Dialogue Breakdown Detection with Long Short Term Memory

Tittaya Mairittha[1], Tsuyoshi Okita[2], Sozo Inoue[3],
Kyushu Institute of Technology[4]

Abstract: *This paper aims to detect the utterance which can be categorized as the breakdown of the dialogue flow. We propose a logistic regression-based and a Long Short-Term Memory (LSTM)-based methods. Using the input with utterance-response pairs the performance of the LSTM-based method is superior to that of the logistic regression-based method in 36% measured with F-measure. We also measured the performance using the performance with utterance-response pairs: the performance with the input only with responses is unexpectedly inferior to those with responses in 6% to 23% measured with F-measure.*

I. INTRODUCTION

Among dialogue services, the voice agent service provides critical functionalities in daily life. This agent gives a series of answers to the questions by the caller, continuing the conversation for a minute. Ideally, this agent should respond appropriate answers as if a human being is behind the line. This goal is far from being attained. One reason for this lies in our lack of understanding what is the natural way we humans talk and produce a response, leading to dialogue breakdowns in conversation. Here, dialogue breakdowns refers to a situation in which users cannot proceed with the conversation, [**?**]. The dialogue breakdown detection will be useful for continuing the conversation in dialogue systems and may improve the conversation with the agent. Dialogue breakdown detection can be treated as a classification the problem, where the input to the classification model is a sequence of words and the output is a predicted class. More structured neural network approaches for utterance classification include using recursive neural network [**?**]

In this paper, we aim to detect inappropriate utterances that cause dialogue breakdown. Our approach is two-fold: (1) analyze the system utterances judged as breakdowns, we have experimented with two methods to train the dialogue act classification one is a logistic regression which is a robust model for simple classification tasks, another is a powerful type of deep learning recurrent network designed to handle sequence classification problems is called LSTM [**?**] and (2) we want to find out their impact on the input only with responses compared to utterance-response pairs. The results of these experiments show that LSTM-based method can successfully handle dialogue breakdown detection.

II. RELATED WORK

There have been a few papers published that have presented dialogue breakdowns in the conversation. One is a report from dialogue breakdown detection challenge [**?**] where the task is to detect a system's inappropriate utterances that lead to dialogue breakdowns in chat, but the research was conducted only in Japanese.

III. METHODS

In this section, we describe our classification methods in detail.

A. Logistic regression-based method

We will start by considering logistic regression. The logistic regression refers to a classifier that classifies an observation into one of two classes, and multinomial logistic regression is used when classifying into more than two classes. We considered these tasks as multi-label classification tasks which our targets there are three types of labels: O (not a breakdown), T (possible breakdown), and X (breakdown).

The inputs to the multinomial logistic regression are the features we have in the dataset, the utterances we converted them into numerical values by sum all of the vectors for each in the in the utterance, concatenate the two together, and then use this as the input to the linear model. We then use the softmax function which calculates the probabilities for the given score and return a probability distribution on the output breakdowns. And the last stage of multinomial logistic regression uses the cross-entropy function to find the similarity distance between the probabilities calculated using the softmax function.

B. LSTM-based method

We turn now to a second algorithm for classification called LSTM to classify the breakdowns. The first layer is the embedded layer that uses 32 length vectors to represent each word. The next layer is the LSTM-based layer with 100 memory unit, in and the output layer we used a softmax activation function. Finally, the network uses the efficient Adam gradient descent optimization algorithm [**?**] with a logarithmic loss function. The hyper-parameters of the neural network model are as follows: the model parameters are a learning rate of 0.01, a dropout rate of 0.2 for both LSTM and softmax layer outputs and a batch size of 64 to use when training the model, we trained with 5 epochs with early stopping. As discussed above, there are many factors that can influence the model and effect of training. To achieve the best performances, we may fine tune hyper parameters.

IV. BREAKDOWN DETECTION TASK

For training and development data sets, annotations were provided by the dialog system technology challenges (DSTC6) [**?**]. The original dialogue datasets were collected by the WOCHAT dataset and the Conversational Intelligence Challenge (CIC). A total of four data sets, resulting in 4,150 dialogues. Each dialogue was controlled to be 20-21 utterance sequences from the user and the system. All utterances are annotated by 30 annotators (only for system utterances), each element of the annotation fields one the followings: (NB) Not a breakdown, (PB) Possible breakdown, and (B) Breakdown. Example dialogue is shown in Fig.**??**

Possible breakdown	Not a breakdown	Breakdown
7% (266)	52 (5192)	41% (6181)

TABLE I
Distributions of breakdown labels

V. EXPERIMENTAL SETUP

A. Data preprocessing

The data cannot be used directly for performing analysis process. Therefore, certain steps are executed to solve this problem to convert the raw data into a clean data set. We also generated output predictions from a baseline detector that the DSTC6 prepared. The baseline program uses words included in each utterance as features (Bag-of-Words) and detects dialogue breakdowns by using Conditional Random Fields (CRFs). This program outputs three kinds of labels with probability distributions. The correct label for each instance is basically the majority of the labels given to the instance. We also split randomly the dataset into train (90%) and test (10%) sets.

	Speaker	Utterance	NB	PB	B
0	U	Hi Alex	0	0	0
1	S	Hi.	28	1	1
2	U	can you say something different?	0	0	0
3	S	How different?	20	10	0
4	U	very different please	0	0	0
5	S	What you reading?	17	7	6
6	U	harry potter	0	0	0
7	S	'Mr. Potter...	11	16	3
8	U	yes, have you read it?	0	0	0
9	S	No. Not yet. Soon.	22	2	6
10	U	because it is funny	0	0	0
11	S	He looks like a carrot, honey.	4	2	24
12	U	what!!!!	0	0	0
13	S	Zordon... he's gone.	3	4	23
14	U	hello Alex	0	0	0
15	S	Nice to see you.	24	3	3
16	U	well I do not remember saying that	0	0	0
17	S	Do you remember me?	11	10	9
18	U	of course	0	0	0
19	S	I'll see you Thursday night.	7	15	8

Fig. 1. An example of a dialogue with the number of breakdown labels for each system utterance (S means the system and U means the user)

B. Models

Our proposed model encodes utterances in context into fixed-length vectors, then processes these encoded vectors in the model. Our first step is to encode raw text inputs from the utterances into distributed vector representations, each utterance is represented by a sequence word index, the index is ordered by the most frequent words. For each sequence, we truncated words to make the consistent length. Then, looked up the embedding for each individual index, we also initialized the embedding with GloVe [**?**] embeddings on 50-dimensional vectors of Wikipedia 2014 and return a matrix of d-dimensional vectors for passing them to the model. The second step is to create the models that detect dialogue breakdowns as implemented in the

Keras framework. We can see that in the section **??**.

C. Analyzing

We used a type of evaluation metrics to evaluate the classification of the breakdown labels. We evaluated each breakdown's classification performance by using precision, recall, F-measure and evaluated the total performance of all breakdown labels using macro averages.

- Accuracy: The number of correctly classified labels divided by the total number of labels to be classified.
- Precision, recall, F-measure (B): The precision, recall, and F-measure for the classification of the B labels.
- Precision, recall, F-measure (PB+B): The precision, recall, and F-measure for the classification of PB + B labels; that is, PB and B labels are treated as a single label.

VI. RESULTS

Model	Accuracy	Breakdown			Possible breakdown + Breakdown		
		Precision	Recall	F-measure	Precision	Recall	F-measure
LSTM pair	0.636	0.58	0.59	0.59	0.78	0.67	0.71
LSTM single	0.598	0.52	0.48	0.5	0.85	0.52	0.57
Logistic pair	0.609	0.13	0.43	0.2	0.13	0.43	0.2
Logistic single	0.562	0.1	0.33	0.14	0.5	0.35	0.2

TABLE II

The precision, recall, and F-measure for the classification of (B) labels. and (B + PB) labels

Table **??** reports the best results we obtained from the methods. As can be seen from the table, deep learning LSTM, achieved good precision, also seem to be reasonable from the input only with responses and utterance-response pairs. However, the B label is still more difficult to classify compared to PB+B labels. This indicates that it is difficult to discriminate between possible breakdowns and breakdowns, but it is possible to discriminate breakdowns from non-breakdowns.

Table **??** shown in patterns of errors of in dialogue breakdown detection, presenting a confusion matrix.

Actual	Prediction outcome		
	NB	**PB**	**B**
NB	U: How are you today? R: Good. How are you?	U: So, what else? R: Okay, okay. What else you got?	U: What? , R: We're skinny dipping, Danny.
PB	U: Just like dogs. R: What kind of dog	U: Lock him up! R: Okay! Who else should we lock up?	U: well, I can do many things R: Ten.
B	Oh, my god ! 1 plus 1?	U: Amoroso renato. R: where are you from?	U: Who's Mr.Vitti? R:Where's Vitti? I don't see Vitti.

TABLE III

An example results for a three class classifier.

Figure **??** shows example dialogue between our dialogue system and a user where the annotated breakdown.

	Speaker	utterances
0	U	what did happen to juliet?
1	S	So was Juliet – at fifteen.
2	U	Why?
3	S	It's not right. I feel horrible. Besides...
4	U	Still incorrect :(I think that correct answer is: first:)
5	S	I don't know. what do you think
6	U	You sure like to not talk about thing
7	S	I don't know. That's interesting.
8	U	I don't hate anything about it
9	S	So you're less intelligent than a computer?!
10	U	You are so strange.
11	S	Actually... I do not have any idea!!!
12	U	I don't know.
13	S	You've just said that you want to discuss news.
14	U	You are repeating yourself.
15	S	I repeat myself repeat myself.
16	U	my day was ok
17	S	Hide your laptop here.
18	U	Do you really want to talk with me?
19	S	Talk to me.

Fig. 2. An example of a dialogue breakdown

When we looked at the incorrect predictions cases in Figure **??**, we found that those utterances may breakdown from a proper noun or a named entity. So future studies will include identifying and extracting entities (people, places, companies, etc.) to improve the performance of classification efficiency.

	Speaker	utterances
0	U	goodYou are rigth!How can I reward you?
1	S	... not for you and this post.
2	U	Grover Cleveland Golf Course
3	S	Who is Aung San Suu Kyi?
4	U	Nope. I don't know that she's anything like Christmas.
5	S	Do you like my girlfriend?
6	U	What?
7	S	We're skinny dipping, Danny.
8	U	Still incorrect. I think that true answer is: jesus.
9	S	jesus
10	U	i'm sorry . i'm so sorry .
11	S	no worries broI took it easy
12	U	I like blackwidow's body
13	S	I love Black Widow's green eyes.
14	U	Move!Still incorrect :(Lets speak about something else...
15	S	sad(
16	U	iron man
17	S	I love Iron Man's blue eyes.
18	U	So, what else?
19	S	Okay, okay. What else you got?

Fig. 3. An example dialogues which incorrect predictions

VII. CONCLUSIONS

In this paper, we verified the implementation of our models by achieving results very similar to those achieved by previously published works within the challenge, the accuracy for the former is 0.643 while that of the latter is 0.636. When we look at the details, we can see the performing of utterance-response pairs, rather than the input only with responses. However, the accuracies are low, it is still difficult to detect breakdowns. Future research should aim to improve the systems with sentence-based classification.

References

[1] Martinovsky, B., Traum, D.: The error is the cue: Breakdown in human-machine interaction. In: Proc. Error Handling in Spoken Dialogue Systems. pp. 11–16 (2003)

[2] Ravuri, Suman V., and Andreas Stolcke. "Recurrent neural network and LSTM models for lexical utterance classification." INTERSPEECH. 2015.

[3] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." Neural computation 9.8 (1997): 1735-1780. APA

[4] Higashinaka, Ryuichiro, et al. "The dialogue breakdown detection challenge: Task description, datasets, and evaluation metrics." LREC. 2016.

[5] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C., Ng, A., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In EMNLP, 2013.

[6] Kingma, Diederik, and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).

[7] Dialogue System Technology Challenge 6 (DSTC6), http://workshop.colips.org/dstc6/call.html

[8] Pennington, Jeffrey, Richard Socher, and Christopher Manning. "Glove: Global vectors for word representation." Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.