Saracevic Mirhet, BSc

Potential of Bots for Encyclopedias

Master's Thesis

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme: Computer Science

submitted to

Graz University of Technology

Supervisor

Priv.-Doz. Dipl.-Ing. Dr.techn. Martin Ebner

Co-Advisor

Dipl.-Ing. Markus Ebner

Institute of Interactive Systems and Data Science

Head: Univ.-Prof. Dipl-Ing. Dr. Stefanie Lindstaedt

Graz, October 2019

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

Date	Signature

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Dissertation identisch.

Datum	Unterschrift

Abstract

Every day people search for information on the world wide web. Over time, new technologies have been developed that could facilitate the information search. Chatbots have become a hot topic in the last few years. The capability to process and understand natural languages, and their broad application area has made them very popular. In the field of information retrieval, chatbots are changing the way users search for information.

This work provides a short introduction, the motivation behind this master thesis, the problem statement and the goal of the work. Furthermore, history and background, state of the art, types and applications of chatbots are outlined. In the scope of this thesis, a chatbot was developed for the online encyclopedia Austria Forum. The chatbot provides functionalities for the information search and the information upload in the geography category of the online encyclopedia. The description of design decisions, architectural structure and implementation of the chatbot is described. The last part of this document is devoted to the comparison between search results of the existing search engine and the chatbot, future work and conclusion.

Kurzfassung

Jeden Tag suchen Menschen nach Informationen im World Wide Web. Im Laufe der Zeit wurden neue Technologien entwickelt, welche die Informationssuche erleichtern konnten. Chatbots sind in den letzten Jahren ein heißes Thema geworden. Die Fähigkeit, natürliche Sprachen zu verarbeiten und zu verstehen und ihr breites Anwendungsgebiet haben sie sehr populär gemacht. Im Bereich des Information Retrievals verändern sie die Art und Weise, wie Benutzer nach Informationen suchen.

Diese Arbeit bietet eine kurze Einführung, in der die Motivation für diese Masterarbeit, die Problemstellung und das Ziel der Arbeit beschrieben werden. Weiterhin werden die Geschichte und der Hintergrund, der Stand der Technik, verschiedene Typen und Anwendungen von Chatbots dargestellt. Im Rahmen dieser Arbeit wurde ein Chatbot für die Online-Enzyklopädie Austria Forum entwickelt. Der Chatbot bietet Funktionalitäten für die Informationssuche und das Hochladen der Informationen in der Geographie Kategorie der Online-Enzyklopädie. Die Beschreibung der Designentscheidungen, der architektonischen Struktur und der Implementierung des Chat-

bots wird präsentiert. Der letzte Teil dieses Dokuments beschäftigt sich mit dem Vergleich der Suchergebnisse der bestehenden Suchmaschine und des Chatbots, der zukünftigen Arbeit und dem Fazit.

Acknowledgements

I would like to thank my co-advisor Dipl.-Ing Markus Ebner and my supervisor Dr. Martin Ebner for their support during the research. A thank you goes to Dr. Hermann Maurer, who came up with the idea for this master thesis. I would also like to thank Gerhard Wurzinger for providing the API to Austria Forum and his support during the implementation of the practical part.

A big thank you goes to my parents, siblings, girlfriend and friends for the enormous support both during the whole study and during the research. Special thanks go to my uncle and his family who gave me the opportunity to study and live in Austria. Completing my studies would not be possible without all these people.

Contents

List of Figures

List of Tables

Listings

1 Introduction

The online encyclopedias, also called digital encyclopedias, provide a large amount of information on various topics and areas. The number of articles in online encyclopedias is constantly increasing. The computers are mostly used for information search and communication. It is known that finding relevant information has always been one of the biggest issues of search engines (Croft, Metzler, and Strohman, 2009). The new technologies, known as chatbots, combine the two activities of searching and communicating. The main focus of this work will be on information search and relevance of search results. In addition, further potentials of chatbots in online encyclopedias will be examined.

This chapter describes the motivation behind this master thesis, the problem statement and the final goal of the work. Chapter 2 provides an overview of the history and background, types of chatbots and their usage nowadays. Chapter 3 describes the practical part of the master thesis. It lists tasks and requirements, continues with architecture and design, evaluation of the tools used, and ends with the concrete implementation of the chatbot.

Chapter 4 will compare the search results of an existing search engine in an online encyclopedia and the chatbot. Chapter 5 will show the future work. The last chapter 6 contains the conclusion of this work.

1.1 Motivation

Conversational agents, dialogue systems or simply chatbots have become very popular in the past few years. Graphical user interfaces provided by chatbots are very familiar to users, as they have been used for the human to human communication in applications such as mIRC[1], Facebook Messenger[2], WhatsApp[3], Viber[4]. Therefore, no additional knowledge is needed in order to use chatbots.

One of the biggest advantages of chatbots is the wide range of practical applications. Among other practical applications, chatbots can be used as an information retrieval tool, especially for information search. The user communicates with chatbot in a natural language and can therefore formulate the questions better. The chatbot evaluates the question and provides an answer to the user. If the chatbot cannot determine what the user is searching for, it will ask questions to collect more information. Thus, the chatbot gains a better understanding and can deliver better results.

[1] https://www.mirc.com/ (last visited on 02 June 2019)
[2] https://www.messenger.com/ (last visited on 02 June 2019)
[3] https://www.whatsapp.com/ (last visited on 02 June 2019)
[4] https://www.viber.com/ (last visited on 02 June 2019)

1.2 Problem Statement

The online encyclopedias are big sources of information.
One of the ways to get the desired information is to navigate through the website. On digital encyclopedias, individual pages are linked to each other. Following these links could bring the user to the desired information. The most commonly used way is the use of the search functionality of the online encyclopedia. The search is based on keyword matching. The user enters a query and receives search results in form of a list. The user needs to browse the list in order to find desired information.
Navigating to and searching for information on online encyclopedias can be time consuming. The information retrieved can be noisy and does not satisfy the user's needs. Since the first personal computers and search engines were built, finding relevant information has been one of the big issues. The relevance is measured by how well the retrieved information satisfies the query of a user. Despite the improvement of the information search techniques the issue of relevance remained.

1.3 Goal

The aim of this project is to develop a chatbot that will serve as an information retrieval tool. The geography area of the online encyclopedia Austria

1 Introduction

Forum[5] is used as a knowledge base. The task of the system is to enable a communication with the users in a natural language. Users will actively participate in information search by providing questions and additional information if needed to the system. In this way the chatbot will be able to deliver better search results. Another task of the system is to enable users to contribute to the online encyclopedia. Users will be able to upload audio, video and pictures. In this way the knowledge base gets extended and enriched.

[5] https://austria-forum.org/af/Geography/Index (last visited on 02 June 2019)

2 State of the Art

This chapter describes definitions of the term chatbot. It also depicts the background and development of chatbots through history. It provides an overview of approaches and practical applications. The last part is dedicated to the chatbots that are used as information retrieval tools.

2.1 What is Chatbot?

Chatbot is a software program with which users communicate using natural languages.

"*chatbot* a computer program designed to have a conversation with a human being, especially over the internet" (Cambridge Dictionaries Online, 1999)

"*chatbot* a computer program in the form of a virtual e-mail correspondent that can reply to messages from computer users" (dictionary.com, 1995)

Chatbots are also called conversational agents, dialog based systems, virtual agents or machine conversation systems depending on the area where they find their application (Shawar and Atwell, 2007). The commonly provided chatbot communication is either text or voice based. Chatbots can be seen as graphical user interfaces to information, data or service. Types of the chatbots are listed and described in section 2.4.

2.2 Background and History

The first conversational system was developed by Joseph Weizenbaum in 1966 at the Massachusetts Institute of Technology. This system, known as ELIZA, has enabled users to communicate with computers in natural language. It was programmed with specific scripts in order to be able to provide responses similar to those provided by a real person. The goal was to imitate the human conversation. For example, the script called DOCTOR was programmed to mimic a psychotherapist.
The ELIZA functionality was based on the keyword matching algorithm. The first step of the ELIZA algorithm was finding a keyword in the question. If the keyword was found, the sentence was transformed and manipulated according to the rule associated with the found keyword. As an example, the sentence with the keyword "mother" can be used. The message from ELIZA can be "Tell me more about your family". If the input sentence did not match any rule, ELIZA responded with fixed answers, for example "Very interesting. Please go on." or "Can you think of a special example?". The aim

was to encourage the patient to continue with the conversation. ELIZA could not communicate truly understanding - it matched the found keywords with predefined rules. Even though, some users had long conversations with ELIZA thinking they were talking to a real person. (Weizenbaum, 1966; Shawar and Atwell, 2007)

The next famous chatbot was PARRY, developed in 1972 by the psychiatrist Keneth Colby. The goal of PARRY was to simulate a paranoid individual. This program was more complex and advanced than ELIZA. ELIZA and PARRY met each other and had a conversation in 1972 at the first International Conference on Computer Communications (ICCC). (Computer History Museum, 1996)

Another famous chatbot called A.L.I.C.E. was developed in 1995 by Wallace. The abbreviation A.L.I.C.E. stands for Artificial Linguistic Internet Computer Entity. The functionality of the chatbot is based on a pattern matching algorithm. The brain, or in other words, the knowledge base of A.L.I.C.E consists of Artificial Intelligence Markup Language (AIML) objects. AIML is based on Extensible Markup Language (XML) and is A.L.I.C.E's own scripting language. Every AIML object contains heuristic conversation rules. A.L.I.C.E was firstly build to amuse and entertain the users. It is an open source software that enables users to create their own pattern matching rules and in this way build their own chatbots. (Shawar and Atwell, 2007)

In the past 60 years many chatbots were built inspired by ELIZA. The reason for the increased development of chatbots was the fact that the people having

the conversation with ELIZA thought they are talking to a real person. At the beginning of the 90s a contest for the chatbots called Loebner Prize[1] was founded. At this competition the chatbots were subjected to the Turing Test. (Dale, 2016)

The Turing Test consists of an individual having text based conversations with two parties. The interactor does not know which party is a machine and which is a human. It is free to conduct the conversation in any direction. At the end of the conversation the interactor has to decide which party is a machine and which one is a human. If the decision is wrong meaning the machine has fooled the interactor, the machine passes the Turing Test (Turing, 1950). The Loebner Prize had a big impact on building chatbots and has increased the interest in the area of artificial intelligence (AI).

All the chatbots mentioned in this chapter were used for a chatting purpose, to amuse or to entertain the user. The growth of the use of personal computers had a big impact on the development and widespread of conversational agents (Wilks, 1999). The development and improvement of graphical user interfaces also increased the human desire to communicate with personal computers in the same way they do with human beings (Zadrozny et al., 2000). The increased amount of data was a trigger for the improvement of certain algorithms, methods and techniques in data processing (Shawar and Atwell, 2007). All these factors enabled the development of chatbots that are not only used to amuse or to entertain the users, but also to find a role in

[1] https://en.wikipedia.org/wiki/Loebner_Prize (last visited on 18 June 2019)

many practical applications in different areas.

2.3 Present Day

The drivers in the past that led to the increased interest in chatbots were discussed in section 2.2. The year 2015 was a turning point that increased the interest in building chatbots again. This was caused because the use of messaging applications surpassed the use of social networks. Messaging applications are considered a fertile ground for the chatbots.

Peter Rojas, Entrepreneur in Residence at Betaworks said "People are now spending more time in messaging apps than in social media and that is a huge turning point. Messaging apps are the platforms of the future and bots will be how their users access all sorts of services" (Chatbot Magazine and Schlicht, 2016).

There are 2.1 billion users of messaging applications. Chatting became the most usual way of communication in the last decades. "90% of our time on mobile is spent on email and messaging platforms. I would love to back teams that build stuff for places where the consumers hang out!" said Niko Bonatsos, Managing Director at General Catalyst (Chatbot Magazine and Schlicht, 2016).

Many big companies saw an opportunity for e commerce, marketing or

customer service in chatbots. Facebook[2], Slack[3], Viber[4] and Kik[5] provided tools for chatbots integration in their messaging platforms. Facebook[6] and Microsoft[7] launched the platforms for creation and integration of the chatbots in 2016. A year later the number of chatbots on Facebook Messenger[8] was over thirty thousand. Other platforms also recorded increased numbers of the chatbots launched. (Dale, 2016)

Artificial Intelligence (AI) became better due to the upgrading in machine learning. The computing power available nowadays helped in improving the machine learning algorithms such as artificial neural networks. The artificial neural networks are algorithms that simulate the biological neural networks in order to make the computers able to learn. The neural networks require a large amount of data for the learning process. It needs to be trained with examples of human conversation. The availability of data and the data mining techniques contributed a lot to the improvement and development of the neural networks. It widened the range of the chatbot applications. Because of that it is possible to build more complex chatbots that are able to carry out a various number of tasks and have more than one purpose (Shawar and Atwell, 2007).

[2] https://developers.facebook.com/docs/workplace/integrations/custom-integrations/bots (last visited on 18 June 2019)
[3] https://api.slack.com/bot-users (last visited on 18 June 2019)
[4] https://developers.viber.com/ (last visited on 18 June 2019)
[5] https://www.kik.com/ (last visited on 18 June 2019)
[6] https://www.facebook.com/ (last visited on 23 July 2019)
[7] https://www.microsoft.com/en-us/ (last visited on 23 July 2019)
[8] https://www.messenger.com/ (last visited on 18 June 2019)

The chatbots are able to understand the users' requests and to provide reasonable answers. Examples of artificial intelligence based chatbots include amongst others Siri[9], Alexa[10], Google Assistant[11] and Cortana[12].

2.4 Types of Chatbots

A strict classification of the chatbots is not possible. There are many parameters which the classification depends on. In this section the following three parameters are considered: knowledge domain, approach and conversation length.

2.4.1 Knowledge Domain

The knowledge domain can be considered the brain of the chatbots. The larger the knowledge domain the more answers the chatbot can provide. It can be differentiated between closed and open domain chatbots. (Nimavat and Champaneria, 2017)

The chatbots with a closed domain are considered as very limited. These chatbots are able to provide answers regarding one specific topic or to carry out one specific task. If the user asks something that does not have anything

[9] https://www.apple.com/siri/ (last visited on 15 August 2019)
[10] https://www.alexa.com/ (last visited on 15 August 2019)
[11] https://assistant.google.com/ (last visited on 15 August 2019)
[12] https://www.microsoft.com/en-us/cortana/ (last visited on 15 August 2019)

in common with the particular topic the chatbots will fail to answer that question. The examples for this kind of chatbots are technical customer support and shopping assistant chatbots. (Wildml and Britz, 2016)

In comparison to closed domain chatbots, the open domain chatbots are able to provide answers about general topics. They can carry out several tasks and provide multiple services. The communication can go in any direction the users want. One example for this kind of chatbot is A.L.I.C.E that has won the Loebner Prize already three times. Of course it is not possible nowadays to develop chatbots that are able to provide reasonable answers to every users question. (Wildml and Britz, 2016)

2.4.2 Approach

Basically there are two approaches when developing chatbots. There are rule based approaches or approaches based on the concept of machine learning.

Rule based chatbots, also called script based chatbots operate based on the predefined set of rules. These rules depend on the task of the chatbot and are written by the developer. When communicating with the chatbot, it analyses every question and searches for the keywords. Then the keywords are matched against predefined rules and the response is retrieved from a database. If there is no match between keywords and rules an error message or a predefined response is provided. The set of rules can be extended

and edited. This can improve the functionality and increase the number of tasks. The rule based approach is appropriate for question answer systems (Wildml and Britz, 2016). An example is the chatbot that is used as natural language interface to Arabic Web QA. (Shawar, 2011)

The chatbots using machine learning can be classified into two groups depending on the model they are based on. The first group contains retrieval based chatbots. These chatbots are equipped with a knowledge base consisting of possible user questions and their responses. When engaged with the question, the chatbot searches for the most relevant response in the knowledge base and provides the user with it. They do not generate any new responses. Retrieval based chatbots need to be trained with a large number of datasets in order to be able to answer a wide range of questions. Chatbots based on generative models belong to the second group. As the name already states these chatbots generate their responses instead of retrieving them from the knowledge base. In comparison to retrieval based chatbots, they can handle the questions previously unknown to them. Both of the chatbots are more efficient than rule based chatbots and also considered intelligent. They are capable of learning from the user inputs and get more intelligent with time. (Wildml and Britz, 2016)

2.4.3 Conversation Length

The conversation length is an important parameter for the chatbot classification. Short conversations are typical for the chatbots that provide various

information to the users. Engaged with the question, the chatbot's task is to respond to the user with an appropriate answer. The earlier questions or conversation do not matter and are not kept track of (Wildml and Britz, 2016). The CNN news chatbot on the Facebook Messenger[13] fits in this group of chatbots. Another example would be the FAQchat, trained with the Frequently Asked Questions (FAQ) at the University of Leeds. (Shawar, Atwell, and Roberts, 2005)

A long conversation means that also earlier conversations should be taken into account in order to properly define the contexts. It also means getting as much information as possible from the user. This type of conversation is common among the chatbots that are used for customer support. (Wildml and Britz, 2016)

2.5 Practical Applications

2.5.1 Customer Service

The companies saw an opportunity in using chatbots for customer service. The working time for chatbots is 24 hours per day, seven days per week and they give answers immediately. Of course the chatbots might not be capable of answering all the questions, but handling the repetitive requests

[13] https://www.messenger.com/ (last visited on 23 July 2019)

from users should not be a problem. The requests chatbots cannot understand are forwarded to real humans for further processing. The chatbots gather information data from the user and thus can personalize customer experiences. (Accenture Interactive, 2016)

2.5.2 Marketing

There are several benefits in using chatbots for marketing purposes. The social platforms are a great place to find potential customers. Here it is also important to learn the customers' interests and needs in order to get to know them better and to be able to make better recommendations. These chatbots need to be based on AI to be able to understand the customer and build great user experiences.

2.5.3 Finance

Chatbots in finance have a great potential. They are not there only to make money transfers, but also to provide information about customer accounts like account amount, transactions and so on. It also can be used for credit applications. The chatbot is able to determine whether an application is valid or not. These chatbots use AI in order to be able to solve this task.

2.5.4 Human Resources

These chatbots are used to help new employees with the onboarding process. They provide information requested by the employees to help them to adapt faster to the company environment. They also provide distinct information about payment, maternity leave, holidays, vacation and so on. The chatbots can be trained for the application and recruitment processes. In this manner it can be helpful to find potential employees. One possible task would also be the integration in transactional processes. Intel already built a chatbot called Ivy. Stella[14] is an another example.

2.5.5 E Commerce

The chatbots in this field are seen as personal shopping assistants. Through the conversations with the users the chatbots learn about their preferences and needs. This is what they use to make better and more precise product recommendations. Additionally, notifications about new products that are relevant to the user can be sent. The chatbots are able to set price alerts or help the customers choose a gift. H&M[15] and eBay[16] for instance have already launched personal shopping assistants.

[14] https://www.stella.ai/ (last visited on 23 July 2019)

[15] https://www.chatbotguide.org/h-m-bot (last visited on 23 July 2019)

[16] https://www.ebayinc.com/stories/news/say-hello-to-ebay-shopbot-beta/ (last visited on 23 July 2019)

2.5.6 Entertainment

Many chatbots were built with the purpose to simulate different personalities, to amuse and entertain the user. Mitsuku[17] is a chatbot that is built based on AIML and simulates an eighteen year old female. This chatbot has won the Loebner Prize four times. Another example is Disney's chatbot Zootopia[18]. It helps fans of the movie Zootopia to investigate and solve crimes with Lieutenant Judy Hopps. Chatbot Alpha 5[19] helps users to learn how to be a Power Ranger. It is available on Facebook Messenger[20], Kik[21] and Twitter[22].

2.6 Chabtbot as Information Retrieval Tool

In the section 2.5 an overview of the chatbots practical applications has been given. Among the other tasks, bots also deal with information retrieval. The chatbots usually provide different information to the user. There are also chatbots that gather information from users.

[17] https://www.pandorabots.com/mitsuku/ (last visited on 23 July 2019)
[18] https://chatbottle.co/bots/zootopia-for-messenger (last visited on 23 July 2019)
[19] https://www.kik.com/bots/powerrangersalpha5/ (last visited on 23 July 2019)
[20] https://www.messenger.com/ (last visited on 23 July 2019)
[21] https://www.kik.com/ (last visited on 23 July 2019)
[22] https://mobile.twitter.com/ (last visited on 23 July 2019)

"Information retrieval is a field concerned with the structure, analysis, organization, storage, searching, and retrieval of information." (Croft, Metzler, and Strohman, 2009)

The information search has always been a big challenge for the researchers. Most search engines function based on keyword matching algorithms. The improvement of Natural Language Processing (NLP) and Natural Language Understanding (NLU) techniques as well as progress in Artificial Intelligence (AI) have improved information retrieval . They opened the door for chatbots to be used in information retrieval, since the user can formulate queries in natural language. The famous bots in this field are chatbots to Question-Answering (Q&A) systems and Frequently Asked Questions (FAQ).

2.6.1 Chatbots to Q&A Systems

The aim of the question answer system is to provide answers to user questions. This means that simple answers are preferred rather than entire documents or multiple links that must be browsed by the user (Quarteroni and Manandhar, 2007).

Nowadays it is not possible to build a chatbot that has answers to any given question. It depends on the size of the chatbot knowledge base. The larger the knowledge base, the more answers can be provided by the chatbot.
Retrieval based bots to Q&A systems are suitable for closed domains. The

knowledge base is most of all a structured one and programmed in advance, such as a simple database containing question-answer pairs. So these chatbots are able to answer questions regarding one specific topic. Their knowledge base is finite and limited, therefore they cannot provide answers to questions that do not exist in their knowledge base.
On the other side, these chatbots are very precise and provide answers with high accuracy. The generative models of chatbots are suitable for open domains and are considered more efficient than retrieval based chatbots. The answers are generated from the knowledge base. The conversation last longer and the chatbots try to gather additional information regarding the question in order to provide a satisfactory answer. ALICE, mentioned in section 2.2, is an example of such a chatbot that was trained and used for access to QA. (Shawar, 2011)

2.6.2 Chatbots to FAQ

The chatbots in this category provide a different access to frequently asked questions. These kinds of chatbots are mostly used in customer support. In this way, long waiting times on phones can be avoided. Every FAQ website can be used as a knowledge base. In addition to the functionality of information search, the chatbots also have the capability to manage the knowledge base. For example, if an unknown question has been asked more than once, it will be added to the knowledge base.
The chatbot that is used to answer University related questions, is one

example of chatbots to FAQ (Ranoliya, Raghuwanshi, and Singh, 2017).

3 Search Chatbot for Austria Forum

The chapter lists requirements for the practical part. Furthermore, architecture, design decisions and implementation will be described. It also includes the evaluation of the used tools.

3.1 Task and Requirements

The task in this thesis was to develop a chatbot that will mainly be used as an alternative to the search engine integrated in online encyclopedia. Of course, the information search will not be the only purpose of the chatbot; further potential has to be examined and implemented if possible. The chatbot can be seen as an information retrieval tool that on one hand retrieves and provides information to the users and on the other hand gathers additional information and content from them. Following requirements were set at the beginning of this thesis:

- the chatbot should be a stand-alone application
- the chatbot should be developed with Java 8 technologies
- the chatbot should be used for information search
- the chatbot should use the geography category of the online encyclopedia as an information source

3.2 Architecture

Since it was one of the requirements, the chatbot is a stand-alone application. The architecture can be divided into three main parts: a single page client (3.3.2), a chatbot system and an interface to third party NLU platform (3.4.5). Figure 3.1 illustrates the chatbot architecture. Basically, the flow

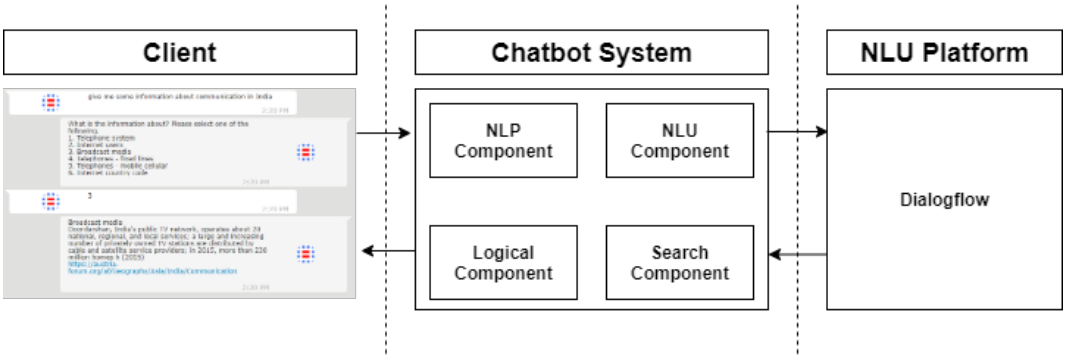

Figure 3.1: Architecture

of information begins when the user sends a message and ends with the response of the chatbot. The steps between these two events are parts of the general pipeline (Nimavat and Champaneria, 2017). Figure 3.2 shows the

levels of the general pipeline. The chatbot system developed in this work follows a variant of the general architecture and includes four components: logical component (3.4.2), natural language processing component (3.4.3), natural language understanding component (3.4.5) and full-text search component (3.4.7). Invoking the chatbot in a browser leads the user to the

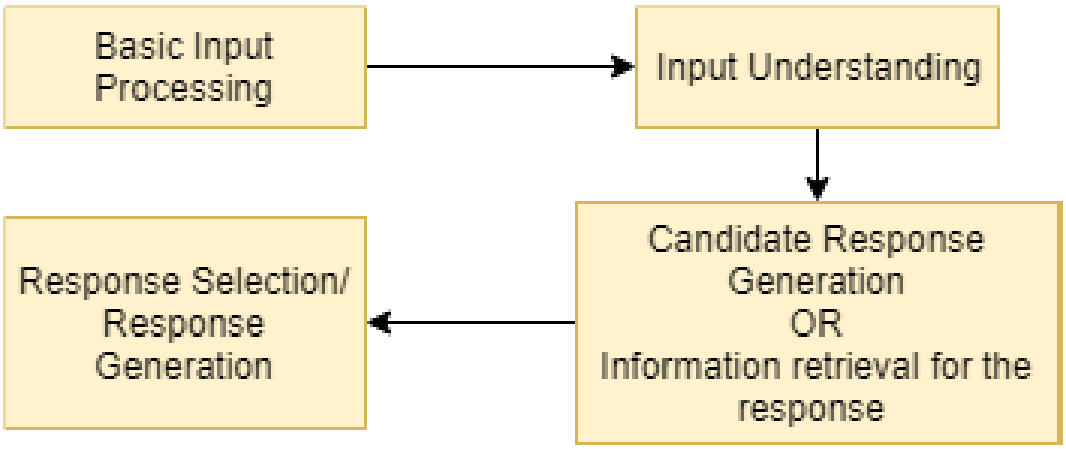

Figure 3.2: General chatbot pipeline (Nimavat and Champaneria, 2017)

single page client. The connection between client and background system is realized with the help of web sockets that are well suited for bidirectional communication. The communication begins when the user enters a question. The input is forwarded to the chatbot system that runs in the background. In order to gain additional information from the input, the NLP component is used. The NLP is used for keyword and location tag extraction. Additional natural language processing is performed on a third party NLU platform where intent and entity recognition is carried out. The logical component interprets the response from the NLU platform, sets the conversation's

context and performs certain actions. Depending on the intent matched with the user input the chatbot either searches for information in the geography knowledge base or supports the user by uploading the content. To clarify how the chatbot behaves depending on matched intents the following two use cases are assumed:

Use case 1. (3.3) The user wants to find out the capital city of a country and enters a question ("What is the capital city of Austria?") into the input field. The chatbot system performs NLP on the user input. The NLP component extracts keywords ("capital, city") and location tags ("Austria"). A request with the user input is sent to the NLU platform. The agent on the NLU platform performs intent recognition and sends a response ("search"). The bot logic component supplies the search component with information from the NLU. The search component tries to construct a query with the given data. The query is executed on the knowledge base and answers are retrieved. In case of a large quantity of search results, the chatbot will try to ask additional questions and to update the search query. This process is repeated until a certain amount of search results is retrieved. Finally, the search results are displayed to the user. **Use case 2.** (3.4) The user wants to upload information to the geography site and enters a question ("I would like to upload a video?"). The chatbot system performs NLP on the user input. The NLP component extracts keywords ("upload, video"). A request with the user input is sent to the NLU platform. It responds with matched intent ("clip upload intent"). The response is parsed, the intents and data processed. The chatbot asks a finite number of questions and collects the answers from the user. The necessary information for the video upload

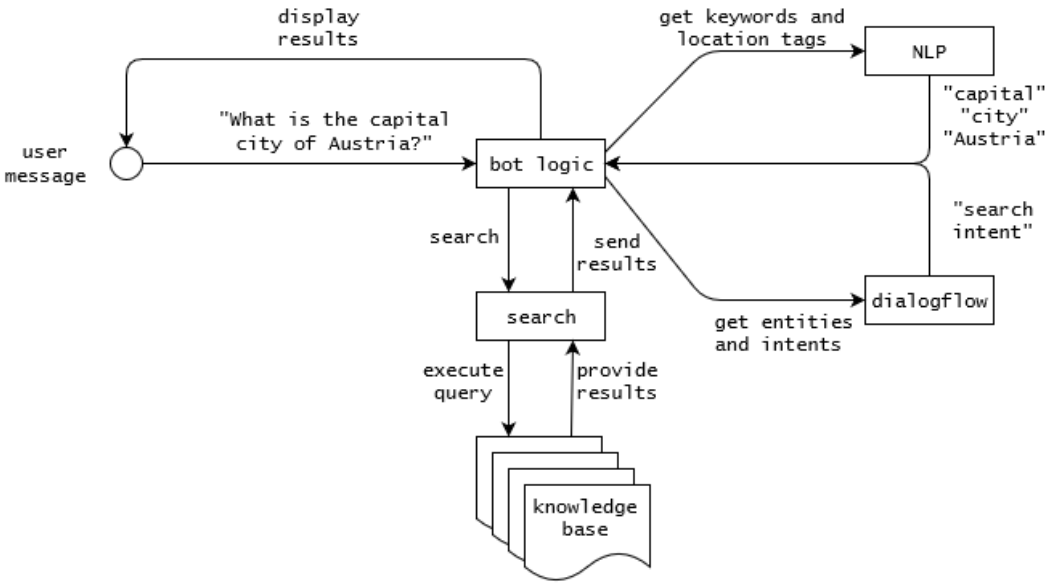

Figure 3.3: Information search use case

include the country name, the link, the title and the duration of the video. The collecting of the necessary information ends by asking the user for submission. If the user decides to upload the information, the chatbot verifies it. If the verification results in success, the gathered information is uploaded. Otherwise, the process is aborted and the chatbot expects the next question.

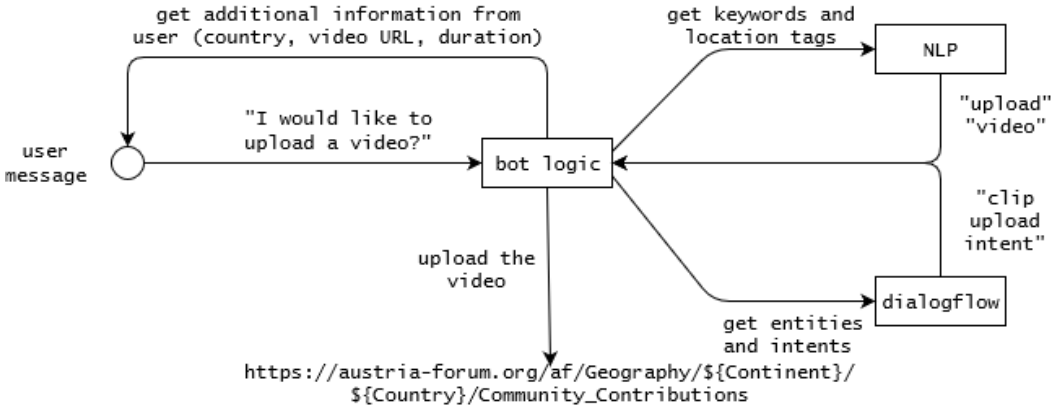

Figure 3.4: Information upload use case

3.3 Design

3.3.1 Knowledge Base

Since the chatbot is mainly used for information search, the knowledge base represents an essential part. The chatbot developed in this work uses the geography website of the Austria Forum[1] as an information source. To clarify how the content of the pages is structured the following list has to be considered:

- the index page lists continents and countries in form of links
- the country page contains a general information in form of text and further links to category pages
- the category page contains information in different forms

[1] https://austria-forum.org/af/Geography/Index (last visited on 02 June 2019)

- table with various number of rows and columns
- pictures
- text
- links to extern websites
- links to intern pages
- the mixture of all the above listed forms

Austria Forum[2] provides API that enables the retrieving of information. The information retrieved is represented in JSON format. An example of an unstructured JSON object can be seen in 3.1.

```
Geography/Asia/Lebanon": {
  "path":"Geography/Asia/Lebanon",
  "paths":[
    "Geography/Asia/Lebanon/Communication",
    "Geography/Asia/Lebanon/Community_Contributions",
    "Geography/Asia/Lebanon/Culture",
    "Geography/Asia/Lebanon/Economy ",
    "Geography/Asia/Lebanon/Energy",
    "Geography/Asia/Lebanon/Geography",
    "Geography/Asia/Lebanon/Government",
    "Geography/Asia/Lebanon/Maps",
    "Geography/Asia/Lebanon/People_Society",
    "Geography/Asia/Lebanon/Pictures",
    "Geography/Asia/Lebanon/Special_Information",
    "Geography/Asia/Lebanon/Transportation"
```

[2] https://austria-forum.org/ (last visited on 02 June 2019)

```
  ],
  "title":"Lebanon",
  "url":"https://austria-forum.org/af/Geography/Asia/
     Lebanon"
}
```

Listing 3.1: Unstructured JSON object

Since the JSON is considered as a semi structured data format, it is difficult to analyze, query or edit the data. In contrast to structured data, for example to relational database where all information is easily stored in tables with rows and columns, JSON format includes objects that differ in number of properties, structure depth and nested fields. In order to facilitate querying, editing and maintenance of the JSON objects, the mapping to a structured data format has to be done. An auxiliary tool was developed which is used for converting the unstructured JSON objects to structured JSON objects. A structured JSON object includes a finite number of key-value pairs. Multiple examples of structured JSON objects can be seen in 3.2 In order to keep the objects as simple as possible the key and the value are represented as string data types. This decision was made because of the used full-text search library (see 3.4.7).

```
{
"continent": "Asia",
"content_title": "Telephones - mobile cellular",
"country": "Lebanon",
"content_text": "total : 4.4 million ; subscriptions
```

```
per 100 inhabitants : 71 ( July 2015 est .)",
"category": "Communication",
"url":"https://austria-forum org/af/Geography/Asia/
   Lebanon/Communication"
},
{
"continent": "Asia",
"content_title": "Internet country code",
"country": "Lebanon",
"content_text": ".lb",
"category": "Communication",
"url": "https://austria-forum.org/af/Geography/Asia/
   Lebanon/Communication"
},
{
"continent": "Asia",
"content_title": "Natural gas - proved reserves",
"country": "Lebanon",
"content_text": "0 cu m (1 January 2014 es)",
"category": "Energy",
"url": "https://austria-forum.org/af/Geography/Asia/
   Lebanon/Energy"
},
{
"continent": "Asia",
"content_title": "Electricity - exports",
```

```
"country": "Lebanon",
"content_text": "0 kWh (2013 est .)",
"category":  Energy",
"url": "https://austria-forum.org/af/Geography/Asia/
   Lebanon/Energy"
},
{
"continent": "Asia",
"content_title": "Natural gas - consumption",
"country": "Lebanon",
"content_text": "150.1 million cu m (2010 est .)",
"category": "Energy",
"url": "https://austria-forum.org/af/Geography/Asia/
   Lebanon/Energy"
}
```


Listing 3.2: Structured JSON object

3.3.2 Single Client Page

Since the chatbot will be mainly used for information search, small attention was paid to the design of the client page. The idea was to develop a client page which is as simple as possible. The page consists of the following parts:

- input text field

- send button and
- chat history field

The input text field is located at the bottom of the page. The input text is not limited to a particular number of characters. The message is sent by clicking the send button or pressing the enter key. The history field stores the messages exchanged between user and chatbot. Figure 3.5 illustrates the single client page. The most important thing is that the user does not need additional knowledge in order to communicate with the chatbot. If the user is unsure what the chatbot is able to do or what its task is, it would be enough to send "help" message. The interaction is simple, natural and intuitive.

3.4 Implementation

In this section the implementation of the practical part is presented. The technologies used for the implementation and the components of the chatbot system are briefly described. It also gives an overview of the evaluation that was performed on NLP and search libraries, and as well on NLU platforms.

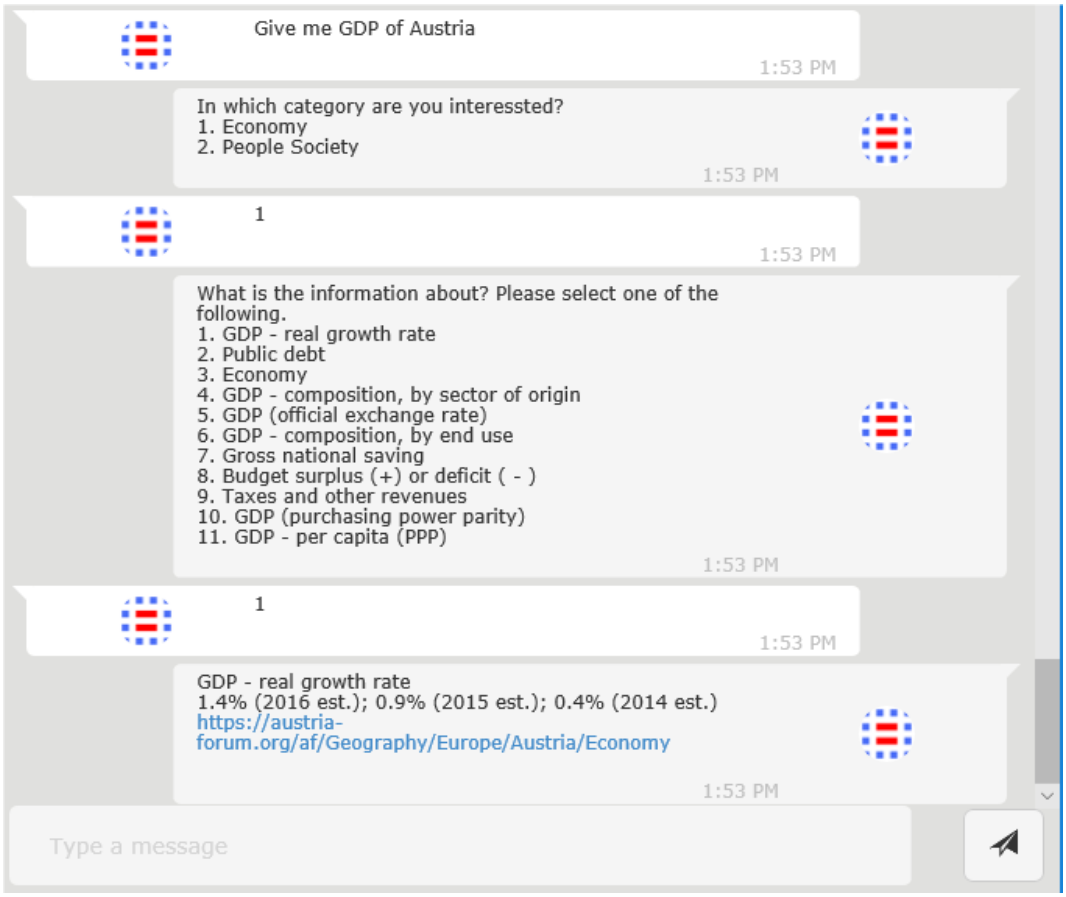

Figure 3.5: Single page client

3.4.1 Bot Server

The bot server component is based on a Web Socket server that runs on Apache Tomcat[3]. It is developed using Java API for Web Sockets. It is known that the Web Socket protocol is suitable for communication between the server and the client. Furthermore, the Web Socket protocol provides real time and bi-directional communication.

Also there is no need for the client to send requests every time it communicates with the server. It is sufficient to open the connection to the server and then it uses it every time to send messages. In order to handle the connection and communication, the server must be annotated with @ServerEnpoint annotation. This annotation defines the server as a WebSocket endpoint. Every method that the bot server provides must also be annotated.

- @onOpen is used to annotate the method that is invoked if the connection between the server and the client is established successfully
- @onClose is used to annotate the method that is invoked if the connection between the server and the client is closed
- @onMessage is used to annotate the method that is invoked when the data from the client is received
- @onError is used to annotate the method that is invoked when an error occurrs

[3] http://tomcat.apache.org/ (last visited on 02 June 2019)

3.4.2 Logical Component

As the name already suggests, the bot logic component includes all the logic of the chatbot system. This component concerns with the initialization of other components and their interaction. Furthermore, it handles the conversation context and manages the conversation flow.

When the connection between the server and the client is established, the bot logic component gets initialized. The user input is forwarded to the NLP component and a request is sent to the NLU platform. This way, additional data is collected that enables decision making and determines the conversation flow. Based on the responses from the NLP component and the NLU platform, the bot logic component executes certain actions. A default answer is sent to the user if its input had signs of a small talk conversation. If the user asks for help, the help text is displayed. In this case, the user would like to upload a content and therefore he will be asked to input additional information. The information gets processed, verified and in case of a successful verification, uploaded to the geography site. The most important task of the bot logic component occurs when the NLU recognizes a search intent. The necessary data is forwarded to the search component, the search results retrieved and displayed to the user. If the search component delivers a large number of results, additional questions have to be asked in order to gain further information and make the results more relevant to the user.

3.4.3 NLP Component

The task of the natural language processing component is to analyze the user input and make it understandable for the chatbot by detecting the meaning of the content. The component is based on Stanford CoreNLP library[4] and uses its functionality for keyword and location extraction. The user input is passed through the pipeline of CoreNLP tools that perform different actions like tokenizing, sentence splitting, parts of speech (POS) tagging, parsing, named entity recognition (NER) and lemmatization. In 3.3 can be seen how the pipeline is programmatically applied in JAVA.

```
Properties props = new Properties();
props.put("annotators", "tokenize, ssplit, pos, lemma,
   ner, parse");
pipeline = new StanfordCoreNLP(props);
```

Listing 3.3: Pipeline annotators

The language model used by the component is the Stanford English Model. Since the part of speech tagger and the named entity recognizer are used for the keyword and location extraction, figures 3.6 and 3.7 show how those tools process a text input.

[4] https://stanfordnlp.github.io/CoreNLP/ (last visited on 15 August 2019)

Figure 3.6: CoreNLP POS tagger tool

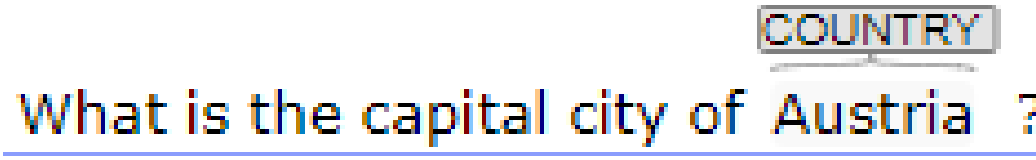

Figure 3.7: CoreNLP NER tool

Stanford CoreNLP vs. Apache OpenNLP

Stanford CoreNLP[5] and Apache OpenNLP[6] libraries are developed in Java. Table 3.1 illustrates the list of NLP tools provided by the libraries. When considering table 3.1 it can be seen that the libraries provide almost the same tools for natural language processing. The evaluation of NLP libraries is based on keyword and location entity extraction. For the keyword extraction both libraries require tokenizer and part of speech tagger tool. OpenNLP uses different models for tokenizing and POS tagging, whereby CoreNLP needs one model for all its tools. The Stanford NLP is faster and more accurate than Apache OpenNLP when considering the POS tagger tool

[5] https://stanfordnlp.github.io/CoreNLP/ (last visited on 15 August 2019)
[6] https://opennlp.apache.org/ (last visited on 15 August 2019)

NLP tool	Stanford NLP	Apache OpenNLP
Tokenization	✓	✓
Sentence segmentation	✓	✓
Part of speech taging	✓	✓
Named entity extraction	✓	✓
Chunking	✗	✓
Parsing	✓	✓
Language detection	✗	✓
Coreference resolution	✓	✓
Sentiment analysis	✓	✗
Bootstraped pattern learning	✓	✗
Open information extraction	✓	✗
Lemmatizing	✓	✓

Table 3.1: List of NLP tools

(Nanavati and Ghodasara, 2015). The comparison of POS tagger accuracy between the libraries can be seen in figure 3.8. For the location extraction the

Sr. No.	Type of Sentence	No. of Sentences tested	No. of tokens tagged	Performance (No. of tokens correctly tagged)		Accuracy (%)	
				Stanford NLP	Apache Open NLP	Stanford NLP	Apache Open NLP
1	Simple Present Tense	5	20	20	20	100	100
2	Continuous Present Tense	5	20	20	20	100	100
3	Perfect Present Tense	5	20	20	20	100	100
4	Simple Past Tense	5	20	20	20	100	100
5	Continuous Past Tense	5	20	20	20	100	100
6	Perfect Past Tense	5	20	20	20	100	100
7	Simple Future Tense	5	20	20	20	100	100
8	Continuous Future Tense	5	20	20	20	100	100
9	Perfect Future Tense	5	20	20	20	100	100
10	Ambiguous (Verb similar to Noun)	10	50	44	42	88	84
11	Use of conjunctives	5	25	24	23	96	92
12	Negative	5	25	25	25	100	100
13	Emphasis	5	25	21	20	84	80
14	Direct speech	10	50	47	45	94	90
15	Indirect speech	10	50	47	44	94	88
16	Highly Complex	20	200	171	167	86	84
Total		110	605	559	546	92	90

Figure 3.8: POS tagger comparison (Nanavati and Ghodasara, 2015)

named entity recognizer is used. Table 3.2 provides the list of entities that can be detected. At the University of Coimbra the comparison of performance of the common used NLP toolkits was made. The comparison was performed based on precision, recall, their combination and other common metrics. There were seven toolkits, whereby three of them were developed in Python and the rest in Java. The evaluated tools were tokenization, POS tagging, chunking and name entity recognition. The tools were applied on texts from social networks and texts from newspapers. The researchers showed that

Entity	Stanford NLP NER	Apache OpenNLP NER
Person	✓	✓
Location	✓	✓
Organization	✓	✓
Misc	✓	✗
Money	✓	✗
Number	✓	✗
Ordinal	✓	✗
Percent	✓	✗
Date	✓	✓
Time	✓	✓
Duration	✓	✗
Set	✓	✗

Table 3.2: List of NER entities

none of the toolkits has outperformed others, despite the differences in programming languages, used tools and text. (Pinto, Gonçalo Oliveira, and Alves, 2018)

The results of the performance comparison, POS tagger comparison and the number of entities that can be detected by NER were the main reasons for choosing Stanford CoreNLP over Apache OpenNLP.

3.4.4 NLU Platforms

The natural language platforms enrich chatbots with the Natural Language Understanding (NLU).

Dialogflow

Dialogflow[7] is an NLU platform that is owned by Google[8]. It was formerly released under the name api.ai. The NLU platform can be used free of charge. It offers a simple user interface to design so called agents. The main concepts are entities and intents. An entity can be a word or phrase in input text. They represent an important part (e.g. location, date, city, state) of a user input. An intent is a connection between the user input and the possible output. In chatbot context, an intent is seen as a mapping between a question and possible answers. Dialogflow supports multiple

[7] https://dialogflow.com (last visited on 14 July 2019)
[8] https://www.google.com (last visited on 14 July 2019)

natural languages and includes pre-built agents. There is the possibility to train agents since the platform provides machine learning features. A more detailed description can be found in section 3.4.5.

LUIS

Language Understanding Intelligent Service[9] (LUIS) is Microsoft's NLU Platform and is included in the Microsoft Azure[10] cloud services. LUIS also uses entity and intention concepts to extract meaningful information from user input. It provides pre-built entity dictionaries and can easily be integrated into the Azure bot services[11]. Learning can be enabled to acquire new knowledge, update and extend language models. LUIS supports 12 natural languages. It is intended for commercial use.

Watson Conversation

Watson Assistant[12] is an IBM[13] platform for building conversation interfaces. It is supported by NLU, which is based on intention and entity concept. It also offers a visual dialogue builder and different skills for handling questions in comparison to already described platforms. If Watson cannot

[9] https://www.luis.ai/home (last visited on 14 July 2019)
[10] https://azure.microsoft.com/de-de/ (last visited on 14 July 2019)
[11] https://azure.microsoft.com/en-us/services/bot-service/ (last visited on 14 July 2019)
[12] https://www.ibm.com/cloud/watson-assistant/ (last visited on 14 July 2019)
[13] https://www.ibm.com/at-de (last visited on 14 July 2019)

provide an answer to a user input, it tries to find an answer in different sources. Watson Assistant supports 13 natural languages. Using Watson Assistant is not free of charge.

wit.ai

Wit.ai[14] is an NLU platform that belongs to Facebook[15]. The platform enables developers to design conversational interfaces with which they can communicate via text or voice. NLU is also based on entity and intention concepts. The chatbot can be trained by providing it with conversation examples. The chatbot also learns from user input over time. Developers can define stories that determine the conversation flows and define actions which chatbot takes when it receives a message from a user. Wit.ai supports many natural languages and can be used free of charge.

Amazon Lex

Amazon Lex[16] is a service that is part of Amazon Web Services[17]. It is used to build conversation interfaces for text and voice in any application. The service provides functionalities for machine learning, converting voice messages to text and NLU. With NLU it is possible to discover intentions

[14] https://wit.ai (last visited on 14 July 2019)
[15] https://www.facebook.com (last visited on 14 July 2019)
[16] https://aws.amazon.com/lex/ (last visited on 14 July 2019)
[17] https://aws.amazon.com (last visited on 14 July 2019)

in user inputs. Amazon Lex and Amazon Alexa are supported with the same machine learning technologies. Examples of use cases are Call Center Bots, Informational Bots, Application Bots, Enterprise Productivity Bots and Internet of Things. The only language supported is American English. Amazon Lex can be used without charge for one year, with ten thousand text requests and five thousand language requests per month. After the trial year, each language request costs $0.004 and each text request $0.00075.

3.4.5 Interface to NLU Platform

The chatbot for Austria Forum takes the advantages of the third party platform Dialogflow to build great conversational experiences. It provides some concepts that help the chatbot to understand and to extract information from user's input, to control the conversation context and to learn with time.

In order to start using the NLU platform, an agent has to be created. An agent can be seen as natural language understanding module. The agent needs intents and entities to be able to understand user input and to extract useful information. Every intent refers to one or more responses. If the user input matches one of the defined intents, the agent responds with pre-defined response. With entities the agent is able to extract additional information from parts of the user input. The logical component interacts with the NLU platform. It sends the user's input, receives and processes the response. To understand what users are saying and what they are searching

for, the intents showed in figure 3.9 were defined. The following list clarifies

Figure 3.9: Chatbot intents

the meaning of the intents.

- search.country intent is matched when the user asks for information about a specific country
- search.continent intent occurs if the user wants information about a continent
- search.category intent is triggered if the user is interested in a category of a country
- search.help intent is triggered if the user asks for help
- search.choice intent is used in cases where the chatbot needs a confirmation from the user
- clip.upload intent occurs in case the user wants to upload a video or an audio clip to the website

- picture.upload intent occurs in case the user wants to upload a picture to the website

3.4.6 Search Libraries

Since the chatbot is an information retrieval tool, the choice of the search library is very important. The following two sections give an overview about search libraries considered during the research.

Apache Lucene

Developer	Apache Software Foundation
Initial Release	1999
Programming Language	Java
Operationg System	Cross-platform
Type	Index and Search
License	Apache License 2.0
Website	https://lucene.apache.org/

Table 3.3: Apache Lucene software library

Apache Lucene[18] is an open-source full-text search software. It includes Lucene Core[19], a library written in Java by Doug Cutting. It is suitable for

[18] https://lucene.apache.org/ (last visited on 15 July 2019)
[19] https://lucene.apache.org/core/index.html (last visited on 15 July 2019)

applications that incorporate information retrieval features. It offers scalable and high-performant indexing as well as ranked, fielded and multiple-index searching (The Apache Software Foundation, 1999). General information on Lucene can be found in table 3.3.

The indexing process deals with index creation and mapping of text data into index documents. The indexes enable accurate and efficient searching processes. The searching process deals with querying and retrieving of the indexed documents. Lucene provides a wide range of query types like phrase, boolean, wildcard, proximity, range query and more. The documents are retrieved based on ranking. (Balipa and Ramasamy, 2015) The working process of a Lucene application is shown in figure 3.10.

Apache Solr

Developer	Apache Software Foundation
Initial Release	2006
Programming Language	Java
Operationg System	Cross-platform
Type	Search and Index API
License	Apache License 2.0
Website	http://lucene.apache.org/solr/

Table 3.4: Apache Solr enterprise search platform

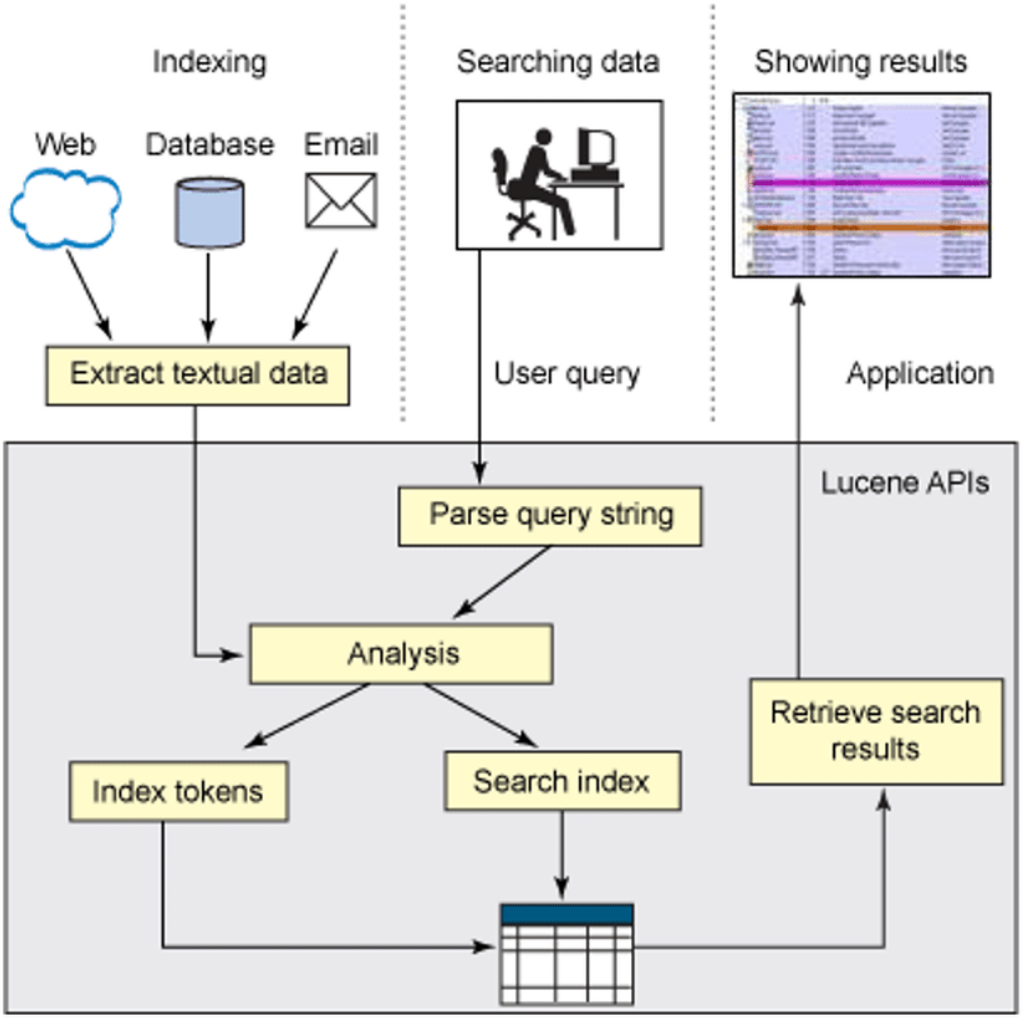

Figure 3.10: Workflow of Lucene application (Balipa and Ramasamy, 2015)

Apache Solr[20] is a standalone search server. Originally, Solr was developed by Yonik Seeley at CNET Networks[21]. In 2006 it became a part of the Apache Software Foundation[22]. It is based on top of Lucene Core[23] and provides APIs for XML, HTTP, JSON, Python and Ruby. Its major features are hit highlighting, faceted search, caching, replication and a web administrator interface. The Solr search server is accessed over a REST-like API. The documents are uploaded over HTTP in form of JSON, XML, CSV or binary. The querying is based on HTTP GET request and results are retrieved in JSON, XML, CSV or binary form. (Vikash and Barwal, 2016) General information on Solr can be found in table 3.4.

3.4.7 Search Component

The search component is based on full-text search library Lucene Core[24] and uses its functionality for indexing and searching. Therefore the search component includes an indexer and a searcher. The Lucene indexer is concerned with index creation. The Lucene index consists of documents that include one or more fields. The index of the chatbot stores all the data that is retrieved from the geography part of Austria Forum. Each document in Lucene index holds a structured JSON object. The key-value strings of

[20] https://lucene.apache.org/solr/ (last visited on 15 July 2019)
[21] https://www.cnet.com/ (last visited on 15 July 2019)
[22] https://www.apache.org (last visited on 15 July 2019)
[23] https://lucene.apache.org/core/index.html (last visited on 15 July 2019)
[24] https://lucene.apache.org/core/index.html (last visited on 15 July 2019)

the object are mapped to the document fields.The mapping of JSON objects is shown in 3.4. For the simplicity of the search process each field was represented as a text field.

```
private void addDocuments() throws IOException {
  for(JSONObject o : (List <JSONObject>) this.data ) {
    Document doc = new Document();
    String continent = o.get(Constants.CONTINENT ).
       toString();
    String country = o.get(Constants.COUNTRY).toString();
    String category = o.get(Constants.CATEGORY ).toString
       ();
    String title = o.get (Constants.TITLE).toString();
    String text = o.get(Constants.TEXT).toString();
    String url = o.get(Constants.URL).toString();

    doc.add(new TextField(Constants.CONTINENT, continent
       ,Field.Store.YES));
    doc.add(new TextField(Constants.COUNTRY, country,
       Field.Store.YES));
    doc.add(new TextField(Constants.CATEGORY, category,
       Field.Store.YES));
    doc.add(new TextField(Constants.TITLE, title, Field.
       Store.YES));
    doc.add(new TextField(Constants.TEXT, text, Field.
       Store.YES));
    doc.add(new TextField(Constants.URL, url, Field.Store
```

```
        .YES));

    this.writer.addDocument(doc);
  }
}
```

Listing 3.4: Mapping of JSON objects to index documents

The searcher provides capabilities for searching and querying. The searcher creates a query and passes it on to the index searcher object, which executes the query and returns the list of the relevant documents. Furthermore, it provides methods for query creation and update. It is possible to create queries that are executed on multiple fields as well the queries that are run on a specific document field. An example of the query that is executed on country field is shown in 3.5.

```
public void createLocationQuery (String location) {
  BooleanQuery.Builder builder = new BooleanQuery.Builder
     ();
  Query q = new TermQuery(new Term(Constants.COUNTRY ,
     location.toLowerCase ()));
  builder.add(q, BooleanClause.Occur.MUST);
  setQuery(builder.build());
}
```

Listing 3.5: Creation of a location query

The queries are created with the parameters that are made available by

the NLP and the NLU components. The methods regarding the search are responsible for query execution and retrieval of the relevant results. The query is executed by the search function that is provided by the index searcher. The results are presented in form of documents and are listed based on the document score. In order to avoid large numbers of documents and to provide only relevant results to the users, a threshold had to be defined. Therefore the documents with a higher score than the threshold are considered relevant. If the number of the retrieved documents is still large, the user is asked for additional information; the query is updated and executed again. The search component also takes care about the length of the retrieved results. The length of the result is limited to three hundred characters. For more detailed information the user can click on the link included in the search result.

The reason why Lucene was chosen over Solr is that Lucene is more suitable for building prototypes. Also, the existing embedded search engine on Austria Forum is based on Lucene Core. Lucene gives the programmer full control over all internal processes. Solr is a standalone search server that is used in content management and enterprise management systems.

4 Comparison of Search Results

In this chapter the search results of the integrated search engine on the website Austria Forum and those of chatbots are compared. In order to compare the search results, six questions were defined, shown in table 4.1. The questions contain information that is included in the knowledge base.

The search engine of Austria Forum is based on keyword matching algorithm. It functions like most search engines and provides a list of links as search results for any given query. It does not matter how the query was defined and what information it contains. The search results of the search engine for the defined questions (4.1) are shown in 4.1, 4.3, 4.5, 4.7, 4.10 and 4.12.

The chatbot, on the other hand, accepts queries in natural language as well as keywords. For the purpose of this comparison the number of answers to be displayed to the user was set to three. Since the knowledge domain is a closed domain containing geographical information, the main focus was on location information. When the chatbot receives a message, it searches for

Question	Austria Forum	Chatbot
Q1	Nepal	Provide some information about Nepal
Q2	communication India	Give me some information about communication in India
Q3	airports Croatia	How many airports does Croatia have?
Q4	population	Need information about population
Q5	energy	I need information about energy
Q6	natural hazards	What are the most common natural hazards

Table 4.1: Questions

the location tag, constructs a query and performs search on the knowledge base. In case of the Q3, the location tag is found and the search results retrieved. The number of results is within defined range and they are displayed to the user as can be seen in 4.6. The questions Q1 and Q2 also contain location information. The amount of the found results is bigger than the defined threshold. The chatbot engages the user with followup and clarification questions in order to gather additional information. The search query is updated and the search performed again. This procedure is repeated until the number of results is lower than the threshold. At the end the results are displayed to the user. The conversation flow and the results for Q1 and Q2 can be seen in figures 4.2 and 4.4.

The questions Q4, Q5 and Q6 does not contain any location information. The chatbot constructs query and searchs for information. If the number of results is too large the chatbot asks for location information (country name). The query is updated and the results retrieved. If the entered country does not exist, the chatbot would not find anything and a pre-defined answer is displayed. Otherwise, the results are retrieved. Depending on the number of the results, the chatbot either poses clarification questions or displays the results to the user. The conversation flow and the results for questions Q4, Q5 and Q6 are shown in 4.8, 4.9, 4.11 and 4.13.

The chatbot does not provide results for every search query. If it receives a question that does not contain any information related to the geography domain, it would answer with one of the pre-defined responses. In this case the search results would be empty.

The chatbot provides also information about its tasks and usage. This information is provided if a user enters “Can you help?”, “I need help” or similar types of questions. The answers are pre-defined.

The chatbot and the search engine of Austria Forum differ in the way they display search results and the form of those results. In contrast to the search engine of Austria Forum that displays a list of links as search results, the chatbot displays search results in two different forms. The chatbot answers either contain a chunk of content and a link to the page containing it or include only the link to the page containing relevant information. The answers are limited to three hundred characters.

Considering the search results of both the chatbot and the search engine of Austria Forum it can be concluded that the chatbot provides more satisfactory results. The design and the structure of the knowledge base mentioned in 3.3.1 have a big effect on the relevance of search results. Also, the capability of the chatbot to engage users with clarification questions contributes to the higher relevance of search results.

Seite	Relevanz
Nepal (Geography > Asia > Nepal > Maps)	100
Nepal (Geography > Asia)	87
Nepal in South Asia (Geography > Asia > Nepal > Maps)	59
Impressions of Nepal (Geography > Asia > Nepal > Pictures)	57
Faces of Nepal (Geography > Asia > Nepal > Pictures)	57
Schwarzenegger in Nepal (Geography > Asia > Nepal > Special Information)	50

Figure 4.1: Search engine results for Q1 (see table 4.1)

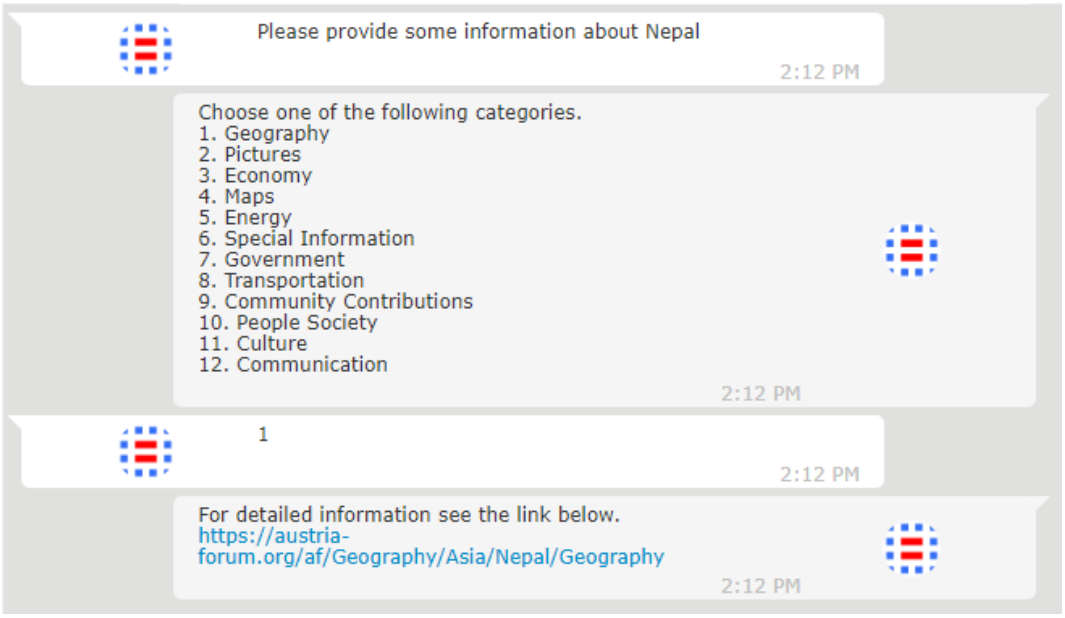

Figure 4.2: Chatbot search results for Q1 (see table 4.1)

Page	Score
India (Geography > Asia > India > Maps)	100
India (Geography > Asia)	73
Communication (Geography > Asia > Nepal)	69
Communication (Geography > Territories, Oceans and Islands > Southern Ocean)	69
Communication (Geography > Territories, Oceans and Islands > Navassa Island)	69

Figure 4.3: Search engine results for Q2 (see table 4.1)

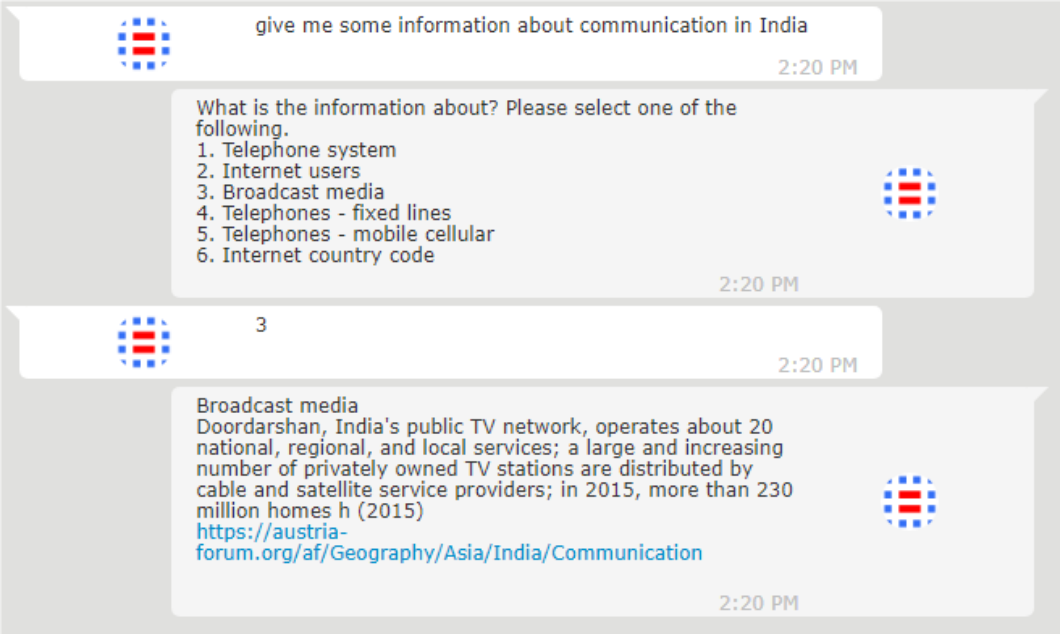

Figure 4.4: Chatbot search results for Q2 (see table 4.1)

Page	Score
Croatia (Geography > Europe > Croatia > Maps)	100
Airport (Geography > Asia > Nepal > Pictures > Kathmandu)	76
Croatia (Geography > Europe)	62
Adelaide-Airport (Geography > Australia > Australia > Pictures > Various Pictures from Australia)	58
Croatia in Europe (Geography > Europe > Croatia > Maps)	58

Figure 4.5: Search engine results for Q3 (see table 4.1)

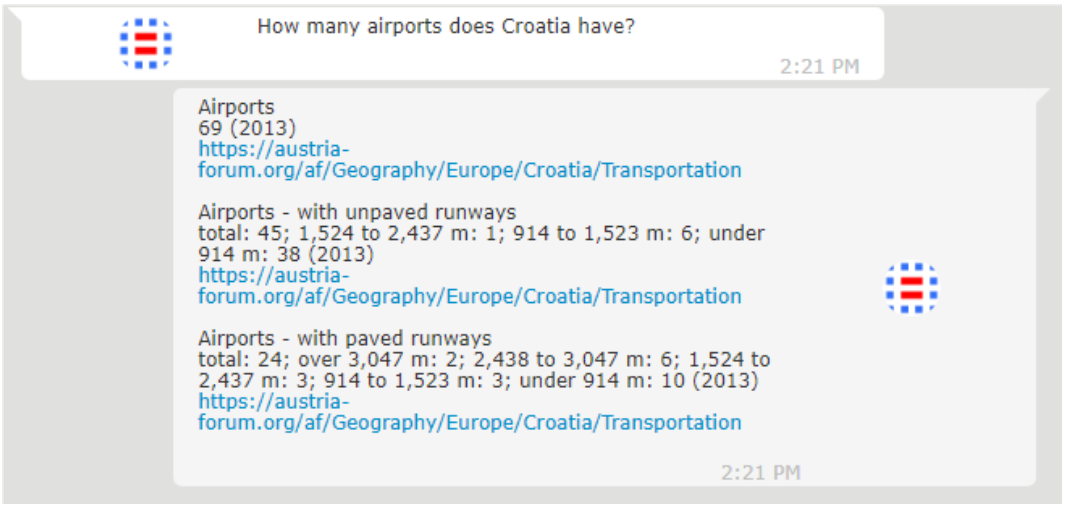

Figure 4.6: Chatbot search results for Q3 (see table 4.1)

Seite	Relevanz
Population of Austria (Geography > Visualizations)	100
Population statistics of the world (Geography > Visualizations)	97
Population (Geography > Asia > Philippines > Pictures > Philippinen)	96
People Society (Geography > Territories, Oceans and Islands > Bouvet Island)	13
People Society (Geography > Territories, Oceans and Islands > Clipperton Island)	13

Figure 4.7: Search engine results for Q4 (see table 4.1)

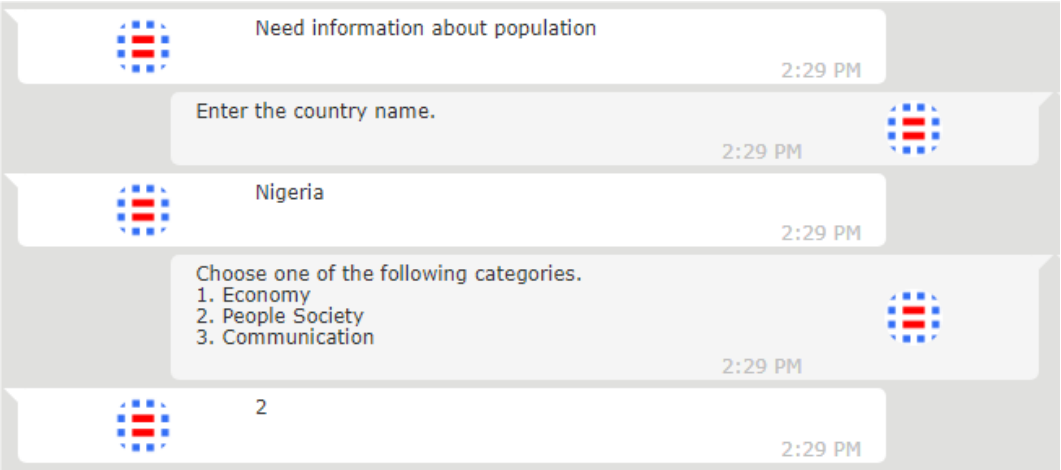

Figure 4.8: Chatbot search results for Q4 (see table 4.1)

The functionality of the chatbot to support users when uploading information to the website should also be mentioned. Each country page on Austria forum include a link to the page "Community Contributions". This page provides forms for upload of interesting pictures and audio or video clips. If the user asks "I would like to upload a picture", "I would like to upload an audio clip" or similar questions, the chatbot reacts and engages the user with followup questions. It collects information like country name, title of picture or clip, link to information and duration of clips. The chatbot verifies the information and in case of success, asks the user for confirmation.

In order to upload information via "Community Contributions" the user should know which pages does provide upload possibility. To find the page for upload is time consuming. With the chatbot this scenario is straightforward. The only task of the user is to ask right questions.

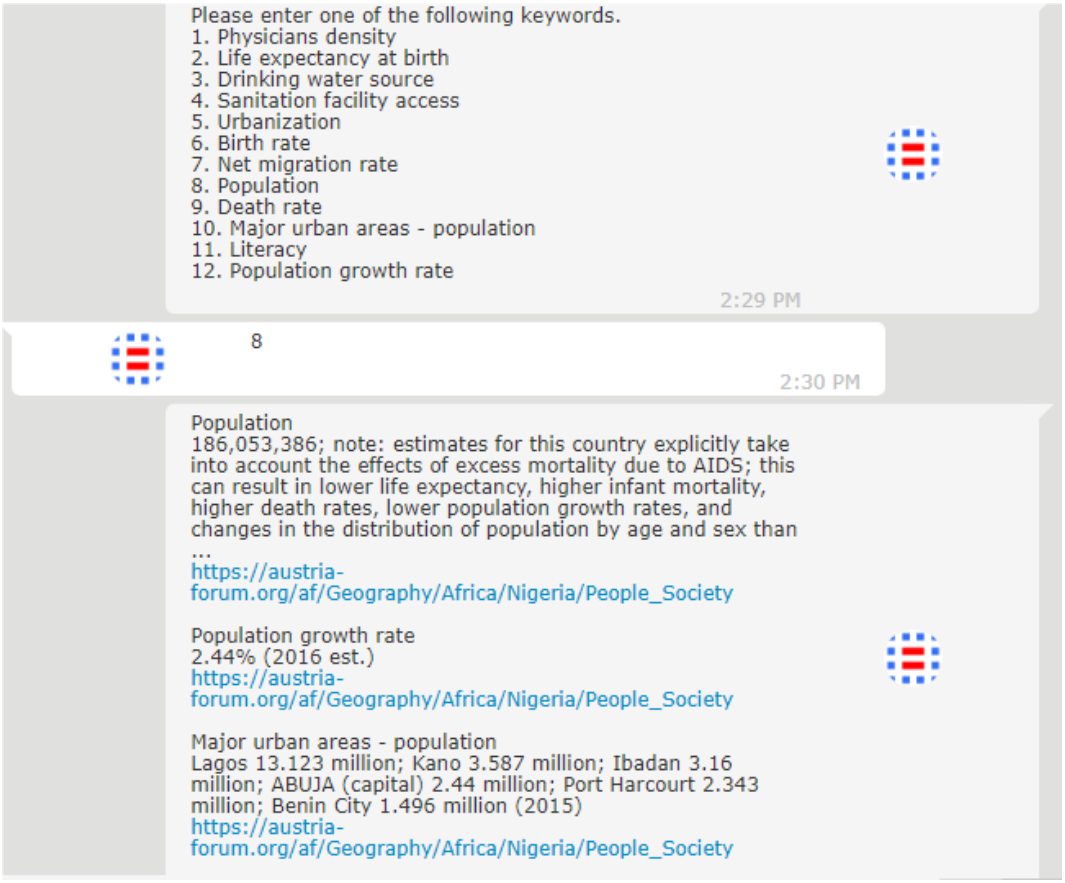

Figure 4.9: Search engine results for Q4 (2) (see table 4.1)

Seite	Relevanz
Energy (Geography > Territories, Oceans and Islands > Guernsey)	100
Energy (Geography > Australia > Palau)	100
Energy (Geography > Territories, Oceans and Islands > Antarctica)	100
Energy (Geography > Territories, Oceans and Islands > Akrotiri)	100
Energy (Geography > Territories, Oceans and Islands > Anguilla)	100

Figure 4.10: Search engine results for Q5 (see table 4.1)

Figure 4.11: Chatbot search results for Q5 (see table 4.1)

Seite	Relevanz
Natural Bridge Natural Bridge (Geography > America > United States > Pictures > Virginia)	100
Natural Gas Production (Geography > Visualizations)	85
Natural Bridge Canyon (Geography > America > United States > Pictures > Factbook Pictures)	85
Natural Gas Consumption (Geography > Visualizations)	85
Garachico - Piscinas Naturales (2) (Geography > Europe > Spain > Pictures > Tenerife (2016))	79

Figure 4.12: Search engine results for Q6 (see table 4.1)

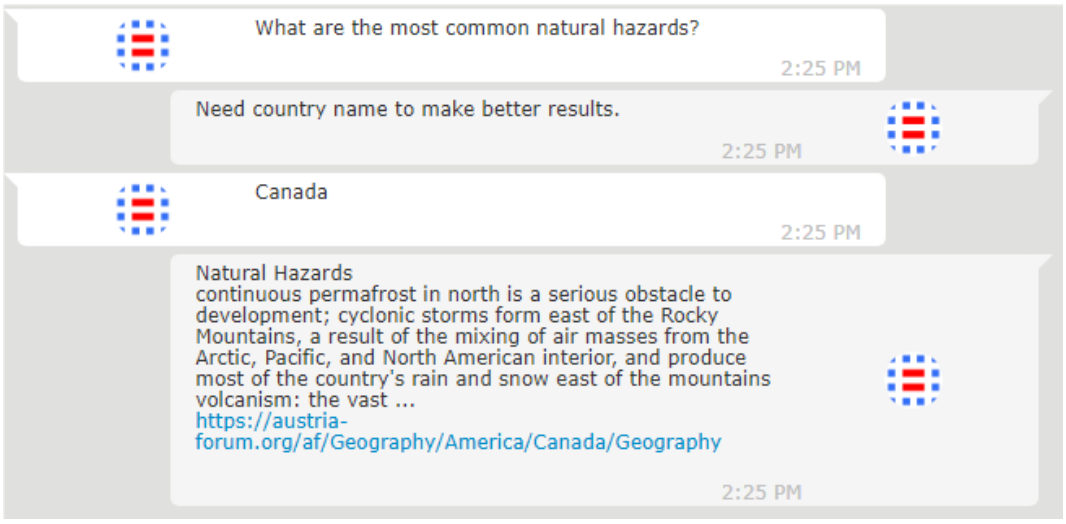

Figure 4.13: Chatbot search results for Q6 (see table 4.1)

5 Future Work

Considering the research, it can be concluded that some parts of the chatbot system could be improved. Most potential for improvement lies in the search component. Since the named entity recognizer of the NLP component is capable of detecting several entities, the search component could provide functionalities for a person, number or organization search. It would request an adaption of the knowledge base and an introduction of different field types other than text.

Since Austria Forum also provides information on other domains besides geography, it would be interesting to see how the chatbot would function on multiple domains. It is imaginable that users write and upload whole articles via chatbot and in this way contribute to online encyclopedias. Of course Austria Forum API should be extended for this to be feasible.

The ability of the chatbot to understand user inputs depends heavily on the NLU platform. Dialogflow already provides V2 API with some improvements. The NLU platforms will continue to develop and become more powerful.

6 Conclusion

The first ideas about chatbots and how they developed over time were described in Chapter 2. To better understand the potential of chatbots today, types and practical applications have been listed. The biggest focus was on chatbots that can be used as information retrieval tools.

The goal of this master thesis was to design and develop a chatbot that users can use as an information retrieval tool. The chatbot allows information retrieval in natural language but also supports keyword-based queries. Chapter 3 describes the design, architecture and implementation of the chatbot. Communication with NLU Platform, which adds some intelligence to the chatbot, has been described. The choice of tools used for search functionalities and for natural language processing was discussed.

Basic input processing was performed with the natural language processing library Stanford CoreNLP. The number of entities that can be detected by a named entity recognizer and the performance of the post of the speech tagger were crucial in the selection process of the NLP library.

A big focus was set on the choice of the search library, since the search component is an essential part of the chatbot system. The search engine of Austria Forum is based on the Lucene library and was chosen as a base for the search component of the chatbot system. This library provides a very powerful full-text search functionality. It gives the user the possibility to map the knowledge base onto so called indexes and to maintain, query and retrieve results from them.

The chatbot for Austria Forum is used as an information retrieval and contains properties of several chatbot types. It is definitely a closed domain chatbot with a geographical knowledge base. Engaged with a question the chatbot tries to find relevant information or to help the user to upload information. The chatbot does not store chat history, the conversation is short and question-answer based. Once engaged with the question the chatbot takes over the conversation and requires additional input. The procedure of requiring input by information upload is pre-defined. The NLU platform adds some intelligence to chatbot and gives capability to learn with time. It can be inferred that the chatbot is a hybrid type, including properties of rule and machine learning based chatbots.

A comparison of the results in chapter 4 showed that search engines and chatbots differ in functionality and displaying of search results. The chatbot accepts whole sentences as well as keywords as input. Additionally, the form of the input has an effect on the search results. The comparison also showed that chatbot provides satisfying results. It has potential to be used as an information retrieval tool in a closed domain.

Bibliography

Accenture Interactive (2016). *Chatbots in Customer Service*. last visited on 23 July 2019. URL: https://www.accenture.com/t00010101t000000__w__/br-pt/_acnmedia/pdf-45/accenture-chatbots-customer-service.pdf (cit. on p. 15).

Balipa, Mamatha and Balasubramani Ramasamy (2015). "Search Engine using Apache Lucene." In: *International Journal of Computer Applications* 127, pp. 27–30. DOI: 10.5120/ijca2015906476 (cit. on pp. 46, 47).

Cambridge Dictionaries Online (1999). *Cambridge Dictionaries Online*. last visited on 22 July 2019. URL: https://dictionary.cambridge.org/ (cit. on p. 5).

Chatbot Magazine and Matt Schlicht (2016). *The Complete Beginner's Guide to Chatbots*. last visited on 18 June 2019. URL: https://chatbotsmagazine.com/the-complete-beginner-s-guide-to-chatbots-8280b7b906ca (cit. on p. 9).

Computer History Museum (1996). *Internet History of 1970s*. last visited on 22 July 2019. URL: https://www.computerhistory.org/internethistory/1970s/ (cit. on p. 7).

Croft, W., Donald Metzler, and Trevor Strohman (2009). *Search engines: Information retrieval in practice*. ISBN: 978-0-13-136489-9 (cit. on pp. 1, 18).

Dale, Robert (2016). "The return of the chatbots." In: *Natural Language Engineering* 22, pp. 811–817. DOI: 10.1017/S1351324916000243 (cit. on pp. 8, 10).

dictionary.com (1995). *dictionary.com*. last visited on 22 July 2019. URL: http://dictionary.com/ (cit. on p. 5).

Nanavati, Jay and Yogesh Ghodasara (2015). *A Comparative Study of Stanford NLP and Apache Open NLP in the view of POS Tagging*. last visited on 25 July 2019. URL: http://www.ijsce.org/wp-content/uploads/papers/v5i5/E2744115515.pdf (cit. on p. 38).

Nimavat, Ketakee and Tushar Champaneria (2017). "Chatbots: An overview. Types, Architecture, Tools and Future Possibilities." In: (cit. on pp. 11, 22, 23).

Pinto, Alexandre, Hugo Gonçalo Oliveira, and Ana Alves (2018). "Comparing the Performance of Different NLP Toolkits in Formal and Social Media Text." In: 51, 3:1–. DOI: 10.4230/OASIcs.SLATE.2016.3 (cit. on p. 40).

Quarteroni, Silvia and Suresh Manandhar (2007). "A Chatbot-based Interactive Question Answering System." In: (cit. on p. 18).

Ranoliya, Bhavika, Nidhi Raghuwanshi, and Sanjay Singh (2017). "Chatbot for university related FAQs." In: pp. 1525–1530. DOI: 10.1109/ICACCI.2017.8126057 (cit. on p. 20).

Shawar, Bayan (2011). "A Chatbot as a Natural Web Interface to Arabic Web QA." In: *International Journal of Emerging Technologies in Learning* 6. DOI: 10.3991/ijet.v6i1.1502 (cit. on pp. 13, 19).

Shawar, Bayan and Eric Atwell (2007). "Chatbots: Are they Really Useful?" In: *LDV Forum* 22, pp. 29–49 (cit. on pp. 6–8, 10).

Shawar, Bayan, Eric Atwell, and Andrew Roberts (2005). "FAQchat as an Information Retrieval System." In: (cit. on p. 14).

The Apache Software Foundation (1999). *Lucene Core*. last visited on 25 July 2019. URL: https://lucene.apache.org/core/ (cit. on p. 46).

Turing, A. M. (1950). "I.—COMPUTING MACHINERY AND INTELLIGENCE." In: *Mind* LIX.236, pp. 433–460. ISSN: 0026-4423. DOI: 10.1093/mind/LIX.236.433. eprint: http://oup.prod.sis.lan/mind/article-pdf/LIX/236/433/9866119/433.pdf. URL: https://doi.org/10.1093/mind/LIX.236.433 (cit. on p. 8).

Vikash, Kumar and P.N. Barwal (2016). *Implementation of Highly Optimized Search Engine Using Solr*. last visited on 25 July 2019. URL: http://www.ijirset.com/upload/2016/march/116_Implementation.pdf (cit. on p. 48).

Weizenbaum, Joseph (1966). "ELIZA&Mdash;a Computer Program for the Study of Natural Language Communication Between Man and Machine." In: *Commun. ACM* 9.1, pp. 36–45. ISSN: 0001-0782. DOI: 10.1145/365153.365168. URL: http://doi.acm.org/10.1145/365153.365168 (cit. on p. 7).

Wildml and Denny Britz (2016). *Deep Learning for Chatbots, Part 1 – Introduction*. last visited on 23 July 2019. URL: www.wildml.com/2016/04/deep-learning-for-chatbots-part-1-introduction/ (cit. on pp. 12–14).

Wilks, Yorick (1999). *Machine Conversations*. Norwell, MA, USA: Kluwer Academic Publishers. ISBN: 0792385446 (cit. on p. 8).

Zadrozny, Wlodek et al. (2000). "Natural Language Dialogue for Personalized Interaction." In: *Commun. ACM* 43.8, pp. 116–120. ISSN: 0001-0782. DOI: 10.1145/345124.345164. URL: http://doi.acm.org/10.1145/345124.345164 (cit. on p. 8).