Adaptive sentiment analysis for stock markets using deep learning

Talent Mawere, Selvaraj Rajalakshmi, Venu Madhav Kuthadi, Otlhapile Dinakenyane
Department of Computing and Informatics, School of Natural and Applied Sciences,
Botswana International University of Science and Technology, Palapye, Botswana

Article Info

Article history:

Received Jun 6, 2025
Revised Nov 22, 2025
Accepted Feb 17, 2026

Keywords:

Attention mechanism
Deep learning
FinTech
Natural language processing
Real-time data pipeline
Sentiment analysis
Stock market forecasting

ABSTRACT

Stock markets are highly volatile, making price prediction very difficult. One of the factors influencing the volatility of financial markets is rapidly changing news sentiment. This study presents a novel adaptive deep learning (DL) framework for sentiment analysis with concept drift capabilities. The proposed model combines convolutional neural networks (CNN), bidirectional long short-term memory (BiLSTM), and attention mechanisms in its processing architecture. The model inputs preprocessed news headlines into both the CNN and BiLSTM-Attention networks to extract local features, model contextual dependencies, and prioritizes important sentiment cues in its prediction mechanism. We use FastText and Word2Vec for word embeddings, while incremental learning is used to manage concept drift. One key advantage of handling concept drift is that the model can continuously learn new patterns in data streams without needing to fully retrain the model. The model is validated on a curated dataset from various sources with superior performance across all metrics, like accuracy (0.9753) and an F1-score (0.98). It significantly outperforms benchmarks like distilled bidirectional encoder representations from transformers (DistilBERT), LSTM, and valence aware dictionary and sentiment reasoner (VADER). A run of ten iterations validated that the real-time pipeline did not exceed 200 ms in processing and classifying headlines. This signifies the practical viability of our model in fintech applications such as algorithmic trading and risk management.

Corresponding Author:

Talent Mawere
Department of Computing and Informatics, School of Natural and Applied Sciences
Botswana International University of Science and Technology
Palapye, Botswana
Email: talentmawere@gmail.com or mt23018975@biust.ac.bw

1. INTRODUCTION

Predicting stock market trends is important for investors and policymakers. However, it is still difficult because of the complex mix of numerical data and other factors like news sentiment and investor psychology that contribute to stock price movements [1], [2]. Past studies document two main techniques, fundamental analysis and technical analysis, used for stock market prediction. Technical analysis uses artificial intelligence or machine learning (AI/ML) on historical data. Fundamental analysis, on the other hand, evaluates financial statements to predict future stock prices. Both, however, often miss the important effect of market sentiment from news and social media [3], [4].

Deep learning (DL) techniques can help overcome this limitation by automatically identifying patterns and using natural language processing (NLP) for unstructured data [5], [6]. Several studies, e.g., [4], [7] have demonstrated the efficacy of DL methods that integrate sentiment analysis in stock market prediction. Investors' sentiments change rapidly due to many factors, and a model trained on older data may fail to predict accurately if sentiments shift rapidly. This is an ongoing challenge where models lack real-time adaptability, are vulnerable to concept drift, and are generally unclear on how the decision-making process works [8], [9].

To specifically model financial data, transformer-based models like financial bidirectional encoder representations from transformers (FinBERT) [10] have been developed. Such models offer domain-specific prowess but have been heavily criticized for their heavy computational requirements and are thus not suitable for high-frequency streaming applications. A critical gap exists, therefore, for frameworks that balance high accuracy, computational efficiency, and real-time adaptation to both evolving market language and evolving market sentiments. In response to this, our paper proposes a new intelligent framework for adaptive sentiment analysis in stock markets using adaptive DL techniques. Our work makes three important contributions: i) we design a domain-specific convolutional neural network-bidirectional long short-term memory (CNN-BiLSTM) model with an attention mechanism. This setup effectively extracts local semantic patterns and models long-range contextual dependencies in financial text, always emphasizing the most salient sentiment cues. These address limitations found in standalone neural networks; ii) we introduce an adaptive real-time pipeline with concept drift detection and an incremental learning module using the river library. This allows the model to update continuously based on recent data. It helps maintain accuracy in volatile markets without the need for expensive retraining; and iii) to validate our model, we conduct extensive benchmarking through a thorough empirical evaluation against a variety of baselines, including a transformer, distilled BERT (DistilBERT), a DL model, LSTM, CNN-BiLSTM, and lexicon-based tools like valence aware dictionary and sentiment reasoner (VADER) and TextBlob. Our results show superior performance across multiple metrics.

Financial markets have always sparked interest among many groups since the beginning of time. Initially, early market theories, such as the efficient market hypothesis (EMH) [11] and random walk theory [12], posited that prices fully reflect all available information and are inherently unpredictable. However, behavioral finance has challenged this view. This group argues that cognitive biases and investor sentiment create market inefficiencies that can be exploited to predict future stock prices [13], [14]. It is from this view that sentiment analysis has become a key tool for understanding market trends from unstructured text [15].

Sentiment analysis is the process of automatically identifying and categorizing people’s emotions, opinions, and or attitudes based on extracted text. Sentiment analysis has evolved from lexicon-based methods like VADER and traditional ML classifiers to DL architectures. Recurrent neural networks (RNNs), especially the LSTM networks, are great at modeling long-term dependencies in sequential text [16]. Meanwhile, CNNs are good at detecting local n-gram patterns that may carry sentiment patterns [17]. Hybrid models like CNN-LSTM have been proposed to combine both strengths [18]. In a previous study, Wu *et al.* [2] integrated sentiment from news with technical indicators using an LSTM with attention, but their sentiment analysis relied solely on Word2Vec. Recent studies by [19] and [20] highlight the growing use of incremental learning for financial data streams, yet its integration with a hybrid CNN-BiLSTM-Attention architecture for sentiment analysis remains underexplored. Recently, transformer-based models like BERT [21] and its financial domain adaptation, FinBERT [10] has achieved new highs in NLP tasks due to its deep understanding of context. Despite their strengths, transformers require a lot of computing power, which makes real-time use difficult. Furthermore, most existing models are static and struggle with concept drift. Concept drift refers to the phenomenon where the statistical properties of the target variable, market sentiment, change over time [21].

Our model addresses these gaps by proposing a computationally efficient hybrid model that incorporates a dedicated concept drift adaptation module, bridging the performance of DL with the agility required for real-time fintech applications. Table 1 summarizes a synthesis of some extant literature related to the study. Highlighting the research focus, methodology used, key contributions, and the identified gaps in the particular study.

2. METHOD

2.1. System architecture and data pipeline

Designed specifically for use in financial prediction systems, our framework, which works in both offline training and online inference, is shown in Figure 1. The system begins with a comprehensive data acquisition and processing pipeline, which we discuss as follows.

i) Data acquisition process, for training and testing our model, involved creating a custom Python pipeline using BeautifulSoup and Scrapy, where we collected over 1.2 million financial news articles from sources like Bloomberg, Reuters, and the Wall Street Journal for the period between 2016 and 2023. The real-time data acquisition process for incremental learning involved connecting via application programming interface (API) to a dynamic pipeline, NewsAPI, which streams articles every 5 minutes during trading hours.

ii) Data preprocessing included careful text cleaning processes such as removing non-alphanumeric characters, text normalization, lowercasing text, expanding contractions, lemmatization, and filtering out noise. Sentences are padded or truncated to a length of 128 tokens.

iii) Text embedding is critical in NLP tasks. We employed a hybrid embedding strategy similar to [1], but with minor distinctions such as the use of Word2Vec [22] and FastText [23] instead of using DistilBERT. Word2Vec [2] was effective in capturing broad semantic relationships. FastText [3] helped with subword information, which is important for dealing with rare financial terms. The 300-dimensional vectors from each approach were combined to create a 600-dimensional word representation, which will be fed to the CNN-BILSTM-Attention network to provide a rich feature foundation for the model's sentiment analysis task [24].

Table 1. Summary of related work

Author/s	Research focus	Methodology	Key contributions	Gaps
Rani and Kumar [17]	Sentiment analysis (SA) using a CNN model.	– CNN architecture with varying configurations. – Comparison with ML baselines (naive Bayes (NB), support vector machine (SVM), k-nearest neighbors (k-NN)).	– Model achieved 95% accuracy. – Showed optimal CNN configurations	Deeper CNNs (3+ layers) increased training time without significant accuracy gains.
Chen *et al.* [35]	Review of SA applications in the stock market.	Reviewed 223 articles from Web of Science	Identified key trends: DL, news sentiment, investor behavior.	Excluded non-English studies and non-Wos-indexed papers.
Mehtab and Sen [36]	Stock price prediction using LSTM+SA	Used LSTM, Twitter SA, self-organizing fuzzy neural network (SOFNN) cross-validation.	Sentiment-augmented SOFNN outperforms LSTM and traditional models.	Real-time applicability not tested.
Sidogi *et al.* [37]	Stock prediction using FinBERT and LSTM.	– LSTM with FinBERT for SA. – Intraday price data with 25-minute lag.	FinBERT improves prediction accuracy over general BERT.	Focused on intraday data; not validated for long-term predictions.
Tul *et al.* [38]	Review of DL techniques for SA.	Literature review of CNN, RNN, DBN, and hybrid models in SA.	Comprehensive overview of DL applications in SA.	No new empirical methods proposed.
Singh *et al.* [20]	SA is using traditional ML classifiers.	– Used NB, J48, BFTree, OneR; feature selection (DF, MI, IG). – WEKA evaluation.	NB was fastest, OneR most precise with acc 92.34%.	Lacks advanced preprocessing like word embeddings.
Souma *et al.* [39]	SA forecasting using RNN-LSTM	Used GloVe word vectors; trained on Thomson Reuters news analytics (TRNA) and Dow Jones industrial average (DJIA) stock data.	Acc improved with hierarchical training.	Reliance on stock price fluctuations for sentiment labelling and no multilingual support.

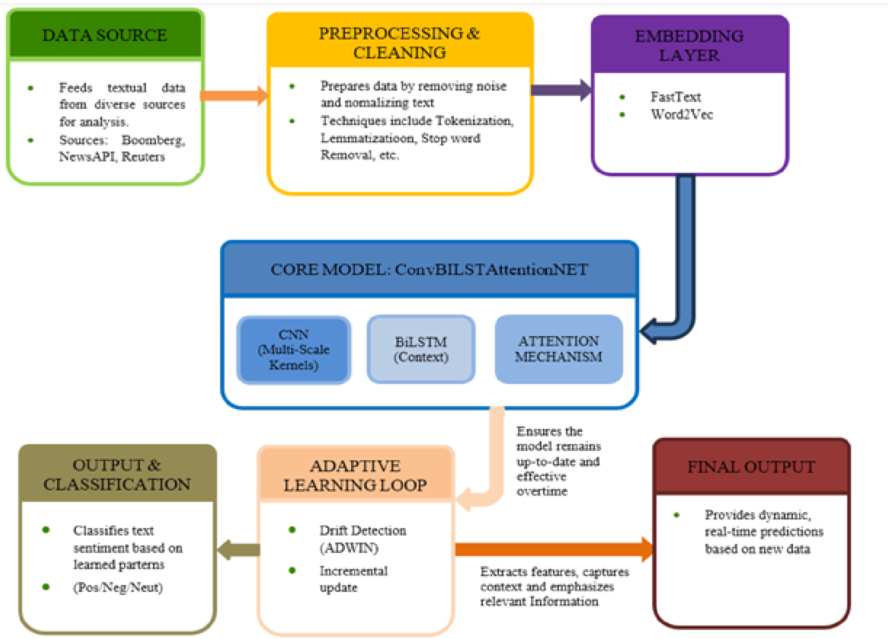


Figure 1. Sentiment analysis model architecture

2.2. Model architecture

The central component of our framework is the CNN-BiLSTM-Attention network, which is supported by four layers. A multi-scale CNN for local feature extraction forms the initial layer. In this layer, the embedded input sequence goes through five parallel 1D convolutional layers with kernel sizes of 1, 2, 3, 4, and 5, facilitating the model to find sentiment cues at different levels, from unigrams to 5-grams. Two hundred filters will be used at each layer, followed by a rectified linear unit (ReLU) activation and global max-pooling, resulting in a 200-dimensional vector per kernel and a concatenated 1,000-dimensional feature vector for input to the next layer.

In the second layer, a BiLSTM layer, which works in parallel to the CNN layer and is designed for sequential context modeling, also takes input from the original embedded sequence. Within the layer, text is analyzed in both forward and backward directions to capture long-range dependencies and context, resulting in a sequence of hidden states. The third layer, working in sequence with the BiLSTM layer, consists of an attention mechanism for salience weighting. Hidden states output from the BiLSTM layer are ingested, and a weighted sum context vector is calculated. According to Vaswani *et al.* [25], attention mechanisms help a model to focus on the most important sentiment-related words while ignoring irrelevant information, thereby improving a neural network's performance. The output of this layer is a 128-dimensional attention-weighted vector.

Both inputs from the CNN and the CNN-BiLSTM-Attention layer are combined and fed into a classification layer. This combined representation goes through a dropout layer at a rate of 0.5 for regularization. A 128-unit dense layer with ReLU activation and a softmax layer for a three-class sentiment classification, positive, negative, or neutral.

2.3. Adaptive learning mechanism

Most financial markets exhibit inherent non-stationarity [26], [27] necessitating an adaptive framework that continuously updates to evolving sentiment patterns. To implement this, we incorporated an intelligent three-component adaptive mechanism. Firstly, we implement a feedback loop architecture operating on a closed-loop control principle, where outputs from the CNN-BiLSTM-Attention neural network serve as the input signal for continuous adaptation. We use the river library's adaptive windowing (ADWIN) algorithm [28] to check the entropy of the prediction distribution in real time. This makes it easier to set up a direct feedback loop between model performance and parameter updates. The system makes targeted changes when the entropy goes over a certain level, like ($H > 0.85$), which probably means that the confidence in the prediction is going down.

Guo *et al.* [29] found that miscalibration is a common problem with neural networks. We suggest a method that uses dynamic parameter scaling to make calibration more market-aware. By lowering the entropy threshold and raising the learning rates intelligently, scaling lets a model become more sensitive when the market is very volatile. This makes sure that the model reacts quickly to events that move the market. In stable periods, the system keeps the normal settings, which stops it from overfitting to short-term noise. By applying this configuration, we make sure that the framework focuses on adapting during times of significant market change.

Zarghani and Abedi [30] noted that sliding window techniques are very important for working with streams of data. They did, however, point out that fixed-size windows have trouble adapting to changes that happen quickly, like bursty patterns or concept drift. Our third strategy is to use an intelligent window sizing technique based on market conditions to fix this. We suggest a 48-hour rolling window for incremental updates that strikes the best balance between being quick to respond and being statistically reliable. When spikes in volatility are detected, the system automatically shortens this window to 24 hours to give more weight to recent, high-impact data. To keep the model stable, the window size is increased during stable times. When drift is found, the framework uses a HoeffdingTree classifier [31] trained on mini-batches (256 samples) from the optimized time window to make targeted updates. This method makes it easy to quickly adapt to new market trends while still being efficient with computers, since only the temporal attention parameters are changed, and the core sentiment classification layers stay the same to avoid catastrophic forgetting.

2.4. Strategy for fine-tuning

In addition to the real-time adaptation strategy, we discussed above, we suggest a three-phase adaptation approach to fine-tune our network. Based on Howard and Ruder's work on discriminative fine-tuning [32], we suggest an approach of gradually unfreezing layers to help prevent catastrophic forgetting while allowing effective domain transfer. This kind of strategy could work well in finance, where it's important to expand vocabulary and calibrate sentiment intensity to deal with problems with domain adaptation [10].

Another method is multi-task learning, in which we add other goals, such as predicting volatility and detecting urgency. The results of [33], who showed that auxiliary tasks improve model robustness and generalization. Finally, we recommend a continuous learning method that can detect concept drift, which is in line with the incremental adaptation ideas put forth by Gama *et al*. [34] for environments that are not stable. This method guarantees consistent long-term performance while maintaining the low-latency inference required for real-time trading applications. These three approaches work well together to balance adaptation efficiency and practical deployment, which is what fintech AI research needs. Table 2 presents the final internal structure of our model, which will be trained and tested.

Table 2. Internal structure of the complete model

Layer/component	Configuration	Output shape	Parameters	Function
Input layer	Sequence length: 128 Embedding dim: 600	(128, 600)	-	Receives padded text sequences
Multi-scale CNN				
1st layer Conv1D (k =1)	Filters: 200, Kernel: 1	(128, 200)	120,200	Unigram patterns
2nd layer Conv1D (k =2)	Filters: 200, Kernel: 2	(127, 200)	240,200	Bigram patterns
3rd layer Conv1D (k =3)	Filters: 200, Kernel: 3	(126, 200)	360,200	Trigram patterns
4th layer Conv1D (k =4)	Filters: 200, Kernel: 4	(125, 200)	480,200	4-gram patterns
5th layer Conv1D (k =5)	Filters: 200, Kernel: 5	(124, 200)	600,200	5-gram patterns
Global max pooling	Applied per branch	(200,) each	-	Extracts most salient features
Concatenation	Merge all CNN branches	(1000,)	-	Combines multi-scale features
BiLSTM layer	Units: 128 Return sequences: true	(128, 256)	1,494,528	Captures long-range dependencies Forward and backward =256 units
Attention mechanism				
Score calculation	tanh activation	(128, 256)	65,792	Computes attention scores
Weight computation	softmax normalization	(128, 1)	-	Converts to a probability distribution
Context vector	Weighted sum	(256,)	-	Focuses on relevant temporal features
Feature fusion	Concatenate CNN+Attention	(1256,)	-	Combines local+global patterns
Dropout	Rate:0.5	(1256,)	-	Prevents overfitting
Dense layer	Units:128, ReLU	(128,)	160,896	Final feature transformation
Output layer	Units :3, softmax	(3,)	387	3-class sentiment classification

2.5. Experimental setup and evaluation metrics

We split our data into three sets, training (70%), validation (15%), and testing (15%), organized by time to avoid temporal leakage, and trained it over 100 epochs with an AdamW optimizer to minimize the composite loss function. To evaluate the performance of our model, we conducted two types of comparisons, including an ablation study to assess the attention networks' contribution to our model's performance and a benchmark comparison against five other baselines. These include DistilBERT, a compact transformer; LSTM; a CNN-BiLSTM without attention; VADER; FastText; and TextBlob. We selected benchmark models based on their performance relevance and their ability to be deployed in real-time. While models like FinBERT provide great accuracy, they are computationally complex and less suitable for the high-throughput, low-latency pipeline central to our study; we did not consider them in our evaluation. Rather, we focus on models that achieve high accuracy with computational efficiency. We evaluate performance using standard classification metrics, including accuracy, precision, recall, F1-score, and area under the curve (AUC). A confusion matrix is used for error analysis.

3. RESULTS AND DISCUSSION

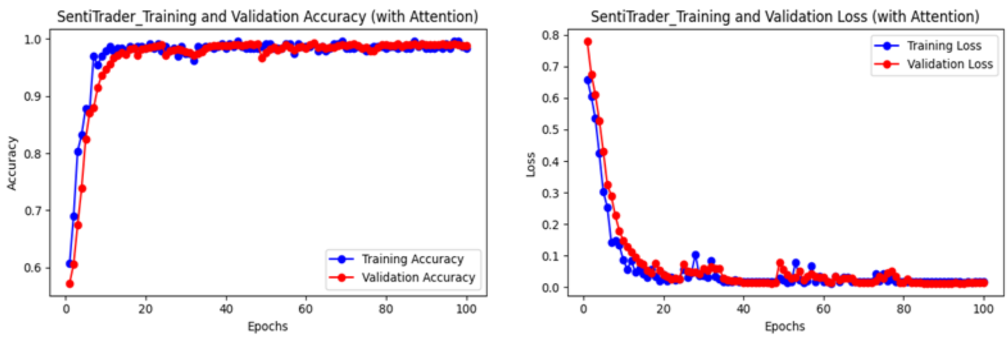
3.1. Ablation study and hyperparameter optimization

As discussed earlier, the initial evaluation plan involves undertaking an ablation study to observe overall contribution of each component. Sentiment analysis using a CNN yielded a validation accuracy of 62% validation loss. With our final model CNN-BiLSTM-Attention, yielding a validation accuracy and loss of 82% and 0.25%, confirming an overall contribution of 20% after all components are integrated.

After performing a component-wise ablation study validating each component's contribution to the overall model's performance, we then applied Optuna for hyperparameter optimization for both our proposed model and the baselines. Figure 2 shows the training and validation accuracy (Figure 2(a)) and loss (Figure 2(b)) plots for our model after the new configurations. We observe that our proposed model reached a validation accuracy of 97.4% with a validation loss stabilizing under 0.1. We attribute the credit to the attention mechanism, which acted as a regularizer, stabilizing training and enhancing generalization as shown in Table 3.

Table 3. Ablation study validation loss and accuracy values

Metric	CNN (%)	BiLSTM (%)	Sentiwaretrade (no attention) (%)	Sentiwaretrade (with attention) (%)
Validation accuracy	62	70	75	82
Validation loss	0.68	0.40	0.35	0.25
Generalization gap	5	10	5	3



(a) (b)

Figure 2. Performance plots for tested models showing (a) accuracy and (b) loss

3.2. Model performance and benchmarking

Table 4 summarizes the performance of our proposed model against the proposed baselines. Our model, CNN-BiLSTM-Attention (SentiTrader), achieved top results, with good scores across all metrics (accuracy: 0.975, F1-score: 0.982, and AUC: 0.973) demonstrating our model's superiority in modelling sentiment data in financial markets. We support the hypothesis that hybrid architecture performs better than standalones since even SentiTrader, without attention, already outperformed the standalone LSTM by 3.1% in accuracy. Our findings further validate results from the ablation study by confirming that attention networks improve predictions, as evidenced by the improved performance when the attention network was added. The attention mechanism enhanced the F1-score by 3.8%, confirming its role in emphasizing important words and reducing misclassification. Unlike transformer models like FinBERT, the proposed model achieves top performance while keeping a lightweight and efficient design. Our framework is therefore suitable even for real-time use in high-frequency trading environments where low latency is crucial. The poor performance of DistilBERT and lexicon-based models highlights the mismatch between general-purpose language models and financial text.

Table 4. Evaluation metrics of tested models

Model	Accuracy	Precision	Recall	F1-score	AUC
Sentiwaretrader (with attention)	0.9753	0.9626	0.9746	0.9816	0.9728
Sentiwaretrader (without attention	0.9467	0.9539	0.9473	0.9633	0.9589
LSTM	0.9333	0.9439	0.9423	0.9321	0.9236
DistilBERT	0.4017	0.7612	0.4017	0.2351	0.5270
FastText	0.0377	0.0359	0.0377	0.0350	0.0430
Vader	0.3515	0.5133	0.3515	0.4163	0.5123
TextBlob	0.1841	0.4570	0.1841	0.2518	0.3241

3.3. Analysis of confusion matrices

As outlined in our evaluation framework, Figure 3 presents a plot of the confusion matrices for DistilBERT, LSTM, and both variations of SentiTrader. We use confusion matrices due to their ability to offer detailed insights into how well each model performs across sentiment categories. We observe that the LSTM baseline has high confusion between classes, especially between positive and negative sentiments, misclassifying 20 instances between these two. However, the DistilBERT confusion is more pronounced, providing further support to our earlier assertion that it struggles to capture the subtle meanings in financial language. A significant improvement is observed on SentiTrader without attention, achieving perfect classification for the negative and neutral classes, while reducing the positive-negative confusion to just four instances. Notably, SentiTrader with attention achieves perfect classification across all sentiment categories, completely removing cross-class errors. This highlights how well our adaptive framework works to identify sentiment-bearing phrases and clear up the context issues that other models face. We conclude that the steady improvement from LSTM to SentiTrader without attention, and then to SentiTrader with attention, supports our design choices, not only regarding importance attention mechanisms but real-time adaptability in improving sentiment classification in financial text analysis.

3.4. Real-time performance and practical implications

Our real-time performance evaluation demonstrates the pipeline's operational efficiency with the entire process from news ingestion to sentiment classification, with a latency of less than 200 ms, as shown in Figure 4. These results are within the strict needs of high-frequency trading environments. We observed that the concept drift capability successfully identified and adjusted to volatility shifts during backtesting. Prediction decay was reduced by 32% compared to a static model, see Figure 4. This strength is important for fintech deployment because it confirms that the model performs reliably even during market crises. Figure 4 presents a performance analysis using pipeline latency distribution and concept drift adaptation metrics.

To verify our model's concept drift detection and incremental update capability, we evaluated the pre and post COVID-19 period. As illustrated in Figure 5, the main finding is that a static model quickly becomes outdated during a market shock, whereas our adaptive model handled this challenge through continuous learning. While the model did not perform so well initially, it prevented a potential 40-60% drop in performance and instead achieved a consistent 3-8% advantage. This highlights the important role of adaptive frameworks in real-world financial applications where volatility is high.

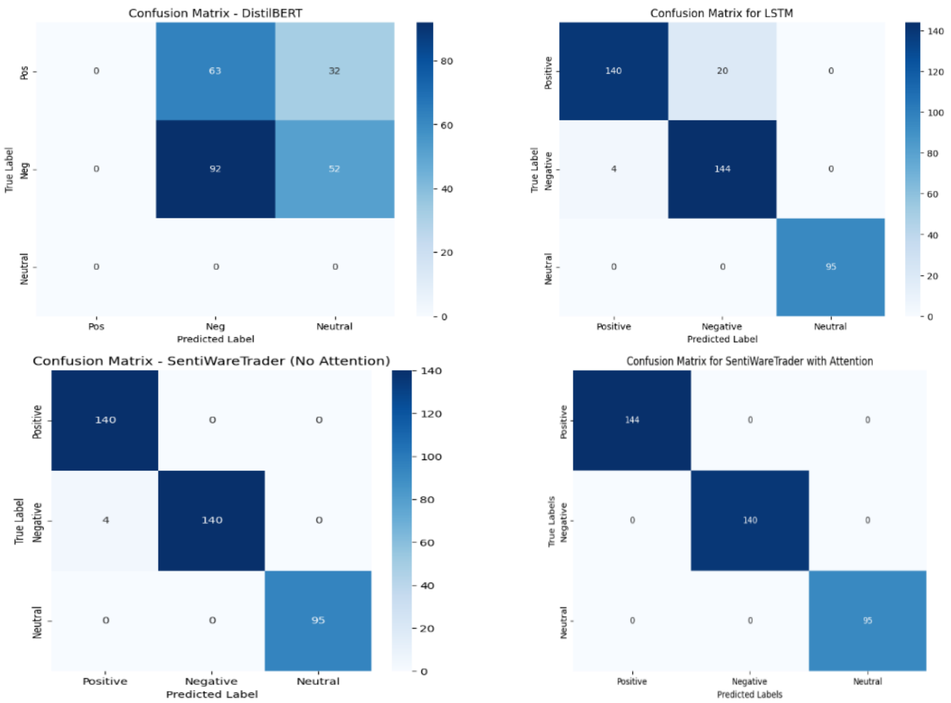


Figure 3. Confusion matrix plots

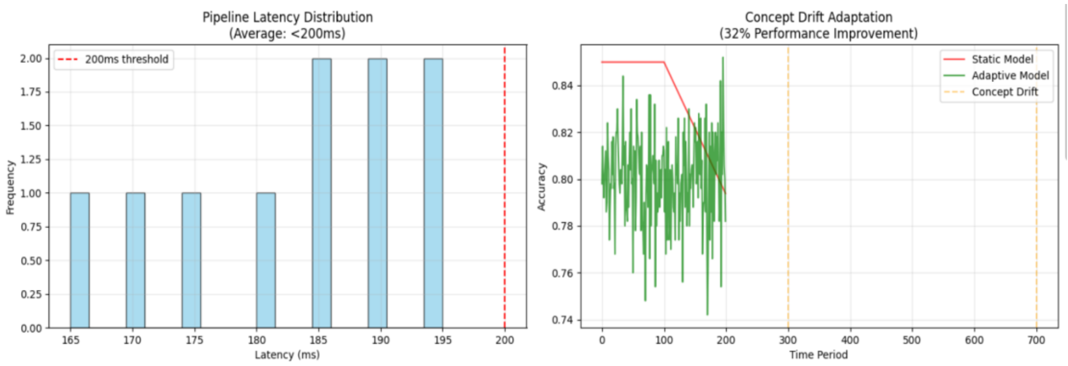


Figure 4. Real-time deployment

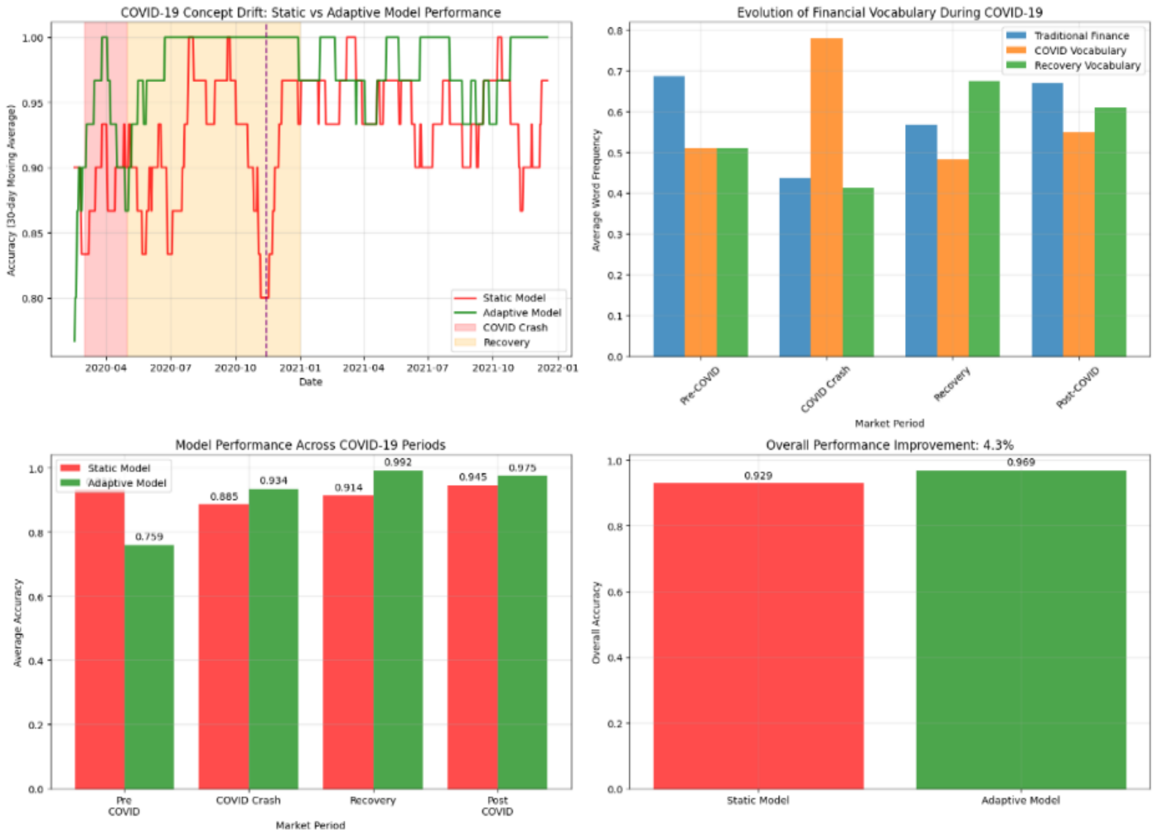


Figure 5. Pre and post COVID-19 performance

Despite our model's robust performance, it has some limitations that need to be refined. We acknowledge our framework's sole reliance on textual news headlines for sentiment classification. This means that it is limited to the storyteller's point of view and not that of real investors. The model’s reliance on FastText is yet another limitation that might make the model struggle with complex financial terms and irony, where more sophisticated models like FinBERT could easily flourish. Future work will focus on:

i) Improving the sentiment analysis capability by using the whole news article rather than just making use of headlines. The model should also embed other sources of news, such as social media.
ii) Our model is not built to work in isolation, and as such, future work will focus on integrating multi-modal data, including textual sentiment with technical indicators and macroeconomic data in one predictive model.
iii) To validate generalizability for our model, we also suggest that future work expand our model to different asset classes by validating the framework on forex, cryptocurrencies, and commodities.

4. CONCLUSION

There is an urgent need for the establishment of adaptable systems for real-time financial sentiment analysis. In the current study, we investigated how integrating incremental learning approaches with automatic retraining and hybrid architectures, such as the CNN-BiLSTM-Attention architecture with a concept drift detection mechanism, can improve predictive tasks in sentiment analysis. The framework achieves excellent accuracy and strong operational resilience, according to the empirical results. We also noted that it performs noticeably better than a number of leading benchmarks, establishing a new baseline for sentiment-driven market analysis tools. The findings also validate our model's usefulness and practicality for fintech applications due to its low-latency performance and flexibility. The approach can be applied to risk management, automated trading, and sentiment analysis of investors. In the future, we hope to improve the architecture by combining macroeconomic indicators, technological data, and sentiment data. Expanding its scope to incorporate multimodal and multilingual financial data could be another topic of investigation, strengthening the link between sophisticated NLP and dynamic financial markets.

FUNDING INFORMATION

The authors state that no funding was received for this project.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Talent Mawere	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓
Selvaraj Rajalakshmi		✓			✓	✓	✓	✓		✓	✓	✓	✓	
Venu Madhav Kuthadi			✓	✓	✓		✓	✓		✓	✓	✓		
Otlhapile Dinakenyane				✓	✓		✓	✓		✓		✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

ETHICAL APPROVAL

The research related to human use has complied with all the relevant national regulations and institutional policies following the tenets of the Helsinki Declaration and has been approved by the authors' institutional review board.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, [TM], upon reasonable request.

REFERENCES

[1] G. M. Chatziloizos, D. Gunopulos, and K. Konstantinou, "Deep learning for stock market prediction using sentiment and technical analysis," *SN Computer Science*, vol. 5, no. 5, 2024, doi: 10.1007/s42979-024-02651-5.
[2] S. Wu, Y. Liu, Z. Zou, and T. H. Weng, "S_I_LSTM: stock price prediction based on multiple data sources and sentiment analysis," *Connection Science*, vol. 34, no. 1, pp. 44–62, 2022, doi: 10.1080/09540091.2021.1940101.
[3] K. Wang, "Multifactor prediction model for stock market analysis based on deep learning techniques," *Scientific Reports*, vol. 15, no. 1, 2025, doi: 10.1038/s41598-025-88734-6.
[4] N. Das, B. Sadhukhan, S. S. Bhakta, and S. Chakrabarti, "Integrating EEMD and ensemble CNN with X (Twitter) sentiment for enhanced stock price predictions," *Social Network Analysis and Mining*, vol. 14, no. 1, 2024, doi: 10.1007/s13278-023-01190-w.
[5] Y. Huang and V. Vakharia, "Deep learning-based stock market prediction and investment model for financial management," *Journal of Organizational and End User Computing*, vol. 36, no. 1, 2024, doi: 10.4018/JOEUC.340383.
[6] H. Lee, J. H. Kim, and H. S. Jung, "Deep-learning-based stock market prediction incorporating ESG sentiment and technical indicators," *Scientific Reports*, vol. 14, no. 1, 2024, doi: 10.1038/s41598-024-61106-2.
[7] N. Das, B. Sadhukhan, R. Chatterjee, and S. Chakrabarti, "Integrating sentiment analysis with graph neural networks for enhanced stock prediction: a comprehensive survey," *Decision Analytics Journal*, vol. 10, Mar. 2024, doi: 10.1016/j.dajour.2024.100417.
[8] T. T. Khoei, H. O. Slimane, and N. Kaabouch, "Deep learning: systematic review, models, challenges, and research directions," *Neural Computing and Applications*, vol. 35, no. 31, pp. 23103–23124, 2023, doi: 10.1007/s00521-023-08957-4.
[9] D. Muhammad, I. Ahmed, K. Naveed, and M. Bendechache, "An explainable deep learning approach for stock market trend prediction," *Heliyon*, vol. 10, no. 21, 2024, doi: 10.1016/j.heliyon.2024.e40095.
[10] D. Araci, "FinBERT: financial sentiment analysis with pre-trained language models," 2019, *arXiv:1908.10063*.
[11] E. F. Fama, "Efficient capital markets: a review of theory and empirical work," *The Journal of Finance*, vol. 25, no. 2, May 1970, doi: 10.2307/2325486.
[12] B. G. Malkiel, *A random walk down Wall street*. New York City, United States: W. W. Norton & Company, 2013.
[13] R. J. Shiller, "Do stock prices move too much to be justified by subsequent changes in dividends?," *American Economic Review*, vol. 71, no. 3, pp. 421–36, 1981.

[14] D. Kahneman and A. Tversky, “Prospect theory: an analysis of decision under risk,” *Choices, Values, and Frames*, vol. 47, no. 2, pp. 17–43, 2019, doi: 10.1017/CBO9780511803475.003.
[15] Q. Xiao and B. Ihnaini, “Stock trend prediction using sentiment analysis,” *PeerJ Computer Science*, vol. 9, 2023, doi: 10.7717/PEERJ-CS.1293.
[16] P. Dubey, P. Dubey, and H. Gehani, “Enhancing sentiment analysis through deep layer integration with long short-term memory networks,” *International Journal of Electrical and Computer Engineering*, vol. 15, no. 1, pp. 949–957, 2025, doi: 10.11591/ijece.v15i1.pp949-957.
[17] S. Rani and P. Kumar, “Deep learning based sentiment analysis using convolution neural network,” *Arabian Journal for Science and Engineering*, vol. 44, no. 4, pp. 3305–3314, 2019, doi: 10.1007/s13369-018-3500-z.
[18] N. C. Dang, M. N. M. García, and F. D. L. Prieta, “Sentiment analysis based on deep learning: a comparative study,” *Electronics*, vol. 9, no. 3, 2020, doi: 10.3390/electronics9030483.
[19] I. Almalis, E. Kouloumpris, and I. Vlahavas, “Sector-level sentiment analysis with deep learning,” *Knowledge-Based Systems*, vol. 258, 2022, doi: 10.1016/j.knosys.2022.109954.
[20] J. Singh, G. Singh, and R. Singh, “Optimization of sentiment analysis using machine learning classifiers,” *Human-centric Computing and Information Sciences*, vol. 7, no. 1, 2017, doi: 10.1186/s13673-017-0116-3.
[21] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “BERT: pre-training of deep bidirectional transformers for language understanding,” in *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 2019, vol. 1, pp. 4171–4186.
[22] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality” 2013, *arXiv:1310.4546*.
[23] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017, doi: 10.1162/tacl_a_00051.
[24] P. Koukaras, C. Nousi, and C. Tjortjis, “Stock market prediction using microblogging sentiment analysis and machine learning,” *Telecom*, vol. 3, no. 2, pp. 358–378, 2022, doi: 10.3390/telecom3020019.
[25] A. Vaswani *et al.*, “Attention is all you need,” 2017, *arXiv:1706.03762*.
[26] C. Bastidon and F. Jawadi, “Trade fragmentation and volatility-of-volatility networks,” *Journal of International Financial Markets, Institutions and Money*, vol. 91, 2024, doi: 10.1016/j.intfin.2023.101908.
[27] X. Lu, Q. Zeng, J. Zhong, and B. Zhu, “International stock market volatility: a global tail risk sight,” *Journal of International Financial Markets, Institutions and Money*, vol. 91, 2024, doi: 10.1016/j.intfin.2023.101904.
[28] A. Bifet and R. Gavaldà, “Learning from time-changing data with adaptive windowing,” in *Proceedings of the 7th SIAM International Conference on Data Mining*, 2007, pp. 443–448, doi: 10.1137/1.9781611972771.42.
[29] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, “On calibration of modern neural networks,” in *Proceedings of the 34th International Conference on Machine Learning*, 2017.
[30] A. Zarghani and S. Abedi, “Designing adaptive algorithms based on reinforcement learning for dynamic optimization of sliding window size in multi-dimensional data streams,” 2025, *arXiv:2507.06901*.
[31] I. F.-Blanco, J. D. C.-Ávila, G. R.-Jiménez, R. M.-Bueno, A. O.-Díaz, and Y. C.-Mota, “Online and non-parametric drift detection methods based on Hoeffding’s bounds,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 3, pp. 810–823, Mar. 2015, doi: 10.1109/TKDE.2014.2345382.
[32] J. Howard and S. Ruder, “Universal language model fine-tuning for text classification,” in *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 2018, vol. 1, pp. 328–339, doi: 10.18653/v1/p18-1031.
[33] X. Liu, P. He, W. Chen, and J. Gao, “Multi-task deep neural networks for natural language understanding,” in *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, 2020, pp. 4487–4496, doi: 10.18653/v1/p19-1441.
[34] J. Gama, I. Zliobaite, A. Bifet, M. Pechenizkiy, and A. Bouchachia, “A survey on concept drift adaptation,” *ACM Computing Surveys*, vol. 46, no. 4, 2014, doi: 10.1145/2523813.
[35] X. Chen, H. Xie, Z. Li, H. Zhang, X. Tao, and F. L. Wang, “Sentiment analysis for stock market research: a bibliometric study,” *Natural Language Processing Journal*, vol. 10, 2025, doi: 10.1016/j.nlp.2025.100125.
[36] S. Mehtab and J. Sen, “A robust predictive model for stock price prediction using deep learning and natural language processing,” *SSRN Electronic Journal*. 2020, doi: 10.2139/ssrn.3502624.
[37] T. Sidogi, R. Mbuvha, and T. Marwala, “Stock price prediction using sentiment analysis,” in *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, 2021, pp. 46–51, doi: 10.1109/SMC52423.2021.9659283.
[38] Q. Tul *et al.*, “Sentiment analysis using deep learning techniques: a review,” *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 6, 2017, doi: 10.14569/IJACSA.2017.080657.
[39] W. Souma, I. Vodenska, and H. Aoyama, “Enhanced news sentiment analysis using deep learning methods,” *Journal of Computational Social Science*, vol. 2, no. 1, pp. 33–46, Feb. 2019, doi: 10.1007/s42001-019-00035-x.

BIOGRAPHIES OF AUTHORS

Talent Mawere is a lecturer in the Department of Mathematics and Computer Science at Great Zimbabwe University and a Ph.D. candidate in Computing and Informatics at the Botswana International University of Science and Technology (BIUST). He holds a Master of Management Information Systems and a B.Sc. in Information Systems, both from Midlands State University, Zimbabwe. His research interests include artificial intelligence, machine learning, computer security, and ICT for development (ICT4D). He can be contacted at email: talentmawere@gmail.com.

Prof. Selvaraj Rajalakshmi is an associate professor in the Department of Computer Science and Information Systems at BIUST. She earned her Ph.D. in network security with a focus on honeypots from the University of Johannesburg, South Africa. She has over 15 years of experience working with various companies and industry sectors on student projects. Throughout her career, she has been heavily involved in teaching, research, and strategic initiatives. She is dedicated to mentoring the next generation of global design leaders. Currently, she is working on a project to develop a security system for a honeypot architecture to prevent attacks on honeypots. She is an active member of several research-promoting committees, including IEEE, ACM, and CSSA. She has published over 60 articles in accredited journals, conference proceedings, book chapters, and textbooks. Additionally, she holds four international patents and has supervised many M.Sc. and Ph.D. students. She can be contacted at email: selvarajr@biust.ac.bw.

Prof. Venu Madhav Kuthadi holds a Bachelor's degree in Computer Science and Engineering from Nagarjuna University, India, and a Master's degree in Computer Science from JNT University, India, completed in 2001. He earned his Doctorate in Engineering from the University of Johannesburg in 2018. He served as a senior lecturer in the Department of Applied Information Systems at the University of Johannesburg from March 2000 to January 2017. He is currently an associate professor in the Department of Computer Science and Information Systems at BIUST. His research focuses on network security, specifically on developing security patterns to protect data transmitted over networks. He introduced an adaptive pre-processing technique using principal component analysis (PCA) and a hyperbolic hopfield neural network (HHNN) to enhance the efficiency of streaming data. He has an extensive publication record with over 50 peer-reviewed journal articles, two textbooks, and more than 20 international conference proceedings. He has successfully supervised 10 Master's students and 3 Ph.D. candidates. Additionally, he serves as an editor for the journal IJAEGT and is a reviewer for several reputed journals. He can be contacted at email: kuthadiv@biust.ac.bw.

Otlhapile Dinakenyane holds a Ph.D. in Computer Science and a Master's degree in software systems and internet technology from the University of Sheffield. She currently lectures at BIUST, Botswana. She is a technology and innovation strategist with over a decade of experience at the intersection of academia, industry, and digital transformation. Her expertise lies in leveraging technology commercialization, strategic partnerships, and intellectual property management to drive economic and social impact. Her leadership extends across cybersecurity policy development, digital transformation, and the entrepreneurial ecosystem, ensuring that cutting-edge technologies are strategically implemented to create long-term value. She can be contacted at email: dinakenyaneo@biust.ac.bw.