Generating Cafeteria Conversations with a Hypernetwork Dialogue Model

지도교수 : 장병탁

이 논문을 공학학사 학위 논문으로 제출함.

2013년 12월 3일

서울대학교 공과대학

컴퓨터공학부

오준혁

2013년 12월

Generating Cafeteria Conversations with a Hypernetwork Dialogue Model

Jun-Hyuk Oh [1] Hyo-Sun Chun [1] Byoung-Tak Zhang [1,2]

[1]School of Computer Science and Engineering
[2]Cognitive Science and Brain Science Programs
Seoul National University, Seoul 151-744, Korea

{jhoh, hschun, btzhang}@bi.snu.ac.kr

Abstract. This paper introduces a data-driven dialogue system that generates the cashier's response to the customer's order. The proposed system learns dialogue rules from the corpus without prior knowledge which manage dialogue flows. The support vector machine combined with bag-of-words model is used to recognize and classify dialogue act (DA) of the customer. The hypernetwork dialogue model is used to manage dialogue flows and generate responses. The experimental results show that proposed system can generate appropriate dialogues without dialogue rules and the performance of dialogue generation is significantly improved as the data grows.

1 Introduction

These days, in the field of machine learning and natural language processing, many researchers are studying sentence generation [1], [2]. However, many of these models use a lot of prior knowledge such as a specific grammar to generate sentences. These approaches are not scalable because they are bounded by their specific language or a specific situation. For example, a method to generate sentences based on several Korean grammar rules cannot be applied to other language. There are also many researches about dialogue management (DM) [12-14]. This is a challenging problem because a model is required to not only recognize the speech act of a user and also generate appropriate dialogues. So, in order to reduce the complexity, most of works define a set of rules or dialogue flows with prior knowledge. In [12], for example, they provide a script language in which the dialogue generation rules can be coded. Thus, developers need to foresee all possible situations in order to code the dialogue rules. In other words, if a new situation occurs, developers should modify the rules manually. Therefore, instead of these approaches, data-driven approaches are more advisable so that many researches are in progress today [4], [8], [10].

This paper describes a data-driven dialogue system in cafeteria situations. The system plays a cashier's role by automatically recognizing a customer's dialogue act (DA) and responding to the customer. The corpus is composed of Korean dialogues between customers and cashiers. The system consists of two parts: DA classification

and dialogue generation. In the DA classification part, Bag-of-words (BoW) model is used for feature extraction of spoken sentences, and a Support Vector Machine (SVM) classifier is used to classify them. In the dialogue generation part, a hypernetwork dialogue model learns the flow of DA sequences from the corpus and predicts the next DA in the given situation. If the predicted DA is one of cashier side DAs, an appropriate response is retrieved from the corpus corresponding to the predicted DA. Otherwise, the system requires the user to input a new spoken sentence of the customer side. The architecture of the system is illustrated in Fig. 1. The proposed method does not need prior knowledge such as a set of dialogue rules because the system learns the rules from the corpus. The experimental results show that it is possible to manage dialogues without prior dialogue rules based on a hypernetwork dialogue model.

The rest of this paper is organized as follows. In Section 2, DA classification is described. Dialogue generation based on a hypernetwork dialogue model is described in Section 3. In Section 4, experimental results are presented. Finally, we conclude with summary and future works in Section 5.

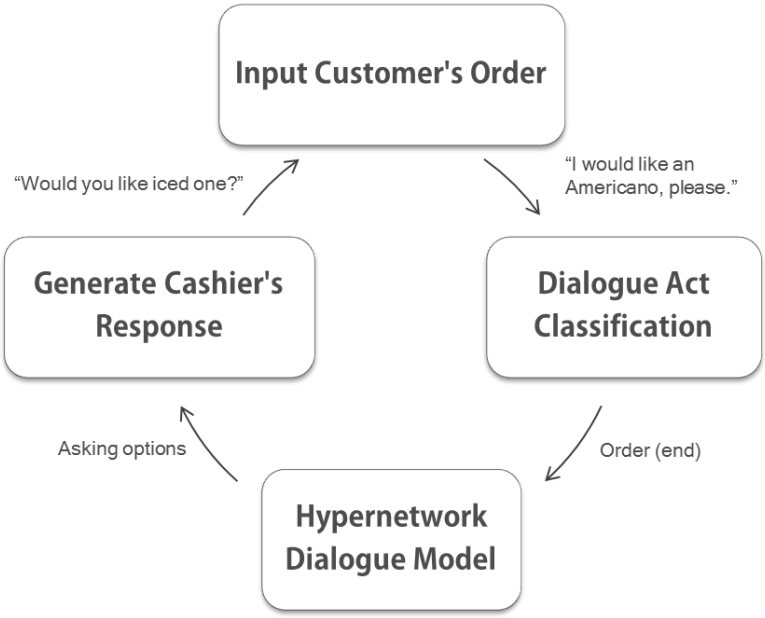

Fig. 1. Architecture of the dialogue system: First, a user inputs an order at the customer side. Next, the system recognizes DA of the input sentence. Then, a hypernetwork dialogue model predicts the next DA. If the predicted DA is one of cashier side DAs, an appropriate response is retrieved from the corpus.

2 Dialogue Act Classification

Table 1. Dialogue acts in cafeteria situations

DA	Role	Example
Greeting	Both	Hello.
Inducement to order	Cashier	Can I take your order?
Order (End)	Customer	I would like an Americano, please.
Order (Continue)	Customer	I would like a Café latte and
Payment information	Cashier	The total is 15,000 won.
Asking options	Cashier	Would you like iced one?
Deciding options	Customer	No. I would like an iced one.
Order Confirmation	Cashier	One Americano and two Café latte.
Question about menu	Customer	Is there café mocha?
Modification on order	Customer	I would like to change it to a Café latte.
Asking packing	Cashier	Here or to go?
Menu information	Cashier	Ice flake is 9,000 won.
Question about payment	Customer	Is it 8,000 won?
Sign request	Cashier	Sign on the screen, please.
Yes/No	Both	Yes.
Backchannel	Both	Yes.
Thanks	Cashier	Thanks.
Paying	Customer	Here you are.
Begin	-	(dummy for the system)
End	-	(dummy for the system)

Dialogue act (DA) is the meaning of a spoken sentence at the level of illocutionary force [9]. For DA classification, DAs in cafeteria situations are defined (Section 2.1). Then, BoW model for feature extraction and a SVM classifier for learning algorithm are used to classify a new sentence into pre-defined DA (Section 2.2).

2.1 Dialogue Acts in Cafeteria Situations

DAs in cafeteria situations are bounded [6]. Based on this idea, we define a set of DAs that commonly occur in cafeteria situations described in Table 1. DA 'Begin' and 'End' are dummies for the system. The first DA is always 'Begin', and the last one is 'End'. Note the difference between 'Yes/No' and 'Backchannel'. The spoken sentence 'Yes' is belongs to 'Yes/No' when it is used to respond to a question like 'Would you like iced one?' On the other hand, 'Yes' can also belong to 'Backchannel' when it is used to agree with the partner without meaning.

2.2 Prediction of Dialogue Acts

BoW model is used to extract feature of sentences and a SVM classifier is used to predict dialogue acts of given spoken sentences. In BoW model, a dictionary T is constructed from all of the distinct words in the corpus as follows:

$$T = \{w_1, w_2, \cdots, w_n\}$$

$w_1, w_2, \cdots, w_n$ are distinct words in the corpus. Based on the dictionary T, a sentence S is represented as a n-dimensional vector of word frequencies as follows:

$$< f(S, w_1), f(S, w_2), \cdots, f(S, w_n) > \in R^n$$

$$f(S, w_i) \equiv \text{the number of word } w_i \text{ in the sentence } S$$

After labeling DA for each sentence according to Table 1, a SVM with linear kernel is trained to predict DA of given sentences. Each training instance $< x_i, y_i > \in$ D is represented as BoW model x_i and its DA y_i.

3 Dialogue Generation based on Hypernetwork Dialogue Model

Before generating dialogues, a hypernetwork dialogue model predicts the following DA (Section 3.1, 3.2). If the predicted DA is one of customer side DAs, the system requires the user to input an order and classify it into DA, which is discussed in Section 2. Otherwise (if predicted DA is one of cashier side DAs), the system generates an appropriate response by retrieving the most probable response from the corpus corresponding to the predicted DA (Section 3.3). This cycle is repeated until the predicted DA reaches 'End'.

3.1 Hypernetwork Model

Hypernetwork model is an extension of graphical model [3]. By the definition of hypergraph, an edge can connect more than two vertices, which is called a hyperedge. A hypernetwork model is represented as $\boldsymbol{H} = (\boldsymbol{X}, \boldsymbol{E}, \boldsymbol{W})$ where $\boldsymbol{X} = \{x_1, x_2, \cdots, x_n\}$, $\boldsymbol{E} = \{E_1, E_2, \cdots, E_{|\boldsymbol{E}|}\}$, and $\boldsymbol{W} = \{w_1, w_2, \cdots, w_{|\boldsymbol{E}|}\}$. $\boldsymbol{X}, \boldsymbol{E}, \boldsymbol{W}$ are sets of vertices, hyperedges, and weights, respectively. Each hyperedge is represented as $E_i = \{x_{i_1}, x_{i_2}, \cdots, x_{i_{|E_i|}}\}$ where $|E_i|$ is the cardinality of the hyperedge. A Hypernetwork model can be used as a probabilistic associative memory to store a data $\boldsymbol{D} = \{x^{(n)}\}_{n=1}^{N}$ where $x^{(n)}$ is n-th pattern to store. The energy of the hypernetwork is defined as follows:

$$\varepsilon(x^{(n)}; \boldsymbol{W}) = -\sum_{i=1}^{|\boldsymbol{E}|} w_{i_1 i_2 \cdots i_{|E_i|}} x_{i_1}^{(n)} x_{i_2}^{(n)} \cdots x_{i_{|E_i|}}^{(n)}$$

$x_{i_1}^{(n)} x_{i_2}^{(n)} \cdots x_{i_{|E_i|}}^{(n)}$ are vertices connected by the hyperedge $x^{(n)}$ and $w_{i_1 i_2 \cdots i_{|E_i|}}$ is the weight of the hyperedge. $\boldsymbol{W}$ is a set of weights of hyperedges and represents the parameters for the hypernetwork model. The probability of the data generated from the hypernetwork is given as Gibbs distribution:

$$\mathrm{P}(x^{(n)} | \boldsymbol{W}) = \frac{1}{Z(\boldsymbol{W})} exp\{-\varepsilon(x^{(n)}; \boldsymbol{W})\}$$

$\exp\{-\varepsilon(x^{(n)}; \boldsymbol{W})\}$ is called Boltzmann factor and the normalizing term $Z(\boldsymbol{W})$ is defined as :

$$Z(\boldsymbol{W}) = \sum_{x^{(m)}} exp\{-\varepsilon(x^{(n)}; \boldsymbol{W})\} = \sum_{x^{(m)}} exp\left\{\sum_{i=1}^{|\boldsymbol{E}|} w_{i_1 i_2 \cdots i_{|E_i|}} x_{i_1}^{(m)} x_{i_2}^{(\mathrm{m})} \cdots x_{i_{|E_i|}}^{(\mathrm{m})}\right\}$$

3.2 Prediction of Dialogue Act Sequence

It is assumed that a DA sequence follows Markov process. Markov process satisfies a condition that a transition to the next state is solely dependent on the current state and independent on the full history. More generally, k-order Markov process follows a condition that a transition depends on the k-nearest states. Given this assumption, the probability of a DA sequence $U_1, U_2, \cdots, U_n$ is [7]:

$$\begin{aligned} P(U_1, U_2, \cdots, U_n) &= P(U_1)P(U_2|U_1)P(U_3|U_1, U_2)\cdots P(U_n|U_1, U_2, \cdots, U_{n-1}) \\ &= \prod_{i=1}^{n} P(U_i|U_1, \cdots, U_{i-1}) \approx \prod_{i=1}^{n} P(U_i|U_{i-k}, \cdots, U_{i-1}) \end{aligned}$$

Chain rule is applied to the first line, and Markov process assumption is applied to the next equation approximating $P(U_i|U_1, \cdots, U_{i-1})$ to $P(U_i|U_{i-k}, \cdots, U_{i-1})$, which means that a random variable U_i is dependent on the k-nearest DAs in the DA sequence. Since the amount of the corpus is small, k value is set to 1 or 2 as described in Fig. 2. Thus, given DA sequences as $U_1, U_2, \cdots, U_n$, the probability of the next DA U_{n+1} follows the equation:

$$P(U_{n+1}|U_1, U_2, \cdots, U_n) \approx P(U_{n+1}|U_{n-1}, U_n) \approx P(U_{n+1}|U_n)$$

The system predicts the next DA according to $P(U_{n+1}|U_{n-1}, U_n)$. When the probability $P(U_{n-1}, U_n)$ is zero (due to the lack of data), $P(U_{n+1}|U_{n-1}, U_n)$ is replaced by $P(U_{n+1}|U_n)$.

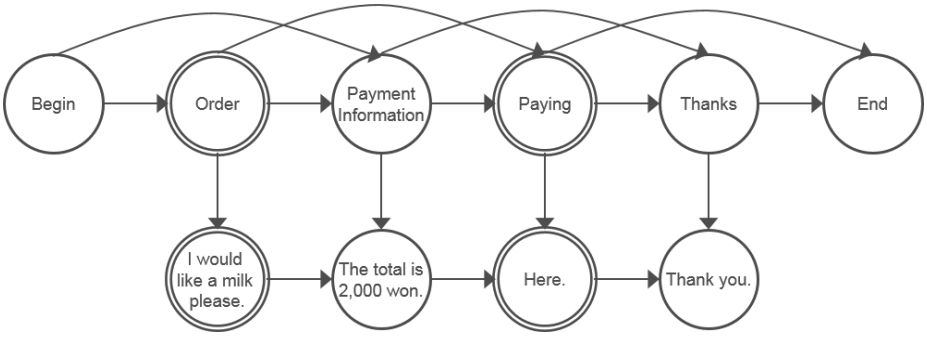

Fig. 2. Graphical representation of dialogue process. The upper line represents a DA sequence that follows 2-order Markov process, and the other line represents sentences. Double-lined and single-lined circles indicate customer and cashier part respectively.

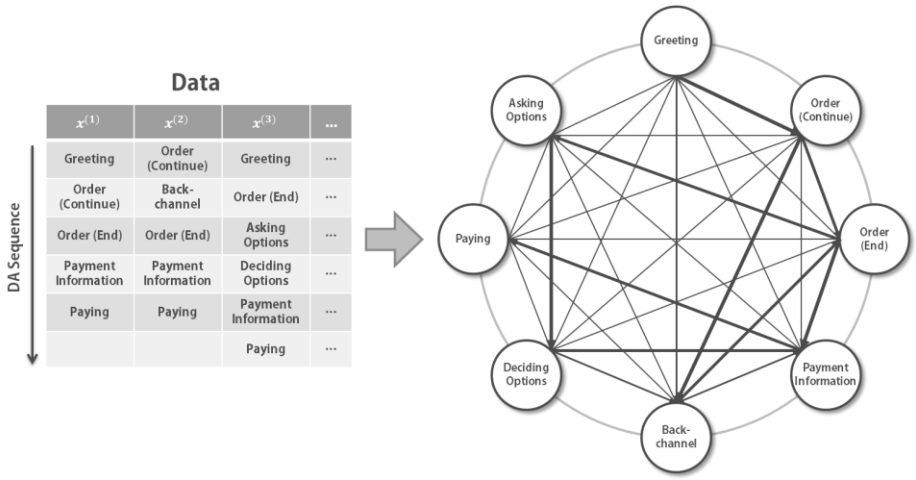

Fig. 3. Hypernetwork dialogue model. The hypernetwork stores DA sequences from the data. The frequency of a DA sequence is represented as a weight of a hyperedge. A hypernetwork dialogue model is capable of predicting DA sequences based on its hyperedges.

A hypernetwork dialogue model is constructed from the corpus to calculate the conditional probability. In this model, as illustrated in Fig. 3, each vertex represents each DA, and each weight of hyperedges represents the frequency of DA sequence in the corpus. In other words, the probability distribution of DA sequences is represented by a hypernetwork dialogue model using a population of hyperedges and their weigths. With the model, $P(U_{n+1}|U_{n-1}, U_n)$ can be calculated from 3-order hyperedges and $P(U_{n+1}|U_n)$ can be calculated from 2-order hyperedges.

3.3 Sentence Generation

If the predicted DA is one of cashier side DAs, the system generates a response by retrieving it from the corpus. Since the DA is determined, the system only considers sentences corresponding to the predicted DA in the corpus. Thus, DA can be viewed as a latent variable because it is not observed and reduces the complexity of data. Our previous work [10] is used to select the most probable sentence S^* which is defined as follows:

$$S^* = \arg\max_{S \in D_U} P(S|S')$$

S' is the previous sentence (customer's order in most cases). D_U is a set of spoken sentences in the corpus which correspond to the predicted DA. Similar to factored language model [11], it is assumed that each sentence can be viewed as a set of words.

$$\mathrm{S} = \{w_1^S, w_2^S, \cdots, w_n^S\}$$

$$S' = \{w_1^{s'}, w_2^{s'}, \cdots, w_m^{s'}\}$$

So, the probability $\mathrm{P}(S|S')$ is defined as:

$$\mathrm{P}(S|S') = \mathrm{P}\left(w_1^s, w_2^s, \cdots, w_n^s \middle| w_1^{s'}, w_2^{s'}, \cdots, w_m^{s'}\right)$$

$$\approx \prod_{i=1}^{n} P\left(w_i^s \middle| w_1^{s'}, w_2^{s'}, \cdots, w_m^{s'}\right)$$

To estimate the probability, the number of occurrence of words between consecutive sentences is measured. However, $P\left(w_i^s \middle| w_1^{s'}, w_2^{s'}, \cdots, w_m^{s'}\right)$ is zero in most cases because of Zipf's law. So, the generalized backoff method [11] is used as smoothing technique in order to estimate the probability.

4 Experimental Results

Korean dialogue corpus was collected via recording conversations from two cafeterias and translating them into text form. The corpus consists of total 130 episodes (a complete form of conversation) and 706 sentences in total. The number of sentences for each DA is given in Table 2.

Table 2. The number of sentences in the corpus

DA	# of sentences
Greeting	20
Inducement to order	19
Order (End)	131
Order (Continue)	68
Payment information	143
Asking options	30
Deciding options	39
Order confirmation	66
Question about menu	16
Modification on order	9
Asking packing	2
Menu information	22
Question about payment	6
Sign request	29
Yes/No	51
Backchannel	231
Thanks	16
Paying	8
Total	706

4.1 Dialogue Act Classification

The test data is created by randomly sampling 10% of 706 sentences. For the rest 90%, we trained a linear SVM on the different size of the training data varying from 10% to 90%. This trial is independently repeated for 20 times with dictionary size fixed at 450. As shown in Fig. 4, the classifiers trained on larger training data get better performance. The best accuracy is 80.14% when the training data is 90% of total sentences.

The effect of dictionary size is shown in Fig. 5. Intuitively, the accuracy increases as the dictionary size increases. The interesting point is that the accuracy is high enough when the size of dictionary is about 100. Considering that the total number of distinct words in the corpus is 461, it can be inferred that DA can be determined by only a small set of words. It is observed that few words such as 'please', 'like', and 'yes' are discriminative words in DA classification.

We also evaluated the classification error for each DA shown in Table 3. 'Yes/No' shows the highest misclassification rate. Most of misclassified sentences were 'Yes' and they were misclassified into 'Backchannel'. As discussed in Section 2.1, 'Yes' can belong to both of 'Yes/No' and 'Backchannel'. The answer depends on the context. The similar problem occurs on 'Modification on order' and 'Deciding options' that also show high misclassification rate. Most of the misclassified sentences are like 'Café mocha, please' and misclassified into 'Order (End)'. In this case, the answer also depends on whether the customer has already ordered or not. But, the system considers only the current sentence without considering the previous context. If such high-level context were considered, the accuracy would be improved.

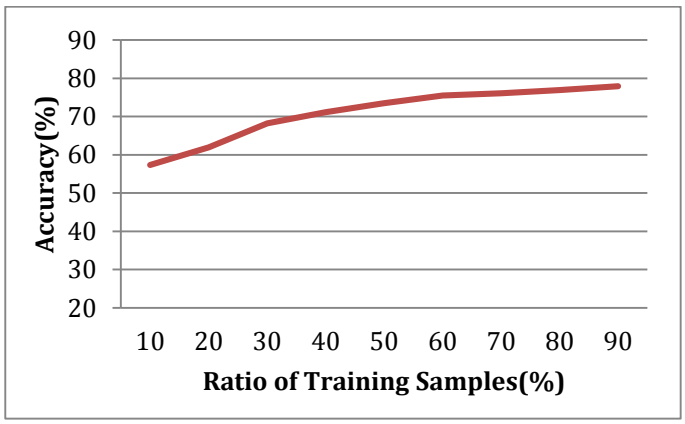

Fig. 4. DA classification accuracy according to the amount of training samples

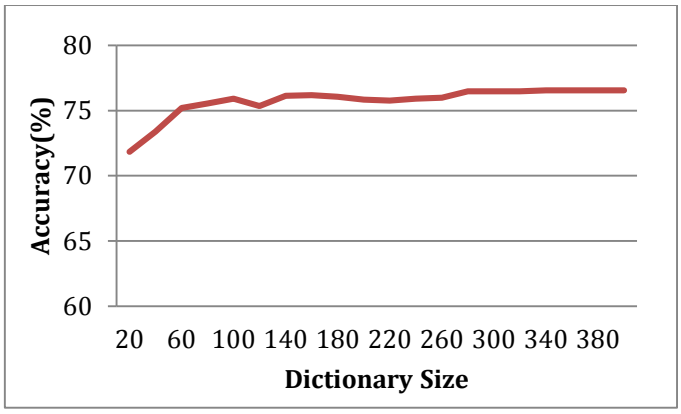

Fig. 5. DA classification accuracy according to the size of dictionary

Table 3. Classification error for each DA

DA	Error rate(%)
Yes/No	27.45
Modification on order	22.22
Deciding options	12.82
Menu information	4.55
Backchannel	1.73

Table 4. Example of dialogue generation

Role	Dialogue	Dialogue Act
Customer	안녕하세요. → Hello.	Greeting
Cashier	안녕하세요. → Hello.	Greeting
Customer	생체리 우유 하나랑요. → I would like a cherry milk and	Order (Continue)
Cashier	네. → Yes.	Backchannel
Customer	라즈베리 요구르트요. → a raspberry yogurt please.	Order (End)
Cashier	4,500 원이요. → The total is 4,500 won.	Payment information
Customer	여기 있습니다. → Here.	Paying
Cashier	감사합니다. → Thank you.	Thanks

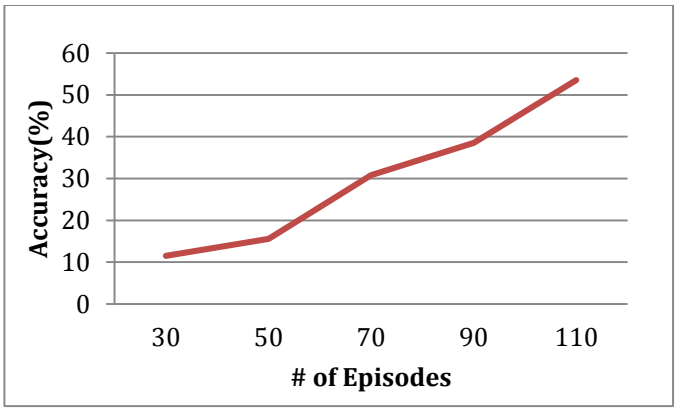

Fig. 6. Dialogue generation accuracy according to the number of training episodes

4.2 Dialogue Generation

An example of dialogue generation is shown in Table 4. A hypernetwork dialogue model determines the flow of DA sequence. The 'Customer' part is given as user input. The 'Cashier' part is automatically generated by the system.

The test set is composed of randomly sampled 20 episodes of total 130 episodes. For the rest 110 episodes, we measured the accuracy by varying the amount of the training data. The results of 10 independent trials are averaged. In this experiment, if one of generated responses is out of the context, the whole episode is regarded as incorrect. Fig. 6 shows that the accuracy significantly increases as the number of episodes increases. Most of error cases are caused by DA misclassification discussed in Section 4.1. Considering 130 episodes are very little amount of data, it is expected for the system to get good result for large dataset.

5 Conclusion

This paper proposes a new method to generate dialogues in cafeteria situations based on a hypernetwork dialogue model. In this method, DAs in cafeteria situations are defined. Then, BoW model is used to extract feature of sentences, and a linear SVM is trained to classify a new spoken sentence into DA. Finally, a hypernetwork dialogue model is trained to predict DA sequence and appropriate responses are retrieved from the corpus. The experimental results show that the accuracy is significantly increased as the amount of the corpus grows. Although the result is not perfect, our method has advantages for two reasons. First, our method does not require a lot of prior knowledge for dialogue management such as dialogue rules. This is because the model can learn dialogue flows from the corpus. Second, our method can be easily applied to other languages, since both of BoW model and a hypernetwork dialogue model are not influenced by languages and grammars.

On the other hand, there are some future works to research further. First, high-level context should be considered in DA classification. As discussed in Section 4.1, even the same spoken sentence might belong to different DA depending on the context. If such high-level context were considered together, the result would be improved. Second, the proposed method also needs some prior knowledge in that DAs should be defined manually. This makes it difficult to apply this method to other situation easily. It would be more scalable to model this problem in unsupervised way (without manually labeling DAs). Finally, the proposed method generates spoken sentences by retrieving them from the corpus with a whole sentence as a unit. This approach cannot deal with detailed information such as specific menus or prices. Therefore, if a model learns the association between words and generate spoken sentences with a word as a unit, we expect it to perform better.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2010-0017734-Videome) and was supported in part by KEIT grant funded by the Korea government (MKE) (KEIT-10044009).

References

1. Y Shim and M Kim, Automatic short story generator based on autonomous agents, *In Proceedings of PRIMA*, pp. 151-162, 2002.
2. E Reiter and R Dale, Building Natural-Language Generation Systems, *Cambridge University Press*, 2000.
3. Zhang, B.-T., Hypernetworks: A molecular evolutionary architecture for cognitive learning and memory, *IEEE Computational Intelligence Magazine*, 3(3) pp. 49-63, 2008.
4. N McIntyre and M Lapata, Learning to Tell Tales: A Data-driven Approach to Story Generation, *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pp. 217-225, 2009.
5. Lee, D.-H., Unsupervised modeling of user actions in a dialogue corpus, *Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5061-5064, 2012.
6. Seok-Jae Choi, The processing of Language for the Dialogue System –Especially for an order of a fast-food and coffee, *Journal of the Korean Language and Literature Society* v.31, pp. 253-279, 2012.
7. Stolcke, Andreas, at al., Dialogue act modeling for automatic tagging and recognition of conversational speech, *Computational Linguistics*, v.26 n.3, pp. 339-373, 2000.
8. Bado Lee, Ho-Sik Seok, Byoung-Tak Zhang, Sentence Generation Capability of a Random Hypergraph based Language Model considering Word-distance, *Journal of the Korea Information Science Society*, v.37, 2(C), pp. 278-281, 2010.
9. Austin, J. L., How to do Things with Words. *Clarendon Press, Oxford*, 1962.
10. Ha-Young Jang, Byoung-Tak Zhang, Context Analysis based on Hypergraph Language Model for Automatic Storytelling, *Journal of the Korea Information Science Society*, v.38, 2(B), pp. 291-294, 2011.

11. Bilmes, J.A. and Kirchhoff, K., Factored language models and generalized parallel backoff. *In HLT-NAACL 2003: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, Association for Computational Linguistics, pp. 4–6, 2003.
12. Boye, Johan, Dialogue management for automatic troubleshooting and other problem-solving applications, *Proceedings of 8th SIGdial Workshop on Discourse and Dialogue*, 2007.
13. Person, N., Graesser, A. C., Harter, D., Mathews, E., & Tutoring Research Group, Dialogue move generation and conversation management in AutoTutor. *In Workshop Notes of the AAAI'00 Fall Symposium on Building Dialogue Systems for Tutorial Applications*, pp. 45-51, 2000.
14. Bohus, Dan, and Alexander I. Rudnicky, RavenClaw: Dialogue management using hierarchical task decomposition and an expectation agenda, *CMU Computer Science Department, Paper 1392*, 2003.