LIPI at the FinNLP-2022 ERAI Task: Ensembling Sentence Transformers for Assessing Maximum Possible Profit and Loss from Online Financial Posts

Sohom Ghosh[1,2] and **Sudip Kumar Naskar**[2]
[1]Fidelity Investments, Bengaluru, India
[2]Jadavpur University, Kolkata, India
{sohom1ghosh, sudip.naskar}@gmail.com

Abstract

Using insights from social media for making investment decisions has become mainstream. However, in the current era of information explosion, it is essential to mine high-quality social media posts. The FinNLP-2022 ERAI task deals with assessing Maximum Possible Profit (**MPP**) and Maximum Loss (**ML**) from social media posts relating to finance. In this paper, we present our team LIPI's approach. We ensembled a range of Sentence Transformers to quantify these posts. Unlike other teams with varying performances across different metrics, our system performs consistently well. Our code is available here[1].

1 Introduction

Over the last few years, financial opinion mining has emerged to be an interesting area of research (Chen et al., 2021b). Several research (Mao et al. (2012), Sprenger et al. (2014), Lee et al. (2015), Pagolu et al. (2016), Asur and Huberman (2010), Elliott et al. (2018), Crowley et al. (2021)) highlight the importance of social media posts for predicting stock markets. Although the wisdom of the crowd matters, it is still necessary to mine quality posts from the rest. Quantifying social media posts in terms of the expected profitability is an open area for research. Chen et al. (2021a) proposed two metrics: Maximum Possible Profit (**MPP**) and Maximum Loss (**ML**) for evaluating such posts. They recently hosted the FinNLP-2022 ERAI Task[2] (in conjunction with EMNLP-2022[3]). It comprises pairwise comparison (Task-1) and unsupervised ranking (Task-2) of financial social media posts with respect to **MPP** and **ML**. In this paper, we describe our best-performing systems (Task-1 → **MPP**: 57.47% & **ML**: 59.77%; Task-2 → **MPP**:18.27% & **ML**: -3.90%).

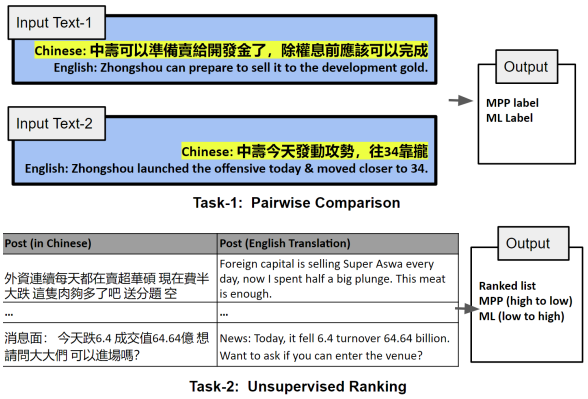

Figure 1: ERAI FinNLP-2022 Tasks

2 Problem Statement

For Task-1, given two posts, the task is to develop a system for evaluating which of them will lead to greater **MPP** and lower **ML**.
For Task-2, given a set of posts, the task is to develop a system for ranking these posts in terms of higher **MPP** and lower **ML** values.
Results of Task-1 were evaluated using accuracy. For Task-2, average **MPP** and **ML** values of top 10% posts were considered for evaluation.

3 Datasets

The organizers initially provided the participants with two datasets. The first dataset (corresponding to Task-1) had 200 instances out of which 2 were null. We dropped the null instances from our experiments. Each instance consists of two posts (in Chinese as well as in English), their **MPP** and **ML** values, and labels corresponding to each post. In the dataset, the **ML** label is set to '1' for an instance (i.e., a pair of posts) when the **ML** value of the first post is less than that of the second post, otherwise the ML label is set to '0'. On the contrary, the **MPP**

[1]https://github.com/sohomghosh/LIPI_ERAI_FinNLP_EMNLP-2022/
[2]https://sites.google.com/nlg.csie.ntu.edu.tw/finnlp-2022-emnlp/erai-shared-task
[3]https://2022.emnlp.org/

Proceedings of the Fourth Workshop on Financial Technology and Natural Language Processing (FinNLP), pages 111 - 115

is set to '0' for an instance (i.e., a pair of posts) when the **ML** value of the first post is less than that of the second post, otherwise the **MPP** label is set to '1'. The posts in the dataset were collected from social media platforms like PTT[4] and Mobile01[5]. We refer to this as **D1**. For Task-2, a dataset consisting of 210 unlabelled posts (in Chinese as well as in English) were provided. This dataset is referred to as **D2**. **D2** serves as the test set for Task-2. Subsequently, the organizers released a test set consisting of 87 pairs of unlabelled posts (in Chinese and English) for pairwise comparison. We refer to this as **D3**.

Data Preparation

We created training and validation sets from **D1** maintaining a split ratio of 80:20. We extended **D1** in two ways.

Firstly, we treat each post from the pair individually, i.e.,tuple (post-1, post-2, MPP-1, MPP-2, ML-1, ML-2) is converted into 2 tuples – (post-1, MPP-1, ML-1) and (post-2, MPP-2, ML-2). This gave us 320 instances for training and 80 for validation. We refer to this training set as **D4**. For sub-systems SB-1 (§4.1), SB-2 (§4.2) and SB-4 (§4.4), we used this set.

Secondly, we expanded **D4** by comparing each post to every other post after removing the null instances. It resulted in 97,032 instances of training. This is referred to as **D5**. The validation set was kept as it is. We use this in sub-systems SB-3 (§4.3) and SB-5 (§4.5).

Chen et al. (2022) narrates the dataset and problem statement in more detail. The formulas for calculating **MPP** and **ML** are mentioned in (Chen et al., 2021a). In Figure 1, we present the problem statement and a sample dataset.

4 Sub-systems

Since our submitted systems are ensemble of multiple sub-systems, we explain each of the sub-systems here. More details regarding the hyper-parameters of each sub-system are reported in the shared codebase.

4.1 Sub-System 1 (SB-1)

For all the Chinese posts in **D4**, we extracted the corresponding embeddings using sbert-chinese-qmc-finance.[6] We trained a linear regression model using the embedding as input to learn either **MPP** values or **ML** values based on requirements. We chose linear regression to start with as we did not have much data to train.

4.2 Sub-System 2 (SB-2)

This sub-system is similar to SB-1 (§4.1). The only difference is that we trained a neural network (multi-layer perceptron model) for 50 iterations instead of linear regression.

4.3 Sub-System 3 (SB-3)

For this sub-system, we used the **D5** dataset. For each pair of Chinese posts present in **D5**, we concatenated the embeddings for each of the posts obtained using sbert-chinese-qmc-finance[7]. We trained a linear regression mode to learn the difference of either **MPP** values or **ML** values between each post present in a given pair.

4.4 Sub-System 4 (SB-4)

We customised the BERT model's architecture (Devlin et al., 2019) for the task of regression such that its last layer learns to predict either the **MPP** values or the **ML** values. This was done by passing the representation of the [CLS] token through a fully connected linear layer having 128 neurons followed by a layer with $tanh$ activation. We initialised it with the weights from the FinBERT model (Araci, 2019). We used only the English posts present in **D4** for this.

4.5 Sub-System 5 (SB-5)

We extracted FinBERT (Araci, 2019) embeddings corresponding to all the English posts present in **D5**. We trained a multi-layer perceptron model for 500 iterations which takes this embedding as input and predicts the difference between either **MPP** values or **ML** values corresponding to each post present in a given pair.

5 Best Performing Systems

In this section, we narrate the systems corresponding to our best-performing submissions.

[4] https://www.ptt.cc/index.html accessed on 09/17/2022

[5] https://www.mobile01.com/ accessed on 09/17/2022

[6] https://huggingface.co/DMetaSoul/sbert-chinese-qmc-finance-v1 accessed on 09/17/2022

[7] https://huggingface.co/DMetaSoul/sbert-chinese-qmc-finance-v1 accessed on 09/17/2022

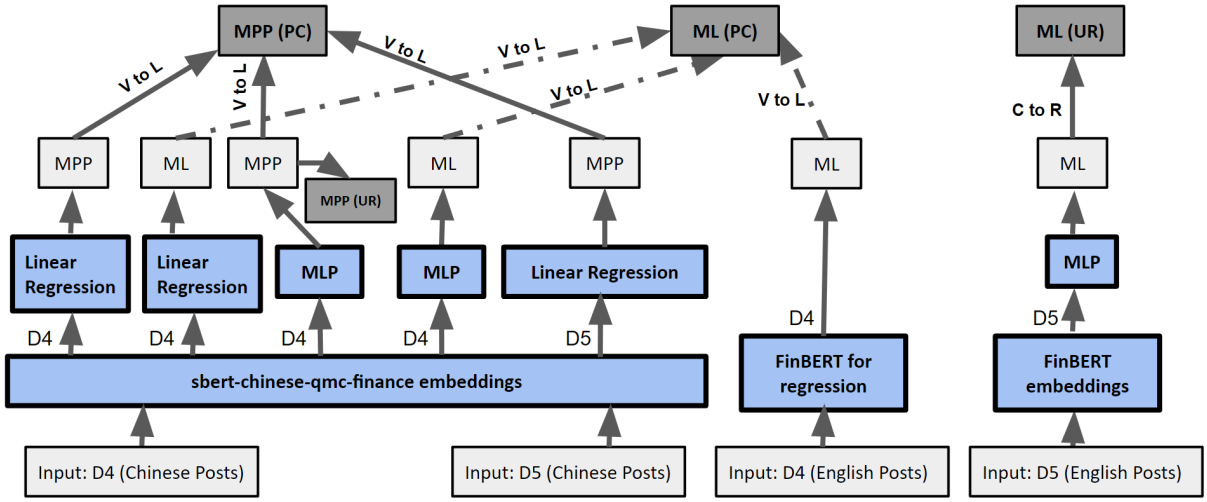

Figure 2: Ensemble Architecture. PC: Pairwise comparison, UR: Unsupervised Rankings, V to L: values to labels by comparison, C to R: comparison to rankings.

5.1 MPP calculation for Pairwise Comparison

This is an ensemble of three subsystems SB-1 (§4.1), SB-2 (§4.2) and SB-3 (§4.3). While SB-1 and SB-2 were trained with the objective of learning the **MPP** values, SB-3 was trained with the objective of learning the difference in **MPP** values for a given pair of posts. For SB-1 and SB-2, to obtain labels from raw **MPP** values, we computed and compared the **MPP** values of the posts constituting each pair in the test set. When **MPP** value of the first post was greater than **MPP** value of the second post, we assigned label '1', otherwise we assigned label '0'. For SB-3, we assigned label '1' when the predicted difference in **MPP** is greater than 0, otherwise we assigned label '0'. The final decision for the **D3** is made based on majority voting.

5.2 ML calculation for Pairwise Comparison

This system consists of selecting the final output from the predictions made by SB-1 (§4.1), SB-2 (§4.2) and SB-4 (§4.4) based on majority voting. Each of these constituent sub-systems were trained with the objective to learn the **ML** values. We scored each of these sub-systems on every post present in **D3**. Subsequently, we compared the raw **ML** values of posts constituting each pair in the test set. Label '1' was assigned when **ML** value of the first post was lesser than that of the second post, otherwise label '0' was assigned.

5.3 MPP calculation for Unsupervised Ranking

SB-2 (§4.2) was trained to predict the **MPP** value for a given post. We scored **D2** using SB-2 and ranked the posts in decreasing order of predicted **MPP** values.

5.4 ML calculation for Unsupervised Ranking

We trained SB-5 (§4.5) to learn the difference in **ML** values for a given pair of posts. We used this system to compare and sort the instances in **D2** in increasing order of predicted **ML** values.

Figure 2 gives a pictorial representation of all the ensemble models.

6 Experiments and Results

This section states various experiments we performed and their results. We started with SB-1 which is a linear regression model trained over sentence embeddings. We tried financial sentence embeddings available for Chinese as well as the English language. Subsequently, we replaced the linear regression model with a multi-layer perceptron model. We further experimented by transforming the original training set **D1** to **D4** and **D5**. We also tried altering the last layer of the BERT (Devlin et al., 2019) model for the task of regression. For the pairwise classification task, we used the regression models to get the **MPP/ML** values for each post in a pair. We then assigned a label to the pair by comparing these values as mentioned in §3. The results are presented in Tables 1 and 2. In this paper we focus on the best-performing systems among

		Train/Valid.		MPP (Pairwise Comparison)			MPP (Unsupervised Ranking)		
Sl.#	**Model**	**Data**	**Language**	**Train**	**Valid.**	**Test (D3)**	**Train**	**Valid.**	**Test (D2)**
1.1	SB-1	D4	Chinese	100.00%	70.00%	54.02%	8.04%	2.98%	11.83%
1.2	SB-2	D4	Chinese	62.18%	67.50%	48.28%	3.89%	2.45%	**18.27%**
1.3	SB-3	D5	Chinese	99.63%	60.00%	41.38%	-	-	17.46%
1.4	SB-4	D4	English	51.92%	47.50%	50.57%	2.11%	3.94%	4.17%
1.5	SB-5	D5	English	99.59%	45.00%	55.17%	-	-	16.63%
1.6	Ensemble (§5.1)	-	-	-	72.50%	**57.47%**	-	-	-

Table 1: MPP Results

		Train/Valid.		ML (Pairwise Comparison)			ML (Unsupervised Ranking)		
Sl.#	**Model**	**Data**	**Language**	**Train**	**Valid.**	**Test (D3)**	**Train**	**Valid.**	**Test (D2)**
2.1	SB-1	D4	Chinese	97.44%	52.50%	50.57%	-10.26%	-2.16%	-7.81%
2.2	SB-2	D4	Chinese	57.69%	55.00%	50.57%	-5.55%	-8.01%	-5.56%
2.3	SB-3	D5	Chinese	99.65%	52.50%	47.12%	-	-	**-3.90%**
2.4	SB-4	D4	English	58.00%	50.00%	**59.77%**	-1.87%	-1.35%	-6.29%
2.5	SB-5	D5	English	91.24%	55.00%	44.83%	-	-	-4.11%
2.6	Ensemble (§5.2)	-	-	82.05%	57.50%	50.57%	-	-	-

Table 2: ML Results

all our submissions due to page constraints. The other approaches we tried include classification of posts separated by *[SEP]* token using various variants of BERT (Devlin et al., 2019). Since the **D4** dataset consists of single posts, we use the same training and validation set for both the tasks. As the **D5** dataset comprises only of pairs of posts, we are unable to provide its performance in the unsupervised ranking task corresponding to the training and validation set. We ensembled models with varying lengths of the training set, therefore we do not report the performance of the model mentioned in §5.1 for the training set. Similarly, for the unsupervised ranking task, we do not report the performances of the models describe in §5.1 and §5.2 as these models were suitable for pairwise comparison task only. The performance of the participating teams has been reported here(Chen et al., 2022).We used labelled instances from **D4** to assess the performance of the unsupervised ranking models as well. This helped us in choosing the best performing models. As **D5** was suitable for pairwise comparison task only, we could not use it to evaluate the models which were developed for the unsupervised ranking task. It is interesting to observe that our ensemble system's performance (Sl.# 1.6) is next only to that of team *Jetsons* in the pairwise comparison task using **MPP**. Moreover, in the same task using **ML** our subsystem SB-4 (Sl.# 2.4) performs as good as that of the best performing team *DCU-ML* (accuracy: 59.77%). However, we did not submit this sub-system separately as it did not perform well on the validation set and submitted the results of the ensemble model (Sl.# 2.6) instead. In the unsupervised ranking using **MPP** task, only team *PromptShots*'s system performed better than that of ours (Sl.# 1.2). However, in the unsupervised ranking using **ML** task, the performance of the system developed by team *Yet* and the baseline solution were better than that of our systems (Sl.# 2.3 and 2.5). In this case as well we did not submit the result corresponding to SB-3 (Sl.# 2.3) where **ML** of top 10% post is -3.90% on the test set because the underlying system could not be evaluated on the validation set obtained from **D5**. We submitted results of SB-5 (Sl.# 2.5) instead.

7 Conclusion

Comparing the performance of our models with that of the other participants, we conclude that our models performed consistently well. We also observe that in most cases we achieve better performances using the Chinese texts than the translated version in English. This is because we are losing out on the nuances during translation. We further observe that ensembling helps in improving the overall performance.

Collecting more financial posts in a resource-rich language like English and incorporating prices of the stock whose **MPP** and **ML** are being discussed as input to the model are interesting directions for future work.

8 Limitations

The training dataset is very small in size and does not assure how the system will perform in real life. Fine-tuning large language models like BERT on **D5** is compute intensive. Moreover as the **MPP** and **ML** calculation differs for bullish and bearish market, it would be nice to take market conditions into consideration.

Ethics Statement

This research has been done for academic purposes. The authors declare that there are no underlying commercial interests. Investment in stock markets is risky and may lead to monetary losses. Investors are advised to use their discretion instead of blindly relying on these models' output.

Disclaimer

The opinions expressed in this paper are of the authors. They do not reflect the opinions of their affiliations.

References

Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models.

Sitaram Asur and Bernardo A. Huberman. 2010. Predicting the future with social media. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, volume 1, pages 492–499.

Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2021a. Evaluating the rationales of amateur investors. In *Proceedings of the Web Conference 2021*, WWW '21, page 3987–3998, New York, NY, USA. Association for Computing Machinery.

Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2021b. Financial opinion mining. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*, pages 7–10, Punta Cana, Dominican Republic & Online. Association for Computational Linguistics.

Chung-Chi Chen, Hen-Hsen Huang, Hiroya Takamura, and Hsin-Hsi Chen. 2022. Overview of the finnlp-2022 erai task: Evaluating the rationales of amateur investors. In *Proceedings of the Fourth Workshop on Financial Technology and Natural Language Processing*, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Richard M Crowley, Wenli Huang, and Hai Lu. 2021. Executive tweets. *Available at SSRN 3975995*.

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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

W Brooke Elliott, Stephanie M Grant, and Frank D Hodge. 2018. Negative news and investor trust: The role of $ firm and # ceo twitter use. *Journal of Accounting Research*, 56(5):1483–1519.

Lian Fen Lee, Amy P Hutton, and Susan Shu. 2015. The role of social media in the capital market: Evidence from consumer product recalls. *Journal of Accounting Research*, 53(2):367–404.

Yuexin Mao, Wei Wei, Bing Wang, and Benyuan Liu. 2012. Correlating s&p 500 stocks with twitter data. In *Proceedings of the First ACM International Workshop on Hot Topics on Interdisciplinary Social Networks Research*, HotSocial '12, page 69–72, New York, NY, USA. Association for Computing Machinery.

Venkata Sasank Pagolu, Kamal Nayan Reddy, Ganapati Panda, and Babita Majhi. 2016. Sentiment analysis of twitter data for predicting stock market movements. In *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pages 1345–1350.

Timm O Sprenger, Andranik Tumasjan, Philipp G Sandner, and Isabell M Welpe. 2014. Tweets and trades: The information content of stock microblogs. *European Financial Management*, 20(5):926–957.