PolyU_CBS-CFA at the FinSBD task: Sentence Boundary Detection of Financial Data with Domain Knowledge Enhancement and Bilingual Training

Mingyu Wan[1*], **Rong Xiang**[2], **Emmanuele Chersoni**[1], **Natalia Klyueva**[1], **Kathleen Ahrens**[3], **Bin Miao**[4], **David Broadstock**[4], **Jian Kang**[4], **Amos Yung**[3] and **Chu-Ren Huang**[1]

[1]Department of Chinese and Bilingual Studies, The Hong Kong Polytechnic University
[2]Department of Computing, The Hong Kong Polytechnic University
[3]Department of English, The Hong Kong Polytechnic University
[4]School of Accounting and Finance, The Hong Kong Polytechnic University
{mingyu.wan, kathleen.ahrens, bin.miao, david.broadstock, jian.kang, amos.yung, churen.huang}@polyu.edu.hk, {xiangrong0302, emmanuelechersoni, natalka.kljueva}@gmail.com

Abstract

Sentence Boundary Detection is a basic requirement in Natural Language Processing and remains a challenge to language processing for specific purposes especially with noisy source documents. In this paper, we deal with the processing of scanned financial prospectuses with a feature-oriented and knowledge-enriched approach. Feature engineering and knowledge enrichment are conducted with the participation of domain experts and for the detection of sentence boundaries in both English and French. Two versions of the detection system are implemented with a Random Forest Classifier and a Neural Network. We engineer a fused feature set of punctuation, digital number, capitalization, acronym, letter and POS tag for model fitting. For knowledge enhancement, we implement a rule-based validation by extracting a keyword dictionary from the out-of-vocabulary sequences in FinSBD's datasets. Bilingual training on both English and French training sets are conducted to ensure the multilingual robustness of the system and to extend the relatively small training data. Without using any extra data, our system achieves fair results on both tracks in the shared task. Our results (English[1]: F1-Mean = 0.835; French: F1-Mean = 0.86) as well as a post-task quick improvement with self-adaptive knowledge enhancement based on testing data demonstrate the effectiveness and robustness of bilingual training with multi-feature mining and knowledge enhancement for domain-specific SBD task.

*Contact Author

[1]This is the adapted result as illustrated in Section 5.

1 Introduction

Sentence Boundary Detection (SBD), which aims at detecting/disambiguating sentence boundaries of texts, is a fundamental step in many Natural Language Processing (NLP) applications. It should be carried out before other critical components of NLP, *e.g.* part-of-speech (POS) tagging, syntactic-semantic-discourse parsing, information extraction or machine translation. Existing SBD approaches have shown promising results for languages that have dependable orthographic conventions to mark beginning and ending of sentences, such as in English and many Europeans languages. Relevant recent work include (*e.g.* [Riley, 1989; Reynar and Adwait, 1997; Mikheev, 2002; Palmer and Hearst, 1997; Read, 2012]). However, previous work in SBD mainly dealt with well-formed and clean data such as articles from the Brown corpus [Hearst, 1994] or Wall Street Journal [Palmer and Hearst, 1997].

SBD remains challenging in two scenarios. The first involves documents encoded in non-text formats, such as Adobe PDF format, or other image formats. They provide the exact layout of a human readable document on a wide range of machines. However, texts converted from PDF documents by OCR software are usually noisy and potentially with the loss of substantial formatting features. This chaos leads to difficulties in SBD, and so far has been under-researched. The second involves languages whose orthography does not mark sentence boundary conventionally. For instance, [Huang and Chen, 2017] shows that using the period punctuation will lead to significant divergence from sentence boundaries. What they proposed are followed by [Hou *et al.*, 2019] in their Menzerath-Altamann based power relations between a clause and its constituent words (instead of between sentences and words). In this current paper, we deal with the first challenge.

There are a number of issues to be addressed when applying SBD to financial documents. Unlike passages of

Proceedings of the First Workshop on Financial Technology and Natural Language Processing (FinNLP@IJCAI 2019), pages 122-129, Macao, China, August 12, 2019.

formal texts, financial documents are often heavily populated with rich tables of data—sometimes stretching over multiple pages—and figures, titles, dates and keywords of various types. The presence of such non-textual information is admittedly not unique to financial documents, but it should be noted that many financial documents also do not come in clean/easily machine readable structures. Detecting sentence boundaries on the basis of periods/stops may also be less straightforward, for example the presence of company tickers in a document may introduce some difficulties in cleanly identifying sentence boundaries, especially if appended with exchange, for example the full ticker for China Light and Power (CLP) company listed in Hong Kong can be written as '0002.hk'. Additionally, sentences may contain various financial terms/acronyms, including company name abbreviations, that may impact the syntactic structure and hence generate sentence confusion.

As such financial documents present a range of challenges that result in the need to use a hybrid of language processing tools in combination with knowledge enrichment specific to the financial domain. With such endeavors, we can expect chances of achieving SBD with reasonable levels of accuracy.

In the following sections, we will review some related work in Section 2, describe the features and methodology in Section 3, show and discuss the results in Section 4 and finally conclude this work in Section 5.

2 Related Work

Sentence Boundary Detection is a fundamental issue in Natural Language Processing, which can be viewed as a classification issue. Current studies normally tackle the problem as the identification of the truthful sentence ending markers among the ambiguous ones. The history of SBD development witnessed machine learning as the earliest attempts (*e.g.* [Riley, 1989; Reynar and Adwait, 1997; Palmer and Hearst, 1997]), with rule-based systems coming afterwards (*e.g.* [Mikheev, 2002; Mikheev, 2000]). There has been some work occasionally using unsupervised techniques (*e.g.* [Kiss and Strunk, 2006]).

Early attempts have already shown promising results of SBD, but all with well-formed data. For example, since Riley [1989]'s very first work of SBD, a 99.8% accuracy was reported by investigating only a single punctuation mark, *i.e.* period, with the use of Decision Tree classifier trained on 25 million words of newswire texts. Hearst [1994] achieved a 1.5% error rate by using Feed-forward Neural Network of POS features trained on the Brown corpus. Later on, Palmer and Hearst [1997] developed SATZ, a system that used features of contextual POS distribution and words-as-vectors of the target punctuations *via* NN and DT with training on the 30 million WSJ corpus. Their work hit the record of the state-of-the-arts of SBD with error rates of 1.1% for NN and 1% for DT. Synchronically, Reynar and Adwait [1997] adopted supervised Maximum Entropy learning with two sets of features: in-domain financial uses, *e.g.* honorifics (Mr., Dr., etc.) and corporate designations (Corp.); domain-independent abbreviations, as well as '!', '.' or '?' as potential boundaries. This work, however, was slightly inferior in performance with accuracies of 98.8% and 97.9% respectively for the domain-dependent system and accuracies of 98.0% and 97.5% respectively on the portable system.

Modern machine learning techniques provide us with a series of statistical models focusing on data patterns, nonlinear features and forecast accuracy. With the breakthrough of computing technology, the nonlinear methods became feasible in 1980s, as represented by Breiman *et al.* [1984]'s work with tree-based and regression models. Since then, an increasing number of tree-based models, both supervised and unsupervised, were developed and promptly emerged, such as Random Forests, Boosting Trees, *etc.* Prior to traditional classifiers, Neural Network methods were introduced to SBD by McCulloch and Pitts [1943]. From 1980s, the Neural Network, incorporating the Bayesian Neural Network, was resurged by the upgrades of computing technology as well as the appearance of back-propagation algorithms. Unlike tree-based methods, NN methods present smooth functions of parameters, which facilitate the development of Bayesian inference.

Complementary to machine learning, Mikheev [2000; 2002] employed rule-based systems for SBD, and reported error rates of 0.31% and 0.20% with training on WSJ and the Brown corpus respectively. Recently, Read *et al.* [2012] and Griffis *et al.* [2016] both adopted several state-of-the-art NLP toolkits for SBD with mixed datasets of varied formality and specificity. Both works showed that the existing toolkits for SBD in specific domains are worse without resistance to domain-transfer or formality change.

To approach the above-mentioned problem of SBD in recent decade, we need to enforce renewed efforts of further shaping the NLP tools as well as addressing to domain-specific and informality issues, with aim of refreshing a new record in the SBD history.

In this paper, we propose a feature-oriented and knowledge-enriched approach to detecting the beginnings and endings of financial data in both English and French by using a Random Forest Classifier (RFC) and a Neural Network (NN). In addition to feature engineering, model fitting and parameter tuning, we conduct post-classification knowledge enhancement with a rule-based keyword validation on the predictions by automatically identifying and extracting the out-of-boundary word sequences from the FinSBD datasets [Ait Azzi *et al.*,]. In addition, the main body of noise in the datasets provides us with a useful resource as a by-product of the task for rectifying the ambiguous boundaries.

3 Features and Methodology

In this section, we describe the features and methodology of this work with a pipeline of feature engineering,

classification and aspect knowledge enrichment (post-classification validation).

3.1 Feature Engineering

Feature engineering plays an important role in machine-learning, involving the selection of a subset/fused set of informative and discriminative features with dual purposes of dimension reduction and classification leverage [Garla and Brandt, 2012]. In general classification tasks, features typically include bags of characters, words, n-grams and/or concepts in a text corpus, which, however, causes high dimensionality of feature space in lowering classification efficiency.

Feature selection is necessary when feature space is overloaded or in redundancy. Algorithms of term frequency, chi-square, information gain, mutual information or relevance score are usually adopted in automatic feature selection (*e.g.* [Lee and Lee, 2006; Chen *et al.*, 2009]); domain knowledge is also helpful to guide the feature engineering process. In this sentence segmentation task, we utilize a semi-automatic selection method that both considers high frequency words and keyword knowledge, as shown in Sections 3.1 and 3.3 below.

By a close observation and comparison of the scanned documents with the gold boundaries in the datasets [Ait Azzi *et al.*,], we introspect the key sections of erroneous predictions both manually and statistically. This leads to the inclusion of the following sets of salient features for fitting the classifiers.

- **Two sets of punctuation:** Punctuation serves as important cues for SBD and has been proved as the most useful feature in SBD. In addition to using period as the baseline feature set, we also include a set of special punctuation-symbols that are prevalent in financial data, such as the dollar signs, math operators and copyright symbols, as listed below:
  - PUNC_SET1 = [‘.’]
  - PUNC_SET2 = [‘?’, ‘!’, ‘;’, ‘%’, ‘-’, ‘/’, ‘”’, ‘\’, ‘)’, ‘(’, ‘∗’, ‘⊡’, ‘<’, ‘≥’, ‘>’, ‘≤’, ‘•’, ‘e’, ‘$’, ‘£’, “‘”, ‘©’, ‘®’]

  By adding the second punctuation set in the attribute table, we got 1% F1 improvement for the validation sets of both languages.
- **Initially capitalized words:** As suggested in the samples of gold boundaries, most BEGINs are marked with initially capitalized words (*e.g.* “Distribution_BEGIN”) or the ENDs are largely preceding such words (*e.g.* “._Enter_END The_BEGIN sales”). Although it introduces some confusing information for the classifiers, such as the keywords in titles, tables, figures, etc., the inclusion of this type of feature on average improves 2% F1 score of validation for both languages. In order to associate such feature with both BEGINs and ENDs, we use a feature array of three dimension in the pre-, current, post- positions to maximally represent its discriminativeness in predicting boundaries.
- **Acronyms or Abbreviations:** Acronyms or abbreviations are also salient features for marking boundaries as indicated in both the existing literature and the FinSBD datasets. For example, “UBS_BEGIN” “co” or “kiid” show that acronyms tend to co-occur (all capitalized words) or not occur (all lower cased words) with boundaries. As such, we construct a three-dimension attribute array of storing the Boolean value of all-word-capitalization in the pre-, current, post- positions. This feature set also improves around 1% F1 in the validation set for both languages.
- **Digital numbers:** Digital numbers is a common property of financial data which causes confusion for disambiguating *e.g.* decimal points from endings, as in “10.3”. To identify both the left and right context of the target period, we also construct a three-dimension attribute array for representing such cases. This feature set helps improve 2% F1 for the French validation set and 1% F1 for the English validation set.
- **Letters or Roman numbers:** As another salient feature in financial data, letters (*e.g. A-Z, a-z* alphabetical letters) or Romance numbers (*I, II, . . . , XII*) are highly suggestive of non-boundary tokens. Therefore it also serves as a useful feature for excluding the wrong boundaries. A tri-gram feature array is also constructed to represent such information in the pre-, current, post- positions, which helps around 1% F1 improvement of both validation sets.
- **POS tags:** Despite the fact that sentence segmentation occurs prior to part-of-speech tagging in NLP processing, the pos information of individual tokens can, in turn, indicate the phrasal structure of a word sequence which may provide useful cues to the identification of verbal sentences or alternatively, the non-clausal noun phrases (keywords). By including the three-dimension POS feature (the UPOS tag set by UDPipe[2]) in our experiment, our system is further optimized with 3% F1 increase for both validation sets.
- **Enter (‘\n’):** Enter (‘\n’) seems a universal feature for any type of document. But after a close look into the converted pdf documents in the FinSBD datasets, we found that Enters (‘\n’) is strongly associated with the conversion errors caused by the pdf scanning. With the inclusion of such feature in a three-dimension array, we further improve the system with 1% F1 for both validation sets.

For maximizing the discriminative power of the above features, we construct a fused feature set of 24 dimensions to fit the machine learning models and get an optimized performance (English: F1-Mean = 0.87; French: F1-Mean = 0.85) in the validation sets.

[2]http://ufal.mff.cuni.cz/udpipe

3.2 Classification Models

Ensemble Learning of Random Forest

Random Forest Classifier (RFC) is a tree-based ensemble classifier. It combines the decision of multiple Decision Tree (DT) classifiers where each classifier is generated using a random vector independently sampled from the input vector, and each tree casts a unit vote for the most popular class to classify an input vector [Breiman, 1999].

The ensemble RFC is generally more accurate than all the individual classifiers as it makes use of many naive classifiers that randomly use a subset of the vector, thus it is more robust to overfitting in comparison to traditional decision trees. As such, it is our first choice of classifier in this study. The RFC classifier is imported from the sklearn package[3] where a random forest is taken as a meta estimator (n_estimators is 10 by default) that fits a number of decision tree classifiers on various subsamples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

The design of a decision tree required appropriate attribute selection measure and a pruning method. In our experiment, we use randomly selected features at each node to grow a tree with an optimized setting of min_samples_split as 8, max_features as "log2" and random_state as 10. In addition, we set oob_score true to use out-of-bag samples to estimate the generalization accuracy.

The above setting of parameters of RFC work out the best performance in our estimation on the validation sets.

Neural Network

Resembling the biological neural networks, artificial Neural Networks (NN) approaches were proposed and led to great improvements in a number of NLP tasks. An artificial NN is usually composed of many simple processors (neurons) that are interconnected, operate in several layers and learn from input of examples. Considering the similar characteristics of financial data, we implemented a NN-based approach as a complementary work to the RFC model.

In the validation, we trained the Multi-layer Neural Network using Tensorflow[4] following several runs of parameter optimization. The optimization was done with bilingual training on both English and French training sets and testing on the English validation set. The optimal setup for training the model was a network of one input layer (density 300) and 1 hidden layer(density 100) with the relu activation function, and one output layer (3 categories) with the softmax activation function. As a loss function we used categorical crossentropy, and Adam as an optimizer. The batch size was set to 32, and the number of epochs was 5.

The input units were the feature vectors as described in Section 3.1, but certain features were excluded for the Neural Network run of the French trial for reasons of optimization.

[3]https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[4]https://www.tensorflow.org/api_docs/python/tf/nn

3.3 Domain Knowledge Enhancement

Knowledge enrichment has been proven to be a useful guidance for the post-processing of confused classification by statistical models [Ghosh and NAG, 2002]. In this task, OCR conversion of financial documents introduced large chunks of out-liars, such as the titles, dates, tables, figures, etc., which fail to fall into the traditional category of sentence boundary. These non-textual sections, as finely segmented with the "Enter" marks, cause erroneous predictions of boundaries. To solve this problem, we implemented a post-processing procedure to correct false positive predictions, as realized with the following two algorithms.

Keyword Extraction

Following the above-mentioned principle, we constructed a keyword dictionary with Algorithm 1. A broader definition of keyword is adopted here, including out-of-boundary words, symbols and phrases. We utilize the well segmented training and validation sets [Ait Azzi *et al.*,] for resource construction. Intuitively, if any word sequence locates between an "END" and the next "BEGIN", we regard it as a potential out-lier and construct keywords with further segmentation marked by "Enter". The final keyword dictionary with respect to both languages is then constructed, containing elements of key-value (keyword-frequency) pairs.

Algorithm 1 Keyword Extraction

Input: dataset
Output: keyword_dict

```
for i in len(dataset) do
    curword = dataset[i].
    nxtword = dataset[i+1].
    if "END" in curword then
        if "BEGIN" in nxtword then
            continue
        else
            add keyword to keyword_dict.
            update frequency in keyword_dict.
        end if
    end if
end for
refine keyword_dict with length threshold.
refine keyword_dict with frequency threshold.
return keyword_dict
```

To further control the quality of extracted keywords, we introduced *length threshold* and *frequency threshold* to filter those patterns that are too short or rarely occurring. As as result, a keyword dictionary of 16,501 keyword-frequency pairs is generated for the rule-based validation, as well as providing a financial resource for potential use in future IR inquiries.

Rule-based Validation

With the keyword dictionary generated, we used a rule-based approach for correcting the potentially wrong boundaries that are not in the keyword list, as illustrated in Algorithm 2.

Algorithm 2 Rule-based validation

Input: dataset, raw_pred, keyword_dict
Output: updated_prediction

```
for keyword in keyword_dict.key do
    for i in len(dataset)-len(keyword) do
        if word sequence match keyword then
            update raw_pred with NO_BOUNDARY
        end if
    end for
end for
return raw_pred
```

As shown in Algorithm 2, each keyword in the dictionary is used as a rule. Every word sequence that matches a keyword in the dictionary shall be forced to the "NO_BOUNDARY" class. This process will be executed iteratively until all the predictions are validated. As to be shown in the following section, the experimental results on the validation sets and the final results on the test sets of both languages have consistently verified the usefulness of knowledge enhancement in domain-specific classifications.

4 Results

In this section, we show our classification results with the following four aspects of comparisons.

4.1 Classifiers

This section focuses on the comparison of the classification performance of the two classifiers, *i.e.* RFC and NN, with the same experimental setting. A fused feature set is used and bilingual training is conducted. The classification results on both the validation (Dev) set and the test set of the two languages are shown in Figure 1 below:

As it is easy to see in Figure 1, RFC shows superior performance with 1% or 4 % F1 gain compared to NN for both the validation tasks (Dev_en and Dev_fr) and the testing of the French track (Test_fr). This is highly suggestive that the ensemble random forest is more fitted to the selected feature set in this work, while NN seems to demonstrate no advantage of wininning traditional classifiers despite having the same salient feature set in this task.

However, what is contradictory to our estimation is: NN outperforms RFC with 3.5% F1 discrepancy in the English track (Test_en), whereas the results of our validation on the English development set is opposite (RFC: 0.875 vs. NN: 0.86). Another obvious observation is: both classifiers' performance on the English test set

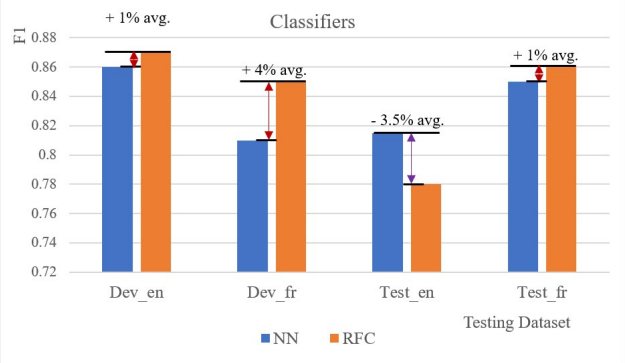

Figure 1: Classification Results of NN vs. RFC

drops significantly with a slip of 4-10% F1. By reviewing the released test set with gold labels, we found that a large number of new acronyms and code series are introduced. This causes a systematic deterioration of both classifiers, but NN presents a higher robustness to the feature surprise.

The unexpected performance of RFC in the English test set can be attributed to an over-fitting problem and hence draws our attention to look for a semi-supervised or un-supervised mechanism in complementing the feature mismatch between the validation set and the test set, which shall lead to a more stable result in similar tasks.

4.2 Bilingual Training

This section focuses on the comparison of the classification performance of bilingual vs. monolingual training with the same experimental setting. A fused feature set and the RFC classifier is adopted. As what we submitted for the contest are all based on bilingual training, the current comparison is based on the validation (Dev) set only. The classification results on the validation set of the two languages are shown in Figure 2 below:

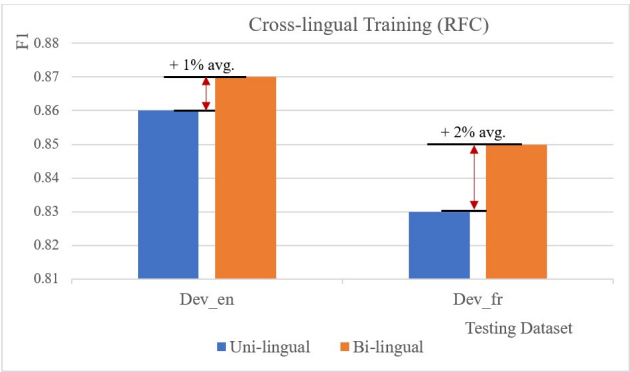

Figure 2: Classification Results of bi- vs. mono-lingual training

As shown in Figure 2, bilingual training brings a consistent benefit to the classification performance with 1%

F1 improvement for English and 2% F1 improvement for French. This impact can be counted as huge for any kinds of competition. By referring to this set of validation results, we finally conduct bilingual training for all the runs of submission.

4.3 Features

This section focuses on the comparison of the feature discriminativeness in the SBD classification task by adding the individual feature set separately in each implementation. The basic experimental setting is the same, including using the RFC classifier and bilingual training. As the submissions are all based on the full feature set, the current comparison is implemented on the validation (Dev) set only. The classification results on the validation set of English are shown in Table 1 below:

Features\F1	BS[a]	ES[b]	Mean	δ(%)[c]
Punc1	0.70	0.82	0.76	baseline
+Punc2	0.71	0.83	0.77	1 ↑
+Cap	0.72	0.86	0.79	2 ↑
+Acro	0.73	0.87	0.80	1 ↑
+Dig	0.73	0.89	0.81	1 ↑
+Lett	0.74	0.90	0.82	1 ↑
+POS	0.78	0.92	0.85	3 ↑
+Enter/All	0.80	0.92	0.86	1 ↑

[a] Beginning boundaries
[b] Ending boundaries
[c] F1 improvement in percentage

Table 1: Performance of feature mining in the English Dev set with RFC

In Table 1, we can see that by using the period punctuation as the baseline feature set, the classification performance is already decent, with 0.76 F1-Mean. And the 'ES' prediction is apparently more accurate than the 'BS' prediction, which is intuitively reasonable as a period usually marks an end of a sentence.

By adding the other feature set one by one, as mentioned in Section 3.1, the performance consistently increases with 1-3% F1 improvement. Some features help on both 'BS' and 'ES', such as 'Punc2', 'Cap', 'Acro', 'Lett', 'POS'; some help only on 'BS', such as 'Enter'; and some help only on 'ES', such as 'Dig'. Among all the feature sets, POS shows the greatest improvement to the identification of both 'BS' and 'ES', which implies the usefulness of incorporating certain syntactic information into sentence detection.

4.4 Keyword Validation

This section focuses on the comparison of post-classification validation vs. non-validation of keyword knowledge to demonstrate the effectiveness of knowledge enhancement. The basic experimental setting is the same, including the fused feature set, the RFC classifier and bilingual training. As we submitted 2 runs of both languages with the RFC method for the contest, the current comparison is based on both the validation (Dev) set and the test set. The corresponding classification results are shown in Table 2 below.

	F1	NO[a]	YES[b]	δ(%)
Dev	BS	0.83	0.83	0
	ES	0.86	0.87	1 ↑
	Mean	0.845	0.85	0.5 ↑
Test	BS	0.81	0.84	3 ↑
	ES	0.88	0.88	0
	Mean	0.845	0.86	1.5 ↑

[a] Without keyword validation
[b] With keyword validation

Table 2: Performance of RFC in the French Dev and Test sets in terms of keyword validation

Informative knowledge can be a very useful guidance to correcting the confused predictions of statistical models, as evidenced in this experiment. We implemented the keyword extraction and validation procedure, as shown in Section 3.3, to post-process the predictions of the RFC model with the aim of rectifying certain wrong labels caused by confusion of the out-of-boundary keywords.

The results in Table 2 indicate that keyword validation is indeed successful for both validation and testing. Notably, the improvement of predicting 'BS' is significant (3%↑), which leads to a 1.5% F1-Mean gain for our system in the final contest, and this result is comparable to the top teams. The success of keyword validation in our experiment suggests that by using adequate domain knowledge in NLP tasks, we could optimize the classification performance in an efficient way. Moreover, the domain knowledge itself serves a valuable resource for text processing and information extraction of the specific domain.

5 Knowledge Adaptation to the Test Sets and the Final Results

This section aims to fill in the gap of our mistake in missing the implementation of the knowledge enhancement procedure on the test sets, which causes an unexpected low result of the English trial for RFC.

In order to remedy the above mistake, we simply run the script of the same procedure in Section 3.3 by including the keywords of test sets in the knowledge dictionary so as to cover the additional domain specific words that are not included/recoverable in the training data[5]. The corresponding results and ranks of our system are shown in Table 3 below.

As Table 3 shows, the results of RFC for the French trial are consistent among the three types of validation and our team achieves a stable rank (No. 8) in the competition. However, for the English trial,

[5]https://github.com/ClaraWan629/FinSBD_RFC_r1

Test Set	F1-Mean	Rank
Dev_en_rfc1	0.875	—
Test_en_rfc1	0.78*	16◇
Test_en_rfc1_adapted	**0.835**⋆	**10**⋇
Dev_fr_rfc1	0.85	—
Test_fr_rfc1	0.86*	8◇
Test_fr_rfc1_adapted	0.86⋆	8⋇

* Result without adaptation to the test set
◇ Rank without adaptation to the test set
⋆ Result with adaptation to the test set
⋇ Rank with adaptation to the test set

Table 3: Performance of RFC *w.r.t.* knowledge adaptation

there is a 9.5% F1-Mean gap between Dev_en_rfc1 and Test_en_rfc1, which is surprisingly different from our estimation. As mentioned above, we conduct a keyword adaption step on the test set and obtained a more reasonable result of the English trial with a rank of 10, as highlighted in Table 3. This adaptation step necessarily proves that our method of feature engineering and knowledge enhancement is effective and robust and it is important to resolve the over-fitting problem by applying it to the test sets.

6 Conclusion

In this work, we demonstrate the efficiency and robustness of combining feature engineering, bilingual training and knowledge-enriched approaches to the detection of sentence boundaries for noisy data in Financial NLP. We first conduct document introspection and error analysis in mining salient features for fitting the models and the final features include a 24-dimension array of punctuation, digital number, capitalization, acronym, letters, Enter, and POS tags. We then tune the classifiers on parameters of min_samples_split and max_features for RFC and batch size, epochs for NN with optimized performance on Dev sets. We also implement rule-based validation of keyword knowledge extracted from the out-of-vocabulary word sequences in FinSBD's datasets. Lastly, we train on both English and French datasets to make predictions with maximal training data. The results of the four aspects of comparisons suggest the following findings: 1) NN does not showing significant advantage over traditional classifiers as RFC is better fitted to the selected feature set in this work; 2) NN performs better in terms of new features not originally selected; 3) The significant improvement of using POS information of a three-dimension sequence in the task shows that syntactic information may be helpful for sentence detection; 4) Informative knowledge enhancement shows a double benefit for both the correction of misclassification of statistical models and resource construction in domain-specific NLP tasks. It is important to note that our unexpectedly lower result of RFC for the English trial is found to be caused by the mistake of not implementing knowledge enhancement on the test set. After we conduct a knowledge adaptation to the test set, the outcome achieved is close to our estimation (English: 0.835%; French: 0.86%) and within reasonable range of the best results. Although our result is not currently the best, our system is designed to be highly adaptive with minimal training data for a new language and/or a novel domain. We hope to conduct additional studies to verify the effectiveness of this feature design.

Acknowledgments

This work is partially supported by the fund of the GRF grant (RGC Ref No. 15608618).

References

[Ait Azzi *et al.*,] Abderrahim Ait Azzi, Houda Bouamor, and Sira Ferradans. The FinSBD-2019 Shared Task: Sentence boundary detection in PDF Noisy text in the Financial Domain.

[Breiman *et al.*, 1984] Breiman, Friedman, Olshen, and Stone. Classification and regression trees. *Wadsworth Int. Group*, 37(15):237–251, 1984.

[Breiman, 1999] L. Breiman. Random forests—random features. *Technical Report 567*, 1999.

[Chen *et al.*, 2009] Chen, Huang, Tian, and Qu. Feature selection for text classification with naïve bayes. *Expert Systems with Applications*, 36(3):5432–5435, 2009.

[Garla and Brandt, 2012] Vijay N. Garla and Cynthia Brandt. Ontology-guided feature engineering for clinical text classification. *Journal of biomedical informatics*, 45(5):992–998, 2012.

[Ghosh and NAG, 2002] Joydeep Ghosh and Arindam C. NAG. Knowledge enhancement and reuse with radial basis function networks. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290)*, pages 1322–1327. IEEE, 2002.

[Griffis *et al.*, 2016] Griffis, Shivade, Lussier E. Fosler, and A. M. Lai. A quantitative and qualitative evaluation of sentence boundary detection for the clinical domain. In *AMIA Summits on Translational Science Proceedings*, page 88, 2016.

[Hearst, 1994] Marti A Hearst. Multi-paragraph segmentation of expository text. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 9–16. Association for Computational Linguistics, 1994.

[Hou *et al.*, 2019] Renkui Hou, Chu-Ren Huang, and Hongchao Liu. A study on chinese register characteristics based on regression analysis and text clustering. *Corpus Linguistics and Linguistic Theory*, 15(1):1–37, 2019.

[Huang and Chen, 2017] Chu-Ren Huang and Keh-Jiann Chen. Sinica treebank. In *Nancy Ide and James*

Pustejovsky (Eds.), Handbook of Linguistic Annotation, pages 641–657. Dordrecht: Springer, 2017.

[Kiss and Strunk, 2006] Tiber Kiss and Jan Strunk. Unsupervised multilingual sentence boundary detection. *Computational linguistics*, 32(4):485–525, 2006.

[Lee and Lee, 2006] Changki Lee and Gary G. Lee. Information gain and divergence-based feature selection for machine learning-based text categorization. *Information processing & management*, 42(1):155–165, 2006.

[McCulloch and Pitts, 1943] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[Mikheev, 2000] Andrei Mikheev. Tagging sentence boundaries. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, pages 264–271. Association for Computational Linguistics, 2000.

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

[Palmer and Hearst, 1997] David D. Palmer and Marti A. Hearst. Adaptive multilingual sentence boundary disambiguation. *Computational linguistics*, 23(2):241–267, April–June 1997.

[Read, 2012] Jonathon Read. Sentence boundary detection: A long solved problem? In *Proceedings of COLING 2012: Posters*, pages 985–994, Mumbai, India, 2012. Association for Computational Linguistics.

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

[Riley, 1989] Michael D. Riley. Some applications of tree-based modelling to speech and language. In *Proceedings of DARPA ,5'peech and Language Technology Workshop*, pages 339–352, Cape Cod, Massachusetts, 1989.