Technical Report on Shared Task in DialDoc21

Jiapeng Li[*], **Mingda Li**[*], **Longxuan Ma**[*], **Weinan Zhang**[†], **Ting Liu**
Research Center for Social Computing and Information Retrieval
Harbin Institute of Technology, Harbin, Heilongjiang, China
{jpli,mdli,lxma,wnzhang,tliu}@ir.hit.edu.cn

Abstract

We participate in the DialDoc Shared Task sub-task 1 (Knowledge Identification). The task requires identifying the grounding knowledge in form of a document span for the next dialogue turn. We employ two well-known pre-trained language models (RoBERTa and ELECTRA) to identify candidate document spans and propose a metric-based ensemble method for span selection. Our methods include data augmentation, model pre-training/fine-tuning, post-processing, and ensemble. On the submission page, we rank 2nd based on the average of normalized F1 and EM scores used for the final evaluation. Specifically, we rank 2nd on EM and 3rd on F1.

1 Introduction

Our team SCIR-DT participates in the DialDoc shared task in the Document-grounded Dialogue and Conversational QA Workshop at the ACL-IJCNLP 2021. There are two sub-tasks based on the Doc2Dial dataset (Feng et al., 2020). The dataset contains goal-oriented conversations between a user and an assistive agent. Each dialogue turn is annotated with a dialogue scene, which includes role, dialogue act, and grounding in a document (or irrelevant to domain documents). The documents are from different domains, such as Social Security and Veterans Affairs. Sub-task1 is **Knowledge Identification** which requires identifying the grounding knowledge in form of document span for the next agent turn. The input is dialogue history, current user utterance, and associated document. The output should be a text span. The evaluation metrics are Exact Match (EM) and F1 (Rajpurkar et al., 2016). Sub-task2 is text generation which requires generating the next agent response in natural language. The input is dialogue history and associated document. The output is agent utterance. The evaluation metrics are SacreBLEU (Post, 2018) and human evaluations. We only participate in sub-task 1.

[*]These three authors contributed equally.
[†]Corresponding author.

2 Related Work

2.1 Document-grounded Dialogue (DGD) & Conversational QA (CQA)

The DGD maintains a dialogue pattern where external knowledge used in dialogues can be obtained from the given document. Recently, some DGD datasets (Moghe et al., 2018; Dinan et al., 2019) have been released to exploiting unstructured document information in open-domain dialogues. The Doc2Dial dataset is also document-grounded dialogue. However, the dialogue in Doc2Dial is goal-oriented which guides users to access various forms of information according to their needs.

The CQA (such as CoQA (Reddy et al., 2019), QuAC (Choi et al., 2018) and DoQA (Campos et al., 2020)) task is also based on background document, which aims to understand a text passage and answering a series of interconnected questions that appear in a conversation. The difference between DGD and CQA is the dialogue of DGD is more diversified (including chit-chat or recommendation) and not limited to QA. The Doc2Dial task is closely related to the CQA tasks. It shares the challenges and additionally introduces the dialogue scenes where the agent asks questions when the user query is identified as under-specified or additional verification required for a resolute solution.

2.2 Pre-trained Language Model (PLM)

The traditional word embeddings (Pennington et al., 2014) are fixed and context-independent, they could not resolve the out-of-vocabulary (OOV) problem and the ambiguity of words in different contexts. To address these problems, Pre-trained

Proceedings of the 1st Workshop on Document-grounded Dialogue and Conversational Question Answering, pages 52–56

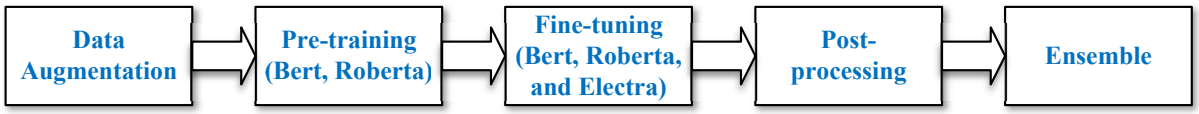

Figure 1: The pipeline methods we used in the competition.

Language Models (PLMs) such as BERT (Devlin et al., 2019) were introduced. BERT employed a Masked language modeling (MLM) method that first masked out some tokens from the input sentences and then trained the model to predict the masked tokens by the rest of the tokens. Concurrently, there was research proposing different enhanced versions of MLM to further improve on BERT. Instead of static masking, RoBERTa (Liu et al., 2019) improved BERT by dynamic masking and abandoned the Next Sentence Prediction (NSP) loss. Instead of masking the input, ELECTRA (Clark et al., 2020) replaced some input tokens with plausible alternatives sampled from a small generator network and trained a discriminative model that predicted whether each token in the corrupted input was replaced by the generator or not. When used for downstream tasks, these PLMs were first trained on a large corpus, then fine-tuned on specific tasks. The contextualized embedding has been proven to be better for the downstream NLP tasks (Qiu et al., 2020) than traditional word embedding. We adopt the BERT, RoBERTa, and ELECTRA in this competition.

3 Our Method

We first use two data augmentation methods to obtain a 5-times larger augmented dataset. We use the augmented data to re-train BERT and RoBERTa with the whole word masking technique and fine-tune BERT, RoBERTa, and ELECTRA models. We test several span post-processing methods and then propose an ensemble method with trainable parameters for final text span selection. The pipeline we used in this competition is illustrated in Figure 1.

3.1 Problem Statement

In sub-task 1, we focus on selecting the correct text span as knowledge from a document. For each example, the model is given a conversational context $\mathbf{C} = [C_1, C_2, ..., C_{|C|}]$ with $|C|$ turns from different speakers and a document $\mathbf{K} = [K_1, K_2, ..., K_{|K|}]$ with $|K|$ spans as external knowledge. Each span is labeled with start and end positions in $\mathbf{K}$. The

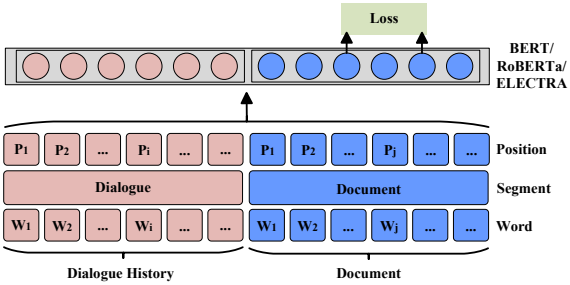

Figure 2: The models we used in the competition.

Table 1: Doc2Dial dataset statistics.

dataset	documents	dialogues	turns
Train	488	3474	44149
Validation	488	661	8539
dev-test	488	198	1353
final-test	573	787	5264

model learns to select a document span K_i for the response with probability $P(K_i|\mathbf{K}, \mathbf{C}; \Theta)$, Θ is the model's parameters. Specifically, our model adopts the BERT-QA (Chadha and Sood, 2019) method and predicts the start and end positions of a span, if the predicted positions are not the boundaries of an existing span, we use some post-processing methods to modify them to the nearest K_i. The selected span K_i is used for sub-task 2 to generate a response. The model structure is shown in Figure 2. The input of the model is the sum of positional/segment/word embedding of dialogue and document. The output is a document span.

3.2 Data augmentation

The statistics of the Doc2Dial dataset are shown in Table 1. The final test set has an unseen domain that is not included in the training set. Besides the final test page, the organizers provide a dev-test page that uses a small set for additional testing. We use back-translation and Synonym substitution as data augmentation methods. We adopt the google translation service[1] to translate English into other languages (such as Spanish/German/Japanese/French),

[1]https://translate.google.com

then back-translated them into English[2]. Finally, we obtain 5-times document+dialogue data to pre-train the PLMs. Then we pair the 5-times dialogue data with documents translated from different languages, which gives 25 times data for fine-tuning.

3.3 Pre-training and Fine-tuning

We use the augmented data to pre-train two models: BERT and RoBERTa. We follow the Masked Language Model method with the whole word masking technique. We do not pre-train the ELECTRA model because we hope our ensemble method could leverage the prediction results from RoBERTa and ELECTRA to achieve a good performance on both seen and unseen domains. We pre-train RoBERTa on the augmented data to get a good performance on the seen domains. Meanwhile, we hope that ELECTRA can get a good prediction on the unseen domain. The unseen domain in the final-test set requires the knowledge packed in the parameters of the pre-trained model. Pre-training ELECTRA will lose this knowledge.

When fine-tuning these models (BERT, RoBERTa, and ELECTRA), the model structure and training objective is the same as the common method used in the span-extraction Reading Comprehension task. The training objective is defined as the sum of negative log probabilities of the true start and end positions by the predicted distributions, averaged over all N examples:

$$L = -\frac{1}{N}\sum_{n=1}^{N}[\log P(S_n^{start}) + \log P(S_n^{end})], \quad (1)$$

where S_n^{start} and S_n^{end} are the ground-truth span start and end positions of the n-th example .

3.4 Post Processing

Since the document is divided into consecutive spans and the task requires identifying a single span, we propose two different post-processing methods to fix the wrong predictions. The goal of these methods is to process the predicted incomplete span into a complete one. The first method is to expand the predicted start/end to the boundary of one standard span when the predicted positions are within it. The second is to move the predicted start/end to the boundary of the nearest span when the predicted positions are across two spans.

[2]When the back-translation sentence is the same as the original sentence, we employ synonym substitution with Wordnet (https://wordnet.princeton.edu/) to increase diversity.

3.5 Ensemble Method

Algorithm 1: Metric-based ensemble method.

```
During training: Metric = F1 or EM;
Input: S^R, S^E, S, W̃^R, W̃^E, S_gt.
Output: Weight for each model.
for p ∈ range(start=0, stop=1, step=0.1) do
    Score = 0
    for k ∈ {validation set} do
        Initialize W: {W_i = 0, i = 1, 2, ..., T}
        for i ∈ [1, T]; do
            W_i = p · W̃_i^R + (1 − p) · W̃_i^E
        end for
        Score += Metric(S_argmax(W), S_gt)
    end for
    Record weight p* for the Best Score.
end for
During test:
for k ∈ {test set} do
    Initialize W: {W_i = 0, i = 1, 2, ..., T}
    for i ∈ [1, T]; do
        W_i = p* · W̃_i^R + (1 − p*) · W̃_i^E
    end for
    S_k = S_argmax(W)
end for
```

We propose a simple but efficient ensemble method (**Algorithm 1** shows the details) to utilize the advantages of different models. For each example, we calculate top N span candidates from each model and sort them in descending order with respect to model confidence. Each span is given a weight which is the reciprocal of its ranking number plus one. For example, candidates from RoBERTa are S_j^R, $(j = 1, 2, ..., N)$, and the corresponding weight is $W_j^R = \frac{1}{j+1}$. Similarly, S_j^E and W_j^E for ELECTRA. Then we use these candidates to form a final candidate dictionary S_i, $(i = 1, 2, ..., T)$, $N \leq T \leq 2N$, and the ensemble weight W_i of S_i, is calculated by W_i = $p \cdot \tilde{W}_i^R$+$(1-p) \cdot \tilde{W}_i^E$, $(i = 1, 2, ..., T)$. p is a hyperparameter and W_i^R = W_j^R if there is a j such that $S_j^R \cong S_i$, 0 otherwise. $\cong$ means exact match here and $\tilde{W}_i^E$ follows the same definition. Then we use a specific metric, such as F1 or EM, to learn the optimal p* with all examples in the validation set. When testing, we select one candidate as our final prediction using the learned weight[3].

[3]For example, a text span ranks 3rd in RoBERTa and ranks 4th in ELECTRA, p*=0.2, then the final weight to re-rank this span in S is 0.2*0.25+0.8*0.2 = 0.21.

Table 2: Experimental results. "DA/FT/PT/PP" means "data augmentation/fine-tuned/pre-trained/post-processing", respectively.

Models	On dev-test set		On final-test set	
	F1%	EM%	F1%	EM%
BERT (baseline - w/o DA)	66.84	48.48	66.45	48.67
BERT (FT)	67.62	50.01	67.29	49.82
RoBERTa (FT)	71.86	56.77	70.46	54.23
ELECTRA (FT)	72.51	57.58	70.91	54.64
RoBERTa (PT/FT)	72.08	60.10	71.55	58.70
ELECTRA (FT/PP)	72.79	58.08	71.27	55.65
RoBERTa (PT/FT/PP)	72.37	60.61	71.57	59.09
RoBERTa (PT/FT/PP) + ELECTRA (FT/PP)	**74.09**	**63.13**	**75.64**	**63.91**

4 Experiments and Analysis

4.1 Experimental Settings

Our implementations of BERT, RoBERTa, and ELECTRA are based on the public Pytorch implementation from Transformers[4]. All models are in large size. During pre-training, we follow the hyper-parameters setting of the original implementation. During fine-tuning, we truncated the length of the dialogue context to 60 tokens and maximum input length to 512 tokens. The maximum predicted span length is set to 90 words. Candidate span size N is set to 20. We use EM as the **Metric** in the ensemble method. We use a single Tesla v100s GPU with 32gb memory, the pre-training time is around 48 hours and fine-tuning time is around 24 hours for each model.

4.2 Experimental Results and Analysis

In this competition, each team has five submission opportunities on the final test page[5]. Table 2 shows the experimental results on dev-test/final-test sets of different models. The baseline given by the organizer is a BERT-large model without pre-trained on Doc2Dial data, we fine-tune the baseline on the training set of Doc2Dial data and get the F1 of 66.84 and EM of 48.48 on the dev-test set. When using augmented data to fine-tune the BERT-large model, we get 67.62 F1 and 50.01 EM. The results prove the effectiveness of dialogue data augmentation. We fine-tune RoBERTa and ELECTRA with the augmented data and they both outperform BERT. We use augmented data to pre-train the RoBERTa model before we fine-tune it. The F1 and EM increase to 72.08 and 60.10, respectively. It proves that pre-training on task data can further improve performance. Then we find Post-processing helps ELECTRA on both F1 and EM. We employ the PT/FT/PP on RoBERTa and get 72.37 F1 and 60.61 EM. At last, we employ our ensemble method on the best performance RoBERTa and ELECTRA models and achieve 74.09 F1 and 63.13 EM on the dev-test set. The last method also achieves our best F1 and EM on the final-test set, the ensemble results outperform the best single model (RoBERTa) more than 4% on both F1 and EM. For EM, the contribution ranks from big to small are Ensemble>Pre-training>Data Augmentation>Post Processing.

The ensemble method uses both PLM (RoBERTa) that is pre-trained with augmented data and PLM (ELECTRA) that is not pre-trained with augmented data. In this way, we can leverage the knowledge packed in the parameters of ELECTRA for the unseen domain of the final-test data. The ELECTRA(FT/PP) got an EM of 55.65 on the final-test set and the RoBERTa(PT/FT/PP) got an EM of 59.09. The ensemble method increased the EM to 63.91, indicating that the two models have a great difference of choice in spans and our ensemble method leverages the difference between the two models to achieve a better result.

5 Conclusion

We introduced our submission for Doc2Dial Shared Task. In sub-task 1, our model is based on RoBERTa and ELECTRA. We propose a simple but efficient ensemble method for knowledge selection in multi-turn dialogue. Our team SCIR-DT ranks 2nd on the final submission page. Apart from the methods we introduced, there are other methods that could further improve the performance of

[4]https://github.com/huggingface/transformers

[5]Each team has 20 more submission opportunities after the competition to help finish their technical report.

our model. For example, Feng et al. (2020) proved the dialogue act information was useful for subtask 1; there are some noisy data such as empty responses in the dialogue data could be filtered out during training; employing machine reading comprehension dataset such as SQuAD (Rajpurkar et al., 2016) or CQA dataset such as CoQA (Reddy et al., 2019) for pre-training and fine-tuning may also be helpful. However, due to the time limitation, we did not try all these methods during the competition. We hope these methods and experiences would be helpful for future contestants.

Acknowledgments

We thank the thoughtful suggestions from the reviewers. This paper is supported by the National Natural Science Foundation of China (No. 62076081, No. 61772153, and No. 61936010) and the Science and Technology Innovation 2030 Major Project of China (No. 2020AAA0108605).

References

Jon Ander Campos, Arantxa Otegi, Aitor Soroa, Jan Deriu, Mark Cieliebak, and Eneko Agirre. 2020. Doqa - accessing domain-specific faqs via conversational QA. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7302–7314. Association for Computational Linguistics.

Ankit Chadha and Rewa Sood. 2019. BERTQA - attention on steroids. *CoRR*, abs/1912.10435.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. Quac: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2174–2184. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Song Feng, Hui Wan, R. Chulaka Gunasekara, Siva Sankalp Patel, Sachindra Joshi, and Luis A. Lastras. 2020. doc2dial: A goal-oriented document-grounded dialogue dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8118–8128. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Nikita Moghe, Siddhartha Arora, Suman Banerjee, and Mitesh M. Khapra. 2018. Towards exploiting background knowledge for building conversation systems. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2322–2332. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 186–191. Association for Computational Linguistics.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *CoRR*, abs/2003.08271.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2383–2392. The Association for Computational Linguistics.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. Coqa: A conversational question answering challenge. *Trans. Assoc. Comput. Linguistics*, 7:249–266.