Finance document Extraction Using Data Augmentation and Attention

Ke Tian
OPT Inc, Tokyo, Japan
tianke0711@gmail.com

Zijun Peng
Harbin Institute of Technology (Weihai),China
2986320586@qq.com

Abstract

This paper mainly describes the aiai that the team submitted to the FinToc-2019 shared task. There are two tasks. One is the title detection task from non-titles in the finance documents. Another one is the TOC (table of contents) prediction from the finance PDF document. The data augmented and attention-based LSTM and BiLSTM models are applied to tackle the title-detection task. The experiment has shown that our methods perform well in predicting titles in finance documents. The result achieved the 1st ranking score on the title detection leaderboard.

1 Introduction

In the finance field, a great number of financial documents are published in machine-readable formats such as PDF file format for reporting firms' activities and financial situations or revealing potential investment plans to shareholders, investors, and the financial market. Official financial prospectus PDF documents are the documents that describe precisely the characteristics and investment modalities of investment funds. Most prospectuses are published without a table of contents (TOC) to help readers navigate within the document by following a simple outline of headers and page numbers and assist legal teams in checking if all the contents required are fully included. Thus, automatic analyses of prospectuses by which to extract their structure are becoming increasingly more vital to many firms across the world. Therefore, the second Financial narrative processing (FNP) workshop is the first proposal of the FinTOC-2019 shared task to focus on the financial document structure extraction (Rmi Juge, 2019). Two tasks are contained in the FinTOC-2019 task.

Title detection (task 1): This is a two-label classifications task that detects text block as titles or non-titles in the financial prospectuses document. For example, in the training data, there are about 9 fields:

(1) text blocks: a list of strings computed by a heuristic algorithm; the algorithm segments the documents into homogeneous text regions according to given rules.

(2) begins_with_numbering : 1 if the text block begins with a numbering such as 1., A, b), III., etc.; 0 otherwise

(3) is_bold: 1 if the title appears in bold in the PDF document; 0 otherwise

(4) is_italic: 1 if the title is in italic in the pdf document; 0 otherwise

(5) is_all_caps: 1 if the title is all composed of capital letters; 0 otherwise

(6) begins_with_cap: 1 if the title begins with a capital letter; 0 otherwise

(7) xmlfile: the xmlfile from which the above features have been derived

(8) page_nb: the page number in the PDF where appears the text bock

(9) label: 1 if text line is a title, 0 otherwise

There are eight fields, which are the same as the training data in the test data except the label field. The goal of this task is to detect the text blocks as titles or non-titles.

TOC generation (task2): this subtask will predict the TOC from the PDF document. There are annotated TOCs in the XML format in the document structure as well as PDFs. The XML file is composed of TOC-titles with three attributes:

(1) title: a title of the document

(2) page: the page number of the title

(3) matter_attrib: whether the title appears on the front page, the body, or the back matter of the documents.

There are about five levels of titles that can be inferred from the hierarchy of the XML file. The training documents are the same as those for the

Proceedings of the Second Financial Narrative Processing Workshop (FNP 2019), pages 1–4
Turku, Finland, 30 September, ©2019 Linköping University Electronic Press

title detection sub-task. The test documents are the same as the training data with the title labels. The PDF and XML documents are provided in the test data. The goal of this task is to generate the TOC XML file of the test data.

In this research, we first recreate the training and test data using data augmentation to be new training and test data for task1, and then we use attention-based LSTM and BiLSTM modes to detect the title in task1. 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 Methods

The structure of the proposed method for tackling with task 1 is shown in Figure 1. The recreation of training, test data and word embedding using data augmentation are described in Section 2.1. The attention of the long short-term memory (LSTM) (Sepp and JRgen, 1997) model and BiLSTM (Mike and Kuldip, 1997) are described in Section 2.2, and the ensemble result is presented in Section 2.3.

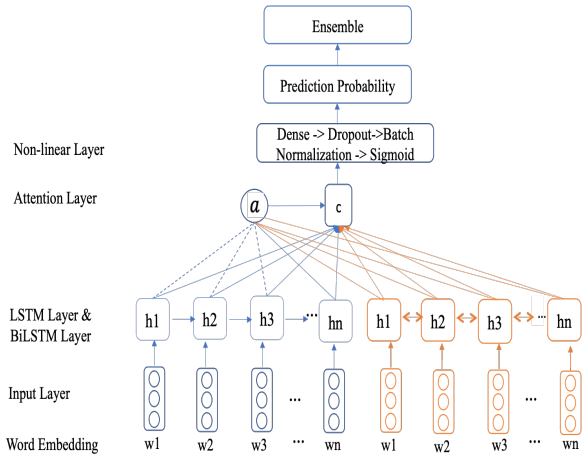

Figure 1: The Structure of attention-based LSTM and BiLSTM

2.1 Data Augmentation

The train and test data are provided in the title detection task, except the label field. There are eight fields used to predict the label. Before using these data for prediction, the train and test data are recreated. The procedure for recreating the new training and test data is shown in Figure 2.

As with the text blocks, we used the NLTK first to tokenize the text, and then all the tokenized words were converted to be lower. Secondly, we

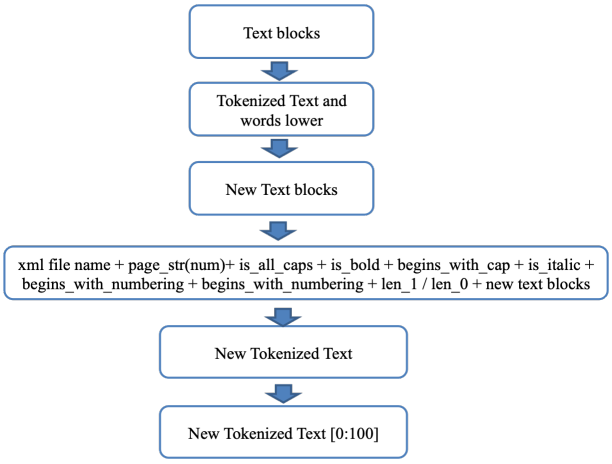

Figure 2: The procedure of data augmentation

computed the length of all text blocks labeled 1, namely the title in the train data. We observed the length of all title text blocks is less than 60. Therefore, if the length of text block is more than 60, the len_1 word is added before the new text block; otherwise the len_0 is added. Thirdly, the begins_with_numbering is added if the value of this field is 1 before len_1 or len_0, the same as is_italic, begins_with_cap, is_bold, is_all_caps words are added subsequently if the field value is 1. Finally, the page number, and xml file name of text blocks are added in the front of the previous new text blocks. For example, take the first text block DB PWM I in the train data to explain the procedure. The other seven fields are as follows: begins_with_numbering (0), is_bold (1), is_italic (0), is_all_caps (1), begins_with_cap (1), xmlfle (LU0641972152-LU0641972079_English_2015_DBPWMIGlobalAllocationTracker-.xml), page_nb (1). Based on the data augmentation procedure, the new text block is LU0641972152-LU0641972079_English_2015_DBPWMIGlobalAllocationTracker.xml page_1 is_all_caps is_bold begins_with_cap len_0 db pwm i.

Word embedding is the foundation of deep learning for natural language processing. We use the new train and test text data to train the word embedding. In the recreated text data, there are recreated sentences with 14,285 unique token words from the training, dev, and test data. The CBOW model (Tomas et al., 2013) is taken to train word vectors for the recreated text block, and the word2vec dimension is set to 100.

2.2 Attention-based LSTM and BiLSTM Model

After the data augmentation is completed, we only take the previous 120 words of each text block as the input sentence. In the structure of the proposed model as shown in Fig. 1, the LSTM and BiLSTM layer, the embedding dimension and max word length of word embedding are set to be 100 and 120, respectively, as the embedding dimension. The embedding layer of the word embedding matrix is an input layer of LSTM, and the size of the output dimension is 300.

Through the task train data, we observe that some keywords could help indicate the label of the text block. For example, most of title text blocks have the following features: len_0, begins_with_cap, is_bold, is_all_caps. hus, some keywords in the new data have more importance to predict the label of the text block. 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 (Tian and Peng, 2019), attention mechanism is used to solve the task. In this paper, we mainly use the feed-forward attention mechanism (Colin and Daniel, 2015). The attention mechanism can be formulated with the following mathematical formulation:

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

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.

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 title detection task, two results for each language are submitted: one result is based on the word embedding of attention-based LSTM, and the other result is based on word embedding of the BiLSTM.

Team name	Score
Aiai_1	0.9818997315023511
Aiai_2	0.9766402240293054
UWB_2	0.9723895009266195
FinDSE_1	0.9700572855958501
FinDSE_2	0.9684006306179805
UWB_1	0.9653446892789734
Daniel_1	0.9488117489093626
Daniel_2	0.9417339436713312
YseopLab_1	0.9124937810249167
YseopLab_2	0.9113421072180891

Table 1: Leader board of title detection task.

3 Experiment and Result

In the experiment, the proposed deep attention model has been implemented in the task. We have submitted two results. One result is attention-based LSTM. The other one is the attention-based BiLSTM. The evaluation metric used for this title detection task is the weighted F1 score. The final result of attention-based LSTM and BiLSTM ranking 1st and 2nd in the leader board are shown in Table 1.

4 Conclusion

We have described how we tackle title detection in the FinToc-2019 shared task. Firstly, we augmented the text block and added another 7 fields to recreate the new training and test data. Then, the attention-based LSTM and BiLSTM models are experimented on. The experimental result showed that the proposed model could effectively solve the goal of the task and achieve a very good performance in carrying out this task.

For future work, more models or methods will be implemented for the task. Moreover, we have planned to tackle Task 2.

References

Raffel Colin and P. W. Ellis Daniel. 2015. Feed-forward networks with attention can solve some long-term memory problems. arXiv:1512.08756.

Schuster Mike and K. Paliwal Kuldip. 1997. Bidirectional recurrent neural networks. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 45:2673–2681.

Sira Ferradans Rmi Juge, Najah-Imane Bentabet. 2019. The fintoc-2019 shared task: Financial document structure extraction. In *The Second Workshop on Financial Narrative Processing of NoDalida 2019*, Turku, Finland.

Hochreite Sepp and A Schmidhuber JRgen. 1997. Long short-term memory. *Neural Computation*, 9:1735–1780.

Ke Tian and Zijun Peng. 2019. aiai at finnum task: Financial numeral tweets fine-grained classification using deep word and character embedding-based attention model. In *The 14th NTCIR Conference*.

Mikolov Tomas, Sutskever Ilya, Chen Kai, Corrado Greg, and Dean Jeffrey. 2013. Distributed representations of words and phrases and their compositionality. arXiv:1310.4546.