Multilingual Financial Word Embeddings for Arabic, English and French

Nadhem Zmandar
UCREL NLP Group
Lancaster University
UK
n.zmandar@lancaster.ac.uk

Mahmoud El-Haj
UCREL NLP Group
Lancaster University
UK
m.el-haj@lancaster.ac.uk

Paul Rayson
UCREL NLP Group
Lancaster University
UK
p.rayson@lancaster.ac.uk

***Abstract*—Natural Language Processing is increasingly being applied to analyse the text of many different types of financial documents. For many tasks, it has been shown that standard language models and tools need to be adapted to the financial domain in order to properly represent domain specific vocabulary, styles and meanings. Previous work has almost exclusively focused on English financial text, so in this paper we describe the creation of novel financial word embeddings for three languages: English, French and Arabic. In order to evaluate the effectiveness of the embeddings, we started by evaluating the English embeddings on a sentiment analysis classification task using the existing FinancialPhrase dataset and show improved performance over a standard GloVe based model using convolutional neural networks.**

***Index Terms*—Deep neural networks, sentence classification, financial sentiment analysis, word embedding.**

I. Introduction

Financial reporting requirements have been extended dramatically in recent years especially after the financial crisis in 2008. Financial communication and investor relation management are becoming crucial parts of the financial markets and the fund management industry. All listed companies in regulated markets are required by law to regularly communicate their financial activities to their stakeholders. They are required to publish their financial reports and several other financial narratives regularly. There are different ways in which firms can communicate with their stakeholders and we have different kinds of financial narratives (e.g. Annual Financial Reports, Preliminary Earning Announcements, Earning Announcements, Conference Calls, Corporate Social Responsibility Reports, Risk Management Reports, Audit Reports and IPO Prospectus) [1]. The choice of the reporting language and the format of reports is made by the financial regulator in a specific market. Most of the previously released language models are focused on English language. However, very little work has been carried out on other languages [2, 3, 4]. In this paper, we focus on two other major financial markets, France and Saudi Arabia, in order to begin addressing this imbalance. The market cap of listed French companies is around $2.898 Trillion (99 companies). The market cap of Saudi companies which reports in Arabic is around $2.234 Trillion.

This research is built on the hypothesis that using task oriented word embeddings in finance will improve the accuracy for financial text classification or sentiment analysis tasks. Liu et al. [5] proved in their paper that using a task oriented word embedding would improve the performance of the model. They developed a novel method to train task oriented word2vec models.

The main novel contributions of this research are the multilingual financial word embeddings themselves along with a proof that training financial word embeddings will improve results on financial NLP tasks.

II. Related work

With the increasing growth of the volume of financial disclosures in different languages and forms, financial NLP research is growing drastically and rapidly becoming a major research area. Kumar and Ravi [6] presented a survey of the applications of text mining in financial domain. Moreover, Zhao et al. [7] presented a Study on the Text Classification for Financial News Based on Partial Information.

There are two main options to obtain embedding vectors for a given corpus of documents:

1) Use pre-trained embeddings learned from a generic large corpus such as Wikipedia or Google News. There are several sources for pre-trained word embeddings. Popular options include Stanford's GloVe: Global vectors for word representation[1] [8] and SpaCy's built-in vectors.
2) Train a task oriented model using documents that reflect the domain of interest (e.g. finance, healthcare ... etc).

In fact, many tasks require embeddings of domain-specific vocabulary that models pre-trained on a generic corpus may not be able to capture. Standard word2vec models are not able to assign vectors to out-of-vocabulary words and instead use a default vector that reduces their predictive value. The less generic the content of the subsequent text modelling task, the more preferable the second approach. However, quality word embeddings are data hungry and require informative documents containing hundreds of millions of words.

Financial documents include words that appear in any general purpose pre-trained word embedding such as GloVe. However the usage of these words will be different and therefore the link in the vector space should be different

[1] https://github.com/stanfordnlp/GloVe

as well. The domain specific vocabulary used in financial disclosures is different from 'general' language. LOUGHRAN and MCDONALD [9] showed that the meaning of words can change substantially in a financial context. In fact, the context of a word tells you what type of words tend to occur near that specific word. The context is important in finance as this is what will give meaning to each word embedding.

For example, corporate earnings releases use nuanced language not fully reflected in GloVe vectors pre-trained on Wikipedia articles. Moreover, when working with industry-specific documents, the vocabulary or its usage may change over time as new technologies or products emerge. For all these reasons, working on training custom word embedding for financial domain would have an added value.

III. Methodology

To create word embeddings we always need to choose an embedding method. In order to build the financial word embeddings, we used the word2vec model introduced by Mikolov et al. [10]. **Word2vec** is developed by using two-layer neural networks. The choice of Word2Vec Model is justified by the fact that it is a powerful unsupervised word embedding technique. In fact, it is not a single algorithm, rather, it is a family of model architectures and optimizations. The usefulness of Word2vec is to group the vectors of similar words together in vector space. That is, it detects similarities mathematically. Word2vec creates vectors that are distributed numerical representations of word features such as the context of individual words.

To implement the word2vec model, we used the Gensim[2] library.

The two variants of word2vec model are:

- **CBOW**: The continuous-bag-of-words model predicts the target word using the average of the context word vectors as input so that their order does not matter. CBOW trains faster and tends to be slightly more accurate for frequent terms, but pays less attention to infrequent words.
- **SG**: The skip-gram model uses the target word to predict the words surrounding a given input word. It works well with small datasets and finds good representations even for rare words or phrases. The skip-gram model implicitly factorizes a word-context matrix that contains the pointwise mutual information of the respective word and context pairs [11].

A. Training Setup

One epoch of word embedding training takes approximately 10 minutes on a modern `4-core i7` processor and `40 Gb` RAM. The training speed can be significantly improved by using parallel training on multiple-CPU machine.

The main choices to make that impact the performance of the model are:

- Architecture: skip-gram (slower, better for infrequent words) vs CBOW (fast).
- The training algorithm: hierarchical softmax (better for infrequent words) vs negative sampling (better for frequent words, better with low dimensional vectors).
- alpha: The initial learning rate - (0.01, 0.05)
- Sub-sampling of frequent words: can improve both accuracy and speed for large data sets (useful values are in range 1e-3 to 1e-5).
- Dimensionality of the word vectors: Default value is 100. 300 is the dimension we recommended for this task.
- Context (window) size: for skip-gram usually around 10, for CBOW around 5.

The parameters we used to train word2vec model are shown in Table I:

sg	min_count	window	size	sample
1	3	2	300	6e-5
alpha	negative	workers	epochs	—
0.05	20	16	15	—

TABLE I
WORD2VEC PARAMETERS

B. Training process

In this paper, we train and evaluate domain-specific embeddings using financial annual reports from UK firms. We will first describe how we pre-processed the data for this task, then demonstrate how the skip-gram architecture outlined in the first section works, and finally visualize the results. We also will introduce alternative, faster training methods. Pre-processing typically involves phrase detection, that is, the identification of tokens that are commonly used together and should receive a single vector representation.

Figure 1 shows how word embeddings are trained starting from a corpus.

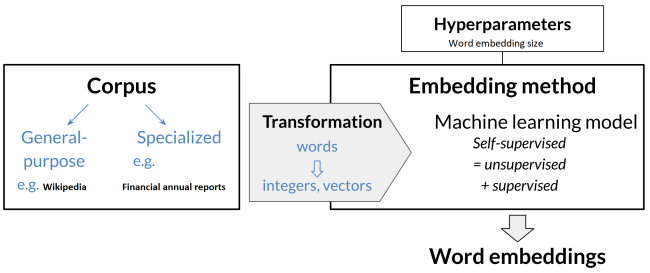

Fig. 1. word embedding process

We perform the pre-processing using NLTK[3] library. We deleted non alphanumeric values, and replaced some special characters by their equivalent (e.g. "m" is replaced with "million"). Moreover, we convert all words into lowercase. Finally, we extract tokenized sentences of the dataset using the NLTK tokenizer and created a vocabulary of the training dataset in the form of dictionary where keys are words and values are number of occurrence. The tokenized sentences

[2]https://radimrehurek.com/gensim/

[3]https://pypi.org/project/nltk/

were passed as input to the word2vec tool from Gensim library which produced the word vectors as output.

IV. Dataset

We used three different datasets to train our multilingual word embeddings.

- **FNS + Annual Reports Key Sections Corpora**[4]: FNS dataset is UK annual report dataset in English from the financial summarization shared task 2020 [12, 13, 14]. The dataset is composed of 3,000 UK firm annual reports. Annual Reports Key Sections Corpora is a Plain text content extracted from an initial sample of 31,464 annual reports published between January 2002 and December 2017 by firms listed on the London Stock Exchange (LSE) [4].
- **COFIF** [15]: A Corpus of Financial Reports in French Language[5]. It contains over 188 million tokens in 2,655 reports from French listed companies in the CAC40 (the French stock market index). An example of a 2D plot of French embedding is shown in Figure 2.
- **ABMC**: Arabic in Business and Management Corpora (ABMC) 2016[6]. It is composed of 1,200 Arabic articles as plain text and also tagged using Stanford Arabic Part of Speech Tagger.

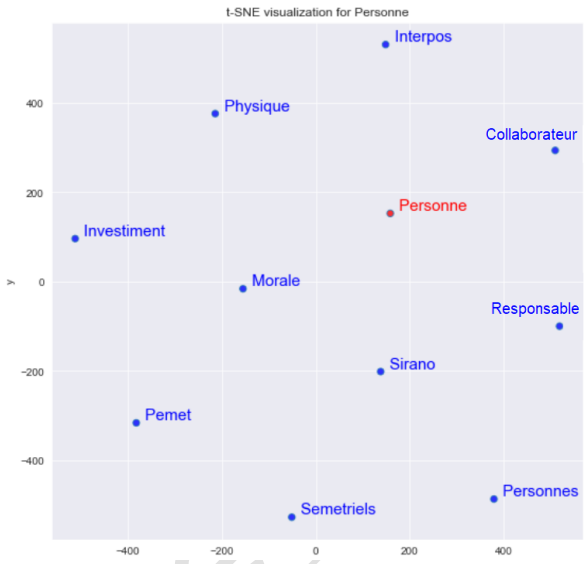

Fig. 2. T-SNE 2D plot for the French word “Personne”

V. Experiments

For our experiments, we used the Financial phrasebank dataset[7]. The dataset was collected by [16]. This release of the financial phrase bank covers a collection of 4,840 sentences. The selected collection of phrases was annotated manually by 16 people with adequate background knowledge on financial markets. Given the large number of overlapping annotations (five to eight annotations per sentence), there are several ways to define a majority vote based gold standard. The authors have formed four alternative reference datasets based on the strength of majority agreement. For our experiments, we used the sentences with more than 50 per cent agreement. To pre-process the classification dataset, we separated it into inputs and labels. The inputs are financial related sentences and the labels are sentiments (positive, neutral, negative). Then we encoded our labels as follows ‘positive’: 0, ‘neutral’:1, ‘negative’:2. We then split our dataset into training (80%) and testing (20%) and we ensured that our split respected a normal distribution of our labels. The length of the training and testing datasets is 3,876 and 970 respectively. We trained three different neural networks simultaneously: CNN, DNN and RNN on our training dataset using a dropout of 0.05, a batch size of 2 and sparse categorical value of 0.and we pass the financial English word embedding and the finance related sentences as input to our model. We used the Random Multimodel Deep Learning for Classification (RMDL)[8] library introduced by Kowsari et al. [17] to perform our experiments. It is a new ensemble, deep learning approach for classification. The architecture of the RMDL is detailed in Figure 3.

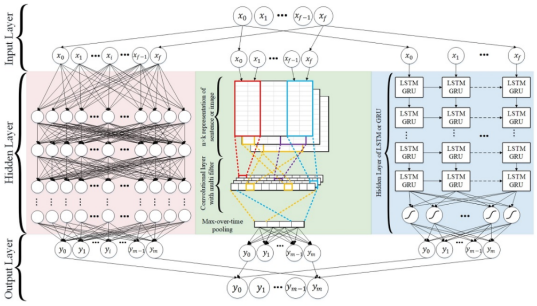

Fig. 3. Random Multimodel Deep Learning (RMDL) architecture for classification (see [17])

RMDL solves the problem of finding the best deep learning structure and architecture while simultaneously improving robustness and accuracy through ensembles of deep learning architectures. In short, RMDL trains multiple randomly generated models of Deep Neural Network (DNN), Convolutional Neural Network (CNN) as shown in Figure 4 and Recurrent Neural Network (RNN) in parallel. In our case we run three randomly generated models for every neural network.

We used the Adam optimizer which combines aspects of RMSProp with Momentum. It is considered fairly robust and often used as the default optimization algorithm [19]. Adam optimizer has several hyperparameters with recommended default values (`learning_rate=0.001, beta_1=0.9, beta_2=0.999, epsilon=1e-07`.)

VI. Results

Our results show that using a financial word embedding in a financial text sentiment classification task improves the result

[4]https://doi.org/10.17635/lancaster/researchdata/271
[5]https://github.com/CoFiF/Corpus
[6]https://sourceforge.net/projects/arabic-business-copora/
[7]https://huggingface.co/datasets/financial_phrasebank
[8]https://github.com/kk7nc/RMDL

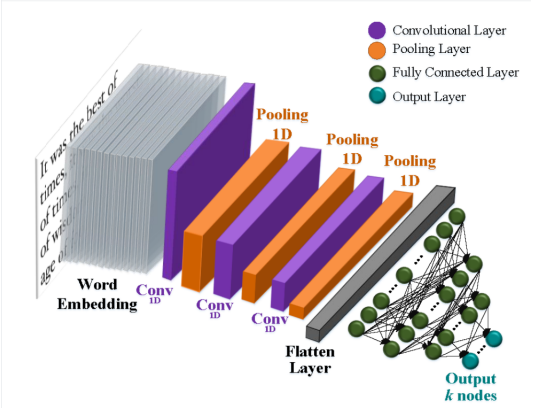

Fig. 4. Architecture of CNN max pooling and word embedding for text classification (see [18]).

using a convolutional neural network. We achieve comparable results using the deep neural network. However, recurrent neural networks performed better using the GloVe embeddings. Detailed results are available in Table II. We use accuracy and F1 weighted scores to evaluate our models.

The accuracy score is a practical evaluation that calculates a straightforward true or false value for each result. Either the model's outputs matches the correct predictions for a given subset samples i of a set of samples or not. Eqn. 1 shows the equation of the accuracy function:

$$Accuracy(y, \hat{y}) = (\frac{1}{n_samples}) \sum_{i=0}^{n_samples-1} 1(y = \hat{y}) \quad (1)$$

The F1 score introduces a more flexible approach that can help when faced with datasets containing uneven class distributions. F1 score uses weighted values of precision and recall. It is a weighted average of precision and recall values. Eqn. 2, 3, 4 show the equation of the accuracy function, precision and recall respectively.

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

Model	Accuracy	F1 Score
CNN + GloVe	0.696	0.669
CNN + financial WE	**0.722**	**0.675**
DNN + GloVe	0.771	0.765
DNN + financial WE	0.769	**0.766**
RNN + GloVe	0.789	0.788
RNN + financial WE	0.688	0.662

TABLE II
TEXT CLASSIFICATION RESULTS

VII. New frontiers

Although Pretrained Word2vec and GloVe embeddings capture more semantic information than the bag-of-words approach and allow a better results on different NLP tasks, they are unable to differentiate between context-specific usages. To address unsolved problems like polysemy, several models have emerged that build on the attention mechanism designed to learn more contextualized word embeddings. In December 2017, Vaswani et al [20] published their seminal paper, Attention Is All You Need, describing their work at Google Research and Google Brain, presenting the original Transformer model.

Since then, the use of bidirectional language models that process text both left-to-right and right-to-left for a richer context representation has emerged, and the use of semi-supervised pretraining on a large generic corpus to learn universal language aspects in the form of embedding that can be used for fine-tuning for specific tasks.

The paper [21] presented the FinBERT transformer which BERT model pre-trained on financial communication text. It is trained on a corpora of 4.9B tokens. There is also a fine-tuned FinBERT model for financial sentiment classification which is available on Huggingface's transformers library[9]. This model achieves superior performance on financial sentiment classification task.

VIII. Conclusion and Future Work

In this paper, we have reported on training financial word embeddings in three languages[10]. We also explored the task of financial sentiment classification using a financial language model and showed improved performance over a standard GloVe model. In addition to exploring transformer based models, future work will include collecting more financial corpora in English, French and Arabic in order to improve the performance of the WE models. Moreover, we aim to perform experiments on French and Arabic financial text classification datasets. The biggest limitation of such experiments would be the access to human annotated financial sentiment classification datasets in Arabic or French.

[9]https://huggingface.co/yiyanghkust/finbert-tone
[10]https://github.com/UCREL/multilingual-financial-word-embeddings.

References

[1] M. El-Haj, P. Rayson, M. Walker, S. Young, and V. Simaki, "In search of meaning: Lessons, resources and next steps for computational analysis of financial discourse," *Journal of Business Finance & Accounting*, vol. 46, no. 3-4, pp. 265–306, 2019.

[2] M. El Haj, P. E. Rayson, P. Alves, and S. E. Young, "Towards a multilingual financial narrative processing system," 2018.

[3] M. El-Haj, P. Rayson, P. Alves, C. Herrero-Zorita, and S. Young, "Multilingual financial narrative processing: Analyzing annual reports in english, spanish, and portuguese," in *Multilingual Text Analysis: Challenges, Models, And Approaches*. World Scientific, 2019, pp. 441–463.

[4] M. El-Haj, P. Alves, P. Rayson, M. Walker, and S. Young, "Retrieving, classifying and analysing narrative commentary in unstructured (glossy) annual reports published as pdf files," *Accounting and Business Research*, vol. 50, no. 1, pp. 6–34, 2020.

[5] Q. Liu, H. Huang, Y. Gao, X. Wei, Y. Tian, and L. Liu, "Task-oriented word embedding for text classification," in *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, Aug. 2018, pp. 2023–2032. [Online]. Available: https://aclanthology.org/C18-1172

[6] B. S. Kumar and V. Ravi, "A survey of the applications of text mining in financial domain," *Knowledge-Based Systems*, vol. 114, pp. 128–147, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705116303872

[7] W. Zhao, G. Zhang, G. Yuan, J. Liu, H. Shan, and S. Zhang, "The study on the text classification for financial news based on partial information," *IEEE Access*, vol. 8, pp. 100 426–100 437, 2020.

[8] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. [Online]. Available: https://aclanthology.org/D14-1162

[9] T. LOUGHRAN and B. MCDONALD, "When is a liability not a liability? textual analysis, dictionaries, and 10-ks," *The Journal of Finance*, vol. 66, no. 1, pp. 35–65, 2011. [Online]. Available: http://www.jstor.org/stable/29789771

[10] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[11] Y. Goldberg and O. Levy, "word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method," *CoRR*, vol. abs/1402.3722, 2014. [Online]. Available: http://arxiv.org/abs/1402.3722

[12] M. El-Haj, "Multiling 2019: Financial narrative summarisation," in *Proceedings of the Workshop MultiLing 2019: Summarization Across Languages, Genres and Sources*, 2019, pp. 6–10.

[13] M. El-Haj, A. AbuRa'ed, M. Litvak, N. Pittaras, and G. Giannakopoulos, "The financial narrative summarisation shared task (FNS 2020)," in *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*. Barcelona, Spain (Online): COLING, Dec. 2020, pp. 1–12. [Online]. Available: https://aclanthology.org/2020.fnp-1.1

[14] N. Zmandar, M. El-Haj, P. Rayson, M. Litvak, G. Giannakopoulos, N. Pittaras *et al.*, "The financial narrative summarisation shared task fns 2021," in *Proceedings of the 3rd Financial Narrative Processing Workshop*, 2021, pp. 120–125.

[15] T. Daudert and S. Ahmadi, "CoFiF: A corpus of financial reports in French language," in *Proceedings of the First Workshop on Financial Technology and Natural Language Processing*, Macao, China, Aug. 2019, pp. 21–26. [Online]. Available: https://aclanthology.org/W19-5504

[16] P. Malo, A. Sinha, P. Korhonen, J. Wallenius, and P. Takala, "Good debt or bad debt: Detecting semantic orientations in economic texts," *Journal of the Association for Information Science and Technology*, vol. 65, 2014.

[17] K. Kowsari, M. Heidarysafa, D. E. Brown, K. J. Meimandi, and L. E. Barnes, "Rmdl: Random multimodel deep learning for classification," in *Proceedings of the 2Nd International Conference on Information System and Data Mining*, ser. ICISDM '18. New York, NY, USA: ACM, 2018, pp. 19–28. [Online]. Available: http://doi.acm.org/10.1145/3206098.3206111

[18] Kowsari, J. Meimandi, Heidarysafa, Mendu, Barnes, and Brown, "Text classification algorithms: A survey," *Information*, vol. 10, no. 4, p. 150, Apr 2019. [Online]. Available: http://dx.doi.org/10.3390/info10040150

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[21] Y. Yang, M. C. S. UY, and A. Huang, "Finbert: A pretrained language model for financial communications," 2020.

[22] M. Heidarysafa, K. Kowsari, D. E. Brown, K. Jafari Meimandi, and L. E. Barnes, "An improvement of data classification using random multimodel deep learning (rmdl)," vol. 8, no. 4, pp. 298–310, 2018.

[23] G. Andrew and J. Gao, "Scalable training of L1-regularized log-linear models," in *Proceedings of the 24th International Conference on Machine Learning*, 2007, pp. 33–40.

[24] M. S. Rasooli and J. R. Tetreault, "Yara parser: A fast and accurate dependency parser," *Computing Research*

Repository, vol. arXiv:1503.06733, 2015, version 2. [Online]. Available: http://arxiv.org/abs/1503.06733

[25] R. K. Ando and T. Zhang, "A framework for learning predictive structures from multiple tasks and unlabeled data," *Journal of Machine Learning Research*, vol. 6, pp. 1817–1853, Dec. 2005.

[26] P. Malo, A. Sinha, P. Takala, P. Korhonen, and J. Wallenius, "Good debt or bad debt: Detecting semantic orientations in economic texts," 2013.

[27] G. Wang, C. Li, W. Wang, Y. Zhang, D. Shen, X. Zhang, R. Henao, and L. Carin, "Joint embedding of words and labels for text classification," in *ACL*, 2018.

[28] Y. Kim, "Convolutional neural networks for sentence classification," 2014.

[29] M. Toshevska, F. Stojanovska, and J. Kalajdjieski, "Comparative analysis of word embeddings for capturing word similarities," *6th International Conference on Natural Language Processing (NATP 2020)*, Apr 2020. [Online]. Available: http://dx.doi.org/10.5121/csit.2020.100402