

A Comparison of Various Chatbot Frameworks

Vatsal Shah^{#1}, Dr.Seema Shah^{*2}

[#]Department of Computer Engineering, Mukesh Patel School of Technology Management and Engineering, NMIMS University

¹vatsalshah7878@gmail.com

²seema.shah@nmims.edu

Abstract— The technological advances in artificial intelligence and machine learning have led to the growth in the development of chatbots. Chatbots are the virtual agents which mimic a user to give a response to the user's queries. Chatbots have gained a lot of popularity in the business as it can automate the customer service and reduce the efforts of the human. Chatbot frameworks are the prebuilt bot engines which help users to reduce the development time. In this paper, overview and comparison of various chatbot frameworks is presented along with this it also helps the users to identify how to select one of the chatbot framework for their business. Chatbot frameworks are compared on the basis of features it provides to end users.

Keywords— Chatbots, Chatbot Frameworks, NLP (Natural Language Processing), Chatbot Architecture, Machine Learning

I. INTRODUCTION

As the rising demand for self-service and the technological advances in AI and NLP the platform for chatbot has risen. According to Gartner the 50% of medium and large enterprises will deploy a chatbot. A chatbot can carry out many of the functions of mobile applications or website, all within a messaging platform. Also, People may interact with computers in a completely new way Instead of downloading an app to get information; they might just add the corresponding chatbot to their preferred messenger platform and they would not have to wait for the response as using chatbot instant service is provided compared to traditional method where user have to wait for long to get connected to the service provider to get solution.

The chatbots are computer program which offers a conversational experience using artificial intelligence and natural language processing to mimic conversations with real people. Chatbot are also used in many types of applications such as booking flight tickets, shopping, news updates, ordering pizza, get updates about weather, updates about news, product suggestions, customer support, finding a restaurant for you etc. are just examples of how chatbot can be used there are limitless possibilities for what chatbot can be implemented to engage, help or sell the products to the user.

Chatbots are getting popular and they are now creeping into our smartphones. People spend a lot of time on the apps installed in the smartphones every day. The underlying principle of every chatbot is to interact with a human user via text messages and behave as though it were capable of understanding the conversation and reply to the user appropriately. Chatbot that functions based on rules: -This bot is very limited. It can only respond to very specific commands. If anything, apart from set of question is asked, the bot doesn't know the answer. This bot is only as smart as it is programmed to be. Chatbot that functions using machine learning: -This bot has an artificial brain i.e. artificial intelligence. You don't have to be specific when you are talking to it. It understands language, not just commands. This bot continuously gets smarter as it learns from conversations it has with people [1].

The general chatbot architecture is shown in Figure 1. The User input will be processed through two modules intent classification and entity recognition. Intent classification checks the input message and based on the input sentence identifies the intent i.e. what exactly user wants to convey.

Entity recognition module extracts the keyword from the input. Response generator uses the algorithm to generate the response. The response generator uses the identified intent and entities as well as the context of the conversation which is extracted from the last input message and selects the appropriate response. E.g. In a flight bot the identification of users input is necessary to confirm whether he wants to book a flight or want to inquire something about flight, Entities extraction is necessary as it identifies the city, date etc. for response generation.

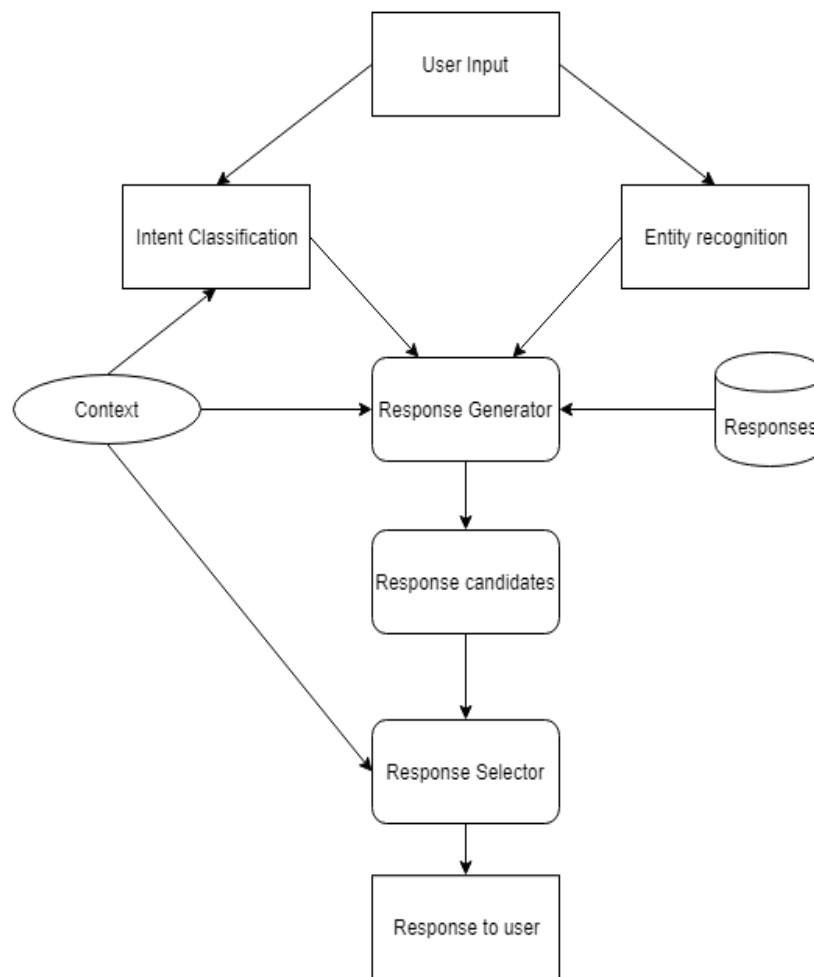


Fig. 1. Chatbot Architecture

User Input: - A user can give an input message to the system.

Pre-Processing: - consist of many steps like removal of stop words, tokenization, and conversion in lower case, etc.

Intent Classification: - Intent classification module identifies the intent of user message.

Entity Recognition: - Entity Recognition module extracts structured bits of information from the message.

Response Generator: - Response generator is doing all the domain-specific calculations to process the user request.

Response Candidates: - Response candidates are the set of responses which can be the output to the user input.

Response Selector: - Response selector scores the response candidates and selects a response which should work better for the user.

For each step-in architecture, the methods or algorithm to the corresponding step can be utilized. From [3] part of speech tagging, chunking, split into words etc. are some input processing algorithms which can be done through various libraries which can be utilized as a function. Intent Classification entity detection takes place for understanding the input. Once this input is structured the response can be summarized. Chatbot can be divided into two parts on basis of its response generation [4].

Retrieval based response models: these types of models are easy to build as it depends on the rules defined or on the defined question answers pair. When the information is constrained and the discussion area is limited to a couple of situations retrieval-based models can be utilized. Additionally, this type of bots does not guarantee to give response every time sometimes it may deny knowing the response. It can work well for use cases of Flight Booking, FAQ bots etc. **Generative based response models:** these types of models are hard to build as the domain of the chatbot is not fixed it can be from any domain. to train these types of model's large corpus of data is required.

For intent detection various algorithms can be used to classify the domain of the input query. Algorithms like Random forest, SVM, Naïve Bayes, LSTM [5] CRF [6], BI- LSTM [7] [8] etc. are some approaches to classify the intent of the user query. For named Entity Recognition various algorithms can be used to identify entities like name, place etc. Algorithms like CRF, SVM,

HMM, MeMM [9] etc. are some approaches to extract entities from the user query. All the steps are not required to be developed from scratch as there are many chatbot frameworks which provide features to complete all the required steps for development of chatbot.

II. RELATED WORK

In 1950, Alan Turing asked the question “Can machines think?” Turing conceptualized the problem as an “imitation game” (now called the Turing Test), in which an “interrogator” asked questions to human and machine subjects, with the goal of identifying the human. If the human and machine are indistinguishable, we say the machine can think [10].

In 1966, Joseph Weizenbaum at MIT created the first chat- bot that, arguably, came close to imitating a human: ELIZA. Given an input sentence, ELIZA would identify keywords and pattern match those keywords against a set of pre-programmed rules to generate appropriate responses [11]. Since ELIZA, there has been progress in the development of increasingly intelligent chatbots.

In 1972, Kenneth Colby at Stanford created PARRY, a bot the impersonated a paranoid schizophrenic [12]. In 1995, Richard Wallace created A.L.I.C.E, a significantly more complex bot that generated responses by pattern matching in- puts against (input) (output) pairs stored in documents in a knowledge base. These documents were written in Artificial Intelligence Markup Language (AIML), an extension of XML, which is still in use today.

ALICE is a three-time winner of the Loebner prize, a competition held each year which attempts to run the Turing Test, and awards the most intelligent chatbot [13]. Modern chatbots include: Amazon’s Echo and Alexa, Apple’s Siri, Microsoft’s Cortana and Google assistant.

In [2], the authors discuss the overview of chatbot frame- work (IBM Watson, LUIS Wit.ai etc) along with challenges of programming a chatbot. In [14], the authors discuss how it is possible to implement a chatbot by using chatbot frameworks, but they do not provide any experimental evaluation for the adopted NLU system. In [15], the authors developed a chatbot by using Dialog Flow for the visualization of lecture talk. They developed it without comparison with other chatbot frameworks. In [16], the authors compare 4 NLU platforms only from a performance-based point of view without providing the difference between the frameworks.

Despite the spread of chatbots in customer care support, food order, shopping assistant, reservation etc. the literature has not proposed yet a comparison concerning the chatbot framework available nowadays. In this paper, we plan to provide comparison of the various chatbot frameworks available nowadays.

III. CHATBOT FRAMEWORKS

Chatbot frameworks help to build, connect, publish and manage chatbots which are smart and interactive to give the best user experience. It helps to create new chatbots with ease. The chatbot frameworks which we identify are: IBM Watson, Dialog Flow, Wit.ai, Rasa NLU, LUIS, Agentbot, Pyepstream, Kitt.ai, Semantic Machines, Pandorabots Gupshup and Digital Genius. All these platforms are powered by machine learning algorithms. The main operation performed by the frameworks is to identify intent and entity. An intent represents a mapping between what a user says and what action should be taken by the chatbot. It represents a portion of a conversation. An entity, instead, is a tool for extracting parameter values from natural language inputs. Any important keyword you want to get from a user’s request, will be linked to an entity. for e.g. “Book a flight to Paris?” is the input of the intent for asking to book a flight, while “Paris” is an individual of a possible entity named City that collect the names of any city in the world.

A. IBM Watson

Watson Assistant is an offering for building conversational interfaces into any application, device, or channel.it has ability to identify when it needs to search in knowledge base, ask again for clarity or transfer to human. It is built on a neural network which has three main components: Intents, Entities, Dialog. It has uses cases in Healthcare, Finance, Legal, Retail, Fantasy Football, customer service etc. [17].

Key features: -

- Channel integration
- Skill routing

- Session management
- Hand-off to human agent
- Provide a specific and customized solution

B. *Dialog flow*

Dialog flow matches the query to the most suitable intent based on information contained in the intent and the agent's machine learning model.

It returns output data as a JSON response object. It leverages predefined knowledge packages collected over several years [18].

Key features: -

- Using Intents and Contexts
- The modeling of large and complex systems.
- Ability to handle the code proactively and decreases server-side coding.
- It offers one click integration with the several platforms such as Twitter, Facebook.

C. *Microsoft Language Understanding (LUIS)*

LUIS is a machine learning-based service to build natural language into apps, bots, and IoT devices. User can use enterprise ready, custom models that continuously improve.

LUIS contains a domain-specific natural language model. It has many prebuilt domain models including intents, utterances, and prebuilt entities. Prebuilt domain models include the completed design of the conversation which can be utilized for quick start with LUIS [19].

Key features: -

- It Uses intents and entities.
- Domain-specific topic or content related.
- Active learning.
- Pre-existing and pre-built models from Bing and Cortana.
- JSON for easy use.
- It gives HTTP endpoint for integration.

D. *Rasa NLU*

The Rasa Stack is a set of open source machine learning tools for developers to create contextual chatbots and assistants. It has solutions in Contextual AI Assistant, On premises Chatbots, Bots & Digital Assistants, Lead Generation & Sales, Internal Processes Automation, and Customer Service [20].

Key features: -

- User can tweak and customize models for training data.
- It can be deployed on premises.

E. *Wit.ai*

Wit.ai works on stories as a key concept to model the behavior of Chatbot. Every story gives an example of all possible conversation. In this bot the developers basically teach the Wit.ai using examples and when a user writes about the similar type of object. It will be able to process the request and get the extract entities and apply developer logic on the scenario. It is based on the web hook integration in which information bot sends command into web service and gets the results from it [21].

Key features: -

- The story concept is very useful and powerful.
- Branches lead to better control the conversation also conditions on the actions.

F. *Agent Bot*

Agent Bot's NLU engine allows it to interpret conversational language and it learns from each interaction. Agent Bot is an Artificial Intelligence solution that integrates in customer service [22].

Key features: -

- Understands natural language.
- Memory to maintain coherence during long conversations.

- Gathers customer information to deliver customized solutions.
- Continuous evolution.

G. *Pypestream*

Pypestream uses AI-based machine learning classifiers that can be trained to identify the entities based on the intents. It has uses cases in Telecom, Insurance, Travel, and Financial Services [23].

Key features

- Customer service automation
- It can process payments.
- Security, privacy, and compliance
- User authentication, audit trails, and strict data privacy standards.

H. *Semantic Machines*

Semantic Machines is a language-independent technology platform that understands the conversations. Its conversational AI represents a powerful new paradigm, enabling computers to communicate, collaborate, understand our goals, and accomplish tasks.

It has use cases in E-Commerce, Travel, Concierge, Calendar, Business, Search, Productivity and Automotive [24].

Key features: -

- Conversation Engine
- Speech Recognition
- Speech Synthesis
- Reinforcement Learning
- Language Independent

I. *Pandorabots*

Pandorabots offers various services to meet business needs from custom solutions to chatbot development. The Pandorabots API allows you to integrate bot hosting service and natural language processing engine into one's own application. The bot built with it can be integrated with wide range of web platform and applications.

Its use cases include advertising, virtual assistance, e- learning, entertainment and education [25].

J. *Gupshup*

Gupshup has created an API called 'NLP on the fly' that can be used to process natural language queries. This API consists of intent and entities extraction which can be used easily. The main advantage is there is no need to train the model before implementing it.

The API works on the basis of the variations for intent. The JSON is created that contains few variations of all the intents long with their corresponding entities. Gupshup's messaging platform is used by over 30,000 businesses [26].

Key features: -

- Deploy on any platform with one click integration option

K. *Kitt.ai*

Kitt.ai uses a simple drag-and-drop interface that visually describes a dialogue and at the same time implements the flow that can be executed on the server as the dialogue is designed. It liberates developers to focus on the dialogue flow, rather than the code [27].

Key features: -

- Hot word Detection (no internet required)
- Semantic Parsing
- Natural Language Understanding
- Conversational Engine (multi-turn)
- Neural network powered machine learning model

L. Digital Genius

Digital Genius is the AI-powered platform for customer service that uses conversational process automation to seamlessly connect conversations with processes. AI predicts case metadata and suggests the right answers to your agents. AI learn from every agent interaction. It uses a deep neural network model, word vectors and statistical operations [28].

Key features: -

- Reduce Customer Support Costs
- Dazzle Your Customers
- Rapid Time to Value
- Unlock Critical Time for Customer Support

IV. COMPARISON OF CHATBOT FRAMEWORKS

In this section we compare the chatbot frameworks which we presented in the previous section.

The chatbot frameworks discussed here has a common feature of text processing. In which they aim to detect the intent of the user query and identify the entity from the query provided by the user. The basic feature for all framework is intent detection and entity recognition.

In Table 1 We will be comparing the frameworks on the following additional features they provide to the users.

Speech Processing it tells whether the framework can process speech as well for conversation.

Pre-build Entities indicates how many pre-build entities are offered. E.g. of prebuilt entities are date, location, name etc.

Pre-build Intents indicates h many pre-build groups of in- tents are offered. E.g. of prebuilt intents are greeting, currency converter weather news etc.

Default Fallback Intent indicates fallback mechanism al- lows classification of sentences that are not recognized as part of a defined intent; without a dedicated fallback mechanism, every sentence will belong to a defined intent.

Online Integration indicates which third-party integration are available.

Machine learning indicates whether the framework automatically learns from its previous inputs and improve over the time.

License is the pricing for using the frameworks.

Languages tells how many languages are supported by the framework.

The additional features speech processing, predefined entities, predefined intents, default fallback intent, online integration and machine learning are additional features which are provided by the frameworks it plays a vital role as user can get various extra features apart from text processing. The selection of right chatbot frameworks is also a vital step for a user, based on the steps and comparison of frameworks given one can identify the right framework for use.

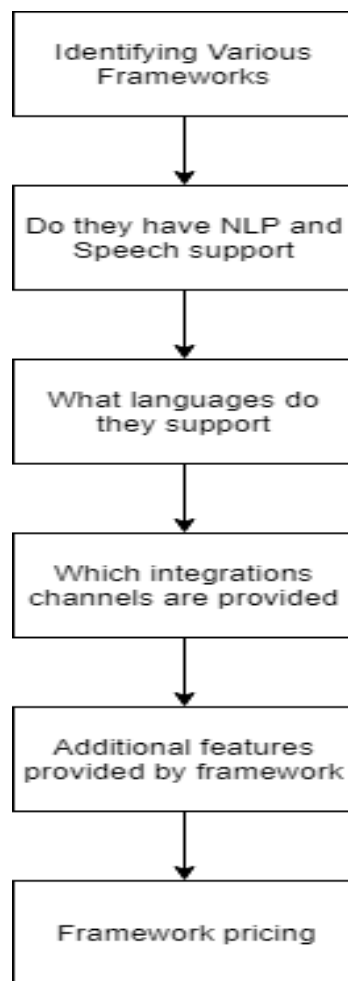


Fig. 2. Steps for framework selection

From figure 2 We can identify how to select the chatbot framework which suits best to the user. To identify which frameworks to select for the development of the chatbot one has to identify various framework first. The next step is to identify whether they support NLP and speech processing and what different languages do they support for processing of the queries. After identifying this for the framework it is important to know what applications integration this framework provides easily so development can be easy. Also, additional features like predefined intent and entities. Machine learning features etc. are key element to identify the frameworks. The pricing of chatbot frameworks is also important factor for deciding the chatbot framework.

In the Table 1 we compared the various chatbot framework on the criteria's discussed above. These criteria specify the features which each framework provides. The basic features provided by all the framework are text processing, intent detection and entity recognition apart from these the frameworks gives some other features which are useful for ease the development of chatbot. The criteria discuss here are the features which helps developer to develop the chatbots faster due to number of features provided by the framework.

From the Table 1 it can be analyzed from a descriptive point of view, we can identify that Dialog Flow is the most complete framework since it provides the best solutions for the majority of features considered for selecting the framework for development of chatbots.

TABLE 1
COMPARISON OF CHATBOT FRAMEWORKS

Name	Speech Processing	Prebuilt Intent	Prebuilt Entities	Default Fallback Intent	Online Integration	Machine learning	Languages Supported	License
IBM Watson	Yes	No	Yes	Yes	No	Yes	Multilingual	Free Trial
Dialogflow	Yes	Yes	Yes	Yes	Yes	Yes	Multilingual	Free Trial
LUIS	Yes	Yes	Yes	Yes	No	Yes	Multilingual	Free Trial
Rasa NLU	No	No	No	No	No	No	English, German	Opensource
Wit.ai	Yes	No	Yes	No	No	Yes	Multilingual	Free with certain limits
AgentBot	Yes	No	No	No	Yes	No	English	Free Trial
Pypestream	No	No	No	No	No	No	English	Paid
Semantic Machines	Yes	No	No	No	No	No	Language Independent	Paid
Pandorabots	No	No	No	No	Yes	No	Multilingual	Paid
Gupshup	No	No	Yes	No	No	No	English	Paid
Kitt.ai	Yes	No	No	No	No	No	English	Paid
Digital Genius	No	No	No	No	Yes	No	English	Paid

V. CONCLUSION

In this paper, we introduced the chatbot, its architecture and compared the chatbot frameworks. The chatbots can be developed for any purpose like booking flight tickets, shopping, customer services, news updates etc. The main steps from the chatbot architecture given are intent and entity recognition. All the frameworks discussed have common feature to identify the intents and entity and also are used to process the text. The frameworks will help developer to develop any chatbot with ease and reduce the development efforts. All the chatbot frameworks discussed in this paper gives an overview of each and also states its features. We compared these frameworks on the certain features criteria and identify that Dialog flow framework by Google works best on the basis of descriptive comparison. We also discussed the steps on how to select a framework for the business and which would be suitable on the basis of what the users needs the features in chatbot.

As future work, we plan implement the frameworks and evaluate to understand at which extent each platform is able to detect the intention of the user queries.

REFERENCES

- [1] [Online]. Available: <https://chatbotsmagazine.com/the-completebeginner-s-guide-to-chatbots-8280b7b906ca>
- [2] AM Rahman et.al., "Programming challenges of Chatbot: Current and FutureProspective", 2017 IEEE Region 10 Humanitarian Technology Conference, 2017
- [3] [Online]. Available: <https://chatbotsmagazine.com/machine-learningneural-networks-and-algorithms-5c0711eb8f9a>
- [4] [Online]. Available: <https://medium.com/>
- [5] [Online]. Available: <https://medium.com/@surmenok/natural-language-pipeline-for-chatbots-897bda41482>
- [6] Lirong Qiu et.al., "Query intent recognition based on multi-class features", IEEE Access, 2017.
- [7] Thai-Le Luong et.al., "Intent Extraction from Social Media Texts Using Sequential Segmentation and Deep Learning Models", IEEE, 2017.
- [8] Joo-Kyung Kim et.al., "Intent Detection using Semantically enriched Word Embeddings", IEEE, 2016.
- [9] P. and, Fousiya K.K., "Named Entity Recognition Approaches: A Study applied to.
- [10] A. M., "Computing Machinery and Intelligence,," Mind, LIX, vol. 236, pp. 433–460, 1950.
- [11] J. Weizenbaum, "ELIZA—a computer program for the study of natural language communication between man and machine,," Communications of the ACM, vol. 9, no. 1, pp. 36–45, 1966.
- [12] K. M. Colby, "Modeling a paranoidmind,," Behavioral and Brain Sciences, vol. 4, no. 4, p. 515, 1981.

- [13] R. S. Wallace, "The Anatomy of A.L.I.C.E. <http://www.alicebot.org/anatomy.html>."
- [14] P. Amit, K. Marimuthu, R. A. Nagaraja, and R. Niranchana, "Comparative study of cloud platforms to develop a chatbot" *International Journal of Engineering & Technology*, vol. 6, no. 3, pp. 57–61, 2017.
- [15] S. M. R. Ishimura, Chatbot and dialogue demonstration with a humanoid robot in the lecture class., 2017, vol. 10279.
- [16] D. B, Evaluating natural language understanding services for conversational question answering systems, 2017.
- [17] [Online]. Available: <https://www.ibm.com/watson>
- [18] [Online]. Available: <https://dialogflow.com/>
- [19] [Online]. Available: <https://www.luis.ai>
- [20] [Online]. Available: <https://rasa.com/>
- [21] [Online]. Available: <https://wit.ai/>
- [22] [Online]. Available: <https://aivo.co/en/agentbot-automatic-support-chat/>
- [23] [Online]. Available: <https://www.pypestream.com>
- [24] [Online]. Available: <http://www.semanticmachines.com/>
- [25] [Online]. Available: <https://www.pandorabots.com/>
- [26] [Online]. Available: <https://support.gupshup.io/>
- [27] [Online]. Available: <http://kitt.ai/>
- [28] [Online]. Available: <https://www.digitalgenius.com/>