System Utterance Generation by Label Propagation over Association Graph of Words and Utterance Patterns for Open-Domain Dialogue Systems

Hiroshi Tsukahara   Kei Uchiumi

Denso IT Laboratory, Inc.

Shibuya Cross Tower 28F, 2-15-1 Shibuya, Shibuya-ku, Tokyo, Japan

Abstract

A novel graph-based utterance generation method for open-domain dialogue systems is proposed in this paper. After an association graph of words and utterance patterns from a dialogue corpus is constructed, a label propagation algorithm is used for generating system utterances from the words and utterance patterns in the association graph that are found to strongly correlate with the words and utterance patterns that appeared in previous user utterances. We also propose a crowdsourcing framework for collecting annotated chat data so that we can implement our method in a cost effective manner. Crowdsourcing is also used for conducting subjective evaluations and the results will show that the proposed method can not only provide interesting and informative responses but it also can appropriately expand the topics by comparing them to a well-known chat system in Japanese.

1   Introduction

Chatting plays a lot of important roles in human communications for naturally exchanging diverse information, facilitating collaborative tasks, or even enhancing the quality of the conversations themselves. For dialogue systems as well, the functionality of being able to create chats is considered to have a significant importance regardless of whether task-oriented or non-task-oriented. There are currently many types of smart devices in our daily life and most of them have spoken dialogue interfaces, although they are basically limited to question-answering. However, there are cases where people do not always have the clear intent on searching for something but they just want to know whether there is anything interesting they should know. In such cases, if the systems could offer a chats function instead, people may be able to make such unconscious or potential intentions clear by themselves through chats with these systems.

However, it is quite challenging for dialogue systems to automatically generate chat responses because of the wide variety of topics in user utterances. In ordinary dialogue systems, i.e., rule-based systems, a very large number of hand-crafted rules and utterance patterns, or templates, would need to be prepared for extending the coverage of topics they can handle. However, this would be a very formidable task both to create them and to maintain them while keeping them up to date. Thus, a data-driven approach that makes use of the huge amount of conversational resources currently on the web, such as microblogs or social network media, as corpora have been recently investigated (Shibata et al., 2009; Sugiyama et al., 2013). These corpora contain a large number of sentences that cover a wide range of topics, but there are many noisy sentences that do not contain meaningful content themselves. Another issue with this approach is that it basically selects sentences that are similar to the user utterances on the surface-level. Thus, the generated responses tend to be monotonous and the topic of conversation is not naturally changed by these systems.

We propose a graph-based approach to address these issues. It is based on a dialogue corpus with a considerably large number of utterances. Out of a corpus, we construct an association graph, which

is a bipartite graph with word and utterance pattern nodes, where a word represents a named entity and an utterance pattern represents a template of utterances reduced by replacing their named entities with slots holding the type of named entities that are originally placed there. The association graph is used for finding words and utterance patterns that belong to the same semantic category, or topic of conversation, and formed dynamically using label propagation over the association graph with the words and utterance patterns of previous utterances. The system utterances are synthesized out of those words and utterance patterns.

This paper is organized as follows. First, we explore the use of crowdsourcing for efficiently constructing a dialogue corpus in Sec. 2. The details of the proposed method are described in Sec. 3. We discuss the results from a subjective evaluation in Sec. 4. These results support the concept that the proposed method can create responses with significant and interesting information and that it can appropriately expand the topics. We introduce the related works in Sec. 5. Finally, we give a summary and present some future prospects for the present study in Sec. 6

2 Framework for Constructing a Dialogue Corpus with Crowdsourcing

We describe our framework for constructing a dialogue corpus in text chats by making use of crowdsourcing. The utilization of crowdsourcing is now getting popular for collecting data and conducting user assessments. (Eskenazi et al., 2013; Lasecki et al., 2013; Mitchell et al., 2014). The merits for using crowdsourcing are that many workers can work simultaneously at low cost.

In addition, we can now find many kinds of online collaboration platforms like slack[1]. We can create a number of rooms, which are called channels in slack for example, where a number of workers can simultaneously create chats on those platforms. They also support highly interactive customizable browser interfaces and many APIs for connecting to other services provided outside themselves. Therefore, we can define our own markers that can be used for annotation, or we can send utterances to bot servers

outside the system for watching the progress of a conversation or violations of the guidelines in real-time. The notification to workers can be sent from the bot server to the channels as well. The chat logs can also be exported using those APIs. Thus, we found these are ideal environments for collecting chat data, annotating them online, and remotely managing them.
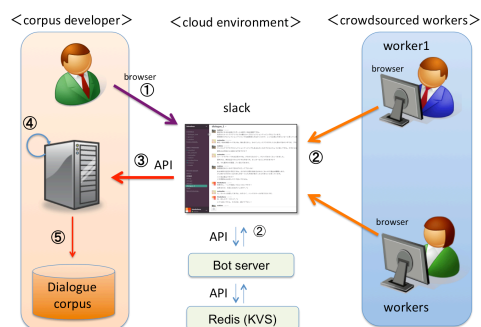

Figure 1: Framework for collecting chat data, annotating them, and exporting into a structured database as a corpus

We show our framework for collecting text chats, annotating them, and exporting the annotated chat logs into a structured database as a corpus in Fig. 1. We selected Slack as our online platform for this paper. The numbers in Fig. 1 show the procedural flow.

In the first procedure, the corpus developer creates a team in Slack and customizes the markers for quickly and correctly inputting annotations. Emoticons are used to represent these markers in a chat stream on the browser. We can create plural channels so that several pairs of workers are able to simultaneously input their chats. The connection to a bot server is also created so that the system can automatically watch the inputs by each worker. We use Hubot[2] for creating a bot server and Redis[3] for storing the working data of each worker. The bot server is placed on a cloud server hosted by Heroku[4]. In fact, all these components are open platforms and open source software. Thus, anyone can create such an environment without incurring any costs, so you can at least try this framework if you want.

In the second procedure, workers access the URL of a channel introduced by the corpus developer at a scheduled time. Once both of the workers arranged

---

[1] https://slack.com

[2] https://hubot.github.com/

[3] http://redis.io/

[4] https://www.heroku.com/

as a conversation pair come online, they start inputting utterances according to the prescribed guidelines. They begin with greetings and introducing themselves and expand the topics by selecting them from the specified genres. In our case, the workers are required to chat by choosing from news on current affairs, sports, entertainment, or gourmet information. The utterances input by the workers are sent to the bot server and the number of utterances are then counted. The check as to whether or not the utterances are in accord with the guidelines may also be checked here or presenting suggestions for annotation may also be sent on the fly to the workers in a working channel. A notification is sent to the channel directly if the number of utterances reaches a required amount so that the workers can notice the completion of a dialogue session. An example of the annotations for utterances are presented in Fig. 2. We designed the way of annotating so that the workers can easily input in the message format of the browser. We define only three kinds of annotations: (a) the location and type of named entities, (b) the dialogue acts of the utterances, and (c) the topics of the utterances. The types of dialogue acts is also limited to the following eight types so that even workers without a good knowledge of natural language processing can understand: (1) greetings, (2) yes-no questions, (3) yes-no answers, (4) provision of information/self-disclosure, (5) presentation of new topics, (6) questions, (7) answers, and (8) feedback/opinions. We found that it is useful to define some of the special annotations for smoothly managing the dialogue input tasks. For example, we define the annotation string "rem", which can be put at the beginning of the utterance, for indicating this utterance is in fact a comment. This annotation can be used to exchange messages between workers, corpus developers, and proofreaders directly on a channel. We also define the annotation string "New-Dial", which is used by itself, to indicate the beginning of a new dialogue session.

In the third and fourth procedures, the chat logs are exported by the corpus developers in charge of proofreading the annotations. The annotations for each utterance are checked by two proofreaders in the present study. Thus, including a dialogue input worker, the annotations are checked by at least three people. We explain the proofreading procedures in
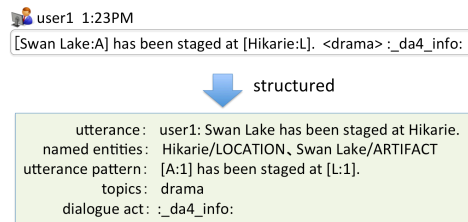

Figure 2: Example of annotations for an utterance. The square brackets ([and]) annotate the position of a named entity and its entity type is also supplied after a colon. The words enclosed in angle brackets ($<$ and $>$) annotate the topics of the utterance. A string enclosed by colons (e.g., :_da_info:) annotates the dialogue act of the utterance.

detail in the following paragraphs. After finishing proofreading the annotations, the fifth procedure is performed for exporting the annotated chat logs into a relational database to store the structured data of a dialogue corpus.

We recruited native Japanese-speaking crowd workers and collected twenty thousand annotated utterances over a period of about three months. The workers were distributed all over Japan from the north to the south and their ages ranged from 20 to 49. Only two workers were male. This size of the corpus was quite moderate compared to a web-scaled corpus, but it is still large enough for our proposed method to work. The speed of collecting annotated utterances depends on how many proofreaders are used. Proofreaders familiar with the work can check about six hundred utterances a day and two proofreaders were used in the present study.

We show the detailed procedures for proofreading annotations in Fig. 3. The numbers represent the flow of the proofreading procedures. The annotated utterances are exported for the first time in procedure (2). Then, two proofreaders check it in procedure (3), and the requests for revision are dispatched to each crowd worker from procedure (4) to (6), where the requests for revision are copied as a backup. Then, the workers revise the annotations by accessing the channel on Slack in procedure (7). The results of the revisions by the workers are checked by comparing them with the backup in procedure (8). If there are any differences, then new requests for revision are created in procedure (9). Procedures (4) to (9) are repeated until the differences are eliminated.
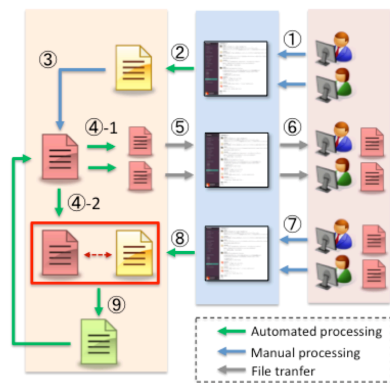
Figure 3: Procedures for proofreading annotations

The characteristic of our framework for collecting a corpus using crowdsourcing is that the workers are not independent but they collaborate with each other in one task. We can collect the utterances of the workers by using a dialogue system as a conversation partner. It might be reasonable to collect the user interaction behaviors using dialogue systems and make use of them to construct a dialogue corpus (Mitchell et al., 2014). We are interested in collecting worker dialogues to learn how people develop their conversations and how topics are naturally explored by them.

3 Graph-based Method for Generating Utterances

In this section, we describe our proposed method for generating utterances. It relies on an algorithm in semi-supervised learning called label propagation over graphs, and we apply it to the association graph of words and utterance patterns, and Fig. 4 depicts an example. We can see in the figure that the label propagation with the regularized Laplacian can successfully extract semantic categories depending on the structure of a bipartite graph of instances and patterns, i.e. words and utterance patterns in the present paper (Zhou et al., 2004; Komachi et al., 2009). Roughly speaking, the words and utterance patterns that are linked to each other are considered to share the same semantic relevance to some extent. This semantic relevance is called a semantic category and can be regarded as a topic talked about in the conversations. By making use of the label propagation over the association graph, we can extract words and utterance patterns that share the same se-

mantic category with words and utterance patterns that appeared in previous utterances. It is expected that synthesizing those words and utterance patterns can help to generate utterances that expand the topics while maintaining the relevance to the current topic in a conversation.
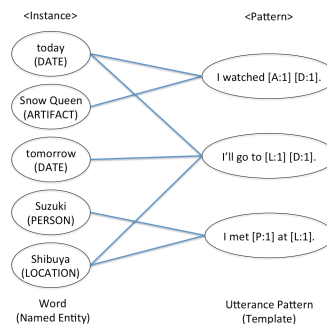

Figure 4: Association graph of words and utterance patterns.

We depict the architecture of our dialogue system in Fig. 5. Procedures (2) to (4) should be performed in advance to obtain the graph Laplacian data out of a corpus of procedure (1), which is necessary in the label propagation procedure. Procedure (2) extracts the named entities and utterance patterns making use of the annotations. The utterance patterns are obtained by replacing the named entities with slots that specify the type of named entities that can be applied. Then, an association graph of words and utterance patterns is constructed by linking the word and utterance pattern nodes if they co-occur in an utterance in the corpus. We introduce an instance-pattern matrix W, which represents the frequency of the co-occurrence of instances and patterns. Let us denote a word as w_i and a utterance pattern as p_j, and then, the instance-pattern matrix W is defined by

$$W_{ij} = \frac{|w_i, p_j|}{\sum_k |w_i, p_k|}, \qquad (1)$$

where $|w, p|$ represents the frequency of the co-occurrence of a word w and an utterance pattern p.

In Laplacian label propagation, the similarity matrix A between instances is measured using a regularized Laplacian

$$L = I - D^{-1/2}(A)AD^{-1/2}(A), \qquad (2)$$

in-stead of the naive product $A = W^T W$ of the instance-pattern matrix W, where $D(A)$ is a diagonal degree matrix defined as $D(A)_{ii} = \sum_j A_{ij}$.
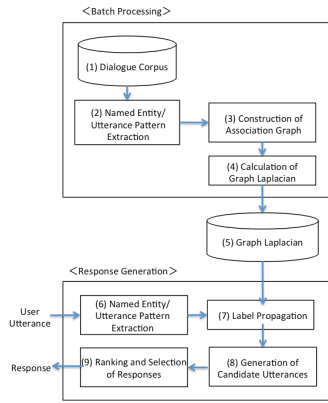
Figure 5: System architecture for utterance generation.

The regularized Laplacian has the effect of reducing the self-reinforcement by removing the contribution from the self-loops.

The procedure for generating responses to user utterances goes as follows. First, the named entities and an utterance pattern are extracted from the last utterance and the word and utterance pattern nodes in the association graph are matched to them if there are some that have the same word or utterance pattern. If these nodes are found, we assign a 1 as their initial score. For the other nodes that do not match, we assign a 0 as their initial score. We may take into consideration the history of the utterances before the last utterance as well. Let τ be the length of the turns that an utterance appeared in the past, i.e., $\tau = 1$ for the last utterance. Then, we extract the words and utterance patterns from those past utterances and search the association graph for word nodes or utterance patterns that match them. If some nodes are found, we assign $\lambda^{\tau-1}$ as their initial scores, where $\lambda \in (0, 1]$ is the decay rate. In practice, we limit τ to some extent T, such as $\tau \leq T$. By denoting the initial scores on the association graph as F_0, it is recursively spread using the following equation,

$$F_{t+1} = \alpha(-L)F_t + (1-\alpha)F_0, \qquad (3)$$

where a parameter $\alpha \in [0, 1)$ controls the contribution from the seeds and the graph structure. The contribution from the graph structure becomes dominant as the value of α approaches 1. The recursion is continued until the F_t score converges to F. In practice, this procedure is truncated within a finite number of recursions.

Let the resulting scores for each word w and utterance pattern p be $F(w)$ and $F(p)$. We define the

score for an utterance s generated from an utterance pattern p and words $\{w_i | i = i_1, i_2, \ldots, i_n\}$ as

$$F(s) = \frac{1}{n} \sum_{k=1}^{n} F(w_{i_k})F(p). \qquad (4)$$

We output the utterance s^*, which has the highest score among the generated utterances. There are cases where the utterance with the highest score already appeared in the current context of the dialogue, so we choose the utterance with the next highest score, and this is repeated until a new utterance is found.

4 Evaluation

We evaluated the performance of the proposed method by conducting a subjective assessment using crowdsourced workers. As the baseline for the evaluation, we adopted a well-known chat system in Japanese provided by NTT DoCoMo, Inc. The system is available through the Web API[5]. We call this chat system the baseline dialogue system in the paragraphs that follow.

We recruited twelve native Japanese-speaking crowd-workers in their 20's to 40's (seven females and five males). Each subject was presented five types of dialogues made for the same given seed utterance. We prepared 26 seed utterances as explained in the following paragraphs.

We selected topics from the top ten rankings of query keywords in Japan in 2014 provided by Google [6]. There were 55 different Japanese keywords among them. We limited them to 27 keywords from the genres of current affairs news, sports, entertainment, and TV programs which were covered by the corpus we constructed. However, our dialogue system failed to generate a response for one of them, so we omitted that keyword. Thus, we selected 26 keywords, among which 6 were from the field of current affairs, 6 from sports, 6 from entertainment, 2 from TV animations, and 6 from TV dramas (Tab. 1). As a seed utterance for each keyword, we picked the first sentence from the Wikipedia page with the given keyword as its title.

Starting from a seed utterance, we produced a dialogue using a pair of participants taking turns.

Current affairs news	Nobel prize, Mt. Ontake, Haruko Obokata, Mamoru Samuragochi, Academy Awards, iPhone6
Sports	Asian Games, Yuzuru Hanyu, Kei Nishikori, Seiko Yamamoto, 2014 Winter Olympics, Mao Asada
Entertainment	Zawachin, May J., Takako Matsu, Sota Fukushi, Kanna Hashimoto, Japan Electric Union
Animations	Yokai Watch, Frozen
Dramas	Hanako to Anne, Massan, Sorry youth!, Gochisousan, Ken Takakura, First class

Table 1: List of keywords used for evaluation.

We produced five types of dialogues as indicated in Tab. 2. Each dialogue contained ten utterances including the seed utterance. In the cases of dialogues produced by human and system participants, the seed utterance was regarded as created by the human participant. For the cases of dialogues between two human participants, males in 30's and 40's were used separately from workers in charge of the assessment. One participant in their 30's was involved in all the type (a), (b) and (c) dialogues to avoid any fluctuation caused by the difference in participants. The type (a) dialogues were prepared to check whether the workers honestly assessed the dialogues. In the case where the evaluation of a worker for the type (a) dialogues was unnaturally bad, we could detect that the worker was cheating. We also added the type (d) and (e) dialogues to exclude the influence of the choice of human participant for creating the dialogues. These five types of dialogues were produced for all 26 keywords, respectively.

a	human vs. human
b	human vs. proposed dialogue system
c	human vs. baseline dialogue system
d	proposed dialogue system vs. proposed dialogue system
e	baseline dialogue system vs. baseline dialogue system

Table 2: Types of produced dialogues

We specify an example of a dialogue produced in the type (b) dialogues in Tab. 3. The initial topic keyword is Kei Nishikori in this example. The system successfully extracted the keyword Kei Nishikori from the seed utterance and generated the utterance to inform what organization he is endorsed by. A

U	Kei Nishikori is a male Japanese professional tennis player from Shimane Prefecture.
S	Kei Nishikori is endorsed by Nissin Food, so Nissin Food is benefiting from a worldwide advertising effect.
U	You mean Nissin's Cup Noodles could become hot sellers?
S	The Cup Noodle's TV commercial "Real intention and stated reason" in which the comedian trio Dacho Club appeared in is better known, isn't it?
U	Is there such a TV commercial? I didn't know.
S	I've seen the Cup Noodle's TV commercial "Real intention and stated reason" in which Dacho Club appeared only once.
U	What kind of story was it?
S	If you get to know the stories background, you could better enjoy watching the Cup Noodle's TV commercial "Real intention and stated reason" in which Dacho Club appeared.
U	Recently there are many commercials that continue in a series like dramas.
S	It was interesting watching the Cup Noodle's TV commercial "Real intention and stated reason" in which Dacho Club appeared.

Table 3: Example of dialogue in type (b) dialogues. U represents the turns by a human participant and S represents the turns by the proposed dialogue system.

new keyword Cup Noodle was presented by a human participant, then the system expanded the topic by informing there is a series of interesting Cup Noodle's TV commercials that the comedian trio Dacho Club appeared.

We present these prepared dialogues to the workers in charge of the assessment by randomly shuffling the order of the dialogues for each keyword. We set up nine criteria to collect the workers' judgments on the dialogues as classified in Tab. 4. For the question concerning criterion C1, the workers answered in the order of their preference for the dialogues. The order was 1 for the best one and 5 for the worst. While for the question concerning criterion C2, the workers answered using a 4-point Likert scale: 4 for strongly agree, 3 for agree, 2 for disagree, 1 for strongly disagree. For the rest of the criteria, the workers answered using 6-point Likert scale: 6 for excellent, 5 for good, 4 for rather good, 3 for rather poor, 2 for poor, and 1 for terrible. It

must be noted that criteria C2 and C7 are related and they are also used for checking the reliability of the judgments by the workers.

A Kruskal-Wallis test was performed on the results of the judgments by the workers to see whether there were statistically significant differences in the distributions of the scores selected by the workers for the five types of dialogues. The results are indicated in the notched box plots for each criterion in Fig. 6 – 14. It must be noted that we divided the workers into two groups with six workers to check the influence of the selection of workers. The boxes for dialogue types (a) to (e) are placed from the left to right in each plot. Although slight deviations are seen, the plots for the two worker groups basically agree with each other, which confirms the reliability of the results of the judgments by the workers. In addition, we found the type (a) dialogues have the best assessment for all the criteria with only slight deviations; there are cases where the boxes even collapse as seen in Figs. 6, 7, and 14. The plots in Figs. 7 and 12 do not qualitatively contradict each other as well. Thus, we can confirm all the workers earnestly conducted their evaluations.

C1	Personal preference for dialogues (like or dislike)
C2	Quality of expanding topics as a chat
C3	Quality of naturalness of utterances as a chat
C4	Quality of interest of content in utterances
C5	Quality of usefulness of information in utterances
C6	Quality of naturalness of continuity in two consecutive utterances
C7	Quality of continuity from topic to topic
C8	Grammatical appropriateness of utterances in Japanese
C9	Semantic appropriateness of utterances

Table 4: Criteria for assessment

The type (b) dialogues were judged as the highest next to the type (a) dialogues, outperforming other types of dialogues in most of the plots. The widths and shifts of the notches of the boxes indicate that the type (b) dialogues were statistically significantly superior to the other types of dialogues from

(c) to (e), showing the effectiveness of our proposed method. Qualitatively the same plots were observed for other criteria as well. Surprisingly, in criteria C4 and C5, the type (d) dialogues gained a better assessment than the type (c) dialogues; nevertheless the type (d) dialogues were made only by the systems and the type (c) dialogues involved the human participant. This shows our method is especially superior in generating interesting and informative utterances compared to the baseline dialogue system.
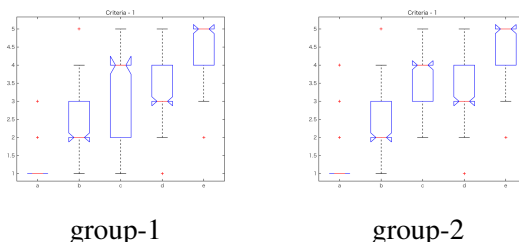

group-1 group-2
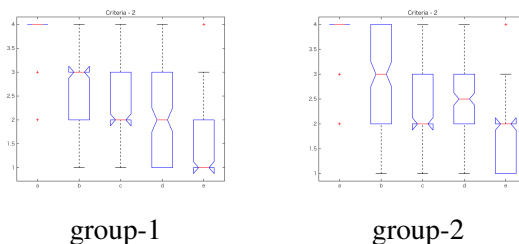
Figure 6: Plots of Kruskal-Wallis test for criterion C1.

group-1 group-2

Figure 7: Plots of Kruskal-Wallis test for criterion C2.
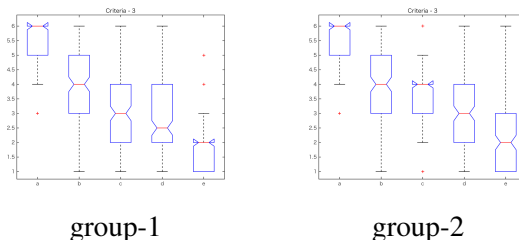

group-1 group-2

Figure 8: Plots of Kruskal-Wallis test for criterion C3.

5 Related Works

There are several modeling in the data driven approach. There is a statistical machine translation modeling for generating chat responses to the user utterances (Ritter et al., 2011). In this approach, generating a response to an input utterance is regarded as a mapping in translation. Collaborative filter modeling was also investigated, where responses are selected in terms of the user preference (Jafarpour and Burges, 2010). Making use of
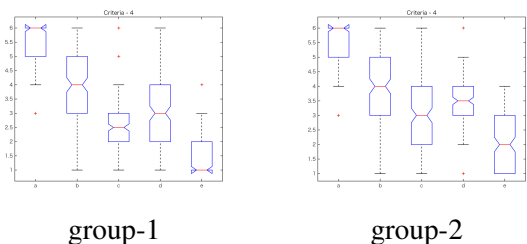
group-1   group-2

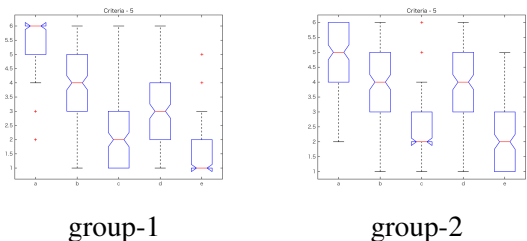Figure 9: Plots of Kruskal-Wallis test for criterion C4.

group-1   group-2

Figure 10: Plots of Kruskal-Wallis test for criterion C5.

group-1   group-2
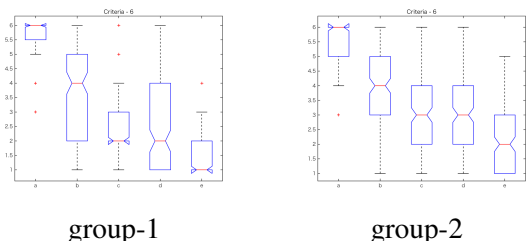
Figure 11: Plots of Kruskal-Wallis test for criterion C6.

group-1   group-2
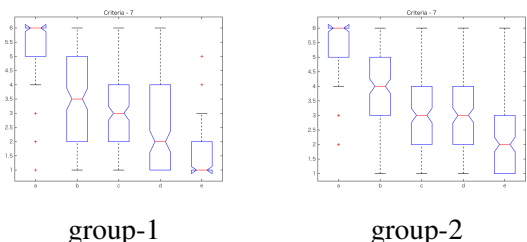
Figure 12: Plots of Kruskal-Wallis test for criterion C7.

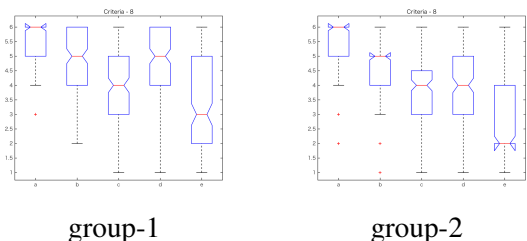group-1   group-2

Figure 13: Plots of Kruskal-Wallis test for criterion C8.
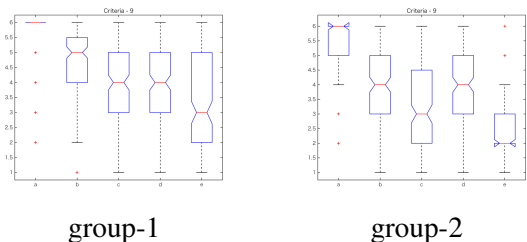

group-1   group-2

Figure 14: Plots of Kruskal-Wallis test for criterion C9.

recently raising crowdsourcing as a novel dynamical resource has also been proposed (Bessho et al., 2012).

6 Conclusion

We proposed a novel graph-based approach in this paper for the generation of system utterances in open-domain dialogue systems. Being different from ordinary statistical approaches, which basically select system responses from the utterances in a dialogue corpus that match an input utterance in surface-level similarity, utterances that semantically match an input utterance are generated by making use of the label propagation algorithm over an association graph of words and utterance patterns extracted from a dialogue corpus in the proposed approach. Thus, it is possible to generate non-trivial utterances that do not match the input utterances in surface-level and the topics of a dialogue can be expanded naturally.

We also proposed a framework for effectively collecting utterances and denoting the annotations simultaneously by making use of crowdsourcing and cloud open collaboration platforms. This framework is considered to have an advantage over the frameworks that make use of microblogs or Wikipedia in that we can control the quality of the contents of a dialogue corpus.

We implemented the proposed algorithm by constructing a considerably large dialogue corpus. The subjective evaluation was performed by using crowdsourcing workers and the effectiveness of the proposed approach was confirmed.

Our future work will focus on creating a dialogue act classifier making use of the annotations of the constructed dialogue corpus and integrating the filtering of responses so that they are in accordance with the recognized dialogue act of a previous utterance. Then it is expected that we can generate more natural responses than the current system can.

The framework of the label propagation can also be extended by adding more layers to the association graph, such as adding a layer of topic nodes. Then, the expansion of topics can be controlled by specifying the target topics that a system wants to move to.

References

Fumihiro Bessho, Tatsuya Harada, and Yasuo Kuniyoshi. 2012. Dialog system using real-time crowdsourcing and twitter large-scale corpus. In *SIGDIAL '12*, pages 227–231.

Maxine Eskenazi, Gina-Anne Levow, Helen Meng, Gabriel Parent, and David Suendermann. 2013. *Crowdsourcing for Speech Processing: Applications to Data Collection, Transcription and Assessment*. Wiley Publishing, 1st edition.

Sina Jafarpour and Christopher J.C. Burges. 2010. Filter, rank, and transfer the knowledge: Learning to chat. Technical Report MSR-TR-2010-93, July.

Mamoru Komachi, Shimpei Makimoto, Kei Uchiumi, and Manabu Sassano. 2009. Learning semantic categories from clickthrough logs. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 189–192, Suntec, Singapore, August. Association for Computational Linguistics.

Walter S. Lasecki, Ece Kamar, Dan Bohusa, and Eric Horvitz. 2013. Conversations in the crowd: Collecting data for task-oriented dialog learning. In *Workshop at Conference on Human Computation and Crowdsourcing 2013*.

Margaret Mitchell, Dan Bohus, and Ece Kamar, 2014. *Proceedings of the INLG and SIGDIAL 2014 Joint Session*, chapter Crowdsourcing Language Generation Templates for Dialogue Systems, pages 172–180.

Alan Ritter, Colin Cherry, and William B. Dolan. 2011. Data-driven response generation in social media. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 583–593, Stroudsburg, PA, USA. Association for Computational Linguistics.

Masahiro Shibata, Tomomi Nishiguchi, and Yoichi Tomiura. 2009. Dialog system for open-ended conversation using web documents. *Informatica*, 33(3):277–284.

Hiroaki Sugiyama, Toyomi Meguro, Ryuichiro Higashinaka, and Yasuhiro Minami, 2013. *Proceedings of the SIGDIAL 2013 Conference*, chapter Open-domain Utterance Generation for Conversational Dialogue Systems using Web-scale Dependency Structures, pages 334–338. Association for Computational Linguistics.

Dengyong Zhou, Olivier Bousquet, Thomas N. Lal, Jason Weston, and Bernhard Schölkopf. 2004. Learning with local and global consistency. In S. Thrun, L.K. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, pages 321–328. MIT Press.