

2024 Fall

Introduction to Natural Language Processing

HW3: Situated Proactive Response Selection

TA: Yu-Chien Tang
tommytyc.cs13@nycu.edu.tw

Task introduction

- Response Classification
 - You are given an user's question and the current situation. Please assess the quality of the response.
 - There is **no limit to the methods you can use**. Feel free to employ any method you prefer to build your model or process the data.
- Requirement
 - Upload your submission to Kaggle.
 - Submit a report and your source code to E3.

Deadline is 12/20 (Friday) 23:59, no late submission.

Dataset

English conversation with situational background information.

- [HW3_dataset.zip](#)
 - train.json
 - val.json
 - test.json
 - sample_submission.csv



Data introduction

Each object in the json file contains the following information :

- **u**: utterance (question)
- **s**: 12 situational statements
- **s.type**: semantic categories of the situational statements (Location, Possession, Time, Date, Behavior, Environment)
- **s.gold.index**: index of the situational statement related to the utterance (**only present in the train and val sets**)
- **r**: response
- **r.label**: quality assessment of the response (0: bad, 1: good)

Tip : you can use LLM (e.g., GPT) to assist in identifying relevant situations.

Data example

```
{'u': 'Please find my glasses.',  
 's': ['The wind blows really strong.',  
       '[user] is home.',  
       "[user] knows [someone]'s phone number.",  
       '[user] has a birthday coming up.',  
       "[user]'s contacts include a locksmith.",  
       '[user] owns a camera.',  
       '[user] has bad vision.',  
       'There are other people in the parking lot that may have a spare light bulb.',  
       '[user] has a letter from [someone].',  
       '[user] has eye contacts.',  
       'The overhead light bulb is burned out.',  
       "[user]'s glasses are broken."],  
 's.type': ['environment',  
            'location',  
            'possession',  
            'date',  
            'environment',  
            'possession',  
            'behavior',  
            'possession',  
            'possession',  
            'possession',  
            'environment',  
            'environment'],  
 's.gold.index': [11, 8, 6, 9, 1],  
 'r': 'Sorry, but your glasses are broken. Shall I bring your contacts?',  
 'r.label': 1}
```

Kaggle Submission Format



Your model is expected to determine whether the response in <test.json> is correct or not, and then upload your model's predictions to Kaggle. the submission format should be:

- A 793*2 .csv file, first row for column name and the last 792 rows for your result (0: bad, 1: good)
- First row must match the one shown in the sample_submission.csv, make sure the order is correct!

	A	B	C
1	index	response_quality	
2	0	0	
3	1	1	
4	2	0	
5	3	1	

Kaggle Submission (70%)

- [Kaggle link](#)
- **Change your team name into your student ID**, or there will be a deduction of 5 points for HW3.
- There'll be a simple baseline and a strong baseline. Beat them to get higher score.
- The scoring metric is **accuracy**. Higher score means better performance.
- You can submit **at most 5 times each day** and choose 2 of the submissions to be considered for the private leaderboard, or will otherwise default to the best public scoring submissions.

#	Team	Members	Score	Entries
	Strong Baseline		0.86616	
	Simple Baseline		0.81313	

Report Submission (30%)

Answer the following 3 questions :

1. Describe how you implement your model, including your choice of packages, model architectures, model input, loss functions, hyperparameters, etc.
2. What processing did you do with the data? Is there an improvement in predictive accuracy when utilizing both situations and utterances for prediction, compared to solely relying on utterances? Why or why not?
3. Compare all the methods you have tried and use a table to display their respective performances. Which method performed the best, and why?

Please answer the questions in detail to get the full point of each question.

Grading policy

- Kaggle (70%)
 - 30% based on the public leaderboard score and 70% based on the private leaderboard score. Both public and private board are the 50% split of test set.
 - Basic score :
 - Over strong baseline : 55
 - Over simple bassline : 40
 - Under simple baseline : 25
 - Ranking score:
 $15 - (15/N) * (\text{ranking} - 1)$, N=numbers of people in the interval
- Report (30%)
 - 10% for each quesiton

You will receive 0 points if you do not submit the source code.

E3 Submission

Submit your source code and report to E3 before 12/20 (Friday) 23:59.

No late submissions will be accepted!

Format:

- HW3_<student ID>.zip
 - source code: HW3_<student ID>.py or HW3_<student ID>.ipynb
 - report: HW3_<student ID>.pdf

Feel free to contact TA Yu-Chien if you have any question about HW3.

mail: tommytyc.cs13@nycu.edu.tw

Have Fun !

