UNIT III

CHATBOT

Chatbots

- Chatbots are computer-programmed services that can interact like humans through a chat interface, both textually and auditorily,….etc.
- Also known as talkbots, smartbots, bots, chatterbots, or interactive agents, Virtual Assistant, Chatterbox,..etc. these programs are intended to communicate with the user and behave as if a real human is behind the conversation.

- Chatbots have a wide array of uses in different areas such as business, education, information, and entertainment,...etc.

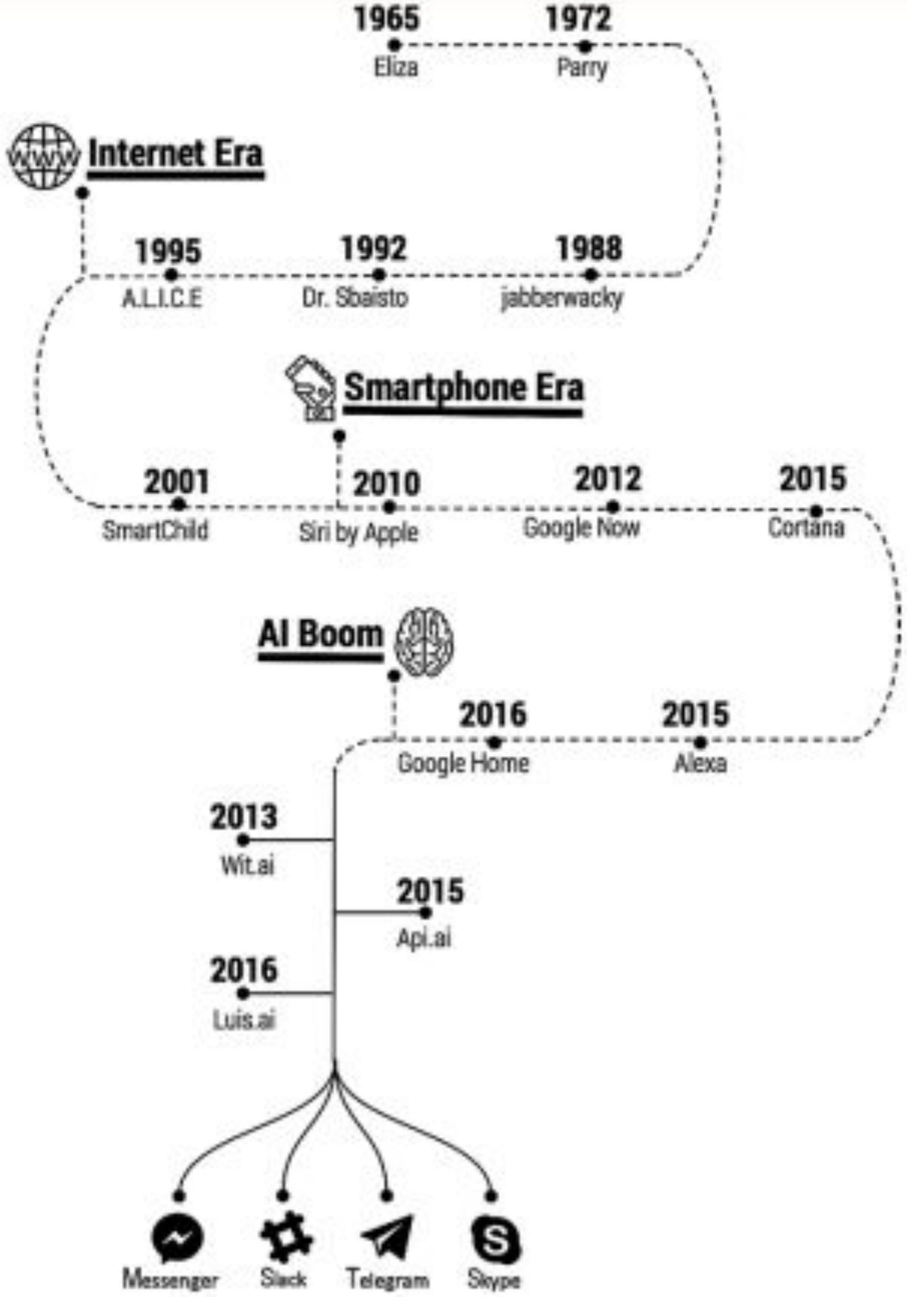
- **First used in online interactive games** and **instant messaging**, these bots are now being classified based on their usage in communication, analytics, design, travel, sports, shopping, personal, food, and health.

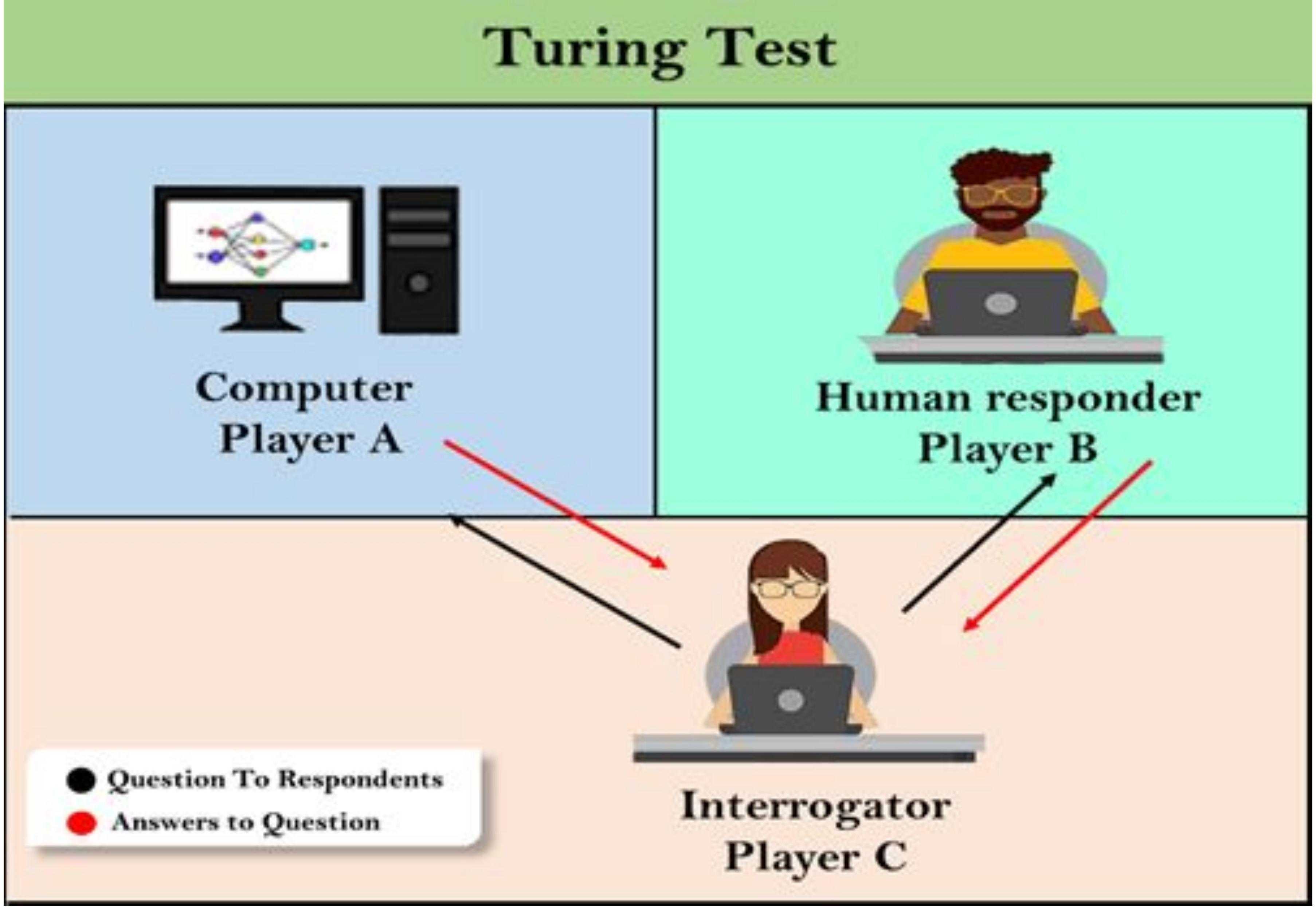
- Airlines and e-commerce companies have also employed chatbots on their websites to provide information and answer inquiries to their clients and customers.

For Example

- Instead of browsing the Amazon website, you might have a conversation with Alexa to find and order the product you need. Alexa is a voice-controlled intelligent personal assistant service that is part of the Amazon Echo smart speaker.

- Alexa might converse with you in the same way you might with **a shop assistant in a retail store. It could provide personalized recommendations, and assist with on-the-spot purchasing. Alexa could also automatically connect to your existing customer data and confirm your identity.**

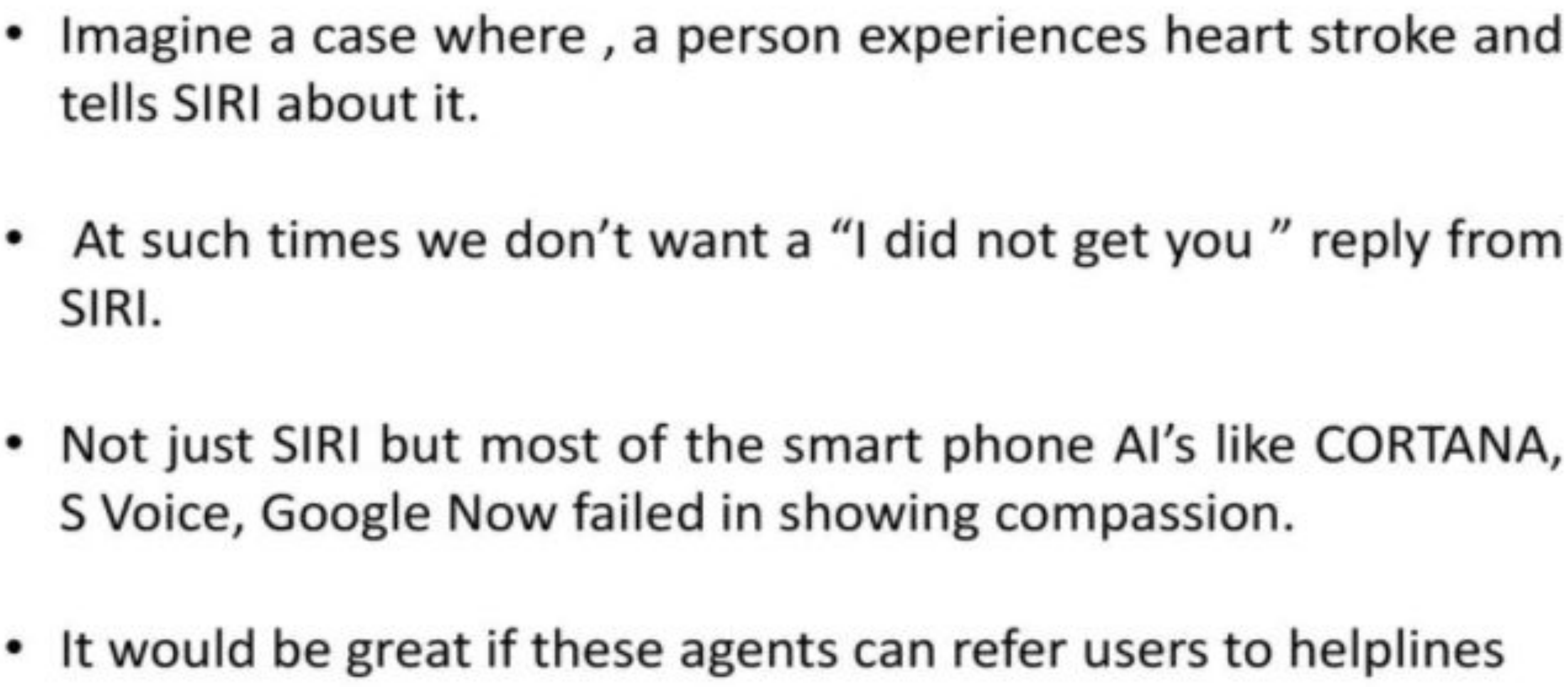

1965
Eliza
1972
Parry
Internet Era
1995
A.L.I.C.E
1992
Dr. Sbaisto
1988
jabberwacky
Smartphone Era
2001
SmartChild
2010
Siri by Apple
2012
Google Now
2015
Cortana
AI Boom
2016
Google Home
2015
Alexa
2013
Wit.ai
2015
Api.ai
2016
Luis.ai
Messenger
Slack
Telegram
Skype



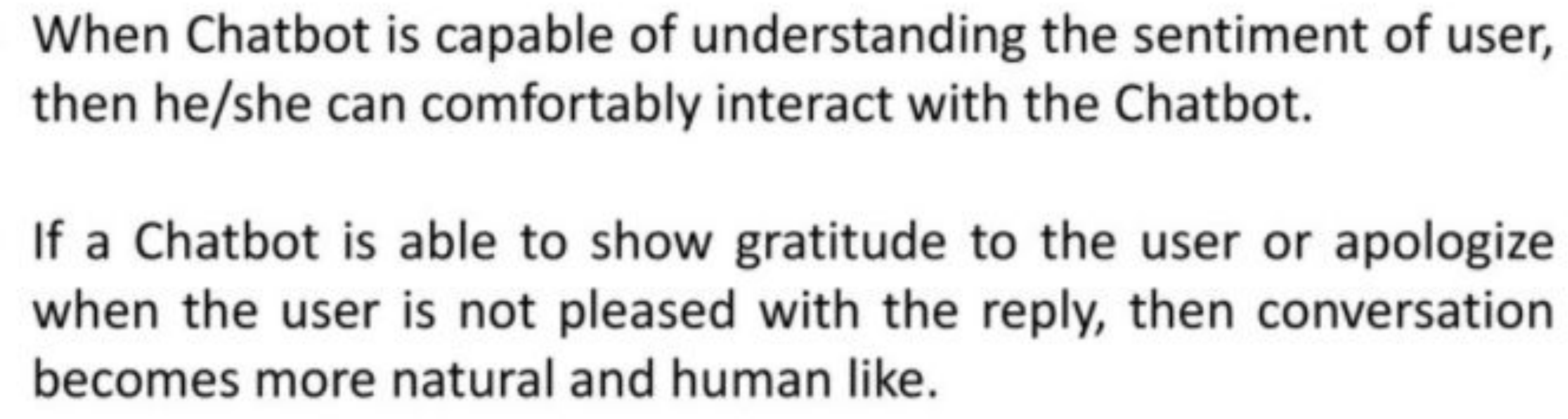
Turing Test was introduced by Turing in his 1950 paper, "Computing Machinery and Intelligence," which considered the question, "Can Machine think?"

- Features required for a machine to pass the Turing test:
- **Natural language processing:** NLP is required to communicate with Interrogator in general human language like English.
- **Knowledge representation:** To store and retrieve information during the test.
- **Automated reasoning:** To use the previously stored information for answering the questions.
- **Machine learning:** To adapt new changes and can detect generalized patterns.
- **Vision (For total Turing test):** To recognize the interrogator actions and other objects during a test.
- **Motor Control (For total Turing test):** To act upon objects if requested.

The Rise of Chatbots



- Imagine a case where , a person experiences heart stroke and tells SIRI about it.
- At such times we don’t want a “I did not get you ” reply from SIRI.
- Not just SIRI but most of the smart phone AI’s like CORTANA, S Voice, Google Now failed in showing compassion.
- It would be great if these agents can refer users to helplines



When Chatbot is capable of understanding the sentiment of user, then he/she can comfortably interact with the Chatbot.

If a Chatbot is able to show gratitude to the user or apologize when the user is not pleased with the reply, then conversation becomes more natural and human like.

So is it Possible to apply sentiment analysis on Chatbot for such kind of Applications

- 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

NLP in the cloud

- Conversational AI is a central sub-field of Natural Language Processing that makes it possible for a human to have a conversation with a machine.
- Everytime the human says or asks something to the AI, the whole conversation history is sent too, so the AI can have the context in memory and make relevant responses.
- For example they can detect search documents, understand the customer tone and adapt their own tone (anger, joy, sarcasm...).

- There are several acronyms in the world of automation and AI that are relevant for understanding chatbots. Here are **four key terms** that you need to know:
- **NLP, or Natural Language Processing,** is a branch of AI that helps computers read and understand natural human language. Its main goal is to improve human-machine communication.
- **NLU, or Natural Language Understanding,** is a branch of NLP. It is all about machine reading comprehension and making sure the machine understands the text’s actual meaning. In more scientific terms, NLU takes place when the machine converts the user’s inputted data (what they’re saying) into a logical form that the computer's algorithms understand.

- **NLG, or Natural Language Generation**, is another subset of NLP, which is essentially NLU in reverse: the machine generates a logical response which it then converts to a natural language response that a human reader can easily understand.

- **NLI, or Natural Language Interaction**, another branch of NLP, refers to the communication that takes place between humans and machines—the process of translating from programming language to human language, and back again, like a very complex Google Translate for machines.

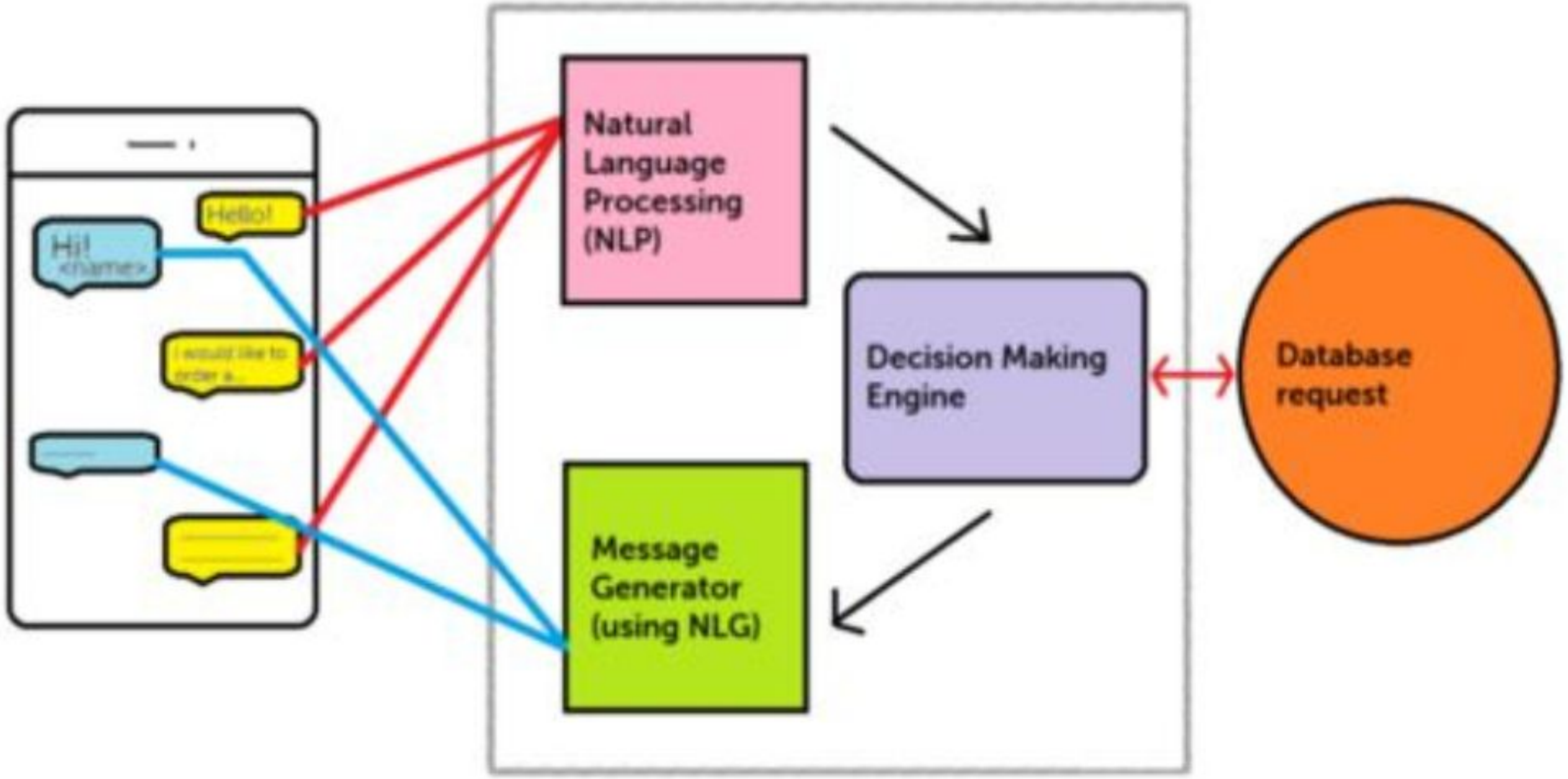
NLP in the cloud

- Until recently, Chatbots have been very limited. But with the creation of huge modern models like GPT-3(Generative Pre-trained Transformer 3) and GPT-J, it is now possible to easily create advanced chatbots that are both fluent and relevant.
- GPT-J is the most advanced open-source NLP model which is the best GPT-3 alternative
- This model is so big that it can adapt to many situations, and perfectly sounds like a human

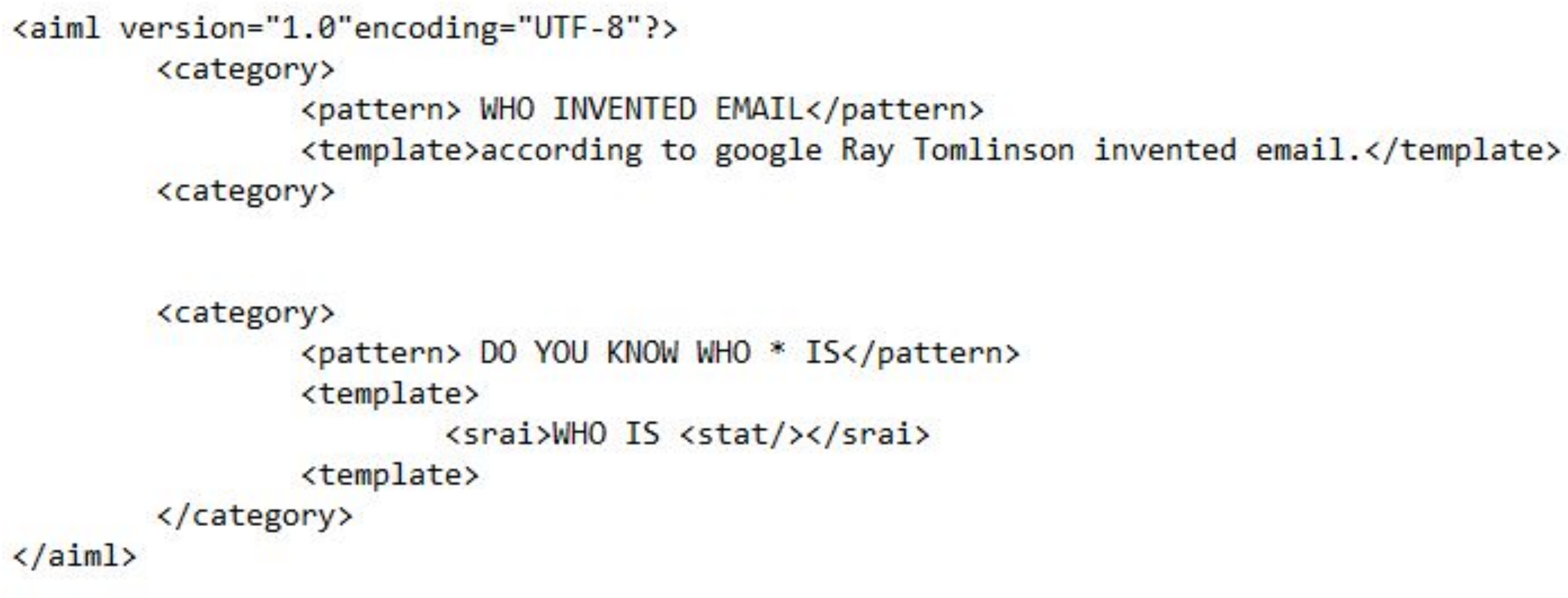
- The NLP Engine is the central component of the chatbot architecture. It interprets what users are saying at any given time and turns it into organized inputs that the system can process.
- The NLP engine uses advanced machine learning algorithms to determine the user’s intent and then match it to the bot’s supported intents list.

- Architecture & Work Methods of Chatbots.
- Pattern Matches: Bots utilize pattern matches to group the text and it produces an appropriate response from the clients. “Artificial Intelligence Markup Language (AIML), is a standard structured model of these Patterns.

Architecture & Work Methods of Chatbots


Hello!
Hi! <name>
I would like to order a...
Natural Language Processing (NLP)
Decision Making Engine
Database request
Message Generator (using NLG)

A simple example of Pattern matching is;



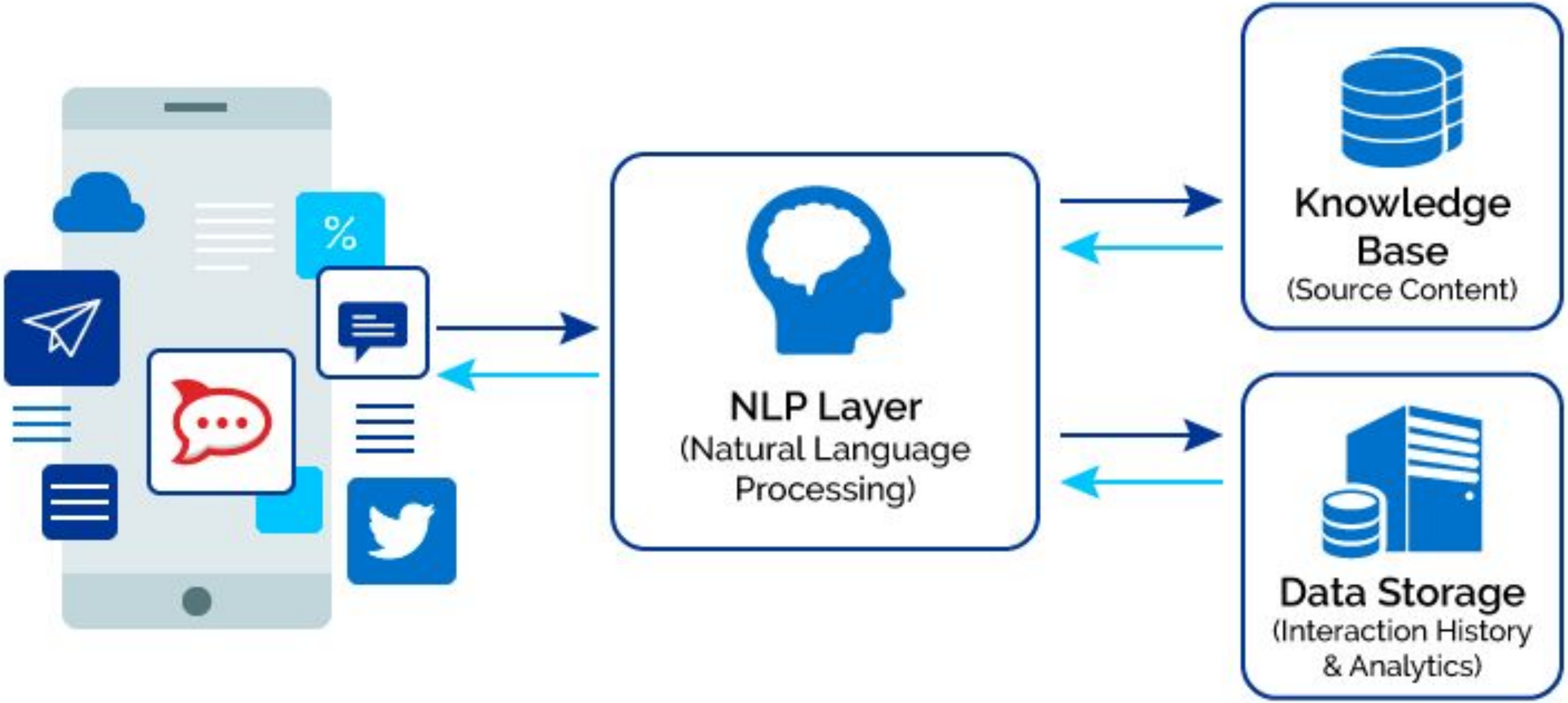
```
<aiml version="1.0"encoding="UTF-8"?>
        <category>
                <pattern> WHO INVENTED EMAIL</pattern>
                <template>according to google Ray Tomlinson invented email.</template>
        <category>


        <category>
                <pattern> DO YOU KNOW WHO * IS</pattern>
                <template>
                        <srai>WHO IS <stat/></srai>
                <template>
        </category>
</aiml>
```

Then the machine gives the following output:

- Human: Who invented the email?
- Robot: According to Google, Ray Tomlinson invented email.

The Chatbot knows the appropriate answer because her or his name is in the related pattern.

- For every sort of question, a remarkable pattern must be accessible in the database to give a reasonable response.
- With a number of pattern combinations, it makes a hierarchical structure. We utilize algorithms to lessen the classifiers and produce the more reasonable structure


NLP Layer
(Natural Language
Processing)
Knowledge
Base
(Source Content)
Data Storage
(Interaction History
& Analytics)
%

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
- The Backend

The Messaging Channel

- There are a lot of messaging channels out there
- 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
- Text classification is a machine learning technique that assigns a set of predefined categories to open-ended text. ... Text classification is one of the fundamental tasks in natural language processing with broad applications such as sentiment analysis, topic labeling, spam detection, and intent detection.

Here’s a more specific example. When talking to a customer serviceagent, 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.

Using a framework versus building your own

There are many chatbot frameworks available, including API.ai, Wit.ai, Microsoft Bot Framework, and IBM Watson Conversation.

Microsoft Bot Framework:

They provide all the ingredients to build, connect, deploy and manage intelligent bots. Bots built on Microsoft Bot Framework can be deployed on any website, app and also on messengers like Slack, Facebook Messenger etc.

Facebook Bot Engine (Wit.ai):

Wit.ai provides APIs that can process Natural Human Language, which can be in the form of both text and voice input.

Dialogflow (Api.ai): Api.ai allows you to create engaging voice and text-based conversational interfaces via which your users can interact with your product.

Based on a framework, the backend of chatbot consist of three main parts:

- Intents
- Entities
- 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.

Utterance:

Anything the user says. For example, if a user types “show me yesterday’s financial news”, the entire sentence is the utterance.

Intent:

An intent is the user’s intention.

For example, if a user types “show me yesterday’s financial news”, the user’s intent is to retrieve a list of financial headlines. Intents are given a name, often a verb and a noun, such as “showNews”.

Entity:

An entity modifies an intent.

For example, if a user types “show me yesterday’s financial news”, the entities are “yesterday” and “financial”. Entities are given a name, such as “dateTime” and “newsType”. Entities are sometimes referred to as slots.

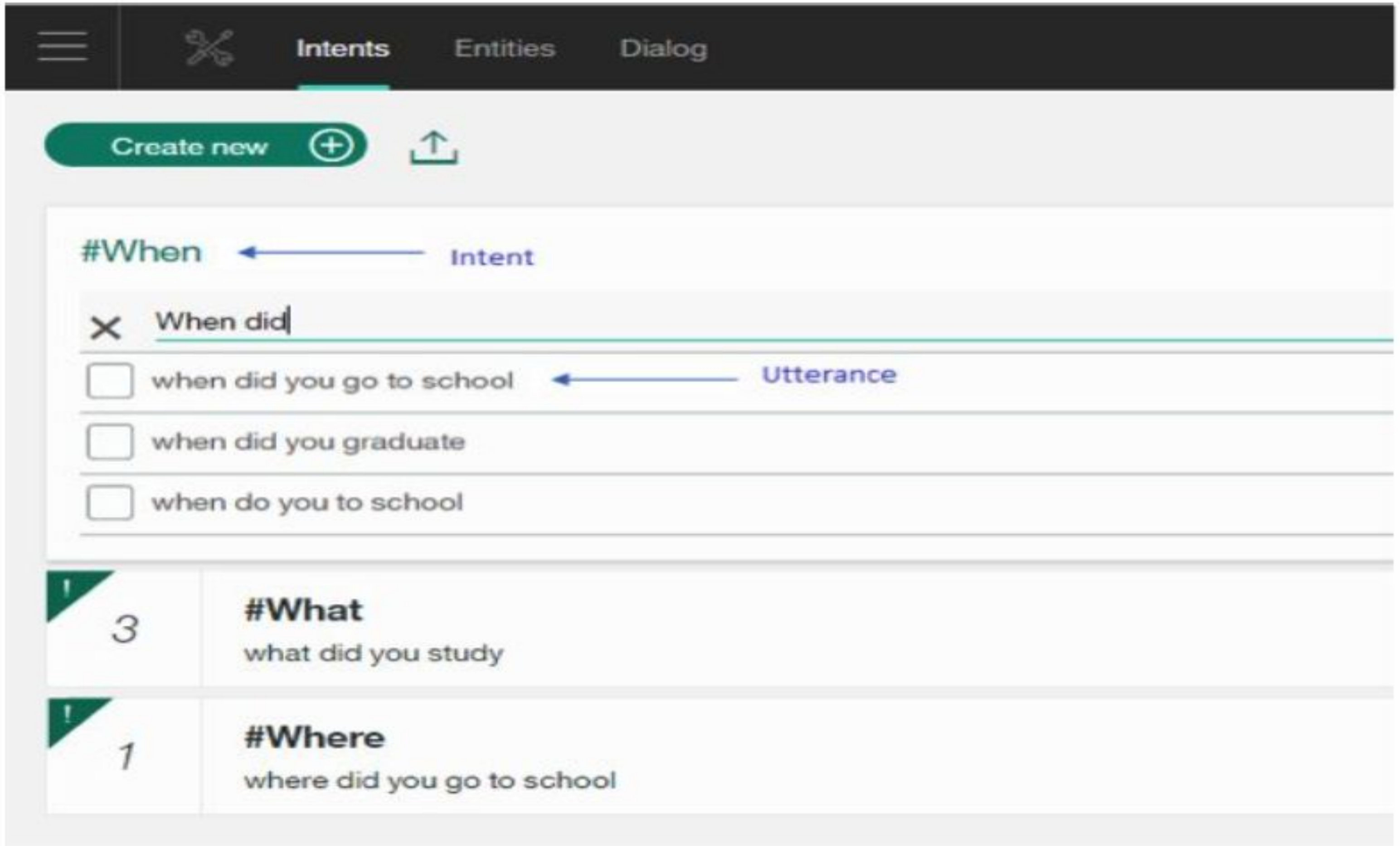
Step 1: Define Intents

Step 1: Define Intents

- Intents are the objectives of every bot.
- A chatbot that answers and questions about you has the broad objective of providing information about a human being.
- Therefore, it helps to think of this information as it pertains to where,

  what, and when.

Intents



#When

When did

□ when did you go to school

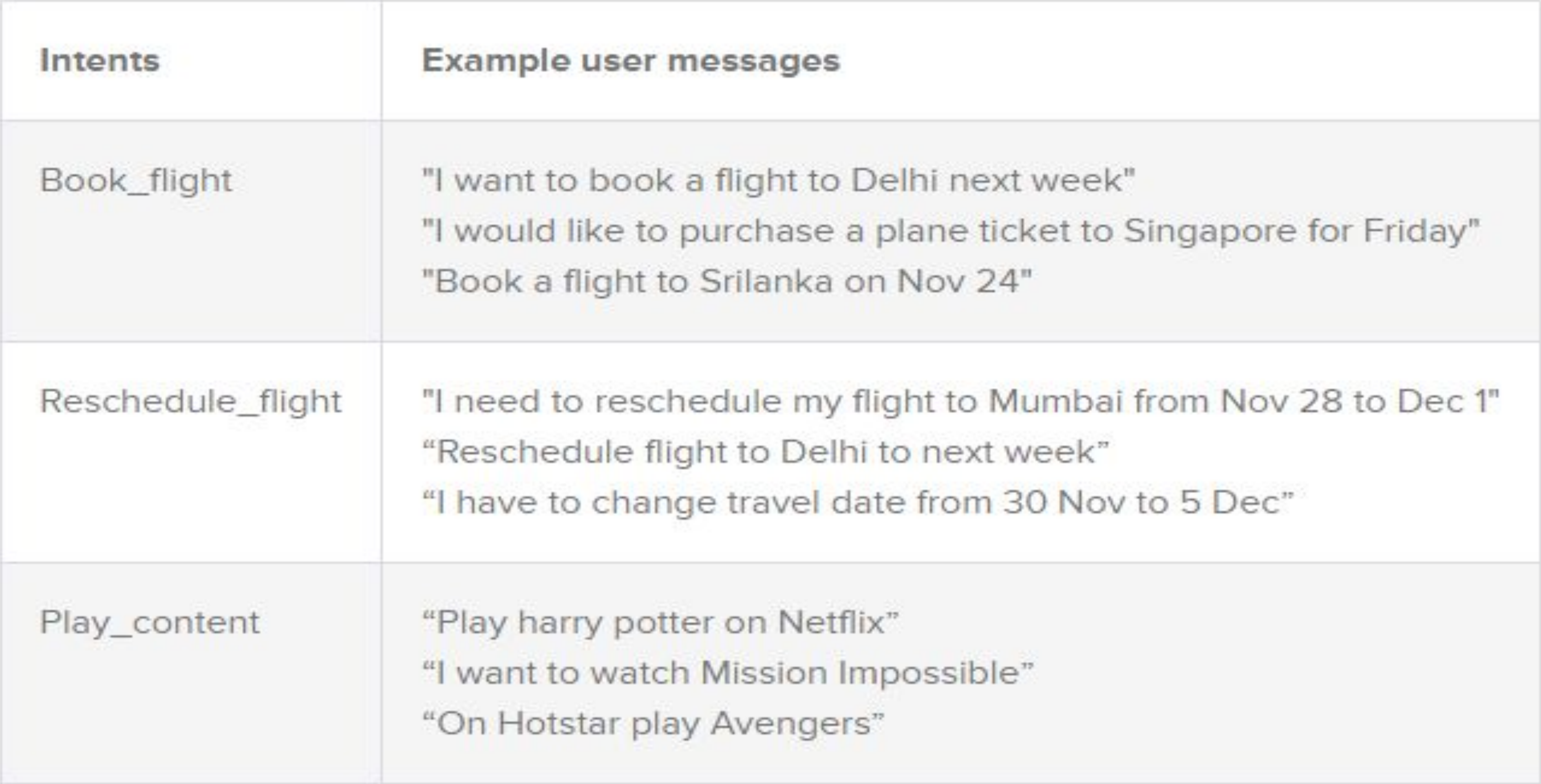
□ when did you graduate

□ when do you to school

3 #What
what did you study

1 #Where
where did you go to school

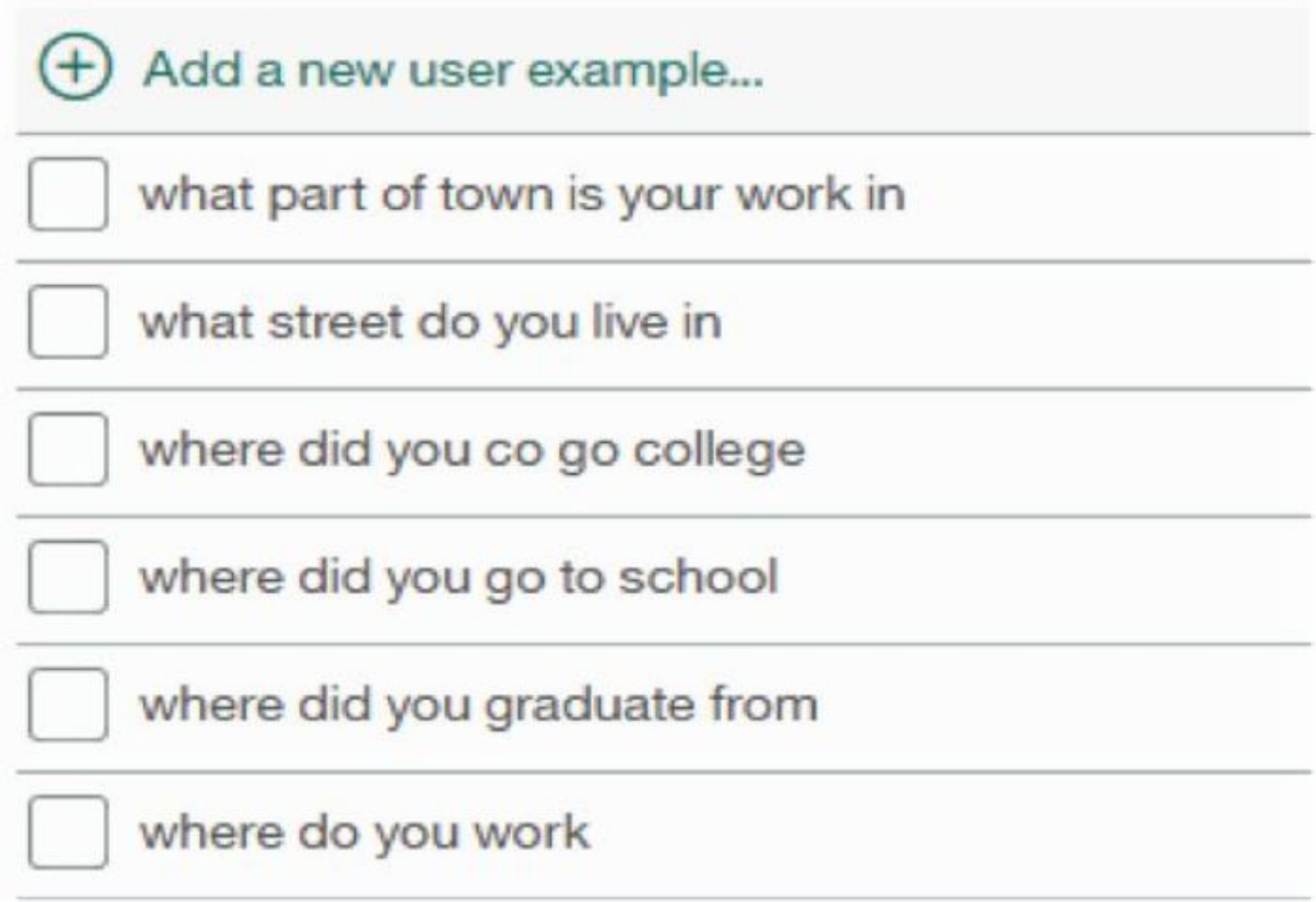
INTENT



Intents	Example user messages
Book_flight	"I want to book a flight to Delhi next week" "I would like to purchase a plane ticket to Singapore for Friday" "Book a flight to Srilanka on Nov 24"
Reschedule_flight	"I need to reschedule my flight to Mumbai from Nov 28 to Dec 1" “Reschedule flight to Delhi to next week” “I have to change travel date from 30 Nov to 5 Dec”
Play_content	“Play harry potter on Netflix” “I want to watch Mission Impossible” “On Hotstar play Avengers”

Intent Example

- Say you are building a chatbot for a store that sells fashion-related products.
- Before you start building the chatbot, you need to keep in mind what actions your chatbot will be able to perform.
- In this case, you would want your chatbot to respond to the user with appropriate textual and visual information when a user wants to see the products that the store sells by saying,
- for example, "I want to buy a red shirt."
- Also, when the user sends the chatbot messages such as "Do you have any stores in Berlin?" it should be able to locate all the nearby shops for that particular location.
- To perform each of these actions, the chatbot needs to decide whether the user is looking for a product or store location from the chat message.
- So, you can say your chatbot will have two intents: product lookup and location lookup.



□ what part of town is your work in

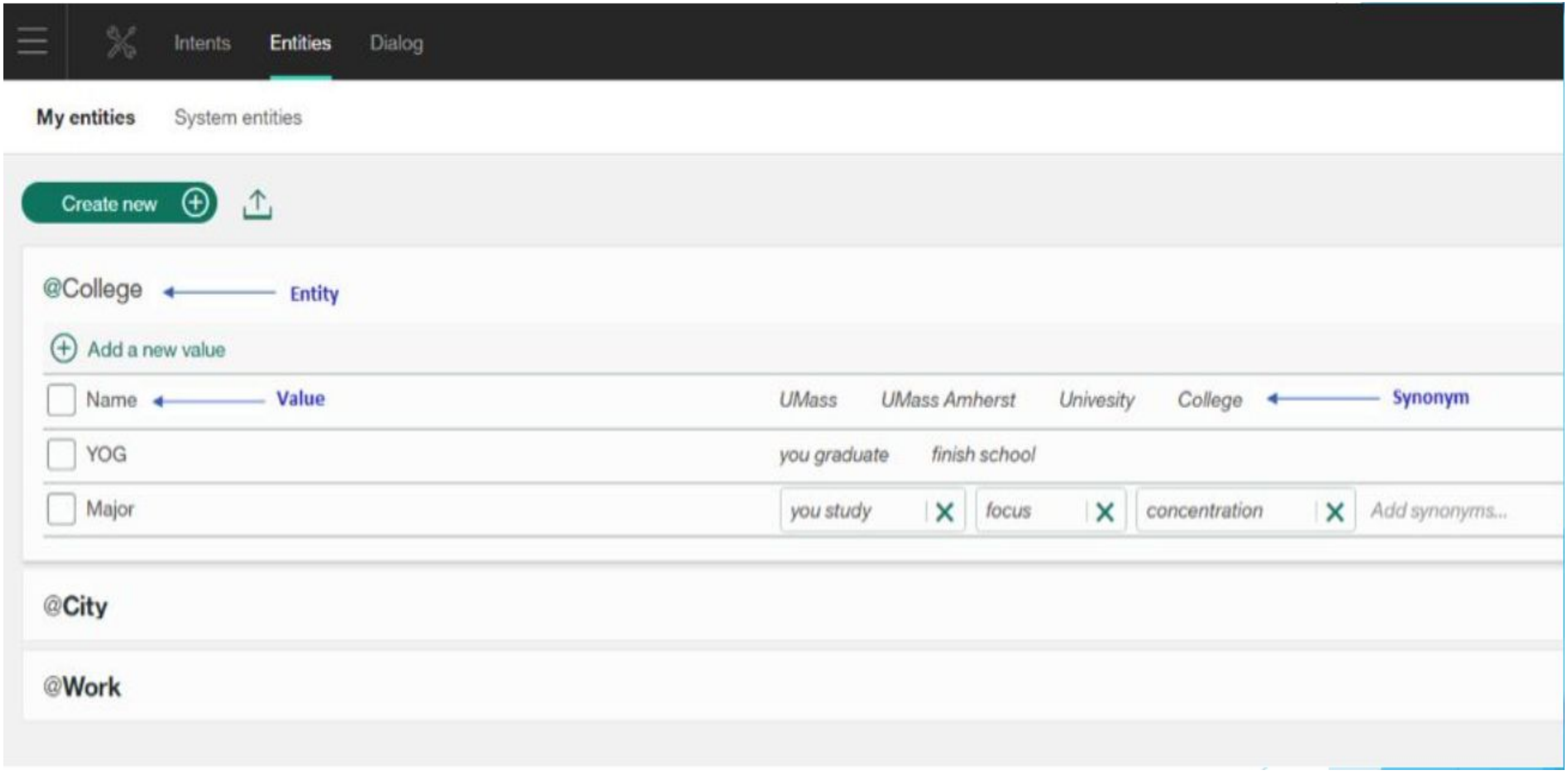
□ what street do you live in

□ where did you co go college

□ where did you go to school

□ where did you graduate from

□ where do you work



@College ← Entity

□	Name ← Value	UMass	UMass Amherst	Univesity	College ← Synonym
□	YOG	you graduate	finish school		
□	Major	you study ✕	focus ✕	concentration ✕	Add synonyms...

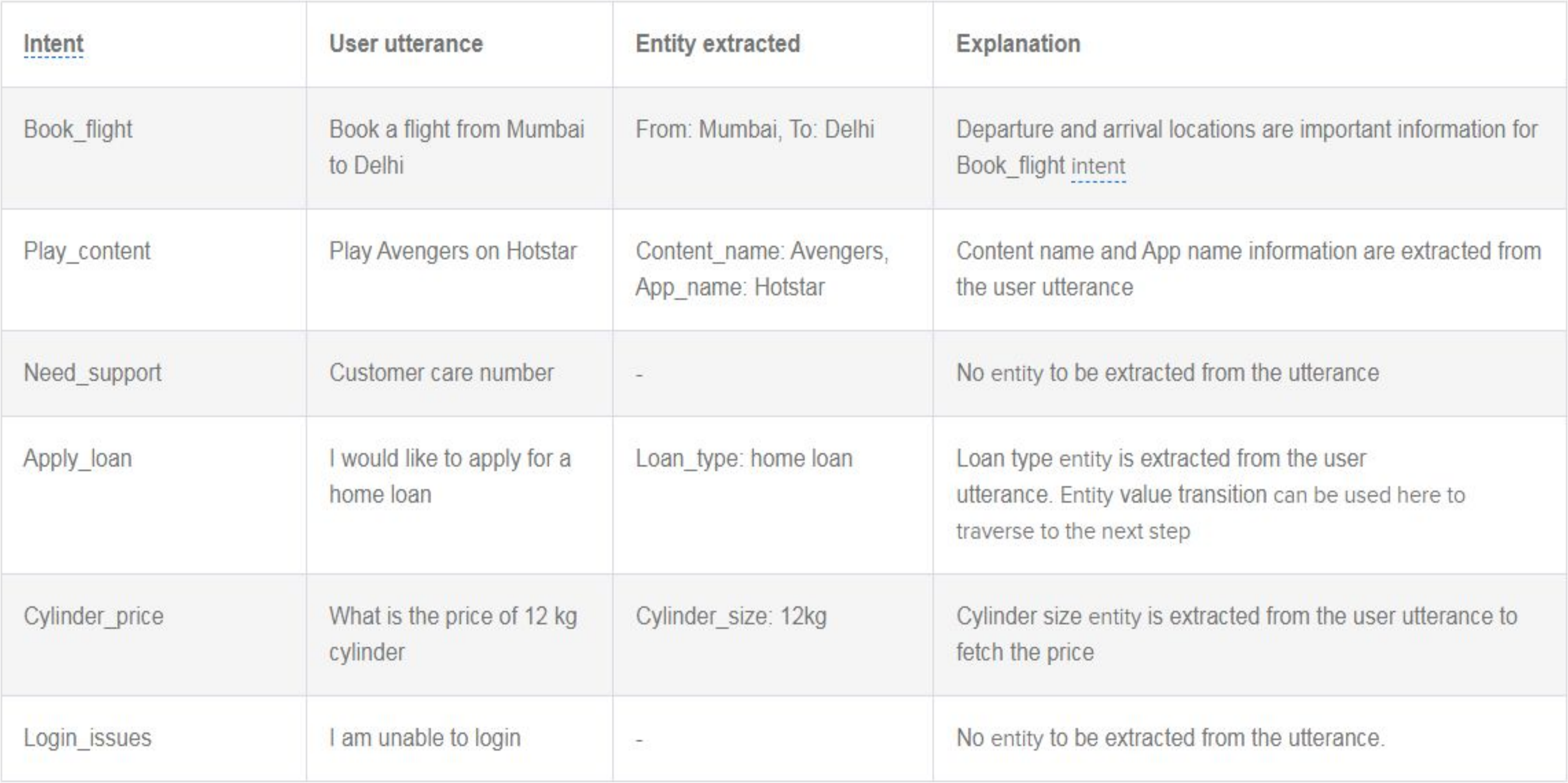
@City

@Work

Entity Example

- Entities are the important keywords/phrases that your chatbot looks for in a user message.
- These entities help the chatbot identify the subject of the conversation and deliver targeted information to the user, providing a better experience.
- Imagine when you message a chatbot with “I want to buy shirts” and it understands that you’re looking to buy fashion products but not able to “identify” what you are trying to buy, thus giving you information with all kinds of products that you are not interested in

Entity&Intents



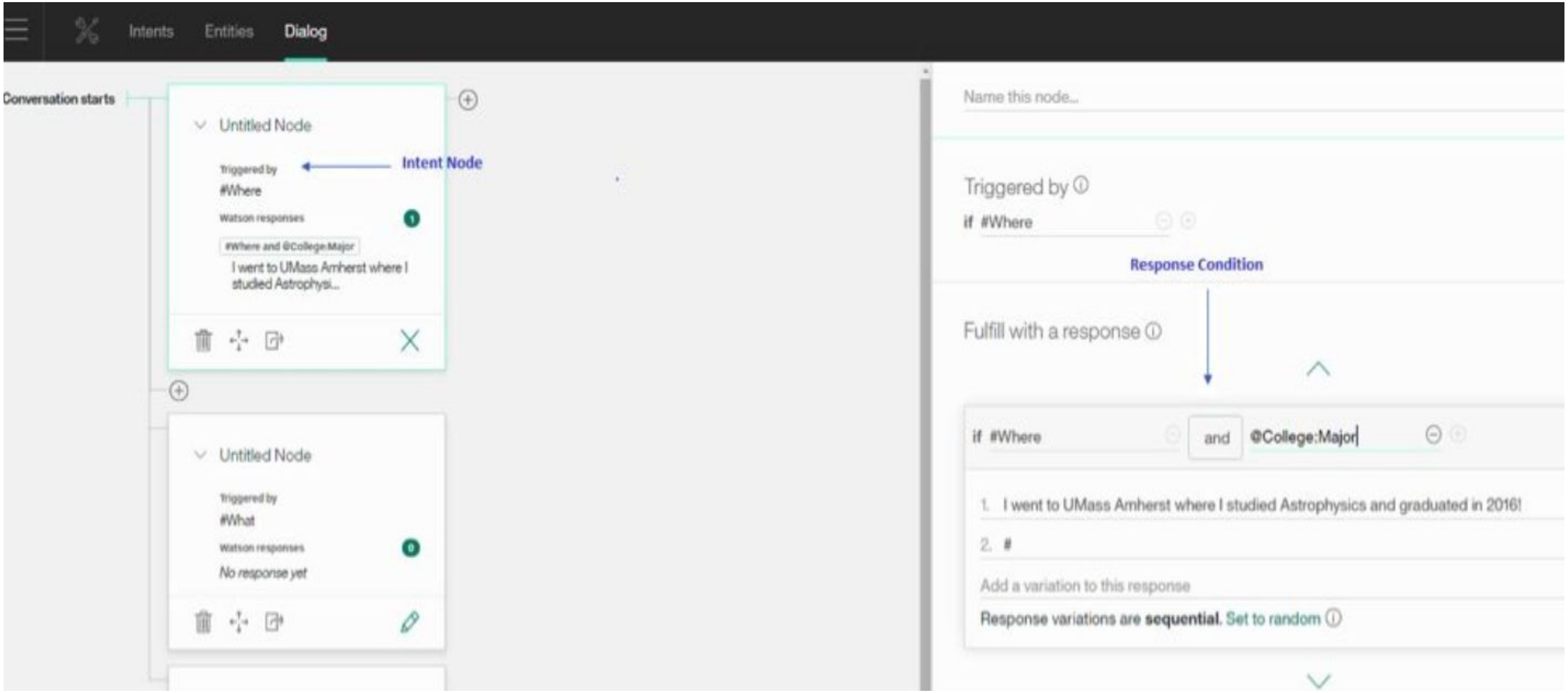
Intent	User utterance	Entity extracted	Explanation
Book_flight	Book a flight from Mumbai to Delhi	From: Mumbai, To: Delhi	Departure and arrival locations are important information for Book_flight intent
Play_content	Play Avengers on Hotstar	Content_name: Avengers, App_name: Hotstar	Content name and App name information are extracted from the user utterance
Need_support	Customer care number	-	No entity to be extracted from the utterance
Apply_loan	I would like to apply for a home loan	Loan_type: home loan	Loan type entity is extracted from the user utterance. Entity value transition can be used here to traverse to the next step
Cylinder_price	What is the price of 12 kg cylinder	Cylinder_size: 12kg	Cylinder size entity is extracted from the user utterance to fetch the price
Login_issues	I am unable to login	-	No entity to be extracted from the utterance.

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.

Dialog



Untitled Node

Triggered by
#Where

Watson responses 1

#Where and @College:Major
I went to UMass Amherst where I studied Astrophysi...

Untitled Node

Triggered by
#What

Watson responses 0

No response yet

Name this node...

Triggered by ⓘ

if #Where

Fulfill with a response ⓘ

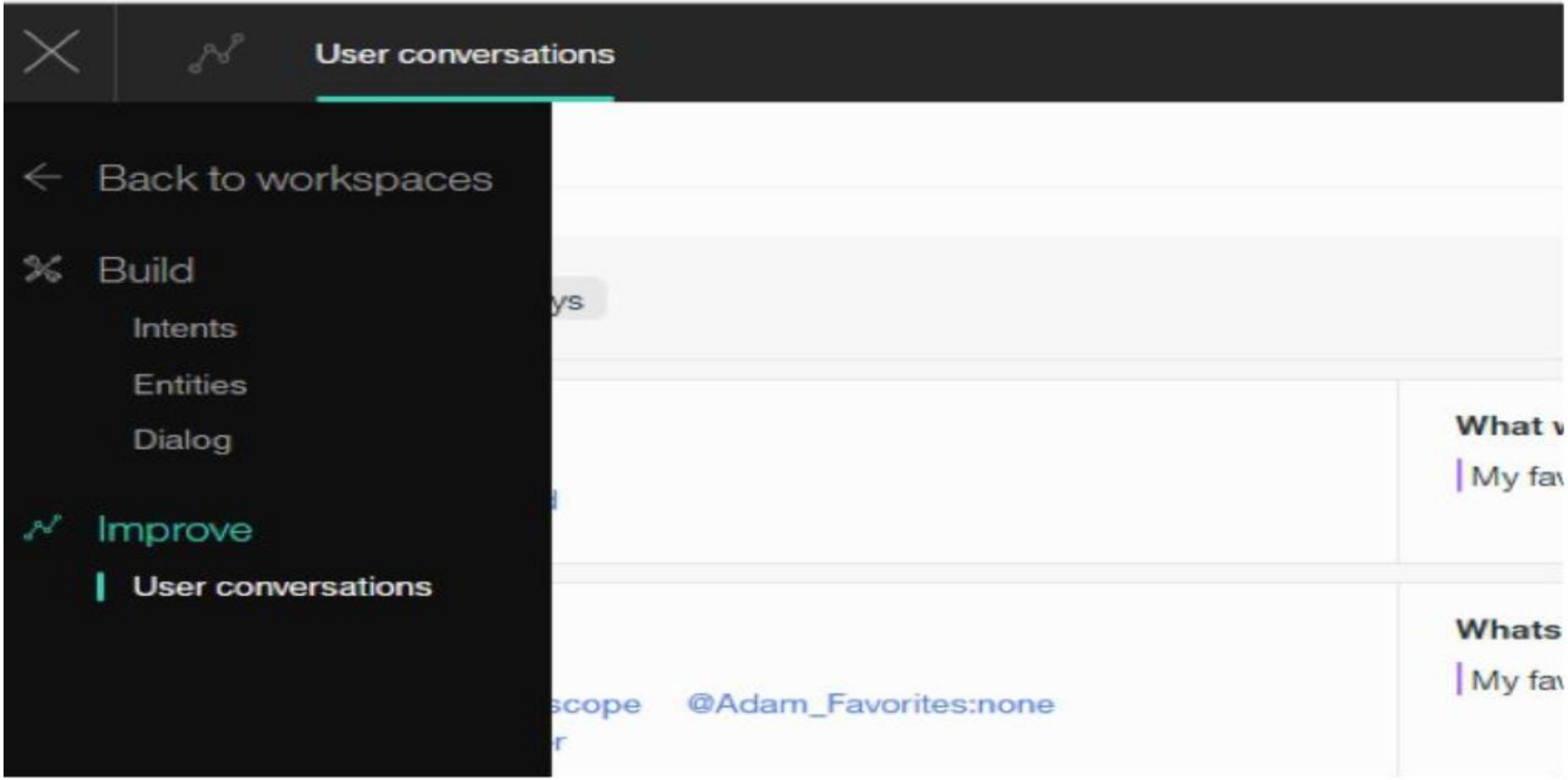
if #Where and @College:Major

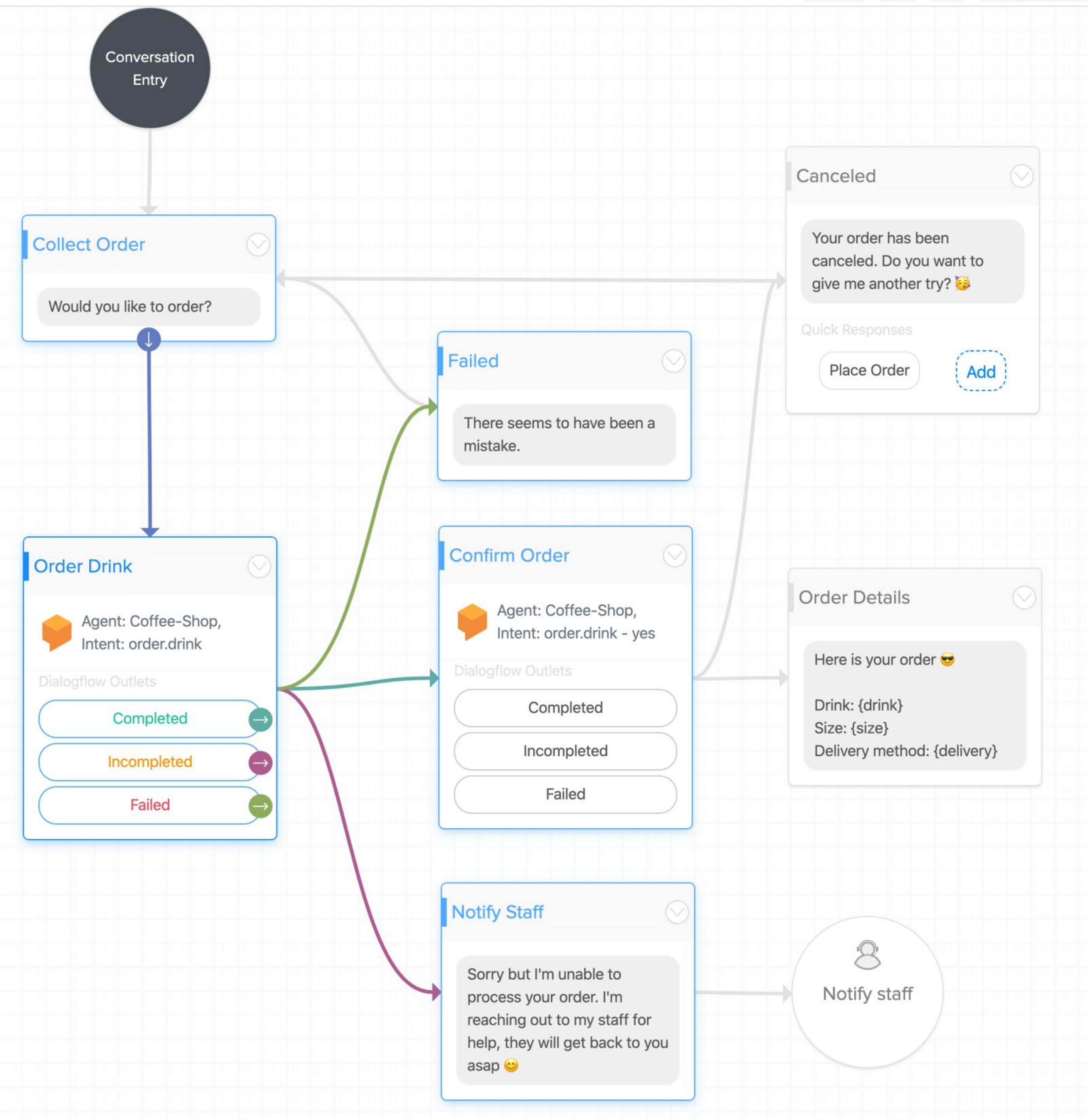
1. I went to UMass Amherst where I studied Astrophysics and graduated in 2016!
2. #

Add a variation to this response

Response variations are **sequential.** Set to random ⓘ

Training and Improving





Would you like to order?

Agent: Coffee-Shop, Intent: order.drink

There seems to have been a mistake.

Agent: Coffee-Shop, Intent: order.drink - yes

Sorry but I'm unable to process your order. I'm reaching out to my staff for help, they will get back to you asap 😊

Your order has been canceled. Do you want to give me another try? 🥳

Here is your order 😎

Drink: {drink}
Size: {size}
Delivery method: {delivery}

• Types of Chatbots

Chatbots can be classified into two different types, based on how they are built:

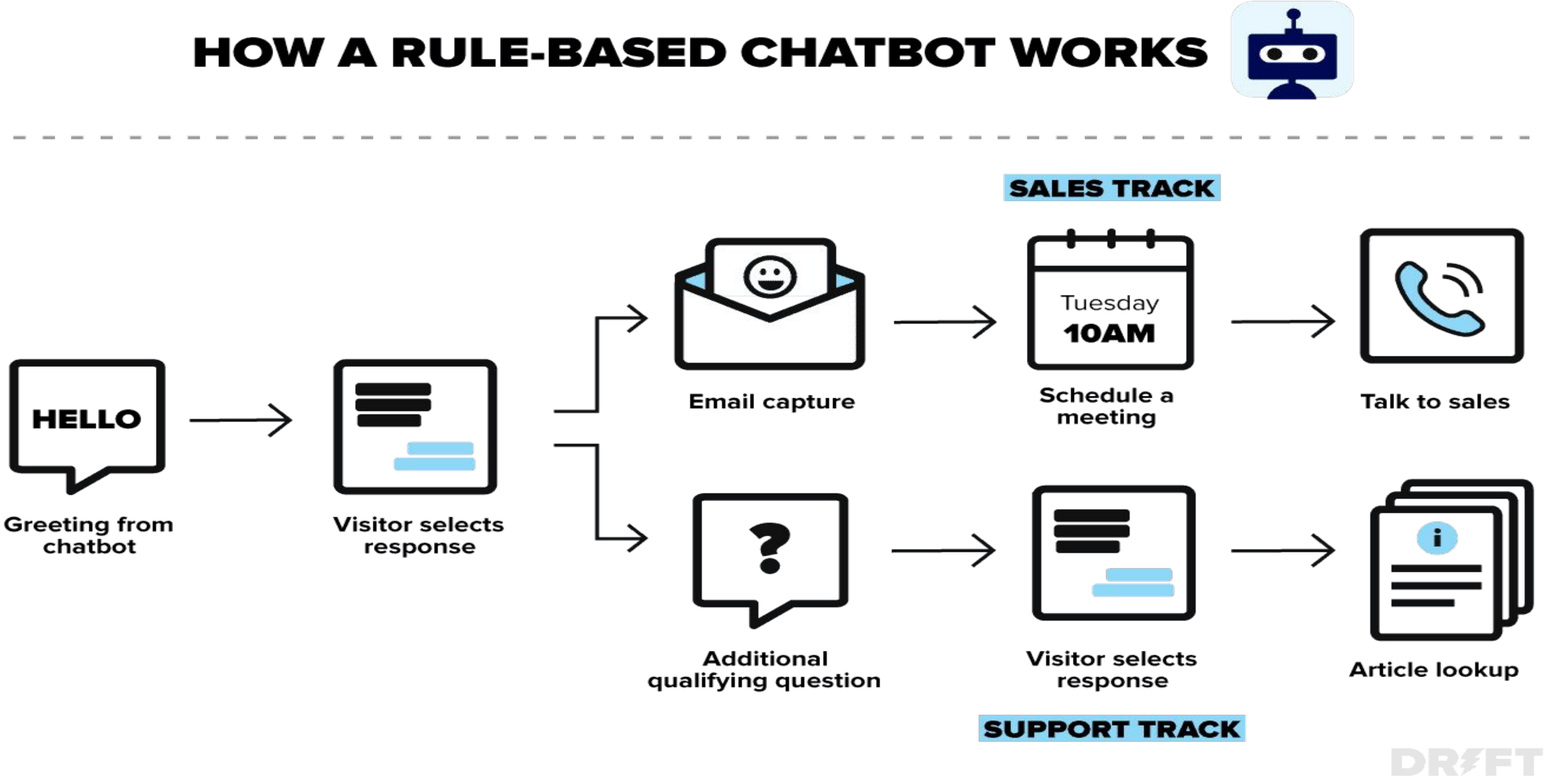
Rule-Based Chatbots

- Rule-based chatbots are pretty straight forward. They are provided with a database of responses and are given a set of rules that help them match out an appropriate response from the provided database.

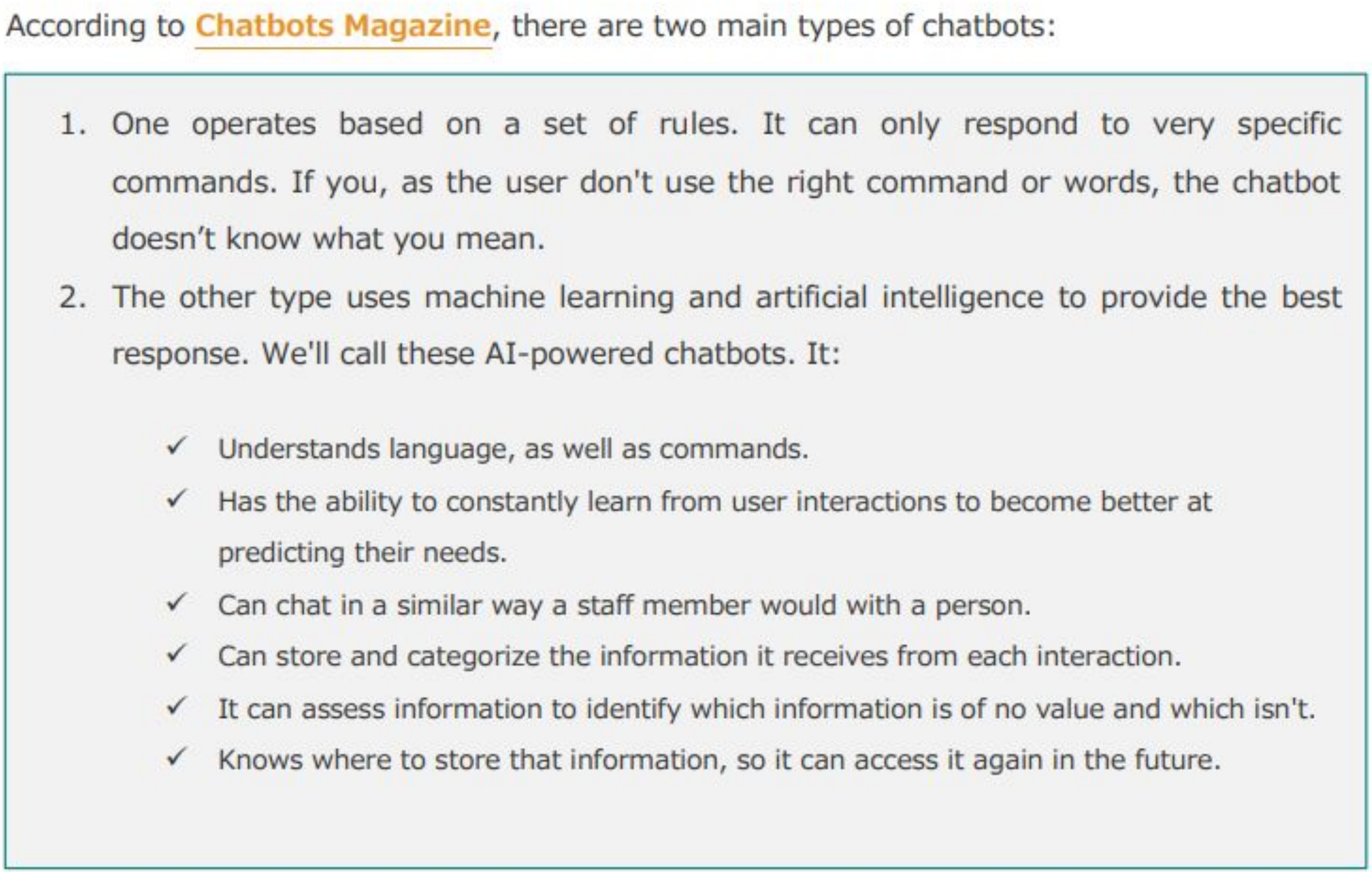
AI-Based Chatbots

- With the rise in the use of machine learning in recent years, a new approach to building chatbots has emerged. Using artificial intelligence, it has become possible to create extremely intuitive and precise chatbots tailored to specific purposes.

Rule based Chabot



The Two Types of Chatbots



According to **Chatbots Magazine**, there are two main types of chatbots:

1. One operates based on a set of rules. It can only respond to very specific commands. If you, as the user don't use the right command or words, the chatbot doesn’t know what you mean.
2. The other type uses machine learning and artificial intelligence to provide the best response. We'll call these AI-powered chatbots. It:

   ✓ Understands language, as well as commands.

   ✓ Has the ability to constantly learn from user interactions to become better at predicting their needs.

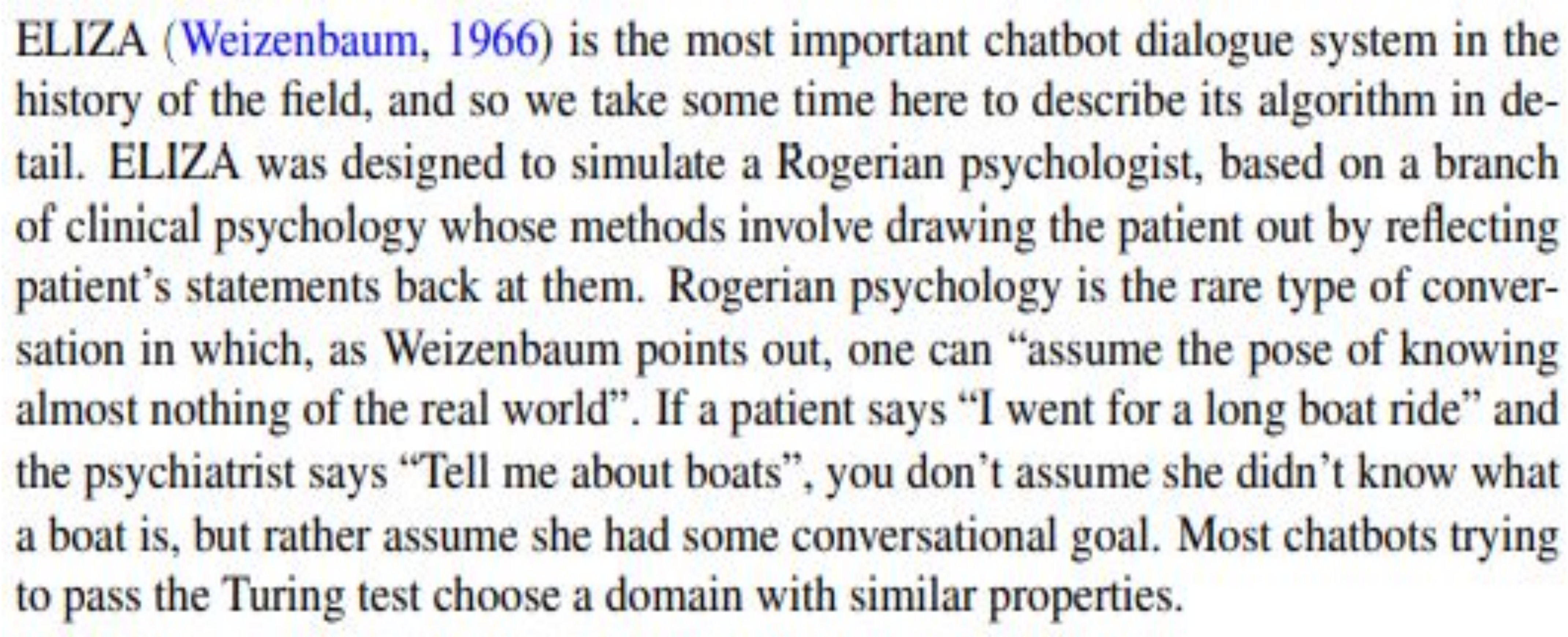
   ✓ Can chat in a similar way a staff member would with a person.

   ✓ Can store and categorize the information it receives from each interaction.

   ✓ It can assess information to identify which information is of no value and which isn't.

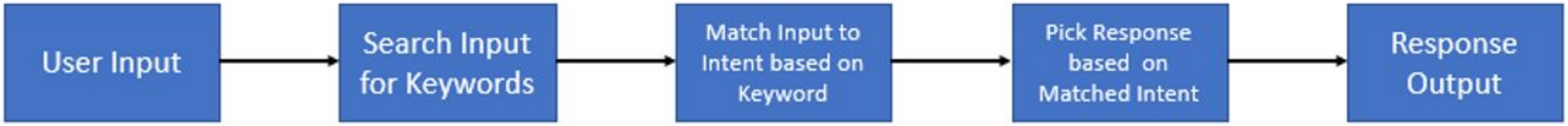
   ✓ Knows where to store that information, so it can access it again in the future.

Rule-based chatbots: ELIZA and PARRY

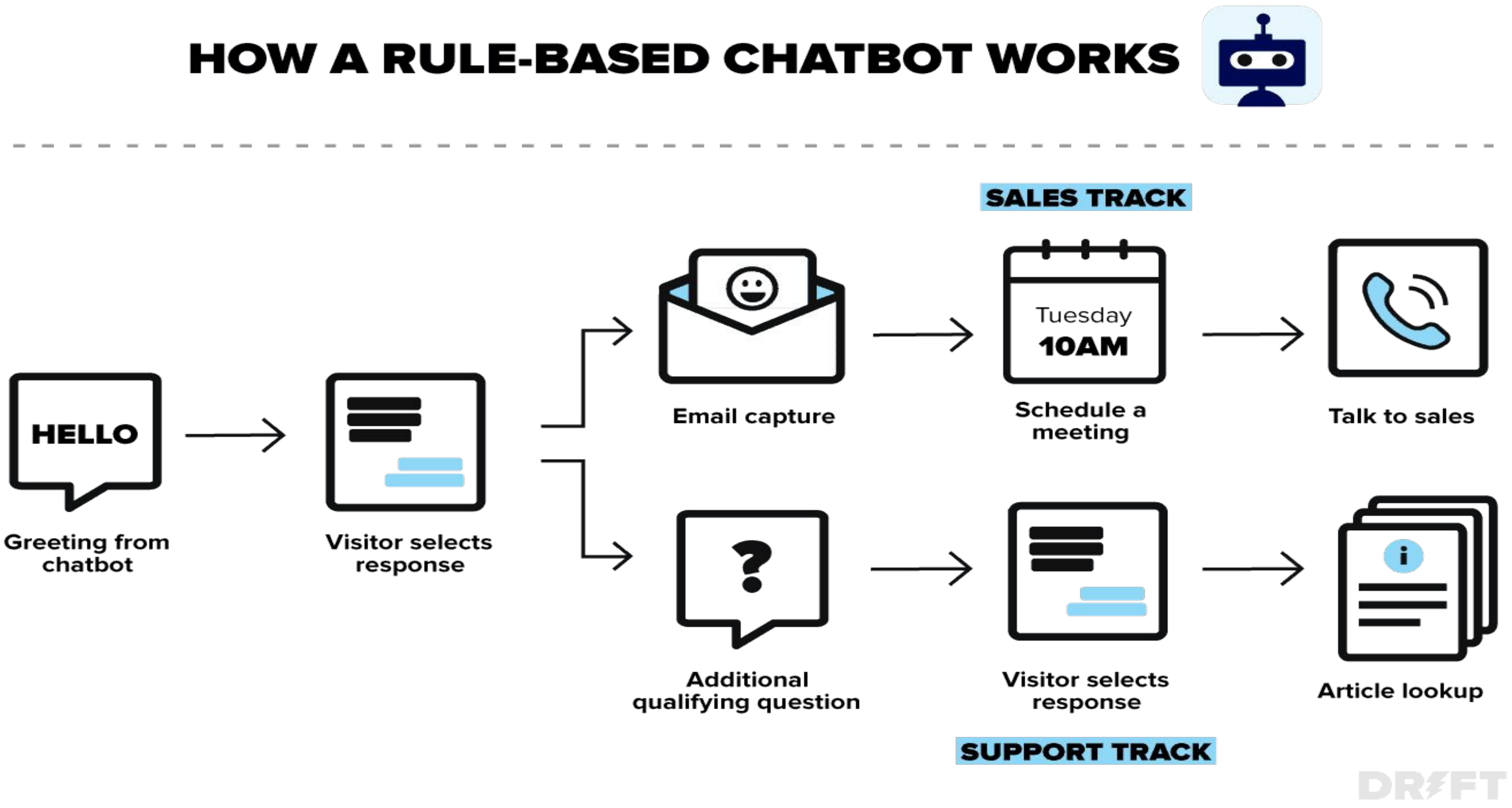


ELIZA (Weizenbaum, 1966) is the most important chatbot dialogue system in the history of the field, and so we take some time here to describe its algorithm in detail. ELIZA was designed to simulate a Rogerian psychologist, based on a branch of clinical psychology whose methods involve drawing the patient out by reflecting patient's statements back at them. Rogerian psychology is the rare type of conversation in which, as Weizenbaum points out, one can "assume the pose of knowing almost nothing of the real world". If a patient says "I went for a long boat ride" and the psychiatrist says "Tell me about boats", you don't assume she didn't know what a boat is, but rather assume she had some conversational goal. Most chatbots trying to pass the Turing test choose a domain with similar properties.

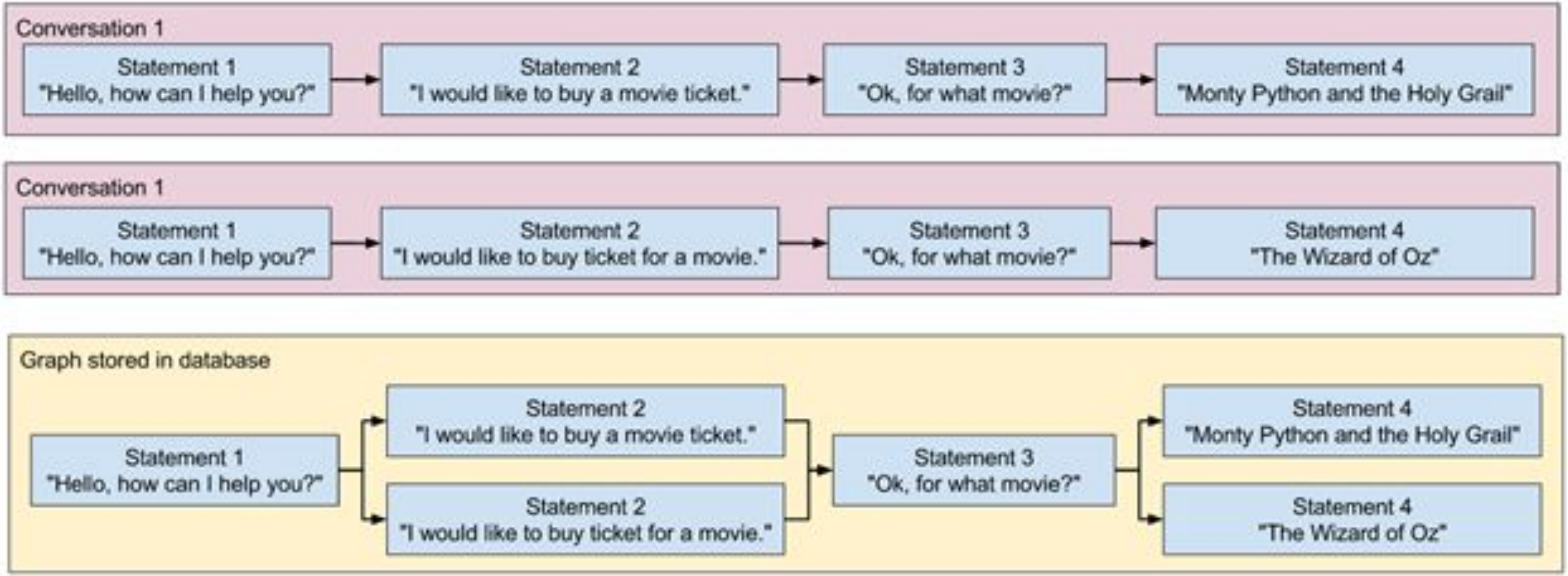
- RuleBased Chatbot


User Input
Search Input for Keywords
Match Input to Intent based on Keyword
Pick Response based on Matched Intent
Response Output

Rule based Chabot



Building an AI-based Chatbot in Python



Conversation 1

Statement 1 "Hello, how can I help you?"

Statement 2 "I would like to buy a movie ticket."

Statement 3 "Ok, for what movie?"

Statement 4 "Monty Python and the Holy Grail"

Conversation 1

Statement 1 "Hello, how can I help you?"

Statement 2 "I would like to buy ticket for a movie."

Statement 3 "Ok, for what movie?"

Statement 4 "The Wizard of Oz"

Graph stored in database

Statement 1 "Hello, how can I help you?"

Statement 2 "I would like to buy a movie ticket."

Statement 2 "I would like to buy ticket for a movie."

Statement 3 "Ok, for what movie?"

Statement 4 "Monty Python and the Holy Grail"

Statement 4 "The Wizard of Oz"

Transformative user experience(UX) of chatbots

- Chatbots are programmed to answer customer questions and resolve customer issues quickly and accurately via AI and machine learning.
- **Good chatbots can improve (User Experience)UX by making customer service faster, more personalized, and available 24/7.**
- Bad chatbots can actually damage UX, so it’s important to invest heavily in the design and testing processes before launching your bot.
- https://www.emarkable.ie/2020/08/chatbots-improve-user-experience-on-your-website/

- Benefits of chatbots from a business perspective are obvious: well-designed chatbots are cheaper than live agents and can resolve exponentially more issues than humans in the same amount of time. Plus, they don’t need to sleep or take breaks, making customer service available 24/7.

- But the **benefits of chatbots** are also becoming clear to users themselves. UX designer, mentor, and writer Mallory Kim outlines the following central principles of UX:

- User-centricity
- Simplicity
- Consistency
- Confirmation and feedback
- Forgiveness
- Accessibility
- Digestibility
- Familiarity
- Navigability
- Data-driven foundation
- Control
- Helpfulness

Designing elements of a chatbot

https://azumo.com/chatbot-design-elements

- Step 1. Frame the Conversational Bot's Purpose
- Step 2. Understand your Bot Platform
- Step 3. Choose the Best Chatbot for Your Purpose
- Step 4. Use the Right Chatbot Design Elements
- Step 5. Track Data and Analyze Behaviour

Best practices for chatbot development

https://outgrow.co/blog/best-practices-for-chatbots

- Set Expectation for your Chatbots
- Be mindful of the Chatbot’s greeting
- Be Upfront About Bot Functionality
- Try to make the messages as human as possible
- Make it Easy For Your Customers To Leave
- Reengage Users Through The Chatbot

How are bots different from humans

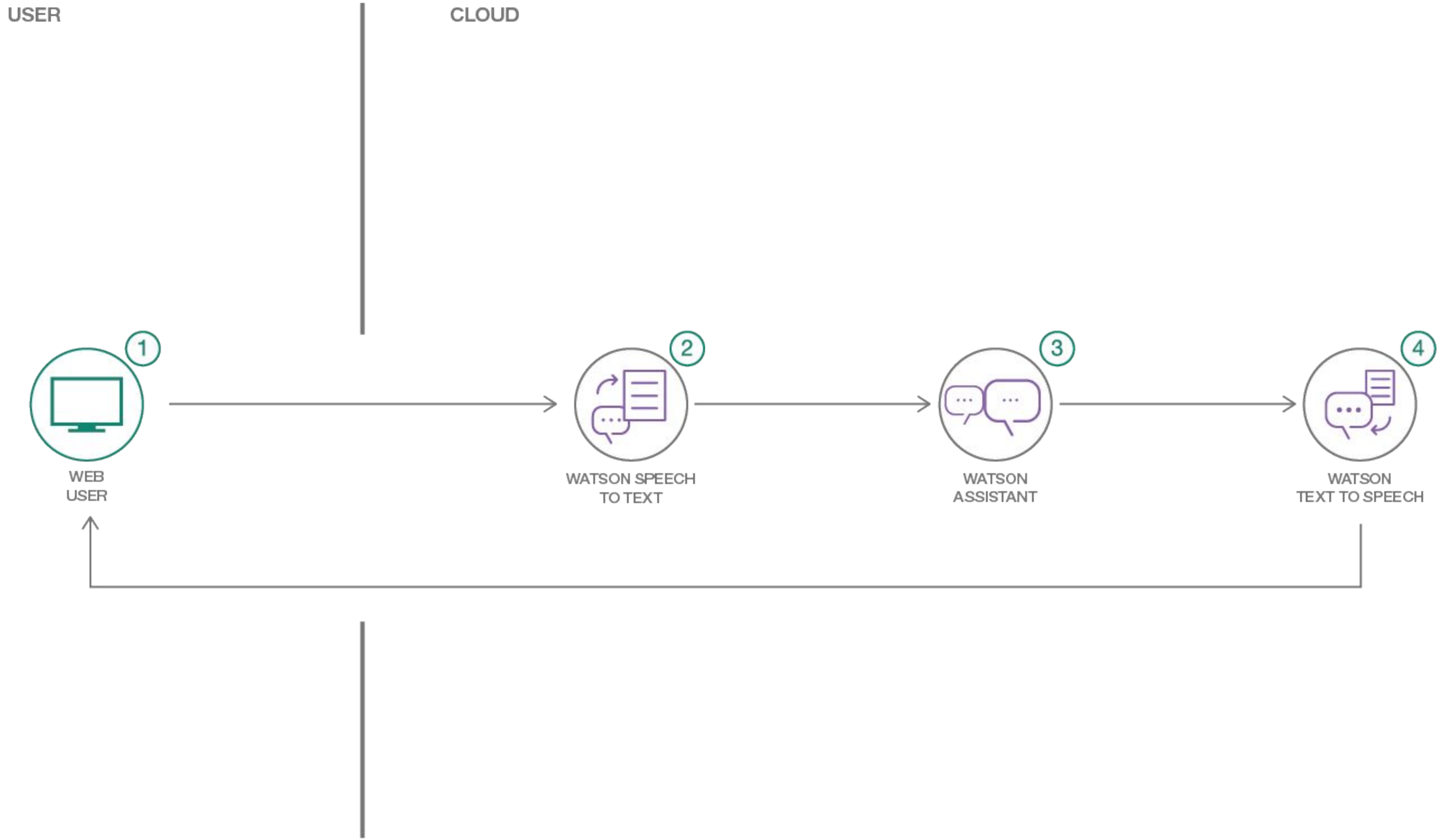
- Bots have **no online status** and **no last seen timestamps**, the interface shows the label ‘bot’ instead.
- Bots have **limited cloud storage** — older messages may be removed by the server shortly after they have been processed.
- Bots can't initiate conversations with users.
- A user must either add them to a group or send them a message first. People can use telegram.me/<bot_username> links or username search to find your bot.
- **Bot usernames always end in ‘bot’** (e.g. @TriviaBot, @GitHub_bot).
- When added to a group, bots do not receive all messages by default (see Privacy mode).
- **Bots never eat, sleep or complain** (unless expressly programmed otherwise).

Working Principle of AudioBots

- Use IBM Watson Speech to Text, Watson Text to Speech, and Watson Assistant services to build a web-based chatbot that has audio as input and output
- Chatbots are used to improve customer service and cost savings.
- Watson Speech to Text and Watson Text to Speech are speech recognition APIs to convert speech to text and text to speech.
- By combining these, you can create a voice-enabled chatbot.

The basic flow is:

- The user speaks through the web UI.
- The user’s voice is recorded using an appropriate web-based recorder.
- The recording is passed to the Watson Speech to Text service on the cloud where it is transcribed into text.
- The text is then passed to Watson Assistant, which identifies the intents and entities to give an appropriate response.
- This text response is passed to Watson Text to Speech, which provides the audio response in a natural voice.


USER
CLOUD
1
WEB USER
2
WATSON SPEECH TO TEXT
3
WATSON ASSISTANT
4
WATSON TEXT TO SPEECH

- The code on the client side is primarily in Javascript, jQuery, and Flask. All of the REST API calls to the different Watson services are made from Flask.

MusicBot

- MusicBot is the original Discord music bot written for Python 3.5+, using the discord.py library.
- It plays requested songs from YouTube and other services into a Discord server (or multiple servers). Besides, if the queue becomes empty MusicBot will play through a list of existing songs with configuration. The bot features a permission system allowing owners to restrict commands to certain people. As well as playing songs, MusicBot is capable of streaming live media into a voice channel (experimental)..
- https://awesomeopensource.com/project/Just-Some-Bots/MusicBot
- https://github.com/Discord4J/Discord4J/wiki/Music-Bot-Tutorial#introduction

- Discord is a popular real-time messaging platform with robust support for programmable bots.
- Bots are present on Discord and provide a wide range of services, including moderation assistance, games, music, internet searches, payment processing, and more.

https://www.toptal.com/chatbot/how-to-make-a-discord-bot

MusicBot

- **Discord** is a proprietary freeware VoIP application and digital distribution platform—designed initially for the video gaming community—that specializes in text, image, video and audio communication between users in a chat channel. Discord runs on Windows, macOS, Android, iOS, Linux, and in web browsers. As of 21 July 2019, there are over 250 million unique users of the software.

- https://en.wikipedia.org/wiki/Discord_(software)

- https://www.igeeksblog.com/how-to-use-apple-music-bot-in-messenger/