Ict_spring Belief and Sentiment System at TAC 2017

Yihan Ni [*], **Yue Zhao** [*], **Bingbing Xu** [*], **Hui Du**
University of Chinese Academy of Sciences
CAS Key Laboratory of Network Data Science and Technology,
Institute of Computing Technology, Chinese Academy of Sciences,
Beijing, China
{niyihan, zhaoyue, xubingbing, duhui}@ software.ict.ac.cn

Abstract

We present the systems of the Ict_spring team for the 2017 TAC KBP BeSt Evaluation.

1 Introduction

The 2017 TAC KBP BeSt Evaluation aims to predict beliefs and sentiments of targets. We use the following data sets from the LDC.

- LDC2016E27_DEFT_English_Belief _and_Sentiment_Annotation_V2
- LDC2016E61_DEFT_Chinese_Belief _and_Sentiment_Annotation
- LDC2016E62_DEFT_Spanish_Belief _and_Sentiment_Annotation
- LDC2016E114_TAC_KBP_2016_Belief _and_Sentiment_Evaluation_Gold _Standard_Annotation

We submitted three different systems which show different performance. This notebook has divided into three parts. Firstly, we will introduce three different systems according to their feature engineering and models. The second part is the most important part which includes feature extraction, models and the process of source finding. And then the results of our team has illustrated. Lastly, we concluded our work.

2 Types of Systems Submitted

The ICT_Spring team submitted three systems, each of them contains English Chinese and Spanish.

[*] These authors contributed equally to this work.

2.1 System 1

The first system includes gold ere and predicted ere. For the gold ere, sentiment classification applies regression method, the difference is that the discussion forum data take general features, and the newswire data using doc2vec features. In belief classification, we take the doc2vec features to train a two-layer neural networks with 20 iterations. For the predicted ere, we use the general features to take regression prediction in sentiment classification and neural networks in belief classification.

2.2 System 2

The second system only submits the gold ere part, in which sentiment classification use dec2vec features to train a two-layer neural network, but the English discussion forum data uses general features.

2.3 System 3

The third system is submitted as a baseline, which sets all sentiment as neg and all belief as cb.

3 Approaches

3.1 Feature Extraction

In aspect of features extraction, three types of features are extracted. And we determine which type to use according to the performance in different situations. We introduce them as follows.

3.1.1 General Features

These features includes tfidf, the number of sentiment words, the quantified sentiment values of words, the type of entities and so on. They are extracted in the level of target texts and contexts respectively.

The **context** of a target includes the sentence which contains the target and 2 sentences before this sentence as well as 2 after this sentence, totally 5 sentences. Of course we can change the number of sentences to 3, 7 and so on. But through tests, setting the number to 5 leads to the best performance.

In practice, we determined the best combination of the features mention above by experiments. In addition, these features can be added to the embedding matrix features as well when using CNN as the classification model.

For the features about sentiment, we use the python package pattern.en.sentiment[1]. Lack of sentiment dictionaries for Chinese and Spanish, features about sentiment words are removed for these two languages.

3.1.2 Word Embeddings

Another type of features is based on word embeddings. We use pretrained word embeddings here, combining the word embeddings of context, window text, and target text of each target. For English, what we use is the ready-made Glove word embeddings on Wikipedia. We have tried both the embeddings with 300 dimensions and that with 100 dimensions and it turned out that there is not obvious difference in terms of performance. For Chinese, we have embeddings trained with Chinese blogs by ourselves. We do not have appropriate Spanish corpus, so we did not try this kind of feature on this language.

The way of combination is different for different models. For example, to apply to CNN, we put them together to construct a matrix, and to apply to a two-layer network, we compute the average vector of these word embeddings.

However, the embedding features did not perform good in our experiments on the development set, so they are not used in the final submissions.

3.1.3 Doc2vec

We use the model of doc2vec to generate features as well. The model is trained with official data, using the context of each target as one document. We have external data for English and Chinese, so in these two languages we train the model with official data combined with external data. For English the external data set is imdb[2], and for Chinese we use a weibo data set.

3.2 Models

In the aspect of models, mainly two models are used to extract the results, **regression** and **a fully connected neural network with two layers**. In addition, we do some other attempts such as CNN, SVM, logistic regression and so on.

3.2.1 Regression

To apply the task to regression, we treat the labels as real values, and also predict real values for the samples of test data. After that, we set thresholds to turn these val-

[1] https://www.clips.uantwerpen.be/pages/pattern-en#sentiment

[2] http://ai.stanford.edu/ amaas/data/sentiment/

ues into labels. The thresholds are determined artificially according to repeated experiments.

3.2.2 Fully Connected Neural networks

The fully connected neural networks is of two layers, trained as a multi-classification model via a softmax function. Due to the small scale of training data, we only constructed this model with two layers to reduce parameters.

3.2.3 Other Models

In addition, we do some other attempts such as CNN, SVM, logistic regression and so on. CNN is also trained as a multi-classification model and uses the embedding matrix as features. Classifiers like SVM and LR is used in the form of 1vs1 or 1vsothrs. We also tried to use the sentiment word dictionary to filter the results. And we merged these models by voting. But these models tend to predict the labels of most samples as None, which is not what we desire.

We have tried some other methods proposed in some papers as well, but the performance is not desirable all the same. The two models mentioned above, i.e., regression and the two-layer network, perform best. Therefore, we submit the results of these two models eventually.

3.3 Resampling

The training data are imbalanced in this task. There are too many samples labeled with None in sentiment annotations and too many samples labeled with cb in belief annotations.

For belief classification, the problem can be omitted since predicting all samples as cb is acceptable and even good. However, in the sentiment classification task, most samples are labeled with None, which will be removed when evaluating the performance, and as a result it is meaningless to predict all samples as None. In other words, we need to detect samples with sentiment.

Hence, to highlight the importance the positive and negative samples, we do up resampling in sentiment classification. For instance, to apply to a multi-class classification model, we should make both the number of the positive samples and the negative samples become the same as the number of None samples.

3.4 Sources Finding

Our method to find source is to use the author as its source. Most newswires and some discussion forums do not have author, and so we set the sources to None.

4 Evaluation and Results

The results of the evaluation is showed in Table 1-6. In the three submitted systems we submitted, system3 is treated as the baseline, which for sentiment all samples are set as neg, and for belief cb.

We first look at the sentiment task. In the case of gold ERE, as expected, our System 1 has high performance in terms of precision and F-measure. But in predicted ERE, System 3 performs best on F-measure. In contrast, System 2 has a higher precision at the cost of much lower recall, which leads to the final F-measure lower than System 3. Thus, in terms of F-measure, our best system is, disappointingly, our baseline system, System 3.

We now consider the belief task. For gold ERE, among our three systems, System 3 performance best in F-measure, which mainly caused by higher recall. This is because the actual label of most samples is exactly cb. However, Our System 1 got the highest F-measure among all three languages in predicted ERE, which is attributed to higher precision.

As expected the results in predicted ERE are significantly lower due to the errors introduced by entity, event and relation annotation. Meanwhile, the score in discussion forums is higher than that in newswires in both tasks.

5 Conclusion

In conclusion, feature engineering is a crucial part which can determine the final result. Deep neural network has the ability of feature extraction, but in our experiments, linear regression had better performance than deep neural network. In our opinion, features, parameters and the structure of model all are the reasons. On the other hand, we only used some rules to get source via our experience. In the following work, we can try to use machine learning to find source. Also, every aspect should be considered more carefully.

References

[1] Jeffrey Pennington, Richard Socher, Christopher D. Manning: Glove: Global Vectors for Word Representation. EMNLP 2014: 1532-1543

System	System	Gold ERE			Predicted ERE		
		Prec.	Rec.	F-measure	Prec.	Rec.	F-measure
Sys 1	DF	0.122104116	0.519119351	0.197705207	0.034161491	0.033604888	0.033880903
	NW	0.04612231	0.196793003	0.074730141	0.023952096	0.016477858	0.0195241
Sys 2	DF	0.109589041	0.315179606	0.162630792	—	—	—
	NW	0.04597171	0.145286686	0.069843495	—	—	—
Sys 2	DF	0.091805576	0.708381615	0.162545422	0.030173944	0.057705363	0.03962704
	NW	0.0405366	0.337706511	0.072384523	0.017468944	0.046343975	0.025373555

Table 1: Results for our three English Sentiment systems on KBP Eval Data

System	System	Gold ERE			Predicted ERE		
		Prec.	Rec.	F-measure	Prec.	Rec.	F-measure
Sys 1	DF	0.087306377	0.286468647	0.133826704	0.033685269	0.032891507	0.033283656
	NW	0.014787879	0.181547619	0.027348128	0.007211538	0.010135135	0.008426966
Sys 2	DF	0.059016393	0.213861386	0.092505353	—	—	—
	NW	0.007866936	0.052083333	0.013669205	—	—	—
Sys 2	DF	0.048667387	0.713531353	0.091119821	0.018424809	0.069710358	0.029146141
	NW	0.01365961	0.313988095	0.026180284	0.00389314	0.032657658	0.006956939

Table 2: Results for our three Chinese Sentiment systems on KBP Eval Data

System	System	Gold ERE			Predicted ERE		
		Prec.	Rec.	F-measure	Prec.	Rec.	F-measure
Sys 1	DF	0.128500192	0.350052247	0.187991021	0.043097643	0.019826518	0.027158922
	NW	0.036312305	0.206877427	0.061780538	0.022685185	0.020493517	0.021533729
Sys 2	DF	0.090831191	0.332288401	0.142664872	—	—	—
	NW	0.029677552	0.160288408	0.050082315	—	—	—
Sys 2	DF	0.09703601	0.657958899	0.169128839	0.049319728	0.071871128	0.058497226
	NW	0.034990414	0.283416528	0.062290486	0.021918289	0.052279381	0.030887077

Table 3: Results for our three Spanish Sentiment systems on KBP Eval Data

System	System	Gold ERE			Predicted ERE		
		Prec.	Rec.	F-measure	Prec.	Rec.	F-measure
Sys 1	DF	0.725126848	0.843144799	0.779695191	0.026490066	0.002727583	0.004945904
	NW	0.737846673	0.507249529	0.601194312	0.026680896	0.004172926	0.00721709
Sys 2	DF	0.735429177	0.801077337	0.766850829	—	—	—
	NW	0.738643634	0.502174859	0.597876748	—	—	—
Sys 2	DF	0.725644386	0.852122611	0.783814074	0.022857143	0.002727583	0.004873591
	NW	0.737322835	0.509134406	0.602341438	0.026511135	0.004172926	0.007210845

Table 4: Results for our three English Belief systems on KBP Eval Data

System	System	Gold ERE			Predicted ERE		
		Prec.	Rec.	F-measure	Prec.	Rec.	F-measure
Sys 1	DF	0.808107225	0.836831415	0.822218527	0.007984032	0.001796461	0.002932981
	NW	0.638713667	0.387251594	0.4821662	0.009541397	0.001629606	0.002783764
Sys 2	DF	0.807994758	0.834800271	0.821178821	—	—	—
	NW	0.638703291	0.387101612	0.482046972	—	—	—
Sys 2	DF	0.808107225	0.836831415	0.822218527	0.007432181	0.001796461	0.002893519
	NW	0.638713667	0.387251594	0.4821662	0.009402487	0.001629606	0.002777778

Table 5: Results for our three Chinese Belief systems on KBP Eval Data

System	System	Gold ERE			Predicted ERE		
		Prec.	Rec.	F-measure	Prec.	Rec.	F-measure
Sys 1	DF	0.798959189	0.707408755	0.750401955	0.039215686	0.001747488	0.00334588
	NW	0.649757504	0.454687213	0.534995413	0.035377358	0.003232759	0.005924171
Sys 2	DF	0.813054499	0.622286892	0.704993475	—	—	—
	NW	0.649294671	0.455971381	0.535725833	—	—	—
Sys 2	DF	0.789348371	0.763792894	0.776360387	0.029962547	0.001747488	0.003302374
	NW	0.651126651	0.46119978	0.539948454	0.034883721	0.003232759 7	0.00591716

Table 6: Results for our three Spanish Belief systems on KBP Eval Data

[2] Yoon Kim: Convolutional Neural Networks for Sentence Classification. EMNLP 2014: 1746-1751

[3] OwenRambow et al.: The 2016 TAC KBP BeSt Evaluation. In Proceedings of Text Analysis Conference 2016.

[4] OwenRambow et al.: The Columbia-GWU System at the 2016 TAC KBP BeSt Evaluation. In Proceedings of Text Analysis Conference 2016.

[5] Vlad Niculae et al.: Cornell Belief and Sentiment System at TAC 2016. In Proceedings of Text Analysis Conference 2016.