Hierarchical Committee Machines for Fraud Detection in Mobile Advertising

S. Shivashankar S.SHIVASHANKAR@ERICSSON.COM
Ericsson Research,
Chennai, India

P. Manoj MANOJ.M.P@ERICSSON.COM
Ericsson Research,
Chennai, India

Abstract:

In recent years, a significant amount of attention is being paid towards fraud clicks in online advertisements. Researchers have started paying an equal amount of attention to it as towards other problems such as placing right ads on a page, personalizing it for an user, etc. In this paper, we elaborate the method we used to predict fraud clicks in mobile ads. The dataset was provided by BuzzCity as part of the machine learning contest held in conjunction with Asian Conference on Machine Learning 2012 [1]. As in the case of any fraud detection problem, this particular challenge also involves class imbalance issues. More importantly, as repeatedly said by experts [2] feature engineering is the key for good performance. We built a lot of derived attributes which played a critical role in improving the performance. We used hierarchical committee machines to combine a set of diverse cost sensitive classifiers built using different set of attributes (datasets). More details about feature engineering and methods used can be found in later sections of the paper.

1. INTRODUCTION:

Trends are rapidly changing due to the influence of new technologies. Conventional newspaper advertisements, television advertisements etc., to attract customers are overtaken by online advertisements. With the growth in smartphones usage [6], mobile advertisements are booming in volume. Though online advertisements have become one of the primary avenues, handling fraud clicks is still an open problem [7].

Problem scenario is explained as follows: an advertising commissioner is provided with advertisements, budget for advertisements and a commission for each click by the advertiser. Content publishers in agreement with the advertising commissioner publish advertisements in their websites for a publishing cost. Dishonest publishers earn money by charging the advertisers a humongous cost, inflating the number of clicks (*click fraud*). Click frauds reduce the reliability of online advertising systems. It is the duty of an advertisement commissioner to prevent click frauds to make the online advertising systems more reliable and to convince advertisers with fair accounts. There is a need for a click fraud detection system to identify dishonest publishers and maintain the reliability of online advertisement systems in a long-term. In this paper, we would discuss our method to identify fraud clicks in the data provided by BuzzCity [5].

Given data contains publisher database with the publisher's profile and clicks database with details about click traffic. The goal is to build a method for effective detection of dishonest publishers. In particular, the goal is to detect “Fraud” publishers and separate them from “OK” and “Observation” publishers, based on their traffic properties and profile. This would aid in revealing the underlying fraud scheme and the concealment strategies of dishonest publishers.

In Section 2, we explain the feature engineering performed. In Section 3, we elaborate the method used for fraud click detection. In Section 4, we provide key insights based on the study performed.

2. DATA PREPROCESSING AND FEATURE EXTRACTION:

Given data has the publisher data base and click database in CSV format. The publisher database records the publisher’s profile and comprises several fields: partner-id, bank-account, address and status – “OK”, “Observation” and “Fraud”. The click database records the click traffic and has several fields: id, IP address, phone model, partner-id, campaign id, country, time-stamp, category and referrer URL.

Since the number of click records are huge (3.1 million records in training set and 2.6 million records in validation set), we used Hadoop map reduce for feature engineering [2]. Each publisher was represented using the properties of the clicks made. Publisher database fields such as bankaccount and address were not very useful as the score using the proposed method on the validation dataset was 28.94. The common intuition to use duplicate IP address or repetitive clicks from the same IP address (gave a score of 39.64). Repetitive clicks from the same IP address were found to be common for publishers with "OK" status also. Similarly, country information was not found to be a discriminating field. Derived attributes based on other fields that characterized total/average number of clicks were useful, for example total number of clicks, average clicks per campaign id, etc. Timestamp was a crucial field to be modeled for good performance. We derived attributes such as number of clicks per day, sum/average/standard deviation over difference in click timestamp between subsequent clicks for a publisher, etc. It might be fruitful to invest more efforts on the timestamp field in order to improve the results further. Surprisingly, phone model information helped with a score of 47.57. Scores of different attribute sets with diverse classifiers is given in Table 1. *Basic features* derived for each publisher includes: number of unique IP, unique categories, unique countries, unique campaign ids, category of clicks and total number of clicks per publisher.

Attributes	K-Star	Decorate with j48	AODE
Basic features + top 5 countries and top 5 categories with maximum number of clicks per publisher	14.57	17.94	12.87
Basic features + average clicks per agent, IP, category, campaign id per publisher	25.08	32.98	27.54
Basic features + clicks per category, country, day per publisher	24.06	27.87	17.64
Basic features + sum, average, standard deviation over difference in click timestamp between subsequent clicks for a publisher	28.64	41.99	37.54

Table 1 Performance of various features with K Star, Decorate with j48 and AODE. “+” indicates that the additional features were appended to the set of basic features.

A list of all derived attributes given in Table 2.

S.NO	Feature	Description
1	Number of unique ip	No of unique ip's per pid
2	Number of unique cid	No of unique cid's per pid
3	Number of unique cntr	No of unique cntr's per pid
4	Number of unique category	No of unique categories per pid
5	Total clicks	Total clicks per pid
6	Category	Category name for which clicks exist per pid
7	Country feature vector	Country wise clicks per pid. 1 X C vector, where C is the total number of countries
8	Category feature vector	Category wise clicks per pid. 1 X N, where N is the number of categories
9	Clicks per category	No of clicks per category per pid
10	Countries with highest number of clicks	Countries sorted according to number of clicks and K countries with highest clicks per cid are appended.
11	Bank account given or not	Boolean attribute : 0 if the bank account for pid is not given, else 1
12	Address given or not	Boolean attribute : 0 if the address for

		pid is not given, else 1
13	Top country	Country with highest clicks per pid
14	Cluster id	Cluster clicks data into predefined number of clusters, say 5, add the distribution of clicks within the clusters as a feature vector.
15	No of referrers	No of unique referrers per pid
16	Number of days	Number of days pid is active
17	Clicks per day	Number of clicks per day per pid
18	Sum of difference in time	Sum of time difference between each click for a pid
19	Average of difference in time	Average over difference in time between each click for a pid
20	Standard deviation of sum of difference in time	SD of time difference between each click for a pid
21	Clicks per category	Total number of clicks per category per pid
22	Average clicks - day	Average of clicks per day per pid
23	Average clicks - referrer	Average of clicks per referrer per pid
24	No of agents	No of unique agents per pid
25	Sum of difference of clicks – ip and cid	Sum of difference of clicks per ip per cid per pid
26	Sum of clicks	Duplicate clicks sum
27	Average clicks – agent	Average clicks per agent per pid
28	Average clicks - ip	Average of clicks per ip per pid
29	Average clicks - cid	Average of clicks per cid per pid
30	Average clicks - cntr	Average of clicks per cntr per pid

Table 2 : Derived Attributes

3. METHODS USED AND EXPERIMENTAL CONFIGURATION:

The problem to identify fraud publishers was posed as a binary classification problem since there are efficient algorithms to solve binary class classification problems. Two ways to solve it as a binary class classification problem were investigated. In the first approach, “Observation” and “OK” instances were combined together as “OK” instances. Maximum score obtained using this approach with the proposed method was 32.68. In the second approach “Observation” and “Fraud” instances were grouped together as “Fraud” instances. Maximum score obtained using

this approach for the proposed method was 51.49. Table 3 describes various attributes in Dataset A, B, C and D. As mentioned in Section 2, *basic features* derived for each publisher includes: number of unique IP, unique categories, unique countries, unique campaign ids, category of clicks and total number of clicks per publisher.

Dataset A	Basic Features + class variable, clicks of category, number of days, number of clicks per day, number of referrers per publisher
Dataset B	Dataset A + sum, average, standard deviation over difference in click timestamp between subsequent clicks for a publisher
Dataset C	Dataset B + number of agents per publisher
Dataset D	Dataset C + average clicks per campaign id, country, referrer, day per publisher

Table 3 : Description of datasets. “+” indicates that the additional features were appended to the set of basic features.

This particular fraud detection challenge also had class imbalance issues. Fraud instances were less compared to OK instances. Cost sensitive classification and ensemble learning were used to address the class imbalance problem. Algorithms like J48, REP tree, LAD tree, AODE were applied on all datasets. It was found that Dataset D performs well with LAD tree only (score: 47.57).

Hierarchical committee machines as shown in Figure 1 was used to infer the probability of fraud in a given instance. Each committee machine was used to combine the responses of diverse classifiers on datasets that included different sets of derived attributes. The diverse classifiers include j48, K Star, LAD tree, AODE and REP tree. Finally, a committee machine was used to combine the responses from individual committee machines that were built on different datasets. The score on the validation set was 51.49 and the score on the test set was 38.0744. The parameters (weights) of the committee machines were found empirically.

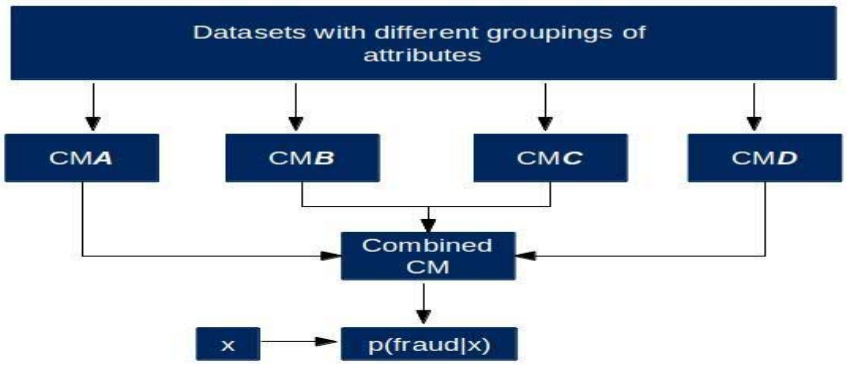

Figure 1: Hierarchical Committee Machines

Results using classifiers that performed well and that had diverse results (to help in ensemble learning) are given in Figure 2. Not all classifiers that were tried are given in Figure 2. The five classifiers used are Decorate with j48, Bagging with REP tree, Bagging with cost-sensitive classifier with LAD tree, K Star and AODE respectively. It can be seen that Dataset D performed well with most of the methods. From the figure, it is easy to interpret that dataset D gave the best performance (47.57) using Bagging with cost-sensitive LAD Tree.

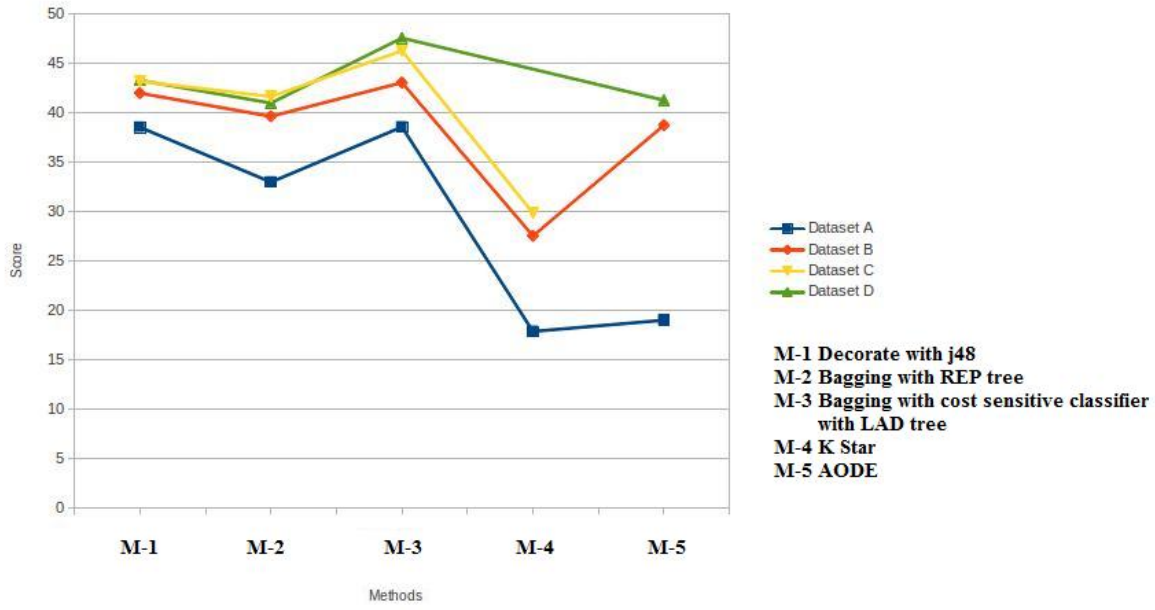

Figure 2: Results

4. RESULTS AND KEY INSIGHTS ON FRAUDULENT BEHAVIOUR:

INSIGHTS ON METHODS:

- Typical methods such as over-sampling, under-sampling, SMOTE, HDDT did not perform very well (score was less than 25). Since it is a widely accepted method for scenarios with class imbalance, sampling methods have to be investigated carefully to see how they can be useful,
- Popular methods like Random Forest did not perform better than other tree counterparts.
- It is important to try ranking methods, as the evaluation metric was average precision. Bayesian based ranking methods such as AODE performs well (score greater than 35) with more derived attributes.
- Cost-sensitive classification performed well with a score greater than 40 with only few classifiers like LAD Tree.
- With more attributes LAD tree performs well individually(score greater than 45), but does not produce so diverse results on dataset C and D. Memory based methods such as k-star do not perform well individually(score less than 28), but helped as part of the committee.

INSIGHTS ON ATTRIBUTES:

- Most of the fraud clicks belonged to publishers whose category was ‘AD’ or ‘MC’.
- Common intuition to use duplicate ip per publisher did not improve the score.
- Country information made negligible improvement in the score.
- Surprisingly phone agent (model) information of the users improved the score significantly (score : 47.57).
- Time information was critically important for good performance. Further investigations/refinements would be worthwhile to improve the results.

REFERENCES:

1. http://palanteer.sis.smu.edu.sg/fdma2012/
2. Pedro Domingos, A Few Useful Things to Know about Machine Learning. Communications of the ACM, 55 (10), 78-87, 2012
3. Yu, Hen Hu and Jenq-Neng Hwang (eds.),Volker Tresp,”Handbook for Neural Network Signal Processing”, July 31, 2001
4. http://en.wikipedia.org/wiki/MapReduce
5. http://www.buzzcity.com
6. http://www.ericsson.com/res/docs/2012/ericsson_superior_network_performance.pdf
7. http://www.google.com/ads/adtrafficquality/index.html