GOAT at the FinSim-2 task: Learning Word Representations of Financial Data with Customized Corpus

Yulong Pei*
Eindhoven University of Technology
Eindhoven, the Netherlands
y.pei.1@tue.nl

Qian Zhang*
Rogers Communications
Toronto, Canada
zhangqiandut@gmail.com

ABSTRACT

In this paper, we present our approaches for the FinSim 2021 Shared Task on Learning Semantic Similarities for the Financial Domain. The aim of the FinSim shared task is to automatically classify a given list of terms from the financial domain into the most relevant hypernym (or top-level) concept in an external ontology. Two different word representations have been compared in our study, i.e., customized word2vec provided by the shared task and FinBERT. We first create a customized corpus from the given prospectuses and relevant articles from Investopedia. Then we train the domain-specific word2vec embeddings using the customized data with customized word2vec and FinBERT as the initialized embeddings respectively. Our experimental results demonstrate that these customized word embeddings can effectively improve the classification performance and achieve better results than the direct utilization of the provided word embeddings. The class imbalance issue of the given data is also explored. We empirically study the classification performance by employing several different strategies for imbalanced classification problems. Our system ranks 2^{nd} on both Average Accuracy and Mean Rank metrics.

CCS CONCEPTS

• **Information systems → Clustering and classification**; • **Computing methodologies → Natural language processing**.

KEYWORDS

Word representations, BERT, word2vec, imbalance classification

ACM Reference Format:
Yulong Pei and Qian Zhang. 2021. GOAT at the FinSim-2 task: Learning Word Representations of Financial Data with Customized Corpus. In *Companion Proceedings of the Web Conference 2021 (WWW '21 Companion), April 19–23, 2021, Ljubljana, Slovenia.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3442442.3451385

1 INTRODUCTION

Hypernymy, i.e. the capability to relate generic terms or classes to their specific instances, lies at the core of human cognition [3] and hypernymy modeling has been widely studied in natural language processing (NLP) for decades. Although recent studies have shown promising results, especially methods based on embeddings [12, 13], hypernymy modeling for general corpus may not work well in specific areas such as financial domain because there are many abbreviations and polysemy (e.g., ETF and Option) that are difficult to classify without context. FinSim 2020 shared task [6] was the first hypernymy categorization task for the financial domain to fill this gap. FinSim 2021 shared task continues to focus on this problem by providing an enriched dataset in terms of volume and quality.

In this paper, we present our approaches for the FinSim 2021 shared task on Task on Learning Semantic Similarities for the Financial Domain [9]. This task aims to automatically classify given financial terms into the most relevant hypernym concept in an external ontology. Most methods submitted in FinSim 2020 view a hyponym-hypernym pair as a *is-a* relation and model this problem as a classification problem. Considering the effectiveness of previous studies using classification model and the power of word embeddings[1] in capturing the semantics of text, we follow previous study to focus on learning domain-specific representations and exploring good classifiers.

In specific, we exploit two different word representations, i.e., customized word2vec provided by the shared task and FinBERT [1]. We first create customized corpus from the given prospectuses and relevant articles from Investopedia[2]. Then we train the domain-specific word2vec embeddings using the customized data with different initialization strategies: (1) initialized word embedding using customized word2vec and (2) initialized word embedding using FinBERT. Different classifiers have also been explored and compared empirically. Our experimental results demonstrate that these customized word embeddings can effectively improve the classification performance and better than the direct utilization of the provided word embeddings. Besides, we explore the class imbalance issue of the given data. We study the classification performance by employing several different strategies for imbalanced classification problems.

The rest of this paper is organized as follows. Section 2 introduces the technical details of our proposed approaches. Section 3 empirically evaluates the performances of our methods. Section 4 briefly discusses the related works in hypernymy research. Finally we conclude in Section 5.

2 PROPOSED APPROACHES

We make use of customized corpus and exploit different word embeddings in word representation learning. Several classifiers are

*Both authors contributed equally to this work.

[1]We use the terms embedding and representation interchangeably.
[2]https://www.investopedia.com/

WWW '21 Companion, April 19–23, 2021, Ljubljana, Slovenia

ACM ISBN 978-1-4503-8313-4/21/04.
https://doi.org/10.1145/3442442.3451385

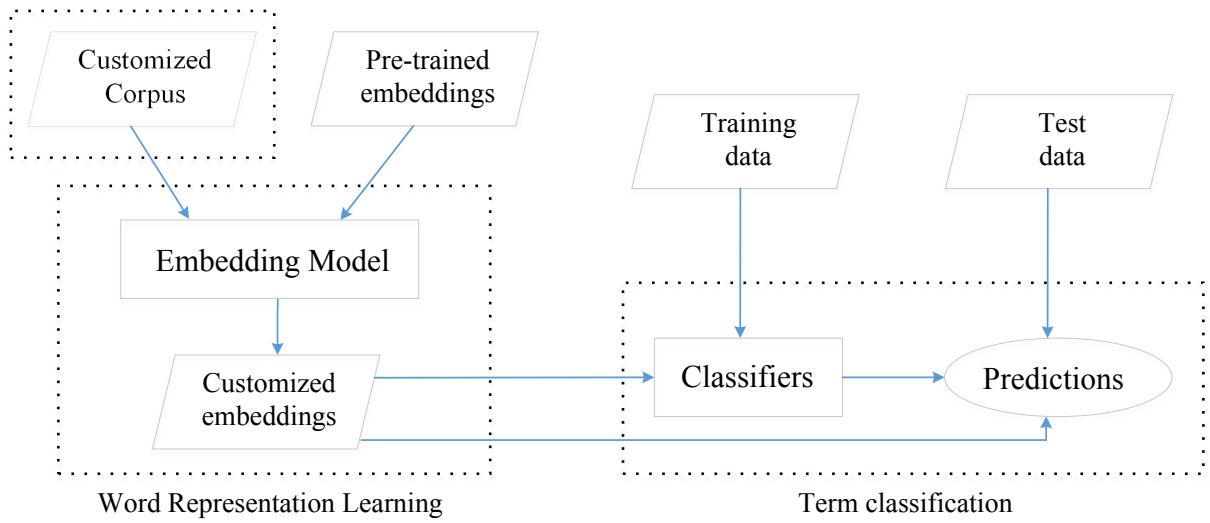

Figure 1: Framework of our proposed approach.

tested for term classification in our approaches. The general framework is shown in Figure 1. This framework consists of customized corpus collection, word representation learning and term classification. Each component will be discussed in details below.

2.1 Customized Corpus Collection

General word embeddings are trained on domain-independent corpus such as Wikipedia. However, different domains contain specific semantics. Therefore, to learn domain-specific representations for financial data, we collect customized corpus in our work.

In practice, we collect corpus from two sources. The first one is a set of English prospectuses provided by FinSim task organizers. The corpus extracted from the prospectuses contains 203 documents and the size is estimated to about 10 million tokens. The second data is from Investopedia. Specifically, we enrich the customized corpus using Investopedia definitions and topics. Using the predefined tags (`Bonds`, `Forward`, `Funds`, `Future`, `MMIs`, `Option`, `Stocks`, `Swap`, `Equity Index`, `Credit Index`) as keywords, we search for related articles from Investopedia. In the returned articles and explanations matched the keywords, we select 1,403 articles. Note that for some query, e.g., MMIs, it returns empty results. After cleaning the raw text, we obtain about 0.2 million tokens. These customized tokens are used to train the domain-specific word representations.

2.2 Word Representation Learning

In this component, we utilize two widely used word representations, i.e., word2vec [10] and BERT [5], to capture the semantic and syntactic properties of terms. Thanks to researchers in NLP for financial data, customized word2vec is provided by the FinSim task organizers and FinBERT [1] is proposed to apply BERT in financial domain. Thus, we directly use the customized word2vec and FinBERT for word representations.

2.2.1 word2vec. Word2vec [10] is a dense low-dimensional representation of a word and it can capture semantic and syntactic properties of words. Since word2vec is not a very deep model, one can easily fine-tune it on customized data. In this task, two versions of customized word2vec have been provided with dimension of 100 and 300 respectively. In this work, we fine-tune word2vec using the 300 dimension version on the customized corpus introduced in Section 2.1. `word2vec-100` and `word2vec-300` are used to denote the given embeddings respectively. `word2vec-c` denotes our fine-tuned embeddings trained on the customized corpus with the given `word2vec-300` as the initialized embeddings.

2.2.2 BERT. Similar to word2vec, BERT can also be fine-tuned on the financial data to learn better representations for financial domain. However, due to the limited computational resources, we leave this direction as the future work. Instead, we select to use FinBERT [1] as the domain-specific word representations for this study. FinBERT is a pre-trained language model to analyze sentiment of text in financial domain. It further trains the BERT model in the finance domain by using financial corpus including a subset of Reuters TRC2 dataset[3] and Financial PhraseBank [8]. For simplicity, we use the version of FinBERT provided by Python Project Index [4].

Although it is difficult to fine-tune BERT or FinBERT, we propose a combination strategy which can make use of customized corpus and BERT/FinBERT. In specific, we use FinBERT to initialize word embeddings and fine-tune word2vec on the customized corpus. Two different dimensions of word embeddings have been exploited: (1) same dimension to original BERT. i.e., 768, denoted as `FinBERT-768` and (2) compressed dimension, i.e., 300, denoted as `FinBERT-300`.

2.3 Classification Methods

Word representations are used as features to train classifiers for term classification. A term is represented by the sum of word embeddings of each word contained in the term. To find the best classifiers, we test several widely used classification methods including *Logistic*

[3]https://trec.nist.gov/data/reuters/reuters.html
[4]https://pypi.org/project/finbert-embedding/

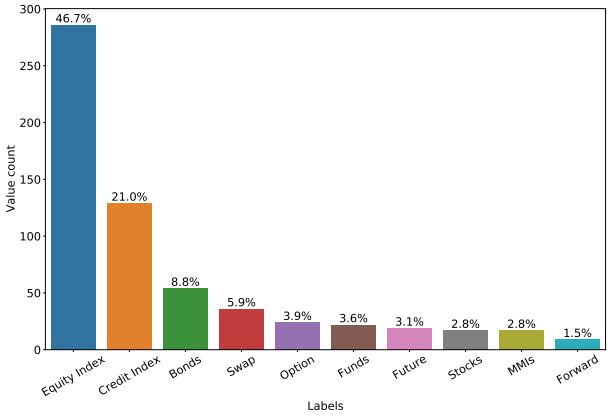

Figure 2: Numbers of different labels in the training data.

Regression, *linear SVM*, *Decision Tree*, *Random Forest*, and *AdaBoost*. Experimental studies will be discussed in Section 3.

2.4 Data Imbalance

Another issue in this task is the imbalanced distribution of different tags. As shown in Figure 2, more than 50% terms belong to categories `Equity Index` and `Credit Index`. However, only 1.5% terms, i.e., 9 terms, are in the `Forward` category. Classification of imbalance data has been widely studied in machine learning and data mining [11]. Thus, we refer to the imbalanced classification problem and employ some representative approaches to tackle this issue. Considering the small number of minor classes, it is more suitable to use over sampling strategies. Specifically, we empirically study the following over sampling methods:

- *Random Over Sampling*. It over-samples the minority class by picking samples randomly with replacement.
- *SMOTE (Synthetic Minority Oversampling TEchnique)* [4]. It is an over-sampling approach in which the minority class is over-sampled by creating "synthetic" examples.
- *ADASYN (Oversample using Adaptive Synthetic)* [7]. It uses a weighted distribution for different minority class examples according to their level of difficulty in learning.

Experimental studies will be introduced in Section 3.

3 EXPERIMENTS

3.1 Data Description

The FinSim shared task has a total of 613 entries with 2 column data: `terms` and `label`. There are in total 10 hypernyms (originally 11, merged `Swaps` and `Swap` into single one). For the test data, there are 212 terms to be classified into the correct hypernym.

Labels such as `Equity Index` and `Credit Index` are self explanatory, but there are also abbreviations and polysemy (e.g., `MMIs` and `Future`) that are difficult to classify without context. To tackle this issue, we use external corpus to learn domain-specific representations. We enrich the training data using Investopedia definitions and topics containing the 10 labels, along with the 10 million tokens extracted from prospectuses provided by FinSim organizers.

3.2 Results and Analysis

According to the proposed approaches introduced in Section 2, we have two provided embeddings, i.e., `word2vec-100` and `word2vec-300` and three new embeddings trained on the customized corpus, i.e., `word2vec-c`, `FinBERT-300` and `FinBERT-768`. We test different classifiers on these embeddings and the results are shown in Table 1. Note that each result is the average of 5 runs and the training/test ratio is 50%/50%. From the experimental studies, we find that complicated classifiers, e.g., Random Forest and AdaBoost, achieve worse performance than linear classifiers, so we select Logistic Regression as the classifier in our submitted systems. This observation is consistent with findings in the FinSim 2020 shared Task, that models learn linear boundaries perform better for this task [6]. Another conclusion is that `FinBERT-300` using word2vec to compress the dimension of FinBERT to 300 achieves the best performance.This conclusion is consistent to the evaluation results on test data, which will be discussed in Section 3.3.

We also explore different imbalanced classification methods. The results are reported in Table 2. Intuitively, using imbalanced classifiers can reduce the effect of imbalanced distribution of data. However, empirical studies show that these methods cannot improve the classification performance. A possible reason is that although for some tags there are only limited number of terms, these terms contain strong indicators/patters to be distinguished from other tags. For instance, 8 out of 9 terms belonging to category `Forward` contain the word *forward*. Therefore, in our submitted systems, imbalanced classification strategies have not been utilized.

3.3 Submitted Systems

In our submitted results, we collect customized corpus from provided prospectuses and articles from Investopedia. Logistic regression is used in all three submissions as the classifier. The differences are how to initialize word embeddings and the dimensions of representations. The final results are reported in Table 3.

3.3.1 GOAT_1. In this submission, we use FinBERT to initialize the word embedding and train a word2vec model. The dimension of representations stays the same to FinBERT, i.e., 768.

3.3.2 GOAT_2. In this submission, we use customized word2vec provided by the shared task to initialize the word embedding and train a word2vec model. The dimension of representations stays the same to the initialization, i.e., 300.

3.3.3 GOAT_3. In this submission, we use FinBERT to initialize the word embedding and train a word2vec model. The dimension of representations is set to be 300.

Among these three submissions, the third one, GOAT_3, performs best and this submission ranks the 2^{nd} on both Average Accuracy and Mean Rank metrics.

4 RELATED WORK

Hypernymy, i.e. the capability to relate generic terms or classes to their specific instances, lies at the core of human cognition [3]. Therefore, hypernymy modeling has been widely studied in NLP for decades. These methods can be categorized into pattern-based, distributional, supervised classification based and projection-based

Table 1: Results of different word representations and classification models on the training data. ACC is short for Accuracy and MR is short for Mean Rank.

	word2vec-100		word2vec-300		word2vec-c		FinBERT-300		FinBERT-768	
Classifier	ACC	MR	ACC	MR	ACC	MR	ACC	MR	ACC	MR
Logistic Regression	0.859	1.248	0.890	1.193	0.882	1.225	0.928	1.131	0.895	1.189
SVM (Linear)	0.884	1.196	0.875	1.233	0.775	1.422	0.869	1.294	0.817	1.369
Decision Tree	0.695	1.812	0.692	1.824	0.627	1.997	0.725	1.758	0.716	1.752
Random Forest	0.757	1.517	0.765	1.486	0.716	1.644	0.824	1.356	0.752	1.595
AdaBoost	0.486	2.177	0.573	1.980	0.520	2.026	0.611	1.892	0.546	2.078

Table 2: Results of different imbalanced classification methods on the training data.

	word2vec-c		FinBERT-300		FinBERT-768	
Strategy	ACC	MR	ACC	MR	ACC	MR
Standard	0.882	1.225	0.928	1.131	0.895	1.189
Random	0.873	1.225	0.908	1.147	0.859	1.304
SMOTE	0.879	1.261	0.889	1.212	0.886	1.232
ADASYN	0.863	1.268	0.879	1.186	0.876	1.252

Table 3: Results of submissions on the training and test data.

	Training Data		Test Data	
Method	ACC	MR	ACC	MR
GOAT_1	0.895	1.189	0.887	1.198
GOAT_2	0.882	1.225	0.868	1.330
GOAT_3	0.928	1.131	0.896	1.193

approaches [12]. Most recent studies explore embedding methods for hypernymy modeling [12, 13].

There are some shared tasks for the problem of hypernymy modeling. For instance, SemEval provides a series of tasks for hypernym modeling including Taxonomy Extraction Evaluation (TExEval) [2] which aims to find hypernym-hyponym relations between given terms and Hypernym Discovery [3] which aims to retrieve (or discover) its suitable hypernyms from a target corpus given an input term. Among these tasks, the FinSim 2020 shared task [6] is the first hypernymy categorization task for financial domain.

5 CONCLUSIONS

In this paper, we investigated the problem for the FinSim 2021 Shared Task on Learning Semantic Similarities for the Financial Domain. We utilized word2vec and FinBERT embeddings to capture the semantic representations for text in financial domain. We collected external corpus from Investopedia to enrich the customized data for a better representation learning. We also explored different classifiers and imbalance classification methods for this task. From the experimental studies, some conclusions can be drawn: (1) BERT/FinBERT is more effective in capturing semantics; (2) linear classifiers are better choice in relatively small-scale data; and (3) imbalance classification methods are not effective in this task. For future research, more advanced classification methods such as deep learning based classifiers can be investigated. Other data resources can be exploited for data augmentation such as financial articles in Google news and financial terms in Wikipedia.

REFERENCES

[1] Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063* (2019).

[2] Georgeta Bordea, Els Lefever, and Paul Buitelaar. 2016. Semeval-2016 task 13: Taxonomy extraction evaluation (texeval-2). In *Proceedings of the 10th international workshop on semantic evaluation (semeval-2016)*. 1081–1091.

[3] Jose Camacho-Collados, Claudio Delli Bovi, Luis Espinosa-Anke, Sergio Oramas, Tommaso Pasini, Enrico Santus, Vered Shwartz, Roberto Navigli, and Horacio Saggion. 2018. SemEval-2018 task 9: Hypernym discovery. In *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018); 2018 Jun 5-6; New Orleans, LA. Stroudsburg (PA): ACL; 2018. p. 712–24.* ACL (Association for Computational Linguistics).

[4] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16 (2002), 321–357.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 4171–4186.

[6] Ismail El Maarouf, Youness Mansar, Virginie Mouilleron, and Dialekti Valsamou-Stanislawski. 2021. The finsim 2020 shared task: Learning semantic representations for the financial domain. In *Proceedings of the Second Workshop on Financial Technology and Natural Language Processing*. 81–86.

[7] Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. 2008. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*. IEEE, 1322–1328.

[8] Pekka Malo, Ankur Sinha, Pekka Korhonen, Jyrki Wallenius, and Pyry Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology* 65, 4 (2014), 782–796.

[9] Youness Mansar, Juyeon Kang, and Ismail El Maarouf. 2021. FinSim-2 2021 Shared Task: Learning Semantic Similarities for the Financial Domain. In *Proceedings of The Web Conference 2021 (Virtual Edition)*.

[10] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[11] Yanmin Sun, Andrew KC Wong, and Mohamed S Kamel. 2009. Classification of imbalanced data: A review. *International journal of pattern recognition and artificial intelligence* 23, 04 (2009), 687–719.

[12] Chengyu Wang and Xiaofeng He. 2020. Birre: learning bidirectional residual relation embeddings for supervised hypernymy detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 3630–3640.

[13] Zheng Yu, Haixun Wang, Xuemin Lin, and Min Wang. 2015. Learning term embeddings for hypernymy identification. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.