A voicebot for social and psychological studies

Georg Friedrich Eduard Kruse
Technische Universität Berlin
Berlin, Germany
g.kruse@campus.tu-berlin.de

Joel John
Technische Universität Berlin
Berlin, Germany
joel.john@campus.tu-berlin.de

Wolf Siegfried Rieder
Technische Universität Berlin
Berlin, Germany
w.rieder@tu-berlin.de

***Abstract*—In the last decade artificial intelligence (AI) has greatly evolved due to the combination of neural networks and increased computational power. This has led to a variety of new AI applications like chat- and voicebots. Even though such applications have become increasingly visible, the moral impacts they have on humans are relatively unknown. The analysis of these impacts is an emerging research field in sociology and psychology. Researchers therefore need easy-to-use chat- and voicebots to carry out their experiments. The frameworks and applications needed for such bots should be open-source and free to use under a public licence so that the approaches can be shared freely among researchers and be further developed.**

In this paper, we develop a chatbot with voice support using only open-source frameworks with public licences. It is used in a social experiment about moral decisions to prove its applicability to real world experiments. Our framework is capable of detecting and processing speech in an online manner with a latency below one second.

***Keywords*—chatbot, AI, morality**

I. Introduction

AI has evolved a lot in the last few decades and has become an indivisible part of our lives - ranging from virtual assistants to humanoid robots. AI has helped humans in making our work easier. It has immense applications in various fields. We use AI applications in our everyday lives - from setting up meetings, navigation, shopping, asking questions, learning, etc.

Because of daily interactions, the voice assistants even have an impact on our attitudes and behaviours [1]. Since AI technologies are constantly evolving, in the future we will have deeper interactions with AI and many of our day-to-day decisions might be automated with minimal intervention from us [2].

Even though we interact with AI on a day-to-day basis, the moral impacts the AI applications have over us are widely unknown. The question remains - what are the moral implications AI has over us. Researches are going on to answer this question. This is one of the emerging research fields in sociology and psychology [3]. But in order to conduct effective researches, the researchers need an easy-to-use method to evaluate the impact AI applications have on human morality. One of the simplest and effective methods for such an evaluation is with the use of chatbots. Chatbots are one of the predominant AI applications that we use. Chatbots can behave human-like, are easy to reprogram, and provides instant feedback. A chatbot with voice support is known as ”voicebot”. Although a voicebot has shortcomings like perceiving empathy and accents, they are much more interactive and are gaining popularity [4] [5]. In this project, we are developing a chatbot with voice support that will help researchers evaluate the moral impact AI has over humans.

The major constraints in making an efficient chatbot for such research purposes are licensing issues, privacy concerns, and the need for a suitable method of evaluation. For efficient research, researchers need easy-to-use open-source chatbots. An open-sourced platform allows the researchers to intervene in most aspects of the chatbot implementation [6]. Since of the current state-of-the-art chatbots are closed-source and cloud-based, a major concern with those are the lack of proper security and privacy [7]. Especially when the purpose of this chatbot is for social and psychological studies, the privacy of the participants is an extremely important factor. Studies have shown that information disclosure will be more when the chatbot behaves more human-like [8].

In order to steer clear of licensing issues and privacy concerns, we are using open-sourced frameworks for the implementation of the chatbot as well as for the voice integration. And as for the method of evaluation, human participants will engage in an economic decision game with the voicebot. By analyzing the results, researchers can evaluate how much the bot can influence the decisions of the participants by providing them with varying information.

II. Related Work

A preliminary Web of Science search with the terms ”Chatbot” and ”Conversational Agent” results in 1435 publications over the past decades (Fig. 1). However, 768 publications were released since 2019 and indicates that this topic is becoming more relevant than ever before.

Therefore, we will take a more detailed look at past research efforts. This section starts with an overview of this topic, i.e. the past, the current state of research, and future directions. Then, developed frameworks for classifying or evaluating chatbots will be discussed. Finally, relevant publications at the intersection of chatbots and social or psychological research as well as user studies are presented.

The first known chatbot in history was developed by Joseph Weizenbaum in 1966 [9]. ELIZA was able to identify certain keywords and respond based on predefined rules. However, the chatbot had its limitations as it was only used in the context of psychotherapy and all keywords had to be included in a predefined dictionary.

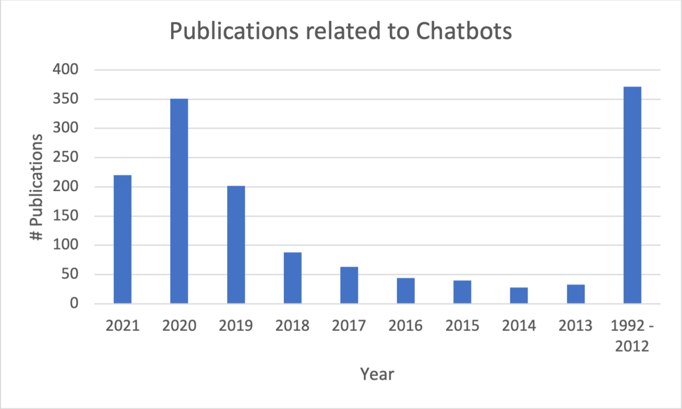

Fig. 1. Web of Science search results regarding publications.

PARRY was the next well known chatbot, introduced by Colby et al. [10], simulating a patient with schizophrenia. In 1995, the chatbot A.L.I.C.E. was created by Wallace and further improved in the following years [11]. It utilized pattern-matching and 41 000 categories to respond appropriately to the user [11]. The next century brought to light many sophisticated chatbots thanks to the use of machine learning (ML) and natural language processing (NLP). First and foremost, programs like Siri, Google Assistant, or Watson, followed by the rise of many tools and platforms to develop chatbots for various use-cases.

The literature offers several definitions of a chatbot, Dale [12, p. 813] defined them as: “any software application that engages in a dialog with a human using natural language.” and is similar to others with the extra notion of AI [13].

Hussain et al. [13] used four criteria for the classification: (i) interaction mode, which refers to the available modalities, (ii) goals, i.e. are they designed for specific tasks (task-oriented) or lengthy conversations without a certain goal (non-task oriented), (iii) knowledge domain, i.e. the range of knowledge, and (iv) design approach.

Other researchers examined these criteria more in-depth, for example Almansor et al. [14] divided each goal into two subitems. Task-oriented can be further split into supervised and unsupervised approaches, similar to the ML terms. Non-task-oriented can be distinguished between retrieval-based and generation-based chatbots.

Mnasri et al. [15] divided the design approach into five different architectures, namely rule-based, data-driven, information retrieval based, ML, and hybrid. However, all have the emphasis of ML models and learning approaches in common, from deep learning, reinforcement learning, (recurrent) neural networks to the sequence-to-sequence algorithm. Nevertheless, it might be better to follow a hybrid approach as each model has its advantages and disadvantages. In addition, the author suggests separating the approach from the chatbot engine. Lastly, it is no surprise that ML is the driving force behind today’s chatbot systems, but on the other hand it is more dependent on advances in this area and others like NLP. However, even with these advanced approaches, the highly complex nature of natural language cannot yet be fully solved, which limits the ability of chatbots [16].

Radziwill et al. [17] examined 42 publications from industry and academic sources to analyze the quality attributes of chatbots and CAs. They first identified three top-level categories, namely efficiency, effectiveness, and satisfaction, which were further subdivided into categories measured by different quality attributes, e.g., robustness or accurate speech synthesis. In addition, they used the Analytic Hierarchy Process (AHP) to evaluate quality in terms of performance, humanity, affect, and accessibility to compare different chatbot systems. Interestingly, this method can also be used to compare two versions of the same chatbot, and given the learning effects of these systems, it is vital to periodically assess quality over time.

Another framework by Parikari et al. [18] from 2018, identified six categories comprising two to four metrics. Each of them addresses either the chatbot itself, e.g., purpose and high-level functioning, or the interactions between a human and the chatbot, e.g., modalities offered and human similarity. The authors compared ten chatbots and surprisingly, all were simple, mostly informative, and used only text to communicate. However, due to the small sample size, it is not apparent how good and useful their framework might be.

A comprehensive framework was developed by Pérez-Soler et al. [19] in 2021. The authors focused mainly on technical attributes, 26 in total, and eight administrative attributes that take into account important developer considerations, such as open source or pricing. Their framework takes into account recent advances and requirements of today’s chatbots, such as the use of NLP, intents, or sentiment analysis. In contrast to the work of Parikari et al. [18], fourteen prominent chatbots were selected that have much more sophisticated functionality. However, the human-friendliness or actual performance of the chatbot does not matter here.

The framework developed by Braun et al. [20] attempts to address multiple stakeholders and support their decision-making process. The framework includes six categories with two to five subdivisions and focuses mainly on interaction and how well a chatbot understands its users rather than technical features. Compared to other studies, they incorporated another category called Timing, which deals with the chatbot’s response behavior. Subsequently, they applied their framework to four realized chatbot systems and then to three tools, namely Rasa, Kaldi, and Chatfuel.

Finally, Janssen et al. [21] published a taxonomy comprising the three levels intelligence, interaction, and context. Each level is subdivided into dimensions, which in turn are subdivided into features. The latter level is of particular interest because it takes into account the chatbot’s operating environment, which has been measured only briefly in previous frameworks. The technical level, however, is not as detailed as the work of Pérez-Soler et al. [19] and is examined more broadly, such as whether it is rule-based. The authors analyzed 103 chatbots, more than any other previously mentioned study, and provided some interesting results. Most chatbots are rule-based, goal-oriented, reactive, and designed primarily to communicate with a single person. There are other predominant characteristics, but this taxonomy, combined with the larger sample size, provides a comprehensive view of the state of chatbots in 2020.

The use of CAs in studies as part of a game is not new. The interaction between an embodied CA and two users in a dice game was observed by Rehm et al [22] in 2005. The goal was to test this form of communication as well as the effects on the users themselves. Training the CA with video recordings enabled a more human-like interaction, and the participants not only showed similar emotional behavior to other humans, but also viewed the CA as competent. Matthias Rehm [23] published a more extensive follow-up study three years later that confirmed some of the earlier findings. Although participants interacted with the CA similarly to other people, they tended to behave differently, for example, by observing the CA's behavior more closely or discussing the CA with the other participant.

Völkle et al. [24] analyzed the acceptance of chatbots on multiple dimensions in three different scenarios. Their laboratory experiment with 30 participants revealed that the acceptance is higher in scenarios where the chatbot was used to solve simple problems or retrieve information. This was not observed in complex or emotional situations like health which might be due to the participants lack of trust in the chatbot's competence compared to a human.

The aspect of trust was further examined by Følstad et al. [26] in an interview study with thirteen users of customer service chatbots. Their results revealed that not only the chatbot itself contributes to the user's perceived trust, but also the context in which the chatbot is located, e.g. the chatbot provider or the perceived security, influence the trust level. In addition, human likeness, which is part of some classification systems, is related to trust. The performance of the chatbot, i.e., the responses and the ability to accurately recognize the user's request, is also relevant.

In 2020, Følstad et al. [25] conducted a questionnaire study with 207 participants from the United States that used chatbots in various scenarios, e.g. customer service or productivity. Positive experiences were summarized in seven categories and negative ones in six. Participants emphasized useful and efficient chatbots in order to achieve their goals. In addition, a chatbot with an entertaining way of interacting was highlighted as a particularly positive feature. On the other hand, interpretation issues, the inability to assist, or questions already asked were rated especially negatively, but less than half of the participants had something negative to say. How positively or negatively a chatbot is evaluated by the subscriber depends on the concrete use case, i.e. different types of chatbots have to meet different user expectations.

III. Technology Review

In this section, we will review the three main technologies involved in this project - the NLP framework, the speech-to-text (STT) engine and the text-to-speech (TTS) engine. We will discuss the open-source as well as the proprietary technologies available.

A. Natural Language Processing

Due to recent advances in machine learning and NLP, there are lot of approaches for building a state-of-the-art chatbot framework. But most of these frameworks are not yet capable of creating a truly conversational chatbot.

Currently there are no highly effective recommended systems for automated comparisons of chatbots. Therefore chatbot comparisons are often done by humans. For selecting the NLP framework for this project, our criteria involved - Input and output modalities, NLP type, pricing model, licensing and testing & deployment options.

Table I shows the comparison of the most popular NLP frameworks available today. It includes both the NLP platforms as well as chatbot frameworks. The chatbots offered by most of the cloud-based platforms have a paid pricing model with the exception of Facebook's wit.ai which offers the service free of charge even for commercial use. While the cloud-based platforms provide better features, they are mostly closed-source. Open-source chatbot frameworks provide greater freedom of development and guarantee better privacy.

B. Speech-To-Text Engine

The system which is used to recognize and transcribe spoken language to text is known as automatic speech recognition (ASR) or STT engine. There are various STT tools available today. The major metrics that are useful for evaluating these tools are word error rate (WER), Real Time Factor (RTF), and Type. Table II shows the comparison of major speech to text engines.

The Word Error Rate is the ratio of Levenstein distance between words in a reference transcript and words in the output of the STT engine. The word error rate of a human transcriptionist is considered to be 4%, while most STT engines provide a WER between 6% to 25% [27].

The real time factor (RTF) is the ratio of processing time to the length of the input speech file. A lower RTF means the engine is computationally more efficient.

Mainly there are two different types of speech recognition systems - offline, on-device systems as well as cloud based systems. Although cloud-based speech recognition systems provide better WER and RTF compared to offline systems, they are mostly proprietary. Offline systems are better suited for situations where privacy and open source public licences are important.

C. Text-To-Speech Engine

The system which is used to artificially create human speech from text is known as speech synthesis system or text-to-speech (TTS) engine. Instead of playing recorded speech, a TTS engine generates speech using plain text as input [29].

TTS engine evaluations are usually done by humans, evaluating its ability to be understood clearly and its similarity to

	DialogueFlow	Watson	Rasa	Wit.ai	Botkit	Botpress	Lex	Microsoft Bot Framework
Company	Google	IBM	Rasa Gmbh	Facebook	Microsoft	Botpress	Amazon	Microsoft
Custom NLP Support	No	No	Yes	No	Yes	No	No	No
Hosting	Cloud	Cloud	On-Premise	Cloud	On-Premise	On-Premise	Cloud	Cloud
License Type	Proprietary	Proprietary	Open-Source	Proprietary	Open-Source	Open-Source	Proprietary	Open-Source
Pricing	Free & Paid	Free & Paid	Free	Free	Free	Free	Free & Paid	Free & Paid
Speech Support	Yes	Yes	No, But can Integrate Voice Platforms	Yes	No	No	Yes	Yes

TABLE I
COMPARISON OF MAJOR CHATBOTS

	WER	RTF	Hosting
Amazon Transcribe	8.21%	N/A	Cloud
Google Speech-to-Text	12.21%	N/A	Cloud
Mozilla DeepSpeech (0.6.1)	7.55%	0.46	On-Device
Picovoice Leopard (v1.0.0)	8.34%	0.46	On-Device

TABLE II
COMPARISON OF SPEECH-TO-TEXT ENGINES [28]

the human voice. The major metrics considered for these evaluations are - Speed, Quality, Clarity, and listening experience. While most of the TTS systems are on par with humans on Speed and Clarity metrics, they are lacking in the quality and listening experience metrics [30].

IV. APPROACH

For the implementation of this project, we use ”Rasa Open Source” as the NLP framework along with the open source models for STT and TTS from the Mozilla Deepspeech project. [32], [33], [34]. The main advantages Rasa has over its competitors are on-premise installation and open source licence. Even though Rasa provides additional functionalities over Rasa Open Source as ”Rasa X”, we refrained from using it since it was not open-source.

Mozilla Deepspeech is used as the voice engine because it an offline open-source and privacy-friendly framework. Mozilla Deepspeech is the tensorflow implementation of Baidu’s Deep Speech architecture [31]. Another advantage of Deepspeech is that it has a low word error rate even with the default, pre-trained model. Since both Mozilla Deepspeech and Rasa Open Source are written in Python, it allows easier integration.

The project consists of two parts - a SocketIO Flask server on which the voice engines are installed and a Rasa chatbot container. This modulized setup allows good treatability and adaptability for future development. For the delivery/deployment of the project, docker-compose is used.

A. System Architecture

Social experiments in the field of psychology and sociology are often conducted through web applications. Therefore the voicebot will be deployed as a web server and the participants will interact with it through their web browser.

In figure 2 the overall system architecture of the project can be seen. The python web framework Flask is used as a server, on which the voice processing is performed. It communicates with the clients through web sockets. This ensures a fast way to transfer the audio data from the participants web browser to the backend. The server then processes the audio data (see section IV-C) and sends the resulting test data to the rest api channel of the rasa chatbot. The chatbot then sends a text response back to the Flask server where it is turned into speech. The speech data is then sent through the websockets to the clients and played in the browser.

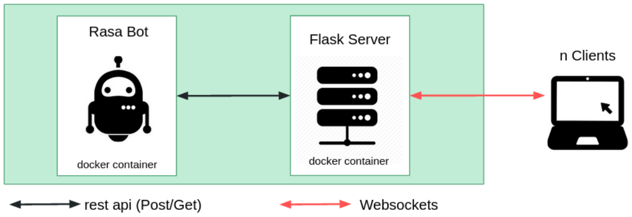

Fig. 2. Architecture - System.

B. Rasa Chatbot

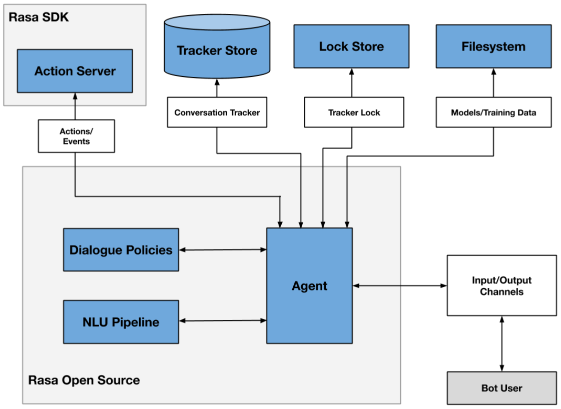

Fig. 3. Architecture - Rasa. [35]

Fig. 3 shows the architectural overview of Rasa Open Source. Since Rasa has a scalable design, it allows easier integration with other systems. The primary components involved are Natural Language Understanding (NLU) and dialogue management (Rasa Core).

NLU is responsible for intent classification, entity extraction and response retrieval, while dialogue management decides on the next action (conversation) to perform [35].

The above diagram (Fig. 4) shows the message handling inside Rasa Open Source. The messages a user sends are passed on to an interpreter (Rasa NLU) which converts and extracts the entities and intents within the message. Then interpreter passes this message to the tracker, which keeps

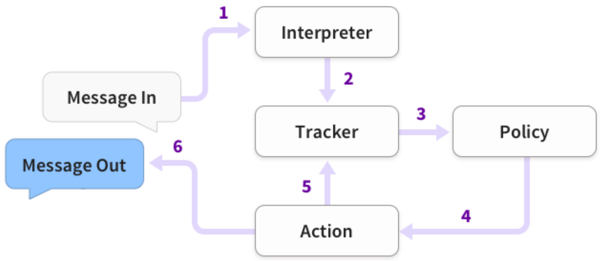

Fig. 4. Message Handling in Rasa. [36]

track of the conversation state and logs the actions that are taken. The current state of the tracker is sent to each policy, which in turn chooses the next action to take. And finally, the response is sent to the user [36].

C. Voice Integration

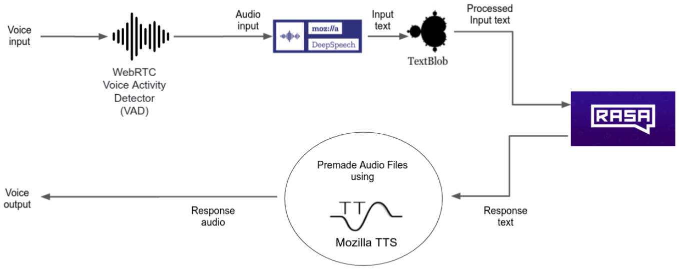

Fig. 5. Architecture - Voice [36]

Fig. 5 illustrates the functioning of the voice integration. The browser of the client activates the microphone of the device and continuously records the users audio input. This input is sent through websockets as audio blobs of 0.5 sec length to the flask server. There it is split into smaller audio frames of 20ms and analyzed by a WebRTC voice activity detector (VAD) with an aggressiveness of three in order to evaluate if the user is speaking. Then these classified frames are fed into an audio buffer. Once the VAD detects speech the audio is fed into a Deepspeech audio context stream until the VAD stops detecting speech. Then the audio context stream is analysed by the Deepspeech STT model and the output text is spell checked by the Textblob framework. The checked text data is then sent through a Post request to the Rasa chatbot container and an text answer is received. This answer is analysed in order to determine if previously processed voice files of the text answer are stored locally as .ogg files on the web server. If so, these files are sent through the websocket connection to the clients browser. If no voice files of the chatbots answer exist, the text data is fed into the Deepspeech TTS model and the newly created output voice file is sent to the client.

V. Evaluation

The major challenges in creating chatbots with voice integration are creating a conversation flow which seems human like and keeping the latency of the responses at a minimum. These capabilities are tested in an economic decision game, where the participants will interact with the voicebot before either investing in a ”green” or ”normal” investment fond. The participants will have to ask the voicebot about the task and the voicebot will try to influence their behavior. The main focus on the analysis will lie on the performance of the voice model and if the Rasa chatbot is capable of understanding imperfect input data.

The Rasa framework provides a state of the art NLU model combined with dialogue management. The NLU model shows good performance in understanding and classifying the users intents if the text input doesn’t show any spelling mistakes. The higher the input error rate is, the lower the resulting performance. Due to the relatively high error rate of the STT models, this can compromise the NLU model to an extent, at which it is no longer capable of correct intent classification. In order reduce such spelling mistakes, spell checking is performed on the text data before analyzing it with the NLU model.

The Deepspeech SST model has an official word error rate of 7.55%. The experiment shows, that this is only the case, if the participants speak slow and clearly in an almost inhuman manner. Once the participants start speaking fast and unclear or with a strong accent, the model only understands single words correctly. This leads to an inability in finding the users intents correctly. Since the amount of intents in the used NLU model is comparably low, the Rasa chatbot still manages to conduct the experiment, but it will struggle to do so for more complex tasks. New versions of the Rasa framework might be capable to handle imperfect input data better.

Since the Deepspeech project is under continuous development, newly developed models might soon be available, which reach ever lower word error rates. The modularity of the STT integration on the Flask server would allow quick exchanges of models. The exchange of the model will therefore be simple and fast, without making major changes of the overall architecture necessary.

The success of a voicebot framework mainly depends on a fast response creation. To achieve low latencies, mostly the STT and the TTS models need to be investigated, because they are mainly responsible for the processing time. The Rasa chatbot, the spell checking by Textblob as well as the audio data transfer through the websockets do not account for considerable time delays. The Deepspeech STT model need for an average sentence (e.g. ”Can you help me with my task?”) 0.26 seconds on an laptop. This is fast enough to be human like. The TTS on the other side has a real time factor 0.6. This means, that the processing of a phrase like ”Hello! I am Juila. I will help you today with your task.” takes 2.58 seconds. The processing time increases even more for longer phrases which are needed in the experiment to explain the task and answer the questions. Creating speech in an online fashion is therefore not feasible.

The processing time can be greatly reduces by generating the voicebot answers in advance before launching the experiments. As all the answers are carefully reviewed by the researchers before conducting the experiments, there is no need to generate voice live. The audio files are saved on the server and played on demand to the users, if they math the response of the Rasa bot. The response time can be kept below one second, which is reasonable.

VI. Conclusion

In this paper we introduced and open source voicebot framework which can be used under public licence. The simple communication structure through websockets and the rasa rest api makes it transferable to other server setups. Voice detection is added to make the communication with the voicebot more human like. The speech processing time is kept below one second by previously generating voice files for all Rasa chatbot answers.

Remaining obstacles for real world social experiments are the poor performance of the Deepspeech STT model on imperfect voice as well as the undercomplexity of the Rasa dialogue management. This leads to the failure of the Rasa stories. A better STT model would greatly improve the overall performance of the voicebot.

References

[1] Atieh Poushneh, ”Humanizing voice assistant: The impact of voice assistant personality on consumers’ attitudes and behaviors,” Journal of Retailing and Consumer Services, Volume 58, 2021.

[2] Janna Anderson, Lee Rainie and Alex Luchsinger, “Artificial Intelligence and the Future of Humans, Pew Research Center, December 2018.

[3] Morgan R. Frank. ”The evolution of AI research and the study of its social implications.” Medium.com. https://medium.com/mit-media-lab/the-evolution-of-ai-research-and-the-study-of-its-social-implications-4a9598b3d7db.A

[4] Santhosh, ”Chatbot Vs. Voicebot. Who’s the winner?.” Agra.ai. https://agara.ai/conversational-ai-blog/customer-service/chatbot-vs-voicebot-whos-the-winner/

[5] H. N. Io and C. B. Lee, ”Chatbots and conversational agents: A bibliometric analysis,” 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), 2017, pp. 215-219, doi: 10.1109/IEEM.2017.8289883.

[6] Adamopoulou E., Moussiades L. (2020) ”An Overview of Chatbot Technology.” Artificial Intelligence Applications and Innovations. AIAI 2020. IFIP Advances in Information and Communication Technology, vol 584. Springer, Cham. https://doi.org/10.1007/978-3-030-49186-4_31

[7] Barbara Ondrisek. ”Privacy and Data Security of Chatbots.” Medium.com. https://medium.com/@electrobabe/privacy-and-data-security-of-chatbots-6ab87773aadc

[8] Ischen C., Araujo T., Voorveld H., van Noort G., Smit E.(2020) ”Privacy Concerns in Chatbot Interactions.”Chatbot Research and Design. CONVERSATIONS 2019. Lecture Notes in Computer Science, vol 11970. Springer, Cham. https://doi.org/10.1007/978-3-030-39540-7_3

[9] Weizenbaum, J.. “ELIZA—a computer program for the study of natural language communication between man and machine.” Commun. ACM 9 (1966): 36-45.

[10] Colby, K., F. D. Hilf, S. Weber and H. Kraemer. “Turing-like Indistinguishability Tests for the Calidation of a Computer Simulation of Paranoid Processes.” Artif. Intell. 3 (1972): 199-221.

[11] Wallace, R.. “The Anatomy of A.L.I.C.E.” (2009).

[12] Dale, R.. “The return of the chatbots.” Natural Language Engineering 22 (2016): 811 - 817.

[13] Hussain, Shafquat, Omid Ameri Sianaki and N. Ababneh. “A Survey on Conversational Agents/Chatbots Classification and Design Techniques.” AINA Workshops (2019).

[14] Almansor, E. H. and F. Hussain. “Survey on Intelligent Chatbots: State-of-the-Art and Future Research Directions.” CISIS (2019).

[15] Mnasri, Maali. “Recent advances in conversational NLP : Towards the standardization of Chatbot building.” ArXiv abs/1903.09025 (2019): n. pag.

[16] Lokman, Abbas Saliimi and M. A. Ameedeen. “Modern chatbot systems: a technical review.” (2018).

[17] Radziwill, N. and Morgan C. Benton. “Evaluating Quality of Chatbots and Intelligent Conversational Agents.” ArXiv abs/1704.04579 (2017): n. pag.

[18] Paikari, Elahe and A. Hoek. “A Framework for Understanding Chatbots and Their Future.” 2018 IEEE/ACM 11th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE) (2018): 13-16.

[19] Pérez-Soler, Sara, Sandra Juarez-Puerta, E. Guerra and J. de Lara. “Choosing a Chatbot Development Tool.” IEEE Software 38 (2021): 94-103.

[20] Braun, Daniel and F. Matthes. “Towards a Framework for Classifying Chatbots.” ICEIS (2019).

[21] Janssen, A., Jens Passlick, Davinia Rodríguez Cardona and M. Breitner. “Virtual Assistance in Any Context.” Business & Information Systems Engineering 62 (2020): 211-225.

[22] Rehm, M. and M. Wissner. “Gamble - A Multiuser Game with an Embodied Conversational Agent.” ICEC (2005).

[23] Rehm, M.. “”She is just stupid” - Analyzing user-agent interactions in emotional game situations.” Interact. Comput. 20 (2008): 311-325.

[24] Völkle, Christiane and Patrick Planing. “Digital Automation of Customer Contact Processes – an Empirical Research on Customer Acceptance of different Chatbot Use-cases.” (2019).

[25] Følstad, A. and P. B. Brandtzaeg. “Users’ experiences with chatbots: findings from a questionnaire study.” Quality and User Experience 5 (2020): 1-14.

[26] Følstad, A., Cecilie Bertinussen Nordheim and C. Bjørkli. “What Makes Users Trust a Chatbot for Customer Service? An Exploratory Interview Study.” INSCI (2018).

[27] R. P. Lippmann, “Speech recognition by machines and humans”, Speech Communication, vol. 22, pp.1–15, 1997

[28] Speech-to-Text Benchmark. https://github.com/Picovoice/speech-to-text-benchmark

[29] J. O. Onaolapo, F. E. Idachaba, J. Badejo, T. Odu, and O. I. Adu , “A Simplified Overview of Text-To-Speech Synthesis”, 2014

[30] Julia Cambre, Jessica Colnago, Jim Maddock, Janice Tsai, and Jofish Kaye. 2020. ”Choice of Voices: A Large-Scale Evaluation of Text-to-Speech Voice Quality for Long-Form Content”. Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. Association for Computing Machinery, New York, NY, USA, 1–13. DOI:https://doi.org/10.1145/3313831.33767894

[31] Awni Y. Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, and Andrew Y. Ng. ”Deep speech: Scaling up end-to-end speech recognition.” 2014

[32] Rasa Open Source. https://rasa.com/docs/rasa/

[33] Mozilla Deepspeech. https://github.com/mozilla/DeepSpeech

[34] Mozilla TTS. https://github.com/mozilla/TTS

[35] Rasa Open Source Architecture. https://rasa.com/docs/rasa/arch-overview/

[36] Message Handling in Rasa. https://rasa.com/docs/rasa/architecture/