Content

[Abstract 3](#_Toc48473637)

[1. Introduction 4](#_Toc48473638)

[1.1 Background 4](#_Toc48473639)

[1.2 Motivation 4](#_Toc48473640)

[1.3 Chatbot 5](#_Toc48473641)

[1.4 NLU 5](#_Toc48473642)

[1.5 Rasa 6](#_Toc48473643)

[2. Literature Review 8](#_Toc48473644)

[2.1 Chatbot 8](#_Toc48473645)

[2.1.1 Social chatbot - XiaoIce 8](#_Toc48473646)

[2.1.2 Task-oriented chatbot 11](#_Toc48473647)

[2.2 NLU 11](#_Toc48473648)

[2.3 Rasa 11](#_Toc48473649)

[2.3.1 Rasa Architecture 11](#_Toc48473650)

[2.3.2 Compare NLU framework 11](#_Toc48473651)

# Abstract

# Introduction

In this chapter, this essay will mention about the background and the motivation of this project. Also, introduce some primary concepts about the chatbot, natural language understanding and Rasa.

## Background

Machines can think. It is a prevalent story in a fiction movie or novel. Such as in movie Prometheus[[1]](#footnote-1), David is an AI robot which helps main actors to analyse information, answer questions and support them during their universe adventure. It is hard to believe that would become true in part of regular life one day in the few decades. However, it seems not absurd anymore.

In recent years, artificial intelligence has explored widely. One of the major topics to be investigated in this field is the chatbot which connects with Machine Learning, Big Data and Natural Language Processing. These research let the machine can understand what does human say and respond to a proper answer to people. For example, Watson is the question-answering system posted in natural language and developed by IBM. In 2011, it defeated the human player to win the first-place prize on Jeopardy![[2]](#footnote-2) Moreover, in 2018, google demo the Google Assistant to make an appointment in an actual phone call.[[3]](#footnote-3)

Due to those significant achievements from the chatbot, people are developing a lot of different type of application in the business market. Also, implement the chatbot on social media like Facebook, WhatsApp, Telegram, etc. to help people solve their problem. It not only reduces the personnel costs but also increases the convenient. That is why the chatbot has gotten popular.

## Motivation

## Chatbot

The chatbot is a software application designed for extended conversations and set up to mimic the human to human chat behaviour. Also, it used to conduct an online chat conversation via text instead of providing direct contact with a live human agent.

Chatbots are regularly utilized in dialogue systems for different purposes including client service, request routing, or for data collection. Some of the chatbot applications use extensive word-classification processes, Natural Language processors, and complicated AI, others scan for common keywords and generate replies using general phrases obtained from an associated library or database.

According to Mnasri (2019), there are two categories of property for the chatbot. One is social chatbots which designed to offer unstructured human-like conversations. The other is task-oriented chatbots which aimed to carry short conversations and accomplish simple tasks related to specific things.

J. Daniel and J. Martin (2018) indicated that chatbot architectures could be divide into two building approaches. First are rule-based systems which involve the early famous ELIZA and PARRY systems. Second are corpus-based systems. It analyses large datasets of human-human conversations, which can be done by using information retrieval or by using a machine translation typical example such as neural network sequence-to-sequence systems, to learn to map from a user announcement to system response.

## NLU

Natural language understanding (NLU) is a part of natural language processing (NLP) in artificial intelligence that uses software to understand what the meaning is from sentences in text or speech.

Natural language processing (NLP) is a subset of AI, and it involves programming computers to process large volumes of language data. It comprises numerous tasks that split natural language into smaller components to understand the relationships between those components and how they work together. Typical tasks include parsing, speech identification, tagging part-of-speech, and data extraction. NLP focuses mainly on transforming text into structured data.

NLU is a vital and challenging subset of natural language processing (NLP). NLU is narrower in purpose, focusing mainly on machine reading comprehension. Although NLU and NLP understand human language, NLU is tasked with interacting with untrained individuals and understanding their intent. It means that NLU goes beyond understanding words and interprets meaning.

NLU uses algorithms to reduce human speech into a structured ontology. AI can find out things like intent, time, locations, and sentiments. For example, I want to reserve a table for two people at the Birmingham restaurant on the 4th of September might break down into this:

intent: [reserve a table] for reservation,

intent: [two people] for how many people,

time: [4th of September] for reservation date,

and locations: [Birmingham restaurant] for the restaurant name.

<https://www.uctoday.com/collaboration/what-is-natural-language-understanding/>

<https://blog.rasa.com/nlp-vs-nlu-whats-the-difference/>

<https://en.wikipedia.org/wiki/Natural-language_understanding>

<https://www.bmc.com/blogs/nlu-vs-nlp-natural-language-understanding-processing/>

## Rasa

Rasa is an open-source Machine Learning framework for making chatbots. It supports not only text but also voice-based dialogue. Additionally, it can connect to messaging social media such as Facebook, Telegram, etc., and APIs. Rasa also has an external feature named Rasa X, which uses actual conversations to enhance the chatbot and building chatbot as well.

Rasa helps you create contextual assistants capable of having layered conversations with lots of backwards and forwards. For a human to have a significant exchange with a contextual assistant, the assistant needs to be capable to use context to build on something that was previously discussing.

<https://rasa.com/>

<https://pypi.org/project/rasa-nlu/>

# Literature Review

In this chapter, it will cite different approaches to analysis the chatbot. Section 2.1 lists several past studies for chatbot architecture and applications. Section 2.2 introduces the purpose of the NLU and some NLU studies. Section 2.3 explains how does the Rasa architecture work and contains some critical difference with another framework - Dialogflow. Through these differences to understand why to choose Rasa.

## Chatbot

In this section, it will offer two case studies which are using different methods to construct. First one is social chatbot. Second is task-oriented chatbot.

### Social chatbot

Here will introduce a famous social chatbot example – XiaoIce.

**Introduction**

In May 2014, Microsoft released XiaoIce, which is one of the famous social chatbots. It is like a real friend who can understand users' emotional needs and engage communications.

A recent study by Shum et al. (2018) concluded that challenges and opportunities with social chatbots. With the reproduction of smartphones and the progression of broadband wireless technology, social chatbots are created to assist people's needs for interaction, affection, and social belonging. Consequently, social chatbots must be able to identify the emotion and trace emotional changes during a conversation.

This chatbot also needs the ability to handle multiple tasks when users are talking about the context of random chats. For this reason, the chatbot has to develop a set of ways to serve users' demands. Therefore, the chatbot takes time to talk like a human, express results, offering perspectives, generating new topics to retain the conversation going.

**Design Principles**

The method introduced by Zhou et al. (2020) mentions three main design principles.

1. **IQ + EQ + Personality:**

Knowledge and memory modelling, image and natural language understanding, reasoning, generation, and prediction comprise IQ capacities. These parts are fundamental for the procession of dialogue ability. Additionally, meeting users' specific requests and accomplish their tasks are necessary for social chatbot.

EQ contains empathy and social skills. A social chatbot requires to have the ability to recognise the users' emotions from conversations, detects how the emotion change during chatting and knows the users' emotional requirements to achieve empathy ability. In other words, query understanding, user profiling, emotion detection, sentiment recognition, and dynamically tracking the mood of the user in a conversation are the key points to reach this goal.

Personality is represented as a particular set of actions, cognition, and emotional patterns that form an individual’s unique character.

1. **Conversation-turns Per Session:**

Conversation-turns Per Session (CPS) which is considered to the vital point for social chatbots. CPS is an average number of conversation-turns between the user and the chatbot during the conversation. If CPS is large, it means that the chatbot has better engaged.

1. **Hierarchical Decision-Making**

To attract users’ interest, XiaoIce tries to promote a variety of conversation modes. For each conversation mode, it runs by a skill that handles a specific type of conversation segment.

XiaoIce uses Markov Decision Processes (MDPs) (Sutton, Precup, and Singh 1999) as the mathematical framework which aims to use a mathematically way to cast human-machine social chat as a hierarchical decision-making process.

Through the MDP and conversation with users, the chatbot would observe the current dialogue state and choose a proper option according to a hierarchical dialogue policy. Afterwards, the chatbot receives a reward and keeps watch new state until the dialogue end.

The design purpose of the chatbot is to obtain optimal policies and skills to maximize the expected CPS (rewards).

**System Architecture**

Zhou et al. (2020) also introduced that XiaoIce system architecture which consists of three layers, user experience, conversation engine and the data.

1. **User experience layer:**

This layer integrates to chat platforms. Moreover, it supports two modes. One is the full-duplex mode which deals with the voice-stream-based conversation. The other is taking turns mode, which handles message-based communications.

1. **Conversation engine layer:**

This layer contains several vital components which are dialogue manager, empathetic computing module, Core Chat, and dialogue skills. The dialogue manager decides to select dialogue skill or Core Chat using the dialogue policy to generate replies by tracking of the dialogue state. The empathetic computing module can understand the content of user input and the empathetic aspects of the conversation.

1. **Data layer:**

This layer consists of the different dataset, which is collected human conversational data, non-conversational data, knowledge graphs, etc.

**Conversation Engine**

Conversation Engine is the primary part of XiaoIce, and this engine helps XiaoIce makes the proper response to the user. According to the above system architecture, conversation contains Dialogue Manager, Empathetic Computing, Core Chat and Dialogue Skills. These components have more detail elements to accomplish the duty. Here will mention some important features.

1. **Dialogue Manager**

Dialogue Manager controls the whole dialogue system. It involves three parts. One of them is the Global State Tracker, which is responsible for tracking the state of the conversation. Another one is Dialogue Policy, which would base on the dialogue state to select an action. Either top-level policy to respond to the user's specific needs or a response suggested by a skill-specific low-level policy can activate the action. The last one is the Topic Manager, which depends on the dialogue turn to make classifier decide to switch a new topic. This behaviour is simulating that human change topic during a conversation.

1. **Contextual Query Understanding**

Contextual Query Understanding (CQU) contains below steps, First is named entity identification, which links all entity mentioned label to the entities saved in working memory of the state tracker and update new entities into the working memory. Next is Co-reference resolution which replaces all pronouns to the corresponding entity names. The last step is Sentence completion, which completes the sentence using contextual information if the sentence is not complete.

**Summary**

In summary, social chatbots have to behave like a human. Not only understand the user meaning and finish the task but also have a persona which is always reliable, sympathetic, affectionate and sense of humour. In these purposes, social chatbots have to contain IQ, EQ and personality. XiaoIce achieves this goal because of the system architecture. Primarily, the conversation engine layer helps that XiaoIce can understand what the user needs and response the proper action back. Moreover, this layer is the key to let chatbot obtain the personal characteristic.

[]<https://www.mitpressjournals.org/doi/pdf/10.1162/coli_a_00368>

[From Eliza to XiaoIce: challenges and opportunities with social chatbots]

### Task-oriented chatbot

**Introduction**

The majority of prior research has applied that Task-Oriented Chatbot aims to help the user to achieve their needs (Hussain S et.al, 2019). It means the task-oriented chatbot is designed for dealing with specific situations. For example, booking flight, hotel, accommodation, or scheduling an event, etc. The task-oriented chatbot is good at restricted domains. However, this kind of chatbots cannot process general knowledge or answer the question without their task domain. Instead, they are goal-oriented chatbots concentrated on serving people to accomplish a specific goal.

**Framework**

A series of recent studies has indicated that domain ontology is the primary frame of the recent task-based dialogue system (J. Daniel, J. Martin, 2018). Also, domain ontology is a knowledge structure that draws the sets of intentions the system can extract from user sentences. Additionally, the ontology determines one or more frames, each a collection of slots, and represents the values that each slot can take.

In 1977, Genial Understander System (GUS) introduced this influential frame-based architecture for travel planning (Bobrow et al., 1977). Furthermore, it has influenced most modern commercial digital assistants.

In GUS-style frame, the set of slots designate what the system needs to know. And the filler of each slot is restricted to a value of specific semantic type. The purpose of control architecture of frame-based dialogue systems is to collect the user intends and then process the associated action for the user. Most frame-based dialogue systems are hand-designed for the task by a dialogue designer.

One of the frameworks is using Natural Language Processing (NLU) to be the core of chat interaction. Handoyo et al. (2018) built a ticket chatbot by using Wit.AI, which is an application as one of the natural language interfaces for turning the sentence into structured data. Also, Wit.AI can detect particular intent word in sentences by a built-in block and allow the developer to define the customised entities.

**Summary**

In conclusion, the task-oriented chatbot needs to know the intent of the user sentences. And analysis these sentences to a specific entity. It will be filled into a slot or be detected as a special word. Through this procession, the chatbot executes a corresponding action which designed from developers.

[A survey on conversational agents/chatbots classification and design techniques]

## NLU

What is Natural Language Understanding? A more comprehensive description can be found in the paper from Dahlgren et al. (1991).

The Natural Language Understanding (NLU) is using a similar way which is human understanding to understand a natural language. The natural language is highly ambiguous and redundant. Because the same word may have many different meanings or the same meaning sentence can use a different way to express. Therefore, the NLU module is used to analyse this complicated structure and resolve its meaning layer by layer.

## Rasa

### Rasa Architecture

### Compare NLU framework

There are many different frameworks to develop chatbots on the internet like google Dialogflow and Rasa. This essay will compare Rasa and Dialogflow and explain why I use Rasa to build my chatbot.

(https://www.ideas2it.com/blogs/battle-of-the-bots-rasa-vs-google-dialogflow-vs-aws-lex/)

https://chatbotslife.com/dialogflow-vs-rasa-which-one-to-choose-206fb98b0e90

Lakshmanan (2020) mentions the meaningful difference between Rasa and Dialogflow. It is including model design, integration, and customisation. Here, propose main points influence this dissertation using Rasa.

**Intent/Entity Model**

Rasa is using the concept of a pipeline for intent and entity. They have adapted a recent state-of-the-art model – Bidirectional Encoder Representation from Transformer (BERT) into the pipeline, which helps developers to build a better model. Moreover, Rasa supports another advantage of combining our customise model into the pipeline for any task.

Dialogflow is a solid platform which contains good models and pre-trained entities. However, it does not support custom models.

**Conversation Flow**

Rasa is based on the data or conversation flow from given and processes by a transformer model (recent state-of-the-art model). Through this method, it can help developers to create their flow generically.

Dialogflow has dialogue management which drives the conversation by the context provided for intents. Also, it is slightly rule-based and not as flexible.

**Model Tweaking**

Rasa is open-source, which helps the developer base on their needs to estimate and configure the model.

In Dialogflow, models are closed base. Developers cannot modify or evaluate the models.

**API Integration**

Any external API such as SQL, Graph DB or our original API can be integrated into Rasa.

Dialogflow provides API integration as well. But Rasa supports the Knowledge graph integration which is helpful to let the developer process a better conversation.

**Deployment**

Rasa can be set up at the could-based and on-premise as well.

Dialogflow is cloud-based platforms.

**Customization**

The developer can use Rasa to customise the business logic to implement, model, deployment, and integration. Furthermore, the developer can use SDK which Rasa provides to execute your custom logic.

Dialogflow can let developer customise training data and the input rule-based dialogue flow. But it does not support any customisation based on the business needs.

**Summary**

This is an essential finding in the understanding of the difference between these two frameworks. We could know that Rasa is more flexible to function customised to implement into the chatbot. Additionally, using these customisation supports can help the program be better and more general. Although the developer requires technical knowledge and python coding skill, Rasa still provides better performance than Dialogflow.

1. <https://www.imdb.com/title/tt1446714/> [↑](#footnote-ref-1)
2. <https://en.wikipedia.org/wiki/Watson_(computer)> [↑](#footnote-ref-2)
3. <https://www.theverge.com/2018/5/8/17332070/google-assistant-makes-phone-call-demo-duplex-io-2018> [↑](#footnote-ref-3)