Chatbot in the Service Industry: Challenges and Perspectives

Haoyuan Wang

University of California, San Diego San Diego, United States

h8wang@ucsd.edu

Abstract. The chatbot is a software application that could initiate and participate in a meaningful conversation with a live human being agent. With proper algorithms and sufficient training, a chatbot could comprehend inquiries and replies in a way that mimics human response. The chatbot allows companies to operate at a low cost and provides more efficient and professional customer service. This article aims to provide a symmetrical overview of chatbot building and comments on frequently used models and mainly focuses on the models that researchers utilize to build chatbots. The models are separately discussed in terms of query understanding and response generation, and the knowledge base, evaluation, and optimization methods are also included. Chatbots building is a combination of Natural Language processes (NLP) and Machine Learning (ML), and there is a trend shifting from rule-based to more complex models involving Neural Networks.

Keywords: Chatbot, natural language processes, machine learning.

1. Introduction

Chatbot has more than 70 years of development, and it is gradually evolving into a more intelligent form with the ability to perform complex tasks. Currently, three main categories of chatbots form the mainstream of the industry. The first one is social chatbots. Social chatbots are usually more humorous and intended to initiate a casual conversation with a human being. For example, a computer program named Microsoft Xiaoice is intended to offer entertaining conversations with users. The second category is general task-oriented chatbots. These chatbots are intended to perform simple general tasks such as setting reminders and alarms. For example, Apple employs Siri to assist users with making calls and composing messages. The last category is specialist task-oriented chatbots, which means the chatbot will only perform a specialized task and could accomplish more complex tasks. For example, medical, commerce, and financial industries have developed chatbots to substitute customer service, request routing, or gather information. The focus of this paper is to discuss the model building and development of specialized task chatbots in the service industries.

This kind of chatbot does not need humorous elements to entertain the users. However, they are more focused on understanding the users' inquiries and generating accurate responses because misunderstanding the users and generating irrelevant responses come with a high cost. For example, a medical chatbot misunderstands the user's syndrome, and providing wrong suggestions might cause severe complications for the patient and lawsuits for the company.

Building a well-functioning Chatbot that understands the text input and responds to inquiries with desired response is a challenging topic that requires sophisticated ideas and well-rounded solutions. Computers, unlike human brains, are exact and rule based. Therefore, a simple model is not feasible to process ambiguous information. A simple model has difficulty associating phrases with different expressions, although they have the same meaning for a human brain. For example, if the model is to retrieve data from a Question-Reply pair database that has a pair: "Where can I buy a cat– Petco." Computers are expected to reply to the same message if the user asks the same question differently– " where is an excellent place to shop for cats." However, the computer would return an error message in this scenario since this question is not given in the database. The ambiguity feature of human language use must be taken care of since it is ubiquitous for people to use different wording in conversation.

The review first analyzes the primary data source and different preprocess approaches used to develop the chatbot. Then in the modeling session, five models are introduced. Although the models

behind chatbots have different methodologies, chatbot development follows a similar structure. Such a generalized structure can be separated into two steps: query understanding and response generation. After that, the model optimization includes how researchers add remediation to the existing model to improve performance. Then, two evaluation methods are introduced in the model evaluation section to assess how well the model responds to the user's inquiry. Lastly, the discussion session provides comments on different models and the trend of developing chatbots.

2. Data Sourcing

The developers of the article usually construct their dataset according to the chatbot's purpose. The chatbot usually has a paired Question and Response(Q&R) database. In the database, researchers would anticipate users' questions based on the chatbot's scope of functions. According to those questions, developers usually manually input the best response. The anticipated question and its best response form a Q&R pair, and many Q&R pairs construct the whole database. For example, Abidah Elcholiqi's team built a Q&R database based on FAQ from Bank BTPN [1]. The data source used in this paper is shown in Table I.

The other primary data source is Application Programming Interface (API) technology. When the chatbot needs specific information, it will use API to retrieve it from a database containing the information.

For example, Eko Handoyo's team designed a chatbot to provide users with information about an airline ticket. Keeping track of fluctuating flight information such as departure time and updates in a self-built database is hard. As a result, he utilized API technology to acquire up-to-date ticket information [2]. Conversation history could also be utilized as the training data for chatbots. For example, Koji Tanaka utilize Switchboard Dialogue Act Corpus, a dataset containing 1,155 five-minute conversations between two participants [3].

Table 1. The main data source introduced in this paper

Author	Country	Publishing Date	Data source
Divia Maduhu et al. [4]	India	2017	Self-built symptoms-disease dataset
Rohit Binu Mathew et al. [5]	India	2019	Self-built symptoms-disease dataset
Bushra Kidwai et al. [6]	India	2019	Self-built database with 50 disease and their symptom
Abidah Elcholiqi et al. [7]	India	2020	Self-built database based on FAQ from Bank BTPN
Anran Jiao [8]	China	2020	Iex-finance API and Self-built database
Eko Handoyo et al. [2]	Indonesia	2018	Ticket.com API
Minghui Qiu et al. [9]	China	2017	Conversation history from Alibaba online customer service center
Pengfei Zhu et al. [10]	China	2018	Conversation history of E commerce platform Taobao
Koji Tanaka et al. [3]	Japan	2019	SwDA corpus

3. Preprocessing

Computers cannot directly process and understand long, complex sentences like the human brain. Therefore, the user's input must be preprocessed into a more analyzable form. Researchers have used several NLP methods to accomplish this goal.

The first step is tokenization, which separates a long sentence from the user into smaller segments and stores it for future use. However, not all these words are helpful for chatbots to understand a user's query and analyzing every single word would significantly increase noise and runtime of the model. Thus, stop word removal and Terms Frequency-Inverse Document Frequency (TF-IDF) is introduced to keep meaningful words for chatbots. Stop words are common words like "a," "an," and "the" that do not contribute much to the meaning of the sentence, and researchers usually remove those words before they process a user's queries. Stop words can be removed by predefining a set of common words and removing them from the tokenized dataset. TF-IDF is a statistical method that calculates the uniqueness of a word across all the documents, and it can be separated into two parts: Term Frequency (TF), which refers to how often certain words appear in a particular document, and Inverse Document Frequency (IDF), which refers to how common a word appears across all documents.

$$W_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right) \tag{1}$$

The TF-IDF approach would return a TF-IDF score between 0 and 1 for all the words, and higher TF-IDF stands for a higher uniqueness, therefore a higher probability that this word would contain pivot information. Researchers could focus on those words with higher TF-IDF scores and get much information from a subset of the tokenized dataset. After strategically choosing important words, Researchers usually further manipulate data sets using lemmatization and stemming, which reduces all the words to their root form and then treats those words that have the same root form the same. This method reduces the word range a chatbot needs to understand and makes our chatbot more efficient. After the above method, researchers usually employ Part-of-Speech (POS) tagging to tag each word. POS tag shows the word's position in the sentence and could facilitate future analysis.

An example of the above preprocessing method would be the user input: "I want to buy 3 cats." The chatbot would first tokenize these sentences into "I," "want," "to," "buy," "3", and "cats" and store those words in a dataset. Then, the TF-IDF and the stop words removal probably would remove "to" in the dataset. After that, lemmatization and stemming convert "cats" into its root form–" cat." Then the POS tagging would tag "I" as subject, "want" and "buy" as a verb, "3" as an adjective, and "cat" as a noun. As for existing databases, researchers would manually do the POS tagging to train the chatbot.

4. Modeling

The modeling behind chatbot can be divided into two parts-Query Understanding and Response Generation. Both parts have different models and logic behind it, which will be separately discussed in the following two sections.

4.1. Query Understanding

Chatbots must first comprehend the user's words to prepare for a relevant response. There are several ways that chatbots use to comprehend the user's intention

4.1.1. Support Vector Machine

Support Vector Machine (SVM) is a supervised machine learning method usually used for classification. Essentially SVM maps all the data in a plane and tries to find boundaries to create these data into different categories. For the 2-dimension binary classification, basic version of SVM is linear, which means a straight line is used as the boundary. However, SVM has a kernel trick.

Nonlinear kernels could be implemented on the SVM model based on the traits of the data, and the boundary would take more complicated forms such as curve or circle. As for the chatbots, multi-dimension SVM are used to match the user's input sentence into the best category. For example, B. Tamizharasi use nonlinear SVM to classify the symptom that user described into different types of diseases [11].

4.1.2. Dialogue act recognition

Dialogue Act (DA) is a crucial topic in NLP, and it refers to an utterance that has a specific function in the dialogue. There are types of utterances that will be shown in the Table. II.

Table 2. The type of utterances

Utterance	Type
I like cats.	Statement
Where is the cat?	Question
Halo	Greeting
kijaspd	Undefinable

DA could also be used for information extraction. Researchers usually use machine learning to classify each sentence into its utterance type according to the words contained in it and its structure. Researchers incorporate the Bayesian rule to predict the DA type. The formulas are given as follows.

$$DA_{Max} = argmax_{DA} P(DA) * P(U|DA) \quad (2)$$

Maximizing the equation above shows what DA has the highest ability occurring given an utterance. From SVM, the chatbot would have a general idea about the main themes the user wants to convey. However, this is not enough. To fully understand the user's input and make relevant responses, the chatbot need also understand the user's DA. Understanding the user's DA is very important in making relevant responses. Knowing DA allows the chatbot to understand whether the users are asking a question or making a request. For example, "Book me a ticket from San Diego to Sacramento on December 30th." and "Are there flights from San Diego on December 30th?" has very similar content, but they represent very different inquiries. The former represents a request, while the latter is a question. Chatbots need to understand the difference in the DA to make relevant responses.

4.1.3. Named entity recognition

Name and Entity Recognition (NER) is quite like POS tagging which involves tagging words and phrases. The difference is that NER tags the entity and names of the words rather than the grammatical component, as shown in the following Fig. I.

Fig.1 The example of name and entity recognition (NER)

This entity tag is helpful for the chatbot to extract information from the conversation and understand the user's query. For example, in the ticketing chatbot, these labels could quickly help the chatbot to understand which ticket a person is interested in and how many he or she wants. Researchers achieve this entity recognition by using NLU (Natural Language Understanding) in RASA Open Source, a machine learning framework for Artificial Intelligence assistants. RASA NLU recognizes features such as capitalization and POS tagging and returns entity tags for the sentence.

4.2. Response Generation

Response Generation (RG) is an essential component of a chatbot. Chatbots not only need to understand human input but also need to generalize a humanlike, relevant response to keep the conversation flowing. Two models could help chatbots achieve this goal and will be discussed in the rest of this session.

4.2.1. Rule based model

The rule-based model is the first approach developed for generating responses and is widely used in earlier chatbots. The most representative rule-based model is Artificial Linguistic Intelligence Computer Entity (ALICE), created in Artificial Intelligence Markup Language (AIML). AIML has a fundamental unit called a category, and each category contains at least one pattern, which is intended to match the user's input and one template on which the chatbot's response is based.

Alice chatbot has a complex set of decision trees and uses the Depth First Search (DFS) algorithm. DFS would look for matches between users' input and pattern word by word and respond with a template. ALICE will use the default model if there is no good match between the user's input and patterns, which might not be a proper response [12] .

4.2.2. Information retrieval model

Information Retrieval (IR) models are similar compared to Rule-Based models. IR also looks for matches between the user's input and existing patterns and pairs up a response associated with that pattern. However, unlike rule-based models, all the patterns and templates are manually defined. IR models use millions of conversations on the internet as training sets. After feeding the model enough input, the chatbot will form a Q&A database. Whenever the chatbot encounters a question, it will match with a similar question in the Q&A database. Then all these candidate responses are reranked through a system such as Neural Networks. Finally, response with the highest rank would be the chatbot's response to that question.

5. Model Optimization

Even with a delicate model, chatbots still face many challenges and have the potential of achieving better performance. Although the chatbots in the service industry are designed to perform specific tasks and offer professional services, some users might try to initiate a casual conversation that might not be relevant to the service. For example, a user might say, "Hope you have a nice day!" Without special consideration, chatbot often fails to retrieve proper answers from the database, which results in user dissatisfaction. However, a Chit-chat model could be aggregated into the existing model to handle this conversation. Chitchat is an attention-based model trained on millions of conversations on Twitter [13]. Such a large proportion of training data helps the chatbot handle almost every kind of everyday conversation. Peifeng Zhu added a chitchat model to his information retrieval model and only activated it when all the candidate responses were below the threshold of 0.3 [10]. Recall that the IR model would return a list of candidate answers with scores. When the model calculates a meager score for all the candidate's responses, the user's inquiries are likely irrelevant to the chatbot's service range. Then, Chitchat would come in and keep the conversation flowing. Another researcher, Kai Sun, adopted the Chitchat model to his existing neural dialogue model and reported that chatbot has improved in terms of engagement, interestingness, humanness, and knowledge [13]. Aggregating Chitchat to the existing model improves the chatbot's performance and helps the chatbot take a huge step towards its goal—mimicking humans in a conversation.

6. Model Evaluation

Evaluation is crucial for the development and enhancement of chatbots. Having the accurate universal method allows the developers to compare the performance of different models better, and researchers could improve or switch the model based on its evaluation. Different authors have

developed their methods for evaluating chatbot responses, and these methods are generally forums on the correctness and relevance of the chatbot's responses.

6.1. Mean Average Precision Method

Mean Average Precision (MAP) is a delicate statistical method aiming to report object detection performance. In this case, MAP measures the relevance of all the candidate responses to the user's input， the confusion matrix as shown in Table III.

Table 3. Confusion Matrix

		Actual	
		True	False
Predicted	True	True Positive (TP)	False Positive (FP)
	False	False Negative (FN)	True Negative (TN)

Recall the confusion matrix above, Precision, which measures how well True Positive can be found out of all optimistic predictions.

$$Precision = TPTP + FP \tag{3}$$

Moreover, the Recall measures how well True Positive can be found out of all predictions.

$$Recall = TPTP + FN \tag{4}$$

Then the Precision and recall curve is generated, and the AP is the area under the curve. The mAP is calculated as the AP of each Precision divided by the number of classes.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{5}$$

The mAP takes care of both Precision and Recall and thus could offer a researcher a good idea of the model's ability to classify the target object according to class. As a result, mAP can be used as a valid indicator for query understanding. For example, Cui utilizes mAP to assess the performance of his model, and the result he got is 0.7742.

6.2. Mean Reciprocal Rank

Mean Reciprocal Rank (MRR) is an information retrieval system measurement and is widely used to evaluate chatbots-based IR systems. MRR is calculated as the mean of Reciprocal Rank (RR), which is how the relevant response is ranked among all the responses in the candidate pool.

$$MRR = \frac{1}{Q}\sum_{i=1}^{Q} \frac{1}{rank_i} \tag{6}$$

If the MRR is high, on average, the chatbot did an outstanding job of retrieving target responses for users' inquiries and vice versa. As a result, MRR evaluates how well the chatbot could retrieve relevant responses.

Both methods are the statistical evaluation of chatbot's performance; however, as it is developed to serve users, users should be the ultimate judge for chatbots. After the chatbot has been put into a real-world scenario, the user often receives a survey question after the response is generated. A typical survey question would be: "Does the information provided answer your question?" These evaluations would provide chatbot developers with valuable case study opportunities. Researchers could investigate each case and analyze why or why not the chatbot is functioning as desired.

7. Discussion

The chatbot can be based on different models. However, there is a shift of models on which different chatbots are based. The earliest chatbots like ALICE tend to be rule-based. They have relatively direct logic, matching the user's input with patterns and output templates. The simple logic

comes with a tradeoff: building a rule-based model tends to be very time-consuming, and the rule-based chatbot can only perform a limited function. The reason behind this is that all the rules are manually crafted. For example, the research team of Alice manually implements more than 5000 rules. Hard coding will make rule-based models time-consuming and labor-intensive and restrict the scope of knowledge that chatbots have. Developers must predict the user's conversation and program it into the chatbot. However, in the actual world application, users might not follow the pattern and thus cannot obtain an ideal response. With the development of ML and NLP, IR models gradually became popular. IR does not need manual implementation of rules because it can learn from the enormous conversation flow happening daily on the internet. Moreover, such an extensive training data set allows the IR model to learn more patterns and answer a broader scope of questions. The current trend for Chatbot building is utilizing neural networks and hybrid models using API technology. Neural networks allow chatbots to interpret inquiries and generate responses like the human brain. Neural Network is a milestone for chatbot because it helps the chatbot towards its final goal–to mimic a human in a conversation with an actual human being. On the other hand, API technology helps free up the local storage by allowing access to other databases on the cloud. Therefore, the developer does not need to build and maintain an extensive database to have enough up-to-date information. The researchers never fight alone in the battle of chatbots. Advancements in statistical models, NLP, and ML will also initiate a revolution in the history of chatbots.

8. Conclusion

The recent development of chatbots shows a remarkable convergence. The primary database is still a self-constructed QA database, but more researchers are gradually aware of the benefits of using a large volume of online conversations to train the chatbot. The most used model has shifted from a rule-based to an IR model accompanied by different machine learning techniques like SVM and neural networks. The combination of these progress facilitates researchers to create chatbots with more knowledge and human resemblance. While these handcrafted models have remarkable performance, it is time-consuming and requires much brainwork. The future of chatbots requires researchers to develop a standard way of building chatbots suitable for all service industries. This could be accomplished as a chatbot is equipped with a more complex neural network and various optimization methods like Chitchat. This paper provides an introduction and overview of different models to those who want to take a first step toward exploring chatbots.

References

[1] Elcholiqi A, Musdholifah A. "Chatbot in Bahasa Indonesia using NLP to provide banking information." IJCCS (Indonesian Journal of Computing and Cybernetics Systems) 14.1 (2020): 91-102.

[2] E. Handoyo, M. Arfan, Y. A. A. Soetrisno, M. Somantri, A. Sofwan and E. W. Sinuraya, "Ticketing Chatbot Service using Serverless NLP Technology," 2018 5th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), 2018, pp. 325-330.

[3] Tanaka, K., Takayama, J., and Arase, Y. "Dialogue-act prediction of future responses based on conversation history." Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop. 2019.

[4] R. B. Mathew, S. Varghese, S. E. Joy and S. S. Alex, "Chatbot for Disease Prediction and Treatment Recommendation using Machine Learning," 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019, pp. 851-856.

[5] Kidwai, B., and Nadesh, R. K. "Design and development of diagnostic Chabot for supporting primary health care systems." Procedia Computer Science 167 (2020): 75-84.

[6] Elcholiqi, A., and Musdholifah, A." Chatbot in Bahasa Indonesia using NLP to provide banking information." IJCCS (Indonesian Journal of Computing and Cybernetics Systems)14.1 (2020): 91-102.

[7] Jiao, A. "An intelligent chatbot system based on entity extraction using RASA NLU and neural network." Journal of Physics: Conference Series. Vol. 1487. No. 1. IOP Publishing, 2020.

[8] Qiu, M., Li, F. L., Wang, S., Gao, X., Chen, Y., Zhao, W., and Chu, W. "Alime chat: A sequence to sequence and rerank based chatbot engine." Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. 2017.

[9] Zhu, P., Zhang, Z., Li, J., Huang, Y., and Zhao, H. "Lingke: A fine-grained multi-turn chatbot for customer service." arXiv preprint arXiv:1808.03430 (2018).

[10] McTear, M., Callejas, Z., and Griol, D. "The Conversational Interface: Talking to Smart Devices: Springer International Publishing."

[11] AbuShawar, B., and Atwell, E. "ALICE chatbot: Trials and outputs." Computación y Sistemas 19.4 (2015): 625-632.

[12] S Sun, K., Moon, S., Crook, P., Roller, S., Silvert, B., Liu, B., andCardie, C. "Adding chit-chat to enhance task-oriented dialogues." arXiv preprint arXiv:2010.12757 (2020).

[13] Singh, D., Suraksha, K. R., and Nirmala, S. J. "Question Answering Chatbot using Deep Learning with NLP." 2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT). IEEE, 2021.