Sequential Decision Making Predictions under the Influence of Observational Learning

Shing H. DOONG
Department of Information Management, ShuTe University
YanChao District, KaoHsiung City, Taiwan 82445

ABSTRACT

Today’s corporate managers face challenges in information technology (IT) adoption with great stakes. The waiting period for an IT investment to be realized, if any, could be long; thus word-of-mouth information propagation may not help them to make wise decisions. Though information used by early adopters to make their decisions may not be available to the public, late adopters can often observe the decisions made by the early adopters and infer hidden information to supplement their own private information. Observational learning theory applies when a person uses observed behavior from others to infer something about the usefulness of the observed behavior. Walden and Browne proposed a simulation procedure to model the influence of observational learning in sequential decision makings. Previously, we proposed a dynamic Bayesian network (DBN) to model sequential decision makings under the influence of observational learning. In the present study, we show how to infer a DBN model from simulated data. Hidden Markov model and artificial neural networks were used to infer the DBN model. Their performance will be discussed.

Keywords: Sequential Decision Making, Observational Learning, Dynamic Bayesian Network, Hidden Markov Model, Artificial Neural Networks

1. INTRODUCTION

Today’s corporate managers face challenges in information technology (IT) adoption with great stakes. IT together with telecommunication has been considered the main driver for the economic growth of many countries in the new economy era since 2000s. To many companies, IT has become an indispensable part of their core competence with several characteristics. First, IT is becoming so powerful and complex that a fair assessment of its merits is difficult. Second, capital investments in IT are substantial, yet returns on investments often take time to materialize. Brooks has shown that software and other technological components are complex artifacts ever built by human beings [1]. In some cases, impacts of new technologies may take years to be realized [2]. Owing to these reasons, corporate managers need different kinds of tools and practices to help them make wise decisions in IT adoptions.

When people make decisions with limited or asymmetric information, they use different practices to correct this information deficiency. Observational learning occurs “when one person observes the behavior of another person and infers something about the usefulness of the behavior based on that observation” [3]. Research shows that, due to information asymmetry, people use what they observe from others to update their own private information or belief about a decision making [4]. Observational learning often leads to an interesting phenomenon called informational cascades [5].

An informational cascade occurs “if an individual’s action does not depend on his private information signal” [5]. Walden and Browne [3] developed a theoretical extension of the observational learning model in [5], where a binary private signal is generated for each decision maker who chooses to adopt or reject an action. In [3], a continuous private signal is issued to each individual who also chooses to adopt or reject an action. Changing the private information signal from binary to continuous has produced many interesting results. For example, unlike the easy informational cascading in the case of binary signals, Walden and Browne showed that there are always late decision reversals in a sequence of decision makings. That is, informational cascades do not occur in the case of continuous signals.

A simulation procedure was used to investigate the extended observational learning theory [3]. In a later study, we showed that the Walden and Browne (WB) model can also be investigated from the perspective of a dynamic Bayesian network (DBN) [6]. In the present study, we consider the problem of inferring the DBN from simulated data. Hidden Markov model (HMM) and artificial neural networks (ANN) are used to infer the DBN model.

This paper is organized as follows. We briefly discuss the WB model and our DBN perspective first. Then, HMM and ANN are introduced to learn the DBN, given simulated data. Experimental results will be presented next, followed by discussions and conclusions.

2. MATERIALS AND METHODS

Observational learning with continuous signals

Walden and Browne used a continuous signal to denote the private information received by an individual [3]. A sequence of individuals will make a decision of choosing technology A (e.g., adopt the cloud computing IT) or technology B (e.g., reject the cloud computing IT). Assume that technologies A and B emit signals from the normal distributions $N(\mu_A, \sigma^2_A)$ and $N(\mu_B, \sigma^2_B)$ respectively, and $\mu_A > \mu_B$. An individual chooses A if the following condition is satisfied:

$$\frac{p(s \mid \mu_A)}{p(s \mid \mu_B)} > \beta$$  \hspace{1cm} (1)

Here $s$ is a private signal received by the individual, $p(s \mid \mu_A)$ and $p(s \mid \mu_B)$ are the probability distribution
functions (pdfs) of \( N(\mu_A, \sigma^2) \) and \( N(\mu_B, \sigma^2) \), respectively. Plugging in pdfs to solve for \( s \), we obtain the following decision rule:

\[
\begin{aligned}
  s \geq r(\beta) & \Rightarrow A \\
  s < r(\beta) & \Rightarrow B 
\end{aligned}
\]  

(2)

\[
r(\beta) = \frac{2\ln(\beta)\sigma^2 - \mu_A^2 + \mu_B^2}{2(\mu_A - \mu_B)}
\]  

(3)

Using signal detection theory [7], Walden and Brown set the decision threshold \( \beta \) as follows:

\[
\beta = \frac{\Pr(M_B)}{\Pr(M_A)}k
\]

(4)

Here \( k \) is common to all individuals and involves the relative benefit of B to A. For the first individual, we can assume technologies A and B are equally good, thus \( \Pr(M_B) = \Pr(M_A) \).

In order to describe the fact that \( \beta_i \) scales up or down depending on the decision of the \( t \)-th individual, a causal link from \( Y_t \) to \( X_{t+1} \) must be established as follows:

![Diagram](image_url)

Figure 1. A DBN for observational learning

The \( X_{t+1} \) variable depends on \( X_t \) and \( Y_t \) by rules in Table 1. This random variable has a discrete distribution and its value depends on the previous hidden state (\( X_t \)) and the previous decision (\( Y_t \)). The WB model is now converted to a DBN in Figure 1 with relevant pdfs given in Eq. (7) and Table 1.

| \( \Pr(X_{t+1} | X_t, Y_t) \) | (\( \beta_t, a \)) | (\( \beta_t, b \)) |
|----------------|----------------|----------------|
| \( \Pr(a | M_B, A_t) \) | 1 | 0 |
| \( \Pr(a | M_A, A_t) \) | \( \Pr(b | M_B, A_t) \) | 0 | 1 |

Table 1. Conditional probability \( \Pr(X_{t+1} | X_t, Y_t) \)

The DBN perspective has a few advantages over the original WB model. First, the signal receiving and decision making step has been simplified to a binomial sampling step. Second, the dynamic updating of decision thresholds is replaced by clear rules in Table 1. The DBN in Figure 1 is also easy to interpret. It tells causal relationships among all relevant variables and shows how the system evolves as time moves ahead.

**Dynamic Bayesian Network**

In order to model the WB model as a DBN, we need to use two sequences of random variables to describe the dynamic involved in a sequential decision model [6]. The variable \( X_t \) represents a decision threshold \( \beta_t \), thus \( \Pr(X_t = \beta_t ) = 1 \), and the \( Y_t \) variable represents the outcome of a decision. Thus, assuming that signals are drawn from technology A, then \( \Pr(Y_t | X_t) \) is given by:

\[
\begin{aligned}
\Pr(a | \beta_t) &= \int_{-\infty}^{r(\beta)} p(s | \mu_A) ds, \\
\Pr(b | \beta_t) &= \int_{r(\beta)}^{\infty} p(s | \mu_A) ds 
\end{aligned}
\]  

(7)

**Hidden Markov Models**

A hidden Markov model (HMM) is represented by a 5-tuple \((S, V, A, B, \pi)\) where \( S = \{s_1, s_2, \ldots, s_N\} \) consists of \( N \) states that are not directly observable, \( V = \{v_1, v_2, \ldots, v_M\} \) denotes \( M \) observable outcomes emitted by a state, \( A = (a_{ij}) = \Pr(s_j | s_i) \) represents the transition probabilities between states, \( B = (b_{jk}) = \Pr(v_k | s_j) \) represents the emitting probabilities of outcomes by states, and \( \pi = (\pi_s) \) represents the initial probabilities of states [8].

Given an HMM with all needed components, a sequence of outcomes can be generated by (1) choosing an initial state according to the initial probability vector \( \pi \); (2) emitting an outcome from this state by using the emitting probability matrix
On the other hand, given sequences of outcomes, an HMM may be learned from the data and used for future predictions. This is a patterns induction procedure used in most data mining algorithms.

**Artificial Neural Networks**

Artificial neural network (ANN) has been successfully applied to solve many function approximation problems in engineering and social sciences. An ANN simulates the neural system of a brain to learn patterns from examples and uses the learned knowledge to make predictions for new data [9]. A basic data processing unit in a neural net is called a neuron which is connected to other neurons via synapses. The structure of an ANN refers to the number of neurons and the way they are distributed and connected.

To simplify the computation, neurons are scattered into layers and information is transferred from layer to layer. The input layer represents the independent variables in a function approximation problem. The output layer corresponds to the dependent variable(s). Layers between the input and output layers are called hidden layers. An ANN with hidden layers is also called a multilayer perceptron (MLP). Without a hidden layer, a simple perceptron has limited learning capability [10]. It has been shown that an MLP can approximate arbitrarily well any continuous decision region provided that there are enough layers and neurons [11]. Learning an ANN from data is to find optimal synaptic weights to fit training data with known input-output pairs. Attention must be paid to the network structure so that we do not overfit the model with data. A trained ANN can be used to predict output values for new input values.

**3. A SIMULATION STUDY**

To examine informational cascades of sequential decision makings under the influence of observational learning, both [3] and [6] presented a simulation study.

Assume that two alternative technologies A and B are to be selected by a sequence of individuals. Suppose A is the better technology, thus all private signals will be emitted by its pdf, which is assumed to be a normal distribution $N(\mu_A, \sigma^2)$. We assume that $\mu_A = 1$, $\mu_B = 0$, and $k = 1, \sigma = 1$ in the previous section of observational learning with continuous signals.

A simulation run of decision makings consists of 100 sequential decisions as explained previously. For the WB approach, this includes (1) drawing a signal from the pdf of technology A; (2) making a decision based on the signal, Eq. (2) and Eq. (3); (3) updating the new threshold according to the decision made and Eq. (6); and (4) continuing the process until the 100th decision is made. On the other hand, for the DBN approach, this includes (1) drawing a sample from the uniform distribution on $[0, 1]$ to decide technology A or B according to Eq. (7); (2) updating conditional probability $P(X_{t+1}|X_t, Y_t)$ in Table 1; and (3) continuing the process until the 100th decision is made.

Because the simulation is based on probabilistic samplings, one run of simulation can differ from another run of simulation substantially. Thus, a total of 1000 runs of simulation are conducted to smooth out fluctuations between runs. At the end, the average correct decision rate for each decision position (from 1 to 100) is reported. The average correct decision rate at a position is the number of correct decisions (i.e., choosing A) at that position out of total runs divided by 1000.

Figure 2 shows that both approaches yield very similar curves of average correct decision rate. Both approaches have an average correct rate curve that starts low at around .70 and increases to around .95 at the later stage. The correlation value between these two sequences of average correct decision rates is .994 and the mean absolute error (MAE) is .013. Other simulation types including random updating of decision thresholds and cases of tertiary decisions can be considered with the DBN approach.

**4. LEARNING PATTERNS OF SEQUENTIAL DECISIONS**

The last section presents a simulation study based on WB and DBN models. We now consider the reverse process of discovering models from data. Since real sequential decisions are hard, if not impossible, to obtain in business, we use simulated data from the DBN approach to learn patterns of sequential decisions under the influence of observational learning.

**Training samples**

The DBN approach is used to generate training samples for learning patterns of sequential decisions. In total, 1000 observation sequences are outputted from the simulations. Each observation sequence consists of 100 sequential decisions of 1 (for choosing A) or 0 (for choosing B).

**Using HMM as a learning tool**

HMM is a special DBN when it is spread out in steps. In order to use HMM, we need to decide the number of hidden states and the number of observable outcomes. Since there are only two possible decisions (1 or 0), we choose 2 hidden states and 2 emitted outcomes. The 1000 observation sequences of training samples are fed into a Baum-Welch (also called a forward-backward) learning algorithm to learn parameters of an HMM [8]. These parameters include the initial probability for each state, outcomes emitting probabilities and states transition probabilities.

The trained HMM is used to generate 1000 sequences of simulated outcomes. The simulation is obvious and straightforward, given the full parameters of an HMM. Each sequence consists of 100 sequential decisions. The average correct decision rate is computed as before and compared with that from the DBN approach.
Using ANN as a learning tool
In order to use ANN as a learning tool, we need to set up an input-output correspondence, i.e., input variables and output variables. Using the DBN perspective (Figure 1) as a guideline, we can set up a correspondence as

\[ \beta_{t+1} = f(\beta_t, D_t) \]

Since \( \beta_t \) determines the distribution of \( D_t \) according to Eq. (7), we will use the probability of choosing A as the surrogate variable. Let \( p_t \) denote the probability of choosing A at the \( t \)-th position. \( p_t \) can be determined from Eq. (7). Then, we will approximate the following function with ANN.

\[ p_{t+1} = f(p_t, D_t) \] (8)

To prepare training samples for the ANN learning, we use the average correct decision rate from the DBN simulations to denote \( p_t \), i.e. \( p_t = \frac{\text{number of decision A at position } t}{1000} \).

The \( D_t \) variable is extracted from the 1000 observation sequences of the DBN simulations. Instead of using the full set of 1000 observation sequences to train a single ANN model, we train 10 ANN models with smaller data sets and average outputs from these 10 ANN models to make predictions. More specifically, we randomly choose 100 observation sequences from the DBN simulation to train an ANN model. This procedure is repeated 10 times to get 10 ANN models, which are bagged to get the final predictor. The idea is similar to a bagging predictor [12].

Since our model in Eq. (8) has only two inputs and one output, we do not need to use a complicated network structure. One or two hidden layers will suffice for our data set. Though our data set may be potentially large, e.g., 100 observation sequences with 100 sequential decisions will produce 9900 input-output pairs, of which many are simply duplicates. After using a trial-and-error approach with test data, we decided to use a two hidden layer structure – the first hidden layer has 4 neurons and the second hidden layer has 2 neurons. Our final MLP has 2, 4, 2, 1 neurons in the respective layer. The Sigmoid function was chosen to be the activation function.

After the bagging aggregator is trained, it is used to predict the probability \( p_t \) in a simulation of sequential decisions. The first decision is simulated by using the average correct rate at position 1 from the DBN simulation. A random sample is drawn from the uniform distribution on \((0, 1)\) and compared with this average correct rate to choose technology A or B. After the decision is made, it is plugged into Eq. (8) with the learned bagging ANN predictor to predict the next probability of choosing A. This process continues until the 100\(^{th}\) decision is made. Again, 1000 runs of simulation are conducted to calculate the average correct decision rate from the learned ANN model.

5. EXPERIMENTAL RESULTS

In this section, we present the experimental results from different simulation scenarios.

The standard case

In the standard case, we assume \( k=1 \). Thus, the relative benefit of choosing A or B is equal to one. The previous simulation study has shown that the average correct decision rate increases from around .70 at position 1 to around .95 at position 100. Figure 3 shows the average correct decision rate curves from DBN, HMM and ANN. The DBN simulation was used to generate training samples for the other two to learn. Both HMM and ANN learn their model from the training samples, and use the learned model to simulate sequential decisions. The average correct decision rate curve reports the simulation results using the trained model.

![Figure 3. Comparison with DBN, HMM and ANN (k=1)](image1)

The MAE between DBN and HMM is .013 and the same measure for DBN and ANN is .008. On the other hand, the correlation between DBN sequence and HMM sequence is .969, and the same measure for DBN and ANN is .982. Thus, ANN has learned a better prediction model for this standard case.

Technology B has a higher relative benefit

In this case, we assume \( k=10 \), thus technology B has a higher relative benefit than technology A. This gives individuals less incentives to choose technology A.

![Figure 4. Comparison with DBN, HMM and ANN (k=10)](image2)
order to make a decision of choosing A. The probability of choosing technology A is small at the beginning. When more individuals select technology A, later individuals increase their belief in technology A through observational learning.

The simulations show that the average correct decision rate increases from less than .10 at position 1 to around .60 at position 100. The MAE between DBN and HMM and ANN is .028 and .031 respectively. The correlation between DBN sequence and HMM sequence is .997, while the same measure for DBN and ANN is .992. Therefore, HMM is a better prediction model in this case.

**Technology B has a higher relative benefit and only partial sequences are used**

In this case, we assume that technology B has a higher relative benefit \((k = 5)\), and only partial sequences from the DBN simulations are used to train HMM and ANN models. We assume that only the first 50 decisions in an observation sequence are used to train prediction models.

![Figure 5. Comparison of models (k=5, only 50 decisions used)](image)

Since the relative benefit of B to A is not as big as the one in the previous case, we expect individuals to have more incentives to choose technology A. Figure 5 verifies this with an initial average correct decision rate of around .15 to the last rate of around .80 at position 100. Since we only use the first 50 decisions to train HMM and ANN, their performance for the second half of decision sequences is more interesting. Figure 5 shows that the HMM model performs better than the ANN model for this part of decision sequences. Overall, the HMM model also produces a smaller MAE ( .026 vs. .030) and a higher correlation (.993 vs. .970) than the ANN model.

**6. DISCUSSIONS**

The experimental results show that HMM has a better capability than ANN in learning patterns of sequential decisions. When \(k\) is big, the average correct decision rate curve resulting from the ANN model is much jagged than that from the HMM model. This is interesting if we consider the fact that the HMM model has no causal links between hidden states only. Using transition probabilities, the next state is sampled based on the current state only. Outcomes from the current state have no effect on the sampling of the next state in HMM. This seems to contradict the causal model explained by the DBN perspective of observational learning.

The jaggedness of the ANN average correct decision rate curve may come from an over-fitted neural network. Because we have a small network structure with a bundle of data, though many of them are duplicates, we may over-fit the network to produce a sensitive predictor. The bagging procedure does not seem to overcome this difficulty. Other prediction algorithms such as support vector regressions with known capabilities in over-fitting control may be considered in the future.

**7. CONCLUSIONS**

Today’s corporate managers face challenges in IT adoption with great stakes. Corporate IT has become so powerful and complex that a fair assessment of its merits is difficult. Capital investments in IT are substantial, yet returns on investments often take time to materialize. Conventional word-of-mouth information propagation procedures may work for consumer IT decisions, but not for corporate IT decisions.

Though it is usually difficult to obtain the private information that other companies use to make their IT adoption decisions, it is possible to observe what the other companies have decided in their IT adoption. Observational learning theory applies when a person uses observed behavior from others to infer something about the usefulness of the observed behavior. Corporate managers may practice observational learning to help them make better IT adoption decisions.

Observational learning is known to create informational cascades, a phenomenon when an individual’s action does not depend on his private information signal. When informational cascades occur, belief inferred from observational learning has overshadowed the private information signal that an individual uses to make his decision.

Walden and Browne [3] proposed a simulation model to show that informational cascades do not occur when the private information signal is continuous. We presented a DBN perspective of the WB model in [6]. The DBN approach demonstrated similar simulation results as the WB approach.

This study is focused on learning the DBN model resulting from observational learning impacted sequential decisions. Two machine learning tools are used to learn the DBN. The first one, hidden Markov model, is itself a special case of DBN. The second one, artificial neural network, is a popular learning algorithm in artificial intelligence.

The HMM learning approach does not consider impacts of the current decision \((D_i)\) on the sampling of the next state. It also uses a limited number of hidden states to represent continuous information signals. On the other hand, the ANN learning approach uses the DBN perspective to model a functional form for approximation. Its continuous output variable meets the type of private information signals in Walden and Browne [3].

The experimental results show that HMM has a better learning capability than ANN in our study. In the future, we plan to run...
more tests with different learning algorithms and diverse training samples. Learning patterns of sequential decisions constitutes the reverse process of simulation studies as presented in [3, 6]. Together, simulation studies and patterns learning can help us better understand how observational learning impacts sequential decisions.

Acknowledgements: This research has been supported in part by a grant from the National Science Council of Taiwan under the contract number NSC99-2410-H-366-006-MY2.

8. REFERENCES