Robot process automation – a chatbot implementation

ABSTRACT

In this task, I created a chatbot for a bank, with appropriate functionality and prototype implementation. This paper also explores the benefits of implementing Artificial Intelligence into chatbots.

1 Introduction

A chatbot is a conversational software agent, which interacts with the users by using natural language. The idea of a chatbot was originated at Massachusetts Institute of technology [1]. In 1966 Joseph Weizenbaum developed ELIZA a computer program that interacts and demonstrates communication between a human and a computer via natural language [2]. When the bot responded it took the role of a psychotherapist, with a basis of a structured dictionary, and looked for keywords in entered text [3]. Even that ELIZA was a very straightforward bot with the predefined direction of dialogue and keyword are still used in today's chatbots. In the past few years, bots have been growing exponentially way, with faster developments of artificial intelligence, platforms, communications devices, and speech recognition [4].

Communication and interaction between bots and humans have been increasingly controlled and determined by algorithms, this affects the efficiency and convenience advantages as the next logical level of evolution. At the very start, bots were able to answer simple, repetitive questions like "what is the weather like today? ". With the advantages of fixed chatbots and artificial intelligence chatbots, it takes the bots to another level with more demanding tasks. Some problems still occur, it is not always easy to see the difference between a machine and a human in a conversation, with some artificial intelligence can create an illusion of a natural human interaction and fixed chatbots acts like humans but stands out more. One of the problems for the developers is to create a common language between robots and humans to elevate efficiency and communication higher. Therefore speech is one of the most powerful tools in the form of communication between humans, is the ambition in the human-computer interaction to improve speech interaction between humans and computers to simulate human to human speech interaction [5]. These days, it has been receiving interest with contribution from Google, IOS and Android, because they are more than graphics-based interfaces, which manipulating graphical icons, like buttons, scroll bars, windows, tabs, menus, cursors, and even the mouse pointing device. Speech dialogue systems are the primary interaction method with a machine [6], speech interaction will have a significant role to make computers more human-like in the future [7]. A lot of focus has been improving the recognition rate of human voice and the technology has been approaching for speech-based human-computer interaction. But when it comes to communication in chatbots the speech has been split into speech recognition, speech parsing, natural language processing, keyword identification, chatbot design/personality, artificial intelligence, etc. A chatbot is a program that has the ability to have a conversation with a human by using natural language speech [5].

In this paper, a service chatbot for a banking system will demonstrate how it handles requests and deliver the services to the users. The paper will be reflecting on the solution for the application and its design decisions founded in the established knowledge. There has been a long history of various chatbots with various techniques such as information retrieval and template rules, and recently deep learning and machine learning have been implemented as a chatbot technique. The focus question is, what is the limitation of a fixed chatbot against an AI chatbot and what actions need to be done to make the user experience better?

2 Related Work

In this section, it will go through classification of chatbots and some existing solutions for different chatbots.

2.1 Classification of chatbot

A messaging service chatbot has grown and spread rapidly over recent years, its various functions from payment, ordering, and booking, etc which requires an application or a website. It is easier for the users to perform all the tasks through their messaging apps, for instance, Facebook Messenger, WhatsApp, etc [8]. Chatbot applications can be grouped into four different categories, service, commercial, entertainment, and advisory chatbot [9]. For example, in this case, a chatbot for a bank will respond to questions then provide the customer with bank services and functionalities. Commercial chatbots can be take out restaurants that can take delivery orders or notify the customers through a messaging app. Entertainment chatbots are designed to make the customer engaged with sports, festivals, concerts, movies, etc. Advisory chatbots offer to provide suggestions, recommendations on service. A chatbot can be divided based on three different groups as shown in Fig 1.

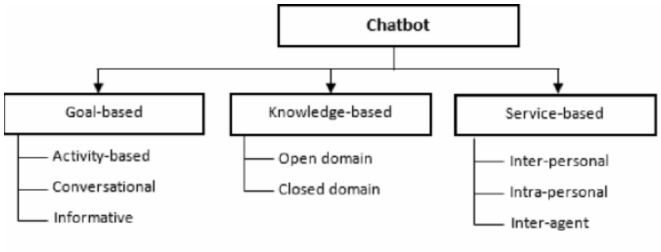

Figure 1. Chatbot applications.

There are two types of chatbots, a chatbot based on Artificial Intelligence(AI) being constantly updated through multiply interaction with customers, it is very intelligent with a complex design. But a fixed chatbot, has the limitation of having fixed information, limited help, they are often used for customers with limited access or repetitive questions without being able to understand human behavior.

2.1.1 Goal based chatbot

Goal based chatbot's main goals to help the user to archive the needs. They are designed for a particular task to have a short conversation to extract information from the user to complete the desired task [10]. For example, a website selling computers deploys a chatbot on their website to help the customer to answer the questions or address the problems.

2.1.2 Knowledge based chatbot

Knowledge based chatbots are classifying based on the knowledge given and the amount of the data they are trained on. There are two data sources open domain and closed domain. Open-domain answers depend on general topics and respond. A closed domain is focused on a particular knowledge domain, where the data is provided from the company itself [11].

2.1.3 Service based chatbot

Service based chatbots depend on the functions and services it can provide to the customer. It can either be a personal or commercial purpose, for instance, a banking company wants to provide the customer with their bank services and functionalities.

2.2 Existing chatbots

In June 2017, the department of FUJITSU Financial Services Solution Finplex Robot Agent Platform (FERP) released an enterprise chatbot service for the financial industry. The concept behind FERP is to respond to user's needs to access information as quickly as possible anytime anywhere and to services provider needs for visualization, in this case, FUJITSU made a deal with Sony Bank [12]. Sony Bank intention behind creating a chatbot was to improve the level of service for existing users, and the application should handle Frequently asked questions(FAQ) from existing customers who already have a loan contract. The system includes a script editing function for creating different scenarios involving dialogs with robots combined with an automatic response mechanism. The chat screen operation is shown in Fig 2.

Figure 2. FRAP chat screen.

The idea was to enable the customer to construct the services provided through a chatbot, it also simplifies the creating and editing of the script that exchanges between the chatbot and the user. This function enables the customers to create and edit scenarios on their own and launch their chatbot. FERP also implemented AI to automatically generate words or phrases to machine learning to text in already existing FAQs, call center, response manuals, business manuals, etc and data in official documents. This system is a goal- and service based chatbot, where it uses different approaches to make the user experience better.

Google created a framework for chatbot called [13], DialogFlow. The framework has a natural language processing(NLP) platform that can build a conversational bot for various customers in various languages through multiple platforms. It let various companies develop their unique chatbot and let the users interact through an interface in either voice or text exchange. It is powered by machine learning, and combining it with NLP it will recognize the intent and the context behind the user's inputs. It will use the relevant data to provide a conversational chatbot with responses, depending on what the user is asking for.

Xu, Liu, Guo, Sinha, and Akkiraju [14] has provided with a new conversational system for customer services on social media. This is a service based chatbot which makes it easier for internet users to use social media for help. They created state-of-the-art deep learning techniques such as long short-term memory (LSTM) networks were first applied on social media, where the system requests an input, computes its vector representation then feel it to LSTM, then respond with an output. With the growth of social media and the number of user requests, it has become a lot more challenging process to respond to every single incoming request [15]. With the lack of manpower, a lot of companies form their dedicated customer services to respond to users on social media. However, manually addressing a lot of requests it is very time-consuming and often fails what the user expects. A recent study shows that 72% of the users who contacted a brand through social media expect to get a response within one [16].

Cui, Huang, Wei, Tan, Duan, and Zhou [17] provided with their customer service conversational chatbot called SuperAgent. This chatbot is gathering information about the product from HTML source code, product information, questions &answers, and customer reviews. This is an example of a knowledge-based chatbot, where it is being fed with information and it all depends on the knowledge it has. Bank of America [18] recently launched its own AI chatbot, Erica. It helps a customer with banking related problems, functions from credit score changes, refund confirmation, bill reminders, balance watch, duplicate charge monitor, etc.

3 Scenario and idea description

In this section the scenario behind the whole chatbot idea for a bank will be described and

3.1 Scenario

It is Monday afternoon and the customer got a notification on his phone from the bank that today is the due day to pay a bill. He saw that he needs to pay 760 Kr for the electricity bill, therefore he decided to use his hone which he already holds in his hands instead of turning the computer. He opened the bank app and contacted the bank chatbot, and the scenario for what things he wants to do through the chatbot is:

1. Log in to verify who he is.
2. Pay his bills and look at the receipt
3. Take a look at his recent transactions to make sure everything is right.
4. He wants to check how much money left he have.
5. He wants to update his email and phone number.

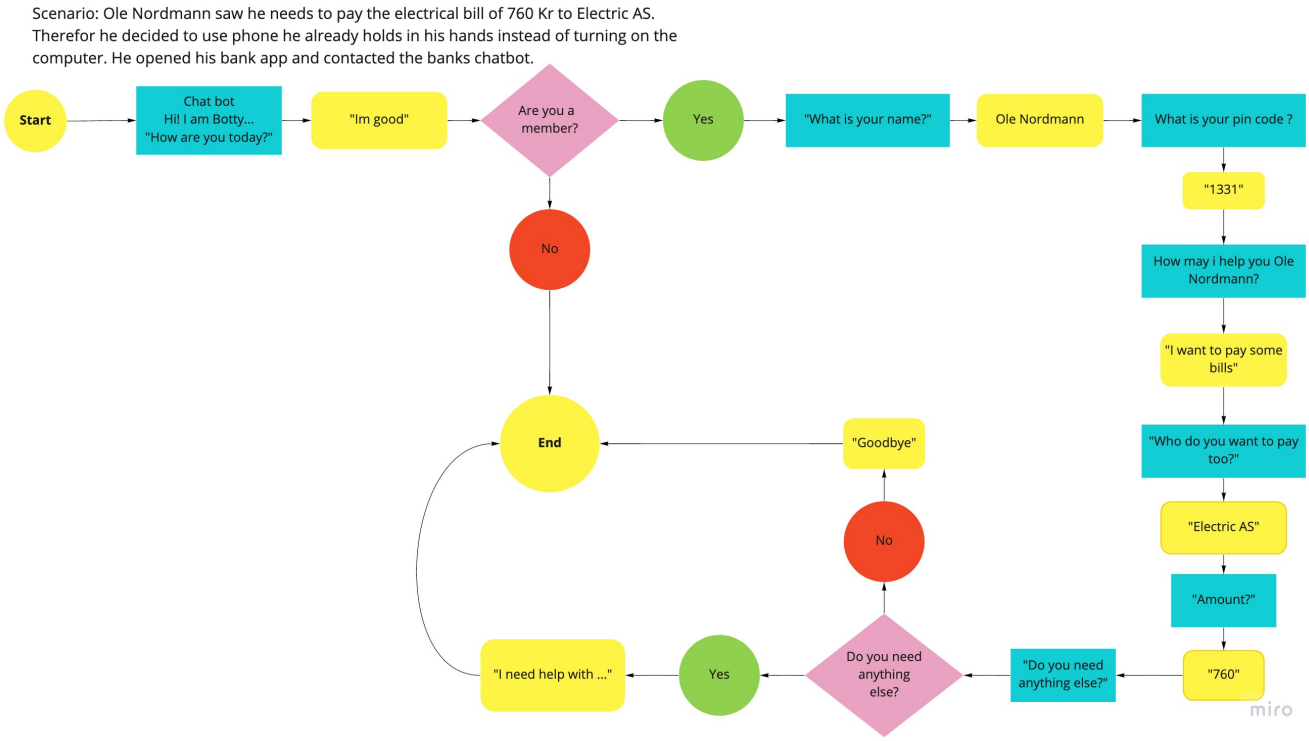

Figure 3. This flowchart shows how Ole Nordmann pays his bills through the chatbot

The school did not provide us with any data or any specification on what kind of chatbot it should be. Therefore it was decided the chatbot will be developed was for a bank. When it comes to the idea of designing a chatbot for a bank, it needs some features like login, makes a payment, checks balance, sees latest transactions, show account info, and change account info. With all of these features, a chatbot should be able to help the customer and make it easier for both parts, as shown in Fig 3. The developed chatbot does not have AI implementation and it is a fixed chatbot. However, it should be flexible enough to understand whats the user wants by looking for specific keywords or intents. The customer should be able to use the provided services from the bank, as illustrated in Figure 4.

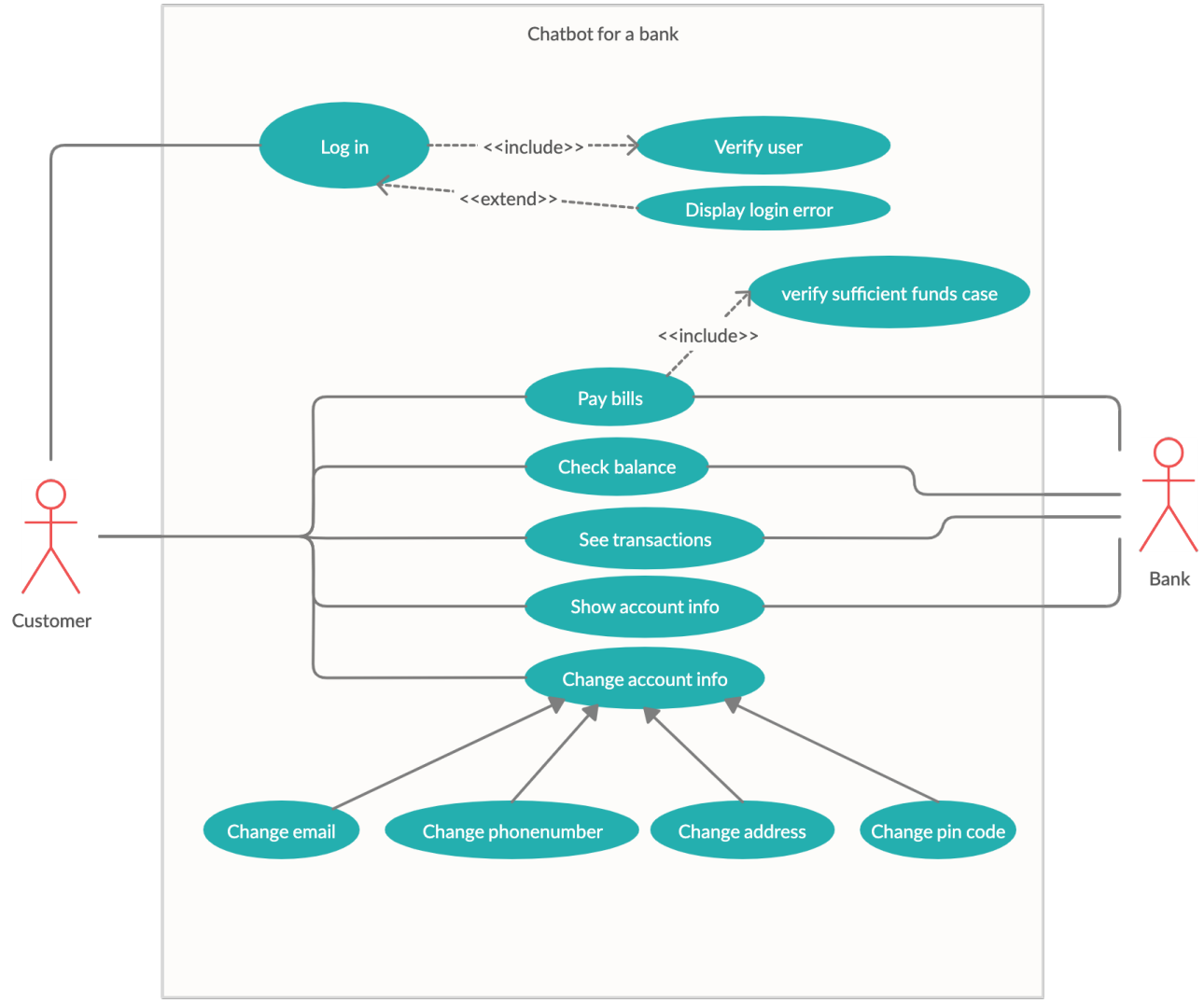

Figure 4. User case diagram for a banking chatbot.

4 Design (/ development)

Listing 1. My Javascript Example

When it comes to developing a chatbot it is important to identify the opportunities the bot can provide for the company, and what kind of chatbot it needs to be created. Microsoft provided with a chatbot dialogs with a Bot Framework SDK [19], allows us to model a conversation and managing conversation flow. Communication is organized into dialogs, where the dialog contains waterfall steps and prompts. So it depends on how the bot is developed for the different interactions between a human person and the bot. By combing the Bot Framework with an emulator [20] allows us to test and debug bots. Using the emulator it is possible to chat with the bot and receive responses. This section will be explained the ideas and functionality behind each service.

4.1 Data for the chatbot

In this task a dataset was not available, therefore it was created a test dataset for the chatbot to function. It was created data for three different people, Ole Nordmann, Hans Petter, and Julie Hansen. Each person was given their own "database" with a set of information from id number, income, loan, personal information, etc. For the chatbot to function it was important to create a test dataset with a structural way, illustrated in Listing 1.

To test bankfunctions, it was needed a fake dataset with some test data inside. The data set contains three different people, Ole Nordmann, Hans Petter and Julie Hansen. Each person was given their own set with data from id number, income , loan , personal information, as shown in Listing 2. In order for the chatbot to function it was important to create a test dataset to see if the chatbot works like it should.

```
const data = [
  {
    name: "Ole Nordmann",
    pin: 1331,
    Id: 0,
    Job: "Yes",
    AccountNumber: 1234,
    income: 50000,
    loan: 1000,
    Savings: 10000000,
    Balance: 5500,
    Email: "Olenordmann@gmail.com",
    Address: "Main road 1 ",
    Phonenumber: 12345678,
    TransactionHistory: [
      {
        date: "31.01.2020", Place: "Coop Mega", Price: 500,
      },
      {
        date: "03.02.2020", Place: "Coop Prix", Price: 200,
      },
      {
        date: "07.02.2020", Place: "Coop Obs", Price: 1000,
      },
    ],
  },
  {
    name: "Hans Petter",
    pin: 1234,
    Id: 1,
    Job: "No",
    AccountNumber: 4321,
    income: 0,
    loan: 50000,
    Savings: 50,
    Balance: 420,
    Email: "Hans Petter@gmail.com",
    Address: "Main road 2 ",
    Phonenumber: 241512311,
    TransactionHistory: [
      {
        date: "02.03.2020 ", Place: "Rema 100", Price: 500,
      },
      {
        date: "02.02.2020", Place: "Narvesen", Price: 200,
      },
      {
        date: "23.03.2020", Place: "Coop Obs", Price: 1000,
      },
    ]
  }
```

Listing 2. Data for bank customers.

4.2 Log inn

Login is a security measure designed to prevent unauthorized access to confident data, in this case, customer "Ole Nordmann" should not be able to access "Hans Petter" data. For the simplicity of the chatbot, all it needed was a name and the pin code attached to that name to log in. For example, Bank of America requires you to log in before you can access your data and services. The reason behind this login system was to show how a bank login system might be, instead of the name it should be a social security number. As mentioned earlier in the scenario the person "Ole Nordmann" wants to log in as shown in Fig 5.

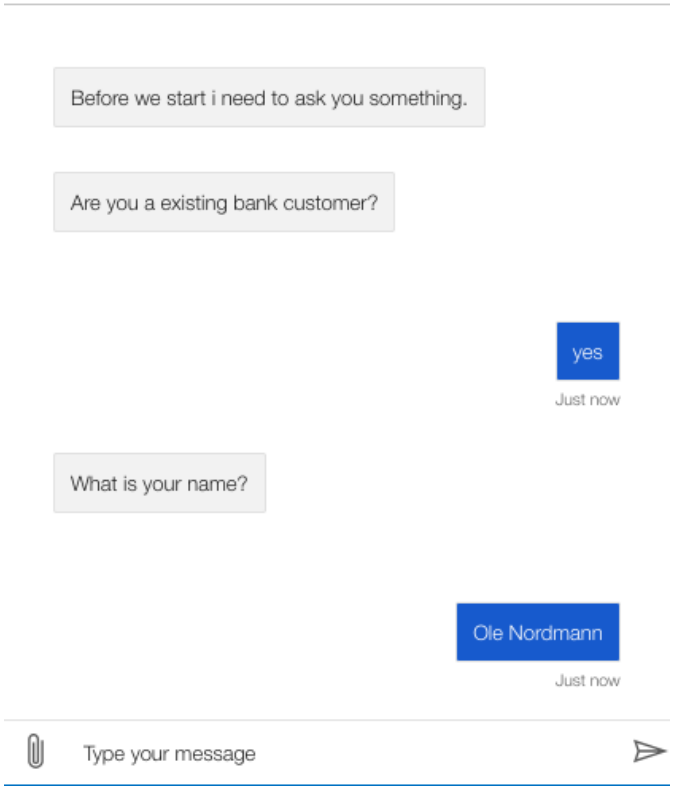

Figure 5. The customer is an existing customer and the bot asks who the person is.

A person can forget what his pin code is, and might like to have a few try to get the pin code right. Instead of starting the whole conversation from the start and make the customer write the same thing again, the chatbot will ask the customer to type in the pin code again, as shown in Fig.6.

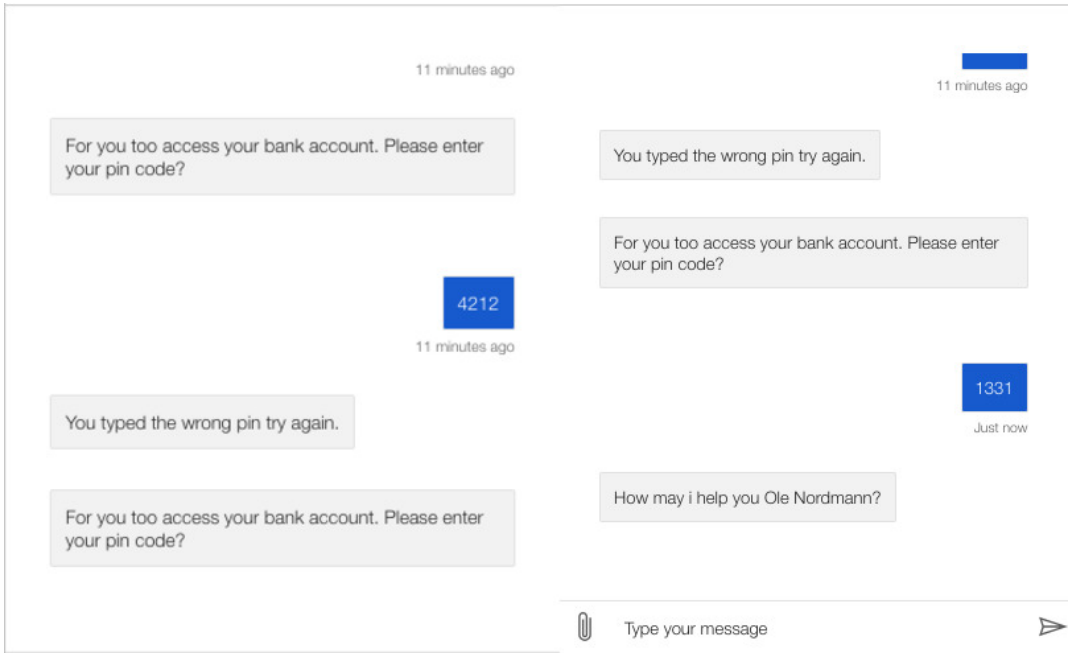

Figure 6. Chatbot conversation if the pin code is wrong.

4.3 paying bills, transaction history, and balance checking

After the customer gained access to his account, the customer can finally start to use chatbot services and functions. The chatbot will start with "How may I help you, Ole Nordmann". If we are following the scenario, Ole Nordmann wants to pay his bills to Electric AS. For the chatbot to know which function to use, it had been added a keyword "bills" to access the paying bill part. The bot will ask whom he wants to pay and the amount and when it is done it will print out a confirmation as illustrated in Fig 7.

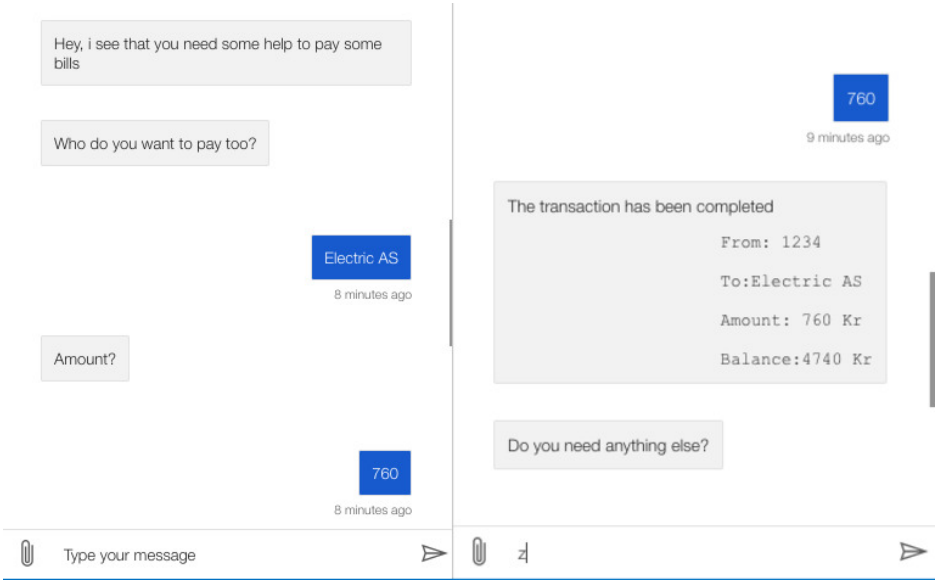

Figure 7. The flow of paying bills

To make the chatbot less buggy and more pleasant for the customer it is important to add some extra steps to check the input data is valid. For example, Ole Nordmann has to pay a bill, money is a numeric value and what if he miss typed something like "760b" in Listing 3 line 9, there is a function called isNaN(), the function checks if there are letters in the input. The chatbot will be able to detect that proceed to ask the customer to try again, as illustrated in Fig 8.

```
1  bot.dialog("/TransferMoney", [
2    function (session) {
3      builder.Prompts.text(session, "Amount?");
4    },
5    function (session, results) {
6
7      var temp4 = results.response;
8
9      if (isNaN(temp4)) {
10       session.send("The number is invalid, it can only contain numbers. Try again");
11       session.beginDialog("/TryAgainTransfer");
12     } else {
13
14       toMoney = results.response;
15       if (GlobalBalance > toMoney) {
16         GlobalBalance -= toMoney;
17         result = "";
18
19         var datetime = new Date();
20         let date = datetime.getDate();
21         let month = datetime.getMonth() + 1;
22         let year = datetime.getFullYear();
23
24         var CurrentDate = date + "." + month + "." + year;
25
26         dict = {
27           date: CurrentDate,
28           Place: toAccount,
29           Price: toMoney
30         }
```

```
        var item = data[GlobalUserId].TransactionHistory;

        item.push(dict);



        session.send(result);
        session.endDialog();
      } else {
          session.send("You dont have enough balance on your bank account.")
          session.send("Try again with another total amount.")
          session.beginDialog("/TryAgainMoney");
        }
    }
  }
  }
```

Listing 3. Checks if the input is numeric.

Another situation could be if the person is typing the wrong number "7600" instead of "760", as shown in Fig 8. The Chatbot will double-check if the customer has the balance to pay the bill, it will act as an extra safety layer and make the customer beware of the typo.

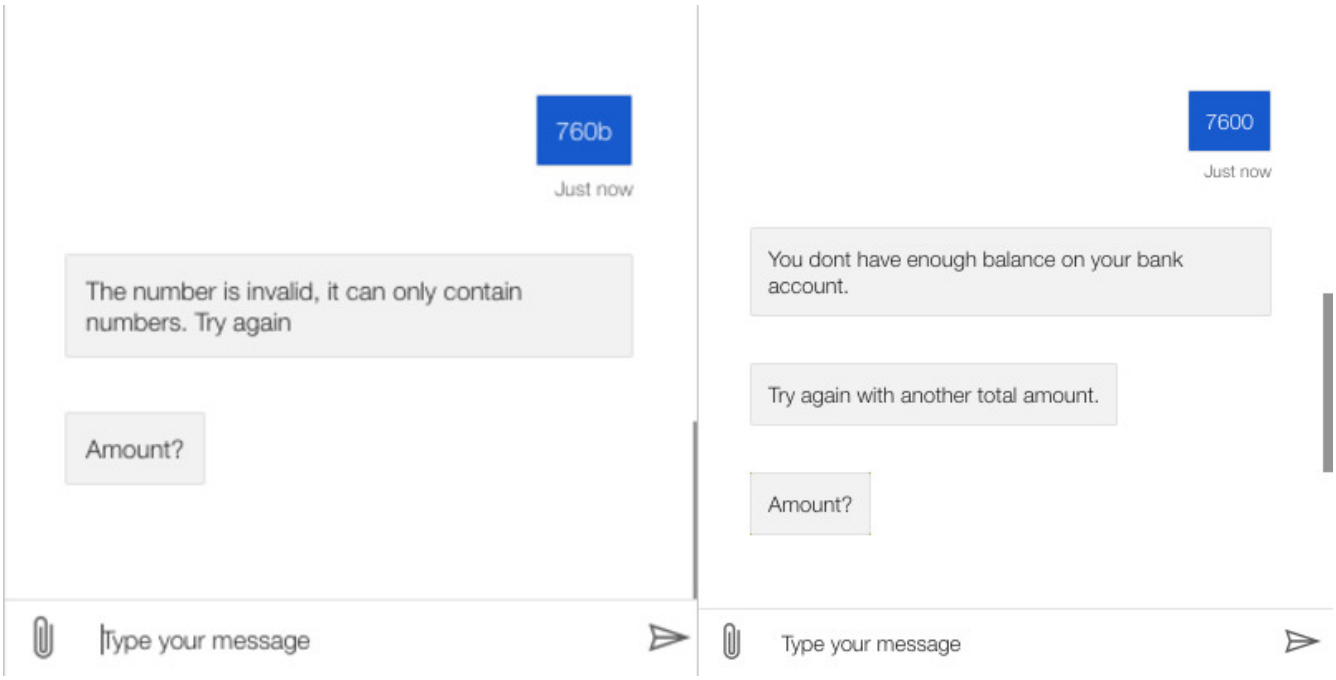

Figure 8. On the left side, it shows it there is a letter in the input. On the right side, it shows if the person does not have enough money.

In the next step of the scenario, the customer wants to cross-check with a receipt, transaction history, and the balance he has left. For the chatbot to know which functions the customer wants, the inputs need to contain the keywords "transaction" or "balance" in a sentence, as illustrated in Fig 9. A basic bank function, if there has been a payment it should be added to transaction history.

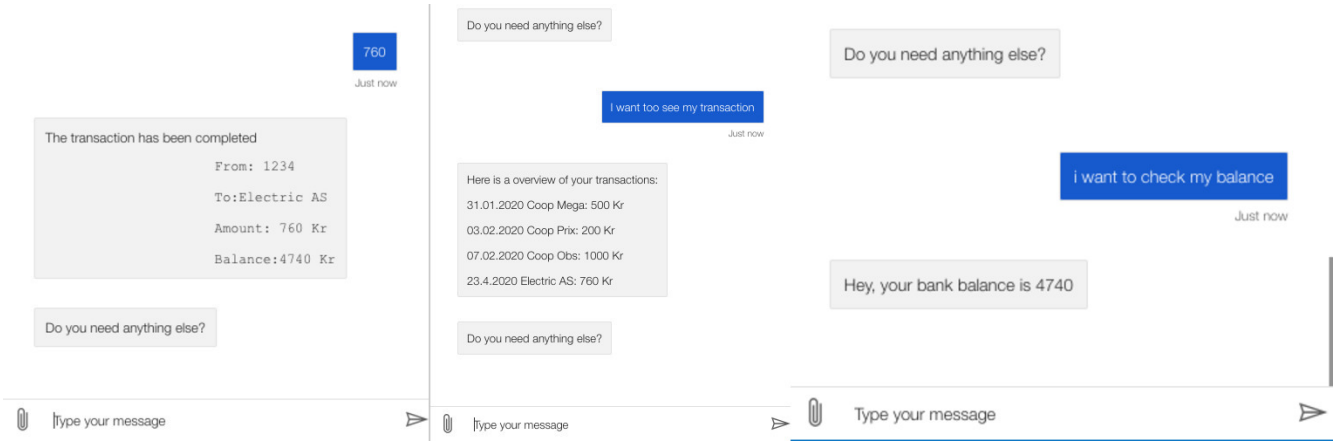

Figure 9. Shows how the customer can get the transaction history and his balance.

4.4 Update and show account info

In the next step of the scenario, the customer wants to update his email and phone number. He asks the chatbot about this account info and update. For the chatbot to know which functions the customer wants, the input needs to contain the keywords "show account info" and "update account info", as illustrated in Fig 10.

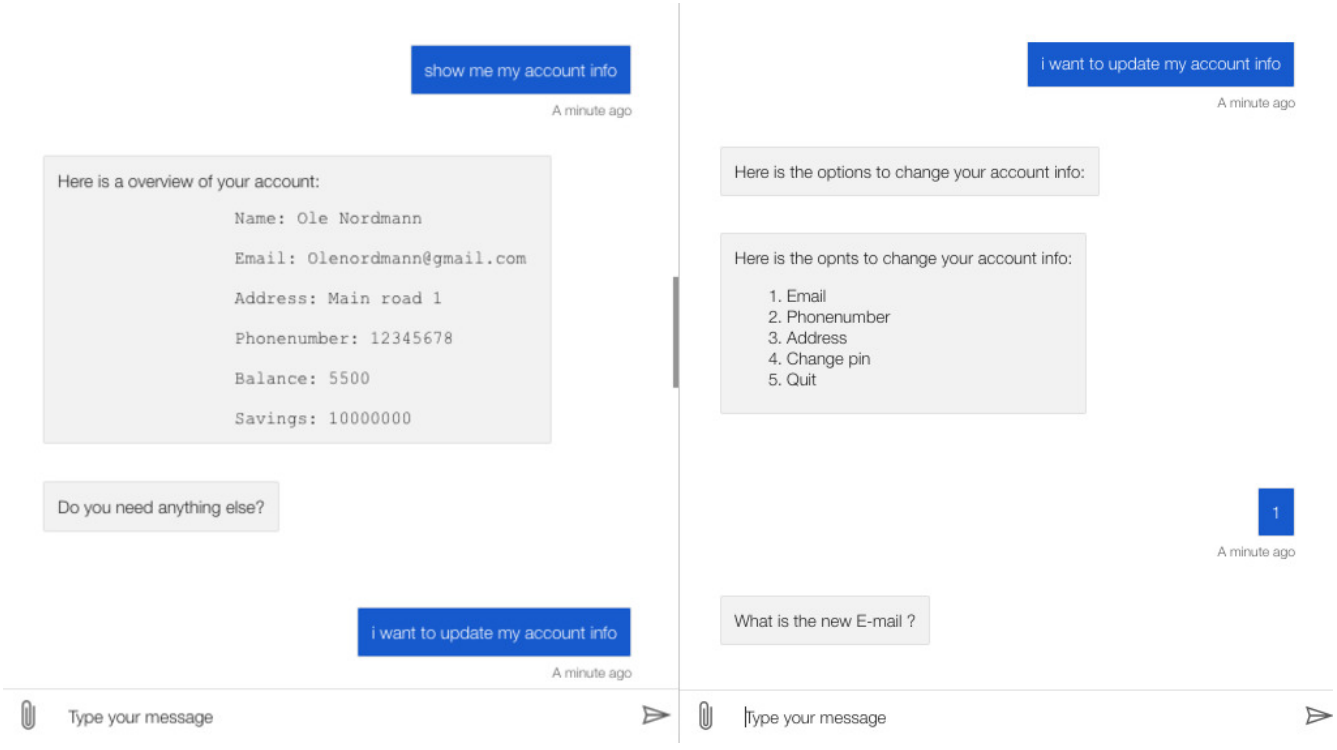

Figure 10. Conversation to show account info and update account info.

When data is being processed, storing of the data should also update the dataset. If the customer changes something, the chatbot will be able to change things in the data, as illustrated in Figure 11.

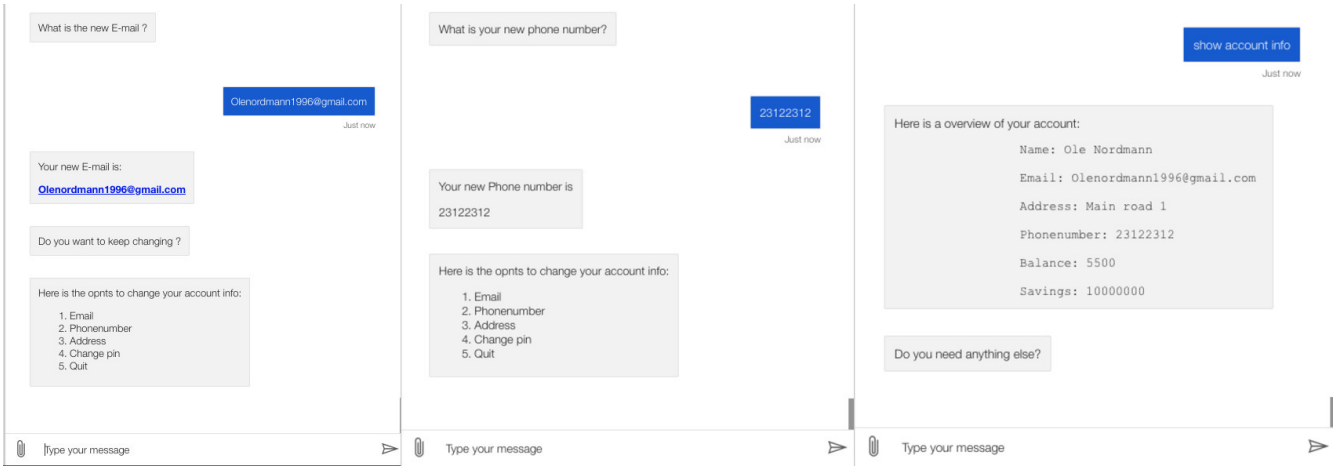

Figure 11. Conversation to show how things are being updated.

As mentioned earlier about the security layer it is important to provide the service the users want, even if he manages to type in the wrong things, for example forgetting "@" or having letters when writing his number, as illustrated in figure 12.

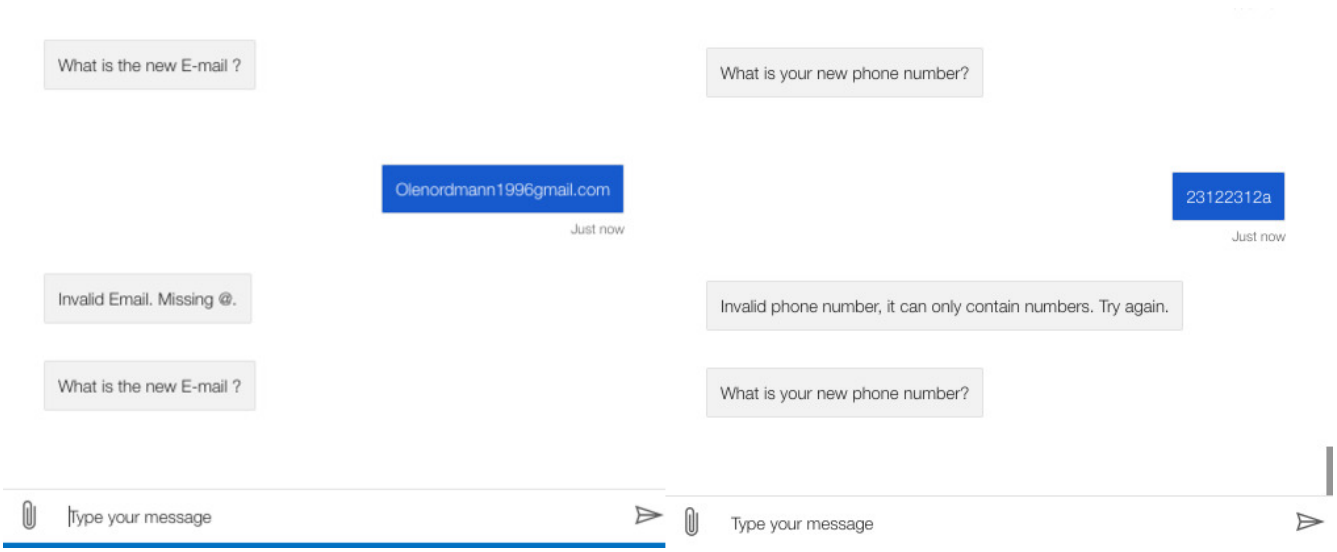

Figure 12. Shows an error if the input has a letter since the input can only be numeric.

5 Discussion and future work

In this work, it has been developed a prototype conversation chatbot for Bank related services, and functionalities and a user case diagram of a proposed chatbot are shown in Fig.4. As mentioned earlier in the Development of a chatbot, the development of this chatbot is a fixed chatbot without any human understanding and have a fixed path and limited help and does not achieve similar performance as human agents. Therefore some of the functionalities of the chatbot are clunky. The chatbots can be case sensitive in some parts in the dialog, for instance, it is case sensitive when the bot asks the name of the user as shown in Fig.5. If the user miss types his name with an uppercase or a lowercase the program will crash and everything has to be restarted. A chatbot not being able to memorize a conversation they already had with the same person comes to bad memory, which forces the user to write the same thing over and over again. This can be very annoying for the customer for the effort required, therefore it is important to be careful when designing and developing a chatbot, to make sure the program can understand the user's questions and needs to respond accordingly.

Fixed chatbots are generally easier to build with less infrastructure than bots with more complicated infrastructure like AI bots. In this case is to build a chatbot for a single purpose, customer services to provide very linear and single-dimensional support. They usually have a very specific technology as mentioned earlier in the development with a Bot Framework with an

emulator. It is also easier to create an architecture requirement and quicker and easier to build, deploy and therefore the cost of implementing this kind of chatbot is cheaper and less scalable and less robust when it comes to the long run with reliability, analytics, learning, and less investment. In our case this chatbot won't learn anything new and very static, it does understand or try to understand if something is typing wrong.

AI bots in the other hand are also similar to fixed chatbots, but it is far more complex, with more advanced levels, AI components, natural language processing capability, and the learning center once a chatbot is implemented, it can handle the request and queries anytime and anywhere, this is one of the main advantages of AI chatbot. It will provide better overall functionality and better customer experience and reducing time waste. The benefits of implementing machine- or deep learning into the chatbot, it can handle unseen data from interaction to update independently. For example, what if a person with special needs to use a chatbot. The person has dyslexia and trying to have a conversation with a fixed chatbot. The chatbot will not be able to understand the person if the user types something wrong. However, since AI chatbot is a lot better handling unseen data, it will try to learn something from the person and maybe answer something similar "Do you mean .... ? ". Chatbots are installed to speed up responses and improve customer interaction, but due to limited data and the time it needs for a self-update, this process and be slow and costly if a fixed chatbot is getting stuck, and the query does not relate to anything "service or data" it won't understand it. This can lead to a customer with frustration with bad customer experience. Customer service plays an important role in the organization's ability to generate income and revenue. It is often one of the most resource-intensive within a company. Support staffs spend a lot of time answering questions via telephone or message application to make sure their customer is satisfied with their business. With this traditional service, it has two problems: First, the staff usually receive often repetitive questions asked by various customers, which a company can save a lot of money by using a conversational chatbot. Second, it is hard to have support open 24/7 especially for a known global organization.

Chatbots can handle multiple clients on the same platforms, humans can serve a limited number of customers at the same time but this does not concern for chatbots. With a chatbot, no customer is left unattended while solving different problems at the same time. But designing a chatbot can be complicated and complex and need a lot of time to understand what the customer wants. A fixed chatbot was created to help customers with frequent problems, but with an AI bot, it can understand the emotions of the customer and what they want. But AI bots should have decision making if they are not able to make a decision it can lead to bigger problems. For example, In 2016 Microsoft launched a chatbot for Twitter called Tay in less than 24 hours [21]. The twitter chatbot was an experiment for conversational understanding, and the more you chat with the bot it smarter it gets. The content is received from users turned out to be a racist conversational bot. It is important to set restrictions to a chatbot, and being well optimized so it can avoid what happened to Microsoft.

6 Conclusion

Fixed chatbots and artificial intelligence chatbots are becoming popular for web services and systems. In this paper, it was proposed and implemented a fixed interaction chatbot for a bank, which shows what kind of benefits it can provide and what needs to be considered. Some of the benefits that an Artificial intelligence chatbot it can adapt to different situation and keep learning and becoming better, while a fixed chatbot is very limited. As future work, it can be implemented artificial intelligence into the chatbot to improve customer experience.

References

1. J. Weizenbaum, "Eliza—a computer program for the study of natural language communication between man and machine," *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.
2. K. M. Colby, "Human-computer conversation in a cognitive therapy program," in *Machine conversations*. Springer, 1999, pp. 9–19.
3. P. Gentsch, *AI in Marketing, Sales and Service: How Marketers without a Data Science Degree Can Use AI, Big Data and Bots*, 1st ed. Palgrave Macmillan, 2018.
4.
5. S. A. Abdul-Kader and J. Woods, "Survey on chatbot design techniques in speech conversation systems," *International Journal of Advanced Computer Science and Applications*, vol. 6, no. 7, 2015.
6. C. I. Nass and S. Brave, *Wired for speech: How voice activates and advances the human-computer relationship*. MIT press Cambridge, MA, 2005.
7. Y.-P. Yang, "An innovative distributed speech recognition platform for portable, personalized and humanized wireless devices," in *International Journal of Computational Linguistics & Chinese Language Processing, Volume 9, Number 2, August 2004: Special Issue on New Trends of Speech and Language Processing*, 2004, pp. 77–94.

8. M. Nuruzzaman and O. K. Hussain, "A survey on chatbot implementation in customer service industry through deep neural networks," in *2018 IEEE 15th International Conference on e-Business Engineering (ICEBE)*, 2018, pp. 54–61.
9. S. Barker *et al.*, "How chatbots help," *MHD Supply Chain Solutions*, vol. 47, no. 3, p. 30, 2017.
10. V. Ilievski, C. Musat, A. Hossmann, and M. Baeriswyl, "Goal-oriented chatbot dialog management bootstrapping with transfer learning," *arXiv preprint arXiv:1802.00500*, 2018.
11. J. Huang, M. Zhou, and D. Yang, "Extracting chatbot knowledge from online discussion forums." in *IJCAI*, vol. 7, 2007, pp. 423–428.
12. T. Okuda and S. Shoda, "Ai-based chatbot service for financial industry," *Fujitsu Scientific and Technical Journal*, vol. 54, no. 2, pp. 4–8, 2018.
13. (2020, April). [Online]. Available: https://dialogflow.com/
14. A. Xu, Z. Liu, Y. Guo, V. Sinha, and R. Akkiraju, "A new chatbot for customer service on social media," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, ser. CHI '17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 3506–3510. [Online]. Available: https://doi.org/10.1145/3025453.3025496
15. J. Nielsen, "State of the media: The social media report," *Nielsen Company. http://blog. nielsen. com/nielsenwire/social*, 2011.
16. (2020, April). [Online]. Available: https://blog.hubspot.com/marketing/twitter-response-time-data
17. L. Cui, S. Huang, F. Wei, C. Tan, C. Duan, and M. Zhou, "Superagent: A customer service chatbot for e-commerce websites," in *Proceedings of ACL 2017, System Demonstrations*, 2017, pp. 97–102.
18. (2020, April). [Online]. Available: https://promo.bankofamerica.com/erica/
19. (2020, April). [Online]. Available: https://docs.microsoft.com/en-us/azure/bot-service/nodejs/bot-builder-nodejs-dialog-overview?view=azure-bot-service-3.0
20. (2020, April). [Online]. Available: https://dev.botframework.com/
21. (2020, April). [Online]. Available: https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist