Definition Extraction with Pre-trained RoBERTa Model

Zhuo Li
Key Lab of High Confidence Software Technologies
lizhmq@pku.edu.cn

Abstract

We describe our contribution to the three subtasks of SemEval 2020 Task 6, *DeftEval: Extracting term-definition pairs in free text*. We explore the performance of using pre-trained RoBERTa model for definition extraction. For *Subtask 1: Sentence Classification*, we fine-tune the RoBERTa model to learn sentence representations for the downstream binary classification task. For *Subtask 2: Sequence Labeling*, we use RoBERTa to generate the contextual representation for each token and pass the representations to a feed-forward neural network with CRF layer for BIO tag classification. The system for *Subtask 3: Relation Extraction* is based on the fine-tuned RoBERTa model in the previous tasks and a Random Forest classifier. Our systems achieve respectively 0.785, 0.526 and 0.477 F_1-scores on the official test set, which rank the 29th, 22nd and 23rd in the submissions, respectively. We open source our code at: https://github.com/Lizhmq/DeftEval.

1 Introduction

Definition Extraction is the task to automatically extract terms and their definitions from text. It has been a hot research topic in natural language processing (NLP) community and attracted numerous researchers' work. However, definition extraction is a challenging problem due to the flexibility of free text or semi-structured text. To help address this challenge, Spala et al. (2020) introduced the Definition Extraction from Texts (DEFT) corpus, a human-annotated English dataset that contains term-definition pairs of different domains (*e.g.*,) from two types of documents - Textbook and Contracts. In addition, SemEval-2020 proposed a shared task which was aimed at evaluating the performance of submitted systems on three subtasks of definition extraction. The three subtasks are the following:

Subtask1: Sentence Classification. This subtask is to predict whether there is a formal definition in a sentence. For example, The following sentence "In short, **collective behavior** is any group behavior that is not mandated or regulated by an institution." gives a formal definition to "**collective behaviour**" and is labeled as a positive example. On the contrary, the sentence "There are three primary forms of collective behaviour: the crowd, the mass, and the public." enumerates three typical examples of collective behaviour but does not contain a formal definition. We are required to give a binary classifier to detect definitions in sentences in this subtask. The F_1-score of the positive class is reported as the final result.

Subtask2: Sequence Labeling. Given a dataset of tokenized sentences, we aim to label each token with one of the following classes: Term, Alias-Term, Referential-Term, Definition, Referential-Definition, Qualifier, or None. The average value of F_1-scores of the six positive classes is the evaluation metrics of subtask2.

Subtask3: Relation Extraction. Given a dataset of tokenized sentences and the tag id of each token, our goal is to predict, for each token, the type of relationship it had with another token, and also the tag id of the token it had a relation with. This is a classical relation extraction task and the relations that are considered for evaluation include: Direct-defines, Indirect-defines, Refers-to, AKA, Supplements or None. Similar to subtask2, the evaluation metrics is the average F_1-score of the five positive classes.

To tackle the three subtasks, we employ the powerful pre-trained RoBERTa model to learn meaningful sentence embeddings and contextual token embeddings and use them for downstream classification.

For subtask1, we utilize the representation of the special token "[CLS]" produced by RoBERTa as representation of the input sentence and feed it into a feed-forward neural network for binary classification. Furthermore, we compare the performance of RoBERTa with several traditional machine learning methods (*e.g.*, Naive Bayesian, Decision Tree) and demonstrate the effectiveness of RoBERTa on Definition Detection. For subtask2, our base model use the contextual token representations to classify the tokens. And we add a Conditional Random Fields (CRF) layer upon the token classifier to further evaluate the performance of incorporating CRF. For subtask3, we use the fine-tuned RoBERTa model in subtask2 to produce token representations and use the features to train a Random Forest classifier to predict whether there is some relation between two tokens. Our systems achieve respectively 0.785, 0.526 and 0.477 F_1-scores on the official test set, which rank the 29th, 22nd and 23rd respectively in the submissions.

Our main contributions are listed as follows:

- We explore the performance of the pre-trained RoBERTa model for Definition Extraction task.
- We give an in-depth evaluation of different traditional machine learning techniques on subtask1, and demonstrate the powerfulness of RoBERTa.
- We test the influence of introducing CRF in subtask2 through an ablation study.
- We propose to use feature learned by RoBERTa in subtask2 to train a Random Forest based model for Relation Extraction in subtask3. This prove that the representations learnt by RoBERTa in subtask2 are meaningful and useful and thus can be applied for token relation extraction directly without further mapping.

2 Related Work

2.1 Definition Extraction

Definition extraction methods in the research literature fall into one of the following categories. The earlier research on the subject applies rules-based approaches, such as (Klavans and Muresan, 2001) whose system extracts definitions from medical publications. Similar approaches can also be found in Cui et al. (2004, 2005) as well as in Westerhout and Monachesi (2007). These methods find definitions by locating words such as is called, means or is, and using grammatical rules on top of that. They generally suffer from low recall, but this issue has been addressed by features-based methods. Fahmi and Bouma (2006) have trained a sentence classifier with features based on bag-of-words and n-grams, position of the sentences in the text and syntactic information. Westerhout (2009) expands the set of features with additional linguistic and structural information, and uses a hybrid approach between a rules-based system and a machine learning classifier.

The above approaches do not generalize well to new domains, as rules and handcrafted features are often relevant to specific tasks and time-consuming to create, hence the development of Machine Learning and Deep Learning models. Li et al. (2016) use a Long Short-Term Memory neural (LSTM) network classifier, where features are automatically created from raw input sentences and part-of-speech sequences. Contextual words embeddings depend on the adjacent text, and are pretrained in an unsupervised manner to automatically learn semantic concepts before being used on different downstream tasks (Akbik et al., 2018). Lin and Lu (2018) apply transfer learning to their downstream token classification problem, which consists in fine-tuning the context string embeddings to a particular task without having to learn them from scratch. Transformer-based architectures have been recently used with transfer learning: Alt et al. (2019) solve a relation extraction problem with GPT (Radford et al., 2018) embeddings, and Veyseh et al. (2020) use BERT (Devlin et al., 2019) embeddings for a definition extraction task.

2.2 DeftEval

Participants in DeftEval 2020 used a varous array of methods for the three subtasks. For subtask1 and subtask2, many participants used pre-trained language models (*e.g.*, BERT, RoBERTa, and XLNet) (Xie et al., 2020; Avram et al., 2020; Singh et al., 2020; Jeawak et al., 2020; Caspani et al., 2020). For subtask3, Caspani et al. (2020) used a Random Forest model to extract the relations between definitions. The

participants are also interested in Data Augmentation (Caspani et al., 2020) and Jointly Training (Xie et al., 2020; Avram et al., 2020) for subtask1 and subtask2.

3 System Overview

3.1 RoBERTa Representation

Robustly optimized BERT pretraining approach (RoBERTa) (Liu et al., 2019) is an optimized pre-trained bi-directional representation model based on the work of BERT. RoBERTa uses the MLM strategy of BERT, but removes the NSP objective. Moreover, the model was trained with a much larger batch size and learning rate, on a much larger dataset and showed that the training procedure can significantly improve the performance of BERT on a variety of NLP tasks.

RoBERTa has reached the top position on the GLUE leaderboard, which demonstrate its capability of learning meaning contextual representations. Thus we use RoBERTa to generate sentence and token-level representations for downstream tasks. The procedure is formulated as the following formula:

$$h_0, h_1, h_2, \cdots, h_{n+1} = \text{RoBERTa}([CLS], s_1, s_2, \cdots, s_n, [SEP]) \tag{1}$$

where $s_1, s_2, \cdots, s_n$ is the subtokens generated by RoBERTa tokenizer using Byte-Pair-Encoding (BPE) and h is the output of the last transformer layer of RoBERTa. We use h_0 as the representation of the whole sentence and h_i for the token representation of subtoken s_i.

3.2 Subtask1: Sentence Classification

RoBERTa classifier. We represent the sentence embedding with h_0 from RoBERTa and feed it into a one layer feed-forward neural network to produce the prediction:

$$(p_0, p_1)^T = \text{Softmax}(Wh_0 + b) \tag{2}$$

where $p_1 = 1 - p_0$ is the probability of the sentence being classified as positive (*i.e.*, it contains definition).

Other baseline models. We further apply several traditional machine learning methods including Naive Bayesian, K-Nearest-Neighbors, CART, Logistic Regression, Support Vector Machine and another deep model - Long-Short-Term-Memory (LSTM) on subtask1. For traditional methods, we compute TF-IDF as the feature representation of the sentences. For LSTM, we use the average pooling of each time-step's output to produce the sentence representation, and a feed-forward layer is used for classification.

3.3 Subtask2: Sequence Labeling

Simple classification. In this implementation, we simply use a feed-forward layer to produce the probability of each token being in different classes which is similar to the method in subtask1:

$$p^{(i)} = \text{Softmax}(W' h_i + b') \tag{3}$$

However, this model does not take the label dependency of different tokens into consideration. For example, a token can be labeled as "I-Definition" only if the previous token is labeled with "B-Definition" or "I-Definition". And the simple method does not model the joint probability. Thus we augment this method with CRF further more.

Classification with CRF. To model the probability distribution better, we add an additional CRF layer upon the base classification model mentioned above. The details of CRF layer is described in details as follows. For a sequence x of input words and another sequence y of output tags, CRF works by constructing a conditional probability distribution in the following manner:

$$p(y|x; W, b) \propto \exp(\sum_{i=1}^{n} W^T_{y_{i-1}, y_i} x_i + b_{y_{i-1}, y_i}) \tag{4}$$

where the parameters W_{y_{i-1}, y_i} and b_{y_{i-1}, y_i} are called the weight matrix the bias, respectively.

To estimate the parameters W and b, we perform a maximization of the log-likelihood function:

$$L(W, b) = \sum_{j=1}^{N} \log(y^{(j)}|x^{(j)}; W, b) \tag{5}$$

Once the CRF is trained, we use the Viterbi algorithm to find the most probable sequence among all possible tag sequences.

3.4 Subtask3: Relation Extraction

In subtask3, we are given the token classes and required to predict the relation between tokens. Each token is related to one different token at most. We solve this problem by predicting the relation of each token pair. In other word, assume there are n tokens in the test set. We enumerate all the $n(n-1)/2$ token pairs and predict the relation of each pair. We then describe our model to learn the relation between token pairs.

We train a **Random Forest** model to predict the relations between token pairs. The following **features** are extracted as model's input:

- The contextual representation of the two input tokens, respectively. We compute the 768-dim token embedding from the fine-tuned RoBERTa model in subtask2.
- The type (*e.g.*, "Definition", "Term") of the two input tokens. We encode the type into a ont-hot vector.
- Whether the two input tokens are from the same sentence.

If one token is predicted to be related to multiple tokens, we choose the target token that appears at the closest position in the original corpus as output.

3.5 Other Tricks

To further investigate the performance of our systems, we try to incorporate the following two tricks in our models: data augmentation and jointly training.

Data augmentation. In subtask2, we find that the numbers of label in different classes differ greatly, and our model performs poorly in the classes with low sample quantity (*e.g.*, "Qualifier", "Referential-Term"). We try to increase the training data of these classes by data augmentation. As an early step trial, we simply duplicate the training examples containing tokens of these classes.

Jointly training. We notice that subtask1 and subtask2 are strongly related. For example, if a sentence is classified as negative (*i.e.*, not containing definition) in the subtask1, each token in this sentence should be labeled as "O" in subtask2. Thus we try to jointly train our model on the two subtasks.

However, there may be some defects in our implementation, the two tricks do not bring significant performance improvement to our systems in the experiments.

4 Experimental Settings & Results

Settings. We use the RoBERTa-base model in our experiments, which shares the same setting with BERT-base. The parameter size of RoBERTa-base is about 110M. We set the block size (maximum input subtoken length) to 512. The batch size at training time is set to 32. We use the AdamW optimizer with learning rate 5e-5 for optimization. In subtask1 and subtask2, we train our models for 8 epochs and choose the one checkpoint with best performance on the validation set. For traditional ML methods in subtask1, we compute TF-IDF on the whole vocabulary of training set. In subtask3, we define the tree number of Random Forest as 100.

Results. We list the experiment results in Table 1, Table 2, and Table 3.

Model	NB	KNN	CART	LR	SVM	LSTM	RoBERTa
Precision	0.71	0.71	0.57	0.67	0.77	0.71	**0.76**
Recall	0.19	0.02	0.47	0.32	0.40	0.67	**0.82**
F_1-score	0.30	0.03	0.52	0.44	0.53	0.69	**0.79**

Table 1: Results of Subtask1.

Model	F_1-score
Base Model	0.511
+ CRF	**0.526**

Table 2: Results of Subtask2.

Model	F_1-score
Random Forest	0.477

Table 3: Results of Subtask3.

5 Conclusion & Future Work

We propose to utilize RoBERTa to solve the Definition Extraction task. As an early step trial, our model produce comparable results with the published submissions.

Additionally, to our surprise, the data augmentation and jointly training tricks do not significantly improve the performance of our system. We leave the study of them to the future work. The result of our system in subtask3 is not satisfactory and the time complexity is $O(n^2)$ with respect to the labeled token numbers which means our model is inefficient in subtask3. We will solve these problems in the future work.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 1638–1649. Association for Computational Linguistics.

Christoph Alt, Marc Hübner, and Leonhard Hennig. 2019. Improving relation extraction by pre-trained language representations. *arXiv preprint arXiv:1906.03088*.

Andrei-Marius Avram, Dumitru-Clementin Cercel, and Costin-Gabriel Chiru. 2020. UPB at semeval-2020 task 6: Pretrained language models for definition extraction. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (online), December 12-13, 2020*, pages 737–745. International Committee for Computational Linguistics.

Fabien Caspani, Pirashanth Ratnamogan, Mathis Linger, and Mhamed Hajaiej. 2020. ACNLP at semeval-2020 task 6: A supervised approach for definition extraction. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (online), December 12-13, 2020*, pages 479–486. International Committee for Computational Linguistics.

Hang Cui, Min-Yen Kan, and Tat-Seng Chua. 2004. Unsupervised learning of soft patterns for generating definitions from online news. In *Proceedings of the 13th international conference on World Wide Web, WWW 2004, New York, NY, USA, May 17-20, 2004*, pages 90–99. ACM.

Hang Cui, Min-Yen Kan, and Tat-Seng Chua. 2005. Generic soft pattern models for definitional question answering. In *SIGIR 2005: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, August 15-19, 2005*, pages 384–391. ACM.

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.

Ismail Fahmi and Gosse Bouma. 2006. Learning to identify definitions using syntactic features. In *Proceedings of the Workshop on Learning Structured Information in Natural Language Applications@EACL 2006, Trento, Italy, April 3, 2006*. Association for Computational Linguistics.

Shelan S. Jeawak, Luis Espinosa Anke, and Steven Schockaert. 2020. Cardiff university at semeval-2020 task 6: Fine-tuning BERT for domain-specific definition classification. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (online), December 12-13, 2020*, pages 361–366. International Committee for Computational Linguistics.

Judith L. Klavans and Smaranda Muresan. 2001. Evaluation of the DEFINDER system for fully automatic glossary construction. In *AMIA 2001, American Medical Informatics Association Annual Symposium, Washington, DC, USA, November 3-7, 2001*. AMIA.

Siliang Li, Bin Xu, and Tong Lee Chung. 2016. Definition extraction with LSTM recurrent neural networks. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data - 15th China National Conference, CCL 2016, and 4th International Symposium, NLP-NABD 2016, Yantai, China, October 15-16, 2016, Proceedings*, volume 10035 of *Lecture Notes in Computer Science*, pages 177–189.

Bill Yuchen Lin and Wei Lu. 2018. Neural adaptation layers for cross-domain named entity recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2012–2022. 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.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. .

Aadarsh Singh, Priyanshu Kumar, and Aman Sinha. 2020. DSC IIT-ISM at semeval-2020 task 6: Boosting BERT with dependencies for definition extraction. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (online), December 12-13, 2020*, pages 710–716. International Committee for Computational Linguistics.

Sasha Spala, Nicholas A. Miller, Franck Dernoncourt, and Carl Dockhorn. 2020. Semeval-2020 task 6: Definition extraction from free text with the DEFT corpus. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (online), December 12-13, 2020*, pages 336–345. International Committee for Computational Linguistics.

Amir Veyseh, Franck Dernoncourt, Dejing Dou, and Thien Nguyen. 2020. A joint model for definition extraction with syntactic connection and semantic consistency. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9098–9105.

Eline Westerhout. 2009. Definition extraction using linguistic and structural features. In *Proceedings of the 1st Workshop on Definition Extraction*, pages 61–67.

Eline Westerhout and Paola Monachesi. 2007. Extraction of dutch definitory contexts for elearning purposes. *LOT Occasional Series*, 7:219–234.

Shu-Yi Xie, Jian Ma, Haiqin Yang, Lian-Xin Jiang, Yang Mo, and Jian-Ping Shen. 2020. UNIXLONG at semeval-2020 task 6: A joint model for definition extraction. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation, SemEval@COLING 2020, Barcelona (online), December 12-13, 2020*, pages 730–736. International Committee for Computational Linguistics.