A neural network based dynamic forecasting model for Trend Impact Analysis

Nedaa Agami a, Amir Atiya b, Mohamed Saleh a,* Hisham El-Shishiny c

a Decision Support Department, Faculty of Computers and Information, Cairo University, Giza, Egypt
b Computer Engineering Department, Faculty of Engineering, Cairo University, Giza, Egypt
c Advanced Technology and Center for Advanced Studies, IBM Cairo Technology Development Center, Egypt

ARTICLE INFO

Article history:
Received 23 August 2008
Received in revised form 9 December 2008
Accepted 19 December 2008

Keywords:
Trend Impact Analysis
Forecasting
Neural networks

ABSTRACT

Trend Impact Analysis is a simple forecasting approach, yet powerful, within the Futures Studies paradigm. It utilizes experts’ judgements to explicitly deal with unprecedented future events with varying degrees of severity in generating different possibilities (scenarios) of how the future might unfold. This is achieved by modifying a surprise-free forecast according to events’ occurrences based on a Monte-Carlo simulation process. Yet, the current forecasting mechanism of TIA is static. This paper introduces a new approach for constructing TIA by using a dynamic forecasting model based on neural networks. This new approach is designed to enhance the TIA prediction process. It is expected that such a dynamic mechanism will produce more robust and reliable forecasts. Its idea is novel, beyond state of the art and its implementation is the main contribution of this paper.

© 2009 Elsevier Inc. All rights reserved.

Keywords: Trend Impact Analysis Forecasting Neural networks

1. Introduction

“Futures Studies is the science, art and practice of postulating possible, probable and preferable futures, and the worldviews and myths that underlie them” [1]. It seeks to understand what is likely to continue, what is likely to change and what is novel through analyzing the sources, patterns, and causes of change and stability. In an attempt to develop foresight and map possible futures within the Futures Studies paradigm, futurists employ a wide range of methodologies, either purely quantitative, purely qualitative or a combination of both, to cover the diverse different views of the future [2–4]. One prominent hybrid Futures Studies method is the Trend Impact Analysis (TIA).

As Gordon, the founder of the method puts it—“TIA is a simple approach to forecasting in which a time series is modified to take into account perceptions about how future events may change extrapolations that would otherwise be surprise-free. It permits an analyst, interested in tracking a particular trend, to include and systematically examine the effects of possible future events that are believed to be important.” [5, p.3] The major steps of TIA as defined by Gordon—are as follows:

1. The Forecasting engine generates the base-case scenario (a surprise-free forecast) using an appropriate forecasting technique.
2. Expert judgments are used to identify a list of unprecedented future events that, if they were to occur, could cause deviations (positive or negative) from the extrapolation of historical data (the source of such a list might be a Delphi study [6,7], some form of other informal consensus among experts or a literature search [8]). For each such event, experts judge the probability of occurrence as a function of time and its expected impact, should the event occur, on the future trend.
3. Using Monte Carlo Simulation, the TIA algorithm combines the impact and event-probability judgments with results of the base-case scenario to generate a randomly selected subset of possible future scenarios. Note that, the usual practice is to generate a very large number of scenarios (e.g. one million).

* Corresponding author.
E-mail addresses: n.agami@fci-cu.edu.eg (N. Agami), amir@alumni.caltech.edu (A. Atiya), Saleh@SalehSite.info (M. Saleh), shishiny@eg.ibm.com (H. El-Shishiny).

0040-1625/$ – see front matter © 2009 Elsevier Inc. All rights reserved.
The TIA algorithm was further enhanced by developing a mechanism to assess the impact of the occurrence of an unprecedented future event given how severe the occurrence is. This was achieved by allowing the analyst to supply three levels of impact/probability pairs; where each pair is associated with one of three degrees of severity: Low, Medium and High.

The basic steps of the Enhanced approach for TIA are as follows:

1. Randomly generate the Degree of Severity 'D' (see Fig. 3 in [9]).
2. Accordingly (knowing the event and its degree), identify the corresponding event impact parameters: Maximum Impact, Steady-State Impact, Time to Maximum Impact and Time to Steady-State Impact. This is done by indexing the associated matrices.
3. Randomly generate the number of the month 'M' in given year 'Y' on which the event would occur.
4. Compute the Fractional Change Vector using the estimated event impact parameters (see Figs. 4, 5 and 6 in [9]).
5. Update the current scenario (column) 'S' of the Scenarios Matrix accordingly.

While in the literature there was emphasis on studying and analyzing the future events aspect of TIA, there was a little attention paid to the surprise-free forecast aspect. In this paper we focus on this particular issue. We develop a dynamic forecasting mechanism using a neural network model to enhance the prediction process of the current TIA instead of the static mechanism already applied.
The rest of this paper is organized as follows: In Section 2, we explain the research objective. In Section 3, we give an overview on Neural Networks. Then in Section 4, we explain in details our proposed algorithm including the inputs, output and the algorithm itself. Finally in Section 5, we give a numerical example and in Section 6 we conclude.

2. Research objective

TIA, as explained earlier, is a forecasting tool of considerable power that is relatively simple and easy to use. Although it enhances the usefulness of purely quantitative forecasting methods by utilizing qualitative information from experts to generate alternative future scenarios instead of a single surprise-free scenario, it suffers a basic limitation. The current TIA process employs a static forecasting mechanism, i.e. it imports the base-forecast from an external forecasting engine and modifies it according to the unprecedented future events’ occurrences during the simulation process. Our main hypothesis in this paper is that: For any given scenario, in a certain year, if an unprecedented event occurred, then feeding back this information to a neural network and allowing the network to re-train (before forecasting), will produce more responsive and robust forecasts. Moreover to improve this dynamic process, in our research, we iterate this process every year as shown in the following figure.

![Diagram of neural network]

**Fig. 5.** Standard method of performing time series prediction using a sliding window over the input sequence—in this case, three time steps.
This proposed mechanism allows us to extend the forecasting time horizon while minimizing the degradation that occurs as time goes by in the quality of produced forecasts.

A disadvantage of this proposed feedback process is that it takes considerable time to re-train the neural network for each scenario in each year. However, as explained later in Section 4, we have devised a simplified mechanism to overcome this disadvantage.

To summarize this section, our proposed approach suggests a dynamic forecasting mechanism rather than the static one already applied. This is achieved by re-generating the forecasts (for all scenarios) each year after testing the occurrences of unprecedented future events using a NN. Initially, the NN is trained on available historical data; then (for each scenario) before generating a new forecast, the NN is re-trained on the new time-series (specified by item no. 3 in Fig. 1).

3. Neural networks—Theoretical background

A neural network is defined as a powerful data modelling tool that is able to capture and represent complex input/output relationships [10–13]. The motivation for the development of neural network technology stems from the desire to develop an artificial system that could perform "intelligent" tasks similar to those performed by the human brain. Neural networks resemble the human brain in the following two ways:

i. A neural network acquires knowledge through learning.
ii. A neural network’s knowledge is stored within inter-neuron connection strengths known as synaptic weights.

The true power and advantage of neural networks lies in their ability to represent both linear and non-linear relationships and in their ability to learn these relationships directly from the data being modelled. Traditional linear models are often inadequate as many complex problems are characterized by non-linear behaviour. In this paper, one model of neural network is selected among the main network architectures, namely the Multi-Layer Perceptron (MLP). The basis of the model is the neuron structure as shown in Fig. 2. These neurons act like parallel processing units. An artificial neuron is a unit that performs a simple mathematical operation on its inputs and imitates the functions of biological neurons and their unique process of learning.

From the previous figure, we will have:

$$v_k = \sum_{j=1}^{m} x_j w_{kj} + b_k$$

The neuron output will be:

$$y_k = f(v_k)$$

Where $$w_{kj}$$’s are the interconnection weights and $$f$$ is a non-linear activation function (usually the logistic function—see Fig. 4). NNs are characterized in principle by a network topology, a connection pattern, neural activation properties, training strategy and ability to process data. The most common neural network model is the Multilayer Perceptron (MLP). The basis of the model is the neuron structure as shown in Fig. 2. These neurons act like parallel processing units. An artificial neuron is a unit that performs a simple mathematical operation on its inputs and imitates the functions of biological neurons and their unique process of learning.

Table 1

<table>
<thead>
<tr>
<th>Short name</th>
<th>Full name</th>
<th>Type</th>
<th>Dimensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>NumS</td>
<td>Number of Scenarios</td>
<td>Scalar</td>
<td>–</td>
</tr>
<tr>
<td>NumY</td>
<td>Number of Years</td>
<td>Scalar</td>
<td>–</td>
</tr>
<tr>
<td>NumE</td>
<td>Number of Events</td>
<td>Scalar</td>
<td>–</td>
</tr>
<tr>
<td>HistVect</td>
<td>Time Series Vector</td>
<td>Vector</td>
<td>–</td>
</tr>
<tr>
<td>MaxImpMx</td>
<td>Maximum Impact Matrix</td>
<td>2D-Matrix</td>
<td>(NumE, 3)</td>
</tr>
<tr>
<td>SSImpMx</td>
<td>Steady-State Impact Matrix</td>
<td>2D-Matrix</td>
<td>(NumE, 3)</td>
</tr>
<tr>
<td>TMaxMx</td>
<td>Time to Maximum Impact Matrix</td>
<td>2D-Matrix</td>
<td>(NumE, 3)</td>
</tr>
<tr>
<td>TSSMx</td>
<td>Time to Steady-State Impact Matrix</td>
<td>2D-Matrix</td>
<td>(NumE, 3)</td>
</tr>
<tr>
<td>ProbOccMx</td>
<td>Probability of Occurrence Matrix</td>
<td>3D-Matrix</td>
<td>(NumE, NumY, 3)</td>
</tr>
</tbody>
</table>

Table 2

<table>
<thead>
<tr>
<th>Short name</th>
<th>Full name</th>
<th>Type</th>
<th>Dimensions</th>
</tr>
</thead>
<tbody>
<tr>
<td>SrMx</td>
<td>Scenarios Matrix</td>
<td>2D-Matrix</td>
<td>(NumM, 9)</td>
</tr>
</tbody>
</table>
they get summed then processed by the nonlinear activation function (see Fig. 4). As the processed data leaves the first hidden layer, again it gets multiplied by interconnection weights, then summed and processed by the second hidden layer. Finally the data is multiplied by interconnection weights then process one last time within the output layer to produce the neural network output.
To perform any task, a set of experiments of an input/output mapping is needed to train the neural network. These data are one of the most important factors to obtain reliable results from any trained NN.

There are a number of NN learning algorithms. For the purpose of our research, we used the Back Propagation algorithm based on Gradient Descent method.

NNs have various applications that can generally be grouped into four categories: Clustering, Classification (pattern recognition), Function Approximation and Forecasting. When used in forecasting, the NN task is to forecast some future values of a time-sequenced data. First, the NN is trained over the input sequence and the input variables are assumed to be the lags $X(t), X(t-1), \ldots$, etc. Then the prediction is given by (Fig. 5):

$$X_{t+1} = F(X(t), X(t-1), \ldots, X(t-L+1))$$

In a nutshell, a neural network can be considered as a learning model that is able to predict an output pattern when it recognizes a given input pattern. Once trained, the neural network is able to recognize similarities when presented with a new input pattern, resulting in a predicted output pattern [17,18].

Based on the previous and the fact that, a NN is a powerful forecasting tool that can be trained and hence learn from experience, we proposed a NN-based enhancement for TIA. This proposed algorithm is explained in details in the next section.

4. Proposed algorithm

The newly proposed approach suggests re-generating the forecast each year, for each scenario after testing the occurrence of unprecedented future events using a neural network model. It uses the Enhanced TIA [9] as the point of departure. After conducting the steps of the Enhanced TIA and generating the scenarios multipliers matrix, we first train the NN on the available time-series. Then, we use the trained network to produce forecasts for one year ahead. These produced forecasts are then modified using the corresponding multipliers for each scenario. Afterwards, we update the scenarios matrix (where each column represents an output scenario) accordingly. Finally, we append the one year updated values to the end of the old time-series. The previous steps are repeated for each year until the end of the forecasting horizon.

All the assumptions, inputs and output of the enhanced TIA algorithm hold true. However instead of the input Base Forecast Vector (BaseFrVect), we restrict ourselves to use the input Historical time-series (HistVect) added as shown in the next table.
4.1. Inputs

The following table lists the inputs used in our proposed algorithm (Table 1):

4.2. Output

The single output of the algorithm is the Scenarios Matrix shown in the table below (Table 2):

4.3. Algorithm

Below, we shall explain in details, the functions of the proposed approach illustrated in the previous flow charts: Fig. 6 outlines the 'Main' function. Basically the function consists of a triple loops structure (the same case as in the enhanced algorithm). The outer-loop is a counter on the number of scenarios to be generated ($S = 1, ..., \text{NumSc}$). The inter-mediate loop is a counter on the number of years we wish to study ahead ($Y = 1, ..., \text{NumY}$); while the inner loop is a counter on the number events

![Flow chart of the 'GenSrMx' function.](image-url)
incorporated in the study (E = 1,..., NumE). Before entering the inner loop, the Scenarios Multipliers Matrix (Sr_Mult_Mx) is initialized such that each element is set to 1. The inner loop then carries out the same steps of the Enhanced TIA [9] explained earlier resulting in the Scenarios Multipliers Matrix (Sr_Mult_Mx).

Fig. 7 outlined the ‘Compute_Sr_Mult_Mx_Compact’ function. This function computes the set of deciles associated with the set of scenarios. Afterwards, we deal with the computed set of deciles instead of the original set of scenarios to lessen the computational time and effort of the algorithm.

Fig. 8 outlines the function ‘GenSrMx’ function. The function consists of double loops structure. The outer loop is a counter on the number of scenarios to be generated (S = 1,..., NumSc), while the inner loop is a counter on the number years we wish to study ahead (Y = 1,..., NumY). Before entering the inner loop, an auxiliary (vector) variable (TimeSeries) is defined. At the very beginning, the value of this variable is set to equal the historical time series (HistVect). The inner loop then carries out the following steps:

1. Trains the NN on available historical data.
2. Produces forecasts of one year ahead using the trained network.
3. Modifies the values of the current year ‘Y’ in a given scenario ‘S’ using the corresponding multiplier.
4. Updates the scenarios matrix (SrMx) accordingly.
5. Appends the updated year values and the old time series (TimeSeries) to form the new time series.

The previous steps are repeated for each year until the end of the forecasting horizon.

Note that the algorithm was implemented and tested using the MATLAB software and its associated Neural Network toolbox. It was developed in such a way that enables it to be flexible in the number of scenarios, events and years according to the decision maker’s point of view and the data available.¹

This paper is a proof of concept. However, we are currently developing the proposed algorithm using the R Language [19] and the associated AMORE (a more flexible neural network) package. We selected the R Language in order to provide an open source tool that can be easily used by futurists as it is a free scientific software environment for statistical computing.

5. Numerical example

The following example illustrates the use of the new approach:

Assume that the Egyptian Ministry of Tourism needs to study how some unprecedented events would affect the number of tourists’ arrivals in the future if they were said to occur. Two events (NumE = 2) are incorporated in the study: Epidemic Diseases (E1) and Currency Devaluation (E2). Should any of these events occur, the first have negative impacts whereas the second have positive impacts. 10,000 scenarios are generated (NumS = 10,000) and the study is for five years ahead (NumY = 5).

¹ The reader interested in the code can e-mail the authors.
values (time series) for the last fifteen years (180 months) are available. The data of events (supposedly estimated earlier by experts) is listed below (Table 3).

A. Probability of Occurrence Matrix (ProbOccMx)
B. Maximum Impact Matrix (MaxImpMx)
C. Steady-State Impact Matrix (SSImpMx)
D. Time (in months) to Maximum Impact Matrix (TMaxMx)
E. Time (in months) to Steady-State Impact Matrix (TSSMx)

From Tables 4 and 5, it is clear that for every event, the expected impact associated with the high degree of severity is greater than that associated with medium degree of severity which is greater than the one associated with low degree of severity (Tables 6 and 7).

The neural network MATLAB toolbox was used to create a fully connected feed-forward neural network with the following parameter values, and was trained by back propagation technique:

- Number of input nodes (lags) = 10 (months)
- Number of output nodes = 12 (months, i.e. one year)
- Hidden layers and hidden nodes = 1 hidden layers with 3 hidden nodes
- Activation functions = ‘tansig’ (hidden layer) and ‘purelin’ (output layer)
- Learning rate = 0.005
- Error goal = 0.05
- Number of training epochs = 1000
- Momentum factor = 0.1

After running the MATLAB program, we obtain the scenarios matrix. The next figure displays the 90th, 50th and 10th percentile scenarios respectively (Fig. 10).

<table>
<thead>
<tr>
<th>Table 6</th>
<th>TMaxMx.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>High</td>
</tr>
<tr>
<td>E1</td>
<td>4</td>
</tr>
<tr>
<td>E2</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 7</th>
<th>TSSMx.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>High</td>
</tr>
<tr>
<td>E1</td>
<td>7</td>
</tr>
<tr>
<td>E2</td>
<td>2</td>
</tr>
</tbody>
</table>

Fig. 9. Historical time series.
We have run several experiments to validate the developed algorithm. The conducted experiments included incrementally changing the impact values, directions and the probability of occurrence for each level of severity. By visually inspecting the figures associated with these experiments (as shown in Fig. 7), we can verify that the algorithm works logically and responds to changes in a systematic way.

6. Concluding remarks and possible future work

- The proper way to study the future is in terms of alternatives (scenarios); i.e. one should never anticipate a single or unique future, but rather a spectrum of different possible futures.
- Exploration of multiple futures is fundamental to the Futures Studies paradigm. Trend Impact Analysis (TIA) is an efficient Futures Studies method designed specifically for this purpose.
- Neural Networks, as a Machine Learning technique, are powerful forecasting tools that seek pattern-based understanding of the past through learning and training.
- Integrating TIA and Neural Networks yields a powerful forecasting tool that adds a learning component and allows that model to adjust to novel situations.
- In a future research, a neural network forecaster can be used (as an external forecasting engine) in an RT-Delphi study. Along the span of the study, experts can be guided by viewing the neural network forecasting output online before determining their responses every time they visit the web-site to update their opinions.
- Moreover, one might think of implanting a neural network within system dynamics models to identify the equations of rates or auxiliary variables when it is not possible or considerably difficult to construct explicit equations as discussed by Alborzi [20]. This is because neural networks are capable of capturing the underlying structural relations between input and output variables through training schemes using actual data.
- Furthermore, one might as well think of building an Agent-Based Model (ABM) based on a neural network approach.

Acknowledgments

We would like to acknowledge the useful discussions and insightful remarks of Professors Mohamed Hassan Rasmy and Mohammed El-Beltagy of Cairo University and Professor Ali Hadi, the Vice Provost and Director of Graduate Studies and Research at the American University in Cairo (AUC). We as well, would like to thank Nesreen Ahmed from Cairo University for her help during the development process of this work. This work is part of “the Data Mining for Improving Tourism Industry Revenue in Egypt” research project within the Egyptian Data Mining and Computer Modeling Center of Excellence.

References


Nedaa Agami is currently an assistant lecturer in the Department of Operations Research and Decision Support, Faculty of Computers and Information at Cairo University, and has just received her M.Sc. degree with the Data Mining and Computer Modeling Center of Excellence in conjunction with the Ministry of Communications and Information Technology (MCIT) in Egypt. Her current research interests include futures studies, scenario planning, systems thinking and simulation and modeling.

Amir Atiya received his B.S. degree in 1982 from Cairo University and the M.S. and Ph.D. degrees in 1986 and 1991 from Caltech, Pasadena, CA, all in electrical engineering. Dr. Atiya is currently a Professor at the Department of Computer Engineering, Cairo University. He recently held several visiting appointments, such as in Caltech and in Chonbuk National University, S. Korea. His research interests are in the areas of neural networks, machine learning, pattern recognition, computational finance, and Monte Carlo methods. He obtained several awards, such as the Kuwait Prize in 2005. Currently, he is an associate editor for the IEEE Transactions on Neural Networks.

Mohamed Saleh is currently an Associate Professor at the Department of Decision Support, Faculty of Computers and Information at Cairo University. He has got a Ph.D. in System Dynamics from the University of Bergen, Norway. He also got his M.Sc. from Bergen University, and a Master of Business Administration (MBA) from Maastricht School of Management, Netherlands. Mohamed is also an Adjunct Professor in the System Dynamics Group at the University of Bergen, Norway. His current research interests are mainly system dynamics, simulation, futures studies and revenue management optimization.

Dr. Hisham El-Shishiny is currently Senior Scientist and Manager of Advanced Technology and Center for Advanced Studies at IBM Technology Development Center in Cairo. He received his Ph.D. and M.Sc. in Computer Science with highest honors from Nancy University in France and his B.Sc. in Electronics and Communications Engineering from Cairo University. He has also completed a two year post-graduate program in Operations Research and conducted post-graduate studies in Solid State Sciences at the American University in Cairo. His current research interests are Data Mining, Machine Learning, Image Processing and Visualization, HPC, Optimization, Knowledge Management and Natural Language Processing.