2014 International Symposium on Technology Management and Emerging Technologies (ISTMET 2014), May 27 - 29, 2014, Bandung, Indonesia

Applicability of Machine-Learning Techniques in Predicting Customer Defection

Niken Prasasti[1, 2] and Hayato Ohwada[2]
1. School of Business and Management, Bandung Institute of Technology, Indonesia
2. Department of Industrial Administration Department, Tokyo University of Science, Japan
niken.prasasti@sbm-itb.ac.id, ohwada@rs.tus.ac.jp

Abstract—**Machine learning is an established method of predicting customer defection from a contractual business. However, no systematic comparison or evaluation of the different machine-learning techniques has been performed. In this study, we provide a comprehensive comparison of different machine-learning techniques with three different data sets of a software company to predict customer defection. The evaluation criteria of the techniques are understandability of the model, convenience of using the model, time efficiency in running the learning model, and performance of predicting customer defection.**

Keywords: Customer defection, Machine learning, Classification, J48 Decision Tree, Random forest, Neural network, SVM

I. INTRODUCTION

Companies and industries are adopting machine-learning techniques in a wide range of applications. The major focus of machine-learning research is to extract information automatically from data using computational and statistical methods. From a broad perspective, machine learning involves giving software the ability to build knowledge from experience, derived from patterns and rules extracted from a large volume of data [1].

Current research in machine learning assists companies in developing their business strategies. For instance, in the insurance, mass media, and telecommunications industries, machine learning is applied to identify customers with high probability of defecting from a service that they provide. They do so by considering information derived from the usage-patterns of past customers. Previous techniques in predicting customer defection include logistic regression [2], decision trees [3], support-vector machines (SVMs) [4], neural artificial networks [5], and random forests (RFs) [6]. In our previous study [7], we investigated predicting customer defection using the SVM and the J48 Decision Tree, both of which perform well for the prediction model.

While recent research has focused on evaluating the performance of each machine-learning technique, no comparison of other machine-learning features (e.g., understandability, convenience, time efficiency, and visualization of the techniques) has been performed. This paper presents a comprehensive comparison of machine-learning techniques, especially in predicting customer defection. It evaluates not only the performance, but also the features of machine learning that have not yet been addressed in the literature. Based on experiment results, recommended machine-learning technique should be considered when predicting customer defection.

The remainder of this paper is organized as follows. Section 2 reviews the problem description. Section 3 describes the data sets and variables used for machine-learning procedures. Section 4 presents the machine-learning techniques used in this study. Section 5 provides results and a comparison of machine-learning techniques used in predicting customer defection. Section 6 presents a tabulation and discussion of the results. Finally, the last section provides conclusions.

II. PROBLEM DESCRIPTION

The term “defection” is widely used in businesses with a contractual customer base. A characteristic of a contractual business is that usage and retention are related concepts; customers must renew their contracts to continue access to a service [8]. In this study, we focused on applying machine-learning techniques to analyze customer defection from a software company as an example of a contractual business. There is a one-year contract between the customer and the company. The company offers three main products that vary by product price: Low-Price, Mid-Price, and High-Price.

The company has an e-commerce site that sends a confirmation of auto-renewal e-mail to each customer at least twice between zero days and fifty days before their renewal

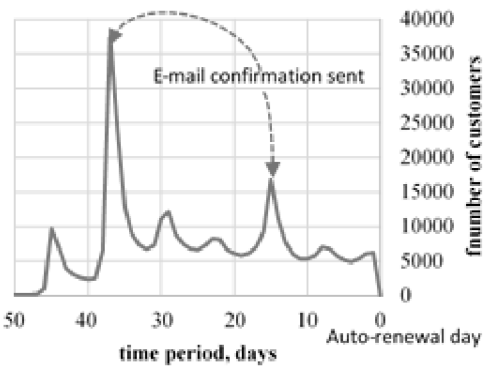

Fig. 1. Customer defection during the confirmation period

time. The customer must choose whether to opt-in or opt-out. Customers choose to opt-in if they want to be contacted with a particular form (e.g., a renewal form). In contrast, they choose to opt-out if they prefer not to renew (i.e., defect). Figure 1 plots statistics regarding the number of customers who defect within fifty days before the renewal deadline.

Typically, customer defection can be predicted by machine learning using customers' basic demographics and records of usage. In this study, we predicted customer defection using historical data of customers' opting-in and opting-out activity. Data sets and variables used will be described in the following section.

III. Data Sets

The data sets used in the experiments were provided by the software company. We executed learning procedures on different data sets of three different products categories: Low-Price, Mid-Price, and High-Price. Each data set has more than 20,000 records for 2007 through 2013, with six predictor variables. One issue in the data is that some customers tend to opt-in for another product from the same company after they have opted-out from a previous one (which should not be defined as defection), while the e-commerce site is able to record only opt-out data. Therefore, data preparation is very important in this research.

The original records contain the pattern of cancellation of customers after they choose the opt-out option. Before applying the data to the prediction models, we screened data so that only data that represent real defection (when the customer who chose to opt-out does not opt-in for another product). The variables used in the machine-learning procedures are listed in Table 1.

TABLE I. Variables Used in Machine-Learning Procedures

Variable	Definition
UPDATE_COUNT	Total count of renewals and purchases (first purchase is excluded).
CC_PRODUCT_PRICE	Recently purchased product price.
OPTIONAL_FLAG	Whether customer used optional service flag.
ORG_FLAG	Type of customer, whether individual or organization.
MAIL_STATUS	Delivery status of e-mail.
CLASS	Type of customer (defecting or retained).

TABLE II. Number of Samples Used in Machine-Learning Procedures

Product	Positive	Negative
Low-Price	13,709	5,302
Mid-Price	8,013	1,764
High-Price	10,961	2,265

UPDATE_COUNT is calculated using the results of data preparation and describes the total count of renewal and purchase records of customers, not including the first purchase. CLASS is the main variable that defines whether or not a customer is classified as defecting. The class distribution for machine learning for each dataset is presented in Table 2.

IV. Machine-Learning Procedures

Several machine-learning techniques are applied to predict customer defection. Intuitively, defection prediction is a simple classification problem. It can be solved by using a classifier that discriminates between customers, based on the variables of customer records. A set of labelled training examples is given to the learner, and the classifier is then evaluated on a set of instances. We applied universal learning techniques in predicting customer defection: decision tree, neural network, and SVM. We used the WEKA J48, RF, multilayer perceptron (MLP), and sequential minimal optimization (SMO) classifiers. We performed parameter-tuning on all machine-learning techniques in order to achieve the best performance on the given data sets. With many approaches used in previous research, some machine-learning algorithms are not tuned at all if the performance of the defection prediction is already sufficient with the default parameters set by the learning tools.

A. J48 Decision Tree

The decision tree is a predictive machine-learning technique that determines the target value (dependent variable) of a new sample based on various attribute values of the available data [9]. Like other decision-tree techniques, the WEKA J48 Decision Tree follows a simple algorithm. Using the attributes of available training data, it first creates a decision tree to classify a new item. It then analyzes the attribute that most obviously discriminates the various instances and looks for another attribute that gives the highest information gain. It continues the process until it obtains a clear decision as to what combination of attributes gives a particular target value, and it stops when it runs out of attributes.

B. Random Forests

The RF involves three main concepts: trees, bootstrap, and aggregation. This learning technique consists of bagging unpruned decision-tree learners with a randomized selection of features at each split [10]. It follows the same algorithm for both classification and regression. First, it draws n_{tree} bootstrap samples from the original data. For each bootstrap sample, it grows an unpruned classification or regression tree. Each tree gives a classification and votes for the most popular class. Next, the forest chooses to classify the case according to the label with the most votes over all trees in the forest [11].

C. Neural Networks

Neural networks can be classified into single-layer perception and multilayer perceptron (MLP). They have a remarkable ability to derive meaning from complicated data and generally can be used to extract patterns and detect complex problems that are not easily noticed using other techniques. We used the MLP function in WEKA. The MLP neural network is a nonlinear predictive model where the inputs are transformed to outputs using weights, bias terms, and activation functions [12]. The MLP neural network is considered in this study because nonlinear relationships were found in some previous research in customer defection.

D. *Support Vector Machines (SVM)*

We used the WEKA sequential minimal optimization (SMO) algorithm for training the support vector classifier. It is one of the most universal algorithms for large-margin classification by SVM. SVM is a classification technique based on neural network technology using statistical learning theory [13]. It looks for a linear optimal hyperplane so that the margin of separation between the positive class and the negative class is maximized. In practice, most data are not linearly separable; thus, to make the separation feasible, transformation is performed using a Kernel function. The input is transformed into a higher dimensional feature space using nonlinear mapping [14].

A decision on the Kernel function is needed in implementing SVM. The kernel defines the function class with which we are working. Instead of using the linear, sigmoid, or polynomial kernel, we used the squared exponential kernel (RBF) because it is generally more flexible than the other kernels and thus can model more functions with its function space.

V. RESULTS

In order to provide comprehensive comparison of machine-learning techniques for predicting customer defection, we use four criteria to evaluate the techniques: understandability of the model, convenience of using the model, time efficiency in running the learning model, and performance of predicting customer defection.

A. *Understandability of the Model*

The understandability of a machine-learning model is difficult to formalize, as it is a very subjective concept. However, in developing the measurement of understandability, we defined our judgment based on the following questions.

- Is it easy to know whether the model works or not?
- Does the learning algorithm help us understand the model better?
- Are the results of the technique easily interpreted?

Decision trees are well known for their simplicity and understandability. The decision tree is produced by algorithms that identify various ways of splitting a data set into branches (segments). It follows the simple and understandable algorithm described in the previous section. The visualization of the J48 Decision Tree output is clear and readable.

```
UPDATE_COUNT <= 2
|   CC_PRODUCT_PRICE <= 4700: TRUE (3397.0/81.0)
|   CC_PRODUCT_PRICE > 4700
|   |   UPDATE_COUNT <= 0
|   |   |   MAIL_STATUS = TRUE
|   |   |   |   CC_PRODUCT_PRICE <= 7048
|   |   |   |   |   VSSA_FLAG = FALSE
|   |   |   |   |   |   CC_PRODUCT_PRICE <= 5200: TRUE (406.0/167.0)
|   |   |   |   |   |   CC_PRODUCT_PRICE > 5200: FALSE (24.0/7.0)
|   |   |   |   |   VSSA_FLAG = TRUE: FALSE (4.0)
|   |   |   |   CC_PRODUCT_PRICE > 7048: TRUE (211.0/8.0)
|   |   |   MAIL_STATUS = FALSE: TRUE (1120.0/69.0)
|   |   UPDATE_COUNT > 0
|   |   |   CC_PRODUCT_PRICE <= 4743: FALSE (873.0/297.0)
|   |   |   CC_PRODUCT_PRICE > 4743
|   |   |   |   CC_PRODUCT_PRICE <= 8800: TRUE (306.0/15.0)
|   |   |   |   CC_PRODUCT_PRICE > 8800
|   |   |   |   |   ORGFLAG = FALSE: FALSE (53.0/17.0)
|   |   |   |   |   ORGFLAG = TRUE
|   |   |   |   |   |   CC_PRODUCT_PRICE <= 13315
|   |   |   |   |   |   |   MAIL_STATUS = TRUE: TRUE (4.0/1.0)
|   |   |   |   |   |   |   MAIL_STATUS = FALSE: FALSE (6.0/1.0)
|   |   |   |   |   |   CC_PRODUCT_PRICE > 13315: TRUE (6.0)
UPDATE_COUNT > 2
|   UPDATE_COUNT <= 3
|   |   VSSA_FLAG = FALSE
|   |   |   CC_PRODUCT_PRICE <= 8700
|   |   |   |   CC_PRODUCT_PRICE <= 4743: TRUE (1114.0/436.0)
|   |   |   |   CC_PRODUCT_PRICE > 4743
|   |   |   |   |   CC_PRODUCT_PRICE <= 8300
|   |   |   |   |   |   CC_PRODUCT_PRICE <= 6648: FALSE (26.0/5.0)
|   |   |   |   |   |   CC_PRODUCT_PRICE > 6648: TRUE (3.0)
|   |   |   |   |   CC_PRODUCT_PRICE > 8300: FALSE (5.0)
|   |   |   CC_PRODUCT_PRICE > 8700
|   |   |   |   CC_PRODUCT_PRICE <= 8800: TRUE (85.0/9.0)
|   |   |   |   CC_PRODUCT_PRICE > 8800: FALSE (18.0/4.0)
|   |   VSSA_FLAG = TRUE: FALSE (6.0/1.0)
|   UPDATE_COUNT > 3
|   |   CC_PRODUCT_PRICE <= 8300: FALSE (2009.0/585.0)
|   |   CC_PRODUCT_PRICE > 8300
|   |   |   CC_PRODUCT_PRICE <= 8800: TRUE (77.0/11.0)
|   |   |   CC_PRODUCT_PRICE > 8800
|   |   |   |   UPDATE_COUNT <= 5: FALSE (22.0/5.0)
|   |   |   |   UPDATE_COUNT > 5: TRUE (2.0)

Number of Leaves  :     23

Size of the tree :      45
```

Fig. 2. Visualization of J48 Decision Tree classification results

The J48 Decision Tree is one of learner that can have a tree structure visualized. Figure 2 presents a decision tree constructed using the J48 classifier and illustrates how the classifier uses the attributes to make a decision. The leaf nodes indicate the class to which an instance will be assigned if that node is reached. The numbers in parentheses after the leaf nodes indicate the number of instances assigned to that node, followed by how many of those instances are incorrectly classified as a result. With other classifiers, some other output is given to indicate how the decisions are made (e.g., a rule set). RF produces an ensemble of trees (not just one, like J48); thus, the output only calculates learning performance.

In generating neural networks, WEKA has its own graphical user interface (GUI) function that can be set to true before the learning process starts, to help us understand the model better, can be seen in Fig. 3. The model of neural network prediction using the MLP algorithm is presented in Fig. 4.

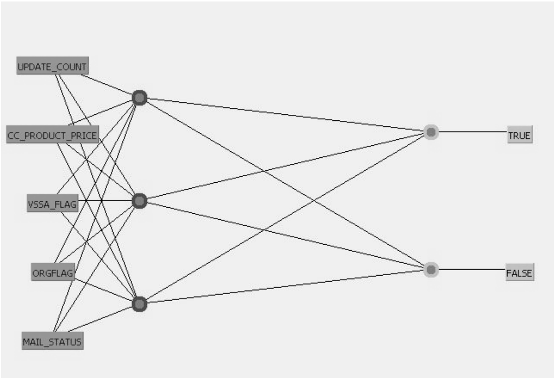

Fig. 3. The GUI of MultiLayerPerceptron at the beginning of running the model

```
=== Classifier model (full training set) ===

Sigmoid Node 0
    Inputs    Weights
    Threshold    -1.0450140675045605
    Node 2    2.3197643557624232
    Node 3    0.7750567095888843
    Node 4    1.9140976755512467
Sigmoid Node 1
    Inputs    Weights
    Threshold    1.0450140675045605
    Node 2    -2.3197643557624237
    Node 3    -0.7750567095888847
    Node 4    -1.9140976755512462
Sigmoid Node 2
    Inputs    Weights
    Threshold    -11.624510298738192
    Attrib UPDATE_COUNT    -19.356026030966717
    Attrib CC_PRODUCT_PRICE    10.154993835402767
    Attrib VSSA_FLAG    0.4508535673283764
    Attrib ORGFLAG    -0.4073094811798154
    Attrib MAIL_STATUS    3.708201552853084
Sigmoid Node 3
    Inputs    Weights
    Threshold    -13.379433657648118
    Attrib UPDATE_COUNT    -8.07082353592674
    Attrib CC_PRODUCT_PRICE    -2.011440494392915
    Attrib VSSA_FLAG    -3.8306305737742643
    Attrib ORGFLAG    0.17965146614814542
    Attrib MAIL_STATUS    -3.6070164508068965
Sigmoid Node 4
    Inputs    Weights
    Threshold    9.527502749473129
    Attrib UPDATE_COUNT    -14.824765439819926
    Attrib CC_PRODUCT_PRICE    31.20062432935239
    Attrib VSSA_FLAG    -17.34613299880761
    Attrib ORGFLAG    -0.8236167322743257
    Attrib MAIL_STATUS    -0.021572173627486806
Class TRUE
    Input
    Node 0
Class FALSE
    Input
    Node 1
```

Fig. 4. The learning model of MLP

The SMO algorithms implement the sequential minimal-optimization algorithm for training a support vector classifier using kernel functions; here we used the RBF kernel. Figure 5 presents the output of SMO on the customer defection data. Since the customer defection data contains two class values, two binary SMO models have been output, with one hyperplane to separate each possible pair of class values. The hyperplanes are expressed as functions of the attribute values in the original space [20].

```
Kernel used:
  RBF kernel: K(x,y) = e^-(0.1* <x-y,x-y>^2)

Classifier for classes: TRUE, FALSE

BinarySMO

       1      * <0.142857 0.089977 0 0 0 > * X]
 +       1      * <0.285714 0.281697 0 0 0 > * X]
 -       1      * <0.285714 0.078493 0 0 1 > * X]
 -       1      * <0.428571 0.078493 0 0 0 > * X]
 +       1      * <0.571429 0.078493 0 0 0 > * X]
 +       1      * <0.571429 0.078493 0 0 1 > * X]
 -       1      * <0.285714 0.078493 0 0 0 > * X]
 +       1      * <0.571429 0.078493 0 0 0 > * X]
 +       1      * <0.285714 0.089977 0 0 0 > * X]
 -       1      * <0.142857 0.078493 0 0 0 > * X]
 +       1      * <0.285714 0.675063 0 0 1 > * X]
 -       1      * <0.285714 0.078493 0 0 0 > * X]
 -       1      * <0.285714 0.078493 0 0 0 > * X]
 +       1      * <0 0.089977 0 0 1 > * X]
 -       1      * <0.714286 0.087945 0 0 1 > * X]
 +       1      * <0.571429 0.087945 0 0 0 > * X]
 +       1      * <0.571429 0.078493 0 0 0 > * X]
 -       1      * <0.714286 0.078493 0 0 1 > * X]
 -       1      * <0.285714 0.078493 0 0 0 > * X]
 +       1      * <0.428571 0.078493 0 0 0 > * X]
 +       1      * <0.571429 0.078493 0 0 1 > * X]
 -       1      * <0.428571 0.078493 0 0 0 > * X]
 -       1      * <0.714286 0.078493 0 0 0 > * X]
 -       1      * <0.428571 0.078493 0 0 0 > * X]
 +       1      * <0 0.089977 0 0 0 > * X]
 +       1      * <0 0.089977 0 0 1 > * X]
 +       1      * <0.571429 0.078493 0 0 0 > * X]
 +       1      * <0.571429 0.078493 0 0 0 > * X]
 -       1      * <0.571429 0.078493 0 0 0 > * X]
 +       1      * <0.571429 0.078493 0 0 0 > * X]
 +       1      * <0 0.089977 0 0 1 > * X]
 -       1      * <0.428571 0.078493 0 0 0 > * X]
 -       1      * <0.428571 0.281697 0 0 0 > * X]
 +       1      * <0 0.089977 0 0 0 > * X]
 +       1      * <0.285714 0.089977 0 0 1 > * X]
 +       1      * <0.428571 0.078493 0 0 0 > * X]
 -       1      * <0.285714 0.078493 0 0 0 > * X]
 -       1      * <0.142857 0.078493 0 0 1 > * X]
 +       1      * <0.428571 0.087945 0 0 1 > * X]
 -       1      * <0.714286 0.078493 0 0 0 > * X]
 +       1      * <0.428571 0.089977 0 0 0 > * X]
 +       1      * <0.142857 0.281697 0 0 0 > * X]
 -       0.8576 * <0.142857 0.078493 0 0 0 > * X]
 +       1      * <0.571429 0.078493 0 0 0 > * X]
 -       1      * <0.142857 0.078493 0 0 0 > * X]
```

Fig. 5. Part of the output of SMO on the customer defection data

B. *Convenience of Using The Model*

The learning method using a customer-defection model consists of a set of algorithms. It requires setting parameters to achieve expected results. In this study, the convenience of using each model is represented by the ease of tuning the parameters before proceeding with the algorithm. From the machine-learning perspective, classification can be defined as a method of searching for a function that maps the space of attributes of the domain to the target classes [15].

Decision trees are probably the most common learning method used for the customer-defection problem. Generally, with the WEKA J48 Decision Tree, the default parameter values already give the best performance across all data sets. Previous research [16] indicated that by reducing error pruning (using the –R –N 3 flag) with J48 we can improve model performance; however, with the present customer defection prediction case, the default values give better performance.

Like other decision trees, RFs have very few parameters to tune and can be used quite efficiently with the default parameters. Using the WEKA RF, we changed one main parameter in RF, the number of trees. We found that increasing the number of trees while tuning to the default value of 500 (for 20,000 predictors [17]) greatly improved performance.

SMO is more complicated to tune. When using it with WEKA, two parameters can be tuned: the complexity value of SMO and the gamma value of the kernel used by SMO. To find the best parameter for the model, we used the GridSearch function in WEKA, which enables us to optimize two parameters of an algorithm by setting it on maximum value, minimum value, base value, and step value for how much a parameter can be increased for each test [18]. The main advantage of GridSearch is that it is not limited to first-level parameters of the base classifier, and we can specify paths to the properties that we want to optimize.

The default parameters in the WEKA MLP are quite sensible for the model. With MLP, determining the learning rate is very important [19]. Hence, we made changes in the learning rate parameter -L to 0.1 and 0.5, and determined that using default –L 0.3 results in optimum performance.

C. *Time Efficiency in Building the Model*

It is important to consider time when using machine-learning techniques for predicting customer defection. We compared the time required to run the learning model of each classifier using WEKA. For three different data sets, decision trees require the least time to build the model and to calculate performance. J48 performed more quickly than RF, especially when we increased the number of trees in the RF above the default.

Although MLP requires more time than decision trees, it is still acceptable, since it takes less than 10sec for one run on every data set. The SMO SVM required the longest time up to more than 5 minutes to build the model after we tuned the kernel function in the RBF kernel.

TABLE III. TIME REQUIRED BY CLASSIFIER FOR EACH DATA SET

Product	**Time needed to build model (seconds)**			
	J48	***RF***	***MLP***	***SVM***
Low-Price	0.11	4.35	5.6	280.7
Mid-Price	0.13	5.66	4.3	299..8
High-Price	0.13	5.44	4.3	342.4

D. *Performance on Predicting Customer Defection*

A classification task involves deciding which set of categories or labels should be assigned to data according to attributes of the data. In predicting customer defection, there are two possible classes: defect or retain. Commonly, the performance of a classifier task is measured by accuracy. For a data set, if a classifier can correctly guess the label of half the examples, its accuracy is 50%. In this study, we also calculated the precision and recall of each classifier to avoid regarding one classifier model as better than another one based on accuracy alone.

TABLE IV. COMPARISON OF CLASSIFIER PERFORMANCE

Product	**Classifier**	**Accuracy**	**Recall**	**Precision**
Low-Price	J48	72.12%	83.91%	74.10%
	RF	72.28%	84.21%	74.14%
	MLP	68.81%	80.51%	72.04%
	SMO	68.81%	84.93%	70.42%
Mid-Price	J48	81.95%	85.80%	88.14%
	RF	82.32%	86.12%	88.22%
	MLP	78.73%	91.32%	80.83%
	SMO	82.28%	90.41%	80.92%
High-Price	J48	82.87%	76.39%	92.87%
	RF	83.13%	77.68%	92.61%
	MLP	68.57%	67.57%	76.54%
	SMO	82.71%	75.21%	91.51%

Table 4 compares the accuracy, recall, and precision scores of four classifiers for three data sets. The table presents experiment results for all 10-fold cross-validations. It can be safely concluded that no single model had the highest accuracy for all three data sets. The accuracies of the four classifiers on the Low-Price product data sets were similar. The performance of every algorithm differed, depending on the characteristics and type of data. However, decision trees and SVM gave more stable results.

VI. DISCUSSION

Table 5 summarizes the results of evaluation criteria of all classifier techniques (high indicates good value and low indicates poor value). To the best of our knowledge and based on the results of the experiment, the J48 Decision Tree gives higher understandability (from the algorithm and the result visualization), convenience of use, and time efficiency. Its high performance should be considered for predicting customer defection.

Though RFs have high accuracy in each prediction on all data sets, in practice they have poorer understandability than the J48 Decision Tree in the present defection prediction case. Hence, convenience of use and time efficiency are advantages of this decision-tree model. Some recent research applied RF when the number of predictor variables was high.

The neural-network model seems to be unsuitable for predicting customer defection using data sets with the characteristics described in the third section. It has lower performance on all data sets, although it has high understandability and time efficiency.

The last classifier, SMO as the SVM tool, has higher predicting performance. SVMs are well-known for their good learning performance. However, SMO is more complicated than the other classifiers. One of its weaknesses is the time required to build and run the model, especially with a huge amount of input data.

TABLE V. COMPARISON OF CLASSIFIER PERFORMANCE

Criteria	Classifiers			
	J48	*RF*	*MLP*	*SMO*
Understandability	Higher	Low	High	Low
Convenience	Higher	Higher	Low	Low
Time efficiency	Higher	Higher	High	Low
Performance	High	High	Lower	Higher

VII. CONCLUSION

Machine learning is an established method of predicting customer defection from a contractual business. We applied some machine-learning classifier techniques to predict customer defection from a software company and provided a comprehensive comparison of four classifiers: J48 Decision Tree, RFs, neural networks, and SVM. We used four evaluation criteria: understandability of the learning model, convenience of using the model, prediction performance, and time efficiency.

Finally, we conclude that for predicting customer defection, each classifier has its best criteria. In this study, we concluded that the J48 Decision Tree and SVM models were excellent due to compatibility with the data sets. However, these findings are limited to only customer defection cases with typical data sets. The results may be different for other data sets with other prediction variables.

REFERENCES

[1] Mitchell, T.: Machine Learning: McGraw Hill, 1997.

[2] Nie, G., Rowe, W., Zhang, L., Tian, Y., & Shi, Y. (2011). Credit card churn forecasting by logistic regression and decision tree. *Expert Systems with Applications*, *38*(12), 15273–15285. doi:10.1016/j.eswa.2011.06.028

[3] Bin, L., Peiji, S., & Juan, L. (2007). Customer Churn Prediction Based on the Decision Tree in Personal Handyphone System Service. *2007 International Conference on Service Systems and Service Management*, 1–5. doi:10.1109/ICSSSM.2007.4280145

[4] Coussement, K., & Poel, D. Van Den. (2006). Churn Prediction in Subscription Services: an Application of Support Vector Machines While Comparing Two Parameter-Selection Techniques Kristof.

[5] Sharma, A. (2011). A Neural Network based Approach for Predicting Customer Churn in Cellular Network Services. *International Journal of Computer Application*, *27*(11), 26–31.

[6] Ying, W., Li, X., Xie, Y., Johnson, E., & Engineering, S. (2008). Preventing Customer Churn by Using Random Forests Modeling. *IEEE International Conference on Information Reuse and Integration*, *3*, 429–435

[7] Prasasti, N., Okada, M., Kanamori, K., & Ohwada, H. (2013). Forthcoming. Customer Lifetime Value and Defection Possibility Prediction Model using Machine Learning : An Application to a Cloud-based Software Company. *Lecture Notes in Computer Science*, *8398*.

[8] Ascarza, E., & Hardie, B. G. S. (2013). A Joint Model of Usage and Churn in Contractual Settings. Journal of Marketing Science, (February), 1-5.

[9] Padhye, A. (n.d.). Chapter 5: Classification Method. Retrieved from http://www.d.umn.edu/~padhy005/Chapter5.html

[10] Montillo, A. A. (University of P. (n.d.). Random Forests. Retrieved from http://www.dabi.temple.edu/~hbling/8590.002/Montillo_RandomForests_4-2-2009.pdf

[11] Coussement, K., & Poel, D. Van Den. (2008). Improving Customer Attrition Prediction by Integrating Emotions from. Gent.

[12] Glady, N., Baesens, B., & Croux, C. (2009). Modeling Churn Using Customer Lifetime Value. *European Journal of Operational Research*, *197*(1), 402–411. doi:10.1016/j.ejor.2008.06.027

[13] Vapnik, V.N. 1995. *The Nature of Statistical Learning Theory*. New York: Springer – Verlag.

[14] Coussement, K., & Poel, D. Van Den. (2006). Churn Prediction in Subacription Services: an Application of Support Vector MAchines While Comparing Two Parameter-Selection Techniques.

[15] Koblar, V. (2012). Optimizing Parameters of Machine Learning Algorithms.

[16] Kotthoff, L., Gent, I. P., Miguel, I., Building, J. C., & Haugh, N. (n.d.). An Evaluation of Machine Learning in Algorithm Selection for Search Problems, 1–15.

[17] Janitza, A. B. S. (2012). Overview of Random Forest Methodology and Practical Guidance with Emphasis on Computational Biology and Overview of Random Forest Methodology and Practical Guidance with Emphasis on Computational Biology and Bioinformatics, (129).

[18] Weka - Optimizing Parameters. (n.d.). Retrieved from http://weka.wikispaces.com/Optimizing+parameters

[19] Why MultiLayer Perceptron / Neural Network ? (n.d.). Retrieved from http://courses.media.mit.edu/2006fall/mas622j/Projects/manu-rita-MAS_Proj/MLP.pdf

[20] Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data mining: practical machine learning tools and techniques*. Morgan Kauffman Publisher.