Forecast of Mobile Ad Click through Logistic Regression Algorithm

M. Sree Vani
Dept of CSE, MGIT , Hyderabad-500075

Abstract

Mobile advertising gives opportunities for advertisers to only bid and pay for measurable user responses, such as clicks on ads. As a result, click prediction systems are central to most mobile advertising systems. With millions of mobile user's daily activities and active advertisers, predicting clicks on Mobile ads is a challenging machine learning task. This paper presents an experimental study of using different machine learning techniques to predict whether an mobile ad will be clicked or not. Our approach is applied on Avazu mobile Ad click data set. In this paper feature selection is performed to remove features that do not help improve classifier accuracy. Several supervised classification algorithms are applied in experiments and observe that logistic regression with feature selection produces better classification accuracy.

I. INTRODUCTION

Mobile advertising is a multi-billion dollar industry and is growing dramatically each year. In most online advertising platforms the allocation of ads is dynamic, tailored to user interests based on their observed feedback. Machine learning plays a central role in computing the expected utility of a candidate ad to a user, and in this way increases the efficiency of the marketplace. In sponsored search advertising, the user query is used to retrieve candidate ads, which explicitly or implicitly are matched to the query. In Mobile, ads are not associated with a query, but instead specify demographic and interest targeting. As a consequence of this, the volume of ads that is eligible to be displayed when a user browses internet in mobile can be larger than for sponsored search. In order to tackle a very large number of mobile ads per request, where a request for ads is triggered whenever a user browses internet, Logistic Regression is usually considered a good supervised classification algorithm for most of the datasets. So, application of logistic regression with stochastic gradient descent produces better logloss score for this dataset.

The rest of the paper describes about methodology in section 2 and Experimental setup is described in section 3 in detail. Results are discussed in section 4 and finally section 5 gives conclusions.

II. METHODOLOGY

This paper initially describes the data pre-processing and feature selection techniques that have been applied. Thereafter, it explains the approach that produces better results - logistic regression with stochastic gradient descent and weights regularization.

A. Data Pre-Processing

Feature Selection is a very important step in data pre-processing that determines the accuracy of a classifier. It helps remove unnecessary correlation in the data that might decrease accuracy. As expected the most important thing is to have the right features: those capturing historical information about the user or ad dominate other types of features. Initially Principal Component Analysis [4] is implemented to reduce the number of attributes in train data. PCA is often useful to measure data in terms of its Principal Components rather than on a normal x-y axis. Principal Components are the underlying structure in the data. They are the directions where there is most variance, the directions where the data is most spread out.

Afterwards feature selection is performed manually on the usefulness of historical and contextual data items. The features used in our methodology can be categorized into two types: contextual features and historical features. The value of contextual features depends exclusively on current information regarding the context in which an ad is to be shown, such as the device used by the users or the current page that the user is on. Some example contextual features can be local time of day, day of week, etc. On the contrary, the historical features depend on previous interaction for the ad or user, for example the click through rate of the ad in last week, or the average click through rate of the user.

B. Classification

Initially, naive bayes is applied on our dataset. However, it did not give us very good results because it assumes feature independence. Also, as the number of unique values is each feature is high and the training dataset is significantly large in size, it is computationally expensive to compute all the probabilities. Next, Vowpal Wabbit which is an open source fast out-of-core learning system library is applied. The produced results were good and significantly better than naive bayes. Due to more volume in the data set Logistic Regression is usually considered a good supervised classification algorithm for most of the datasets. So, application of logistic regression with stochastic gradient descent produces better logloss score for this dataset.

Logistic Regression states that

$$P(y=1|x;w)=\sigma\big(g(x)\big) \qquad (1)$$

Where

$$g(x)=\textstyle\sum_d w_d x_d = w^T x \qquad (2)$$

$$\sigma(a)=\frac{1}{1+e^{-a}} \qquad (3)$$

The regularization parameter (λ) is estimated using cross validation. Figure 3 shows all the values of λ and the corresponding values of Log Loss. It can be seen that the value of 0.00025 has given the least Log Loss and thus it is chosen as the optimum λ value for this data. It could be possible that this value of lambda is a local optimum and there might be some global optimum that could produce better results. But, this is the best value produced by using greedy approach. Because the data is so huge, it is not possible to load the entire data into memory. So, batch gradient descent and newton method were not possible. Hence, the suitable approach is Stochastic Gradient Descent approach for updating the weights. The learning rate λ for the Stochastic Gradient Descent has also been experimented upon. The initial weights for our approach are 0. Different values are considered for the learning rate and chose the optimum value as 0.01.

Stochastic Gradient Descent (SGD) algorithm

An ad impression is given in terms of a structured vector x = ($e_{i_1}, e_{i_2}, e_{i_3}, \dots e_{i_n}$) where e_i is the i^{th} unit vector and $i_{1,}i_2,\dots i_n$ are the values of the n categorical input features. In the training phase, we also assume that we are given a binary label y $\in$ {+1,-1} indicating a click or no-click. Given a labeled ad impression (x, y), let us denote the linear combination of active weights as

$$s(y,x,w) = y.w^T x = y\sum_{j=1}^{n} w_j i_j \qquad (4)$$

Where w is the weight vector of the linear click score.

III. EXPERIMENTAL SETUP

This work performs prediction of Ad clicks from Avazu data set. The Avazu data set contains 5.87 GB of training data and 673 MB of test data. The train data has 24 attributes including the class label. Figure 1 shows data distribution of class label in training dataset. It can be seen that there are very less number of clicks. This is expected because out of the large number of ads displayed on a webpage, people hardly click on any ads.

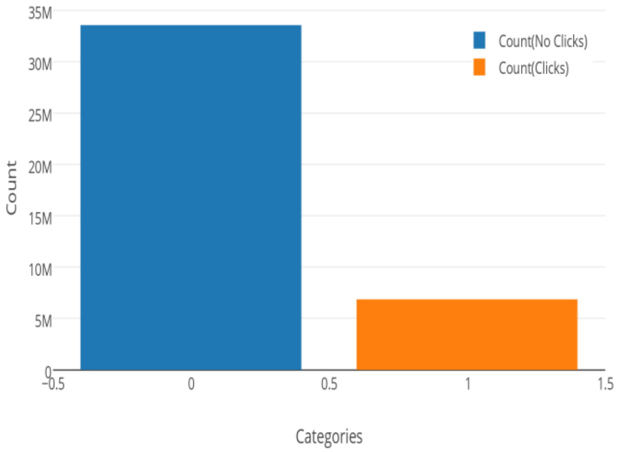

Figure 1: Class Label

A. Evaluation Metrics:

In this work, Normalized Entropy (NE) and calibration are considered as major evaluation metrics because the accuracy of prediction model is concerned more compared to metrics directly related to profit and revenue.

Normalized Entropy is equivalent to the average log loss per impression divided by what the average log loss per impression would be if a model predicted the background click through rate (CTR) for every impression. In other words, it is the predictive log loss normalized by the entropy of the background CTR. The background CTR is the average empirical CTR of the training data set. It would be perhaps more descriptive to refer to the metric as the Normalized Logarithmic Loss. The lower the value is, the better is the prediction made by the model. The reason for this normalization is that the closer the background CTR is to either 0 or 1, the easier it is to achieve a better log loss. Dividing by the entropy of the background CTR makes the NE insensitive to the background CTR. Assume a given training data set has N examples with labels y_i ={+1, -1} and estimated probability of click pi where i = 1, 2, …. N. The average empirical CTR as p

$$NE = \frac{-\frac{1}{N}\sum_{i=1}^{n}\left(\frac{1+y_i}{2}\log(1-p_i)\right)}{-(p*\log(p)+(1-p)*\log(1-p))}$$

NE is essentially a component in calculating Relative Information Gain (RIG) and RIG = 1 – NE.

Calibration is the ratio of the average estimated CTR and empirical CTR. In other words, it is the ratio of the number of expected clicks to the number of actually observed clicks. Calibration is a very important metric since accurate and well-calibrated prediction of CTR is essential to the success of online bidding and auction. The less the calibration differs from 1, the better the model is. We only report calibration in the experiments where it is non-trivial.

IV. RESULTS

It is evident that unnecessary features act as noise and degrade the performance of the system. This work investigates several approaches for data reduction using feature selection namely Naïve bayes [3, 5], Vowpal Wabbit [1]. The performance of our approach may depend on the two types of features. Firstly consider the relative importance of the two types of features. Relative importance can be calculated by sorting all features by importance, then calculate the percentage of historical features in first k-important features. The result is shown in Figure 2. The historical features provide considerably more explanatory power than contextual features. The top 10 features ordered by importance are all historical features. Among the top 20 features, there are only 2 contextual features despite historical feature occupying roughly 75% of the features in this dataset.

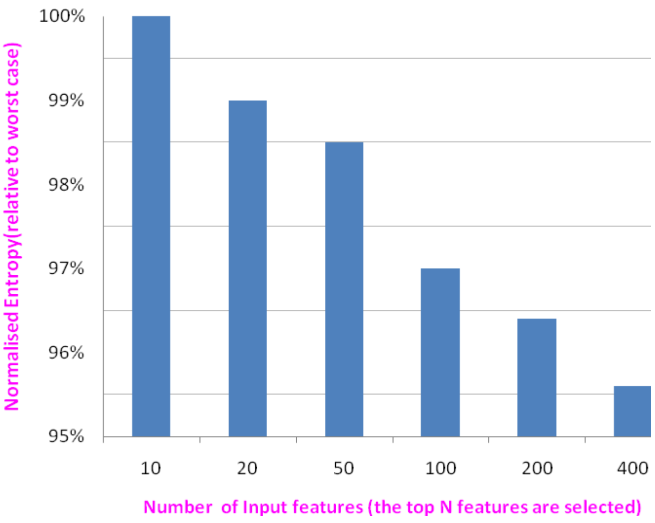

Figure 2: Results for Historical Feature Percentage.

The Table 1 shows the scores achieved for the different techniques that are implemented in this work. It can be seen that logistic regression with proper data pre-processing produces the best score and good ranking. It produces the best score of 0.393862 which is a significant improvement over many other approaches.

Table 1: Different Approaches Implemented and Scores

Approach	**Score**
Naïve bayes	0.6013
Vowpal Wabbit	0.47
Vowpal Wabbit(Feature Selection)	0.4007
Logistic Regression	0/396
Logistic Regression(Feature Selection)	0.3938

Figure 3 plots the different log-loss scores for different value of regularization parameter λ for logistic regression. The selected value of λ that gave us the best results i.e. 0.00025.

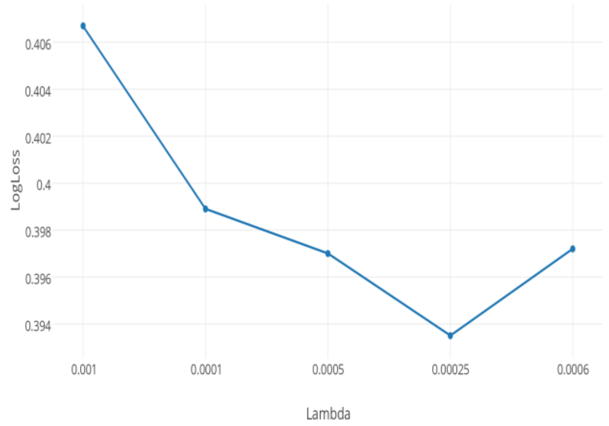

Figure 3: Parameter Estimation

V. CONCLUSION

This work investigates several approaches for data reduction using feature selection. It is evident that unnecessary features act as noise and degrade the performance of the system. Therefore, it is essential to remove these fields. In this paper several supervised classification algorithms are applied to see which one will work best for our dataset. Logistic regression with feature selection produces better results. Random Forest algorithm will be implemented on such large datasets in future.

REFERENCES

[1] Vowpal Wabbit open tool. http://hunch.net/˜vw/. Accessed: 2010-09-30.

[2] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. The Journal of Machine Learning Research, 12:2121–2159, 2011.

[3] Zhipeng Fang, Kun Yue, Jixian Zhang, Dehai Zhang, and Weiyi Liu. Predicting click-through rates of new advertisements based on the bayesian network. Mathematical Problems in Engineering,2014, 2014.

[4] Dejian Lai. Principal component analysis on human development indicators of china. Social

[5] Indicators Research, 61(3):319–330, 2003.

[6] David D Lewis. Naive (bayes) at forty: The independence assumption in information retrieval.In Machine learning: ECML-98, pages 4–15. Springer, 1998.5

[7] H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady,Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, et al. Ad click prediction: a view from the trenches. In Proceedings of the 19th ACM SIGKDD international conference on

[8] Knowledge discovery and data mining, pages 1222–1230. ACM, 2013.

[9] Steffen Rendle. Scaling factorization machines to relational data. In Proceedings of the VLDB Endowment, volume 6, pages 337–348. VLDB Endowment, 2013.6