Financial Sentimental Analysis

Elanthamil Jeeva
Elanthamil.Elilarasu-Premalatha@city.ac.uk

1. Problem statement and Motivation

Financial markets are highly reactive to news and events, which can have a large impact on stock prices. Various forms of text data, including news articles and social media posts, can shape trading behaviours of investors and ultimately influence stock price movements. However, deciphering the impact of news on stock prices is a multidimensional challenge that involves not only capturing the emotional information and sentiment of news articles but also understanding the hidden relationships between news and market fluctuations. Analyzing financial sentiment poses a significant challenge due to the unique terminology and scarcity of labeled data in this domain. Traditional models often fail because they struggle to comprehend the specialized language used in financial contexts. Therefore, we explored the pre-trained language models that can help bridge this gap. These models require fewer labeled examples to achieve accurate results and can be fine-tuned on domain-specific datasets, making them a promising solution for financial sentiment analysis. Sentiment analysis is a vital component of various financial applications, including stock price forecasting, credit risk assessment, economic report analysis, and investment decision-making. To tackle these tasks, researchers have employed a range of techniques to fine-tune pre-trained language models (PLMs). While these models have demonstrated impressive results, there is still room for improvement. Therefore, we are finetuning the PLM on finance text so as to inject the financial domain knowledge into the existing PLM.

This project aims to create a robust sentiment analysis model that can effectively capture the emotional tones of news articles and assess their impact on stock market volatility. By analyzing the sentiment of news articles i.e positive, negative or neutral, we can identify emerging trends and patterns that may influence stock prices, ultimately empowering investors to make more informed decisions and refine their investment approaches.

2. Research hypothesis

Fine-tuning transformer-based models such as BERT and RoBERTa will beat traditional machine learning models such as Support Vector Machines (SVM) in reliably classifying sentiment in financial text because of their greater capacity to capture contextual and semantic nuances in language. This hypothesis argues that when the additional capabilities of transformer-based models are fine-tuned, they will outperform classic models such as SVM in the specialised task of financial sentiment analysis.

3. Related work and background

Researchers in natural language processing (NLP) have leveraged various forms of textual data to provide new insights in the finance domain, including stock price prediction (Fortuny et al,2014), risk prediction (Li et al,2020), financial entity extraction (Loukas et al,2022), and other financial applications. These efforts have employed a range of methodologies, including rule-based methods, deep learning, and financial text analytics. Financial text analytics possess distinct characteristics that set it apart from text

analytics in general domains. Firstly, unstructured data such as news articles, reports, and social media posts are constantly updated in real-time, making them crucial for the financial domain. Secondly, metrics closely related to investment information, such as sentiment scores, are essential. Finally, words like 'increase' and 'decrease' hold significant meaning in financial texts compared to general domains. This study focuses specifically on financial news datasets, which differ from corporate reports or 10-K filings in that they are text data uploaded in real-time. Moreover, these datasets are unique in having financial news sentiment analysis benchmarks annotated by experts, allowing for precise and objective evaluations. Previous studies have shown that financial sentiment analysis plays a significant role in enhancing stock prediction performance (Kalyani et al,2016, Li et al,2022, Souma et al, 2019). To facilitate language models in learning these characteristics, researchers typically use pre-trained language models (PLMs). This is also called Transfer learning, where a PLM is further trained for a specific task (Weiss et al, 2016), offers the advantage of training the model to be domain or task specific. This process enables the language model to learn semantic relations from text in the target domain, as well as tasks that may have a different data distribution from that of the general corpus. This is particularly useful in domain specific research.

Sentiment analysis has been a topic of extensive research, with many surveys and review articles exploring its methodologies and applications. A notable survey by (Medhat et al, 2014) provides a comprehensive overview of sentiment analysis techniques and their applications, as well as related disciplines such as emotion recognition and resource development. Recent studies have also made significant contributions to the field of sentiment analysis. For instance, (Abdul et al, 2019) conducted an investigation on the polarity of the IMDB dataset using sentiment analysis to develop a transformer-based model, specifically BERT. Their findings showed that the BERT model outperformed previous machine learning and deep learning-based models on the their dataset.

Another notable study by (Gong et al, 2022) proposed a transformer-based method that combines knowledge distillation and text augmentation. This approach reduces processing costs and training time while improving overall performance. The results of their study demonstrated the effectiveness of their method, with an accuracy of 93.28% for emotion recognition in text, outperforming other models such as BERT (93.38%), ALBERT (92.06%), and mobileBERT (92.74%).

Machine learning approaches have employed a range of techniques to analyze sentiment data. For instance, researchers have used support vector machines (SVM), Naive Bayes, and decision trees to classify stock market data into bullish and bearish categories, with SVM demonstrating superior performance (Wang et al, 2015). Based on this work, subsequent studies have integrated machine learning with lexicon-based techniques, leading to the development of more effective models (Cortis et al, 2017). However, despite the advancements in methodology, challenges persist, including issues with accuracy and data sparsity (Sohangir et al, 2018). To address the challenges in sentiment analysis, researchers have explored the use of deep learning approaches. Prior to the development of the Transformer architecture by Google, researchers employed various word embedding

techniques, including tf-idf, countvector GloVe, Word2Vec, and ELMo, to represent tokens as vectors and store information. Additionally, the introduction of convolutional neural networks, recurrent neural networks (RNNs), long short-term memory (LSTM), and attention mechanisms has led to significant advancements in sentiment analysis (SA) (Sohangir et al, 2018 , Yang et al, 2016). The emergence of BERT, a transformer-based model, has shown a great impact on SA performance.

4. Data Set

(Malo et al, 2013) introduced the Financial Phrase Bank dataset, a freely available resource for financial sentiment classification. This dataset consists of 4846 phrases taken from financial news stories from LexisNexis' financial news database and manually labelled by 16 financial market experts. Each phrase is assigned one of the three sentiment labels: positive, neutral, or negative. They were asked to classify the sentences based on how they thought the information may affect the stock price of the corresponding company. We divided the dataset into three sets: training, validation, and testing, with 20% of the phrases kept for testing and 10% of remaining 80% training data is for validation. Fig-1 and Fig-2 shows quantity of various sentiment data and their bar plot respectively.

	Sentiment	count
0	neutral	2879
1	positive	1363
2	negative	604

Figure-1: Distribution of sentiments

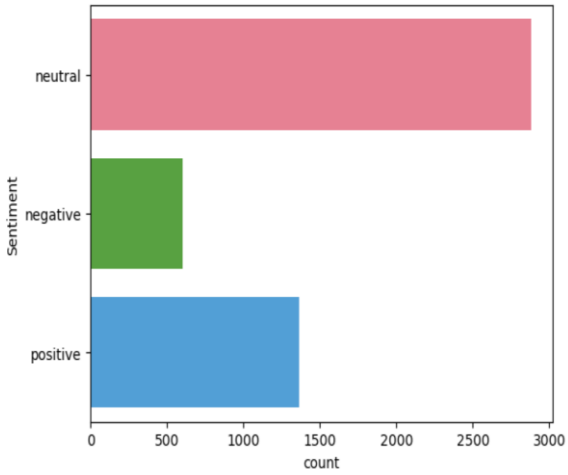

Figure-2: Bar plot for sentiment Distribution

5. Accomplishments

1. Task-1: Financial phrasal bank dataset is loaded using pandas. Since Transformer based model does not require text to be pre-processed, therefore we are not applying any kind of pre-processing methods for any of the experimented models to perform fair comparison - *Completed*

2. Task-2: To prepare the data for SVM, we applied TF-IDF vectorization as well as Count vectorization - *Completed*

3. Task-3: Build the BERT/RoBERTa model and perform finetuning on training set of Financial phrasal bank dataset- *Completed*

4) Task-4: Save those finetuned model (checkpoints) on huggingface account for future reuse to perform testing- *Completed*

5) Task-5: Load those finetuned model one by one and Perform testing – *Completed*

6. Baselines and Pros/Cons

6.1 BERT: BERT (Bidirectional Encoder Representations from Transformers) is a pre-trained language model developed by

Google in 2018. It's a significant improvement over traditional word embeddings, such as Word2Vec, as it uses a multi-layer bidirectional transformer encoder to generate contextualized representations of words in a sentence. This means that BERT can capture the nuances of language, including syntax, semantics, and pragmatics, and can understand the relationships between words in a sentence. BERT is trained on a massive corpus of text data, including the Wikipedia and BookCorpus datasets, which totals over 16 GB of text. This large-scale training enables BERT to learn a rich representation of language that can be fine-tuned for specific tasks such as sentiment analysis, question answering, and text classification. When BERT processes a sentence, it takes in the input sentence and generates a vector representation of each word in the sentence. This vector representation captures the context in which the word is used, including the surrounding words, syntax, and semantics. BERT's contextualized representations can capture complex relationships between words, which can improve the accuracy of downstream tasks. BERT's ability to capture complex relationships between words is due to its use of self-attention mechanisms, which allow it to weigh the importance of different words in the sentence. This enables BERT to focus on the most relevant words and ignore irrelevant ones, resulting in more accurate representations of the input sentence. BERT's pre-trained model can be fine-tuned for specific tasks by adding a task-specific layer (regression or classification) on top of the pre-trained model. This allows BERT to adapt to different tasks and datasets, making it a versatile and widely applicable language model.

Pros:

1. State-of-the-art performance: BERT has achieved state-of-the-art performance on many text classification tasks, including finance-related tasks.
2. Pre-trained on large corpus: BERT is pre-trained on a large corpus of text data, which allows it to learn general language patterns and relationships.
3. Contextualized representations: It generates contextualized representations of words, which can capture complex relationships between words and improve the accuracy of downstream tasks.
4. Easy to fine-tune: It can be easily fine-tuned for specific tasks, such as sentiment analysis or topic modeling.

Cons:

1. Computational resources: It requires significant computational resources to train and fine-tune, which can be a challenge for individual user and smaller organizations.
2. Limited interpretability: BERT's complex architecture can make it difficult to interpret its predictions and understand the relationships between words.
3. Requires large dataset: BERT requires a large dataset to achieve optimal performance, which can be a challenge for finance-related tasks where data is often limited.

6.2 RoBERTa: RoBERTa (Robustly Optimized BERT Pretraining Approach) is a variant of the BERT model and is an improvement over BERT, with several key differences in its architecture and training objectives. One of the main differences is the use of a dynamic masking technique, which randomly masks some of the input tokens during training. This technique helps RoBERTa to learn more robust representations of language that are less dependent on the specific input data. RoBERTa is trained on an even larger corpus of text data than BERT, including the Wikipedia and BookCorpus datasets, as well as the Common Crawl dataset. This larger training corpus enables RoBERTa to learn a more comprehensive representation of language that can capture more nuanced relationships between words. RoBERTa uses a similar architecture to BERT, with a multi-layer bidirectional transformer encoder. However, RoBERTa has a larger model size, with 355 million parameters compared to BERT's 110 million parameters. This larger model size enables RoBERTa to capture more complex relationships between words and to handle longer sentences. When RoBERTa processes a sentence, it takes in the input sentence and generates a vector representation of each word in the sentence. This vector representation captures the context in which the word is used, including the surrounding words, syntax, and semantics. RoBERTa's improved performance is due to its ability to capture more nuanced relationships between words and its ability to handle longer sentences. RoBERTa's pre-trained model can be fine-tuned for specific tasks by adding a task-specific layer on top of the pre-trained model. This allows RoBERTa to adapt to different tasks and datasets, making it a versatile and widely applicable language model.

Pros:

1. Improved performance: RoBERTa has achieved improved performance on many text classification tasks, including finance-related tasks.
2. Robust to overfitting: RoBERTa is more robust to overfitting than BERT, especially when fine-tuned on small datasets.

Cons:

1. Requires more computational resources: RoBERTa requires more computational resources to train and fine-tune than BERT.
2. More complex architecture: RoBERTa's architecture is more complex than BERT's, which can make it more difficult to understand and interpret.

6.3 SVM: Support Vector Machine is a type of supervised learning algorithm that is commonly used for classification tasks. SVM is a linear algorithm that works by finding the hyperplane that maximally separates the classes in the feature space. This hyperplane is the decision boundary that separates the classes, and it is determined by the training data. SVM is a robust algorithm that can handle high-dimensional data, making it suitable for text classification tasks. SVM can be used for both binary and multi-class classification tasks, and it can handle non-linear relationships between the features and the target variable. SVM uses a kernel function to map the input data into a higher-dimensional space where the classes are linearly separable. This is known as the kernel trick, and it allows SVM to handle non-linear relationships between the features and the target variable. Common kernel functions used in SVM include the

linear kernel, polynomial kernel, and radial basis function (RBF) kernel. It is a widely used algorithm in text classification tasks, including spam detection, sentiment analysis, and topic modeling. SVM can be used with a variety of feature extraction techniques, including bag-of-words, TF-IDF, and word embeddings. SVM's performance can be improved by tuning the hyperparameters, including the regularization parameter, the kernel function, and the kernel parameters. Additionally, SVM can be combined with other algorithms, such as gradient boosting and random forests, to improve its performance.

Pros:

1. Easy to implement: SVM is a simple and easy-to-implement algorithm that can be used for text classification tasks.
2. Interpretable: SVM's predictions are easy to interpret, which can make it easier to understand the relationships between words.
3. Less sensitive to hyperparameters: SVM's performance is less sensitive to hyperparameters than BERT's and RoBERTa's.

Cons:

1. May not perform as well as BERT and RoBERTa: SVM may not perform as well as BERT and RoBERTa on many text classification tasks.
2. Requires feature engineering: SVM requires feature engineering, which can be time-consuming and challenging.
3. May not be as effective for long texts: SVM may not be as effective for long texts, which can be a challenge for finance-related tasks.

7. Approach and Methodology

Since PLM may not perform well on domain specific task, Therefore We have finetuned BERT and RoBERTa with last layer as classification head. Once it is finetuned, to make it flexible and available online, it has been uploaded on Huggingface account, their respective checkpoints are listed in Table-1. Hence just by using corresponding checkpoints, the tuned model can be accessed.

Table-1: Custom Finetuned Checkpoints

Model	Finetuned Checkpoints
BERT	Elanthamiljeeva/BERT_FPB_finetuned_v2
RoBERTa	Elanthamiljeeva/RoBERTa_FPB_finetuned_v2

For fine-tuning BERT and RoBERTa models on the Financial Phrasal Bank dataset, we employed the Hugging Face Transformers library's Trainer class with a specific set of training arguments as listed in Table-2. In the same way to train and test on SVM, we used two varients of vectorization 1) Tf-Idf 2) Countvector

Table-2: BERT/RoBERTa Finetuning setup

Epochs	4
Training Batch Size	16
Evaluation Batch Size	64
Warmup Steps	500
Weight Decay	0.01
Logging Steps	10

And used linear kernal with other parameters as default by sklearn library. Model finetuning/training has been done on 80% of the data. Dataset is having class imbalance, but BERT/RoBERTa finetuning does not affect and discriminate while performing prediction because of being

more data for few sentiments and less for others. This is the beauty of this PLM as compared to SVM, which suffer from class imbalance.

8. Result & Discussion

The Table-3 shows the performance of all the models used in the experiments. SVM with count vector is the worst performer, achieving low accuracy and F1 score, whereas SVM with TF-IDF outperforms the count vectorizer method, showing better results. RoBERTa is the best performer, achieving more than 88% accuracy and F1 score. BERT's performance is lagging behind RoBERTa by approximately 2% in accuracy and F1 score. This means that RoBERTa is more effective than BERT in classifying sentiment. Overall, the results show that RoBERTa is the top-performing model. Using Fig-3 and Fig-4, we found that negative and positive sentiment have more jumps as compared to neutral from BERT to RoBERTa model. Therefore, RoBERTa is giving robust and more confident model to detect financial sentiment of text. Using Fig-5, we can conclude that SVM is biased towards class having more data i.e. neutral class.

Table-3: Model performance

Model	Accuracy	F1
SVM+ CountVectorizer	0.751	0.748
SVM+ TfidfVectorizer	0.773	0.761
BERT	0.859	0.859
RoBERTa	0.881	0.881

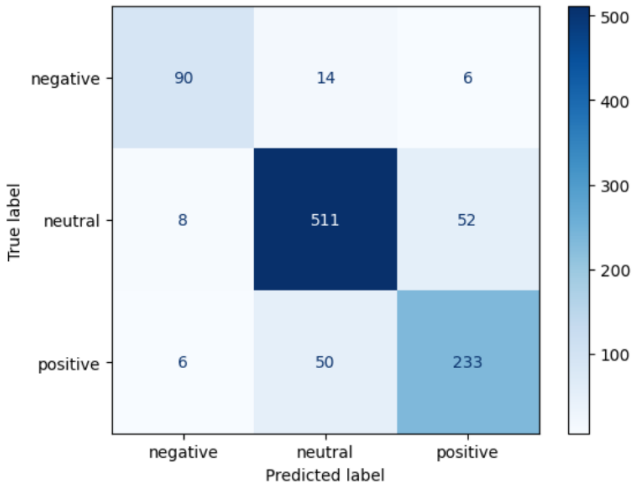

Figure 3: Confusion matrix for BERT

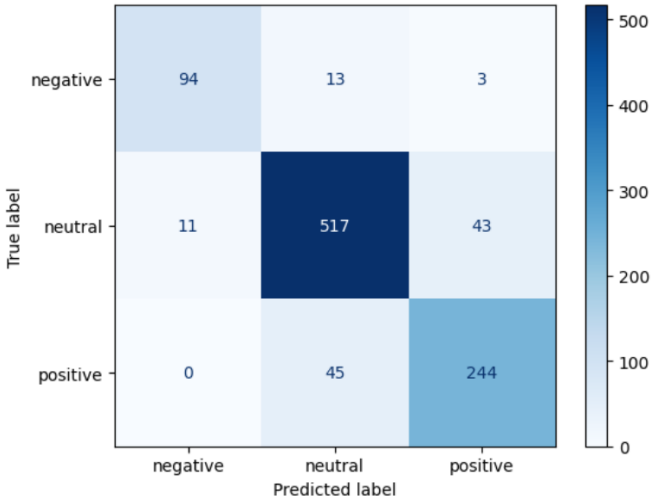

Figure 4: Confusion matrix for RoBERTa

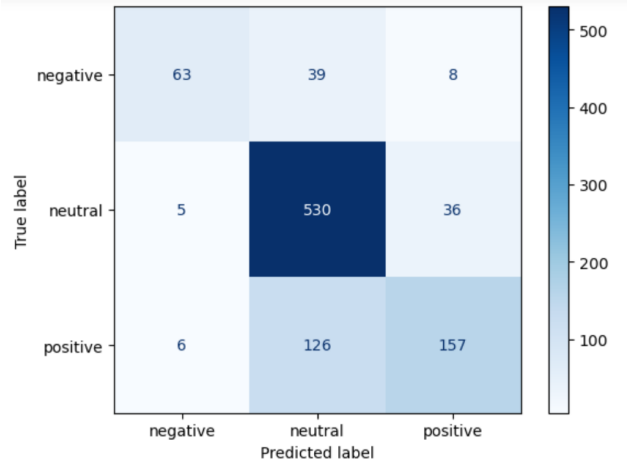

Figure 5: Confusion matrix for SVM-Tf-Idf

9. Lessons learned and conclusions

The fine-tuning of RoBERTa and BERT surpassed the classic SVM model, demonstrating transformer models' greater capacity to read and process complicated financial documents. The comparatively lower performance of the SVM model highlights the limitations of classical machine learning methods in handling the intricacies and subtleties of financial sentiment analysis. The accuracy gap between RoBERTa and BERT shows that architectural changes in RoBERTa, such as more sophisticated language representation, can lead to improved performance in sentiment classification tasks.

Future work will concentrate on improving model performance using advanced techniques like LLM , as well as tackling issues like class imbalance and real-time sentiment analysis for financial news.

References

Abdul, S., Qiang, Y., Basit, S.A., & Ahmad, W. (2019). Using BERT for Checking the Polarity of Movie Reviews. International Journal of Computer Applications.

Akita, R., Yoshihara, A., Matsubara, T., & Uehara, K. (2016). Deep learning for stock prediction using numerical and textual information. 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), 1-6.

Cortis, K., Freitas, A., Daudert, T., Hürlimann, M., Zarrouk, M., Handschuh, S., & Davis, B. (2017). SemEval-2017 Task 5: Fine-Grained Sentiment Analysis on Financial Microblogs and News. International Workshop on Semantic Evaluation.

Fortuny, E.J., Smedt, T.D., Martens, D., & Daelemans, W. (2014). Evaluating and understanding text-based stock price prediction models. Inf. Process. Manag., 50, 426-441.

Gong, X., Ying, W., Zhong, S., & Gong, S. (2022). Text Sentiment Analysis Based on Transformer and Augmentation. Frontiers in Psychology, 13.

Kalyani, J., Bharathi, P.H., & Jyothi, P.R. (2016). Stock trend prediction using news sentiment analysis. ArXiv, abs/1607.01958.

Li, J., Yang, L., Smyth, B., & Dong, R. (2020). MAEC: A Multimodal Aligned Earnings Conference Call Dataset for Financial Risk Prediction. Proceedings of the 29th ACM International Conference on Information & Knowledge Management.

Li, X., Wu, P., & Wang, W. (2020). Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong. Inf. Process. Manag., 57, 102212.

Loukas, L., Fergadiotis, M., Chalkidis, I., Spyropoulou, E., Malakasiotis, P., Androutsopoulos, I., & Paliouras, G. (2022). FiNER: Financial Numeric Entity Recognition for XBRL Tagging. Annual Meeting of the Association for Computational Linguistics.

Malo, P., Sinha, A., Korhonen, P.J., Wallenius, J., & Takala, P. (2013). Good debt or bad debt: Detecting semantic orientations in economic texts. Journal of the Association for Information Science and Technology, 65.

Medhat, W., Hassan, A.H., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. Ain Shams Engineering Journal, 5, 1093-1113.

Sohangir, S., Wang, D., Pomeranets, A., & Khoshgoftaar, T.M. (2018). Big Data: Deep Learning for financial sentiment analysis. Journal of Big Data, 5, 1-25.

Souma, W., Vodenska, I., & Aoyama, H. (2019). Enhanced news sentiment analysis using deep learning methods. Journal of Computational Social Science, 2, 33 - 46.

Tsai, M., & Wang, C. (2017). On the risk prediction and analysis of soft information in finance reports. Eur. J. Oper. Res., 257, 243-250.

Wang, C., Tsai, M., Liu, T., & Chang, C. (2013). Financial Sentiment Analysis for Risk Prediction. International Joint Conference on Natural Language Processing.

Wang, G., Wang, T., Wang, B., Sambasivan, D., Zhang, Z., Zheng, H., & Zhao, B.Y. (2015). Crowds on Wall Street: Extracting Value from Collaborative Investing Platforms. Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing.

Weiss, K.R., Khoshgoftaar, T.M., & Wang, D. (2016). A survey of transfer learning. Journal of Big Data, 3, 1-40.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E.H. (2016). Hierarchical Attention Networks for Document Classification. North American Chapter of the Association for Computational Linguistics.