Quality Controlled Stock Prediction Model

Shawket Guirguis*, Fatma Zada** and Tawfik Khattab

Abstract - **This paper is the first attempt to improve the quality of investing in the highly volatile Egyptian Stock Exchange (ESE) by combining the concepts of statistical process control and artificial intelligence. Control charts were used to construct a statistically controlled stock market prediction model to support the decision of stock investors. The suggested model is mainly based on the concepts of Case-based Reasoning (CBR) which is an artificial intelligent methodology that imitates the human problem-solving and reasoning behavior. Hit rate was applied as a performance measure of the quality of prediction for the suggested model. Results of predicting 900 next day stock predictions during January 2012 had a mean absolute prediction error of 2.096 LE and a hit ratio of 67%. After using the quality controlled process, the mean absolute prediction error was reduced to 1.92 L.E. and the hit ratio increased to 72%.**

I. INTRODUCTION

Stock Market Predictions have always been the focus of many investors and researchers from numerous fields such as economics, psychology, statistics and more recently artificial intelligence. The high dimensionality, noisiness, and non-stationary characteristics of stock prices have made the problem of predicting future stock prices of extreme complexity. Movement of stocks is influenced by numerous known and unknown microeconomic, macroeconomic, political and other events. This problem further complicates within highly volatile emerging markets such as the Egyptian Stock Exchange (ESE).

This paper establish a novel stock price prediction model that combines the concepts of quality control and artificial intelligence aiming to improve the quality of investing within the ESE to support the decision of stock investors. The forecasting model implements the Case-based Reasoning (CBR) artificial intelligent and statistical quality control methodologies. CBR imitates the human problem-solving and reasoning behavior. When humans are faced with uncertainty, they make decisions by analogy and use their experience in previously encountered cases to construct and justify arguments for new similar cases (Lenz, 1999).

II. Background & Related Work

Reasoning in artificial intelligence is often modeled as a process of drawing conclusions by chaining generalized rules. Case-based reasoning (CBR) takes on a different approach by using knowledge of previously experienced situations in the form of cases rather than relying on general knowledge in the form of rules (Lenz, 1999). Motivation for CBR came from modeling of the human problem-solving and reasoning behavior. Humans are robust problem-solvers; who solve hard problems despite incomplete and uncertain knowledge, and their problem-solving competence im-

proves with experience. When faced with uncertainty, humans make decisions using analogy rather than using probabilistic models.

Related work was conducted in 2004, where Kim suggested a simultaneous optimization method of a case-based reasoning (CBR) system using a genetic algorithm (GA) for stock market predicting. In 2007, Sushmita and Chaudhury presented a framework for using a case-based reasoning system for stock analysis in financial market. The unique aspect of their research was the use of a hierarchical structure for case representation. In 2009, Chang, Liu, Lin, Fan, and Ng claimed to make the first attempt in the literature to predict the sell/buy decisions instead of stock price itself. They proposed an integrated system for stock trading prediction by combining dynamic time windows, case based reasoning (CBR), and neural networks. In 2011, Srinivasan, Singh and Kumar proposed a conceptual framework for a Decision Support System (DSS) which is based on Multi-Agent system (MAS) utilizing Data Mining (DM) and Case-base Reasoning (CBR) techniques. In 2011, Chang, Fan and Lin took a different approach to predict stock price movements. They proposed a novel case based fuzzy decision tree (CBFDT) model by applying a step-wise regression (SRA) method to select the most important factors from the set of input features.

III. DATA & METHODOLOGY

Data used in this research were the historical daily stock records of the ESE from the 1st of January 1998 to the 30th of April 2012 for 218 distinct Egyptian stocks. A total of 293,131 records were collected. Data was downloaded from the Egyptian Mubasher Data Service (www.mubasher.info) in the form of comma delimited text files. Daily stock records included the following features: date, open price, low price, high price, close price, volume and the CASE30 index. The CASE30 index is the most widely known performance indicator of the change of value of the most active 30 ESE stocks (El Telbany, 2004). To account for the interdependence of the ESE with major stock markets of the US and UK, S&P 500 daily historical records were also downloaded from finance.yahoo.com from 1st of January 1998 to 30th of April 2012. All stock records from comma delimited text files were imported in the MYSQL database. Programming script in hypertext Preprocessor (PHP) was coded to compute differentiated and normalized features. These features were added to the collected raw features and stored in one main stock record table.

A successful forecasting system relies on the quality of its data. To ensure the quality and integrity of used data, errors and inconsistent data were detected and removed. Records that included mistyped values due to human entry error were detected and removed. Entries due to human recording error were identified by using descriptive z-values to detect outliers within each stock feature and then removed from database. After data cleaning and removing outliers 24,349 records were removed due to missing and wrong entries and 268,782 records were used.

The quantitative research methodology was utilized within this research. Historical daily stock records were first analyzed using statistical descriptive and correlation analysis. Next, the CBR and SPC methodologies were utilized to construct a quality controlled artificial intelligent stock price prediction model. This model in implemented and tested within the highly volatile Egyptian Stock Exchange (ESE). The CBR methodology was recently used in predicting stock prices in Kim (2004), Li and Ho (2009), Li and Sun (2009), Chang, Liu, Lin, Fan, and Ng (2009) and Chang, Fan and Lin (2011). The SPC methodology was used to monitor and control the process of stock price prediction. Control charts plot the normal operation data of the prediction process in order to check for violations that are outside the control limits. The Statistical Process Control (SPC) methodology was used to control the prediction of stock prices in Wheeler and Chambers (1992).

IV. QUALITY CONTROLLED STOCK PREDICTION MODEL

Eddington (1928) stated that there are two types of causes for process variation: assignable causes and common causes. Common (or chance) causes are unpredictable and considered an intrinsic part of the process. He defined common causes as "*Something unknown is doing something we don't know what."* Common causes of variation are random causes that cannot be identified (Reid and Sanders, 2010). They are unavoidable and could only be removed by changing the process because they are results of the process itself (Deming, 2000). A certain amount of common or normal variation occurs in every process due to differences in materials, workers, machines, and other factors. On the other hand, assignable (or special) causes of variation can be predictable and once detected can be removed. For example poor quality in raw materials would increase variation within the production process, but once this cause is detected it can be removed and thus return to natural process variation (Reid and Sanders, 2010).

The suggested stock prediction model integrates the CBR cyclic process of Aamodt & Plaza (1994) with SPC control chart. The quality control process is appended to the four CBR processes as shown in Figure 1. The case base contains all available historical stock cases. The retrieval process finds the most similar historical case to the query stock case. The reuse process formulates the predicted next day stock prices. The revise process validates the recommended predictions through feedback from the user. The control process ensures that prediction accuracies are still within acceptable range and due to common causes and not special causes. Finally, the retain process readjust case feature weights used within the retrieval process and number of K-nearest neighbors (KNN) neighbors used within the reuse process based on computed prediction accuracies.

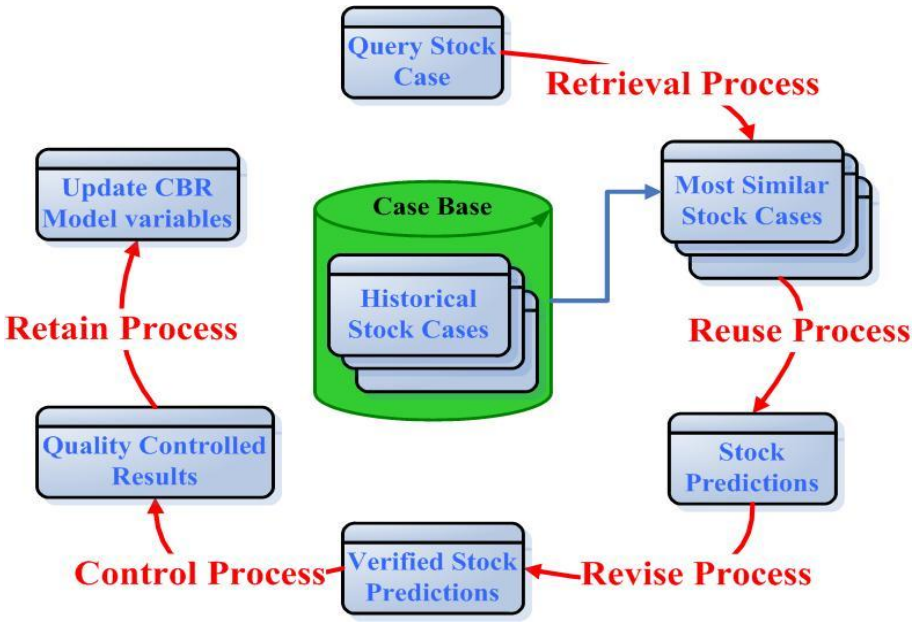

Figure 1: Quality Controlled Stock Prediction Model

A. Retrieval Engine

The retrieval engine captures the knowledge embedded in the historical stock prices of ESE in its similarity assessment and query relaxation components. Similarity computation depends on distance-based methods. Since distance-based methods are sensitive to the choice of features, it is necessary to carry out data reduction using feature selection methods. Data reduction and removal of the irrelevant and redundant features shorten the running time of reasoning and yield more accurate results specially when the size of data is large (Kim, 2004).

1) Similarity Assessment

The similarity assessment module is the most important module of the retrieval engine and considered the heart of the CBR system. The most important assumption in CBR is that similar experiences can guide future reasoning, problem solving, and learning. Similarity between all cases retrieved from the database and the query case is calculated using similarity measures. The concept of similarity between two cases depends on the fact that the less the difference between the two cases, the more similar they are. In the implementation of the similarity measure, weighted feature-based similarity is the most common form (Li and Sun, 2009). The similarity between two cases is computed as the weighted average of local similarities of features as shown in equation 1. Weights were computed using the previously mentioned feature selection module.

$$Similarity(Q,C) = \frac{1}{n}\sum_{i=1}^{n} f(Q_i, C_i)w_i$$

.....*(1)*

Where

Q is the query stock case
C is the retrieved case to be examined
n is number of features in each case
i is an individual case feature from i to n
f is the local similarity function for feature i in cases Q & C
w_i is the weight of feature i

Local similarity of each case attribute is computed using equation 2. The absolute difference between both respective features is computed then divided by the standard deviation of feature standard deviation. Maximum similarity is 100 percent and minimum similarity is 0 percent.

$$local\ similarity\ percentage = 1 - \left|\frac{open\ price\ query\ case - open\ price\ candidate\ case}{standard\ deviation\ of\ open\ prices}\right| \text{.....(2)}$$

For example, to compute the similarity between a stock open price of 22.72 and another open price of 16.69, the standard deviation of open prices of 62.18. Assuming equal feature weights, a similarity score of 90.3% is computed between both features as shown in following equation:

$$local\ similarity\ of\ open\ prices = 1 - \left|\frac{22.72 - 16.69}{62.18}\right| = 90.3\%$$

2) Candidate Stock Case Query Relaxation and Formation Module

The candidate stock cases query relaxation and formation module of the CBR retrieval engine selects candidate stocks to be analyzed. Given the query stock case, a database query is formulated to retrieve candidate stock cases from the case base. The query is formulated to retrieve all stock records from the database whose feature values are one standard deviation more or less than current value. For example, if current close different percent of query stock case was 2 percent, since the standard deviation of close different percent is 8.06, then all stock records with values from -6.06 and 10.06 are considered as candidate stock cases. Candidate Stock cases considered depend on open different percent, close different percent, and volume different percent. Table 1a demonstrates a sample query case and Table 1b demonstrates the formulated database query in SQL Language.

Table 1a: Query Stock Case

Open Different Percent	Close Different Percent	Volume Different Percent
1.86	2.14	154

Table 1b: Candidate Stock Case Retrieval Database Query in SQL Language

```
SELECT * FROM a_all_stocks WHERE opendiffpercent>=-2.54 AND opendiffpercent<=6.26 AND closediffpercent>=-2.26 AND closediffpercent<=6.54 AND date <> '2012-04-23' LIMIT 0, 100000
```

B. The Reuse Process

The Reuse process utilizes the stock prediction formation module of the CBR system to formulate next day stock price predictions based on the retrieved top matching stock cases. To formulate the most accurate predictions, it is important to determine the number of top matching nearest neighbors to use while computing the next day prediction and determine the number of previous stock cases to include. To determine the optimal number of nearest neighbors to use, the accuracy of prediction 1, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 using top matching cases were computed as shown in Table 2. The accuracy of prediction was computed using the mean absolute percent error (MAPE) in Equation 1. Table 2 shows that using 60 top matching stock cases generated the least forecasting error.

Table 2: Mean Absolute Percent Error of Next Day Stock Prediction using Different Numbers of Nearest Neighbor Matching Stock Cases

Number of Nearest Neighbours	MAPE
1	147.75
10	126.225
20	118.295
30	121.393
40	116.553
50	113.125
60	112.815
70	115.507
80	116.117
90	118.134
100	126.464

The similarity between a query case and a candidate case could be improved by computing and averaging the similarity between both cases one or two days before. To determine the best number of previous historical cases to use to compute the similarity between a query case and a candidate case, one, two and three previous days were used during similarity computation as shown in Table 3. It was found that including and averaging the similarity of two day back provided the best prediction accuracy and the least MAPE.

Table 3: Mean Absolute Percent Error of Next Day Stock Prediction using Different Numbers of Previous Stock Cases

Number of Nearest Neighbors	Number of Previous Stock Cases used to compute similarity	MAPE
60	1	112.815
60	2	111.139
60	3	116.901

Table 2 and 3 demonstrated that it is best to use 60 nearest neighbors and one previous stock case to formulate the next day close percent difference with least forecasting error.

C. The Revise Process

The revise process validates the stock predictions suggested by the model by presenting the user with visual and tabular data of the top similar retrieved stock cases. The user inspects stock cases to be used to formulate the prediction to apply own expert opinion to reject/accept candidate stock cases. For example if the prediction model was presented with a query stock case for Alexandria Containers and goods (ALCN) on 4th of April 2012, the most similar stock cases suggested by the model was Abou Kir Fertilizers (ABUK) on 5th of June 2006 as shown in Table 4.

Table 4: Top Matching Similar Stock Case presented to User in the Revise Process

Date	Stock Symbol	Stock Sector	Day Diff	Open	Opendp	High	Highdp	Low	Lowdp	Close	Closedp	Volume	Volumedp	Egx30	Egx30dp	SAP500	SAP500dp
2012-04-04	ALCN	21	1	68.4	-0.0	68.4	-0.0	65.5	-4.03	65.5	-4.37	540	440	1398	-1.0	4838	-1.41
2006-06-05	ABUK	4	1	66.5	-0.3	66.5	-0.3	63.4	-4.68	63.7	-2.31	3713	8151	1265	-1.7	5474	-1.16

The user is also presented with graphical comparison between query stock case (ALCN) and most similar stock case (ABUK) in terms of close price and percent change in close price as shown in Figure 3. The dash vertical line represents the separation between historical stock close prices and future stock close prices. The user compares historical stock close prices between query (ALCN) and most similar case (ABUK) suggested by the system to determine whether to accept or reject that case to be used to formulate the prediction.

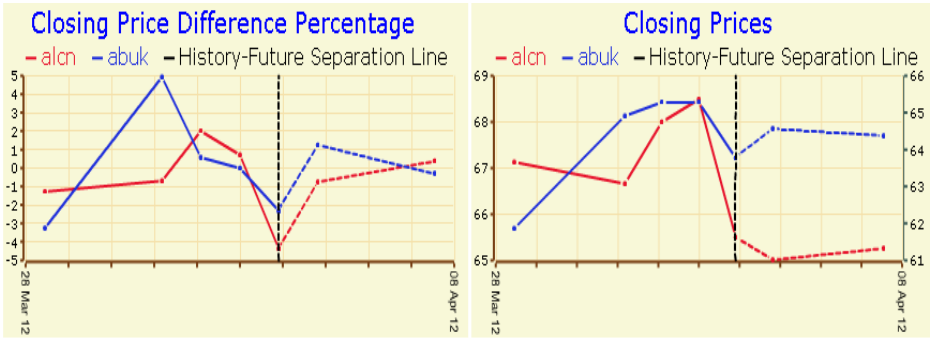

Figure 3: Comparison between Query Stock Case and Most Similar Stock Case in terms of Close Price and Percent Change in Close Price.

D. The Control Process

The control process aims to improve the quality of the stock price prediction process by ensuring that the accuracy of the prediction process remains within acceptable error range. SPC control charts were utilized to ensure that the process is stable and in control and all variations are due to common and not special causes (Moan, Loan and Provost, 1991). If a special cause exists, it is necessary to investigate and try to eliminate it in order to bring the process back to a control state. Special causes not considered within the prediction model may include: omission of an important stock feature, a change or shift in the trend or cycle of the market, entrance of new types of stock investors, severe economic or political events, or other unknown causes of variation. Thus control charts were used to monitor prediction errors and detecting errors due to assignable/special causes and thus provide useful insight on whether or not the prediction process is performing satisfactory. As a result, SPC applications will be very useful in terms of analyzing the stock price prediction process. Hence, the stock investor will have a better understanding on how efficient the prediction model is performing.

The control process used three control charts to monitor the accuracy of stock predictions suggested by the model. Two variable and one attribute control charts were used: the mean (X-bar chart), the standard deviation (S-chart) and the proportion conforming (P-Chart). A sample of sixty stock predictions were randomly collected per day and used to construct the above mentioned control charts. All charts were constructed using the statistical program SPSS v.17.

In order to establish a firm knowledge on the measures of quality of the stock prediction process, a theoretical research was conducted and it was decided to use absolute prediction error and the hit ratio quality measures which were defined by a number of studies (Chen, Wang & Lai, 2008). The mean (X-bar) and S-charts were used to monitor the absolute prediction error which is a variable quality metric, while the P-chart

was used to monitor the hit ratio which is an attribute quality metric. The absolute prediction error indicates how close the forecasted variation mirrored the actual stock price variation. It is computed as the difference between the forecast and actual values as shown in equation 4.4:

$$Absolute\ Prediction\ error = \frac{1}{N}\sum_{i=1}^{N} Predicted\ Stock\ Variation_i - Actual\ Stock\ Variation_i \quad \ldots(4.4)$$

Where

Predicted Stock Variation is the value suggested by the prediction model

Actual Stock Variation is the actual stock price difference.

N is the number of prediction samples per day.

Hit ratio is the most common metric used to determine the quality of predicting stock price variation (Yu, Chen, Wang & Lai, 2008). Hit ratio is defined as

$$Hit\ ratio = \frac{1}{N}\sum_{i=1}^{N} R_i \quad \ldots\ldots\ldots(4.5)$$

Where $R_i = 1$ when sign of prediction equals sign of actual value

$R_i = 0$ when sign of prediction is not equal sign of actual value

N is the number of prediction samples per day.

The mean (X-bar) chart was used to monitor the variation in the average prediction accuracy to show how consistent the prediction process is at achieving its mean. The X-bar chart was chosen because it is the most commonly used chart when the quality measured is a continuous variable (Chou, Li & Wang, 2001). The standard deviation (S-chart) was used to monitor the variation and dispersion of data within samples. An S-chart was used rather than an R-chart because the sample size was sixty and R-charts are efficient only when sample size is less than ten.

The X-bar control chart was used to monitor the absolute prediction error from 2nd January 2012 to 24th January 2012 is shown in Figure 4. The control limits were computed from the collected samples so that the mean center line equals 2.096, the upper control limit equals 2.666 and the lower control limit equals 1.526. From 18th January to 24th January 2012, five consecutive out of control points indicate the existence of assignable (special) causes of variations which require further interpretation. During this period significant economic and political events were taking place in Egypt including the start of president Mubarak court trial and declaration of first elected people party in Egypt after the 25 January revolution. The Egyptian stock market index EGX30 increased by more than 13% during these days. The X-bar control chart guided the stock investor not to use the suggested model to predict next day stock prices

from 19th to 24th January 2012. Accordingly, the mean absolute prediction error was reduced to 1.92.

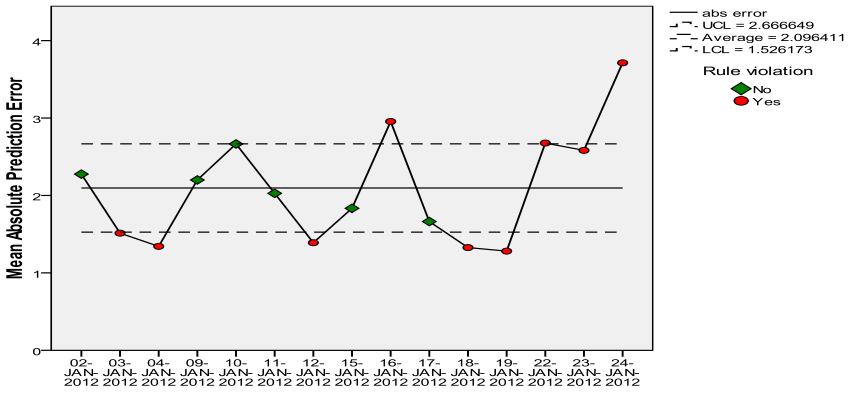

Figure 4: X-bar Control Chart monitoring Absolute Prediction Error from 2nd January 2012 to 24th January 2012

The S-chart was used to monitor the absolute prediction error from 2nd January 2012 to 24th January 2012 is shown in Figure 5. The control limits were computed from the collected samples so that the mean center line equals 1.466, the upper control limit equals 1.872 and the lower control limit equals 1.06. At 10th of January, the S-chart detected an out of control situation within collected samples. During the 9th of January President Mubarak was charged with treason and official from the World Bank visited the country causing the Egyptian stock market index EGX30 to increase by 2.4% during the 10th of January 2012.

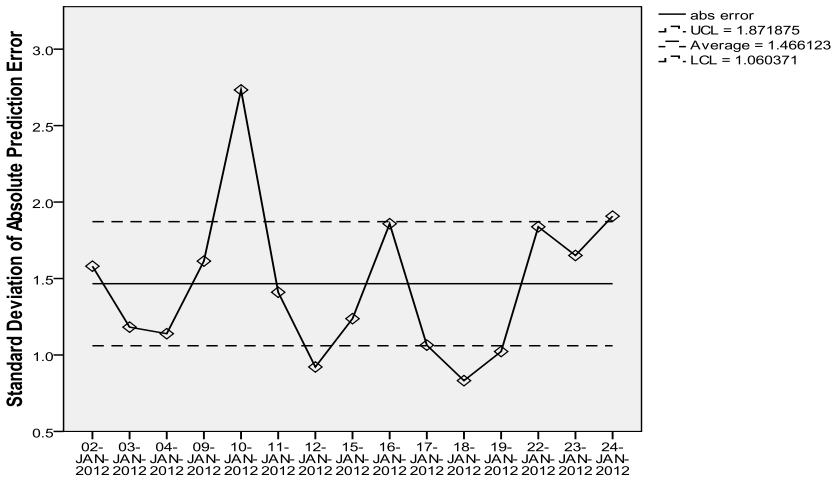

Figure 5: S-Chart of Absolute Prediction Error from 2nd January 2012 to 24th January 2012

Proportion (P) charts plot the fraction of correct predictions defined as the ratio of the correct predictions to the total number of predictions. P-charts were chosen over Count (C-charts) because the probability of correct predictions is large and more than five percent of the total number of predictions. The P-chart was used to monitor the quality attribute hit ratio from 2nd January 2012 to 24th January 2012 as shown in Figure 6. The control limits were computed from the collected samples so that the mean center line equals 67%, the upper control limit equals 85% and the lower control limit equals 48%. The P-chart detected out of control conditions from the 18th of January 2012 till the 24th of January 2012. The P-chart detected and confirmed the same instability within the stock prediction process detected by the X-bar chart. The P-chart chart guided the stock investor not to use the suggested model to predict next day stock prices from 19th to 24th January 2012. Accordingly, the hit ratio increased to 72%.

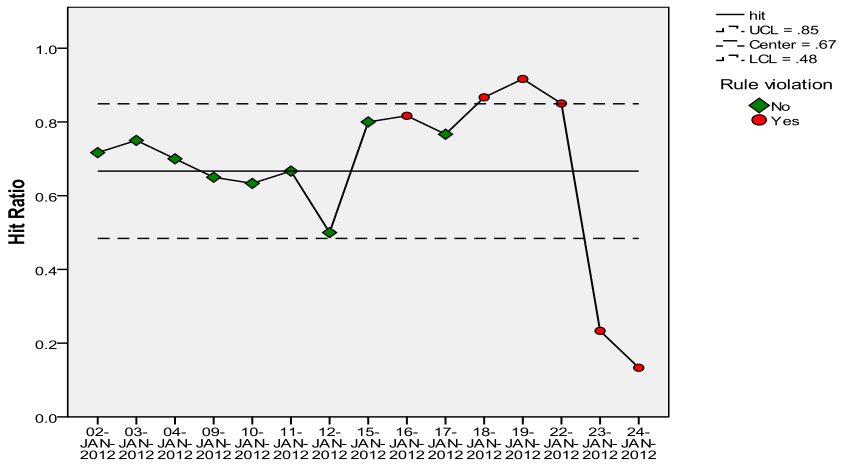

Figure 6: P-Chart of prediction hit ratio from 2nd January 2012 to 24th January 2012

The above three control charts were used to improve the quality of the prediction process by monitoring its prediction accuracy and detecting instability and out of control conditions due to assignable and not common causes of variations such as economic or political instability.

E. The Retain Process

Finally, the retain process saves the new stock cases together with the accuracy of the predictions suggested by the model within the case base. The prediction model is retrained upon addition of these new stock cases in order to account for changes within the factors determining future stock prices within the chosen stock market. Retraining involves readjust and fine tuning the weights of different case feature weights of the retrieval process in addition to the number of K-nearest neighbors (KNN) used within the reuse process that provide least prediction error and optimal accuracies.

V. MODEL OPTIMIZATION

A. Stock Case Base Authoring

The case base stores all historical cases to be included in the reasoning process. One case must contain as much information of the historical situation as possible. Based on literature survey and the data considered for this research, a stock case was designed to contain the following raw stock features: open price, low price, high price, close price, volume, the close value of the Case-30 Egyptian stock exchange market index (egx30) and the close value of the United States S&P500 market index. S&P500 market index was included in the stock case due to the close integration of the Egyptian stock market with US stock market as suggested by (Cuza, 2009). To

obtain stationary and remove the effect of measurement units among the raw data, differencing and normalization operation were performed on all stock features and included in the stock case.

The stock case base was partitioned into three clusters based on the price change percentage of next closing price. Partitioning the case base into three case bases is based on the notion that different features of stock prices attain different importance before the stock price increase, decrease or remain without significant change. The first case base included all increasing stock records that yielded an increase in the next closing price by a value equal or more than one percent. The second case base included all stock records that yielded a decrease in the next closing price by a value equal or less than one percent. The third case base included all stocks that yielded a change in next closing price between one percent increases to one percent decrease. This technique is similar to the technique used by (Chang, Fan & Lin, 2011) which classified the case base into subsections by applying a data matrix to find similarly weighted data, and applies the gradient method to find the clustering groups. This is effective in finding and describing patterns in data to make predictions, and also to build an explicit representation of the knowledge of stock movement. This approach in classifying the case base was chosen for simplicity as compared to the more complex clustering techniques such as K-means clustering used by Bezdek (1981) and Bereta and Burczyn'ski (2009), Kohonen' self-organizing network in Kohonen (1988) and case-based clustering (Shiu et. al., 2002; Chang et. al., 2011).

B. Learning Case Feature Weights

Feature weight optimization and determination of relative importance of different stock features involved partitioning the case base into three clusters then the relative importance of different stock feature was determined for each case base. Feature selection tries to pick a subset of features that are relevant to the target concept and remove irrelevant or redundant features. Feature weighting includes feature selection, since selection is a special case of weighting with binary weights. But, feature selection is more efficient and effective than feature weighting when harmful features are existing (Kim, 2004). Learning case feature weights is assigning a weight to each feature based on its relative importance. Wettschereck, Aha and Mohri (1997) presented various feature weighting methods based on distance metrics in the machine learning literature. Kim (2004) proposed a GA-based feature weighting method for *k*nearest neighbor.

Methods of feature subset selection into two categories: a *filter* and *wrapper* approach. The filter approach, a feature subset is selected by the characteristics of data itself and independent of learning algorithm (Wang, Bell and Murtagh, 1998). The wrapper approach depends on the results of learning algorithm for the selection of relevant feature subset. Although the filter approach is rather simple to implement, the wrapper approach may be effective because it operates with the learning algorithm in a synergistic way (Kim, 2004). The wrapper approach was selected as the method to optimize feature weights based on the forecasting error percent. The goal is to find

feature weights that produce the minimum forecasting error. Within each class of stock records, large weights are assigned to those sensitive key features, while small (or zero) weights will be assigned to irrelevant features. Deciding the trend of the future stock price movement will be more effective giving more weight to relevant features and ignoring irrelevant features. A greedy algorithm was chosen to find the optimized feature weights for each of the clusters defined in the previous section. The algorithm is continuously executed upon addition of new cases to adapt these weights to new market trends. This algorithm is extracted from feature selection algorithm utilized in (Mitra, Murthy and Pal, 2002). Given the stock case base, the optimization algorithm generates the set of optimized feature weights as shown in Figure 7 as follows:

Step 1: initialize all feature weights to zero.
Step 2: For each feature
Assign weight from 0 to 20.
Calculate forecasting error for each weight.
Iterate 100 times or until convergence (a certain weight continuously generates the minimum forecasting error).
Choose the weight that generates the minimum forecasting error.
Step 3: Upon addition of new stock records, repeat step 2.

Figure 7: Feature Weight Optimization Algorithm

The forecasting error is computed using the mean absolute percent error as shown in equation 4.1.

$$MAPE\ of\ next\ day\ prediction = \frac{\sum_{m}^{1} \frac{actual\ close\ different\ percent - predicted\ close\ different\ percent}{actual\ close\ different\ percent} * 100}{m} \quad (3)$$

Where m is number of cases used to determine prediction accuracy
Upon addition of new stock cases, step two is repeated utilizing newly added cases to update the feature weight set. The above algorithm was run on the stock case base to yield the following optimized feature weights for each of the three clusters as shown in Table 5.

Table 5: Optimized stock feature weights

Stock Feature	Num. of Runs	Increasing Stocks	Decreasing Stocks	No Change Stocks	All Stocks
Month	293	0	5	4	0
Day	308	6	14	5	5
Daydiff	264	20	18	0	1
stock_id	267	0	10	0	0

sector_id	282	20	10	0	1
Open	227	1	12	1	1
Opendiffpercent	420	0	9	20	10
High	258	1	1	15	1
Highdiffpercent	177	4	6	16	16
Low	212	0	0	20	1
Lowdiffpercent	198	5	1	3	1
Close	150	18	12	8	12
Closediffpercent	208	19	9	19	9
Volume	212	10	6	13	13
Volumediffpercent	180	20	20	20	20
egx30close	202	7	13	0	0
egx30closediffpercent	180	3	20	7	6
sap500close	211	2	20	9	6
sap500closediffpercent	237	7	11	12	8

The above feature optimization algorithm yielded that the most important stock features for increasing stock price were day difference, sector, previous closing price, previous closing price change percentage, previous volume change percentage. The most important stock features for decreasing stock prices were day difference, previous volume change percentage, previous EGX30 close change percentage and SAP500 close. The most important stock features for non changing stock prices were previous open change percentage, low price and previous volume change percentage. These feature weights were used to initialize the similarity retrieval engine of the stock CBR model.

VI. **Results**

The suggested quality controlled stock case-based prediction model was tested by computing the accuracies of predicting 900 actual next day different stock prices from the 2nd of January 2012 to the 24th of January 2012. The stocks were chosen randomly to include distinct stocks during different days and months. The model was then verified by comparing these results with other models from the literature. The mean absolute prediction error of the randomly chosen stocks was 2.096 LE and ranged between 1.526 and 2.666 L.E. The hit ratio was 67% and ranged between 48% and 85%. After

using the quality control process, the mean absolute prediction error was reduced to 1.92 L.E. and ranged between 1.376 and 2.48 L.E. The hit ratio increased to 72% and ranged between 54% and 89%.

Table 6: Quality Controlled Stock Prediction Model Performance Results

Feature	Absolute Prediction Error (L.E.)			Hit Ratio		
	Max.	**Min.**	**Mean**	**Max.**	**Min.**	**Mean**
No Quality Control	2.666	1.526	12.09	85%	48%	67%
Quality Control	2.48	1.376	1.92	89%	54%	72%

The suggested prediction model assigned relative importance to stock case features which is similar to (Chun and Park, 2005; 2008), (Li and Ho, 2007) and (Chang, Tsai, Huang, and Fan, 2009) which assigned relative importance weights for different stock features and updated these weights dynamically then use these weighted features to find nearest neighbor cases to predict next day stock prices. The suggested model is also in accordance with the model suggested by (Chang, Fan and Lin, 2011) which selected the most important factors from the set of input features. The suggested model is also in accordance with the model suggested by (Ahn, Kim and Han, 2006) which searched for optimal k parameter for the k-nearest neighbor to improve the performance of the CBR system.

The suggested model clustered stock cases into different classes similar to (Sushmita and Chaudhury, 2007) which utilized CBR to analyze stocks in financial market and created a hierarchical case structure and similar to (Chang, Tsai, Huang, and Fan, 2009) which integrated a data clustering technique with Case Based Reasoning (CBR). It is also similar to (Chang, Fan and Lin, 2011) which divided the original case base into eight different subsets of similar cases

The suggested model utilized a man-machine interface to support the decision making for predicting stock price future movements using an expert to improve the prediction accuracy which is similar to (Li and Sun, 2009c), (Li and Ho, 2007) and (Srinivasan, Singh and Kumar, 2011) which proposed a Decision Support System (DSS) based on Case-base Reasoning (CBR) and Data Mining (DM) to predict stock market prices from huge data stores.

Literature review did not reveal the previous usage of control charts as a statistical process control method to control the quality of stock predictions. The suggested quality controlled stock price prediction model results indicated a mean hit ratio of

72% which is in accordance with the case-based fuzzy decision tree model suggested by Chang, Fan and Lin (2011) which provided an average hit ratio of 91% and the integrated support vector machine (SVM) and also in accordance with the CBR stock prediction model suggested by Chang, Tsai, Huang, and Fan (2009) which had a hit ratio of 93.85% for stocks within the US S&P500 index as shown in Table 7.

Table 7: Comparison of Hit Ratio between Proposed Quality Controlled CBR Stock Prediction Model and other Models

Stock Prediction Model	Stock Market	Hit Ratio
Quality Controlled CBR Stock Price Prediction Model	Egyptian Market	72%
CBR Fuzzy Decision Tree Model (Chang, Fan and Lin, 2011)	S&P500	91%
SVM and CBR Model (Chang, Tsai, Huang, and Fan (2009)	S&P500	94%

VII. Conclusion

This major contribution of this research is to improve the quality of investing in the stock market by combining the concepts of statistical process control and artificial intelligence in the highly volatile Egyptian Stock Exchange (ESE). The suggested quality controlled stock prediction model appends a control process to the traditional four cyclic process of CBR: retrieval, reuse, revise and retain. The suggested model generated acceptable results for the highly volatile Egyptian Stock Exchange (ESE). The mean absolute prediction error of the randomly chosen stocks was 2.096 LE and ranged between 1.526 and 2.666 L.E. The hit ratio was 67% and ranged between 48% and 85%. After using the quality control process, the mean absolute prediction error was reduced to 1.92 L.E. and ranged between 1.376 and 2.48 L.E. The hit ratio increased to 72% and ranged between 54% and 89%. The model was verified through comparison with the case-based fuzzy decision tree model suggested by Chang, Fan and Lin (2011) which provided an average hit ratio of 91% and the CBR stock prediction model suggested by Chang, Tsai, Huang, and Fan (2009) which had a hit ratio of 93.85% for stocks within the US S&P500 index.

VIII. References

Aamodt A. and Plaza E. (1994). Case-based reasoning: foundational issues, methodological variations and System approaches. AI Communications 7(1), 39-59.

Bereta, M., & Burczyn'ski, T. (2009). "Immune K-means and negative selection algorithms for data analysis". *Information Sciences*, 179(10), 1407–1425.

Bezdek, J. C. (1981). "Pattern recognition with fuzzy objective function algorithms". Plenum, NewYork.

Chang, P. C., Liu, C. H., Lin, J. L., Fan, C. Y. and Ng, C. S. P. (2009), "A neural network with a case based dynamic window for stock trading prediction". *Expert Systems with Applications*, No. 36, pp. 6889-6898.
Chang, P., Fan, C. Lin, J. (2011). "Trend discovery in financial time series data using a case based fuzzy decision tree". Department of Information Management, Yuan Ze University, Taoyuan 32026, Taiwan. Expert Systems with Applications 38 (2011) 6070–6080
Deming, W.E., (2000). *Out of the Crisis*, MIT Press, Cambridge, Massachusetts. ISBN: 0-262-54115-7.
Eddington, A.S. (1928). *The Nature of the Physical World.* Cambridge University Press. London
El Telbany, M. E. (2004). The egyptian stock market return prediction: a genetic programming approach. Electrical, Electronic and Computer Engineering International Conference, 2004.
Kim, K. J. (2004). "Toward Global Optimization of Case-Based Reasoning Systems for Financial Forecasting". Applied Intelligence 21, 239–249, 2004.
Mitra P., Murthy C.A., Pal S.K. (2002). "Unsupervised feature selection using feature similarity". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 24, Issue 3, 301-312.
Lenz, M., (1999). Case Retrieval Nets as a model for building flexible information systems. PhD dissertation, Humboldt Uni. Berlin. Faculty of Mathematics and Natural Sciences.
Li, H. and Sun, J. (2009). "Predicting business failure using multiple case-based reasoning combined with support vector machine". *Expert Systems with Applications* 36(6) 10085–10096.
Li, S. T. and Ho, H. F. (2009), "Predicting financial activity with evolutionary fuzzy case-based reasoning", *Expert Systems with Applications*, No. 36, pp. 411-422.
Kohonen, T. (1988). "Self-Organization and Associate Memory", Springer, Berlin.
Reid, R. D. & Sanders, N. R. (2010). *Operations Management, 4th Edition*. New York: Wiley.
Shiu, S. C. K., Sun, C. H., Wang, X. Z., & Yeung, D. S. (2002). "Transferring Case Knowledge To Adaptation Knowledge: An Approach for Case-Base Maintenance". *Computational Intelligence*, Volume 17, Issue 2, pages 295–314, May 2001. Blackwell Publishers, Inc.
Srinivasan, S., Singh, J., Kumar, V. (2011). "Multi-agent based decision Support System using Data Mining and Case Based Reasoning". *IJCSI International Journal of Computer Science* Issues, Vol. 8, Issue 4, No 2, July 2011.
Sushmita, S. & Santanu C. "Hierarchical Fuzzy Case Based Reasoning with Multicriteria Decision Making for Financial Applications". Pattern Recognition and Machine Intelligence Lecture Notes in Computer Science Volume 4815, 2007, p226-234.
Wang H., Bell D. and Murtagh F. (1998). "Relevance approach to feature subset selection in Feature Extraction, Construction and Selection: A Data Mining Perspective". Kluwer Academic Publishers: Boston, 1998.
Wheeler, D and Chambers, D. (1992). *Understanding Statistical Process Control*. SPC press.

Wettschereck D., Aha, D.W. and Mohri, T. (1997). “A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms”. *Artificial Intelligence Review*, vol. 11, pp. 273–314.
Yu L., Chen H., Wang S., Lai, K. K. (2008). “Evolving Least Squares Support Vector Machines for Stock Market Trend Mining”. Chinese Academy of Sciences, Beijing, China.