Supporting an Iterative Conversation Design Process

Yoonseo Choi
School of Computing
KAIST
Daejeon, South Korea
yoonseo.choi@kaist.ac.kr

Nyoungwoo Lee
School of Computing
KAIST
Daejeon, South Korea
leenw2@kaist.ac.kr

Hyungyu Shin
School of Computing
KAIST
Daejeon, South Korea
hyungyu.sh@kaist.ac.kr

Jeongeon Park
School of Computing
KAIST
Daejeon, South Korea
patricia021912@kaist.ac.kr

Toni-Jan Keith Monserrat
Institute of Computer Science
University of the Philippines Los Baños
Los Baños, Laguna, Philippines
tpmonserrat@up.edu.ph

Juho Kim
School of Computing
KAIST
Daejeon, South Korea
juhokim@kaist.ac.kr

Author Keywords
Iterative design; Conversation design; Chatbot design process; Early-stage design support; Crowdsourcing

CCS Concepts
•Human-centered computing → Systems and tools for interaction design; Empirical studies in interaction design; *User interface design;*

CHI '20 Extended Abstracts, April 25–30, 2020, Honolulu, HI, USA.

ACM ISBN 978-1-4503-6819-3/20/04.
DOI: https://doi.org/10.1145/3334480.3382951

Abstract
Conversation design is an essential step in building a chatbot. Much like visual user interface design, conversation design benefits from prototyping and user testing to allow for conversation exploration and improvement. However, it can be overwhelming to quickly iterate on the conversation design as the iterative process requires not only designing a conversation but also building and testing a working chatbot equipped with the conversation. We developed *ProtoChat*, a prototype system that supports an iterative conversation design by allowing designers to (1) prototype conversations, (2) test the conversations with the crowd, and (3) review and analyze the crowdsourced conversation data. Results of an exploratory study with four conversation designers show that the designers successfully iterated on their conversation design by reviewing how the crowd followed the conversation, which provided insights into concrete action items for improving their conversation design.

Introduction
One of the first steps in crafting a chatbot is to design its possible conversations with the user. Designers use human-to-human conversation as the basis to create an effective flow of interactions between the user and the chatbot [11]. Similar to other design tasks like designing a website, designing conversations of a chatbot could benefit from iterative design, as well as rapid prototyping and testing.

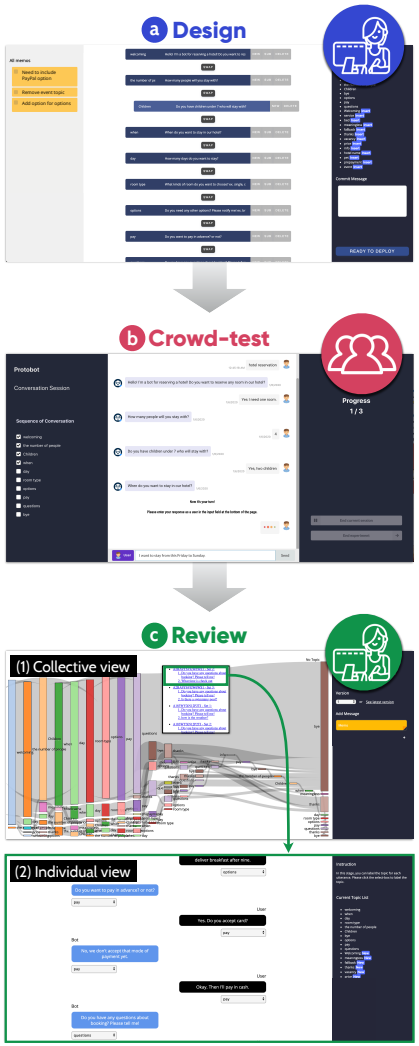

Figure 1: Conversation design process with ProtoChat. Designers can design a conversation (a), test with crowds (b), and browse and analyze the crowdsourced conversation (c(1),(2)).

Conversation design can be defined as planning the flow of the conversation and its underlying logic [11]. There are many guidelines for conversation design from companies like Google [9, 10] and from literature [1, 4, 3]. Even though it is possible to apply such guidelines for a better design, conversation design prototypes need to be iterated but it is hard to be tested without a properly working chatbot. Adding to the guidelines above, work has been done to investigate possible ways to design conversations that can be used for chatbots. Existing approaches collect conversation data from humans by Wizard-of-Oz [12, 6] and workshops [7]. Other approaches formulate the conversation by analyzing existing data sources such as Twitter conversation data [5, 13], mail threads of DBpedia [2], and existing chatbot logs [14]. Despite the guidelines and techniques for designing a conversation, little research has investigated how designers iterate on their conversation design, what challenges they encounter in the process, and what system support can be applied to the process.

To understand how designers iterate on their conversation design ideas and what challenges they face during the process, we conducted semi-structured interviews with two professional conversation designers with at least one year of experience and seven students with prior experiences in conversation design. Results show that designers find it difficult to discover possible conversations in the design process, especially when they are not familiar enough with the chat domain. Also, it is overwhelming to rapidly iterate on the conversation design as the iterative process requires not only the design of a conversation but also prototyping and testing a working chatbot. A low-level working chatbot can be built using frameworks such as Google Dialogflow [1], BotKit [2], and Chatfuel [3], but they require technical abilities which distract designers from focusing on the conversation design itself. Supporting the iterative design of conversation requires designers to get a sense of how potential users might follow the current conversation design.

To address these challenges, we developed *ProtoChat*, a prototype system for supporting designers to quickly iterate on their conversation design. With ProtoChat, designers can (1) prototype conversational flows, (2) test various conversation flows with the crowd, and (3) review and analyze the crowdsourced conversation data, all of which encourage designers to iteratively improve their conversation design. ProtoChat consists of two interfaces: the designer interface and the crowd-testing interface. The designer interface supports designers to (1) craft the conversation with the unit of 'topic-utterance set' in *Draft* page, (2) explore and analyze the crowdsourced data in *Review* page, and (3) review their previous design versions and test settings in *History* page. The crowd-testing interface is an independent interface to test the designed conversation. The crowd can not only test a conversation by responding to the bot utterances but also suggest new bot utterances that can be used in the conversation.

Through our exploratory study with four conversation designers, we found that ProtoChat provides insights into conversation design improvements by visualizing how the crowd followed the conversation. Participants designed the conversation with four interactions: addition of topics, removal of topics, modification of utterances, and change of topic order based on decisions made during their design process. Participants mentioned that ProtoChat enables

[1] https://dialogflow.com/

[2] https://botkit.ai/

[3] https://chatfuel.com/

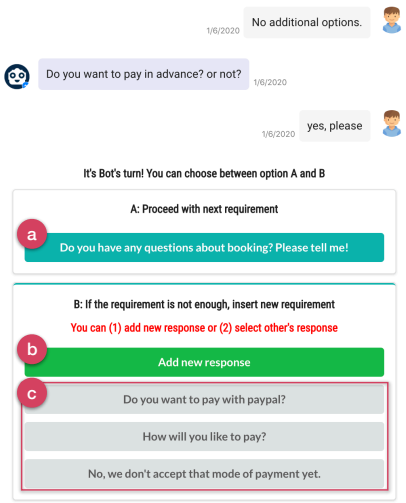

Figure 2: Chabot's turn on the crowd interface. The crowds could either (a) proceed conversation with the designed scenario, (b) insert a new scenario, or (c) follow new scenarios that the crowd created.

them to quickly get a sense of user needs even in a very early stage of chatbot building process.

System Design

ProtoChat supports designers to quickly iterate on the conversation by allowing them to create conversation sequences, test the designed conversation with the crowd, analyze the crowd-tested conversation data, and revise the conversation design. These features are manifest in two main interfaces: the designer interface and the crowd-testing interface.

Designer Interface

The designer interface supports designers to insert and edit low-fidelity conversations, plan how to test the conversation design, review the crowdsourced data, and refer to previous designs and analysis as well. Three main features are provided in designer interface: *Draft*, *Review* (See Figure 1-a, c), and *History*.

In the Draft page, designers can create a conversation by defining a set of 'topics' and 'utterances' (See Figure 1-a), where topic and utterance are two basic building blocks for designing a conversation. *Topic* refers to what kind of questions need to be asked in the domain (e.g., payment) and *utterance* refers to how the topic is addressed within the conversation (e.g., "How would you like to pay?"). When designers finish creating a conversation version to be tested, they can launch custom tests by configuring parameters for collecting user data. Designers can configure (1) the number of crowd workers who will test the conversation flow, (2) the number of chat sessions each crowd worker receives in testing, (3) an option for showing or hiding other crowds' answers (which would be applied in Figure 2-c), and (4) a deployment method (either Amazon Mechanical Turk or a custom link to the crowd-testing interface).

In the Review page, designers can browse and review the crowdsourced conversations. The collective view is where all the crowd-collected conversations are displayed to help designers identify the conversational flow (See Figure 1-c). The conversations are displayed with a Sankey diagram [4], where topics are displayed as nodes and the thickness between the nodes implies the number of utterances (See Figure 1-c(1). When users click on the gray flow between the nodes, they can move to an individual view, where individual conversation performed by each tester is displayed (See Figure 1-c(1),(2)). The new conversations that the crowd testers added are not yet assigned a topic, and the designer can label those conversations with existing or new topics to build up the collective view of crowdsourced data. The collective view is automatically updated with the modified topic labels. Moreover, designers can leave notes to help them revise the design of the scenario.

Additionally, ProtoChat provides the History page, where designers can review their previous design versions and test settings.

Crowd-testing Interface

To help designers quickly test conversation ideas and collect responses from the crowd, the crowd-testing interface is designed as a chat interface. Based on the drafted conversation and testing parameters, a web link is generated so that designers can either deploy the testing interface to a crowdsourcing platform (e.g., Amazon Mechanical Turk [5]) or share it with testers of their choice. The crowd-testing interface (See Figure 1-b) has a unique feature which enables the tester to either hold a conversation by responding to pre-written utterances of a chatbot (Figure 2-a), insert new utterances within an existing conversation flow (Fig-

[4] https://www.d3-graph-gallery.com/sankey.html
[5] https://www.mturk.com

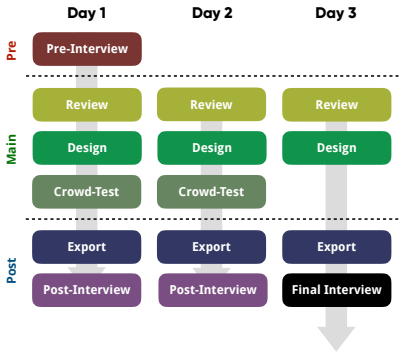

Figure 3: The procedure of the 3-day experiment.

Design

(Day 1) Create a conversation flow to successfully guide users to finish the task, in the form of chatbot.
(Day 2, 3) Revise their previous version of conversation design after reviewing and analyzing crowd data in the Review phase.

Crowd-test

Configure test settings after designing the overall conversation flow and run the test. The testing method was fixed to MTurk.

Review

Browse and analyze the data crowdsourced by the Crowd-testing interface. Participants could move back-and-forth between the individual and the collective view.

Export

Export the design into a hi-fidelity design prototype each day. By requiring participants to export their conversational flow into a design prototype that supports end-to-end scenarios, they were able to recognize whether their ongoing design needs more editing or iterations.

Figure 4: Detailed explanation of each task performed in the experiment.

ure 2-b), or follow new scenarios that other crowd testers created (Figure 2-c). If the crowd chooses to insert new conversations, it is performed as a self-dialogue as the crowd need to design both utterances for a chatbot and a user.

Exploratory Study

The initial evaluation focused on exploring two research questions: *(1) how does the designer utilize ProtoChat to design a goal-oriented conversation?* and *(2) what kind of support is needed for the iterative design of conversation?*. To examine ProtoChat's role during the overall design process, we conducted a 3-day long experiment. During the experiment, we asked participants to mainly work on three tasks with our system: (1) *Design*, (2) *Crowd-test*, and (3) *Review* the conversation each day, so that they run three design iterations. The overall procedure of the 3-day experiment is shown in Figure 3.

We recruited 4 designers (3 female, 1 male) who work on research related to conversation design and have prior experience in chatbot conversation design. Participants received KRW 50K (about 43 USD) for their participation. Figure 4 shows details of each task performed in the experiment.

In addition to the main design activities, the study had an export phase and interviews. After completing the three main tasks, participants were asked to export their design (details are described in Figure 4-*Export*). The pre-interview (Day 1) mainly focused on prior experiences, needs, and challenges in the conversation design process. The post-interview (Day 1 & 2) asked about participants' experience and the design process with ProtoChat; and the final interview (Day 3) additionally asked about the overall usability and feature suggestions.

Result

Participants chose different domains for their conversation design (P1: movie reservation, P2: ice cream order, P3: hotel reservation, P4: house fixing). P1, P2, and P3 were already familiar with the domain, whereas P4 was not knowledgeable about the domain but selected it based on personal interest. To answer the two research questions, we used two axes to analyze the study results. For the first question, we analyzed the micro interactions of participants based on the design process with *ProtoChat*. To answer the second question, we analyzed the participants' design goal of each iteration, which is captured in ProtoChat usage patterns.

Designer interactions during design iterations

Participants improved their design with four main interactions: adding a new topic, removing an existing topic, modifying utterances, and changing the topic order. Figure 5 shows examples of interactions made by two participants.

Add & remove topics

Two frequent interactions during the design process were adding (11 topics added) and removing topics (7 topics removed). P3 (hotel reservation) added the topic *PayPal Option* ("Do you want to pay in Paypal?"), which was prevalently asked by crowd testers in the payment phase. P1 (movie reservation) removed the topic *SnackMenu* – "Which snack would you like to buy? We have popcorn and coke." in the first iteration and added it back with minor utterance modification in their second iteration. P1 mentioned that they thought the topic *SnackPurchase* – "Would you like to order snacks?" would be a good enough question to collect information of the snacks the users want, thus removed the SnackMenu topic. However, after crowd-testing, P1 realized that many users wanted to know what snacks are available snacks, and appended the topic again.

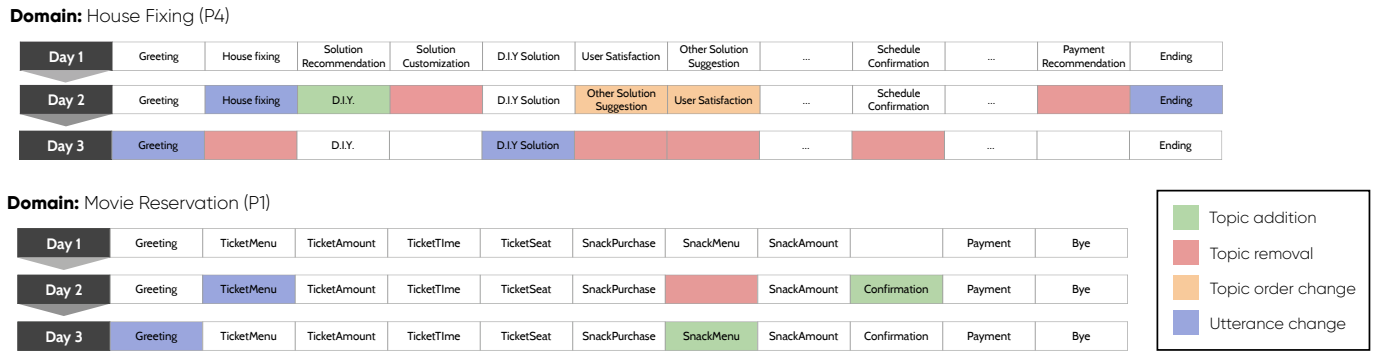

Figure 5: Topic and utterance change during the 3-day experiment in two different domains. Design iteration of domain 'House fixing', a more unfamiliar topic to the designer, shows more complex design change than design iteration of domain 'Movie reservation'.

Modify utterances

Participants sometimes kept the original topics but performed modification at the utterance level. Participants added options to provide users with enough information to proceed. P2 (ice cream order) added five flavor options when asking which three flavors the user wants for the topic "flavors" – "Let me know three flavors :) *(1) mint, (2) chocolate, (3) strawberry, (4) vanilla, (5) peanut*". P2 stated that they added the flavor options to the utterance after seeing the crowd asking for available options, and some not knowing what to answer. P4 (house fixing) specified information for introducing *Plumbing Technician List* – "Mr. Rooter Plumbing is *the closest and least expensive* with quality service and fast responses. *Average price is around 40-50 dollars.*". The modification happened when the participant saw newly crowd-added conversation that asked for the price of the plumbing service, which made them realize that price can be the primary concern for users.

Change the order of topics

The least common interaction was changing the order of topics: only one participant switched the order. P4 (house fixing) swapped the topic order within the topics *User Satisfaction* ("Did you satisfy [sic] with the solution?") and *Other Solution Suggestion* ("If you still have a problem with a clogged sink, mix a cup of baking soda with a half-cup of salt and pour down the drain. Let the mixture sit for several hours, then flush with boiling water.") after their first iteration.

Design goal of design iteration

On days 2 and 3, before reviewing the result, participants were asked about their expectations towards crowd-testing the scenario. Three significant expectations were: (1) to see if the crowd's divergent exploration of new scenarios can improve the current scenario to handle more cases, (2) to verify whether the overall flow of the conversation is easy to follow, and (3) to fix errors in the current conversation.

Responses from the crowd are effective in need-finding and exploring possible scenarios

With crowd-testing, participants were able to observe both the desired flow of a conversation by reviewing the collective view and the needs from the crowd by analyzing individual chat logs. Specifically, participants were able to do need-finding within the context of a conversation. Need-finding includes collecting responses about specific topics/questions or collecting new questions that can be asked for a more natural conversation. They agreed that the responses collected from the crowd helped them discover user needs and make internal decisions for their next de-

sign. For example, P1 and P4 mentioned that based on the responses from open-ended questions, they were able to make UI decisions in the Export step such as deciding between the button UI or free-form question, or even the content of the answer format. P2 realized that they missed out on the option between choosing between a cup or a cone, after seeing the new utterance – *"A cup or a cone?"*, which they felt was an essential topic that needs to be included in an ice cream order.

Make decisions with evidence acquired from the crowd
When participants tried to make decisions about their design, they looked for paths a majority of the crowd testers followed in the Sankey diagram to verify whether their conversation flow makes sense and is easy to follow. Using this information, participants refined their utterances during the iterative process, not the overall sequence or the order of topics. Detailed revisions were made such as changing the tone of the bot utterances (e.g., format of the questions), but the overall sequence of topics remained the same across multiple iterations.

Discussion and Future work

Further investigation toward realistic conversation design
In a real-world setting, it is hard to run iterations in the early stage of a chatbot design process, as it involves prototyping and testing a working chatbot with potential users. P2 mentioned that "*This tool helps in collecting feedback for my own design with a large number of crowds in a lightweight manner, which enables experiencing a quick iterative process.*". As this paper presented an exploratory study, we believe that the design space of supporting iterative design of conversation needs to be further investigated, especially focusing on how to support more diverse forms of conversations. For example, the proposed system did not cover how to support branching interactions in a conversation, which is one of the fundamental building blocks in chatbots.

Improving the conversation design with the crowd
ProtoChat has potential to support quick iterations on the conversation design with the crowd. Leveraging the crowd can be also useful for not only covering the most common scenario but also exploring uncommon cases in a real life. Chorus [8] demonstrated how the crowd could come up with not only a diverse set of responses but also a diverse set of variations of descriptions on a given topic, where they expected crowdsourcing as a potential approach to explore diverse conversations in the chat domain. It is important to investigate how to construct a crowdsourcing pipeline that allows designers to explore various possible conversations so that the designers can get a sense of the overall design space of the conversation. Furthermore, the crowd can decide how to iterate on the conversation with little intervention of the designers. Supporting such interaction between designers and the crowd can be an interesting research direction.

Supporting more complex control flows in conversation
Designers pointed out that branching out the conversation threads based on user input would be necessary to support a more realistic conversation design than the current version. For example, if a chatbot asks the user "Would you like to order snacks?" then an adaptive conversation flow can respond differently based on user input between "yes" or "no". With increased flexibility of the role of crowd testers, we envision the crowd-driven expansion of the scenario tree with moderate designer intervention. Yet, the proposed system is still useful in the early stage of design that requires rapid prototyping and outlining. The branching support can increase the the level of complexity in conversation design that our system can handle.

Acknowledgments

This work was supported by Samsung Research, Samsung Electronics Co.,Ltd.(IO180410-05205-01).

REFERENCES

[1] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N. Bennett, Kori Inkpen, and et al. 2019. Guidelines for Human-AI Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, Article Paper 3, 13 pages. DOI: http://dx.doi.org/10.1145/3290605.3300233

[2] Ram G. Athreya, Axel-Cyrille Ngonga Ngomo, and Ricardo Usbeck. 2018. Enhancing Community Interactions with Data-Driven Chatbots–The DBpedia Chatbot. In *Companion Proceedings of the The Web Conference 2018 (WWW '18)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 143–146. DOI: http://dx.doi.org/10.1145/3184558.3186964

[3] Leigh Clark, Nadia Pantidi, Orla Cooney, Philip Doyle, Diego Garaialde, Justin Edwards, Brendan Spillane, Emer Gilmartin, Christine Murad, Cosmin Munteanu, and et al. 2019. What Makes a Good Conversation? Challenges in Designing Truly Conversational Agents. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, Article Paper 475, 12 pages. DOI: http://dx.doi.org/10.1145/3290605.3300705

[4] Jonathan Grudin and Richard Jacques. 2019. Chatbots, Humbots, and the Quest for Artificial General Intelligence. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, Article Paper 209, 11 pages. DOI: http://dx.doi.org/10.1145/3290605.3300439

[5] Tianran Hu, Anbang Xu, Zhe Liu, Quanzeng You, Yufan Guo, Vibha Sinha, Jiebo Luo, and Rama Akkiraju. 2018. Touch Your Heart: A Tone-aware Chatbot for Customer Care on Social Media. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 415, 12 pages. DOI: http://dx.doi.org/10.1145/3173574.3173989

[6] Meng-Chieh Ko and Zih-Hong Lin. 2018. CardBot: A Chatbot for Business Card Management. In *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion (IUI '18 Companion)*. ACM, New York, NY, USA, Article 5, 2 pages. DOI: http://dx.doi.org/10.1145/3180308.3180313

[7] Rafal Kocielnik, Lillian Xiao, Daniel Avrahami, and Gary Hsieh. 2018. Reflection Companion: A Conversational System for Engaging Users in Reflection on Physical Activity. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 2, Article 70 (July 2018), 26 pages. DOI: http://dx.doi.org/10.1145/3214273

[8] Walter S. Lasecki, Rachel Wesley, Jeffrey Nichols, Anand Kulkarni, James F. Allen, and Jeffrey P. Bigham. 2013. Chorus: A Crowd-Powered Conversational Assistant. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13)*. Association for Computing Machinery, New York, NY, USA, 151–162. DOI: http://dx.doi.org/10.1145/2501988.2502057

[9] Actions on Google. Online; Accessed: 2020-01-05a. Capitalization & punctuation. https://designguidelines.withgoogle.com/conversation/style-guide/capitalization-punctuation.html. (Online; Accessed: 2020-01-05).

[10] Actions on Google. Online; Accessed: 2020-01-05b. Language. https://designguidelines.withgoogle.com/conversation/style-guide/language.html. (Online; Accessed: 2020-01-05).

[11] Actions on Google. Online; Accessed: 2020-01-05c. What is conversation design? https://designguidelines.withgoogle.com/conversation/conversation-design/what-is-conversation-design.html. (Online; Accessed: 2020-01-05).

[12] Archana Prasad, Sean Blagsvedt, Tej Pochiraju, and Indrani Medhi Thies. 2019. Dara: A Chatbot to Help Indian Artists and Designers Discover International Opportunities. In *Proceedings of the 2019 on Creativity and Cognition (C&C '19)*. ACM, New York, NY, USA, 626–632. DOI: http://dx.doi.org/10.1145/3325480.3326577

[13] Anbang Xu, Zhe Liu, Yufan Guo, Vibha Sinha, and Rama Akkiraju. 2017. A New Chatbot for Customer Service on Social Media. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 3506–3510. DOI: http://dx.doi.org/10.1145/3025453.3025496

[14] Li Zhou, Jianfeng Gao, Di Li, and Heung-Yeung Shum. 2018. The Design and Implementation of XiaoIce, an Empathetic Social Chatbot. *CoRR* abs/1812.08989 (2018). http://arxiv.org/abs/1812.08989