BUILDING AN E-COMMERCE CHATBOT

by

SONDOS MAHMOUD

AFAF ABDLBAKRY

A thesis submitted to the

School of Computing

in conformity with the requirements for

the degree of Master of Science

Queen's University

Kingston, Ontario, Canada

October 2022

Abstract

Chatbots have been used in many fields, ranging from education to healthcare and are also used in e-commerce settings. This study describes the artificial intelligence chatbot and focuses on its implementation within the e-commerce strategy. With the assistance of AI chatbots enhance human-machine communication with the assistance of AI. To structure the e-commerce datasets, this chatbot is built with Python and a SQL (Structured Query Language) server as the database. Recently, many NLU platforms have been provided to serve as an off-the-shelf NLU component for chatbots like Rasa and Google Dialogflow, but Google Dialogflow supports one language at a time. To know how it'll be made, we are going to make a simple English chatbot via Dialogflow. To address this gap, we will develop a multilingual chatbot system that will allow companies and organisations to customise and deploy their own multilingual AI-chatbot service. The chatbot will also be able to understand and reply in English and Arabic. To achieve this, we will use the deep learning technique.

Keywords— E-commerce multilingual-chatbot, Artificial intelligence, Google Dialogflow, Natural Language Understanding.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Chatbots are an example of technology that will shape the future of marketing. The business world has changed in several ways in recent years. You can now interact with businesses one-on-one through chatbots. Needless to say, bots are becoming more popular every day, and they interact with customers as a human would and without cost. The chatbot will be accessible on computers, which customers can log in to anywhere and anytime. Many companies are using them for different functions or to deal with their customers. This work attempts to develop an AI(artificial intelligence) chatbot for an e-commerce website. A chatbot that functions through deep learning has an artificial neural network inspired by the neural nodes of the human brain. The AI chatbot is programmed to self-learn as it is introduced to new dialogues and words. By linking different questions, AI chatbots can give personalised responses to the customer's questions. Compare this to rule-based chatbots, where the response to any question is programmed and comes across as mechanical and non-engaging. The conversation between customers and rule-based chatbots doesn't easily jump between different questions. The chatbot's purpose is to have a smart, accurate, and real-time conversation with customers. Customers can converse with the bot in this manner to inquire about specific items they wish to purchase. Chatbots can bring many benefits to a business, and they are the best digital marketing tool. Businesses need technology to operate today. Chatbots are excellent examples of this technology.[20]

1.1 Techniques to develop a chatbot

A chatbot will never be able to replace a human-to-human interaction, but it can help customers in finding necessary information about your business and e-commerce. There are many services available on the market which contain the following:

1. There are widely-used NLUs, like Rasa, Dialogflow, Microsoft Azure, IBM WAtson, BotMan, Botpress, Chatterbot, LUIS, SAP Conversational AI, but Gregori evaluated NLUs using frequently asked questions by university students and found that Dialogflow performed best.

2. The development of the chatbot with deep Natural Language Processing (NLP) and datasets.

3. The development of database items.

4. Introduction of administrator privileges, which can be updated by an assigned staff to update the current list of items and the number of stocks for each item.

5. System testing to ascertain the proper functionality of the developed chatbot, alongside, user testing by a focus group to provide constructive feedback on the developed chatbot.

1.2 Motivation

Chatbots are no longer optional but business essentials. The impact they have had on different business aspects is excellent. AI Chatbots are smart chatbots that answer customer questions directly. Implementing these chatbots on your websites means that you shouldn't depend on the customer support team to answer every question, they personalize communication for every customer. Besides, e-commerce visitors get lost in the millions of products, so chatbots can help customers make and track orders. Businesses put a lot of effort into marketing themselves. Chatbots have emerged as the best digital marketing tools. They have helped businesses enhance their marketing efficiency. In addition to, they have improved customer satisfaction. This will go a long way in saving them money and time. When chatbots that support only one language come to deal with larger companies and public services, the users with unfamiliar language for this chatbot are going to face a problem dealing with it, so the companies will be lost a lot of customers. To handle this problem, we will build a multilingual chatbot that supports English and Arabic languages. Building an AI chatbot that will answer smartly is still a challenge until now. In the end, it will help businesses achieve a significant return on investment.[13]

1.3 Problem

E-commerce sites contain different types of items and categories. So the customer has a problem finding a particular item or ordering, returning, canceling and shipping the order. In addition, users need to know seller information and product reviews without wasting time. Most companies use chatbots that deal with only one language, so they will lose some of their customers. The chatbot can make recommendations based on previous transactions, remind customers of any product they want to know about, or ask questions to provide details on the product they are looking for.

chatbot challenges

1. Lack of memory resources.
2. Problems with data translation from English to Arabic
3. Orthographic ambiguity:if two nouns are spelled the same in an orthography, then this is lexical ambiguity.
4. The term ”morphologically rich languages” (MRLs) refers to languages in which substantial grammatical information, i.e., information concerning the arrangement of words into syntactic units or cues to syntactic relations, is expressed at the word level.
5. Dialectal variation in grammar and vocabulary, in addition to sound variations in orthographic consistency.
6. Resource poverty(data and tools)
7. Limited research.

1.4 Objective

The goal is to develop an open-domain intelligent multilingual chatbot with English and Arabic language support that is designed by using the seq2seq model to retrieve all sorts of information, provide and analyse information and answers for narrow scenarios, and develop real-time discussion. Build a language detection model to make the chatbot classify and reply in English and Arabic. Build a website to deploy our multilingual chatbot on it.

1.5 Thesis Organization

Firstly, in the first chapter, we define the problem and goal of building a multilingual chatbot. In the second chapter, we discussed how to make a multilingual e-commerce chatbot by using English and Arabic language. In the third chapter, we will describe the dataset and its preprocessing steps, what the used model is, and how we can deploy our chatbot with the website. In the fourth chapter, we will cover the chatbot result. Finally, in the fifth chapter, we mentioned the limitations and challenges of chatbot.

Chapter 2

Background

The two major reasons for using a chatbot are to increase efficiency and save time. There are a lot of ways to improve communication between your company and its customers. One efficient method (both in terms of cost and results) for any business to improve their customer service is by building chatbots. The general workflow of a chatbot is as follows: Chatbot frameworks generally provide an interface for customising a set of intents, entities, and post-completion actions in certain states in the conversation and a test tool for the newly developed bot. According to research, chatbots will handle 80 percent of routine customer queries. There are different kinds of chatbots. We will discuss both in the next section. [6].

2.1 chatbot's types

We need a chatbot in our life to make our daily life easier, but what types of chatbots actually exist? There are rule-based bots and bots with artificial intelligence (AI bots). In more complicated cases, AI chatbots are used to fully resolve customers' issues. Also, rule-based bots are limited by grammatical errors or wrong keywords that people might use. This is why rule-based chatbots require more data for automated customer service training.

2.1.1 Rule-based Chatbots

Here, a chatbot system works on a series of defined rules. However, when the input pattern doesn't match with any predefined rule, then this chatbot system is inefficient at answering the question. Most developers use Artificial Intelligence Markup Language to write rules for chatbot systems. AIML is an XML-based language. These rules are the foundation for the kinds of problems the chatbot is familiar with and can deliver solutions for. And it is difficult to write down rules for every possible situation. Rule-based chatbots, such as Google-Dialogflow, can use simple or complex rules. They can't answer any questions outside of the known rules. These chatbots don't learn through interactions. Also, they just perform and deal with the scenarios or intent you train them for.[19].

2.1.2 Self-learning Chatbots

They are also called AI chatbots. Here the chatbots use machine learning algorithms, which enables them to learn more and more. These bots include two types: By Using Retrieval-Based Models: These bots are trained for many different inquiries and their likely answers. For each question, the bot can give the most important answers from the set of every possible answer. Likewise, there is no issue with the language and sentence structure, as the suitable responses are predetermined, and it can't turn out badly in a sentence structure way. By using generative models, it doesn't reply with the same answer from a set of answers. They take word by word from the inquiry and give suitable responses. These models must be prepared more precisely because they will handle errors in spelling and grammar easily.

2.1.3 NLU chatbots

By dealing with language, users ask the chatbot to perform specific tasks or inquire about a couple of pieces of knowledge. Internally, a chatbot then uses the NLU to research the proposed query and act on the user's request. So, NLU is employed to extract structured data from unstructured language input. Specifically, it extracts intents and entities from users' input queries: intents represent the user's purpose of the question, while entities represent important informational data within the query. Most NLUs include a collection of built-in entities which are pre-trained on general domain queries. To use an NLU in a specialized domain, developers must define a collection of custom intents and entities. Any custom intent requires NLU training and a series of queries representing alternative ways a user might express this intention. The response principle is to match the user question to a collection of reference sentences and calculate a score of the similarity of the sentence to the system-trained reference sentences. NLUs must be trained to acknowledge custom entities. The misclassification of intents and entities negatively impacts the chatbot responses.[10]

2.1.4 Dialogflow platform

Dialogflow is a common conversational tool for natural language understanding developed by Google that simplifies the design of conversational user interfaces into mobile applications, web applications, devices, bots, interactive voice response systems, etc. It provides an enjoyable way for the users to interact with the products they produce.

Dialogflow can integrate and handle text or audio data.[12]

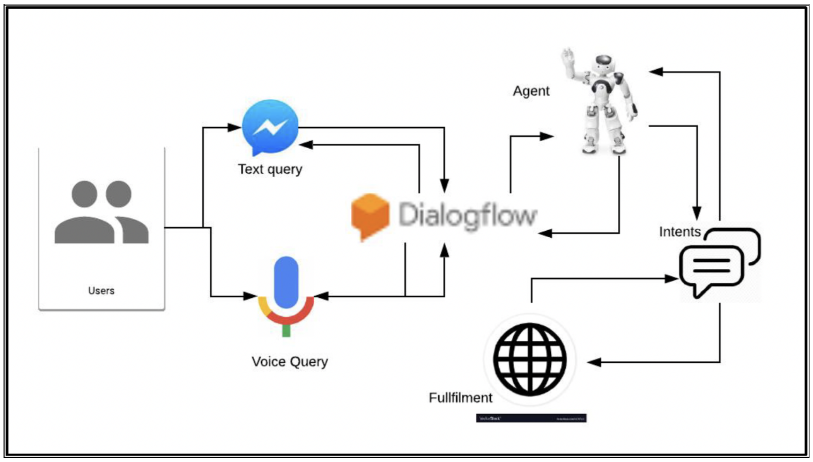


Figure 2.1: Dialogflow's Architecture.[16]

Components of DialogFlow

Following are the components of DialogFlow used for designing Chatbot:

1. User: User can be Humans or Machines.

2. Text Query/Voice Query: The user interacts with an app like Facebook Messenger/Google home to start the interaction with the bot.

3. Dialogflow: This is the Chatbot platform used to create Placement-Activity agent.

4. Virtual Agent: Is an AI virtual character that is a Chatbot. It's the first step that must be created before creating a chatbot, because it will train and understand human language.

5. Intent: Is a specific action that can be recalled by the user in the Dialogflow by using one of the known terms.

6. Fulfillment: Is a piece of code that is deployed as a webhook and the DialogFlow interface agent calls the business on a per intent basis. During a conversation, this allows the user to use the information extracted from DialogFlow's NLP to generate the response or to move action at the back end.[17]

2.1.5 Multilingual AI-chatbot

AI provides many opportunities, as it includes such capabilities that allow the software to perform such tasks that humans perform. Natural language processing is the foundation of AI based chatbots. By using complex algorithms of NLP, chatbots can process the input text: understand, conclude, and determine what was said or written and then state a list of all suitable actions. For the multilingual chatbot, we will translate the English dataset into Arabic dataset by using translate library[6].

2.1.6 Rasa platform

RASA is the only open source NLU and it's a framework to create user-interactive AI chatbot using python and NLU. The processes like tokenizing, stemming and Tf-idf all of this will be completed by the rasa framework. It doesn't interact deeply with deep learning and machine learning. Rasa itself has built models which are in abstraction. The commands such as rasa train will train the model and it will be ready for use. All developers have access to configure, deploy, and run the NLU on local servers. Thus, increasing the processing speed by saving the network time compared to cloud-based platforms[15]. Rasa has two main components:

1. Rasa NLU (Natural Language Understanding): which takes the user input and tries to conclude the intent (purpose) and extract the occasion entities.

2. Rasa Core: A conversation management solution tries to build a probability model which decides the set of actions to perform based on the previous set of user inputs.

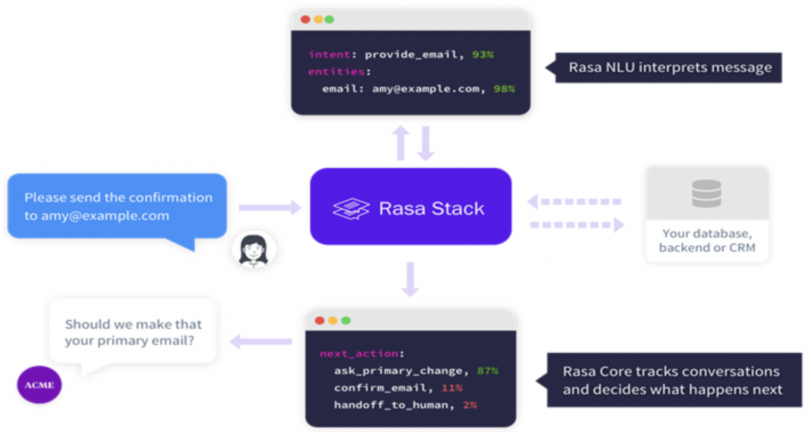


Figure 2.2: Rasa Architecture.[7]

Language Understanding Intelligent Service (LUIS) platform

An NLU cloud platform from Microsoft. LUIS contains various prebuild domains such as music and weather, and supports five programming languages: C charp, Go, Java, Node.js, and Python.

Chatterbot

Chatterbot is a Python library that implements a conversational dialog engine for chatbots using several human languages and provides a training and logic adapter that matches the user input against the training data[5]. It likes the following :

1. Botpress is a framework for creating bots using independent modules and third parties.

2. BotMan is a PHP library for building a chatbot that can be used by itself or in a Laravel-based bundle and supports several popular messaging channels as well as its own drivers.

Microsoft Azure

It is Microsoft's artificial intelligence (AI) chatbot offered as a service on the Azure cloud service marketplace. Azure Bot Service can add intelligent agents that are qualified for conversation without having to commit the resources to develop one's own AI.

IBM WAtson

An AI service in IBM Cloud that lets you build, train and deploy Chabot over the cloud. Anyone can build a Chatbot. You need not have any sort of technical experience to build a chatbot using IBM Watson Assistant. The conjectural interface helps you to easily create dynamic conversational flows.

2.2 Related work

Most algorithms that generate or choose pre-defined chatbot answers still rely on rule-based approaches, in which answers are selected from a predefined set. Despite the advantages of neural-based methods, such as RNN, CNN, and GAN, these have been proposed in only a few papers.[10]

AS, A. R. D. B. Landima in [10] presents the chatbot's dialog policy decides which action the chatbot should take in the next iteration, in order to help users complete a task. Although many different approaches have been proposed, we have identified

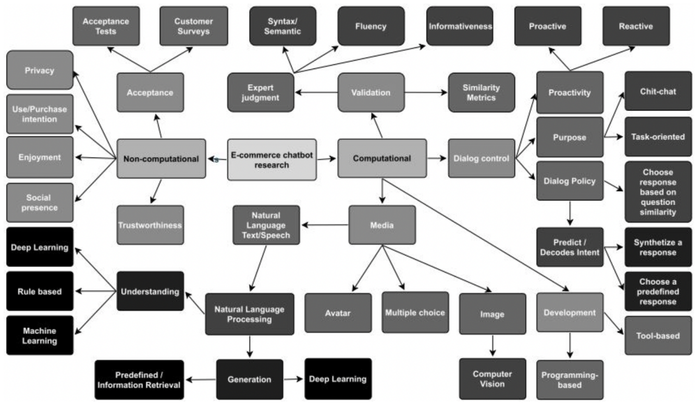


Figure 2.3: E-commerce chatbot research.[10]

three main subcategories in the literature: firstly, to predict the user's intent and then choose a specific answer from a finite set of predefined answers. The second to analyze the similarity between the user's questions and the questions in the data set and the corresponding choice of answer, and sequential DL approaches first decode the user's intent and synthesize the response.

Evaluation and Validation

The literature on measuring chatbots' sentence generation effectiveness generally falls into two categories: Similarity metrics, which compare the chatbot's answer to that of a human, and/or expert evaluation, which relies on human judgment. To quantify the quality of a sentence generated by chatbots. Another way to validate a chatbot is by having human experts help interpret the bot's sentences. Starting in 2017, despite the popularity of modern machine learning methods, classical methods are still in use in some of the best research. Rule-based algorithms use hand-crafted hard rules using, for example, Artificial Intelligence Markup Language (AIML). Classical ML algorithms in general combine the extraction of hand-crafted features from unstructured textual data, such as the number of n-grams or the number of terms. Statistical measure "Frequency - inverse document frequency" (TF-IDF). DL algorithms deliver cutting edge performance through the use of deep nerves.

Future work

The Future research on non-computational aspects of chatbots for e-commerce that can be applied in the fashion domain appears more promising in the following 4 areas: firstly is, chatbot acceptance and design for different demographics: Rese et al. (2020) found that females had a negative attitude towards chatbots due to the technology immaturity and privacy issues. The second is consumer autonomy and identity in chatbot consumer experience:consumer autonomy (Ameen, Hosany, and Tarhini, 2021) is related to the perceived sense of control that consumers have over the interaction with chatbots, and it can be attached to motivational factors (e.g. Self-Determination Theory). The third is chatbot design, consumer trust and privacy. The fourth is perceived enjoyment, usability, and usefulness: perceived enjoyment and the utilitarian nature of the interaction (e.g. whether useful or not) also influence consumer acceptance.

The result

Most research on chatbot computational aspects had English as their primary language (76.3 percent), followed by papers on Indonesian chatbots (6.8 percent) and other languages like Chinese and Bangle. However, the resulting papers were mostly not fashion-specific (87.7 percent). Contrastingly, a few papers like Liao, Zhou, Ma,

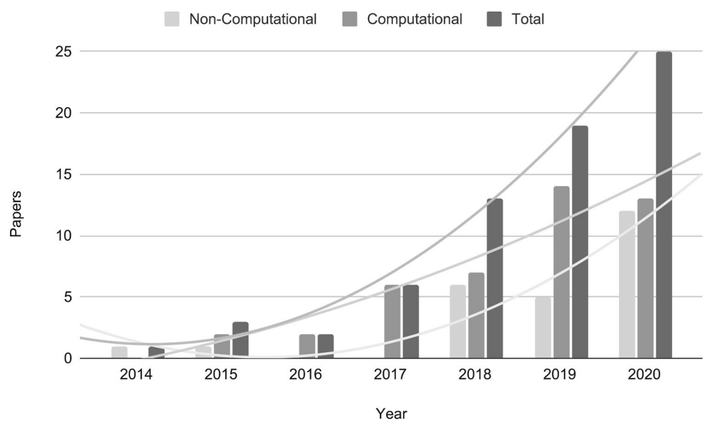


Figure 2.4: Evolution of chatbots for e-commerce research from 2014 to 2020.[10]

Hong, and Chua (2018) and Vaccaro, Agarwalla, Most recent papers already consider natural language text as the input method (79.4 percent), except for Aarthi (2020). A few studies (15.1 percent) are voice-based. He proposed Dialog control as a tool to build a chatbot. The investigated dialog policies were based on variations of deep learning architectures, such as recurrent neural networks (RNN) and Convolutional neural networks (CNN).

Furthermore, researchers as Victoria Oguntosin in[20] proposed the chatbots, that have been utilized in numerous areas extending from education to healthcare and are too utilized in e-commerce settings. The e-commerce chatbot application for Covenant University Shopping Mall (CUSM) seeks to provide an easy, smart, and comfortable shopping experience for the Covenant University Community. The main goal of creating a web-based chatbot called Hebron is Real-time discussion with the understudies. In addition to providing the information about what products are currently on offer, what the pricing of the goods sold.

Chatbot Interface

Utilizing React.js, a front-end framework for developing single-page web applications React.js aids in the creation of responsive web pages. Users (students/staff) can completely interact with Hebron (the chatbot) through this presentation layer and receive accurate and current responses. The official CUSM customer service department is Hebron. Here, the user can ask the bot questions about what products are currently on offer, what the pricing of the goods sold. The user can then utilize the CUSM payment platform in the mall are, and when CUSM closes and opens to pay for the goods they want to buy. Then Message Backend contains: Python and SQL will be employed in its development. Python is a high-level, simple-to-understand language. Additionally, it supports ML and AI. Relational database management systems are programmed and managed using the domain-specific language SQL. It will be useful in this situation to manage the e-commerce datasets kept in a DBMS (MySQL). A web database manager is MySQL.

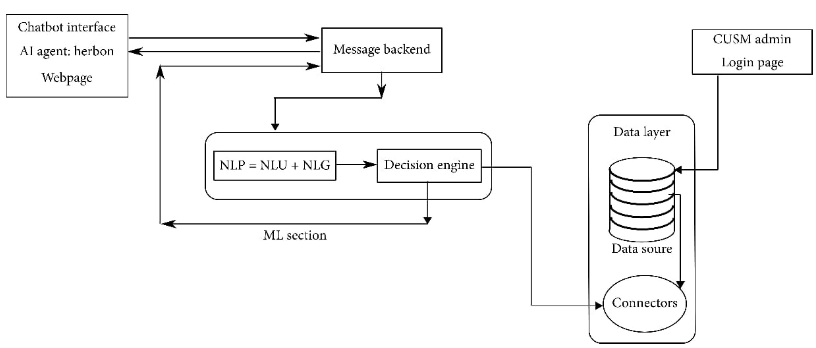


Figure 2.5: Chatbot for University Shopping Mall System Architecture Design.[20]

The algorithm

It's contains utilizing Spacy, Recast.ai, and an open-source Python library and API, the ML part will be created. Spacy is the NLP component of Hebron's first ML component that will aid in understanding and translating the massive amounts of texts (data type) that Hebron will come across during conversations with its target users, particularly in terms of the grammatical organization of each sentence that Hebron will come across. This will support Hebron's deep learning approach to NLP. In contrast, Hebron is trained using the Recast.ai API, it includes subdivisions like the user's intents and preprogrammed expressions under each intent and skill in the chatbot's conversational flow. Recast.ai will assist with establishing a webhook for the chatbot interface and linking Hebron to the external DBMS. Prefixes, suffixes, and infixes are subsequently taken into account and divided into tokens according to its exception rules. Lemmatization feature, third. This function makes it possible to determine a word's root form. For instance, the words "tables" and "standing" have the lemma "table," "stand," and "understand," respectively. The Linguistic Annotation Function. This function reveals the grammatical structure of the sentence.

The result

Shows that about 50 percent of users thought that Hebron's functionality was good and satisfactory, suggests that Hebron fulfilled about 60 percent of its functionality, and 60 percent of users rated Hebron's interface as good.

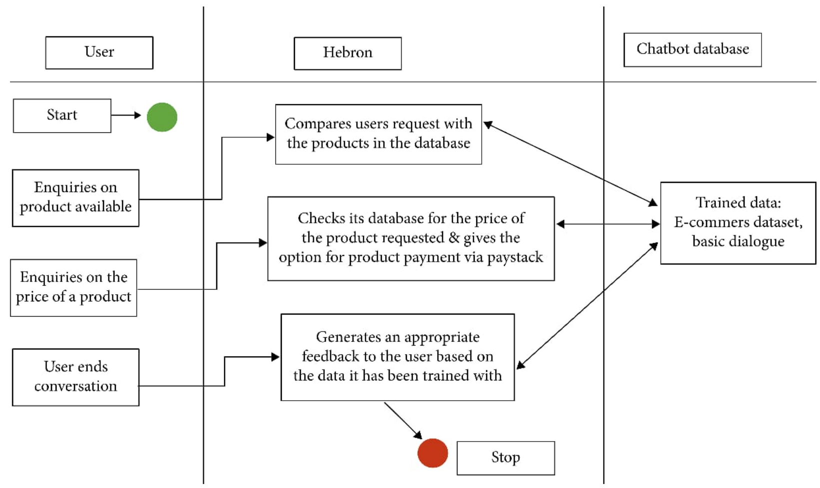


Figure 2.6: The activity diagram for the inquiry of products, availability of products alongside the requested.[20]

2.2.1 Different platforms

Comparison of evaluating the performance of different NLUs(IBM Watson, Dialogflow, Rasa, and LUIS)using SE tasks, so the researchers in [16] proposed that solution approach by making intents classification, confidence score, and entity extraction with precision, recall, and F1-measure measurements.

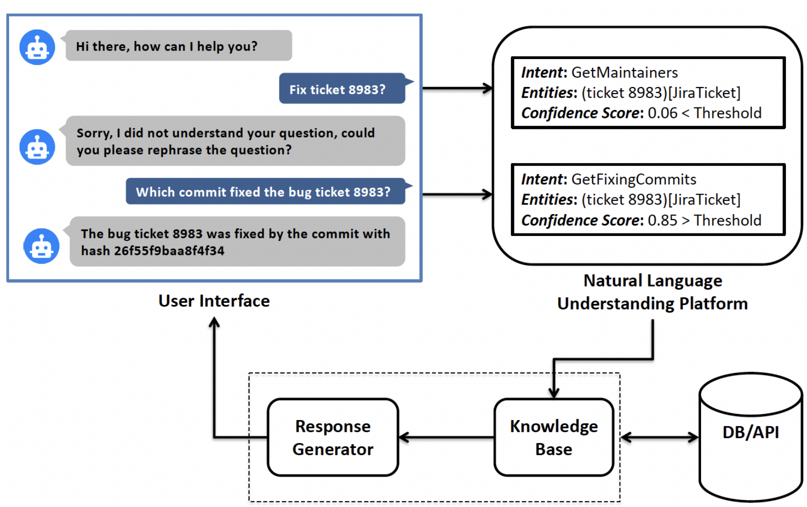


Figure 2.7: Overview of chatbot interaction.[2]

Platforms

Then they choose to evaluate the following tools : Watson Conversation (IBM Watson): is the first choice as a bot-building platform for 61 percent of businesses. One of the Watson's great parts is a Conversation Service. Dialogflow: An NLU developed by Google. It supports more than 20 spoken languages, that makes it easy to design and integrate a conversational user interface into your mobile app, web application, device, bot. Language Understanding Intelligent Service : It's NLU cloud platform from Microsoft. LUIS has several prebuilt domains such as music and weather, and supports five programming languages: C-charp, Go, Java, Node.js, and Python. Rasa is the only open-source NLU, it allows developers to configure, deploy, and run the NLU on local servers. It increases the processing speed by saving the network time compared to cloud-based platforms.

The dataset

There are two data corpora (Data corpus refers to all data collected for a particular research project, while data set refers to all the data from the corpus that is being used for a particular analysis), which includes 10 intents with their queries and entities with their lists of synonyms. The first one is repository corpus: it includes questions posed to a chatbot. The second one is stack Overflow: it covers question and answering, uses to train chatbots.

The result

In intents classification, IBM Watson performs the best with an F1-measure=85.5 percent, but in confidence scores, Rasa is the best with median confidence score ¿

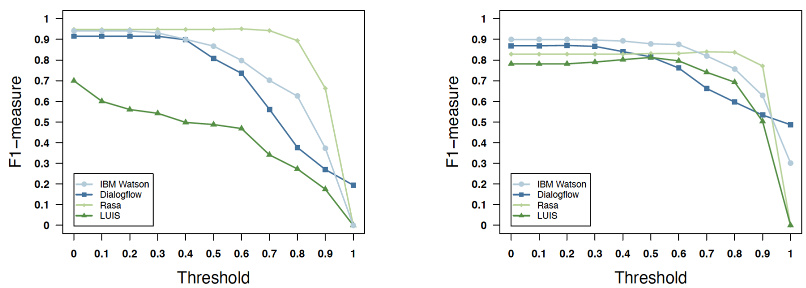
Task	Intent	Intent's characteristics # Training Samples	% Queries w. Exclusive Words	Distinct Entity Types	F1-measure IBM Watson	Dialogflow	Rasa	LUIS	Avg.
Repository	**BuggyFile**	37	92	-	96.0	96.0	**100.0**	83.3	93.8
	FixCommit	31	10	CommitHash	**100.0**	91.7	**100.0**	68.8	90.1
	BuggyCommit	52	7	JiraTicket	**94.7**	**94.7**	**94.7**	64.3	87.1
	OverloadedDev	15	88	-	**94.1**	80.0	88.9	58.1	80.3
	BuggyCommitByDate	66	39	-	72.7	69.6	80.0	**91.7**	78.5
	CountCommitsByDate	11	97	-	88.4	**91.3**	80.0	53.3	78.3
	BuggyFixCommit	32	74	-	**100.0**	85.7	63.2	48.0	74.2
	ExperiencedDevFixBugs	15	72	-	**92.3**	88.9	**92.3**	13.3	71.7
	CommitsByDate	8	70	-	70.0	50.0	**76.2**	0.0	49.1
	FileCommits	10	77	-	55.6	**80.0**	28.6	11.1	43.8
Stack Overflow	**LookingForCodeSample**	132	100	Platform	**90.9**	85.9	84.5	84.0	86.3
	LookingForBestPractice	12	92	-	81.3	81.3	78.0	**83.3**	81.0
	UsingMethodImproperly	51	100	-	**79.1**	66.5	64.4	57.5	66.9
	FacingError	10	100	-	**80.0**	60.0	33.3	10.0	45.8
	PassingData	10	70	-	36.7	**50.0**	36.7	0.0	30.9

Figure 2.8: Intents' characteristics and classification performance as F1-measure of the four NLUs.[2]

0.91. In the results also show that all NLUs, excepting for google-dialogflow, usually provide trustable confidence scores. Intents Classification : NLUs rank similarly in both tasks in intents classification, with IBM Watson outperforming all other NLUs, followed by Dialogflow, Rasa, and LUIS. Aside from the training sample size, intents that contain limited words and special entity types are easier to identify by all NLUs. NLUs Confidence scores :IBM Watson, Rasa, and LUIS provide higher median confidence scores, ranging between 0.68 : 0.96, for correctly classified intents. If the confident score is less than the threshold the chatbot return (Sorry, I did not understand your question) then the user rephrases the question with high confident score. so in high confidence - typically a good answer that completely answers the user's query. Entity Extraction : with LUIS and Rasa outperforming when using Repository task, while IBM Watson and Dialogflow perform better when the entities need to be predicted (Stack Overflow task).

2.2.2 Rasa platform

For making multilingual-chatbot there is Rasa chatbot, it represents in [11]. Firstly we should ask ourselves, what is the problem? the problem is, how can you build a



(a) Ranking in Repository Task (b) Stack Overflow Task

Figure 2.9: Evaluating differents NLUs.[2]

chatbot by rasa?

The solution

We will add a bot to the platform which can understand the human language and generate an appropriate response (answer). This bot will be helpful for filtering the products from whatever the e-commerce sites it has been integrated it with and also replying to some of the issues before they get to the customer call center. Rasa increases the processing speed by saving the network time compared to cloud-based platforms.

Approaches

Rasa has two main components: rasa NlU-core and Rasa NLU. For rasa NLU is an open-source NLP tool for intent classification, it extracts the entity from the bot in structured data form and helps the chatbot understand what the user is needing. Rasa Core: a chatbot framework with machine learning-based conversation management, which takes the structured input from the NLU and predicts the next best action using LSTM neural network, and it also uses reinforcement learning to improve the prediction of the next best action.

The dataset

The initial project files must found in rasa: data/nlu.md, data/stories.md, domain.yml, endpoints.yml, actions.py, data/nlu.md. The important component contains, intents What the user is implying is called an intent, entities are the most important part of data that can be extracted from a user message and data/stories.md: It gives the bot an idea of how the conversation should be.

As Rakhra mentioned in [15], the problem is e-Commerce sites contain a different types of items and categories.so the user has the problem to find the particular item with save time. The Chatbot can make recommendations on the basis of previous transactions, remind users of any product they want to know about, or ask questions to provide details on the product they are looking for according to their interaction.

The solution

There are three reasons to understand for the creation of the architectural flow of an AI chatbot: background of understanding: by using NLU and AI to make Chatbot understand what the user's question, response selection:AI prepares the answer while machine learning allows the bot to learn to deal with similar or unfamiliar scenarios, (Artificial intelligence lets you pick a suitable response) and output the result: Natural Language generates the chosen question answer in the user's own language. A chat has four main components: a front end, an information base, a back-end, and a corpus of training data. The front end is what connect the bot and the customer. It's define the intent (user purpose), NLU Engine :It takes text as input and returns intent, entities and confidence about its prediction, which selects a suitable response and takes part in the chatbots' status to execute the requested action. For the dataset, it is created manually and contains many processes for Chatbot, including data parsing (Data parsing refers to the process of transforming data by making tokenization, stemming, ... (removing irrelevant information)), data crawling and pattern matching.

The algorithm

There are waterfall models involves stages such as specification acquisition, analysis, architecture, implementation, and testing. Each operation is carried out one by one. The main disadvantage of the waterfall model, since the project is split into stages, is its inflexibility. If there is any step falls, then the other steps won't continue. The gradual model is the most fitting approach, it's incremental model, so we can add the new requirements during the entire development period.

The dataset

The QA dataset is saved as a CSV file or JSON. The Dataset contains many processes for Chatbot, including data parsing (Data parsing refers to the process of transforming data by making tokenization, stemming, ... (removing irrelevant information), data crawling and pattern matching.

Today's chatbots come in a variety of forms, including general-purpose conversational agents, task-oriented chatbots, open-domain agent systems, and domain-specific agent systems. As Angelo indicates on its paper [5] he presents a distributed chatbot system for supply chain. The system includes several services: chatbot, natural language processing services as well as supply chain service.

Approaches

The supply chain microservice aims to provide the user with the required information about orders and supplies. NLU service A chatbot then uses the NLU internally to analyze the request made and respond to user requests. The main purpose of an NLU is to extract structured data of unstructured linguistic input. In particular, it extracts intents and entities from user searches: represent intents the purpose of the user of the query, while the entities. In addition, spring Framework, Hibernate ORM, Spring Data JPA, GraphQL, Redis, Kafka, MongoDB are used to implement the suggested system architecture as distributed and scalable chatbot system.

The first docker container is for the MySQL database image, and it contains the specific supply chain information that the chatbot will be able to process. The Kafka docker is used to allow the chat service to scale vertically. The MongoDB docker container is used in the distributed chatbot system to store user history. The Redisserver container is used to store the context of the user.

The result

Show correct recognition in 90 percent of the template test user sentences and 65 percent recognition rate for the user queries with synonyms of the sentences. Also show that the system was able to process up to 10,000 user sessions without scaling any of the servers for 60 minutes. The quality of the NLU service recognition is further increased to 83 percent for the user queries varying from the reference templates by additional training of the NLU system with the test data that are not properly resolved. The main latency of the system came from the inability to implement many Web Socket connections at a time. The other system latency is observed due to network latency and cascading system crash when one of the services fails to process the messages, leading to fail of the other services as well. A chatbot then uses the NLU internally.

2.2.3 Observation

Most of the researches that use rule-based-method aren't efficient for developing chatbot because they are trained by using predefined answers, so chatbots can't interact well. So we purposed deep learning techniques but, the training of DL algorithms heavily relies on the availability of sizable datasets. Our analysis of the literature indicates a dearth of publicly available datasets for chatbot training.

let's show some result of papers served as a basis to estimate the current state of chatbot evaluation.

Paper	Data	Platform	Result(F1-measure)
A.K.D.E. Shihab[2]	Repository	Dialogflow	83.1
		IBM Watson	85.5
		Rasa	80.7
		LUIS	48.8
	Stack Overflow	Dialogflow	84.4
		IBM Watson	78.1
		Rasa	74.8
		LUIS	70.7

Table 2.1: Evaluation scores.[2]

Papers	Important notes	Results
A. R. D. B. Landim[10]	Creating a web-based chatbot called Hebron for the Contract College Community Shopping center by Utilizing Spay, Recast.ai, lemmatization feature, the Linguistic Annotation function and Prefixes, suffixes and infixes.	No result is provided.
Victoria Oguntosin[20]	Chatbot Interface: Utilizing React.js. Message Backend: Python and SQL and ML Section:Utilizing Spay, Recast.ai, and an open-source Python library and API, the ML part will be created Recast.ai will assist with establishing a webhook for the chatbot interface and linking Hebron to the. external DBMS.	No result
M. Mamatha[11]	By using pattern matching and data creeping. From this Rasa E-commerce chatbot, we will filter any products from any site like Amazon and Flipkart.	95 accuracy rate
A.K.D.E. Shihab[2]	IBM Watson has the best performance NLU for the SE tasks. Dialogflow has small confidence score. Rasa has the most trustworthy NLU in confidence score. LUIS and IBM Watson obtain the best results in extracting entities.	F1-score greater than 84.0
Rakhra[15]	The efficient Chatbot applications need to Grammar-based data parsing. For successful pattern matching process, the implemented pattern must extract information that is useful for evaluating the related text then crawling data come to scan a database for data that fits the results of pattern matching.	No result is provided.
Angelo[5]	They use Kafka broker and a NoSQL database to scalability the chat service. The distributed chatbot system keeps the context of the conversation so, the missing information in question will fill automatically them from the previous question.	No result is provided.

Table 2.2: The summary of all papers.

Chapter 3

Methodology

As we mentioned in chapter 2 for[10], most research on the chatbot computational side had English as their main language 76.3 percent like DailogFlow. So, our research will focus on how to build a multilingual smart chatbot. We also built a simple chatbot by using Dialogflow, as you can find its result in the section of the result.

The chatbot system only uses two processes: sorting and data crawling. Only about 1200 of the 1500 queries can be answered correctly, representing about 80 percent accuracy. Other researchers only recommended pattern matching and data creeping with a 95 percent accuracy rate and response times starting from 7.5 seconds to 48 seconds[11].

Sentences can come in different forms from a list of individual words to sentences, to multiple paragraphs with special characters, so we will use natural language processing with deep learning techniques to build a conversational AI chatbot that is able to support the Arabic language also, as Dialogflow supports many languages except the Arabic language.

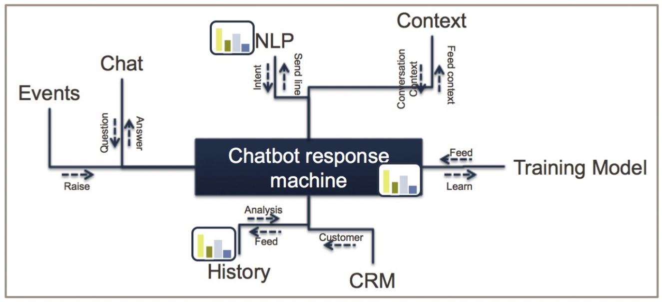


Figure 3.1: How a chatbot work[14].

3.1 English Chatbot

The first step to build an English AI-chatbot is how we can make the dataset ready to use, then use the appropriate model. Natural language text is already used as the input method in 79.4 percent of recent papers. [10].

3.1.1 The first English dataset

The dataset contains 55k questions collected from Amazon QA E-commerce Data as mentioned in [3]. The format of the dataset is JSONl, where each line in the file is a JSON string that represents a question, answers related to the question and the extracted review. We convert this format into CSV.

questionText	questionType	review_snippets	random_sentence	top_sentences_IR	top_review_wilson	top_review_helpful	answers
What ISO range is covered when in set on auto-...	descriptive	['I set up my 5d mark ii with my canon 24-105 ...	["A common shooting technique is to meter for ...	['1) light and small for a FF camera, almost t...	['This camera has top-tier image quality in a ...	['This camera has top-tier image quality in a ...	[{'answerText': 'I believe it is 12800 or 2560...
Will this fit the Nordic Ware 13x18 Baking Pan...	descriptive	["I bought this cooling rack, along with a Nor...	["This is the second rack I've purchased."]	['The fit is a little tight with the Nordic Wa...	['I just got my cooling rack yesterday and ema...	['I just got my cooling rack yesterday and ema...	[{'answerText': "YES!!! I purchased both and...
Will these fit 35"-36" inseam?	descriptive	["and they are very water tight, and the size ...	['A little hard to bend over or get up from ou...	['However, the rest of the wader, especially t...	['My husband bought a pair of Frogg Togg unins...	['My husband bought a pair of Frogg Togg unins...	[{'answerText': 'I use a 32" and I would say a...
How long does it take for flowers to blossom?	descriptive	["If you have loved the scent of flowers wafti...	["And it's such a great price - I bought anoth...	["If you have loved the scent of flowers wafti...	["I've waited a long time to get this set as i...	["I've waited a long time to get this set as i...	[{'answerText': '1 to 3 minutes.', 'answerType...
Will these fit a 1997 Dodge 2500 4wd pickup?	descriptive	['This is a good item, just too bad they did N...	['it feels more rigid and stable.']	['It said for Dodge Ram 2500/3500 and that is...	["Put a set of these airbags on my '06 Dodge M...	['This is a good item, just too bad they did N...	[{'answerText': 'No, the kit you need is made ...

Figure 3.2: Head of the first dataset.

Columns	Description
QuestionText	Represents the question
QuestionType	Yes or no for a boolean question or descriptive for a non-boolean question.
Review snippets	Extracted reviews related to the question.
asin	product ID
qid	question ID
Category	Represents many Product categories.
Top review wilson	The review with the highest Wilson score.
Top review helpful	The helpful review for the user.
Is answerable	0 for the question is already answered or 1 for the question isn't answer.
Top sentences IR	A list of top sentences based on IR score with the question.
Answers	Each answer contains answerText, answerType and helpful.

Table 3.1: All English dataset's columns

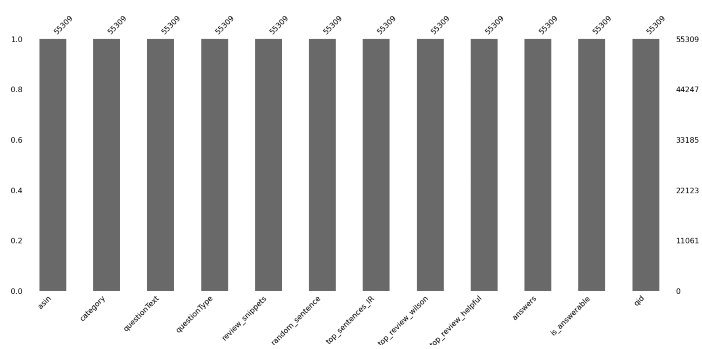


Figure 3.3: Visualisation of missing data.

Firstly, we check for null values and duplicated values. To identify the missing values, we use the missingno Python library, and as we see in figure 3.2, there are no missing values in our dataset. Also, we checked if there are duplicated values, and we found that our dataset doesn't contain any duplicated values. We will work with the questionText column as the input and the answer column as the target.

Figure 3.3 is ideal to display the composition of the data category. How many categories are in our e-commerce dataset?

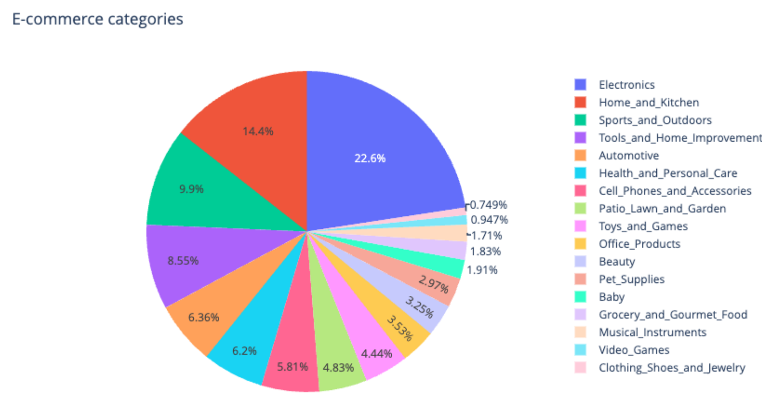


Figure 3.4: Percentage of each E-commerce category in the data.

What percentage of all categories were represented in the dataset?

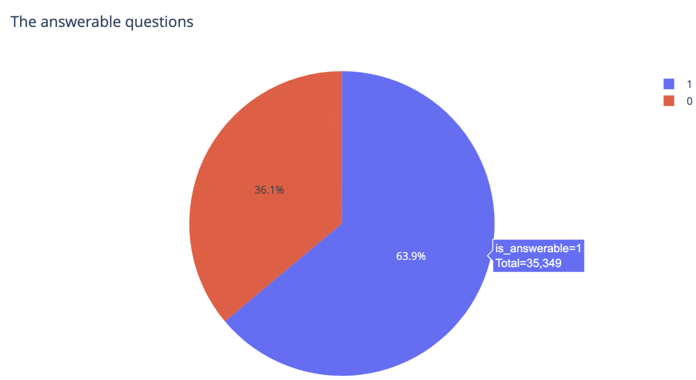


Figure 3.5: Percentage of how many questions are answerable.

The two figures below represent the words per question and the answers per question. Additionally, those show the normalisation of the statistic computed within each bin to estimate density.

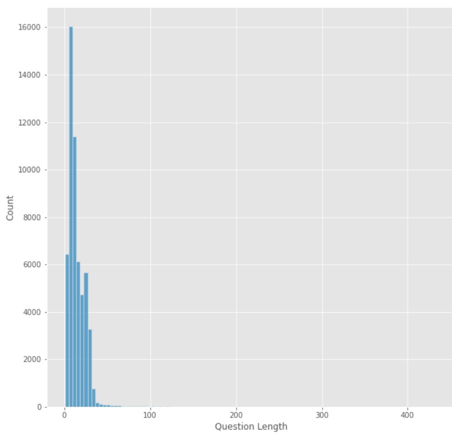


Figure 3.6: Question distribution.

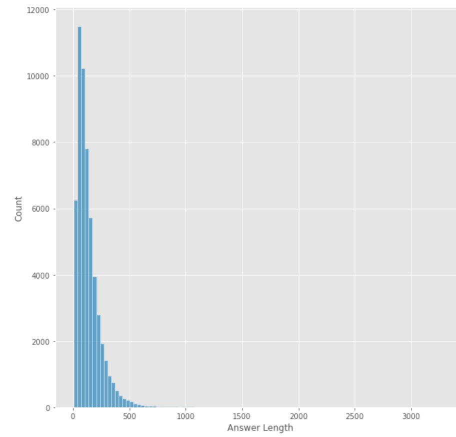


Figure 3.7: Answers distribution.

3.1.2 The second English dataset

The dataset contains 314263K questions collected from Amazon QA E-commerce Data specialised in the electronics category is mentioned in [4]. Firstly, we find 148k rows are null in the answerType column, and we don't need this column, so we drop it. We check the duplicated values and we find 969 duplicated, so we drop them. We will use just the question and answers columns only, so we will drop the others.

	questionType	asin	answerTime	unixTime	question	answerType	answer
0	yes/no	0594033926	Dec 27, 2013	1.388131e+09	Is this cover the one that fits the old nook c...	Y	Yes this fits both the nook color and the same...
1	yes/no	0594033926	Jan 5, 2015	1.420445e+09	Does it fit Nook GlowLight?	N	No. The nook color or color tablet
2	open-ended	0594033926	2 days ago	NaN	Would it fit Nook 1st Edition? 4.9in x 7.7in ?	NaN	I don't think so. The nook color is 5 x 8 so n...
3	yes/no	0594033926	17 days ago	NaN	Will this fit a Nook Color that's 5 x 8?	Y	yes
4	yes/no	0594033926	Feb 10, 2015	1.423555e+09	will this fit the Samsung Galaxy Tab 4 Nook 10.1	N	No, the tab is smaller than the 'color'
...	...	...	...	...	...	...	...
314258	yes/no	BT008UKTMW	Feb 20, 2015	1.424419e+09	Is the space from bottom of desktop to tray ad...	N	No
314259	yes/no	BT008UKTMW	Oct 13, 2014	1.413184e+09	can the mouse extension be mounted on the LEFT...	Y	yes, you can put it on which ever side you want
314260	yes/no	BT008UKTMW	Feb 26, 2014	1.393402e+09	does it come with all the hardware	Y	It's been a while since I bought this, but I'm...
314261	open-ended	BT008UKTMW	Nov 8, 2013	1.383898e+09	how wide is it? I need a 19 inch length tray f...	NaN	We just measured the tray and it is 21 inches ...
314262	yes/no	BT008UKTMW	Nov 27, 2013	1.385539e+09	Can this be adapted to be clamped underneath a...	N	I do not think so.

314263 rows × 7 columns

Figure 3.8: Head of the second dataset.

3.1.3 Data preprocessing

Data preprocessing transforms the data into a format that is more easily and effectively processed in data mining, machine learning and other data science tasks. The techniques are generally used at the earliest stages of the machine learning and AI development pipelines to ensure accurate results.

Now, we apply some of preprocessing steps:

Cleaning

Consists of removing the less useful parts of a text through any HTML tag removal and dealing with capitalization characters and other details. We also removed the unwanted and repeated parts at the beginning of each sentence in the answers column.

- Remove any HTML tags.
- Convert all white spaces (tabs, etc.) to a single white space.
- Remove punctuation like (!"?[]).
- Converting to lower case: Text contains different capitalization that reflects the beginning of sentences, so we convert every character into lower case.

Tokenization

The processing of splitting text into sentences of words or characters, but we select the splitting into words, so it gets the text into an easy format to convert to raw numbers. There are two types of tokenization:

1. Word tokenization is used to separate words with a distinct space character. Also, it may tokenize multi-word expressions.

2. Sentence Tokenization/Segmentation: This is commonly performed based on punctuations such as ",.?!" as they tend to mark the sentence boundaries.

Pad sequences

This function converts a list of sequences into a 2D NumPy array of shapes (number of samples, number of timesteps). The number of time steps is either the max length argument if provided, or the length of the longest sequence in the list. Sequences that are shorter than the number of time steps are padded with the value until they are several time steps long.

3.2 Arabic chatbot

To be able to build a conversational Arabic chatbot, we will apply the NLP life cycle (data extraction, data preprocessing, tagging, normalization(noise), text to number, word embedding and neural network model evaluation).

Motivation

- Closed source
- Lack of flexibility
- Fragmented standards
- Fragmented packages

3.2.1 Arabic Dataset

As we want to build a multilanguage conversational AI chatbot, we have two options:

1. The first thing that we need to translate the English dataset to the Arabic dataset by downloading the translation library and using the apply function to translate the dataset.

2. Use The Arabic dataset is specialised in the fashion category and it contains 13K questions. The format of the dataset was text, so we converted it into CSV. We considered the question column as an input and the answer column as a target. It contains 596 duplicated values, and it doesn't have any null values.

3.2.2 Data preprocessing

If the textual data is not properly cleaned or processed, incorrect words, punctuations and URLs are added to the data. The performance will be impacted when we create static or dynamic embeddings and analyse the sentence or word vectors. In the context of embeddings, (and subsequently models), we will find that if we do not remove these inconsistencies, the vectors will not be properly placed. For example, if we apply a SOTA language embedding such as GPT-2 on unclean data containing such redundancies, the tokenizer will create separate tokens for them; which will lead the model to associate certain weights with them. These add to the redundancy and increase the complexity of the model. When we will be extracting individual entries (word or sentence vectors), then these redundancies get added to the vector space and interfere with different metrics such as semantic similarity, classification, question answers, etc.

Cleaning

Eliminating the redundancies such as punctuation and estimating the meaningfulness of the data.

- Remove punctuation by defining Arabic punctuation and then applying a small function to remove punctuation.
- Remove noise, which is called normalise Arabic sentences and it includes removing Tashdid, Fatha, Tanwin Fath, Damma, Tanwin, Damm, Kasra, Tanwin, Kasr, Sukun and Tatwil/Kashida.
- Remove extra whitespace, longation, diacritics and repeating characters.
- Tokenization from keras is used for vectorizing a text corpus. For this, each text input is converted into an integer sequence or a vector that has a coefficient for each token in the form of binary values.
- Make pad of sequence to encode all our sentences with the same length.

3.3 The Proposed model

After the English and Arabic dataset is cleaned, we will pass the question as an input and the answer as a target into the model. We use the seq2seq model, also known as the encoder/decoder model, which uses Long Short Term Memory (LSTM) for text generation from the training corpus. The seq2seq model is also useful in machine translation applications. What does seq2seq or encoder/decoder model simply do? It predicts a word given in user input and uses the probability of that word occurring to predict each of the following words. So the seq2seq model consists of three parts :

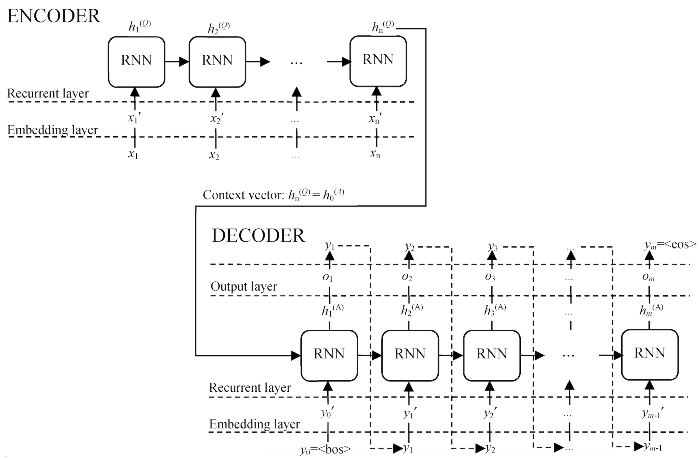


Figure 3.9: Sequence-to-Sequence Model[9].

- Embedding can be of type word or character or other forms such as bigram, trigram or even a hybrid between them. The function of the embedding layer is to convert the input into a vector of real numbers that represents the input.

- The encoder block is an LSTM cell. It is fed in the input-sequence over time, and it tries to encapsulate all its information and store it in its final internal states h (hidden state) and c (cell state). Then the internal states are passed onto the decoder part, which uses them to try to produce the target-sequence. The function of the encoder is to process (encode) the input embeddings which are variable length vectors and produce intermediate states which are fixed lengths vectors. This is the context vector that we were earlier referring to.

- The decoder block is also an LSTM cell. The main thing to note here is that the initial states (h, c) of the decoder are set to the final states (h, c) of the encoder. These act as the context vector and help the decoder produce the desired target-sequence. Each recurrent unit accepts a hidden state from the previous unit and produces an output from it as well as its own hidden state.

The output sequence in a question-answering problem is a collection of all the words from the answer.

The encoder outputs a final state vector (memory) that is the initial state of the decoder. We train the decoder using a method called teacher forcing. This allows us to predict the next word in the target sequence given by the previous word. State is passed through the encoder to each layer of the decoder. "Hi", "like", "are" and "you" are called input tokens, and "I", "am" and "good" are called target tokens. The probability of the token 'am' depends on the previous word and the state of her encoder. Add an ¡END¿ token to tell the decoder when to stop.

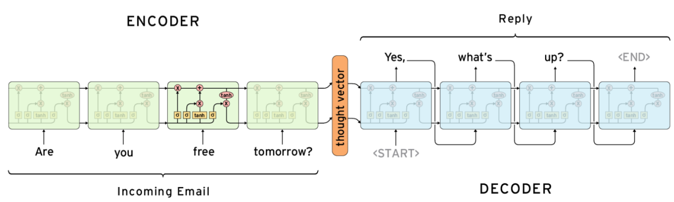


Figure 3.10: Sequence-to-Sequence Model[18].

LSTMs are specifically designed to avoid long-term dependency problems. LSTM also provides a solution to the vanishing/exploding gradient problem. Long-term memorization of information is effectively their default behaviour and not something they learn the hard way. All recurrent neural networks are in the form of chains of repeating modules of neural networks. In a standard RNN, this repeating module has a very simple structure. A long-short-term memory network consists of four different gates for different purposes, as described below:

1. Forget Gate(f): This item determines how far to forget previous data.

2. Input Gate(i): It determines the extent of information to be written onto the Internal Cell State.

3. Input Modulation Gate(g): It is often considered as a sub-part of the input gate and much of the literature on LSTM's does not even mention it and assumes it is inside the input gate. It is used to modulate the information that the input gate will write onto the internal state cell by adding non-linearity to the information and making the information zero-mean. This is done to reduce the learning time as zero-mean input has faster convergence. Although this gate's actions are less important than the others and it is often treated as a finesse-providing concept, it is good practise to include this gate into the structure of the LSTM unit.

4. Output Gate(o): It determines what output(next hidden state) to generate from the current Internal Cell State. The basic work-flow of a Long-Short Term Memory Network is similar to the work-flow of a Recurrent Neural Network, with the only difference being that the Internal Cell State is also passed forward along with the Hidden State.

Algorithm steps

- Load the dataset.
- Pass the input.
- The input is split into a sequence of words.
- Each input is given to each LSTM cell.
- For each input vector, the text is converted into vector format by the embedding layer.

- At the end of the encoder phase, the last cell will give an output, which is the context vector.
- This context vector is now passed to the decoder.
- In the decoder phase, for every LSTM cell, there is an output and it will act as an input to the next cell.
- This process is done until we reach the ¡END¿ token.
- After reaching the End of String token, the decoder cell outputs are combined together and given as the output of the chatbot.

To define the architecture of the model, we use different classes available in the Keras Python library.

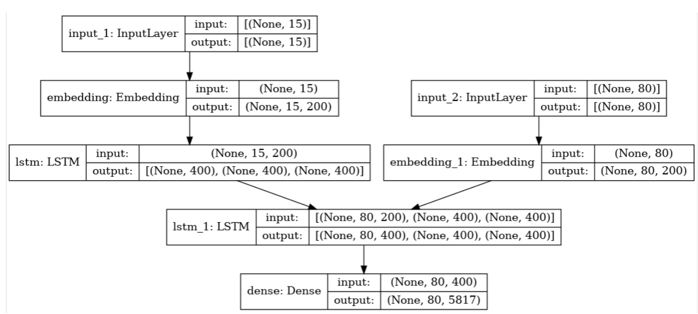


Figure 3.11: The architecture of the English model.

The input is the question column, and the answer column is the target. Firstly, we have the encoder input shape = (None, max length of the question) and the decoder input shape = (None, max length of answers). After that, we have two embedding layers that enable us to convert each word into a fixed-length vector of a defined size. We pass the padded sequences as input to the embedding layer with shape (input dim

= size of the vocabulary, output dim = length of the vector for each word). Then, we insert LSTM layers with dense layers to predict the label.

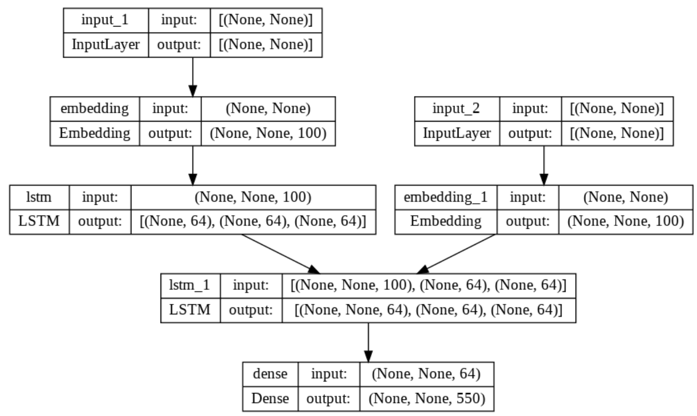


Figure 3.12: The architecture of the Arabic model.

For figure 3.12, it's the same architecture as the English model but with an input shape (none,none) to be any scalar number, so you use this model to infer an arbitrarily long input.

Now our model is ready for training, which can be done by calling the fit method on training dates. The training phase is one of the most important phases in the construction of our model. The time required to train the model depends on the computational ability of the machine on which the model is built and the number of epochs.

An epoch refers to up to one training cycle across the entire dataset. We can form our data for any number of epochs that we desire. The performance of our model depends on the number of epochs. In addition to the epochs, the batch size is also an important factor in determining the model's performance. There is no hard and fast rule on how these values are chosen, just try different combinations at random and choose the optimal values.

3.4 Simple DialogFlow-chatbot

Dialogflow is a natural language understanding platform that has the easiest way to design and integrate a conversational user interface into your mobile app, web application, device, bot, interactive voice response system, and so on. Using Dialogflow, you'll be able to provide new and interesting ways for users to interact with your product. Dialogflow is part of the Conversational AI offering within Google Cloud.

Components of building chatbots using the English language

Those parts consist of three sections (contact info, order pizza intent, special intent, training phrases, action and parameters, and text response).

- Create an agent (pizza bot).
- Create intents (when a user says something similar to a training phrase, Dialogflow matches it to the intent).
- Entities part consists of three sections(pizza crust style, pizza ingredient and pizza size).

3.5 Language detection

We use a dataset that contains English and Arabic sentences with the label English or Arabic. We remove punctuation from English sentences, and we remove English letters from Arabic sentences.

Then, we make tf-idf, which is split into two parts : Term frequency works by looking at the frequency of a particular term you are concerned with relative to the document (number of times the word appears in a document) and IDF (inverse

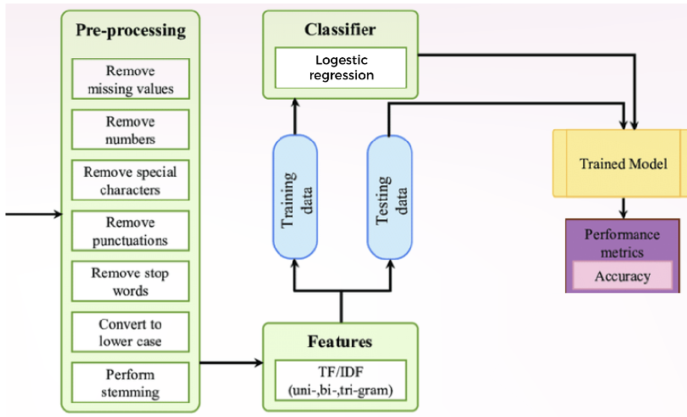


Figure 3.13: Logistic regression model [8].

document frequency) looks at how common (or uncommon) a word is amongst the corpus. The reason we need IDF is to help correct for words like "of," "as," "the," etc. since they appear frequently in an English corpus. Thus, by taking inverse document frequency, we can minimise the weighting of frequent terms while making infrequent terms have a higher impact. Following that, we built a logistic regression model to allow the chatbot to respond and categorise the sentences as Arabic or English.

3.6 Deploy chatbot

Now we will design an online shopping website where the user will input the question and the data will be given to the model. Based on the training given to the model, it will return the answer. We built it by using HTML, CSS and Java Script and merged our chatbot with the website by using Flask.

HTML (markup language) is the language used for creating detailed instructions concerning style, type, format, structure and the makeup of a web page before it gets displayed to us.

CSS (design language) stands for Cascading Style Sheets, and you use it to improve the look of a web page. By adding thoughtful CSS styles, you make your page more attractive and creative for the end user to view and use.

Chapter 4

Result

The result of the chatbot is not accurate when we used sequential model with LSTM layers only and it was very slow, so we decided to use a sequence-to-sequence model to generate accurate answers in addition to LSTM layers, so our results have improved a lot.

Another thing that helped us to improve our results was Turning. We tried different epochs and different batch sizes. For the first English dataset, it doesn't accurate for the chatbot because it contains long sentences and this is one of the chatbot's limitations. Finally, the accurate response for the model with the number of epochs = 50 and batch size= 32 for English data set and we use only 6000 because we don't have enough memory resources. For the Arabic dataset, the result is very accurate with the number of epochs = 23 and the batch size = 32.

The accuracy in the second Arabic dataset is better than the accuracy of the first Arabic dataset because we used Google Translate in the first dataset and there are many issues with using Google Translate:

- The meaning can be lost in translation because there is no way to incorporate context.

- It often produces translations that contain significant grammatical errors.
- It doesn't have a system to correct for translation errors.
- The language pair affects the translation's quality.

	Accuracy	Loss
The first English dataset	0.85	0.21
The second English dataset	0.94	0.024
The first Arabic dataset	0.1	0.02
The second Arabic dataset	0.97	0.038
Language detection model	100	

Table 4.1: Scores for comparing results.

4.1 Simple DialogFlow-chatbot

As we mentioned in chapter 3, we built a simple chatbot by using DialogFlow. This chatbot makes it easier for the customer to purchase. Here are screenshots that explain the chatbot and how to have a conversation with it.

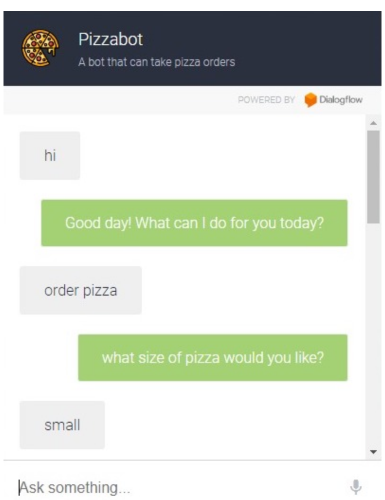


Figure 4.1: The outcome of dialogflow.

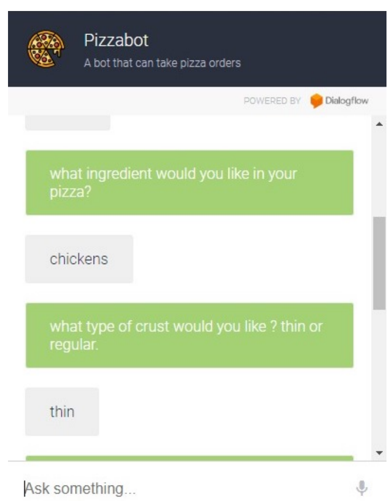


Figure 4.2: Details of the order.

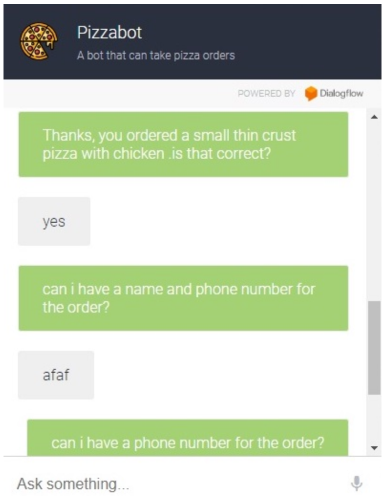


Figure 4.3: The confirmation.

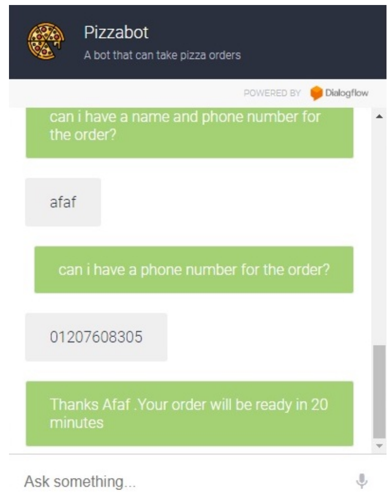


Figure 4.4: Customer's information.

The figures displayed, What is the customer's order? What are the ingredients of the order and its crust? Also, display how the chatbot confirms the order. In the last step, the chatbot asks for the customer's information and determines the time of its arrival.

Here[1] you can find our simple chatbot.

4.2 Multilingual chatbot

Finally, our chatbot is called EMO (E-commerce Multilingual Chatbot) we merge all models in flask and make a website.

In this figure 4.5, we built a website with an icon chatbot on the down left side. In the outcome conversation, the chatbot replies automatically in English or Arabic text because we made a language detection model to make it easier to answer quickly without the need to select a language first.

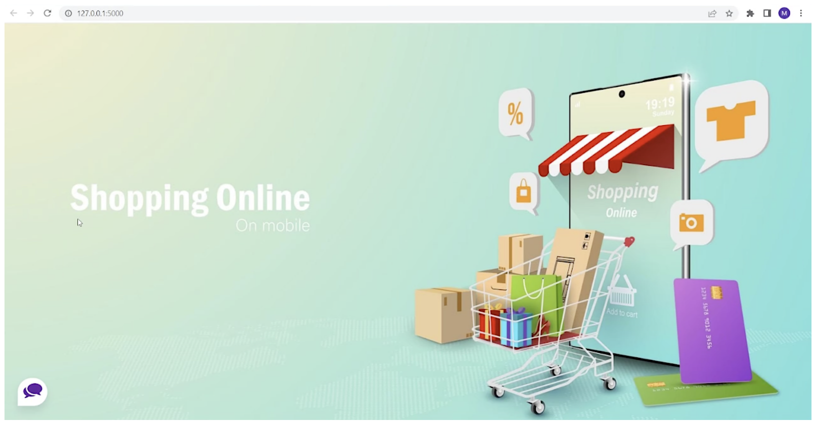


Figure 4.5: Online shopping website.

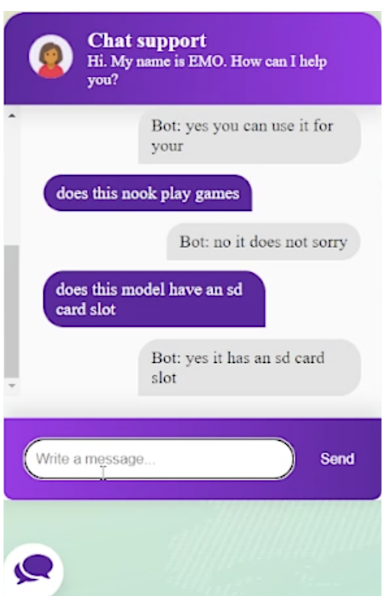


Figure 4.6: The outcome conversation.

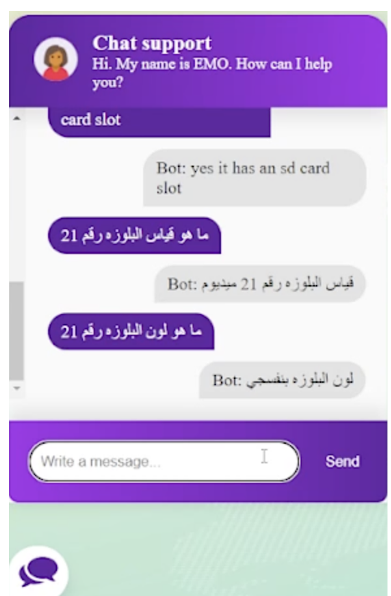


Figure 4.7: English and Arabic conversation.

Chapter 5

Discussion

5.1 Challenges

Training is a long process that requires more processing power and a configured computing machine. Another problem is finding good hyperparameters to optimise the model.

Developing a generic chatbot is very difficult. The model used in this experiment is for machine translation, and dialogue generation is treated as a translation problem where the history of previous conversations is not considered.

Therefore, the performance of the model may be limited for long conversations. Many of the results were iterative and general. Due to the lack of real-time high-quality data, the chatbot's performance was somehow inferior to that of a human interaction. Many utterances were discarded due to length or other discrepancies.

5.2 Limitations

The truth is, chatbots are machines not people. Although you can attempt to give them a casual tone, chatbots will never understand human context. Chatbots are great at providing facts and data. But they can never really create an emotional connection with the customer.

And establishing an emotional connection can be a crucial factor in today's competitive environment. They often fail in long conversations and are less relevant in dialogue generation also, they Can't make decisions. Most of these chatbots are developed for limited(closed) domains.

Most of them use simple rule-based techniques. They behave well in highly structured question and answer sessions as well as in conversational modes. They have zero research skills and only have the answers to the available queries; they can't research for anything outside of their domain.

5.3 Conclusion

Chatbots are the best digital marketing tools. We discovered that seq2seq is the best for building chatbots that contain the LSTM model. The encoder and decoder are built using Keras with Tensorflow. LSTM is a special type of RNN that is capable of learning long-term dependencies. The Encoder-Decoder model contains a two-part encoder, which is the vector representation of the input sequence and maps it to an encoded representation of the input. This is then used by the decoder to generate an output. We found that 80 percent of customers are satisfied with their interactions with chatbots.

5.4 Future Work

Add an attention layer to the seq2seq model to focus on important parts. Also, at each decoder step, it decides which source parts are more important. In this setting, the encoder does not have to compress the whole source into a single vector; it gives representations for all source tokens.

Bibliography

[1] pizzabot. https://console.dialogflow.com/api-client/demo/embedded/d76390d4-a9d1-4b82-819d-100a947fe02e.

[2] A. Abdellatif, K. Badran, D. Costa, and E. Shihab. A comparison of natural language understanding platforms for chatbots in software engineering. *IEEE Transactions on Software Engineering*, PP:1–1, 05 2021. doi: 10.1109/TSE.2021.3078384.

[3] Amazonqa. Amazonqa/amazonqa: Evidence-based qa system for community question answering., . URL https://github.com/amazonqa/amazonqa.

[4] Amazonqa. Amazonqa/amazonqa: Amazon question/answer data., . URL https://jmcauley.ucsd.edu/data/amazon/qa/.

[5] S. Angelov and M. Lazarova. E-commerce distributed chatbot system. In *Proceedings of the 9th Balkan Conference on Informatics*, BCI'19, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450371933. doi: 10.1145/3351556.3351587. URL https://doi-org.proxy.queensu.ca/10.1145/3351556.3351587.

[6] S. Ayanouz, B. A. Abdelhakim, and M. Benhmed. A smart chatbot architecture based nlp and machine learning for health care assistance. In *Proceedings of the 3rd International Conference on Networking, Information Systems amp; Security*, NISS2020, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450376341. doi: 10.1145/3386723.3387897. URL `https://doi.org/10.1145/3386723.3387897`.

[7] G. G. Understanding rasa: A look under the hood of rasa nlu, May 2020. URL `https://medium.com/@gauravgpunjabi/getting-started-with-machine-learning-ea0dc29e3ff6`.

[8] I. Ismiguzel. Applying text classification using logistic regression. URL `https://medium.com/analytics-vidhya/applying-text-classification-using-logistic-regression`.

[9] J. Kapočiūtė-Dzikienė. A domain-specific generative chatbot trained from little data. *Applied Sciences*, 10(7), 2020. ISSN 2076-3417. doi: 10.3390/app10072221. URL `https://www.mdpi.com/2076-3417/10/7/2221`.

[10] A. R. D. B. Landim, A. M. Pereira, T. Vieira, E. de B. Costa, J. A. B. Moura, V. Wanick, and E. Bazaki. Chatbot design approaches for fashion e-commerce: an interdisciplinary review. *International Journal of Fashion Design, Technology and Education*, 15(2):200–210, 2022. doi: 10.1080/17543266.2021.1990417. URL `https://doi.org/10.1080/17543266.2021.1990417`.

[11] M. Mamatha et al. Chatbot for e-commerce assistance: based on rasa. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 12(11):6173–6179, 2021.

[12] A. F. Muhammad, D. Susanto, A. Alimudin, F. Adila, M. H. Assidiqi, and S. Nabhan. Developing english conversation chatbot using dialogflow. In *2020 International Electronics Symposium (IES)*, pages 468–475, 2020. doi: 10.1109/IES50839.2020.9231659.

[13] D. Murphy. 6 reasons why chatbots are the future of marketing, Sep 2021. URL `https://www.business2community.com/marketing/6-reasons-why-chatbots-are-the-future-of-marketing-02429815`.

[14] D. A. Nayyar. Chatbots and the open source tools you can use to develop them, Jan 2019. URL `https://www.opensourceforu.com/2019/01/chatbots-and-the-open-source-tools-you-can-use-to-develop-them/`.

[15] M. Rakhra, G. Gopinadh, N. S. Addepalli, G. Singh, S. Aliraja, V. S. G. Reddy, and M. N. Reddy. E-commerce assistance with a smart chatbot using artificial intelligence. In *2021 2nd International Conference on Intelligent Engineering and Management (ICIEM)*, pages 144–148, 2021. doi: 10.1109/ICIEM51511.2021.9445316.

[16] S. Ranavare and R. Kamath. Artificial intelligence based chatbot for placement activity at college using dialogflow. 12 2020.

[17] S. S. Ranavare and R. Kamath. Artificial intelligence based chatbot for placement activity at college using dialogflow. *Our Heritage*, 68(30):4806–4814, 2020.

[18] K. Shridhar. Generative model chatbots. URL `https://medium.com/botsupply/generative-model-chatbots-e422ab08461e`.

[19] S. A. Thorat and V. Jadhav. A review on implementation issues of rule-based chatbot systems. In *Proceedings of the International Conference on Innovative Computing & Communications (ICICC)*, 2020.

[20] A. O. Victoria Oguntosin. Development of an e-commerce chatbot for a university shopping mall. *Applied Computational Intelligence and Soft Computing*, 14, 2021. URL `https://doi.org/10.1155/2021/6630326`.