Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 2458-9403
Vol. 3 Issue 4, April - 2016

Forecasting Shares Trading Signals With Finite State Machine Variant

***Ismaila W.Oladimeji.**
Department of Computer Science & Engineering,
Ladoke Akintola University of Technology,
Ogbomoso, Nigeria woismaila@lautech.edu.ng

Ismaila Folasade. M.
Department of Computer Science,
Osun State Polytechnic, Iree, Nigeria
ismailafolasade@yahoo.com

.

***Abstract*—Shares or Stocks trading has been developed for over twenty years, and has gone deeply into all aspects of daily economic life and attracted more and more investors' attentions. Therefore, researches on finding internal rules and establishing an efficient stock forecast model to help investors minimize risks and maximize returns are very practical and amazing. Computer science plays vital role to solve this problem. Different techniques are available for the prediction of stock market. Very popular some of these are Neural Network, Data Mining, Hidden Markov Model (HMM) And Neuro-Fuzzy system. From this Hidden Markov Model is the most leading machine learning techniques in stock market index prediction area. This paper is aimed at modeling and predicting the stock market situations with the concept of a Hidden Markov Model. The stock market signals considered for the model are hold, buy and sell for weekly and monthly data analysis. The model is being trained and tested HMM variants, that is, Viterbi, Forward-Viterbi and Baum-Welsh algorithms. The performance evaluation of the three algorithms were compared and analysed. The results showed that Forward-Viterbi and Baum-Welsh gave good directional prediction and a very low misclassifications.**

Keywords—Stock market, Computer Science, Hidden Markov Model, Posterior-Viterbi, Baum-Welsh

I. Introduction

The financial market is a complex, evolutionary, and non linear dynamical system. The field of financial forecasting is characterized by data intensity, noise, non stationary, unstructured nature, high degree of uncertainty, and hidden relationships. Therefore, predicting finance market price movements is quite difficult. Increasingly, according to academic investigations, movements in market prices are not random. These days stock prices are affected due to many reasons like company related news, political events, natural disasters … etc. The fast data processing of these events with the help of improved technology and communication systems has caused the stock prices to fluctuate very fast. Thus many banks, financial institutions, large scale investors and stock brokers have to buy and sell stocks within the shortest possible time. To invest money in the stock market there is need to have an idea whether the prices of stocks are going to increase or decrease on the next day. Thus in this work we are trying to predict whether it is safe for stockers to sell or buy shares. [1].

The approaches to predict stock price could be categorized into two kinds, fundamental analysis and technical analysis. Fundamental analysis is based on macroeconomic data, such as exports and imports, money supply, interest rates, foreign exchange rates, and the basic financial status of companies such as dividend yield, cash flow yield, etc. Technical analysis requires history of market. "*The idea behind technical analysis is that constantly changing attributes of investors in response to different forces/factors make trends/move of stock prices"*. Prediction is made by exploiting implications hidden in past trading activities, and by analyzing patterns and trends shown in price and volume charts. [2]. Some of the methods of analyzing stocks to predict if the following day's closing price would increase or decrease. Include were Typical Price (TP), Relative Strength Index (RSI), and Moving Average (MA) (extracted from [3]**).** Although technical indicators appear to be very helpful, there are many shortcomings. First, most indicators only analyze historical prices and do not take any other factor into account. Thus, it is limited to what kind of information it can take in as an input. This becomes problematic because it does downtrend/uptrend. This is where statistical learning theories, such as HMMs can make a significant difference. They allow us to model the market based on many different information. For instance, it can model how historical prices, fundamentals, and current news affect stock prices. They eliminate the weaknesses of being limited to just analyzing historical prices and they can be exploited to take other factors in the economy into account. [4]

Around the world, the Hidden Markov Models (HMM) are the most popular methods in the machine learning and statistics for modeling sequences. According to the number of patent applications for speech recognition technology from 1988 to 1998, the trend shows that this method has become very mature. In a statistics framework, the HMM is a composition of two stochastic processes, a Hidden Markov chain, which accounts for temporal variability, and an observable process, which accounts for spectral variability. The combination contains uncertainly status just likes the stock walk trace. HMM has the following advantages viz, Strong mathematical foundations, Computational efficiency in learning and classifying/predicting, and Robust handling of over time changing data.

II METHODOLOGY

A. Hidden Markov Model

The Hidden Markov Model (HMM) is a finite set of states, each of which is associated with a probability distribution. Transitions among the states are governed by a set of probabilities called transition probabilities. In a particular state an outcome or observation can be generated, according to the associated probability distribution. It is only the outcome, not the state visible to an external observer and therefore states are ``hidden'' to the outside; hence the name Hidden Markov Model. In order to define an HMM completely, following elements are needed.

1. The number of states of the model, *N*. Let denote the set of states S = {S_1, S_2,…S_N}, where S*i*, i=1,2,….N is an individual state.
2. The number of observation symbols per state, *M*. Let denote the set of symbols V = {V_1, V_2,…V_M}, where V_i, i =1,2,….M is an individual symbol.
3. A set of state transition probabilities, that is, A=[a_{ij}], where
   $a_{ij}=p\{q_{t+1}=S_j \backslash q_t=S_i\}$, 1≤i,j≤N ; t=1, 2,…
   where q_t denotes the current state.
4. The observation symbol probability distribution in each of the states, B = {$b_j(k)$}, where
   $b_j(k)=P(V_k \backslash S_j)$, 1≤j≤N, 1≤k≤M
5. The initial state distribution, π = [π_i], where π_i = P{q_1 = S_i},1≤ i ≤ N, such that $\sum_{i=1}^{N} \pi i = 1$

Therefore, we can use the compact notation
λ = (A, B, π) (1)

As stated by [5], HMM is a variant of a finite state machine having a set of states, Q, an output alphabet, O, transition probabilities, A, output probabilities, B, and initial state probabilities, Π. The current state is not observable. Instead, each state produces an output with a certain probability (B). The HMM has variant algorithms that are been modified to produced better outputs. Three of these variants are considered in this work as stated below.

1. ***The Viterbi algorithm***: Finding the most probable path.

• Initialization: v_0 (0) = 1; $v_k(0)$ = 0, for *k* that are not silent states

• Recursion for emitting states (*i* =1…*L*):

$$v_l(i) = e_l(x_i)\max_k\left[v_k(i-1)a_{kl}\right]$$
$$\mathrm{ptr}_i(l) = \arg\max_k\left[v_k(i-1)a_{kl}\right] \quad (2)$$

• Recursion for silent states

$$v_l(i) = \max_k\left[v_k(i)a_{kl}\right]$$
$$\mathrm{ptr}_i(l) = \arg\max_k\left[v_k(i)a_{kl}\right] \quad (3)$$

• Termination

$$P(x,\pi) = \max_k \left(v_k(L)a_{kN}\right)$$
$$\pi_L = \arg\max_k \left(v_k(L)a_{kN}\right) \quad (4)$$

• Traceback; follow pointers back starting at π_L

2. ***Posterior-Viterbi Decoding***

The specific problem to be solved by the Posterior-Viterbi algorithm is to use posterior decoding to find the path which maximizes the product of the posterior probability of the states and after having computed the posterior probabilities, Viterbi algorithm is used to find the best allowed posterior path through the model. In the PV algorithm, the basic idea is to compute the path π^{PV}

$$\pi^{PV} = argmax_{\{\pi \in A_p\}} \prod_{i=1}^{L} P(\pi_i \mid O, M) \quad (5)$$

Where A_p is the set of the allowed paths through the model, and *P*(π_i |*O*,*M*) is the *posterior* probability of the state assigned by the path πat position *i* (as computed in equation 5).

3. ***Baum-Welch algorithm***

The Baum-Welch re-estimation formulas aim at adjusting the parameters of the model λ = (A,B, π), such that we achieve the maximum value of P(O | λ). Noting that

$$\sum_{t=1}^{T-1} \gamma_t(i) = \text{Expected number of transitions from state i}$$
$$\sum_{t=1}^{T-1} \varepsilon_t(i,j) = \text{Expected number of transitions from state i to state j} \quad (6)$$

Then, the Baum-Welch alpha-beta recursion update equations are as follows:

$$\bar{\pi}_i = \gamma_t(i), \quad 1 \le i \le N$$
$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$
$$\hat{b}_j(k) = \frac{\sum_{t=1 \text{ and } O_t=k}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} \quad (7)$$

B. HMM Structures

Figure 1 shows examples of HMM structure. Figure 1 (a) shows a 3-state ergodic model, in which every state of the model could be reached from every other state of the model in a single step, and Fig. 1 (b) shows a 3-state left-to-right model, in which the state index increases or stays the same as time increases. Some authors employed left-right HMM model because they believe that there can be no transition from higher order states to lower order states. But in this work, ergodic model will be employed to give room for transitions from one states to another and vice versa.

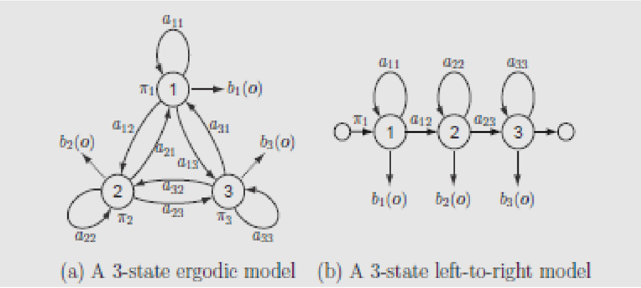

Figure 1: Examples of HMM structure.

B. Training of HMM

Processing of the input sequence on a given HMM develops a HMM training diagram, which we call "HMM with all estimated parameters". Those parameters include transition/emission probabilities. HMM with all estimated parameters" is also called "training of HMM" [6]. This "training of HMM", as shown in figure 2, gives all possible sequences of states or categories which can be assigned to the input sequence. All these possible sequences of states or categories are also called hidden states sequence because these sequences are not known unless we train the original HMM for a given input sequence. Each hidden state sequence of states or categories assigned to the input sequence is also called the output sequence. Brute Force expansion of the HMM is usually intractable for most real world classification problems, as the number of possible hidden state sequences is extremely high and scales exponentially with the length of input sequence.

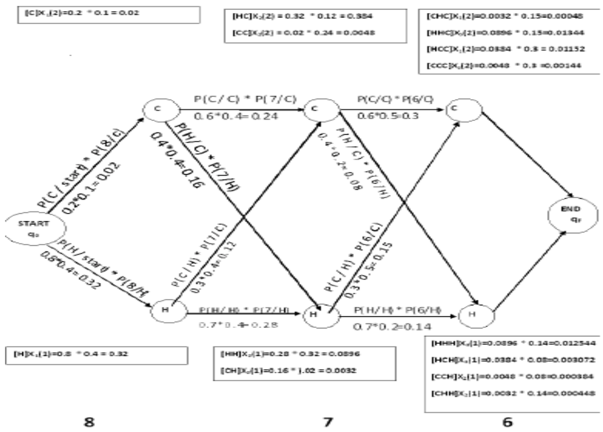

Figure 2. Sample of HMM training diagram

All output sequences that are assigned to a given input sequence have probability of likelihood. This probability defines how likely the output sequence is as an appropriate assignment for the input sequence. These probabilities of likelihood are calculated with the help of transition probabilities and emission probabilities of the original HMM. For a given input sequence, HMM chooses the state sequence that maximizes in the following formula.

*P(input symbol/state) * P(state/previous 'M' states)*

In the above formula, Transition probability = P(state/previous 'M' states) and Emission probability = P(input symbol/state) . Transition probabilities (TPs) of each state of HMM either depend on one previous state or more than one previous state. If they depend on one previous state (i.e., M=1), this HMM is called a 1st order HMM. If all TPs of states of HMM depend on two previous states (i.e., M=2), this HMM is called a 2nd order HMM and so on. [7].

C. Prediction Using Hidden Markov Model

Hidden Markov Models are heavily used to solve classification problems. Prediction is a new domain and not much work been done here. To predict with HMM we will utilize our knowledge of the past data. The rational is if something happened in the past i.e. some pattern appeared, than it will with high probability appear in the future. Hence we will look for resembling pattern and within this patter most probable observation that will be used to calculate our prediction results.

Given trained Hidden Markov Model we calculate the likelihood value of last observation, o_t, in the sequence. From historical data we collect all observations with likelihood in predefined interval, Δ, from likelihood of o_t. The base assumption is that the predicted observation will act similarly to past observation patterns. We calculate the difference between most similar likelihood observation to o_t, o_i and observation o_{i+1} that follows it. The difference $o_t+(o_{i+1}-o_i)$ is returned as our prediction. In case that no observations are found in predefined likelihood range Δ we use adaptive approach and take observation that is closest to o_t without range limitation. Our algorithm can run on single day or on predefined days range. In case of date range we add predicted values to historical data after each iterations. [3].

D. Mapping of Stock Market to HMM

In order to map the stock market operations in terms of an HMM, it is better to decide the hidden states and observation symbols. The situations of stock market are Bear (when prices are falling), Bull (when the market is generally rising) and stable (no change) represent the observation symbols. The urge that the shareholders' have to sell, buy or hold the stocks in transactions form the hidden states at the stock house. The diagram of the mapping is shown in figure 3.

The States are not directly observable. The situations of the stock market are considered hidden. Given a sequence of observation we can find the

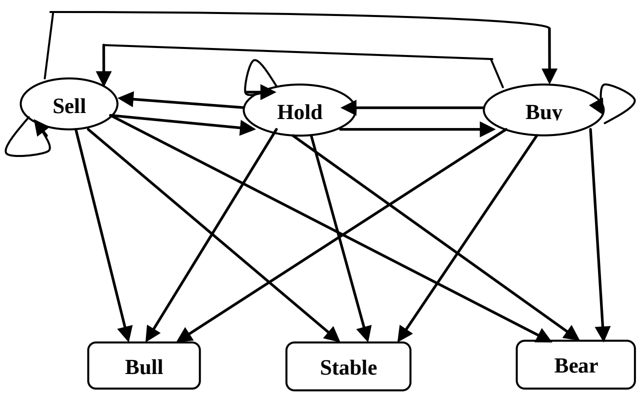

Figure 3: Three-State Ergodic HMM

hidden state sequence that produced those observations.

So this HMM has • 3 states: "sell", "buy", and "hold". • 3 observation symbols: "bear", "bull", "stable".

III. IMPLEMENTATION

There are a number of sources of stock and financial data on web including Google Finance and Yahoo! Finance. It seems that Google Finance does not provide the required data format as required. As an alternative, Yahoo! Finance was used to load and store the stock data from 2003 to 2013. Data-analysis reports for weekly and monthly highest, lowest and average prices are shown in Fig 4.

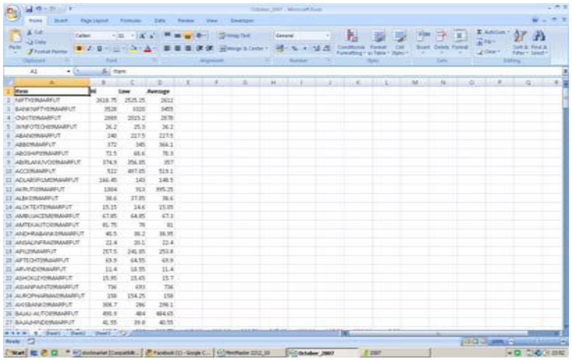

Fig. 4. Sample of Stocks Analysis .

The data was compressed to 450 weekly and 100 monthly data points to take care of some missing data. Also, weeks and months that contain no transaction is assumed to be the hold signal. The data set is divided into two parts; 60% of the data is used to build and train the model, and the remaining 40% is reserved for out-of-sample evaluation.

Supposing the initial input is given. The probability values of Transition Probability Matrix, Emission Probability Matrix and Π for difference in twelve weeks and six months close values are estimated as shown in figure 5.

Let Sell→s ; Buy→b ; Hold→ h.; Bear → e; Bull→ u; Stable→t

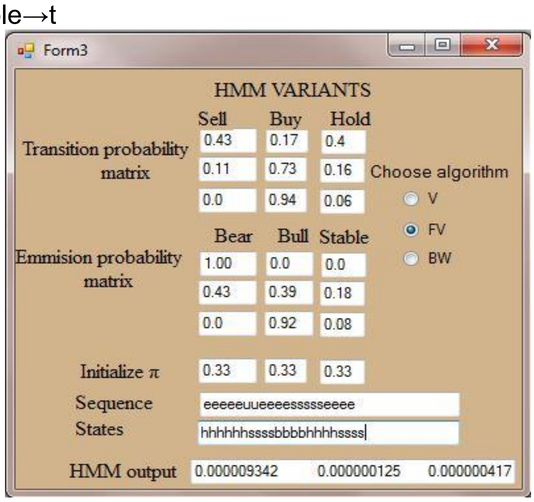

Fig. 5: Sample Output of HMM program

IV. DISCUSSION OF RESULTS

Firstly, the distribution of out of sample data among the three classes was investigated. Table 1 shows the distribution of the actual 'Buy', 'Hold' and 'Sell' signals within the test sample (the first 180 weeks and 40 months). Note: (SS-Sell Signal, BS-Buy Signal, HS- Hold Signal; Viterbi-V; Posterior-Viterbi-PV; Baum-Welch- BW)

Table 1: Distribution Data belongs to Test Samples

	SS	BS	HS
Weeks	42	59	79
Months	8	11	21

The model is being trained by HMM training algorithms viz; Viterbi, Posterior-Viterbi and Baum-Welch. The HMM algorithms estimation selection process is then followed by an empirical evaluation which is based on the out-sample data. Table 2 shows the result of predictions of each algorithm.

Table 2: Results of Correctly Classified Stock Signals

HMMs	SS		BS		HS	
	Wks	Mths	Wks	Mths	Wks	Mths
V	26	4	29	6	44	11
PV	31	4	38	7	56	13
BW	37	6	54	9	65	17

The relative performance of the algorithms is measured by hit ratio (Table 3 shows the experiment results). We use direction prediction to test the performance of the model. [8, 9]. If the predicted signal is the same with present signal and then it is counted as one time correct prediction. Direction prediction percentage (hit_ rate) is calculated as follows:

$$\text{Hit\_rate} = h / n \qquad (8)$$

where h is the number of correct prediction and n is the number of weeks/months in testing data.

Table 3: Prediction performance of HMMs Using Hit Ratio.

HMMs	SS		BS		HS	
	Wks	Mths	Wks	Mths	Wks	Mths
V	0.14	0.1	0.16	0.15	0.24	0.27
PV	0.17	0.1	0.21	0.18	0.31	0.33
BW	0.21	0.15	0.3	0.22	0.36	0.43

The result from table 3 shows that the BW produced the highest prediction accuracy with total hit ratio of 0.87 and 0.8 for weekly and monthly prediction performance respectively, followed by Posterior-backward algorithm which produced total hit ratio of 0.69 and 0.61 for weekly and monthly prediction performance respectively, while Viterbi algorithm produced total hit ratio of 0.54 and 0.52 for weekly and monthly prediction performance respectively.

However, some signals are misclassified, e.g., Hold signal is being classified as Buy signal. From a trader's point of view, the misclassification of a 'Hold' signal to 'Buy' class or 'Sell' class is a more serious mistake than misclassifying a 'Buy' signal or a 'Sell' signal as a 'Hold' signal. The reason is in the former case a trader will loses the money by taking part in an unwise investment while in the later case he/she only lose the opportunity of making a profit, but no monetary loss. The most serious mistakes are the misclassification of 'Buy' signal to 'Sell' signal and vice versa. For instance the results of misclassification of the three algorithms for weekly data are shown in table 4.

Table 4: Misclassification results from the three algorithms

HMMs	SS-BS	SS-HS	BS-SS	BS-HS	HS-SS	HS-BS
V	10	6	18	12	19	16
PV	5	6	14	7	13	10
BW	2	4	-	4	8	6

It can be observed from table 4 that although no serious misclassifications were predicted by the three algorithms, both Posterior-viterbi and Baum Welch algorithms give hope that traders can benefit from the predictions obtained. As a result of highest classification rate corresponding to 'Buy' signals, the Baum Welch yields higher profits than others.

V. CONCLUSION

Today the stock price prediction has become very complex than before as stock prices are not only affected due to company's financial status but also due to socio economical condition of the country, political atmosphere and natural disasters etc. The return from the share market is always uncertain and ambiguity in nature hence traditional techniques will not give accurate prediction. In this paper, a stochastic tool, the hidden markov models, has been used to perform a stock prediction using three variants of HMM algorithm. The study considered weekly and montly data for ten years range. The results of the study provide evidence that the Baum Welsh performed better than Viterbi and Posterior-Viterbi algorithms by means of performance evaluation. In the future, we hope to carry out an empirical study to see how useful financial managers would consider this application. Also, we find imperative to compare the HMM forecasting performance with that of linear discriminant analysis, supervised and unsupervised neural networks, and support vector regression.

VI. RELATED WORKS

Investors in stock market primarily traded stocks based on intuition before the advent of computers. The continuous growth level of investing and trading necessitate a search for better tools to accurately predict the market in order to increase profits and reduce losses. As reported by [10], Gupta and Dhingra used Hidden Markov Model (HMM's) for the predicting the stock market. By using historical stock prices they present he Maximum a Posteriori HMM approach for forecasting stock values for the next day. For training the continuous HMM they consider the intra day high and low values of the stock and fractional change in stock values. Over all the possible stock values for the next day this HMM is used to make a maximum posteriori decisions. Also [11] in which alternative scenario is investigated with a novel methodology, aimed at analyzing short (local) financial trends for predicting their sign (increase or decrease). This is achieved by modeling directly the signs of the local trends using *two* separate Hidden Markov models, one for positive and one for negative trends. The authors in [12], in their thesis adopted generally Left-Right Hidden Markov Model. The HMM is assumed that the day trade signal was produced status series (hidden layer) by a random process. Then produce an observation sequence by the status series and the other random process.

REFERENCES

[1]. P. K. Sahoo, K. Charlapally (2015).Stock Price Prediction Using Regression Analysis, International Journal of Scientific & Engineering Research, Volume 6, Issue 3.

[2]. R. K. Dase, D. D.Pawar and D. S.Daspute (2011) Methodologies for Prediction of Stock Market: An Artificial Neural Network. International Journal of Statistika and Mathematika, vol. (1).

[3]. L. Brailovskiy and M. Herman. Prediction of Financial Time Series Using Hidden Markov Models, Dept. Mathematics and Computer Science, The Open University of Israel, Raanana, Israel

[4] R. Satish and H Jerry. (2010). Analysis of Hidden Markov Models and Support Vector Machines in Financial Applications, Technical Report No. UCB/EECS-2010-63

[5] Y. Ephraim and N. Merhav (2002). Hidden Markov processes. IEEE Transactional Information Theory. vol. 48, pp. 1518-1569.

[6] R. Hassan (2009). A combination of hidden markov model and fuzzy model for stock market forecasting. Elsevier, 2009.

[7] A. Jamil, M.W. Stephen, and A. Sherji (2012). State Complexity of Hidden Markov Model, ICCGI 2012: The Seventh International Multi-Conference on Computing in the Global Information Technology .

[8] R. B. Caldwell, (1999). Performances Metrics for Neural Network-based Trading System Development, NeuroVe$t Journal, 3(2). 22-26.

[9] A. N Refenes,., A. Zapranis, and G. Francis (1994). Stock Performance Modeling Using Neural Networks: A Comparative Study with Regression Models. Neural Network, Vol 5, pp961-970.

[10] S., Kute S., Tamhankar (2015). A Survey on Stock Market Prediction Techniques, International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064, Volume 4 Issue 4.

[11] B. Manuele G. Enrico and O. Edoardo (2008).A Hidden Markov Model Approach to Classify and Predict the Sign of Financial Local Trends, SSPR&SPR 2008, LNCS 5342, pp. 852–861, 2008. Springer-Verlag Berlin Heidelberg 2008.

[12] Wen-Chih Tsai An-Pin Chen (2002).Using Hidden Markov Model for Stock Day Trade Forecasting, The Second International Conference on Electronic Business, Taipei, Taiwan, December 10-13.