Cognitive Computation (2019) 11:262–270
https://doi.org/10.1007/s12559-018-9608-3

# Large-scale Ensemble Model for Customer Churn Prediction in Search Ads

Qiu-Feng Wang[1] · Mirror Xu[2] · Amir Hussain[3]

Received: 21 August 2018 / Accepted: 23 October 2018 / Published online: 13 November 2018

## Abstract

Customer churn prediction is one of the most important issues in search ads business management, which is a multi-billion market. The aim of churn prediction is to detect customers with a high propensity to leave the ads platform, then to do analysis and increase efforts for retaining them ahead of time. Ensemble model combines multiple weak models to obtain better predictive performance, which is inspired by human cognitive system and is widely used in various applications of machine learning. In this paper, we investigate how the ensemble model of gradient boosting decision tree (GBDT) to predict whether a customer will be a churner in the foreseeable future based on its activities in the search ads. We extract two types of features for the GBDT: dynamic features and static features. For dynamic features, we consider a sequence of customers' activities (e.g., impressions, clicks) during a long period. For static features, we consider the information of customers setting (e.g., creation time, customer type). We evaluated the prediction performance in a large-scale customer data set from Bing Ads platform, and the results show that the static and dynamic features are complementary, and get the AUC (area under the curve of ROC) value 0.8410 on the test set by combining all features. The proposed model is useful to predict those customers who will be churner in the near future on the ads platform, and it has been successfully daily run on the Bing Ads platform.

**Keywords** Churn prediction · Ensemble model · Machine learning · Search ads · Static features · Dynamic features

## Introduction

Customer churn is one of the mounting issues of today's rapidly growing and competitive search advertising market, which is a multi-billion market per year. Search advertising is a three-player game, including advertisers, search users, and the company of the ads publisher, where advertisers participate in auctions to show their advertisements to users who come to search engine (e.g., google.com or bing.com) to search information, and the company of the ads publisher (e.g., Google or Microsoft) will charge those advertisers if their ads are clicked by search users [1]. The overall process of the search advertising is like this: if one ad of the advertiser wins in the auction, it will get an impression shown on the search engine, and if it is clicked by the search user, the ad will get a click and the advertiser will pay it (i.e., spending); furthermore, if the product of the ad is bought by the search user, the ad gets a conversion. Usually, these metrics (i.e., impressions, clicks, spending, and conversions) are related positively, and the aim of the advertisers is to get higher conversions. In the whole search advertising market, there are millions of advertisers, and most of them are small business with a little budget on the search ads. It is difficult for all of advertisers to understand the complex logic of search ads, so ads publisher companies provide an ads platform to help advertisers to manage

Qiu-Feng Wang is currently with XJTLU, but carried out the most of work described here while being affiliated with Microsoft.

✉ Qiu-Feng Wang
qiufeng.wang@xjtlu.edu.cn

Mirror Xu
mirrorxu@microsoft.com

Amir Hussain
a.hussain@napier.ac.uk

[1] Xi'an Jiaotong-Liverpool University (XJTLU), NO. 111 Renai Road, Suzhou 215123, People's Republic of China

[2] Microsoft Corporation, No. 5 Danling Street, Beijing 100080, People's Republic of China

[3] Edinburgh Napier University, 10 Colinton Road, Edinburgh EH105DT, Scotland

their ads business, e.g., AdWords on Google[1] or BingAds on Bing.[2] In this context, the customers of ads publisher companies are those advertisers on the ads platform, so the terms "advertiser" and "customer" are used interchangeably in this paper unless otherwise mentioned.

Customer churners are those advertisers who will leave the ads platform, which means there will be no performance on the ads platform. Figure 1 shows two customers with the performances on the platform of Bing Ads for about 1 year. Considering the seasonal performance, we define the customers as churners who have no performance for at least 3 months. So, the object of this work is to predict the customers whether they will leave or not based on their historical performance, and give some suggestions to help them get more conversions if they will be nonvoluntary to leave. This work will benefit all of the three players: customers sell their products to those search users who are really interested and ads of the publishers achieve the charge from those customers.

To better understand a churn prediction model, it is necessary to know some reasons of the customer churn. The reasons of customer to be churner are various and mostly divided into two categories, corresponding to two churn types: voluntary and nonvoluntary churn, respectively [2, 3]. Sometimes customers are forced into dropping their service with the ads' platform due to the customers themselves. This is known as voluntary churn. Some examples include customers reduce budget on the advertisements plan due to the financial status. Nonvoluntary churn happens when customers do not get desirable effect from their advertisements, for example, their ads do not get enough impressions, clicks, or conversions. It is difficult to predict those voluntary churn due to its nature: it is the customer's own decision to churn [4], who usually have no uncommon performance on the ads' platform until they suddenly stop the service. For nonvoluntary churn customers, we can carefully design a model to predict based on the analysis of their historical activities in the ads' platform using machine-learning techniques.

In the competitive marketplace, it is the fact that the cost for customer acquisition is much greater than the cost of customer retention.[3] Therefore, with the purpose of retaining customers, academics as well as practitioners find it crucial to build a customer churn prediction model that is both accurate and comprehensible, in order to identify respectively the customers who are about to churn and their reasons to do so, which are also essential business intelligence applications [5]. Over the last decade, there has been increasing interest for churn prediction in various fields, including telecommunication industry [5–10], banking [11–14], insurance [15, 16], social networks [17], and online games [18–21]; yet, there are not many research papers for churn prediction in search ads. To the best of our knowledge, only one paper [4] introduced the churn prediction model in search ads, which applied simple tree-based ensemble algorithms for Google AdWords.

Ensemble learning is based on multiple learning models, which are strategically generated, and optimally combined for prediction problems. The idea behind this is inspired by human cognitive system: two minds are better than one [22]. During the past decade, ensemble models have been well developed and widely applied in various machine-learning applications [23–27] and cognitive science research [28–30]. Among various ensemble models, the gradient-boosting decision tree (GBDT) [31] is one of the most popular models due to its theory analysis and efficient optimization.

The contributions of our work are twofold: scientific and business. From scientific perspective, the main contributions are the following: (1) we introduce a GBDT-based churn prediction model in search ads, which is efficient, high accuracy and explainable. (2) To improve the prediction accuracy further, we carefully consider two different types of features: static and dynamic features. Static features are based on the information that customers provided when they created their accounts in search ads' platform, which show the properties of the customers, e.g., big customers or small customers. Dynamic features are based on the activities of the customers during the historical time period, which show the customers' historical performance and change ratio. Both features are very useful to build the churn prediction models.

From a business perspective, the main contributions include two points: (1) we propose an early prediction model, which means our model can identify customers whether to be churner much time ahead of their beginning of inactive status. This is necessary in the real business products, because the customers do not temporarily pause their accounts but continue to be inactive for a long-time period [4], and it is too late to retain the customers if those customers begin to be inactive. (2) According to the prediction results, we also give the analysis for churn preservation. This also makes us use GBDT algorithm instead of the deep learning model, which is usually a black box[4] to lack enough explanation for the prediction results, although it has powerful prediction ability [32].

The remainder of the paper is structured as follows. The next section provides a brief literature review on churn prediction model. Then, "Proposed Method" describes

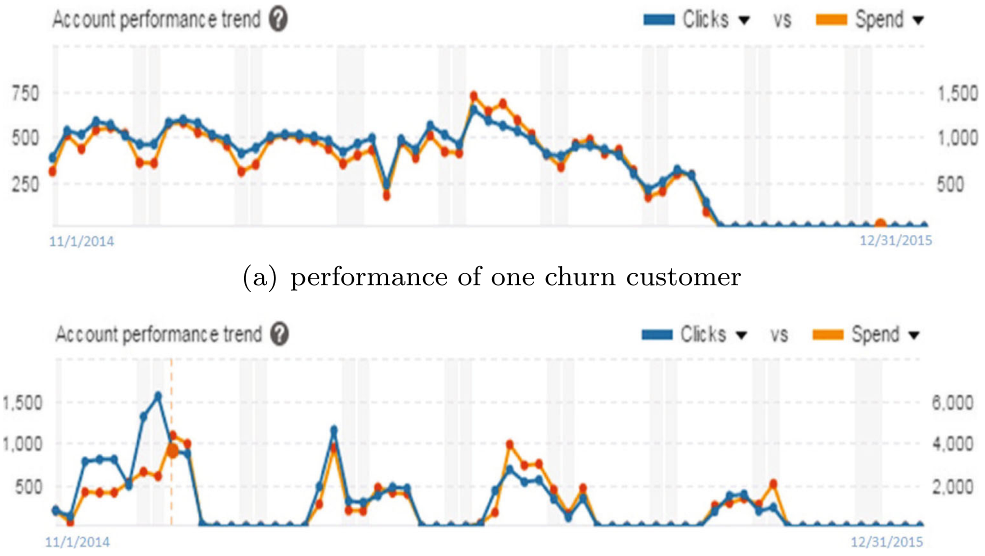
[1] https://adwords.google.com/home/

[2] https://bingads.microsoft.com/

[3] https://www.invespcro.com/blog/customer-acquisition-retention/

[4] https://www.quora.com/Why-do-many-research-studies-claim-that-deep-learning-is-a-black-box

**Fig. 1** Examples of the click and spend performance with two customers



(a) performance of one churn customer

(b) performance of one seasonal customer (non-churner)

our proposed system in details, including the churn definition, feature engineering, ensemble modeling, and churn preservation. In "Experimental Results," we introduce the evaluation setup and experimental results. Finally, we draw our conclusions in "Conclusions."

## Related Works

Churn prediction is a common task in the industry, even there are few works about churn prediction in search ads [4], many related works have been investigated in other areas, such as telecommunication industry [5–10], banking [11–14], insurance [15, 16], social networks [17], and online games [18–21]. In this section, we discuss related works according to the steps of a typical pattern classification system, which includes feature engineering and statistical classifier.

### Feature Engineering

In a typical pattern classification system, the suitable feature engineering (feature extraction) of the data representation plays an important role [33]. Before the popularity of deep neural network (end-to-end model), most systems need handcrafted feature engineering, which is time-consuming and very dependent on the specific tasks. The work [21] extracted features by three categories, including normal features, monetization features, and gameplay style features for the churn prediction in mobile social games. The sentiment-based and emotion-based features (expressed in customer emails) were adopted in the work [34], and domain knowledge was incorporated in the work [35]. The work [36] created the features from those key performance indicators (KPIs) in the prepaid mobile markets. The work [4] considered two types of features for churn prediction in Google AdWords, including static (related to the customers) and time-varying (related to the customers' activity on AdWords during some periods).

As the feature engineering work is usually time-consuming, recently, many researchers focus how to extract features from the original data directly. The work [37] proposed an automatical feature extraction method based on stacked auto-encoder, then a linear regression classifier was applied for Telecom churn prediction. The work [38] adopted the neural embedding techniques to represent the customer and combine the original handcrafted features in the prediction model of the online fashion retailer. Furthermore, the popular deep learning model combines the feature engineering and classifier design, which will be introduced in the following.

### Statistical Classifier

Once the data are represented as a suitable formation by the feature engineering, one statistical classifier is used to separate the data into different classes. Churn prediction is most naturally formalized as a binary classification problem, classifying customers into two classes: those who are likely to churn (positive class) and those who are likely continue playing in the ads' platform (negative class). So numerous classifiers from the research fields of machine learning have been adopted for churn prediction [5, 10], among which, the linear regression (logistic regression) is a usual starting point as it represents a simple robust linear model [4, 13], which uses a weighted linear combination of features to output the probability of instances belonging to the positive or negative class [21].

To improve the customer prediction accuracy, some advanced classifiers have been adopted, for example, decision trees (C4.5) [4, 12, 19], kernel-based classifiers (support vector machines: SVM) [39, 40], artificial neural networks [41, 42], and k-nearest neighbors [43]. All of these works implement the prediction model as a single model. Researchers have also explored the use of so-called ensemble-learning techniques, for example, boosting and bagging models were used in the work [44], and the work [45] combined SVM with Naive Bayes trees on analyzing bank credit card churn prediction. Although various statistical classifiers have been applied in the churn prediction, there appears no single dominating method which is useful in all churn-related context [5, 10].

The class imbalance is a significant challenge in learning of the statistical classifiers for customer churn prediction, where the number of churners is significantly outnumbered by the non-churners. In order to solve this issue, many solutions have been proposed in the machine learning area, which can be found in these works [46–49]. In the churn prediction, the work [50] proposed a balanced random forests by combing the weighted forests, and the work [51] comprehensively compared the sampling techniques for imbalance learning in churn prediction.

### End-to-End Learning Model

Compared to the typical machine-learning system with separated stages of feature engineering and classifier learning, the popular deep neural network (DNN) can learn the model directly from the data, which is an end-to-end learning model [32]. The DNN has demonstrated superior performance in many tasks, including speech, vision, and natural language processing, and the most successful DNNs are deep convolutional neural networks (CNN) and recurrent neural networks (RNN). In the churn prediction, the work [52] represented each customer as an image based on their usage behavior (columns) over a period (rows), then a deep CNN was successfully used in the telecommunications. The work [53] applied the RNN with reinforcement learning for churn prediction in mobile phone users. The work [54] described the details about how to build a DNN in customer churn prediction, which is abstract and company independent.

The DNNs in churn prediction are not as popular as those in other domains. Although DNNs can get higher accuracy, they are usually not explainable, which is very important in business intelligence (BI), like the churn prediction. Another reason is that the DNNs sometimes are not robust, this is not allowed in a stable BI system. In this research, we carefully design features and use the GBDT-based ensemble-learning method to predict churners.

## Proposed Method

In this section, we firstly introduce the churn definition in our research, then give the feature description and ensemble model classifier, and finally, we introduce some analysis on churn preservation. Due to the confidentiality of the company, we can not provide all implemented details used in our work.

### Churn Definition

To better understand the churn prediction system, we first explain our definition of customer churn in search ads. Customers can churn at any time and sometimes show seasonal behavior (see Fig. 1); therefore, we need carefully consider to define customer churn to avoid incorrect identification of churned customers, and give early prediction to have enough time to preserve these churners. To this end, a long period of about 10 months was taken into account for the customer activity analysis, including half-year observation for feature extraction, 1 month for the window of early prediction, and another 3 months to determinate the labels of customers. This is best understood by viewing Fig. 2 from left to right.

We run the churn prediction system on "Today" shown in Fig. 2 to predict the customer whether to become churner or not in the next 1 month (the "1 month window" in Fig. 2). This window gives the ads' platform enough time to do analysis and increase efforts for retaining the potential churners. However, if the window is too long, the churn prediction accuracy will be decreased due to more uncertainty, so we use one month in our system. This machine-learning system dependents on the features extracted from the previous half year (180 days), such as the customer activities and monetization performances (see, "Feature Description"). The customer is identified as a churner if the one has no clicks (non-performing[5]) in consecutive 13 weeks (90 days). The time window between "Today" and the day of the last click happened before churn is the early prediction window, which is at most one month (30 days, the "1 month window" in Fig. 2). To remove plenty of zombie customers, who are always non-performing, we only consider the customers who have clicks in the last week ("P-week 1" shown in Fig. 2).

[5] Because CPC strategy is widely used in the search ads, the metric of click is usually used to show the performance of the advertisers.

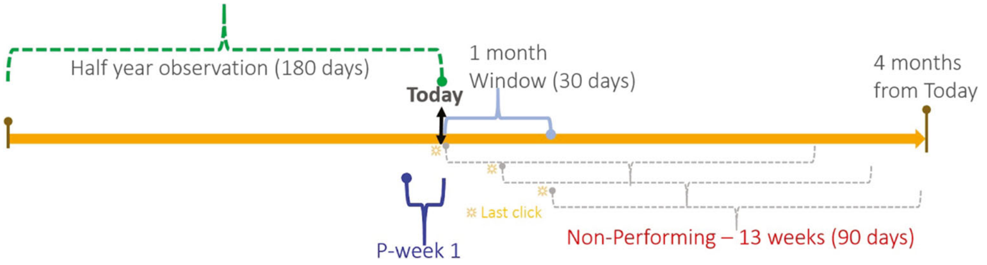

Half year observation (180 days)
Today
1 month
Window (30 days)
4 months
from Today
Last click
P-week 1
Non-Performing – 13 weeks (90 days)

**Fig. 2** The time windows for churn definition in search ads

## Feature Description

To model the customer churner, we need to extract useful features to represent the customer. In this research, we extract two types of features, including static and dynamic features [4]. The static features are based on the information that customers provided when they created their accounts in search ads' platform, such as creation time, customer type, coupon information, and budget setting. In our experiments, we used 23 static features. Table 1 gives more details about the static features.

The dynamic features are based on the customers' activities within the period of the previous half year (see, Fig. 2), such as the monetization activities of impressions, clicks, cost and conversions, and the number of active sub-objects of accounts, campaigns, ad group, order items.[6] All of these activities can be found in the Table 1 for detailed description. Each activity is calculated daily during the period, which formats a sequence of activity data. From this sequence data, we extract some statistical values, e.g., mean value and variance, the change ratio features, e.g. slope ratio, and the activities in the last days (e.g., the change ratio in last week) [36]. Moreover, we transform this sequence data via Fourier transformation, and use some representative coefficients in the transformation space. Finally, we extracted 35 features for each sequence data, and we used 25 activities in our experiments, so totally we have 875 dynamic features. Further details about the sequence transformation and extraction are ignored due to the confidentiality policy.

## Ensemble Model Classifier

Ensemble models are widely used in the data mining, and almost half of data mining competitions are won via some variants of tree ensemble methods [55], which can learn higher order relationship between features. These tree ensemble models can be scalable, and they are popular in the industry.

The gradient boosting is a high-performance technique in ensemble models, which produces a prediction model in the form of an ensemble of weak prediction models. The basic idea is to construct additive regression models by sequentially fitting a simple parameterised function (weak prediction model) to current "pseudo"-residuals, which can be simulated by the gradient of the loss functional being minimized at each iteration [31]. This procedure can be described briefly in the following Algorithm [56], where the input is a training set $\{(x_i, y_i), i = 1, ..., n\}$, the loss function is $L(y, F(x))$, and the number of iterations is $M$.

**Algorithm 1** Gradient Boosting

```
Start with an initial model, say $F_0(x) = \arg\min_{\gamma} \sum_{i=1}^{n} L(y_i, \gamma)$.
For m=1 to M do:
    Calculate so-called "pseudo"-residuals, say $r_{mi} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}$;
    Train a weak learner $h_m(x)$ using the training set $\{(x_i, r_{mi}), i = 1, ..., n\}$;
    Calculate the coefficient $\gamma_m$, say $\gamma_m = \arg\min_{\gamma} \sum_{i=1}^{n} L(y_i, F_{m_1}(x_i) + \gamma h_m(x_i))$;
    Update the model, say $F_m(x) = F_{m-1}(x) + \gamma_m h_m(x)$.
Output $F_M(x)$.
```

The decision trees are typically used as the weak prediction models in the gradient boosting, which is usually abbreviated to GBDT (i.e., gradient-boosting decision trees). A decision tree is a binary tree-like flow chart, where at every interior node, one decides which of the two-child nodes to continue to, based on the value of one of the input features.

The ensemble of trees is produced by computing, in each step, a regression tree that approximates the gradient of the loss function, and adding it to the previous tree with coefficients that minimize the loss of the new tree (see, the Algorithm in above). The output of the GBDT on a given instance is the sum of the tree outputs. For a binary classification problem of churn prediction, the output is converted to the probability by using some form of calibration, like the sigmoid transformation. The more information can be found in the work [31] or the wikipedia entry for boosted trees, the section about gradient tree boosting [56].

[6] Each advertiser creates the accounts in search ads with tree structures, including accounts, campaigns, ad group, and order items [1].

**Table 1** Some representative features in the proposed churn prediction model

| Features | Description | Type |
|---|---|---|
| Creation time | The date when the customer created the accounts | Static |
| Customer type | The type of this customer, e.g., industry category | sStatic |
| Location | The location of the customer | Static |
| Coupon class | The class of coupon used | Static |
| Budget | The budget of customer set on all campaigns | Static |
| Impressions | The number of ads shown on ads' platform | Dynamic |
| Clicks | The number of ads clicked by the users | Dynamic |
| Cost | The money charged by the ads' platform | Dynamic |
| Conversions | The number of users' action valuable to the business of ads | Dynamic |
| #active accounts | The number of active accounts | Dynamic |
| #active campaigns | The number of active campaigns | Dynamic |
| #active ad groups | The number of active ad groups | Dynamic |
| #active order items | The number of active order items | Dynamic |

## Churn Preservation

The final aim of churn prediction system is to preserve the potential churners. Our system can predict customers to become churners about one month early. This gives enough time to help analyze the activities of potential churners and give corresponding suggestions to them.

Figure 3 shows the diagram of our churn preservation system. Firstly, the features of each customer are input into our churn prediction model and get the probability to be churner. Once this probability is larger than a threshold, we think this customer is a potential churner, and we will analysis the information of the customer by using the churn prediction model analysis. The GBDT model gives the importance of all features, which can be used for the reason analysis with the help of the diagnosis tool provided by the ads' platform. Then, we can get some reasons to get the corresponding suggestions from the opportunity tool also provided by ads' platform [1], like adding keywords and modifying bids. All of these opportunity suggestions can help the customers get more performance on the ads' platform. Both diagnosis and opportunity tools are public and free for the customers in the ads' platform. Lastly, we will notice these potential churners by emails or call those large-spending churners directly.

## Experimental Results

To evaluate the performance of our approach, we randomly selected customers from Bing Ads during the period of 2015/04/08 to 2016/02/01 (300 days). As in the Fig. 2, the day of "Today" is 2015/10/04, and the period of 2015/04/08 to 2015/10/04 is for the feature extraction, which is introduced in "Feature Description," and there are 898 features totally in our experiments. All of the selected customers have at least one click during the period of 2015/9/28 to 2015/10/04 ("P-Week1"). The period of 2015/10/05 to 2016/02/01 is used to label the customers: if the customer has no clicks in consecutive at least 90 days with the end date of 2016/02/01, then the customer is labeled as a churner (positive class); otherwise, the customer is labeled as a non-churner (negative class). To avoid the class imbalance issue, we randomly select the negative

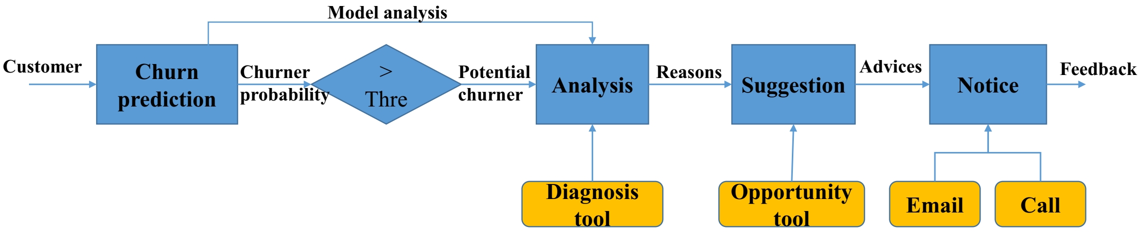


**Fig. 3** The diagram of churn preservation

**Table 2** The number of customers in our data set

| Dataset | Positive | Negative | All |
|---|---|---|---|
| Training | 19342 | 33568 | 52910 |
| Test | 4846 | 8303 | 13149 |
| All | 24188 | 41871 | 66059 |

**Table 3** The AUC of different churn prediction setting

| Setting | GBDT | GBDT-dynamic | GBDT-static | LR |
|---|---|---|---|---|
| AUC | 0.8410 | 0.8236 | 0.8215 | 0.8159 |

samples to make the ratio of negative samples number to positive samples number not too large. Finally, we get 66059 customers for the evaluation, including 52910 customers as training data, and 13149 customers as test data, with the ratio about 4:1. In these customers, there are 24188 and 41871 customers, as churner (positive class) and non-churner (negative class), respectively. Table 2 shows the details of our evaluate data set.

The receiver operating characteristic (ROC) curve is a plot that is widely used to represent the performance of probabilistic or a threshold based classifier turned into a binary classifier for varying discrimination thresholds. The ROC curve is defined by plotting the true positive rate (TPR) against false positive rate (FPR) at different discrimination thresholds. The TPR and FPR are defined as follows:

$$\mathrm{TPR} = \frac{\text{churners correctly classified}}{\text{total churners}}$$

$$\mathrm{FPR} = \frac{\text{non-churners incorrectly classified}}{\text{total non-churners}}.$$

The area under the ROC curve is also commonly used to compare the performance of different classifiers, which is usually abbreviated to AUC (area under curve). AUC represents the likelihood that the probability assigned to a randomly selected churner is higher than the probability assigned to a randomly selected non-churner, so the AUC is larger then the performance of this classifier is better.

Figure 4 shows the ROC of the churn prediction performance on the four classification setting. As can be seen from Fig. 4, the performance of GBDT only with dynamic features (note as GBDT-dynamic with the green curve) and only with static features (note as GBDT-static with the blue curve) are comparable, and combining all features (note as GBDT with the red curve) improves the performance a lot, which is much better than logistic regression with all features (note as LR with magenta curve). The Table 3 shows all AUC value for these classification setting, which is consistent with the founding from Fig. 4.

Finally, this system has been successfully run on the Bing Ads platform, and we choose different thresholds to satisfy the various requirements, e.g., the lower threshold for large spending customer to get higher recall, while higher threshold for small spending customer to get higher precision.

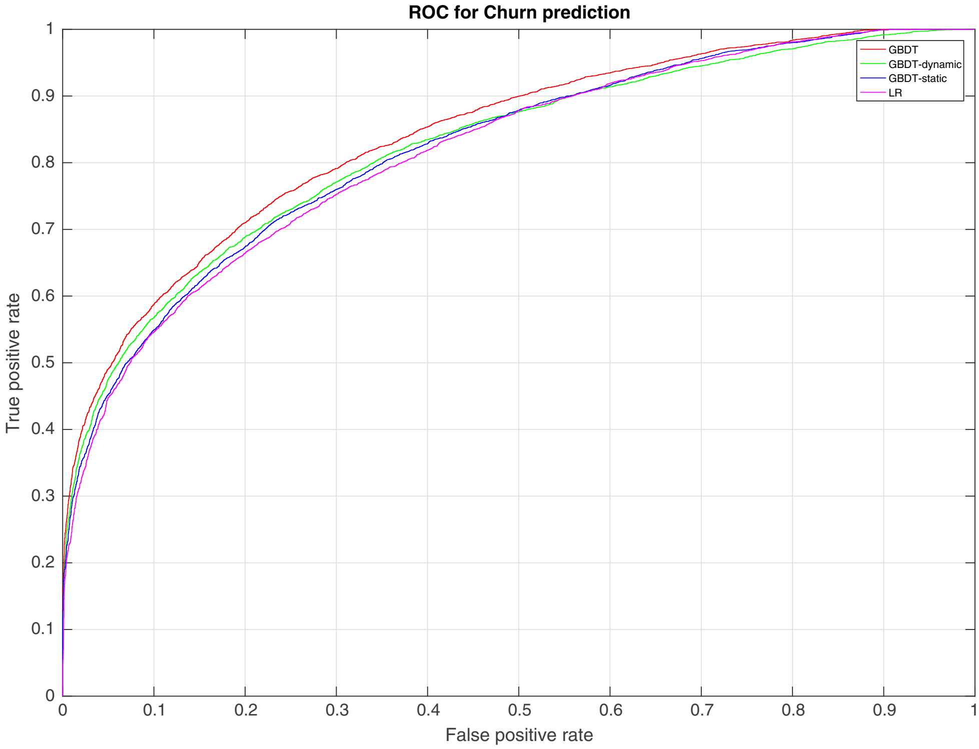


**Fig. 4** ROC curve for different churn prediction setting

## Conclusions

Churn prediction is an important tool for the ads' platform to stay competitive in a rapidly growing search advertising market. In this paper, we propose a large-scale ensemble model of gradient-boosting decision tree (GBDT) for customer churn prediction in search ads. In the feature engineering, we carefully consider two types of features, including static features of customer original setting and dynamic features of the customer historical activities, where the Fourier transformation is also used to extract the dynamic features. To help the ads' platform has enough time to preserve the potential churners, we propose the churn prediction model to identify customers whether to be churner plenty of time in advance (1 month window), and a pipeline of churn preservation is also given. We also evaluate the proposed model on a large-scale data set of randomly selected customers from Bing Ads platform, and get the AUC value 0.8410 by combing both static and dynamic features. Finally, this model has successfully daily run on the Bing Ads platform to predict and preserve the potential churned advertisers.

Recently, deep learning models have shown the powerful prediction ability in many other domains, especially the CNN- and RNN-based models. However, they are not so popular for churn prediction in search ads due to the limit of the explainability and stability. Many works have been proposed to overcome these limits, and we will try to apply them in the research of churn prediction in future, especially the RNN-based model, which is suitable for the time-series problem. Obviously, our churn prediction is a sequence problem. Moreover, the churn preservation system will be deeply investigated to combine the churn prediction model, although we have proposed a simply churn preservation pipeline.

**Acknowledgements** We also would like to thank all of the members in Bing Ads Adinsight team and PM team at Microsoft for their discussion and help on this work.

**Funding Information** This study was funded by Natural Science Foundation of the Jiangsu Higher Education Institutions of China under no. 17KJB520041 and 17KJD520010; Natural Science Foundation of Jiangsu Province BK20181189 and BK20181190; Open Project Fund of the National Laboratory of Pattern Recognition 201800020, Key Program Special Fund in XJTLU under no. KSF-A-10, KSF-A-01 and KSF-P-02; and XJTLU Research Development Fund RDF-16-02-49. In addition, A. Hussain was supported by the UK Engineering and Physical Sciences Research Council (EPSRC) grant (AV-COGHEAR, grant reference number: EP/M026981/1).

## Compliance with Ethical Standards

**Conflict of interest** The authors declare that they have no conflict of interest.

**Ethical Approval** This article does not contain any studies with human participants performed by any of the authors.

**Informed Consent** Informed consent was obtained from all individual participants included in the study.

## References

1. Wang Q, Huang K, Li S, Yu W. Adaptive modeling for large-scale advertisers optimization. BMC Big Data Analytics. 2017;2:8.
2. Kim HS, Yoon CH. Determinants of subscriber churn and customer loyalty in the Korean mobile telephony market. Telecommun Policy. 2004;28(9-10):751–65.
3. Hadden J, Tiwari A, Roy R, Ruta D. Computer assisted customer churn management: state-of-the-art and future trends. Comput Oper Res. 2007;v34(10):2902–17.
4. Yoon S, Koehler J, Ghobarah A. Prediction of advertiser churn for google adwords jsm proceedings. 2010.
5. Vafeiadis T, Diamantaras KI, Sarigiannidis G, et al. A comparison of machine learning techniques for customer churn prediction. Simul Model Pract Theory. 2015;55:1–9.
6. Kraljević G, Gotovac S. Modeling data mining applications for prediction of prepaid churn in telecommunication services. Automatika. 2010;51(3):275–83.
7. Jadhav RJ, Pawar UT. Churn prediction in telecommunication using data mining technology. Int J Adv Comput Sci Appl. 2011;2(2):17–9. https://doi.org/10.14569/IJACSA.2011.020204.
8. Kim K, Jun CH, Lee J. Improved churn prediction in telecommunication industry by analyzing a large network. Expert Syst Appl. 2014;41(15):6575–84.
9. Qureshi SA, Rehman AS, Qamar AM, et al. Telecommunication subscribersćhurn prediction model using machine learning, 8th International Conference on Digital Information Management. IEEE. pp. 131–136. 2014.
10. Amin A, Anwar S, Adnan A, Nawaz M, Alawfi K, Hussain A, Huang K. Customer churn prediction in the telecommunication sector using a rough set approach. Neurocomputing. 2017;237:242–54.
11. Xie Y, Xiu L. Churn prediction with linear discriminant boosting algorithm. IEEE International Conference on Machine Learning and Cybernetics, pp. 228–233. 2008.
12. Glady N, Baesens B, Croux C. Modeling churn using customer lifetime value. Eur J Oper Res. 2009;197(1):402–11.
13. Nie G, Wei R, Zhang L, et al. Credit card churn forecasting by logistic regression and decision tree. Expert Syst Appl An International Journal. 2011;38(12):15273–85.
14. Ali ÖG, Aritürk U. Dynamic churn prediction framework with more effective use of rare event data: the case of private banking. Expert Syst Appl. 2014;41(17):7889–903.
15. Risselada H, Verhoef PC, Bijmolt THA. Staying power of churn prediction models. J Interact Mark. 2010;24(3):198–208.
16. Günther C-C, Tvete IF, Aas K, et al. Modelling and predicting customer churn from an insurance company. Scand Actuar J. 2014;1:58–71.
17. Ngonmang B, Viennet E, Tchuente M. Churn prediction in a real online social network using local community analysis. In: Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining; 2012. p. 282–288.
18. Borbora ZH, Srivastava J. User behavior modelling approach for churn prediction in online games. In: 2012 international conference on privacy, security, risk and trust, PASSAT 2012, and 2012 international conference on social computing, SocialCom 2012, Amsterdam, Netherlands; 2012. p. 51–60.
19. Runge J, Gao P, Garcin F, et al. Churn prediction for high-value players in casual social games. In: 2014 IEEE conference on Computational Intelligence and Games; 2014. p. 1–8.

20. Castro EG, Tsuzuki MSG. Churn prediction in online games using playersĺogin records: a frequency analysis approach. IEEE Transactions on Computational Intelligence and Ai in Games. 2015;7(3):255–65.
21. Milošević M, Živić N, Andjelković I. Early churn prediction with personalized targeting in mobile social games. Expert Syst Appl. 2017;83:326–32.
22. Gudivada VN, Irfan MT, Fathi E, Rao DL. Cognitive analytics : going beyond big data analytics and machine learning. Handbook of Statistics. 2016;35:169–205.
23. Wang Q-F, Cambria E, Liu C-L, Hussain A. Common sense knowledge for handwritten chinese text recognition. Cogn Comput. 2013;5(2):234–42.
24. Yin X-C, Huang K, Hao H-W. DE2: dynamic ensemble of ensembles for learning nonstationary data. Neurocomputing. 2015;165:14–22.
25. Saliha M, Swindle AH. From spin to identifying falsification in financial text. Cogn Comput. 2016;8(4):729–45.
26. Ortín S, Pesquera L. Reservoir computing with an ensemble of time-delay reservoirs. Cogn Comput. 2017;9(3):327–36.
27. Wen GH, Hou Z, Li HH, Li DY, Jiang LJ, Xun EY. Ensemble of deep neural networks with probability-based fusion for facial expression recognition. Cogn Comput. 2017;9:597–610.
28. Ayerdi B, Savio A, Graña M. Meta-ensembles of classifiers for Alzheimerś disease detection using independent ROI features. Natural and Artificial Computation in Engineering and Medical Applications. Springer; 2013. pp. 122–130.
29. Gu Q, Ding YS, Zhang TL. An ensemble classifier based prediction of G-protein-coupled receptor classes in low homology. Neurocomputing. 2015;154:110–18.
30. Mogultay H, Vural F T Y. Cognitive learner: an ensemble learning architecture for cognitive state classification. In: IEEE 25th Signal Processing and Communications Applications Conference; 2017. p. 1–4.
31. Friedman JH. Greedy function approximation: a gradient boosting machine. Ann Stat. 2001;29(5):1189–232.
32. Goodfellow Ian, Bengio Yoshua, Courville A. Deep Learning. Cambridge: MIT Press; 2016.
33. Duda RO, Hart PE, Stork DG. Pattern classification, 2nd ed. New York: Wiley; 2001.
34. Coussement K, Van den Poel D. Integrating the voice of customers through call center emails into a decision support system for churn prediction. Information & Management. 2008;45(3):164–74.
35. Lima E, Mues C, Baesens B. Domain knowledge integration in data mining using decision tables: case studies in churn prediction. J Oper Res Soc. 2009;8(8):1096–106.
36. Meher AK, Wilson J, Prashanth R. Towards a large scale practical churn model for prepaid mobile markets. Advances in Data Mining Applications and Theoretical Aspects, pp. 93–106. 2017.
37. Li R, Wang P, Chen Z. A feature extraction method based on stacked auto-encoder for telecom churn prediction. In: Zhang L, Song X, and Wu Y, editors. Theory, Methodology, Tools and Applications for Modeling and Simulation of Complex Systems. AsiaSim 2016, SCS AutumnSim. Communications in Computer and Information Science. Singapore: Springer; 2016.
38. Chamberlain BP, Cardoso A, Liu CHB, et al. Customer lifetime value prediction using embeddings. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; 2017. p. 1753–1762.
39. Coussement K, Van den Poel D. Churn prediction in subscription services: an application of support vector machines while comparing two parameter-selection techniques. Expert Syst Appl. 2008;34(1):313–27.
40. Gordini N, Veglio V. Customers churn prediction and marketing retention strategies. An application of support vector machines based on the AUC parameter-selection technique in B2B e-commerce industry. Ind Mark Manag. 2016;62:100–7.
41. Huang Y, Kechadi T. An effective hybrid learning system for telecommunication churn prediction. Expert Syst Appl. 2013;40(14):5635–47.
42. Hadiji F, Sifa R, Drachen A, et al. Predicting player churn in the wild. In: IEEE conference on Computational intelligence and games (CIG). IEEE; 2014. p. 1–8.
43. Keramati A, Jafari-Marandi R, Aliannejadi M, et al. Improved churn prediction in telecommunication industry using data mining techniques. Appl Soft Comput. 2014;24:994–1012.
44. Lemmens A, Croux C. Bagging and boosting classification trees to predict churn. J Mark Res (JMR). 2006;43(2):276–86.
45. Farquad MAH, Ravi V, Raju SN. Churn prediction using comprehensible support vector machine: an analytical CRM application. Appl Soft Comput. 2014;19:31–40.
46. Huang K, Yang H, King I, Lyu MR. Imbalanced learning with biased minimax probability machine. IEEE Trans Syst Man Cybern B. 2006;36(4):913–23.
47. Sun Y, Wong AK, Kamel MS. Classification of imbalanced data: a review. Int J Pattern Recognit Artif Intell. 2009;23(4):687–719.
48. He H, Garcia EA. Learning from imbalanced data. IEEE Trans Knowl Data Eng. 2009;21(9):1263–84.
49. Huang K, Zhang R, Yin X-C. Imbalance learning locally and globally. Neural Process Lett. 2015;41(3):311–23.
50. Xie Y, Xiu L, Ngai E, Ying W. Customer churn prediction using improved balanced random forests. Expert Syst Appl. 2009;36(3):5445–9.
51. Zhu B, Baesens B, Backiel A, et al. Benchmarking sampling techniques for imbalance learning in churn prediction. J Oper Res Soc. 2018;69(1):49–65. https://doi.org/10.1057/s41274-016-0176-1.
52. Wangperawong A, Brun C, Laudy O, et al. Churn analysis using deep convolutional neural networks and autoencoders. 2016. arXiv:1604.05377.
53. Kasiran Z, Ibrahim Z, Mohd Ribuan MS. Customer churn prediction using recurrent neural network with reinforcement learning algorithm in mobile phone users. Int J Int Inf Process. 2014;5(1):1–11.
54. Spanoudes P, Nguyen T. Deep learning in customer churn prediction: unsupervised feature learning on abstract company independent feature vectors. 2017. arXiv:1703.03869.
55. Chen T. Introduction to boosted trees, University Of Washington. 2014. http://homes.cs.washington.edu/~tqchen/pdf/BoostedTree.pdf.
56. https://en.wikipedia.org/wiki/Gradient_boosting.