Dialogue Breakdown Detection Using BERT with Traditional Dialogue Features

Hiroaki Sugiyama

Abstract Despite of the significant improvements of Natural Language Processing with Neural networks such as machine reading comprehensions, chat-oriented dialogue systems sometimes generate inappropriate response utterances that cause dialogue breakdown because of the difficulty of generating utterances. If we can detect such inappropriate utterances and suppress them, dialogue systems can continue the dialogue easily.

1 Introduction

Chatting with people is an important function of dialogue systems in building social relationships with users. This not only provides therapeutic and entertainment benefits but also plays an important role in drawing out the user's potential requirements and constructing a good relationship with the user. Furthermore, such conversational dialogue has the potential to improve the performance of task-oriented dialogue [1]. Thus, the construction of conversational dialogue systems (also called non-task oriented dialogue systems or chat-oriented dialogue systems) has recently gained attention [2–4].

The difficulty of developing chat-oriented dialogue systems is that such systems are required to respond to a very wide range of topics expressed by user utterances. Since it is still difficult for the current dialogue systems to continue outputting appropriate responses, utterances that cause the dialogue to collapse are often generated. It is assumed that the continuation of dialogue becomes easy when we can detect and suppress such problematic utterances.

In a previous dialogue breakdown detection challenge (DBDC), the author proposed a dialogue breakdown detection system that captures frequently appearing error patterns that are specific to each utterance generation approach [5]. The authors

H. Sugiyama (✉)
NTT Communication Science Laboratories, Kyoto, Japan
e-mail: h.sugi@ieee.org

E. Marchi et al. (eds.), *Increasing Naturalness and Flexibility in Spoken Dialogue Interaction*, Lecture Notes in Electrical Engineering 714,
https://doi.org/10.1007/978-981-15-9323-9_39

proved that traditional dialogue features such as dialogue-acts or sentence similarities calculated with word vectors are effective to such errors.

From the other viewpoint, machine reading comprehension with neural networks are so popular recently, and many significant improvements are frequently proposed. Especially, BERT achieves SOTA in many natural language processing tasks and has gained attention. BERT adopts pre-training task called Next Sentence Prediction (NSP) that evaluates the cohesion of sentence pairs. Since NSP task resembles dialogue breakdown detection, we expect that BERT improves dialogue breakdown detection performance. In this paper, we propose a novel dialogue breakdown detection method that combines BERT and traditional dialogue features, and examine the effectiveness of the additional features.

2 Systems

We utilize BERT [6] to detect dialogue breakdown. In addition to the original BERT, we introduce traditional dialogue features such as dialogue-act to improve the estimation performance. In this section, we explain the structure of our model and the details of additional features.

2.1 BERT with Additional Features

BERT is a Transformer-based method that achieves SOTA performances in many kinds of Natural Language Processing tasks [6]. The important advantage of BERT is that, once a transformer model is pre-trained with large-scale text corpus, we can fine-tune the pre-trained model with fewer data than the scratch.

When we adopt BERT to classification tasks, we use the final hidden state of Transformer corresponding to first token ([CLS]) as the aggregate sequence representation, to which we adopt feed-forward networks and softmax functions for final outputs (classification results). The feed-forward network is pre-trained with Next Sentence Prediction (NSP) task, where BERT predicts whether randomly chosen two sentences A and B are actually connected (A follows B) or not. Since this NSP task resembles our dialogue breakdown task, we expect that BERT improves the estimation performance. However, since BERT pre-trained models in public are not trained with dialogue corpus, the prediction possibly does not suit for our task. Besides, pre-training of the model with dialogue data is difficult because it requires a huge size of texts.

To overcome this difficulty, we concatenate additional dialogue features that represent the naturalness of dialogue flow to the aggregate representation. Figure 1 illustrates our model. Original BERT estimates only aggregate representation C in the left side. We utilize word (token) vectors T obtained from BERT to calculate

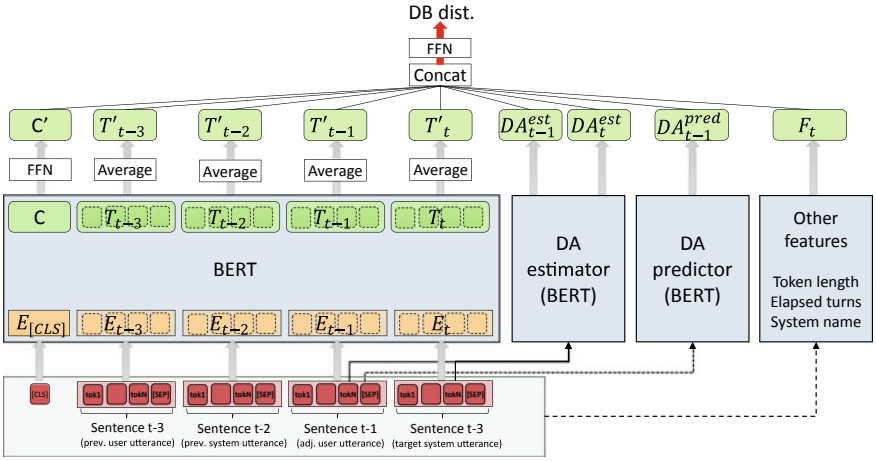

Fig. 1 Distribution of annotated winning rates between annotators

sentence vectors of the dialogue context. Dialogue acts are separately estimated and predicted using other BERT models, and are concatenated to the vector C and T.

2.2 Features

In this section, we explain the details of the dialogue features shown in Fig. 1.

2.2.1 Dialogue Act

Dialogue act represents the action types of each utterance such as *question* or *greeting*. This is a very common dialogue feature to represent dialogue flow [7], and is proved as an effective feature for dialogue breakdown detection [5]. For example, in DBDC, all of the systems sometimes respond with questions even when the user utterance is a question. Dialogue acts are useful to capture such kinds of errors. In this study, we utilize *estimated dialogue acts* of the target system utterance and adjacent user utterance, and in addition, we use *predicted dialogue acts* that are expected to be suitable for the next utterance after the user utterance. We use a dialogue acts definition proposed in [8], in which they categorize utterances into 33 dialogue acts.

We train dialogue acts estimator and predictor with regression of multi-label expressions (e.g.., [0,0,1,0,1,0,..]), since one utterance possibly contains multiple dialogue acts.

We train Japanese dialogue acts estimator and predictor with NTT's Japanese chat dialogue corpus (3680 dialogues) [4], using BERT with words that are tokenized with

sentencepiece. English dialogue acts estimator and predictor were trained with NTT's English situation dialogue corpus (4000 dialogues), using BERT with words.

2.2.2 Sentence Length

With the technology of the current dialogue system, it is difficult to estimate the consistency of the user utterance and the system utterance. Therefore, there is the problem that the longer the system utterance is, the greater the possibility that an unrelated element is included. In particular, DIT system of Japanese task tends to generate very long utterances, and thus, the utterance does not match the content of the user utterance. Here, we add token length of the target utterance to the features.

2.2.3 Number of Elapsed Turns

All three dialogue systems generate relatively appropriate utterances at the beginning of a dialogue, but the proportion of inappropriate utterances tends to increase as the dialogue proceeds. Therefore, the number of elapsed turns from the start of the dialogue is added to the features.

2.2.4 Sentence Embedding and Similarities

DIT and IRS systems in Japanese task sometimes generate system utterances with topics completely different from user utterances. DCM system in Japanese task sticks to a specific topic, and, as a result, there are many cases where an utterance with almost the same contents repeatedly occurs. In order to detect these errors, we adopt word-based sentence embedding features, the difference vectors of the sentence embedding, and sentence similarity features between system and user utterances and between the target and a previous system utterance. Sentence embedding is calculated as the average vector of the last layer of BERT corresponding the tokens in target sentences.

2.2.5 System Names

It is shown in previous DBDC papers that each dialogue system has a specific dialogue breakdown patterns [5]. To capture the system-dependent error patterns, we adopt a system name feature with one-of-k vector representation.

Table 1 Dataset distributed in DBDC English task [11]

	DBDC3 (eval)				DBDC4 (dev/eval)						
	CIC	TKTK	YI	IRIS	Bot001	Bot002	Bot003	Bot004	Bot005	Bot006	IRIS
No. of dialogues	100/50	100/50	100/50	100/50	39/46	38/33	42/47	41/38	2/2	6/7	43/27
No. of annotators	30	30	30	30	15	15	15	150	15	15	15

Table 2 Dataset distributed in DBDC Japanese task [11]

	DBDC1 (dev/eval)	DBDC2 (dev/eval)			DBDC3 (eval)			DBDC4 (dev/eval)			
	DCM	DCM	DIT	IRS	DCM	DIT	IRS	DCM	DIT	IRS	LiveComp
No. of dialogues	20/80	50/50	50/50	50/50	50	50	50	0/50	0/50	0/50	73/73
No. of annotators	30	30	30	30	30	30	30	15	15	15	30

3 Experiment

3.1 Dataset

3.1.1 Dataset for English Task

The DBDC organizers have distributed previous DBDC3 dataset as development dataset for English task [9]. Table 1 shows the statistics of the dataset. The target dialogue systems of DBDC4 are dialogue systems submitted for Conversational Intelligence Challgenge 2 (ConvAI2)[1] and IRIS [10], which are changed from those of DBDC3 (CIC, TKTK, IRIS and YI).

3.2 Dataset for Japanese Task

The DBDC organizers have distributed development dataset for Japanese task using previous DBDC1, DBDC2 and DBDC3 [9]. Table 2 shows the statistics of the dataset. DBDC4 task contains a new dialogue system group called LiveComp, which consists of four dialogue systems developed for Dialogue system live competition [12].

3.3 Experiment Settings

In this research, we examined the effectiveness of the features described in Sect. 2.2 by adding certain features to the case of original BERT (no additional features).

[1] https://github.com/DeepPavlov/convai/tree/master/data.

Table 3 Feature addition analysis in DBDC English task

	DBDC3-eval		
	Accuracy	MSE	JSD
Original BERT	0.5180	0.0231	0.0410
+Dialogue act(DA)	0.5250	0.0227	0.0409
+Sentence vector(SV)	0.5375	0.0225	0.0408
+SV +Sentence distance(Dis)	0.5055	0.0231	0.0408
+SV +Difference of SV(Diff)	0.5205	0.0226	0.0405
+System names(Sys)	0.5165	0.0225	0.0405
+Elapsed turns and sentence length(Other)	**0.5375**	0.0227	0.0405
+DA +SV	0.5265	0.0225	0.0403
+DA +SV +Dis +Diff	0.5175	0.0227	0.0406
+DA +SV +Dis +Diff +Sys + Other(ALL)	0.5315	**0.0222**	**0.0399**

Since importance is placed on distribution-related metrics (Mean Squared Error and JS divergence), we examined the features that minimize MSE.

We used DBDC1-dev, DBDC1-eval and DBDC2-dev, DBDC2-eval for the training data in Japanese task, and DBDC3-dev in English task. DBDC3-eval is used for evaluation data in both tasks. DBDC4-dev data is distributed but its size is very small. Since DBDC4-dev data shows similar behaviors of the performance with DBDC3-dev, we add DBDC4-dev to train data instead of using for validation data.

because this analysis adopted MSE as model selection and optimization function.

Each model with a certain feature is trained using Adabound optimization method [13]. For choosing models submitted to DBDC4 shown in Sect. 4, we use optuna[2] developed by Preferred Networks to search hyper-parameters (final lr and training batch size) and optimum feature sets. However, after the submission, we noticed that random seed is more dominant to find out the best performance of a certain feature set. Therefore, for the analysis shown in Sect. 3.4, we examine the feature sets with the following three parameter tuning steps. First, we tune *final lr* of Adabound with optuna. Second, using the optimum *final lr*, we search for better random seed randomly. Finally, using the best random seed, we tune *final lr* again around the firstly chosen value.

3.4 Result

Tables 3 and 4 show the result of the feature addition analysis of English and Japanese tasks. They illustrate that models with all the features achieved the best performance

[2] https://optuna.org/.

Table 4 Feature addition analysis in DBDC Japanese task

	DBDC3-eval		
	Accuracy	MSE	JSD
Original BERT	0.6018	0.0401	0.0765
+Dialogue act(DA)	0.5885	0.0400	**0.0757**
+Sentence vector(SV)	0.6012	0.0398	0.0762
+SV +Sentence distance(Dis)	0.6000	0.0397	0.0758
+SV +Difference of SV(Diff)	**0.6030**	0.0397	0.0758
+System names(Sys)	0.5982	0.0406	0.0781
+Elapsed turns and sentence length(Other)	**0.6030**	0.0404	0.0768
+DA +SV	0.6018	0.0399	0.0760
+DA +SV +Dis +Diff	0.6012	0.0399	0.0760
+DA +SV +Dis +Diff +Sys + Other(ALL)	0.5964	**0.0395**	0.0759

in MSE both English and Japanese tasks, and the models are superior to the original BERT. In the comparison of each feature, sentence vectors(SV) seems effective in both tasks.

On the other hand, Accuracy and JSD metrics are not consistent with MSE result,

4 Submitted Systems

We adopt models trained with all the features for submitted systems. We prepare four systems for each language with the combination of their training data (the use of DBDC4-dev), and metrics used for model selection (highest accuracy or lowest MSE). In addition, as extra trials after the competition, we evaluate another run that utilizes only DBDC4-dev for training.

Table 5 shows the result of DBDC4 English task. Run 4, which is trained only with DBDC3-dev and is selected as the best accuracy model, shows the best performance

Table 5 Submitted systems and extended trials for DBDC4-en

Runs	DBDC3-dev	DBDC4-dev	Metric	DBDC3-eval			DBDC4-eval		
				Acc	MSE	JSD	Acc	MSE	JSD
Run 1	✓	✓	MSE	0.521	0.0229	0.0410	0.488	0.0378	0.0732
Run 2	✓	✓	Accuracy	0.521	0.0230	0.0439	0.532	0.0376	0.0752
Run 3	✓	–	MSE	0.525	0.0223	0.0404	0.534	0.0360	0.0708
Run 4	✓	-	Accuracy	0.538	0.0225	0.0409	**0.556**	**0.0350**	**0.0692**
*EX 1	–	✓	MSE(RUN 3)	0.547	0.0233	0.0412	**0.601**	**0.0299**	**0.0580**

Table 6 Submitted systems and extended trials for DBDC4-ja

Runs	DBDC1,2-dev/eval	DBDC4-dev	Metric	DBDC3-eval			DBDC4-eval		
				Acc	MSE	JSD	Acc	MSE	JSD
Run 1	✓	✓	MSE	0.587	0.0414	0.0801	**0.584**	**0.0462**	**0.0953**
Run 2	✓	✓	Accuracy	0.605	0.0427	0.0809	0.572	0.0504	0.1013
Run 3	✓	–	MSE	0.598	0.0406	0.0779	0.480	0.0653	0.1259
Run 4	✓	–	Accuracy	0.605	0.0414	0.0795	0.444	0.0714	0.1360
*EX 1	–	✓	MSE(Run 3)	0.596	0.0421	0.0794	**0.599**	**0.0451**	**0.0763**

among Run 1–4. Although the English task of DBDC4-eval does not contain DBDC3 systems, Run 1 and 2 are lower performance than Run 3 and 4. Considering the model EX 1 trained only with DBDC4-dev is superior to the other settings (including other teams), DBDC4-dev is crucial for the training. This indicates that the dialogue systems' behaviors of DBDC3-eval resemble those of DBDC4-eval in the English task. We assume that the larger and more various training data of Run 1 and 2 make the Run 1 and 2 models search parameters more difficult than Run 3 and 4.

Table 6 illustrates the result of DBDC4 Japanese task. Run 1 trained with DBDC1,2-dev/eval and DBDC4-dev achieves the best score in Run 1–4. Run 3 and 4 trained only with DBDC1,2-dev/eval show significantly lower performance than Run 1 and 2. This indicates that DBDC4-dev is necessary to achieve high performance in DBDC4-eval.

More interestingly, EX 1 (extended trials) that leverages only DBDC4-dev for training shows the best result in all the runs including other teams. This shows that DBDC1,2-dev/eval is not necessary to achieve high performance in DBDC4-eval even though DBDC4-eval contains systems same as DBDC1,2.

5 Conclusion

We examined the effectiveness of additional features to improve the performance of BERT for dialogue breakdown detection. Through the analysis, sentence vectors are effective for the estimation but models trained with all the features show the best performance in both English and Japanese tasks. In addition, a comparison of training dataset shows that the training data domain is dominant for the performance. We plan to investigate the effectiveness of pre-training with huge dialogue data in English.

References

1. Bickmore T, Cassell J (2001) Relational agents: a model and implementation of building user trust. In: Proceedings of the SIGCHI conference on human factors in computing systems, pp 396–403
2. Ritter A, Cherry C, Dolan WB (2011) Data-driven response generation in social media. In: Proceedings of the 2011 conference on empirical methods in natural language processing, pp 583–593
3. Wong W, Cavedon L, Thangarajah J, Padgham L (2012) Strategies for mixed-initiative conversation management using question-answer pairs. In: Proceedings of the 24th international conference on computational linguistics, pp. 2821–2834
4. Higashinaka R, Imamura K, Meguro T, Miyazaki C, Kobayashi N, Sugiyama H, Hirano T, Makino T, Matsuo Y (2014) Towards an open-domain conversational system fully based on natural language processing. In: Proceedings of the 25th international conference on computational linguistics, pp 928–939
5. Sugiyama H (2017) Dialogue breakdown detection based on estimating appropriateness of topic transition. In: Proceedings of dialog system technology challenges, vol 6
6. Devlin J, Chang MW, Lee K, Toutanova K (2019) BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the annual conference of the North American Chapter of the Association for Computational Linguistics
7. Meguro T, Higashinaka R, Dohsaka K, Minami Y, Isozaki H (2009) Analysis of listening-oriented dialogue for building listening agents. In: Proceedings of the SIGDIAL 2009 conference: the 10th annual meeting of the special interest group on discourse and dialogue (September), pp 124–127
8. Meguro T, Higashinaka R, Minami Y, Dohsaka K (2010) Controlling listening-oriented dialogue using partially observable markov decision processes. In: Proceedings of the 23rd international conference on computational linguistics, pp 761–769
9. Higashinaka R, D'Haro LF, Shawar BA, Banchs R, Funakoshi K, Inaba M, Tsunomori Y, Takahashi T, Sedoc Ja (2019) Overview of the dialogue breakdown detection challenge 4. In: Proceedings of WOCHAT
10. Banchs RE, Li H (2012) IRIS: a chat-oriented dialogue system based on the vector space model. In: Proceedings of the 50th annual meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp 37–42
11. Higashinaka R, Funakoshi K, Inaba M, Tsunomori Y, Takahashi T, Kaji N (2017) Overview of dialogue breakdown detection challenge 3. In: Proceedings of the dialogue system technology challenge, vol 6
12. Higashinaka R, Funakoshi K, Inaba M, Tsunomori Y, Takahashi T, Akama R (2019) Dialogue system live competition: identifying problems with dialogue systems through live event. In: Proceedings of international workshop on spoken dialogue systems technology
13. Luo L, Xiong Y, Liu Y, Sun X (2019) Adaptive gradient methods with dynamic bound of learning rate. In: Proceedings of the international conference on learning representations