ],
  {
    "label": "Sarileru Nekevaru Sarileru Neekevaru",
    "value": {
      "input": {
        "text": "Sarileru Nekevaru Sarileru Neekevaru"
      }
    }
  }
],
  "response_type": "option"
},
{
  "title": "Kakinada",
  "options": [
```

```

    {
      "label": "Jaanu",
      "value": {
        "input": {
          "text": "Jaanu"
        }
      }
    },
    {
      "label": "V (film)",
      "value": {
        "input": {
          "text": "V (film)"
        }
      }
    }
  ],
  "response_type": "option"
}
],
"parent": "node_9_1603118785262",
"conditions": "@Movies-availability:City",
"dialog_node": "response_2_1603118878844"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "values": [],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  }
},
"parent": "node_10_1603127279080",

```

```
"disabled": true,
"dialog_node": "response_4_1603168396999"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "Good Evening"
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  },
  "parent": "node_10_1603114659514",
  "conditions": "@Greetings:(Good Evening)",
  "dialog_node": "response_5_1603115205809",
  "previous_sibling": "response_6_1603115184625"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "Good Afternoon"
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  }
}
```



```
},
"parent": "node_10_1603114659514",
"conditions": "@Greetings:(Good Afternoon)",
"dialog_node": "response_6_1603115184625",
"previous_sibling": "response_7_1603114991217"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "Hi"
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  },
},
"parent": "node_10_1603114659514",
"conditions": "@Greetings:hi",
"dialog_node": "response_6_1603115233775",
"previous_sibling": "response_5_1603115205809"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "Good Morning"
          }
        ],
        "response_type": "text",

```

```
        "selection_policy": "sequential"
      }
    ]
  },
  "parent": "node_10_1603114659514",
  "conditions": "@Greetings:(Good Morning)",
  "dialog_node": "response_7_1603114991217"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "Hello"
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  },
  "parent": "node_10_1603114659514",
  "conditions": "@Greetings:hello",
  "dialog_node": "response_7_1603115243150",
  "previous_sibling": "response_6_1603115233775"
},
{
  "type": "response_condition",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "Hi"
          }
        ]
      }
    ]
  }
}
```

```
    ],
    "response_type": "text",
    "selection_policy": "sequential"
  }
],
},
"parent": "node_10_1603114659514",
"conditions": "anything_else",
"dialog_node": "response_9_1603115299126",
"previous_sibling": "response_7_1603115243150"
},
{
  "type": "slot",
  "parent": "node_8_1603124489383",
  "variable": "$Time",
  "dialog_node": "slot_10_1603124898138",
  "previous_sibling": "slot_4_1603124612331"
},
{
  "type": "slot",
  "output": {},
  "parent": "node_8_1603124489383",
  "variable": "$Movies",
  "dialog_node": "slot_4_1603124612331",
  "previous_sibling": "handler_2_1603124610462"
},
{
  "type": "slot",
  "parent": "node_8_1603124489383",
  "variable": "$Payment",
  "dialog_node": "slot_5_1603125270357",
  "previous_sibling": "slot_9_1603125198263"
},
{
  "type": "slot",
  "parent": "node_8_1603124489383",
  "variable": "$number",
```

```
"dialog_node": "slot_9_1603125198263",
"previous_sibling": "slot_10_1603124898138"
},
{
  "type": "standard",
  "title": "Welcome",
  "output": {
    "generic": [
      {
        "values": [
          {
            "text": "Hello. I am movie ticket booking bot, I am capable of showing Movies availability,Book tickets and Inquiry.... How may help you ?"
          }
        ],
        "response_type": "text",
        "selection_policy": "sequential"
      }
    ]
  },
  "conditions": "welcome",
  "dialog_node": "Welcome"
}
],
"counterexamples": [],
"system_settings": {
  "off_topic": {
    "enabled": true
  },
  "disambiguation": {
    "prompt": "Did you mean:",
    "enabled": true,
    "randomize": true,
    "max_suggestions": 5,
    "suggestion_text_policy": "title",
    "none_of_the_above_prompt": "None of the above"
  }
},
```

```
"system_entities": {  
  "enabled": true  
},  
"human_agent_assist": {  
  "prompt": "Did you mean:"  
},  
"spelling_auto_correct": true  
},  
"learning_opt_out": false,  
"name": "My first skill",  
"language": "en",  
"description": ""  
}
```