Anomaly detection methods to detect click fraud in online advertising

Predictive Analytics : Spring 2015 Final Project

Aiman Arisha, Dimple Sharma

University of California Santa Cruz Silicon Valley Extension

Abstract

The purpose of this project is to create predictive model to identify fraudulent publishers who generate illegitimate clicks or click fraud - the deliberate clicking on advertisements with no real interest on the product or service offered - and distinguish them from normal publishers. It makes use of fraud data from BuzzCity, a global mobile advertising company, who provided real world fraud data for 3 days, for Fraud detection in mobile advertising 2012 competition. We implemented a variety of different classification algorithms based on supervised and semi-supervised learning technique to 30 predictive features, which consist of 10 click behaviors, 10 click duplications, and 10 high-risk click behavior features to determine whether illegitimate clicks came from a fraudulent publisher. Our key data insights are that invalid or fraudulent clicks have temporal and spatial characteristics, which distinguish them from normal clicks, and that simple statistical feature can retain predictive power over time and reduce the chances of over-fitting the training data. Our principle finding is that an ensemble of neutral network methods provides a promising solution to highly-imbalance nonlinear classification task with missing and noisy patterns.

Introduction

A reliable click fraud detection system is needed to assist the advertising commissioner proactively prevent click fraud and assure advertisers that their dollars have been well spent. Online advertising has become an ideal choice for small and large business to target appropriate market segments in real time. In the online advertisement business model, the advertising commissioner is the main coordinator. See figure 1 in Appendix C. A content publisher displays advertisements on their website and earns a commission for the traffic it drives to the advertisers. Buzz City adopts cost per click (CPC) payment scheme with most publishers in the network; hence the payment scheme is subject to abuse by malicious publisher through click fraud.

Data

This project makes use of a real world click fraud data provided by BuzzCity for a Fraud detecting in mobile advertising (FDMA) 2013 competition. The competition aimed at building data-driven methods to detect fraudulent publishers.

The raw dataset consist of two categories publisher database and click database, provided in a comma-separated values (CSV) format. The publisher database records publisher profile along with the status label, fields listed in Table 1 in Appendix A. On the other hand, click database captures the click traffic associated with the publishers, fields listed in Table 2 in Appendix B.

Feature Engineering

The 30 predictors can be grouped into three categories of features: 10 click behavior, 10 click duplication, and 11 high-risk click behavior; and their relative influences of all features are shown in figure 2 in Appendix D. The figure shows that duplicated clicks are invalid clicks with a high influence and more predict power.

The features capture temporal and spatial aspects of clicks for each publisher. We use simple statistics features based on average, standard deviation, and percentages, which retain predictive power over time and reduce the chances of overfitting the data.

A key data insight is that fraudulent clicks have significantly more duplicates than normal clicks within short duration of one minute interval. To capture some temporal dynamics of click fraud behavior we divided a day into 6 hour periods – night (12 am to 5:59 am), morning (6 am to 11:59 am), afternoon(12 pm to 5:59 pm), and evening (6 pm to 11:59 PM). To capture spatial characteristics we create features to identify high risk countries, as fraudulent clicks tend to come from some countries and finer grain spatial regions. We calculated percentage of invalid clicks originating from high risk countries such as Indonesia.

Algorithms

This section provides the details of the algorithms used in the experiment. For the anomaly detection task of detecting the fraudulent publishers, a wide range of single and ensemble learning algorithms were implemented focusing on Supervised as well as Semi-supervised classification techniques. These algorithms have been compared and evaluated in terms of precision, recall, F-score, Matthews Correlation Coefficient and the area under the ROC curve to determine which machine learning method produces better results to click fraud detection in a class imbalance scenario.

I. Rule Based Model

The first algorithm that was implemented was a Rule Based model using C5.0 algorithm. The classifier was trained on the 30 predictive features in the training set using C5.0 decision trees and tested on the validation and test set. However, keeping in mind that decision trees generally do not have the same level of predictive accuracy as other robust classification algorithms we set out to test different approaches.

II. Naïve Bayes

Our next approach was to build a Naïve Bayes classifier, which is based on the Bayesian theorem of conditional probability. Naïve Bayesian based methods have been used for anomaly detection to estimate the posterior probability of observing a class label (from a set of normal and anomaly class labels). The class label with largest posterior is chosen as the predicted class for the given test data instance. Laplace smoothing was incorporated into the algorithm in order to handle zero probabilities, especially for the anomaly class. An inspection of the results from Naïve Bayes results revealed a high false-positive rate. The model was tuned to minimize the FPR by setting a lower threshold probability value.

III. Replicator Neural Networks

As a next approach a Replicator Neural Network was implemented. The neural network was fit with three-size hidden-layer with skip-layer connections. The final parameters used on the training data set are:

- size (number of units in the hidden layer): 3
- maxit (maximum number of iterations): 150
- decay (parameter for weight decay): 5e-4
- entropy (switch for entropy fitting): T
- skip (switch to add skip-layer connections): T
- rang (Initial random weights): -0.25

IV. Ensemble Method: Random Forest

Random Forest is a machine learning technique based on the bagging method and the random selection of features. In Random Forest each tree in the forest is built from a sample drawn with replacement based on bootstrap method from the train set. An ensemble of random forests of 15 trees was implemented. The final parameters used on the training data set are:

- ntree (number of trees to grow): 15
- mtry (number of variables randomly sampled as candidates at each split): 5
- importance (Should importance of predictors be assessed?): TRUE

V. Ensemble Method: GBM

Gradient boosting is a machine learning technique used for classification problems with a 'bernoulli' loss function, which produces a final prediction model in the form of an ensemble of weak prediction decision trees. The implementation of generalized boosted regression model (GBM) in R's gbm package was used. The final parameters used on the training data set for best performance on the test data set are:

- distribution (loss function): "bernoulli"
- n.trees (number of iterations): 5000
- shrinkage (learning rate): 0.001
- interaction.depth (tree depth): 5
- n.minobsinnode (minimum observations in terminal node): 5

In Figure 2 Appendix D, the relative influence or importance from the 30 predictive features obtained using specific parameters of GBM in the final training dataset is shown.

VI. Multi-Layer Perceptron

The multi-layer perceptron (MLP) is a simple feed-forward neural network with an input layer, several hidden layers and one output layer. The implementation involves three hidden layers sandwiched between an input layer and an output layer and an ensemble of 15 neural networks. The final parameters used on the training data set are:

- hidden1 (hidden1 number of hidden nodes in the first hidden layer): 3
- n.ensemble (number of ensemble members to fit): 15
- monotone (monotonicity constraint): 1
- bag: TRUE

VII. One class Support Vector Machine

The final approach was to be implemented a semi-supervised technique called One-class SVM for anomaly detection. In this case, a training dataset containing only the normal instances is used to learn a model. The learned model is then applied on the test dataset containing both normal and anomalous instances. And the records that do not comply with the model are labeled as outliers. The final parameters used on the training data set are:

- type (svm type): 'one-classification'
- scale (scaling the variables): TRUE
- nu (parameter needed for one-classification): 0.10
- gamma (parameter needed for all kernels): 0.01

However, an inspection of the results of the model shows high false alarm rate (or FPR), which means that legitimate records are being classified as anomalies. This is a general drawback of a Semi-Supervised technique.

Evaluation Metrics

For model evaluation of anomaly detection problem, we will use precision, recall, F beta-score, Matthews Correlation Coefficient and the area under curve or ROC to evaluate classification models. Our key model evaluation insight is that F beta-score or F2-score provides a good solution to evaluate model classification error as it weights recall higher than precision - recall gives true positive rate while precision gives false positive rate. Another key finding includes Matthews's correlation coefficient which takes into account all four measures – true positive, false negative, true negative and false positive- and provides a balanced measure for binary classification with classes of very different sizes.

As accuracy provides proportion of true results combining both true negative and true positive, it cannot be used as a measure to evaluate models due to high imbalance in class of normal and anomalous instances, where only 2% or 3% of data contains anomalies.

Following contains results of key matrices for each classification method:

Algorithms Tested	Precision (%)	Recall (%)	F Beta Score (%)	MCC (%)	Value of AUC
Random Forest	66.667	26.829	30.471	41.352	63.226
Rule Based	57.692	36.585	39.474	44.766	67.916
GBM	64.103	30.488	34.060	43.195	65.004
One-class Support Vector Machine	11.688	43.902	28.302	18.577	67.290
Naïve Bayes	24.219	37.805	33.991	27.818	67.240
Neural Networks	29.333	26.829	27.295	26.123	62.506
Ensemble Neural Network	**42.453**	**54.878**	**51.843**	**46.621**	**76.394**

Conclusion

We analyze the effect of classification techniques ranging from single classifier algorithms to ensemble learning algorithms such as random forest, GBM and ensemble neural networks. Based on our results we can conclude that ensemble learning algorithms tend to produce better results on the highly imbalanced dataset.

Feature engineering plays a critical role in any classification problem and it is even critical when class imbalance problem exists. Form our experimentation results, it can be seen that carefully crafted features set along with ensemble learning algorithms could yield to better results in an imbalanced class distribution classification problem.

Challenges

There are two key challenges that we faced. The first would be the class imbalance of the normal and anomalous instances, which made learning a classifier quite challenging. From machine learning point-of-view, algorithms can provide high interpretability to fraud detection. However, the most important step is to engineer the best features using domain knowledge and subject matter expertise.

Appendix

A. Publisher database

Table 1: Publisher database

Field	Description
Publisherid	Unique identifier of a publisher
Bankaccount	Bank account associated with a publisher
Address	Mailing address of a publisher
Status	OK: Publisher who have a healthy traffic Fraud: Publisher deemed as fraudulent with clear proof

B. Click database

Table 2: Click database

Field	**Description**
Id	Unique identifier of a clicks
Numericip	Public ip address of a visitor/clicker
Deviceua	Phone model used by a visitor/clicker
Publisherid	Unique identifier of a publisher
Adscampaignid	Unique identifier of a given advertisement campaign
Usercountry	Country from which the surfer is
Clicktime	Timestamp of a given click
Publisherchannel	Publishers channel type: ad - adult sites, co- community, es –entertainment, and lifestyle, gd - glamour and dating, in – information, mc – mobile content, pp – premium portal, se – search, portal services.
Referredurl	URL were ad banners were clicked.

C. Online advertising Business model

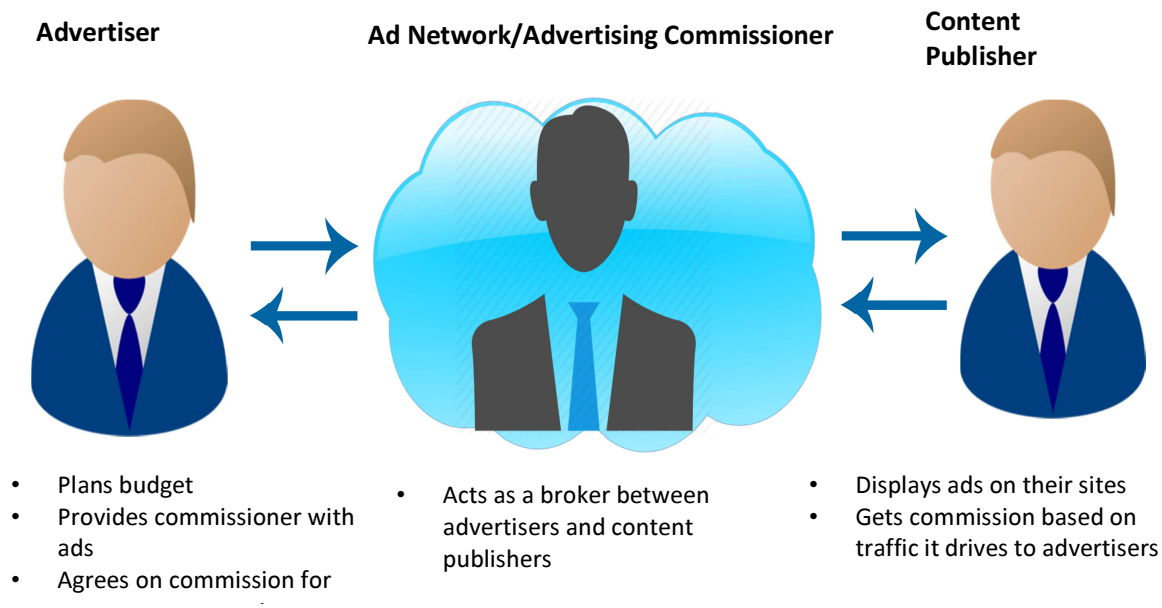

Figure 1 - Online advertising business model

D. Relative influence of features

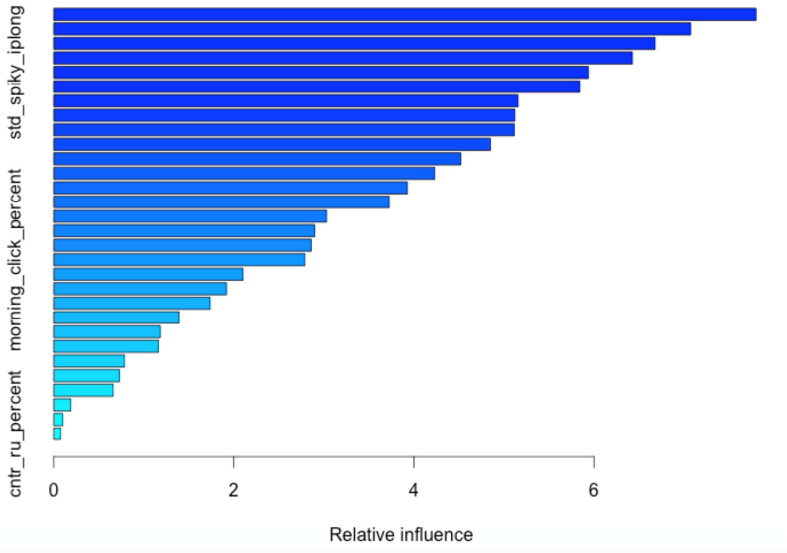

Figure 2 – Relative Influence of Features

References

[1] FDMA competition website (http://palanteer.sis.smu.edu.sg/fdma2012/)

[2] Detecting Click Fraud in Online Advertising: A Data Mining Approach, Journal of Machine Learning Research 15 (2014) 99-140, http://www.jmlr.org/papers/volume15/oentaryo14a/oentaryo14a.pdf

[3] V. Chandola, A. Banerjee and V.Kumar, “Anomaly Detection: A Survey”, University of Minnesota

[4] Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani, An Introduction to Statistical Learning with Applications in R, http://www-bcf.usc.edu/~gareth/ISL/ISLR%20First%20Printing.pdf

[5] Dhaval Patel, Chen Wei, Hybrid Models for Click Fraud Detection in Mobile Advertising, http://palanteer.sis.smu.edu.sg/fdma2012/doc/ThirdWinner-DB2-Paper.pdf