L3i_LBPAM at the FinSim-2 task: Learning Financial Semantic Similarities with Siamese Transformers

Nhu Khoa Nguyen
University of La Rochelle, L3i
La Rochelle, France
nhu.nguyen@univ-lr.fr

Emanuela Boros
University of La Rochelle, L3i
La Rochelle, France
emanuela.boros@univ-lr.fr

Gaël Lejeune
Sorbonne University
Paris, France
gael.lejeune@sorbonne-universite.fr

Antoine Doucet
University of La Rochelle, L3i
La Rochelle, France
antoine.doucet@univ-lr.fr

Thierry Delahaut
La Banque Postale Asset Management
Paris, France
thierry.delahaut@labanquepostale-am.fr

ABSTRACT

In this paper, we present the different methods proposed for the FinSIM-2 Shared Task 2021 on *Learning Semantic Similarities for the Financial domain*. The main focus of this task is to evaluate the classification of financial terms into corresponding top-level concepts (also known as hypernyms) that were extracted from an external ontology. We approached the task as a semantic textual similarity problem. By relying on a siamese network with pre-trained language model encoders, we derived semantically meaningful term embeddings and computed similarity scores between them in a ranked manner. Additionally, we exhibit the results of different baselines in which the task is tackled as a multi-class classification problem. The proposed methods outperformed our baselines and proved the robustness of the models based on textual similarity siamese network.

CCS CONCEPTS

• **Computing methodologies** → **Lexical semantics**; **Neural networks**.

KEYWORDS

Hypernym detection, siamese networks, semantic similarities

ACM Reference Format:
Nhu Khoa Nguyen, Emanuela Boros, Gaël Lejeune, Antoine Doucet, and Thierry Delahaut. 2021. L3i_LBPAM at the FinSim-2 task: Learning Financial Semantic Similarities with Siamese Transformers. In *Companion Proceedings of the Web Conference 2021 (WWW '21 Companion), April 19–23, 2021, Ljubljana, Slovenia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3442442.3451384

1 INTRODUCTION

While there exists a few predominant applications of natural language processing (NLP) regarding finance, for example analyzing sentiment of financial news or reports, many practices remain under-represented. One of these unexplored research topics is hypernym extraction. A hypernym indicates a word/concept that has the highest level of category abstraction and generally has hyponyms, which are terms describing a more specific concept in said category. The task of hypernym extraction generally deals with finding the hypernym-hyponym association that usually belongs to the kind of "is-a" relationship.

To the best of our knowledge, FinSim[1] is the first shared task that tackles hypernym extraction in the financial domain. Rather than using news or annual reports, the organizers targeted prospectus [13], a type of financial document that describes investment offering to the public, which is mandatory to file and submit to the Securities and Exchange Commission, according to Investopedia[2]. The shared task consists of a list of financial terms extracted from a set of prospectuses that need to be assigned to their corresponding top-level concepts. These hypernyms are known beforehand, hence the task can be considered as multi-class classification.

The remainder of this paper is organized as follows. In Section 2, we present and discuss a selection of works concerning hypernym extraction methods. Then, in Section 3, the dataset explored in this work is presented. The proposed model is detailed in Section 4 and the experiments are described in Section 5. We present and discuss the obtained results in Section 6. Finally, Section 7 concludes this paper and hints at future work.

2 RELATED WORK

According to a recent survey [12], hypernym extraction consists of two general approaches: pattern-based and distributional-based. Pattern-based methods are traditional approaches to this problem that attempt to find the pair of terms that satisfy certain patterns. For example, a method that analyzed the co-occurrences of words to discover hyponym-hypernym couples was proposed [5]. With distributional methods, they have been given more attention recently with the advances of word embeddings such as Word2Vec [7], and GloVE [9] (context-independent), or BERT [3] (context-dependent).

Word embeddings can capture different similarities between terms and their top-level concepts. For example, the SentEval[3]

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

ACM ISBN 978-1-4503-8313-4/21/04.
https://doi.org/10.1145/3442442.3451384

[1] https://sites.google.com/nlg.csie.ntu.edu.tw/finweb2021/shared-task-finsim-2
[2] https://www.investopedia.com/
[3] https://github.com/facebookresearch/SentEval

toolkit evaluated the semantic similarities between texts [2]. Another notable research designed a siamese network using BERT as an encoder in order to measure similarities between sentences [10].

In 2020, the FinSim shared task had a number of entries, most of which were distributional-based techniques. The winning team proposed a system using a context-free word embedding with the Naive Bayes classifier, a classical supervised machine learning method [6]. The use of the pre-trained word and sentence embeddings are also explored in [1]. The authors treat the task in an unsupervised manner with the use of cosine similarity. Furthermore, this approach tried to include additional data from the Financial Industry Business Ontology (FIBO) ontology. While most of the entries from last year used some form of embedding, nonetheless, a team proposed a method that utilized Linear SVM classifier using TF-IDF features extracted from external, supplemented data [4], thus treated this problem as a multiclass classification problem.

3 DATA

For training the models, we were provided with 600 terms along with their respective hypernyms/labels, which include ten top-level financial concepts: *Bonds, Forward, Funds, Future, MMIs, Option, Stocks*, and *Swap*.

Upon observation, we noticed several challenges. First, the training dataset is relatively small, with only 600 entries. While it is much larger than last year (100 terms for the training data), it is still not enough to apply neural-based approaches. As for the characteristic of the terms given in the training set, we realized that, occasionally, these terms contain the top-level concepts to which they belong to. Most of the time, terms usually do not have any hypernyms as part of them. However, there are terms that contain hypernyms of other classes while belonging to different top-level concepts. In some extreme cases, a term can have both hypernyms that it belongs to and hypernyms that are irrelevant. A number of 119 related hypernyms was observed, while the number of unrelated hypernyms was of 53, and 12 were common to both, in a total of 160 hypernyms.

Here are a few examples extracted from the dataset: "Corporate Bonds" contains the "Bonds" hypernym, which is also the category this phrase belongs to. With "Fixed Recovery Swap", while it has the "Swap" concept in it, this term is not a lower concept of said hypernym, but belongs to "Credit Index". Including the "Bond" and "Future" hypernyms, the term "Single Name Bond Future", however, has as top-level ontology term "Future".

4 METHODS

The architecture we proposed is based on a siamese neural network that contains two pre-trained BERT encoders proposed by [3] with the same configuration and the same parameters[4], as presented in Figure 4. This type of architecture allows updating the weights of both encoders such that the produced term embeddings are semantically meaningful and can be compared. We use cosine similarity for comparing the two-term representations, which is defined as follows:

$$\cos(\mathbf{t}, \mathbf{e}) = \frac{\mathbf{te}}{\|\mathbf{t}\|\|\mathbf{e}\|} = \frac{\sum_{i=1}^{n} \mathbf{t}_i \mathbf{e}_i}{\sqrt{\sum_{i=1}^{n} (\mathbf{t}_i)^2} \sqrt{\sum_{i=1}^{n} (\mathbf{e}_i)^2}} \quad (1)$$

where t is the vector representation of the term and e is the vector representation of the ontology term.

The model adds a pooling operation to the output of the BERT encoder using the output of the [CLS]-token and we computed the mean of all output vectors, as this strategy proved the best results in our preliminary experiments.

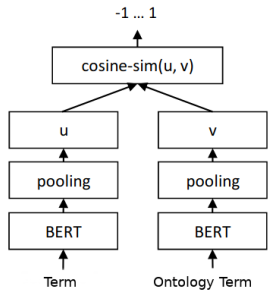

Figure 1: Architecture on calculating semantic similarities using siamese network with BERT encoders [10].

5 EXPERIMENTS

In this section, we describe in details the architecture of the baseline and the more sophisticated systems we employed.

5.1 Baseline

We investigated four commonly used text classification models as baselines: logistic regression, random forest, support vector machine and decision trees. For all models, we used the default parameters and a TF-IDF (term frequency-inverse document frequency) weighting measure[5]. We also consider as features two different embeddings: the GloVE [9] embeddings and the *FinSim* embeddings provided by the organizers that were trained on a set of financial articles.

5.2 Metrics

Regarding what metrics to evaluate the results, we followed the two criteria that FinSim organizers employed: *Mean Rank* and *Accuracy*. Accuracy implies the rate of correct prediction from the system compared to the groundtruth. On the other hand, the *Mean Rank* expresses how far off the correct label was in the prediction from the first rank. However, to ensure that this metric stays consistent, the organizers imposed a limit where if the correct label is not in the top-3 of the prediction, the rank of that prediction is automatically assigned as four regardless of how low the label is.

[4]The method is implemented and available at https://github.com/UKPLab/sentence-transformers

[5]https://scikit-learn.org/

5.3 Data Pre-processing

In order to decide which are the most performing systems, we split the provided annotated data into 80% for training and 10% for each development and test sets. For every top-level ontology term (10), the input entry is (term, ontology term, distance). 0 means that the term is irrelevant to the ontology term and 1 indicates a close relationship between the term and the hypernym, with a maximum distance of 1.
(1) *Credit Default Swap.*, Swap, 1
(2) *Credit Default Swap.*, Credit Index, 0
(3) *Credit Default Swap.*, Bonds, 0 . . .

5.4 Parameters

For our baseline models, we used the default parameters. For the pre-trained encoders, we experimented with several English ones (`bert-base`, `bert-large`, `bert-STS`[6] *cased* and *uncased* models). We trained for 150 epochs, with Adam optimizer with weight decay, 2×10^{-5} learning rate and a mini-batch of dimension 8.

6 RESULTS AND DISCUSSION

Table 1 describes the results of different methods on the test dataset that was split from the given 600 terms. We used different machine learning techniques as baselines: logistic regression, support vector machine, random forest and decision trees. Unigram TF-IDF along with two pre-trained embeddings, the provided FinSim embeddings and GloVE[7], were used. Among these baselines, logistic regression and SVM coupled with GloVE yielded the best results, where SVM performed better in the mean rank metric while logistic regression has better accuracy by a small margin.

In the results from Table 1 we remark that the chosen baseline models with TF-IDF weighting, GloVe and FinSim embeddings are competitive with the siamese-based models with pre-trained BERT encoders. We also notice that using FinSim pre-trained embeddings obtain in general lower performance scores than those with GloVe pre-trained embeddings, which might indicate that the articles on which the model was trained were not sufficient.

With our approach, siamese network coupled with multiple pre-trained BERT encoders, we were able to achieve superior results, in both metrics, compared to the best performing baselines. Between the BERT `cased` and `uncased` models, it is worth noticing that the `uncased` models perform better than the `cased` ones, which confirms that the character morphology is not important for this task, due to the fact that the capitalization is not connected to the presence of named entities or other capitalized terms. The model `bert-base-STS` (`cased` or `uncased`) did not perform as expected. We assume that the large corpus on which it was fine-tuned does not necessarily contribute to the financial domain, and thus it decreased the performance of the system. However, the difference is not statistically significant.

We also evaluated at which hypernyms our proposed methods failed to detect accurately. Table 2 illustrates the performance, using precision, recall and F-1 measure, of our best baseline (SVM using GloVE embeddings) and the siamese network with the best performing architecture, with the pre-trained `bert-base-uncased` for the encoders. Upon evaluating these metrics as well as the predictions made, we discovered that terms containing their respective hypernym get often miss-classified, with an imbalance between the precision and the recall. "MMIs", has exceptionally poor F1 score. We suspect that since this hypernym is an acronym, its representation might be unclear, hence could appear irrelevant.

To take a step further, we utilized the siamese-based systems with additional information about the definition of the hypernyms to add more informative features and to obtain a better distinction between them. The definition of each concept was added to the model. These definitions were extracted from the Financial Industry Business Ontology (FIBO). as shown in Figure 6.

They were concatenated with the ontology top-terms in the following manner:
(1) *Credit Default Swap.*, Swap + <hypernym definition in FIBO>, 1

Table 1: Experimental results for our chosen baseline models and proposed siamese-based methods.

Model	Mean Rank	Acc
Baseline Models		
– GloVE		
Logistic Regression+GloVE	1.322	0.844
Linear SVM+GloVE	1.306	0.841
Random Forest+GloVE	1.514	0.759
Decision Tree+GloVE	1.918	0.661
– FinSim embeddings		
Logistic Regression+FinSim	1.495	0.788
Linear SVM+FinSim	1.322	0.841
Random Forest+FinSim	1.527	0.743
Decision Tree+FinSim	1.951	0.657
– TF-IDF unigram		
Logistic Regression+TF-IDF	1.776	0.657
Random Forest+TF-IDF	1.469	0.743
Decision Tree+TF-IDF	1.743	0.735
Linear SVM+TF-IDF	1.453	0.8
BERT-based siamese networks		
`bert-base-uncased`	**1.2**	**0.894**
`bert-base-cased`	1.384	0.824
`bert-large-uncased`	1.241	0.886
`bert-large-cased`	1.331	0.829
`bert-base-uncased-STS`	1.220	0.882
`bert-base-cased-STS`	1.232	0.885
– definitions		
`bert-base-uncased+definitions`	1.387	0.816
`bert-base-cased+definitions`	1.363	0.832
`bert-large-uncased+definitions`	1.363	0.848
`bert-large-cased+definitions`	1.379	0.828
`bert-base-uncased-STS+definitions`	1.346	0.844
`bert-base-cased-STS+definitions`	1.359	0.840

[6]Fine-tuned on the STSbenchmark (semantic textual similarity benchmark) dataset
[7]We used the model pre-trained on Wikipedia 2014 and Gigaword 5 (vector size 300).

Table 2: Comparing baseline with proposed system using F1 measure

Model	Hypernym	Precision	Recall	F1
SVM+GloVE				
	Bonds	0.611	0.647	0.629
	Credit Index	0.797	0.895	0.843
	Equity Index	0.964	0.973	0.968
	Forward	**1.000**	0.500	0.667
	Funds	0.750	0.375	0.500
	Future	0.800	0.800	0.800
	MMIs	0.222	0.286	0.250
	Options	0.923	0.923	0.923
	Stocks	0.500	0.200	0.286
	Swap	0.812	0.765	0.788
`bert-base-uncased`				
	Bonds	0.560	0.824	0.667
	Credit Index	0.902	0.807	0.852
	Equity Index	0.948	**1.000**	0.973
	Forward	**1.000**	0.667	0.800
	Funds	0.741	0.625	0.667
	Future	**1.000**	**1.000**	**1.000**
	MMIs	0.333	0.143	0.200
	Options	**1.000**	**1.000**	**1.000**
	Stocks	**1.000**	0.400	0.571
	Swap	0.737	0.824	0.778

Figure 2: A definition from FIBO for the *Swap* ontology term.

(2) *Credit Default Swap.*, Credit Index + <hypernym definition in FIBO>, 0 . . .

However, the outcomes showed that adding more information will deteriorate the performance of all the siamese-based models. While it is expected that adding the definition to the label would increase the result by adding more informative features and context to the encoders, the experiment showed the opposite as both metric slightly decreased compared to having no added information. We suspect that the definition can cause noise which affects the encoding process.

To analyze the impact of the ontology terms that can be present in the terms (cf. Table 2), we propose to mark the common hypernym tokens in the term [8, 11] in order to uprise their relevance. We implemented our best performing siamese-based model with BERT encoders and *EntityMarkers* [11]. A BERT encoder with *EntityMarkers* consists in augmenting the input data with a series of special tokens, here named *TermMarkers*. Thus, if we consider a sentence $x = [x_0, x_1, \ldots, x_n]$ with n tokens, we augment x with two reserved word pieces (<Term> and </Term>) to mark the beginning and the end of each term in the sentence, as shown in the following example:
Swap $\subset$ *Credit Default Swap* $\implies$ (*Credit Default Swap*, Swap) $\rightarrow$ (*Credit Default <Term> Swap </Term>*, Swap)

Table 3: The results for the best performing system with and without marked entities.

Model	Average Rank	Acc
`bert-large-uncased`	1.241	0.886
`bert-large-uncased`+*TermMarkers*	1.404	0.779

From Table 3, we notice that if we give more importance to the common tokens in the term and the ontology top-level term (hypernym), the results are decreasing considerably which proves that by looking at the hypernym tokens presence in the term can only diminish the performance of the system.

Table 4: Results of our top-3 systems on the test set provided by the organizers. Median and Best (maximum accuracy and minimum mean rank) scores are computed on the submissions from each participant, as shared by FinSim organizers.

Model	Mean Rank	Acc
L3i-LBPAM_1	1.42	0.811
L3i-LBPAM_2	1,325	0.858
L3i-LBPAM_3	1,434	0.821
Median	1.285	0.858
Best	1.189	0.906

Table 4 shows the results of our top-3 systems on the final test set given by the organizers. Compared to our experiments, it can be clearly seen the mean rank metric from every system is lower than expected. As for accuracy, our best results only stands among the average compared to other teams, which might hint towards the difference in the test set and the provided training sets distributions.

7 CONCLUSIONS AND FUTURE WORK

In this paper, we attempt to solve the problem given in FinSim-2 the Shared Task on *Learning Semantic Similarities for the Financial Domain* by using a siamese network with BERT encoders to compute a similarity score between terms and hypernyms in a ranked manner. Our preliminary experimental results clearly outperformed the baseline, but only ranked around the median in the official scoring.

For future improvement, since the dataset was rather small, we plan to approach the task with few-shot learning. Moreover, due to the potential that this type of method could have, another aspect we need to work on is improving the model by focusing the attention mechanism on the contextual words in order to a better disambiguation between the ontology and the terms.

ACKNOWLEDGMENTS

This work has been supported by the European Union's Horizon 2020 research and innovation program under grants 770299 (NewsEye) and 825153 (Embeddia).

REFERENCES

[1] Vivek Anand, Yash Agrawal, Aarti Pol, and Vasudeva Varma. 2021. FINSIM20 at the FinSim Task: Making Sense of Text in Financial Domain. In *Proceedings of the Second Workshop on Financial Technology and Natural Language Processing*. -, Kyoto, Japan, 104–107.

[2] Alexis Conneau and Douwe Kiela. 2018. SentEval: An Evaluation Toolkit for Universal Sentence Representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA), Miyazaki, Japan. https://www.aclweb.org/anthology/L18-1269

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[4] 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*. -, Kyoto, Japan, 81–86.

[5] Gregory Grefenstette. 2015. INRIASAC: Simple Hypernym Extraction Methods. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*. Association for Computational Linguistics, Denver, Colorado, 911–914. https://doi.org/10.18653/v1/S15-2152

[6] Vishal Keswani, Sakshi Singh, and Ashutosh Modi. 2020. IITK at the FinSim Task: Hypernym Detection in Financial Domain via Context-Free and Contextualized Word Embeddings. (5 Jan. 2020), 87–92. https://www.aclweb.org/anthology/2020.finnlp-1.14

[7] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546* (2013).

[8] Jose G Moreno, Emanuela Boros, and Antoine Doucet. 2020. TLR at the NTCIR-15 FinNum-2 Task: Improving Text Classifiers for Numeral Attachment in Financial Social Data. In *Proceedings of the 15th NTCIR Conference on Evaluation of Information Access Technologies*. -, Tokyo Japan, 8–11.

[9] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, 1532–1543. http://www.aclweb.org/anthology/D14-1162

[10] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *CoRR* abs/1908.10084 (2019). arXiv:1908.10084 http://arxiv.org/abs/1908.10084

[11] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. *arXiv preprint arXiv:1906.03158* (2019).

[12] Chengyu Wang, Xiaofeng He, and Aoying Zhou. 2017. A Short Survey on Taxonomy Learning from Text Corpora: Issues, Resources and Recent Advances. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Copenhagen, Denmark, 1190–1203. https://doi.org/10.18653/v1/D17-1123

[13] Mansar Youness, Kang Juyeon, and El Maarouf Ismail. 2021. FinSim-2 2021 Shared Task: Learning Semantic Similarities for the Financial Domain. In *Proceedings of The Web Conference 2021, Virtual Edition, 2021*. Association for Computing Machinery, Ljubljana, Slovenia.