TREND: Trigger-Enhanced Relation-Extraction Network for Dialogues

Anonymous ACL submission

Abstract

The goal of dialogue relation extraction (DRE) is to identify the relation between two entities in a given dialogue. During conversations, speakers may expose their relations to certain entities by some clues, such evidences called "triggers". However, none of the existing work on DRE tried to detect triggers and leverage the information for enhancing the performance. This paper proposes **TREND**, a multi-tasking BERT-based model which learns to identify triggers for improving relation extraction. The experimental results show that the proposed method achieves the state-of-the-art on the benchmark datasets. [1]

1 Introduction

The goal of relation extraction (RE) is to identify the semantic relation type between two mentioned entities from a given text piece, which is one of the basic and important natural language understanding (NLU) problems (Zhang et al., 2017; Zhou and Chen, 2021; Cohen et al., 2020). In terms of the problem setting, we are usually given a written sentence and a query pair containing two entities and asked to return the most possible relation type from a predefined set of relations. Dialogue Relation Extraction (DRE), on the other hand, aims to excavate underlying cross-sentence relation in natural human communications (Yu et al., 2020; Jia et al., 2020). The problem itself is well-motivated, relations between entities in dialogues could potentially provide dialogue systems with additional features for better dialogue management (Peng et al., 2018; Su et al., 2018a) and generating more appropriate responses (Su et al., 2018b).

Figure 1 illustrates an example of the recently-proposed dataset, DialogRE (Yu et al., 2020). Given a conversation and a query pair, we aim to identify the interpersonal relationship between the entities, the entities can be not only human but other types of entities like locations. Furthermore, with longer context than a single sentence, DialogRE also has annotation on the evidences of relations within conversation flow, called **Triggers**. A trigger can be a short phrase or even a single word, and different part-of-speech of words are possible. In the example, the clue for knowing Speaker 3 has negative impression on Speaker 3 is that Speaker 2 once said *"You are arrogant."* Such hint is intuitively useful for identifying the relations. However, none of previous work tried to explicitly leveraged the trigger information for DRE.

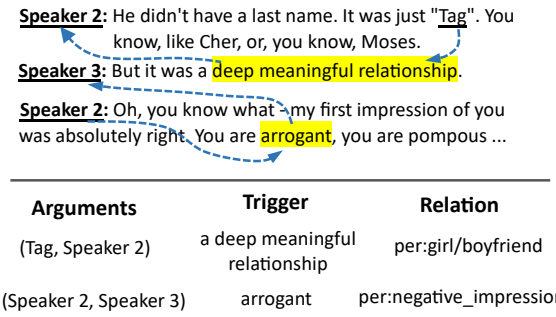

Figure 1: An example of dialogue relation extraction from DialogRE dataset, the blue dashed arrow lines connect the subjects, the triggers, and the objects. Triggers are clues of relations between entities, the DialogRE dataset has annotation of them.

Prior work can be divided into two main lines, one of which is graph-based methods. DHGAT (Chen et al., 2020) presents an attention-based heterogeneous graph network to model multiple types of features; GDPNet (Xue et al., 2020b) construct latent multi-view graphs to model possible relationships among tokens in a long sequence, and then to refine the graphs by iterative graph convolution and special graph pooling techniques. Another branch is BERT-based (Devlin et al., 2018) methods (Yu et al., 2020; Xue et al., 2020a), the backbones of the model architectures are BERTs. SimepleRE

[1] The source code will be released once accepted.

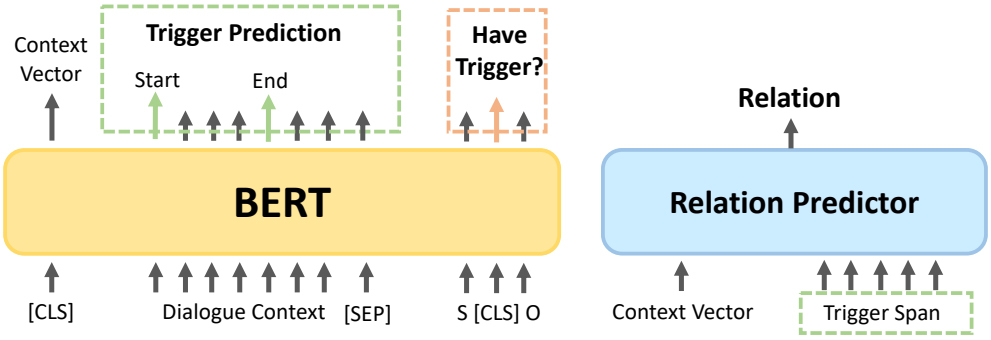

Figure 2: The proposed method is composed of two components: (1) a multi-tasking BERT with two fine-tuning tasks (trigger prediction, and prediction of having a trigger or not), and (2) a relation predictor with feature fusion by attention.

(Xue et al., 2020a) is a simple BERT model with an additional refinement gate for iteratively finding high-confidence prediction. LSR (Nan et al., 2020) proposed a latent structure refinement method for better reasoning in the document-level relation extraction task.

In this work, we propose **TREND**, a multi-tasking model base on BERT with an attentional relation predictor, where we design some auxiliary tasks for trigger prediction. Specifically, TREND has (1) extractive-style trigger identification by predicting start-end pointers, and (2) binary classifier for existence of triggers.The proposed methods are simple and flexible, and the experimental results show that our model achieves the state-of-the-art on DialogRE (Yu et al., 2020) and DDRel (Jia et al., 2020).

2 Proposed Method

The core idea of this work is to identify trigger spans and accordingly leverage the information of them for improving the relation extraction. We hereby propose **T**rigger-enhanced **R**elation-**E**xtraction **N**etwork for **D**ialogues, **TREND**.

2.1 Problem Formulation

Given a piece of dialogue context $\mathcal{D}$ composed of text tokens $\mathcal{D} = \{x_i\}$ and a query pair q containing a subject entity and an object entity $q = (s, o)$, we aim to find a function f to find the most possible relations between the entities from a predefined set of relations $\mathcal{R}$,

$$f(\mathcal{D}, q) \rightarrow \mathcal{R}.$$

Note that a single query pair can have multiple relations but we follow the setting of previous work where if a query has multiple relation labels, it will be split into multiple data samples with the same input $(\mathcal{D}, q)$ and different single target label.

2.2 TREND

The proposed TREND has two main modules, (1) a multi-tasking BERT (Devlin et al., 2018) for encoding context and identifying triggers, and (2) a relation predictor for predicting relation by fusing the context feature and the trigger span.

Trigger Prediction As illustrated in Figure 2, an input $(\mathcal{D}, q)$ will be first augmented into a BERT-style sequence. Specifically, the format is "[CLS] $\mathcal{D}$ [SEP] s [CLS] o", [CLS] and [SEP] are classification and separator special tokens, respectively. We also follow the method in (Yu et al., 2020) to replace the speaker tokens in $\mathcal{D}$. The [CLS] tokens at different position in the sequence may carry different meaning after encoding by BERT. In our model, we assume the encoding of the [CLS] token in the beginning of the sequence contains contextual information of whole input sequence.

Since triggers certainly exist in the input dialogue context, we propose to use an extractive-style method by predicting start-end pointers (Devlin et al., 2018), which is prevalent in Question-Answering area (Lee et al., 2016; Rajpurkar et al., 2016). The task is a single-label classification problem of predicting the most possible positions, hence the cross entropy loss is conducted.

Binary Gate Not every given query has a relation, in these cases, the labels are "Unanswerable". Certainly, such samples would not have trigger annotations. We hereby propose to learn a binary

	Model	DialogRE	DDRel					
			4-class		6-class		13-class	
		F1	Acc	Macro-F	Acc	Macro-F	Acc	Macro-F
(a)	CNN	48.0	42.7 / 47.3	33.3 / 35.0	37.8 / 38.5	31.5 / 30.4	32.3 / 22.2	9.2 / 7.1
(b)	BERT	60.6	47.1 / 58.1	44.5 / 52.0	41.9 / 42.3	39.4 / 38.0	39.4 / 39.7	20.4 / 24.1
(c)	GDPNet	64.3	-	-	-	-	-	-
(d)	SimpleRE	*60.4	-	-	-	-	-	-
(e)	**TREND**	**66.8**	51.5 / **65.4**	**46.5 / 61.2**	40.3 / **52.6**	**43.0 / 55.0**	**40.5 / 46.2**	**21.2** / 34.7
(f)	**TREND-L**	**67.8**	**51.6** / 60.3	**46.5** / 54.0	**42.5** / 46.2	**43.0** / 48.2	34.4 / 43.6	19.9 / **36.3**
(g)	(e) - BG	65.2	52.5 / 53.8	45.3 / 49.7	37.0 / 43.6	41.8 / 45.9	36.6 / 43.6	26.4 / 36.3
(h)	(f) - BG	66.2	41.5 / 47.4	40.3 / 44.9	39.0 / 42.3	43.1 / 42.9	38.5 / 34.6	17.3 / 21.1

Table 1: The performance of the models on automatic metrics, the official DDRel has different level (session-level/pair-level) and different granularity of evaluation settings (4,6,13-class). * Though SimpleRE reports 66.7 in the paper, their problem setting is different from the others; here we take the one with the same setting for fair comparison, we will detail this in Section 3.

classifier as a gate, if the binary gate gives over 0.5 score, we suppose the given sample does not have triggers and accordingly use empty trigger spans for prediction. The binary cross entropy loss is conducted as the loss function.

Relation Predictor Now we have a context vector (encoded [CLS] token) and a predicted trigger span, we then feed them into the predictor for relation prediction, as depicted in Figure 2. The features are fused by the following generic attention mechanism, the query is the context vector and the keys and the values are trigger words:

$$\sum \text{softmax}(\mathbf{c} \cdot \mathbf{x_i}) \cdot \mathbf{x_i},$$

where $\mathbf{c}$ is the context vector and $\mathbf{x_i}$ is the BERT encoding of words. The merged feature is then fed into a 1-layer feed-forward network for final relation prediction. Because the task is a single-label classification problem, hence the cross entropy loss is conducted.

Finally, all the losses from the above objectives are linearly combined and the whole model can be trained in an end-to-end manner. For each objective, we have a weight parameter to adjust the impact of it. We also apply schedule sampling (Bengio et al., 2015) on trigger prediction and binary signal when feeding into the relation predictor.

3 Experiments

In all the experiments, we use mini-batch Adam with learning rate $3e{-}5$ as the optimizer on Nvidia Tesla V100. The ratio of teacher forcing and other hyper-parameters were selected by grid search in (0,1] with step 0.1. The whole training takes 30 epochs without any early-stop method. The entire implementation was based on PyTorch and HuggingFace transformers[2] package. Other details will be reported in Appendix A.

Speaker 1: What's up?		
Speaker 2: Monica and I are **engaged**.		
Speaker 1: Oh my God. Congratulations.		
Speaker 2: Thanks.		
Argument	**Relation**	**Predicted Trigger**
(Speaker 2, Monica)	girl/boyfriend	engaged

Table 2: An example of the predicted results of our model on the DialogRE dataset.

3.1 Datasets

The benchmark datasets conducted in our experiments are DialogRE (Yu et al., 2020) and DDRel (Jia et al., 2020), both are DRE datasets. The official DialogRE dataset has two versions, we chose the latest version (v2) of English part. Since the conversations in DialogRE are quite natural and colloquial, the preprocessing process includes text normalization like lemmatization and expanding contractions. Because of the different characteristics of the datasets, the batch size for DialogRE is 16 while the one for DDRel is 4.

3.2 Analysis

The experimental results are shown in Table 1. In our experiments, we take CNN (row (a)), BERT (row (b)), GDPNet (row (c)) (Xue et al., 2020b), and SimpleRE (row (d)) (Xue et al., 2020a) as the baselines for comparison. GDPNet and SimpleRE

[2] https://huggingface.co/transformers/

did not conduct the DDRel dataset in their experiments, so we only report the performance of CNN and BERT. Although SimpleRE reported 66.7 F1-score of their best-performing model in their paper, their problem setting is different from the other work. Specifically, they concatenate all the argument pairs of a dialogue sample as a long query and likewise append it after the dialogue context. Because a dialogue sample could have up to 20 argument pairs, SimpleRE proposes to augment more contextual information by means of the concatenation. For fair comparison, we take the same setting of SimpleRE taking a single argument pair.

TREND (row (e)) utilizes BERT-base model while TREND-L (row (f)) is based on the BERT-large, both models outperform all the baselines on DialogRE and achieve the state-of-the-art, and TREND-L could further improve the performance for 1.0 F1-score. Unlike SimpleRE (row (d)) and GDPNet (row (c)) need to iteratively refine the latent features or latent graphs, the prediction of the proposed TREND is straight-forward. Such design makes training and inference efficient and robust. Table 2 shows a generated example of our model, in this example, our model successfully identify the correct trigger and hereby help the model to predict the right relation. In terms of exact-match, the trigger prediction of our model still has lots of space to improve, however, exact-match is not really necessary. For instance, if the ground truth trigger is "Mom" but the predicted trigger is "Dad", it could still facilitate the prediction regarding the label might be "parent". Partial matches are another case, if the ground truth trigger is "got married" but the predicted trigger is "married", such prediction apparently helps.

Our trained binary gate has about 85% accuracy while the trigger prediction has no more than 50% accuracy. Though these sub-modules are not perfectly-trained, we found them somewhat useful by the ablation test (row (g)-(h)). To examine the effectiveness of our modeling, we further try to estimate the upper-bound performance by using the ground truth trigger spans for final relation prediction. The estimated upper-bound of TREND (row (e)) is 75.3, in other words, our design of relation predictor is validated and the potential of the proposed model could be unleashed once we enhance the trigger prediction.

Transfer Learning Since the DDRel dataset does not provide annotations of triggers, the reported numbers in (row (e)-(h)) are the transferred results where the models were first pre-trained on DialogRE and then fine-tuned on DDRel. Since the output space is different, the last prediction layer is replaced. From Table 1, we can see that TREND ((row (e)-(f))) are the best-performing models in all the evaluation settings. Especially for pair-level evaluation, which takes much longer dialogue context as input, the improvement over the baselines is more. We suppose this is because when longer context is provided, extracting key evidences becomes more important to overcome information overload. Suprisingly, TREND-L does not keep outperforming TREND. Table 3 is an example of the predicted results of our model adapted to DDRel, the model identify the word "father" as the trigger, which is reasonable for the target relation "Child-Parent". All the results show that TREND is capable of transferring learned knowledge to a new dataset and new domain.

"BETSY: That's all.",		
"JIM: That's all?!"		
"BETSY: You don't see it, do you, **father**?"		
"JIM: No. Fellow wants to sell a house ...		
Argument	**Relation**	**Predicted Trigger**
(BETSY, JIM)	Child-Parent	father

Table 3: An example of the predicted results of our model adapted to the DDRel dataset.

4 Conclusion

In this paper, we propose TREND, a multi-tasking model predicting relation triggers for improving dialogue relation extraction. TREND is a simple, flexible, end-to-end model based on BERT, which has three main components: (1) extractive-style trigger indetifier by predicting start-end pointers, (2) binary classifier for existence of triggers, and (3) a relation predictor with attentional feature fusion. On the DRE benchmark datasets, DialogRE and DDRel, the proposed method achieves the state-of-the-art performance. The experiment results also show that the proposed TREND: (1) can transfer the learned knowledge from DialogRE to DDRel, extracting the informative evidence without further instruction, and (2) has great potential to boost performance more based on the proposed ideas.

References

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 1171–1179.

Hui Chen, Pengfei Hong, Wei Han, Navonil Majumder, and Soujanya Poria. 2020. Dialogue relation extraction with document-level heterogeneous graph attention networks.

Amir DN Cohen, Shachar Rosenman, and Yoav Goldberg. 2020. Relation classification as two-way span-prediction. *arXiv preprint arXiv:2010.04829*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Qi Jia, Hongru Huang, and Kenny Q Zhu. 2020. Ddrel: A new dataset for interpersonal relation classification in dyadic dialogues. *arXiv preprint arXiv:2012.02553*.

Kenton Lee, Shimi Salant, Tom Kwiatkowski, Ankur Parikh, Dipanjan Das, and Jonathan Berant. 2016. Learning recurrent span representations for extractive question answering.

Guoshun Nan, Zhijiang Guo, Ivan Sekulic, and Wei Lu. 2020. Reasoning with latent structure refinement for document-level relation extraction. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Baolin Peng, Xiujun Li, Jianfeng Gao, Jingjing Liu, Kam-Fai Wong, and Shang-Yu Su. 2018. Deep dyna-q: Integrating planning for task-completion dialogue policy learning. *arXiv preprint arXiv:1801.06176*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Shang-Yu Su, Xiujun Li, Jianfeng Gao, Jingjing Liu, and Yun-Nung Chen. 2018a. Discriminative deep dyna-q: Robust planning for dialogue policy learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3813–3823.

Shang-Yu Su, Kai-Ling Lo, Yi-Ting Yeh, and Yun-Nung Chen. 2018b. Natural language generation by hierarchical decoding with linguistic patterns. In *Proceedings of The 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Fuzhao Xue, Aixin Sun, Hao Zhang, and Eng Siong Chng. 2020a. An embarrassingly simple model for dialogue relation extraction. *arXiv preprint arXiv:2012.13873*.

Fuzhao Xue, Aixin Sun, Hao Zhang, and Eng Siong Chng. 2020b. Gdpnet: Refining latent multi-view graph for relation extraction.

Dian Yu, Kai Sun, Claire Cardie, and Dong Yu. 2020. Dialogue-based relation extraction. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45.

Wenxuan Zhou and Muhao Chen. 2021. An improved baseline for sentence-level relation extraction. *arXiv preprint arXiv:2102.01373*.

A Training Details

A.1 Hyperparameters

All the hyper-parameters were selected by grid search in (0,1] with step 0.1. The loss functions are linearly combined and each of them has a adjustable weight.

TREND

- Loss: $0.3 \cdot \mathcal{L}_{\text{trigger}} + 1.0 \cdot \mathcal{L}_{\text{relation}} + 1.0 \cdot \mathcal{L}_{\text{binary}}$
- schedule sampling: 0.7 for trigger prediction, 0.7 for binary classification

TREND-L

- Loss: $0.3 \cdot \mathcal{L}_{\text{trigger}} + 1.0 \cdot \mathcal{L}_{\text{relation}} + 1.0 \cdot \mathcal{L}_{\text{binary}}$
- schedule sampling: 0.5 for trigger prediction, 0.7 for binary classification

A.2 Cost of Time

DialogRE

- Training: 15min$\times$30
- Inference: 5min

DDRel (session-level)

- Training: 15min$\times$30
- Inference: 5min

DDRel (pair-level)

- Training: 1.5min$\times$30
- Inference: 10s