International Research Journal of Computer Science (IRJCS) ISSN: 2393-9842
Issue 11, Volume 3 (November 2016) www.irjcs.com

A NOVEL APPROACH TO INVESTIGATE THE USE OF HYBRID DATA OVER REAL TIME DATA

Veena K K
Research Scholar, VTU,RRC,Belgavi, Karnataka
Veena.bichagal@gmail.com

Dr.Basavaraj S Patil
Professor, Department of Information Science & Engineering, AMC Engineering College, Bengaluru, Karnataka
Chief Data Scientist, Predictive Research Pvt Ltd., Bengaluru

Manuscript History
Number: **IRJCS/RS/Vol.03/Issue11/NVCS10085**
Received: 05, September 2016
Final Correction: 20, October 2016
Final Accepted: 02, November 2016
Published: November 2016

Abstract: The main aim of the proposed study is to develop a hybrid temporal model that provides learning pattern for classifying the temporal data. These results are unusual, which is in contrast with the Hidden Markov Models (HMM). The system is evaluated in terms of the capabilities of a hybrid learning algorithm, which is applied over the temporal data. Performance of the hybrid algorithm depends entirely on the dynamic data, which is fed into the system. The data fitting is an important concern, to find, analyse and predict the future instance. Hence, the difficulty in making a hybrid algorithm to fit the dynamic data is increasing, however, the data fits in better proportion over the expert system. An expensive research is required to build the required module for data pre-processing, analyzing and prediction. Also comparing such systems' performance with the conventional schemes is required to prove its effectiveness. The study aims at developing a most generic artificial neural network hybrid algorithm, which predicts well the stock market data without the knowledge of past outputs. Hence, the end user does not trouble the recognition system and that is regarded as the virtues of soft computing tools

Keywords: Hybrid Approach, real time data, hybrid temporal model, learning algorithm, stock market data

---

I. INTRODUCTION

The learning of temporal pattern requires the self-organizing network architecture, which corresponds to the temporal patterns. The conventional temporal pattern recognition and neural network methods are considered as better solutions for the temporal recognition problems. Most real-world considerations require a proper solution using temporal pattern recognition. In a non-trivial environment, the temporal pattern recognition with neural network pattern will provide solutions related to temporal information. The information is considered dynamic as it enters the system for the purpose of processing and analysis. However, under relative time order, it is difficult to process the temporal information in the incoming dynamic data. In several fields like robotics, speech recognition, analyzing the situation and fusion of data, requires the use of temporal behaviour of the associated dynamic data. During the use of non-temporal approach for pre-processing or for other block processing, the information lost is considered expensive. Hence, the use of neural network approach for obtaining the temporal information to obtain the data is considered effective.

The neural network to process the temporal information in the present study considers the incoming events as the input data. The sequence of the events, (mostly the letter, abstracts, phonemes and objects) accepted by the system over certain intervals of time is categorized specifically as sequence for the temporal data recognition system.

The temporal data recognition system uses Gaussian classification process, which signifies the temporal data statistics. Here, the temporal data recognition system uses neural network learning scheme with mean and co-variance factors, which updates the self-formation classes. When the system finds the temporal sequences in the presence of noisy environment and the output with Gaussian distance measurement for the stored data sequence, hence, an analog measure is chosen to provide the solution using closeness fit for the present recognized temporal patterns. Also, the temporal system with neural network learning identifies well the temporal sequence, even in the presence of missing entities and unordered sequences. Further, the system can be used well as a prediction abled system, which realizes the length of each sequence before the introduction of entire sequence. This leads to multi-dimensional distance measurement using Gaussian distribution [1].

1.1.1 Temporal Patterns

The capability of understanding the environment is an important characteristic in an indefinite intelligence search pattern, which is not administrated through static temporal recognition pattern. The occurrence of events is more important than the individual events, whatever may be the event is, and it is follow the order of the events. The dimensions over the timing patterns allow the access to information, which is related to its present situation, past events and future expectation. The capacity of incorporating the time into temporal information process is an essential ability for the temporal recognition of the order of events, understanding its cause and effect and making better planning and predictions [1]. The capability of recognizing the identifying the sequences is an important consideration for multiple tasks, which are involved in vision and audition process. Here, the sequence may consist of phonemes stream, letters typed or movie frames. Upon completion of the preprocessing task, the members of each temporal sequence are recognized and there occurs the shifting of the task. Followed by this is the processing task, which is concerned in finding the known sequences and they are characterized by the input of the system. The answer depends purely on the context of received input data and the system should return the sequences, which represents the confidence rating or the matching similarity between the input and the recognized sequences [1]. The main task is recognizing the required sequences, which are present in an order. However, there is a possibility of the occurrence of the errors, which is due to fluctuations in the input data with variable environmental conditions. Even in the presence of error, the system must handle the order of the sequence in an effective manner to deal with such ambiguities in temporal recognition pattern [1].

1.1.2 Temporal Information System

The processing system of the temporal system consists of a multi-layer network architecture that varies distinctly from conventional artificial neural network models. Here, the neural network organizes the temporal learning phase and develops the categories of Gaussian distribution for the learnt sequences. The Gaussian categories are represented in the form of single nodes in a particular level, which depends on the level of activation in a field. When the input stimuli arrives the information system, the decay factor is experienced by the field, which provides the ordering of the activation nodes based on its input time [3]. The temporal system or the Gaussian classifier mainly provides support to process the stimuli of the static inputs through the combination of temporal decay and mean and covariance factors [2], which obtains to represent the input patterns statistics and categories update statistics. Hence, the system tires to classify the representation of the input stimuli, which is very far from the conventional category of the temporal recognition pattern. When the system fails such pattern, the new category of sequences is collected from the present sequences. The available categories are available in parallel and independent form to perform distance measurement with a novel input. This further determines the better categorizations over the new input samples [1]. The Gaussian classification is utilized to achieve the process of unsupervised input sequence partitioning. Here, the individual nodes have multi-dimensional Gaussian activation function, which is adapted better with the input data using its mean and covariance matrix. The data statistics of the system is learnt and that represents the data as a sum of multi-dimensional normal distributions. However, the individual inputs are represented as a dissimilar class, where the individual nodes'acquires the data to produces the Voronoi classifier. In other cases, the entire points are classified as a single input and that fits the normal distribution, where the input adheres to the mean and covariance of the system and output fits the normal distribution [1].

II. TEMPORAL DATA MINING

The temporal data mining is considered as the process of knowledge discovery from the temporal databases, which possess dynamic data and that count the temporal patterns over the temporal data. Here, any nature inspired algorithm especially neural networks counts the temporal patterns and fits the temporal data to form a mining algorithm [4]. The objective of the temporal data mining is to discover the proper temporal patterns, sudden trends or hidden relations over the sequential data. Here, temporal sequences refer to the nominal symbols from the alphabet and the time series is represented by the continuous real-valued elements 'sequence. This is obtained as a combination of technique through the process of machined learning, database technology and statistics. The temporal mining or data classification consists of three major categories that include:

- Representation of the temporal input data
- Definition of the similarity measurement, and finally
- The mining tasks.

2.1 TEMPORAL DATA REPRESENTATION

The temporal data representation here refers to the effective way of representing the temporal data before the process of mining. This is classified into three major approaches, namely:

- Time domain representation,
- Transformation representation, and
- Generative model representation.

2.1.1 Time Domain Representation

The time domain representation is the simple method, which represents the minimal computation of the required temporal data. This is placed in either in the original temporal form for the given data or it is segmented into several parts to form temporal sequences. Here, the occurrence of the original temporal form is represented in time domain representation and the segmented temporal patterns are represented using linear function. The time domain application provides easier approach, since; implementation is an easier task without the loss of temporal information. On the other hand, it poses heavy demand on higher computation resource and power for mining the required input data. This is considered infeasible in case of real-world problems that consists of large volume of temporal data with high dimensionality

2.1.2Transformation based Representation

The transformation domain representation is typically a process of transferring into representation space, the temporal data. The features contain the discriminatory information, which are extracted and processed for representing it in a temporal data form. However, in general the transformation domain representation is categorized into two, they are:

- Piecewise representation domain
- Global representation domain.

The piecewise representation refers to the temporal data partitioning into segments at a certain critical points. The critical points are chosen based on various criteria, where each segment is modelled into a precise representation. The representation of segments constitutes and forms piecewise data representation. Some of them includes: adaptive piecewise constant approximation [5] and curvature-based principal component analysis segments [6]. The global representation domain is modelled using the temporal data through the basis function set. The temporal coefficients available in the parametric space refer to the representation of the temporal data in a global manner. This is used to reconstruct approximately, the temporal data. Some of the commonly used representations include:

- POLYNOMIAL/SPLINE CURVE FITTING [7, 8],
- DISCRETE FOURIER TRANSFORMS (DFTS) [9]
- DISCRETE WAVELET TRANSFORMS (DWTS) [10].

The main advantage related to transformation representations reduces the high dimensional temporal data in order to attain uniformity in lower dimensional feature space. This significantly helps in improving the computational efficiency of the system. However, it is realized that none of the representation could cater perfectly the needs of different temporal data set and each representations capture the limited characters, which is acquired from temporal data set [11-13].

2.2.1.1 PIECEWISE Local Statistics

The piecewise local statistics is inspired by the speech signal processing, motifs in time series [4], and bit-level time series representation [14]. The piecewise local statistics analysis uses window-based time series statistics. Initially, the fixed window size is used to block the time series data into a group of segments. The individual segments uses 1storder and 2ndorder statistics for estimating the features of the segment set. Hence, the segment 'n' with its local statistics, having its mean μ_n and standard deviation (SD) σ_n for each observation is calculated by

$$\mu_n = \frac{1}{|W|}\sum_{t=1+(n-1)|W|}^{n|W|} x(t)$$

$$\sigma_n = \sqrt{\frac{1}{|W|}\sum_{t=1+(n-1)|W|}^{n|W|}\left[x(t)-\mu_n\right]^2} \tag{1.1}$$

Where,| W| denotes the window size.

The time series piecewise distribution uses PLS representation with a fixed number of dimensions is created with the set of notations belonging to the features of local statistics for all the given segments. Here, the estimate is affected at the destination of the time series, since the window is made delimited. Hence, the PLS representation is considered as an extension of [15], which uses 1storder statistics.

2.2.1.2 Piecewise Discrete Wavelet Transform

The Discrete Wavelet Transform (DWT) is considered as an efficient tool for multi-scale analysis. Similar to the pre-processing procedure in the PLS, the time series $\{x(t)\}_{t=1}^{T}$ with the |W|, window size is blocked into segment set. The local information in the multi-scale wavelet analysis is acquired with the use of Daubechies wavelets over each segment set. This acquires the information accurately during the multi-scale analysis. Consider n^{th} segment with level '*J*', multi-scale analysis is carried out by applying DWT and that leads to the Piecewise DWT representation under its collective coefficients.

$$\{x(t)\}_{t=(n-1)|W|}^{n|W|} = \left\{\Psi_L^J \{\Psi_H^J\}_{j=1}^{J}\right\} \quad 1.2$$

The decomposition of each segment in DWT time series analysis at different levels is carried out with the low- and high-pass filters. At j^{th}level, 2^{-j} |W| coefficients encodes the entire information using Ψ_H^J of high-pass filters. While, coarse information is acquired using Ψ_L^J of the low-pass filters.

2.2.1.3 Polynomial Curve Fitting

The main aim of the polynomial curve fitting procedure finds the derivations related to the signals, which is influenced less by the noises. Least-square approximation or least-squares polynomial is the most common approach that finds the best coefficients that fits well with the sequential data. The coefficients belong to the polynomial equations and the time series model makes an appropriate fit with parametric polynomial function [16].

$$x(t) = \alpha_P t^P + \alpha_{P-1} t^{P-1} + \ldots + \alpha_1 t + \alpha_0 \quad 1.3$$

Here,α_p (p= 0,1,…,P) is considered as the best polynomial coefficient of order p. Generally, polynomial coefficient belonging to the 4^{th} order performs well as per the studies. However, the coefficients from the higher order perform in a lower order without any significant improvement. Here, the most appropriate fit is carried out with the least square error minimization function that takes into account the points from the sequential time series.

Further it considers the given order polynomial order w.r.t. α_p (p = 0,1,…,P). Here, the entire coefficients are obtained through the process of optimization using PCF representation, which is a location based time series global representation with the points in sequential series. The time series complex structure accounts for abrupt changes in a larger manner with the points, observed. Hence, the appropriate approximation is carried out with high order polynomial curve. Other main disadvantage of PCF function is that, it is unable to retrieve the local information, which is essentially important for accurate approximation.

2.2.1.4. Discrete Fourier Transform

The PCF representation contains the generic time series information to analyse the time domain trajectory. Yet, the Fourier transform provides greater decomposition of the time series data in the frequency domain. The Fourier series analysis is an effective mechanism to transform the sequence and functions of data elements into frequency domain from its native time domain representation. The main advantage of analyzing the data sequence is that it reveals the most important properties, which is not noticeable in the time domain representation. Fundamentally, the Fourier transforms is divided into two types:

- Continuous Fourier Transform (CFT) and
- Discrete Fourier Transform (DFT).

The CFT aims to decompose the continuous waveform into a continuous component of frequency spectrum. The inverse CFT synthesizes a continuous function from the frequency components spectrum. On contrary, the DFT over discrete sample signal observes the discrete sequences from the time series. This observation is carried out to represent the data in the form of temporal data representation. The DFT aims to map the sequential discrete information from the observed time domain to the sequential discrete information in the frequency domain using its related coefficients. The DFT is applied for the purpose of deriving the global time series representation in the frequency domain [9]. The time series DFT $\{x(t)\}_{t=1}^{T}$ attains the Fourier coefficients' set:

$$a_k = \frac{1}{T}\sum_{t=1}^{T} x(t)\exp\left(\frac{-j2\pi kt}{T}\right) \quad 1.4$$

Where, $k = 0,1,...,T\text{-}1$

The robust representation is formed under noise with few top K coefficients agrees with the low frequencies, where $K<<T$. The coefficients form the descriptor of the Fourier series. As of now top 16 coefficients that corresponds to the lower frequency captures the utmost features of the frequency component, where necessary improvements is not made by the DFT coefficients.

III. GENERATIVE REPRESENTATION

The generative model or generative representation is treated as a mixture of the models, where the temporal data is obtained. Hence the temporal data obtained from the various models include:

- 1st order Markov chain model
- Dynamic Bayesian networks model
- Hidden Markov Model and
- Autoregressive Moving Average Model

Hence, the temporal data with the relevant model parameters is represented as,

$$p(x|\theta) = \sum_{k=1}^{K} w_k p(x|\theta_k) \qquad 1.5$$

Where, $\theta = \{\theta_k\}_{k=1}^{K}$ denotes the model parameters (unknown), w_k is the weighted coefficient with $\sum w_k = 1$ and $0 \le w_k \le 1$ K denotes the component model, which represents the total data set. Out of all regenerative model representation techniques, Hidden Markov Model provides the better ability for acquiring the information related to temporal features. The values of such features changes dynamically during the process of observation and this satisfies the Markov property. Hence, the temporal data is represented as Hidden Markov Model and that depicts the data as an unobservable stochastic process with finite states, which is related with other observable stochastic process. Consider an observation is related with initial probability p_j at the state j. The emission probability emitted by the observation model is given as b_j. Consider the next state I, which is selected as per the state transition probability (a_{ij}) with emission probability b_i generates a symbol. The entire process gets repeated until the stopping criterion is reached. Here, the process of states during each event is eliminated and the process outcomes produce sequence of events from the observable stochastic process. This is considered as the "hidden" procedure and the parameters obtained from such process are considered as a triplet:

$$\lambda = \{p=\{p_j\}, A=\{a_{ij}\}, B=\{b_j\}\} \qquad 1.6$$

The emission probability of the continuous temporal data or time series data is defined in terms of multivariate Gaussian distribution. However, the simulation used in the proposed study considers Hidden Markov Model as a single Gaussian distribution. This is used as an emission function with $b_j = \{\mu_j, \sigma_j^2\}$ and this reduces the cost of computing the stock market datasets and avoids the over fitting problem with the limited data set.

Thus, for temporal data with emission probability, the observed data set is represented in the form of a Hidden Markov Model set where the value of λ varies from 1 to K. The model contains M states and observed data set is a single Gaussian distributed element, where the component with parameters is shown below:

- Probability vector p (initial state) with M-dimensions
- State transition matrix A = M×M
- Mean vector, μ varies from1 to M and
- Variance vector, σ varies from1 to M

The applications related to Hidden Markov Model approaches three major problems, which are shown below:

a. Using the parameters λ={p, A, B} and the probability p(x|λ) generates a specific set of observations $x = \{x(t)\}_{t=1}^{T}$, the problem associated with Hidden Markov Model approaches is solved using forward algorithm and backward algorithm [17, 18].

b. Using the parameters λ = {p, A, B} and the hidden state generates a specific set of observations $x = \{x(t)\}_{t=1}^{T}$, the problem associated with Hidden Markov Model approaches is solved using Viterbi algorithm [19,20].

c. The most likely parameter from the observed sequence $x = \{x(t)\}_{t=1}^{T}$ is solved with expectation-maximization algorithm [21]. This model helps in achieving the solutions related to the problems of temporal data.

The generative representation helps in identifying the data regularity and dependent nature of the dynamic temporal data.

However, this approach leads to higher computational cost due to severe mining operation over the temporal data. This is because the temporal data is considered to appear in several models in a larger volume with high dimensionality. Hence, this method is not applicable for the solutions of real world problems.

IV. SIMILARITY MEASURES

From the previous sections, the temporal data representation in appropriate manner is found. However, with rapid temporal data, the similarity between the temporal data varies in the representation environment. This is commonly referred as the similarity measurement between the temporal data. Multiple researches are carried out to find the similarity measurement between varied mining patterns for the temporal data. However, the relationship between the representative method and the similarity measurement is ruled by the data mining objectives. Hence, the similarity measure for the temporal data instances are classified into three types [22], namely,

- Time Similarity
- Shape Similarity and
- Change Similarity

4.1 Time Similarity

The main purpose of the similarity measurement is to measure the similarities between the observed two different temporal data w.r.t to its timing intervals. This is achieved using Euclidean distance measurement over the time domain or transformation representation over the temporal data instances. The Euclidean distance measurement or correlation between the observed temporal data is used as a similarity measurement metric that is constructed to classify the instances correctly. The iterative clustering procedure is carried out to identify the sequence of the values with unique temporal data having similar length in the form of vector. Hence, the Euclidean distance between the observed temporal pair is carried out. The clustering of the temporal data using neural networks uses feature vector acquired from the unique temporal data. This method corresponds to the transformation representation and calculates the Euclidean distance between the paired temporal vectors as part from the original temporal data. Hence, the Euclidean distance between the temporal pairs is defined as:

$$D(x,y)=\sqrt{(x,y)\left[\sum(x-y)^{T}\right]^{-1}} \qquad 1.7$$

Where, $x=\{x(t)\}_{t=1}^{T}$ and $y=\{y(t)\}_{t=1}^{T}$ are the temporal data pair and T is the length of each pair.

4.2 Shape Similarity

The shape similarity is very similar to the timing similarity, where the main objective is to find the similarity measurement between the two observed unique temporal data, which varies in speed or time and the common part is observed at varied time is noted. The common pattern or similar sub-pattern refers to the common observation at similar time instant in two different temporal datasets. The dynamic time warping achieves such objective over different temporal data. This warping pattern aligns the temporal sequence pair i.e. it aligns the time series pattern follows the path of warping. This distance metric reduces the usage of Euclidean distance metric. Consider the time series, $x=\{x(t)\}_{t=1}^{T}$ and $y=\{y(k)\}_{k=1}^{K}$ with a warping path $z=\{z_l\}_{l=1}^{L}$ is used to satisfy the following three conditions:

a. Boundary condition: between two sequences makes the warping path to start and finish between the beginning and ending instances with z_1= {x(1), y(1)} and z_L= {x(I), y(J)}.
b. Continuity condition: restricts the required warping paths if z_l= {x(i), y(j)}, then the condition is given as z_{l-1}= {x(i'), y(j')} with i - i' and j - j' < 1
c. Monotonicity condition: forces the pairs in Z to align themselves in a monotonically spaced environment, if z_l = {x(i), y(j)}, then the condition is given as z_{l-1} = {x(i'), y(j')} with i - i' and j - j' > 0.

Further, the DTW is given as:

$$D^{DWT}(x,y)=\min\left(\frac{1}{L}\sum_{l=1}^{L}D^{Euclidean}(z_l)\right) \qquad 1.8$$

4.3 Change Similarity

The change similarity finds the similarity behaviour between the dynamic temporal data, which significantly changes its value or state during the period of observation. Certain generative representations like ARMA and Hidden Markov Model has the ability to acquire the characteristics of such dynamic data.

Depending on log-likelihood characteristics between the sequences, which is generated from adjacent sequences, the distance is considered symmetric between the required sequences. The distance between the sequences $x=\{x(t)\}_{t=1}^{T}$ and $y=\{y(t)\}_{t=1}^{T}$ [24], is given as:

$$D^{sym}(x,y)=\frac{1}{2}\left[L_{xy}+L_{yx}\right] \quad 1.9$$

Instead, the log-likelihood measurement between the distance metric is formulated using:

$$D^{BP}(x,y)=\frac{1}{2}\left[\frac{L_{xy}+L_{xx}}{L_{xx}}-\frac{L_{xy}+L_{yy}}{L_{yy}}\right] \quad 1.10$$

Where ,L_{xy}= log($p(X|\theta_y)$), and from the θ_y, the sequence $y=\{y(t)\}_{t=1}^{T}$.

Depending on the log-likelihood distance between the sequences, resultant generator model and another model is used as a reference point to calculate the Kullback-Leibler distance [23, 24] between the models. This is computed using single, complete, and average linkage methods [25], given as:

$$D^{\min KL}(\lambda_i,\lambda_y)=\min_{x\in C_i}\left(\log p(x\mid\lambda_i)-\log p(x\mid\lambda_j)\right) \quad 2.11$$

$$D^{\max KL}(\lambda_i,\lambda_y)=\max_{x\in C_i}\left(\log p(x\mid\lambda_i)-\log p(x\mid\lambda_j)\right) \quad 2.12$$

$$D^{BoundaryKL}(\lambda_i,\lambda_y)=\frac{1}{|B_x|}\sum_{x\in C_i}\left(\log p(x\mid\lambda_i)-\log p(x\mid\lambda_j)\right) \quad 2.13$$

Where, x is grouped into C_i, $B_x$is denoted as the fraction of x, with smallest value log $p(x|\lambda_x)$, log $p(x|\lambda_y)$. Hence, the value of log $p(x|\lambda_x)$ - log $p(x|\lambda_y)$ is considered zero, since the boundary is estimated between the i and j clusters.

V. MINING TASK

After suitable representation of the dynamic temporal data in a required type and after defining the similarity measurement between the sequences, it is better for the study to rely on an algorithm. The algorithm operates over the temporal data to attain the required task, called as mining operation. Depending upon the different objectives of the dynamic temporal data [4], the task of mining operation is categorized into five types:

- Prediction task
- Classification task
- Clustering task
- Search and Retrieval task
- Pattern discovery task.

Prediction task: Mining operation requires the use of prediction task to forecast the dynamicity of the temporal data or its evolution based on time series pattern with past output. Generative model is built to achieve such task that represents the prediction ability of the temporal data. However, the problem in production task is formulated using clustering, classification and association.

Classification task: Classification is a typical operation to study the environment based on supervised learning, however, most classification algorithms is made to adopt with special procedures, since there exist dynamic change in environment due to the presence of the temporal data. The temporal data classification presents the data, which belongs to class or category in the data environment. The data is trained in prior with the assigned training pattern and the classification task determines the respective category or class automatically for the given datasets.

Clustering task: This task is referred as the collection or a partition of time series data into groups or clusters. Here, the items within a single group possess similar characteristics. The clustering process undergoes unsupervised learning pattern and that determines the basic structure of the temporal data, which is hard visualize. This is because the temporal data possess large voluminous data with very high dimensionality. Hence, the research on such data is specially paid attention and applying the research over varied applications. The main constraints associated with clustering includes: Searching the number of inherent clusters and temporal data grouping using the similarity measurement.

Search and retrieval task: This task detects well the activity of an objective over large sequential databases; however, it searches for the temporal data, which is an important task in data mining. The rapid growth of temporal data makes the search and retrieval processes an important one in exploring such large databases e.g. online media library. However, the main limitation is associated with locating the queries in such archives.

Pattern discovery task: The pattern discovery discovers the exciting patterns that possess periodic and abnormal temporal data characteristics. The interesting discovery over such patterns is now becoming an important task in the data mining process.

This is applicable over any domain, especially, when the domain expert is unavailable, the use of such discovery is used to complete the entire process. Also, use of algorithm is an important function that discovers well the interesting pattern for the specified domain. These algorithms do not hold any prior knowledge or any information in prior. Hence, the present study relies on pattern discovery tasks for predicting the stock market instances.

VI. NEURAL NETWORK TEMPORAL CLASSIFICATION

The most important algorithm that helps in discovering the pattern using the process of classification is the artificial neural network algorithm. This algorithm is designed to get trained for recognizing the sequence within a specified time. This algorithm is associated with the temporal integration process that provides better implementation with fade memory over each node. Further, the usage of interconnect pattern added with the neural network improves the ability of the classifying the data instances. The present study utilizes the neural network to identify the pattern set associated with the temporal integration process, which has the ability to recognize well the sequence of the associated events. Here, the timing element is used by the nodes that hold the fading memory. The individual nodes in the network are considered as an individual sequence or event. A positive excitation signal from the node is send, when the neural network finds the suitable events. The main problem here is, the node that sends the excitation signal is made partially active, since, the nodes with due respect to the prior event is active.

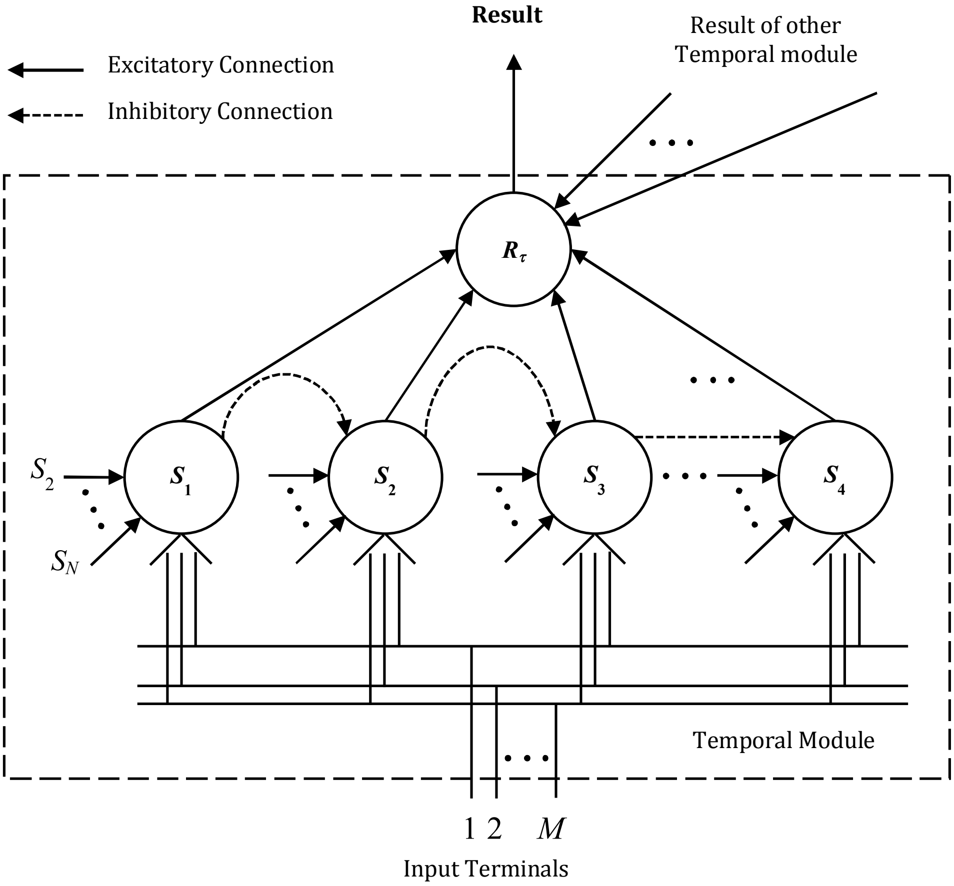

Fig 1.1 Temporal Modules for Neural Network [26]

The entire sequence of the events in the neural network is predicted with the integration of temporal activity associated with the nodes. With the use of required threshold value, the total number of events of the sequences is found with the use of such network, prior the decision control is made. The addition of temporal integration process results in a short noise burst, which appears at the input nodes, however, the results of the successful samples are not affected by such recognition process. The entire process is shown in Figure 1.1.The temporal neural module consists of sequence of nodes, N that varies from S_1 to S_N with a result node. The sequences of each node form excitatory connections between the nodes and that forms a complete forward chain. An inhibitory input is received by the individual sequence of nodes from the other nodes apart from its upstream neighbour. Finally, the output is collected from the individual node sequence, which inturn is connected with the result input node that possesses a static excitatory connection. If the neural network possesses multiple temporal modules, the output of each node receives the inhibitory inputs within a single module from the output nodes of other modules.

The recognized events in the neural network should be stored in a temporary location and that should hold information of an individual event. In temporal recognition module, the timing pattern is formed from the sequence nodes with the fading memory. Hence, the node activated by the particular event remains active over a shorter duration, once the input response is removed. With the other active nodes, the node downstream is fired to form subsequent sequence pattern. Finally, the time flow direction is made active with the excitatory connections with the inputs given to the system at regular intervals. Once the neural node is activated partially w.r.t to an event, the excitation form the other nodes is not encouraged. Upon recognizing the perfect sequences of event, the other nodes send inhibitory signals to the relevant nodes expect the upstream node. Both the excitatory connection and the inhibitory connections, and with the partial past output, the recognition of the temporal sequences from the chained node sequence is enable with external inputs from other modules. The nodes are fired from the left to right location due to use of correct input sequence of events. The correct sequence in the temporal module is used; at least a sequence of node is made active with the use of such correct sequence. This is proved in terms of the time constant function w.r.t fading memory, which should be greater than the inter-arrival rate of the sequence of the input events. This active node sends the positive excitation signal to the result node and that integrates these signals with a summation factor based on time. The summation node is made active, when the integral value of the node exceeds the predefined threshold value of the summation factor. Finally, when the constant integration time factor with threshold value recognizes the total correct input before the firing of the result nodes [27].

VII. PROBLEM FORMULATION

The problems associated with speech processing technique, motion detection technique and predictive signal processing technique is that the information contains the temporal pattern sequence. In order to handle static problem associated with the network, the artificial neural network algorithm is used. The solution that fits the static problem is achieved using the temporal pattern recognition. The integration of other algorithms with neural network is carried out, since most of the algorithms fails to identify the temporal data patterns considering buffers and time delay as an important factor. No proper conventional solutions are available to transform the temporal data sequence into static patterns that provides indirect time representation. If such solution exists, it leads to integration of temporal process, which recognizes the sequences of event w.r.t. time. Hence, a solution to such problem is brought about by the use of hybrid algorithm that integrates neural network with genetic algorithm and neural network with fuzzy logic to carry out the temporal data sequence. This hybrid solution is encouraged with timing element that introduces nodes with fading memory. Here, each node in the neural network is represented as an event and an excitation signal is sent to the nodes. The current node is activated partially, which represents the past event. The sequences are temporally integrated with the activity of the neural nodes, which represents the sequence of relevant events. The threshold data determines the total number of events and the detected sequence before making a better decision. To further, improve the results of real-time data, the hybrid algorithms combined with temporal objective is used.

7.1 OBJECTIVE OF THE WORK

The main aim of the proposed study is to develop a hybrid temporal model that provides learning pattern for classifying the temporal data. These results are unusual, which is in contrast with the Hidden Markov Models (HMM). The system is evaluated in terms of the capabilities of a hybrid learning algorithm, which is applied over the temporal data. The data involved is in this research e.g. stock market data with more of temporal information is fed over the hybrid learning algorithm. Hence, the study takes into account the real world dynamic data i.e. stock market prediction data. Performance of the hybrid algorithm depends entirely on the dynamic data, which is fed into the system. The data fitting is an important concern, to find, analyse and predict the future instance. Hence, the difficulty in making a hybrid algorithm to fit the dynamic data is increasing, however, the data fits in better proportion over the expert system. An expensive research is required to build the required module for data pre-processing, analyzing and prediction. Also comparing such systems' performance with the conventional schemes is required to prove its effectiveness. The study aims at developing a most generic artificial neural network hybrid algorithm, which predicts well the stock market data without the knowledge of past outputs. Hence, the end user does not trouble the recognition system and that is regarded as the virtues of soft computing tools. The main objectives of the proposed techniques are stated as follows:

1. To encompass the temporal pattern recognition through various stages that evolves over time: modeling, recognition, classification, pattern generation, analysis and prediction.
2. To strongly develop an interpretation, extraction and prediction of stock market data using temporal data patterns for better decision making.
3. To carry out powerful analysis over real time dynamic data that changes w.r.t time. The information is stored in files, repositories and databases of stock market.

4. To develop a dynamic technique that performs temporal pattern recognition using hybrid meta-heuristic models. This involves developing and combining two major algorithms and use it for predicting the stock market data.
5. To develop a hybrid model that involves combination of neural network with genetic algorithm that ensures better generalization capability and optimization on predicting the temporal data.
6. To develop a hybrid model that involves combination of neural network with Fuzzy system that ensures better adaptation and greater efficiency over the temporal data.
7. To implement a fine tune strategies for evaluating the stock market data for predicting the future changes in stock market and evaluating the accuracy of testing hybrid models.

7.2 CONTRIBUTIONS OF THE WORK

The contributions of the proposed study involve the use of hybrid approach over real time data, which is briefed as follows:

1. The system is designed to undergo the certain stages for processing the temporal data, which provides effective results in predicting the future outcomes of the stock market data. This involves: data preprocessing that fits with the hybrid neural network model, detection of relevant data as per the current instances, recognition module for predicting the future instances and decision modules for accurate prediction.
2. A hybrid neural network model is developed, which combines the use of artificial neural network (ANN) algorithm with genetic algorithm. Genetic algorithm is used in finding the parameters using its chromosome pattern. Here, the neural network is made to adapt with the genetic algorithm, which fits accurately with the stock market data. Hence, the over-fitting problem is eliminated, which arises in the presence of noise. Also, genetic algorithm is made to perform more local search in the temporal data and the ANN performs the global data search with back-propagation. The module is shown in Figure 1.2.

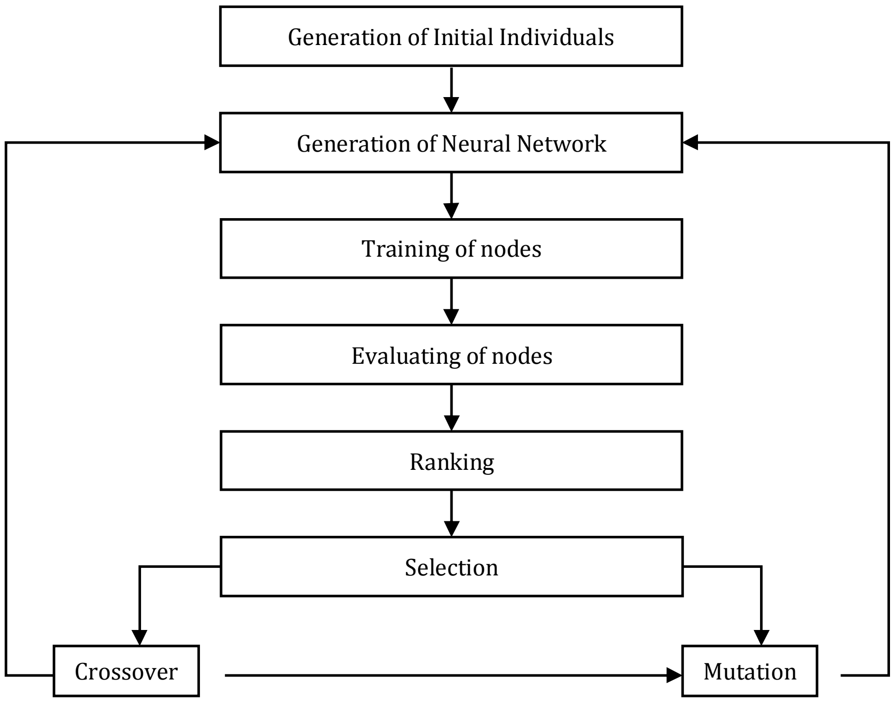

Figure 1.2: Genetic Algorithm with ANN

3. Another, hybrid model is developed with the help of ANN with Fuzzy logic system. Here, the rules are defined in the interference engine to carry out the process of prediction. The ANN is mainly responsible for handling the data for processing and fuzzy logic maintains the reasoning task with the defined rules. The modifications are done to handle the dynamic change and prediction of the stock market data.
4. Finally, a comparisons is made between the proposed hybrid models using the evaluation sets that includes mean absolute deviation, mean squared error, mean absolute percentage error and root mean squared error are calculated to find the accuracy of the hybrid models. Thereby, maintain low false positive ratio of the given stock market inputs using the proposed algorithms.

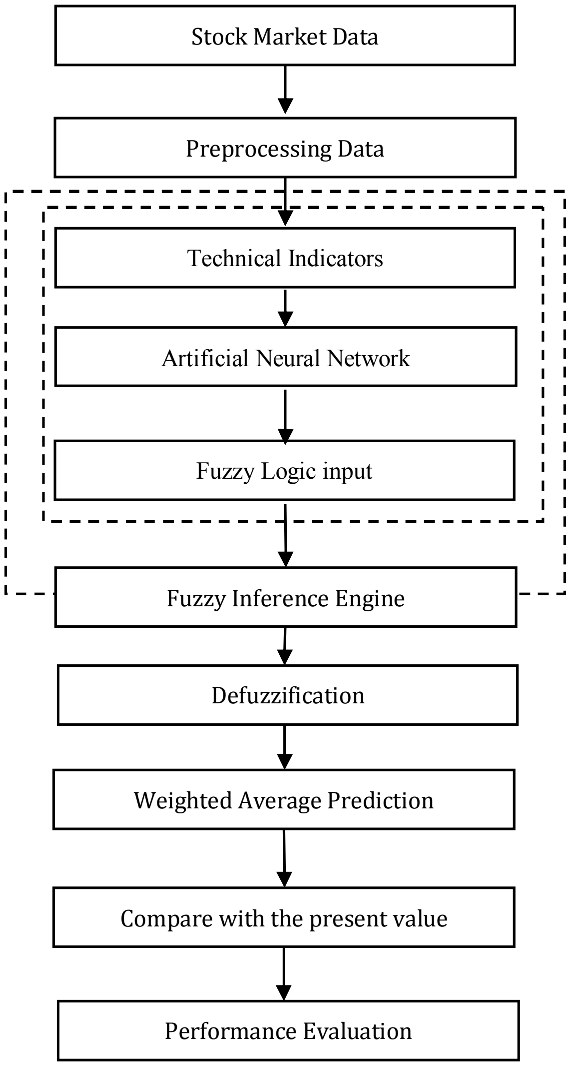

Figure 1.2: Fuzzy Logic with ANN

REFERENCE

1. Sung, C.H., 1988, July. Temporal pattern recognition. In Neural Networks, 1988. IEEE International Conference on (pp. 689-696). IEEE.
2. Chan, Golub, & LeVeque, Algorithms for computing the sample variance: Vol analysis and recommendations (1983), The American Statistician, (37) 242-247.
3. Grossberg (1986) in Pattern recognition bv humans and machines, Vol. 1, eds. Schwab & Nusbaum, Academic Press, 216-226.
4. Lin, W., Orgun, M.A. and Williams, G.J., 2002, December. An overview of temporal data mining. In Proceedings of the 1st Australian data mining workshop (pp. 83-90).
5. Keogh, E., Chakrabarti, K., Pazzani, M. and Mehrotra, S., 2001. Locally adaptive dimensionality reduction for indexing large time series databases. ACM SIGMOD Record, 30(2), pp.151-162.
6. Bashir, F., 2005. MotionSearch: Object motion trajectory-based video database system—Index, retrieval, classification and recognition (Doctoral dissertation, PhD thesis, Dept. of Electrical Eng., Univ. of Illinois, Chicago).
7. Dimitrova, N. and Golshani, F., 1995. Motion recovery for video content classification. ACM Transactions on Information Systems (TOIS), 13(4), pp.408-439.
8. Chen, W. and Chang, S.F., 2000, January. Motion trajectory matching of video objects. In Storage and Retrieval for Media Databases (Vol. 3972, pp. 544-553).
9. Faloutsos, C., Ranganathan, M. and Manolopoulos, Y., 1994. Fast subsequence matching in time-series databases (Vol. 23, No. 2, pp. 419-429). ACM.
10. Sahouria, E. and Zakhor, A., 1997, October. Motion indexing of video. In Image Processing, 1997. Proceedings., International Conference on (Vol. 2, pp. 526-529). IEEE.

11. Yang, Y. and Dr Supervisor Chen, 2006. Clustering Ensemble with Multiple Representations for Temporal Data Clustering Analysis. University of Manchester.
12. Yang, Y. and Chen, K., 2006, July. An ensemble of competitive learning networks with different representations for temporal data clustering. In Neural Networks, 2006. IJCNN'06. International Joint Conference on (pp. 3120-3127). IEEE.
13. Yang, Y. and Chen, K., 2007. Combining Competitive Learning Networks of Various Representations for Sequential Data Clustering. In Trends in Neural Computation (pp. 315-336). Springer Berlin Heidelberg.
14. Bagnall, A., Keogh, E., Lonardi, S. and Janacek, G., 2006. A bit level representation for time series data mining with shape based similarity. Data Mining and Knowledge Discovery, 13(1), pp.11-40.
15. Keogh, E., Chakrabarti, K., Pazzani, M. and Mehrotra, S., 2001. Dimensionality reduction for fast similarity search in large time series databases. Knowledge and information Systems, 3(3), pp.263-286.
16. Policker, S. and Geva, A.B., 2000. Nonstationary time series analysis by temporal clustering. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 30(2), pp.339-343.
17. Baum, L.E. and Eagon, J.A., 1967. An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. Bull. Amer. Math. Soc, 73(3), pp.360-363.
18. Baum, L.E. and Sell, G., 1968. Growth transformations for functions on manifolds. Pacific Journal of Mathematics, 27(2), pp.211-227.
19. Viterbi, A., 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE transactions on Information Theory, 13(2), pp.260-269.
20. Forney, G.D., 1973. The viterbi algorithm. Proceedings of the IEEE, 61(3), pp.268-278.
21. Dempster, A.P., Laird, N.M. and Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the royal statistical society. Series B (methodological), pp.1-38.
22. Bagnall, A. and Janacek, G., 2005. Clustering time series with clipped data. Machine Learning, 58(2), pp.151-178.
23. Sinkkonen, J. and Kaski, S., 2002. Clustering based on conditional distributions in an auxiliary space. Neural Computation, 14(1), pp.217-239.
24. Juang, B.H. and Rabiner, L.R., 1985. A probabilistic distance measure for hidden Markov models. AT&T technical journal, 64(2), pp.391-408.