A Customer Churn Prediction Model in Telecom Industry Using Boosting

Ning Lu, Hua Lin, Jie Lu, *Member, IEEE*, and Guangquan Zhang

***Abstract*—With the rapid growth of digital systems and associated information technologies, there is an emerging trend in the global economy to build digital customer relationship management (CRM) systems. This trend is more obvious in the telecommunications industry, where companies become increasingly digitalized. Customer churn prediction is a main feature of in modern telecomcommunication CRM systems. This research conducts a real-world study on customer churn prediction and proposes the use of boosting to enhance a customer churn prediction model. Unlike most research that uses boosting as a method to boost the accuracy of a given basis learner, this paper tries to separate customers into two clusters based on the weight assigned by the boosting algorithm. As a result, a higher risk customer cluster has been identified. Logistic regression is used in this research as a basis learner, and a churn prediction model is built on each cluster, respectively. The result is compared with a single logistic regression model. Experimental evaluation reveals that boosting also provides a good separation of churn data; thus, boosting is suggested for churn prediction analysis.**

***Index Terms*—Boosting, churn prediction, customer relationship management, digital marketing, logistic regression, telecommunication.**

I. INTRODUCTION

CUSTOMER relationship management (CRM) is a strategic approach which targets the development of profitable, long-term relationships with key customers and stakeholders [1]. Due to saturated markets and intensive competition, more and more companies have recognized the importance of CRM and have changed their product-centric mass marketing champion strategies toward customer-centric targeted marketing. Nowadays, the rapid development of digital systems and associated information technologies provide enhanced opportunities to understand customers and build reliable digital CRM systems [2].

Manuscript received May 16, 2012; revised July 16, 2012; accepted September 04, 2012. Date of publication October 12, 2012; date of current version May 02, 2014. This work was supported by the Australian Research Council (ARC) through the Discovery Project DP110103733. Paper no. TII-12-0362.

N. Lu and G. Zhang are with the Department of Information Technology, University of Technology, Sydney NSW 2007, Australia (e-mail: Ning.Lu@uts.edu.au; Guangquan.Zhang@uts.edu.au)

J. Lu is with the Department of Information Technology and the Decision System & e-Service Intelligence Laboratory, QCIS Centre, Faculty of Engineering and Information Technology, University of Technology Sydney, Broadway, NSW 2007, Australia (e-mail: Jie.Lu@uts.edu.au;)

H. Lin is with Singtel Optus Pty Ltd., Sydney NSW 2113, Australia (e-mail: Hua.Lin@optus.com.au)

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2012.2224355

Customer churn management, as a part of CRM, has become a major concern. In mobile telecommunications, the term "churn" refers to the loss of subscribers who switch from one provider to another during a given period. Based on an earlier study [3], the estimated average churn rate for mobile telecommunications is about 2.2% per month. This means that one in fifty subscribers of a given company discontinues their services every month. As it is more profitable to retain existing customers than to constantly attract new customers [4]–[6], it is crucial to build an accurate churn prediction model for identifying those customers who are most prone to churn.

Established literature on customer churn uses various data mining technologies, such as neural networks [7], clustering [8], decision tree [7], [9], regression [10], [11], support vector machine [4], [12], and ensemble of hybrid methods [13], to provide more accurate predictions. According to a review on customer churn prediction modeling [14], Regression is the most commonly adopted technique, probably because of its high reported accuracy and interpretability for understanding key drivers, as well as for providing information to set up retention actions. As the churner usually takes only a fraction of the customer base, the problem of customer churn prediction is always combined with the problem of highly skewed class distribution or lack of churner data. One of the most common techniques for dealing with rarity is sampling [15]. Methods that adopt the sampling technique alter the distribution of training examples and generate balanced training set(s) for building churn prediction model(s) [8], [9], [12], [13]. However, a recent study [11] on the class imbalance issue in churn prediction reveals that advanced sampling technique does not increase predictive performance. Although the weighted Random Forests technique is suggested [11], tree ensembles, such as Random Forests, are often criticized for being hard to interpret [16], [17], i.e., it is difficult to identify risk factors which can be addressed by the retention process to prevent a customer from leaving, thus they are not the preferred methods in this study. Other research explores the power of new features for churn prediction, such as social network [18] and text information of customer complaints [19], which is beyond the scope of this paper.

This paper presents a churn prediction model in the telecommunication industry using a boosting algorithm which is believed to be very robust [11] and has demonstrated success in churn prediction in the banking industry [20]. The established literature only uses boosting as a general method to boost accuracy, and few researchers have ever tried to take advantage of the weight assigned by boosting algorithms. The weight also provides important information, specifically, outliers. The testing results show that boosting provides a good separation of the

1. Initialize the weight of sample s_i, $D_1(i)=1/N$.
2. For $t=1,\ldots,T$:
   a) Train a base classifier $h_t: X \rightarrow \{-1,+1\}$, using weighted training set D_t
   b) Compute the estimated error of h_t, $\varepsilon_t = \Pr_{i\sim D_t}[h_t(x_i) \neq y_i]$
   c) Choose $\alpha_t = \frac{1}{2}\ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$
   d) Set $D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t} & if \quad h_t(x_i) = y_i \\ e^{\alpha_t} & if \quad h_t(x_i) \neq y_i \end{cases} = \frac{D_t(i)\exp(-\alpha_t y_i h_t(x_i))}{Z_t}$
   where Z_t is a normalize factor, so that $\sum_i D_{t+1}(i) = 1$
3. Output the classifier $sign[H(x)] = sign\left[\sum_{t=1}^{T} \alpha_t h_t(x)\right]$

Fig. 1. AdaBoost algorithm.

customer base, which also leads to a better overall performance. This paper is organized as follows. Section II reviews related techniques concerning boosting (Section II-A) and logistic regression (Section II-B). Section III articulates our churn prediction model. Section IV shows the results of experimental evaluations and is followed by the conclusion and future work (Section V).

II. Methodologies

A. Boosting

Boosting refers to a general and provably effective method that attempts to "boost" the accuracy of any given learning algorithm [21]. Although boosting is not algorithmically constrained, most boosting algorithms involve learning iteratively and adding weak classifiers to come up with a final strong classifier. Each added weak classifier is usually weighted according to its accuracy and trained with reweighted training data.

One of the earliest and best-known boosting algorithms is AdaBoost [21]. The AdaBoost algorithm takes a training set $S = \{(x_i, y_i)\}$ as inputs, where $i = \{1, 2, \ldots, N\}$, $x_i = (x_{i_1}, x_{i_2}, \ldots, x_{i_n}) \in X$ and the label $y_i \in Y = \{-1, +1\}$. AdaBoost works by repeatedly training a base classifier based on a weighted training set and synthesizing these trained classifiers. Initially, all training samples are weighted equally, but the weights of incorrectly classified samples are increased for the next round, so that the base classifier is forced to focus on examples with higher weights in the training set. The pseudocode for AdaBoost is given in Fig. 1. As their algorithm uses a discrete base classifier (hypothesis) $h : X \rightarrow \{-1, +1\}$, their algorithm is also called Discrete AdaBoost in later literature.

Schapire and Singer [22] studied boosting in a more generalized framework. They proposed the use of a base classifier $f : X \longrightarrow \mathbb{R}$ to replace the discrete classifier $h(x)$, where $|f(x)|$ represents the confidence of its prediction. They also improved the choice of α_t by

$$\alpha_t = \frac{1}{2}\ln\left(\frac{W_t^+}{W_t^-}\right) \tag{1}$$

1. Initialize the weight of sample s_i equally, $D_1(i)=1/N$.
2. For $t=1,\ldots,T$:
   a) Estimate $f_t(x)$ by weighted least-squares fitting of y_i to x_i: $f_t = \arg\min_f \left(J_t = \sum_{i=1}^{N} D_t(i)(y_i - f_t(x_i))^2\right)$
   b) Set $D_{t+1}(i) = \frac{D_t(i)\exp(-y_i f_t(x_i))}{Z_t}$
   where Z_t is a normalize factor, so that $\sum_i D_{t+1}(i) = 1$
3. Output the classifier $sign[F(x)] = sign\left[\sum_{t=1}^{T} f_t(x)\right]$

Fig. 2. Gentle AdaBoost algorithm.

where $W_t^b = \sum_{i:y_i f_t(x_i)=b} D_t(i)$, $b \in \{-, 0, +\}$. When $W^0 = 0$, the choice of α will be identical to Discrete AdaBoost, as the latter can be reformed as $\alpha_t = (1/2)\ln\left(W_t^+ + (1/2)W_t^0/W_t^- + (1/2)W_t^0\right)$ according to their generalized framework.

From a statistical point of view, Friedman *et al.* [23] showed that boosting algorithms are step-wise procedures for fitting additive logistic regression and proposed the Gentle AdaBoost algorithm. The Gentle AdaBoost does not require the computation of the log ratios which may lead to very large updates, but always updates within range $[-1, +1]$. Thus, the Gentle AdaBoost is believed to be more reliable and stable. The detail of Gentle AdaBoost is given in Fig. 2.

We intentionally omit boosting algorithms for classification with multiple classes, as there are only two classes, churner or nonchurner, in our case. For a comprehensive review of boosting algorithms, please refer to [24].

B. Logistic Regression

Logistic Regression is a simple but effective analytic method which is used to describe and test hypotheses about relationships between a categorical variable and one or more categorical or continuous variables. Given a sample set $S = \{(x_i, y_i)\}$, where $i = \{1, 2, \ldots, N\}$, and $x_i = (x_{i_1}, x_{i_2}, \ldots, x_{i_n}) \in X$, to evaluate the relationship between a set of independent variables (inputs) $x_i \in R^n$ and a corresponding target label $y \in Y = \{-1, +1\}$, the logistic regression estimates the probability of $P(y = 1|x_i) = \hat{p}$ by

$$\hat{p} = \frac{1}{1 + \exp(-\sum_{j=0}^{n} \beta_i x_{i_j})} \tag{2}$$

where $\sum_{j=0}^{n} \beta_j x_{i_j} = \beta_0 + \beta_1 x_{i_1} + \cdots + \beta_n x_{i_n} = \ln(\hat{p}/1 - \hat{p}) = \text{logit}(\hat{p})$ is called the regression equation, with intercept β_0 and regression coefficients β_j, $j = \{1, 2, \ldots, n\}$.

In its application, the maximum-likelihood estimation is used to maximize the likelihood of the regression coefficients given a set of observations (samples).

III. Churn Prediction Modeling

A. Problem Description

There are two types of churn behavior: voluntary, in which a customer decides to terminate services, and involuntary,

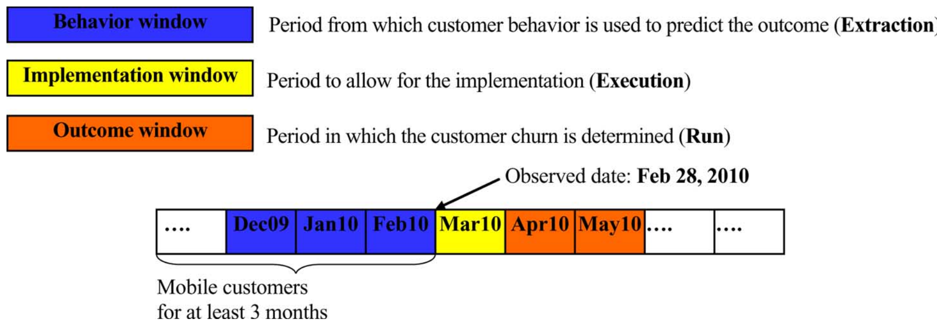

Fig. 3. Timeline of churn prediction model.

in which the service provider decides to terminate a customer's services (typically because of unpaid bills) [7]. This research considers only voluntary churners, because involuntary churners are easier to identify and are of less importance from a churn management perspective.

In contrast to most existing churn prediction models, our prediction model allows for an "Implementation Zone" within which the company is able to perform retention actions. As a result, customers who churn within the "Implementation Zone" are excluded from model building. The modeling timeline is given in Fig. 3. The timeline below shows that our model in essence aims to predict all customers who will churn voluntarily in the future two-month period (in the outcome window), based on their latest three-month information (in the behavior window).

B. Data Collection

We choose to study a data set from a telecommunication company which includes a segment of mobile customers (in the number of millions) who are active at a point of time in the year 2010. Because the churn prediction model may be biased due to a particular observed point of time, we base our studies on the annual churn records of the year 2010. We initially extract 700+ variables from the mobile customer database, including mobile plan and contract information, billing, usage, and product holding information, as well as customer care inbound/outbound information. All of these possible variables are defined as independent variables $x_i = (x_{i_1}, x_{i_2}, \ldots, x_{i_n})$. For each mobile customer, the churner label (yes/no) is collected in the two-month period following the initial date of extraction. This becomes the dependent variable or target variable y_i. For example, if we extract data from Dec, 2009 to Feb, 2010, then the churner label is determined in two-month period of time after March 2010 (April 2010–May 2010), note that we allow for an "Implementation Zone." This churner label will be the dependent variable y_i.

Our training set includes 7190 customers drawn randomly, with 678 churners and 6512 nonchurners. To validate our churn prediction model, a testing set of six-month customer information is collected in the year of 2011. A churn prediction is made based on each three-month period, in order to simulate the real-world scenario.

C. Variable Selection

In Section III-B, we extract customer information on hundreds of variables. Most of these variables, however, are irrelevant to churn prediction or of little relevance. The excess variables not only burden the computational process in generation of a model, but also interfere with the heuristic search for all practical model generation algorithms and result in an inferior model [25], [26]. Thus, we continue to evaluate the variable set in an attempt to eliminate some variables based on their predictive performance and produce a more predictive model.

We use a two-step process for variable selection. First, we select a set of variables that have the most impact on customer churn by the chi-square automatic interaction detection (CHAID) Analysis [27]. The CHAID Analysis is a form of analysis that determines how variables best combine to explain the outcome in a given dependent variable, which incorporates a sequential merge and split procedure based on a chi-square test statistic. CHAID uses suboptimal split on each predictor instead of searching for all possible combinations of categories, which reduces computation time. We choose CHAID as it is especially useful for data expressing categorized values, which suits our case; its output is highly visual, having a tree image, and is easy to interpret, which is very important from the perspective of checking by expert knowledge and retention management. With the CHAID Analysis, we narrow down the 700+ variables to 70 variables. Nevertheless, the number of variables is still far too many for an optimized prediction process, so we perform a stepwise logistic regression using a forward-selection approach. In Step 1, these 70 variables are split according to the optimum divisions obtained from a decision tree. We use these splits to define each group as a dummy variable. The interaction variables are obtained by looking at the correlations of the coefficients (>0.3) of the modeling variables. These 70 variables, plus a number of interaction variables, will be the inputs for a stepwise logistic regression. Our study employs a forward-selection procedure, in which variables are added to the model one at a time until a pre-set stopping rule is satisfied. Finally, 21 optimized significant variables are identified and used to form a logistic regression model.

D. Customer Separation

In general, a boosting algorithm works by repeatedly reweighting the data, so that examples that are misclassified gain weight and examples that are classified correctly lose

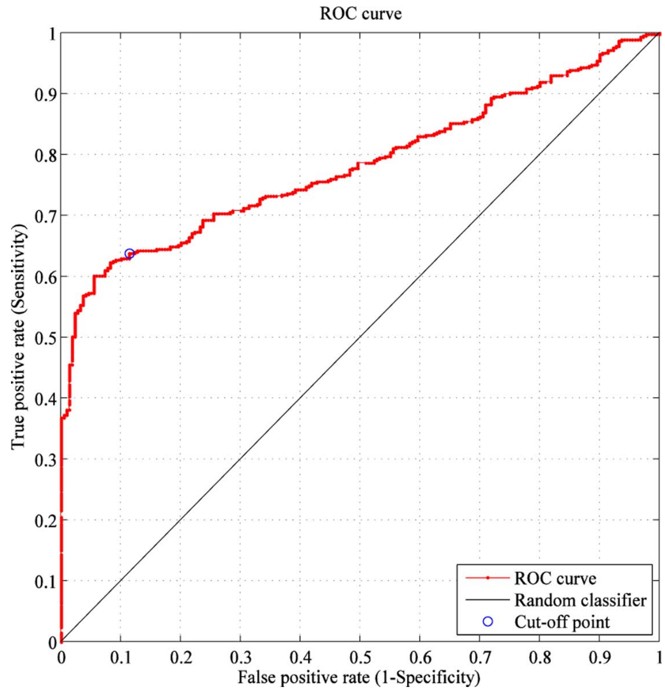

Fig. 4. ROC curves of the separation.

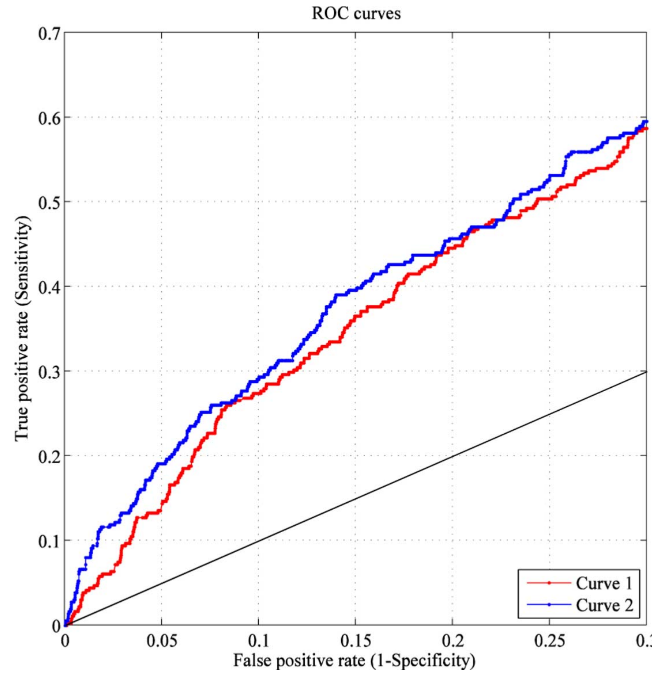

Fig. 5. ROC curves of predictions on Cluster-1.

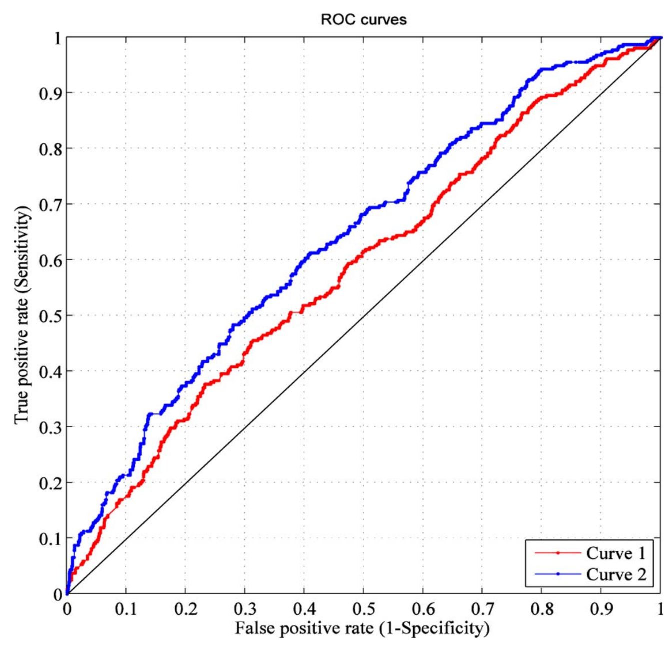

Fig. 6. ROC Curves of predictions on Cluster-2.

weight, thus forcing a base learning algorithm to focus on hard examples. This fact inspires our idea of separating customers based on the assigned weight.

We choose Gentle AdaBoost in this study because it adopts a more stable rule of updating the weight, and use the logistic regression as the base learner. The procedure of how Gentle AdaBoost works has already been depicted in Fig. 2. To fit a regression equation $g(x)$ using weighted least-squares with sample weight D, the objective function is defined as

$$J = \sum_{i=1}^{N} D(i) \times \left(\frac{1+y_i}{2} - p(x_i)\right)^2 \tag{3}$$

where $p(x_i) = 1/1 + e^{-g(x_i)}$, $y_i = \{+1, -1\}$.

We use the gradient descent algorithm to optimize the regression equation by minimizing J. This gives the gradient for coefficients

$$\nabla J = \frac{\partial J}{\partial \vec{\beta}} \tag{4}$$

where $\partial J/\partial \beta_j = -2\sum_{i=1}^{N} D(i)\left((1+y_i)/2 - p(x_i)\right) \left(e^{-g(x_i)}/\left(1+e^{-g(x_i)}\right)^2\right)x_{i_j}$

Each round, the regression coefficient vector $\vec{\beta}$ is then updated by

$$\beta_{k+1} = \beta_k - \mu \nabla J_k \tag{5}$$

where μ is a constant learning rate and k is the step count in the optimization process.

The Gentle AdaBoost is forced to stop after 5 rounds. Nine (approximately 1‰) nonchurners with the highest weight are removed as noise. The weight on all of the training data is fitted to a normal distribution. All churned customers with a weight greater than the weight of 95% of the population are identified and defined as a group, which leaves the remaining churners to form the other group. This finally yields two groups of churners, with 220 customers and 478 customers, respectively. A logistic regression model is then trained to separate these two groups. Fig. 4 shows the performance of the separation, with the cutoff point for best sensitivity and specificity (blue circle in plot) $=0.7292$. This logistic regression model is then applied back to the training data and generates two customer clusters, where Cluster-1 contains 5486 records with 361 churners, a

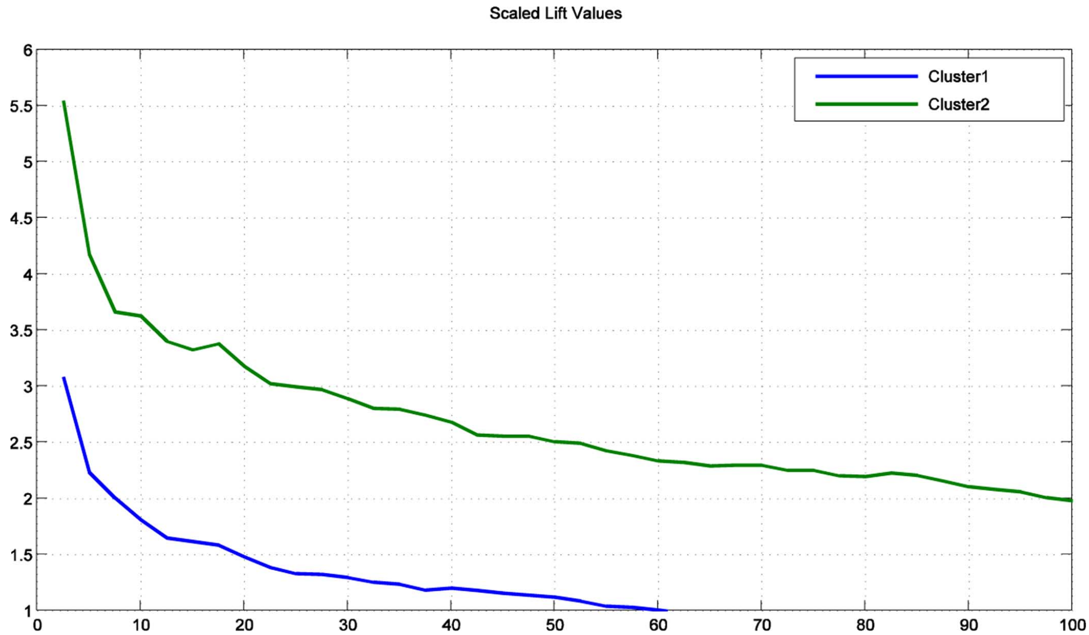

Fig. 7. Lift curves for predictions.

churn rate of 6.58%, and takes 76.4% of the training sample; and Cluster-2 contains 1695 records, with 318 churners, a churn rate of 18.76%, and takes 23.6% of the training sample.

As the training set is highly skewed (with the churners account for less than 10% of the data), separating churners based on their hardness of fitting a logistic regression model also gives a chance to form a high churn risk group. In our case, customers who belong to Cluster-2 are defined as high risk, and the average churn rate is about three times that of the average churn rate of customers in Cluster-1.

E. Model Generation

A logistic regression model is trained for each customer group to predict the likelihood that a customer will churn sometime in the future. Customers with a higher predicted likelihood have a higher propensity to churn. This likelihood can be stored on the customer's file and used in all kinds of retention efforts. Practically, this predictive model allows a scientific basis for managing business development efforts and therefore optimizes marketing costs. The results of the model will be used as a basis for generating lists and prioritizing contact customers and offers.

IV. Prediction Results

We evaluate the performance of our churn prediction model using a training set of customer information collected over a six-month period. Each customer is classified into one of two predefined groups and his/her churn propensity is monitored and updated according to his/her latest three-month information. In this way, we can simulate the real-world scenario of churn prediction.

A churn prediction system should be measured by its ability to identify churners for marketing purpose [18], and we therefore use the receiver operating characteristic (ROC) curve and top-quantile-lift values to give a comprehensive evaluation of our prediction model. We compare the results with a logistic regression model without customer separation.

Fig. 5 shows that, for customers who belong to the Cluster-1, the ROC curve of our churn prediction model (Curve 2) is located above but close to the curve of the logistic regression model (Curve 1), with the Area Under Curve (AUC) increased from 70.71 to 72.35. Although this improvement is not significant by Z-test $(z = 0.742, p = 0.28)$, our model achieves a much better result when the false positive rate is smaller than 5%. We will show why this improvement is still considered to be important later in this section.

Fig. 6 shows that, for customers who belong to the Cluster-2, the ROC curve of our churn prediction model (Curve 2) is located on the curve of logistic regression model (Curve 1), with the area under curve (AUC) increased from 58.74 to 64.08. This improvement is reported to be significant $(z = 2.08, p = 0.018)$ when $\alpha = 0.05$, which shows that our churn prediction model is able to better distinguish churners from nonchurners on Cluster-2.

The lift value reflects the increase in density with regard to the churn event relative to the density of churners in the customer base [19]. The higher the lift is, the better the predictive model is. For marketing purposes, where budgets are often limited and only a small fraction of customers can be targeted for retention actions, the top-quantile-lift is more important and of more practical value. As the churn rate of customers in Cluster-1 is much lower than the churn rate of customers in Cluster-2, only a small fraction of customers in Cluster-1 will be selected in order to identify as many churners as possible; thus the improvement that our prediction model has made for Cluster-1 is still of very important practical value.

We scaled the lift values based on the overall churn rate of the whole testing set to show how well our churn prediction model performs on each customer cluster. The results are given in Fig. 7. Note that although a better ROC curve is reported for predictions on Cluster-1 (Fig. 5) than on Cluster-2 (Fig. 6), the actually lift value of predictions on customers of Cluster-2 is much better than customers of Cluster-1 as a result of the difference in the base churn rates between the two customer clusters.

If we want to obtain lift value of the top-10%-percentile in real world application, we should take the top-3.3% from Cluster-1 and the top-31% from Cluster-2. That gives a top-10%-lift of all customers, catching 849 churners in 3259 customers, and giving a lift value of 2.83, which is improved by 0.42 if compared with a Logistic Regression model built on the whole training set, where only 728 churners in 3259 customers are caught, giving a lift value of 2.41.

V. Conclusion and Future Work

This research conducts an experimental investigation of customer churn prediction based on real-data sets. In contrast to most churn prediction models, our model allows for an "Implementation Zone" where customers with the highest churn propensity can be addressed for retention actions. Another difference is the way in which this research uses the boosting algorithm. Rather than trying to boost a base learner directly by a boosting algorithm, like most researchers did, this study tries to separate the training data based on the hardness of fitting a base learner, and builds a distinct prediction model for each defined cluster. The results are tested on a living data-stream and are compared with a logistic regression model fitted by the entire training set. Experimental evaluation shows that in customer churn data, which is highly skewed, the weight given by Gentle AdaBoost algorithm also suggests a good separation, and provides an opportunity to define a high risk customer group.

There is still extensive work to do from both a technique and business point of view. On one hand, to further improve performance, other classification methods as well as other techniques addressing class rarity should be used and compared; for example, will the performance be improved by a hybrid of different classifiers or by building a multiple boosted regression model using the sampling technique? On the other hand, accurate churn prediction only provides a basis for generating lists and prioritizing contact customers. Identifying the reason for a particular customer's churn behavior and providing what the customer really needs are also important for targeted marketing research.

Acknowledgment

The authors would like to thank the anonymous company for their efforts in providing us with their data as well as their expert insights.

References

[1] A. Payne and P. Frow, "A strategic framework for customer relationship management," *J. Marketing*, vol. 69, no. 4, pp. 167–176, Oct. 2005.

[2] L. D. Xu, "Enterprise systems: State-of-the-art and future trends," *IEEE Trans. Ind. Inf.*, vol. 7, no. 4, pp. 630–640, Nov. 2011.

[3] A. Berson, K. Thearling, and S. Smith, *Building Data Mining Applications for CRM*. New York: McGraw-Hill, 1999.

[4] K. Coussement and D. V. Poel, "Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques," *Expert Syst. Appl.*, vol. 34, no. 1, pp. 313–327, Jan. 2008.

[5] W. Verbeke, K. Dejaeger, D. Martens, J. Hur, and B. Baesens, "New insights into churn prediction in the telecommunication sector: A profit driven data mining approach," *Eur. J. Oper. Res.*, vol. 218, no. 1, pp. 211–229, Apr. 2012.

[6] W. J. Reinartz and V. Kumar, "The impact of customer relationship characteristics on profitable lifetime duration," *J. Marketing*, vol. 67, no. 1, pp. 77–99, Jan. 2003.

[7] P. Datta, B. Masand, D. R. Mani, and B. Li, "Automated cellular modeling and prediction on a large scale," *Artif. Intell. Rev.*, vol. 14, no. 6, pp. 485–502, Dec. 2000.

[8] D. Popović and B. D. Bašić, "Churn prediction model in retail banking using fuzzy C-means algorithm," *Informatica*, vol. 33, no. 2, pp. 235–239, May 2009.

[9] C.-P. Wei and I. T. Chiu, "Turning telecommunications call details to churn prediction: A data mining approach," *Expert Syst. Appl.*, vol. 23, no. 2, pp. 103–112, Aug. 2002.

[10] M. Owczarczuk, "Churn models for prepaid customers in the cellular telecommunication industry using large data marts," *Expert Syst. Appl.*, vol. 37, no. 6, pp. 4710–4712, Jun. 2010.

[11] J. Burez and D. V. Poel, "Handling class imbalance in customer churn prediction," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 4626–4636, Apr. 2009.

[12] N. Kim, K.-H. Jung, Y. S. Kim, and J. Lee, "Uniformly subsampled ensemble (USE) for churn management: Theory and implementation," *Expert Syst. Appl.*, vol. 39, no. 15, pp. 11839–11845, Nov. 2012.

[13] D. A. Kumar and V. Ravi, "Predicting credit card customer churn in banks using data mining," *Int. J. Data Anal. Tech. Strategies*, vol. 1, no. 1, pp. 4–28, Aug. 2008.

[14] J. Hadden, A. Tiwari, R. Roy, and D. Ruta, "Computer assisted customer churn management: State-of-the-art and future trends," *Comput. Oper. Res.*, vol. 34, no. 10, pp. 2902–2917, Oct. 2007.

[15] G. M. Weiss, "Mining with rarity: A unifying framework," *ACM SIGKDD Explorations Newslett.*, vol. 6, no. 1, pp. 7–19, Jun. 2004.

[16] N. Meinshausen, "Node harvest," *Ann. Appl. Stat.*, vol. 4, no. 4, pp. 2049–2072, Dec. 2010.

[17] J. Dutkowski and A. Gambin, "Research on consensus biomarker selection," *BMC Bioinformat.*, vol. 8, no. 5, pp. 1–13, May 2007.

[18] Y. Richter, E. Yom-Tov, and N. Slonim, "Predicting customer churn in mobile networks through analysis of social groups," in *Proc. SDM 10, Soc. Ind. Appl. Math.*, Philadelphia, PA, 2010, pp. 732–741.

[19] K. Coussement and D. V. Poel, "Integrating the voice of customers through call center emails into a decision support system for churn prediction," *Inf. Management*, vol. 45, no. 3, pp. 164–174, Apr. 2008.

[20] J. Shao, X. Li, and W. Liu, "The application of AdaBoost in customer churn prediction," in *Proc. IEEE Comput. Soc. Press*, Los Alamitos, CA, 2007, pp. 1–6.

[21] Y. Freund and R. E. Schapire, "A short introduction to boosting," *J. Jpn. Soc. Artif. Intell.*, vol. 14, no. 5, pp. 771–780, Sep. 1999.

[22] R. E. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," *Mach. Learning*, vol. 37, no. 3, pp. 297–336, Dec. 1999.

[23] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *Ann. Stat.*, vol. 28, no. 2, pp. 337–407, Apr. 2000.

[24] A. Ferreira and M. T. Figueiredo, "Boosting algorithms: A review of methods, theory, and applications," in *Ensemble Machine Learning*. New York: Springer, 2012, pp. 35–85.

[25] S. Hong and S. Weiss, "Advances in predictive data mining methods," in *Proc. MLDM*, London, U.K., 1999, pp. 13–20.

[26] M. Kafai and B. Bhanu, "Dynamic bayesian networks for vehicle classification in video," *IEEE Trans. Ind. Inf.*, vol. 8, no. 1, pp. 100–109, Jan. 2012.

[27] G. V. Kass, "An exploratory technique for investigating large quantities of categorical data," *Appl. Stat.*, vol. 29, no. 2, pp. 119–127, 1980.

Ning Lu received the B.S. degree in software engineering from Zhejiang University, Hangzhou, China, in 2006. He is currently working toward the Ph.D. degree from the University of Technology Sydney, Sydney, Australia. His dissertation explores how a case-based reasoning system is able to better communicate with users, and continuously tune according to feedback in concept drift environments.

His research interests include adaptive information system, concept drift, case-based reasoning, and maintenance. He is currently a Research Assistant with the University of Technology Sydney, Sydney, Australia.

Hua Lin received the Degree (with honors) in mathematics from Murdoch University, Perth, Australia, in 1990, and the M.S. degree in mathematics and statistics from Curtin University, Perth, Australia, in 1999.

She was a Tutor with the School of Mathematics and Statistics, Curtin University. She worked with ING, Bancorp Advantage Market Solutions, and Commonwealth Bank. She is currently a Marketing Customer Insights Manager with SingTel Optus. She has many years working experience in marketing research.

Jie Lu (M'06) received the Ph.D. degree from Curtin University, Perth, Australia, in 2000.

She is Head of the School of Software, University of Technology Sydney, Sydney, Australia. Her research interests lie in the area of decision support systems and e-Service intelligence. She has published five research books and 270 papers and won five Australian Research Council discovery grants and ten other grants. She serves as Editor-In-Chief for *Knowledge-Based Systems*, Editor-In-Chief for *International Journal of Computational Intelligence Systems*, and editor for a book series on Intelligent Information Systems and has delivered six keynote speeches at international conferences.

Prof. Lu was the recipient of a University Research Excellent Medal in 2010.

Guangquan Zhang received the Ph.D. degree in applied mathematics from Curtin University of Technology, Perth, Australia.

He is an Associate Professor with the Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, Australia. From 1979 to 1997, he was a Lecturer, Associate Professor, and Professor with the Department of Mathematics, Hebei University, China. His main research interests lie in the area of multi-objective, bilevel and group decision making, decision support system tools, fuzzy measure, fuzzy optimization, and uncertain information processing. He has published four monographs, four reference books and over 250 papers including more than 140 refereed journal articles. He has won four Australian Research Council (ARC) discovery grants and many other research grants. He has served, and continues to serve, as a guest editor of special issues for four international journals.