A dynamic procedure for defection detection and prevention based on SOM and a Markov chain

Young-ae Kim[a], Hee-seok Song[b], and Soung-hie Kim[c]

[a]*Graduate School of Management, KAIST,*
207-43 Cheongyangri2-dong, Dongdaemun-gu, Seoul 130-012 , South Korea
Tel: +82-2-958-3684, Fax: +82-2-958-3604, E-mail Address: kya1030@kgsm.kaist.ac.kr

[b]*Department of Management Information Systems, Hannam University,*
133 Ojung-dong, Taeduk-gu, Taejon, 306-791 , South Korea
Tel: +82-42-629-7614, Fax: +82-42-629-7667, E-mail Address: hssong@kgsm.kaist.ac.kr

[c]*Graduate School of Management, KAIST,*
207-43 Cheongyangri2-dong, Dongdaemun-gu, Seoul 130-012 , South Korea
Tel: +82-2-958-3684, Fax: +82-2-958-3604, E-mail Address: seekim@kgsm.kaist.ac.kr

Abstract

Customer retention is a common concern for many industries and a critical issue for the survival in today's greatly compressed marketplace. Current customer retention models only focus on detection of potential defectors based on the likelihood of defection by using demographic and customer profile information. In this paper, we propose a dynamic procedure for defection detection and prevention using past and current customer behavior by utilizing SOM and Markov chain. The basic idea originates from the observation that a customer has a tendency to change his behavior (i.e. trim-out his usage volumes) before his eventual withdrawal. This gradual pulling out process offers the company the opportunity to detect the defection signals. With this approach, we have two significant benefits compared with existing defection detection studies. First, our procedure can predict when the potential defectors could withdraw and this feature helps to give marketing managers ample lead-time for preparing defection prevention plans. The second benefit is that our approach can provide a procedure for not only defection detection but also defection prevention, which could suggest the desirable behavior state for the next period so as to lower the likelihood of defection. We applied our dynamic procedure for defection detection and prevention to the online gaming industry. Our suggested procedure could predict potential defectors without deterioration of prediction accuracy compared to that of the MLP neural network and DT.

Keywords:

Customer Relationship Management, Data Mining, Customer Defection, Defection Prevention, Self-Organizing Map, a Markov chain

1. Introduction

Customer retention is a common concern for many industries and a critical issue for the success or bottom-line survival in today's greatly compressed marketplace (Wei, & Chiu, 2002). The longer a customer stays with a company the more profit a customer generates. This is the outcome of a number of factors such as the higher initial costs of attracting new customers, the increase in the number of purchases, the customer's better understanding of the company, and positive word-of-mouth (Trubik, & Smith, 2000).

Many studies (Datta et al., 2000; Mozer et al., 2000; Ng, & Liu, 2000; Raghavan et al., 2000; Smith et al., 2000; Wei, & Chiu, 2002) have been conducted for customer retention to demonstrate the potential of data mining through experiments and case studies. Especially in the service industry such as telecommunications and Internet service companies, there exist enormous customer behavior data for the purpose of billing. These behavior data provide a good opportunity to predict future customer behavior (Datta et al., 2000; Mozer et al., 2000; Ng, & Liu, 2000; Raghavan et al., 2000; Song et al., 2001) Here, the term 'customer behavior' means action-oriented activities like calling, visiting websites, and making purchases.

This paper aims to develop a procedure for defection detection and prevention using past and current customer behavior by integrating SOM and a Markov chain. The basic idea originates from the observation that a customer has a tendency to change his behavior (i.e. trim-out his usage volumes) before his eventual withdrawal. This gradual pulling out process offers the company the opportunity to detect the signals of defection.

With this approach, we have significant benefits compared with existing defection detection studies based on the data mining techniques. First of all, we can compose a *dynamic* defection detection procedure. Because existing defection detection models only focus on who the potential defectors are, they cannot predict when the potential defectors would withdraw. But our defection detection procedure can predict the lead-time for eventual defection by the dynamic feature. This feature helps to give a marketing manager ample lead-time for preparing defection prevention plans. The second benefit is that our approach can provide a procedure for not only defection detection but also defection prevention which recommends the desirable behavior state for the next period so as to lower the likelihood of defection.

However, building a dynamic procedure for defection detection is not easy because multi-features of customer behavior are related to customer defection in a highly nonlinear way. To overcome this difficulty, we define the possible behavior states for a specific domain using SOM, and then use this state representation in monitoring shifts of customer behavior. Based on these data on the flow of customers among all these states, we adopted a Markov chain to predict potential defectors and lead-time for eventual defection. Then, we set up an automated defection prevention procedure which assists building campaigns. We applied our defection detection and prevention procedure to the online gaming industry, because the histories of its customer behavior are the most prominent predicator for future defection.

We begin by reviewing the previous studies on defection detection in section 2. The concept of SOM and a Markov chain which are prerequisites for our research is summarized in section 3. The domain for the case study is briefly introduced in section 4. In section 5, our suggested procedure is explained step by step. Some experiments are provided to evaluate our procedure in sections 6. In section 7, finally we summarize our contributions and outline areas for further research issues.

2. Existing studies in defection detection using data mining techniques

Many studies (Datta et al., 2000; Mozer et al., 2000; Ng, & Liu, 2000; Raghavan et al., 2000; Smith et al., 2000) have been conducted to detect potential defectors using data mining techniques. Most of their prediction models for defection are developed with a logit regression, a decision tree and a neural network using demographic and customer profile information.

In particular in the service industry such as telecommunications and Internet services providing companies, customer behavior data are mainly used to gauge and predict the likelihood of defection (Datta et al., 2000; Mozer et al., 2000; Ng, & Liu, 2000; Raghavan et al., 2000). Raghavan et al. (2000) developed a prediction model for customer defection in which they used each user's online activity patterns as independent features and assessed the goodness-of-fit of their model in an Internet service provider (ISP). Ng and Liu (2000) proposed a solution that integrates various techniques of data mining to form an intuitive and novel approach to gauging customer loyalty and predicting their likelihood of defection. Their studies conclude that behavioral features can capture intangible value more effectively than gross demographic variables, which are traditionally used to predict defection.

However, these studies only focused on providing the likelihood of defection. The marketing manager needs not only the likelihood of defection but also the lead-time to conduct a defection prevention campaign effectively before eventual defection. But, we have not yet found such a study in the field of customer retention, capable of handling a dynamic procedure for defection detection and prevention.

3. Background

3.1 SOM

SOM is able to map structured and high-dimensional data onto a much lower-dimensional array of neurons in an orderly fashion. This mapping tends to preserve the topological relationships of the input data. Topological preserving means that the data points lying near each other in the input space will be mapped onto nearby map units. Due to this property, SOM is able to cluster input data and their relationships on the map. SOM facilitates the understanding of processes so that several variables and their interactions may be inspected simultaneously (Alhoniemi et al., 1999; Simula et al., 1999). We adopt SOM to determine every possible behavior state for a specific domain.

3.2 Markov Chain

A Markov chain, which is one special type of discrete-time stochastic process, can be used to predict the evolution of customer states (Ha et al., 2002).

A Markov chain has the following properties and assumptions:

① The probability distribution of the state i at time $t+1$ only depends on the state at time t.

② For all states i and j, $p(X_{t+1} = j | X_t = i)$ is independent of time t. These two assumptions can be specified by the conditional probabilities for a Markov chain:

$$P\left(X_{t+1} = j \mid X_t = i, X_{t-1} = i_{t-1}, \ldots, X_1 = i_1, X_0 = i_0\right)$$
$$= P\left(X_{t+1} = j \mid X_t = i\right) = p_{ij}$$
$$\text{for all states } i_0, i_1, \ldots, i_{t-1}, i, j \qquad (1)$$

We adopted an absorbing Markov chain in which some of the states are absorbing states and the rest are transient ones.

An absorbing state has a special transition probability, $p_{ii} = 1$ (Winston, 1994). For example, once a customer is in a defector state at time t (i.e. current period), he/she cannot be in non-defector states at time $t+1$(i.e. next period) or afterwards. Our research question to detect potential defectors can be summarized as follows based on this absorbing Markov chain: How much time does the customer stay in given transient states before absorption (i.e. defection) take place?

4. Domain

The dataset for the case study is prepared from an online game company in Korea. We sampled 255 customers and collected totally 114,736 transactions for those customers from various sources such as web log data and the transaction database. The collected input data for defection prediction contain the total session length for a week, the total number of sessions for a week, the total number of access errors caused by congestion, and the average length per session. Customer profiles were maintained for the four input features during four consecutive weeks for each customer with an actual defection indicator. We defined a defector as the customer who has no session time for one month, because there is no voluntary sign of withdrawal in this Internet game site.

5.Proposed procedure

5.1 Overall procedure

The key idea of our proposed procedure for defection detection is adopted from the work on monitoring the evolution of customer behavior states over time. To monitor the customer behavior states, a representation scheme describing the state of an individual customer has to be developed based on individual customer behavior data. The SOM model is adopted to determine the states of customer behavior because it facilitates understanding of complex customer behavior, so that several variables and their interactions can be inspected simultaneously even when non-linear dependencies between customer behavior variables exist (Alhoniemi et al., 1999; Simula et al., 1999). Based on this state representation, a Markov chain is applied to provide probabilities of state shift and predict the elapsed time to eventual defection. We present the overall procedure of our dynamic defection detection and prevention as shown in Figure 1.

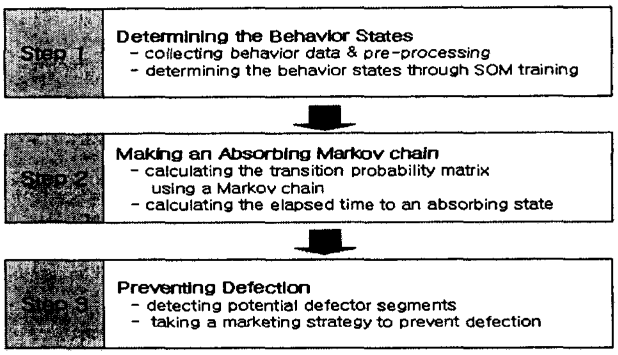

Figure 1 - Overall procedure for dynamic defection detection and prevention

5.2 Step 1: Determining the behavior states

5.2.1 Collecting behavior data and pre-processing

First of all, features which can affect defection are selected based on the prior knowledge of domain experts, and the values of features are collected over the multi-periods from web log files and databases. After collecting behavior data, data preparation such as session identification, user identification, normalization and feature selection is needed to build behavioral profiles (Almuallim, & Dietterich, 1994; Chakraborty G. et al., 2000; Cooley et al., 1999; Hall, & Smith, 1998; Kira & Rendell, 1992). In our online game domain, the total session time during a week, the total number of successful sessions during a week, the total number of access errors caused by congestion during a session time, and the average session time are selected by the marketing manager. Table 1 provides behavior data on five customers. In this table, customer 'AVAL' spent 1343 minutes playing the game during a week with 23 times of successful access, and he/she met 9 access errors four weeks earlier. After that, he/she trimmed out his/her usage volume and used only 19 minutes for one week just before his/her final defection.

Table 1 - Sample behavior profile

Customer ID	Week	Session Time	# of normal accesses	# of error accesses	Average session time	Actual Defection
AVAL	t - 1	19	3	0	6.33	Defected
	t - 2	39	3	0	13.00	Non-defected
	t - 3	147	5	1	29.40	Non-defected
	t - 4	1343	23	9	58.39	Non-defected
DANNYKIM	t - 1	433	18	13	24.06	Defected
	t - 2	2496	15	24	166.40	Non-defected
	t - 3	8217	29	32	283.34	Non-defected
	t - 4	3404	25	24	136.16	Non-defected
MIO935	t - 1	6457	38	10	169.92	Non-defected
	t - 2	5948	41	50	145.07	Non-defected
	t - 3	2462	25	21	98.48	Non-defected
	t - 4	5979	54	36	110.72	Non-defected

5.2.2 Determining the behavior states through SOM training

Based on the selected input features, all the possible states of customer behavior are determined. In our domain, four behavior features in a past unit period for a certain user consist of an input vector for SOM like the record in Table 1. Figure 2 illustrates the possible behavior states which are the results of SOM learning. To interpret each state on the map, a decision tree analysis (C5.0) is additionally conducted. All the four behavior features in Table 1 are regarded as independent variables, and the behavior states that have been assigned as a result of SOM learning are considered as a target class in the decision tree analysis. For example, state C30 can be interpreted as a heavy usage state because its session length is between 0.695 and 0.814. On the contrary, state C04 can be named as a light usage state.

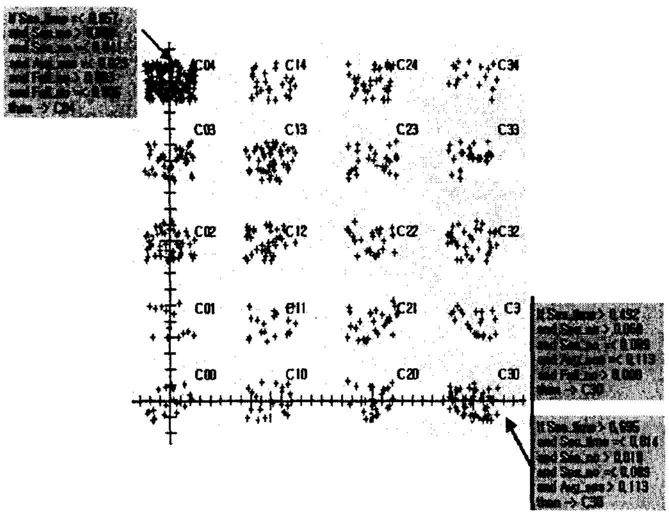

Figure 2 - Behavior states and their interpretations

5.3 Step 2: Making an absorbing Markov chain

To make an absorbing Markov chain, we have two assumptions. First, we assume that customer behavior states have the Markovian property. It implies that given the present state of the customer, the future probabilistic behavior of the customer is independent of its past history and only depends on the current state. This is because the customer's current state information most impacts the decision of the customer's next behavior. Second, we assume that all the customers can be potential defectors because they eventually leave the company some time or other if they are not continuously satisfied with the company's service in today's greatly compressed online gaming industry.

To make a transition probability matrix, we used the states of all the customers for four periods (i.e. four weeks). We rearranged four pairs of transition data for each customer such as all of the transition data from t-4 week to t-3 week, from t-3 week to t-2 week, from t-2 week to t-1 week, and from t-1 week to t week (i.e. an absorbing state). See Figure 3. And then we calculated a transition probability from one week to another week for each state.

Customer ID	t week	t+1 week
순순이[선영]	C04	C13
	C13	C04
	C04	C04
	C04	def
DANNYKIM	C21	C3C
	C3C	C32
	C32	C03
	C03	def
기소중지	C13	C04
	C04	C04
	C04	C04
	C04	def

Figure 3 – The sample transition data for each customer

In our online game domain, a transition probability matrix is calculated as shown in Figure 4. In this figure, state 'def' means 'defector' which is an absorbing state. According to the property of an absorbing Markov chain, if the customer's current state begins in a non-absorbing state (i.e. from state C00 to state C34), then he/she will leave the non-absorbing state and end up in one of the absorbing states (i.e. state 'def') eventually (Winston, 1994).

State at time $t+1$

State at time t

	C00	C01	C02	C03	C04	...	def
C00	0	0	0.29	0.14	0.14	...	0
C01	0	0.08	0.23	0.08	0.15	...	0.08
C02	0	0.05	0.29	0.05	0.19	...	0.14
C03	0	0	0	0.15	0.45	...	0.20
C04	0	0.02	0.02	0.04	0.53	...	0.26
⋮	⋮	⋮	⋮	⋮	⋮	...	⋮
C31	0.08	0	0	0	0	...	0.07
C32	0	0	0	0.05	0.05	...	0.04
C33	0	0	0	0	0.10	...	0
C34	0	0	0	0	0.17	...	0.09
def	0	0	0	0	0	...	1.00

Figure 4 - The transition probability matrix with an absorbing state, 'def'

Figure 5 describes the general transition matrix for an absorbing Markov chain. It consists of four parts which are I, O, R and Q, as illustrated in Figure 5. We can find out the elapsed time to absorbing state 'def' in each non-absorbing state by using the matrix $(I - Q)^{-1}$.

$$
\begin{array}{r|c|c|}
 & \text{non-absorbing states} & \text{absorbing states} \\
\hline
\text{non-absorbing states} & Q & R \\
\hline
\text{absorbing states} & O & I \\
\end{array}
$$

Figure 5 - The transition matrix for an absorbing Markov chain

Figure 6 describes the outcome of the matrix $(I - Q)^{-1}$ for the online game domain. In this figure, we can see that if a customer belongs to state C00 at time t, then he/she will defect 11.5 periods later on average. The elapsed time to absorbing state 'def' can be used as an indicator for classifying defectors and non-defectors and it gives significant information on lead-time for defection prevention.

State	elapsed time (Week)	State	elapsed time (Week)
C00	11.51	C20	14.17
C01	9.74	C21	12.93
C02	8.46	C22	10.77
C03	7.21	C23	9.47
C04	5.97	C24	10.05
C10	13.31	C30	14.36
C11	11.59	C31	12.23
C12	10.47	C32	11.50
C13	8.21	C33	10.16
C14	8.60	C34	9.40

Figure 6 - The elapsed time to an absorbing state

3.4 Step 3: Preventing Defection

3.4.1 Detecting potential defectors

Potential defectors and non-defectors are classified by the current customer state on SOM using its elapsed time to absorbing state 'def'. The threshold for elapsed time can be determined by calculating the hit ratio. In our domain, if we set the threshold for elapsed time as 8.0 weeks, then we gain the highest hit ratio and state C03 and C04 are the potential defection states (See in Figure 6.).

3.4.2 Preventing defection

The object of the defection prevention step is to provide information of To-Be behavior state for the next period to drive their potential defectors into a desirable position. If we can find a behavior state which will extend the elapsed time to absorbing state 'def' (i.e. lower the likelihood of defection), then we can design campaigns to focus on modifying their behavior patterns so as to lead to belonging to that state.

In this section, two strategies are suggested to respond to the question of 'Where to drive potential defectors for the next period?'. The first strategy is to adopt the improvement approach in inducing potential defectors to change their behavior patterns. This strategy is based on the fact that it is very difficult to change customer behavior greatly in such a short period (i.e. a month in our example). Thus, we should induce the gradual change in their behavior patterns through the regular and repetitive operation of this procedure. The second strategy is the innovative approach, which is for the potential defectors who have the shorter lead-time to eventually withdrawal. The innovative approach requires high costs and risk in campaigns because it tries for a radical change of customer behavior in a short period (i.e. a month in our example). Two strategies can be implemented on the map of SOM using the topology preserving property of SOM. It is relatively easy to drive a customer into a closer state (node) from a current behavior state on the output map of SOM because input behavior patterns between the two states are mostly similar according to the topology preserving property. In the improvement approach, we do not consider driving the customers to the second or further neighborhood node from a current node on SOM at a time. Therefore, we can guide potential defectors to a node that will extend the elapsed time the most until he/she eventually defects, among zero and the first nearest neighborhood nodes. In the same vein, in the innovative approach, we can consider driving the customers to the desirable node among all nodes on the map of SOM.

In our domain, we apply an improvement approach to customers in states whose elapsed time to absorbing state 'def' is between 4.0 weeks and 8.0 weeks. And we decide to apply an innovative approach to customers in states whose elapsed time is shorter than 4.0 weeks according to the domain expert's opinion in this field.

(1) Strategy 1: An improvement approach

The improvement approach can be implemented through the algorithm in Figure 7. In Figure 7, a potential defector node (*ExistingNode*) and its elapsed time are the input and a To-Be node (*ToBeNode*) to which this approach recommends to drive is the final output. At line 1 of the algorithm, the output variable and the internal variable are initialized, and zero and the first nearest neighborhood nodes centered on the current node are selected at line 2. *k-NeighborNode* means the set of *k*th nearest neighbor nodes from the current node. In this notation, *k=0* means the set of neighbor nodes which contains only the current node, and *k=1* means the set of first neighbor nodes which contains eight or less neighbor nodes on the rectangular structure SOM. For each selected neighbor node, the elapsed time is calculated and the elapsed time gain is computed by subtracting the elapsed time of the current node from that of a new node at line 4, 5 and 6. Among all selected neighbor nodes, the nodes that have maximum and positive elapsed time gains are selected from line 7 to 13. Those nodes are recommended as To-Be behavior states to guide the customer for the next period. Thus, the behavior information of a To-Be node becomes the target behavior pattern of the next period for customers in the current state (i.e. node).

```
ToBeNode = φ, MaximumGain = -1
0-NeighborNode = {n_1}, 1-NeighborNode = {n_2, n_3, ..., n_m}
for each neighbor node n_i do begin
    NewNode = n_i
    getting ElapsedTime of NewNode
    ElapsedTimeGain = ElapsedTime of NewNode
                      - ElapsedTime of ExistingNode
    if ElapsedTimeGain > 0
      if ElapsedTimeGain > MaximumGain
        ToBeNode = φ, add n_i to ToBeNode,
        MaximumGain = ElapsedTimeGain
      elseif ElapsedTimeGain = MaximumGain
        add n_i to ToBeNode,
        MaximumGain = ElapsedTimeGain
      endif
    endif
endfor
recommend to drive to a node in ToBeNode for next period
```

Figure 7 - Defection prevention procedure for the improvement approach

To demonstrate how well the improvement approach performs in our domain, we detect two potential defector states and compute elapsed time gains. Figure 8 illustrates elapsed time gains when potential defectors in that states are guided to zero or the first nearest neighbor node which has the largest elapsed time gains for the next week. The total elapsed time gains of state 'C03' are 29.34 weeks by multiplying 3.26 by 9 and that of state 'C04' are 147.28 weeks. Therefore our total elapsed time gains are 176.62 weeks for the next period through our defection prevention procedure. Also, we can have larger elapsed time gains from defection prevention through the repetitive operation of this procedure.

T week	T+1 week (To Be)	New Elapsed time	Current Elapsed time	Elapsed time gains (weeks)	The number of defectors	Total gains (weeks)
C03	C02	8.46	7.21	3.26	9	29.34
	C04	5.97				
	C12	10.47				
	C13	8.21				
	C14	8.6				
C04	C03	7.21	5.97	2.63	56	147.28
	C13	8.21				
	C14	8.6				

Figure 8 - Computing elapsed time gains

(2) Strategy 2: An innovative approach

Figure 9 shows an algorithm for the innovative approach. Most of the algorithm is similar to the previous algorithm but the innovative approach investigates every node on the map to provide information for a To-Be state. If there exist two or more nodes in a final To-Be node (*ToBeNode)*, a node that is the nearest one from a current node is selected for the next period. We recommend this approach to keep highly profitable customers who are predicted to defect because of high costs and risk in campaigns.

```
ToBeNode = φ, MaximumGain = -1
for every node n_i do begin
    NewNode = n_i
    getting ElapsedTime of NewNode
    ElapsedTimeGain = ElapsedTime of NewNode
                      - ElapsedTime of ExistingNode
    if ElapsedTimeGain > 0
      if ElapsedTimeGain > MaximumGain
        ToBeNode = φ, add n_i to ToBeNode,
        MaximumGain = ElapsedTimeGain
      elseif ElapsedTimeGain = MaximumGain
        add n_i to ToBeNode,
        MaximumGain = ElapsedTimeGain
      endif
    endif
endfor
recommend to drive to the nearest node
    from current node in ToBeNode for next period
```

Figure 9 - Defection prevention procedure for the innovative approach

5.4.3 Designing marketing campaigns

We can design segmental campaigns based on the information of a To-Be state. Figure 10 provides the real case for a campaign design. An As-Is behavior state and a To-Be behavior state for two segments are presented in this figure. From this figure, the campaign designs for both segment 'C03' and 'C04' should focus on extending session length. Therefore, the most effective campaign for customers in both segments is to provide additional free access time and recommend new attractive games.

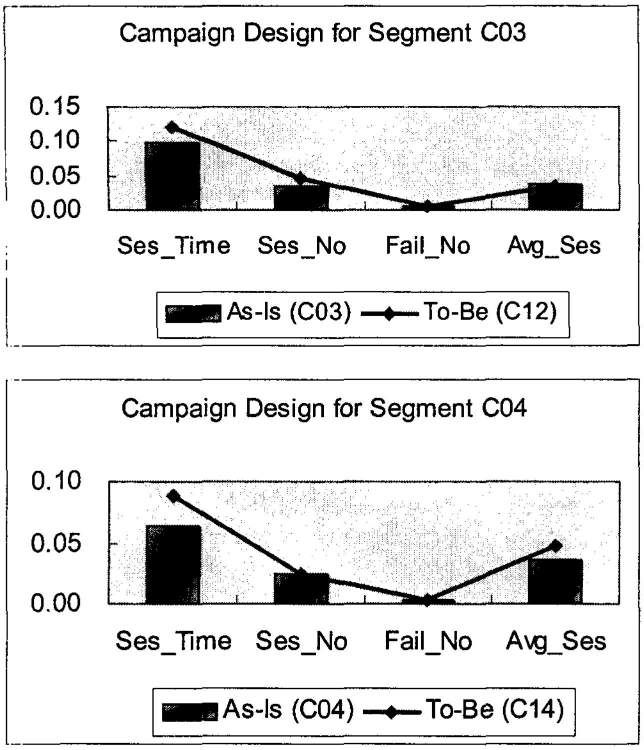

Figure 10 - Segmental campaign design

6. Evaluation

In determining the behavior states using SOM, we have to be very careful in deciding the number of states. The decision for determining the number of states is usually dependent on the characteristics of the domain and the goals of the decision-makers. Therefore we determine the satisfactory number of states by domain experts in this field (i.e. online games).

In this section, we compare the prediction accuracy of the proposed procedure to that of the MLP neural network and a decision tree. To evaluate the performance of our defection detection procedure, we use *FalsePositive* and *FalseNegative* as measures. *FalseNegative* error is more important than *FalsePositive* error because *FalseNegative* is more costly and dangerous than *FalsePositive*. If a non-defector is incorrectly classified as a defector (i.e. *FalsePositive*), he/she will receive a large concession, but if a defector is incorrectly classified as a non-defector (i.e. *FalseNegative*), he/she will terminate services before a company reaches him/her and thus the company will lose his/her revenue (Datta et al., 2000).

Figure 11 shows the comparative results of prediction accuracy among our suggested defection detection and prevention model (MP in Figure 11.), the MLP neural network (with multi nodes in two hidden layers) and a decision tree (C5.0; no cross-validation, no boosting option) based on the same input data in our online game domain. Our proposed procedure for defection detection and prevention resulted in both a slightly higher hit ratio and a lower falsepositive and falsenegative than the MLP neural network model in the test data set. And our procedure gives a performance almost as good as DT. This means that our proposed procedure can cover defection prevention as well as defection detection without deterioration of prediction accuracy compared to that of the MLP neural network and DT.

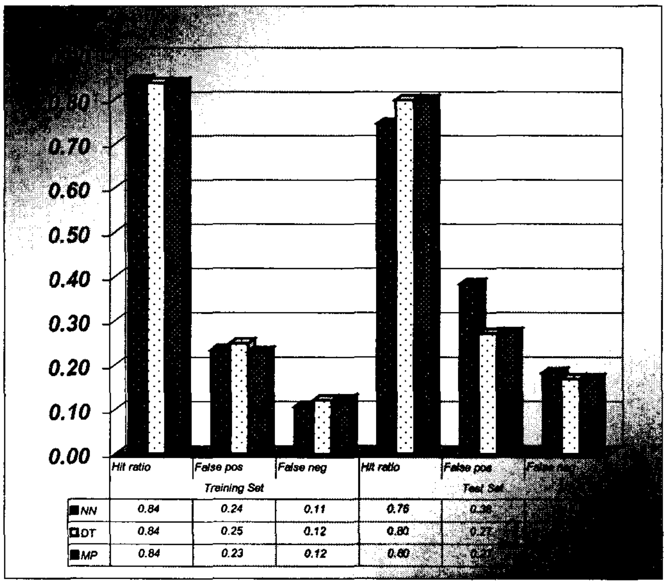

Figure 11 - Comparative results of prediction accuracy

7. Conclusion

We proposed a dynamic procedure for defection detection and prevention based on the observation that potential defectors have a tendency to change their behavior gradually before their eventual withdrawal. For this purpose, possible states of customer behavior are determined from past behavior data using SOM. Based on this state representation, the transition probability matrix can be formed and the elapsed time to actual defection can be predicted using a Markov chain. Also potential defectors are detected according to the elapsed time of each behavior state. In addition, our proposed procedure is extended to defection prevention for potential defectors and it assists in building segmental campaign plans by recommending the desirable behavior state for the next period to lower the likelihood of defection. Our proposed approach is a practical implementation procedure of eCRM because it tries to maintain a relationship with potential customers using an automated campaign procedure continuously.

As an area for further research, we have a plan to develop a system for defection detection and prevention based on our suggested procedure. In this study, we applied our procedure to the online gaming industry, but it will be a promising research area to apply to other service industries such as telecommunications, Internet access services, and contents providing services, and check the effectiveness of our proposed procedure.

References

[1] Alhoniemi, E., Hollmen, J., Simula, O., and Vesanto, J. (1999). "Process monitoring and modeling using the Self-Organizing Map," *Integrated CAE*, Vol. 6(1), pp. 3-14.

[2] Almuallim, H., and Dietterich, T. G. (1994). "Learning Boolean concepts in the presence of many irrelevant features," *Artificial Intelligence*, Vol. 69, pp. 279-306.

[3] Chakraborty, G., and Chakraborty, B. (2000). "A novel normalization technique for unsupervised learning in ANN," *IEEE transactions on neural networks*, Vol. 11(1), pp. 253-257.

[4] Cooley, R., Mobasher, B., and Srivastava, J. (1999). "Data preparation for mining world wide web browsing patterns," *Knowledge and information systems*, Vol. 1(1), pp.5-32.

[5] Datta, P., Masand, B., Mani, D. R., and Li, B. "Automated cellular modeling and prediction on a large scale," *Artificial Intelligence Review*, Vol. 14, pp.485-502.

[6] Ha, S. H., Bae, S. M., and Park, S. C. (2002). "Customer's time-variant purchase behavior and corresponding marketing strategies: An online retailer's case," *Computers & Industrial Engineering*, Vol. 43, pp. 801-820.

[7] Hall, M. A., and Smith, L. A. (1998). "Practical feature subset selection for machine learning," *Proceedings of the 21st Australian Computer Science Conference*, pp. 181-191.

[8] Kira, K., and Rendell, L. A. (1992). "The feature selection problem: Traditional methods and a new algorithm," *Proceedings AAAI-92*, pp. 129-134.

[9] Mozer, M. C., Wolniewicz, R., Grimes, D. B., Johnson, E., and Kaushansky, H. (2000). "Predicting subscriber dissatisfaction and improving retention in the wireless telecommunications industry," *IEEE Transactions on neural networks*, Vol. 11(3), pp. 690-696.

[10]Ng, K., and Liu, H. (2000). "Customer retention via data mining," *Artificial Intelligence Review*, Vol. 4(6), pp. 569-590.

[11]Raghavan, N., Bell, R. M., and Schonlau, M. (2000). "Defection detection: Using online activity profiles to predict ISP customer vulnerability," *Proceedings of the sixth International Conference of Knowledge Discovery and Data Mining*, pp. 506-515.

[12]Simula, O., Vasara, P., Vesanto, J., and Helminen, R. (1999). *The Self-Organizing Map in industry analysis*. CRC Press.

[13]Smith, K. A., Willis, R. J., and Brooks, M. (2000). "An analysis of customer retention and insurance claim patterns using data mining: A case study," *Journal of the operational research society*, Vol. 51, pp. 532-541.

[14]Song, H. S. (2003). "Detecting the change of customer behavior for eCRM: Organization and individual perspective," Ph.D. diss., KAIST, Korea.

[15]Trubik, E., and Smith, M. (2000). "Developing a model of customer defection in the Australian banking industry," *Managerial Auditing Journal*, Vol. 15(5), pp. 199-208.

[16]Wei, C. P., and Chiu, I. T. (2002). "Turning telecommunications call details to churn prediction," *Expert Systems with Applications,* Vol. 23, pp.103-112.

[17]Winston, W. L. (1994). Operations research applications and algorithms, 3rd ed. Duxbury press.