

A first study on a dialogue system for an academic writing aid system

Ni Zihui

Graduate School of Information, Production and Systems,

Waseda University



nizihui@akane.waseda.jp

July 21, 2021

Outline

1 Introduction

2 Method

3 Experiment

4 Conclusion

5 Future work

The goal : Realize a dialogue system for the writing aid task

- To build a dialogue dataset for the writing aid task.
Try to automatically modify the sentences and the annotations in the existing dialogue data set.
- To finetune a dialogue model for the writing aid task.
In the existing Dialogue framework, finetuning the task with the modified data.

An example of the dialogue to the writing aid task

- **The system** (output): Hello, welcome to the writing aid system. You can do many kinds of modification of the paragraph here. Do you want a grammar check first?
- **The user** (input): Yes, please.
- **The system** (output): Okay, do you want an error report or just modify the paragraph?
- **The user** (input): An error report please.

Introduction: The existing dialogue system

- The current task-based dialogue system has been applied to many fields, such as *hotel reservations*, *ticket booking*, and *take-way ordering*.

Name of the system	Dialogue scalability	Natural language management	Natural language Understanding	Natural language generation
<i>OpenDial</i> [Lison and Kennington, 2016]	direct coding	plan	machine learning	template
<i>RASA</i> [Bocklisch et al., 2017]	direct coding	machine learning	machine learning	-
<i>ParIAI</i> [Miller et al., 2017]	direct coding	machine learning	machine learning	template
<i>DeepPavlov</i> [Burtsev et al., 2018]	direct coding	machine learning	machine learning	template
<i>OwlSpeak</i> [Heinroth et al., 2010]	universal plug and play	information state	rule	template
<i>PyDial</i> [Ultes et al., 2017]	direct coding	reinforcement learning	rule and machine learning	rule and machine learning
<i>DialogStudio</i> [Jung et al., 2008]	direct coding	example	machine learning	example
<i>RavenClaw</i> [Bohus and Rudnicky, 2009]	direct coding	plan	rule	template

Table: Some existing dialogue systems

Introduction: The existing dataset

- The current task-based dialogue system has been applied to many fields, such as *hotel reservations*, *ticket booking*, and *take-way ordering*.

Name of the dataset	Size	Source	Quality	Fields
DailyDialog [Li et al., 2017]	13,118	textbooks	auto-extracted	various topics
Wizard-of-Wikipedia [Dinan et al., 2019]	22,311	crowdsourcing	human-written	various topics
Document-grounded [Feng et al., 2020]	4,470	crowdsourcing	human-written	various topics
Persona-Chat [Zhang et al., 2018]	10,981	crowdsourcing	human-written	various topics
Self-dialogue [Fainberg et al., 2018]	24,165	crowdsourcing	human-written	various topics
Cornell Movie Corpus [Danescu-Niculescu-Mizil and Lee, 2011]	304,713	movie scripts	auto-extracted	movie
Self-feeding chatbot [Heinroth et al., 2010]	145,873	human-bot dialogues	human-written (half)	various topics
Twitter corpus [Cieliebak et al., 2017]	9,783	twitter posts/replies	auto-extracted	various topics
Opensubtitles [Lison and Tiedemann, 2016]	5,000,000	movie subtitles	auto-extracted	movie
AirDialogue [Wei et al., 2018]	301,427	human annotators	human-written	travel, flight
MultiWOZ [Budzianowski et al., 2018]	8,438	human behaviors	human-written	hotel, restaurant
DSTC 2 [Henderson et al., 2014]	2235	human behaviors	Amazon Mechanical Turk	food

Table: Some existing datasets of the dialogue system

The goal : Realize a dialogue system for the writing aid task

- To build a dialogue dataset for the writing aid task.
Try to automatically modify the sentences and the annotations in the existing dialogue dataset.
- To finetune a dialogue model for the writing aid task.
In the existing Dialogue framework, finetuning the task with the modified data.

The DSTC 2 dataset

- The dataset we built is based on the DSTC 2 dataset [Henderson et al., 2014].
- DSTC 2 is human-machine dialogue data, a multi-round dialogue data set in the field of restaurant reservations, in which there are 1, 612 training data, 506 validation set, and 1, 117 test set.
- This is the annotation of the DSTC 2 dataset.

annotation	explanation
speaker	the system or the user
text	the dialogue text
goal	information about the goal/task for the system
slots	request parameter (slot list)
acts	the purpose of the dialogue sentence, such as inform, inquiry, etc
dialogue_acts	this is determined by what slots and acts is , directly determines how to generate answers and call the APIs

Table: The annotations of the DSTC 2 dataset

The DSTC 2 dataset

This is an example of the DSTC 2 dataset:

```
{
  "speaker": 2,
  "text": "Hello, welcome to the Cambridge
restaurant system. You can ask for
restaurants by area, price range or food
type. How may I help you?",
  "dialog_acts": [
    {
      "act": "welcomemsg",
      "slots": [
      ]
    }
  ]
}

{
  "speaker": 1,
  "text": "cheap restaurant",
  "goals": {
    "pricerange": "cheap"
  },
  "dialog_acts": [
    {
      "slots": [
        {
          "pricerange",
          "cheap"
        }
      ],
      "act": "inform"
    }
  ]
}
```

Figure: An example of the DSTC 2 dataset (Graph copied from [Henderson et al., 2014])

The introduction of the slot table

- **Slot** is parameters of the query of user.
- For example, in the keyword extraction task:
 - Slot: User needs to appoint **how many keywords** to be extracted.
 - `number_of_keywords = 5`

Slot	Requestable	Informable
food	yes	yes, 91 possible values
name	yes	yes, 113 possible values
pricerange	yes	yes, 3 possible values
addr	yes	no
phone	yes	no
postcode	yes	no
signature	yes	no

Table: The slot table designed in DSTC 2 [Henderson et al., 2014]

The introduction of the slot table

```
"food": {  
  "caribbean": [  
    "carraibbean",  
    "carribean",  
    "caribbean"  
  ],  
  "kosher": [  
    "kosher"  
  ],  
  "tuscan": [  
    "tuscan"  
  ],  
  "french": [  
    "french"  
  ]  
}  
  
"this": {  
  "dontcare": [  
    "dont care",  
    "doesnt  
matter",  
    "any fine",  
    "any noise",  
    "any of town",  
    "noise  
anything",  
    "any type",  
    "anything",  
    "any thing",  
    "what ever",  
    "does not  
matter"  
  ]  
}
```

Figure: An example of the slot and the corresponding slot value in the DSTC 2 dataset (Graph copied from [Henderson et al., 2014])

The goal we need to do in the dataset

We need to modify the data set from **dialogue sentences** and **dialogue annotations**.

The goal : Realize a dialogue system for the writing aid task

- To build a dialogue dataset for the writing aid task.
Try to automatically modify the sentences and the annotations in the existing dialogue data set.
- To finetune a dialogue model for the writing aid task.
In the existing Dialogue framework, finetuning the task with the modified data.

Introduction: The structure of the existing dialogue system

- **User input:** User request.
- **System output:** System answer.

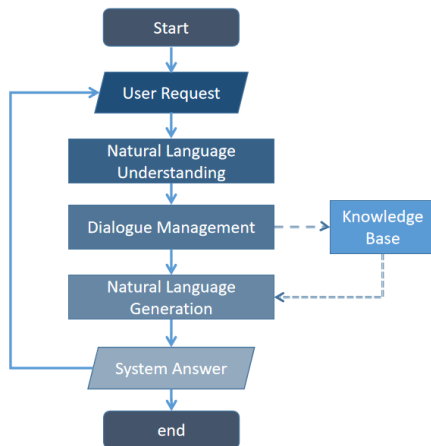


Figure: Task-oriented Pipeline Dialogue System

Introduction: The structure of the our dialogue system

- **User input:** User request and the paragraph.
- **System output:** System answer and the paragraph.

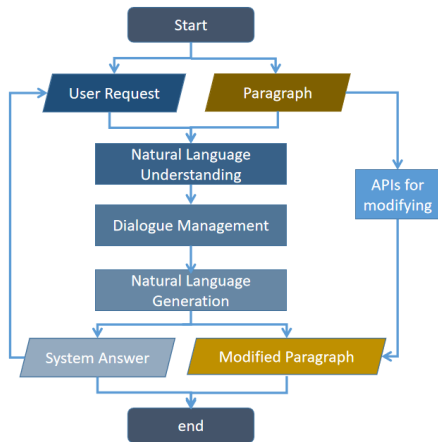


Figure: The flow chart of the pipeline dialogue system for writing aid task

Introduction: The dialogue system

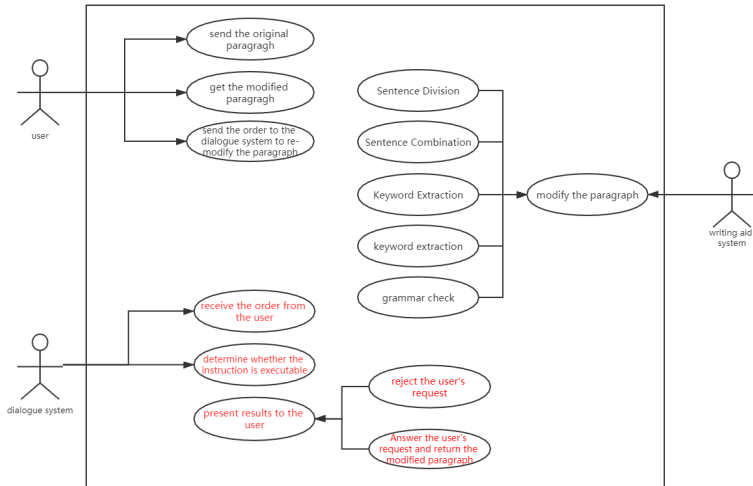


Figure: The user case for an academic writing aid system

Outline

1 Introduction

2 Method

3 Experiment

4 Conclusion

5 Future work

Method: Dataset

- Design the new slot and slot value in the slot table.
- Make the new slot and value appear in the new dialogue sentence.
- Prepare writing aid-related sentence data set.
- Replace the all the words about restaurant reservation in DSTC 2 dataset
- Use grammar check to correct the grammatical errors in the dialogue text.

Method: Dataset

(1) Replace the slot and value of the slot in the original DSTC 2 dataset in the newly designed slot table.

slot	Requestable	Informable
type of sentence division terms	yes	yes. 3 values; and, but, so
type of grammar error correction	yes	yes, 2 values; error report, modified paragraph
number of keyword extraction	yes	yes, 5 values; 1,2,3,4,5
type of text summarization	yes	yes, 2 values; opening sentence, abstract
number of sentences	yes	no
average number of words	yes	no
number of total words	yes	no

Table: The newly designed slot table

Method: Dataset

(2) Make the new slot and value appear in the new dialogue sentence.

```
{"speaker": 2, "text": "What kind of food → type of grammar error correction would you like?", "dialog_acts": [{"act": "request_food →  
type_of_grammar_error_correction", "slots": []}], {"speaker": 1, "text": "gastropub → error report", "goals": {"food" → type of grammar  
error correction : "gastropub" → error report}, "dialog_acts": [{"slots": [{"food" → type of grammar error correction ", "gastropub" → error  
report "}], "act": "inform"}]}
```

```
{"speaker": 2, "text": "What kind of type of grammar error correction would you like?", "dialog_acts": [{"act":  
"request_ type_of_grammar_error_correction ", "slots": []}], {"speaker": 1, "text": " error report", "goals": {"type of grammar error  
correction : " error report}, "dialog_acts": [{"slots": [{" type of grammar error correction ", " error report "}], "act": "inform"}]}
```

Method: Dataset

(3) Prepare writing aid-related sentence data set. We get **84** conversations about writing aid manually and added **2,494** sentences in **ACL-ARC data set** [Bird et al., 2008]¹ as the data set about writing aid.

Please demonstrate it with the result with conjunction of word "and".
I want to see the number of sentences.
I want to see the number of total words.
One possible weakness of discourse-based summarization techniques is that they rely greatly on the accuracy of the discourse parser they use .

Table: The example of the writing aid-related sentence data set

¹use the ACL-ARC sentences extracted in the Mango system
:/files/Data/AcademicWritingAid/backup/cleaned_data

Method: Dataset

(4) We use pre-trained embedding Bert model [Devlin et al., 2019] to replace the extracted words of DSTC 2 to the words in the writing aid dataset. For each word W_i [Li et al., 2018] in DSTC 2, we calculate the probability $P_1(w_i)$ in DSTC 2 and $P_2(w_i)$ in writing aid dataset. And then we calculate

$$W_i = |\log(|P_1(w_i) - P_2(w_i)|)|$$

If W_i of word i is near 0, it should be substituted. The closer W_i is to 0, the greater the frequency difference between the two texts, and the more this word should be replaced.

Method: Dataset

(4) Use Bert model [Devlin et al., 2019] to replace extracted words of DSTC 2 to the words in the writing aid dataset. We set the threshold at 9.72. This is the part of word that we sift out and need to be replaced.

word	result
restauarant	9.714
korean	9.714
baskey	9.714
lcoated	9.714
Graffiti	9.714
tasca	9.714

Table: The words are copied from the output of the dataset. The type errors in "restauarant" and "lcoated" come from the data.

Method: Dataset

(5) Use **grammar check** to correct the grammatical errors in the dialogue text.

Method: The structure of the dialogue system

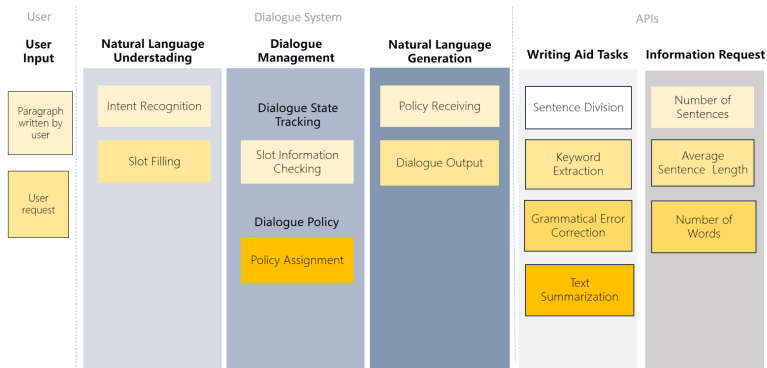


Figure: The structure of dialogue system

Method: The Deepavlov framework

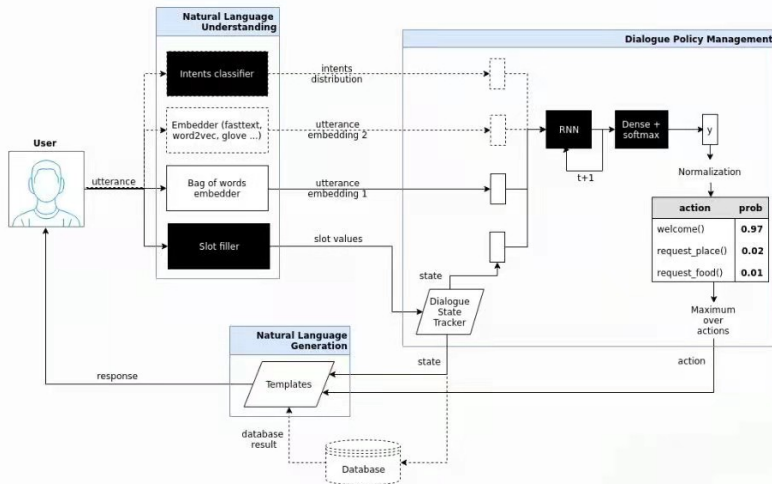


Figure: The structure of deepavlov framework (The graph is copied from [Burtsev et al., 2018])

Outline

1 Introduction

2 Method

3 Experiment

4 Conclusion

5 Future work

Dataset

We generate **2,235** sets of dialogue data, containing **39,420** dialogue sentences with their annotations.

<i>'speaker' : 2, 'text' : ' Hello, welcome to the writing aid system. You can do many kinds of modification of the paragraph here. Do you want a grammar check first?'</i> , <i>'dialog_acts' : 'act' : ' welcomemsg', ' slots' : []</i>
<i>'speaker' : 1, 'text' : ' Yes, please give me the grammar check.', 'goals' : 'grammar check' : ' dontcare', 'dialog_acts' : ' slots' : ['type of text summarization', ' abstract'], ' act' : ' inform'</i>
<i>'speaker' : 2, 'text' : ' What kind of type of grammar error correction would you like?', 'dialog_acts' : 'act' : ' request_type_of_grammar_error_correction', ' slots' : []</i>
<i>'speaker' : 1, 'text' : ' any', 'goals' : 'type of grammar error correction' : ' dontcare', ' type of text summarization' : ' dontcare', 'dialog_acts' : ' slots' : [['this', ' dontcare']], ' act' : ' inform'</i>

Table: The example of our building writing aid dialogue data set

Baseline

- We measure the accuracy of predicting bot answers on different datasets.
- We hope to prove that the prediction effect of our new data set under the same Deepavlov framework can be basically the same as that of the mainstream dataset.

Result

- We take same numbers with the DSTC 2 dataset, with **1,612** training data, **506** validation set, and **1,117** test set.

Dataset	accuracy (DST)	precision (SF)	recall (SF)	F1 (SF)
DSTC 2	47%	99%	97%	98%
MultiWOZ	47%	96%	96%	96%
AirDialogue	22%	59%	56%	55%
Ours	51%	98%	93%	0.96%

Table: The quantitative result of predicting bot answers on datasets.

The result of the dialogue system

Speaker 1: Hello, welcome to the writing aid system. You can do many kinds of modification of the paragraph here. Do you want a grammar check first?
--

Speaker2: Yes, and please send me the error report.

Speaker1: Okay, I'll do it for you.

Table: The example of the chat in the writing aid task

Outline

1 Introduction

2 Method

3 Experiment

4 Conclusion

5 Future work

Conclusion

- We build a writing aid dialogue dataset based on DSTC 2.
- We finetune our writing aid dialogue task on the Deeppavlov framework.

Outline

1 Introduction

2 Method

3 Experiment

4 Conclusion

5 Future work

Future work

- Test our data set with different dialogue system frameworks.
- Analyze the reasons why our data sets have different results in different dialogue system frames.

Thank you for your listening.

Q&A

References I



Bird, S., Dale, R., Dorr, B., Gibson, B., Joseph, M., Kan, M.-Y., Lee, D., Powley, B., Radev, D., and Tan, Y. F. (2008).

The ACL Anthology reference corpus: A reference dataset for bibliographic research in computational linguistics.

In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08), Marrakech, Morocco. European Language Resources Association (ELRA).



Bocklisch, T., Faulkner, J., Pawlowski, N., and Nichol, A. (2017).

Rasa: Open source language understanding and dialogue management.

ArXiv, abs/1712.05181.



Bohus, D. and Rudnicky, A. I. (2009).

The ravenclaw dialog management framework: Architecture and systems.

Comput. Speech Lang., 23:332–361.

References II



Budzianowski, P., Wen, T.-H., Tseng, B.-H., Casanueva, I., Ultes, S., Ramadan, O., and Gašić, M. (2018).

MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling.

In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.



Burtsev, M., Seliverstov, A., Airapetyan, R., Arkhipov, M., Baymurzina, D., Bushkov, N., Gureenkova, O., Khakhulin, T., Kuratov, Y., Kuznetsov, D., Litinsky, A., Logacheva, V., Lymar, A., Malykh, V., Petrov, M., Polulyakh, V., Pugachev, L., Sorokin, A., Vikhreva, M., and Zaynutdinov, M. (2018).

DeepPavlov: Open-source library for dialogue systems.

In Proceedings of ACL 2018, System Demonstrations, pages 122–127, Melbourne, Australia. Association for Computational Linguistics.

References III



Cieliebak, M., Deriu, J. M., Egger, D., and Uzdilli, F. (2017).
A Twitter corpus and benchmark resources for German
sentiment analysis.

*In Proceedings of the Fifth International Workshop on Natural
Language Processing for Social Media*, pages 45–51, Valencia,
Spain. Association for Computational Linguistics.



Danescu-Niculescu-Mizil, C. and Lee, L. (2011).
Chameleons in imagined conversations: A new approach to
understanding coordination of linguistic style in dialogs.

*In Proceedings of the Workshop on Cognitive Modeling and
Computational Linguistics, ACL 2011.*

References IV



Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019).
BERT: Pre-training of deep bidirectional transformers for
language understanding.

*In Proceedings of the 2019 Conference of the North American
Chapter of the Association for Computational Linguistics: Human
Language Technologies, Volume 1 (Long and Short Papers),*
pages 4171–4186, Minneapolis, Minnesota. Association for
Computational Linguistics.



Dinan, E., Roller, S., Shuster, K., Fan, A., Auli, M., and Weston, J.
(2019).

Wizard of wikipedia: Knowledge-powered conversational agents.
ArXiv, abs/1811.01241.

References VI



Henderson, M., Thomson, B., and Williams, J. D. (2014).

The second dialog state tracking challenge.

In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 263–272, Philadelphia, PA, U.S.A. Association for Computational Linguistics.



Jung, S., Lee, C., and Lee, G. (2008).

Dialogstudio: A workbench for data-driven spoken dialog system development and management.

Speech Communication, 50:697–715.

References VII



Li, J., Jia, R., He, H., and Liang, P. (2018).

Delete, retrieve, generate: a simple approach to sentiment and style transfer.

In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1865–1874, New Orleans, Louisiana. Association for Computational Linguistics.



Li, Y., Su, H., Shen, X., Li, W., Cao, Z., and Niu, S. (2017).

DailyDialog: A manually labelled multi-turn dialogue dataset.

In Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 986–995, Taipei, Taiwan. Asian Federation of Natural Language Processing.

References VIII



Lison, P. and Kennington, C. (2016).

OpenDial: A toolkit for developing spoken dialogue systems with probabilistic rules.

In *Proceedings of ACL-2016 System Demonstrations*, pages 67–72, Berlin, Germany. Association for Computational Linguistics.



Lison, P. and Tiedemann, J. (2016).

OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles.

In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).

References IX



Miller, A., Feng, W., Batra, D., Bordes, A., Fisch, A., Lu, J., Parikh, D., and Weston, J. (2017).

ParlAI: A dialog research software platform.

In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 79–84, Copenhagen, Denmark. Association for Computational Linguistics.



Ultes, S., Rojas-Barahona, L. M., Su, P.-H., Vandyke, D., Kim, D., Casanueva, I., Budzianowski, P., Mrkšić, N., Wen, T.-H., Gašić, M., and Young, S. (2017).

PyDial: A multi-domain statistical dialogue system toolkit.

In Proceedings of ACL 2017, System Demonstrations, pages 73–78, Vancouver, Canada. Association for Computational Linguistics.

References X



Wei, W., Le, Q., Dai, A., and Li, J. (2018).

AirDialogue: An environment for goal-oriented dialogue research.

In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 3844–3854, Brussels, Belgium. Association for Computational Linguistics.



Zhang, S., Dinan, E., Urbanek, J., Szlam, A., Kiela, D., and Weston, J. (2018).

Personalizing dialogue agents: I have a dog, do you have pets too?

In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2204–2213, Melbourne, Australia. Association for Computational Linguistics.