

Article

Lightweight Scheme to Capture Stock Market Sentiment on Social Media Using Sparse Attention Mechanism: A Case Study on Twitter

Sihan Wu ^{1,*} and Fuyu Gu ²

¹ School of Accountancy, Shanghai University of Finance and Economics, Shanghai 200433, China

² Department of Computing, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China

* Correspondence: wsh0313543@163.sufe.edu.cn

Abstract: Over the years, people have invested in stock markets in order to maximize their profit from the money they possess. Financial sentiment analysis is an important topic in stock market businesses since it helps investors to understand the overall sentiment towards a company and the stock market, which helps them make better investment decisions. Recent studies show that stock sentiment has strong correlations with the stock market, and we can effectively monitor public sentiment towards the stock market by leveraging social media data. Consequently, it is crucial to develop a model capable of reliably and quickly capturing the sentiment of the stock market. In this paper, we propose a novel and effective sequence-to-sequence transformer model, optimized using a sparse attention mechanism, for financial sentiment analysis. This approach enables investors to understand the overall sentiment towards a company and the stock market, thereby aiding in better investment decisions. Our model is trained on a corpus of financial news items to predict sentiment scores for financial companies. When benchmarked against other models like CNN, LSTM, and BERT, our model is “lightweight” and achieves a competitive latency of 10.3 ms and a reduced computational complexity of 3.2 GFLOPS—which is faster than BERT’s 12.5 ms while maintaining higher computational complexity. This research has the potential to significantly inform decision making in the financial sector.



Citation: Wu, Sihan, and Fuyu Gu. 2023. Lightweight Scheme to Capture Stock Market Sentiment on Social Media Using Sparse Attention Mechanism: A Case Study on Twitter. *Journal of Risk and Financial Management* 16: 440. <https://doi.org/10.3390/jrfm16100440>

Academic Editor: Rakesh Gupta

Received: 24 August 2023

Revised: 4 October 2023

Accepted: 5 October 2023

Published: 10 October 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: sentiment analysis; stock market; transformer; social media; text mining; sparse attention

1. Introduction

A nation’s stock market is one of the foundations of its economy [Gupta and Singh \(2017\)](#); [Sanboon et al. \(2019\)](#). As part of economic liberalization, stock markets play the most significant role in the financial strategies of the worldwide corporate sector [Gandhmal and Kumar \(2019\)](#); [Jiang \(2021\)](#). On the other hand, emotion-driven trading has emerged as a powerful influence on the dynamics of the stock market. Understanding the sentiment around a financial asset can provide valuable insights into its future performance. In this digital era, social media platforms like Twitter serve as a vast source of public opinion and sentiment, which can be used to make more informed financial decisions. The most important choice for investors is what to do with a particular stock, i.e., whether to buy, sell, or hold the stock’s shares. If investors are able to invest in the proper stocks, they will generate substantial profits; otherwise, they risk losing their money, which would be detrimental to them and their country. Therefore, it is necessary to develop such prediction models [Nabipour et al. \(2020\)](#); [Pang et al. \(2020\)](#) that can help more accurately and effectively anticipate the values of stocks. Understanding the sentiment towards a particular stock or the market as a whole is crucial to making informed investment decisions. These decisions, in turn, have far-reaching implications not only for individual investors but also for the broader economic landscape [Gupta and Singh \(2017\)](#); [Sanboon et al. \(2019\)](#). Stock markets serve as the backbone of a nation’s economy. Their performance is a key indicator

of economic health, making it vital to develop tools that can guide investors in making profitable choices [Arora et al. \(2017\)](#); [Saxena et al. \(2021\)](#). However, the volatile nature of financial markets makes it a risky endeavor, where the line between substantial profits and crippling losses is exceedingly thin [Gupta and Singh \(2020\)](#); [Singh and Gupta \(2020\)](#). Given the significant role that stock markets play in economic liberalization and corporate financing strategies worldwide [Gandhmal and Kumar \(2019\)](#); [Jiang \(2021\)](#), accurate and effective prediction models are of paramount importance [Nabipour et al. \(2020\)](#); [Pang et al. \(2020\)](#). This paper proposes a novel and effective model for financial sentiment analysis, with the aim of better equipping investors in this uncertain environment.

Numerous studies in the literature have consistently demonstrated the significant association between the sentiment of social media and the stock market [Liu \(2012\)](#). Consequently, there is substantial value in analyzing the sentiment of the stock market for practical and research purposes. Recently, emerging attention has been paid to analyzing investor sentiment via social media, particularly among young and inexperienced investors. Several research works have focused on using Twitter sentiment to forecast stock market trends [Gandhmal and Kumar \(2019\)](#); [Jiang \(2021\)](#); [Mishev et al. \(2020\)](#); [Pang et al. \(2020\)](#); [Pota et al. \(2020\)](#); [Zhao et al. \(2016\)](#).

Sentiment analysis is regarded as a classical problem in natural language processing (NLP), which aims to determine people’s opinions, sentiments, and preferences regarding entities such as products, services, organizations, and individuals. However, stock sentiment analysis faces two major challenges, as shown below:

- Challenge 1: Mismatch between conventional and stock sentiment.** The first challenges results from the fact that conventional sentiment analysis significantly differs from stock sentiment analysis. In a detailed analysis, it becomes evident that stock sentiment, though bearing certain correlations, markedly diverges from the traditional sentiment often assessed in academic contexts such as consumer feedback studies, literature reviews, and broader public sentiment analyses. Traditional sentiments are primarily anchored in the emotional spectrum, capturing the nuances between positive and negative affective states [Liu \(2012\)](#). On the contrary, stock sentiment is intrinsically tied to market dynamics, reflecting anticipations of stock price movements and whether they indicate bullish or bearish trends. While there are scenarios where stock sentiment aligns with traditional sentiment, there are also instances where the two sentiments manifest stark disparities. For instance, a public discourse may show skepticism toward a particular economic event, yet there could be an underlying optimism about the potential appreciation in stock value for a company like \$TSLA, highlighting a bullish stock sentiment. An extensive compilation of such instances is presented in [Table 1](#).

Table 1. The social media examples on Twitter show the sentiment mismatches between conventional sentiment and stock sentiment due to the difference in sentiment definitions.

Social Media Content	Conventional Sentiment	Stock Sentiment
\$TSLA long.	Negative	Positive
Be Prepared For A DOGE Crash Elon on SNL Dogecoin New Price Predictions.	Negative	Neutral
Buy the f*cking dip! Hold the line! \$AMC \$GME \$NOK	Negative	Positive

- Challenge 2: High computational complexity of deep learning models.** In recent years, deep learning models, particularly transformers, have achieved state-of-the-art performance across a myriad of tasks in natural language processing, computer vision, and beyond. However, a significant impediment to their broader application and scalability remains the high computational complexity associated with their architecture [Lin et al. \(2022\)](#). Such complexity not only demands substantial computa-

tional resources but also poses challenges for real-time processing and deployment in resource-constrained environments. Figure 1 shows that computing the softmax attention constantly dominates (52–58%) the MHA runtime in transformer architecture, particularly as devices grow less powerful and resource constrained. Recognizing these challenges, this paper proposes the adoption of sparse transformers, a variant optimized to reduce computational overhead without compromising the model’s efficacy. By leveraging the sparsity inherent in the transformer’s attention mechanism, we aim to achieve a balance between computational efficiency and model performance, paving the way for more sustainable and scalable deep learning applications.

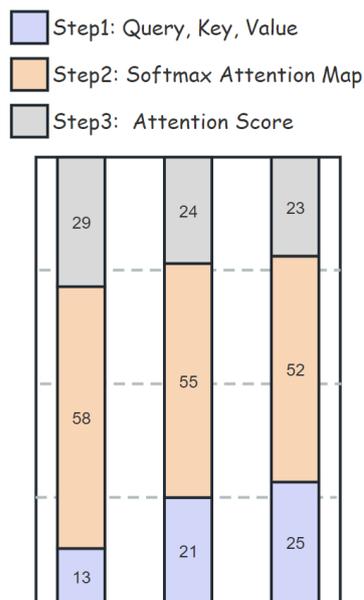


Figure 1. Runtime breakdown of MHA on various devices.

This research realizes more computationally efficient financial sentiment analysis using a sequence-to-sequence model. And the most trending model nowadays is transformer Vaswani et al. (2017), which is a type of natural language processing (NLP) model that can provide outputs that are responsive to context Yang et al. (2020). The transformer model is trained to predict sentiment scores for financial companies using a corpus of financial news items. This sentiment forecast is then utilized to determine the market sentiment Mishev et al. (2020) as a whole. The results demonstrate that the transformer model can generate reliable sentiment ratings and can be used to detect market sentiment in real time. Additionally, the algorithm can generate sentiment scores that are sensitive to the dynamic character of the financial market. In this paper, we present a novel approach for financial sentiment analysis using a sequence-to-sequence model transformer Pota et al. (2020) with sparse attention. The transformer model was first introduced by Google Vaswani et al. (2017) to finish tasks involving machine translation, which is adept at recognizing long-term dependencies from data. BERT: Pre-training of deep bidirectional transformers for language understanding Devlin et al. (2018), a transformer-based model using only encoder modules in natural language processing, attempts to broaden the original transformer’s applicability so that it may serve as a general-purpose backbone for tasks in NLP.

The following is a summary of the key contributions: (1) In this paper, a novel and effective method for financial sentiment analysis is proposed, and its applicability is proven using a real-world sentiment analysis dataset. According to the findings of the trial, the proposed strategy exceeds the most recent methodologies on three performance metrics. (2) According to our knowledge, this is the case. Compared with the original transformer, the performance of this Bert-based transformer structure is superior to SVM, LR, and NBM Neunenschwander et al. (2014); Sohngir et al. (2018); Zhao et al. (2016). The remainder of

this paper is organized as follows. In Section 2, the related work is introduced in detail. The proposed method is subsequently presented in Section 3. In Section 4, the outcomes are depicted. Section 5 concludes with a brief conclusion, limitations, and future work analysis.

2. Related Works

2.1. Sentiment Analysis and Related Financial Applications

Sentiment analysis is a critical workload that has been widely studied in the research community [Aziz et al. \(2022\)](#); [Hasselgren et al. \(2022\)](#); [Pathak et al. \(2021\)](#); [Ruan et al. \(2018\)](#). One of the previous works [Pathak et al. \(2021\)](#) leverages the topic-level sentiment analysis model, which extracts the topic at the sentence level using online latent semantic indexing, and then applies the topic-level attention mechanism in a long short-term memory network.

Financial applications of sentiment analysis include a variety of topics, and previous work performed sentiment analyses at various levels of granularity. The authors in [Aziz et al. \(2022\)](#) propose the Light Gradient Boosting Machine (LGBM) approach to accurately identify fraud for blockchain transactions, such as Ethereum. A trust management framework based on sentiment analysis is proposed in [Ruan et al. \(2018\)](#) to build a trust network for Twitter users. This work considers a reputation mechanism to amplify the correlation between firms' Twitter sentiment valence and the corresponding stock's abnormal returns. [Hasselgren et al. \(2022\)](#) studied how to use the sentiment of public social networks to make investment decisions. The authors present a model to track stock market performance based on the results of sentiment analysis obtained from social media.

2.2. Existing Deep Learning Models for Sentiment Analysis

2.2.1. Seq2Seq Model

Sequence to Sequence (Seq2Seq) models are an effective sort of neural network employed in NLP applications. They are neural networks that receive a data sequence as input and produce another data sequence as output. Seq2Seq models can learn the context of a sentence and derive the meaning of individual words and phrases. They are utilized in numerous applications, including machine translation, chatbot creation, automatic summarization, and text-to-speech conversion. Seq2Seq models like long short-term memory (LSTM) [Hochreiter and Schmidhuber \(1997\)](#), recurrent neural networks (RNNs) [Medsker and Jain \(2001\)](#), and Gated Recurrent Unit (GRU) [Dey and Salem \(2017\)](#) have demonstrated efficacy in a range of tasks, making them an in-demand resource in the field of natural language processing.

2.2.2. LSTM Model

The use of long short-term memory (LSTM) networks has been researched in the area of financial sentiment analysis in recent years [Gupta et al. \(2022\)](#). Financial sentiment analysis is an important issue in stock market businesses, since it can help investors understand the overall sentiment towards a company and the stock market, which can help them make better investment decisions. Sentiment analysis can also help provide insight into general public opinion, which can be useful for making business decisions [Man et al. \(2019\)](#); [Wang et al. \(2016\)](#). LSTM networks, which are a sort of recurrent neural network, are suitable for modeling temporal data and have been proven to be effective in a variety of applications ([Lin et al. 2017](#); [Wang et al. 2019](#); [Zhao et al. 2017](#)), including financial sentiment analysis. LSTM can extract useful information from time series data; however, its performance decreases as the input sequence increases [Qin et al. \(2017\)](#).

2.2.3. Transformer Model

In recent years, the fast development of AI technology has led to the emergence of increasingly powerful algorithms. In general, newer, more potent algorithms have a better data processing capacity [Zhou and Xue \(2018\)](#). The transformer model [Vaswani et al. \(2017\)](#) is a unique and cutting-edge AI program. [Lin et al. \(2022\)](#). Recent research has examined the use of transformer-based models in various complex tasks. A transformer is a type of

neural network design that has been shown to perform well in natural language processing tasks and has been implemented in a number of other disciplines as well [Dong et al. \(2018\)](#); [Dosovitskiy et al. \(2020\)](#); [Khan et al. \(2022\)](#). We adopt a bidirectional transformer for financial sentiment analysis, a BERT-based transformer [Devlin et al. \(2018\)](#), which greatly outperforms the traditional transformer.

2.2.4. BERT

Google AI created BERT (Bidirectional Encoder Representations from Transformers) in 2018 [Devlin et al. \(2018\)](#) as a new natural language processing (NLP) technique. Its performance has surpassed the accuracy of numerous existing cutting-edge NLP models. BERT is a deep learning model based on unsupervised learning that can efficiently learn from unlabeled text, enabling it to perform a variety of tasks like sentiment analysis, text classification, text generation, question answering, and entity extraction. BERT is a powerful tool for natural language processing and comprehension that has been utilized effectively in a variety of applications and is rapidly becoming the industry standard for NLP tasks.

3. Proposed Methods

The primary objective of this paper is a financial sentiment analysis using a deep learning-based sequence model. Hence, a pre-trained model BERT using transformer architecture was used for classification, specifically by first taking financial texts as inputs and then feeding them into BERT. The details will be introduced in Section 3.3.

3.1. Overview of Sentiment Analysis Pipeline

Figure 2 depicts the comprehensive pipeline of our proposed approach. Within this schematic, the letter “E” stands for embedding. This is the preliminary phase where the Twitter dataset undergoes preprocessing to convert its textual content into machine-readable vector representations. Subsequently, the symbols “C” and “T” signify the ultimate hidden states generated by the transformer architecture, encapsulating deep contextual information within the text. In particular, the unique token “[CLS]” in BERT is employed as a specialized marker for classification tasks, serving to encapsulate an aggregated understanding of the entire sentence or text segment.

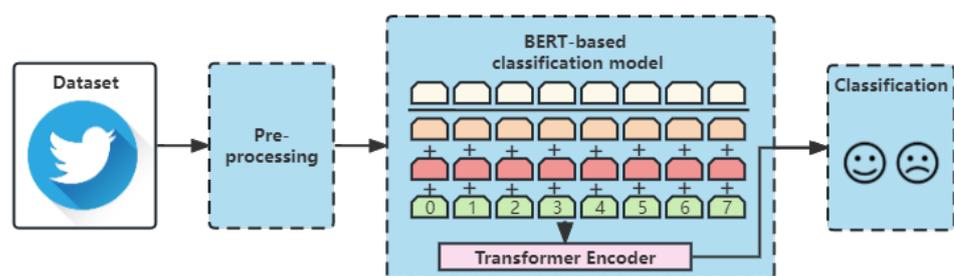


Figure 2. Overview of the pipeline. E stands for embedding, C and T stand for the ultimate concealed states provided by the transformer architecture, and [CLS] is the BERT special classification token.

Central to this pipeline is a BERT-based classification model, an advanced deep learning model particularly specialized in text classification tasks. The process begins with the preprocessing of the Twitter dataset to ensure data quality and uniformity. Upon preprocessing, the data are ingested into the model and traverse through the multi-layered transformer architecture, ultimately resulting in the final classification outcome.

Our selection of the Twitter dataset is motivated by its abundant textual content and its characteristics in real time, which offer a wide range of training samples for our model. Additionally, BERT-based models have previously exhibited exceptional performance in a

diverse range of tasks. Taking advantage of this proven architecture, we aim to achieve efficient and precise classification of Twitter text data.

3.2. Transformer Architecture

Transformer architecture is typically separated into two components, as shown in the figure; one is for the encoder, as shown in Figure 3, and the other for the decoder. Only the encoder needs to travel through the encoder to learn the representation because we only need to classify the texts for sentiment analysis. Devlin et al. (2018); Dosovitskiy et al. (2020); Liu et al. (2021). To produce predictions or perform classification for the downstream model, the transformer encoder is made to take in a sequence of tokens as an input and encode them into a lower-dimensional representation. The model can capture long-range dependencies in the inputs and produce a more accurate representation of the inputs, thanks to the transformer encoder’s self-attention mechanism.

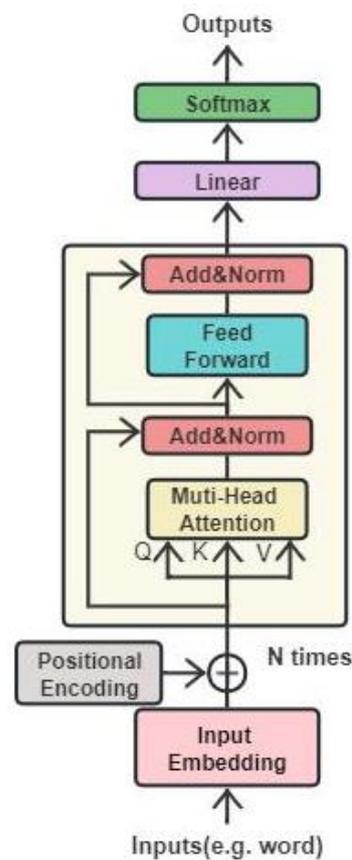


Figure 3. Transformer encoder architecture.

The separation of vectors from input tokens (for example, words, signals, images, etc.), or embeddings, is the initial stage in the encoding process. We assume that a sequence of input length n is $(x_1, x_2, \dots, x_n), x \in \mathbb{R}^{d_{model}}$. These embeddings preserve the meaning of each token in the input sequence and serve as the foundation for the model’s calculation.

Positional Encoding. The order of the tokens is significant in some tasks, but the transformer model, which employs a self-attention mechanism, is not naturally able to capture this order. As a result, the model uses positional encoding (1) to supplement the input embeddings with additional information that encodes the positions of each token in the input sequence.

$$\begin{aligned} PE_{pos,2i} &= \sin(pos/10000^{2i/d_{model}}) \\ PE_{pos,2i+1} &= \cos(pos/10000^{2i/d_{model}}) \end{aligned} \tag{1}$$

The input embeddings are then subjected to self-attention techniques by the transformer encoder. By valuing each input embedding according to its importance to all other input embeddings, self-attention enables the model to capture long-range dependencies in the input text. The transformer encoder adds one or more feed-forward layers to the encoded representation after applying the self-attention methods.

Self-attention mechanism. The input token consists of queries (Q), keys (K) and values (V) of dimension d_{model} . It is created by averaging the input across the three learnable matrices W_q, W_k and W_v .

$$Q, K, V = X \cdot W_q, X \cdot W_k, X \cdot W_v, \tag{2}$$

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \tag{3}$$

Concretely, d_k is the hidden dimension, which can be the same as d_{model} , and scaled dot-product attention is used in this work.

Multi-head attention mechanism. The input embeddings are divided into various “heads” for the multi-head attention mechanism, and self-attention is applied to each head separately. The model can capture various kinds of dependencies in input tokens because each head learns to weight the input embeddings based on their relevance to the other input embeddings in the head. The output of multi-head attention looks like this (4), and it illustrates the detailed information between scaled dot-product attention and multi-head attention, as shown in Figure 4:

$$MultiHead(Q, K, V) = Concat(head_1, head_2, \dots, head_h)W^O, \tag{4}$$

$$head_i = Attention(XW_i^Q, XW_i^K, XW_i^V),$$

where the projections are matrices of parameters $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}, W_i^K \in \mathbb{R}^{d_{model} \times d_k}, W_i^V \in \mathbb{R}^{d_{model} \times d_k}$ and $W^O \in \mathbb{R}^{hd_v \times d_{model}}$. Here, $hd_v = d_k$, usually h , can be set as eight.

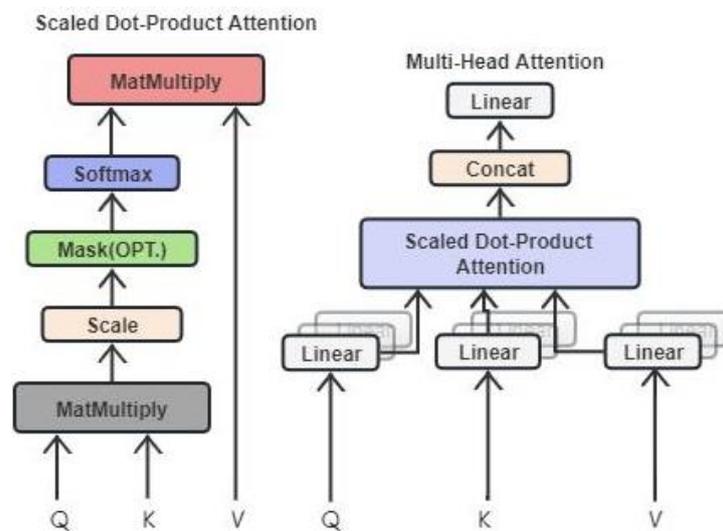


Figure 4. (Left) Scaled dot-product attention. (Right) Multi-head attention consists of numerous concurrent attention levels.

3.3. Pre-Trained Model BERT

BERT Devlin et al. (2018) is one of the most well-liked designs for contemporary language modeling. Its capacity for generalization enables it to be tailored to various downstream tasks depending on the requirements, whether it is NER, classification, question-answering, or sentiment analysis. The parameters of the most internal layers of the archi-

texture are fixed because the core of the architecture was trained on exceptionally huge text corpora. Instead, the layers closest to the surface are those that adjust to the task and are where the so-called fine-tuning is conducted. In Figure 5, a condensed overview is displayed.

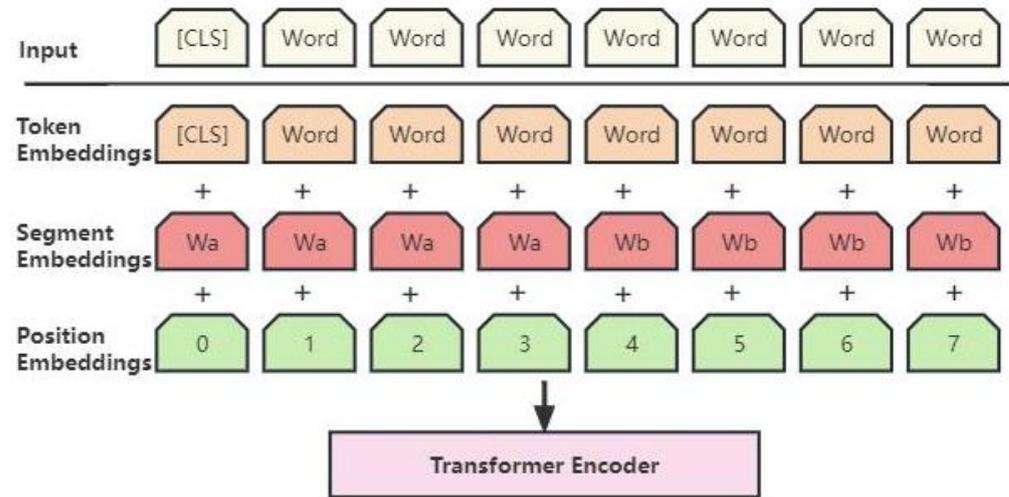


Figure 5. Input representation and the BERT architecture. The total of the token embeddings, segmentation embeddings, and position embeddings constitutes the input embeddings.

The foundation of BERT is the transformer. Think of the input x , which consists of two different phrases. The [SEP] token is situated in a specific position, while the [CLS] token is situated before x . LN is the normalization layer and E is the embedding function. Then, the embedding is obtained by:

$$\hat{h}_i^0 = E_{pos}(i) + E_{seg}(x_i) + E(x_i) \tag{5}$$

The embeddings are subsequently put through M transformer blocks. For each transformer block, it is true that using the Feed Forward (FF) layer, the Multi-Head Self-Attention (MHSA) function mentioned above, and the element-wise Gaussian Error Linear Units (GELU) activation function [Hendrycks and Gimpel \(2016\)](#):

$$\hat{h}^{i+1} = LN(FF, LN(MHSA, h^i)) \tag{6}$$

$$LN(f, h) = LayerNorm(h + Dropout(f(h))) \tag{7}$$

The loss function in BERT is a measure of how well the model is able to predict the correct word in a given context. It is a combination of two objectives: the probability of a correct prediction, and the Masked Language Model (MLM). The MLM objective forces the model to predict randomly masked words from the input sentence, and encourages the model to learn the surrounding context to make the correct predictions. The overall loss is then the sum of the individual losses for each prediction:

$$L_{MLM} = - \sum_{i=1}^k \log(P(MASK_i = token_i | \bar{X}; \theta)) \tag{8}$$

where 15% of the input tokens are randomly masked via the Masked Language Modeling (MLM) method used by BERT. As a result, it may learn the connections between the words in the phrase as well as their context. [Devlin et al. \(2018\)](#). The transformer encoder uses θ to describe the probability P . $MASK_i$ denotes the masked token at the i_{th} point in the token sequence, and \bar{X} represents X after masking.

3.4. Sparse Attention Mechanism

A self-attention layer includes a connection pattern $S = \{S_1, \dots, S_n\}$, where S_i denotes the set of indices of the input vectors to which the i -th output vector attends. A self-attention layer transfers a matrix of input embeddings X to an output matrix. The output vector is a weighted sum of the transformations of the input vectors:

$$A(X, S_i) = \text{Attention}(Q_i, K_{S_i}, V_{S_i}) \tag{9}$$

$$\text{Attention}(Q_i, K_{S_i}, V_{S_i}) = \text{Softmax}\left(\frac{Q_i \cdot K_{S_i}^T}{\sqrt{d_k}}\right) \cdot V_{S_i} \tag{10}$$

For transformer models, full self-attention ($S_i : \{\forall x_j \in X\}$) allows each element to pay attention to both its own position and all prior and subsequent locations, which is shown in the left of Figure 6. According to Child et al. (2019), layers may learn a wide range of specialized sparse structures, which may explain their adaptability to different domains. Several of the network’s early layers learn locally connected patterns that mimic convolution. In a deeper layer, the network learns to divide its attention into rows and columns, essentially factoring the global attention calculation. Moreover, various attention layers exhibit global, data-dependent access patterns. Since the image is being used as an input, a natural approach for computer vision to define a factorized attention pattern in two dimensions is to use strided attention, in which one head attends to the previous l th places while the other attends to the subsequent l th locations; l is usually chosen to be close to \sqrt{n} . The right of Figure 6 shows the length of l is two.

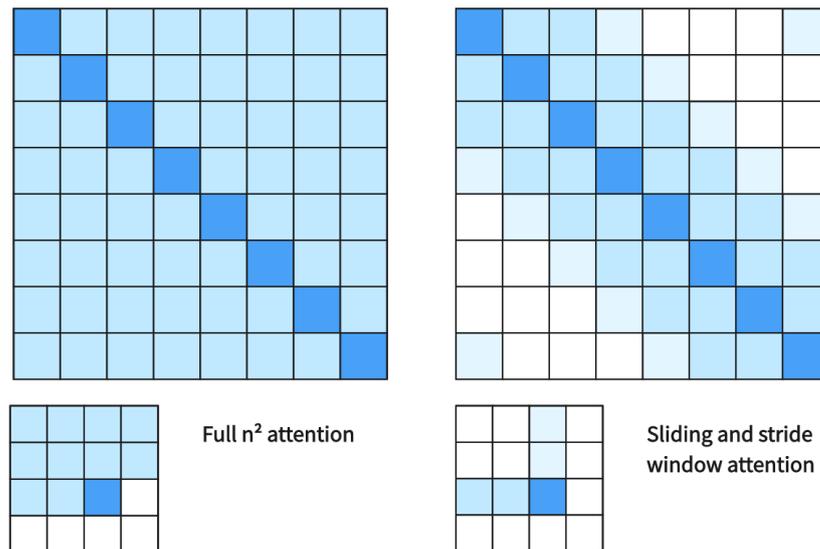


Figure 6. Comparing the full self-attention pattern and the configuration of attention patterns.

Formally, $A_i^{(1)} = \{i - l, i - l + 1, \dots, i + l\}$ and $A_i^{(2)} = \{j : |i - j| \bmod l = 0\}$. This formulation is useful if the data already have a natural structure that fits the stride, such as photos or some kinds of music. In light of the aforementioned advantages of the sparse attention mechanism, we integrated this approach into our customized BERT model for stock sentiment analysis. By doing so, we anticipate not only a substantial reduction in computational complexity but also an enhancement in the model’s ability to discern intricate patterns in stock-related textual data. The adaptability of the sparse attention mechanism, as demonstrated in various domains, holds promise for capturing the nuanced sentiments and fluctuations inherent in stock market discourse. Preliminary results, as will be discussed in subsequent sections, demonstrate that the sparse attention mechanism significantly reduces the computational complexity faced by our BERT model for stock sentiment analysis. This optimization not only streamlines the processing but

also sets a foundation for the development of more efficient models in the domain without compromising performance.

4. Experiments

This section examines and explains the proposed stock sentiment methods based on the BERT transformer. The datasets that were used in this study are thoroughly introduced. The metrics and experimental results of this technique are illustrated in the following sections.

4.1. Experimental Setup

Dataset Introduction and Acquisition

Setup. We performed our experiments on one of the most well-known microblogging platforms, Twitter, which is crucial in sentiment research for a number of areas, including predicting election results and cryptocurrency prices [Abraham et al. \(2018\)](#). We used the official API tool, Tweepy [Almatrafi et al. \(2015\)](#), to collect tweet data for research purposes. We also used the open-source Python text processing toolkit, TextBlob, which offers an API for standard NLP operations like part-of-speech tagging, noun phrase extraction, sentiment analysis, etc. We conducted our experiments on a high-performance computing environment equipped with a 12-core Intel CPU and NVIDIA RTX 3090 graphics card. This configuration allowed us to train and test our models efficiently, thanks to the card's superior computational capabilities.

Evaluation Dataset Overview. We used the TweetFinSent dataset, which is a collection of 2113 tweets, specifically curated for sentiment analysis in the financial domain [Pei et al. \(2022\)](#). Table 2 summarizes the key characteristics of the evaluated dataset. The dataset's sentiments are categorized into positive, neutral, and negative labels, with respective sample counts of 816, 1030, and 267. The dataset mostly covers the retailing sector since the Twitter tickers include the famous retailing brands, such as AMC, GameStop (GME), and Tesla (TSLA). Notably, the dataset exhibits an imbalance in sentiment distribution, with negative samples being the least represented.

Table 2. Key properties of the evaluated TweetFinSent dataset [Pei et al. \(2022\)](#).

Dataset Property	Value
Language	English
Training samples	1113
Testing samples	1000
Total samples	2113
Positive samples	816
Neutral samples	1030
Negative samples	267
Ticker	AMC, BABA, BB, BBBY, CLOV, GME, NOK, PFE, PLTR, SHOP, SOFI, SPCE, SQ, TLRY, TSLA, VIAC, ZM

Data Preparation. After collecting the social media content from the Internet, the raw data cannot be directly loaded into the sentiment analysis pipeline in Figure 2. This is because the collected dataset often contains noise and content (due to the random and creative use of social media by users) that are difficult to be parsed by the transformer model. For instance, tweets from Twitter normally contain special contents such as emojis, emoticons, hashtags, and user mentions, as well as web constructs like email addresses and URLs. Moreover, there are other noises, including phone numbers, percentages, money amounts, times, dates, and generic numbers that impact the effectiveness of down-stream sentiment analysis. In this work, we adopt a series of data preprocessing techniques to convert noisy data into noise-less contents. We preprocess the raw data from social media in the following steps based on the given content: 1. We first preprocess the collected data by removing the impact of various types of data: dates, emails, money amounts, numbers, percentages, and phone numbers. 2.

Secondly, URLs, username, and hashtags are not processed since these contents may indicate meaningful sentiment in the financial domain.

Annotation and Agreement. To ensure the quality and reliability of annotations, the dataset employed a rigorous annotation process. Inter-annotator agreement was assessed using Cohen’s Kappa (κ), yielding an average κ of 0.67, indicating a moderate level of agreement. To further enhance data quality, conflicts in annotations were resolved through discussions among annotators. In the post-conflict resolution, the dataset achieved an impressive overall agreement of 88.5%, surpassing some existing sentiment analysis datasets, such as the Obama–McCain Debate dataset with an agreement of 83.7%.

Sentiment Distribution and Analysis. The dataset’s sentiment distribution reveals insights into the prevailing discussions on social media during the data collection period. The most discussed stocks, often referred to as “meme stocks”, gained significant traction among retail investors. A deeper dive into the dataset’s content is visualized in Figure 7. The most frequent terms in TweetFinSent with different sentiment classes reveal distinct terminologies and expressions associated with each sentiment category. Positive tweets frequently contained phrases like “to the moon” and “buy the dip”, indicating optimistic financial outlooks. In contrast, negative tweets often discussed overvalued stocks and potential sales, reflecting pessimistic sentiments. Neutral tweets, on the other hand, predominantly shared news or statistical insights about the stock market.

Textual Analysis. Further insights into the dataset can be gleaned from Figure 8 on the relationship between (a) word count and (b) sentiment score vs. text length for the evaluated social media dataset. This figure provides a correlation between the length of the tweets and the sentiment scores, offering a nuanced understanding of how text length might influence sentiment in financial tweets.

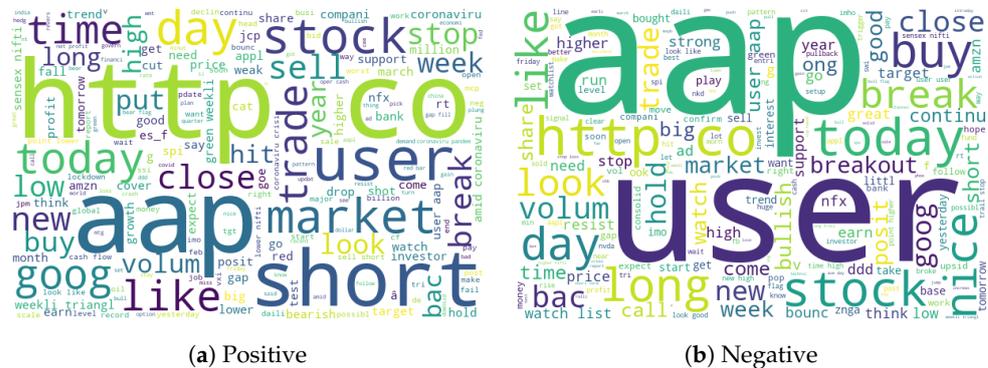


Figure 7. Most frequent terms in TweetFinSent with different sentiment classes.

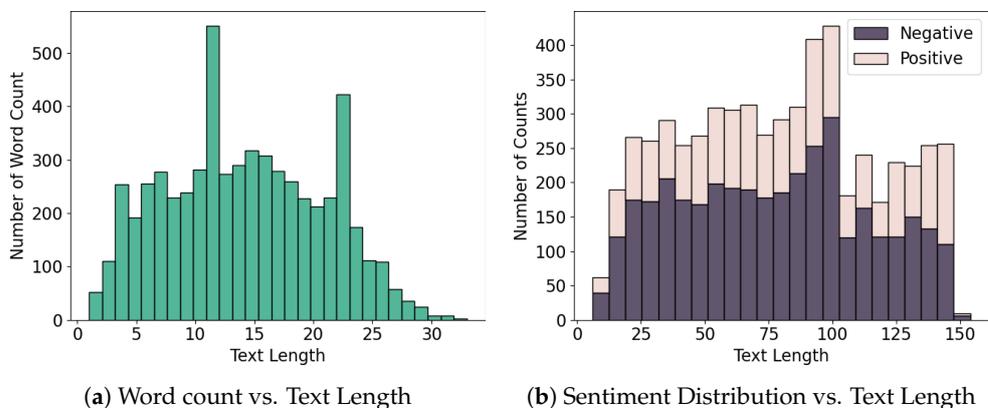


Figure 8. Relationship between (a) word count and (b) sentiment scores vs. text length for the evaluated social media dataset.

4.2. Model Configuration

In our exploration of BERT configurations, we identified key distinctions among BERT-Tiny, BERT-Base, and BERT-Large models. These differences are primarily manifested in

four areas Vaswani et al. (2017): the number of transformer encoder hidden layers, the count of attention heads, the hidden size within feed-forward networks, and the maximum sequence length parameter, which dictates the upper limit of the input vector size. While BERT-Tiny offers a more compact architecture, BERT-Large stands out with its enhanced complexity and capacity, accommodating larger input vectors. For the scope of this article, we have chosen to harness the BERT-Base model, with its corresponding hyper-parameters detailed in Table 3.

Table 3. Hyper-parameters of the fine-tuned financial sentiment analysis BERT model.

Hyperparameter	Value
Attention heads	12
Batch size	8
Epochs	5
Gradient accumulation steps	16
Hidden size	768
Hidden layers	6, 12, 18
Learning rate	0.00003
Maximum sequence length	128

In more depth, the base and the big architecture of BERT can be distinguished. In our study, as detailed in Table 3, we evaluated various BERT model configurations to understand the trade-offs between model complexity and performance. BERT-Tiny, with its 10 M parameters, serves as a lightweight model, while BERT-Large, encompassing 340 M parameters, represents the pinnacle of complexity in our dataset.

4.3. Evaluation Metrics

Using the unknown data as the test dataset, we evaluated the outputs of the training models to gauge the performance of the transformer model. The efficacy of classification is commonly gauged using traditional statistical metrics. One such metric is Precision, which is defined in Equation (11). Here, TP, FP, and FN represent the True Positive, False Positive, and False Negative counts, respectively.

$$Precision = \frac{TP}{TP + FP} \quad (11)$$

Precision provides insight into the model's ability to correctly classify positive instances. A higher precision value indicates that the model is better at distinguishing true positives from false positives.

In addition to Precision, two other crucial metrics for classification are Recall and the F1 Score. Recall, defined in Equation (12), measures the model's capability to identify all relevant instances, or in other words, how many of the actual positives our model captures through labeling them as positive.

$$Recall = \frac{TP}{TP + FN} \quad (12)$$

The F1 Score, defined in Equation (13), is the harmonic mean of Precision and Recall. It provides a single score that balances both the concerns of Precision and Recall in one number. This is particularly useful when the class distribution is imbalanced.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (13)$$

Together, these metrics offer a comprehensive view of the model's classification performance, ensuring that we consider both the identification of positive instances and the avoidance of false alarms.

We also use two additional measures, including the number of parameters (# Params.) and computational complexity (FLOPs), to assess the proposed model's computational effectiveness. Greater memory intensity results from having more parameters, whereas greater computational complexity requires more processing power.

4.4. Results and Analysis

4.4.1. Sentiment Accuracy

The accuracy is the key metric that evaluated the effectiveness for a given sentiment analysis model. In this section, we compare the accuracy of various models in sentiment analysis tasks. These benchmarked models include CNNs [Deriu and Cieliebak \(2016\)](#), LSTM [De Mattei et al. \(2018\)](#), and Multilingual BERT [Magnini et al. \(2020\)](#). To ensure that the comparison is fair, we benchmarked different methods and models in Table 4 based on the same dataset used in this work. The comparison is presented in Table 4. It is evident that our proposed system outperforms the other state-of-the-art models in terms of sentiment accuracy. This superior performance can be attributed to the innovative techniques and methodologies we employed during the model's development. As compared with conventional deep learning models like CNN [Deriu and Cieliebak \(2016\)](#) and LSTM [De Mattei et al. \(2018\)](#), the transformer-based methods show better modeling capabilities for the sequence data. The high accuracy achieved by our system underscores its robustness and reliability in handling sentiment analysis tasks, making it a preferred choice for applications that demand high precision and consistency.

Table 4. Comparison with state-of-the-art algorithms for stock sentiment analysis.

System	F1 Pos	F1 Neg	F1
CNN Deriu and Cieliebak (2016)	0.634	0.706	0.670
LSTM De Mattei et al. (2018)	0.669	0.729	0.699
Multilingual BERT Magnini et al. (2020)	0.723	0.744	0.733
Proposed Design	0.740	0.765	0.752

4.4.2. Case Study

To study the performance difference between different models, we conducted a case study on Tweet data that contain the ticker \$BABA for the Alibaba group. In Table 5, we pick up two representative examples, where our proposed model makes correct predictions, while the rest of three comparing models (CNN [Deriu and Cieliebak \(2016\)](#), LSTM [De Mattei et al. \(2018\)](#), and Multilingual BERT [Magnini et al. \(2020\)](#) in Table 4) make incorrect predictions. For the first example, the correct sentiment label is neutral, but the comparing models incorrectly predict it as positive. This is mainly due to the "lol" keyword in the Tweet, which may cause misinformation to the models. For the second example, we show a more complicated Tweet with multiple tickers. Other models regard it as a negative Tweet because of the "25% down on btc" sentence. However, the actual sentiment for this example is positive. These two examples demonstrate that our proposed model, based on a sparse attention mechanism, has better capabilities to identify the hidden sentiment for the given Tweet because the long-range attention is more helpful to capture the dependency between contents.

Table 5. Two examples to show the potential effects of long-range attention.

Tweet Content	Proposed Prediction	Others' Prediction
Here is my entry \$Baba \$123.63, lol.	Neutral	Positive
update for today: — what a day! what a week! — 25% down on #btc — traded \$bp, \$sq, \$tecs — added \$aapl, \$amzn, \$baba	Positive	Negative

4.4.3. Computational Complexity and Efficiency

We also studied the performance differences of three variants of the BERT model, including BERT-Tiny, BERT-Base, and Bert-Large. This was to analyze the impact of model size on classification precision and then help us select the most cost-effective model. The experiment results are summarized in Table 6. We first calculated the required number of model parameters and computational complexity for three BERT models. BERT-Large has the most model 197 M parameters and a 120 G computational complexity. Meanwhile, BERT-Large also generates the highest precision. It delivers a 0.0794 higher F1 score over the BERT-Tiny model at the expense of more memory and computation consumptions. Here, we regard the BERT-Base model as the most cost-effective model since it balances between complexity and precision well.

Table 6. Performance comparison for different BERT model variants.

Model Name	# Params.	Complexity (FLOPs)	F1 Score
BERT-Tiny	50 M	15 G	0.6731
BERT-Base	110 M	55 G	0.7098
BERT-Large	340 M	120 G	0.7525

Interestingly, despite its intricate architecture, BERT-Large only slightly lags behind BERT-Base in terms of latency, clocking in at 15.8 ms compared with 12.5 ms. This suggests that advanced optimization techniques might have been employed to mitigate the expected latency surge. As computational complexity rises, we observe a corresponding uptick in performance. However, this enhancement comes with the caveat of increased computational demands and potential latency. Such insights underscore the importance of judicious model selection, ensuring a balance between resource constraints and desired performance, especially in real-world applications.

We also study the runtime and computation efficiency for various stock sentiment models in Table 7. The compared baselines include CNN [Deriu and Cieliebak \(2016\)](#), LSTM [De Mattei et al. \(2018\)](#), and the BERT-Large model. We record and calculate the models' parameters that indicate the memory consumption while running the algorithm. The average latency and complexity are also measured to validate runtime and computation efficiency. LSTM has the shortest latency since it requires much less computation complexity as compared with other counterparts. The CNN model with the medium parameter complexity and latency has higher complexity when compared with our proposed algorithm. This is due to the usage of expensive convolution operations. Our proposed model with sparse attention patterns, which has 197M parameters, achieves an average latency of 10.3 ms and a computational complexity of 3.2 GFLOPS. The adopted sparse attention mechanism saves the redundant computation as well as data movement. As a result, our design yields even higher memory and runtime efficiency as compared with the BERT-Large model.

Table 7. Runtime and computation efficiency comparison for various stock sentiment models.

Method	# Params.	Avg. Latency (ms)	Avg. Complexity (GFLOPS)
LSTM	-	2.4	0.3
CNN	28 M	8.2	3.6
BERT-Large	340 M	15.8	4.8
This Work	197 M	10.3	3.2

5. Conclusions

5.1. Summary and Contribution of This Work

The stock market is a crucial component of a nation's economy, and its success or failure has a direct impact on economic growth. There is uncertainty regarding investment outcomes. Social media sentiment has been found to be consistently linked to the stock market, making the analysis of stock sentiment valuable for practical and research purposes. In recent times, there has been a focus on analyzing investor sentiment through social media, particularly among young and inexperienced investors. Numerous studies have explored the use of Twitter sentiment to forecast stock market trends. However, efficient stock sentiment analysis suffers from two challenges: Firstly, there is a mismatch between conventional sentiment analysis and stock sentiment analysis. While traditional sentiment analysis focuses on emotional states, stock sentiment is tied to market dynamics and reflects expectations of stock price movements. This can lead to disparities between the two sentiments. Secondly, deep learning models, such as transformers, have shown great performance improvements but suffer from high computational complexity. This poses challenges for real-time processing and deployment in resource-constrained environments.

To address these challenges, this paper proposes the use of sparse transformers, which reduce computational overhead while maintaining model efficacy, enabling more sustainable and scalable deep learning applications. The use of BERT for financial sentiment analysis has been found to be very effective, with results that are often better than those of other existing methods. In addition, BERT's ability to understand contextual relationships between words makes it well-suited to accurately analyze the sentiment of financial texts. According to our evaluation results, our proposed model with sparse attention patterns, which has 197 M parameters, achieves an average latency of 10.3 ms and a computational complexity of 3.2 GFLOPS. When compared with other models like CNN, LSTM, and BERT, our model demonstrates a competitive latency, being faster than BERT's 12.5 ms while maintaining a higher computational complexity. This indicates that our model efficiently utilizes its parameters to deliver faster results without compromising on computational demands. The improvements are particularly evident when comparing the latency and complexity metrics, showcasing the efficiency and effectiveness of our proposed sparse attention mechanism. As technology continues to evolve and improve, the potential of BERT for financial sentiment analysis will increase. Using BERT to analyze financial texts can provide valuable information and help inform better decision making in the financial sector.

5.2. Limitations and Future Work

While this study primarily centers on leveraging sentiment analysis through BERT and sparse transformer models for stock market predictions, we acknowledge the influence of additional variables such as the behavior of large investors and the role of specialized media. Large investors, such as funds and financial institutions, exert a substantial impact on stock prices that may not be captured on social media platforms. Similarly, specialized financial news outlets and analyst reports can shape public opinion and investor behavior. Looking forward, our research aims to account for these variables by integrating multi-source data, including trading data from large investors and professional news reports, to enhance the model's predictive accuracy. Additionally, we consider incorporating time-series data featuring key milestones or inflection points to offer a more holistic forecasting model.

Author Contributions: Conceptualization and methodology: S.W. and F.G.; software implementation and validation: F.G.; formal analysis and investigation: F.G.; manuscript drafting and visualization, S.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset used in this work is at: <https://github.com/jpmcair/tweetfinsent>, accessed on 4 October 2023.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Abraham, Jethin, Daniel Higdon, John Nelson, and Juan Ibarra. 2018. Cryptocurrency price prediction using tweet volumes and sentiment analysis. *SMU Data Science Review* 1: 1.
- Almatrafi, Omaima, Suheim Parack, and Bravim Chavan. 2015. Application of location-based sentiment analysis using twitter for identifying trends towards indian general elections 2014. Paper presented at the 9th International Conference on Ubiquitous Information Management and Communication, Bali, Indonesia, January 8–10, pp. 1–5.
- Arora, Upasana, Shikhar Verma, Ishu Gupta, and Ashutosh Kumar Singh. 2017. Implementing privacy using modified tree and map technique. Paper presented at the 2017 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA)(Fall), Dehradun, India, September 15–16, pp. 1–5.
- Aziz, Rabia Musheer, Mohammed Farhan Baluch, Sarthak Patel, and Abdul Hamid Ganie. 2022. Lgbm: A machine learning approach for ethereum fraud detection. *International Journal of Information Technology* 14: 3321–31. [CrossRef]
- Child, Rewon, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *arXiv*, arXiv:1904.10509.
- De Mattei, Lorenzo, Andrea Cimino, and Felice Dell’Orletta. 2018. Multi-task learning in deep neural network for sentiment polarity and irony classification. Paper presented at the NL4AI@ AI* IA, Trento, Italy, November 20–23, pp. 76–82.
- Deriu, Jan Milan, and Mark Cieliebak. 2016. Sentiment analysis using convolutional neural networks with multi-task training and distant supervision on italian tweets. Paper presented at the Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, Napoli, Italy, December 5–7.
- Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv*, arXiv:1810.04805.
- Dey, Rahul, and Fathi M Salem. 2017. Gate-variants of gated recurrent unit (gru) neural networks. Paper presented at the 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), Boston, MA, USA, August 6–9, pp. 1597–600.
- Dong, Linhao, Shuang Xu, and Bo Xu. 2018. Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition. Paper presented at the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, April 15–20, pp. 5884–5888.
- Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, and et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv*, arXiv:2010.11929.
- Gandhmal, Dattatray, and Kannan Kumar. 2019. Systematic analysis and review of stock market prediction techniques. *Computer Science Review* 34: 100190. [CrossRef]
- Gupta, Ishu, and Ashutosh Kumar Singh. 2017. A probability based model for data leakage detection using bigraph. Paper presented at the 2017 the 7th International Conference on Communication and Network Security, Tokyo, Japan, November 24–26, pp. 1–5.
- Gupta, Ishu, and Ashutosh Kumar Singh. 2020. Seli: Statistical evaluation based leaker identification stochastic scheme for secure data sharing. *IET Communications* 14: 3607–18. [CrossRef]
- Gupta, Ishu, Tarun Kumar Madan, Sukhman Singh, and Ashutosh Kumar Singh. 2022. Hisa-smfm: Historical and sentiment analysis based stock market forecasting model. *arXiv*, arXiv:2203.08143.
- Hasselgren, Ben, Christos Chrysoulas, Nikolaos Pitropakis, and William J Buchanan. 2022. Using social media & sentiment analysis to make investment decisions. *Future Internet* 15: 5.
- Hendrycks, Dan, and Kevin Gimpel. 2016. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *arXiv*, arXiv:1606.08415.
- Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9: 1735–80. [CrossRef] [PubMed]
- Jiang, Weiwei. 2021. Applications of deep learning in stock market prediction: Recent progress. *Expert Systems with Applications* 184: 115537. [CrossRef]
- Khan, Salman, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. 2022. Transformers in vision: A survey. *ACM Computing Surveys (CSUR)* 54: 1–41. [CrossRef]
- Lin, Tianyang, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. 2022. A survey of transformers. *arXiv*, arXiv:2106.04554.
- Lin, Zhouhan, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *arXiv*, arXiv:1703.03130.
- Liu, Bing. 2012. *Sentiment Analysis and Opinion Mining*. Cham: Springer Nature Switzerland AG.

- Liu, Ze, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. Paper presented at the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, October 11–17, pp. 10012–22.
- Magnini, Bernardo, Alberto Lavelli, and Simone Magnolini. 2020. Comparing machine learning and deep learning approaches on nlp tasks for the italian language. Paper presented at the 12th Language Resources and Evaluation Conference, Marseille, France, May 11–16, pp. 2110–19.
- Man, Xiliu, Tong Luo, and Jianwu Lin. 2019. Financial sentiment analysis (fsa): A survey. Paper presented at the 2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS), Taipei, Taiwan, May 6–9, pp. 617–22.
- Medsker, Larry, and Lakhmi Jain. 2001. Recurrent neural networks. *Design and Applications* 5: 64–67.
- Mishev, Kostadin, Ana Gjorgjevikj, Irena Vodenska, Lubomir T. Chitkushev, and Dimitar Trajanov. 2020. Evaluation of sentiment analysis in finance: From lexicons to transformers. *IEEE Access* 8: 131662–82. [CrossRef]
- Nabipour, Mojtaba, Pooyan Nayyeri, Hamed Jabani, Amir Mosavi, and Ely Salwana. 2020. Deep learning for stock market prediction. *Entropy* 22: 840. [CrossRef]
- Neuenschwander, Bruna, Adriano C. M. Pereira, Wagner Meira Jr., and Denilson Barbosa. 2014. Sentiment analysis for streams of web data: A case study of brazilian financial markets. Paper presented at the 20th Brazilian Symposium on Multimedia and the Web, João Pessoa, Brazil, November 18–21, pp. 167–70.
- Pang, Xiongwen, Yanqiang Zhou, Pan Wang, Weiwei Lin, and Victor Chang. 2020. An innovative neural network approach for stock market prediction. *The Journal of Supercomputing* 76: 2098–18. [CrossRef]
- Pathak, Ajeet Ram, Manjusha Pandey, and Siddharth Rautaray. 2021. Topic-level sentiment analysis of social media data using deep learning. *Applied Soft Computing* 108: 107440. [CrossRef]
- Pei, Yulong, Amarachi Mbakwe, Akshat Gupta, Salwa Alamir, Hanxuan Lin, Xiaomo Liu, and Sameena Shah. 2022. Tweetfinsent: A dataset of stock sentiments on twitter. Paper presented at the Fourth Workshop on Financial Technology and Natural Language Processing (FinNLP), Vienna, Austria, July 23–25, pp. 37–47.
- Pota, Marco, Mirko Ventura, Rosario Catelli, and Massimo Esposito. 2020. An effective bert-based pipeline for twitter sentiment analysis: A case study in italian. *Sensors* 21: 133. [CrossRef]
- Qin, Yao, Dongjin Song, Haifeng Chen, Wei Cheng, Guofei Jiang, and Garrison Cottrell. 2017. A dual-stage attention-based recurrent neural network for time series prediction. *arXiv*, arXiv:1704.02971.
- Ruan, Yefeng, Arjan Durrezi, and Lina Alfantoukh. 2018. Using twitter trust network for stock market analysis. *Knowledge-Based Systems* 145: 207–18. [CrossRef]
- Sanboon, Thaloengpattarakoon, Kamol Keatruangkamala, and Saichon Jaiyen. 2019. A deep learning model for predicting buy and sell recommendations in stock exchange of thailand using long short-term memory. Paper presented at the 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), Singapore, February 23–25, pp. 757–60.
- Saxena, Deepika, Ishu Gupta, Jitendra Kumar, Ashutosh Kumar Singh, and Xiaoqing Wen. 2021. A secure and multiobjective virtual machine placement framework for cloud data center. *IEEE Systems Journal* 16: 3163–74. [CrossRef]
- Singh, Ashutosh Kumar, and Ishu Gupta. 2020. Online information leaker identification scheme for secure data sharing. *Multimedia Tools and Applications* 79: 31165–82. [CrossRef]
- Sohangir, Sahar, Nicholas Petty, and Dingding Wang. 2018. Financial sentiment lexicon analysis. Paper presented at the 2018 IEEE 12th International Conference on Semantic Computing (ICSC), Laguna Hills, CA, USA, January 31–February 2, pp. 286–89.
- Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. Paper presented at the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, December 4–9.
- Wang, Jinjiang, Jianxing Yan, Chen Li, Robert X Gao, and Rui Zhao. 2019. Deep heterogeneous gru model for predictive analytics in smart manufacturing: Application to tool wear prediction. *Computers in Industry* 111: 1–14. [CrossRef]
- Wang, Jin, Liang-Chih Yu, K. Robert Lai, and Xuejie Zhang. 2016. Dimensional sentiment analysis using a regional cnn-lstm model. Paper presented at the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Berlin, Germany, August 7–12, pp. 225–30.
- Yang, Linyi, Tin Lok James Ng, Barry Smyth, and Riu Hai Dong. 2020. Htm: Hierarchical transformer-based multi-task learning for volatility prediction. Paper presented at the Web Conference 2020, Taipei, Taiwan, April 20–24, pp. 441–51.
- Zhao, Bo, Yongji He, Chunfeng Yuan, and Yihua Huang. 2016. Stock market prediction exploiting microblog sentiment analysis. Paper presented at the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, July 24–29, pp. 4482–88.
- Zhao, Rui, Ruqiang Yan, Jinjiang Wang, and Kezhi Mao. 2017. Learning to monitor machine health with convolutional bi-directional lstm networks. *Sensors* 17: 273. [CrossRef] [PubMed]
- Zhou, Yuqing, and Wei Xue. 2018. Review of tool condition monitoring methods in milling processes. *The International Journal of Advanced Manufacturing Technology* 96: 2509–23. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.