

Article

Temporal Convolutional Networks and BERT-Based Multi-Label Emotion Analysis for Financial Forecasting

Charalampos M. Liapis *  and Sotiris Kotsiantis * 

Department of Mathematics, University of Patras, 26504 Patras, Greece

* Correspondence: c.liapis@upnet.gr (C.M.L.); sotos@math.upatras.gr (S.K.)

Abstract: The use of deep learning in conjunction with models that extract emotion-related information from texts to predict financial time series is based on the assumption that what is said about a stock is correlated with the way that stock fluctuates. Given the above, in this work, a multivariate forecasting methodology incorporating temporal convolutional networks in combination with a BERT-based multi-label emotion classification procedure and correlation feature selection is proposed. The results from an extensive set of experiments, which included predictions of three different time frames and various multivariate ensemble schemes that capture 28 different types of emotion-related information, are presented. It is shown that the proposed methodology exhibits universal predominance regarding aggregate performance over six different metrics, outperforming all the compared schemes, including a multitude of individual and ensemble methods, both in terms of aggregate average scores and Friedman rankings. Moreover, the results strongly indicate that the use of emotion-related features has beneficial effects on the derived forecasts.

Keywords: financial forecasting; multivariate forecasting; temporal convolutional networks; BERT; emotion classification; multi-label classification; feature selection; ensembles; sentiment analysis



Citation: Liapis, C.M.; Kotsiantis, S. Temporal Convolutional Networks and BERT-Based Multi-Label Emotion Analysis for Financial Forecasting. *Information* **2023**, *14*, 596. <https://doi.org/10.3390/info14110596>

Academic Editor: Xin Ning

Received: 29 September 2023

Revised: 25 October 2023

Accepted: 1 November 2023

Published: 3 November 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/>).

1. Introduction

In terms of machine learning, time series forecasting, in a broad sense, involves the training and utilization of a model to predict the future values of variables that describe a phenomenon based on historical data. Time series are mathematical formalizations that include sequential and time-dependent observations. In this work, such sequential and time-dependent data are represented by stock market closing prices.

In recent times, financial forecasting seems to be a highly relevant field of research that has the potential to play a critical role in managing risks, making informed decisions, and achieving financial goals, given the increasingly complex and dynamic landscape that constitutes contemporary economies. However, besides this, the financial setting constitutes quite an interesting phenomenon in and of itself from both a modeling and a psychological point of view, inter alia, given the assumption that the various outcomes at their core are informed by social attitudes that are expressed linguistically. Nevertheless, it seems as if this stands out as a characteristic depiction of human agency capable of exemplifying an inherently paranoid aspect of humanity.

In this ever-evolving financial landscape, it is now common knowledge that harnessing the power of deep learning and relevant emotion-related information constitutes a promising path for investigating improvements regarding forecasting endeavors. Furthermore, the emotions and sentiment polarities extracted from posts on social media can be a significantly useful tool for modeling general behavior toward financial markets. Here, the framework to be presented integrates the above two rationales, with, on the one hand, the additional introduction of a thorough benchmarking of a multitude of deep learning and ensemble methods and, on the other, a space of emotion-related features that does not simply contain general sentiment polarities but integrates an extensive description

through a subtle multi-label classification system of 28 distinct emotions, representing a variety of emotional attitudes towards each stock examined. Building on our previous work, we first compare the best-performing algorithmic schemes of a benchmarked set of 30 state-of-the-art methods and then present a method for incorporating the aforementioned fine-grained emotion feature exploitation together with a feature selection procedure.

Thus, this work is about comparing and benchmarking a number of state-of-the-art methods that incorporate both classical sentiment analysis and multi-label emotion classification in the task of financial forecasting, as well as proposing a derived methodology that exploits temporal convolutional networks (TCNs) and emotion analysis to improve medium-term stock market closing price forecasts. Specifically, regarding the latter, the proposed scheme consists of the following distinct modules: TCNs, feature selection, sentiment analysis, and a BERT-based [1] multi-label emotion classifier, all under a multivariate-averaging ensemble scheme. Convolutional networks are a class of neural networks specializing in learning hierarchical features from structured data by applying convolutional operations. Temporal convolutional networks (TCNs) are a type of neural network architecture designed for processing sequential data, such as time series. These networks focus on capturing temporal dependencies and patterns by leveraging convolutional operations with respect to the temporal dimension, allowing them to analyze and learn from the sequential nature of such data. BERT, standing for bidirectional encoder representations from transformers, is a state-of-the-art natural language processing (NLP) mode, a transformer-based neural network architecture specifically designed for language tasks. Hence, the selected day-to-day sentiment and emotion scores extracted from related tweets are incorporated into the feature space and used in a multivariate setting to predict the closing prices of 15 stocks. The investigation builds on the results presented in [2] in the sense that the aforementioned work, which works within the same framework as the present one regarding base learners and data, enables us to reject a fairly large number of methods, keeping only those that exhibit good behavior. Thus, the experimental framework starts with five top-performing methods and includes the investigation of a number of possible weighted ensemble forecasting procedures. It will be shown that the proposed methodology prevails with respect to every evaluation metric, exhibiting the best overall performance in each of the valuations. Furthermore, we will see that the use of multivariate inputs containing specific emotional features always improves the derived predictions.

Given the above, in summary, this work can be seen as both the end piece of a rather extensive comparative study and as presenting a concrete, specific methodology. A novel methodology for improved medium-term stock market closing price forecasts that integrates TCNs and the emotion-related features extracted from tweets is introduced. The method presents the final outcome of a thorough benchmarking and comparison process. The latter includes the investigation of a large number of possible ensemble predictors that incorporate a variety of emotion-related multivariate inputs under a plethora of weighted combinatory schemes. It is shown that the presented methodology clearly outperforms every base or ensemble scheme. Additionally, the incorporation of deep learning as well as fine-grained specific emotion polarities under our simple averaging combinatory scheme not only stands out as good applied practice but has the potential to draw a path towards the creation of semantically rich and diverse feature spaces that represent subtle emotion polarities that can potentially be used in various modeling tasks. We show, through various charts and empirical performance validation, that the incorporation of feature selection, sentiment analysis, and multi-label emotion classification leads to significant prediction improvements. The results demonstrate that the inclusion of multivariate inputs containing specific emotional features consistently leads to improvements in accuracy. Hence, the creation of fine-grained, specific, and distinct emotion polarities stands out as a largely beneficial practice that, quite promisingly, could be utilized in various prediction tasks.

Concluding this introduction, the structure of the present work is as follows: First, some related works are listed. Then, in Section 3, the experimental and evaluation proce-

dures are given. Section 4 contains elements of the proposed methodology. Finally, the results and a summary assessment follow.

2. Related Work

In this section, indicative works from the existing literature are briefly introduced. As already mentioned, emotion and sentiment-related representations have been the center of focus in a multitude of diverse research endeavors.

Starting with some indicative works regarding the latest trend in general sentiment and opinion mining, a novel labeling strategy, together with an effective model for structured sentiment analysis consisting of graph attention networks and an adaptive multi-label classifier, is introduced in [3]. This approach demonstrates significant performance improvements over prior state-of-the-art models on five benchmark datasets across multiple languages. In [4], a novel multiplex cascade framework for unified aspect-based sentiment analysis (ABSA) that maintains the interaction existing between the various ABSA subtasks is introduced. By hierarchically modeling the subtasks and integrating syntax-aware information, the proposed Syntax-aware Multiplex framework improves ABSA results across 28 subtasks with substantial gains. A method that exploits documents' latent target-opinion distribution and then leverages fine-grained sentiment analysis principles to enhance document-level sentiment classification is proposed in [5]. The method, consisting of a variational and a classification part, introduces a hierarchical approach with a variational autoencoder and a transformer-based module, respectively, effectively capturing latent fine-grained target and prior opinion information and achieving state-of-the-art performance on various benchmark datasets. Moreover, in [6], a Three-hop Reasoning chain-of-thought (CoT) framework is presented for implicit sentiment analysis (ISA), with both inspired by and targeting human-like reasoning processes. The method achieves significant improvements, surpassing the state-of-the-art in both supervised and zero-shot setups.

Concerning research on sentiment classifications and economic data, in [7], FinBERT, a domain-specific language model for natural language processing regarding financial-related tasks, is presented. FinBERT is a state-of-the-art BERT-based language model fine-tuned on financial textual datasets. Such fine-tuning procedures are now common practice, and, actually, the model is also used within the experimental framework of the present work. Furthermore, in [8], a deep learning architecture that leverages managerial emotion representations formed by speech recognition using FinBERT-based sentiment analysis applied to earnings conference call transcripts is proposed. In [9], text-based emotion recognition with a focus on deep learning techniques is explored. The work extends existing methods by addressing class imbalances and introducing transfer learning-based strategies, offering comprehensive benchmarking of text-based emotion recognition methods and demonstrating the superiority of deep learning approaches across various datasets. Sentiment polarities generated from tweets are used in [10] to investigate the impact of Twitter on stock market decisions. For this, a methodology that utilizes financial-based sentiment analysis on relevant and influential Twitter accounts is employed. The study contains comparisons regarding the investigation of correlations between tweets and stock market behavior during the H1N1 and COVID-19 periods. A company-specific model for sentiment analysis in financial data is proposed in [11]. The model's architecture is composed of neural networks and aspires to generally detect trend variations in stock prices, transforming pretrained word embeddings that have no financial specificity into embeddings that capture important domain-specific characteristics. A knowledge base extends the financial-related embedding space by enriching the vocabulary. The topic has been investigated relatively extensively with earlier known architectures as well, where various neural network models, such as long short-term memory (LSTM) and convolutional neural networks (CNNs), are employed to model stock market opinions [12]. A hierarchical data structure and a two-step model are used in [13] for financial-related aspect classes and corresponding sentiment polarities in sentence prediction, whereas in [14], a novel semantic and syntactic-enhanced neural model is introduced to improve target

sentiment representation regarding bullish or bearish sentiments in the financial domain by incorporating dependency graphs and context words.

Regarding multi-label emotion analysis-related works, in [15], an emotion prediction framework consisting of a prompt-based generative multi-label emotion prediction model is presented, demonstrating competitive results after being tested on the two datasets. In [16], a novel model called SpanEmo that treats multi-label emotion classification as a span prediction task is introduced. The introduced strategy, in broad terms, aims to present an enhanced model with the capacity to represent the underlying existing associations between emotions as labels and sentences. A topic-enhanced capsule network for multi-label emotion detection consisting of a variational autoencoder that learns latent topic information and a capsule module capturing the corresponding emotion features is introduced in [17]. The proposed method significantly outperforms a variety of previous methods and strong baseline schemes on two benchmark datasets, demonstrating top-level performance. Additionally, a latent emotion memory network (LEM) for multi-label emotion classification that can learn latent emotion distribution without relying on external sources and can efficiently incorporate it into the classification network is presented in [18]. The results from experiments on two benchmark datasets indicate that the suggested model demonstrates state-of-the-art behavior, outperforming well-established baselines.

Moving on to papers regarding financial forecasting and sentiment analysis, a comprehensive literature-based study on investor sentiment analytics and machine learning applied to predict stock prices is presented in [19]. Additionally, review-wise, the work in [20] presents a critical literature review regarding text mining and sentiment analysis for stock market prediction, focusing on stock markets. A systematic review examining works based on using machine learning and text mining techniques applied to news data to predict the stock market is presented in [21]. The study identifies gaps and barriers in the field while highlighting the increasing use of artificial neural networks and advanced natural language processing methods and opportunities for future research. In [22], a sentiment-annotated dataset containing textual data related to Bitcoin taken from Reddit is proposed. The dataset is used to evaluate relevant crypto price change forecasts by incorporating various architectures, such as recurrent neural networks (RNNs) and transformers. A work based on using stock-specific news synopses, together with extracted sentiment features to predict stock prices, is presented in [23]. The study aspires to present a forecasting framework that positively exploits various stock-related aspects, such as discretized stock price movements, valence sentiment analysis, and sentiment polarities. Moreover, ref. [24] introduces weak supervision in financial forecasting, investigating the incorporation of both sentiment analysis (performed on news and social media data) and machine learning methods to the task of cryptocurrency price prediction. Inter alia, the paper employs a BERT classifier to extract sentiment scores, which are then included in a model for predicting daily returns. In [25], again, various past stock-price values and a pretrained BERT model are utilized in combination under a predictive scheme that employs LSTM neural networks. The setup introduces features that contain sentiment scores extracted from news and a relevant online forum, as well as other stock-related historical information such as the opening, closing, highest, and lowest prices. In [26], a new dataset for stock market emotion detection is presented. The set contains data consisting of 12 fine-grained emotion classes concerning investor emotion. The impact of investor emotions extracted is investigated within a time series forecasting setup.

Regarding the architecture that is the core of the methodology proposed here, temporal convolutional networks (TCNs) are used in various forecasting endeavors. In [27], a temporal convolution network model is proposed for multivariable time series prediction, with the authors presenting results that suggest prediction accuracy improvements. The model is employed in a sequence-to-sequence layout applied to nonperiodic datasets. Multichannel residual blocks in parallel with a deep convolution neural network-based asymmetric structure are presented. Moreover, regarding short-term energy load forecasting, a model based on a temporal convolutional network and a light gradient boosting

machine (LightGBM) is proposed in [28]. The TCN is used over the input features to model the underlying information and long-term temporal dependencies. Then, a LightGBM is utilized to predict energy loads. In [29], state-of-the-art temporal convolutional networks are utilized to forecast weather, outperforming LSTMs and various other machine learning architectures. Lastly, in [30], an investor attention factor is employed by combining various trading information as the input and utilizing a temporal convolutional network to predict volatility under high-frequency financial data, and a novel technique combining temporal convolutional networks and recurrent neural networks (RNNs) for greenhouse crop yield prediction is presented in [31].

Closing this literature review, it is rather obvious that the above indicative listing of relevant works does not exhaust the scope of even a small presentation. Therefore, the reader is urged to further follow the relevant literature.

3. Experimental & Evaluation Framework

The central problem of this work is the modeling of specific financial-related time series containing stock market closing prices in order to predict their future fluctuations. The way this task is treated here is as a regression problem.

A time series forecasting task can be formally described as follows: Given a set of time series observations $\mathcal{X} = \{x_1, x_2, \dots, x_t, \dots, x_T\}$, where x_t is the observation at time t , and a set of timestamps $\mathcal{T} = \{t_1, t_2, \dots, t_T\}$, the goal is to build a forecasting model, F , that can predict its future fluctuations $\hat{x}_{T+1}, \hat{x}_{T+2}, \dots, \hat{x}_{T+n}$, where $n \in \mathcal{Z}^+$. This forecasting model, F , can be expressed as:

$$\hat{x}_{t+h} = F(\mathcal{X}, t, h) \quad (1)$$

where \hat{x}_{t+h} represents the prediction at time $t + h$, \mathcal{X} is the historical time series data up to time t , and h is the forecast horizon, that is, the number of future time steps. The objective here is to train and evaluate a model (F) in order to be able to extract predictions that minimize the differences between \hat{x}_{t+h} and the actual observed values x_{t+h} for various h .

As was already mentioned, the present investigation has its starting point in previously drawn conclusions in terms of creating a set of well-performing methods to test as base learners. Specifically, in [2], from a comparison of the 30 state-of-the-art methods for time series forecasting, as depicted in Table A1, a multivariate temporal convolutional network-based method exploiting sentiment analysis was proposed for the task of stock market forecasting. In addition, four more methods stood out. Furthermore, in the same work, we saw that, in terms of generality and a prediction time window that becomes wider, the use of sentiment modeling features improved the predictions.

Given the above, in this paper, a multivariate stock market forecasting methodology based on a variation of the aforementioned temporal convolutional network is proposed. The methodology now exploits both sentiment analysis and a multi-label emotion classification scheme based on BERT applied to stock-related data extracted from Twitter. A series of predictions incorporating various emotion-related time series is first produced and then integrated into an average-weighted scheme, the elements of which are obtained after a feature selection process. The results indicate a general dominance of the proposed method in every tested case and in all metrics. The latter resulted from an extensive evaluation of the outputs of a variety of ensemble configurations compared to our proposed methodology.

3.1. Framework Outline

In short, the experimental framework and evaluation process of the compared methodologies were as follows: We started with a set of five algorithms that were to be used as the base learners and that, in our aforementioned previously related research, exhibited the best behavior. Then, with the exploitation of three sentiment analysis techniques, Vader, TextBlob, and FinBERT, as well as a multi-label classifier of 28 different emotions that we created by fine-tuning the BERT model, a multitude of sentiment polarities on the one hand, and emotion-related outputs on the other, were extracted from stock-related Twitter data. Then, for each of these outputs and in order to create the corresponding time series that

would include a daily observation, a daily average was calculated. Then, for each stock, a dataset with 65 features was formed, consisting of the closing prices, the above sentiment and emotion-related features, and their weekly rolling mean versions. Moreover, for each stock and corresponding dataset, all possible combinations consisting of two features were extracted based on the following rule: every combination had to include, as its main component, the time series of the closing price. Thus, we had 64 different multivariate versions to run—together with the univariate one—for each stock and each base learner. The final input dataset used in the training resulted, on the one hand, from its introduction into the feature space consisting of a number of things resulting from the application of the sentiment analysis and emotion classification and on the other, the incorporation of a smoothed version of every feature used; that is, both the closing price time series and the sentiment and emotion-related ones. This process is outlined schematically in Figure 1.

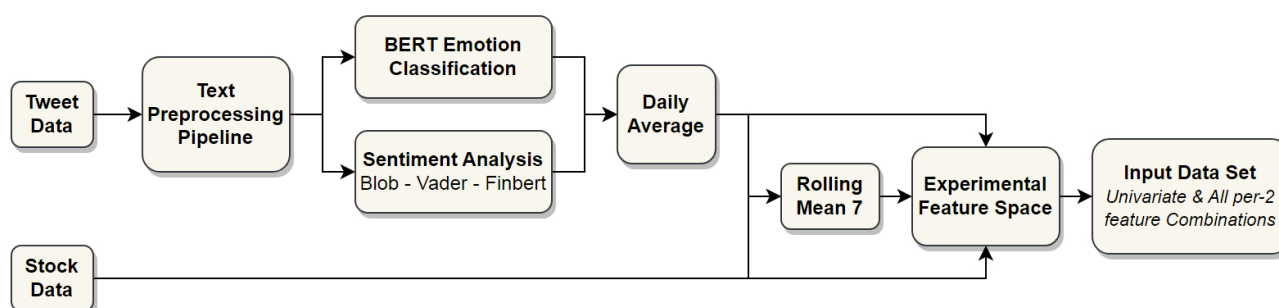


Figure 1. Dataset creation.

In other words, based on the time series of the closing price and given, on the one hand, the three sentiment polarities from the outputs of Vader, TextBlob, and FInBERT and the 28 emotion features of the multi-label BERT classifier, and on the other hand, their smoothed versions resulting from the application of weekly rolling media, 64 features were created that characterized the multivariate layouts. Each of the above characteristics, together with the closing value, constituted an input feature setup.

By using these setups, the first set of experiments was performed for each base learner. Then, according to six evaluation metrics, the best setups regarding each of the five methods investigated were extracted. Next, the possible blended and weighted-average ensemble versions were investigated in the direction of deriving a methodology. Each such ensemble could consist of two to five constituent methods and corresponding input feature setups, each of which was composed of the best-performing multivariate outputs extracted in the previous step. The experimental setting presupposes a first internal benchmarking of the set of 30 state-of-the-art methods presented in Table A1 [2] and continues further investigations of the ensemble methodologies from the best-performing ones. In this context, a new performance ranking is created containing the five base learners to be presented in Table 1 and a number of weighted blended ensemble layouts. The experiments were performed on 80% of the data, reserving the remaining 20% for testing. The following three different time frames were added to the multitude of settings to be tested: single-day, 7-day, and 14-day time shifts.

Thus, all the above methods, together with the proposed methodology to be presented in detail in Section 4, were benchmarked, again, according to the six metrics utilized. Two types of final evaluations were performed: (a) first, the average performance value was calculated regardless of shift and dataset. Here, a ranking based on the average value for each metric was produced. (b) Then, the Friedman rankings [32,33] were calculated, incorporating 15 stock datasets \times 3 shifts per dataset = 45 sets.

Table 1. Best-performing algorithms.

Nº	Abbreviation	Algorithm
1	TCN	Temporal Convolutional Network [34]
2	XCMPPlus	Explainable Convolutional Neural Plus Network [35]
3	LSTM	Long Short-Term Memory Network [36]
4	LSTMPlus	Long Short-Term Memory Plus Network [37]
5	TSTPlus	Time Series Transformer Plus [38]

3.2. Algorithms

We have already seen that the search for possible weighted average ensemble schemes among a series of multivariate methodologies is at the core of this experimental process. Again, given the results in [2], a set with the five best-performing methods was extracted, forming a collection of base learners to experiment with. This set consisted of the methods given in Table 1.

All experiments were carried out using the Python library tsAI [39].

3.3. Data

Two kinds of datasets were used. One involved closing stock prices and the second contained textual data from Twitter.

3.3.1. Stock Data

Regarding the closing prices, Table 2 contains the names of the 15 stock datasets used, along with their corresponding abbreviations. The sets include three years of closing price data for dates ranging from 2 January 2018 to 24 December 2020. Each time series in the set consists of a single daily observation representing the closing price.

Table 2. Stock datasets.

Nº	Dataset	Stocks
1	AAL	American Airlines Group
2	AMD	Advanced Micro Devices
3	AUY	Yamana Gold Inc.
4	BABA	Alibaba Group
5	BAC	Bank of America Corp.
6	ET	Energy Transfer L.P.
7	GE	General Electric
8	GM	General Motors
9	INTC	Intel Corporation
10	MRO	Marathon Oil Corp.
11	MSFT	Microsoft
12	OXY	Occidental Petroleum Corp.
13	RYCEY	Rolls-Royce Holdings
14	SQ	Square
15	VZ	Verizon Communications

3.3.2. Twitter Data

Concerning the tweets, the dataset contains various posts revolving around the investigated companies. The underlying assumption for incorporating such data in financial forecasting is that what is said about these companies can reflect a correlation with their respective future closing prices.

The raw textual dataset has already been extracted from Twitter for the needs of the investigation presented in [40]. There, quite a large number of stock-related tweets written exclusively in English were collected and grouped daily. These Twitter posts consisted

of textual data containing day-to-day views or attitudes towards stocks of interest, that is, tweets that were directly or indirectly linked with the corresponding stock examined. Table 3 depicts some corresponding statistics.

Table 3. Tweet dataset: average statistics.

Nº	Statistic	Value
1	Average number of tweets	15,497
2	Average tweets per day	15
3	Average minimum tweets per day	2
4	Average maximum tweets per day	90
5	Average total tokens per day	496,739
6	Average vocabulary per day (unique tokens)	52,004

Here, after the above procedure of collecting a sufficiently large number of such stock-related tweets, a preprocessing pipeline was applied. Before the cleaning step, the aforementioned initial set was thoroughly inspected in order to retain only the strictly relevant references so as to ensure that the predictive sentiment modeling was built on the basis of as many correlative associations as possible. The preprocessing steps are schematically presented in Figure 2.

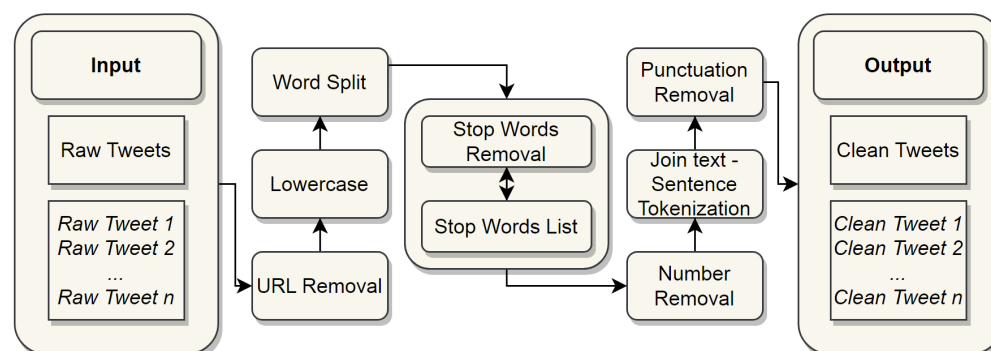


Figure 2. Text preprocessing pipeline.

3.4. Sentiment Analysis and Multi-Label Emotion Classification

Regarding the extraction of insight relative to the emotion and sentiment polarities, two classes of methods were incorporated: one that is relative to classic sentiment analysis and the other containing an emotion classifier that outputs the real-value polarities of a wide range of discrete emotions. Sentiment analysis was applied to the data from Twitter with the incorporation of the following methods: TextBlob [41], Vader sentiment analysis tool [42], and FinBERT [7]. The multi-label, bert-based emotion classifier was created from scratch.

Multi-label means that the model outputs scores for all the labels included in the dataset used in training. Here, the “go_emotions” [43] dataset was used. The dataset includes 58,000 Reddit comments labeled for 28 emotion categories. Thus, for each sequence of text input to the classifier, a vector was extracted that included 28 values within the interval $[-1, 1]$, each of which represents an emotion. These 28 emotions can be seen in Table 4.

Table 4. Emotion Labels.

Nº	Emo	Nº	Emo	Nº	Emo	Nº	Emo
1	admiration	8	curiosity	15	fear	22	pride
2	amusement	9	desire	16	gratitude	23	realization
3	anger	10	disappointment	17	grief	24	relief
4	annoyance	11	disapproval	18	joy	25	remorse
5	approval	12	disgust	19	love	26	sadness
6	caring	13	embarrassment	20	nervousness	27	surprise
7	confusion	14	excitement	21	optimism	28	neutral

The multi-label classifier used was realized by incorporating the uncased version of Squeezebert [44]—that is, "a pretrained model for the English language using a masked language modeling (MLM) and sentence order prediction (SOP) objective"—in a layout that utilizes a simple PyTorch linear transformation layer for the classification output.

Finally, given the respected outputs and in terms of presenting emotion-related statistics for each dataset, it is important to note that their sheer number makes it impossible to provide an exhaustive display. In order to address this, we have included a general correlation heatmap that visualizes the correlations among the average values of the sentiment time series for each stock. Figure 3 depicts the aforementioned linear statistical relations. There, one can easily observe the general absence of correlations between the mean emotional responses to each stock.

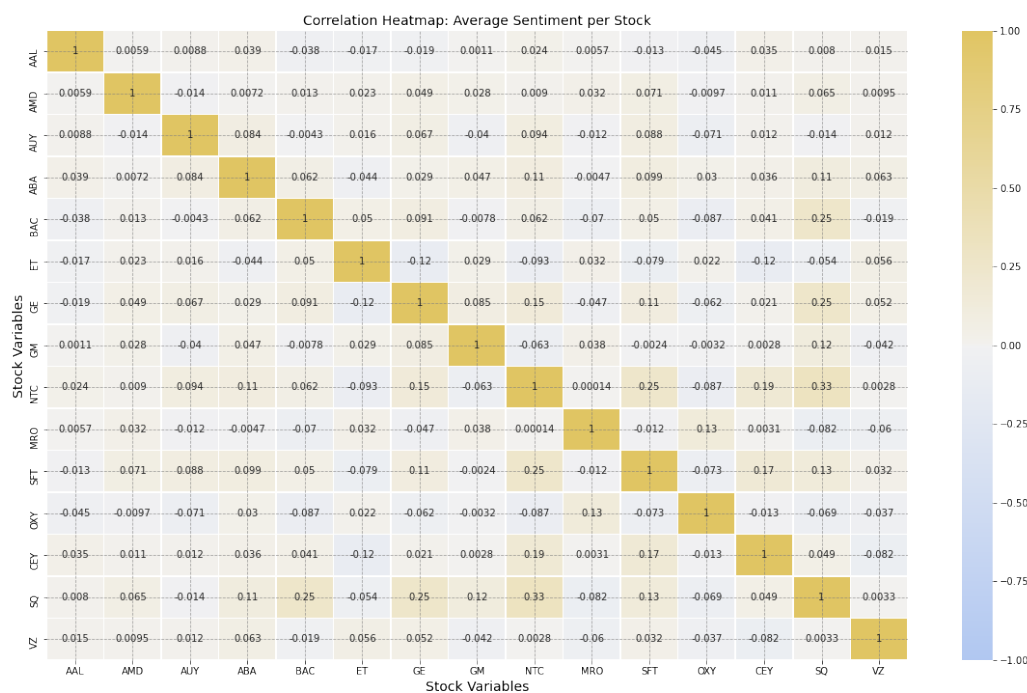


Figure 3. Correlation heatmap: average sentiment per stock.

3.5. Metrics

Lastly, the results were generated based on six widely accepted metrics, each offering distinct insights into the performance of the methods employed. The metrics used are the mean absolute error (MAE), the mean absolute percentage error (MAPE), the mean squared error (MSE), the root mean squared error (RMSE), the root mean squared logarithmic error (RMSLE), and, lastly, the coefficient of determination R2.

4. Proposed Methodology

Here, we arrive at the description of the proposed methodology. First of all, Figure 4 visually depicts a summary of all the steps up to the final prediction.

As one can observe in the illustration, the methodology proposed contains a number of steps. Thus, the presentation that follows is grouped into three more abstract phases. In the first one, the creation and selection of a number of sentiment polarities and emotion score features are undertaken. These will form a multivariate input dataset containing selected features. Here, the process is similar to the one depicted in Figure 1, with one exception: an additional step that incorporates a correlation feature selection procedure. The second phase of the methodology concerns the TCN. Here, every multivariate setup from the previous step will output a TCN prediction. Then, in the last phase, a simple averaging scheme will be imposed on the TCN outputs, forming the final prediction. In other words, it is an ensemble of various multivariate versions of the TCN model that exploit emotion classification. A more detailed description follows below.

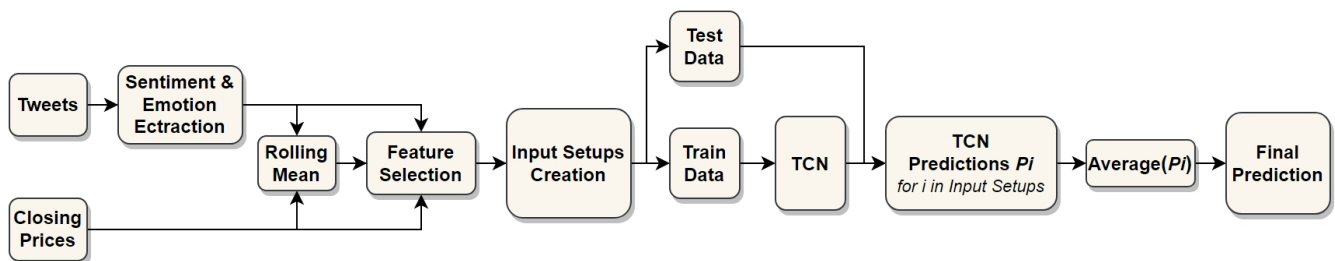


Figure 4. Proposed methodology.

4.1. Emotion Classification and Feature Selection

Looking back at the procedure discussed in Section 3.1, first, the textual data, having undergone the preprocessing presented in Figure 2, are passed through the sentiment and emotion classification modules. From there, three sentiment polarity time series are extracted by the TextBlob, Vader, and FinBERT methods, respectively, as well as another 28 distinct emotion time series concerning each one of the labels shown in Table 4. A simple, 7-day rolling mean is applied, producing smoothed duplicates of all the time series incorporated. All such features, closing prices, and emotion values, smoothed and unsmoothed, are put within the scope of a correlation feature selection procedure that is implemented by exploiting the SelectKBest class from the scikit-learn machine library [45]. Then, we apply a threshold that removes the least correlated features, and the final feature space is ready for training. This threshold is the mean of the feature correlation scores. Finally, all feature combinations consisting of the stock closing price as a fixed input and one of the additional features produced in the previous steps are formed. These will be the input data setups passed through the TCN.

4.2. Temporal Convolutional Network Predictions

For the individual predictions, a variant of the temporal convolutional network from [34] was used. According to the authors, TCNs stand out for their causal convolutions, preventing future-to-past information leakage, and their capacity to map input sequences of any length to corresponding output sequences is akin to recurrent neural networks (RNNs). In the paper [34], TCNs are introduced to tackle sequential data prediction work cases. A network that is able to model sequences is described as a function $f : X^T \rightarrow Y^T$ such that

$$\hat{y}_0, \dots, \hat{y}_T = f(x_0, \dots, x_T) \quad (2)$$

where x_0, \dots, x_T are the inputs and $\hat{y}_0, \dots, \hat{y}_T$ are the corresponding model predictions at a given time. In training, the goal is to find a model, f , such that

$$\min(L(y_0, \dots, y_T, f(x_0, \dots, x_T))) \tag{3}$$

where L is the expected loss and y_0, \dots, y_T is the actual output. The TCN operates by maintaining an output length equal to that of the input, utilizing a 1D fully convolutional network (FCN) architecture, and incorporating causal convolutions, that is, enforcing the absence of information flow from the future to the past:

$$TCN \equiv FCN + CL \tag{4}$$

where CL refers to causal convolutions. Each hidden layer of the FCN utilized has the same length as the input one. Furthermore, zero-padding of a length equal to $ks - 1$, where ks is the kernel size, is included; this is to ensure the consistency of the subsequent layers with their preceding counterparts in terms of length. Causal convolutions represent a particular type of convolutional operation, wherein the output at a given time, t , is exclusively derived from the convolution with elements originating from the same time, t , and those preceding it in the previous layer.

Here, instead of simple, dilated convolutions, ref. [46,47] are included in the architecture. Let F denote the dilated convolution operation, let s be an element of the input sequence $x \in \mathbb{R}^n$, and let $f : \{0, \dots, k - 1\} \rightarrow \mathbb{R}$ be a filter. Then,

$$F(s) = (x *_d f)(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d \cdot i} \tag{5}$$

where d represents the dilation factor, k the filter size, and $s - d \cdot i$ serves as a representation of the past direction.

In this work, the specific TCN implementation tweaked and incorporated can be found in [48]. Given the basic tsAI setup, the specific configuration used also included, inter alia, a learning rate search for each input setup and a validation loss early stop. Here, the important thing is that the model is trained on each of the multivariate input settings. These incorporate past closing prices and emotion-related information relevant to the attitudes toward them. Mostly, however, they retain those sentiment and emotion time series that have been identified as significant by the previous feature selection step. Therefore, at the output of this phase, the predictions are blended, the number of which is equal to the number of significant features of the correlation feature selection process.

4.3. Averaging

All that remains now is the calculation of the average of the predictions from the previous phase:

$$P = \frac{1}{n} \sum_1^n p_i, p_i \in TCN_{p_i} \tag{6}$$

where TCN_{p_i} is the set containing the TCN predictions regarding the setups suggested by the feature selection procedure. With this simple calculation, we have our final prediction, P . In what follows, we will see how the presented methodology displays the best behavior in terms of general performance, always placing at the top of all rankings.

5. Results

The results will now be presented. Here, CD diagrams, bar plots, and tables of the overall averaged results will be used. The CD diagrams will depict the top 10 of the overall Friedman rankings of the competing methods examined regarding all time shifts arranged by metric. Bar plots will depict the numerical values of the Friedman ranks regarding the best-performing schemes.

The tables will contain the top three configurations regarding the average performance of each method in terms of the corresponding metric independent of time shift. This

means that the tables are going to include information about the exact values of the metrics, whereas the CD diagrams and bar plots will show relative rankings.

Thus, Table 5 depicts the average metric values of the three best-performing methods regardless of time shift. Looking at Table 5, one first notices the general superiority of the proposed methodology, that is, the one we refer to as “TCN Mean”, which ranks first in every metric. Beyond that, there is not much to say here about the proposed configuration apart from the fact that its aforementioned prevalence is clear in every metric. Besides the performance of the proposed methodology, however, one can observe various ensemble layouts appearing in the top positions. These methods are the best-performing in the context of every possible weighted average combination version of the base learners presented in Table 1. Regarding this, all the tables and illustrations presented in this section also include a description of the respective weights used in every weighted average ensemble. Specifically, beyond the TCN Mean, one can distinguish such an ensemble configuration: the ensemble consisting of a TCN that incorporates the fear feature—as extracted from the emotion classification process, together with an XCMPlus trained on the setup containing the weekly rolling mean version of the admiration emotion feature, in a linear arrangement with corresponding weights [0.9, 0.1]. This method ranks second in every metric, with the exception of RMSLE, where it ranks third. Additionally, when looking not only at the ranks but also at the values of the metrics, we can further observe that the method clearly loses on average, but in some cases, not by too much.

Table 5. Average performance per metric: top three.

Nº	Method	MAE	Nº	Method	MAPE
1	TCN Mean	4.158	1	TCN Mean	0.171
2	TCN fear & XCMPlus admiration RM7 [0.9, 0.1]	4.309	2	TCN fear & XCMPlus admiration RM7 [0.9, 0.1]	0.174
3	TCN disgust RM7	4.502	3	TCN fear & XCMPlus gratitude RM7 [0.9, 0.1]	0.175
Nº	Method	MSE	Nº	Method	RMSE
1	TCN Mean	74.057	1	TCN Mean	5.120
2	TCN fear & XCMPlus admiration RM7 [0.9, 0.1]	75.891	2	TCN fear & XCMPlus admiration RM7 [0.9, 0.1]	5.321
3	TCN disgust RM7	79.539	3	TCN disgust RM7	5.430
Nº	Method	RMSLE	Nº	Method	R ²
1	TCN Mean	0.093	1	TCN Mean	0.400
2	TCN Close RM7 & XCMPlus gratitude RM7 [0.9, 0.1]	0.098	2	TCN fear & XCMPlus admiration RM7 [0.9, 0.1]	0.209
3	TCN fear & XCMPlus admiration RM7 [0.9, 0.1]	0.101	3	TCN nervousness RM7	0.185

In addition, when looking at the emotion features that appear in the first positions, we can also observe something rather expected: emotions, such as fear, admiration, and disgust, exhibit the best efficiencies, something that seems, even according to common sense, to make sense, given that the emotions in question are attitudes towards stocks that can be related to a general predisposition, and the hypothesis according to which the what is said about stocks subdefines how stocks fluctuate, is valid. Still, our methodology performs best by far.

This can also be seen from the Friedman rankings presented in Figures 5 and 6. The five best-performing methods are presented in Figure 5. The CD diagrams in Figure 6 contain the 10 best-performing schemes as well as aspects of their corresponding statistical mutual relations. Again, our methodology ranks first in every valuation metric by a wide margin. However, here, the ensembles positioned below the proposed methodology are not the same as those that were included in the ranking that contained the average values per metric presented in Table 5. In this context, the following two additional features seem, in our opinion, quite interesting: gratitude and approval. These participate in configurations that generally occupy the top positions in the rankings. Each of these features seems to be able to capture a relatively relevant attitude toward the corresponding stocks in our case study. In conclusion, observe that the CD diagrams also indicate the clear statistical independence of the proposed TCN Mean methodology.

We have two additional remarks: first, each of the best-performing configurations, both regarding the average metric value performances and the Friedman rankings, contains specific features that have been extracted from the BERT-based multi-label emotion classifier. As we will see below, regarding the TCN Mean methodology, emotion features are extracted from the feature selection process, which suggests multivariable setups with TCN predictions that constitute the final averaged ensemble of the proposed methodology. Thus, emotion classification is crucial in our configuration. Second, each of the best-performing methodologies has a TCN as its main component. This is particularly important if we remember the following: the investigation here includes the five best methods from a set of 30 methods tested in [2], and the experiments involved the same context. Thus, the results here build on the results of [2], and this indicates an extremely widespread prevalence of the proposed methodology in a huge number of individual and ensemble methods.

Now, in relation to the configuration presented in Figure 4, the aforementioned ranking of TCN Mean does not include the feature selection process. Actually, the incorporation of the latter leads to a further increase in the efficiency of the proposed method, and this constitutes a further indication of the essentiality of introducing relevant emotion features—beyond, for example, the fact that the univariate version never appears in the higher places in the rankings. Here, in the same way as before, aggregated results, once again, regarding two aggregation tactics will be presented: one containing the average metric values depicted in Table 6 and the other Friedman ranks included in Figures 7 and 8. There are three competing tactics here: initially, TCN Mean without feature selection, a TCN Mean version that incorporates mutual information [49] feature selection, and finally, the proposed methodology: a TCN Mean version that exploits the correlation feature selection procedure mentioned above.

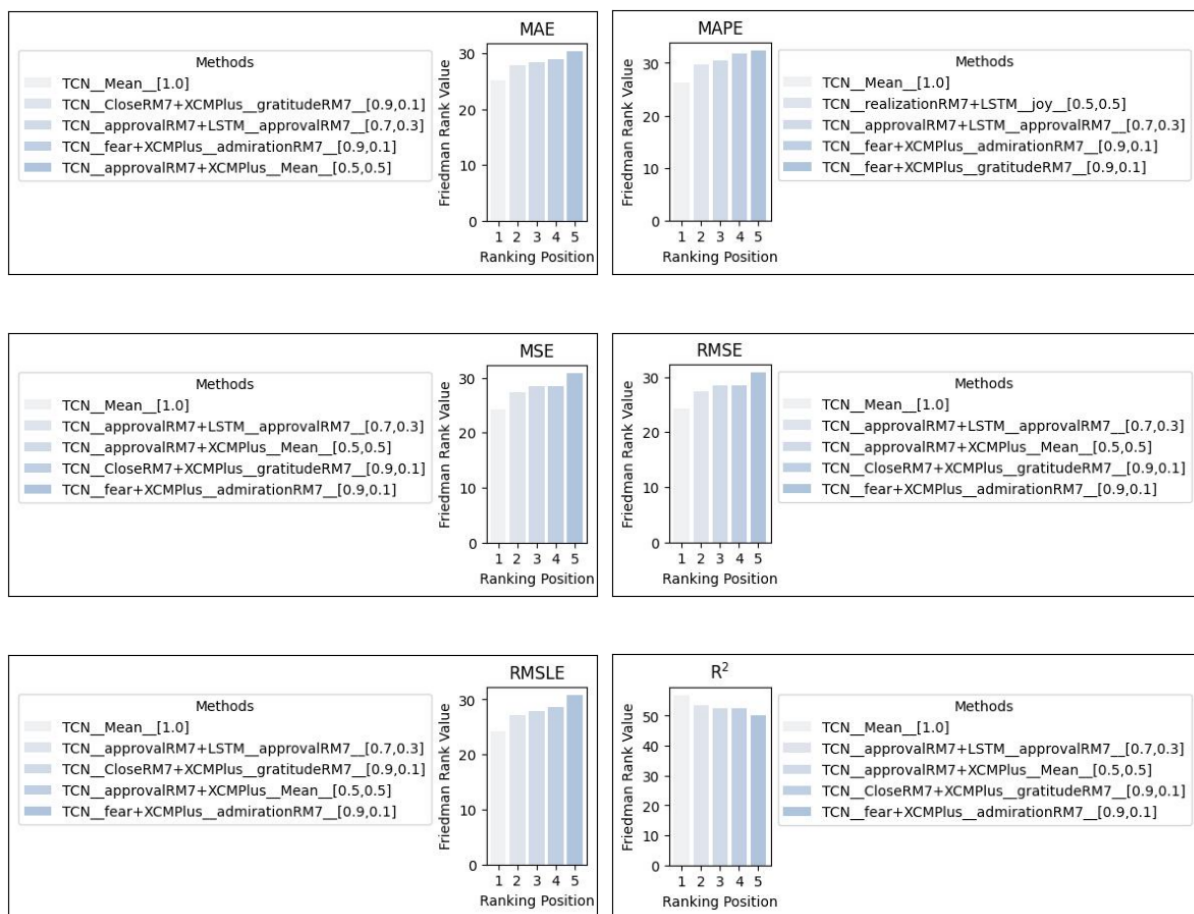


Figure 5. Aggregate Friedman rankings.

Table 6. Average performance per metric: feature selection.

	TCN Mean	TCN Cor	TCN Mut	Best
MAE	4.158	3.804	3.866	TCN Cor
MAPE	0.082	0.074	0.078	TCN Cor
MSE	74.057	68.764	68.622	TCN Mut
RMSE	5.120	4.798	4.865	TCN Cor
RMSLE	0.093	0.087	0.090	TCN Cor
R²	0.400	0.471	0.437	TCN Cor

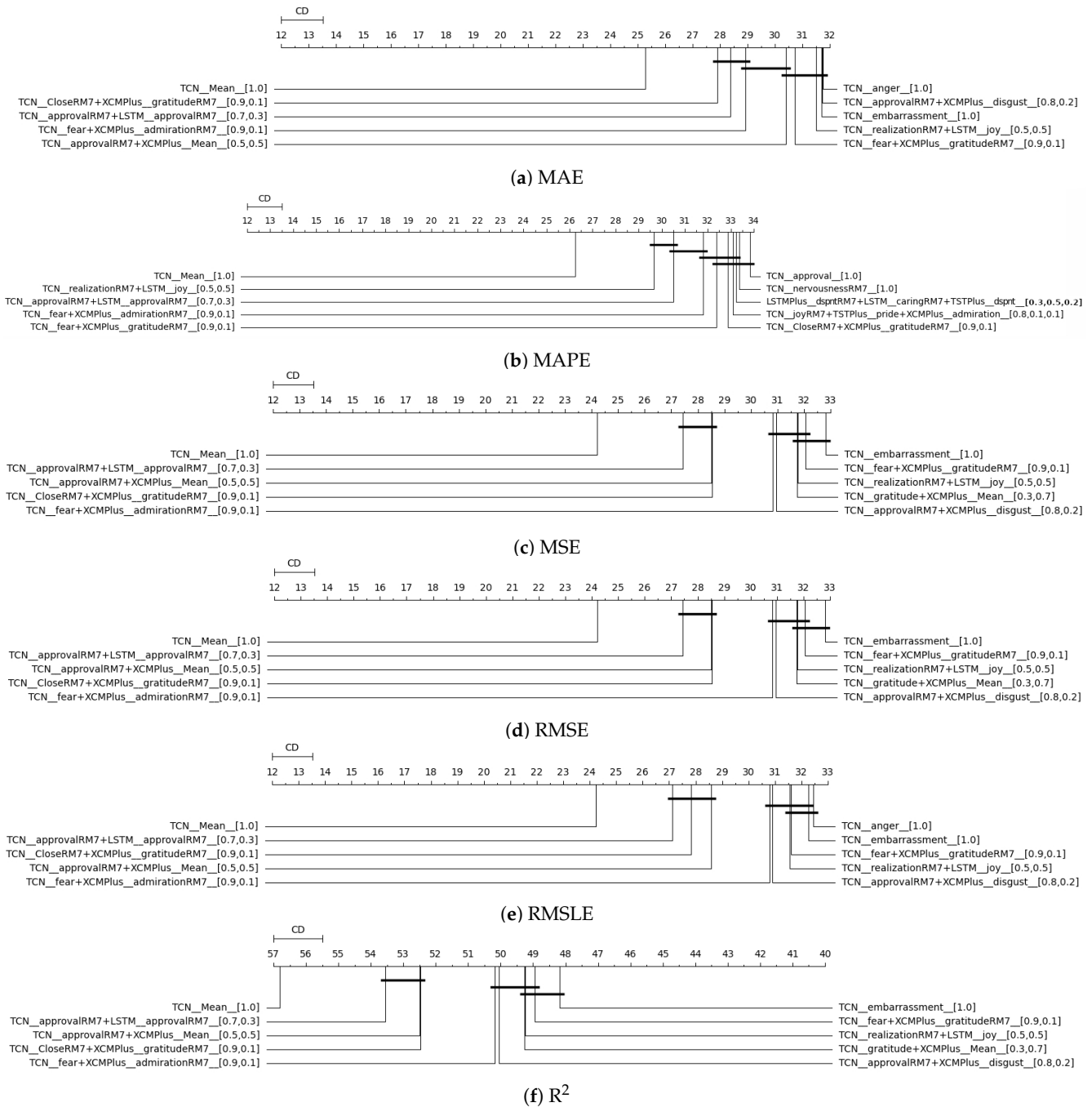


Figure 6. CD diagrams: Friedman rankings.

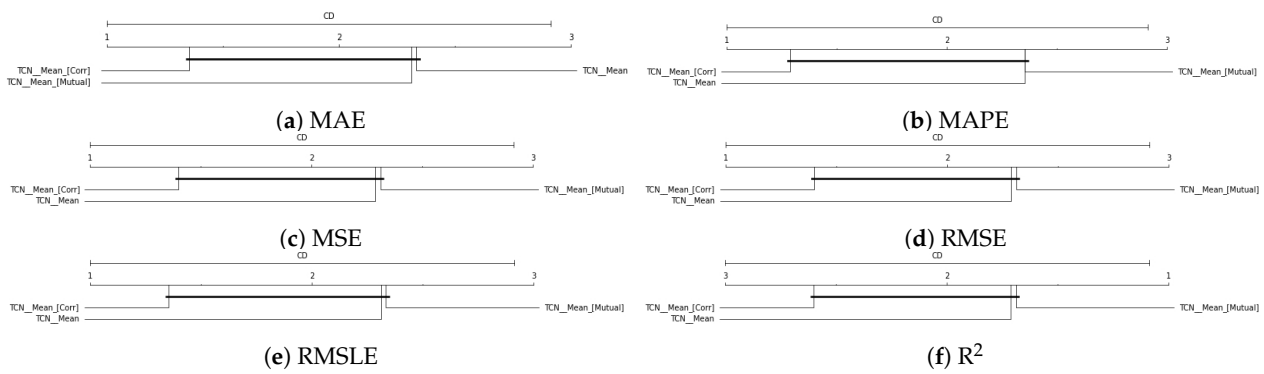


Figure 7. CD diagrams: Friedman rankings.

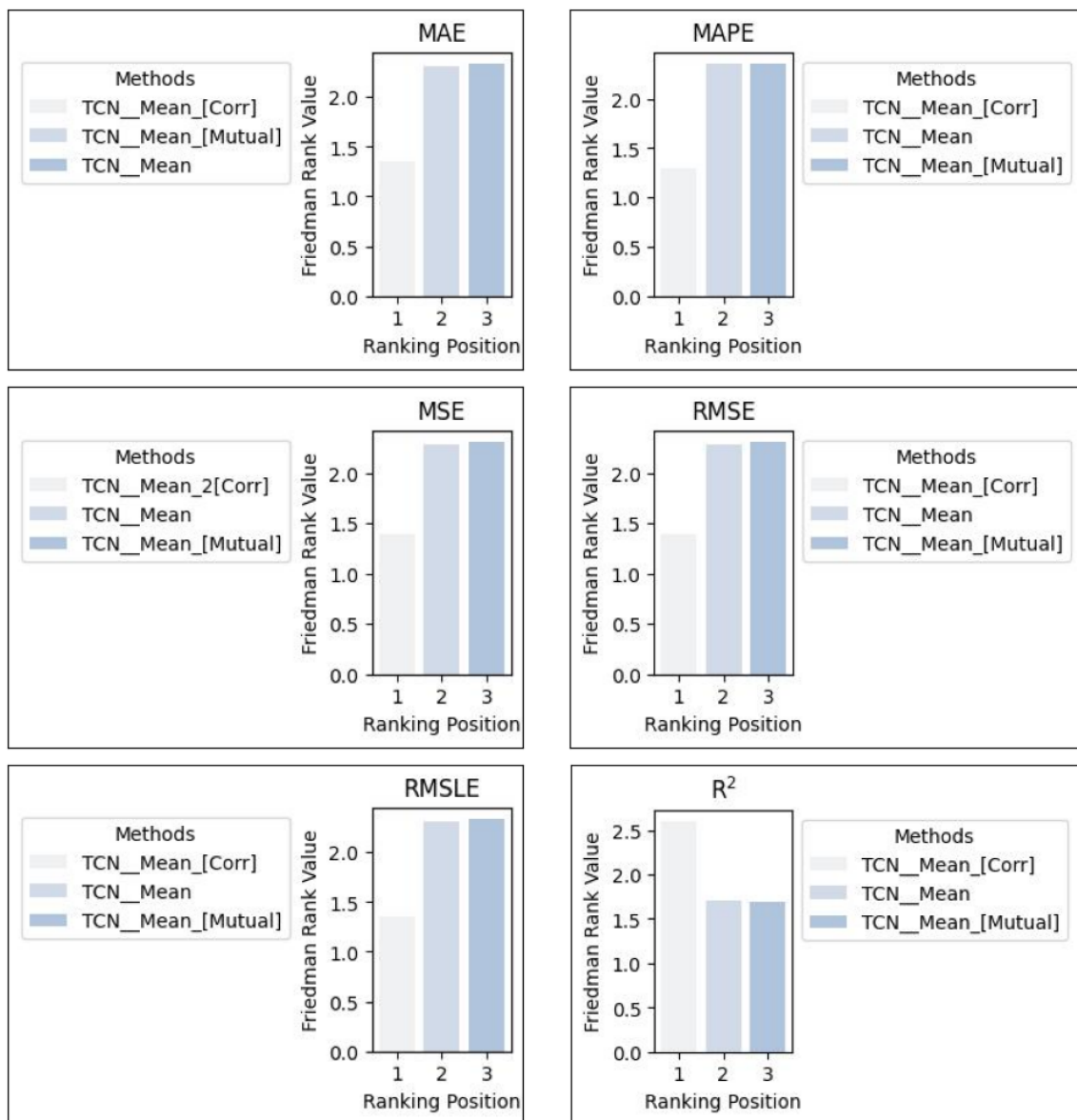


Figure 8. Aggregate feature Friedman rankings.

In short, in both cases, the dominance—in terms of ranking—of the correlation feature selection strategy is easily observed. Both in the CD diagrams and bar plots, as well as in the average rankings, this strategy is placed at the top of the results. An exception is the

MSE metric in the rankings of Table 6. There, the use of mutual information-based feature selection ranks first. However, the general prevalence of the correlation strategy is clear and, therefore, recommended. Specifically, in terms of average metric values, the difference in performance between the strategies is even more evident. This, among others, also indicates that the performance of the methodology we propose is greatly increased in terms of absolute numbers with the incorporation of the correlation feature selection strategy.

6. Conclusions

In conclusion, in this paper, a methodology for stock market forecasting was presented. This methodology makes use of multiple predictions that have been derived from the use of temporal convolutional networks over multivariate input setups made up of features extracted with the use of a BERT-based multi-label emotion classifier over textual data from Twitter. These input feature setups are instantiated according to a correlation feature selection process. On top of that, the forecasts extracted from the TCN are combined using a simple average scheme. The methodology was tested against a large number of individual and ensemble methods for three different time frames: single-day, 7-day, and 14-day time-shift predictions.

Therefore, the entire framework of this work is the last stage of an investigation that includes a multitude of regression, classical machine learning, and deep learning methods [2,40]. Here, the results presented concerned deep learning methods; the experiments started with the results in [2], in terms of only identifying the initial set of the best-performing methods shown in Table 1. Therefore, from a set of 30 algorithms, the five with the best performances were placed under an exploratory, experimental framework for possible weighted ensembles. All the possible ensembles investigated used features extracted from the aforementioned multi-label classifier. After extensive and thorough testing, it emerged that the methodology we propose substantially outperforms any other configuration.

Specifically, we saw our technique outperform every individual and ensemble method over a multitude of six different metrics. The aforementioned aggregate superiority was observed both in the weighted average metric value rankings and in the Friedman rankings. The effects of using the BERT-based multi-label classifier were also observed. Textual data underwent emotion analysis, every output of which was grouped by day and averaged, forming an emotion-based time series related to the investigated stocks. The results show that almost every top-performing method contained features related to some emotion. We also saw that specific well-performing configurations consisted of specific emotions that seemed to be correlated with the stock prices in question, which is also apparent in the context of common sense. Regarding the prediction schemes, the TCN architectures are not only at the top of the rankings but are also constitutive of each of the well-performing configurations. In conclusion, both the use of multi-label emotion classification in combination with the correlation feature selection process and the incorporation of temporal convolutional networks are easily and highly recommended. Given that the performance of the present framework shows the aforementioned promise, it should be further explored over longer time horizons and with broader and more complex combinatorial schemes. Moreover, the integration of additional emotional information extracted from sources other than Twitter, such as news websites and blogs, is a similarly promising perspective that can lead, together with tweet mining procedures, to even more fine-grained sentiment representations, resulting in better correlations and corresponding predictions.

Author Contributions: Conceptualization, C.M.L. and S.K.; methodology, C.M.L.; software, C.M.L.; validation, C.M.L. and S.K.; formal analysis, C.M.L.; investigation, C.M.L.; resources, S.K.; data curation, C.M.L.; writing—original draft preparation, C.M.L.; writing—review and editing, C.M.L.; visualization, C.M.L.; supervision, S.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The stock-related datasets analyzed during the current study consist of data that are available online. Data on stock closing prices can be downloaded, among others, from Yahoo! Finance’s API. The text dataset consists of posts written in the English language posted on <https://twitter.com/> over a period spanning 2 January 2018 to 24 December 2020 and was extracted using the corresponding stock name as a keyword.

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

Appendix A. Initial Set of All Examined Algorithms

Table A1. All Algorithms ¹.

No.	Abbreviation	Algorithm
1	FCN	Fully Convolutional Network [50]
2	FCNPlus	Fully Convolutional Network Plus [51]
3	IT	Inception Time [52]
4	ITPlus	Inception Time Plus [53]
5	MLP	Multilayer Perceptron[50]
6	RNN	Recurrent Neural Network [37]
7	LSTM	Long Short-Term Memory [36]
8	GRU	Gated Recurrent Unit [54]
9	RNNPlus	Recurrent Neural Network Plus [37]
10	LSTMPus	Long Short-Term Memory Plus [37]
11	GRUPlus	Gated Recurrent Unit Plus [37]
12	RNN_FCN	Recurrent Neural—Fully Convolutional Network [55]
13	LSTM_FCN	Long Short-Term Memory—Fully Convolutional Network [56]
14	GRU_FCN	Gated Recurrent Unit—Fully Convolutional Network [57]
15	RNN_FCNPlus	Recurrent Neural—Fully Convolutional Network Plus [58]
16	LSTM_FCNPlus	Long Short-Term Memory—Fully Convolutional Network Plus [58]
17	GRU_FCNPlus	Gated Recurrent Unit—Fully Convolutional Network Plus [58]
18	ResCNN	Residual—Convolutional Neural Network [59]
19	ResNet	Residual Network [50]
20	RestNetPlus	Residual Network Plus [60]
21	TCN	Temporal Convolutional Network [34]
22	TST	Time Series Transformer [61]
23	TSTPlus	Time Series Transformer Plus [38]
24	TSiTPlus	Time Series Vision Transformer Plus [62]
25	Transformer	Transformer Model [63]
26	XCM	Explainable Convolutional Neural Network [64]
27	XCMPlus	Explainable Convolutional Neural Network Plus [35]
28	XceptionTime	Xception Time Model [65]
29	XceptionTimePlus	Xception Time Plus [66]
30	OmniScaleCNN	Omni-Scale 1D-Convolutional Neural Network [67]

¹ Examined in [2].

References

- Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2019**, arXiv:1810.04805.
- Liapis, C.M.; Karanikola, A.; Kotsiantis, S.B. Investigating Deep Stock Market Forecasting with Sentiment Analysis. *Entropy* **2023**, *25*, 219. [CrossRef] [PubMed]
- Shi, W.; Li, F.; Li, J.; Fei, H.; Ji, D. Effective Token Graph Modeling using a Novel Labeling Strategy for Structured Sentiment Analysis. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022.
- Fei, H.; Li, F.; Li, C.; Wu, S.; Li, J.; Ji, D. Inheriting the Wisdom of Predecessors: A Multiplex Cascade Framework for Unified Aspect-based Sentiment Analysis. In Proceedings of the International Joint Conference on Artificial Intelligence, Vienna, Austria, 23–29 July 2022.
- Fei, H.; Ren, Y.; Wu, S.; Li, B.; Ji, D. Latent Target-Opinion as Prior for Document-Level Sentiment Classification: A Variational Approach from Fine-Grained Perspective. In Proceedings of the Web Conference 2021, Ljubljana, Slovenia, 19–23 April 2021.
- Fei, H.; Li, B.; Liu, Q.; Bing, L.; Li, F.; Chua, T.S. Reasoning Implicit Sentiment with Chain-of-Thought Prompting. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Toronto, ON, Canada, 9–14 July 2023.

7. Araci, D. FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. *arXiv* **2019**, arXiv:1908.10063.
8. Hájek, P.; Munk, M. Speech emotion recognition and text sentiment analysis for financial distress prediction. *Neural Comput. Appl.* **2023**, *35*, 21463–21477. [[CrossRef](#)]
9. Kratzwald, B.; Ilić, S.; Kraus, M.; Feuerriegel, S.; Prendinger, H. Deep learning for affective computing: Text-based emotion recognition in decision support. *Decis. Support Syst.* **2018**, *115*, 24–35. [[CrossRef](#)]
10. Valle-Cruz, D.; Fernandez-Cortez, V.; Chau, A.L.; Sandoval-Almazán, R. Does Twitter Affect Stock Market Decisions? Financial Sentiment Analysis During Pandemics: A Comparative Study of the H1N1 and the COVID-19 Periods. *Cogn. Comput.* **2021**, *14*, 372–387. [[CrossRef](#)]
11. Agarwal, B. Financial sentiment analysis model utilizing knowledge-base and domain-specific representation. *Multimed. Tools Appl.* **2022**, *82*, 8899–8920. [[CrossRef](#)]
12. Sohagir, S.; Wang, D.; Pomeranets, A.; Khoshgoftaar, T.M. Big Data: Deep Learning for financial sentiment analysis. *J. Big Data* **2018**, *5*, 1–25. [[CrossRef](#)]
13. Lengkeek, M.; van der Knaap, F.; Frasinca, F. Leveraging hierarchical language models for aspect-based sentiment analysis on financial data. *Inf. Process. Manag.* **2023**, *60*, 103435. [[CrossRef](#)]
14. Xiang, C.; Zhang, J.; Li, F.; Fei, H.; Ji, D. A semantic and syntactic enhanced neural model for financial sentiment analysis. *Inf. Process. Manag.* **2022**, *59*, 102943. [[CrossRef](#)]
15. Chai, Y.; Teng, C.; Fei, H.; Wu, S.; Li, J.; Cheng, M.; Ji, D.H.; Li, F. Prompt-Based Generative Multi-label Emotion Prediction with Label Contrastive Learning. In Proceedings of the Natural Language Processing and Chinese Computing, Guilin, China, 24–25 September 2022.
16. Alhuzali, H.; Ananiadou, S. SpanEmo: Casting Multi-label Emotion Classification as Span-prediction. *arXiv* **2021**, arXiv:2101.10038.
17. Fei, H.; Ji, D.; Zhang, Y.; Ren, Y. Topic-Enhanced Capsule Network for Multi-Label Emotion Classification. *IEEE/ACM Trans. Audio, Speech Lang. Process.* **2020**, *28*, 1839–1848. [[CrossRef](#)]
18. Fei, H.; Zhang, Y.; Ren, Y.; Ji, D. Latent Emotion Memory for Multi-Label Emotion Classification. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020.
19. Ma, H.; Ma, J.; Wang, H.; Li, P.; Du, W.C. A Comprehensive Review of Investor Sentiment Analysis in Stock Price Forecasting. In Proceedings of the 2021 IEEE/ACIS 20th International Fall Conference on Computer and Information Science (ICIS Fall), Xi'an, China, 13–15 October 2021; pp. 264–268.
20. Janková, Z. Critical Review Of Text Mining And Sentiment Analysis For Stock Market Prediction. *J. Bus. Econ. Manag.* **2023**, *24*, 177–198. [[CrossRef](#)]
21. Ashtiani, M.N.; Raahemi, B. News-based intelligent prediction of financial markets using text mining and machine learning: A systematic literature review. *Expert Syst. Appl.* **2023**, *217*, 119509. [[CrossRef](#)]
22. Seroyizhko, P.; Zhexenova, Z.; Shafiq, M.; Merizzi, F.; Galassi, A.; Ruggeri, F. A Sentiment and Emotion Annotated Dataset for Bitcoin Price Forecasting Based on Reddit Posts. In Proceedings of the FINNLP, 2022. Abu Dhabi, United Arab Emirates (Hybrid), December 2022; pp. 203–210.
23. Velu, S.R.; Ravi, V.; Tabianan, K. Multi-Lexicon Classification and Valence-Based Sentiment Analysis as Features for Deep Neural Stock Price Prediction. *Sci* **2023**, *5*, 8. [[CrossRef](#)]
24. Ider, D.; Lessmann, S. Forecasting Cryptocurrency Returns from Sentiment Signals: An Analysis of BERT Classifiers and Weak Supervision. *arXiv* **2022**, arXiv:2204.05781.
25. Ko, C.R.; Chang, H.T. LSTM-based sentiment analysis for stock price forecast. *PeerJ Comput. Sci.* **2021**, *7*, e408. [[CrossRef](#)]
26. Lee, J.; Youn, H.L.; Poon, J.; Han, S.C. StockEmotions: Discover Investor Emotions for Financial Sentiment Analysis and Multivariate Time Series. *arXiv* **2023**, arXiv:2301.09279.
27. Wan, R.; Mei, S.; Wang, J.; Liu, M.; Yang, F. Multivariate Temporal Convolutional Network: A Deep Neural Networks Approach for Multivariate Time Series Forecasting. *Electronics* **2019**, *8*, 876. [[CrossRef](#)]
28. Wang, Y.; Chen, J.; Chen, X.; Zeng, X.; Kong, Y.; Sun, S.; Guo, Y.; Liu, Y. Short-Term Load Forecasting for Industrial Customers Based on TCN-LightGBM. *IEEE Trans. Power Syst.* **2020**, *36*, 1984–1997. [[CrossRef](#)]
29. Hewage, P.R.P.G.; Behera, A.; Trovati, M.; Pereira, E.G.; Ghahremani, M.; Palmieri, F.; Liu, Y. Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station. *Soft Comput.* **2020**, *24*, 16453–16482. [[CrossRef](#)]
30. Lei, B.; Zhang, B.; Song, Y. Volatility Forecasting for High-Frequency Financial Data Based on Web Search Index and Deep Learning Model. *Mathematics* **2021**, *9*, 320. [[CrossRef](#)]
31. Gong, L.; Yu, M.; Jiang, S.; Cutsuridis, V.; Pearson, S. Deep Learning Based Prediction on Greenhouse Crop Yield Combined TCN and RNN. *Sensors* **2021**, *21*, 4537. [[CrossRef](#)] [[PubMed](#)]
32. Friedman, M. The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. *J. Am. Stat. Assoc.* **1937**, *32*, 675–701. [[CrossRef](#)]
33. Dunn, O.J. Multiple Comparisons among Means. *J. Am. Stat. Assoc.* **1961**, *56*, 52–64. [[CrossRef](#)]
34. Bai, S.; Kolter, J.Z.; Koltun, V. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv* **2018**, arXiv:1803.01271.
35. Oguiza, I. tsAI Models: XCMPlus. Available online: <https://timeseriesai.github.io/tsai/models.xcmplus.html> (accessed on 7 November 2022).

36. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
37. Oguiza, I. tsAI Models: RNNS. Available online: <https://timeseriesai.github.io/tsai/models.rnn.html> (accessed on 7 November 2022).
38. Oguiza, I. tsAI Models: TSTPlus. Available online: <https://timeseriesai.github.io/tsai/models.tstplus.html> (accessed on 7 November 2022).
39. timeseriesAI. Timeseriesai/Tsai: Time Series Timeseries Deep Learning Machine Learning Pytorch FASTAI: State-of-the-Art Deep Learning Library for Time Series and Sequences in Pytorch/Fastai. Available online: <https://github.com/timeseriesAI/tsai> (accessed on 9 September 2023).
40. Liapis, C.M.; Karanikola, A.; Kotsiantis, S.B. A Multi-Method Survey on the Use of Sentiment Analysis in Multivariate Financial Time Series Forecasting. *Entropy* **2021**, *23*, 1603. [CrossRef]
41. TextBlob: Simplified Text Processing. Available online: <https://textblob.readthedocs.io/en/dev/> (accessed on 9 September 2023).
42. Hutto, C.J.; Gilbert, E. VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. In Proceedings of the International AAAI Conference on Web and Social Media, Ann Arbor, Michigan USA, 1–4 June 2014.
43. Demszky, D.; Movshovitz-Attias, D.; Ko, J.; Cowen, A.; Nemade, G.; Ravi, S. GoEmotions: A Dataset of Fine-Grained Emotions. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL), Online, 5–10 July 2020.
44. Iandola, F.N.; Shaw, A.E.; Krishna, R.; Keutzer, K.W. SqueezeBERT: What can computer vision teach NLP about efficient neural networks? *arXiv* **2020**, arXiv:2006.11316.
45. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
46. van den Oord, A.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.W.; Kavukcuoglu, K. WaveNet: A Generative Model for Raw Audio. *arXiv* **2016**, arXiv:1609.03499.
47. Yu, F.; Koltun, V. Multi-Scale Context Aggregation by Dilated Convolutions. *arXiv* **2015**, arXiv:1511.07122.
48. Oguiza, I. tsAI Models: TCN. Available online: <https://timeseriesai.github.io/tsai/models.tcn.html> (accessed on 7 November 2022).
49. Ross, B.C. Mutual Information between Discrete and Continuous Data Sets. *PLoS ONE* **2014**, *9*, e87357. [CrossRef] [PubMed]
50. Wang, Z.; Yan, W.; Oates, T. Time series classification from scratch with deep neural networks: A strong baseline. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 1578–1585.
51. Oguiza, I. tsAI Models: FCNPlus. Available online: <https://timeseriesai.github.io/tsai/models.fcncnplus.html> (accessed on 7 October 2021).
52. Fawaz, H.I.; Lucas, B.; Forestier, G.; Pelletier, C.; Schmidt, D.F.; Weber, J.; Webb, G.I.; Idoumghar, L.; Muller, P.A.; Petitjean, F. InceptionTime: Finding AlexNet for Time Series Classification. *arXiv* **2020**, arXiv:1909.04939.
53. Oguiza, I. tsAI Models: InceptionTimePlus. Available online: <https://timeseriesai.github.io/tsai/models.inceptiontimeplus.html> (accessed on 7 October 2021).
54. Chung, J.; Çaglar, G.; Cho, K.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv* **2014**, arXiv:1412.3555.
55. Oguiza, I. tsAI Models: RNN_FCNCN. Available online: https://timeseriesai.github.io/tsai/models.rnn_fcncn.html (accessed on 7 October 2021).
56. Karim, F.; Majumdar, S.; Darabi, H.; Chen, S. LSTM Fully Convolutional Networks for Time Series Classification. *IEEE Access* **2018**, *6*, 1662–1669. [CrossRef]
57. Elsayed, N.; Maida, A.; Bayoumi, M.A. Deep Gated Recurrent and Convolutional Network Hybrid Model for Univariate Time Series Classification. *arXiv* **2019**, arXiv:1812.07683.
58. Oguiza, I. tsAI Models: RNN_FCNCNPlus. Available online: https://timeseriesai.github.io/tsai/models.rnn_fcncnplus.html (accessed on 7 October 2021).
59. Zou, X.; Wang, Z.; Li, Q.; Sheng, W. Integration of residual network and convolutional neural network along with various activation functions and global pooling for time series classification. *Neurocomputing* **2019**, *367*, 39–45. [CrossRef]
60. Oguiza, I. tsAI Models: ResNetPlus. Available online: <https://timeseriesai.github.io/tsai/models.resnetplus.html> (accessed on 7 October 2021).
61. Zerveas, G.; Jayaraman, S.; Patel, D.; Bhamidipaty, A.; Eickhoff, C. A Transformer-based Framework for Multivariate Time Series Representation Learning. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, Singapore, 14–18 August 2021.
62. Oguiza, I. tsAI Models: TSIT. Available online: <https://timeseriesai.github.io/tsai/models.tsitplus.html> (accessed on 7 October 2021).
63. Oguiza, I. tsAI Models: Transformermodel. Available online: <https://timeseriesai.github.io/tsai/models.transformermodel.html> (accessed on 7 October 2021).
64. Fauvel, K.; Lin, T.; Masson, V.; Fromont, E.; Termier, A. XCM: An Explainable Convolutional Neural Network for Multivariate Time Series Classification. *arXiv* **2021**, arXiv:2009.04796.
65. Rahimian, E.; Zabihi, S.; Atashzar, S.F.; Asif, A.; Mohammadi, A. XceptionTime: A Novel Deep Architecture based on Depthwise Separable Convolutions for Hand Gesture Classification. *arXiv* **2019**, arXiv:1911.03803.

66. Oguiza, I. tsAI Models: XceptionTimePlus. Available online: <https://timeseriesai.github.io/tsai/models.xceptiontimeplus.html> (accessed on 7 October 2021).
67. Tang, W.; Long, G.; Liu, L.; Zhou, T.; Blumenstein, M.; Jiang, J. Omni-Scale CNNs: A simple and effective kernel size configuration for time series classification. In Proceedings of the ICLR, 2022, Online, 25–29 April 2022.

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.