A Bert Based Relation Classfication Network for Inter-Personal Relationship Extraction

Cheng Peng

Data Science, GridSum Tech., Shenzhen, China
pengcheng01@gridsum.com

Abstract. Information extraction is an important task in natural language processing and knowledge graph. In this paper, we present our solution on CCKS2019 shared task 3 IPRE(Inter-Personal Relationship Extraction). In the competition data, one sample is composed of a pair of human-entities and a sentence list containing the entities and their relation. Our method can catches the relation hints in raw sentence list with a BERT encoder, then followed by an entity information extraction module which is targeted to build a feature tensor, this tensor can help to calculate the relation categories between two entities with the BERT encoder outputs. In the IPRE competition, using this method, we finally achieve the first place in the sent track with F1-score 0.54279 and the second in the bag track with F1-score 0.62162.

Keywords: Relation Extraction · Information Extraction · Natural Language Processing.

1 Introduction

Relation extraction is a sub problem of information extraction, it's target is to extract relation between a pair of given entities from a list of sentences. In the application of CCKS-IPRE, the pair of entities is restricted to human name, so the target is to find what the relation between character A and character B. It is important to many Articial Intelligence(AI) applications, such as Information Retrieval(IR), Intelligent Question and Answering(QA), and Intelligence Chat-bots(IC). In the CCKS 2019 IPRE competition[1], there are over 402k data which is annotated by distant supervising, then corrected on dev and test part by human. The dataset contains 35 relations including a 'NA' relation. The key challenges of this task are summarized as follows:

Data Noise: Sometimes the relation tag is incorrect in train data because there is no human correct work after annotated with distant supervising. We try to split the incorrect part out, by using our model to test on train data, but the result is too complicated to analysis.

Categories Imbalance: In every data set, 'NA' is the largest part. In train and dev data, we observe that 87.85% are 'NA' samples in sent track data and 92.5% in bag track data, other relation categories have also shown this obvious imbalance.

Entity Position Uncertainty: For example, we can find that in 35.36% bag track train data, one of the two entities repeats more than once in one sentence, but no exact position information is given. As the same entity of different position can contain different semantics, elaborate enumerate or random choose will induce errors into the model.

Extra Long Text: In bag track data, some bag have only one short sentence, but some others have more than one hundred. If we set a max sequence length parameter, we can't confirm which one contains the relation information, some bag may become a noise sample.

To solve the aforementioned challenges, we seleted to train larger amount of models and enlarge the difference of their outputs. We mainly use the following methods: more different model structures, over-sample and randomly under-sample, choosing the sequence order and entity position randomly, at last we ensemble 15 different models. This paper is organized as follows, section 2 contains the data-set analysis and pre-processing. Section 3 introduces details of our solution, section 4 contains the post-processing part, section 5 presents the experimental results and analysis, section 6 includes the conclusion and future works.

2 Data Pre-processing

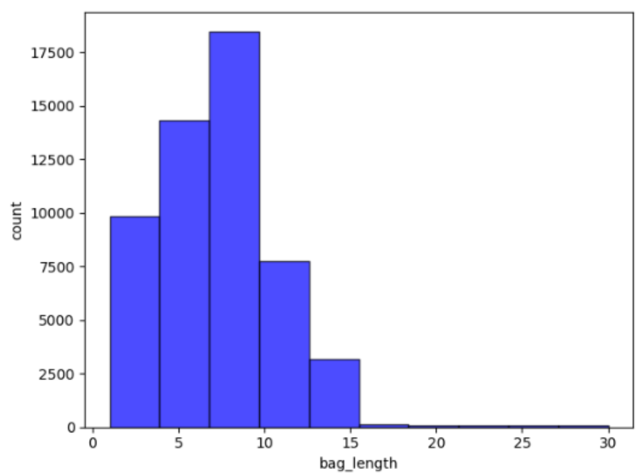

Fig. 1. Histogram of sent number in bag, remove all the samples larger than 30.

Based on our observation that the bag data have better quality than sent data, and there is inner connection between them, one bag sample usually consisted of several sent samples. We considered only design models for bag data, then transformed the best bag track result into sent track result, and used regular expression to help fix the obvious error samples. Our bag track pre-processing method contains the following three parts:

Under-Sample and Over-Sample: the 'NA' relation means there is no relation between two persons, excessived 'NA' samples will harm to our model because model will be trained to focus on how to recognize the 'NA' class. To solve this categories imbalance problem, we over-sampled all classes less than 200 samples to 200. And in every epoch, each 'NA' sample have 90% probability to drop out of train data set. Using this method, we reduce the damage of the 'NA' class to the model, and also use the noise of a large number of 'NA' samples to improve the robustness of the model.

Data Augmentation 1: our primary goal is to implement a classification model for bag data, but there are a lot of bag data that contain too many sentences, so the length of the merged sentence exceeds the range that the model can bear. We need to reduce the length of the merged sentence. However, directly truncating characters beyond the length may make the truncated information not contain valid information that expresses the relationship between the two characters. In order to solve this extra long text problem, in every epoch, the sentence order in bag samples is randomly shuffled, and less than 15 sentences are selected and connected into a long sentence, length of this sentence must less than 510, because BERT model[2] can only support 512 length input.

Data Augmentation 2: either of the two characters may appear multiple times in the sentence, sometimes the short sentence is only expressing some irrelevant information, we call it the entity position uncertainty problem. Due to our principle of increasing the difference between models, we have adopted a method of randomly selecting the location of the entity for this situation. This method will also improve the rebustness of the model.

Data Augmentation 3: also in order to improve the robustness of the model and mitigate the effects of noise in data, we have adopted a data enhancement method: we randomly drop the non-entity words or replace them by other random words, and the ratio should be less than 1%.

3 Relation Recognition

3.1 Single Model of Relation Classification

We designed a BERT encoder based model, the reason why we chose BERT is that we have validated the advantage of BERT in many project or competition like lic2019 Information Extraction. We have been optimizing and improving the structure of the model during the competition, including using ERNIE[3] or BERT-wwm[4] to replace the origin BERT. Figure 2 below is the structure of our final model.

Our model can be divided into three parts:

1) BERT encoder: The encoder encoded the input word, here is Chinese character, to word embedding tensor. The choice of BERT is because we can alway have tremendous improve by using of the bert replacement Bi-LSTM in the previous competitions such as the lic2019 information extraction competition. Therefore, BERT is used as the encoder in this competition. In our experiments, the BERT-wwm encoder obtain a highest single model score: 0.55.

2) Info-extractor: This part of model stucture extracts the embedding of two entities and non-entity words between them depends on the excellent word embedding output by BERT. These three embedding tensors may contain information of subject entity,

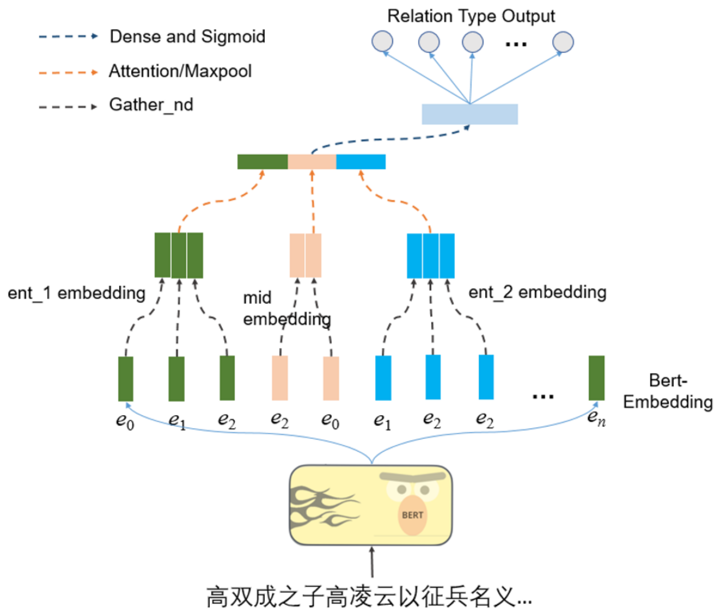

Fig. 2. The relation classification network structure. An encoder is used to encode source sentence embedding, then followed by the info-extractor, at the end is as sigmoid classifier.

object entity and inside relation between two entities. At first we train our model without considering extract the non-entity word vector, but after we analysis the output and found that some key infomation is hidden in the non-entity words, we add the non-entity word extract part.

3) Classifier: Use attention or max-pool to pick the most important information of info-extractor output, then use sigmoid to get the multi-classification result.

3.2 Multi-model Ensemble

We trained 15 different models based on the proposed model structure above. In order to enlarge the difference between the models, 1) we changed the encoder from BERT to ERNIE or BERT-wwm, 2)we used different structure, like attention or maxpooling layer, or just remove the tensor extracted from the word-embedding between two entities, added or removed dense layer, etc. Then we select 15 different models, most of them can get online score better than 0.52, and the output-similarity between every two models is lower than 0.80, the similarity calculation uses F1 equation below, N_r is the count of same predict bag of model-a and model-b, N_{model_a} is the count of not-NA bag of model-a, N_{model_b} is the count of not-NA bag of model-b:

$$P = \frac{N_r}{N_{model_a}} \tag{1}$$

$$R = \frac{N_r}{N_{model_b}} \tag{2}$$

$$F1 = \frac{2PR}{P+R} \tag{3}$$

To ensemble these 15 models, we caculate the average value of 15 probabilities and output all categories bigger than 0.5, if the output is NA, select the mode of 15 categories as the ensemble result.

4 Post-processing

Our post-processing has two purposes: 1. Convert the output of the bag model to the output format of Sent Track; 2. Correct the results using the rules for the two output files. Bag track model predicts the relation of a given person entity pair based on a given sentence set at the bag level, while sent track predicts the character entity correspondence from the sent level, the former's bag level containing the latter's sentences. The bag track results predicted by the model are converted into sent track results, and all the sentences in a bag are uniformly marked as the prediction relation label of the bag. However, there is a problem exists in this direct conversion. Not all sentences in a bag reflect the relation of the entity pairs. For example, Figure 3 shows two sentences composing a bag. The first sentence can not find the relation corresponding to the two characters, should be identified as the relationship "NA". Therefore, it is necessary to add a filtering operation during the conversion process to avoid this type of error in the sent level prediction result.

输入:

- 袁汤 袁安 从袁安起，几代位列三公(司徒、司空、太尉)，出过诸如袁汤、袁绍、袁术等历史上著名人物。
- 袁汤 袁安 袁汤（公元67年—153年），字仲河，河南汝阳（今河南商水西南人，名臣袁安之孙，其家族为东汉时期的汝南袁氏。

输出:

袁汤 袁安 人物关系/亲属关系/血亲/自然血亲/祖父母/爷爷 NA

Fig. 3. The first sentence can not find the relation corresponding to the two characters.

The most succinct and efficient way to handle is to determine whether a certain type of relationship exists in a sentence by a specific word. First of all, it is necessary to classify the relation types. The relation of the same class can be filtered in the same way. After statistical analysis, some of the relation between the characters in the training set and the dev set is very small, which can be neglected because the model is difficult to learn from it. Useful information therefore does not need to deal with these very few

Table 1. Types of relations.

category of the relation	relations
kinship/current spouse	husband/wife
kinship/ex-spouse	ex-husband/ex-wife
kinship/blood relatives/grand-relation	grandfather/grandmother/grandson/granddaughter
kinship/blood relatives/parents and kid	father/mother/son/daughter
kinship/blood relative/brothers and sisters	elder-brother/little-brother/elder-sister/little-sister
kinship/blood relative/others	mother's brother/uncle
friendship	friend
social/relation	like/lover
teacher and student	teacher/student

relations, and the remaining relations to be processed can be divided into 9 categories according to Table 1.

Then, the training set is analyzed on the category of the relation, and the key words that can reflect the category are extracted as the Table 2.

Table 2. Key Words of relations.

category of the relation	key words
kinship/current spouse	marrige/husband/wife
kinship/ex-spouse	divorce/ex-husband/ex-wife
kinship/blood relatives/grand-relation	grandfather/grandmother/grandson
kinship/blood relatives/parents and kid	father/mother/son/daughter/born
kinship/blood relative/brothers and sisters	brother/sister
kinship/blood relative/others	uncle
friendship	friend/pal
social/relation	like/lover/boyfriend/girlfriend
teacher and student	teacher/student/apprentice/disciple

Finally, we can used the key words of the relations to modify the sent track result which is converted from result of bag track, and the showup of key words indicates that the sentence may contains the relation that bag track model predict. Otherwise, we have no way to judge what kind of relationship is contained in the sentence text. And this method can also used to modify the prediction result of the bag track.

5 Experimental Results

Our experiment is first conducted with Bert encoder, and only the first char embedding of two entities is used(model_v1). This baseline model has achieved a good result at the beginning of the competition. Then we try to use all char embedding of the two entities which leads to a signicant improvement(model_v2_BERT). After that, we try to change the bert parameters to ERNIE or BERT-wwm, ERNIE will lead to a minor decline(model_v2_ERNIE) but BERT-wwm can help improve a lot(model_v2_BERT_wwm). At last, we add the embedding between two entities, thus we achive a 0.55 online score of single model(model_v3_BERT_wwm), and then our ensemble model can get 0.5995(ensemble_v1). With the post-processing, bag track can achive 0.6077(ensemble_v2) and sent track 0.5408(sent). Table 3. presents the detail. On list B, we get the F1-score of 0.54279 on sent track, and 0.62162 on bag track, that make us win the first place on the sent track and the second place on bag track.

Table 3. Different model structure score offline/online.

model	P_offline	R_offline	F1_offline	F1_online
model_v1	0.4532	0.4234	0.4378	0.4792
model_v2_BERT	0.4765	0.4928	0.4845	0.5169
model_v2_ERNIE	0.4819	0.5330	0.5150	0.5139
model_v2_BERT_wwm	0.5409	0.5301	0.5354	0.5365
model_v3_BERT_wwm	0.4948	0.5444	0.5184	0.5545
ensemble_v1	0.5755	0.5788	0.5771	0.5995
ensemble_v2	_	_	_	0.6077
sent	_	_	_	0.5408

6 Conclution and Future Works

We analyze the dataset of CCKS 2019 IPRE task, and propose the BERT-encoder based relation classification method. This method successfully solves the problems in Inter-Personal Relationship Extraction. Our experiments reveal the importance of better encoder in NLP tasks and data augmentation in noisy data. During the competition, we have only tested the origin BERT encoder, ERNIE encoder, and BERT-wwm encoder in our experiments with limited time. It's meaningful to have other more explorations on different encoder and decoder structures, and data augmentation strategy

References

1. Haitao Wang, Zhengqiu He, Jin Ma, Wenliang Chen, Min Zhang. 2019. IPRE: a Dataset for Inter-Personal Relationship Extraction. https://arxiv.org/abs/1907.12801
2. Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.
3. Zhang Z, Han X, Liu Z, et al. ERNIE: Enhanced Language Representation with Informative Entities[J]. arXiv preprint arXiv:1905.07129, 2019.
4. Cui Y, Che W, Liu T, et al. Pre-Training with Whole Word Masking for Chinese BERT[J]. arXiv preprint arXiv:1906.08101, 2019.