A Psychologist Chatbot Developing Experience

Abubakr Siddig, Andrew Hines

School of Computer Science, University College Dublin, Ireland
{abubakr.siddig,andrew.hines}@ucd.ie

Abstract. Chatbots are computer programmes that mimic human conversation to interact with users through a variety of messaging channels. They are now regularly deployed on e-commerce and business websites providing customer support. Chatbots have also been employed for research and clinical support in the healthcare domain. In the field of psychology, chatbots have been applied to clinical research where survey or interview data collection are substituted with chatbots that can interact with the subjects via phone messaging apps in a non-clinical setting. This paper examines the design and development of a chatbot for a clinical psychology research study. The stakeholders, functionality, perspectives and technical challenges are presented and discussed. We apply a quality of experience framework to explore the factors that impact stakeholders and influence design priories. We present our conclusions regarding the leveraging cloud platforms and the technical customisation required for non-standard chatbot use cases.

Keywords: chatbot, psychology, dialogflow, QoE

1 Introduction

A conversational chatbot is a computer programme that tries to mimic a conversation with a real person. State-of-the-art chatbots rely on underlying Artificial Intelligence (AI) and Natural Language Processing (NLP) technologies to deliver a natural conversation experience [3]. In general, chatbots can be classified into two categories: task-oriented and conversational chatbots. Task-oriented chatbots operate within the boundaries of a pre-defined set of rules and are limited in their scope of activity. These are as simple as frequently asked question (FAQ) chatbots with a limited number of responses with basic pattern matching.

Conversational chatbots rely on NLP to extract information from the users' text and then react with a highly matched response. Many also use AI to improve the accuracy of the response over time, e.g. virtual agent chatbots.

From a user experience perspective, a general conversational chatbot that naturally handle various scenarios is desirable. In practice, however, chatbots are constrained in their design to specific purposes [18]. Nevertheless, the capacity of chatbots to interpret sentences are required to be free from users input and rely on NLP to parse inputs, extracting and interpreting salient information. NLP can be grouped into two classes: statistical-based methods and rule-based methods.

The statistical-based methods use application-specific text corpora to train statistical Machine Learning (ML) based NLP systems. These methods can identify and parse language structures, grammar and phrases to interpret sentences. The rule-based methods, on the other hand, use a predefined set of rules such as WordNet (electronic lexical database for English [13]) for NLP [2].

Chatbots have been widely adopted for business applications such as web shopping helpers, hotel reservation agents, and FAQ agents. The applications of these chatbots share a general conceptual design with chatbots designed for the healthcare domain including common stakeholders and the use of NLP. In practice, chatbots have four key stakeholders as defined in Table 1 along with domain-specific examples.

Table 1. Chatbot stakeholders

Role	Web Shopping Helper	Healthcare
End user (the people interacting with the chatbots)	Online shopper	Patient
Developer (creators of chatbots)	Software engineer	Researcher
Principal (the people who the chatbots are working on behalf of)	Store owner	Psychologist
Agent (people who are acting on behalf of the principal)	3rd party customer service agent	Graduate student

In this research, we reflect on the Quality of Experience (QoE) for the stakeholders involved with chatbots. QoE is defined as the degree of delight or annoyance of the user of an application or service [1]. The QoE framework evolved within the multimedia technology service community to help highlight the four influence factors beyond the service itself namely: Content, Context, System/Service and Human factors. As a framework, it has been applied beyond traditional multimedia applications to understand QoE factors for stakeholders [17, 9]. In the case of chatbots, the content factor refers to the topic of a conversation. The context factor refers to the use case and the theme of the conversation, such as small talk, educational, early intervention. The system/service factors include technical aspects relating to the messaging platform e.g, Facebook Messenger (Messenger), Skype, chat app etc. Finally, the human influence factors refer to the age of the user and the personal expectation from the chatting process.

From a QoE perspective, we can define the QoE for each stakeholder of a healthcare chatbot as follows:

– **End user:** The satisfaction level for the patient (study participant) will be proportional to how the chatbot responses to irrelevant inputs by users. User

experience (UX) also affects the overall QoE for the end-user such as the use of buttons and dropdown menus. QoE is affected by the ease of engagement as well, e.g using Messenger rather than a custom app to be downloaded separately.
– **Psychologist:** The chatbot will result in the same information being collected as would result from questionnaires/ clinical interviews but with less effort and a more natural environment for subjects to provide more reliable answers and engagement. All results are captured and stored electronically so they can be analysed and studied easily.
– **Developer:** The ease of deployment and integration to multiple messaging channels while maintaining the desired level of functionality.
– **Agent:** The graduate student role has changed from clinical data capture via questionnaire to software development. Their QoE will result from the ease of system development and data presentation.

1.1 Available platforms

Developing of a chatbot platform requires software design, programming skills and knowledge of related fields such as NLP, ML and AI. However, several chatbot cloud platforms have been created in recent years, such as Google dialogflow [5], Microsoft bot framework [12] and IBM Watson conversation [10]. These platforms hide the underlying technologies and integrate the NLP, ML and AI in the background.

A typical user's message would be: `If I want good friendship, I should be honest`, in such a case, the chatbot should be able to do a couple of things, first: what is the best reply? i.e. identifying the user's intention, second: is there any keyword? in this example, `''honest''` is a value that we want to know, and third: do we need to keep any or all of the information for further processing? Using the underlying technologies, these platforms perform these steps for developers.

Moreover, these platforms allow chatbot designers to focus on the tasks and interactions for businesses. However, the platforms target applications where the chatbot conversations are task-oriented, e.g. FAQ, and expect the user to ask a specific question, e.g. `What is the weather forecast today?`. This paper examines how such platforms can be used to develop chatbots for a non-task oriented conversation in the healthcare domain.

This paper reflects on the authors' experience while implementing a chatbot using these platforms, the paper does not investigate thoroughly these platforms nor conduct a comparison among them.

1.2 Applications of Chatbots

Chatbots are being deployed for various research purposes. H. Lo and C. Lee [11] showed that among 583 academic literature on chatbots, 497 are in the domain of computer science, the majority of which focus on a technical point of view and not the perspectives of the application and end-users.

The use of chatbots for psychology purposes has been an active area of study [15][16]. Woebot is an example of a chatbot that was developed for mental health support, specifically in cognitive behavioural therapy (CBT) [4]. Those chatbots usually follow a scripted text. In this domain, the chatbots offer close monitoring and early intervention, if needed. A recent study by Adam Palanica et al. [15] showed that more than 70% of physicians think that chatbots can be used for health consultations. Of concern, 70% of the study participants believed chatbots pose potential risks such as lack of empathy or wrong diagnoses. These findings highlight the need for further research exploring how chatbots can be effectively used in the healthcare field.

Robert Morris et al. [14] discussed the implementation of a chatbot that shows empathy in its responses. Their implementation relies on a post-response format and not a continuous conversation. It uses a text-matching technique to find a possible response within the training corpus. Only 50% of the participants rated the responses as good. This observation highlights low end-user QoE as current chatbot conversations do not yet provide corresponding levels of empathy human-human interactions.

Other researchers are also exploring the application of chatbots to non-traditional use cases, but without a significant focus on the issues surrounding chatbot design and the collections of feedback from chatbot users. Zhou et al. [19] studied the user experience for an interview chatbot. They analyse the questions and responses collected from the participants and the ability of the chatbot to follow the conversation. They conclude that targeted NLP capabilities that suit the purpose of the chatbot are an important consideration.

1.3 Motivation

As discussed earlier, chatbots can generally be classed as task-oriented or conversational. However hybrid chatbots such that it is task-oriented with some small talk conversational capabilities can also be created to fulfil a specific purpose [18]. Nowadays, chatbots are being used for business purposes [3] such as for FAQs and this is the main use case that cloud-based chatbot offerings anticipate. Other uses, e.g. the Woebot [4] for cognitive behavioural therapy do not fit within the question and answer based paradigm. Chatbots for healthcare often require a more of a free discussion within guidelines for an empathy-based user QoE.

This research focuses on the design challenges faced while developing a chatbot for healthcare purposes. The use case was to design and implement an intervention chatbot for psychological studies, named Plybot. Plybot is an intervention informed by Relational Frame Theory which seeks to reduce instances of problematic rule-following. Specifically, Plybot targets generalized pliancy wherein people adhere to rules just because they believe they should. Through conversing with Plybot, users examine why they follow their rules and whether or not these rules are useful for them. This paper is motivated by our need to examine the design factors in developing a psychology chatbot that follows a certain conversational intervention dialogue. We looked into the problem from the QoE perspective of chatbot developers as well as the other chatbot application stakeholders. Plybot, the chatbot used as the case study, was developed in collaborations with researchers at UCD School of Psychology to conduct a clinical research study. The study aims to understand the potential for a chatbot to substitute routine visits to a psychologist while maintaining the same level of integrity provided to the patients. Ethical and clinical issues will be discussed in future publications.

In this paper, we discuss the design, implementation and challenges faced in delivering a functioning psychological self-help chatbot. Unlike previously implemented CBT chatbots such as woebot, this paper discusses the technical and design perspective. We focus on the design considerations and development challenges in implementing a chatbot using a cloud platform where the NLP and ML subsystems are abstracted from the developer.

2 Architecture

This section describes the architecture of the "Plybot" chatbot and discusses the important decisions addressed.

2.1 Plybot design requirements

The Plybot design requirements can be views in contrast to a business-oriented chatbot such as an online shopping helper bot. Table 2 presented the main features required by Plybot and a shopping helper chatbot.

Table 2. Features required by Plybot vs shopping helper

Feature	Plybot	Shopping Helper
Persist the state of the user between messages over daily sessions	Yes	No
Interaction variables - referring back to previous days messages etc.	Yes	No
Easy to use service (Messenger/Skype), no install required by users	Yes	Yes
Initiate the next session at specific time	Yes	No
Data logging, all conversation and specific parameters	Yes	No

This table highlights the technical challenges to be addressed in developing Plybot compared to a simple FAQ-style chatbot that helps people with online shopping queries. Plybot requires the use of previous data into new conversations. It should also have the capability to initiate the conversations (a mandatory requirement). Unlike task-oriented chatbots, plybot should be able to continue the conversation from where it stopped, therefore a persistent user state should be kept even within the same daily session. These requirements make healthcare chatbots in general more demanding than business-oriented chatbots.

2.2 Why use an off-the-shelf cloud platform?

A cloud-based solution such as dialogflow, Microsoft Bot Framework and IBM Watson are cloud-based chatbot platforms. They provide some advantages over custom-built software solutions, namely:

- Customizable dialogue: simple user interfaces to create and to modify the dialogue
- Logging: access to data in an easy way for analysis by non-technical users
- Security and Ethics: privacy of individual users where the data is protected by the policies of the service provider
- Cost: low volume expected, low budget wanted. Cloud platforms pricing models target high volume solutions and are economically attractive for low volume applications.
- Scalability: the system can scale on-demand, and grow with the need.
- Ease of use/reuse: the implementation can adapt and redeployed for future activities

However, these systems bring their challenges, for instance, unlike Microsoft Bot Framework, dialogflow does not support proactive messages (a message to be sent to initiate a conversation or continue a dialogue with a user after a significant pause, e.g. a day later). Yet, dialogflow offers the majority key features required for a chatbot. One of the most important features of cloud-based systems is the easiness of integration with messaging platforms such as Skype and Messenger. Dialogflow allows integrating the communications channels conveniently and flexibly. For this case study, we chose dialogflow having also explored and experimented with the capabilities of Microsoft Bot Framework. We did not investigate the use of IBM Watson conversation for this case study.

2.3 Dialogflow

Figure 1 shows a typical communication flowchart for the chatbot implemented in dialogflow as experienced in our case study.

Chatbots platforms, including dialogflow, use a common terminology relating to dialogue management. The terms are introduced and defined below and summarised in Figure 1.

- **Intent:** is what the idea or message that the users want to convey to the chatbot. Based on interpreting a user input the chatbot sends a reply or performs an action. For instance, a user may indicate a wish to pause a conversation and continue again at a defined time. In such an instance, the chatbot should reply with a confirmation if the time is valid, or ask for valid input otherwise.

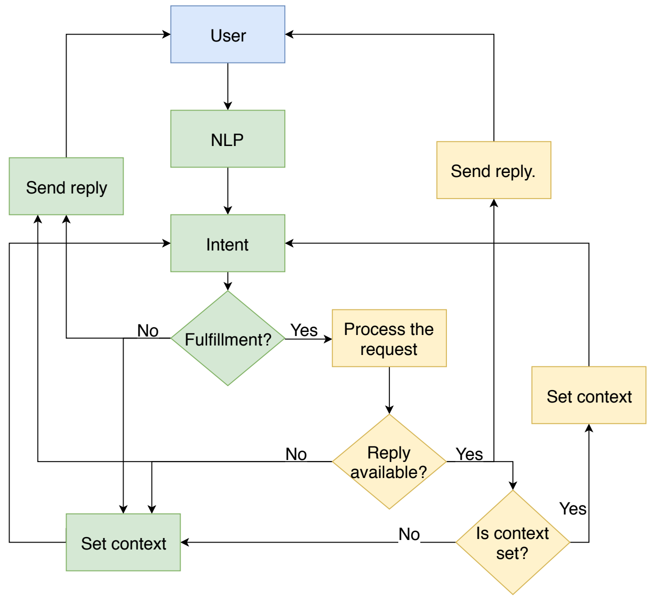

Fig. 1. Communication Flowchart in Dialogflow. The light blue box is user interaction through the messaging platform, light green boxes are dialogflow managed actions and light yellow boxes are custom implemented functionality.

- **Entity:** is a piece of information that has a specific meaning within a user utterance. For example, the user may write `Let's chat at 12:00`, in this case the time, `12:00`, is an entity.
- **Context:** is the glue that holds multiple intents together. It is used to put them within the same conversational space. The intents within the same context are matched first before searching all other intents. For instance, the same keywords (yes/no) can be used multiple times, but the intent is chosen based on the relevant context. Figure 2 shows that for the same response `Yeah` two different replies can be sent to the user based on the context: Jokes or Rules.
- **Fulfilment:** is used when further processing is needed to provide a function other than a simple reply message, for example, accessing a database.

3 Implementation and Challenges

This section describes the details about how Plybot is designed and implemented.

3.1 The messaging platform

As discussed in Section 2.1, one of the main features required by Plybot is the easiness of interacting with the chatbot application. The requirement was that

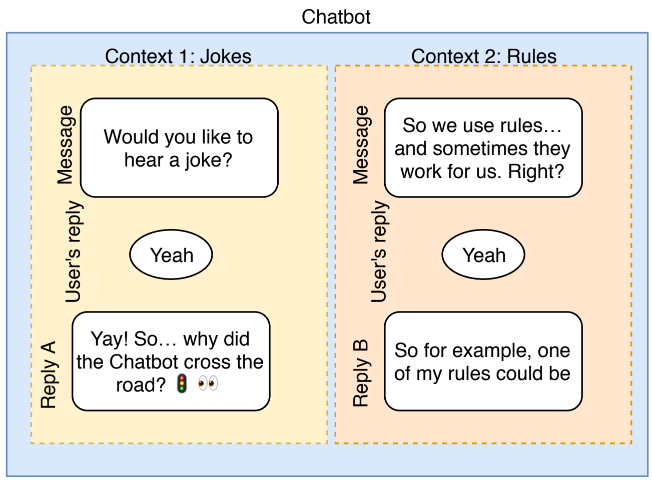

Fig. 2. Context Explained

users barriers to engagement should be minimised, specifically, no need for new app installation on user phones. Skype, Messenger and Slack were the three top choices considered for the chatbot. Slack and Messenger have rich APIs compared to that of Skype. Messenger is widely used in everyday life, unlike Slack which is more of a business-oriented messaging platform. Moreover, healthcare chatbot applications require privacy and security level not provided via Slack as it is based on channels where each user can see all other users. For data privacy, Skype does not send any information about the user_id with message replies, making data logging and retrieval more challenging. However, there is a unique part of the id field for every user, e.g.,

`"id": "4074b0s6-5871-4638-907c-80fcfe8461b3-283dg5df"`

. The last 8 digits are unique for each user and do not change over from day to day. On the other hand, using Messenger is more straightforward as the Facebook Page ID (PID), which is unique for every user, is usually sent with every message, simplifying the implementation when using the Messenger API. This PID is labelled as

`"facebook_sender_id": "4050067363115363"`

which is used to communicate directly with the user using Messenger API and access credentials. For these reasons, Facebook Messenger was selected as the messenger platform for this case study.

3.2 System Architecture

Figure 3 shows the sequence diagram for Plybot.

This architecture uses Messenger as the messaging platform as discussed in Section 3.1. The architecture contains these key components:

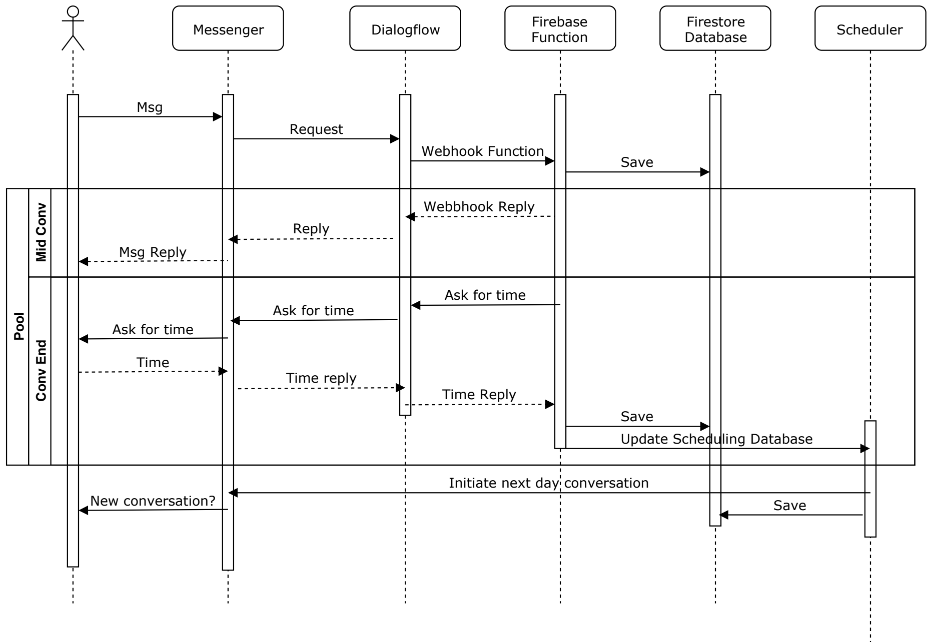

Fig. 3. Sequence Diagram

- **Dialogflow** is the core component of the implementation, where all the NLP, intent classification, and entity extraction algorithms are taking place.
- **Firebase** is where functions are implemented based on the requirement for every single intent, for instance accessing a database to write or retrieve data. Also, these functions are used for input verification, for instance, the rating of an action should take an integer value between 1 - 5. The functions are deployed in Firebase [6].
- **Database** is implemented in Firestore [7], a cloud-based NoSQL.
- **Scheduler Service** is a time-based job scheduler service called cron, responsible for scheduling the time to send the next daily message to the participants at their selected times to engage in a followup conversation.
- **Messenger** is the selected messaging platform where a user interacts with the Plybot.

3.3 Challenges

Most of chatbots cloud platforms, including dialogflow, are business-oriented and task-oriented, where the user is asking for specific information such as `what will tomorrow's weather be?` This restricts the chatbot to only being responsive to users queries.

Our chatbot, Plybot needs to re-initiate a conversation session each day continuing from the previous session. Only Microsoft Bot Framework provides what is called proactive messages which can be sent based on a specific event, for instance, a change in the price of a specific commodity.

This feature is not implemented in dialogflow but the use of the Messenger API provides a solution where cron jobs are meant to send messages using the API at users specified times. These messages are constructed from context and information captured in the previous conversation session and therefore a continued conversation flow is possible.

Another important consideration for chatbot developers is testing their solutions. Although both dialogflow and Microsoft Bot Framework provide a cloud function deployments, relying on cloud deployment for testing a fulfilment function is time-consuming as every deployment action can take a minute or more. In the developing stage, developers need the agility to test any change rapidly. The ability to perform tests using a local server with a public IP address is useful for development and testing.

Inconsistent quality in the cloud service platform API documentation slows development progress. For instance, Microsoft bot framework does not provide examples for integration with other platforms.

Cloud platforms have focused significant effort on increasing security and privacy. As a result, the privacy policies and system security parameters have regularly changed since the platforms were introduced. For instance, the user_id was a standard field in the dialogflow platform until it was deprecated on 30 June 2019, and after that date, the user needs to sign in using a Google account through Google Assistance before the user id can be retrieved [8]. Security and privacy was also an issue with Skype as discussed in Section 3.1. Increased security comes at the cost of flexibility and functionality meaning there is a danger that cloud-based solutions could have key functions or features restricted or compromised through future security informed policy changes.

At the beginning of a conversation, dialogflow associate a `session_id` that will remain active for a period of less than 20 minutes of inactivity. In applications such as Plybot, we should expect the users to discontinue the conversation and resume at their convenience. While this behaviour is not inherently supported by dialogflow, it was overcome by using a database to always keep track of the latest conversation stage to start from.

4 Summary of Lessons Learned

We successfully implemented Plybot, a chatbot that satisfies the requirements for a clinical psychology-based research study as discussed in Section 2.1. It can act as a substitute for the practitioner in specific tasks that are routinely in its nature, using a business-oriented cloud platform. A complete solution is implemented and deployed using dialogflow, firebase, firestore and Messenger. Although platforms like dialogflow are not designed for domains such as healthcare, but it can still be used to achieve the requirements of this domain.

However, the cloud-based implementation means that our solution is at the mercy of the underlying platform for key features including proactive messaging and scheduling. Additionally, the rapid development cycle for the platforms make the implementation of a chatbot volatile and API changes may require updates to keep the Plybot solution working as backwards compatibility is not guaranteed.

From the QoE perspective, we present key observations that will be validated as a future work to be done along with the psychological study, except for the QoE for developer which is our subjective evaluation of the developing experience. These observations are as follows:

– **QoE for User:** the use of Messenger allows easy access but limits the richness of UX widgets. Such implementation requires the trade-off between ease of no app to install against a better and more feature-rich UX, e.g. the use of dropdown menus for faster data input. The ability to interact with a chatbot from your phone rather than having to attend a practitioner interview is also a significant experience improvement.
– **QoE for Psychologist:** it can be concluded that a chatbot implementation provides benefits in terms of structured and consistent data gathering with fewer resources and interaction requirements. Ultimately, the efficacy of the chatbot as an alternative to traditional methods will not be known until the user study is completed.
– **QoE for developer:** The initial benefits of choosing a cloud-based solution (e.g. security, Messenger integration, NLP and AI, deployment) need to be weighed carefully against the platform restrictions (e.g. proactive messages) and maturity (changing APIs and documentation).

In practice, systems convert voice to text before it is processed. However, the effect of using voice instead of text chatting can be addressed in further studies.

Future work will investigate enhancing the users' experience by adding elements that make the interaction more natural such as the typing indicator for a couple of seconds before the reply is sent. These types of enhancements may improve the sense of empathy. Dialogflow allows for only 5 seconds for a response to be received from a webhook function, and this limited time is not sufficient to show the typing indicator before a reply is sent.

Acknowledgement

This publication has emanated from research supported in part by a research grant from Science Foundation Ireland (SFI) and is co-funded under the European Regional Development Fund under Grant Number 13/RC/2289 and Grant Number SFI/12/RC/2077. Thanks to Dr Louise McHugh and Alison Stapleton from UCD School of Psychology for input into the chatbot requirements.

References

1. Callet, P.L., Moller, S., Perkis, A.E.: Qualinet White Paper on Definitions of Quality of Experience. European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003), Lausanne, Switzerland (March 2013)

2. Chowdhury, G.G.: Natural language processing. Annual review of information science and technology **37**(1), 51–89 (2003)
3. Delloite: Chatbots Point of View. Tech. rep., Deloitte Artificial Intelligence (March 2018)
4. Fitzpatrick, K.K., Darcy, A., Vierhile, M.: Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (woebot): a randomized controlled trial. JMIR mental health **4**(2), e19 (2017)
5. Google: Dialogflow, www.dialogflow.com, accessed on 30 Sept 2019
6. Google: Firebase, firebase.google.com, accessed on 30 Sept 2019
7. Google: Firestore, firebase.google.com/docs/firestore, accessed on 30 Sept 2019
8. Help, G.A.: userid is missing, https://support.google.com/assistant/thread/13625684?hl=en, accessed on 30 Sept 2019
9. Hines, A., Kelleher, J.D.: A framework for post-stroke quality of life prediction using structured prediction. 9th International Conference on Quality of Multimedia Experience (QoMEX) (2017)
10. IBM: Ibm watson, https://www.ibm.com/watson/how-to-build-a-chatbot, accessed on 30 Sept 2019
11. Io, H.N., Lee, C.B.: Chatbots and conversational agents: A bibliometric analysis. In: 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM). pp. 215–219. IEEE (dec 2017)
12. Microsoft: Microsoft bot framework, https://dev.botframework.com/, accessed on 30 Sept 2019
13. Miller, G.A.: Wordnet: a lexical database for english. Communications of the ACM **38**(11), 39–41 (1995)
14. Morris, R.R., Kouddous, K., Kshirsagar, R., Stephen, B.., Schueller, M.: Towards an Artificially Empathic Conversational Agent for Mental Health Applications: System Design and User Perceptions. Journal of Medical Internet Research **20**(6) (2018)
15. Palanica, A., Flaschner, P., Thommandram, A., Li, M., Fossat, Y.: Physicians' Perceptions of Chatbots in Health Care: Cross-Sectional Web-Based Survey. Journal of Medical Internet Research **21**(4) (2019)
16. Park, S., Choi, J., Lee, S., Oh, C., Kim, C., La, S., Lee, J., Suh, B.: Designing a Chatbot for a Brief Motivational Interview on Stress Management: Qualitative Case Study. Journal of Medical Internet Research **21**(4) (2019). https://doi.org/10.2196/12231, https://www.jmir.org/2019/4/e12231/
17. Ragano, A., Benetos, E., Hines, A.: Adapting the quality of experience framework for audio archive evaluation. 11th International Conference on Quality of Multimedia Experience (QoMEX), IEEE (2019)
18. Toxtli, C., Monroy-Hernández, A., Cranshaw, J.: Understanding Chatbot-mediated Task Management. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. pp. 1–6. ACM Press, New York, New York, USA (2018)
19. Zhou, M.X., Wang, C., Mark, G., Yang, H., Xu, K.: Building Real-World Chatbot Interviewers: Lessons from a Wizard-of-Oz Field Study ACM Reference format. In: Proceedings of the IUI Workshop. pp. 1–6. ACM Press (March 2019), http://ceur-ws.org/Vol-2327/IUI19WS-USER2AGENT-2.pdf