Abstract
Is the concept of stock market speculations, related with the news in the newspapers? This study mainly focus on the correlation between economy news from one of the highest circulation rate news paper in Turkey and Istanbul stock market closing values. Data set is collected from the web page of news paper in natural language and text mining technique, term frequency – inverse document frequency is applied over these news. On the other hand the stock market values are evaluated as a signal processing job and random walk method has been applied on it. The two feature vectors are correlated with several classification algorithms such as support vector machines, k-nearest neighborhood and artificial neural networks. The results show that there is a weakly relation over 43% between the news and stock market closing values. We believe this research would be beneficiary for the literature to create some stock market estimation tools from the economy news or market strength analysis.

Categories and Subject Descriptors

General Terms
Data Mining, Financial Data Processing, Stock Market Analysis

Keywords
Data Mining, Big Data, Stock Market Analysis, SVM, ANN, KNN, Random Walk, Text Mining

1. INTRODUCTION
This study is built on one of the highly circulating news papers in Turkey which has special pages for economy news. We have collected only these economy news which are free from other news like sports or magazine etc. The properties of the data set will be explained in the experiments section. We have processed the news text via text mining approach called term frequency – inverse document frequency (TF-IDF) which will be explained in the methodology section. On the other hand we have processed the stock market closing values by using a signal processing approach, the random walk (RW). Finally we have investigated the correlation between these two feature vectors by using the support vector machines (SVM), k-nearest neighborhood (KNN) and artificial neural networks (ANN) which are discussed in the section of classification. Also this paper holds the implementation details and the methodology of evaluation over the classification results which are held in the evaluation section.

2. PROBLEM STATEMENT
This study can be categorized as a correlation study built on text mining and signal processing studies. There are two major feature extraction methodologies implement on two data sets which are economy news and stock market closing values and methos are TF-IDF and RW respectively.

Figure 1. Overview of Study
The correlation between news and stock market is one of the indicators of the speculative markets[1].

One of the difficulties in this study is the dealing with natural language data source which requires a feature extraction. The other difficulty is the dealing with a stock market value which is
considered as a signal. On the other hand big data we are dealing with is also problematic. The data set holds 131,248 distinct words and when the feature vector of each economy news is collect the total size of feature vector is beyond 2.5 GByte which is beyond the computation capacity of a single computer with these classification algorithms.

3. Related Work

Researches about stock markets have always been an interesting area of study because of its impact on the business and finance world. Furthermore the developments in computer science field has opened a door to research on the stock markets parallel to the text based news after 2000s. Most of the works are using text mining tools over the news. For example one of the famous method is using the bag of words [2-6] where some uses local dictionaries [2,6] or some uses some text mining tools like IBM Text Miner[5] and some uses the TF-IDF approach [3,6] or some pre-defined term dictionaries [2] and some uses part of speech tagging [9] and concept maps [8]. Also classification varies from latent dirichlet allocation [9] to SVM [3-8] or decision tree [2]. The success rates varies from 45% to 82% and the data sources are reuters market 3000 extra [5], PRNewswire [6], FT Intelligence [8], Australian Financial Review [3], Forbes and Reuters web sites [2], Yahoo Finance [4] or Wall Street Journal [9].

Our aim is putting a new data set into the literature and study the correlation between the new data set and stock market values in Turkey where further researches can be placed for the market structure and strength in the future.

4. Background

We have implemented TF-IDF and RW methods as already explained in the introduction, this section will discuss these methods in detail. Also one of the difficulties is the number of words we are dealing with. We have implemented the information gain calculation for eliminating some of the features.

4.1. Term Frequency – Inverse Document Frequency

For the TF-IDF calculation is given in equation (1).

\[ tfidf(t, d) = tf(t, d)x idf(t, D) \]  

Where t is the selected term, d is the selected document and D is the all documents in the corpus. Also TF-IDF calculation in above formula is built over term frequency (TF) and inverse document frequency (IDF), which can be rewritten as in equation (2).

\[ tf(t, d) = \frac{f(t, d)}{\max \{ f(w, d) : w \in d \}} \]

where f is the frequency function and w is the word with maximum occurrence. Also the formulation of idf is given in equation (3).

\[ idf(t, D) = log \frac{|D|}{|\{d \in D : t \in d \}|} \]

where |D| indicates the cardinality of D, which is the total number of documents in the corpus.

4.2. Stock Market Closing Values

The stock market closing values are collected from the web page of the Istanbul Stock Market which is available for public download and usage. The values are in local currency which is Turkish Lira and we don’t care about these values. Instead we use a relative feature extraction method and get the difference between two consecutive closing values like for any \( C_i \in C \) where C is the set of closing values, we collect pairs of \( <C_{i-1}, C_i> \) for each news where \( i \) is the publication date of the news. The collected features than subtracted each other to see if there is an increase or decrease on the closing value of the stock market as in equation (4).

\[ Feature = C_i - C_{i-1} \] (4)

If the feature value of the closing values of the stock market is in positive (+) we consider it as 1 if the value is negative (-) we consider it as -1. Also there are some dates which do not hold the closing values such as week ends. For those dates we consider the class label of the news as 0.

This approach can be considered as random walk in the literature. The formulation of random walk is given as in equation (5).

\[ s_n = \sum_{j=1}^{n} Z_j \] (5)

where \( Z_j \) is the random event and the initial value of walk starts by 0 where \( S_0 = 0 \). The n in the equation (5) determines the length of the walk and in our study the length of walk is limited with only 2 for each cases.

4.3. Information Gain

The information gain of all the terms are calculated and ordered in descending order. Let \( Attr \) be the set of all attributes and \( E_i \) be the set of all training examples, \( value(x, a) \) with \( x \in E_i \) defines the value of a specific example x or attribute \( a \in Attr \), \( H \), specifies the entropy. The information gain for an attribute \( a \in Attr \) is defined as in equation (6).

\[ IG(Ex, a) = H(Ex) - \sum_{v \in v(a)} \frac{|x \in Ex|v(x, a)|}{|Ex|}H(x \in Ex|v(x, a)) \] (6)

Also entropy in the information gain calculation can be rewritten as in equation (7).

\[ H(X) = \sum_{i=1}^{n} P(x_i)log(P(x_i)) = \sum_{i=1}^{n} P(x_i)log_b \left( \frac{1}{P(x_i)} \right) \] (7)

4.4. Artificial Neural Network

Purpose of ANN studies is adapting the biological neural networks into data processing. Multi-layer perceptron (MLP) is a developed version of ANN which organizes the neurons into layers as input, output or hidden layers. Figure 6 demonstrates the approach in a data flow between layers [11]. The ANN model has been one of the attractive tools used in geo-engineering applications due to its high performance in the modeling of non-linear multi-variate problems. Hecht-Nielsen [12] and Schalkoff [13] indicate that an ANN may be defined as a structure comprised of densely interconnected adaptive simple processing elements that are capable of performing massively parallel computations for data processing and knowledge representation.
The input layer is directly connected to the inputs which will be processed within the system. Hidden layer can hold more than one layer depending on the complexity of the system and finally the output layer holds the results. Figure 6 can be detailed if each of the layers is demonstrated by real number of neurons. Figure 7 is a detailed version of Figure 2 with multiple neurons in each of the layers. Please note that, each neuron on a layer is directly connected to all the neurons on the next layer.

Figure 4. Visualization of sigmoid function

Each of the connections in Figure 3, which are called as synapsis in a biological view, has a weight value, which is a coefficient to the value on the neuron which is indicated by “wij”, where i and j indices are the weight value between the neurons i and j [14].

Aim of this study is calculating the best possible values of the above weights. During this calculation the back propagation algorithm is implemented. The algorithm, changes the weight values on the synapses by the result achieved on the output layer.

Most of the ANN systems use the sigmoid function for the decision of firing the neuron [15]. The generic formula of the sigmoid function is given in equation (1). Mitchell uses the word "logistic function" and the "sigmoid function" synonymously—he also calls this function as "squashing function"—and the sigmoid (aka logistic) function is used to compress the outputs of the "neurons" in multi-layer neural nets. The generic two dimensional view of the function is given in Figure 8.

$$f(x) = \frac{1}{1 + e^{-x}}$$ (8)

Besides the above ANN information, during this study the two parameters of ANN have crucial roles, which are learning rate and the momentum factor.

Learning rate of the ANN is denoted by $\alpha$, which varies from 0 to 1. Learning rate is a parameter to control the amount of weight adjustment at each step of training to determine the learning rate at each step. It affects the convergence of back propagation network (BPN). A large value of $\alpha$ indicates the faster learning with high probability of error, while the lower learning rate indicates slower but robust learning.

Second crucial parameter of ANN is the momentum, which is denoted by $\eta$ and varies from 0 to 1. The momentum is used to speed up the process. BPN has two major disadvantages which are larger training time and slow convergence. Momentum parameter helps to improve the convergence and the training time while reducing the oscillation of learning.

4.5. Support Vector Machine (SVM)

The reason of applying SVM method as in Figure 5 over the dataset is determining the boundaries between classes [25].

$$\omega \cdot \sum_{i=1}^{n} \alpha_i y_i x_i$$ (9)

The margin between the classes is symbolized by $\omega$ symbol in equation (9) and SVM seeks to maximize the value of $\omega$. The above formula can be rewritten as below for the linearly separable classes [26].

$$\|\omega\|^2 = \sum_{i=1}^{J} \alpha_i = \sum_{i=1}^{J} \sum_{i=1}^{J} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$$ (10)

In the equation (10), all the possible cases of $i$ and $j$. Also SVM can use a radial basis function and one of the options is the Gaussian kernel function, quoted in equation (11) [26].
Finally, the class is determined by the result achieved from K function.

4.6. K-Nearest Neighborhood (KNN)
The k, c-neighborhood (or k, c(x) in short) of an U-outlier x is the set of k class c instances that are closest to x (k-nearest class c neighbors of x).

$$K(x, x') = \exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right).$$ (11)

4.7. Error Rate Calculation
The error rate of the system is calculated through root mean square error (RMSE). The calculation of RMSE is given in equation (13) [28].

$$x_{\text{rmse}} = \sqrt{\frac{\sum_{j=1}^{n}(P_j - T_j)^2}{n}}$$ (13)

For this study, above $X$ values are the results achieved from the implementation of the algorithm. The RMSE result of 0 is considered ideal and lower values close to 0 is relatively better.

By the results fetched from the output layer and the calculation of RMSE, the algorithm back propagates to the weight values of the synapses.

Also the results are interpreted by using a second error calculation method RRSE (Root Relative Squared Error) and the calculation is given in equation (14) [17].

$$x_{\text{rrse}} = \sqrt{\frac{\sum_{j=1}^{n}(P_j - T_j)^2}{\sum_{j=1}^{n}(T_j - \bar{T})^2}}$$ (14)

where $P_j$ is the value predicted for the sample case $j$, $T_j$ is the target value for sample case $j$ and $\bar{T}$ is calculated by equation (15) [17].

$$\bar{T} = \frac{1}{n} \sum_{j=1}^{n} T_j$$ (15)

The RRSE value ranges from 0 to $\infty$, with 0 corresponding to ideal.

The third error calculation method is RAE (Relative Absolute Error) and the calculation is given in equation (16) [17].

$$E_i = \frac{\sum_{j=1}^{n} |P_{ij} - T_j|}{\sum_{j=1}^{n} |T_j - \bar{T}|}$$ (16)

$P_{(ij)}$ is the value predicted by the individual program i for sample case j (out of n sample cases), $T_j$ is the target value for sample case j, and $\bar{T}$ is given by the equation (17) [17]:

$$\bar{T} = \frac{1}{n} \sum_{j=1}^{n} T_j$$ (17)

For a perfect fit, the numerator is equal to 0 and $E_i = 0$. So, the $E_i$ index ranges from 0 to infinity, with 0 corresponding to the ideal.

Also the success rate of prediction and expectation can be measured as the f-measure method. F-measure method is built on the Table 1.

<table>
<thead>
<tr>
<th>Predictions</th>
<th>Positive</th>
<th>Negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expectations</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td></td>
<td>False Positive</td>
<td>True Negative</td>
</tr>
</tbody>
</table>

The calculation of f-measure can be given as in equation (18) depending on the Table 1.

$$F_{\text{measure}} = \frac{2TP}{2TP + FN + FP}$$ (18)

5 EXPERIMENTS
In this study the data set is in natural language and some preprocessing for the feature extraction from the data source is required. The first approach is applying the TF-IDF for the all terms in the data source. Unfortunately the hardwares in the study environment was not qualifying the requirements for the feature extraction of all the terms in data source which is 139,434.
5.1. Data Set
We have implemented our approach and Table 2 demonstrates the features of the datasets.

<table>
<thead>
<tr>
<th>Table 2. Properties of the Data Set</th>
</tr>
</thead>
<tbody>
<tr>
<td># of News</td>
</tr>
<tr>
<td>Authors</td>
</tr>
<tr>
<td>Texts per Author</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Average word length</td>
</tr>
</tbody>
</table>

Above data set is collected from the web site of a highly circulating news paper in Turkey. The data is collected directly from database so the noisy parts on the web page like ads, comments, links to other news etc. are avoided. Another problem is the noise of HTML tags in the database entries for formatting the text of news. The data has preprocessed and all the HTML tags are removed from the news and also all punctuations and stop words are removed in the preprocessing phase.

5.2. Feature Extraction
We have implemented a feature extraction algorithm 1 in order to extract two feature vectors.

**Algorithm: Feature Extraction Methods**

1. Let E be Economy News Corpus.
2. Let C be Closings of Stockmarket.
3. For each Ei ∈ E
4. For each Termi ∈ Ei
5. if(count(Termi)>30)
6. Ti ← TF-IDF of Termi
7. Ci ← closing_value(date(Ei)) ∈ C
8. IGj ← Information Gain (Termi,Ei)
9. Vj ← Top300(sort(IG))
10. Vj ← C

Above algorithms demonstrates the extraction of two vectors, one from the economy news corpus and another from the closing values of the stock market. We have limited the number of features by 300 and the Top300 function gets the topmost 300 features from the feature vector.

The V1 feature vector is calculated easily by checking the closing value of the economy news on the date. There are some news which are published during the stock market is closed like in the weekends and we have considered these values as a third class besides the increase and decrease classes.

The correlation algorithms run over the two vectors V1 and V2 extracted via the Algorithm 1.

During the execution of algorithm, the execution requires more memory than the available hardware, where we run the algorithms on a intel 7 cpu and 8GByte of RAM. The required memory is calculated in equation (19).

Memory Requirement = 139,434 words x 9871 news x 6.7 average word length x 4 bytes for each character
= ~34GByte

As a solution we have limited the number of words with the highest occurrences. The number of occurrence on our implementation is 30 and a word is taken into consideration after this number of occurrence. The words appearing above this threshold value is 2878 and the memory required is reduced to 700MByte which is more easier to handle in the RAM.

The feature vector extraction is about 56 minutes in average for the economy news.

5.3. Classification
The results of executions can be summarized in Table 3.

<table>
<thead>
<tr>
<th>Table 3. Error and Success rates of Classification Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>F-measure</td>
</tr>
<tr>
<td>Average</td>
</tr>
<tr>
<td>RAE</td>
</tr>
<tr>
<td>Success</td>
</tr>
</tbody>
</table>

The success rate on Table 3 is the percentage of correctly classified instances. For example the success rate of SVM can be considered as the 43% of the instances are correctly classified to predict an increase, decrease or no change on the stock market value depending on the economy news processed.

The value of success is highly related with the market structure so the success rate here should not be understood as the success rate of the methodology or the classifier. The success rate on the table is the correlation between economy news and the stock market closing values.

6. CONCLUSION
In this paper, we have tried to study the correlation between a stock market and economy news first time for Turkey case. The correlation is strongly related with the market structure and how it is speculative depending on the economy news. The results show that for the best classifier we have applied, there is a 43% correlation between these two data sets.

During this study, we have also dealt with text mining over big data and feature reduction because of the physical limitation and signal processing approach over the stock market.

We believe the results would be beneficiary for further research on the finance and economy fields as well as the data mining field. As a future work more classifiers can be added or an ensemble approach can be applied over the classification phase and also some new feature extraction methods can be applied like using part of speech tagging and so on.

7. REFERENCES


