FMPK Results for CCKS 2019 Task 3 : Inter-Personal Relationship Extraction

Huan Liu, Peng Wang, Zhe Pan, Ruilong Cui, Jiangheng Wu and Zhongkai Xu

School of Computer Science and Engineering, Southeast University, China
{ms, pwang} @ seu.edu.cn

Abstract. Distant supervised relation extraction is widely used in inter-personal relation extraction tasks. However, noisy data will be introduced by distant supervision and models will suffer from the wrong labelling problem. In this paper, we propose a method named *Fusion Model with Prior Knowledge*(FMPK) using model fusion and heuristic rules to augment raw texts and filter noisy data. Experimental results on CCKS 2019 Inter-Personal Relationship Extraction sentence-track task show that our model performs well and achieves 0.41003 F1 value, which is fifth place in all teams.

Keywords: Relationship extraction · Multiple-model fusion · Heuristic rule .

1 Introduction

1.1 Task Description

The goal of CCKS 2019 Inter-Personal Relationship Extraction (IPRE) task [1] is to match the personal relationship against Chinese real-world data[1] . In this task, models should determine which of personal relations two entities are in real-world texts. And there are 35 kinds of inter-personal relations in total including husband, ex-husband, father, uncle, brother, teacher and so on.

In this paper, we focus on sentence-level inter-personal relation extraction task, which means the relation of same pairs of people would be different in different sentences. Take *Yichen He - Mosheng Zhao* as example, their relation will change from lover into spouse in different sentences.

1.2 Challenges

In recent years, with the rapid development of neural network, models have made giant progress in natural language processing. Although Relation Extraction (RE) has advanced considerably, there are many challenges remained in this IPRE task.

[1] `https://biendata.com/competition/ccks_2019_ipre/`

1) First, restricted by real-world texts, there exists severe imbalance in amounts of different relation labels. Furthermore, Not Available (NA) accounts for about 90% of the whole labelled texts, which means that most of people in given labelled sentences do not have inter-personal relation at all.
2) Second, it is no wonder that distant supervision will introduce noisy data into train dataset [2]. And in this task, according to task description, the train dataset is processed by distant supervision while validation and test dataset is labelled by human. Different labelling procedures of train, validation and test dataset lead to different distribution of labels. And it brings lots of challenges in training.
3) Finally, inter-personal relations in some sentences can not be extracted directly and it should be inferred and reasoned by words. Take *spouse* as example, *spouse* relation of two people does not occur in a sentence literally but can be inferred by description of their children's *father* and *mother*. So basic logic reasoning for inter-personal relation is essential faced with this kind of problems.

1.3 Contributions

FMPK is designed to use different methods and strategies to handle with challenges mentioned, which has been proved significant in experiments.

1) For noisy data problems, we use prior knowledge and heuristic rules to filter part of noisy data.
2) Data augments by translation are exploited to enrich minority. In addition, we design a fusion model and weighted softmax to balance different kinds of labels in predicting part.
3) As for reasoning, logic prior knowledge and heuristic rules are designed to do with simple inter-personal inferences.

2 Methods

2.1 Framework

We can divide our method into two parts: data filtering and model fusion. First, we introduce prior knowledge to filter out noisy data and augment data in minority. Second, cleaned train data are fed into 3 models for training and fine-tuning. Finally, the prediction of our fused model would be checked by the same prior knowledge to promote precision of final results. Details about the whole framework are listed as following.

2.2 Prior Knowledge

Due to noisy data introduced by distant supervision, it is necessary to filter out noisy data ahead of training models. And we use prior knowledge and design heuristic rules to make train dataset more reliable. Actually, we filter about half of Not Available(NA) relation in train dataset through ways as following.

Relation Trigger Relation trigger is a word in a sentence when sufficient information has been mentioned to identify a semantic relation. And relation triggers can be verbs, nouns and idioms. Other symbols, such as comma and period, can also be regarded as triggers (the relation type place-of-death can be signified till the comma position). In this paper, we regard all of these mentioned above as trigger words.

Trigger words can be selected by key words in different relations and words similarity. We firstly divide train dataset into different bucks based on labels except Not Available(NA). Then we suppose that if a word w_x is a key word of relation r_i, it occurs in the buck of relation r_i more and in other bucks less. As is mentioned above, we can get top-k candidate key words of each relation through this TFIDF-like method [3]. Meanwhile, we calculate word similarity between relation name and top-k candidate key words to figure out trigger words which are closer to relation names statistically and semantically. As for words similarity calculation, we use word embeddings trained by word2vec [4].

More specifically, we can define words value for trigger generation as following:

$$val_{word} = w_1 \times tf_{i,j} \times log(\frac{N}{df_i + 1}) + w_2 \times \frac{emb_i \cdot emb_{name}}{||emb_i|| \times ||emb_{name}||} \tag{1}$$

Parameters w_1 and w_2 are the weights of each method and they are hyper parameters. According to challenges mentioned in Section 1.2, we loosen the threshold to get more triggers for reasoning. So trigger words set of one relation may have intersection with the others at last. And we suppose that when no trigger exists in a sentence, two entities are likely NA relation (No relation).

Heuristic Rule For inter-personal relation extraction, there are lots of tricks we can use to restrict candidate predictions into narrow limits. For example, father and son share the same family name in Chinese tradition. And family names can also be used in determination of brothers and sisters relation. Furthermore, relation husband and wife can be distinguished by gender information. As is hard to judge one's gender only by name, we sort a list of names out to support gender judgement which does good to relation extraction with gender information. Detailed rules are listed as following.

1) Family Name. Entities have relations named *father*, *son*, *brother*, *sister*, *grandfather*, *grandson*, *aunt(father's sister)* and *uncle(father's brother)* share the same family name.
2) Gender. Objects must be males in relations of *father*, *son*, *uncle* and so on. Females should be *mother*, *daughter*, *aunt* in a similar way. In spite of this, it is hard to distinguish one's gender from the name. And we do not filter out data based on these rules strictly.
3) Reasoning. Take *spouse* as an example again, when *father* and *mother* occur in the same sentence we would add *husband* and *wife* to candidate relations

in addition. Similarly, *son-in-low* is added when *daughter* and *husband* both exit in the same sentence. And *aunt(father's sister)* is added when *father* and *sister* both exit in the same sentence.

These heuristic rules could help eliminate part of noisy data that distant supervision brings in, which makes train dataset more reliable and promotes ability of our model. Meanwhile, rules can benefit precision when used in prediction part.

2.3 Data Augment

The imbalance amount of labels remains after filtered by prior knowledge. Because we only delete lots of NA relation sentences and some parts of others violating prior knowledge. It is necessary to augment some kinds of sentences with smaller amount of labels.

Algorithm 1 Iterative Translation

```
Input: Raw sentence, augmentTimes
Output: Set of augmented sentence(s)
  augmentSet ← ∅
  sentence ← Raw sentence
  for i ← 0 to augmentTimes do
      storage ← ∅
      for word in sentence do
          if word is name then
              replace word with symbol $
              storage ← word
          end if
      end for
      Translate(sentence, Japanese)
      Translate(sentence, Chinese)
      for word in sentence do
          if word is $ then
              name ← storage
              replace $ with name
          end if
      end for
      Similarity = calculateSimilarity(sentence, Raw sentence)
      if Similarity < θ1 or Similarity > θ2 then
          break
      else
          augmentSet ← sentence
      end if
  end for
```

Translation We use Baidu Translator to translate Chinese into Japanese and translate it back into Chinese to enrich train dataset sentences[2]. Nevertheless, names will change a lot during translation and it will influence sentence structure. As for name translation, we replace name with special symbols in translation and replace it back after translation.

Algorithm 1 shows the process of Iterative Translation which is the way we use to enrich train dataset. And two hyper parameters θ_1 and θ_2 are set to ensure that the translated sentence is slightly different from raw sentence. In brief, θ_1 can be regarded as a lower bound of similarity to ensure that translated one can not differ from the raw sentence too much. In comparison, θ_2 is used to stop the process early as it is useless to generate almost the same sentence many times.

Synonym Replacement Although we can replace words with their synonyms, experiments show that simple replacements have little improvement on the experimental results. So we do not try more ways of replacement in this task.

2.4 Model Fusion

Model To promote the ability of models, we fuse 3 kinds of models into FMPK to vote final prediction. And it helps a lot when F1 values of 3 models are similar and prediction distributions are much different.

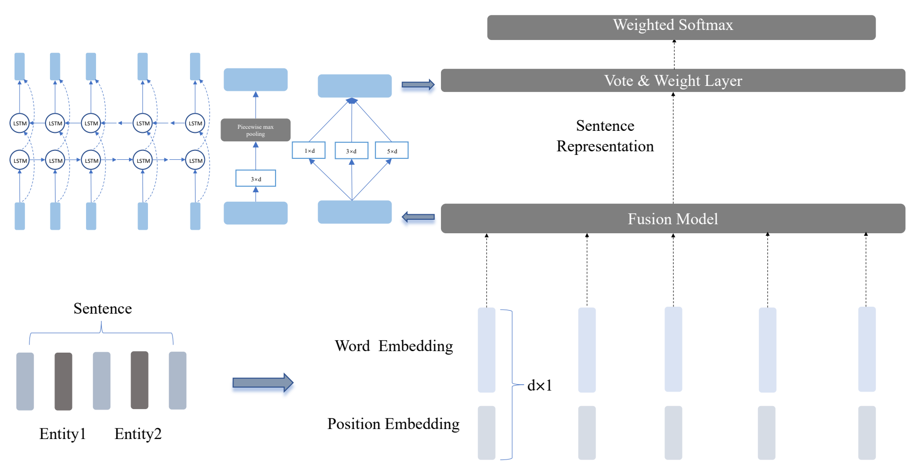

Fig. 1. Fusion Model

[2] fanyi.baidu.com

PCNN Piecewise Convolutional Neural Networks. Single max pooling is not sufficient to capture the structural information between two entities. In relation extraction, an input sentence can be divided into three segments based on the two selected entities. Piecewise max pooling is introduced to get the maximum value in each segment instead of a single maximum value. [5]

Multi-size CNN multi-sized window kernels CNN. It is easy to regard filter as representing some hidden class of the augmented n-grams and the scores as measuring the possibility the augmented n-gram at position belongs to the corresponding labels. [6]

BiLSTM In bidirectional LSTM, for a given sequence, the network computes both a left and a right representations. The final representation is created by concatenating them. [7]

As is shown in Fig 1, 3 models share the same input of concatenation of word embeddings and position embeddings. And then we add weighted layer to learn different parameters for 3 outputs.

Weighted Softmax As for imbalance of label, we use weight Softmax to balance training and predicting probability [8]. Weighted Softmax can be explained as following:

$$S_i = \frac{w_i e^i}{\Sigma_j w_j e^j} \tag{2}$$

2.5 Strengths and Weakness

Strengths The introduction of prior knowledge can reduce misread of noisy data and promote precision of prediction. And fusion models and weighted parameters can balance distributions of labels.

Weakness It is necessary to set threshold and fine-tune over times to generate proper trigger words. In spite of this, intersection of relation trigger sets is inevitable which could cause misleading in training and reasoning. And in the fusion part, hyper parameters are too many to be adjusted meticulously.

3 Experiment

We can only test our FMPK and methods with validation dataset. Because final results did not offer detailed information about our submitted models. As majority of labels are NA, evaluation results filter out NA relation results in final tests by competition sponsors.

As is shown in Table 1, the introduction of prior knowledge promotes the quality of train dataset and ability of model considerably. In general, imbalance of

Table 1. Comparison of different models on CCKS2019 IPRE task

Model name	Method name	Precision	Recall	F1
Baseline	no prior knowledge	–	–	0.220
Baseline	prior knowledge	–	–	0.243
Multi-CNN	no prior knowledge	0.265	0.436	0.330
Multi-CNN	prior knowledge	0.306	0.402	0.354
PCNN	no prior knowledge	0.259	0.447	0.328
PCNN	prior knowledge	0.305	0.425	0.355
BiLSTM	no prior knowledge	0.272	0.411	0.327
BiLSTM	prior knowledge	0.310	0.398	0.349
FMPK	no prior knowledge	0.297	0.463	0.362
FMPK	prior knowledge	0.334	0.441	0.380

labels and noisy data do harm to the whole task and model, which are shown that F1 scores are in a low level in average. And the method of triggers and heuristic rules can release noisy data problems. Meanwhile, fusion of models make great progress when distribution of 3 predictions are much different. Furthermore, heuristic rules benefit promotion of precision in results and slight decrease in recall, which accounts for the raise of F1 score. And it can also be proved by baseline tests.

It shows that FMPK performs better when distributions of outputs are much different from each other. Though fusion model outperforms the others, it can not do with relations with inferences and reasoning well.

4 Conclusion and Future Work

This paper proposes a heuristic method to filter out noisy data introduced by distant supervision and a fusion model FMPK to release the problems of noisy data and imbalance label distributions. Experiments show that it works well but our methods can not solve noisy data problems entirely.

Furthermore, reasoning problems remain and our simple heuristic rules and fusion models are not able to handle these well. We hope that reasoning ability can be boosted when syntax parsing is exploited. And we are going to use sentence structure and syntax analysis to help with reasoning in inter-personal relation extraction in the future.

5 Acknowledgement

The work is supported by National Key R&D Program of China (2018YFD1100302, 2018YFB1003901), the National Natural Science Foundation of China (No.61872177, No.61472077).

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. Mintz, M. , Bills, S. , Snow, R. , & Jurafsky, D. . (2009). Distant supervision for relation extraction without labeled data. ACL 2009, Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP, 2-7 August 2009, Singapore. Association for Computational Linguistics.
3. Ramos, J. (2003, December). Using tf-idf to determine word relevance in document queries. In Proceedings of the first instructional conference on machine learning (Vol. 242, pp. 133-142).
4. Mikolov, T., Chen, K., Corrado, G., & Dean, J. Efficient estimation of word representations in vector space. In Proceedings of the International Conference on Learning Representations. (2013).
5. Zeng, D., Liu, K., Chen, Y., & Zhao, J. (2015, September). Distant supervision for relation extraction via piecewise convolutional neural networks. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (pp. 1753-1762).
6. Nguyen, T. H., & Grishman, R. (2015, June). Relation extraction: Perspective from convolutional neural networks. In Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing (pp. 39-48).
7. Zheng, S., Hao, Y., Lu, D., Bao, H., Xu, J., Hao, H., & Xu, B. (2017). Joint entity and relation extraction based on a hybrid neural network. Neurocomputing, 257, 59-66.
8. Abdel-Hamid, O., Mohamed, A. R., Jiang, H., Deng, L., Penn, G., & Yu, D. (2014). Convolutional neural networks for speech recognition. IEEE/ACM Transactions on audio, speech, and language processing, 22(10), 1533-1545.