A Stacked Ensemble Approach Integrating FinBERT, Classical Machine Learning and LLMs for Financial News Sentiment Analysis*

Kartik M. Avhad† Ganesh Manjhi‡

Abstract

This study develops a hybrid ensemble framework for financial news sentiment classification that integrates FinBERT, classical Machine Learning models, and Large Language model-based sentiment features. FinBERT is fine-tuned to capture nuances of financial language, while Logistic Regression, Random Forest, Decision Tree, Naïve Bayes, Support Vector Machine, and Extreme Gradient Boosting are trained on Term Frequency-Inverse Document Frequency representations to capture lexical frequency patterns. Model outputs are combined through weighted soft voting and stacked meta-learning architectures. Results show that hybrid integration of contextual, statistical, and generative representations yields superior robustness and interpretability in financial sentiment modeling. The proposed stacked ensemble approach outperforms other models, achieving 78.21% accuracy and a Sharpe Ratio of 2.910 (normal returns), while the fine-tuned FinBERT achieves 64.74% accuracy and a Sharpe Ratio of 1.514, an improvement over the Baseline FinBERT, which achieves 48.72% accuracy and a Sharpe Ratio of 0.409. Another LLM, GPT-4o, corresponds to an accuracy of 76.28% and a Sharpe Ratio of 2.484. The accuracy of the average of Classical Machine Learning is 55.13% with a Sharpe Ratio of -0.506.

JEL Classification: C45; C55; G12; G41.

Keywords: Sentiment Analysis; Textual Data; Classical Machine Learning; FinBERT; Large Language Models.

Acknowledgement: Authors are immensely grateful to the discussant, Dr Aicha Bareche, for her insightful comments and thorough reviews in the Financial Economics Meeting (FEM) 2025 Conference at EDC Business School, Paris, $15 - 17^{th}$ December 2025, Dr Antonis Ballis and Dr Sarra Ghadab for a detailed review, readings, and the feedback. We are grateful to the audience for their comments, feedback, and discussion at the FEM Conference, and the 7^{th} Annual Economics Conference at Ahmedabad University, $9 - 10^{th}$ January 2026.

†Centre for Mathematical and Computational Economics (CMCE), School of Artificial Intelligence and Data Science, Indian Institute of Technology Jodhpur, Rajasthan - 342030, India. Email: r24am0001@iitj.ac.in

‡Centre for Mathematical and Computational Economics (CMCE), School of Artificial Intelligence and Data Science, Indian Institute of Technology Jodhpur, Rajasthan - 342030, India. Email: ganeshtrx@gmail.com

1 Introduction

In the modern era of financial modeling, where technology and information flow continuously, there has been a significant increase in the number of text data sources, such as news articles, tweets, social media posts, and quarterly reports, from which useful information can be extracted. The text-to-data approach offers alternative methods, greater predictability, and improved precision for various scientific questions, including economics and finance. The use of Natural Language Processing (NLP) in this domain has predominantly focused on central bank communications, social media sentiment, business cycle detection, and related dynamics of the financial markets (Manjhi and Shah, 2025). There is significant untapped potential in analysing textual data to understand asset markets. Acemoglu et al. (2022) states that the text analysis in economics and finance is expected to expand in the future. In the wake of Large Language Models (LLMs), the study of text data for sentiment analysis has grown rapidly. For example, the study of sentiments in financial markets - Tumarkin and Whitelaw (2001), Baker and Wurgler (2006), Savor and Wilson (2013), Nguyen et al. (2015), Azar and Lo (2016), Ruan et al. (2018), Derakhshan and Beigy (2019), Díaz et al. (2024), etc. This paper aims to develop a novel technique for generating sentiment predictions for news articles. This involves harnessing classical ML models and LLMs, namely fine-tuned Financial Bidirectional Encoder Representations from Transformers (FinBERT), GPT-4o, and a novel stacking-based ensemble technique. This is followed by a comparison of techniques with our proposed framework. Various traditional models have used NLP for sentiment analysis (Kirtac and Germano, 2024; Kamal et al., 2022; Huang et al., 2023; Loughran and McDonald, 2011; Bochkay et al., 2023). Out of these, Huang et al. (2023) has made use of FinBERT, a cutting-edge LLM customised for financial texts based on Google's Bidirectional Encoder Representations from Transformers (BERT) algorithm.

Early computational approaches to financial discourse were mostly lexicon-based and focused on the "bag-of-words" assumption, which ignores grammar, syntax, and context in favor of treating a document as an unordered collection of words (Wallach, 2006; Huang et al., 2023). The Loughran-McDonald (LM) Master Dictionary was a foundational work of this period. The LM lexicon was developed specifically for the financial industry, in contrast to generic word lists (Loughran and McDonald, 2011). However, it frequently leads to incorrect sentiment classification by ignoring word order and context, thereby failing to account for sarcasm, irony, and other linguistic structures. In this context, the use of traditional supervised machine learning models marked a shift in the development of financial text analysis. These models are an improvement over static dictionary lookups since they can learn intricate patterns from labelled input (Wallach, 2006; Huang et al., 2023). Models such as Random Forests, Support Vector Machines (SVMs), and Naïve Bayes have become the contemporary norm in both commercial and academic research (Loughran and McDonald, 2011). Learning the statistical correlations between word patterns and specific outcomes (e.g., positive or negative sentiment) enabled these

approaches to achieve superior results. The Decision Tree and its variants, such as Extreme Gradient Boosting (XGBoost), have also been utilised for sentiment analysis (Hama Aziz and Dimililer, 2021). However, the basic flaw remained the bag-of-words paradigm, which is essentially incompatible with the sequential and contextual aspects of human language.

The use of deep learning algorithms for processing sequential data was the first significant break from the word-independence presumption that had limited earlier approaches. While Recurrent Neural Networks (RNNs), especially Long Short-Term Memory (LSTM) networks, were used to capture long-range dependencies and contextual information across sentences, Convolutional Neural Networks (CNNs) were modified to extract local patterns and features from text, such as n-grams, where Deep learning models routinely outperformed their classical ML counterparts (Bochkay et al., 2023). The disadvantage, though, was that they required a large amount of labelled data for training, which can be expensive and difficult to generate in specialised fields like finance. Rozental and Fleischer (2017) presented two supervised training methods – (i) RNNs, and (ii) Feedforward Neural Network, Naïve Bayes and Logistic Regression and also used the stacking method.

The Transformer architecture proposed in Vaswani et al. (2017), which later helped design BERT, marked a significant change in NLP. Further, Pre-training and fine-tuning are two refined steps in the BERT system. Pre-training requires a vast corpus of unlabeled data, while fine-tuning uses a smaller dataset to adapt the previously trained model. The introduction of LLMs and BERT marked a shift in the financial language from an analysis of mere collections of words to an ability to comprehend linguistic context, such as negation, sarcasm, and domain-specific jargon, which frequently confuses lexicon-based and traditional ML models and hence advances the earlier approaches (Devlin et al., 2019; Huang et al., 2023; Kim et al., 2026). LLMs like GPT 4 are trained on a large set of data, therefore the predictions generated are at par with the traditional models and generates a higher sharpe ratio and other alphas (Kim et al., 2024, 2026). Ensemble techniques combine multiple learning algorithms, particularly for classification tasks (Polikar, 2012; Jain et al., 2022). The stacked ensemble techniques are among the methods used to improve prediction accuracy (Wolpert, 1992; Xia et al., 2021).

Based on the aforementioned text-to-data literature and methodical advancements, this paper addresses the following questions: (i) whether fine-tuning a FinBERT improves the accuracy compared to FinBERT? (ii) whether the GPT-based LLM model can outperform classical ML and the fine-tuned FinBERT models? (iii) whether a novel ensemble approach can be designed such that it outperforms all the mentioned techniques? To address these questions, this paper explores Classical Machine Learning methods, such as Naïve Bayes, Decision Tree, and Random Forest, along with two LLMs, namely FinBERT and OpenAI's GPT-4o (Face, 2023; Achiam et al., 2023), and it further aims to design a novel ensemble technique to capture the news sentiments and their accuracy along with effectively addressing inadequacies of individual models. Based on the methods applied, the basic results are as follows,

- The stacked ensembled framework is the best-performing model, achieving 78.21% accuracy and the corresponding F1-Macro is 77.89%. The Stacked ensemble approach outperforms the 2^{nd} best weighted ensemble by the margin of 0.84% and 1.50% in accuracy and Macro-F1, respectively. The weighted ensemble's corresponding accuracy and Macro-F1 are 77.56% and 76.74%, respectively.
- The LLM (GPT-4o) is the 3^{rd} best-performing approach behind weighted ensemble and stacked ensemble with accuracy of 76.28% and the corresponding Macro-F1 of 75.61%. Note that the 2^{nd} best performing weighted ensemble outperforms the LLM(GPT-4o) by the margin of 1.67% and 1.49% in accuracy and Macro-F1, respectively.
- The fine-tuned FinBERT achieves 64.74% accuracy, an improvement over the Baseline FinBERT, which records 48.72%. The fine-tuned FinBERT approach improves accuracy by 32.9% and F1-Macro by 33.5%.
- The average accuracy of Classical Machine Learning models is 55.13% with a corresponding Macro-F1 of 52.46%. This method performs better than basic FinBERT, but fine-tuned FinBERT outperforms it.
- The proposed stacked ensemble technique outperforms its counterparts with regard to Sharpe Ratio and Mean Daily Return. The Newey-West HAC-corrected value confirms that the stacked ensemble's mean return is statistically significant, indicating that the ensemble strategy's profitability is not driven by random fluctuations but reflects systematic predictive power embedded in sentiment signals.

The basic results of this paper enrich the existing literature, such as Kamal et al. (2022), Kirtac and Germano (2024), Kim et al. (2026). Based on the techniques devised, the calculated sentiment can inform better investment decisions. The remainder of this paper is as follows. Section 2 covers the Data and Methods, followed by the result analysis and conclusion in Sections 3 and 4, respectively.

2 Data and Methods

2.1 Data

The dataset used in this study comprises financial news articles for five major publicly listed firms: Tesla (TSLA), Apple (AAPL), GameStop (GME), Costco (COST), and Palantir (PLTR), representing diverse sectors of the equity market. The tech giants Apple and Tesla were chosen due to their market capitalization ranked as 2^{nd} and 9^{th} respectively, with the former with a market capitalization of $3.8 trillion and the latter with a market capitalization of $1.4 trillion as of January 2026 (Daly, 2026). A relatively small company, GME was chosen to ensure portfolio diversity and prevent the portfolio from being dominated by giants. COST, a company that operates only big-box warehouse retail stores, was chosen to

maintain sectoral heterogeneity. PLTR has been one of the top-performing stocks over the past few months, so it was selected because more articles were available for scraping. The articles were collected via multiple reputable sources, including NewsAPI.org, Yahoo Finance RSS feeds, Finhub.io, ensuring both breadth and temporal diversity. Social media data was not included in the dataset due to imbalance bias and lack of authenticity, unorganized, noisy, cumbersome, and irrelevant, which does not provide textual emotional content. (Rehman et al., 2019; Agüero-Torales et al., 2019; Salur and Aydin, 2020; Priyadarshini and Cotton, 2021; Es-Sabery et al., 2021; Jain et al., 2022)

Over 1,500 raw articles were initially captured using automated scraping pipelines. Given the inherently noisy nature of scraped news data, a multi-stage preprocessing pipeline was utilized to ensure quality and relevance.

i **Relevance filtering:** Only the articles explicitly mentioning a firm's name or ticker in the headline or main text were retained.

ii **Deduplication:** Near-duplicate content (cosine similarity > 0.8) was removed to avoid repeated reporting of the same event.

iii **Length threshold:** Articles with texts shorter than 50 characters were discarded.

iv **Language normalization:** Non-English texts were translated to English via the googletrans API for consistency.

After deploying the pre-processing pipeline, the final corpus had 777 unique, relevant articles, with sentiment manually verified and categorized as Positive, Neutral, or Negative. The articles span a period of 79 calendar days, from 17 July 2025 to 04 October 2025, encompassing a total of approximately 11 weeks of financial news coverage.

2.2 Inter-Annotator Agreement

To construct a reliable gold-standard sentiment dataset in the NLP context, it is important to construct an annotated collection of text (Wissler et al., 2014). Hence, two domain-informed annotators, independently annotated the dataset. Each article was labeled into one of three sentiment categories: *Negative*, *Neutral*, or *Positive*, based on its expected directional impact on the given firm's stock performance.

To evaluate the reliability of annotation, Inter-Annotator Agreement was measured using Cohen's Kappa coefficient, which accounts for agreement occurring by chance. Cohen's Kappa is defined as:

$$\kappa = \frac{P_o - P_e}{1 - P_e}, \tag{1}$$

where P_o denotes the observed proportion of agreement between annotators and P_e represents the probability of agreement expected by random chance.

The computed Cohen's Kappa score $\kappa = 0.83$ indicates substantial agreement between annotators, and also demonstrates the consistency and credibility of the labeling process. A consensus-based gold label was assigned to each article to construct the final dataset used for model training and evaluation.

Table 1: Number of articles and sentiment distribution across firms.

Ticker	Positive	Neutral	Negative	Total
TSLA	79	53	55	187
AAPL	62	52	25	139
GME	54	54	23	131
COST	60	43	26	129
PLTR	80	46	65	191
Total	**335**	**248**	**194**	**777**

Table 1 summarizes the number of articles and sentiment distribution across firms. This moderate-sized yet curated dataset serves as a proof-of-concept corpus for multi-firm sentiment modeling under conditions where data is scarce, typical of emerging financial research contexts.

2.3 Methods

The methodological pipeline follows a three-stage process combining domain-specific transformers, classical machine learning models, and ensemble integration.

Stage 1 - Text Representation and Model Fine-Tuning

FinBERT, which is a BERT-based transformer pretrained on financial corpora (Araci, 2019) is employed. As the checkpoint's native label ordering ({0: positive, 1: negative, 2: neutral}) differs from our canonical scheme ({0: Negative, 1: Neutral, 2: Positive}), all output logits are permuted to the canonical order before computing probabilities. To improve generalization on a small dataset, FinBERT is further fine-tuned using 5-fold stratified cross-validation on the 621-article training set. Each fold used approximately 80% of the training data (≈497 rows) for training and 20% (≈124 rows) for validation. Training was performed with:

- learning rate $= 2 \times 10^{-5}$
- batch size $= 8$
- sequence length $= 256$
- epochs $= 5$
- weight decay $= 0.01$
- warmup ratio $= 0.1$
- mixed-precision (fp16) training on GPU

The fine-tuned model generated out-of-fold (OOF) probability vectors for stacking: each training sample's probability vector was predicted by the fold that excluded it, ensuring no data leakage into the meta-learner. The test-set probabilities were produced by averaging the predictions from all five-fold models (fold-bagging). For comparison, a baseline pre-trained FinBERT (without fine-tuning) was retained alongside the fine-tuned variant.

Stage 2 - Classical ML Baselines

To benchmark transformer performance, six classical classifiers are trained on Term Frequency Inverse Document Frequency (TF-IDF) features. TF-IDF converts each article into a numerical vector in which each dimension corresponds to a vocabulary term; the weight assigned to a term reflects how frequently it appears in the article, downweighted by how common it is across the entire corpus.

This representation rewards distinctive, informative words while suppressing common terms such as "the" or "a" that carry little discriminative signal. The six classifiers trained on these features are:

- **Logistic Regression:** For multiclass classification, Logistic Regression models the probability of class c using the softmax function:

$$P(y = c \mid \mathbf{x}) = \frac{\exp(\mathbf{w}_c^\top \mathbf{x})}{\sum_{k=1}^{K} \exp(\mathbf{w}_k^\top \mathbf{x})}, \tag{2}$$

  where $\mathbf{w}_c$ is the weight vector associated with class c. Model parameters are learned by minimizing the regularized negative log-likelihood:

$$\mathcal{L}(\mathbf{W}) = -\sum_{i=1}^{N} \log P(y_i \mid \mathbf{x}_i) + \lambda \|\mathbf{W}\|_2^2, \tag{3}$$

where λ controls the strength of L2 regularization, a standard technique that penalizes large parameter values to discourage overfitting and improve generalization to unseen data.

- **Random Forest:** Random Forest aggregates multiple decision trees trained on bootstrap samples and random feature subsets. For B trees with predictions $h_b(\mathbf{x})$, the ensemble prediction is:

$$\hat{y}_{RF}(\mathbf{x}) = \underset{b\in\{1,\ldots,B\}}{\text{mode}} \left(h_b(\mathbf{x})\right). \tag{4}$$

  Randomness in sampling and feature selection reduces variance and mitigates overfitting compared to a single Decision Tree.

- **Decision Tree:** A decision tree partitions the feature space by selecting splits that minimize impurity. Using Gini impurity, the quality of a split producing left (L) and right (R) child nodes is:

$$G_{\text{split}} = \frac{N_L}{N} G_L + \frac{N_R}{N} G_R, \tag{5}$$

  where G_L and G_R are the Gini impurity values of the left and right child nodes, respectively (measuring class heterogeneity within each node after the split), N_L and N_R are the corresponding node sizes, and $N = N_L + N_R$ is the parent node size. The Gini impurity for a node is:

$$G = 1 - \sum_{k=1}^{K} p_k^2, \tag{6}$$

  with p_k the proportion of class k samples in the node. A lower G indicates greater purity (i.e., most samples in the node belong to the same class).

- **Multinomial Naïve Bayes:** The Multinomial Naïve Bayes classifier assumes conditional independence of features and models documents as bags-of-words following a multinomial distribution. For a document represented by a word-count vector $\mathbf{x} = (x_1, x_2, \ldots, x_d)$, the posterior probability for class c is:

$$P(c \mid \mathbf{x}) \propto P(c) \prod_{j=1}^{d} P(w_j \mid c)^{x_j}, \tag{7}$$

  where $P(w_j \mid c)$ is estimated using Laplace smoothing:

$$P(w_j \mid c) = \frac{n_{jc} + \alpha}{\sum_{k=1}^{d} n_{kc} + \alpha d}, \tag{8}$$

where n_{jc} denotes the raw count of word j in all training documents belonging to class c; n_{kc} in the denominator similarly denotes the count of word k in class c, so the sum $\sum_{k=1}^{d} n_{kc}$ gives the total word count in class c across the entire vocabulary; d is the vocabulary size; and α is the Laplace smoothing parameter (typically $\alpha = 1$) that adds a small pseudo-count to every word to avoid zero probabilities for words unseen during training.

- **Support Vector Machine (SVM):** We use SVM as a robust margin-based baseline for financial sentiment classification, in addition to probabilistic linear classifiers. The linear SVM solves the following optimization problem in order to learn class-specific weight vectors $\{w_c\}_{c=1}^{K}$ given a TF-IDF feature representation $x \in \mathbb{R}^d$:

$$\min_{W} \ \frac{1}{2} \sum_{c=1}^{K} \|w_c\|^2 + C \sum_{i=1}^{N} \max\bigl(0, 1 - w_{y_i}^{\top} x_i\bigr), \tag{9}$$

where y_i represents the true sentiment label of the document i and C regulates the trade-off between margin maximization and classification error. Since SVMs do not intrinsically produce probabilities, class scores are converted into calibrated probability estimates using Platt scaling, with balanced class weights.

- **Extreme Gradient Boosting (XGBoost):** To capture nonlinear interactions among textual features, we further incorporate XGBoost. XGBoost has stronger predictive power and is approximately 10 times faster than the Gradient Tree Boosting algorithm (Dhaliwal et al., 2018). For a given input x_i, the model prediction is expressed as:

$$\hat{y}_i = \sum_{t=1}^{T} f_t(x_i), \quad f_t \in \mathcal{F}, \tag{10}$$

where each f_t denotes a regression tree belonging to the functional space $\mathcal{F}$. For multiclass sentiment classification, XGBoost minimizes the regularized softmax loss:

$$\mathcal{L} = \sum_{i=1}^{N} \ell\bigl(y_i, \hat{y}_i\bigr) + \sum_{t=1}^{T} \Omega(f_t), \tag{11}$$

with regularization term

$$\Omega(f) = \gamma |T| + \frac{1}{2} \lambda \|w\|^2, \tag{12}$$

where $|T|$ is the number of leaves in a tree and w denotes leaf weights. This formulation enables XGBoost to model complex, nonlinear sentiment patterns while controlling overfitting through structural regularization. The implementation uses 300 estimators, a max depth of 6, a learning rate of 0.1, and a multi-class softmax loss.

Crucially, the TF-IDF vector was fitted only to the training rows of each fold and then used to transform the validation fold and the held-out test set, preventing any information leakage from the validation or test data into the feature space. Classical models followed the same 5-fold cross-validation protocol as FinBERT, producing OOF probabilities for stacking and fold-bagged test probabilities.

Stage 3 - LLM Sentiment Scoring

The dataset is supplemented with LLM-based sentiment predictions generated by GPT-4o. Each article is truncated to 6,000 characters and presented to the model via a zero-shot prompt that instructs it to classify the expected directional impact on the firm's stock as exactly one of *Negative*, *Neutral*, or *Positive*.

Crucially, the implementation requests token-level log-probabilities (`logprobs=True`, `top_logprobs=5`) from the OpenAI API. For each response, the top-5 logprobs at the first content token position are inspected for tokens matching the three canonical labels (case-insensitive, allowing leading whitespace). The extracted log-probabilities are then normalized via softmax:

$$P_{LLM}(c \mid x) = \frac{\exp(\ell_c)}{\sum_{k=1}^{3} \exp(\ell_k)}, \tag{13}$$

where ℓ_c denotes the log-probability for label c. When a label token is absent from the top-5, a floor value $\ell = -15$ is assigned. If logprobs are unavailable (e.g., due to an API error), the LLM output falls back to a one-hot representation:

$$P_{LLM}(c \mid x) = \begin{cases} 1, & c = \hat{y}_{LLM}, \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

This continuous probability representation provides substantially richer information for stacking than discrete one-hot labels.

Stage 4 - Ensembling

Using outputs from the fine-tuned FinBERT, classical ML baselines, and the GPT-4o model, a novel technique is derived through an ensemble process. LLM probabilities are integrated as an additional feature source for ensemble aggregation, alongside FinBERT and classical ML baselines.

2.4 Ensemble Design

Two ensemble frameworks were evaluated.

i **Weighted Soft Voting (WSV):** Probabilities from three component groups: fine-tuned FinBERT, the average of the six classical models, and GPT-4o were aggregated via weighted average. The weights were optimised by sampling 5,000 weight vectors from a symmetric Dirichlet(1,1,1) distribution and selecting the vector that maximised validation (OOF) macro-F1.[1]

ii **Stacked Meta-Learning (SML):** Out-of-fold probability vectors from all eight base models (fine-tuned FinBERT, six classical models, and GPT-4o) were concatenated as input features for a meta-learner, yielding a 24-dimensional feature vector (8 models $\times$ 3 classes) per sample. The meta-learner was implemented as Logistic Regression ($C = 1.0$, liblinear solver, balanced class weights).

All experiments were run in a single-GPU (Tesla T4) Google Colab environment using the PyTorch and scikit-learn libraries. Evaluation metrics included Accuracy, Precision, Recall, F1-score, and Macro-F1, alongside Confusion Matrices for qualitative error inspection.

2.4.1 Mathematical Formulation

The ensemble framework integrates probabilistic classifiers under a unified optimization scheme. Each base model $m \in \{1, \ldots, M\}$ outputs a probability distribution over sentiment classes $k \in \{1, 2, 3\}$, denoted as $P_m(y = k \mid \mathbf{x})$.

FinBERT and Classical Model Objectives: Both FinBERT and the classical classifiers minimize the categorical cross-entropy loss:

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} y_{i,k} \log P(y_i = k \mid \mathbf{x}_i) \tag{15}$$

For FinBERT, each input text is represented by a contextual embedding vector $\mathbf{h}_i$ (the [CLS] token), and the class probabilities are obtained as:

$$P(y = k \mid \mathbf{h}_i) = \text{softmax}(\mathbf{W}\mathbf{h}_i) \tag{16}$$

[1] Optimal soft-vote weights (OOF macro-F1 maximising): FinBERT FT ≈ 0.0001, Classical Average ≈ 0.300, LLM GPT-4o ≈ 0.700. The near-zero weight on fine-tuned FinBERT under linear combination indicates that the LLM channel subsumes most of the transformer's discriminative signal; the classical average retains a non-trivial weight ($\approx 30\%$), suggesting that TF-IDF-derived lexical features provide complementary, non-redundant information relative to the GPT-4o probability distribution. This weight asymmetry does not imply that FinBERT is uninformative in an absolute sense: the stacked ensemble, which uses all eight base-model probability vectors as meta-features, achieves higher accuracy than the soft-vote ensemble, demonstrating that FinBERT's contribution is captured by a non-linear meta-learner even when a linear combination assigns it near-zero weight.

Weighted Soft-Voting Ensemble: In the weighted soft-voting ensemble, class probabilities from each of the three component groups are linearly combined:

$$P_{ens}(y = k \mid \mathbf{x}) = \sum_{m=1}^{M} w_m P_m(y = k \mid \mathbf{x}) \tag{17}$$

subject to the normalization constraints:

$$\sum_{m=1}^{M} w_m = 1, \quad w_m \geq 0. \tag{18}$$

The final sentiment prediction is determined as:

$$\hat{y} = \arg\max_k P_{ens}(y = k \mid \mathbf{x}), \tag{19}$$

where the optimal weights $\mathbf{w}$ are selected to maximize validation OOF macro-F1 via Dirichlet sampling.

Stacked Meta-Learning Ensemble: In the stacked configuration, a meta-learner f_{meta} maps base model probabilities to the final prediction. For each instance i:

$$\mathbf{z}_i = [P_1(y|\mathbf{x}_i), P_2(y|\mathbf{x}_i), \ldots, P_M(y|\mathbf{x}_i)], \tag{20}$$

$$\hat{y}_i = f_{\text{meta}}(\mathbf{z}_i). \tag{21}$$

Here, f_{meta} is implemented as Logistic Regression, learning linear dependencies among base models. LightGBM was also tested as an alternative, but yielded lower accuracy on this dataset. [2]

2.5 Evaluation Metrics:

We evaluate the performance of each model in our analysis using Accuracy and Macro-F1:

$$\text{Accuracy} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(\hat{y}_i = y_i), \tag{22}$$

$$F1_{macro} = \frac{1}{K} \sum_{k=1}^{K} \frac{2 \times Precision_k \times Recall_k}{Precision_k + Recall_k}, \tag{23}$$

[2] LightGBM was also evaluated as an alternative meta-learner but achieved lower performance on this dataset (71.15% accuracy vs. 78.21% for Logistic Regression), suggesting that the limited meta-feature dimensionality (24 features) favours a simpler, well-regularised linear meta-learner over a tree-based approach.

where $Precision_k = \frac{TP_k}{TP_k+FP_k}$ and $Recall_k = \frac{TP_k}{TP_k+FN_k}$, with TP, FP, and FN denoting True Positive, False Positive, and False Negative counts, respectively.

Precision measures the proportion of articles predicted as class k that are truly class k (i.e., how reliable the model's positive predictions are), while *Recall* measures the proportion of true class-k articles that the model correctly identifies (i.e., how completely the model captures each class). Macro-averaging computes these quantities independently for each class and takes the unweighted mean, ensuring that performance on minority classes (e.g., *Negative*) receives equal weight to that on the majority class (Sokolova and Lapalme, 2009).

Sharpe Ratio The Sharpe ratio is a fundamental measure of an investment's risk-adjusted return. It helps identify whether the excess portfolio returns are due to intelligent decisions or to taking on high risk (Sharpe et al., 1998). Mathematically, the Sharpe Ratio, measuring risk-adjusted return, is defined as:

$$\text{Sharpe Ratio} = \frac{\mathbb{E}[R_p - R_f]}{\sigma_p}, \tag{24}$$

where R_p denotes portfolio returns, R_f is the risk-free rate (set to zero in our implementation), and σ_p is the standard deviation of returns. Annualisation is performed by multiplying by $\sqrt{252}$.

Maximum Drawdown The Maximum Drawdown measures the largest decline in cumulative returns from a portfolio's peak to its trough before a new peak is reached. It signals the "downside risk" in a given time period (Magdon-Ismail and Atiya, 2004). Maximum Drawdown (MDD) is defined as:

$$\text{MDD} = \max_{t\in[0,T]} \left(\frac{V_{\text{peak}} - V_t}{V_{\text{peak}}} \right), \tag{25}$$

where V_{peak} is the maximum portfolio value prior to time t.

Newey-West HAC t-test Because the 5-day event window returns overlap temporally, standard t-statistics are biased. We therefore report Newey-West HAC-corrected t-statistics (maxlags $= 5$) to account for serial correlation induced by the overlapping windows.

2.6 Model Architecture and Experimental Setup

The proposed ensemble framework is organized into three interconnected layers: (i) feature representation, (ii) base modeling, and (iii) ensemble fusion as shown in Figure 1. This multi-stage structure captures both contextual and statistical

sentiment cues from financial news.

(a) Feature Representation Layer: Raw text is preprocessed through tokenization, stop-word removal, and lemmatization. Two representations are generated: contextual embeddings from FinBERT and term-frequency–inverse-document-frequency (TF–IDF) features for classical models.

(b) Base Modeling Layer: Three model families are trained. FinBERT is fine-tuned using five-fold cross-validation, minimizing cross-entropy loss. Classical algorithms (Logistic Regression, Random Forest, Decision Tree, Naïve Bayes, SVM and XGBoost) are trained on TF–IDF vectors. GPT-4o provides an independent LLM-based sentiment channel.

(c) Ensemble Fusion Layer: Two ensemble mechanisms are implemented. In the weighted soft-voting scheme, probabilities from three component groups (fine-tuned FinBERT, classical average, and GPT-4o) are linearly combined with weights optimised on OOF macro-F1 via Dirichlet sampling. In the stacked meta-learning setting, concatenated base-model probabilities from all eight individual models (24 features) are used as input features for a Logistic Regression meta-learner f_{meta}.

(d) Evaluation Setup: Performance is measured using accuracy and macro-averaged F1, supported by confusion matrices, per-ticker breakdowns, and financial performance metrics including Sharpe Ratio, Maximum Drawdown, and Newey-West HAC-corrected statistical significance tests.

Figure 1: Conceptual architecture of the proposed ensemble sentiment framework.

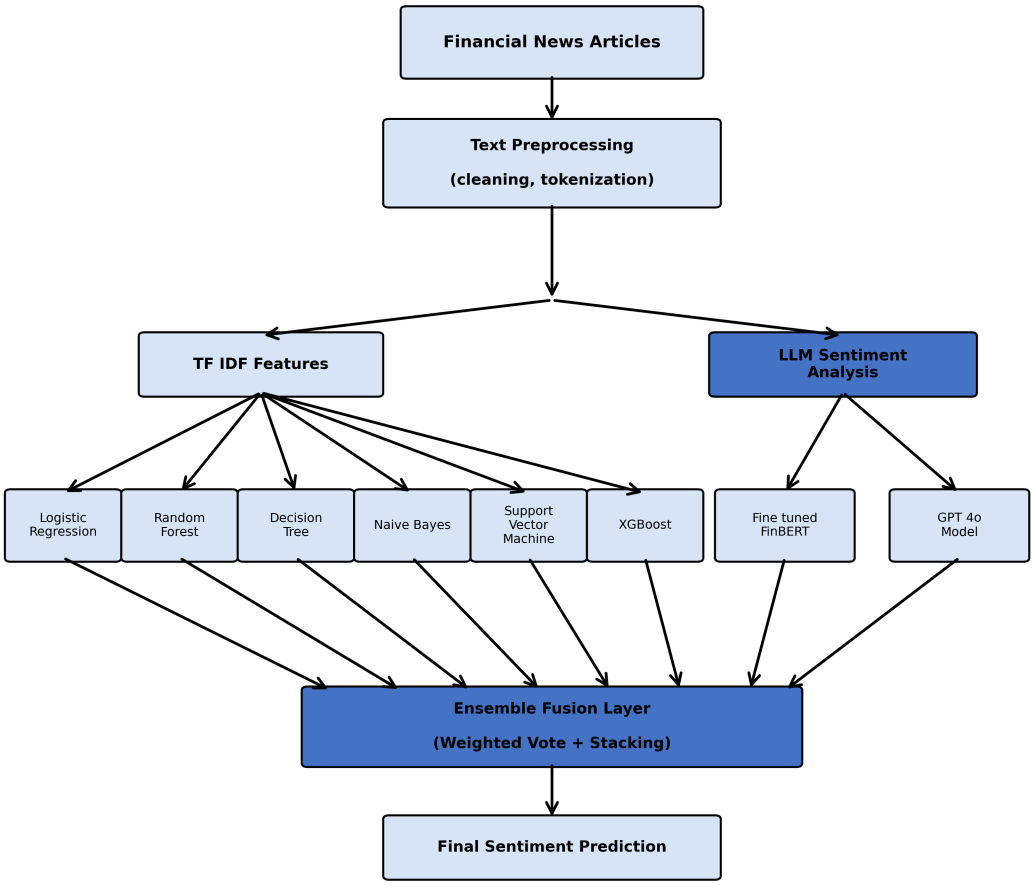


Table 2: Model Parameters and Configurations

Model	Feature Input	Key Parameters	Objective	Cross-val.	Output
FinBERT (baseline)	Contextual embeddings	—	—	—	Probabilities
FinBERT (fine-tuned)	Contextual embeddings	LR=2e-5, Epochs=5, batch=8, max_len=256, warmup=0.1	Cross–entropy	5-fold	OOF probs
Logistic Regression	TF–IDF (1–2 n-grams)	C=2.0, balanced	Cross–entropy	Stratified 5-fold	Probabilities
Random Forest	TF–IDF	$n_{\text{estimators}} = 300$, depth=15	Gini impurity	Stratified 5-fold	Probabilities
Decision Tree	TF–IDF	max_depth=10	Gini impurity	Stratified 5-fold	Probabilities
Naïve Bayes	TF–IDF	alpha=1.0	Log likelihood	Stratified 5-fold	Probabilities
SVM	TF–IDF	Linear kernel, class-balanced, CalibratedCV(3-fold)	Hinge loss + Platt scaling	Stratified 5-fold	Calibrated probs
XGBoost	TF–IDF	$n_{\text{est}} = 300$, depth=6, lr=0.1	Multi-class log-loss	Stratified 5-fold	Probabilities
LLM (GPT-4o)	Article text	temperature=0, logprobs=True, top_logprobs=5	—	—	Softmax log-probs
Stacked Meta	Base-model probs (24-dim)	LR, C=1.0, liblinear	Cross–entropy	—	Final sentiment

2.7 Backtesting Design

To evaluate the economic viability of each sentiment strategy, we construct a simple trading signal from the model's predicted sentiment: Positive $\rightarrow +1$ (long), Negative $\rightarrow -1$ (short), Neutral $\rightarrow 0$ (no position). The event-return horizon spans $t{+}1$ to $t{+}5$ trading days following the article's publication date, with the initial price recorded at or immediately prior to the news article's timestamp and the subsequent price measured at a fixed 5-day forward horizon. This ensures that sentiment signals are evaluated against forward-looking returns, which helps in avoiding the look-ahead bias. The portfolio return for each article is calculated as signal $\times$ event_return, and aggregate performance metrics (Mean Return, Volatility, Sharpe Ratio, Maximum Drawdown) are computed over the full test set. As a robustness check, we also compute abnormal returns by subtracting the contemporaneous S&P 500 return from each stock's event-window return.

To evaluate the economic viability of each sentiment strategy, a simple directional trading signal from the model's predicted label: Positive $\rightarrow +1$ (long), Negative $\rightarrow -1$ (short), Neutral $\rightarrow 0$ (no position) is constructed.

Price Alignment and Look-ahead Bias: For each article, the event anchor is defined as the first *trading day on or after* the article's publication timestamp. [3] The event-window return spans trading days $t{+}1$ through $t{+}5$ relative to this anchor, where $t{+}1$ is the first full trading day after the anchor day:

$$R_{\text{event}}(i) \;=\; \prod_{k=1}^{5}\big(1 + r_{t_0(i)+k}\big) \;-\; 1, \tag{26}$$

where $t_0(i)$ denotes the anchor trading day for article i and r_j is the daily close-to-close return on day j. Anchoring at the *next* trading day after publication and measuring returns from $t{+}1$ onward ensures that no intraday or same-day price information enters the signal, thereby avoiding look-ahead bias for articles published during market hours or on non-trading days.

Abnormal returns: As a robustness check, abnormal returns are computed by subtracting the contemporaneous S&P 500 return from each stock's event-window return:

$$AR_{\text{event}}(i) \;=\; R^{\text{stock}}_{\text{event}}(i) \;-\; R^{\text{S and P 500}}_{\text{event}}(i), \tag{27}$$

where both terms are five-day cumulative returns computed over the identical $[t_0{+}1,\, t_0{+}5]$ window. This is a market-adjusted return, not a CAPM alpha; it removes broad-index co-movement, but does not control for firm-specific factor

[3] The publication timestamp (`published` field) is used throughout—not the scrape timestamp, which post-dates all articles by design. Using the publication time ensures that the strategy can, in principle, act on the information at the next market open following publication.

exposures.

Portfolio construction: Each article constitutes an equally weighted, unit-size position: the portfolio return for article i is signal$(i) \times R_{\text{event}}(i)$. Multiple articles about the same ticker published within five trading days of each other generate overlapping positions that are treated independently. No transaction costs are assumed, consistent with the academic convention for assessing the informational content of sentiment signals rather than a live trading strategy. Because overlapping five-day windows induce serial correlation in the resulting return series, Newey-West HAC-corrected t-statistics (maxlags $= 5$) are reported for all mean-return tests (see Section 2.5).

Horizon rationale: The choice of a five-day holding horizon is motivated empirically. An event-study analysis, presented in Section 3.4, plots the average daily stock return in the $[-5, +5]$ trading-day window around article publication dates across all 777 articles. The results reveal near-zero returns at t=0, suggesting most price impact is not fully realised on the publication day, and negative post-event drift at t+1 through t+5, consistent with incomplete price discovery over a multi-day window. The t+1 to t+5 horizon is thus the shortest window that captures the predominant portion of the sentiment-driven price response while remaining operationally implementable.

Performance metrics: Aggregate performance is evaluated using mean daily return, volatility (standard deviation of portfolio returns), Sharpe Ratio, and Maximum Drawdown as defined in Section 2.5. The Sharpe Ratio is annualised by multiplying by $\sqrt{252}$, following the standard daily-return convention; it should be noted that the underlying observations are five-day cumulative event returns rather than daily returns, so the annualised figures should be interpreted as a comparability convention rather than a literal representation of trading frequency.

3 Result

3.1 Classification Performance

Table 3 presents the performance of each individual classical machine learning model on the test set. The *Classical Average* is obtained by computing the element-wise mean of the probability vectors from all six classifiers:

$$\bar{P}(y = k \mid x) \;=\; \frac{1}{6} \sum_{m \,\in\, \mathcal{M}_{\text{cl}}} P_m(y = k \mid x), \qquad \mathcal{M}_{\text{cl}} = \{\text{LR, RF, DT, NB, SVM, XGBoost}\}, \tag{28}$$

with the final label assigned as $\hat{y} = \arg\max_k \; \bar{P}(y{=}k \mid x)$.

Table 3: Individual Classical ML Model Performance on the Test Set

Model	Accuracy	Macro-F1
Support Vector Machine	0.6026	0.5931
Logistic Regression	0.5962	0.5874
Random Forest	0.5577	0.5312
XGBoost	0.5513	0.5198
Decision Tree	0.5064	0.4817
Naïve Bayes	0.4551	0.4203
Classical Average	**0.5513**	**0.5246**

Table 4 presents the results in terms of accuracy measures and the F1-Macro on the held-out test set ($n = 156$) for the following models: baseline FinBERT, Fine-tuned FinBERT, Classical Average, LLM (GPT-4o), Weighted Ensemble, and the stacked ensemble. The baseline FinBERT (`ProsusAI/finbert`, without fine-tuning) achieves 48.72% accuracy and 48.56% F1-Macro. While prior studies such as Huang et al. (2023) and Peng et al. (2021) report FinBERT accuracies exceeding 80–90% on larger corpora, the relatively modest performance here reflects the small sample size and domain mismatch between the pre-trained model's financial corpus and our five-firm news dataset.

Table 4: Model Performance Metrics (Test Set)

Method	Accuracy	Macro-F1
Baseline FinBERT	0.4872	0.4856
Fine-tuned FinBERT	0.6474	0.6485
Classical Average	0.5513	0.5246
LLM (GPT-4o)	0.7628	0.7561
Weighted Ensemble	0.7756	0.7674
Stacked Ensemble	**0.7821**	**0.7789**

Fine-tuning the baseline FinBERT via 5-fold cross-validation provides a 32.9% increase in accuracy and a 33.5% increase in F1-Macro, with corresponding values of 64.74% and 64.85%. The classical average method, which is an amalgamation of Logistic Regression, Decision Tree, Random Forest, Multinomial Naïve Bayes, SVM, and XGBoost, produces accuracy of 55.13% and F1-Macro of 52.46%. Among the individual classical models, Logistic Regression (59.62%) and SVM (60.26%) perform best, while Naïve Bayes (45.51%) performs worst.

The LLM (GPT-4o) model achieves 76.28% accuracy and 75.61% F1-Macro, substantially outperforming all individual base models. The weighted soft-voting ensemble achieves 77.56% accuracy and 76.74% F1-Macro, with optimal weights strongly favouring the LLM component (FinBERT_FT: 0.0001, Classical_Avg: 0.300, LLM_GPT4o: 0.700). This near-zero weight on FinBERT suggests that, conditional on the LLM and classical signals, the fine-tuned FinBERT provides minimal additional discriminative information under linear combination.

Note that the Stacked Ensemble performed the best with the corresponding accuracy and F1-Macro of 78.21% and 77.89%, respectively. The Stacked Ensemble method outperforms GPT-4o by 2.53% and 3.02% in accuracy and F1-Macro, respectively. The Stacked Meta-Learning ensemble achieved the highest accuracy. Before ensembling, the LLM (GPT-4o) outperformed all its counterparts, achieving the second-highest accuracy and macro-F1 score.

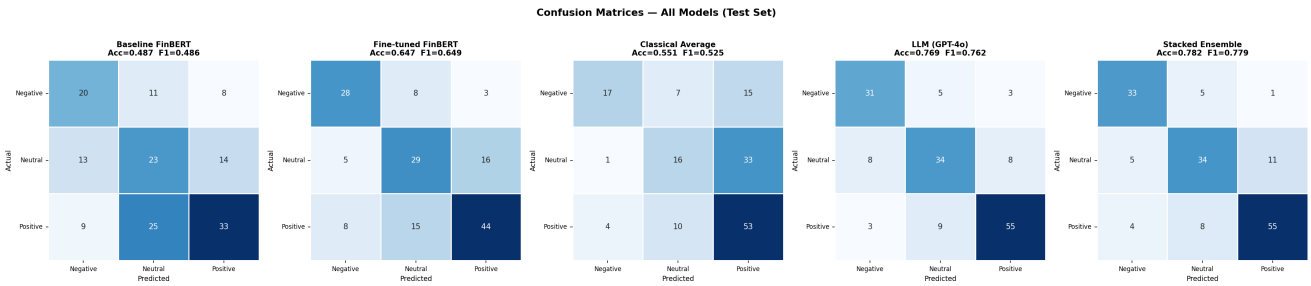


Figure 2: Confusion matrices for all five models on the test set

Figure 2 presents the confusion matrices for all five models on the held-out test set. The Baseline FinBERT (Acc = 0.487, F1 = 0.486) exhibits a rightward prediction bias, systematically incorrectly labeling Negative and Neutral articles as Positive. Fine-tuning substantially corrects this bias (Acc = 0.647, F1 = 0.649). The Classical Average (Acc = 0.551) exhibits the weakest Negative recall, 17 of 39 Negative articles are misclassified as Positive, reflecting the inability of TF-IDF representations to capture negation and contextual sentiment cues. GPT-4o (Acc = 0.769) and the Stacked Ensemble (Acc = 0.782) achieve the most balanced confusion profiles: both correctly identify 55 of 67 Positive articles while substantially reducing Negative misclassification. The Stacked Ensemble additionally improves Negative recall over GPT-4o with 33 versus 31 correct.

Table 5: Per-Ticker Accuracy Breakdown (Test Set)

Ticker	N	Baseline	Fine-tuned	Classical	GPT-4o	Soft Vote	Stacked
PLTR	38	0.447	0.737	0.684	0.684	0.711	0.763
GME	31	0.548	0.677	0.419	0.677	0.677	0.710
COST	32	0.469	0.719	0.594	0.812	0.812	0.844
TSLA	29	0.414	0.448	0.517	0.897	0.897	0.897
AAPL	26	0.577	0.615	0.500	0.769	0.808	0.692

Table 5 presents the per-ticker accuracy breakdown. The stacked ensemble achieves the highest accuracy 76.3% for PLTR, 71.0% GME, and 84.4% COST. For TSLA, the LLM (GPT-4o) and the ensemble approaches all achieve 89.7%. Interestingly, for AAPL, the stacked ensemble wit 69.2% accuracy underperforms the soft-vote ensemble (80.8%) and GPT-4o (76.9%), suggesting that the meta-learner may overfit to patterns less representative of Apple-related news.

3.2 Robustness: Financial Performance

Beyond the classification-based evaluation, we assess the economic viability of the sentiment-driven strategies using standard financial performance metrics, namely the Mean Return, Volatility, Sharpe Ratio, and the Maximum Drawdown. The event-return horizon is $t{+}1$ to $t{+}5$ cumulative.

Table 6: Trading Strategy Performance Metrics ($t{+}1$ to $t{+}5$ cumulative)

Model	Mean Daily Return	StdDev	Sharpe Ratio	Max Drawdown
finbert_baseline	0.0008	0.0316	0.409	−0.282
finbert_finetuned	0.0029	0.0301	1.514	−0.178
classical_avg	−0.0010	0.0321	−0.506	−0.406
gpt4o	0.0046	0.0294	2.484	−0.169
softvote	0.0044	0.0295	2.394	−0.169
stacked	**0.0053**	**0.0290**	**2.910**	**−0.206**

Table 6 reports, the stacked ensemble achieves the highest Sharpe Ratio of 2.910, followed by the GPT-4o model at 2.484 and the soft-vote ensemble at 2.394. A Sharpe Ratio above 2 is generally considered strong in financial applications. In contrast, the classical average yields a negative Sharpe Ratio (-0.506), implying an economically unviable strategy, while the baseline FinBERT achieves a modest positive Sharpe Ratio of 0.409. The fine-tuned FinBERT achieves a Sharpe Ratio of 1.514, substantially higher than the baseline, demonstrating that classification improvement translates directly into trading performance. The stacked model achieves the highest mean daily return (0.0053) with the lowest volatility (0.0290) among all strategies, indicating a favorable return-risk trade-off. The GPT-4o model achieves the lowest maximum drawdown (-16.9%), slightly better than the stacked ensemble (-20.6%). The classical average exhibits the deepest drawdown (-40.6%), underscoring its unsuitability for practical deployment.

Table 7: Abnormal-Return Performance (Stock S&P 500)

Model	Mean Abnormal	StdDev	Sharpe (abnormal)	Max Drawdown
finbert_baseline	−0.0007	0.0306	−0.355	−0.317
finbert_finetuned	0.0007	0.0290	0.400	−0.188
classical_avg	−0.0040	0.0298	−2.108	−0.593
gpt4o	0.0034	0.0284	1.915	−0.164
softvote	0.0032	0.0284	1.802	−0.164
stacked	0.0031	0.0281	1.750	−0.216

Table 7 has reported the same metrics computed on abnormal returns (stock event-window return minus the contemporaneous S&P 500 return). After removing market-level effects, the GPT-4o strategy achieves the highest abnormal Sharpe Ratio (1.915), followed closely by the soft-vote (1.802) and stacked (1.750) ensembles. Notably, the ordering differs from

the raw-return analysis: the stacked ensemble's edge over GPT-4o is narrower for abnormal returns, suggesting that part of its raw-return advantage stems from market timing rather than purely stock-specific sentiment signals.

Figure 3: Comparison of Sharpe Ratio

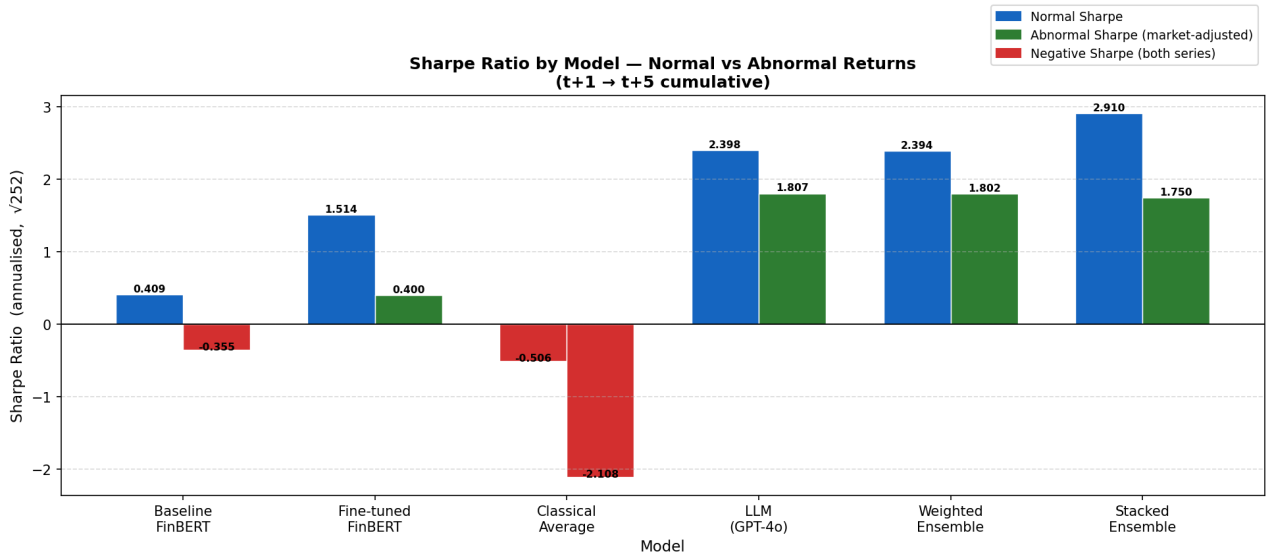


Figure 4: Equity curves (Normal Returns) of sentiment-driven strategies

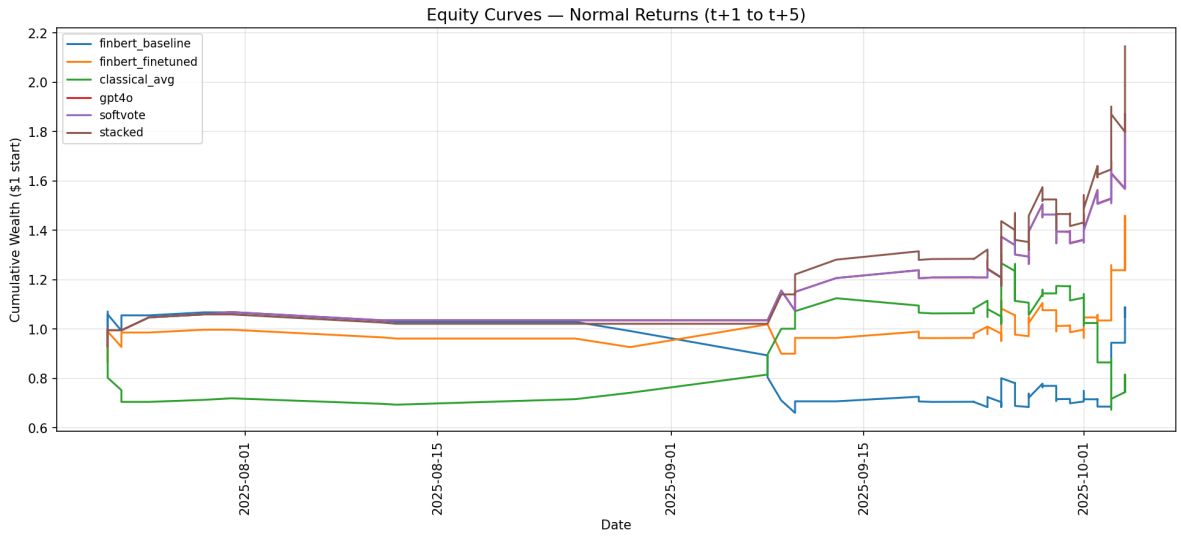


Figure 5: Equity curves (Abnormal Returns) of sentiment-driven strategies

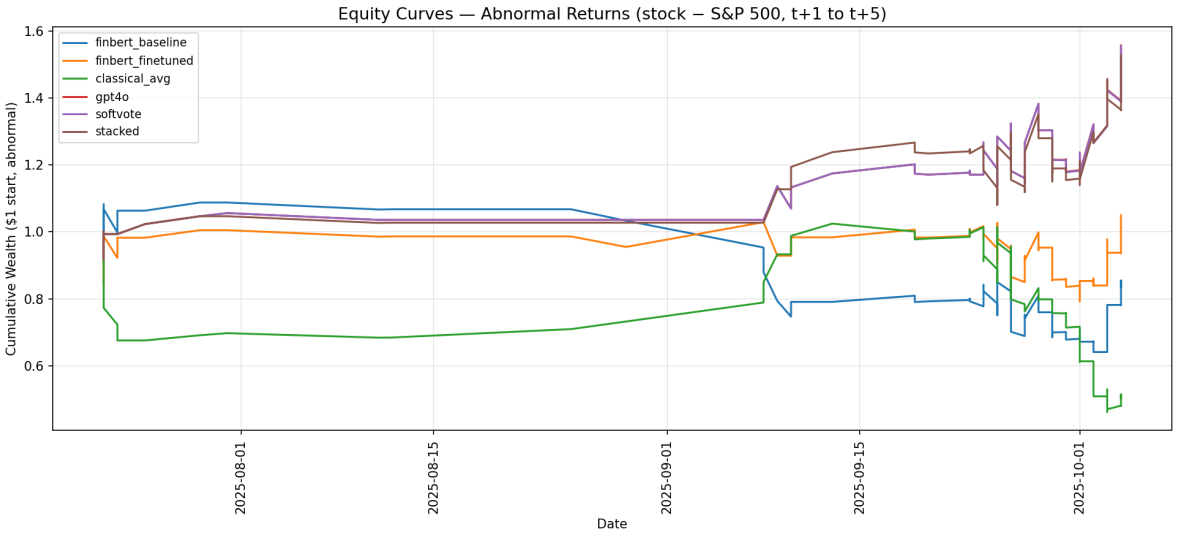


Figure 4 outlines the cumulative wealth evolution of each sentiment-driven strategy over the test period (July to October 2025), assuming a $1 initial investment. Several patterns merit discussion. First, the classical average strategy (green) suffers a steep and sustained decline from the beginning, dropping below $0.70 by mid-August 2025 and never recovering even to its initial value. This persistent underperformance is consistent with its negative mean daily return (-0.10%) and negative Sharpe Ratio (-0.506), confirming that TF-IDF-based frequency features alone are insufficient to generate profitable trading signals in this setting. Second, the baseline FinBERT strategy (blue) traces near $1.00 through August but experiences a sharp fall in early September 2025, falling to approximately $0.70 and remaining suppressed for the remainder of the sample. This behavior reflects the baseline model's poor classification accuracy (48.72%), which generates noisy and frequently incorrect trading signals. Although, the fine-tuned FinBERT (orange) exhibits a far more stable trajectory, remaining close to $1.00 through September before rising modestly toward the end of the period, underscoring the economic value of domain-specific fine-tuning. The three best-performing strategies: GPT-4o (red), Softvote (purple), and Stacked (brown) all remain tightly clustered near $1.00 through August and early September, after which they diverge sharply upward. The stacked ensemble reaches the highest terminal wealth (approximately $2.10 at its peak), followed by the softvote ($1.80) and GPT-4o ($1.60).

All three top strategies exhibit visible volatility in the final week of the sample (late September to early October 2025), with rapid oscillations between gains and losses. This turbulence aligns with the elevated market volatility during that period and suggests that the sentiment signals, while profitable on average, are prone to short-term noise.

Figure 6: Drawdown curves of sentiment-driven strategies.

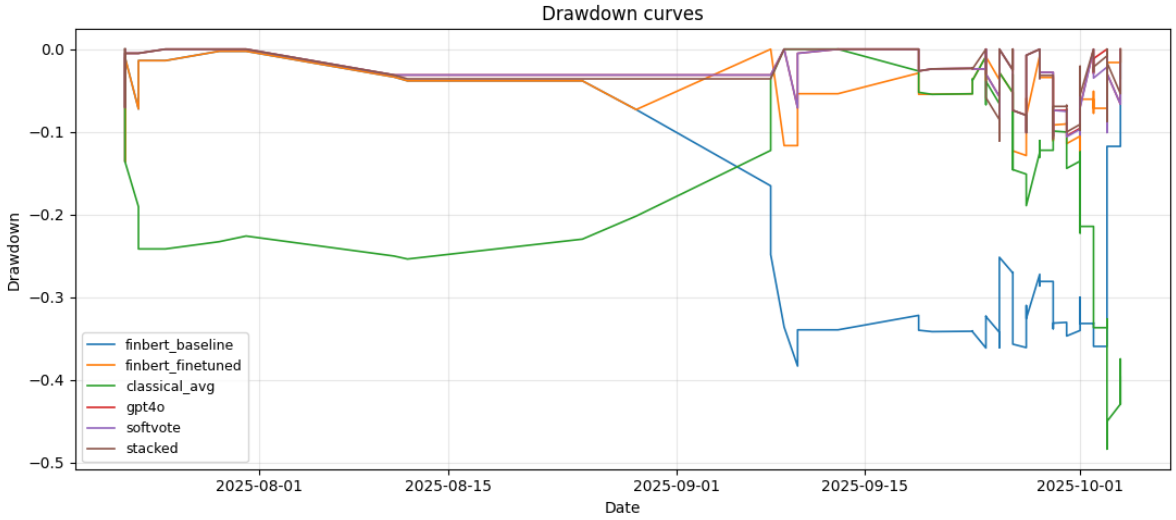


Figure 6 presents the drawdown profiles for all six strategies, measuring the percentage decline from the running peak of cumulative wealth at each point in time. The drawdown analysis reveals downside risk characteristics not fully captured by the Sharpe Ratio alone.

The classical average strategy enters a drawdown almost immediately, reaching -22% within the first two weeks and deepening to its maximum drawdown of -40.6% by early October 2025. It never recovers to within 10% of its peak, indicating a structurally loss-making strategy. Similarly, the baseline FinBERT experiences a prolonged drawdown phase beginning in early September, reaching a maximum drawdown of -28.2% and remaining in a deep drawdown (-30% to -32%) for the final month of the sample. These two strategies exhibit what can be characterised as *non-recovering drawdowns*, a particularly undesirable property for practical deployment, as they indicate persistent capital erosion rather than temporary setbacks. In contrast, the GPT-4o and softvote strategies display the most favourable drawdown profiles, with maximum drawdowns of only -16.9% each. Both strategies remain near their peaks (drawdown $< 5\%$) for the majority of the sample period, with brief dips that are quickly recovered.

The stacked ensemble occupies an intermediate position: while it achieves the highest terminal wealth and Sharpe Ratio, its maximum drawdown (-20.6%) is modestly deeper than those of GPT-4o and softvote. Inspecting the drawdown curve, this maximum occurs during the volatile late-September period, where the stacked strategy briefly gives back gains before recovering to new highs. The fine-tuned FinBERT exhibits a maximum drawdown of -17.8%, comparable to the top strategies, but its drawdown recovery is slower, and its terminal wealth is lower.

Figures 4 and 6 reinforce the quantitative findings in Tables 6 and 8. The stacked ensemble offers the best overall risk-return profile, achieving the highest cumulative wealth and Sharpe Ratio despite a slightly deeper maximum drawdown than the GPT-4o strategy. The GPT-4o model, in turn, provides the most conservative risk profile among the profitable strategies. Both the classical average and baseline FinBERT strategies are clearly unsuitable for practical deployment, as they exhibit persistent capital erosion and non-recovering drawdown patterns.

3.3 Statistical Test of Mean Returns

Table 8: Newey-West HAC t-test

Model	**Mean Return**	**Newey-West t-stat**	**p-value**
finbert_baseline	0.0008	0.346	0.730
finbert_finetuned	0.0029	1.104	0.270
classical_avg	−0.0010	−0.453	0.651
gpt4o	0.0046	2.235	0.025**
softvote	0.0044	2.177	0.030**
stacked	0.0053	2.377	0.018**

*$p < 0.10$; **$p < 0.05$; ***$p < 0.01$. $maxlags = 5$.

Table 8 reports Newey-West HAC-corrected t-statistics for all strategies, accounting for the autocorrelation induced by the overlapping five-day return windows. Among the six strategies, only GPT-4o, the soft-vote ensemble, and the stacked ensemble achieve statistically significant mean returns at the 5% level; the remaining three (Baseline FinBERT, Fine-tuned FinBERT, and Classical Average) fail to reject the null hypothesis of zero mean return. After adjusting for broad market movements (abnormal returns), the stacked ensemble's mean abnormal return is positive but no longer statistically significant at conventional levels, suggesting that part of the raw- return advantage may reflect implicit market timing rather than purely idiosyncratic sentiment signals.

Table 6 complements and strengthens the classification results from Table 4. The proposed ensemble technique outperforms its counterparts in terms of the Sharpe Ratio and the Mean Daily Return. The proposed stacked ensemble and the LLM (GPT-4o) model achieve positive, statistically significant Sharpe ratios relative to their contemporaries. This indicates that the profitability of the ensemble strategy is not driven by random fluctuations but reflects systematic predictive power embedded in sentiment signals.

To improve FinBERT's accuracy, the financial corpus needs to be expanded. The results demonstrate that hybrid ensembling significantly improves financial sentiment classification over the classical pipelines. The stacked ensemble particularly improved recognition of Neutral sentiment, a historically ambiguous class in financial text analysis. Our findings corroborate prior research that domain-specific transformers, when embedded in ensemble architectures, yield tangible

gains in financial NLP tasks (Hama Aziz and Dimililer, 2021; Kirtac and Germano, 2024). The LLM (GPT-4o) is the 2^{nd} best approach, followed by the weighted ensemble based on the accuracy measures.

4 Conclusion

This study introduces a multi-model ensemble framework integrating FinBERT fine-tuning, classical machine learning, and GPT-based LLM sentiment scoring for financial news analysis across multiple companies. The approaches consider a sample of 777 articles distributed across TSLA (187), AAPL (139), GME (131), COST (129), and PLTR (191), split into a training set of 621 articles and a held-out test set of 156 articles. Despite the limited size of the labelled dataset, the GPT-based LLM and the Stacked Ensemble achieved superior generalization and outperformed other techniques. Our results underscore these key insights:

Hybrid modeling improves comprehension and robustness. FinBERT's contextual understanding when combined with classical frequency-based models and LLM-based generative features captures both semantic depth and statistical stability. The proposed stacked ensemble framework achieves 78.21% accuracy, while the fine-tuned FinBERT model achieves 64.74% (a 32.9% improvement over the 48.72% baseline FinBERT). The LLM (GPT-4o) achieves 76.28% accuracy, while the classical ML average is 55.13%. The stacked ensemble achieves the best risk-adjusted trading performance, with a Sharpe Ratio of 2.910, a mean daily return of 0.53%, and moderate volatility (2.90%). The Newey-West HAC-corrected t-test confirms the mean return is statistically significant, indicating that the profitability of the ensemble strategy is not driven by random fluctuations but reflects systematic predictive power embedded in sentiment signals.

The soft-voting weights reveal model complementarity. The near-zero optimal weight on FinBERT in the soft-vote ensemble (0.0001) suggests that, under linear combination, the LLM subsumes much of the transformer's signal. However, the stacked ensemble's superior performance over the soft-vote demonstrates that nontrivial interactions among the eight base models' probability distributions exist and can be exploited by the meta-learner. Abnormal return analysis provides a nuanced view. While the stacked ensemble dominates on raw returns, GPT-4o achieves the highest abnormal Sharpe Ratio (1.915 vs. 1.750 for the stacked ensemble), suggesting that part of the ensemble's raw-return advantage comes from implicit market timing.

Future research should focus on expanding labelled datasets, particularly for underrepresented tickers; experimenting with nonlinear meta-learners on larger feature spaces; domain-adaptive pretraining; active learning loops that prioritise annotating uncertain samples; and out-of-sample validation across different time periods and market regimes.

References

Acemoglu, D., Autor, D., Hazell, J., and Restrepo, P. (2022). Artificial intelligence and jobs: Evidence from online vacancies. *Journal of Labor Economics*, 40(S1):S293–S340.

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. (2023). Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Agüero-Torales, M. M., Cobo, M. J., Herrera-Viedma, E., and López-Herrera, A. G. (2019). A cloud-based tool for sentiment analysis in reviews about restaurants on tripadvisor. *Procedia Computer Science*, 162:392–399.

Araci, D. (2019). Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.

Azar, P. D. and Lo, A. W. (2016). The wisdom of twitter crowds: Predicting stock market reactions to fomc meetings via twitter feeds. *Journal of Portfolio Management*, 42(5):123.

Baker, M. and Wurgler, J. (2006). Investor sentiment and the cross-section of stock returns. *The journal of Finance*, 61(4):1645–1680.

Bochkay, K., Brown, S. V., Leone, A. J., and Tucker, J. W. (2023). Textual analysis in accounting: What's next? *Contemporary accounting research*, 40(2):765–805.

Daly, L. (2026). Largest companies by market cap in 2026 — The Motley Fool.

Derakhshan, A. and Beigy, H. (2019). Sentiment analysis on stock social media for stock price movement prediction. *Engineering applications of artificial intelligence*, 85:569–578.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (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.

Dhaliwal, S. S., Nahid, A.-A., and Abbas, R. (2018). Effective intrusion detection system using xgboost. *Information*, 9(7):149.

Díaz, J. D., Hansen, E., and Cabrera, G. (2024). Machine-learning stock market volatility: Predictability, drivers, and economic value. *International Review of Financial Analysis*, 94:103286.

Es-Sabery, F., Es-Sabery, K., Qadir, J., Sainz-De-Abajo, B., Hair, A., Garcia-Zapirain, B., and De La Torre-Díez, I. (2021).

A mapreduce opinion mining for covid-19-related tweets classification using enhanced id3 decision tree classifier. *IEEE Access*, 9:58706–58739.

Face, H. (2023). Hugging face's transformer models. *URL: https://huggingface. co*.

Hama Aziz, R. H. and Dimililer, N. (2021). Sentixgboost: enhanced sentiment analysis in social media posts with ensemble xgboost classifier. *Journal of the Chinese Institute of Engineers*, 44(6):562–572.

Huang, A. H., Wang, H., and Yang, Y. (2023). Finbert: A large language model for extracting information from financial text. *Contemporary Accounting Research*, 40(2):806–841.

Jain, P. K., Pamula, R., and Yekun, E. A. (2022). A multi-label ensemble predicting model to service recommendation from social media contents. *The Journal of Supercomputing*, 78(4):5203–5220.

Kamal, S., Sharma, S., Kumar, V., Alshazly, H., Hussein, H. S., and Martinetz, T. (2022). Trading stocks based on financial news using attention mechanism. *Mathematics*, 10(12):2001.

Kim, A., Muhn, M., and Nikolaev, V. (2024). Financial statement analysis with large language models. *arXiv preprint arXiv:2407.17866*.

Kim, H., Jeong, J., Ko, H., and Lee, W. (2026). Large language models as financial analysts: Sector-aware reasoning for investment decisions. *Computational Economics*, pages 1–31.

Kirtac, K. and Germano, G. (2024). Sentiment trading with large language models. *Finance Research Letters*, 62:105227.

Loughran, T. and McDonald, B. (2011). When is a liability not a liability? textual analysis, dictionaries, and 10-ks. *The Journal of finance*, 66(1):35–65.

Magdon-Ismail, M. and Atiya, A. F. (2004). Maximum drawdown. *Risk Magazine*, 17(10):99–102.

Manjhi, G. and Shah, R. (2025). What do sentiments of budget speeches mean for stock returns? *Economic Modelling*, page 107156.

Nguyen, T. H., Shirai, K., and Velcin, J. (2015). Sentiment analysis on social media for stock movement prediction. *Expert Systems with Applications*, 42(24):9603–9611.

Peng, B., Chersoni, E., Hsu, Y.-Y., Huang, C.-R., et al. (2021). Is domain adaptation worth your investment? comparing bert and finbert on financial tasks. Association for Computational Linguistics (ACL).

Polikar, R. (2012). Ensemble learning. In *Ensemble machine learning*, pages 1–34. Springer.

Priyadarshini, I. and Cotton, C. (2021). A novel lstm–cnn–grid search-based deep neural network for sentiment analysis. *The Journal of Supercomputing*, 77(12):13911–13932.

Rehman, A. U., Malik, A. K., Raza, B., and Ali, W. (2019). A hybrid cnn-lstm model for improving accuracy of movie reviews sentiment analysis. *Multimedia Tools and Applications*, 78(18):26597–26613.

Rozental, A. and Fleischer, D. (2017). Amobee at semeval-2017 task 4: deep learning system for sentiment detection on twitter. *arXiv preprint arXiv:1705.01306*.

Ruan, Y., Durresi, A., and Alfantoukh, L. (2018). Using twitter trust network for stock market analysis. *Knowledge-Based Systems*, 145:207–218.

Salur, M. U. and Aydin, I. (2020). A novel hybrid deep learning model for sentiment classification. *IEEE Access*, 8:58080–58093.

Savor, P. and Wilson, M. (2013). How much do investors care about macroeconomic risk? evidence from scheduled economic announcements. *Journal of Financial and Quantitative Analysis*, 48(2):343–375.

Sharpe, W. F. et al. (1998). The sharpe ratio. *Streetwise–the Best of the Journal of Portfolio Management*, 3(3):169–85.

Sokolova, M. and Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information processing and management*, 45(4):427–437.

Tumarkin, R. and Whitelaw, R. F. (2001). News or noise? internet postings and stock prices. *Financial Analysts Journal*, 57(3):41–51.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wallach, H. M. (2006). Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, pages 977–984.

Wissler, L., Almashraee, M., Díaz, D. M., and Paschke, A. (2014). The gold standard in corpus annotation. *IEEE GSC*, 21.

Wolpert, D. H. (1992). Stacked generalization. *Neural networks*, 5(2):241–259.

Xia, Y., Chen, K., and Yang, Y. (2021). Multi-label classification with weighted classifier selection and stacked ensemble. *Information Sciences*, 557:421–442.

Statements and Declarations

CRediT authorship contribution statement

Kartik M Avhad: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing - Original draft, Writing - review and editing.

Ganesh Manjhi: Conceptualization, Formal analysis, Investigation, Methodology, Project administration, Resources, Supervision, Validation, Visualization, Writing - Original draft, Writing - review and editing.

Declarations

Funding No funds, grants, or other support were received for the preparation of this manuscript.

Data Availability Data will be made available upon request.

Conflict of Interest The authors have no competing interests to declare that are relevant to the content of this article.