Sentiment Classification of Financial Texts Using the Financial Phrasebank Dataset

Hugo Veríssimo
Complements of Machine Learning 24/25
University of Aveiro
Aveiro, Portugal
hugoverissimo@ua.pt

João Cardoso
Complements of Machine Learning 24/25
University of Aveiro
Aveiro, Portugal
joaopcardoso@ua.pt

***Abstract*—This work explores financial sentiment analysis using the Financial PhraseBank dataset, a benchmark in the field for its annotated financial news snippets. The performance of three different models, fastText, LSTM, and BERT, is evaluated and compared on a selected subset with 75% annotator agreement. BERT-based models significantly outperform the others, motivating further refinement through data augmentation and a novel weighted training strategy that incorporates annotator agreement levels during training. The proposed models achieve performance in line with, or surpassing, literature baselines, demonstrating the importance of both architecture selection and annotation-informed weighting schemes in financial NLP.**

Keywords: financial sentiment analysis, Financial PhraseBank, deep learning, natural language processing

I. Introduction

With the ever-increasing volume of information created and distributed by the minute, access to fast and reliable analysis of encountered information is more important than ever. Especially with the democratized access to financial instruments and capital markets, where individuals have the possibility to invest in virtually any company on the stock exchange, it is important to have ways to leverage against giant institutions with hundreds of financial analysts at their disposal.

Historically, financial analysis relied heavily on fundamental analysis (examining earnings, balance sheets, annual financial reports), which required extensive knowledge in the field (also the strategy that made Warren Buffett one of the richest men in the world), along with technical analysis (studying price and volume trends). Around 2010, after the 2008 global financial crisis, there was a surge in news analysis to evaluate the tone and derive investment strategies from it [1]. Due to the lack of domain specific lexicon these analysis were falible, until the work by Loughran and McDonald was published, a financial lexicon based on 10-K forms (i.e., annual financial reports) and dictionaries [2]. This allowed to use more sofisticated analysis rather than using the presence of negative words as a signal to sell.

Upon the launch of Twitter, information streams increased dramatically, making more and more data available for analysis. But, machine learning was not heavily used, as most data was not annotated, or there was very little data with high-quality annotations. In 2014, P. Malo *et al.* published a fundamental dataset for financial sentimental analysis, that is still used, the Financial Phrasebank. It is unique, for the inclusion of important aspects as directional expressions (e.g., profits decreased), entity polarity shifts (e.g. profits may be negative if decreased), and phrase level context [3].

With this, machine learning models started finding their place, as the field of natural language processing (NLP) grew and niche fields such as financial investments found more useful data. This work explores the Financial Phrasebank dataset by implementing different machine learning and deep learning models to evaluate the sentiment of sentences related to financial news.

II. State of The Art

The field of NLP has grown drastically in the past decade, progressing from recurrent neural networks (RNN) and related models such as Long-Short Term Memory (LSTM), to the transformers-type models, large language models (LLM) and text generative models as ChatGPT. With the Financial Phrasebank the field expanded into financial analysis, with several works of relevance being published in recent years.

In the work of Araci (2019), the author developed a BERT-based model trained specifically on texts with financial data. BERT, Bidireccional Encoder Representations from Transformers, is a large language model developed by Google (2018), benefitting largely from the fact that it can ”hold” in memory large chunks and in both directions, simultaneously [4]. The fact that it is built on the Transformer encoder architecture, it can weigh the importance of different words in a sentence by using a self-attention. The model is pre-trained on large unlabelled corpora (e.g., Wikipedia, BookCorpus), and can be fine-tuned for specific purposes. In this work, the end model was trained on domain specific corpus such as TRC2-financial data and financial specific texts (over 440 000 sentences), and then fine tuned with the Financial Phrasebank. The model achieved an accuracy of 97% and a F1-score of 95% on the Financial Phrasebank dataset with 100% agreement, but only 86% and 84%, respectively, on the dataset with all levels of agreement (the agreement will be further detailed in the Methodology section) [5].

Later on the model was further improved by Sun *et al.* (2025), EnhancedFinSentiBERT, by including dictionary embeddings, expanded corpus that deversified the pre-training stage drastically, and a novel neutral sentiment module, that further enhanced the distinction between neutral and weak sentiments, resulting F1-score (87%). The pre-training stage benefited from the large diversity of the corpus, going from a few million tokens to 2.4 B tokens with the latest version [6].

In a similar direction, but at the fine-tuning level, Atsiwo (2024) improved the data used in fine-tuning, considering that most datasets have relatively short sentences ($<$ 100 tokens), failing to leverage the full context window of LLMs like BERT (512 tokens). This was achieved by augmenting the training data with synthetic sentences generated by GPT-4, with accuracy of 89% and F1-score of 88% for 50% agreement dataset [7].

GPT has been used with different purposes, as in the work by Fatouros *et al.* (2023), where GPT-3.5Turbo was used for zero-shot sentiment classification. These conditions are harder on the model, as it never undergoes specific training for the context, relying solely on its pre-training (hence the poorer performance against finely tuned models). Under the same conditions (not using the Financial PhraseBank, but scraped headlines related with forex trading), it outperformed finely-tuned models with an accuracy of 75% and F1-score of 74% (finely tuned models in zero-shot conditions achieve accuracy of 56% and F1-score of 55%) [8].

The BERT model was revisited by different researchers, but a new iteration from Facebook AI was proposed (2019) named RoBERTa (Robustly Optimized BERT Approach) was developed, that used a signficantly larger corpus for training (10x larger), and a dynamic masking technique during training, that allowed the model to learn new contextual relations while using the same sentences, making it more robust [9]. This model was used to develop financial models, where the work Choe *et al.* (2023) is worth mentioning, where a large corpus of financial texts were fed to the model for training, from a range of sources (e.g., Reuters, SEC fillings, EIA). The model (FiLM, Financial Language Model) benefited from the diversity of training data, rather than simply focusing on fine tuning with highly curated data, showing improved generalization and better metrics than FinBERT and RoBERTa (accuracy 86%, F1-score 84%) [10].

These models are improving substantially over the years, but it is different to put them to test against a controlled dataset from using them in real life, and the variety included as consequence. Competitions such as FinNLP help drive research in this field, by posing ever more diversified test sets, aiming to improve the robustness of models, and the solutions developed by the researchers.

III. Methodology

To address the problem of sentiment classification in sentences related to financial investment, a pipeline was set up for training and testing using the Financial PhraseBank for three types of models, where the best was further explored and tuned for different tests. The dataset and setup are detailed in the following subsections.

A. Dataset

The Financial PhraseBank is a widely used benchmark dataset for financial sentiment analysis. It consists of roughly 4,840 English sentences (mostly news headlines or short statements) about companies, drawn from financial news articles and press releases. Each sentence is labeled with one of three sentiment classes: positive, negative, or neutral, representing the sentence's sentiment from the perspective of an investor [3], [11].

Table I: Financial PhraseBank distribution. Four possible sets within the dataset, depending on how many financial experts agreed with the attributed label. (majority agreement statistics). The dataset with 50% agreement corresponds to the entire dataset.

Sentiment	**Agreement**			
	50%	**66%**	**75%**	**All**
Negative	604	514	420	303
Neutral	2879	2535	2146	1391
Positive	1363	1168	887	570
Total	4846	4217	3453	2264

The dataset was labeled by 16 finance professionals, each responsible for labelling a subset of sentences. Each sentence was labelled by 5-8 annotators, and the resulting agreement score was a result of the fraction of annotators that labelled the sentence in the same manner. This resulted in 4 different subsets, where 50% agreement corresponds to the entire dataset, with the dataset size decreasing as the level of agreement increases. It is important to mention that the agreement level corresponds to the least allowed, so the 50% agreement level dataset contains the other subsets. The dataset sizes and class proportion can be consulted in Table I, along with sample sentences and the attributed sentiment classification in Table II.

This subset strategy allows researchers to find a balance between the amount and the quality of data, representing a common trade-off in the field of machine learning.

Table II: Raw example sentences from the Financial PhraseBank, each annotated with sentiment labels and associated annotator agreement levels.

Sentence	Sentiment	Agreement level
According to Gran, the company has no plans to move all production to Russia, although that is where the company is growing.	Neutral	100%
The fair value of the company 's investment properties went down to EUR 2.768 billion at the end of 2009 from EUR 2.916 billion a year earlier.	Negative	75%
Basic banking activities continued as normal .	Neutral	66%
In banking , Sampo A was unchanged at 14.24 eur and Nordea rose 0.42 pct to 9.51 eur.	Positive	50%

B. Exploratory Data Analysis

Taking into consideration the different possible subsets, the one with 75% agreement was selected, as it offers a balance between quantity and quality in the dataset, while also maintaining class proportion.

In Fig. 1 the number of sentences per class is evidence of how imbalanced the dataset is. As a result, we had to balance the dataset by undersampling all classes to the amount of examples for the *Negative* class, keeping 336 sentences per class.

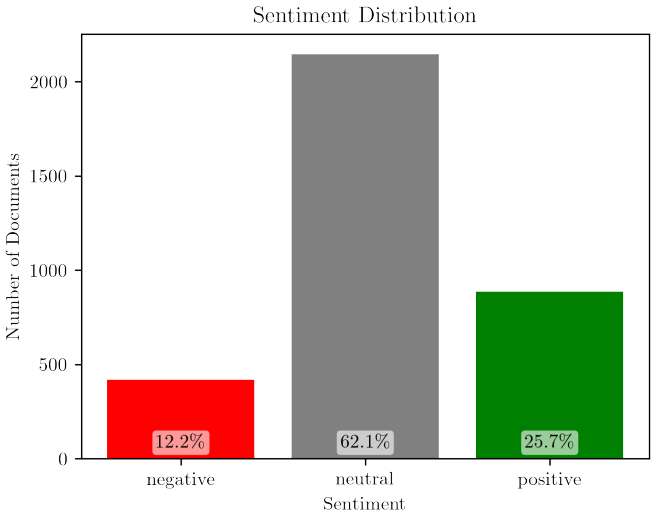

Figure 1: Class distribution in the 75% agreement dataset.

The frequency distribution of document lengths helped determine the maximum number of tokens to use (considering the limit is 512 for BERT), as shown in Fig. 2.

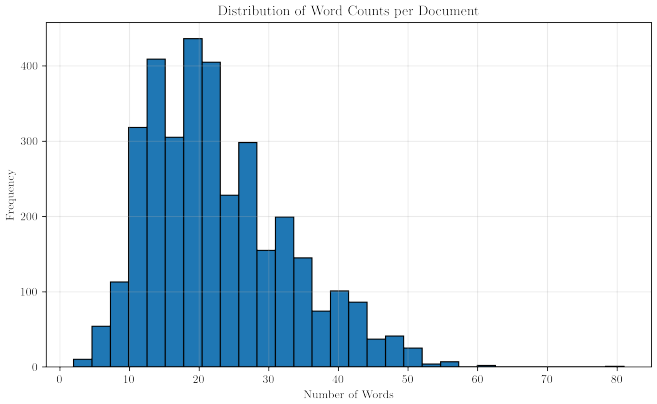

Figure 2: Word count distribution per document for the 75% agreement dataset.

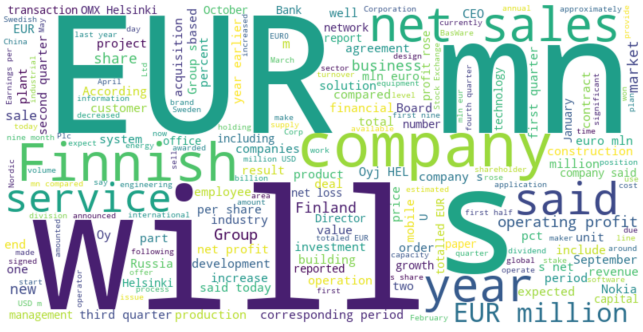

Figure 3: Most frequent words in the dataset, visualized as a word cloud.

Figure 3 shows a word cloud with the most frequent words in the dataset subset with 75% annotator agreement, highlighting key terms that dominate the financial news domain. To identify words most indicative of sentiment class, word frequencies were then compared across sentiment categories. Terms such as *down*, *decreased*, *profit*, *fell*, and *rose* appeared with significantly higher frequency in specific classes, making them particularly relevant for classification.

C. Preprocessing

The selected dataset (75% agreement) was split 80/20 for training/testing. The testing dataset was unique, meaning that all the sentences present in this subset were removed from any other dataset (of all the possible agreement levels), to prevent data leakage.

The preprocessing changed slightly between models and is detailed in the corresponding sections. The preprocessing described here was performed prior to any model training, as was the class balance mentioned earlier.

D. Model Evaluation and Validation Strategy

Prior to model training, 5-fold cross-validation was performed for hyperparameter tuning, followed by training on the full dataset. Models were continuously evaluated using

learning curves and comparative metrics such as the confusion matrix, F1-score, and accuracy.

IV. Model Architectures: LSTM, FastText, and BERT

With the maining goal of classifying the sentiment of sentences, we have selected three models for an initial assessment, and proceeded with more complex iterations on the best model. The final model was further developed with data augmentation, and weighted classes, that will be further detailed below. The models selected were: Long Short-Term Memory (LSTM), fastText, and Bidirectional Encoder Representations from Transformers (BERT).

1) Long Short-Term Memory: LSTM is a type of recurrent neural network (RNN), introduced by Hochreiter and Schmidhuber (1997). Its architecture was designed to solve the vanishing gradient problem common in standard RNNs by introducing memory cells and gating mechanisms (input, output, and forget gates) to retain long term dependencies in sequential data, such as time series or sentences. In the present work, a standard LSTM architecture was used, without pre-training.

2) fastText: Developed by Joulin *et al.* at Facebook AI (2016), fastText is built on the Word2Vec (word representation in a vectorial space) and extended it by incorporating subword information. Rather than representing each word as a single entity, it breaks it down to character n-grams. This allows to represent sentences by averaging word embeddings, making it very lightweight and fast to train on large datasets, with minimal tuning. The lightweightness and little tunability makes it less differenciate and harder to adapt to specific cases. In the present study, the standard supervised fastText implementation was used for sentence classification, without pre-trained embeddings.

3) Bidirectional Encoder Representations from Transformers: BERT was introduced by Devlin *et al.* and colleagues at Google (2018), and is a deep-transformer model pre-trained on large corpora using masked language modeling (hiding one word in the sentence for the model to predict) and next sentence prediction. BERT is capable of considering both forwards and backwards dependencies with a word, simultaneously. This allows for much better understanding of nuanced language patterns and semantics. Despite the higher computational requirements, it still is manageable at a local level, and benefits heavily from fine-tuning for specific NLP tasks. In this work, the `bert-base-uncased` variant was used as the base model.

A. Initial Benchmark

The initial models went through rounds of 5-fold cross validation, with the hyperparameter search spaces as indicated in Table III.

Table III: Hyperparameter search space and selected values for the initial models.

Hyperparameter	Search Space	Selected Value
Epochs	$\{20, 21, \ldots, 99\}$	88
Learning rate	$[10^{-5}, 10^{-2}]$	4×10^{-3}
Embedding dimension	$\{100, 200, 300\}$	200

(a) fastText hyperparameters.

Hyperparameter	Search Space	Selected Value
Epochs	$\{2, 3, 5, 8, 10, 15\}$	3
Learning rate	$[10^{-5}, 10^{-3}]$	10^{-4}
Embedding dimension	$\{32, 64, 128, 256\}$	128
LSTM units	$\{32, 64, 128, 256\}$	32
Dropout	$[0, 0.5]$	0.2
Recurrent dropout	$[0, 0.5]$	0

(b) LSTM hyperparameters.

Hyperparameter	Search Space	Selected Value
Epochs	$\{1, 2, 3, 4, 5\}$	2
Learning rate	$[10^{-5}, 10^{-2}]$	10^{-4}
Weight decay	$[0, 0.5]$	0.1

(c) BERT hyperparameters.

After fitting the models with the best hyperparameters, the models were trained on the complete training set, with the results in Table IV.

Table IV: Initial benchmark metric results across the models for both the training and test sets.

Model	Accuracy		F1 (macro)	
	Train	Test	Train	Test
fastText	0.54	0.65	0.45	0.44
LSTM	0.67	0.66	0.63	0.63
BERT	0.97	0.92	0.97	0.91

From the results, there is a clear gap between the models, with BERT performing best, followed by LSTM and fastText. These findings align with the literature, although LSTM can achieve better performance when using bi-LSTM. However, the purpose here was to evaluate *vanilla* models as an initial assessment.

Given BERT's superior performance, it was selected for more detailed analysis and further experimentation, with its evaluation metrics and class-wise error distribution examined below. From this point onward, this model is referred to as B-BERT.

From the learning curve (Fig. 4) the model seems to learn well, albeit the validation loss does increase slightly towards the end, overlapping the training curve.

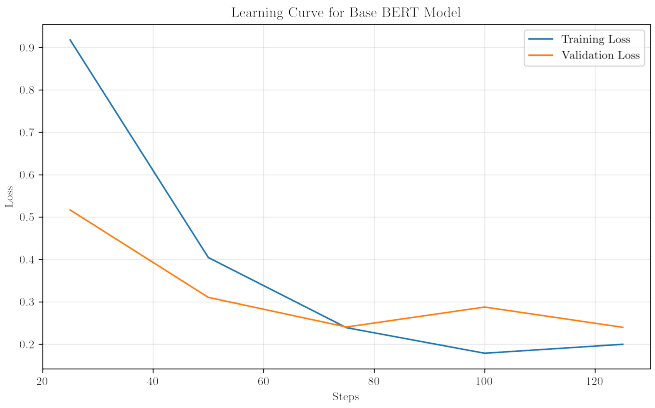

Figure 4: B-BERT learning curve.

From the confusion matrix and training metrics (Fig. 5 and Table V), the model seems to learn well, with no class suffering in particular in terms of performance.

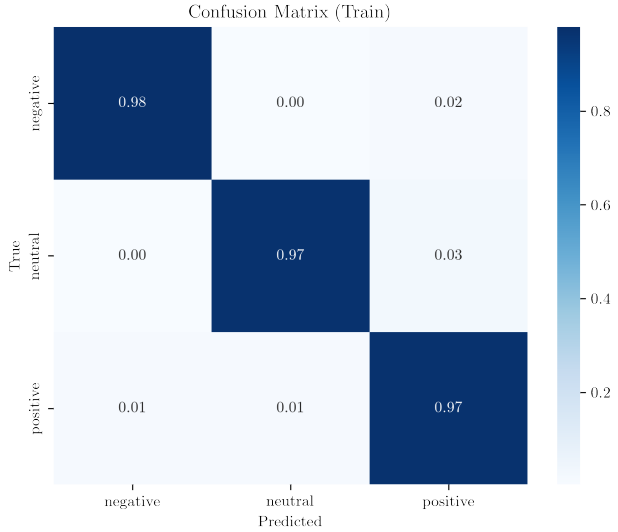

Figure 5: Normalized confusion matrix for the training set with the B-BERT model.

Table V: Classification report for B-BERT on training data.

Class	Precision	Recall	F1-Score	Support
Negative	0.98	0.98	0.98	336
Neutral	0.98	0.97	0.98	336
Positive	0.96	0.97	0.96	336
Accuracy			0.97	1008
Macro avg	0.97	0.97	0.97	1008
Weighted avg	0.97	0.97	0.97	1008

From the test results (Fig. 6 and Table VI), the model appears to be overfit, showing difficulties in generalizing and achieving performance comparable to the training set, with the macro average being significantly lower.

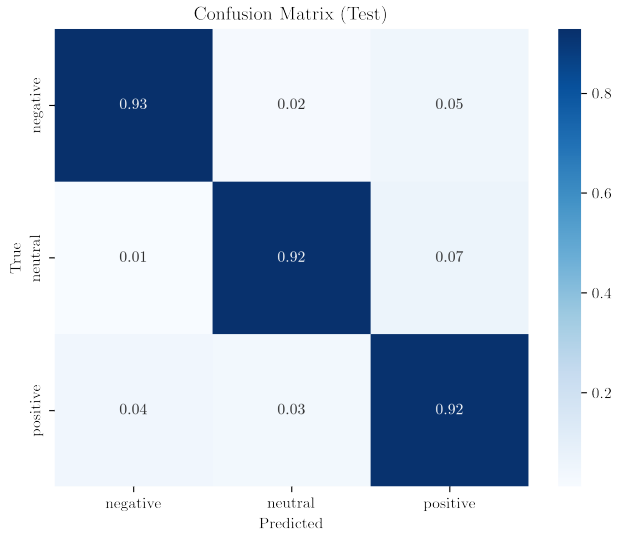

Figure 6: Normalized confusion matrix for the test set with the B-BERT model.

Table VI: Classification report for B-BERT on test data.

Class	Precision	Recall	F1-Score	Support
Negative	0.86	0.93	0.89	84
Neutral	0.98	0.92	0.95	429
Positive	0.84	0.92	0.88	178
Accuracy			0.92	691
Macro avg	0.89	0.92	0.91	691
Weighted avg	0.93	0.92	0.92	691

B. Data Augmented Model - DA-BERT

To improve the model's generalization capability, an online data augmentation strategy was implemented, consisting of back-translation (translation-based augmentation using intermediate pivoting paraphrasing, English to German and back), lexical substitution (where random words are replaced with WordNet-based synonyms), and template-based augmentation (by using named entity recognition, the identified words are replaced with template tokens such as ORG or DATE to generalize the sentence). Some examples of this augmentation are available in Table VII.

Table VII: Example sentences from the Financial PhraseBank and their augmented examples.

Original	Augmented
In the building and home improvement trade, sales decreased by 22.5% to EUR 201.4 mn.	In the building and DIY trade, sales decreased by 22.5% to EUR 201.4 million.
In January–June 2010, diluted loss per share stood at EUR 0.3 versus EUR 0.1 in the first half of 2009.	In the first half of 2009, diluted loss per share stood at EUR 0.3 versus EUR 0.1 in the same period of 2008.

The cross-validation procedure mentioned previously was executed, with the search space and selected values indicated

in Table VIII.

Table VIII: Hyperparameter search space for DA-BERT and selected values after fine-tuning.

Hyperparameter	Search Space	Selected Value
Epochs	$\{1, 2, 3, 4, 5\}$	2
Learning rate	$[10^{-5}, 10^{-2}]$	10^{-4}
Weight decay	$[0, 0.5]$	0.1

With this setup, the model was trained on the full training set, using the same online data augmentation pipeline. As shown in the learning curve (Fig. 7), the model was able to generalize well throughout the training routine, with validation performance slightly increasing between the 50^{th} and 80^{th} steps, then following a downward trend toward the end. Although the curve does not exhibit a stable or horizontal convergence, this behavior is due to an implicit early stopping effect governed by the number of epochs, which was treated as a hyperparameter during fine-tuning.

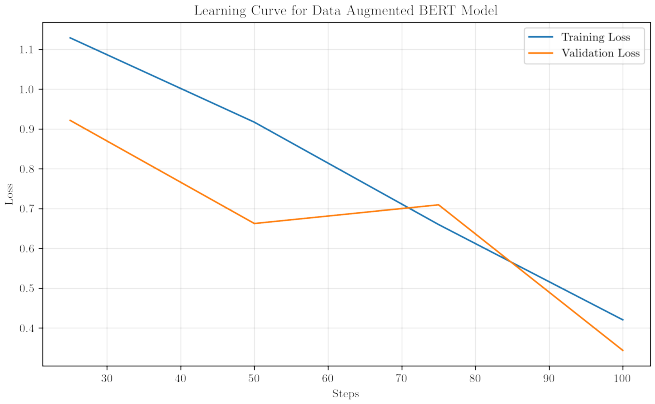

Figure 7: DA-BERT learning curve.

The confusion matrix from the training (Fig. 8) shows that the model performed well, but slightly worse in particular for the *Positive* class, which can also be confirmed from the classification report (Table IX), where the recall is considerably lower. The overall metrics (accuracy, macro average, and weighted average) indicate worse performance compared to the B-BERT model, though the comparison is not entirely fair given that B-BERT showed signs of overfitting.

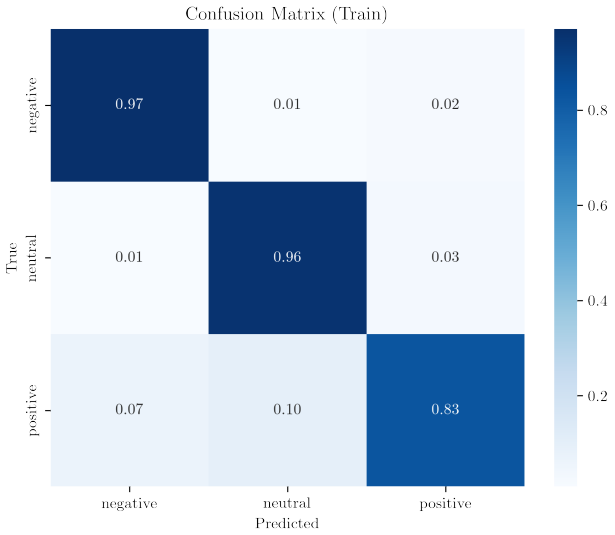

Figure 8: Normalized confusion matrix for DA-BERT model on training data.

Table IX: Classification report for DA-BERT on training data.

Class	Precision	Recall	F1-Score	Support
Negative	0.92	0.97	0.95	336
Neutral	0.90	0.96	0.93	336
Positive	0.94	0.83	0.88	336
Accuracy			0.92	1008
Macro avg	0.92	0.92	0.92	1008
Weighted avg	0.92	0.92	0.92	1008

From the test confusion matrix (Fig. 9) there is a slight decrease for the *Negative* and *Neutral* classes, whereas the *Positive* class keeps as is. The trend is confirmed from the classification report (Table X), showing slightly worse metrics than for the training data.

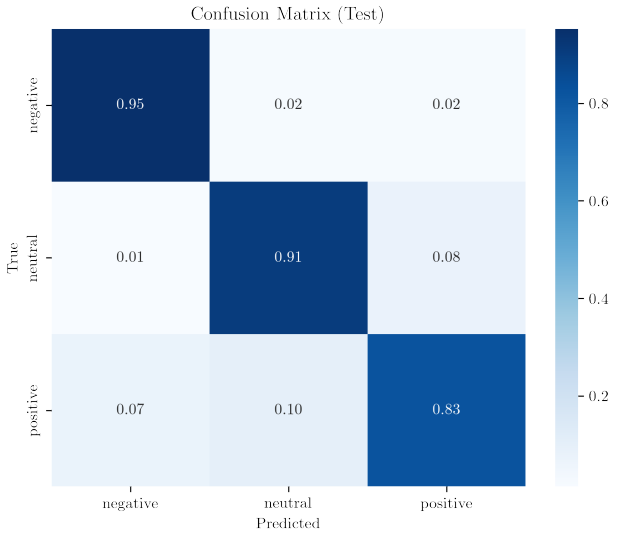

Figure 9: Normalized confusion matrix for DA-BERT model on test data.

Table X: Classification report for DA-BERT on test data.

Class	Precision	Recall	F1-Score	Support
Negative	0.81	0.95	0.87	84
Neutral	0.95	0.91	0.93	429
Positive	0.80	0.83	0.81	178
Accuracy			0.89	691
Macro avg	0.85	0.89	0.87	691
Weighted avg	0.90	0.89	0.89	691

The model seems to be well fit, despite slight differences between training and test metrics. These can be explained given the data augmentation pipeline, that pushes the model towards generalization, while losing more obvious patterns that are kept in the undisturbed test set.

C. Instance Weighted Model - W-BERT

For this model the approach aims to achieve a balance between quantity and quality of data. Rather than filtering documents based on the level of agreement, the goal was to introduce the level of agreement as penalty weights during training, with the model being more penalized for wrong classification of sentences with higher level of agreement and less so for the opposite cases.

After testing, the following formula was adopted for the agreement class weights, with $i \in \{50\%, 66\%, 75\%, 100\%\}$, where each value represents a level of annotator agreement associated with each sentence in the dataset.

$$w_i = \text{scale} \cdot p_i = \left(\frac{N}{\sum_{j=1}^{k} n_j \cdot p_j} \right) \cdot \left(\frac{e^{a_i}}{\sum_{j=1}^{k} e^{a_j}} \right),$$

where a_i is the raw agreement scores for class i, n_i the number of samples in class i, and N the total number of instances.

The rationale behind this formulation starts from a set of empirically defined raw agreement scores, a_i, for each agreement class:

$$\{0.50, 0.66, 0.75, 1.0\}$$

These values represent the degree of annotator consensus and are transformed exponentially using the softmax function,

$$p_i = \frac{e^{a_i}}{\sum_j e^{a_j}}$$

to capture differences in confidence in a non-linear manner.

Following this transformation, a normalization factor is applied to account for agreement imbalance in the training dataset,

$$\text{scale} = \frac{N}{\sum_i n_i \cdot p_i}$$

ensuring that the average weight across all training instances equals 1, preserving the comparability of the loss function with a weightless scenario, such as the test dataset.

This approach leverages confidence information from agreement levels without distorting the overall loss scale during training.

Considering the dataset for this model, and accounting for class balance, the training dataset for this model had 520 observations per class, rather than the 336 from earlier. The hyperparameter search space and selected can be found in table XI.

Table XI: Hyperparameter search space for W-BERT and selected values after fine-tuning.

Hyperparameter	Search Space	Selected Value
Epochs	$\{1, 2, 3, 4, 5\}$	3
Learning rate	$[10^{-5}, 10^{-2}]$	10^{-4}
Weight decay	$[0, 0.5]$	0.1

The learning curve for the instance-weighted training loss and the weightless validation loss (Fig. 10) shows that both training and validation losses decrease during the first half of training. Around the midpoint, the validation loss begins to increase slightly, while the training loss continues to decrease, indicating early signs of possible overfitting.

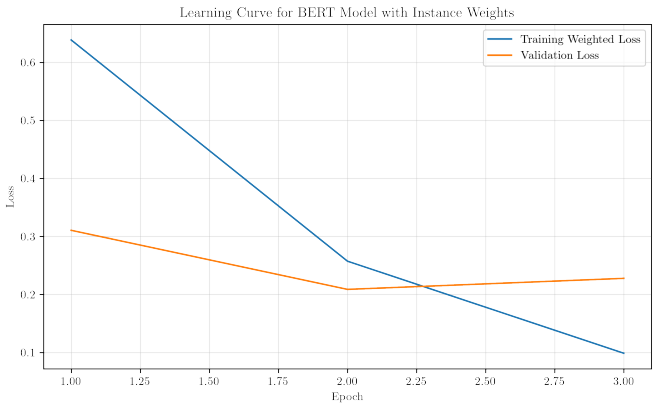

Figure 10: W-BERT learning curve.

For training metrics, only a portion of the training set was used, namely instances with at least 75% agreement. This choice was motivated by the difficulty of incorporating instance weights into the calculation of certain metrics and the potential bias introduced if weighting is ignored. The 75% agreement threshold was a natural selection, consistent with the threshold used throughout the study and for the test set.

The approach under analysis produced notable results. The confusion matrix for the partial training set (Fig. 11) and the classification report (Table XII) provide strong evidence that the model successfully learned to recognize the *Negative* class. The improvement observed in the *Positive* class supports the hypothesis that this category tends to exhibit lower annotator agreement, making it more difficult for models trained without weighting to capture.

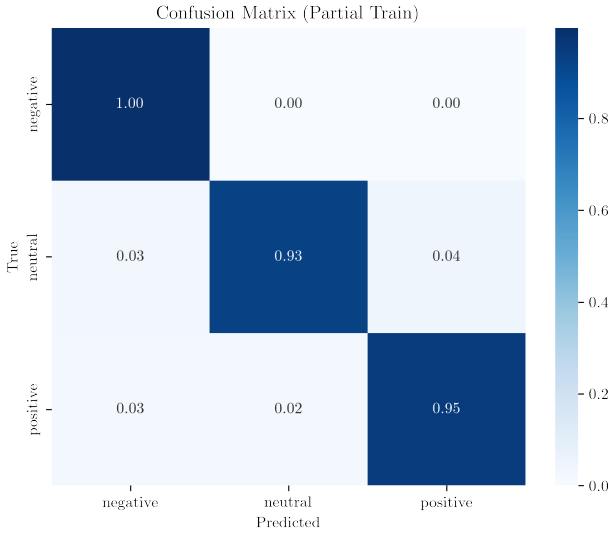

Figure 11: Normalized confusion matrix for W-BERT model on the partial training set.

Table XII: Classification report for W-BERT on the partial training set.

Class	Precision	Recall	F1-Score	Support
Negative	0.83	1.00	0.91	336
Neutral	0.99	0.93	0.96	1717
Positive	0.90	0.85	0.93	709
Accuracy			0.94	2762
Macro avg	0.91	0.96	0.93	2762
Weighted avg	0.95	0.94	0.94	2762

The confusion matrix on the test set (Fig. 12) and the classification report (Table XIII) show a slight decrease in performance metrics, but overall well fit to the data. From the confusion matrix there doesn't seem a considerable change in identifying the classes, nor a straightforward interpretation of any two classes.

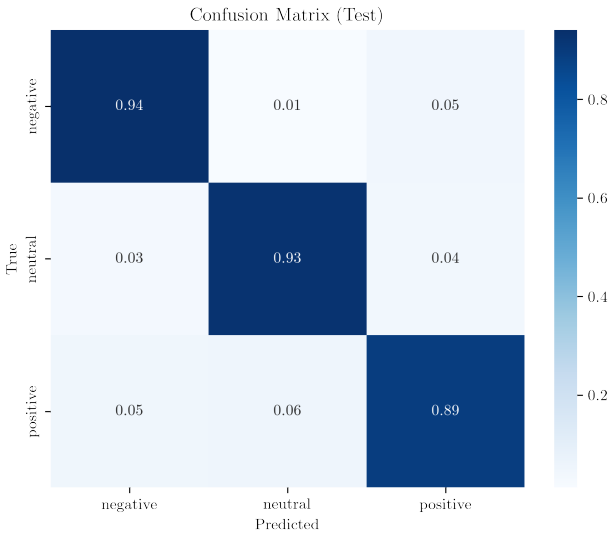

Figure 12: Normalized confusion matrix for W-BERT model on test data.

Table XIII: Classification report for W-BERT on test data.

Class	Precision	Recall	F1-Score	Support
Negative	0.78	0.94	0.85	84
Neutral	0.97	0.93	0.95	429
Positive	0.88	0.89	0.89	178
Accuracy			0.92	691
Macro avg	0.88	0.92	0.90	691
Weighted avg	0.93	0.92	0.92	691

These results may be further improved by implementing more precise methods for determining the raw agreement weight between annotators, whether through expert-driven refinement or fine-tuning. Additional gains could also be obtained by revisiting the weighting strategy applied during the training procedure.

V. Discussion

The metrics of the three BERT model variants are displayed in Fig. 13. At first glance, the B-BERT model slightly outperforms the other two, but only by narrow margins. Given that the test set likely shares significant similarities with the training set, and based on its detailed performance analysis, it is reasonable to conclude that the base model is overfitted and primarily suited to this specific setup. More importantly, the models designed for better generalization, DA-BERT and W-BERT, still perform well in this constrained scenario, with performance metrics closely matching those of the base model.

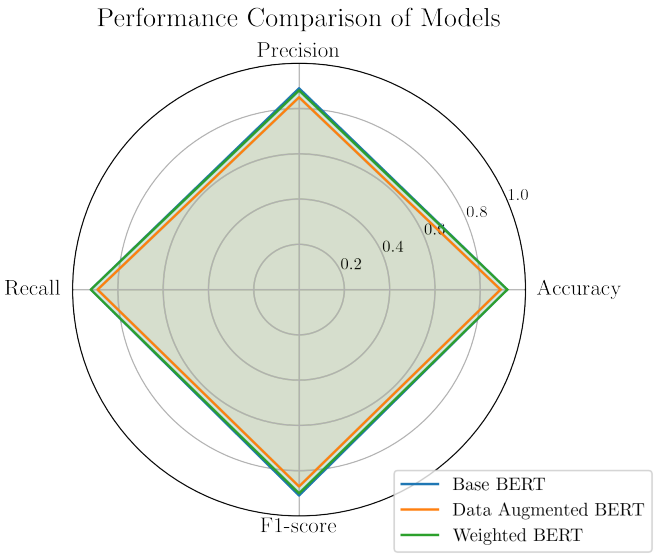

Figure 13: Radar chart of the macro performance metrics of the BERT models.

The models performed well in comparison with the literature, as shown in Table XIV. While the results vary depending on the agreement level of the test set, the trained models consistently fall within the benchmark range reported in previous studies.

Table XIV: Performance comparison on the test set between the trained models and literature baselines.

Model	Accuracy	F1 (macro)	Agreement Level
B-BERT	0.92	0.91	75%
DA-BERT	0.89	0.87	75%
W-BERT	0.92	0.90	75%
Sun *et al.* (2025)	-	0.98	100%
Sun *et al.* (2025)	-	0.87	50%
Atsiwo (2024)	0.89	0.88	50%
Choe (2023)	0.86	0.84	not reported
Araci (2019)	0.87	0.95	100%
Araci (2019)	0.86	0.84	50%
Malo (2014)	0.85	0.78	75%

VI. Conclusion

This work explored different machine learning and deep learning models to perform sentiment classification in financial statements. The approach was defined to assess different models, in order to further explore the best, considering literature's best practices. Among the tested models, BERT performed much better than fastText and LSTM, and was further developed by employing a data augmentation pipeline, and, separately, a weighted approach based on the agreement level. Both cases performed close to the initial model, but, due to improved generalization, showed slightly poorer performance metrics.

For future work, both strategies could be implemented simultaneously, with models tested on different datasets to minimize the impact of performance metrics being influenced by potential overfitting to the dataset. Additionally, the weighted approach requires further investigation, ideally through expert-informed methods, to reach a stage where the model can infer the level of confidence in its predictions based on sentence structure, wording, and other linguistic patterns.

Work Load

Both authors contributed equally to the project.

References

[1] P. C. Tetlock, "Giving content to investor sentiment: The role of media in the stock market," *The Journal of Finance*, vol. 62, no. 3, pp. 1139–1168, 2007.

[2] 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.

[3] 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, no. 4, pp. 782–796, 2014.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2019.

[5] D. Araci, "Finbert: Financial sentiment analysis with pre-trained language models," Master's thesis, Bogazici University, 2019.

[6] Y. Sun, H. Yuan, and F. Xu, "Financial sentiment analysis for pre-trained language models incorporating dictionary knowledge and neutral features," *Natural Language Processing Journal*, vol. 11, p. 100148, 2025.

[7] A. Atsiwo, "Financial sentiment analysis: Leveraging actual and synthetic data for supervised fine-tuning," 2024. [Online]. Available: https://arxiv.org/abs/2412.09859

[8] G. Fatouros, J. Soldatos, K. Kouroumali, G. Makridis, and D. Kyriazis, "Transforming sentiment analysis in the financial domain with chatgpt," *Machine Learning with Applications*, vol. 14, p. 100508, 2023.

[9] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[10] J. Choe, K. Noh, N. Kim, S. Ahn, and W. Jung, "Exploring the impact of corpus diversity on financial pretrained language models," *arXiv preprint arXiv:2310.13312*, 2023. [Online]. Available: https://arxiv.org/abs/2310.13312

[11] P. Malo, A. Sinha, P. Takala, P. Korhonen, and J. Wallenius, "Financialphrasebank-v1.0," 07 2013.