Modeling Human Intelligence in Customer-Agent Conversation Using Fine-Grained Dialogue Acts

Qicheng Ding[1*], Guoguang Zhao[2*], Penghui Xu[1], Yucheng Jin[1], Yu Zhang[1], Changjian Hu[2], Qianying Wang[1], Feiyu Xu[2]

[1] Lenovo Research, Technical Strategy and Innovation Platform
[2] Lenovo Research, AI Lab
* Both authors contributed equally to this research
dingqc1@lenovo.com, zhaogg1@lenovo.com

Abstract. Smart service chatbot, aiming to provide efficient, reliable and natural customer service, has grown rapidly in recent years. The understanding of human-agent conversation, especially modeling the conversational behavior, is essential to enhance the machine intelligence during the customer-chatbot interaction. However, there is a gap between qualitative behavior description and the corresponding technical application. In this paper, we developed a novel fine-grained dialogue act framework specific to smartphone customer service to tackle this problem. First of all, following a data-driven process, we defined a two-level classification to capture the most common conversational behavior during smartphone customer service such as affirm, deny, gratitude etc., and verified it by tagging chatlog generated by human agent. Then, using this framework, we designed a series of technically feasible dialogue policies to output human-like response. As an example, we realized a smart service chatbot for a smartphone customer using the dialogue-act-based policy. Finally, a user study was conducted to verify its efficiency and naturalness. Since the dialogue acts are meaningful abstraction of conversational behavior, the dialogue-act-based chatbot could be more explainable and flexible than the end-to-end solution.

Keywords: Dialogue Act, Dialogue Policy, Customer Service, Chatbot.

1 Introduction

With the proliferation of artificial intelligence (AI), many companies have built a service bot to increase customer engagement and save the cost [1]. Chatbots have been largely used as digital assistants on messaging platforms such as Messenger and WeChat. Messenger announced they had more than 300,000 chatbots until May, 2018. Moreover, according to a Business Intelligence research[1], chatbots allow companies to have a significant yearly cost saving up to 46%.

---

1 https://chatbotsmagazine.com/chatbot-report-2018-global-trends-and-analysis-4d8bbe4d924b?gi=3dd7bc9b669c

Unlike an open-ended dialogue chatbot such as XiaoIce, service chatbot is a task-focused agent that aims to make customers satisfied by solving their issues in focused scope effectively such as after-sale service for their products [2]. Compared to live chat, chatbot has many advantages in customer service such as quick response, 24/7 availability, multitasking. However, as a service provider, we may worry about if putting customer conversations in the hand of machines will degrade the service quality and damage customer experience. Therefore, we need to learn from human agent by extracting their conversational behavior and applying them to chatbots.

Therefore, to bridge the qualitive human conversational strategies with automated chatbot interaction, a technical-feasible approach that quantitatively describe the conversational behavior is necessary. Dialogue acts represent the intention behind an utterance in conversation to achieve a conversational goal [3], and they can be technically recognized like [4, 5]. Analyzing the human conversation in terms of dialogue acts such as statements or requests, can give essential meta-information about conversation flow and content [6], and can be used as a first step to develop automated chatbots [7]. For example, one basic descriptive strategy is "Checking if the customer has any other question when he/she expresses that his/her issue has been solved". This can be codified into a technical solution using dialogue acts, such as "After the agent outputs Proposing Solution act, if the user inputs dialogue acts that relate to problem has been solved, such as Accept, Thank or Will Try, then the agent should output acts combination such as Thank + Offer Help to check if the user has any other question".

Researchers have established various of dialogue framework such as [6, 8, 9], however, in terms of smartphone customer service, a specific dialogue act framework is needed. There is more domain-specific behavior such as "I will try the solution" that need extra act to capture. Besides, we observed that customer and agent demonstrate different conversational behavior, therefore, these two roles need different act frameworks for better description, recognition and prediction. Additionally, one sentence can have multiple intents [5]. Dialogue act framework needs to consider using multiple acts to extract major intentions. However, this paper intends to utilize dialogue acts to construct technically feasible dialogue policies. The multiple acts will increase the complexity of framework dramatically. As a preliminary exploration, we focus on depicting the major act in each user input and representing agent output with multiple acts.

In this work, following a data-driven process, we developed a novel fine-grained dialogue act framework to describe quantitatively conversational behavior in the field of smartphone customer service. Then, to transfer the human intelligence to machine intelligence, we designed a series of technically feasible dialogue policies using dialogue acts to generate human-like response automatically. On top of these, we realized a smartphone customer service chatbot as an example and conduct a user study to verify its efficiency and experience. The major contributions here include 1) a novel two-level fine-grained dialogue act framework specific to smartphone customer service; 2) a technically-feasible process to transfer human intelligence into machine intelligence by dialogue-act-based policy which makes the technical realization more flexible and explainable; 3) a smartphone customer service chatbot to verify the feasibility and experience of our dialogue act framework.

2 Related Work

Many researchers with different backgrounds studying human conversations and developing computational speech and dialogue act models [3, 10]. [8, 11, 12] used more generic labels in order to cover the majority of dialogue acts in a conversation. In 1997, Core and Allen [11] presented the Dialogue Act Marking in Several Layers (DAMSL) as a standard for conversation annotation. The framework contains 220 tags, divided into four main categories: communicative status, information level, forward-looking function, and backward-looking function. [8, 12] established less fine-grained framework for general conversation. However, the generic framework falls short in understanding and analyzing customer service conversations.

Specific to task-oriented dialogue such as customer service, Ivanovic modeled Instant Messaging dialogue using 12 act labels, e.g., Statement, Open-Questions , then proposed a method to predict utterances in task-oriented dialogue using their labels [13]. Following Ivanovic's work, Kim et al. [9] classified dialogue acts in both one-on-one and multi-party Instant Messaging chats. More recent works of dialogue acts modeling for customer service conversations on Twitter can be seen [5] etc.

Considering that our research aims to code conversational behavior in smartphone customer service, we need to expand aforementioned acts and develop a more fine-grained framework. The most similar work to ours is that Oraby et al. on developing a taxonomy of dialogue acts frequently observed in customer service on Twitter [6]. They proposed a two-level framework, the first level is more generic including Greeting, Statement etc. The second level contains over twenty specific acts such as Opening, Yes-No Question etc. Rather than focusing on Twitter, we seek to model the dialogue behavior occurring in smartphone customer service domain.

3 Dialogue Act Framework

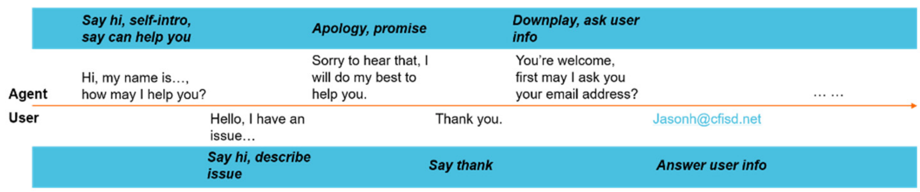

Fig. 1. An example of human agent chatlog abstraction

This section describes how we built a dialogue act framework for smartphone customer service. We selected randomly twenty human conversation chatlog from customer service center, then tagged each sentence using keywords to represent the main meanings (abstraction) as shown in **Fig. 1**. Then we captured major conversational behavior such as asking issue, affirm/deny a feedback etc. Due to the different roles of customer and agent, they exhibited different keyword sets, therefore, we established two frameworks to classify customer and agent act respectively: dialogue act/user and dialogue act bot (We use "user act" and "bot act" due to the usage in human-bot interaction).

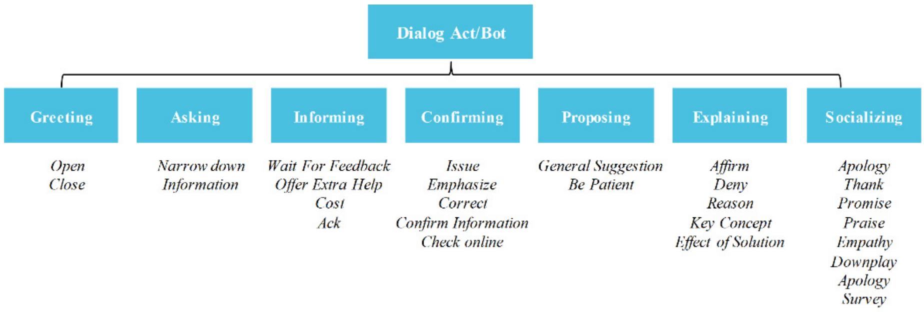

Fig. 2. The dialogue act framework for bot act part

We established the dialogue act framework using iterative reconstruction. At first, we compared each keyword with acts from previous work, then selected the most suitable one as our dialogue act and gave it an operative definition. If there is a keyword that cannot be classified by existing acts, we will define a new one and recheck all tagged core meaning to see if there is any one belong to this new act. We continued this iteration until there is no new act appeared, this procedure could assure that our framework covers the major conversational behavior. Finally, we re-organized all the acts into a two-level hierarchical framework.

Fig. 2 and **Fig. 3** demonstrate the dialogue act/user and dialogue act/bot respectively. Dialogue act/user contains six first-level acts: Greeting, Asking, Informing, Feedback, Explaining and Socializing. Dialogue act/bot has two more first level acts: Confirming and Proposing, and no Feedback category. As for the second level, there are quite different acts because of the different purposes of customer and agent.

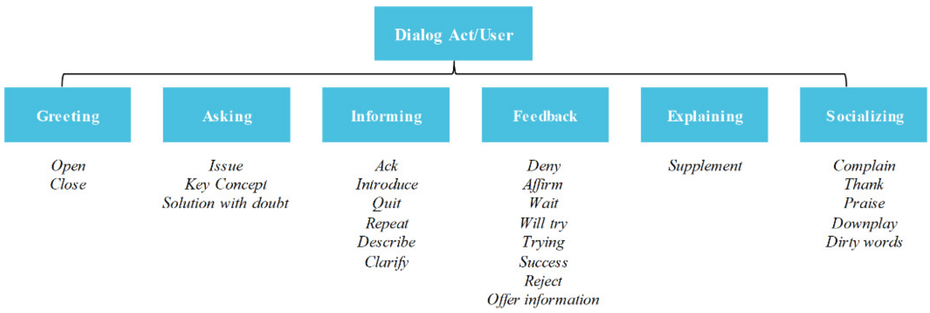

Fig. 3. The dialogue act framework for user act part

4 Dialogue-Act-based Policy

After establishing the dialogue act framework, the next step is to construct technically feasible policies so that human intelligence can be transferred into chatbot. Additionally, since we used dialogue acts as information carrier, the whole process such as what kind of user behavior, what kind of bot response etc., is clear to us, then we could

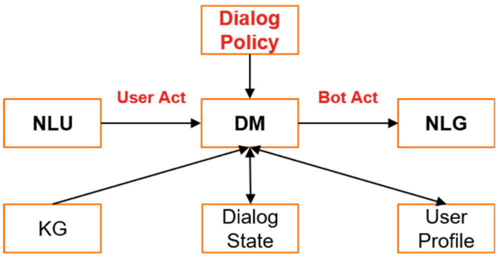

Fig. 4. A service chatbot framework using dialogue act as information carrier.

modify the bot strategy by simply changing the output bot acts. Therefore, the chatbot can be more explainable and flexible.

Fig. 4 illustrates our service chatbot framework using dialogue act as information carrier. The NLU module (natural language understanding) extracts user input such as intents, user behavior and passes to DM (dialogue management) module by dialogue act. The DM combines user act, KG (knowledge graph), dialogue state and user profile together to generate policies, and passes to NLG (natural language generation) through bot act. The NLG is responsible to output human-like sentence based on bot act.

To better handle user response, we divided the conversation into four phases: Opening, Targeting, Solving and Ending. Opening contains conversation at the beginning to the user states phone issues, following is Targeting phase which refers to varied questions (e.g., phone model) leading to final solutions. Solving phase starts with proposing final solutions, and ends with the user accepts the solution or transfers to live agent. Ending phase includes questions before closing the conversation.

With the conversation phased, bot could customize its next step. **Fig. 5** demonstrates dialogue policies examples. For each policy, dialogue act/user describes user's input (e.g., Open). Then bot combines user act, dialogue state (Opening phase) together to plan next bot intent (e.g., Waiting for Issue), and use dialogue act/bot to express its strategy (e.g., Greeting-Open + Informing-Introduce + Asking-Offer Help). After that, the NLG generates human-like response based on the dialogue-act-based policy (e.g., "Hi, my name is XX. How may I help you?").

User Act	Sub-act	Bot Next Intent	Bot Act	Example	Case Phase	Context
Greeting	Open	**Waiting for issue**	Greeting-Open + Informing-Introduce + Asking - Offer Help	Hi, my name is Moli. How may I help you?	Opening	No question left for last visit
		Check the question left on last visit	Greeting-Open + Asking-Before Issue	Hi, how about the XX question you mentioned last time?	Opening	Have question left for last visit
		Treat as Digress			Targeting/Solving/Ending	
	Close	**Close Case by offering extra help**	Greeting-Close + Informing-Offer Extra Help	Bye, if you need other help, please contact ...	Ending	Closing case for the first time
		Close Case directly	Greeting-Close	Bye	Ending	Closing case again

Fig. 5. Examples of dialogue act-based dialogue policy

5 Technical Realization

The main DM tasks include status-tracking, decision-making and topic-management. Therefore, the DM architecture (**Fig. 6**) is roughly divided into three parts: Hypothesis Rank and Select (HRS) and Carry Over, Task Processing Layer, and Making Plans.

Hypothesis Rank and Select & Carry Over

The main purpose of this layer is to determine the user intent. There are two ways to determine the user's intent. One is that the NLU can directly and undoubtedly present the user's intent. The other is that the NLU can present the user vague intent or only some hints. At this moment, the system needs to use context to infer the user's real intent and confirm with users by dialogue acts. Since the first way is self-explanatory, this section will focus less on the first way but more on the second way.

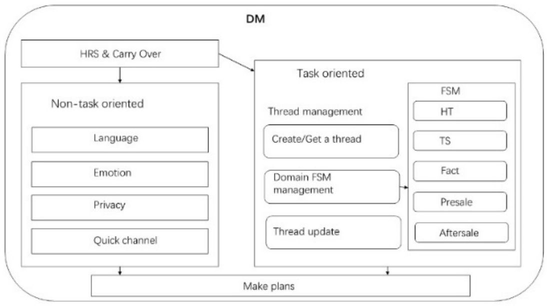

Fig. 6. Dialogue Management architecture, include task-oriented module and non-task-oriented module

The arbitrariness and incompleteness of the user expression brings a challenge for the dialogue system to understand user intent. In a real conversation, the user does not speak a complete sentence, which often lacks a subject or a predicate. At the same time, users are very resistant to repeating previously given information, such as addresses, models, and emails etc. According to the analysis of online logs, the main problems solved by contextual inference are shown in **Table 1** below. If the user's intentions are uncertain, we can confirm with the user through the interaction of dialogue acts.

Table 1. The task of module is slot inference and intention inference

Question Type	Description
Slot inference	Get the current required slots from User Profile, such as mailbox, SN code, IMEI
Intention inference	When the user's intention is unclear, the system can use context reasoning to confirm its real intention

Task Processing Layer

Task processing layer has two major parts. One is the non-task-oriented processing, which includes unsupported languages, emotions, privacy etc. The process in this part is mainly for prompting users with specific words. The second is task-oriented task domain processing. The so-called "task domain" refers to the question asked by the user, which is closely related to specific business problem provided by the current system. A specific task is the user's specific problem, also known as the user intent.

We use a two-tier structure to fully process user intents. We name the first level as "conversation thread", meaning that the current user intent is managed as a session thread that continues until the user solves the problem. On the second level, we use different finite state machines to follow up the processing state of the user intent and adopt different strategies to interact with the user in different states, so as to complete solving the user questions.

Conversation Thread Management

According to specific scenarios, and for the convenience of managing individual conversations, we defined three states: Active, Pause, and Finish. For example, a new intent is started before the user finishes an intent. The original intent is paused, and the new intent is activated. When the new intent completes (Finish), the user can arbitrarily choose whether to continue (Active) or abandon (Finish) the original intent.

Slot Based Finite State Machine

For multiple states in a conversation, it is better to use a state machine for maintenance. This system adopts the hierarchical structure. The first layer state machine is responsible for switching process between the five task domains (how to; trouble shooting; after sale; presale; facts). The state machine design for each task domain is slightly different. This study takes "How to" domain as an example to explain the design process.

Table 2. The state's meaning of HowTo finite state machine and corresponding dialogue acts

FSM state	Description	Dialogue act
init	original state	Opening
slotNotFull	Insufficient slot information	Asking Information
waitUserInput	Wait for information from the user	Waiting Feedback
slotClarify	Verify the information provided by the user	Confirm Information
slotFull	Verify user intent	General Suggestion
deliverAnswer	Push answers to users	Implementable Suggestion
errorHandling	error handling	None

Table 2 shows the specific meaning of discrete states in "How to" domain. In addition, a task-driven dialogue system requires information from users. For example, in the booking system, users are required to provide their location of departure and destination, etc., known collectively as slots. Here, we define states based on slots and use finite state machines to maintain transitions between states (**Fig. 7**).

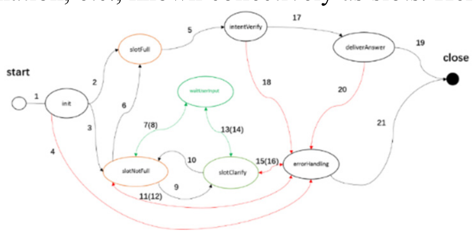

Fig. 7. HowTo state transition example

Make plans

This part will generate dialogue acts of bot, then pass the acts to NLG module. We will not elaborate this part since it is not the main topic of this paper.

6 User Study

In this section, we conducted a preliminary user study to verify our customer service chatbot's efficiency and experience. We pre-defined typical smartphone customer issues such as cannot power on, cannot charge etc., and asked participants to solve these issues using our chatbot. We then recorded their subjective evaluation and comments about their feeling with our chatbot.

Participant.

Considering that our chatbot's target smartphone users mainly speak English, therefore, we recruited 12 native English smartphone users (6 males, 6 females, age range 18~30) to eliminate the extra influence from language and gender.

Task Design.

We selected randomly 25 typical smartphone issues such as order issue, cannot turn on, and each participant was asked to solve 15 issues randomly. The purpose of a task is to give the participant a scenario that his/her phone has something wrong and need to solve with our chatbot. Therefore, for each task we designed a brief background to provide basic information such as who, where, when and what.

Questionnaire.

Refer to [14], we constructed our questionnaire to record their demographics and evaluation of their experience with chatbot. The questionnaires include: demographics (name, age, gender, nationality etc.), post-task (**Table 3**) and post-test (**Table 4**).

Table 3. The post-task questionnaire

Dimension	Sample Item	Question Type
Dialogue efficiency	It took me too much time to complete the task	5 Likert Scale
Dialogue control	I always knew what to say next	5 Likert Scale
Reliability	The chatbot did what I expected	5 Likert Scale

Table 4. The post-test questionnaire

Dimension	Sample Item	Question Type
Dialogue control	I felt the conversation being under my control	5 Likert Scale
Dialogue control	I always knew what to say next	5 Likert Scale
Satisfaction	Overall, I feel satisfied about the chatbot's performance	5 Likert Scale
Satisfaction	I would like to visit the chatbot again	5 Likert Scale
Naturalness	The chatbot behaved naturally	5 Likert Scale
Efficiency	Conversation with the chatbot was efficient	5 Likert Scale
Efficiency	Talking to the chatbot was confusing	5 Likert Scale-Reverse
Usefulness	The chatbot makes issue solving more efficient	5 Likert Scale
Understanding	I had the feeling the chatbot understood me well	5 Likert Scale
Naturalness	What the chatbot said made sense to me	5 Likert Scale
Naturalness	The chatbot's reactions are appropriate	5 Likert Scale

Procedure

Fig. 8 illustrates the user study procedure: set up testing environment, then asked the participant to fill the demographics and let the participant be familiar with this testing. After that, the participant completed 15 tasks with rand sequence. After each task, the participant completes post-task questions, after all tasks, the participant completed post-test

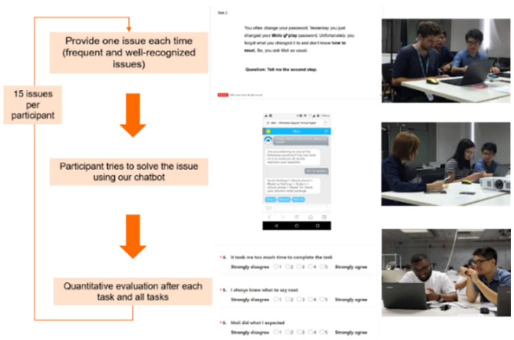

Fig. 8. User study experiment setup and procedure

questions and answered several open questions such as comments to our chatbot. Each participant received 20 USD as incentives.

Result

Fig. 9 shows the statistics of post-task questions. Both dialogue control and reliability achieved acceptable performance. The dialogue efficiency was slightly lower than 3, which might because that participants expected more direct solution rather than an external link. For example, participant thought that they could get the order status directly through our chatbot rather than a website link that need extra interaction to get the answer. **Fig.10** demonstrates the average scores of post-test questions. As we can see, all the scores are higher than 3, which means that our chatbot acquired promising efficiency and experience, especially the perceived conversation naturalness and satisfaction, dialogue control and understanding. Additionally, we collected participants' comments about our chatbot, e.g., "this chatbot solved my issue quickly", "The best part is it responds instantly, unlike the human agent that needs a long-time waiting", "The conversation is not robotic" etc. All these positive feedback support that our chatbot can handle customer service issue properly by using dialogue act framework. However, we also got some negative feedback such as "the chatbot cannot respond properly when it cannot understand my question", "it seems that cannot support multiple-turn conversation". In the future, we will expand this preliminary user study to be more systematic experiment, so that we can draw more solid conclusions.

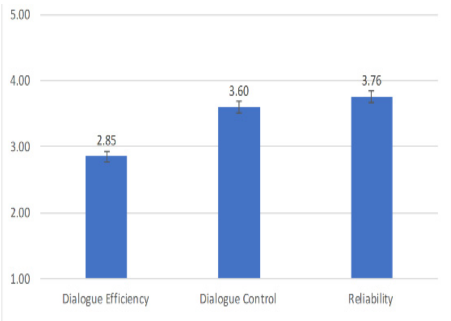

Fig. 9. The post-task results

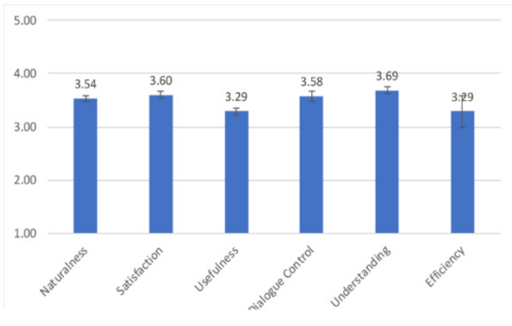

Fig. 10. The post-test results

7 Conclusion and Future Work

This paper established a fine-grained dialogue act framework specific to smartphone customer service. Compared with previous work, this framework contains more domain-related conversational behavior. Besides, two separated frameworks capture customer and agent acts respectively. Then, we designed a series of dialogue-act-based policies to transfer human strategies to automatic human-bot interaction. After that, we realized a customer service chatbot using the established acts and policies. Finally, a user study demonstrated the efficiency and naturalness. Compared with end-to-end chatbots, this dialogue-act-based realization can be more explainable since we know exactly what kind of user behavior the chatbot is dealing with and how. In addition, our dialogue policies are also represented by dialogue acts, we could modify the chatbot's strategy flexibly by changing the bot act combination.

In the future, we would continue to refine the dialogue act framework so that it could cover conversational behavior as more and as accurately as possible. The dialogue act framework is not only an information carrier between qualitative human strategy and technical application, it could also be used in human-bot interaction analysis, i.e., using this act framework to analyze the interactive patterns when the user is talking with our chatbot, and then proposing proper strategies to tackle unraveled problems.

References

1. Jacobs, I., et al., *The top 10 Chatbots for enterprise customer service.* Forrester Report, 2017.
2. Xu, A., et al. *A new chatbot for customer service on social media*. in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 2017. ACM.
3. Austin, J.L., *How to do things with words*. 1975: Oxford university press.
4. Klüwer, T., H. Uszkoreit, and F. Xu. *Using syntactic and semantic based relations for dialogue act recognition*. in *Proceedings of the 23rd international conference on computational linguistics: Posters*. 2010. Association for Computational Linguistics.
5. Zhao, T. and T. Kawahara, *Joint dialog act segmentation and recognition in human conversations using attention to dialog context.* Computer Speech & Language, 2019. **57**: p. 108-127.
6. Oraby, S., et al., *Modeling and Computational Characterization of Twitter Customer Service Conversations.* ACM Transactions on Interactive Intelligent Systems (TiiS), 2019. **9**(2-3): p. 18.
7. Jo, Y., et al. *Modeling dialogue acts with content word filtering and speakpreferences*. in *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing*. 2017. NIH Public Access.
8. Stolcke, A., et al., *Dialogue act modeling for automatic tagging and recognition of conversational speech.* Computational linguistics, 2000. **26**(3): p. 339-373.
9. Kim, S.N., L. Cavedon, and T. Baldwin. *Classifying dialogue acts in multi-party live chats*. in *Proceedings of the 26th Pacific Asia Conference on Language, Information, and Computation*. 2012.
10. Bigelow, J.C., *Language, mind, and knowledge (minnesota studies in the philosophy of Science, Vol. VII).* Linguistics and Philosophy, 1977. **1**(2): p. 301-304.
11. Core, M.G. and J. Allen. *Coding dialogs with the DAMSL annotation scheme*. in *AAAI fall symposium on communicative action in humans and machines*. 1997. Boston, MA.
12. Jurafsky, D., E. Shriberg, and D. Biasca, *Switchboard-damsl labeling project coder's manual.* Technická Zpráva, 1997: p. 97-02.
13. Ivanovic, E. *Using dialogue acts to suggest responses in support services via instant messaging*. in *Proceedings of the Australasian Language Technology Workshop 2006*. 2006.
14. Klüwer, T., et al. *Evaluation of the KomParse Conversational Non-Player Characters in a Commercial Virtual World*. in *LREC*. 2012.