The Self-Learning IBM Watson Chatbot

Stijn van Vilsteren[1][1920286], Jente Insing[2][1903128] and Zoe Youngji Cho[3][1800795]

LIACS, Leiden University, Niels Bohrweg 1, Leiden, The Netherlands
[1]c.j.van.vilsteren@umail.leidenuniv.nl, [2]j.l.insing@umail.leidenuniv.nl,
[3]y.cho.3@umail.leidenuniv.nl

Abstract. In this technical report we discuss how to make a self-learning chatbot using IBM Watson as an artificial intelligent engine. We begin by introducing IBM Watson and placing it in the context of self-learning chatbots. Self-learning means that it does not need external help when it comes to training the system. We illustrate about what the problems with chatbots are and how to solve those problems by making them smarter with offering good user experiences. We will discuss the different kind of API's and services Watson provides to make a chatbot with learning capabilities. Further, we describe how developers can get started with making an IBM Watson chatbot easily and efficiently. We conclude the paper with some final thoughts about our findings including the brief proposal of making an intelligent chatbot.

Keywords: Watson, IBM, Chatbots, Machine Learning, Cloud Computing, Web Intelligence, Big Data.

1 Introduction

Many people use chat applications more than social networks or other apps. The largest services like Whatsapp, Telegram or Facebook Messenger, have hundreds of millions of active users [1]. Because of this trend it is a great place for potential growth for companies. For example, the dealing with an enormous amount of customer questions via mail could be dealt with by a chatbot. Thanks to artificial intelligence and chat applications opening app stores for bots it becomes a lot easier to build chatbots and integrate it into existing chat applications. A lot of companies already made chatbots to serve their customers by automating their customer service with keeping a human touch and without them having to download an extra application on their phone. Facebook Messenger already has more than 34.000 bots on their platform [2]. A company that uses a chatbot is KLM, the Royal dutch airlines, which developed a Facebook Messenger bot which a user can chat with to get information about his flight, information about delays, if a gate has changed and when boarding starts. In this paper we will study the domain of chatbots: what chatbots are, what capabilities they have and how to make a chatbot using the IBM Watson services.

1.1 Chatbots

Chatbots are applications you can interact with using natural language within a chat interface and are often used as automated online assistants where you can interact with speech or via text. Artificial intelligence in chatbots is often used to understand natural language and to give a human like response to try to mimic a human conversation. There are multiple categories of chatbots, but chatbots are most used in the following two categories: [3]

Rule based chatbots which are programmed to see if a certain condition is met (keyword matching). E.g. if you text “I want to book a flight to Amsterdam” it recognizes book, flight and amsterdam as a command and the system will return a list with flights to Amsterdam. If you text “I need a flight to Amsterdam” It does not understand what the user want. Rule based chatbots are much easier to develop than AI chatbots and are mostly used for one single purpose, but also less scalable and less reliable for the user. With rule based chatbots users have to learn how to conversate with the bot, which makes the user experience cumbersome [3].

Artificial intelligence chatbots are more advanced bots than rule based chatbots and are using Natural Language Processing (NLP) and machine learning to understand questions posed in natural language. This means that they can parse a question to understand the meaning and context of it and what kind of question is being asked. E.g. it can make a difference between “I want to book a flight to Amsterdam”, or “I need a flight to Amsterdam”. AI chatbots often have the ability to learn from earlier interactions with a user and can use this data to become smarter and therefore better serve the user. Those kind of chatbots are more expensive and more time consuming to make, but are easier to scale and could easily be used for multiple purposes [3].

1.2 Problems with chatbots

Although chatbots are used a lot, they are not perfect. Most chatbots lack the understanding of context and have problems understanding a basic question when a user talks to a chatbot in a way he would talk to a person, which could be very frustrating, because they try to act as a person but lack the basic understanding of language or context and therefore fail to create a good user experience. One of the problems here is that most chatbots are not good at natural language processing. Questions posed in natural language can be posed in a lot of different ways, for example: give me the weather forecast, is it raining? Do I need an umbrella. Bots often do not understand the relation between rain and using an umbrella like a person can. This gives the feeling that you can not have a simple conversation. Another problem is the lack of understanding follow-up questions. when asking a such a question bots often do not remember the question the users posed before or the answer it self gave [4].

This problem generates another problem, which is that chatbots cannot answer everything. They are focused on a specific topic which means that for every question topic you need another chatbot.

Since that chatbots still offer a cold and poor user experience and its lack of functionality, the user should not assume that the chatbots can solve all of their problems or answer all of their questions correctly, although companies or developers often present those bots that way.

2 AI engines for chatbots

To make a chatbot, multiple services or engines are developed to power an artificial intelligent chatbot with, like Octane ai, Motion.ai, IBM Watson, Dexter, BeepBoopHQ and more. In the next section we will discuss the capabilities of IBM Watson as an AI engine for chatbots. We have chosen for IBM Watson because of the enormous amount of availabilities to experiment with it and Watson has proven in the game Jeopardy that it is good in answering difficult open questions [5]. With IBM Watson you are able to use machine learning without having programming skills, it is easy to train via a web interface and easy to implement in your own system(s).

2.1 IBM Watson

Watson, named after IBM's founder, Thomas J. Watson, is a computer system making use of Artificial Intelligence and cognitive computing technology to become a self-learning "questioning answering (QA) machine" using natural-language processing and had the capability of answering difficult questions within seconds [6]. According to Kelly and Hamm [7] Watson is the first step of the era of cognitive computing and this means that computers can learn from interaction with humans or by analyzing lots of (big) data by simulating the functions like the thought processes of the human brain. This allows Watson to learn from structured- and unstructured data it gathered and has the ability to learn the context of this data and to reason in order to give a precise answer or to make a decision by using machine learning and deep learning algorithms [8]. Furthermore, Watson makes use of the following kinds of technologies, like big data, web intelligence and cloud computing.

2.1.1 Big data, Web intelligence and cloud computing

Big data is a term that refers to an enormous amount of data sets that could be structured or unstructured. With big data there are a lot of big challenges to deal with, e.g. curation, storage infrastructure, security, visualisation and many more. Because of the size and complexity of big data, normal computers are not able to face these challenges [9]. IBM Watson has the ability to tackle these challenges with his supercomputer power and intelligent cognitive learning system. As Bob Picciano explained, Watson is able to go trough data in record speeds. 100 gigabytes per seconds, which allows businesses to integrate their own datasets with external datasets and use them to analyse complex data structures [10]. IBM Watson is an example of web intelligence that uses its A.I. and information technology to solve big questions in a short amount of time.

Web intelligence is a combination of artificial intelligence (AI) and advanced Information technology (IT). Both are part of the rapid growing internet and World Wide Web [11]. Web intelligence is applied in different sectors, like the government, in business and for personal use.

IBM Watson is accessible via different applications and web interfaces, like via mobile, desktop or other devices that are connected to the web [12]. This is called cloud computing. Cloud computing is a term that was called 'The web'. In simple terms, it has a front-end and a back-end. The front end provides the user on-demand access in many applications that are hosted on the web [13]. Examples are mail clients (Google mail, Microsoft Outlook), environments to work together via the browser of a desktop, mobile or tablet (Google Drive, Invision, Adobe cloud). The servers where all those applications run and have their storage can be seen as the back-end. Cloud computing enables users the work faster, work together, work from different devices (smartphones, desktops and other devices) and make use of the power of the servers of the companies.

2.2 DeepQA: How IBM Watson answers questions

IBM Watson makes use of a hybrid Questioning Answering system, called DeepQA. This works differently than knowledge based QA, used by for example Apple's Siri [14][15] or document search used by Google Search. With DeepQA you can ask a question in natural language and the system will analyze its database, including unstructured data like text, images, video and audio, find candidate answers, generate hypothesizes, give them a score, rank them and return a precise answer to the question [16]. In the following section we will explain how Watson answers questions using DeepQA.

Question analysis
When a user asks a question in natural language, Watson will parse the question and analyze it to understand the meaning of it (or double meanings), to detect synthetic subject-verb object properties or semantic relations to identify the different roles the words and phrases in the sentence are playing and how they are connected. In this process Watson tries to find out what type of question is being asked and what the question is asking for. This process determines how the question is further processed by the rest of the system [17].

Hypothesis generation and evaluation
After the system has gathered the results from the question analysis and understands the meaning of the question it will try to find as many of possible candidate answers by searching through its databases which could be thousands of gigabytes. This type of search is called "primary search" where each candidate answer could be seen as a hypothesis. In this phase quantity is more important than accuracy, because when the correct answer is not found in this phase of the process, Watson will not be able to answer the question [17].

Hypothesis and Evidence scoring
The candidate answers that came out of the hypothesis generation phase will be evaluated by collecting positive and negative evidence for each candidate. Scoring algorithms rate the quality of the gathered evidence by many different factors. For example, one of Watson's algorithms named temporal reasoning tries to detect inconsistencies between dates in the question and dates in the candidate answer. When dates are not matching Watson will give this

answer a lower score . Based on those factors it will create a score about how much the evidence supports the candidate answer [17].

Final merging & ranking
In the last step, Watson evaluates all the scores of each candidate answer by merging them together and give those answers a final rank to decide if it is confident enough to answer the question. When Watson is confident enough to give a correct answer, based on the top ranked candidate answers, it will answer the question. If the highest ranked possible answer is rated too low, below a certain threshold, Watson will not answer the question [17].

3 IBM Watson in Applied forms

IBM Watson is currently being used in many different industries and professions. IBM Watson also has many different products available and is already applied in various fields of work. For example in education, financial services, healthcare and commerce. Watson has already proven that it has its excellency on the traditional forms of computing, such as reporting, analysing statistics, business processing solving business problems, and collaboration. This means that Watson could combine statistical analysis with its ability to answer questions about possible events that could be an investment risk [18]. With IBM Watson it is possible to easily make a customized chatbot in existing platforms. Some chatbots in Facebook Messenger, Whatsapp or Telegram are already made with IBM Watson. One of the example of this is Royal Bank of Scotland (RBS). RBS launched the chatbot “Luvo” to help their customers online with their questions. Luvo will pop up on the side of the browser when the user browses the bank's webpage and it is able to answer basic questions. Luvo already covers 10% of the bank’s interaction with their customers online, but if Luvo cannot help the customer, a question like ‘Does this conversations answers your question correctly?’ will show up. If the customer answers ‘no’, a real human service employee will take over the chat. Luvo is built using the IBM Watson Conservation service. More about this service will be described below [19].

4 Experiment: Making a IBM Watson powered chatbot

In this section we are going to make a chatbot with IBM Watson and which services you could use to make a smart chatbot.

Before you start building
First it is important to give your chatbot a purpose or goal. Second is important to understand the capabilities of the platform the chatbot will be integrated in. An example of this is Facebook Messenger where it is possible to craft a greeting window to start chatting with the chatbot. When a user starts chatting with the bot in Messenger it is also important to make use of buttons that give the user the possibility to answer quick and understand what kind of responses the user can give. Third, if a user is able to respond with free-form text input it is essential to help the user in this process by help messages that explain the syntax that the chatbot is working with. Fourth, a user always have to be able to cancel or unsubscribe from

the chat. When you as a developer do not implement something like that it could be experienced as spam, which is not good for the overall user experience [20].

4.1 Watson Conversation

In IBM's Bluemix cloud platform there are multiple services you could use to integrate into your bot. There are services available for natural language processing, to analyze emotions or tone of voice, image recognition and many more. In this part, we will explain how to make a chatbot in the very simple steps among many different tutorials and experiments we conducted.

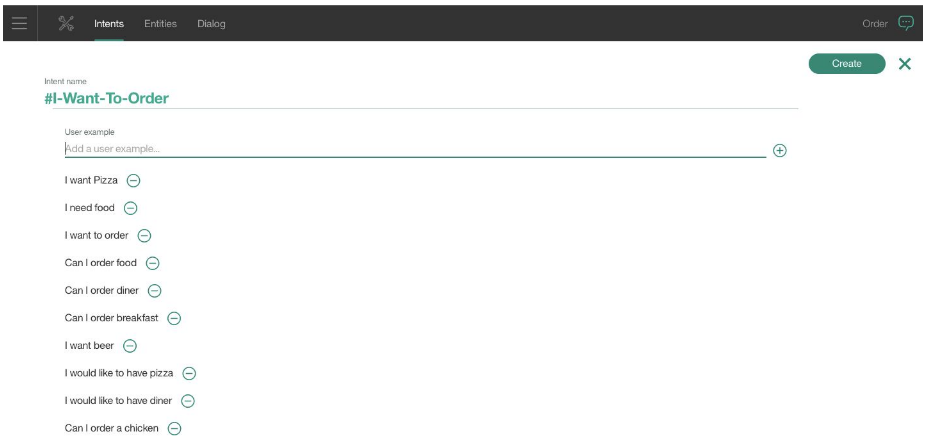
To make a chatbot you could use the Watson Conversation service to create virtual agents (bots). Watson Conversation has the key functions to make a simple chatbot. With this service Watson makes it easy to build a chatbot that understands messages posed in text and integrate them in existing messaging applications. It is not necessarily to have coding experience, because it has a visual dialog builder, which helps you to create these conversations easily [21]. This service already includes natural language processing where you can create a structured conversation and it provides an API you could call from any website or app.

Create a Bluemix account

First you need to create a Bluemix account. You can sign up at IBM Bluemix and get a free 30 days trial if you do not have your Bluemix account yet. When you are signed in you can navigate to the Watson services in the catalog and select multiple services. Here you can add the Conversation service to your account.

Creating conversations

The conversation service provides a web interface where you can add dialogs without having to code. Here you can add intents and entities to let Watson learn to understand what the user is saying and to create a suitable response. For example, when a user wants to know where something is, you can create a intent like: *#I-Want-To-Order.* It is possible that your users use other words to order something. Therefore you can add examples like I want pizza, I need pizza, etc (see figure 1).



Intent name

#I-Want-To-Order

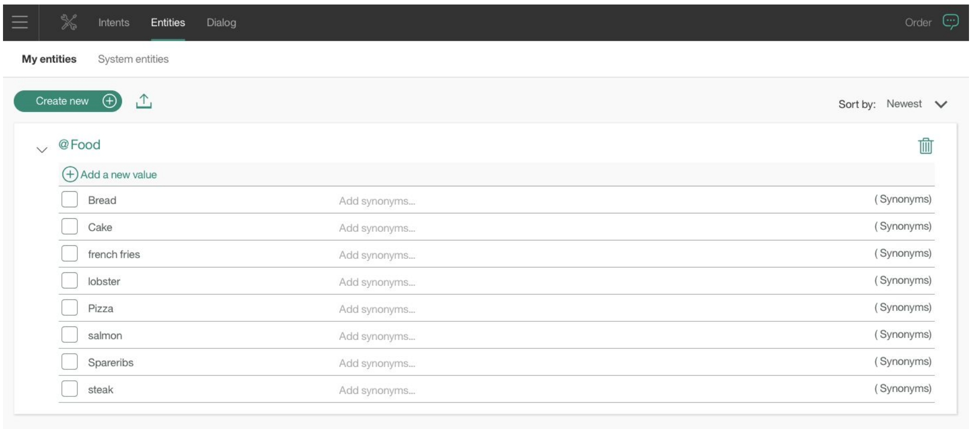
User example

Add a user example...

- I want Pizza
- I need food
- I want to order
- Can I order food
- Can I order diner
- Can I order breakfast
- I want beer
- I would like to have pizza
- I would like to have diner
- Can I order a chicken

Fig 1. Intens.

When you add those examples, Watson can learn the meaning of the intent and now can answer a question when it is asked in a different way. Still, Watson does not know what is being asked or what to the user is looking for. Therefore you have to creates entities, which are parameters of what the user is saying. This could be anything like objects, places or persons, but in this case we use meals or drinks (see figure 2).



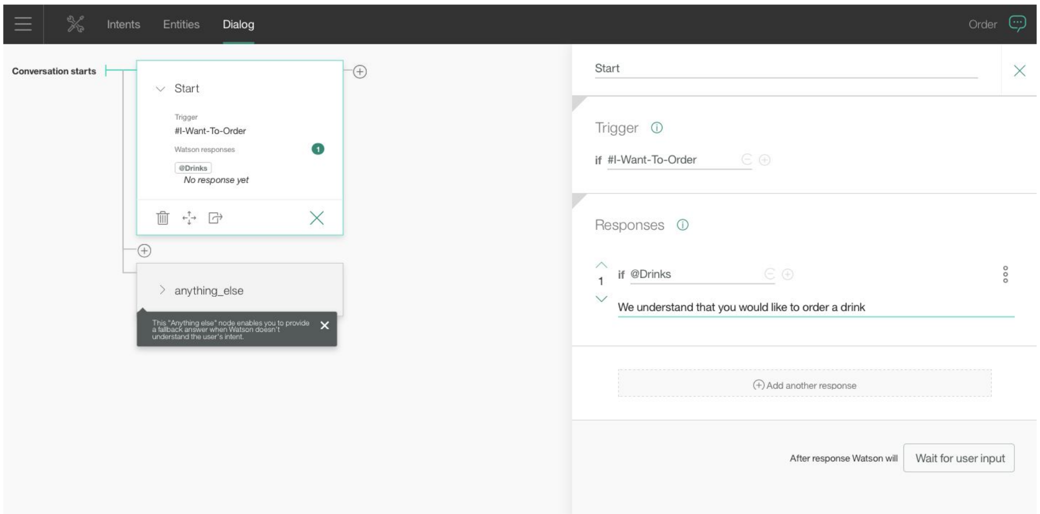
@Food

Add a new value

□	Bread	Add synonyms...	(Synonyms)
□	Cake	Add synonyms...	(Synonyms)
□	french fries	Add synonyms...	(Synonyms)
□	lobster	Add synonyms...	(Synonyms)
□	Pizza	Add synonyms...	(Synonyms)
□	salmon	Add synonyms...	(Synonyms)
□	Spareribs	Add synonyms...	(Synonyms)
□	steak	Add synonyms...	(Synonyms)

Fig 2. Entities.

After you have done this you can start making conversations using the dialog builder where you can create the responses of the chatbot. A response could be: Question: I need a pizza Response" Ok, What kind of pizza do you want? (see figure 3).



Start

Trigger #I-Want-To-Order

Watson responses 1

@Drinks No response yet

This "Anything else" node enables you to provide a fallback answer when Watson doesn't understand the user's intent.

Start

Trigger

if #I-Want-To-Order

Responses

1 if @Drinks

We understand that you would like to order a drink

After response Watson will Wait for user input

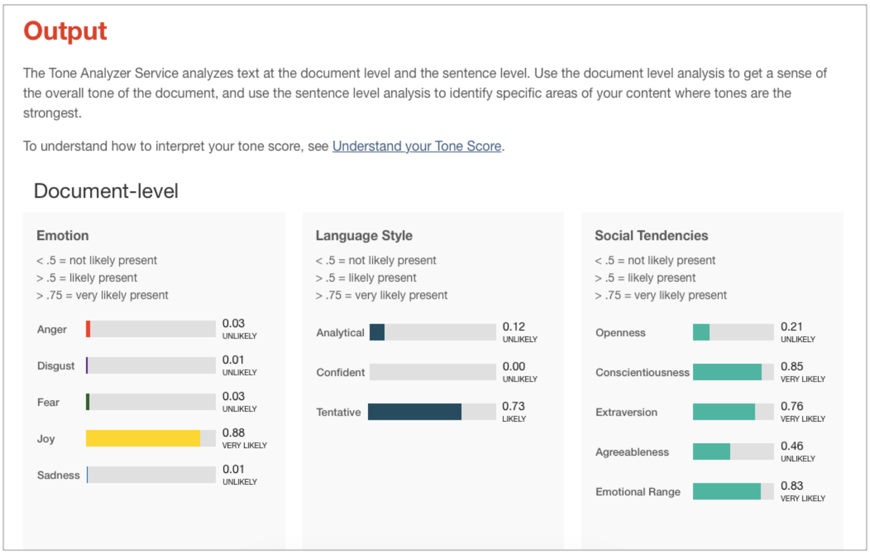
Fig 3. Dialog builder.

In this example you could see that we use an intent and an entity. Because we trained Watson that "I want to order a pizza" could be asked differently it will also use the other examples within the intent. So when someone asks something like "give me a pizza" Watson will understand that too and look for the intent and in the entities to see if it could give a response. Using this dialog builder you also could let Watson learn the context of a follow up question. For example when the user asked "do you have pizza salami" and after Watson's response asks "is there also ham on it" Watson understands that the user is still talking about the pizza with salami.

By making a bot based on the Conversation service you could easily create a bot that can answer simple questions via a script. It needs a lot of dialogs for communicating with the users, which makes it difficult to scale. Also, when a user poses a question which is not mentioned in the dialog builder, Watson is not able to answer that question because it does not know what kind of response to give. To make your chatbot smarter you could make use of other services within the IBM Bluemix platform. In the next sections we will explain more about those services.

4.2 Watson Tone Analyzer

Watson tone analyzer makes it possible to analyze tones, emotions and writing style in written text [22]. With this service you can easily analyze the emotions: happy, angry, disgust, fear, sadness and joy. Social tendencies like: openness, Conscientiousness, extraversion, agreeableness and emotional range and the language styles: confident, analytical and tentative (see figure 4). in order to make a chatbot more human-like or help the user better when they are frustrated with your service Watson makes it possible to respond differently when it analyzes a certain emotion. For example, when someone is waiting for a long time on his food to be delivered and becomes angry the bot can adjust the tone of his responses to calm down the angry customer or a human customer service employee can overtake the bot and chat with the customer to calm it down. In the Watson Conversation service you can use the output of the tone analyzer where you can add different responses on different types of emotions, tones or writing styles.



Output

The Tone Analyzer Service analyzes text at the document level and the sentence level. Use the document level analysis to get a sense of the overall tone of the document, and use the sentence level analysis to identify specific areas of your content where tones are the strongest.

To understand how to interpret your tone score, see Understand your Tone Score.

Document-level

Emotion

< .5 = not likely present
> .5 = likely present
> .75 = very likely present

Anger	0.03 UNLIKELY
Disgust	0.01 UNLIKELY
Fear	0.03 UNLIKELY
Joy	0.88 VERY LIKELY
Sadness	0.01 UNLIKELY

Language Style

< .5 = not likely present
> .5 = likely present
> .75 = very likely present

Analytical	0.12 UNLIKELY
Confident	0.00 UNLIKELY
Tentative	0.73 LIKELY

Social Tendencies

< .5 = not likely present
> .5 = likely present
> .75 = very likely present

Openness	0.21 UNLIKELY
Conscientiousness	0.85 VERY LIKELY
Extraversion	0.76 VERY LIKELY
Agreeableness	0.46 UNLIKELY
Emotional Range	0.83 VERY LIKELY

Fig 4. Tone analyzer.

4.3 Watson Retrieve & Rank

In order to make the chatbot smarter, it is better to let the bot search in database for answers. IBM offers a service called Retrieve and Rank which is a search engine with machine learning capabilities.. The service consists of two parts:

- **Retrieve**: SolR based search component that can retrieve a list of possible matches from your documents giving a query
- **Rank**: A Machine learning component that learns to rank those matches to sort them. So that the best possible match gets a higher ranking. Because it is a machine learning based solution it means that you have to train it. It will learn from the training how to handle your data and improves the relevance of the search results by time and training [23] [24].

Training the data

There is a three-step process of training regarding the Retrieving and Ranking. It collects the data, reviews it and retraining itself. After uploading your files with data you can add question that it can train with. The question must be formated in a plain text file and uploaded to the Retrieve and Ranking environment. After you have completed these steps Watson will ask you to perform an amount of training tasks, e.g. rating the answers Watson found in the documents. The rating is done by giving an amount of stars from 1 to 5. The amount of stars is used to determine which answer is eventually to best one to rank on top. When new data is added it will use your previous ratings to enhance the process of finding the correct answer. This will improve the learning process of Watson.

4.4 Watson Tradeoff Analytics

When Watson has to make decisions and there are a lot of options to choose from, it can be really difficult to choose the best one. Especially when the user gives conflicting criteria. For example, when the user asks “I want to buy the best but cheapest phone. Normally, the best phone is not the cheapest which is a conflicting criteria. With this service you also could personalize the decision making process by using the user’s individual preferences. When the user sets up its preferences, Watson Tradeoff Analytics will analyze these and will give you the best-balanced option by balancing the conflicting criteria [23]. With this service, you could make a chatbot that can make more complicated decisions based on the preferences of the user and becomes better when the bot serves the user satisfiable results every time [25].

4.5 Other services to use

Besides the discussed services of IBM Watson there are several other products and APIs that IBM provide, which are Watson Discovery, Watson Conversation, Watson Virtual Agent and Watson Knowledge Studio.

Watson Discovery
Watson Discovery helps people to quickly search the answers or patterns within a set of data to let them make better decisions with its cognitive computing function. Initially, it explores with a simplified sourcing language datasets quickly. Then, it can discover the certain values from the mixed up data by normalizing, altering and enhancing it [26].

Watson Virtual Agent
Watson Virtual Agent is a pre-trained chatbot you could use for customer service. Because the bot is already trained to work in a lot of domains you only have to add data like company information, documents, etc. It automates certain services to your customers. It also provides some insights and analytics about your customer’s engagement, which help you to understanding your customer’s needs [27]. To use this service you need to have a paid subscription.

Watson Knowledge Studio
Watson Knowledge Studio is a cloud-based application that discovers deep insights and knowledge in random text without any coding. When you teach it to find certain data within the unstructured text , it identifies the relationships of the texts in the given data and easily deploys them. It will accelerate the training process by saving time. You could buy this product with a monthly subscription [28].

APIs

In Watson Developer Cloud, several versions of API's are available. For the products like Watson Discovery and Watson Conversation, you could easily get the API's in there. Other products like Watson Knowledge Studio and Watson Virtual Agent do not have open API's. Other than that, there are also API's available for the functions like natural language understand and visual recognition with color tagging [29].

5 Discussion and conclusion

In this technical report, we have discussed the way to make a chatbot using IBM Watson. Although there are limitations on the methodology and APIs of IBM Watson, we proposed to make a chatbot with the certain API's and the services from IBM that we illustrated above. By using Watson it could be easy and more efficient to make an intelligent chatbot, but the conversations with the customer still are scripted. Furthermore, some important initial steps like training the data have been done manually. It is not possible to make a completely automated chatbot yet, but with using the services of Watson, IBM makes it possible to make a chatbot that learns from earlier interactions or data and respond in other ways in different kind of situations.

If you want to make a smart chatbot, some coding skills are required. The several services do have a web interfaces to train the bot with to make it smarter, but it is still not that easy to make a chatbot that can do a lot of different things.

References

1. Business Insider (2016, September 20). Messaging apps are now bigger than social networks. Retrieved April 08, 2017, from http://www.businessinsider.com/the-messaging-app-report-2015-11?international=true&r=US&IR=T
2. O'Brien, C. (2016, November 11). Facebook Messenger chief says platform's 34,000 chatbots are finally improving user experience. Retrieved April 08, 2017, from https://venturebeat.com/2016/11/11/facebook-messenger-chief-says-platforms-34000-chatbots-are-finally-improving-user-experience/
3. Miller, G. (2016, November 13). Chatbot vs. AI Bot - which is here to stay, which to invest in and why? Retrieved April 03, 2017, from https://chatbotslife.com/chatbot-vs-ai-bot-which-is-here-to-stay-which-to-invest-in-and-why-7ea79a454df3

4. The conversational chatbox design challenge. (2016, August 15). Retrieved April 10, 2017, from https://www.ibm.com/developerworks/library/cc-cognitive-chatbot-conversational-design/index.html
5. Ferrucci, D. (2010, September). Build Watson: an overview of DeepQA for the Jeopardy! challenge. In Proceedings of the 19th international conference on Parallel architectures and compilation techniques (pp. 1-2). ACM.
6. Lally, A., Bachi, S., Barborak, M. A., Buchanan, D. W., Chu-Carroll, J., Ferrucci, D. A., ... & Patwardhan, S. (2014). WatsonPaths: scenario-based question answering and inference over unstructured information. Yorktown Heights: IBM Research. Retrieved April 01, 2017
7. Kelly III, J., & Hamm, S. (2013). Smart Machines: IBMÕs Watson and the Era of Cognitive Computing. Columbia University Press. Retrieved March 28, 2017
8. High, R. (2012). The era of cognitive systems: An inside look at ibm watson and how it works. IBM Corporation, Redbooks. Retrieved March 28, 2017
9. Mayer-Schönberger, V. and Cukier, K. (2014). Big data. 1st ed. Boston [u.a.]: Mariner Books, p.6.
10. Reese, H. (2017). IBM says new Watson Data Platform will 'bring machine learning to the masses' - TechRepublic. [online] TechRepublic. Available at: http://www.techrepublic.com/article/ibm-says-new-watson-data-platform-will-bring-machine-learning-to-the-masses/ [Accessed 5 Apr. 2017].
11. Zhong, N., Liu, J. and Yao, Y. (2003). Web intelligence. 1st ed. Berlin: Springer, pp.1.1.
12. Sridhar Sudarsan, (2017). An Ecosystem of Innovation: Creating Cognitive Applications Powered by Watson. Somers, NY 10589: IMB, p.6.
13. Kenyon, T. (2012). Cloud Computing Whitepaper. Retrieved April 1, 2017
14. Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. International Journal of Information Management, 35(2), 137-144. Retrieved April 01, 2017
15. Bao, J., Duan, N., Zhou, M., & Zhao, T. (2014). Knowledge-based question answering as machine translation. Cell, 2(6). Retrieved April 03, 2017
16. "FAQs." IBM - DeepQA Project: FAQs. IBM, n.d. Web. 10 Apr. 2017. <https://www.research.ibm.com/deepqa/faq.shtml#22>. Retrieved April 05, 2017
17. Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A. A., ... & Schlaefer, N. (2010). Building Watson: An overview of the DeepQA project. AI magazine, 31(3), 59-79.
18. High, R. (2012). The era of cognitive systems: An inside look at ibm watson and how it works. IBM Corporation, Redbooks.
19. Williams-Grut, O. (2016, September 28). RBS is launching an A.I. chatbot called ‘Luvo’ to help customers. Retrieved from https://www.businessinsider.nl/royal-bank-of-scotland-launches-ai-chatbot-luvo-using-ibm-watson-2016-9/?international=true&r=UK
20. Yuan, M. (2016, August 15). The conversational chatbox design challenge. Retrieved from https://www.ibm.com/developerworks/library/cc-cognitive-chatbot-conversational-design/index.html
21. Conversation | IBM Watson. (n.d.). In Watson Developer Cloud. Retrieved April 05, 2017, from https://www.ibm.com/watson/developercloud/conversation.html
22. Watson Developer Cloud. (n.d.). Retrieved April 10, 2017, from https://www.ibm.com/watson/developercloud/tone-analyzer.html?cm_mmc=Earned-_-Leaders

hip%2BAgenda_IBM%2BWatson%2BCore-_-WW_WW-_-twilio&cm_mmca1=000000OF&cm_mmca2=10000409&%3Flang=null

23. Yuan, M. (2017, February 06). Create a chatbot, Part 3 - Use cognitive (or artificial intelligence) services to enhance a chatbot. Retrieved from https://www.ibm.com/developerworks/library/cc-cognitive-chatbot-watson/index.html
24. Retrieve and Rank | Overview of the Retrieve and Rank Service. (n.d.). In Watson Developer Cloud. Retrieved April 05, 2017, from https://www.ibm.com/watson/developercloud/doc/retrieve-rank/index.html
25. Tradeoff Analytics | Using the Tradeoff Analytics service. (n.d.). In Watson Developer Cloud. Retrieved April 05, 2017, from https://www.ibm.com/watson/developercloud/doc/tradeoff-analytics/index.html
26. Discovery | IBM Watson. (n.d.). In Watson Developer Cloud. Retrieved April 05, 2017, from https://www.ibm.com/watson/developercloud/discovery.html
27. IBM Virtual Agent. (n.d.). In IBM. Retrieved April 05, 2017, from https://www.ibm.com/us-en/marketplace/cognitive-customer-engagement
28. IBM Watson Knowledge Studio. (n.d.). In IBM. Retrieved April 05, 2017, from https://www.ibm.com/us-en/marketplace/supervised-machine-learning
29. IBM Watson Developer Cloud. (n.d.). In Watson Developer Cloud. Retrieved April 05, 2017, from https://www.ibm.com/watson/developercloud/