Leveraging Linguistic Coordination in Reranking N-Best Candidates For End-to-End Response Selection Using BERT

Mingzhi Yu, Diane Litman
University of Pittsburgh
Pittsburgh, PA
miy39@pitt.edu, dlitman@pitt.edu

Abstract

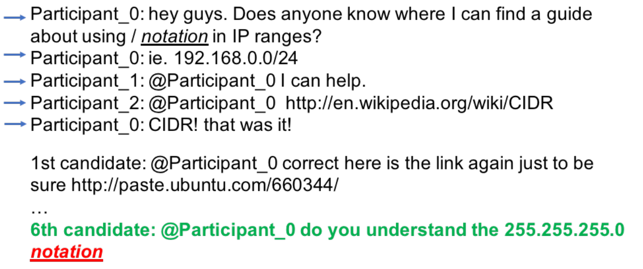
Retrieval-based dialogue systems select the best response from many candidates. Although many state-of-the-art models have shown promising performance in dialogue response selection tasks, there is still quite a gap between R@1 and R@10 performance. To address this, we propose to leverage linguistic coordination (a phenomenon that individuals tend to develop similar linguistic behaviors in conversation) to rerank the N-best candidates produced by BERT, a state-of-the-art pre-trained language model. Our results show an improvement in R@1 compared to BERT baselines, demonstrating the utility of repairing machine-generated outputs by leveraging a linguistic theory.

Introduction

In recent years, end-to-end (E2E) dialogue systems have made remarkable progress. One important type of E2E system is the retrieval-based system (Chen et al. 2016; Zhou et al. 2018), which aims to select the best dialogue response from multiple candidates. The core of these systems is often formalized as a task to match dialogue context and response candidates.

The use of pre-trained language models has recently been attracting attention. With their rich contextualized input representation trained on a large amount of data, models such as ELMO (Peters et al. 2018), XLNET (Yang et al. 2019), and BERT (Devlin et al. 2018) have achieved state-of-the-art performance on various NLP tasks, including the dialogue response selection task of DSTC8 track 2 (Kim et al. 2019). Thus, we also experiment with a pre-trained language model, namely BERT, as our baseline for response selection.

While prior work has shown promising response selection results, there is still a gap between the recall at the best (R@1) and top 10 (R@10) candidates (see section Evaluation Metrics). For many reported models on the leaderboard of DSTC 8 track 2, R@10 can achieve above 90%, while R1 can be approximately 30% lower on average. This gap between R@1 and R@10 indicates that the model failed to select the correct response, but the correct response is highly likely within the 10 best candidates. In the evaluation of our baseline model (see section Results), we also have a similar observation. This observation motivates us to rerank the N-best candidates (in our case, produced by BERT) with the goal of improving model performance as reflected by R@1.



→ Participant_0: hey guys. Does anyone know where I can find a guide about using / *notation* in IP ranges?
→ Participant_0: ie. 192.168.0.0/24
→ Participant_1: @Participant_0 I can help.
→ Participant_2: @Participant_0 http://en.wikipedia.org/wiki/CIDR
→ Participant_0: CIDR! that was it!

1st candidate: @Participant_0 correct here is the link again just to be sure http://paste.ubuntu.com/660344/
...
6th candidate: @Participant_0 do you understand the 255.255.255.0 *notation*

Figure 1: A ranking example. Italic text shows possible coordination. The correct response is in bold text.

Our approach is based on linguistic coordination, a phenomenon that individuals tend to linguistically mimic each other in the conversation. Previous research showed that language may converge in a wide range of linguistic dimensions such as lexical (Nenkova, Gravano, and Hirschberg 2008; Brennan 1996), acoustic-prosodic (Rahimi et al. 2017; Levitan and Hirschberg 2011), and linguistic styles (Gonzales, Hancock, and Pennebaker 2010). Therefore, we propose to leverage linguistic coordination, specifically lexical coordination to improve response selection.

Figure 1 shows example response ranks from a baseline model, where the correct answer is incorrectly ranked as the 6th rather than the top candidate. In the correct response, the word "notation" was mentioned in the context. This can signal higher lexical coordination. Examples such as this lead us to hypothesize that using linguistic coordination to rerank N-best candidates will improve R@1 performance. To our knowledge, this is the first attempt to leverage linguistic coordination in the dialogue response selection task.

Dataset

We evaluate our approach on the Ubuntu IRC dataset (Kummerfeld et al. 2019) provided by the eighth Dialog System Technology Challenge (DSTC8) track 2 subtask 1, which is the disentangled Ubuntu IRC dataset. Each sample in the dataset consists of a multi-turn and multi-party dialogue with 100 response candidates that may be selected for the next turn. The ground-truth response might not be on the candidate list, leading to some no-answer cases. The DSTC 8 track 2 provides a train, development, and test set. A model is expected to rank the list of candidates by their possibility to be the utterance for the next turn. The evaluation metrics include Recall@k. Here the k in the Recall@k means that the true positive response is among the first k ranked candidates. Previous works reported Recall at 1, 2, 5 and 10 (Wu et al. 2020; Dario Bertero 2020). Intuitively, the higher value of a Recall@k indicates a stronger model capability to select the proper response. The smaller k is, the more robust the model is.

We use all conversations from the train set for training. When a conversation has no answer, we use the last utterance in the context as the response and all prior utterances as the new dialogue context. There are two methods to perform evaluation considering there are some no-answer cases. The first method adds a no-answer candidate to the candidate list for each dialogue, and it treats the no-answer candidate as a response. The second method treats a dialogue as a no-answer case if the best candidate selected by the model has a confidence score lower than a certain threshold, e.g., 95%. The method could potentially lead to an overestimated recall at higher values of k. For a naive example, by setting a high threshold for no-answer cases, e.g., 99.9%, many dialogues in the test set can be predicted as no-answer cases. Thus, a model will put no-answer option at the top of its candidate ranking list as the best predicted candidate. Meanwhile, the remaining candidates may be kept the same order. This increases the odds of including the correct candidate in the ranking list predicted by the model. In the example of setting high threshed as 99.9%, if this dialogue is a true no-answer case, then a model will score a hit for Recall@1, Recall@2 and Recall@10. On the opposite, if the dialogue is a false no-answer case, the model will miss the hit for Recall@1, but it is still highly likely to score a hit for Recall@2 and Recall@10.

Therefore, for the simplicity of evaluation, we exclude no-answer cases from the development and test set, and we only evaluate on the dialogues with explicit answers. Note that the previous DSTC 7 (Yoshino et al. 2019) provides similar datasets containing all having-answer conversations, but the conversations are two-party.

We clean our dataset by removing dialogues containing empty candidates. In summary, we have 223,487 dialogues in the train set, 3,837 in the development set, and 4,384 in the test set. Due to the nature of the Ubuntu dataset, there are many typos and technical terms such as urls and symbols. We limit the out-of-vocabulary words by preprocessing the datasets. Certain types of vocabulary are represented as abstract categories including paths, urls, symbols, file extensions, numbers, and addresses. We stemmed words.

Methodology

Dialogue Modeling

We follow previous works to formalize the response selection task as a supervised binary classification problem (Chen et al. 2016; Zhou et al. 2018). Each instance is a triple consisting of a label, dialogue context, and a response. The response in the positive case is the ground-truth response. The response in the negative case is randomly sampled from the remaining candidates. The ratio of positive to negative instance is treated as a hyper-parameter. Our ratio is 1:4. This results in approximately 1 million training cases.

Baseline Model

We use the BERT model as the baseline in a pre-training and fine-tuning manner. To adapt the dialogue to BERT model, we use the dialogue context and response as the two input segments suggested in generic BERT (Devlin et al. 2018). Other configurations also follow the standard BERT for the sentence pairing task. Similar to their sentence pair task model, we add a single fully-connected layer that takes the contextualized class representation T_{cls} as the input and then a softmax layer to perform the binary classification task. We minimized the cross-entropy loss. The final output probability for the positive class will be used as the ranking score.

At the first pass, we will use the baseline BERT to generate the 10-best candidates. Then at the second pass, we will use our algorithm to rerank the 10-best candidates. Our evaluation will focus on the improvement introduced by leveraging linguistic coordination. While there are some existing works on the same dataset from DSTC 8 that are also based on BERT, we do not use them as the first-pass baseline models mainly because the models are not publicly available.

Reranking with Linguistic Coordination

We focus on *lexical* coordination in this work. Inspired by Danescu-Niculescu-Mizil et al. (2012b), we adapt their linguistic coordination measure to our specific problem and data. Compared with their original measure, we modify their formula to measure lexical overlap at the group-level. Coordination between a group and a member is found in other multiparty conversation datasets. Compared to a non-group member, a group member will have higher vocabulary overlap with the group (Rahimi et al. 2017). Thus, we view the dialogue context and the response as a single turn exchange. Then we can adapt the measure to a group-level that measures coordination between a group and a member.

Considering that we only have a turn exchange, we further modify the original formula to address the word coordination in a single exchange. Equation 1 shows the coordination, denoted as $Coor_m$, of a specific word m in the context. Here m functions as one lexical marker to evaluate lexical coordination. The c and r here denote the dialogue context and the current response. The minuend is 1 when m is used in both the context and response. This indicates a word overlap occurs. Otherwise $Coor_m$ equals to 0. $P(\epsilon_r^m)$ reflects how likely this word is used in the current data set. K is a scalar that can be tuned accordingly.

Equation 2 shows the calculation of $P(\epsilon_r^m)$. $Count_r^m$ denotes the count of the marker m in the current responses. $Count^m$ denotes the count of this marker m in the vocabulary count. To ensure $P(\epsilon_r^m)$ can reflect the word usage in the Ubuntu datasets, we create a larger vocabulary set using dialogues from the original task development set, as well as the top 10 candidates selected by our best baseline model. We ignore a union of stop words[1], a list of interjections, English numbers, and the most 200 common words based on the development set performance. In the case of negative $Coor_m$ value, we let $Coor_m$ be 0 so that the range of $Coor_m$ is between 0 and 1. In equation 3, $Coor$ indicates the average coordination for all m when $Coor_m$ is not 0. The generic score G is generated by the baseline model. The final score weights the generic score G and the average coordination score $Coor$. We tune the weights w_g for G and w_{coor} for $Coor$ on the development set. We bypass reranking if a candidate scores are very high in G (>99% based on our development set) because a high value in G implies high confidence from the baseline model.

$$Coor_m = \begin{cases} 1 - K * P(\epsilon_r^m), & \text{m in c and r} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

$$P(\epsilon_r^m) = \frac{Count_r^m}{Count^m} \tag{2}$$

$$S = w_g * G + w_{Coor} * Coor \tag{3}$$

Experiments

Evaluation Metrics

Our response selection evaluation metrics are Recall@1 (R@1) and Recall@10 (R@10). Here the k in the Recall@k means that the true positive response is among the first k ranked candidates. We also include Mean Reciprocal Rank (MRR) to measure the general ranking quality.

Baseline Implementation Details

We use the BERT TensorFlow implementation from Google. The pre-trained model is BERT-base uncased, which we fine-tune in 3 epochs. The maximum sequence length is set to 128. The batch size is 128. We train 2 models. The learning rate is $2e^{-}5$ and $1e^{-}5$ for Model 1 and 2, respectively. Dropout rate is 0.1. Models are trained on one GPU.

Results and Discussion

We use Bert as the baseline and train two models, Model 1 (M1) and Model 2 (M2), to examine the robustness of our reranking method. Both use the same training configuration except for different learning rates, which results in M1 performing worse than M2 (Table 1).

Table 1 shows the results. Overall, R1 is improved by reranking for M1 and M2. Note that R@10 maintains the same before and after reranking because we only rerank the top 10 candidates. For the development set, with the best weights between generic and coordination score, M1 shows the most improvement in R@1 by 6.31%. MRR also increased by 4.22%. M2 shows a small improvement of only 0.86% in R@1. MRR increased by 0.63%. Using the weights found in the development set, the test set results are similar. M1 also shows the most improvement in R@1 by 6.14%. MRR increased by 4.21%. M2 shows again shows a small improvement of 1.18% in R@1. MRR increased by 0.68%. Lexical coordination supports the baseline BERT models to further disambiguate optimal answers, especially for weaker models. However, as the results of baseline models improve, the utility of reranking becomes weaker.

Development Set				
	Model 1		Model 2	
	BERT	Rerank	BERT	Rerank
R@1	31.82	**38.13**	41.93	**42.79**
R@10	64.22	64.22	70.19	70.19
MRR	43.12	**47.34**	51.65	**52.28**
Test Set				
	Model 1		Model 2	
	BERT	Rerank	BERT	Rerank
R1	31.00	**37.14**	39.74	**40.92**
R10	64.10	64.10	69.62	69.62
MRR	42.26	**46.47**	49.97	**50.65**

Table 1: The evaluation results for the development and test set. The recall is shown in percentage.

Table 2 shows the correct response distribution at the top three positions before and after reranking for the test set. For M1 we observe an increase in the 1st position but a decrease in both the 2nd and 3rd positions. One possible explanation is that correct response are initially ranked very closely to the 1st position, and the subsequent reranking slightly adjusts the order based on coordination. Similarly to M1, M2 also shows an increase in the 1st position, but it also shows an increase in the 3rd position. The result implies that some cases are incorrectly reranked.

	Model 1		Model 2	
Ranking	BERT	Rerank	BERT	Rerank
1	31.00	**37.14**	39.74	**40.92**
2	9.58	7.50	9.08	8.00
3	5.68	4.47	5.06	**5.13**

Table 2: The correct answer position in the test set

Case Study

We analyze the reranking outputs. Here error cases are those baseline cases that failed to rank the correct response as first place. For the test set, Table 3 shows the number of error cases that have linguistic coordination in the correct response (and thus potentially could have been corrected by our reranking method), as well as the number of actual corrections and errors after reranking. The caps are 11.13% and 5.82% for M1 and M2 respectively. The higher percentage for M1 indicates that for the weaker model, there is more space for reranking to improve.

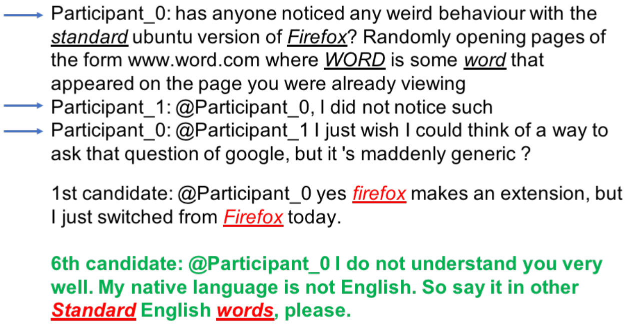
Figure 2 is a typical example of correction from selecting a low-coordinate response to a high-coordinate response. The correct response is in the 6th position. Both the 1st candidate and 6th candidates show some coordination at 'firefox', 'standard', and 'word'. The word 'standard' and 'words' are used less frequently in the dataset but they are both used in the 6th candidate. Thus, the score of the 6th

[1] From NLTK stop words dictionary

		Model 1	Model 2
Test Set	Cap	488	255
	Correction	360	102
	New Error	91	50

Table 3: Test set case analysis. Cap: # of error cases having coordination in the best candidate. Correction: # of corrections after reranking. New Error: # of new errors after reranking

candidate is in higher coordination than the first candidate. This implies that the reranker can capture the coordination in word usage beyond counting word repetition. This is also an interesting example in that the correct response is not directly related to the context but contains context words.



→ Participant_0: has anyone noticed any weird behaviour with the *standard* ubuntu version of *Firefox*? Randomly opening pages of the form www.word.com where *WORD* is some *word* that appeared on the page you were already viewing

→ Participant_1: @Participant_0, I did not notice such

→ Participant_0: @Participant_1 I just wish I could think of a way to ask that question of google, but it 's maddenly generic ?

1st candidate: @Participant_0 yes *firefox* makes an extension, but I just switched from *Firefox* today.

6th candidate: @Participant_0 I do not understand you very well. My native language is not English. So say it in other *Standard* English *words*, please.

Figure 2: A correction example from the test set. Italic text shows potential lexical coordination. Bold text shows the correct response.

Conclusion and Future Work

We proposed a simple yet effective approach to rerank the N-best candidates generated by a pre-trained language model. The results show promising improvement in selecting the correct response. By leveraging linguistic coordination, this work provides a way to rerank the response candidates in an unsupervised manner. Our approach does not increase the computational complexity, and can be applied when there is limited access to computational resources. In the future, we intend to use other state-of-the-art models in the second-pass reranking and experiment on more datasets.

References

Brennan, S. E. 1996. Lexical entrainment in spontaneous dialog. *Proceedings of ISSD* 96:41–44.

Chen, Q.; Zhu, X.; Ling, Z.; Wei, S.; Jiang, H.; and Inkpen, D. 2016. Enhanced lstm for natural language inference. *arXiv preprint arXiv:1609.06038*.

Danescu-Niculescu-Mizil, C.; Lee, L.; Pang, B.; and Kleinberg, J. 2012b. Echoes of power: Language effects and power differences in social interaction. In *Proceedings of the 21st international conference on World Wide Web*, 699–708.

Dario Bertero, Takeshi Homma, K. Y. M. I. K. N. 2020. Model ensembling of esim and bert for dialogue response selection. In *Proceedings of workshop of the 34th AAAI Conference on Artificial Intelligence (AAAI-20)*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Gonzales, A. L.; Hancock, J. T.; and Pennebaker, J. W. 2010. Language style matching as a predictor of social dynamics in small groups. *Communication Research* 37(1):3–19.

Kim, S.; Galley, M.; Gunasekara, C.; Lee, S.; Atkinson, A.; Peng, B.; Schulz, H.; Gao, J.; Li, J.; Adada, M.; et al. 2019. The eighth dialog system technology challenge. *arXiv preprint arXiv:1911.06394*.

Kummerfeld, J. K.; Gouravajhala, S. R.; Peper, J.; Athreya, V.; Gunasekara, C.; Ganhotra, J.; Patel, S. S.; Polymenakos, L.; and Lasecki, W. S. 2019. A large-scale corpus for conversation disentanglement. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Levitan, R., and Hirschberg, J. 2011. Measuring acoustic-prosodic entrainment with respect to multiple levels and dimensions. In *Twelfth Annual Conference of the International Speech Communication Association*.

Nenkova, A.; Gravano, A.; and Hirschberg, J. 2008. High frequency word entrainment in spoken dialogue. In *Proceedings of ACL-08: HLT, Short Papers*, 169–172.

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Rahimi, Z.; Kumar, A.; Litman, D. J.; Paletz, S.; and Yu, M. 2017. Entrainment in multi-party spoken dialogues at multiple linguistic levels. In *INTERSPEECH*, 1696–1700.

Wu, S.; Jiang, Y.; Wang, X.; Miao, W.; Zhao, Z.; Jun, X.; and Li, M. 2020. Enhancing response selection with advanced context modeling and post-training. In *Proceedings of workshop of the 34th AAAI Conference on Artificial Intelligence (AAAI-20)*.

Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R. R.; and Le, Q. V. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, 5754–5764.

Yoshino, K.; Hori, C.; Perez, J.; D'Haro, L. F.; Polymenakos, L.; Gunasekara, C.; Lasecki, W. S.; Kummerfeld, J. K.; Galley, M.; Brockett, C.; et al. 2019. Dialog system technology challenge 7. *arXiv preprint arXiv:1901.03461*.

Zhou, X.; Li, L.; Dong, D.; Liu, Y.; Chen, Y.; Zhao, W. X.; Yu, D.; and Wu, H. 2018. Multi-turn response selection for chatbots with deep attention matching network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1118–1127.