Int. J. Adv. Eng. Pure Sci. 2021, 33(1): 18-27
DOI: 10.7240/jeps.683952

**RESEARCH ARTICLE** / ARAŞTIRMA MAKALESİ

# Stock Market Prediction in Istanbul Stock Exchange by Combining Stock Price Information and Sentiment Analysis

*Duygu Analizi ve Hisse Fiyat Bilgisinin Birleştirilmesiyle Hisse Senedi Piyasası Tahmini*

**Adnan GUMUS[1], C. Okan SAKAR[1]**

[1] *Bahcesehir University, Department of Computer Engineering, 34353, Istanbul, Turkey*

**Abstract**

Predicting the stock market instrument price is a valuable but challenging machine learning task. Researchers use advanced techniques to improve the generalization ability of stock prediction models. However, considering that the stock market highly depends on the political and macroeconomic developments as well as the mood of the related investors, the models that use only stock prices fail to cover all factors affecting the stock market. Therefore, to improve the prediction accuracy of stock market prediction, in this study, we first apply sentiment analysis to the news related with the market and related stock, and then combine the sentiment labels of the news with stock prices and commonly used technical indicators. The obtained cumulative dataset is used to train a long short-term memory recurrent neural network, and the output of this regression model is used in the prediction of the closing price movement to decide whether the closing price next day will be higher. The experiments performed on 8-year data showed that while the F1 score of the model built without sentiment analysis was around 0.56, it has increased to 0.65 when stock prices are combined with sentiment labels. The results show that the model with sentiment labels fits better to the actual prices especially when there is a high volatility in the stock price.

**Keywords:** Recurrent Neural Network, Long Short-Term Memory, Text Classification, Sentiment Analysis, Technical Analysis.

**Öz**

Pay piyasasındaki bir enstrümanın fiyatını tahmin etmek, oldukça değerli ve aynı zamanda oldukça zor yapay öğrenme görevlerinden biridir. Araştırmacılar hisse fiyat tahmin modellerinin genellenebilirlik kabiliyetlerinin arttırılması için ileri teknikler kullanmaktadır. Ancak, hisse senedi piyasasının politik ve makroekonomik haberler ve yatırımcıların duygu durumu ile yakından ilişkili olduğu düşünüldüğünde, yalnızca hisse fiyat bilgisini kullanan modeller hisse senedi piyasasını etkileyen tüm faktörleri kapsama konusunda yetersiz kalmaktadır. Bu nedenle, bu çalışmada, hisse senedi tahmin başarısının arttırılması için piyasa ve ilgili hisse haberlerine duygu analizi uygulanmakta ve daha sonra duygu etiketleri ile hisse fiyatları ve en sık kullanılan teknik indikatörleri birleştirilmektedir. Elde edilen kümülatif veri kümesi bir kısa uzun-hafızalı özyinelemeli sinir ağının eğitilmesinde kullanılmış ve bu regresyon modelinin çıktısı bir sonraki günün kapanış fiyatının daha yukarıda olup olmayacağını tahmin etmek için kullanılmıştır. Sekiz yıllık hisse senedi verisi üzerinde yapılan deneyler, duygu analizi olmadan kurulan modelin F1 skoru 0,56 civarında iken, hisse fiyatı duygu etiketleri ile birlikte kullanıldığında bu değerin 0,65 civarına yükseldiğini göstermiştir. Elde edilen sonuçlar, özellikle piyasada yüksek volatilite olduğunda, duygu etiketini kullanan modelin gerçek hisse fiyatlarına daha yakın olduğunu göstermektedir.

**Anahtar Kelimeler:** Özyinelemeli Sinir Ağları, Uzun Kısa-Dönem Hafıza, Metin Sınıflandırma, Duygu Analizi, Teknik Analiz.

## I. INTRODUCTION

Recently, there is an increasing interest in building stock price prediction models using various methods [1]. Most of the stock prediction works are based on the use of quantitative data consisting of historical stock prices and various technical indicators which can be taken from various financial data providers. These studies generally build statistical or machine learning-based prediction models to estimate the future stock prices [1, 2].

Improved prediction techniques that can cope with complex and non-linear dynamics of the stock market data such as deep learning methods have been applied the stock prediction problem [3-5]. However, the studies that use only numerical data for prediction mostly suffer from the lack of the use of some hidden factors in the input space such as the expectations about the related stock, mood of the society, micro/macro-economic and political developments. The relation between the past and current stock price and the future price refers to such factors which cannot be easily integrated to the prediction system. The investors and experts utilize fundamental analysis techniques to incorporate such factors into their investment decisions. In recent years, with the availability of

**Corresponding Author:** C. Okan Sakar, Tel: +90 212 381 0571, e-posta: okan.sakar@eng.bau.edu.tr
**Submitted:** 03.02.2020, **Revised:** 03.09.2020, **Accepted:** 05.10.2020

different platforms that may possess information about these hidden factors, some researchers have also attempted to include these factors in their prediction models. These studies generally combine historical prices with sentiment analysis of text data collected from various sources [6-9].

In this study, we aim to evaluate the effect of integrating investor sentiment into stock prediction task for Istanbul Stock Exchange. For this purpose, we build a model that predicts next day's stock price movement using historical stock prices, news about the related stock, and financial and political news. We perform sentiment analysis to the news using a recently developed library, FastText, which is used to obtain vector representations for words [10]. We classify each news as positive, negative or neutral, extract features from the sentiment outputs and feed these features to long short-term memory recurrent neural network (LSTM-RNN) algorithm as input together with the stock prices. We present the comparative results, with and without sentiment analysis, on a stock price dataset collected from Istanbul Stock Market.

One of the main contributions of this study is to use a longer period data, 8 years, to build the model. Most of the existing studies that combine sentiment analysis and historical data for stock prediction use a shorter test period [6]. We use a fast and efficient word embedding library for sentiment analysis, FastText, which has shown great success on various datasets [11-14]. Instead of feeding the outputs of sentiment analysis directly to the system, we extract features by combining the probability estimates of classes and the favorite count of each news. Thus, the importance of the news is also taken into account to generate the input. Another contribution is, to the best of knowledge for the first time, to combine sentiment analysis of news in Turkish language with price information and technical indicators for the prediction of a stock in Istanbul Stock Exchange. Although there are some previous works that analyze the impact of investor sentiment on conditional volatility and stock returns in Istanbul Stock Exchange [15, 16], different from these studies we integrate investor sentiment with price information using machine learning techniques for stock prediction. Considering that the financial market in emergent countries is highly sensitive to macro-economic and political news, we aim to test to what extent sentiment analysis can improve the prediction of the stock variability in Istanbul Stock Exchange.

The rest of this paper is organized as follows: Section 2 gives the related works that combine sentiment analysis and stock prices. Section 3 includes the description of the news and stock price datasets. The overview of the methods used in this study are explained in Section 4. Then, in Section 5, the sentiment classification and stock prediction results are presented. Finally, we conclude in Section 6.

## II. RELATED WORK

There are some literature studies that analyze the effect of sentiment analysis in stock prediction. These studies generally follow one of the following two approaches. The first approach is to apply sentiment analysis to the text data related with the stock and then perform a correlation analysis between the sentiment of the text data and price or movement direction of the stock [8]. The second approach is to combine the sentiment information of the related text data with the stock price information and predict the price or movement direction with statistical or machine learning approaches using the combined data as input. In this section, we focus on the studies that follow the second approach since, similar to those studies, we evaluate the effect of sentiment analysis by combining it with the stock price data.

The sentiment analysis in the related studies is performed on the news collected from financial news providers or from comments of the related community in various social media platforms. In one of these studies, Nguyen et al. [6] extracted topics and related sentiments from the text in a message board and combined the obtained data with the historical price information to predict the stock movement. They designed the prediction problem as a binary classification problem, whether the stock price will be up or down, and used support vector machines with linear kernel for prediction. They used only the last two day's prices and sentiment analysis data as input to predict the next day's adjusted closing price. The results of their experiments performed on 1-year data on 18 stocks showed that the stock-price based model can be improved by around 2% accuracy in average by integrating the sentiment analysis into the model. They also noted that for the stocks that are difficult to predict, the improvement was around 10%. This finding might be a motivation to use sentiment analysis in the prediction of emergent countries' stocks since there is a higher volatility in these markets.

Liu et al. [9] aimed to predict the stock volatility in the Chinese market using stock price data with sentiment analysis. They, similar to our study, fed the fused input to recurrent neural networks for prediction. For sentiment analysis part, they extracted emotional words and assigned a weight to each of these words. They used the sum of weighted sentiment scores of all terms to represent the sentiment polarity of the related content. The dataset used in this study belongs to 250 trading days. They have tried many different time-step values, i.e. the length of historical data that will be used for prediction and found 10-days period as the optimal value. The results showed that the accuracy can be improved up to 12%, from 57% to 69%, by using sentimental information as input.

In another related work, Li et al. [17] proposed a sentiment-aware stock market prediction system. They used Naïve-Bayes to classify forum posts as positive, negative, or neutral, combined this information with stock prices and gave it to LSTM-RNN to predict opening and closing prices of the next day. They also compared the LSTM-RNN results to SVM and concluded that LSTM-RNN outperforms SVM due to its ability of handling temporal sequences. They achieved an average F1 score of 0.63 for the sentiment classification task with Naïve Bayes. When the output of sentiment analysis is fed to the prediction model with stock price data, they achieved 88% accuracy for opening prices and 50% accuracy for closing prices.

Zhuge et al. [18] combined sentiment analysis and stock price data to predict the exact stock price in Shanghai Stock Exchange instead of predicting the movement of the stock. They also used Naïve Bayes to classify the posts in the forums as positive or negative. The combined data is fed to LSTM-RNN. We should note that, different from our study, they built the model to predict the opening price of the stock, which is an easier task than predicting the closing price since the input belonging to the previous days contains more recent information about the opening price.

Day and Lee [19] performed experiments to evaluate the effect of using different finance news providers in stock price movement using a deep-neural network architecture. They collected the news data from 4 news providers and the obtained results on the collected data showed that different financial news providers have significantly different effects to investors and their investments.

Weng et al. [20] fused information taken from multiple data sources to better predict the stock movements. They do not perform a sentiment analysis but instead use data from knowledge bases such as Google and Wikipedia to represent the public's collective interest in the related stock. They specifically give the number of visits made to the Wikipedia pages of the related stock and the amount of online content produced for the related stock into the model along with some technical indicators and stock prices. They performed experiments on Apple NASDAQ stock movement data spanning 37 month-period from May 1, 2012 to June 1, 2015. Their findings showed that combining stock price data with the information extracted from such knowledge bases may help to build more generalizable prediction models.

## III. DATASET DESCRIPTION

In this section, we provide details about the news and stock price dataset used in our experiments. We also give the pre-processing operations applied to the text data in this section.

### 3.1. Stock Data

Financial data can be collected from various data providers. In our study, we used Bloomberg platform to collect price information of a stock from Istanbul Stock Exchange. For this purpose, first, we collected daily price of GARAN stock for the years between 2010 and 2018. This dataset has 1892 data points each of which belongs to a trading day in Istanbul Stock Market in the specified time interval. We aim to predict the next day's closing price using the data belonging to the previous *k* days, where *k* represents the length of the historical data that will be used for the prediction. The output variable is a binary-coded variable whose value has been determined according to the sign of the difference between the closing price at time *t* and closing price at time *t-1*. The increase in price is represented with 1 and decrease in price with 0. The stock price historical data fed as input to the prediction model consists of the opening, highest, lowest and closing prices of the previous days.

### 3.2. Financial News Data

For the sentiment analysis part of our study, we received related political and financial news from Twitter. We collected 8 years of news about economy, politics and stock itself. We aim to obtain more reliable results by working with 8 years of data. Another advantage of using a long period of training data is that there are many seasonal movements and trends in the specified interval.

Figure 1 shows the distribution of number of news per day in the specified time interval. The total number of news is 55,341 and on average there are 62 news per day. It is seen that the number of comparably recent news is more than those of the previous years. Also, we have the 'favorite count' information for each news which indicates to what extent the related news is regarded by the related community. We used this information as a measure of the importance of the related news.

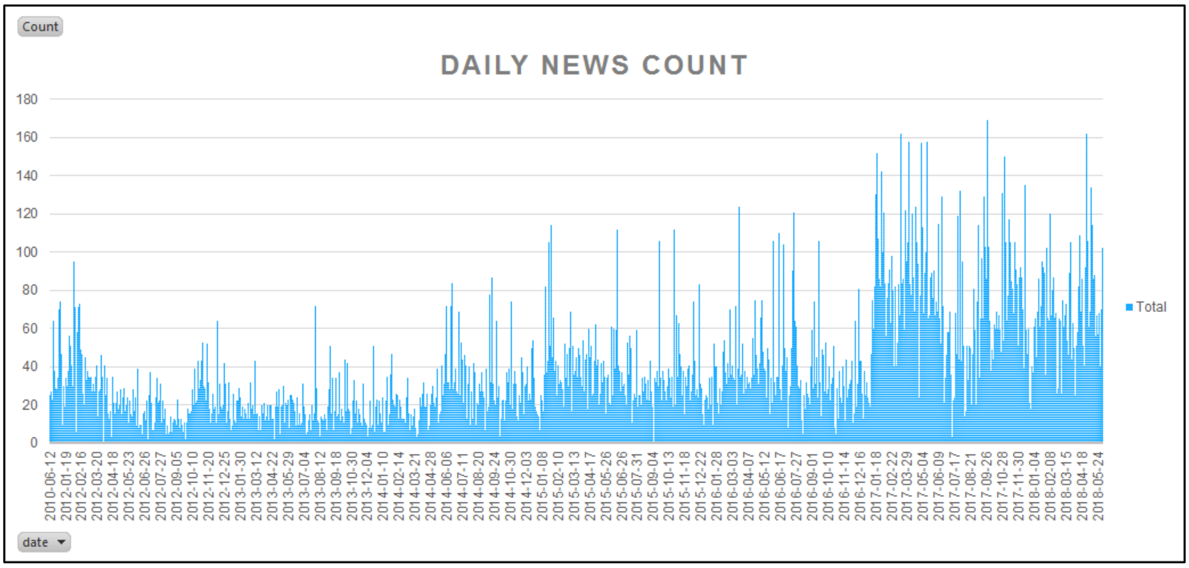

**Figure 1.** Distribution of number of daily news in the test period

### 3.3. Technical Indicators

In addition to the historical stock prices and sentiments of daily news, we calculated some technical indicators and incorporated them into the prediction model. The technical indicators are used in technical analysis by investors to make some predictions and inference about the future movements of the stocks in the market. Technical indicators are defined on the basis of previous stock prices and transaction volume information.

In our study, we used the variations of moving average, volume, and relative strength index. The moving average (MA) indicator is one of the most widely used indicators in technical analysis. To calculate MA, the average price of the last period is taken, and this value is used to evaluate how far the current price is from the average of the last period. We use the MA of 10, 30, and 50 days to represent this technical indicator in our input space. We also use the volume of the previous day and moving average of the volume of the last 5 days as input in our dataset.

The Relative Strength Index (RSI) is an indicator used to determine whether excessive prices of the financial instrument are oversold. The most common period used for RSI is 14 days [21]. We used 3, 9, 14, and 30 days in our study. The RSI value is measured in a range from 0 to 100, with a “high” value located at 70 and a “low” value located at 30.

## IV. METHODS

In this section, the methods used throughout the experiments are described.

### 4.1. Text Processing and Classification

We first applied data cleaning techniques to clear the text data from irrelevant content such as spaces, web addresses, hashtags, and emojis. We have also detected and deleted the duplicated news. Stop words are the most commonly occurring words in a text which are not relevant with the context of the data and do not contribute to the semantic meaning of the phrase. We removed these words from our news dataset. Then, we converted all words to small-case letters. The date information has been kept to be used for matching the news with the corresponding stock price information.

After the pre-processing steps, the obtained dataset is given to a text classifier for sentiment analysis. For this purpose, we first manually labeled 500 news as positive, negative and neutral. The average number of words in each news is 1401. Then, we used 10-fold cross validation to build a classification model using FastText tool. FastText is a library that efficiently performs vector representations and sentence classification of words, especially in languages such as Turkish where the same word differs structurally in grammar [10, 22]. FastText supports both supervised and unsupervised representations of words and sentences. The model built with FastText assigns each news into positive, negative, or neutral categories.

As we have stated in Section 3.2, in addition to the content of the news, we have the ‘favorite count’ information which indicates to what extent the related news is regarded by the related community. To extract more descriptive features from the news, we used the ‘favorite count’ information as a measure of the importance of the related news. For this purpose, we calculated weighted average values using sentiment labels and created PosAVG, NegAVG, and NotrAVG features from the news labeled as positive, negative or neutral, respectively, to represent the distribution of the sentiment of the news in the corresponding trading day. The value of PosAVG feature is calculated as:

$$PosAVG = \frac{1}{N_{pos}} \sum_{i \in X_{pos}} p_{pos}^{i} * f^{i} \qquad (1)$$

where $X_{pos}$ represents the set of news labeled as positive, $p_{pos}^{i}$ the probability of $i^{\text{th}}$ news belonging to positive class, $f^{i}$ the favorite count of $i^{\text{th}}$ news, and $N_{pos}$ the number of positive news in the corresponding trading day. Also, standard deviations for each sentiment label are calculated and the obtained features are combined with historical prices and technical indicators. All features used in our study are shown in Table 1.

**Table 1.** Summary of the features used in this study for stock movement prediction

| **Feature** | **Dataset** | **Explanation** |
|---|---|---|
| Output | Stock Price | Target variable: Stock price at time *t + 1* |
| Open | Stock Price | Opening price at time *t* |
| High | Stock Price | Highest price at time *t* |
| Low | Stock Price | Lowest price at time *t* |
| Close | Stock Price | Closing price at time *t* |
| Volume | Stock Price | Volume at time *t* |
| MAV5Day | Technical Indicator | Moving Average of the volume of last 5-days |
| RSI3Day | Technical Indicator | RSI of last 3-days |
| RSI9Day | Technical Indicator | RSI of last 9-days |
| RSI14Day | Technical Indicator | Relative Strength Index of last 14-days |
| RSI30Day | Technical Indicator | Relative Strength Index of last 30-days |
| MA10Day | Technical Indicator | Moving Average of last 10-days |
| MA30Day | Technical Indicator | Moving Average of last 30-days |
| MA50Day | Technical Indicator | Moving Average of last 50-days |
| PosAVG | Sentiment Analysis | Weighted average of the positive tweets at time t |
| NegAVG | Sentiment Analysis | Weighted average of the negative tweets at time t |
| NotrAVG | Sentiment Analysis | Weighted average of the neutral tweets at time t |
| PosSTDEV | Sentiment Analysis | Standard deviation of the positive tweets at time t |
| NegSTDEV | Sentiment Analysis | Standard deviation of the negative tweets at time t |
| NotrSTDEV | Sentiment Analysis | Standard deviation of the neutral tweets at time t |

### 4.2. Recurrent Neural Networks

We used LSTM-Recurrent Neural Networks (RNN) to predict the closing price of next trading day. In classical feedforward neural network structures, incoming information is processed only in forward direction without a loop structure. The input data is passed through the network and an output is produced in the final layer. In a feedforward network, it is assumed that the training and test samples are independent from each other. RNN, on the other hand, evaluates inputs based on previous inputs in a sequential manner assuming that there is a sequential or temporal relation between inputs and/or outputs of the network [23]. Figure 2 shows the unfolded structure of a RNNs. The decision for the input at moment "*t-1*" which is shown with $x_{t-1}$, also impacts the decision to be made at time "*t*". So, in these networks, inputs produce output, which is shown with *o* in Figure 2, by combining current and previous information. In Figure 2, $h_t$ stands for hidden state at time-step *t* which is an RNN block that combines the current input with the hidden state of the previous time-step. A non-linear activation function is applied to the hidden state to model non-linear dependencies. The parameters of the network are the weights between input and hidden states, *W*, the weights between previous and current hidden states, *U*, and the weights between the hidden state and output, *V*, as seen in Figure 2. In the stock prediction problem, the current stock price, price at time *t*, highly depends on the previous information represented by the stock prices at time *t-1*, *t-2*, …, *t-k* where *k* represents the length of the previous data that is used for prediction. For such problems, feedforward networks may not produce satisfactory results since they do not take the sequential information into account while building the model. Therefore, we use RNNs to build our stock prediction model.

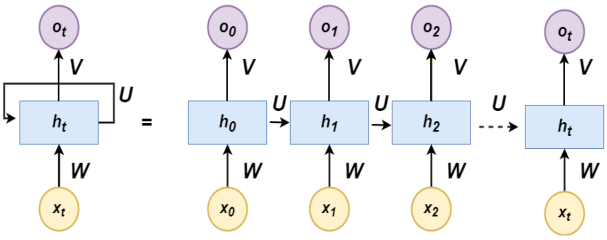

**Figure 2.** Structure of a traditional Recurrent Neural Networks

In theory, RNNs can handle long-term relations in the data. However, it is known that they fail to model dependencies when backpropagating errors across many time steps [23-25]. The related problems are known as vanishing and exploiting gradient problems. To deal with this problem, the units in RNNs have been replaced with some different structures, called memory cell, and an architecture called Long Short-Term Memory-RNNs (LSTM-RNN) [23] has been developed to handle long-term dependencies in sequential learning problems. Each memory cell in an LSTM-RNN has three gates each one dedicated to a particular task. These gates are used to decide how much of the latest content should be stored, how much of the existing memory should be forgotten, and what the next hidden state should be [23].

### 4.3. Proposed Prediction System

The steps of the stock prediction system built in this study are shown in Figure 3. As seen, the inputs of the model consist of sentiment labels, stock prices and values of technical indicators. For the sentiment analysis part, we use manually labeled 500 samples to build a sentiment classification model as described in Section 4.1. We use 10-fold cross-validation to optimize the hyper-parameters of the FastText tool. We also present the sentiment classification accuracies in Section 5.2. The obtained final model is applied to the remaining unlabeled news and sentiment labels for all news are obtained.

As seen in Figure 3, the obtained sentiment labels are combined with stock prices and technical indicators and the combined data is used to predict a numerical output which is the next day's closing price. Then, the difference between the predicted value at time *t* and observed value at time *t-1* is used to decide whether the stock price will be up or down next day.

## V. EXPERIMENTS

### 5.1. Experimental Setup

We have 1892 rows of dataset covering the Istanbul stock exchange data between 06.10.2010 and 01.06.2018. Each row represents one day's stock market attributes. We used the first 1659 days for training and validation and the last 233 days corresponding to the last year for testing. The left-out test set has not been used during training or model selection but only used to evaluate the generalization ability of the model on future examples.

There are various evaluation metrics used to assess the success of machine learning models. Since the first step of our model is to build a regression model that predicts the next day's closing price, we present the actual and predicted values for the test period to evaluate to what extent the actual price can be predicted from the related input. Then, we convert the predicted value to a binary variable indicating whether the movement of the stock price was up or down using the rule given in Section 4.3 and evaluate the results using precision, recall, F1 score, and Matthew's correlation coefficient metrics. We build the prediction model both with and without sentiment labels to evaluate the contribution of the sentiment analysis in the stock movement prediction task.

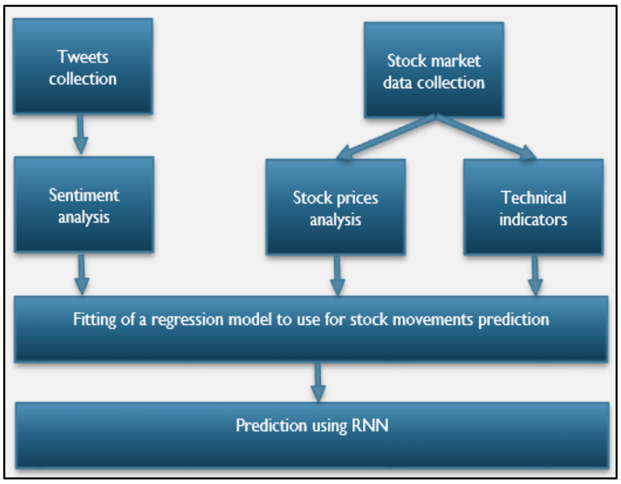

**Figure 3.** Stock price movement prediction architecture

### 5.2. Experimental Results

*5.2.1. Text classification results*

FastText has many hyper-parameters which can be optimized on the training set. We used a grid-search procedure to optimize the related parameters and obtained the best results for the following values: 2 for minn, 4 for maxn 4, 2 for wordNgrams, for epoch, 25, and 1000 for bucket. Overall, the best model gives an accuracy of 98% on the training set and 90.0% on the validation set. Both precision and recall of the best model have been obtained as 0.90 on the validation set. We then applied the best model to the remaining 54,841 news and obtained the probability estimates of each sentiment label. Figure 4 shows the distribution of the negative and positive news labeled by the text classification model in the related period. As it is seen in Figure 4, the number of news increases in time with the wider use of online platforms and social media. As a result of sentiment analysis, 7118 news in the related period were marked as negative, 9548 news positive and 38675 news labeled as neutral summing up to 55,341 news including also manually labeled 500 news.

*5.2.2. Stock price prediction using price data*

Table 2 shows the behavior of root mean squared errors under different epochs of the LSTM-RNN trained with only the historical stock price data. The results show that RMSE decreases up to a point and gets its minimum value at around 200 epochs. After this point, the learning process has become stable and increasing of number of epochs no longer helps to generalize better on the validation set. It takes around 15 minutes to train the network for 200 epochs. The optimal timestep of the LSTM-RNN model has been found as 7. Thus, the next day's stock price prediction has been generated using the previous 7 trading day's historical stock prices. The LSTM-RNN architecture contains two LSTM layers with 60 LSTM cells in each. We used root mean squared prop (RMSprop) as the optimization algorithm [26].

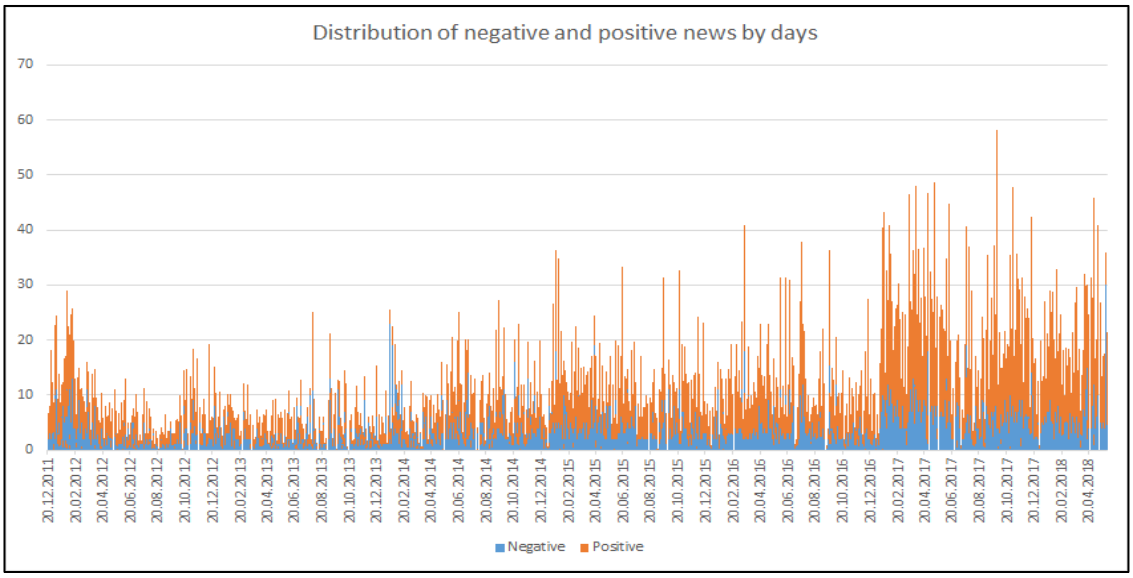

**Figure 4.** Distribution of negative and positive news by days

**Table 2.** RMSE on the validation set with respect to increasing number of epochs

| # of epochs | RMSE on Validation Set | Duration |
|---|---|---|
| 16 | 0.002474 | 00:01:22 |
| 32 | 0.001028 | 00:02:17 |
| 64 | 0.0007549 | 00:04:44 |
| 72 | 0.0007589 | 00:05:04 |
| 100 | 0.0005417 | 00:07:18 |
| 124 | 0.0006015 | 00:08:54 |
| 170 | 0.0005116 | 00:12:42 |
| 200 | 0.0005211 | 00:15:40 |
| 234 | 0.0005775 | 00:16:41 |
| 400 | 0.0005213 | 00:28:52 |

### *5.2.3. Combining multiple datasets*

We constituted two different combined datasets to evaluate the contribution of integrating sentiment analysis into the prediction model. The first dataset consists of 13 features representing the historical stock prices and values of technical indicators. In the second dataset, as given in Section 4.1, there are 6 additional features extracted from the sentiment analysis of the related news belonging to the same time period. We gave both datasets to LSTM-RNN as input to predict the next day’s stock price movement.

Figure 5 shows the predicted and actual prices obtained with the combined datasets for the test data which consists of the last year’s trading days. As it is seen in Figure 5, the second model that is constructed using sentiment analysis along with the stock prices and technical indicators fits better to the actual data points than the first model constructed without sentiment analysis. It is also seen from the circled parts of the input space, especially when there is a high volatility in the stock prices, sentiment analysis helps the model to adapt to the sudden changes in the prices.

Table 3 shows the results obtained by feeding combined dataset as input to LSTM-RNN. We should note that the number of class 1 (up) and class 0 (down) is 120 and 113, respectively, on the test set. In addition to the two models obtained with and without sentiment information, we built another model that uses only stock prices and two of the technical indicators, RSI14 and MA10, to test whether using all technical indicators as input helps LSTM-RNN to model dependencies better. The obtained results seen in Table 3 showed that Model 2, which was built using all technical indicators, gave better performance than Model 1 that uses only two of the technical indicators.

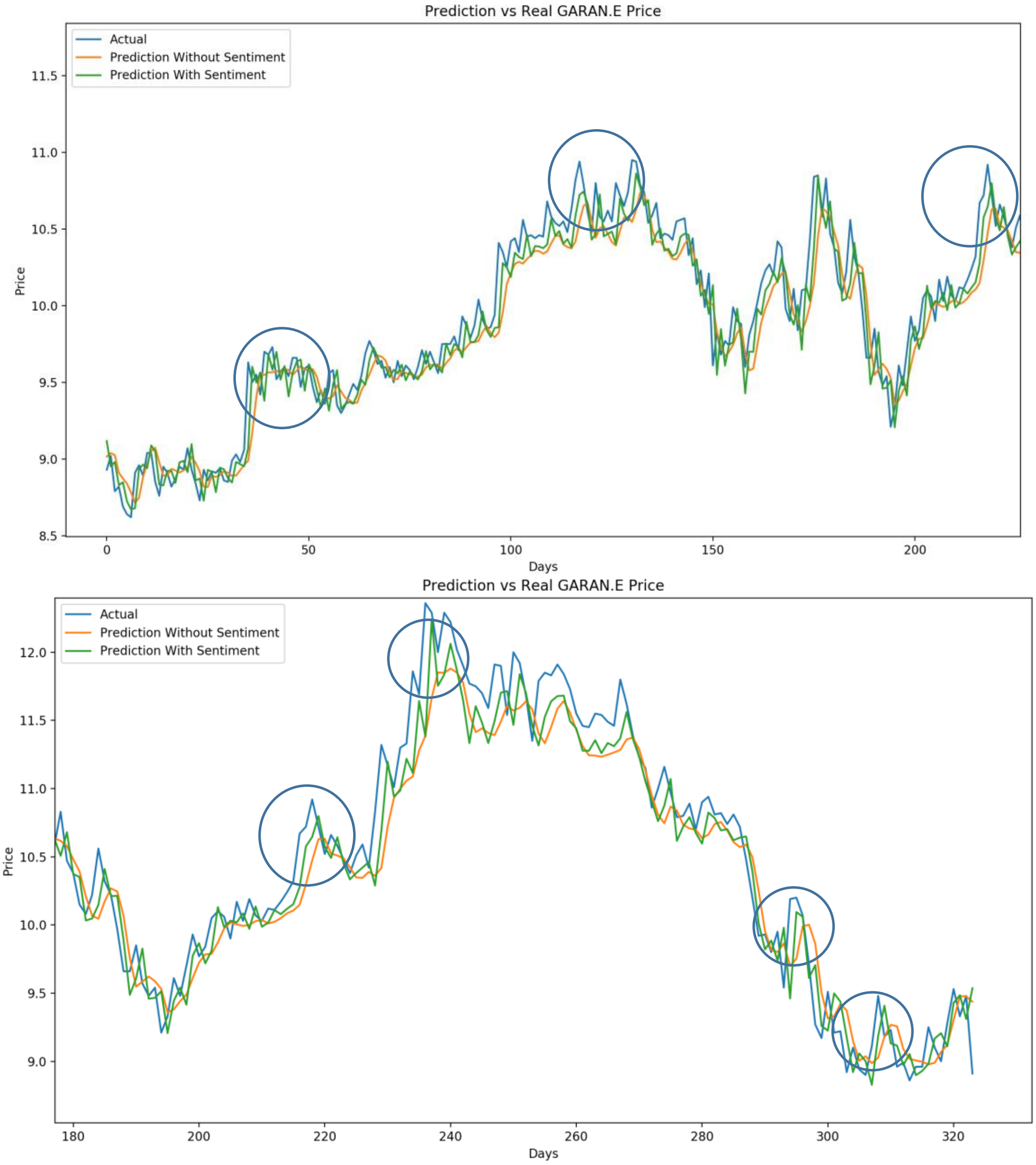

**Figure 5.** Actual stock prices and predictions with and without sentiment analysis for (top) first half (bottom) second half of the test year period

As seen in Table 3, while F1 score of Model 2 was 0.55, it has increased to 0.65 in Model 3, in which sentiment analysis label information is integrated into the input. Model 2 gives 71 true positive movements and 59 true negative movements on the test data whereas Model 3 achieved 83 true positive movements and 70 true negative movements. It is also seen that Matthews correlation Coefficient (MCC) of the model using stock price, technical indicators, and sentiment information together is higher than that of the model using only stock price and technical indicators. The results indicate that the financial news includes important information about the direction of the stock and integrating this information with the help of sentiment analysis into the model with data level fusion increases the success of the model in predicting both up and down price movements. We have also trained the best model, Model 3 shown in Table 3, with changing number of training samples to see the effect of training set size in the model performance. For this purpose, we used the same cross-validation methodology and showed F1 score of the model with respect to the number of yearly data in Figure 6. We should note that, to generate this figure, to keep the sequential information during the learning process we have always used 8th year data as test set and used last *n* years for training where *n* is increased from 1 to 7. As it is seen in Figure 6, in general there is an increasing trend in F1 score with increasing number of trading days used for training. The highest average F1 score with 0.65 was obtained using last 6 or 7 years for training.

**Table 3.** Results obtained using technical indicators and sentiment analysis as input for LSTM-RNN model

| Model | | Recall | Precision | F1 score | MCC |
|---|---|---|---|---|---|
| **Model 1**<br>Stock Price + RSI14 and MA10 | 1 (Up) | 0.55 | 0.56 | 0.55 | 0.06 |
| | 0 (Down) | 0.51 | 0.51 | 0.51 | -0.04 |
| | Average/Total | 0.53 | 0.53 | 0.53 | - |
| **Model 2**<br>Stock Price + All Technical Indicators | 1 (Up) | 0.57 | 0.59 | 0.57 | 0.13 |
| | 0 (Down) | 0.55 | 0.52 | 0.53 | -0.01 |
| | Average/Total | 0.56 | 0.56 | 0.55 | - |
| **Model 3**<br>Stock Price + All Technical Indicators + Sentiment Analysis | 1 (Up) | 0.66 | 0.69 | 0.67 | 0.34 |
| | 0 (Down) | 0.65 | 0.62 | 0.63 | 0.20 |
| | Average/Total | 0.65 | 0.66 | 0.65 | 0.27 |

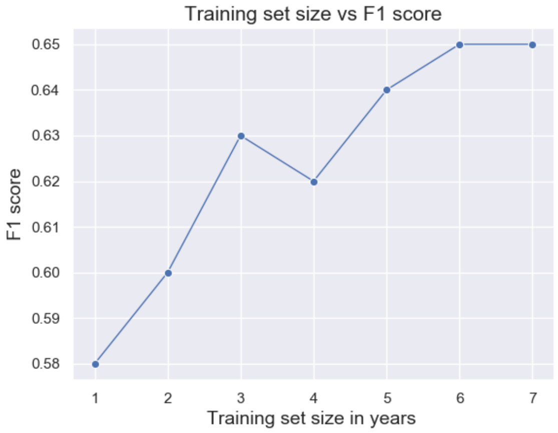

**Figure 6.** Average F1 score of the model built using stock prices and sentiment labels with respect to increasing training set size

## VI. CONCLUSIONS

The number of investors wishing to utilize their savings in the stock market is rapidly growing day by day. Therefore, recently there is an increasing interest in building stock prediction models using machine learning techniques. The researchers also integrate different data sources into the stock price estimation model to improve the accuracy of the models.

The main goal of this study was to assess the effect of using financial news as an additional data source on the success of the stock price-based estimation models. For this purpose, we used Long Short-Term Memory Recurrent Neural Network (LSTM-RNN) to estimate the next day's closing stock price. We collected a news dataset in Turkish language and assigned a sentiment label to each of the news using sentiment analysis tools. Then, the output of the sentiment analysis model was given to the regression model together with the stock price-based features to predict the future price of a stock from Istanbul Stock Exchange. The results of this regression model showed that, even with sentiment analysis inputs, estimating the exact future price of a stock is a difficult task due to the inherent chaotic nature of market or lack of some other important hidden factors that are not included in our dataset. Therefore, we modelled the estimation problem as a binary classification problem and aimed to predict the movement direction of the stock price, i.e. whether the closing price of the stock next day will be higher than today's price. The results showed that classifying economic and political news as positive, negative, or neutral and integrating these values to the model as new inputs increases the success of the stock price movement prediction. The F1 score of the LSTM-RNN model built by using stock prices and technical indicators was measured as 0.56 on test set. When we use sentiment analysis labels of the news as additional inputs, the F1 score of the model increased to 0.65. The obtained results also showed that the performance of the prediction model increases with increasing number of trading days used for training.

Our results showed that with the sentiment analysis information the actual price is predicted better especially when there is a high volatility in the stock price. Based on this finding, as a future work, the proposed model can be used to predict the prices of the stocks with higher volatility. As another future direction, in addition to financial and political news, the reviews and comments of the users in social media platforms can be analyzed and integrated to the model. In this study, we aimed to perform a daily prediction model. In order to build a model that can make shorter-term predictions, the characteristics of an order book may be combined with the inputs used in this study. The analysis of the ITCH market data feed and the information such as the quantity, size, count, side and frequency of orders in every order book level on a specific stock may give us more accurate results for short-term estimation of stock price movement.

## REFERENCES

**[1]** Henrique, B. M., Sobreiro, V. A., & Kimura, H. (2019). Literature review: Machine learning techniques applied to financial market prediction. *Expert Systems with Applications*, 124, 226-251.

**[2]** Gandhmal, D. P., & Kumar, K. (2019). Systematic analysis and review of stock market prediction techniques. *Computer Science Review*, 34, 100190.

**[3]** Akita, R., Yoshihara, A., Matsubara, T., & Uehara,

K. (2016, June). Deep learning for stock prediction using numerical and textual information. *In 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)* (pp. 1-6). IEEE.

**[4]** Chong, E., Han, C., & Park, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications*, 83, 187-205.

**[5]** Yoshihara, A., Fujikawa, K., Seki, K., & Uehara, K. (2014, December). Predicting stock market trends by recurrent deep neural networks. *In Pacific rim international conference on artificial intelligence* (pp. 759-769). Springer, Cham.

**[6]** Nguyen, T. H., Shirai, K., & Velcin, J. (2015). Sentiment analysis on social media for stock movement prediction. *Expert Systems with Applications*, 42(24), 9603-9611.

**[7]** Vargas, M. R., De Lima, B. S., & Evsukoff, A. G. (2017). Deep learning for stock market prediction from financial news articles. In 2017 *IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)* (pp. 60-65). IEEE.

**[8]** Pagolu, V. S., Reddy, K. N., Panda, G., & Majhi, B. (2016, October). Sentiment analysis of Twitter data for predicting stock market movements. *In 2016 international conference on signal processing, communication, power and embedded system (SCOPES)* (pp. 1345-1350). IEEE.

**[9]** Liu, Y., Qin, Z., Li, P., & Wan, T. (2017). Stock volatility prediction using recurrent neural networks with sentiment analysis. *In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems* (pp. 192-201). Springer, Cham.

**[10]** Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5, 135-146.

**[11]** Hövelmann, L., Allee, S., & Friedrich, C. M. (2017). Fasttext and gradient boosted trees at GermEval-2017 on relevance classification and document-level polarity. *In Proceedings of the GermEval 2017 - Shared Task on Aspect-based Sentiment in Social Media Customer Feedback*, (pp. 30-35).

**[12]** Lin, J. W., Gao, Y. C., & Chang, R. G. (2019, February). Chinese Story Generation with FastText Transformer Network. *In 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)* (pp. 395-398). IEEE.

**[13]** Santos, I., Nedjah, N., & de Macedo Mourelle, L. (2017). Sentiment analysis using convolutional neural network with fastText embeddings. *In 2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI) (pp. 1-5).* IEEE

**[14]** Kuyumcu, B., Aksakalli, C., & Delil, S. (2019). An automated new approach in fast text classification (fastText) A case study for Turkish text classification without pre-processing. *In Proceedings of the 2019 3rd International Conference on Natural Language Processing and Information Retrieval* (pp. 1-4).

**[15]** Uygur, U., & Taş, O. (2014). The impacts of investor sentiment on different economic sectors: evidence from Istanbul stock exchange. *Borsa Istanbul Review*, 14(4), 236-241.

**[16]** Canbaş, S., & Kandır, S. Y. (2009). Investor sentiment and stock returns: Evidence from Turkey. *Emerging Markets Finance and Trade*, 45(4), 36-52.

**[17]** Li, J., Bu, H., & Wu, J. (2017). Sentiment-aware stock market prediction: A deep learning method. *In 2017 international conference on service systems and service management* (pp. 1-6). IEEE.

**[18]** Zhuge, Q., Xu, L., & Zhang, G. (2017). LSTM Neural Network with Emotional Analysis for prediction of stock price. *Engineering letters*, 25(2).

**[19]** Day, M. Y., & Lee, C. C. (2016). Deep learning for financial sentiment analysis on finance news providers. *In 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)* (pp. 1127-1134). IEEE.

**[20]** Weng, B., Ahmed, M. A., & Megahed, F. M. (2017). Stock market one-day ahead movement prediction using disparate data sources. *Expert Systems with Applications*, 79, 153-163.

**[21]** Wilder, J. W. (1978). New concepts in technical trading systems. Trend Research.

**[22]** Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., & Joulin, A. (2017). Advances in pre-training distributed word representations. *arXiv preprint* arXiv:1712.09405.

**[23]** Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780.

**[24]** Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2), 157-166.

**[25]** Hochreiter, S., Bengio, Y., Frasconi, P., and Schmidhuber, J. (2011a). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies, In Kremer, S. C. and Kolen, J. F., editors, A Field Guide to Dynamical Recurrent Neural Networks. IEEE Press.

**[26]** Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint* arXiv:1609.04747.