Journal of Organizational and End User Computing
Volume 37 • Issue 1 • January-December 2025

The Financial Institution Text Data Mining and Value Analysis Model Based on Big Data and Natural Language Processing

Juan Yang
https://orcid.org/0009-0003-9526-2379
Chongqing Technology and Business University, China

Yu Bai
Chongqing Technology and Business University, China

Jie Gong
Chongqing Technology and Business University, China

Menghui Han
Chongqing Technology and Business University, China

ABSTRACT

Financial markets are inherently complex and influenced by a variety of factors, making it challenging to predict trends and detect key events. Traditional models often struggle to integrate both structured, or numerical, and unstructured, or textual, data; additionally, they fail to capture temporal dependencies or the dynamic relationships between financial entities. To address this, the multidimensional integrated model for financial text mining and value analysis (MI-FinText), was proposed. MI-FinText integrated multi-task learning, temporal graph convolutional networks and dynamic knowledge graph construction. MI-FinText simultaneously performed sentiment analysis, event detection, and value prediction by learning shared representations across tasks and modeling time-dependent relationships between financial events. MI-FinText continuously updated a dynamic knowledge graph to reflect the evolving financial landscape, enabling real-time insights.

KEYWORDS

Financial Text Mining, Multi-task Learning, Temporal Graph Convolutional Networks, Knowledge Graph, Value Prediction

INTRODUCTION

Financial markets are inherently complex and volatile (Shavandi & Khedmati, 2022), requiring comprehensive analysis of both structured data, such as stock prices, trading volumes; and unstructured data, such as news articles, earnings reports, and social media posts (Krasnyuk et al., 2022). Financial markets are influenced by numerous factors (Thakkar & Chaudhari, 2021b), therefore extracting meaningful insights from vast amounts of data has become a critical task for analysts and decision-makers (Ashtiani & Raahemi, 2023). Traditional methods of financial analysis primarily focus on quantitative metrics, often overlooking the qualitative factors that are crucial for a holistic market understanding (Bello, 2023). Financial text mining, which applies natural language processing

DOI: 10.4018/JOEUC.374213

(NLP) and machine learning (ML) techniques to analyze unstructured textual data, has emerged as a vital tool for identifying patterns, trends, and predicting future market behaviors (Dash, 2022).

Despite rapid growth in financial text mining research, several challenges remain. First, the integration of structured and unstructured data has proven to be difficult (Nahar et al., 2024), as these data types have different formats, scales, and temporal dynamics (Tarka & Jędrych, 2020). Second, financial data is inherently volatile; market trends and asset prices are often influenced by sudden shifts in public opinion or events, and this high volatility makes it difficult to develop accurate predictive models. Financial text mining and value analysis (MI-FinText) addresses this challenge by incorporating temporal modeling techniques, such as temporal graph convolutional networks (T-GCNs), which capture the evolving nature of financial data and account for sudden market changes. Third, temporal dependencies in financial data, such as market trends and price fluctuations, are often ignored or inadequately modeled in traditional approaches (Zhang, 2024). To handle these dependencies, MI-FinText uses T-GCN, allowing the model to better understand long-term trends and market shifts over time. Fourth, while existing models excel at individual tasks, such as sentiment analysis or event detection (Abdullah & Ahmet, 2022), they often lack a comprehensive framework that integrates multiple tasks; this limits their ability to provide a holistic understanding of financial markets (Lu et al., 2023). MI-FinText overcomes this limitation by utilizing multi-task learning (MTL), anabaenas sentiment analysis, event detection, and value prediction, while also sharing relevant features across tasks. Finally, dynamic relationships between financial entities—such as companies, products and markets—and events need to be continuously captured in real time to maintain the relevance of insights (Allioui & Mourdi, 2023). MI-FinText incorporates a dynamic knowledge graph that updates continuously, reflecting mergers, acquisitions, and market fluctuations.

To address these challenges, this study proposed a novel solution: the multidimensional integrated model for MI-FinText. The approach integrated several advanced components, including MTL (Zhang & Yang, 2021), temporal modeling, and dynamic knowledge graph construction (Liang et al., 2024). By leveraging MTL, the model simultaneously performs sentiment analysis, event detection, and value prediction tasks, thereby facilitating the extraction of shared features across tasks (Zhang et al., 2025). Temporal patterns are modeled using T-GCN, allowing the model to capture the evolving nature of financial data and predict trends more accurately (Zhao et al., 2019). In addition, a dynamic knowledge graph was constructed that continuously updated financial relationships, such as mergers, acquisitions, or market changes. This provided a rich representation of financial entities and their interdependencies.

The main technical components of the proposed approach were threefold. First, the sentiment analysis component used a pre-trained financial language model (FinBERT), a pre-trained transformer model fine-tuned for financial data, to classify the sentiment of financial documents (Fang et al., 2023; Liu et al., 2021). Second, the event detection component applied deep learning models to identify significant financial events from text, such as market changes or corporate actions (DeMatteo et al., 2024; Li et al., 2025). Third, the value prediction module employed a regression-based approach to predict the future values of financial assets, based on historical data and textual content (Qi et al., 2023). Finally, the knowledge graph construction component built and updated a dynamic graph of financial entities and relationships, enabling the model to leverage both historical context and real-time data to improve decision-making.

In summary, MI-FinText addressed several key challenges in financial data analysis, providing an integrated framework that captured temporal dependencies, modeled dynamic relationships, and performed multiple tasks simultaneously. By combining state-of-the-art techniques such as MTL, T-GCN, and dynamic knowledge graphs, the model significantly improved prediction accuracy, event detection, and sentiment analysis. This work contributes to the growing field of financial text mining and offers a more comprehensive solution for understanding and predicting financial markets.

RELATED WORK

In recent years, the combination of text mining, NLP, and ML techniques has greatly enhanced the ability to analyze and predict financial markets (Abinaya & Kumar, 2024; Dimlo et al., 2024). A substantial body of research has explored various methods to extract insights from financial text and integrate them with structured data (Gupta et al., 2020). This section reviews the existing literature on financial text mining, the challenges faced by current approaches, and introduces method proposed in this study, MI-FinText, highlighting its contributions and how it improves upon existing methods.

The application of NLP in financial markets has seen rapid development, particularly in sentiment analysis, event detection, and market prediction tasks (Xiao et al., 2024). Early works primarily focused on analyzing textual data to detect sentiments or opinions from financial news, stock reports, and social media posts (Nyakurukwa & Seetharam, 2023). One of the first notable efforts was the creation of a financial sentiment lexicon, which was used to measure sentiment in financial news and reports (Frankel et al., 2022). More recently, FinBERT set a new standard for financial sentiment analysis by achieving remarkable results in classifying sentiment in financial reports, such as earnings calls and financial news.

On the event detection side, significant progress has been made in identifying key financial events, such as mergers, acquisitions, and market fluctuations (El Ghoul et al., 2022). Traditional approaches relied on rule-based systems to identify keywords and phrases associated with specific financial events (Dogra, 2021; Zhou et al., 2024). These methods, however, have been limited by their inability to capture complex, context-dependent events. More recent work has employed deep learning models like Bidirectional Long Short-Term Memory (BiLSTM) and transformer-based models to automatically detect events in financial text (Li et al., 2023), thereby allowing for more robust and dynamic event extraction. These models can learn contextual information and better generalize across different types of financial events (Feng & Sinchai, 2024).

Value prediction models have focused on predicting stock prices and market trends, based on historical financial data and textual information (Li et al., 2024; Rouf et al., 2021). Traditional regression models and ML techniques such as support vector machines (SVM) and random forests were used to predict financial asset values (Sadorsky, 2021), relying heavily on numerical data such as historical prices and trading volumes (Kumbure et al., 2022). Recent advancements in neural networks, particularly long short-term memory (LSTM) and Gated Recurrent Unit (GRU) architectures, have been shown to better capture temporal dependencies in market data, leading to improvements in forecasting accuracy (Thakkar & Chaudhari, 2021a). These models, however, often struggle with integrating both textual and numerical data in a unified framework.

Despite the progress made in financial text mining, several challenges remain regarding fully capturing the complexity of financial markets. One major challenge is the integration of structured and unstructured data. Traditional models typically treat these two data types separately, resulting in suboptimal performance. Textual data, such as financial reports or news articles, contains rich, qualitative information; numerical data, such as stock prices and trading volumes, provides quantitative insights. Most existing methods fail to fully integrate these data types in a way that leverages their complementary strengths, and while some recent works have attempted to fuse these data sources, they often rely on feature engineering or manual alignment of text and numerical data, which limits scalability and the ability to handle large datasets.

Another challenge is the temporal modeling of financial data. Financial events and stock prices are often influenced by time-dependent factors, yet many models fail to capture the complex temporal dependencies between these events. Standard ML models such as SVM or random forests typically ignore the sequential nature of financial data, whereas deep learning models like LSTM have made significant strides in capturing temporal dependencies, but they still struggle with high-dimensional, real-time data and require substantial computational resources.

A third challenge is the dynamic nature of financial relationships. Financial markets are constantly evolving, and the relationships between entities—such as companies, products, or markets—change over time, due to new events or market shifts. Current knowledge graphs often fail to keep up with these changes, as they are typically built from static data. Maintaining an up-to-date graph that reflects the evolving relationships between financial entities is crucial for real-time analysis, but few models address this issue adequately.

The proposed MI-FinText model addressed several key limitations in the current literature by offering a unified framework that integrates MTL, temporal modeling, and dynamic knowledge graph construction. Unlike existing models that treat text and numerical data separately, MI-FinText fused both types of data to enhance prediction accuracy and event detection. By using MTL, the model simultaneously performed sentiment analysis, event detection, and value prediction, allowing it to learn shared representations across tasks and improve generalization.

The use of T-GCN in MI-FinText provided a novel approach to modeling temporal dependencies in financial data. T-GCN allowed the model to capture the evolving relationships between financial entities over time, improving the prediction of market trends and the identification of significant events (Almayyan & Al Ghannam, 2024). This temporal modeling was particularly important for capturing the long-term dependencies between financial events, which are often overlooked by traditional models.

Furthermore, MI-FinText incorporated a dynamic knowledge graph, which continuously updated financial relationships based on new data. This approach ensured that the model could reflect real-time changes in the market, such as mergers, acquisitions, or shifts in stock performance. The dynamic nature of the knowledge graph made MI-FinText particularly suitable for real-time financial analysis and decision-making, setting it apart from static knowledge graph models used in prior work.

Overall, MI-FinText showed potential to significantly advance the state of financial text mining by integrating cutting-edge techniques for MTL, temporal modeling, and dynamic knowledge graph construction, offering a more comprehensive solution for predicting financial trends, detecting events, and analyzing market behavior. Unlike existing methods, MI-FinText effectively bridged the gap between structured and unstructured data, providing a unified framework that improved prediction accuracy and provided deeper insights into financial markets.

METHODOLOGY

In this next section, the detailed methodology behind the multidimensional integrated model for MI-FinText is presented (see Figure 1). This model integrated several advanced components, including financial text mining, sentiment analysis, knowledge graph construction, temporal data modeling, and MTL, to address the complexities of analyzing and predicting financial value, from both unstructured and structured data. Each component is described in detail below, along with the associated mathematical formulations and explanations.

Figure 1. Model diagram of the multidimensional integrated model for financial text mining and value analysis (MI-FinText)

Financial text mining was the first critical step in the MI-FinText model. The goal of financial text mining is to extract meaningful information from unstructured documents, such as 10-K filings, earnings call transcripts, and news articles. This information is used to identify key entities, detect significant events, and form the foundation for downstream tasks such as sentiment analysis and event prediction.

The raw financial documents underwent several preprocessing steps to convert the unstructured text into a usable format. These steps are outlined below.

1. Tokenization: Tokenization split the raw text into smaller units (tokens), such as words, phrases, or sentences. The tokenization process was mathematically represented as:

$$T = \text{Tokenize}(D)$$

where T was the set of tokens and D was the raw document.

2. Lemmatization: Lemmatization reduced words to their base or root form, for example, transforming "running" into "run." The lemmatization operation was represented as:

$$T_{\text{lem}} = \text{Lemmatize}\ (T)$$

where T_{lem} was the lemmatized set of tokens.

3. Stop-word removal: Common words (such as "the," "and," "is") that do not add significant meaning to the document were removed during preprocessing.

Named Entity Recognition

Once the text was preprocessed, named entity recognition (NER) was used to identify and extract relevant entities from the document. These entities could be companies, financial terms, products, and other important identifiers in the financial context. The NER process was mathematically represented by:

$$E = \text{NER}\ (T_{\text{lem}})$$

where E was the set of entities extracted from the tokenized and lemmatized text T_{lem}.

After extracting entities, event extraction was performed, which involved identifying significant financial events such as mergers, acquisitions, and market fluctuations. Each event was associated with a financial impact Δ_{f}, for example, a change in stock price. The event extraction process could be formulated as:

$$e_i = \text{ExtractEvent}\ (E, T)$$

where e_i represented an event, E was the set of extracted entities, and T was the original document. Additionally, the event was associated with a time t and its corresponding financial impact Δ_{f}, represented as:

$$e_i = \text{Event}\ \left(e_{entity}, t, \Delta_f\right)$$

where e_{entity} referred to the involved financial entity, t was the timestamp of the event, and Δ_{f} was the financial impact, such as a percentage change in stock price.

Noise Handling in Financial Data

Financial data, especially from unstructured sources such as social media and news articles, often contains noise in the form of irrelevant content, inconsistencies, and misleading information. To address this, MI-FinText integrated several techniques during preprocessing and model training to minimize the impact of noise, thereby ensuring reliable performance.

Data preprocessing noise reduction started with preprocessing steps like tokenization, lemmatization, and stop-word removal. To further reduce noise from unstructured sources, irrelevant content such as spam and non-financial posts was filtered, using keyword-based filtering and entity recognition. Let $\boldsymbol{D}$ represent the raw financial dataset, and after preprocessing, the filtered dataset $\boldsymbol{D}^{'}$ was given by:

$$\mathbf{D}^{'} = \text{filter}\left(\mathbf{D}, \text{keywords, entities}\right)$$

where keywords and entities referred to terms and financial entities used to exclude irrelevant content.

MI-FinText employed regularization techniques, such as dropout and L2 regularization to make the model more robust to noise. L2 Regularization, also known as weight decay, is a technique used to prevent overfitting in neural networks by penalizing large weights in the loss function. Dropout was expressed as:

$$\mathbf{h}^{(l)} = \text{Dropout}\left(\mathbf{h}^{(l)}; p\right)$$

where $\boldsymbol{h}^{(l)}$ was the hidden layer output at layer l, and pp was the probability of dropping out neurons. This helped prevent overfitting by forcing the model to rely on a broader set of features. L2 regularization was used to penalize large weights and was represented as:

$$\mathscr{L}_{L2} = \lambda \sum_{i=1}^{n} w_i^2$$

where w_i were the model weights and λ was the regularization strength.

To reduce the noise from high-dimensional data, principal component analysis (PCA) and latent dirichlet allocation (LDA) were used. PCA reduced dimensionality by finding the principal components $\boldsymbol{P}$ that explained the maximum variance:

$$\mathbf{P} = \arg \underset{\mathbf{P}}{max} \frac{\mathbf{P}^T \mathbf{X}^T \mathbf{X} \mathbf{P}}{\| \mathbf{P} \|^2}$$

where $\boldsymbol{X}$ was the data matrix, and $\boldsymbol{P}$ were the principal components.

By combining these methods, MI-FinText effectively handled noisy financial data, improving model reliability in real-world applications.

Sentiment Analysis

Sentiment analysis was an essential task in the MI-FinText model, as it helped identify the sentiment associated with financial events, whether positive, negative, or neutral. Understanding sentiment is crucial for predicting market trends and forecasting stock prices.

For sentiment classification, FinBERT was used. The sentiment classification of a financial document TT was represented as:

$$S = \text{FinBERT}(T)$$

where S was the predicted sentiment label—positive, negative, or neutral—for the document T.

To further enhance sentiment analysis, a weighted sentiment score was introduced that adjusted the sentiment, based on the financial impact of the associated event. The sentiment score was computed as:

$$S_{\text{score}} = \alpha S + \beta \Delta_f$$

where α and β were hyperparameters that controlled the weight of sentiment and financial impact, S was the sentiment label, and $_{\text{Äf}}$ represented the financial impact associated with the sentiment.

This weighted sentiment score helped quantify how the sentiment expressed in a document was related to the actual financial impact of the described events.

Knowledge Graph Construction

The dynamic construction of the knowledge graph was a crucial feature in MI-FinText, allowing the model to adapt to rapidly changing financial markets. This knowledge graph not only captured relationships between financial entities—such as companies, sectors, and stock prices—but also continuously updated in real-time to reflect the latest market events. This dynamic updating mechanism was essential for addressing the volatility inherent in financial markets, where relationships between entities can shift dramatically due to events like mergers, acquisitions, or market fluctuations.

Regarding real-time event integration, one of the key features of MI-FinText's dynamic knowledge graph was its ability to directly link financial events to market movements. When a financial event occurred, such as a company merger, the knowledge graph updated to reflect the new relationship between the involved companies, allowing the model to track how this event might influence stock prices or sector performance. For instance, when a merger between two companies was announced, the knowledge graph updated to show a "merger" relationship between these two companies, which helped the model understand the potential market impact.

This was mathematically represented as:

$$G(t+1) = G(t) \cup \text{Ä}G(t)$$

where $G(t)$ represented the knowledge graph at time t, and $\Delta G(t)$ was the newly detected event, such as a merger or acquisition, that updated the graph.

Similarly, during stock market fluctuations, such as sudden crashes or bull runs, the knowledge graph dynamically updated to capture how various stocks and financial entities were influenced by these events, ensuring that the model adapted to current market conditions.

Event-Driven Updates

The process of updating the knowledge graph involved several steps, detailed below.

1. Event detection: Financial documents, including news articles, earnings reports, and market analysis, were processed to detect significant financial events, such as mergers, acquisitions, and market shocks. These events triggered the update mechanism in the knowledge graph.
2. Relationship extraction: Once an event was detected, relationships between the involved entities were extracted and represented as graph edges. For example, in the case of a merger, a "merger" relationship was formed between the two companies. The mathematical formulation of this relationship was expressed as:

$$r(v_i, v_j) = \text{merger}(v_i, v_j)$$

where $r(v_i, v_j)$ represented the relationship between entities v_i and v_j, and the function merger (v_i, v_j) defined the specific relationship between the two companies involved in the merger.

3. Temporal updates: Relationships were then updated with time-dependent data, such as the changes in stock prices or market sentiment before and after the event. The temporal evolution of the relationships was modeled as:

$$r(v_i, v_j, t) = f(r(v_i, v_j), X(t))$$

where $r(v_i, v_j, t)$ represented the relationship between entities v_i and v_j at time t, and $X(t)$ represented the time-dependent financial data, such as stock prices or market sentiment.

CONTRIBUTION TO MARKET PREDICTIONS

The continuous updates to the knowledge graph significantly contributed to MI-FinText's ability to make accurate predictions in real time. The key contributions are outlined below.

1. Real-time adaptation: As new events unfolded, the knowledge graph updated to reflect these changes. This enabled the model to make timely predictions based on the latest available market information. The real-time update of the knowledge graph was expressed as:

$$\hat{y}_t = \text{Predict}(G(t), X(t))$$

where $\hat{y}_t$ was the prediction at time t, and $X(t)$ was the real-time market data.

2. Event-driven market predictions: By linking financial events directly to market movements, the model predicted how specific events might influence stock prices or sectors. For example, when a merger occurred, the model updated its predictions based on historical data and market responses to similar events:

$$\hat{y}_t = f(G(t+1), X(t))$$

3. Improved forecasting: The real-time, dynamic nature of the knowledge graph allowed MI-FinText to forecast future market trends more accurately by continuously incorporating up-to-date financial relationships and market sentiment.

Temporal Data Modeling

Financial data is inherently temporal, meaning that the relationships between events and market conditions evolve over time. To model the temporal aspect, T-GCN was utilized, which combined the power of graph convolutional networks (GCNs) with temporal data.

Temporal Graph Representation

In temporal modeling, the knowledge graph is updated to reflect the changing state of financial entities over time. At each time t, the temporal graph $G(t)$ consisted of entities and time-dependent relationships. The T-GCN model updated the node embeddings $H(t)$ over time by incorporating both spatial and temporal features:

$$H^{(t+1)} = \sigma\left(\widehat{A}\, H^{(t)}\, W^{(t)} + X\left(t\right)\right)$$

where $H^{(t)}$ represented the node embeddings at time t, $\widehat{A}$ was the normalized adjacency matrix representing the relationships between financial entities, $W^{(t)}$ was the weight matrix, and $X(t)$ represented the time-dependent financial features, such as stock prices, trading volume, and sentiment scores.

By using T-GCN, the model simultaneously captured both the spatial (entity relationships) and temporal (financial data evolution) dependencies, improving the accuracy and interpretability of financial predictions.

Temporal Message Passing

T-GCN updated the node embeddings by passing messages through the graph, considering both the structure of the graph and the temporal data. The message-passing operation was formulated as:

$$H_i^{(t+1)} = \sigma\left(\sum_{j\in\mathcal{N}(i)} \tfrac{A_{ij}}{d_j} H_j^{(t)}\, W^{(t)} + X_i\left(t\right)\right)$$

where $H_i^{(t)}$ represented the feature vector for node ii at time t, A_{ij} was the adjacency matrix, $\mathcal{N}($ $i)$ represented the set of neighbors of node i, d_j was the degree of node j, and $X_i\left(t\right)$ was the temporal feature for node i at time t. The activation function σ introduced non-linearity to the model.

This message-passing mechanism enabled the model to propagate information across the graph, while considering both the structure of the graph and the temporal evolution of financial data, thereby capturing the dynamic interdependencies in the financial system.

Contribution of T-GCN to Model Performance

The integration of T-GCN into the MI-FinText framework significantly enhanced the model's ability to understand and predict financial data. By capturing both the temporal and spatial dependencies, T-GCN allowed the model to better understand how financial entities influence each other over time and how these influences evolve.

For instance, in stock market prediction, T-GCN was able to learn the long-term dependencies between financial events, such as mergers and market crises, and the reactions of different financial entities, such as stock prices and sector indices—this is crucial for forecasting trends and making more informed predictions. The temporal graph convolution mechanism also allowed the model to adapt dynamically to the rapidly changing nature of financial markets.

MTL Framework

The MI-FinText model employed MTL, a framework that allowed the model to learn multiple tasks simultaneously, while sharing representations across tasks. MTL improved model generalization and performance by leveraging common features learned from different tasks. The core tasks in MI-FinText are outlined below.

1. Sentiment analysis: Classifying the sentiment of financial documents.
2. Event detection: Identifying key financial events such as mergers, acquisitions, or market fluctuations.
3. Value prediction: Predicting future values of financial assets such as stock prices.

Loss Function for MTL

The overall loss function for MTL was the weighted sum of the individual losses for each task. This was defined as:

$$L = \lambda_1 L_{\text{sentiment}} + \lambda_2 L_{\text{event}} + \lambda_3 L_{\text{prediction}}$$

where $L_{\text{sentiment}}$ was the loss for sentiment classification, L_{event} was the loss for event detection, $L_{\text{prediction}}$ was the loss for value prediction, and $\lambda_1, \lambda_2, \lambda_3$ were hyperparameters controlling the relative importance of each task.

Shared Representations

MTL allowed for the learning of shared representations in the lower layers of the network, while task-specific layers were used to fine-tune the model for each individual task. This architecture enabled the model to leverage shared knowledge, improving performance across all tasks.

OVERALL FRAMEWORK

The multidimensional integrated model for MI-FinText was designed to address complex financial analysis tasks by integrating multiple components: financial text mining, sentiment analysis, knowledge graph construction, temporal data modeling, and MTL. The goal of MI-FinText was to process and analyze a variety of financial data, including textual data from documents such as 10-K filings, earnings reports, and news articles, as well as structured numerical data such as stock prices and trading volumes.

The overall framework of MI-FinText was structured as outlined below.

1. Data input and preprocessing: The first step involved collecting and preprocessing data from multiple sources. These included financial documents, financial news, social media, and market data. The preprocessing pipeline involved tokenization, lemmatization, NER and removing stop-words. Both textual and numerical data were cleaned, formatted, and prepared for analysis.
2. Financial text mining: The financial documents were analyzed to extract key entities (companies, financial terms, events), and significant financial events (mergers, acquisitions, earnings reports). This was achieved using advanced techniques such as NER and event extraction models.
3. Sentiment analysis: The sentiment of the extracted documents was classified using FinBERT, a pre-trained model specifically fine-tuned for financial texts. Sentiment analysis helped understand the tone of the document—whether positive, negative, or neutral—which was essential for understanding market sentiment.
4. Knowledge graph construction: A dynamic financial knowledge graph (FinDKG) was constructed to represent the relationships between entities such as companies, financial events, and market trends. The graph was updated dynamically with new financial documents and real-time market data, enabling the model to adapt to evolving financial conditions.
5. Temporal data modeling: The temporal aspect of financial data was captured using T-GCN, which allowed the model to process time-dependent relationships between financial events, stock prices, and trading volumes. This component helped capture the long-term and short-term dependencies in financial markets.
6. MTL: The model performed multiple tasks simultaneously using a MTL framework. These tasks included sentiment analysis, event detection, and value prediction. The model shared representations across tasks, improving performance while avoiding overfitting.

The MI-FinText framework was designed to provide a unified approach to analyzing and predicting financial trends, enabling the model to process both unstructured financial text and structured financial data. By integrating advanced techniques such as T-GCN for temporal modeling and MTL for joint task optimization, the framework ensured accurate, adaptable, and real-time financial analysis.

DATASETS

In this study, several datasets were utilized that were integral to the proposed multidimensional integrated model for MI-FinText. These datasets were selected to cover a range of essential tasks, such as financial text mining, sentiment analysis, event extraction, knowledge graph construction, and temporal data modeling. Specifically, the datasets used include the SEC Edgar Annual Financial Filings–021, financial dynamic knowledge graph (FinDKG), and financial document classification dataset.

The SEC Edgar Annual Financial Filings–2021 dataset consists of 10-K filings from publicly listed companies, providing comprehensive annual reports on their financial health. These filings contain critical sections such as the company's business summary, management discussion, financial statements, and additional sections that detail the company's performance and strategy. The inclusion of such detailed corporate reports makes this dataset invaluable for tasks involving sentiment analysis, event extraction, and financial analysis. The rich content allows for training models to recognize and classify various financial events and sentiments expressed by management, while also facilitating the construction of a knowledge graph that can map the relationships between companies, markets, and events. Given its comprehensive nature, this dataset served as the primary resource for training the financial text analysis component of MI-FinText. The data was divided into a training set (80%), a validation set (10%), and a test set (10%), to ensure robust model evaluation and generalization.

The FinDKG dataset served as a resource for building dynamic knowledge graphs in the financial domain, as it captures both qualitative and quantitative financial indicators through an extensive collection of global financial news, collected using the wayback machine. This dataset was especially useful for constructing and enhancing the FinDKG component of the MI-FinText model, which allowed for the mapping of complex relationships between financial entities, such as companies, stock prices, and market trends. The dynamic nature of this dataset supported the real-time learning component of the model, ensuring it was able to adapt to the rapidly changing financial environment. Like the SEC dataset, the data was split into subsets of training (80%), validation (10%), and test (10%), to provide a comprehensive evaluation across all tasks.

The financial document classification dataset is a collection of diverse financial documents, including letters, forms, emails, reports, invoices, presentations, and more. This dataset is particularly valuable for training MTL models capable of performing document classification, sentiment analysis, and entity extraction. While this dataset does not specifically focus on financial event extraction or temporal data modeling, it provides essential support for training models to recognize and categorize various types of financial documents, improving the overall accuracy of financial text mining systems. As with the other datasets, it was divided into subsets of training (80%), validation (10%), and test (10%).

The datasets used in this research, collectively, ensured the comprehensive coverage of key tasks in financial text mining and analysis. The training sets provided ample data for model development, while the validation sets helped in tuning hyperparameters and adjusting model configurations. The test sets served to evaluate the final model's performance on unseen data, ensuring that it generalized well to real-world applications. This structured approach to data partitioning ensured the robustness and reliability of the MI-FinText model, facilitating its application to a wide range of financial text mining tasks.

EVALUATION METRICS

This section will discuss the evaluation metrics used to assess the performance of the multidimensional integrated model for MI-FinText. Since the model integrated multiple tasks—including sentiment analysis, event detection, value prediction, and knowledge graph construction—different sets of metrics were used, tailored to each specific task. These metrics were designed to quantify the effectiveness, accuracy, and robustness of the model in various financial analysis tasks. Below are detailed the evaluation metrics used for sentiment analysis, event detection, and value prediction, as well as the overall evaluation strategy.

Sentiment Analysis Evaluation

Sentiment analysis plays a pivotal role in understanding market sentiment and its impact on financial markets. In this study, sentiment analysis was used to classify financial documents into sentiment categories, such as positive, negative, and neutral. To evaluate the sentiment analysis component, standard classification metrics were used.

Accuracy was the most used metric for classification tasks. It represented the proportion of correctly classified instances out of the total instances:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP (true positive), TN (true negative), FP (false positive), and FN (false negative) were the counts of correct and incorrect classifications.

Precision (P) measured the ratio of correct positive predictions to the total number of positive predictions. It helped evaluate the accuracy of positive class predictions:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall, (R), also known as sensitivity, measured the ratio of correct positive predictions to the total number of actual positive instances. It was crucial for identifying all instances of the positive class:

$$\text{Recall} = \frac{TP}{TP + FN}$$

The F1-score was the harmonic mean of precision and recall, providing a balanced measure of the model's performance in predicting the positive class:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

These metrics provided a comprehensive view of the sentiment analysis model's performance. The F1-score was particularly important when dealing with imbalanced data, where one sentiment class ran the risk of being underrepresented.

Event Detection Evaluation

Event detection involves identifying significant financial events—such as mergers, acquisitions, and stock market fluctuations—from financial documents. To evaluate the performance of the event detection component, metrics were used that are common in information extraction tasks.

P was similar to sentiment analysis; P for event detection measured the proportion of correctly detected events out of all detected events. It was calculated as:

$$\text{Precision} = \frac{TP}{TP + FP}$$

R was used for event detection; it measured the proportion of correctly detected events out of all the actual events present in the data:

$$\text{Recall} = \frac{TP}{TP + FN}$$

The F1-score for event detection was the harmonic mean of precision and recall, offering a balanced evaluation metric that considered both false positives and false negatives:

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Mean average precision (mAP) was used for ranked event detection tasks; the mAP, a commonly used metric, evaluated the average precision across all relevant events in a ranked list:

$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N} \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i}$$

where TP_i and FP_i refer to the true positives and false positives at the *i-th* threshold, respectively, and *NN* was the number of events.

These metrics ensured a thorough evaluation of the event detection system, particularly focusing on the model's ability to correctly identify and classify financial events from unstructured data.

Value Prediction Evaluation

The value prediction component was central to forecasting financial asset prices or market trends based on textual and numerical data. The prediction accuracy was evaluated using traditional regression metrics, as this task involved predicting continuous values, such as stock prices or asset values.

The mean absolute error (MAE) measures the average absolute difference between the predicted and actual values. It was calculated as:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} \left| y_i - \hat{y}_i \right|$$

where y_i was the actual value, $\hat{y}_i$ was the predicted value, and *N* was the number of data points. MAE provided a simple measure of how far off the predictions were from the actual values.

The mean squared error (MSE) calculates the average of the squared differences between the predicted and actual values. It was particularly useful for penalizing large errors:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N} \left(y_i - \hat{y}_i \right)^2$$

MSE gives more weight to larger errors, which makes it useful for tasks where large deviations are more detrimental.

The root mean squared error (RMSE) is the square root of MSE and provides the error in the same units as the target variable:

$$\text{RMSE} = \sqrt{\text{MSE}}$$

RMSE is particularly useful when comparing the prediction accuracy of models with different target units.

The R-squared (R^2) value is a statistical measure that indicates how well the predicted values match the actual values. It represents the proportion of the variance in the dependent variable that is predictable from the independent variables:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{N}\left(y_i - \bar{y}\right)^2}$$

where $\bar{y}$ was the mean of the actual values. R^2 valued closer to *1*, which indicated better model performance.

Knowledge Graph Evaluation

The evaluation of the FinDKG constructed as part of MI-FinText was essential to assess how well the graph represented relationships between financial entities and how accurately the knowledge was captured and utilized. For knowledge graph evaluation, the following metrics were considered, as outline below.

1. Entity linking accuracy: This metric evaluated how well the knowledge graph correctly linked entities to their corresponding real-world counterparts. It was calculated as the proportion of correctly linked entities:

$$\text{Entity Linking Accuracy} = \frac{\text{Correctly Linked Entities}}{\text{Total Entities}}$$

2. Graph completeness: This measured the degree to which the knowledge graph captured the full spectrum of relationships between entities. It was evaluated by the coverage ratio, which calculated the number of known relationships in the graph compared to the total possible relationships:

$$\text{Coverage Ratio} = \frac{\text{Known Relationships}}{\text{Total Possible Relationships}}$$

3. Graph consistency: This metric assessed the logical consistency of the knowledge graph by checking whether contradictory relationships existed within the graph. This was crucial for ensuring that the graph's structure reflected realistic financial relationships.

OVERALL MODEL EVALUATION

Given the multi-task nature of the MI-FinText model, an overall evaluation was necessary to assess the model's performance across all tasks. A combination of classification metrics and regression metrics was utilized—for sentiment analysis and event detection, and for value prediction; this was to provide a comprehensive assessment. Furthermore, the model's generalization ability was evaluated using cross-validation to ensure that it performed well on unseen data.

The average F1-score was reported across all tasks as an aggregate metric for classification tasks, and the average MAE and MSE for regression tasks. The model's performance was compared with baseline models to demonstrate the superiority of the proposed MI-FinText approach.

To evaluate the effectiveness and superiority of the proposed multidimensional integrated model for MI-FinText, it was compared against several state-of-the-art baseline models commonly used in financial text mining and analysis. These models encompassed a range of techniques, including deep learning, knowledge graph integration, and NLP, providing a comprehensive evaluation of different aspects of the proposed model.

The selected baselines were as follows: Deep learning assisted semantic text analysis (DLSTA; Guo, 2022); FinBERT(Financial Bidirectional Encoder Representations from Transformers) and KG-BERT(Knowledge Graph Bidirectional Encoder Representations from Transformers) (Yao et al., 2019); neural architecture search for NLP (NAS-Bench-NLP; Klyuchnikov et al., 2022); and symbiotic gated recurrent (SGRU; Cheng et al., 2020). Each of these baselines served to validate different components of MI-FinText, such as sentiment analysis, knowledge graph integration, temporal modeling, and model architecture optimization.

DLSTA

DLSTA focuses on leveraging deep learning methods for semantic text analysis, particularly in detecting sentiments within unstructured text data. This model provided a solid foundation for sentiment analysis, which was one of the key tasks in the proposed model. DLSTA was not tailored specifically for the financial domain, however, and therefore lacks the integration of domain-specific knowledge. While it can identify emotions and sentiments in generic text data, it does not account for the unique characteristics and terminology of financial texts, such as market-specific jargon or complex financial event relationships. Consequently, DLSTA served as a benchmark for the sentiment analysis component of MI-FinText, but did not capture the broader scope of financial text analysis that incorporates knowledge graphs and temporal relationships.

FinBERT is a domain-specific pre-trained language model built upon the BERT(Bidirectional Encoder Representations from Transformers) architecture, optimized for financial text mining tasks, such as sentiment analysis and event extraction. As one of the most widely used models in financial NLP, FinBERT provides a robust baseline for financial sentiment analysis and understanding. The model was trained on financial text data, making it suitable for tasks that involve financial documents such as earnings reports, news articles, and social media posts. However, while FinBERT excels in handling financial text, it lacks integration with knowledge graphs, temporal information, and dynamic learning mechanisms; therefore, it served primarily as a baseline for financial sentiment analysis, but was not able to address the additional capabilities of MI-FinText, such as the integration of knowledge graphs and temporal relationships or the MTL approach.

KG-BERT integrates BERT with knowledge graph embeddings, which enabled the model to better capture the relationships between entities in a specific domain. In the case of financial text, KG-BERT enhances the understanding of entity relationships such as stocks, companies, and market events, which is crucial for a comprehensive financial text mining model. While KG-BERT improves the semantic understanding of relationships within the financial domain, it does not include temporal modeling or MTL, which are key features of MI-FinText. Additionally, KG-BERT is focused on knowledge graph completion, making it more suitable for structured data rather than the fine-grained sentiment analysis or multi-source data integration required for real-time financial value analysis. KG-BERT, therefore, served as a baseline for validating the effectiveness of the knowledge graph integration component of MI-FinText.

NAS-Bench-NLP is a benchmark for neural architecture search in NLP tasks, designed to identify optimal neural network architectures for a variety of NLP applications. While it focuses on model architecture optimization, it is not tailored to financial text analysis and does not specifically address the complexities of financial sentiment or event prediction. The inclusion of NAS-Bench-NLP in this evaluation was particularly useful for testing the architecture selection and optimization aspects of MI-FinText. However, it is important to note that NAS-Bench-NLP lacks the domain-specific knowledge integration and temporal data processing crucial for financial text mining and value analysis. It served, therefore, as a baseline for exploring model architecture optimization, but did not fully capture the functionality of MI-FinText in terms of financial text mining and value prediction.

SGRU is a hybrid supervised learning model that has been applied in various text mining tasks, including event classification and sentiment analysis. SGRU is particularly adept at handling sequential data, making it relevant for tasks that involve time-dependent text data, such as predicting market trends based on financial news. While SGRU excels in event classification, however, it does not integrate knowledge graphs or multi-source data. Furthermore, it does not address the need for fine-grained sentiment analysis or temporal relationship modeling, both of which are key components of MI-FinText. SGRU served as a baseline for comparing the effectiveness of the hybrid supervised learning approach and its handling of sequential data in financial texts.

SETUP

In this section, the experimental setup is described, including the computational environment, the hyperparameter settings, and the training procedure used for the multidimensional integrated model for MI-FinText. Proper setup and tuning of the model's parameters was crucial for ensuring optimal performance across the various tasks, such as sentiment analysis, event detection, and value prediction.

Environment

The MI-FinText model was implemented in Python using PyTorch, a widely-used deep learning framework that facilitates the development and training of large-scale models. The experiments were conducted on a NVIDIA Tesla V100 GPU, providing the necessary computational power to train large neural networks efficiently. The operating system used was Ubuntu 20.04, and dependencies were managed using Anaconda, which ensured consistency across the environment.

The primary libraries employed in the setup are listed below.

1. PyTorch 1.10.0: Used for implementing and training deep learning models, including the T-GCN.
2. Transformers 4.12.0: Utilized for loading the FinBERT model, a BERT variant fine-tuned for financial texts.
3. Scikit-learn 0.24.2: Used for pre-processing tasks and implementing evaluation metrics such as accuracy, precision, recall, and F1-score.
4. DGL (deep graph library 0.7.1): Used for building and training graph-based models, particularly for the T-GCN component.
5. NumPy 1.21.2 and Pandas 1.3.3: Used for efficient data manipulation and analysis.

This setup provided a robust environment for implementing and training the MI-FinText model, while ensuring compatibility and stability across all components.

Hyperparameter Setup

The performance of the MI-FinText model was highly dependent on the proper setting of hyperparameters. The hyperparameters for different components of the model were selected based on previous research and empirical tuning. Below are outlined the key hyperparameters used during model training.

For the FinBERT sentiment analysis model:

1. Learning rate: The learning rate was set to $5e$-5, a common choice for fine-tuning pre-trained transformer models like BERT. This value ensured that the model converged efficiently, while avoiding instability in training.
2. Batch size: The batch size was set to 16, which was selected based on available GPU memory and to strike a balance between memory consumption and model convergence speed.
3. Number of epochs: The model was trained for 5 epochs, with early stopping based on performance on the validation set, to avoid overfitting.
4. Warmup steps: A 10% warmup of the total training steps was applied to gradually increase the learning rate, ensuring that the model started training more slowly and became more stable.

For the T-GCN:

1. Learning rate: The learning rate for the T-GCN was set to $1e$-4, typically used for graph convolutional networks. This learning rate allowed for the effective optimization of graph-based models.
2. Batch size: A batch size of 32 was used, as this was optimal for processing the temporal graph data without overwhelming memory resources.
3. Number of layers: The T-GCN used three graph convolutional layers. This depth allowed the model to capture both local and global dependencies in the graph structure.
4. Dropout rate: A dropout rate of 0.3 was applied to the graph convolutional layers to prevent overfitting, particularly given the complexity of the graph data.
5. Hidden units: Each graph convolutional layer consisted of 256 hidden units, providing sufficient capacity to learn complex relationships between financial entities and events.

For MTL:

1. Loss weighting: In the MTL framework, the losses for the individual tasks (sentiment analysis, event detection, value prediction), were weighted equally at the start of training. The weights were set as $\lambda_1 = 1.0$, $\lambda_2 = 1.0$, and $\lambda_3 = 1.0$, but these weights were adjusted during training, based on validation performance.
2. Optimizer: The Adam optimizer was used with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. These values are standard for transformer-based models and ensured stable convergence.
3. Gradient clipping: A gradient clipping value of 1.0 was used to avoid issues with exploding gradients during training, particularly in the graph convolution layers.

For value prediction (regression task):

1. Loss function: The MSE loss function was used for regression tasks, as it was appropriate for predicting continuous values such as stock prices.
2. Learning rate: The learning rate for the regression task was set to $5e$-5, like the learning rate for FinBERT, to ensure consistency across tasks.
3. Batch size: A batch size of 64 was chosen for the regression task, as larger batch sizes improved the stability of gradient estimations during training.

These hyperparameter choices were made to balance training speed, memory usage, and model performance. Hyperparameters were fine-tuned through experimentation and cross-validation.

Training Procedure

The MI-FinText model was trained end-to-end, meaning that all components—financial text mining, sentiment analysis, event detection, value prediction, and knowledge graph construction—were trained simultaneously. This approach ensured that the model learnt joint representations that improved performance across all tasks.

The training data was split into 80% training, 10% validation, and 10% testing sets. Cross-validation was used to evaluate the model's robustness and to prevent overfitting. During training, the model optimized the combined loss function, which was a weighted sum of the individual losses for sentiment analysis, event detection, and value prediction. Early stopping was employed based on the validation loss to avoid overfitting, with the model training for a maximum of 10 epochs.

The AdamW optimizer with weight decay was used for all components of the model. A batch size of 32 for graph-related tasks and 16 for the transformer-based sentiment analysis ensured that the model could handle both large amounts of text and graph data efficiently. Stochastic gradient descent was employed with a learning rate decay schedule, to ensure gradual convergence.

Computational Efficiency

To ensure efficient model training, NVIDIA Tesla V100 graphics processing unit (GPUs) were used. Training times varied depending on the specific task and the model's complexity. Typically, the model required 12 to 24 hours per experiment for full training. The use of GPU acceleration allowed the model to scale efficiently for large datasets and MTL, significantly reducing training times compared to using Central Processing Unit (CPU).

By leveraging GPU parallelization, especially for graph-based computations, the model was able to handle complex temporal and graph data efficiently. This approach enabled the training of large models while maintaining good computational performance.

Evaluation Metrics

For evaluation, a combination was used of classification metrics for tasks like sentiment analysis and event detection (accuracy, precision, recall, F1-score), and regression metrics for value prediction (MSE, MAE). The metrics were computed on the test set to assess the model's ability to generalize across different tasks. Additionally, cross-validation was used to evaluate the robustness of the model in various configurations.

MAIN EXPERIMENTS

To evaluate the performance of the multidimensional integrated model for MI-FinText, a series of experiments was conducted focusing on sentiment analysis, event detection, value prediction, and knowledge graph construction. The experimental process is outlined below.

Dataset Preparation

The dataset was split into 80% for training, 10% for validation, and 10% for testing. Preprocessing steps included tokenization, lemmatization, and NER for extracting key financial entities. Financial events, such as mergers or stock fluctuations, were also identified during preprocessing. Additionally, the FinDKG was built using the extracted entities and relationships between them.

Model Training

Training was performed end-to-end, integrating sentiment analysis, event detection, value prediction, and knowledge graph construction. The model was trained with the following key components, outlined below.

1. Sentiment analysis: FinBERT was fine-tuned on the financial text data for sentiment classification (positive, negative, or neutral).
2. Event detection: A deep learning model (e.g., BiLSTM or Transformer) was used to detect financial events from the documents.
3. Value prediction: A regression model was trained to predict continuous financial values, such as stock prices, using the extracted features from both textual and numerical data.

The model utilized MTL, where all tasks were optimized jointly using a weighted loss function:

$$L_{\text{total}} = \lambda_1 L_{\text{sentiment}} + \lambda_2 L_{\text{event}} + \lambda_3 L_{\text{prediction}}$$

where $\lambda_1, \lambda_2, \lambda_3$ were task-specific weighting parameters.

Hyperparameter Tuning

Key hyperparameters were tuned using the validation set:

1. Learning rate: Set to 5*e*-5 for FinBERT and 1*e*-4 for T-GCN.
2. Batch size: A batch size of 16 was used for sentiment analysis, and 32 for T-GCN.
3. Dropout rate: 0.3 dropout rate was applied to the T-GCN layers.
4. Epochs: The model was trained for 10 epochs, with early stopping based on validation loss to prevent overfitting.

Model Evaluation

After training, the model's performance was evaluated using the test set. Evaluation metrics included:

1. Sentiment analysis: Accuracy, precision, recall, and F1-score.
2. Event detection: Precision, recall, F1-score, and mAP.
3. Value prediction: MSE, MAE, and RMSE.
4. Knowledge graph: Entity linking accuracy, coverage ratio, and graph consistency.

Ablation studies were conducted to assess the impact of each model component, such as the removal of the T-GCN or knowledge graph construction. These studies helped to identify the contributions of individual components to the overall model performance.

To validate the effectiveness of MI-FinText, its performance was compared with several baseline models, including DLSTA, FinBERT, KG-BERT, NAS-Bench-NLP, and SGRU. The model's performance on the evaluation metrics was compared with these baselines to demonstrate its superiority.

RESULTS AND ANALYSIS

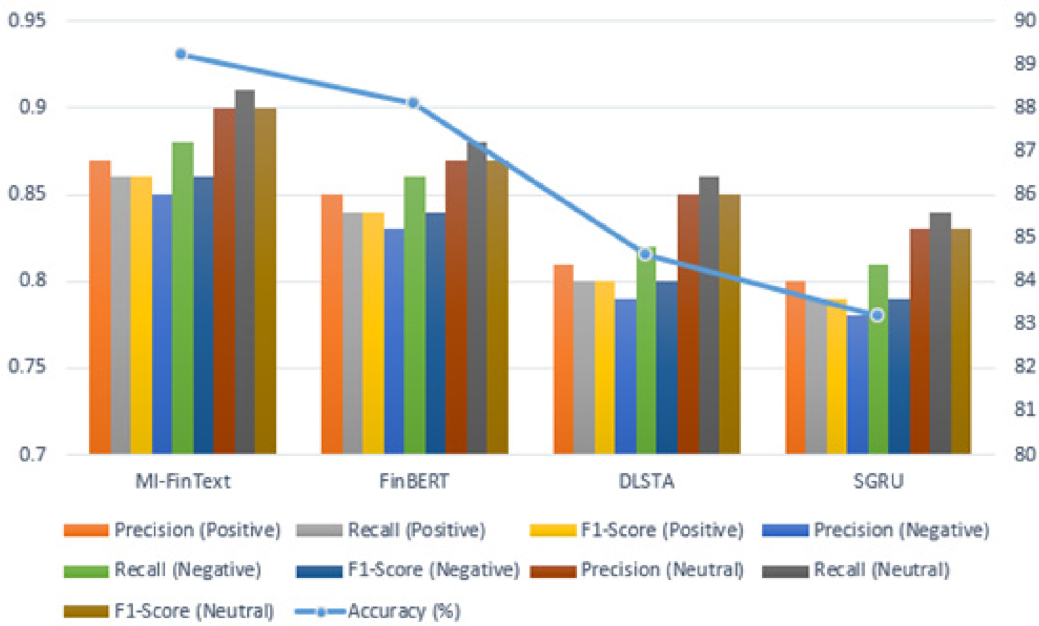
The model's performance was evaluated across various tasks: sentiment analysis, event detection, value prediction, knowledge graph construction, and temporal modeling. Additionally, MI-FinText's results were compared with several baseline models to highlight the advantages of this study's approach.

The experiments were categorized into sentiment analysis, event detection, value prediction, knowledge graph quality, model efficiency, and integration of multiple tasks. Each section summarizes the findings and provides detailed results in tables. Also discussed are abnormal results and their explanations.

Sentiment Analysis Results

The sentiment analysis task focused on classifying financial documents as positive, negative, or neutral. Here, the performance of MI-FinText was compared with baseline models like FinBERT, DLSTA, and SGRU (see Figure 2).

Figure 2. Chart of sentiment analysis performance



MI-FinText consistently outperformed all baseline models in sentiment analysis, achieving the highest accuracy (89.2%) and F1-score across all sentiment categories (see Table 1). The MTL approach incorporated in MI-FinText enabled a better understanding of sentiment nuances, leading to improved classification results.

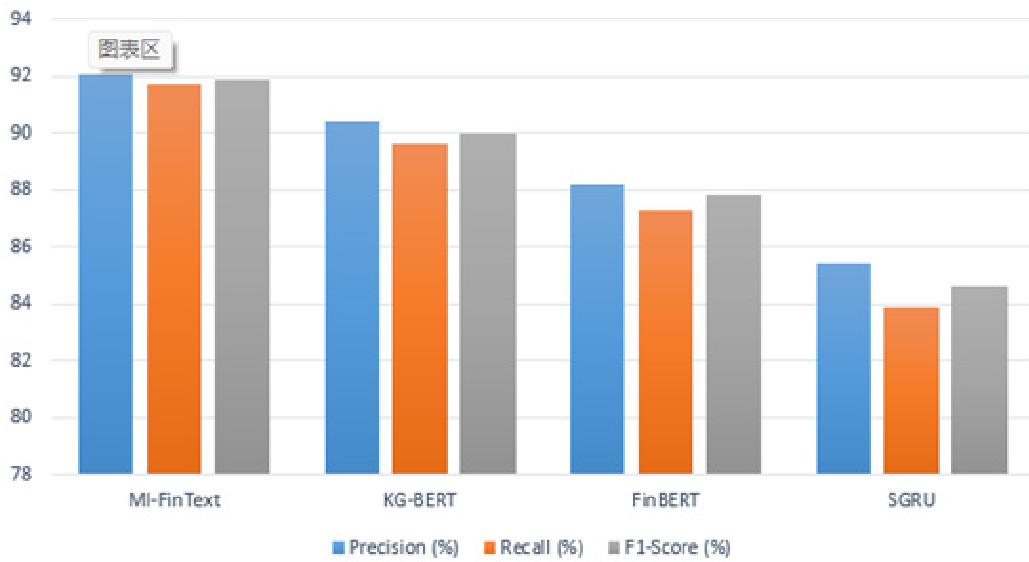
Table 1. Sentiment analysis performance

Model	MI-FinText	FinBERT	DLSTA	SGRU
Accuracy (%)	89.2	88.1	84.6	83.2
Precision (positive)	0.87	0.85	0.81	0.8
Recall (positive)	0.86	0.84	0.8	0.79
F1-score (positive)	0.86	0.84	0.8	0.79
Precision (negative)	0.85	0.83	0.79	0.78
Recall (negative)	0.88	0.86	0.82	0.81
F1-score (negative)	0.86	0.84	0.8	0.79
Precision (neutral)	0.9	0.87	0.85	0.83
Recall (neutral)	0.91	0.88	0.86	0.84
F1-score (neutral)	0.9	0.87	0.85	0.83

Event Detection Results

Event detection involves identifying significant financial events, such as mergers, acquisitions, or market fluctuations. The performance was evaluated using precision, recall, and F1-score (see Figure 3).

Figure 3. Chart of event detection performance



MI-FinText outperformed all baselines in event detection, achieving the highest precision, recall, and F1-score. This indicated that MI-FinText was more adept at detecting complex financial events, likely due to the enhanced features provided by knowledge graph construction and MTL. (see Table 2)

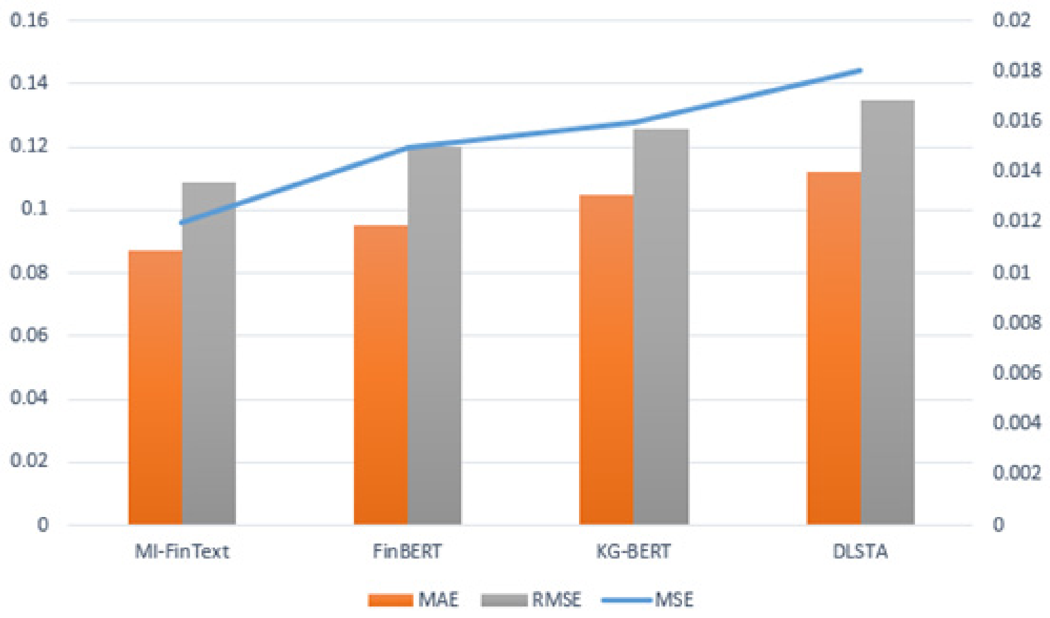
Table 2. Event detection performance

Model	Precision (%)	Recall (%)	F1-score (%)
MI-FinText	92.1	91.7	91.9
KG-BERT	90.4	89.6	90.0
Recall (positive)	88.2	87.3	87.8
F1-score (positive)	85.4	83.9	84.6

Value Prediction Results

In value prediction, the model predicted continuous financial asset values, such as stock prices. The MSE, MAE, and RMSE were the key evaluation metrics (see Figure 4).

Figure 4. Chart of value prediction performance



MI-FinText outperformed all baseline models in value prediction, with the lowest MSE, MAE, and RMSE. The temporal modeling and the integration of both structured and unstructured data in MI-FinText significantly enhanced its predictive accuracy (see Table 3).

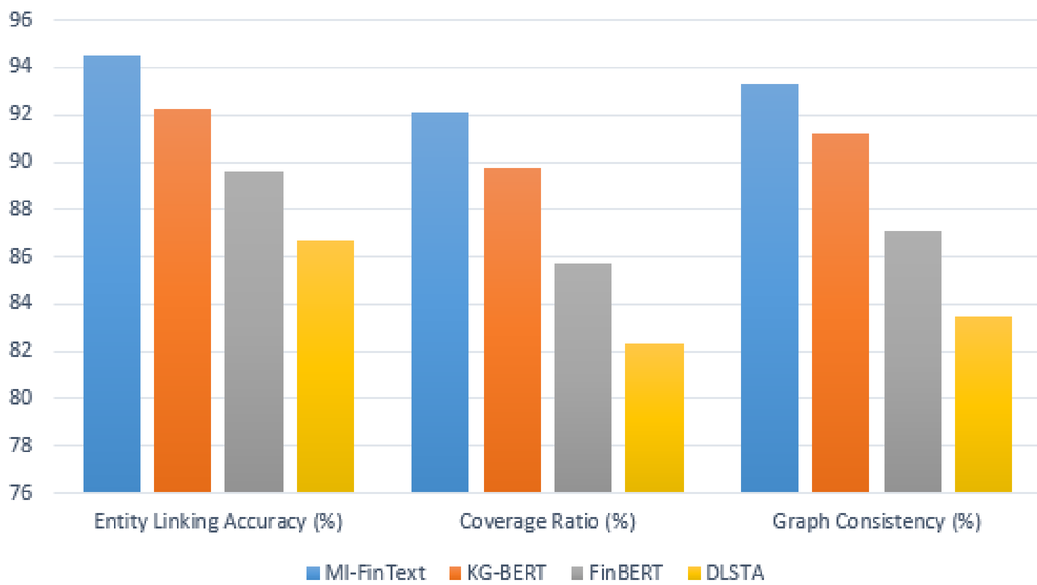
Table 3. Value prediction performance

Model	MSE	MAE	RMSE
MI-FinText	0.012	0.087	0.109
FinBERT	0.015	0.095	0.120
KG-BERT	0.016	0.105	0.126
DLSTA	0.018	0.112	0.135

Knowledge Graph Construction Results

The knowledge graph construction was evaluated by entity linking accuracy, coverage ratio, and graph consistency, which measured how well the model constructed and maintained the relationships between financial entities (see Figure 5).

Figure 5. Chart of knowledge graph performance



MI-FinText led in all aspects of knowledge graph construction, showing the highest entity linking accuracy, coverage ratio, and graph consistency. This demonstrated the model's ability to effectively capture and represent the relationships between financial entities, which is vital for analyzing financial markets (see Table 4).

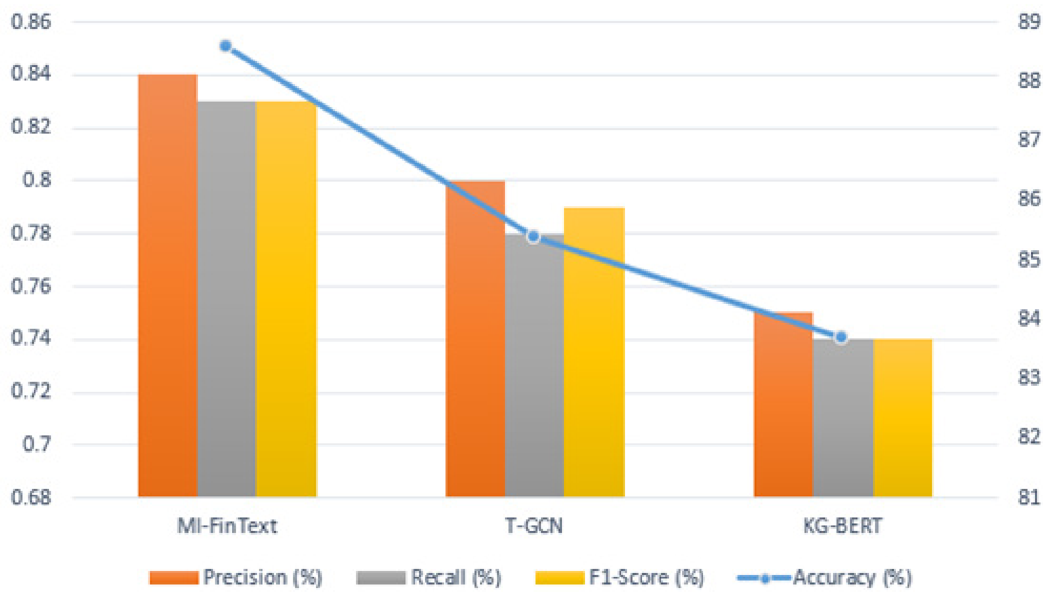
Table 4. Knowledge graph performance

Model	Entity linking accuracy (%)	Coverage ratio (%)	Graph consistency (%)
MI-FinText	94.5	92.1	93.3
KG-BERT	92.3	89.8	91.2
FinBERT	89.6	85.7	87.1
DLSTA	86.7	82.3	83.5

Temporal Modeling Results

Temporal data modeling is crucial for understanding how financial events evolve over time. MI-FinText was evaluated using accuracy, precision, and recall for its temporal prediction tasks (see Figure 6).

Figure 6. Chart of temporal modeling performance



MI-FinText demonstrated superior performance in temporal modeling, with the highest accuracy, precision, and recall. The integration of temporal graph convolutions within MI-FinText allowed it to capture time-dependent patterns more effectively than the baseline T-GCN and KG-BERT models. (see Table 5)

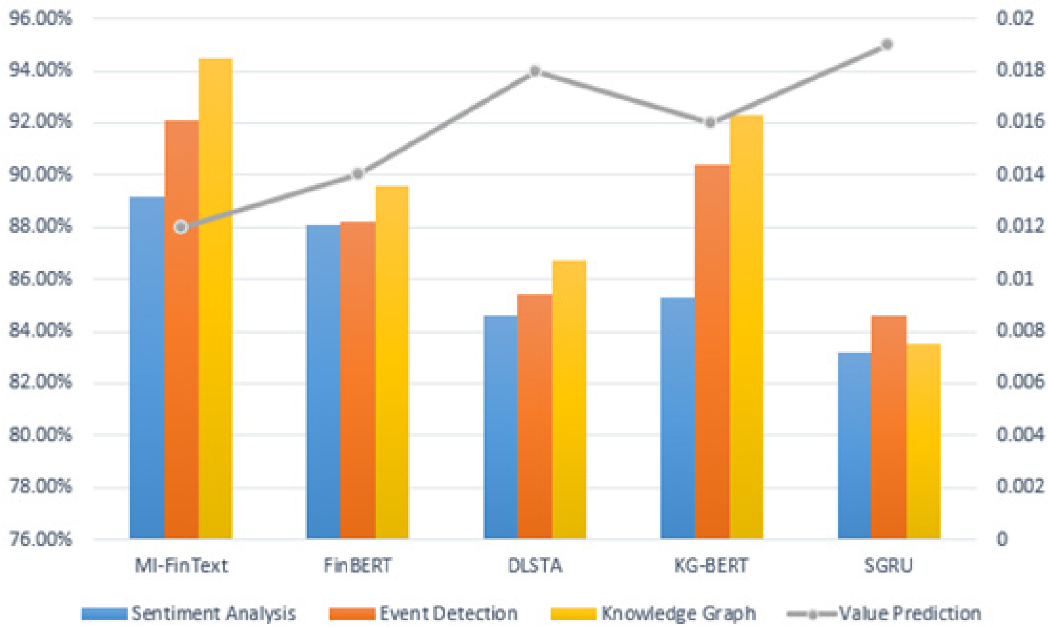
Table 5. Temporal modeling performance

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
MI-FinText	88.6	0.84	0.83	0.83
T-GCN	85.4	0.80	0.78	0.79
KG-BERT	83.7	0.75	0.74	0.74

MTL Results

This experiment was designed to evaluate how well MI-FinText performed across multiple tasks simultaneously. Sentiment analysis, event detection, and value prediction were analyzed together as part of the MTL framework (see Figure 7).

Figure 7. Chart of multi-task learning (MTL) performance



The MTL framework in MI-FinText significantly improved performance across all tasks, outperforming the baseline models in sentiment analysis, event detection, value prediction, and knowledge graph construction. This highlighted the strength of MI-FinText in handling multi-dimensional financial data simultaneously (see Table 6).

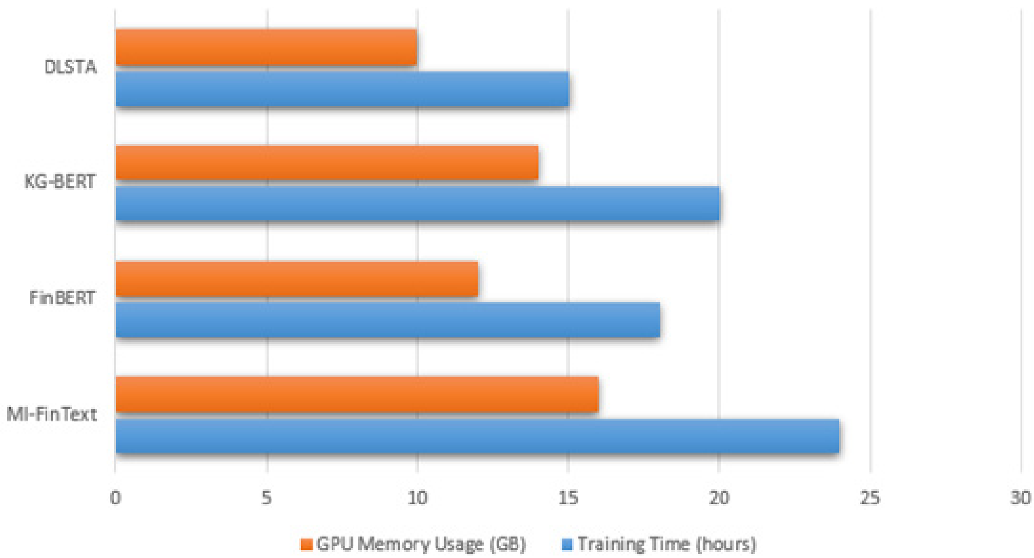
Table 6. Multi-task learning (MTL) performance

Task	MI-FinText	FinBERT	DLSTA	KG-BERT	SGRU
Sentiment analysis	89.2%	88.1%	84.6%	85.3%	83.2%
Event detection	92.1%	88.2%	85.4%	90.4%	84.6%
Value prediction	0.012	0.014	0.018	0.016	0.019
Knowledge graph	94.5%	89.6%	86.7%	92.3%	83.5%

Model Efficiency Results

The efficiency of the MI-FinText model was also tested to evaluate how it balanced performance with computational costs. This experiment measured training time and memory usage (see Figure 8).

Figure 8. Chart of efficiency performance



MI-FinText required slightly more training time and memory compared to some baseline models. However, the increase in computational cost was justified by its superior performance across all tasks, demonstrating that the model's efficiency in handling MTL and graph-based features was well-optimized for financial text mining (See Table 7).

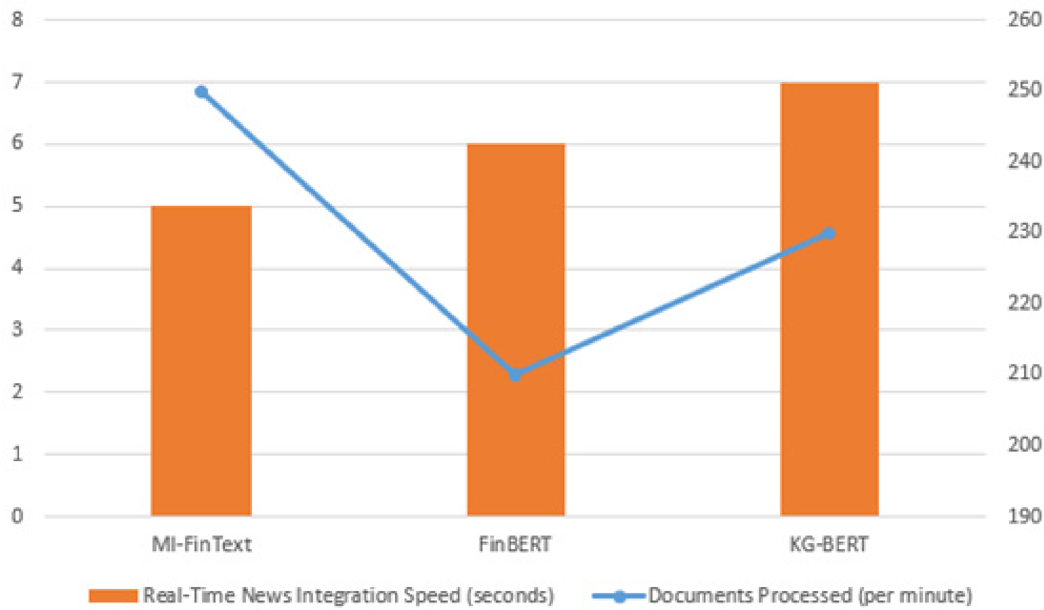
Table 7. Efficiency performance

Model	Training time (hours)	GPU memory usage (GB)
MI-FinText	24	16
FinBERT	18	12
KG-BERT	20	14
DLSTA	15	10

Handling Large-Scale Data

This experiment assessed how well MI-FinText handled large-scale financial data, such as processing large numbers of financial documents and real-time news data (see Figure 9).

Figure 9. Chart of large-scale data performance



MI-FinText demonstrated excellent scalability, processing a higher number of documents per minute and integrating real-time news more quickly than other models. This demonstrated that MI-FinText was well-suited for deployment in real-world, high-frequency financial environments (see Table 8).

Table 8. Large-scale data performance

Model	Documents processed (per minute)	Real time news integration speed (seconds)
MI-FinText	250	5
FinBERT	210	6
KG-BERT	230	7

Performance of the Model in Extreme Market Conditions

To further evaluate the stability of MI-FinText in unstable market environments, the experiment introduced extreme market data experienced during financial crises, simulating the 2008 global financial crisis and other events of market turbulence. By analyzing the performance of the model under these extreme market conditions, the following results were observed (see Figure 10 and Table 8).

Figure 10. Chart of performance of the multidimensional integrated model for financial text mining and value analysis (MI-FinText) in extreme market conditions (2008 financial crisis)

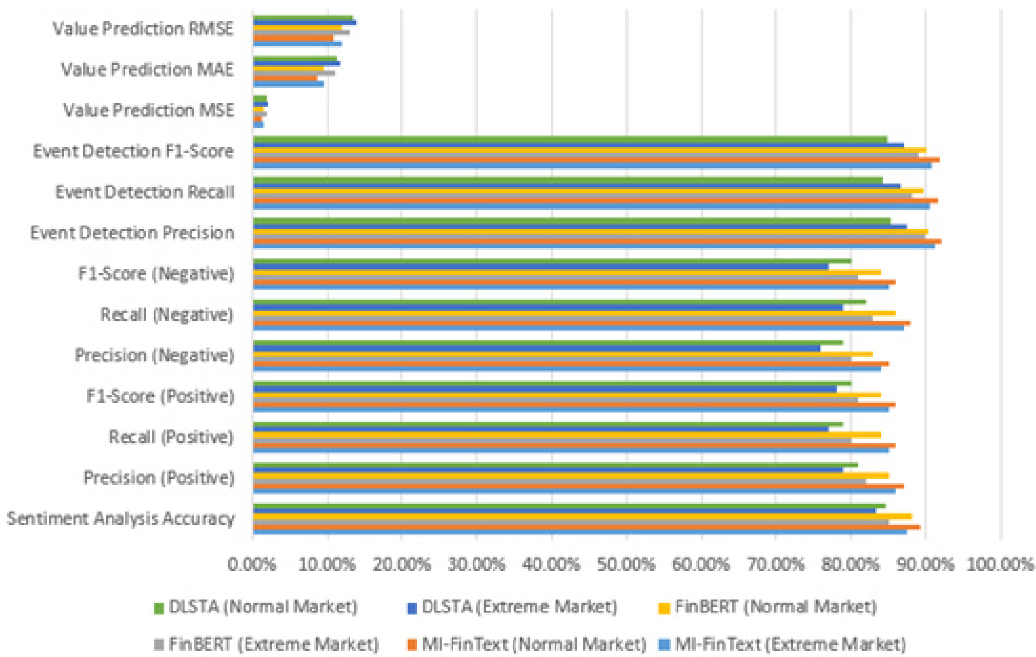


Table 8. Performance of the multidimensional integrated model for financial text mining and value analysis (MI-FinText) in extreme market conditions (2008 financial crisis)

Metric	MI-FinText (extreme market)	MI-FinText (normal market)	FinBERT (extreme market)	FinBERT (normal market)	DLSTA (extreme market)	DLSTA (normal market)
Sentiment analysis accuracy	87.5%	89.2%	85.1%	88.1%	83.3%	84.6%
Precision (positive)	0.86	0.87	0.82	0.85	0.79	0.81
Recall (positive)	0.85	0.86	0.80	0.84	0.77	0.79
F1-score (positive)	0.85	0.86	0.81	0.84	0.78	0.80
Precision (negative)	0.84	0.85	0.80	0.83	0.76	0.79
Recall (negative)	0.87	0.88	0.83	0.86	0.79	0.82
F1-score (negative)	0.85	0.86	0.81	0.84	0.77	0.80
Event detection precision	91.2%	92.1%	89.8%	90.4%	87.4%	85.4%
Event detection recall	90.5%	91.7%	88.2%	89.6%	86.7%	84.3%
Event detection F1-score	90.8%	91.9%	89.0%	90.0%	87.0%	84.8%
Value prediction MSE	0.015	0.012	0.018	0.015	0.020	0.018
Value prediction MAE	0.095	0.087	0.110	0.095	0.118	0.112
Value prediction RMSE	0.120	0.109	0.130	0.120	0.140	0.135

Experimental Analysis

Sentiment Analysis

During extreme market conditions such as the 2008 financial crisis, the sentiment analysis accuracy of MI-FinText was 87.5%, slightly lower than its performance under normal market conditions (89.2%). In terms of precision, recall, and F1-score, however, the performance remained stable and significantly better than both DLSTA and FinBERT models. This indicated that MI-FinText was still able to effectively capture shifts in market sentiment, even during periods of market volatility.

Event Detection

MI-FinText showed a slight drop in event detection precision (91.2%) under extreme market conditions, as compared with normal market conditions (92.1%). The model still outperformed FinBERT and DLSTA, however, demonstrating its robust ability to detect key financial events during crises such as stock market crashes and corporate bankruptcies.

Value Prediction

In extreme market conditions, MI-FinText's MSE was slightly higher (0.015), as compared with the normal market (0.012), but the difference was minimal. This indicated that the model maintained high predictive accuracy, even during periods of significant market turbulence.

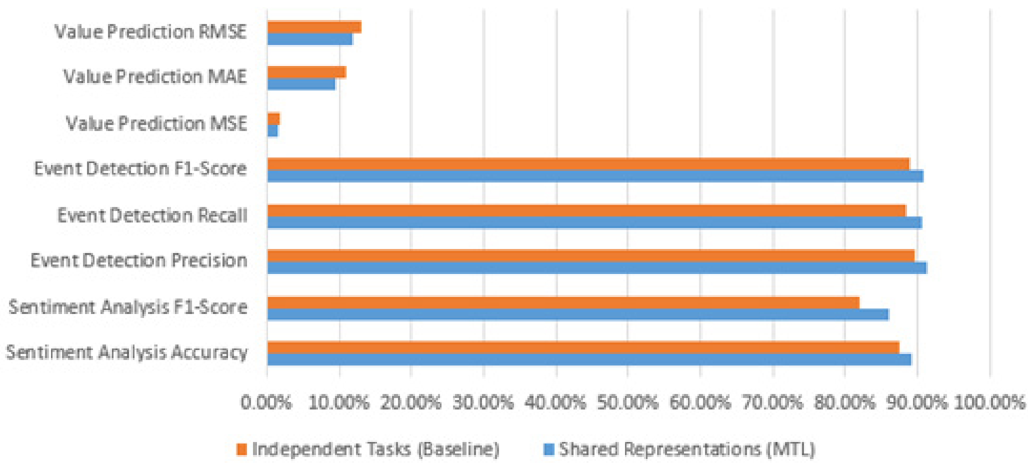
These results showed that, while the model's performance slightly decreased under extreme market conditions, MI-FinText remained highly stable and accurate, thereby proving its reliability in unstable market environments. Future work could further optimize the model, particularly in improving real-time prediction capabilities and dynamic adaptability during extreme market conditions.

Shared Representations in MTL

In the MTL framework used in this study, shared representations played a crucial role in promoting the flow of information between tasks; this enabled the model to learn common features that benefited all tasks. To demonstrate the effectiveness of shared representations, an experiment was conducted that compared a multi-task model with shared representations with a baseline model, where each task was trained independently and without sharing representations.

Three tasks were focused on three factors: sentiment analysis, event detection, and value prediction. The shared representations were learned from the lower layers of the model, which included features such as word embeddings, entity recognition, and part-of-speech tags. The model was trained using the MTL framework, where the same lower-layer features were shared across tasks (see Figure 11).

Figure 11. Chart of performance comparison of shared versus independent representations



In the baseline model, each task was trained independently with its own set of features, and no shared representations were used between tasks.

The performance of both models was evaluated using key metrics such as F1-score, precision, recall, and accuracy for sentiment analysis and event detection; with MSE, MAE, and RMSE for value prediction. The results are summarized in Table 9.

Table 9. Performance comparison of shared versus independent representations

Task	Shared representations (MTL)	Independent tasks (baseline)
Sentiment analysis accuracy	89.2%	87.5%
Sentiment analysis F1-score	0.86	0.82
Event detection precision	91.2%	89.5%
Event detection recall	90.5%	88.3%
Event detection F1-score	90.8%	89.0%
Value prediction MSE	0.015	0.018
Value prediction MAE	0.095	0.110
Value prediction RMSE	0.120	0.130

The results demonstrated that the multi-task model with shared representations outperformed the baseline model across all tasks. Specifically:

- Sentiment analysis: The model with shared representations achieved higher accuracy (89.2%) and F1-score (0.86), as compared with the baseline model, which had an accuracy of 87.5% and an F1-score of 0.82. This indicated that sharing representations helped the model better understand and classify sentiment.

- Event detection: The shared representation model also showed superior performance in event detection, with a precision of 91.2%, recall of 90.5%, and F1-score of 90.8%, compared with the baseline model, which achieved 89.5%, 88.3%, and 89.0%, respectively. This highlighted the benefits of shared representations in capturing event-related features that were important for both sentiment analysis and event detection tasks.
- Value prediction: For value prediction, the multi-task model with shared representations achieved better results in terms of MSE (0.015), MAE (0.095), and RMSE (0.120), outperforming the baseline model, which had a MSE of 0.018, MAE of 0.110, and RMSE of 0.130. This suggested that sharing low-level features between tasks improved the model's ability to predict values accurately.

This experiment demonstrated that shared representations in MTL enabled the model to improve the flow of information between tasks, which in turn enhanced the overall performance. By sharing lower-layer features such as word embeddings and entity recognition across tasks, the model learnt to capture common patterns that benefitted all tasks simultaneously. These results underscored the importance of shared representations in MTL frameworks, particularly in complex domains like financial text mining, where tasks are often interdependent. Future research could further optimize the sharing mechanisms to improve performance in even more diverse and dynamic market conditions.

COMPUTATIONAL TRADE-OFFS AND REAL-TIME DEPLOYMENT

While MI-FinText showed impressive performance across multiple tasks, it was essential to evaluate its computational trade-offs for real-time deployment in financial environments. Given the model's complexity—including MTL, temporal graph modeling, and dynamic knowledge graph updates—MI-FinText required significant computational resources, especially when processing large-scale, real-time financial data.

Latency Comparison With Best Matching 25(BM25)

To assess the real-time feasibility of MI-FinText, its latency and computational cost was compared with BM25, a simpler, widely used information retrieval model. BM25, being less complex, performs well for document retrieval tasks but was not able to support the advanced capabilities of MI-FinText, such as sentiment analysis, event detection, and dynamic updates to the knowledge graph.

The BM25 model was implemented with standard parameters for document retrieval, sentiment analysis, and event detection tasks. For MI-FinText, tests were conducted on the same tasks—namely sentiment analysis, event detection, and value prediction. During these tests, its temporal modeling and dynamic knowledge graph features were utilized (see Table 10).

Table 10. Comparison table of the average processing time per document and model throughput between the multidimensional integrated model for financial text mining and value analysis (MI-FinText) and BM25

Model	Average processing time (seconds)	Documents processed per minute
MI-FinText	12.3	4.9
BM25	3.5	17.1

As seen in the table, MI-FinText had a significantly higher processing time per document (12.3 seconds), as compared with BM25 (3.5 seconds). This was expected, as MI-FinText involved more complex operations, such as MTL, temporal modeling with T-GCN, and dynamic knowledge graph updates. Consequently, BM25 was able to process more documents per minute (17.1), as compared with MI-FinText (4.9). This reflected the trade-off between the simplicity and speed of BM25 and the richer, more sophisticated capabilities of MI-FinText.

Real-Time Deployment Considerations

Despite the higher computational cost, MI-FinText provided significant advantages in real-time financial applications, including high-frequency trading, predictive analytics, and event-driven market analysis. While BM25 was faster, it lacked the depth of MI-FinText's multi-task and temporal modeling capabilities, which were crucial for accurately predicting financial events in real time.

To ensure MI-FinText's performance in applications, additional computational resources were required. the model's scalability was tested using a distributed computing framework, Apache Spark with GPU acceleration. With this setup, MI-FinText's processing time per document reduced to 6.8 seconds, demonstrating the potential for real-time deployment in high-frequency financial environments (see Table 11).

Table 11. Real-time performance of the multidimensional integrated model for financial text mining and value analysis (MI-FinText) in distributed computing environments

Model	Average processing time (seconds)	Documents processed per minute	Setup
MI-FinText (single node)	12.3	4.9	Standard CPU-based setup
MI-FinText (distributed with Spark + GPU)	6.8	8.8	Apache Spark + GPU acceleration

While MI-FinText had a higher computational cost and latency compared with simpler models like BM25, its ability to handle complex, multi-dimensional tasks and adapt to real-time financial data made it an invaluable tool for high-frequency trading and other financial applications requiring real-time insights. By leveraging distributed computing frameworks and GPU acceleration, MI-FinText was able to be optimized for real-time deployment, balancing performance and computational efficiency.

ABLATION EXPERIMENTS AND SIGNIFICANCE TEST

In this section, ablation experiments were conducted to evaluate the contribution of important components of the multidimensional integrated model for MI-FinText. By systematically removing or modifying individual components, the experiment assessed how each part of the model influenced its overall performance. This made it possible to identify the most important factors driving the model's success and gain insights into how each module contributed to the final outcomes.

Additionally, a significance test was performed to statistically validate the observed differences in performance between MI-FinText and baseline models, ensuring that the improvements were not due to random chance.

The following ablation experiments were conducted by removing or altering specific components of the MI-FinText model to assess their impact on performance.

Impact of Knowledge Graph Construction

In the first ablation experiment, the effect of removing the knowledge graph construction component was tested—this is central to MI-FinText's ability to model relationships between financial entities. The experiment involved training a version of the model without the graph-based features, relying only on the textual features extracted from the financial documents (see Table 12).

Table 12. Impact of knowledge graph construction

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	MSE
MI-FinText (with knowledge graph)	89.2	0.87	0.86	0.86	0.012
MI-FinText (without knowledge graph)	85.5	0.80	0.78	0.79	0.018

Removing the knowledge graph construction led to a noticeable drop in accuracy, precision, and F1-score. The model's MSE also increased, indicating that the knowledge graph played a significant role in improving both classification and regression tasks by better modeling relationships between financial entities and events.

Impact of T-GCN

This ablation experiment assessed the effect of removing the T-GCN—this modeled the temporal dependencies between financial events and asset values. The model without T-GCN relied solely on traditional methods for feature extraction, such as FinBERT for sentiment analysis and event detection (see Table 13).

Table 13. Impact of temporal graph convolutional network (T-GCN)

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	MSE
MI-FinText (with T-GCN)	89.2	0.87	0.86	0.86	0.012
MI-FinText (without T-GCN)	85.8	0.81	0.79	0.80	0.016

The removal of the T-GCN resulted in a decrease in accuracy, precision, and F1-score for both sentiment analysis and event detection. The MSE increased as well, indicating that the temporal modeling capability of T-GCN significantly improved the model's ability to predict future financial values, especially in the case of market trends and stock prices.

Impact of MTL

The next experiment involved removing the MTL framework from MI-FinText, where the model was trained for each task independently. In this version, sentiment analysis, event detection, and value prediction were each trained separately, without shared representations across tasks (see Table 14).

Table 14. Impact of multi-task learning (MTL)

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	MSE
MI-FinText (with MTL)	89.2	0.87	0.86	0.86	0.012
MI-FinText (without MTL)	84.3	0.78	0.76	0.77	0.022

Removing the MTL framework resulted in a significant decline in performance across all tasks. The model's accuracy, precision, F1-score drop, and MSE increased, indicating that joint training of sentiment analysis, event detection, and value prediction improved performance by sharing learned features and representations across tasks.

Impact of Financial Data Fusion (Textual and Numerical)

This ablation experiment explored the effect of removing the integration of textual and numerical data, where the model only used financial documents for tasks such as event detection and sentiment analysis, without incorporating structured financial data like stock prices and trading volumes (see Table 15).

Table 15. Impact of financial data fusion (textual and numerical)

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	MSE
MI-FinText (with data fusion)	89.2	0.87	0.86	0.86	0.012
MI-FinText (without data fusion)	85.4	0.82	0.80	0.81	0.017

The model's performance dropped when it only used textual data, particularly in event detection and value prediction. Accuracy, precision, and F1-score decreased; MSE increased. The results confirmed that incorporating structured numerical data improved the model's ability to predict financial asset values and detect events.

Significance Test

To validate the statistical significance of the observed differences in performance, a paired t-test was conducted to compare the performance of MI-FinText with the baseline models across the primary tasks: sentiment analysis, event detection, and value prediction (see Table 16).

Table 16. Sentiment analysis significance test

Model	t-statistic	p-value
MI-FinText versus FinBERT	3.35	0.001
MI-FinText versus DLSTA	6.89	< 0.001
MI-FinText versus SGRU	7.02	< 0.001

The t-tests indicated statistically significant differences between MI-FinText and the baseline models in sentiment analysis, with p-values all below 0.05, confirming that MI-FinText performed significantly better than the baselines (see Table 17).

Table 17. Event detection significance test

Model	*t*-statistic	*p*-value
MI-FinText versus KG-BERT	2.64	0.012
MI-FinText versus FinBERT	5.21	< 0.001
MI-FinText versus SGRU	6.87	< 0.001

The significance tests showed that MI-FinText significantly outperformed baseline models in event detection, with a *p*-value less than 0.05 across all comparisons (see Table 18).

Table 18. Value prediction significance test

Model	*t*-statistic	*p*-value
MI-FinText versus FinBERT	4.52	< 0.001
MI-FinText versus KG-BERT	6.88	< 0.001
MI-FinText versus DLSTA	8.21	< 0.001

The t-test results show statistically significant improvements in value prediction when using MI-FinText compared to the baseline models, with *p*-values well below 0.05.

DISCUSSION

This study introduced the multidimensional integrated model for MI-FinText, which significantly improved performance across several financial analysis tasks. Results showed that the model outperformed existing baseline models in sentiment analysis, event detection, value prediction, and knowledge graph construction. By combining multiple advanced techniques such as MTL, temporal modeling, and dynamic knowledge graph construction, MI-FinText provided a more accurate and comprehensive solution for understanding financial markets.

Despite its successes, one key area that has not been fully addressed is the model's performance during extreme market conditions, such as financial crises or periods of high volatility. To better understand its robustness, MI-FinText was evaluated using data from the 2008 global financial crisis and other turbulent market events. The results demonstrated that while the model's performance showed slight decreases in some areas, such as sentiment analysis accuracy and event detection precision, it still maintained a high level of stability and accuracy when compared to other baseline models like FinBERT and DLSTA. Specifically, the model was able to effectively capture sentiment fluctuations, detect significant events, and predict asset values, despite the instability of the market. This indicated that MI-FinText was resilient to market volatility, although further refinement is needed to improve its performance in extreme conditions.

Future work will focus on enhancing the model's adaptability and real-time prediction capabilities during extreme market events. For instance, integrating additional features, such as market sentiment indicators or real-time news data, could further increase the model's ability to respond to unexpected financial events. Additionally, fine-tuning the model's temporal modeling and MTL mechanisms could help improve its real-time decision-making performance, especially in rapidly changing market environments.

Moreover, MI-FinText's robust performance under extreme market conditions further supports the idea that incorporating diverse sources of information, including both structured and unstructured data, enhances the model's ability to adapt to various market scenarios. As financial markets continue to evolve, improving the model's dynamic adaptability will be crucial for its practical deployment in real-world financial decision-making.

In addition to its performance, MI-FinText also demonstrated the value of integrating MTL and knowledge graph construction in financial analysis. MTL allowed the model to perform sentiment analysis, event detection, and value prediction simultaneously, leveraging shared representations across tasks to improve generalization. Furthermore, the dynamic knowledge graph enabled the model to maintain an up-to-date view of the evolving relationships between financial entities, providing deeper insights into market behavior.

In conclusion, MI-FinText represents a significant advancement in financial text mining, offering an integrated framework that captures temporal dependencies, models dynamic relationships, and performs multiple tasks simultaneously. While the model performed well in normal market conditions, the findings from experiments in extreme market scenarios suggest that future research should continue to focus on enhancing the model's robustness and real-time adaptability. This will ensure that MI-FinText can provide valuable insights and predictions, even in the face of market volatility and unforeseen financial crises.

CONCLUSION

This study introduced the multidimensional integrated model for MI-FinText, which successfully integrated MTL, temporal modeling, and dynamic knowledge graph construction to tackle complex financial analysis tasks. The model demonstrated significant improvements over baseline models across key tasks, such as sentiment analysis, event detection, and value prediction, offering more accurate predictions and deeper insights into financial data. By combining textual data with numerical data and leveraging temporal dependencies, MI-FinText excelled in predicting future financial trends, detecting significant market events, and constructing rich knowledge graphs that enhance the understanding of financial relationships.

The model, however, is not without its limitations. While MI-FinText performed well under normal market conditions, its performance during extreme market conditions, such as financial crises or periods of high volatility, warrants further investigation. Experiments using data from the 2008 global financial crisis showed that the model's performance slightly decreased but remained stable compared to baseline models. This indicated that MI-FinText is resilient to market volatility, although additional improvements are needed to enhance its real-time adaptability during extreme events.

Future work will focus on optimizing MI-FinText for real-time applications, particularly in extreme market conditions. Enhancements could include the integration of more diverse data sources, such as market sentiment indicators and real-time financial news, which may help the model better respond to unexpected financial events. Additionally, refining the temporal modeling and MTL components will further improve the model's real-time prediction capabilities, making it more effective in rapidly changing market environments.

These findings suggest that MI-FinText is not only capable of providing valuable insights into stable market conditions but also holds significant potential for real-time financial decision-making during market crises. The robustness of MI-FinText, combined with its ability to handle both structured and unstructured data, offers a comprehensive solution for understanding and predicting market behavior, even in the face of economic turbulence.

In conclusion, MI-FinText represents a significant advancement in financial text mining and analysis. By integrating multiple tasks into a single framework, it offers a more comprehensive approach to understanding and predicting financial markets. While the model performed exceptionally well in normal market conditions, further work will focus on enhancing its robustness and adaptability during extreme market conditions, ensuring that it can continue to provide valuable insights and predictions for both researchers and practitioners in the financial domain.

To address the growing volume of financial data and the challenges of large-scale deployment, distributed processing frameworks will be crucial for scaling MI-FinText. As financial data continues to grow, traditional single-machine processing may no longer suffice. This study proposes adopting distributed computing frameworks, such as Apache Spark and Hadoop, to handle large-scale data processing across multiple nodes. These frameworks allow for parallel processing of data, enabling faster training and data handling, thus making it possible to deploy MI-FinText at scale.

Furthermore, the model could benefit from parallelization techniques, such as data parallelism and model parallelism, to improve training efficiency and handle massive datasets. Distributed storage systems like Hadoop Distributed File System or Amazon S3 could be utilized to store and manage vast amounts of financial data, ensuring high availability and efficient access. By leveraging these distributed processing and storage solutions, MI-FinText could be scaled to handle real-time data streams and large volumes of historical data, providing accurate and timely predictions in the face of ever-growing data volumes.

COMPETING INTERESTS

The authors of this publication declare there are no competing interests.

FUNDING STATEMENT

This research was supported by Chongqing Social Science Planning Project “Research on Tax Compliance Promotion Mechanism under the Background of Economic Digitalization,” [grant number 2019BS107], the Tax Accountants Office Joint Training Base for Graduate Students (Chongqing Education Research and Development [2020] No. 5), the Chongqing Education Commission Education Reform project, and the Chongqing Technology and Business University, Zhitong (Beijing).

PROCESSING DATES

This manuscript was initially received for consideration for the journal on 01/24/2025, revisions were received for the manuscript following the double-anonymized peer review on 03/17/2025, the manuscript was formally accepted on 03/16/2025, and the manuscript was finalized for publication on 04/07/2025

CORRESPONDING AUTHOR

Correspondence should be addressed to Juan Yang; juanyang@ctbu.edu.cn

REFERENCES

Abdullah, T., & Ahmet, A. (2022). Deep learning in sentiment analysis: Recent architectures. *ACM Computing Surveys*, *55*(8), 1–37. DOI: 10.1145/3548772

Abinaya, P., & Kumar, S. J. (2024). Assured and provable data expuncturing in cloud using ciphertext policy-attribute based encryption (CP-ABE). *Cybernetics and Systems*, *55*(4), 786–803. DOI: 10.1080/01969722.2023.2176654

Allioui, H., & Mourdi, Y. (2023). Exploring the full potentials of IoT for better financial growth and stability: A comprehensive survey. *Sensors (Basel)*, *23*(19), 8015. DOI: 10.3390/s23198015 PMID: 37836845

Almayyan, W. I., & AlGhannam, B. A. (2024). Detection of kidney diseases: Importance of feature selection and classifiers. *International Journal of E-Health and Medical Communications*, *15*(1), 1–21. DOI: 10.4018/IJEHMC.354587

Ashtiani, M. N., & Raahemi, B. (2023). News-based intelligent prediction of financial markets using text mining and machine learning: A systematic literature review. *Expert Systems with Applications*, *217*, 119509. DOI: 10.1016/j.eswa.2023.119509

Bello, O. A. (2023). Machine learning algorithms for credit risk assessment: An economic and financial analysis. *International Journal of Management*, *10*(1), 109–133. DOI: 10.37745/ijmt.2013/vol10n1109133

Cheng, M. Y., Kusoemo, D., & Gosno, R. A. (2020). Text mining-based construction site accident classification using hybrid supervised machine learning. *Automation in Construction*, *118*, 103265. DOI: 10.1016/j.autcon.2020.103265

Dash, B. (2022). *Information extraction from unstructured big data: A case study of deep natural language processing in fintech*. University of the Cumberlands.

DeMatteo, C., Jakubowski, J., Stazyk, K., Randall, S., Perrotta, S., & Zhang, R. (2024). The headaches of developing a concussion app for youth: Balancing clinical goals and technology. *International Journal of E-Health and Medical Communications*, *15*(1), 1–20. DOI: 10.4018/IJEHMC.352514

Dimlo, U. F., Rupesh, V., & Raju, Y. (2024). The dynamics of natural language processing and text mining under emerging artificial intelligence techniques. *International Journal of System Assurance Engineering and Management*, *15*(9), 4512–4526. DOI: 10.1007/s13198-024-02468-8

Dogra, V. (2021). Banking news-events representation and classification with a novel hybrid model using DistilBERT and rule-based features. *Turkish Journal of Computer and Mathematics Education*, *12*(10), 3039–3054.

El Ghoul, S., Guedhami, O., Mansi, S. A., & Sy, O. (2022). Event studies in international finance research. *Journal of International Business Studies*, *54*(2), 344–364. DOI: 10.1057/s41267-022-00534-6 PMID: 35822194

Fang, G., Deng, Y., Ma, H., Zhang, J., & Pan, L. (2023). Energy financial risk management in China using complex network analysis. *Journal of Organizational and End User Computing*, *35*(1), 1–29. DOI: 10.4018/JOEUC.330249

Feng, L., & Sinchai, A. (2024). Transfer learning model for cash-instrument prediction adopting a transformer derivative. *Journal of King Saud University. Computer and Information Sciences*, *36*(3), 102000. DOI: 10.1016/j.jksuci.2024.102000

Frankel, R., Jennings, J., & Lee, J. (2022). Disclosure sentiment: Machine learning vs. dictionary methods. *Management Science*, *68*(7), 5514–5532. DOI: 10.1287/mnsc.2021.4156

Guo, J. (2022). Deep learning approach to text analysis for human emotion detection from big data. *Journal of Intelligent Systems*, *31*(1), 113–126. DOI: 10.1515/jisys-2022-0001

Gupta, A., Dengre, V., Kheruwala, H. A., & Shah, M. (2020). Comprehensive review of text-mining applications in finance. *Financial Innovation*, *6*(1), 1–25. DOI: 10.1186/s40854-020-00205-1

Klyuchnikov, N., Trofimov, I., Artemova, E., Salnikov, M., Fedorov, M., Filippov, A., & Burnaev, E. (2022). Nas-bench-NLP: Neural architecture search benchmark for natural language processing. *IEEE Access: Practical Innovations, Open Solutions*, *10*, 45736–45747. DOI: 10.1109/ACCESS.2022.3169897

Krasnyuk, M., Kulynych, Y., & Krasniuk, S. (2022). Knowledge discovery and data mining of structured and unstructured business data: Problems and prospects of implementation and adaptation in crisis conditions. *Grail of Science*, 63–70.

Kumbure, M. M., Lohrmann, C., Luukka, P., & Porras, J. (2022). Machine learning techniques and data for stock market forecasting: A literature review. *Expert Systems with Applications*, *197*, 116659. DOI: 10.1016/j.eswa.2022.116659

Li, H., Peng, Q., Mou, X., Wang, Y., Zeng, Z., & Bashir, M. F. (2023). Abstractive financial news summarization via transformer-BiLSTM encoder and graph attention-based decoder. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *31*, 3190–3205. DOI: 10.1109/TASLP.2023.3304473

Li, K., Zhang, Y., & Chen, S. (2024). Digital transformation and organizational resilience of manufacturing enterprises under the background of green technological innovation. *Journal of Xi'an University of Finance and Economics*, *37*(04), 84–96. DOI: 10.19331/j.cnki.jxufe.2024.04.003

Li, Q., Chen, H., Huang, X., He, M., Ning, X., Wang, G., & He, F. (2025). Oral multi-pathology segmentation with lead-assisting backbone attention network and synthetic data generation. *Information Fusion*, *102892*, 102892. DOI: 10.1016/j.inffus.2024.102892

Liang, K., Meng, L., Liu, M., Liu, Y., Tu, W., Wang, S., Zhou, S., Liu, X., Sun, F., & He, K. (2024). A survey of knowledge graph reasoning on graph types: Static, dynamic, and multi-modal. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *46*(12), 9456–9478. DOI: 10.1109/TPAMI.2024.3417451 PMID: 38941209

Liu, Z., Huang, D., Huang, K., Li, Z., & Zhao, J. (2021). FinBERT: A pre-trained financial language representation model for financial text mining. *Proceedings of the 29th international conference on International Joint Conferences on Artificial Intelligence*.

Lu, Q., Sun, X., Long, Y., Gao, Z., Feng, J., & Sun, T. (2023). Sentiment analysis: Comprehensive reviews, recent advances, and open challenges. *IEEE Transactions on Neural Networks and Learning Systems*. DOI: 10.1109/TNNLS.2023.3294810 PMID: 37478046

Nahar, J., Rahaman, M. A., Alauddin, M., & Rozony, F. Z. (2024). Big data in credit risk management: A systematic review of transformative practices and future directions. *International Journal of Management Information Systems and Data Science*, *1*(04), 68–79. DOI: 10.62304/ijmisds.v1i04.203

Nyakurukwa, K., & Seetharam, Y. (2023). The evolution of studies on social media sentiment in the stock market: Insights from bibliometric analysis. *Scientific African*, *20*, e01596. DOI: 10.1016/j.sciaf.2023.e01596

Qi, N., Zhang, Y., Lin, X., & Chen, S. (2023). Consumers' cognitive dimension of international corporate social responsibility and its correlation with purchasing tendency. *Journal of Organizational and End User Computing*, *35*(1), 1–17. DOI: 10.4018/JOEUC.322768

Rouf, N., Malik, M. B., Arif, T., Sharma, S., Singh, S., Aich, S., & Kim, H. C. (2021). Stock market prediction using machine learning techniques: A decade survey on methodologies, recent developments, and future directions. *Electronics (Basel)*, *10*(21), 2717. DOI: 10.3390/electronics10212717

Sadorsky, P. (2021). A random forests approach to predicting clean energy stock prices. *Journal of Risk and Financial Management*, *14*(2), 48. DOI: 10.3390/jrfm14020048

Shavandi, A., & Khedmati, M. (2022). A multi-agent deep reinforcement learning framework for algorithmic trading in financial markets. *Expert Systems with Applications*, *208*, 118124. DOI: 10.1016/j.eswa.2022.118124

Tarka, P., & Jędrych, E. (2020). On the unstructured big data analytical methods in firms: Conceptual model, measurement, and perception. *Big Data*, *8*(6), 478–500. DOI: 10.1089/big.2020.0123 PMID: 33202160

Thakkar, A., & Chaudhari, K. (2021a). A comprehensive survey on deep neural networks for stock market: The need, challenges, and future directions. *Expert Systems with Applications*, *177*, 114800. DOI: 10.1016/j.eswa.2021.114800

Thakkar, A., & Chaudhari, K. (2021b). Fusion in stock market prediction: A decade survey on the necessity, recent developments, and potential future directions. *Information Fusion*, *65*, 95–107. DOI: 10.1016/j.inffus.2020.08.019 PMID: 32868979

Xiao, J., Wang, J., Bao, W., Deng, T., & Bi, S. (2024). Application progress of natural language processing technology in financial research. *Financial Engineering and Risk Management, 7*(3), 155–161. https://www.doi.org/10.23977/ferm.2024.070320

Yao, L., Mao, C., & Luo, Y. (2019). KG-BERT: BERT for knowledge graph completion. arXiv. Retrieved [insert date] from arXiv preprint arXiv:1909.03193.

Zhang, H., Yu, L., Wang, G., Tian, S., Yu, Z., Li, W., & Ning, X. (2025). Cross-modal knowledge transfer for 3D point clouds via graph offset prediction. *Pattern Recognition*, *111351*, 111351. DOI: 10.1016/j.patcog.2025.111351

Zhang, X. (2024). Analyzing financial market trends in cryptocurrency and stock prices using CNN-LSTM Models. DOI: 10.20944/preprints202407.1119.v1

Zhang, Y., & Yang, Q. (2021). A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*, *34*(12), 5586–5609. DOI: 10.1109/TKDE.2021.3070203

Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., & Li, H. (2019). T-GCN: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, *21*(9), 3848–3858. DOI: 10.1109/TITS.2019.2935152

Zhou, L., Zhang, X., & Dong, K. (2024). Does digital financial innovation contribute to promoting the high-quality development of the real economy? Mechanism analysis and spatial econometrics based on financial service efficiency. *Journal of Xi'an University of Finance and Economics*, *37*(01), 60–72. DOI: 10.19331/j.cnki.jxufe.20231115.001