aiai at FinSBD task: Sentence Boundary Detection in Noisy Texts From Financial Documents Using Deep Attention Model

Ke Tian[1], **Zi Jun Peng**[2]
[1]Office of CTO, OPT, Inc., Japan
[2]School of Computer Science and Technology, Harbin Institute of Technology(Weihai), P.R.China
[1]tianke0711@gmail.com, [2]986320586@qq.com

Abstract

This paper describes how we tackle the FinSBD-2019 shared task in IJCAI-2019. The deep attention model based on word embedding is proposed to detect the sentence boundary in noisy English and French texts extracted from the financial documents. The experiment is shown that the model has good performance for predicting beginning and end index of sentence in the two tasks which the result achieved the score of F1 for BS, ES are 0.88,0.91 respectively in English task, and score of F1 for BS, ES are 0.91, 0.92 respectively in French task.

1 Introduction

The first step of many language tasks, such as POS tagging, discourse parsing, machine translation, etc., is the sentence boundary detection (SBD), which detects the end of the sentence [Nagmani Wanjaria, 2016]. This makes the task of detecting the beginning and ending very important, which helps in processing the written language text. However, detecting the end of the sentence is a complicated task due to the ambiguousness of punctuation and words in the sentence [Gregory Grefenstette, 1994]. For example, punctuation marks like "." and "!" don't always represent the end part of sentence text and have several functions. The "." can be part of a number like 2.34 or an abbreviation of a phrase, and "!' can represent a word of surprise or shock. A number of research pieces in sentence boundaries mainly used the machine learning methods, such as the hidden Markov model [Mikheev, 2002], Maximum entropy [Jeffrey C.Reynar, 1997], conditional random fields [Katrin Tomanek, 1997], and neural networks [Tibor Kiss, 2006]. Recently, deep learning models have been applied to solve this issue and achieve good performance [Carlos-Emiliano Gonzalez-Gallardoa, 2018] [Carlos Emiliano Gonza lez Gallardo1, 2018]. Until now, research about SBD has been confined to formal texts, such as news and European parliament proceedings, which have high accuracy using rule-based machine learning and deep learning methods due to the perfectly clean text data. There is no research about the SBD in noisy text that was extracted from the files in machine-readable formats. The FinNLP workshop in IJCAI-2019 is the first proposal of FinSBD-2019 shared tasks that detect sentence boundary in noisy text of finance documents [A Ait Azzi, 2019].

The purpose of FinSBD-2019 shared tasks is to detect the beginning and ending parts of sentences in the noisy text extracted from financial pdf documents in two languages: English and French. As shown in the English text T1, the provided dataset is a json file containing "text" that has been word tokenized using NLTK, and begin_sentence and end_sentence correspond to all indexes of tokens marking the beginning and the ending of well-formed sentences in the text.

T1: [{'text': "Invesco Funds , SICAV\n Vertigo Building – Polaris\n Prospectus\n 2 - 4 rue Eugène Ruppert\n L - 2453 Luxembourg\n 20 August 2013 An open - ended umbrella investment fund established under the laws of Luxembourg and harmonised under ……",
'begin_sentence': [22, 50, 79, 120, 128, 1240, 1290, 1315, 1344, 1354,……],
'end_sentence': [49, 78, 119, 125, 156, 1289, 1314, 1343, 1353, 1397,….]}

The goal of the task is to detect the beginning and ending index of English and French sentence text that has been tokenized. We observed that the critical words in the sentence clearly indicated the beginning and ending part of a sentence. For example, in the English text, most of the time, '.', ';' and et al. are at the end of a sentence. The attention mechanism is useful for detecting the weights of words in NLP tasks [Ke Tian 2019]. Therefore, the word2vec-based deep attention model is proposed to detect the beginning and ending index in English and French sentence texts.

Section 2 explains the details of our methods. Section 3 shows experimental configurations and discusses the results. Then, we conclude this paper in Section 4.

2 Deep Attention Model

The structure of the proposed method for detecting the beginning and ending index of sentence texts in English and French is shown in Figure 1. The creating training data and word embedding of English and French texts are first described in Section 2.1. The attention of the long short-term memory (LSTM) [Sepp Hochreiter, 1997] model is described in Section 2.2, and the ensemble result is presented in Section 2.3.

2.1 Recreating Train Data and Word Embedding

In the provided train, dev, and test data in English and French, the words have been tokenized. We observed that the end part of a sentence does not just use punctuation like '.' and ';' and includes some words like 'as' and 'and', which caused the ending part be complex. Like the ending part, the beginning

Proceedings of the First Workshop on Financial Technology and Natural Language Processing (FinNLP@IJCAI 2019), pages 88-92, Macao, China, August 12, 2019.

part of sentence also is not just words which beginning with upper letter like ‘The', ‘Given', ‘This', also include symbol character like ‘(', ‘-'. Therefore, using only the rule to detect the beginning and end of a sentence may be not useful. We found that the unusual beginnings and endings are identifiable by context. Moreover, we found that the provided training data was not easy to use for the deep learning model. We recreated the new training data that can be applied to deep learning model based on provided train, and dev data. The procedure for recreating the new training data is shown in Fig 2.

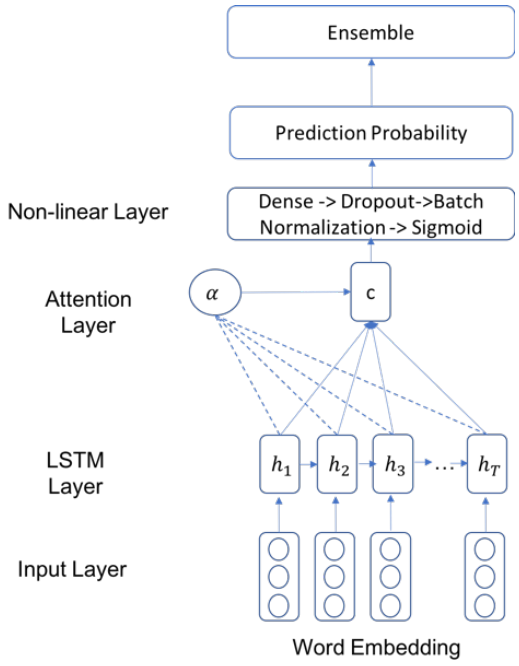

Figure 1: Deep attention model

We take the T1 sentence as an example to describe how to recreate the new train data. As the provided T1 JSON data, each tokenized word is labeled. For example, ‘Invesco', ‘Funds', ‘An', ‘amended', and ‘2016' are labeled ‘O', ‘O', ‘BS', ‘ES', and ‘O' respectively. As each tokenized word, the previous n tokenized words following n tokenized words of each tokenized word are taken to concatenate into a new sentence. For example, take the 5 previous words as an example. As the first word is "Invesco" in the T1, there are no previous 5 words, so we added 5 "pre" words at the beginning of the sentence. Therefore, the new sentence for "Invesco" is the T2 sentence. With the beginning word "An", the new sentence is T3. As the end word of T1, there are no next 5 words, so we added 5 "EOS" words at the end of the sentence. Therefore, the new sentence for "2016" is T4. The labels of T2, T3, and T4 are "O", "BS", and "O" respectively, which are the same as the labels of the tokenized words "Invesco", "An", and "2016" respectively. The train, dev, and test data in English and French both use the same method to recreate data. There are three labels (O, BS, ES) for the train, dev, and test data. Therefore, the goal of the task is changed to classify the labels of new data.

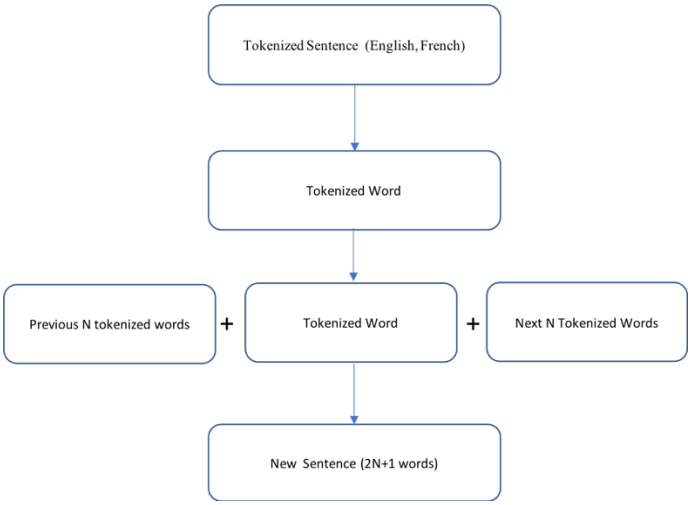

Figure 2: Procedure of recreating data

T2: Pre Pre Pre Pre Pre Invesco Funds , SICAV \n Vertigo Building.

T3: 2453 Luxembourg\n 20 August 2013 An open - ended umbrella investment.

T4: II – Version of APRIL 2016 EOS EOS EOS EOS EOS

Word embedding is the foundation of deep learning for natural language processing. We use the new train, dev, test text data to train the word embedding. At the beginning, the words are not converted into lowercase. In the recreated English text data, there are 1010868 recreated sentences with 14285 unique token words from the training, dev, and test data. In the French text, there are recreated 1053437 sentences with 15784 unique token words from the training, dev, and test data. The CBOW model [Tomas Mikolov, 2013] is taken to train word vectors for the English and French text data, and the word2vec dimension is set to 100.

2.2 Attention-based LSTM Model

Through the task train data, we observe that some keywords could help decide the category of a sentence. For example, ".", ";", and "as" indicate the ending part of sentence. The ES category is indicated by the keywords "The" or "This". Thus, some keywords in the sentence have more importance to predict the label of sentence text. Since the attention mechanism can enable the neural model to focus on the relevant part of your input, such as the words of the input text, the attention mechanism is used to solve the task. In this paper, we mainly use the feed-forward attention mechanism [Colin Raffel, 2015]

The attention mechanism can be formulated with the following mathematical formulation:

$$e_t = a(h_t),\ \alpha_t = \frac{exp(e_t)}{\sum_{k=1}^{T} exp(e_k)},\ c = \sum_{t=1}^{T} \alpha_t h_t$$

In the above mathematical formulation, a is a learnable function and only depend on h_t. The fixed-length embedding c of the input sequence computes with an adaptive weighted average of the state sequence h to produce the attention value.

In the structure of the proposed model, as the LSTM layer, the embedding dimension and max word length of word embedding are set to be 100 and 2n+1 (n is the number words surrounding the tokenized word), respectively, as the embedding dimension. The embedding layer of the word embedding matrix as an input layer of LSTM and the size of the output dimension is 300. We used the feed-forward attention mechanism as the attention layer. As the non-linear layer, the activation function is to dense the output of the attention layer to be 256 dimensions, and by using the dropout rate of 0.25, the output result after the dropout rate will be batch normalization. Finally, the sigmoid activation function that will dense the dimension of batch normalization input will be the length of the label as the final output layer.

2.3 Ensemble Result

In the model training stage, the 10-fold cross validation is used to train the deep attention model for predicting the test data. We sum 10 folds of predict probability and get the mean value of 10 folds for the final predict probability result. In the task, two results for each language are submitted: one result is based on the word embedding of the deep attention model, and the other result is based on word embedding of the convolutional neural networks (CNN) [Kim, 2014] models.

3 Experiments

3.1 Experiment Design and Implementation

In the experiment, rule-based, CNN and the proposed deep attention model have been implemented in the task. Moreover, in the data processing stage, keep the upper letter of words to train the word embedding in the English and French text. In addition, we test the different numbers of words surrounding each tokenized word. The numbers 5, 8, 10 are taken to be tested in the experiment.

As the simple rule-based method in the experiment. In the English task, we just determined the end index by the '.', '.\n', '!', ':', ';' token words, and the beginning index is detected by the word that the beginning character of token word is upper letter or the token word is '('. As the French task, the beginning index is determined by the word that the beginning character of token word is upper letter, and the end index is detected by the '.', '.\n', '!', ':', ';' token words.

The 10-fold cross-validation to predict the test data is used in the model. The deep model in our research was implemented with Keras [Keras, 2019]

Based on the evaluation requirements of the FinSBD Task, the F-scores are taken to evaluate the performance of the proposed model in the paper.

3.2 Result and Discussion

The results of rule-based, CNN, LSTM and the deep attention model are shown as Table 2. From Table 2, as the English task, the worst performance of model is rule-based which the F1 score of ES, BS is 0.17, and 0.72. Moreover, the F1 score of ES, BS of CNN model is 0.82 and 0.89. The F1 score of BS, and ES of the deep attention model is 0.83 and 0.91, respectively, which is better than the CNN and LSTM model. The F1 score of the ES and BS of the deep attention model is also better than the CNN model and Rule-based in the French task. The result showed that deep attention model is effective to predict the beginning and end index of task sentence in English and French.

Model	English			French		
	BS	ES	Mean	BS	ES	Mean
Rule-based	0.17	0.72	0.445	0.34	0.51	0.425
CNN	0.82	0.89	0.855	0.89	0.90	0.895
Deep Attention	0.83	0.91	0.875	0.91	0.92	0.915

Table 1. Experiment Results: F-score of English, French tasks using Rule-based, CNN and Attention-LSTM, and the surrounding number of words is 5

N words	English			French		
	BS	ES	Mean	BS	ES	Mean
5	0.83	0.91	0.875	0.91	0.92	0.915
8	0.86	0.90	0.88	0.91	0.92	0.915
10	0.88	0.91	0.895	0.91	0.92	0.915

Table 2. Experiment Results: F-score of English, French tasks, and the surrounding number of words is 5, 8,10 respectively for deep attention model

In order to test how the number of surrounding words affect performance, the 5, 8, and 10 surrounding words for the deep attention model were implemented in the experiment, and the result is shown in Table 2. Based on the result, the best performance of SBD prediction is the 10 surrounding numbers for tokenized words in which the F1 score of ES and BS in the English task are 0.88 and 0.91, respectively. However, the 5, 8, and 10 surrounding words for the deep attention model in the French task, the score of F1 is the same.

Therefore, we may infer that the more words surround the tokenized word, the better the prediction is in English task. As the French task, the number of words surrounding the tokenized word may not influence the performance of prediction in this task.

Team	ES	BS	Mean
AIG1	0.88	0.89	0.885
seernet1	0.85	0.9	0.875
aiai1	0.83	0.91	0.87
isi1	0.83	0.89	0.86
NUIG1	0.81	0.9	0.855
isi2	0.82	0.89	0.855
AIG2	0.83	0.88	0.855
AI_Blues2	0.82	0.87	0.845
AI_Blues1	0.82	0.87	0.845
mhirano1	0.78	0.89	0.835
aiai2	0.79	0.88	0.835
NUIG2	0.81	0.85	0.83
HITS-SBD2	0.8	0.86	0.83
HITS-SBD1	0.8	0.86	0.83
PolyU_CBS-CFA_NN1	0.77	0.86	0.815
PolyU_CBS-CFA_RFC1	0.7	0.86	0.78
PolyU_CBS-CFA_RFC2	0.68	0.86	0.77
mhirano2	0.58	0.67	0.625

Table 3 Leaderboard FinSBD in English task

Based on the final report about FinSBD-2019 shared tasks as shown in the Table 3 and Table 4. As the English task, the ranking of our team is 3, and aiai1 is the result of deep attention model. As the French task, aiai1 is ranked 2, and some indicators such as ES rank number 1. Due to my busy schedule, we only submitted 5 words around the tokenized word before the deadline, and we found that there is code in error in CNN model in last submission which caused the abnormal score (aiai2) in these tasks. Currently, the updated result is shown in the Table 2. Based on Table 2, if we take the result of 10 surrounding words for submission, our team would rank number 1 in the English task. The result showed that the proposed model could effectively predict the beginning and ending indexes of words in the noisy text of finance documents in these two tasks.

Team	ES	BS	Mean
seernet	0.91	0.93	0.92
aiai1	0.91	0.92	0.915
NUIG1	0.9	0.92	0.91
NUIG2	0.9	0.92	0.91
isi1	0.9	0.91	0.905
isi2	0.89	0.91	0.9
AI_Blues1	0.85	0.88	0.865
AI_Blues2	0.84	0.88	0.86
PolyU_CBS-CFA_RFC1	0.84	0.88	0.86
mhirano1	0.82	0.89	0.855
PolyU2	0.83	0.87	0.85
PolyU_CBS-CFA_NN1	0.83	0.87	0.85
PolyU_CBS-CFA_RFC2	0.81	0.88	0.845
mhirano2	0.67	0.68	0.675
aiai2	0.01	0.02	0.015

Table 4 Leaderboard FinSBD in French task

4 Conclusion

This paper mainly discusses how we tackle the FinSBD-2019 shared task. There are two tasks which predict beginning and ending index of words in the sentence text of finance document in English and French. In order to tackle the these task, we firstly recreate the train, dev, and test data so that can be applied to deep learning model. Then, the deep word embedding-based attention model is proposed to classify the labels of recreated data. The experimented result showed that the proposed model could effectively solve the goal of task and achieve very good performance in these tasks.

References

[Abderrahim, 2019] Ait Azzi Abderrahim, Bouamor Houda, and Ferradans Sira. The FinSBD-2019 Shared Task: Sentence boundary detection in PDF Noisy text in the Financial Domain, The First *Workshop on Financial Technology and Natural Language Processing of IJCAI 2019*:16–19, Macao, China, August 2019.

[Carlos-Emiliano Gonza lez Gallardo1, 2018] Carlos-Emiliano Gonza lez Gallardo1 and Juan-Manuel Torres-Moreno. Sentence boundary detection for french with sub-word level information vectors and convolutional neural networks. *https://arxiv.org/abs/1802.04559*, 2018.

[Carlos-Emiliano Gonza´lez-Gallardoa, 2018] Carlos-Emiliano Gonza´lez-Gallardoa, Elvys Linhares Pontes, Fatiha Sadat, Juan-Manuel Torres-Moreno. Automated sentence boundary detection in modern standard arabic transcripts using deep neural networks. *Procedia Computer Science*, (142):339–346, 2018.

[Colin Raffel, 2015] Colin Raffel and Daniel P. W. Ellis. Feed-forward networks with attention can solve some long term memory problems. *https://arxiv.org/abs/1512.08756*, 2015.

[Gregory Grefenstette, 1994] Gregory Grefenstette and Pasi Tapanainen. What is a word, what is a sentence? problems of tokenization. In *In Proc.3rd International Conference on Computational Lexicography (COM- PLEX'94)*, pages 79–87, 1994.

[Jeffrey C.Reynar, 1997] Jeffrey C.Reynar and Adwait Ratnaparkhi. A maximum entropy approach to identifying sentence boundaries. In *Proceedings of the fifth conference on Applied natural language processing*, pages 16–19, Washington, USA, April 1997.

[Katrin Tomanek, 1997] Katrin Tomanek, Joachim Wermter and Udo Hahn. Sentence and token splitting based on conditional random fields. In *Proceedings of the 12th World Congress on Health (Medical) Informatics*, pages 16–19, Washington, USA, April 1997.

[Ke Tian, 2019] Ke Tian and Zi Jun Peng. aiai at FinNum task: Financial numeral tweets fine-grained classification using deep word and character embedding-based attention model. *The 14th NTCIR Conference*, Tokyo, Japan, June 2019.

[Keras, 2019] Keras. The python deep learning library. *https://keras.io*. Accessed: May 2019

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. *https://arxiv.org/abs/1408.5882*, 2014.

[Mikheev, 2002] Andrei Mikheev. Periods, capitalized words, etc. *Computational Linguistics*, 28(3):289–318, September 2002.

[Nagmani Wanjaria, 2016] Nagmani Wanjari, G. M. Dhopavkar Prof and Nutan B. Zungre. Sentence boundary detection for marathi language. *Procedia Computer Science*, 78:550–555, 2016.

[Sepp Hochreiter, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735-1780, 1997.

[Tibor Kiss, 2006] Tibor Kiss and Jan Strunk. Unsupervised multilingual sentence boundary detection. *Computational Linguistics*, (32):485–525, April–June 2006.

[Tomas Mikolov, 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. https://arxiv.org/abs/1310.4546, 2013.