Adaptive Targeting for Online Advertisement

Andrey Pepelyshev[1(✉)], Yuri Staroselskiy[2], and Anatoly Zhigljavsky[1,3]

[1] Cardiff University, Nizhnii Novgorod, Russia
{pepelyshevan,ZhigljavskyAA}@cardiff.ac.uk
[2] Crimtan, London, UK
yuri@crimtan.com
[3] University of Nizhnii Novgorod, Nizhnii Novgorod, Russia

Abstract. We consider the problem of adaptive targeting for real-time bidding for internet advertisement. This problem involves making fast decisions on whether to show a given ad to a particular user. For intelligent platforms, these decisions are based on information extracted from big data sets containing records of previous impressions, clicks and subsequent purchases. We discuss several strategies for maximizing the click through rate, which is often the main criteria of measuring the success of an advertisement campaign. In the second part of the paper, we provide some results of statistical analysis of real data.

Keywords: Online advertisement · Real-time bidding · Adaptive targeting · Big data · Click through rate

1 Introduction

Online advertising is an important form of marketing where advertisements shown to a user may depend on the user browsing behaviour. Advertising platforms collect big data which may include records of previous conversions, clicks, impressions, visited webpages, account information and search requests. A large part of online advertisements goes through prominent technology companies like Google, Yahoo, Bing and Facebook, which are able to collect enormous amounts of data on the user behaviour, see e.g. [4,6,8,12,19]. Some part of online advertisement spend goes through independent ad exchanges where advertising platforms have less information about users [14]. The present paper deals with the latter case.

Ad exchanges as well as search providers use Real-Time Bidding (RTB), which is a popular way of delivering online advertising, see [3,9,13,20]. As reported in [5], spending on RTB in the US during 2014 increased by 137 % and reached $10 billion and RTB has 45 % of the total spend in online advertising. In contrast to traditional advertising on TV and fixed contracts on showing fixed advertisements on specific websites, RTB enables a demand side to find a favorable ad campaign and submit a bid for a request depending on parameters of the request and behaviour data (i.e. a track record of a user). In our case the demand side is represented by an advertising platform whose core business is

P. Pardalos et al. (Eds.): MOD 2015, LNCS 9432, pp. 240–251, 2015.
DOI: 10.1007/978-3-319-27926-8_21

in delivering efficient advertisements on websites, see [14]. Marketing managers expect that online advertising brings customers at cheaper costs and granular targeting capabilities although the traditional offline advertisement is continued.

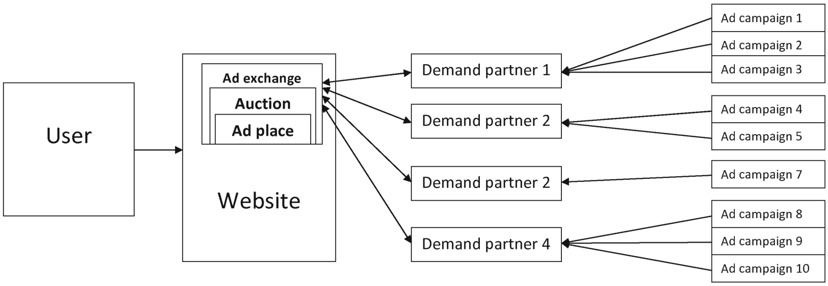

Fig. 1. The scheme of real-time bidding for online advertising.

In Fig. 1 we show the scheme of the RTB system, which consists of 4 components: a user, a webpage with embedded ad place, demand partners (advertising platforms) and ad campaigns; see [17,20] for more detail.

The process of delivering online advertisements occurs billions times each day and consists of the following steps:

- A user comes to a webpage of a web site, where advertisement can be delivered using auction via an ad exchange.
- The web site via the ad exchange notifies several demand partners that there is a possibility to show an ad via bid request (real time auction). Each bid request contains information about user (user id, time of request, IP, geo, user agent) and information about the site (site, url, minimal bid). To make efficient decision demand side can store and analyze information about bid requests. Due to the enormous amount of bid requests storage and analysis of this data is a true big data challenge.
- If a demand partner decides to deliver an ad for the given request, it responds with a bid and a particular advertisement. The demand partners are usually required to return a bid in a short time (e.g. 100 ms) while the webpage is loaded by a user. The bid is given in a certain currency (often USD) multiplied by 1000, corresponding to the commonly adopted cost-per-mille pricing model.
- The website via the ad exchange decides which demand partner won the auction (based on their bids) and delivers the ad of the winner. Note that ad exchanges are working as the second price auction model; that is, the winner pays the second highest bid.
- If a demand partner wins, it delivers the ad and can store information about ad delivery in order to analyze historical efficiency. Note that the user is given the right to opt out from targeted advertisement delivery via demand platform site opt out or via ad itself. In such case the demand partner doesn't store user related information.

- If the user clicks on the delivered ad, the advertiser can store the information about clicks.
- If the user visits the advertised site which contains code of the advertiser, the demand partner can store the information about the visit and can use it to optimize campaign efficiency further.
- If the user buys a product on the advertised site, the demand partner can store the purchase information to evaluate optimization strategies on historical data.

The advertiser has to solve the problem of maximizing either the click through rate (CTR) or the conversion rate by targeting a set of requests under several constraints:

(i) Budget (total amount of money available for advertising),
(ii) Number of impressions (total amount of ad exposures),
(iii) Time (ad campaign is restricted to certain time period).

Campaign size in programmatic segment varies between \$5000 and \$500000 per month and the advertisement company running a campaign needs to choose from 5 mln to 500 mln requests out of 50 bln available ones.

One of the main characteristics of an ad campaign is average cost-per-action (CPA) or average cost per conversion. To identify those parameters of bid request/ impressions, which caused the click/conversion, we have to use all logs.

The problem of adaptive targeting for ad campaigns was addressed in proceedings of the annual WWW conference and in a dozen of papers, however, many of them deal with the sponsored search, see e.g. [8,12,19]. Some papers, for example [2,16], use the look-alike idea implying that a new request will lead to the click/conversion if the new request is similar to (look like one of) the previous successful requests. In 2014 two Kaggle contests were organized [see https://www.kaggle.com/c/avazu-ctr-prediction and https://www.kaggle.com/c/criteo-display-ad-challenge] on algorithms for predicting the CTR using datasets with subsampled non-click records (the CTR for one dataset is about 17 %). The algorithms proposed by many teams are based on different approaches, mainly, ensembles of field-aware factorisation machines (FFM) [15], follow-the-regularized-leader (FTRL) methodology [11], gradient boosting machines (GBM) [7], and are now publicly available, give approximately the same performance with respect to the logarithmic loss criterion

$$logloss = -1/N \sum_{i=1}^{N} (y_i \log(p_i) + (1 - y_i) \log(1 - p_i)),$$

where N is the size of the test set, p_i is the predicted probability of click for the i-th request, and $y_i = 1$ if the i-th request leads to click and $y_i = 0$ otherwise.

All the main strategies mentioned above make learning either about the parameters of the model (like in FFM and FTRL) or the response function directly depending on X. This learning constitutes the main objective at the initial phase of any advertisement campaign. At a later stage in the campaign, when either models or estimates of the response function can be considered satisfactory, they are used for improving the selection of users with the purpose to increase (or even maximize) the CTR. The cost of impressions on the learning stage should be kept on the lowest level but it should be increasing as the choice of users becomes more intelligent since we should be prepared to pay higher price for the users that are more likely to click on our ad.

The present paper is organized as follows. In Sect. 2 we present the formal description of the problem of maximizing the CTR and propose an adaptive strategy which consists of estimating the preference characteristic for a new request and suggesting a relevant bid price; this strategy is based on the 'look-alike' principle and does not use any parametric models similar to those used by FFM and FTRL. In Sect. 3 we perform an analysis of data provided to us by an advertising platform. Specifically, we give the descriptive statistics in Sect. 3.1 and perform the multidimensional scaling in Sect. 3.2. Finally, we evaluate the performance of the proposed strategy in Sect. 3.3 and investigate the sensitivity of the strategy to the choice of factors in Sect. 3.4. Conclusions are given in Sect. 4.

2 Adaptive Strategy for Maximizing the CTR of an Ad Campaign

Suppose that the ad we want to show is fixed. Consider the problem of maximization of the click through rate by an adaptive targeting procedure which should yield the decision whether to show or not the ad to a request from a webpage visited by a user. If the procedure decides to show the ad, it has to propose a bid.

The adaptive decision should depend on the current sample of impressions and clicks which contain the users to whom we have shown the ad before and who have clicked on the ad. We will treat the sample size N as time. We can increase the size of the sample by including all our previous impressions of the same advertisement, so that N could be very large.

Features of an i-th request: $X_i = (x_{i,1}, \ldots, x_{i,m})$, $i = 1, \ldots, N$, where m is the number of features (factors). We equate the i-th request to X_i. Suppose that the requests leading to the click on the ad are $X_{j_1}, \ldots, X_{j_K}$, where $1 \leq j_1 < j_2 < \ldots < j_K \leq N$ and $K = K(N) < N$. Our running performance criterion of the advertising campaign is the click through rate (CTR) defined by $p_N = K/N$. It is clear that the CTR p_N changes as N grows.

We make the following important assumption of independence: if we choose a request with features $X = (x_1, \ldots, x_m)$ then the probability of a click is p_X; different events ('click' or 'no click') are independent. We assume that all possible vectors $X = (x_1, \ldots, x_m)$ belong to some set $\mathbb{X}$ (which is partly discrete and possibly has difficult structure). We also assume that for any two points X and $X' \in \mathbb{X}$ we can define some kind of measure $d(X, X')$ which can be considered as distance (it does not have to satisfy mathematical axioms of the distance function). The properties we require for $d(X, X')$ are: (a) $d(X, X') \geq 0$ for all

$X, X' \in \mathbb{X}$; (b) $d(X, X) = 0$ for all $X \in \mathbb{X}$; (c) small values of $d(X, X')$ indicate on a large degree of similarity between X and X'; (c$'$) large values of $d(X, X')$ indicate on a large degree of dissimilarity between X and X'; (d) $d(X, X') = \infty$ if X and X' can be considered as unrelated (or totally dissimilar).

If $\mathbb{X}$ is a discrete set with all features $X = (x_1, \ldots, x_m) \in \mathbb{X}$ given on the nominal scale then we can use the Hamming distance

$$d(X, X') = \sum_{j=1}^{m} \delta(x_j, x_j'), \quad \delta(x_j, x_j') = \begin{cases} 1 & x_j = x_j', \\ 0 & x_j \neq x_j', \end{cases}$$

or the weighted Hamming distance $d(X, X') = \sum_{j=1}^{m} w_j \delta(x_j, x_j')$, where the coefficients w_j are positive and proportional to the importance of the j-th feature (factor), $j = 1, \ldots, m$.

The purpose of the strategy for maximizing the CTR is to adapt the feature sets for the new requests we will be showing the ad to increase p_N as N increases. Formally, if we assume that $N \to \infty$ then our aim is devising a strategy such that $\lim_{N\to\infty} p_N$ is maximum. In practice, we are given N_{total}, the total number of requests to be exposed to an ad. Correspondingly, we want to maximize $p_{N_{total}}$.

The natural adaptive strategy is an evolutionary one which prefers new requests in the vicinity of the requests that were successful previously, i.e. which follow the look-alike idea. To define the preference criterion, for all N we need an estimator $\hat{p}_N(X)$ of the function $p(X)$, which is defined for all $X \in \mathbb{X}$. We do not need to construct the function $\hat{p}_N(X)$ explicitly; we just need to compute values of $\hat{p}_N(X)$ for a given X, where X is a request which is currently on offer for us. We hence suggest the following estimator $\hat{p}_N(X)$:

$$\hat{p}_N(X) = \frac{\sum_{k=1}^{K} \exp\{-\lambda_N d(X, X_{j_k})\}}{\sum_{i=1}^{N} \exp\{-\lambda_N d(X, X_i)\}} + \varepsilon_N \,, \tag{1}$$

where λ_N and ε_N are some positive constants (possibly depending on N). The sum in the numerator in (1) is taken over all users which have clicked on the ad. If all these (good) requests are far away from X then the value $\hat{p}_N(X)$ will be very close to zero. The constant ε_N is a regularization constant. As $\varepsilon_N > 0$ there is always a small probability assigned to each X, even if in the past there were no successful requests that were similar to X. Theoretically, as $N \to \infty$, we may assume that $\varepsilon_N \to 0$.

Alternative way of determining the estimator of $p(X)$ is the logistic model constructed by the FFM and FTRL approaches [11,15] or the tree-based model constructed by the GBM methodology [7].

Using an estimator $\hat{p}_N(X)$ for $p(X)$, we can suggest how much the advertising platform can offer for the request X in the bidding procedure. For example, the demand side can offer larger bids if $\hat{p}_N(X) \geq p_*$, where p_* is the desired probability we want to reach. Another strategy: the amount of money the advertising platform offer for X is proportional to the difference $\hat{p}_N(X) - K/N$, if this difference is positive.

In the strategy above, we can remove old data from the sample by always keeping the sample size equal to N_0 (assuming $N_0 < N$); in this case the estimator (1) changes to

$$\hat{p}_{N,N_0}(X) = \frac{\sum_{k=1}^{K} 1_{[j_K \geq N-N_0]} \exp\{-\lambda d(X, X_{j_k})\}}{\sum_{i=N-N_0}^{N} \exp\{-\lambda d(X, X_i)\}} + \varepsilon; \qquad (2)$$

in this estimator there is no need to change λ and ε as the sample size is constant (it is always equals N_0). In (2), $1_{[j_K \geq N-N_0]}$ is the indicator of the event $j_K \geq N - N_0$.

3 Analysis of Real Data

Since descriptive statistics for big data are important tools for understanding the data structure, see [1], we show some figures for two ad campaigns named as ad campaign 1 and ad campaign 2. For different subsets of data, we depict the estimator of the CTR computed as $\hat{p} = K/N$ with the 95 %-confidence interval $(\hat{p} - 1.96\sqrt{\hat{p}(1-\hat{p})/N}, \hat{p} + 1.96\sqrt{\hat{p}(1-\hat{p})/N})$, where K is the number of clicks and N is the number of impressions in the selected subset. We use the descriptive statistics to study the influence of each factor on the CTR that helps us to reduce the number of factors for the adaptive strategy.

The estimated CTR for all data is $\hat{p} = 1.7 \cdot 10^{-4}$ for ad campaign 1 and $\hat{p} = 2.4 \cdot 10^{-4}$ for ad campaign 2. These values will serve as a baseline for comparing the CTRs for different subsets of data.

3.1 Descriptive Statistics of the CTR for Two Ad Campaigns

In Fig. 2 we show the CTR on different days. We can see that CTR slightly depends on days. We can observe that the largest CTR of ad campaign 1 was on Dec 20 and the few preceding days, which can be explained by Christmas shopping. The CTR of ad campaign 2 is larger at weekends since the structure of bid requests is different at weekends.

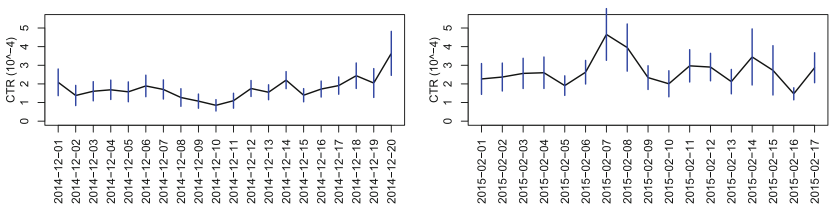

Fig. 2. The click through rate multiplied by 10^4 at different days for ad campaign 1 (left) and ad campaign 2 (right).

In Fig. 3 we show the CTR at different hours. We can see that the CTR for ad campaign 1 is larger from 22:00 to 22:59, which can be explained by activity of certain group of users. The CTR for ad campaign 2 is higher from 9:00 to 9:59 and from 19:00 to 19:59, when a group of users usually use internet in the morning and the evening.

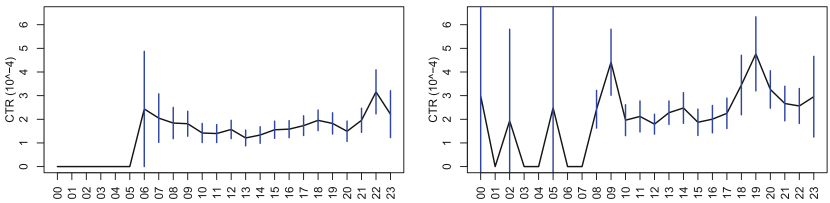

Fig. 3. The click through rate multiplied by 10^4 at different hours for ad campaign 1 (left) and ad campaign 2 (right).

In Fig. 4 we can see that the CTR is nearly the same for many websites except very few websites where the CTR is larger. It is quite natural that the largest CTR is for the website http://www.preloved.co.uk, which is a large classified advertising site. Another large CTR occurs for the website http://www.express.co.uk, which is a portal of the newspaper “Sunday Express”; however, the confidence interval is wide because the number of impressions is small. It is worth noting that the CTR for the websites of other newspapers, “Independent” and “Telegraph”, is very close to the average value.

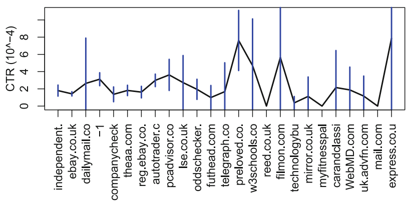

Fig. 4. The click through rate multiplied by 10^4 for 25 websites with largest numbers of requests for the ad campaign 2.

In Fig. 5 we can observe that the CTR does not depend on ad exchange but depends on the user agent. Specifically, the CTR is larger than average for MSIE and smaller than average for Safari.

In Fig. 6 we can see that the CTR for some cities and postcodes significantly differs from the average value. In particular, we can observe that the CTR for London is large but the CTR for Uxbridge and Trowbridge is small. However, the largest CTR occurs for the postcode PO standing for Portsmouth but the number of requests with postcode PO is quite small. The second largest CTR is for the postcode EC standing for Eastern Central, an area in central London. Also the CTR is well above average for postcodes CV (Coventry) and BN (Brighton).

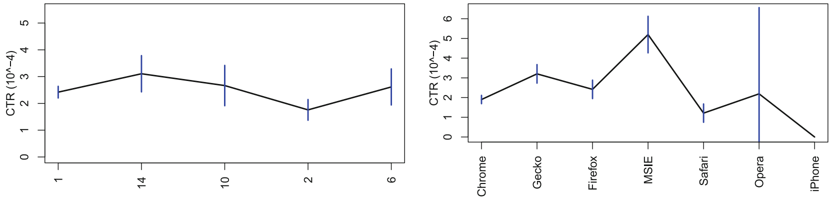

Fig. 5. The click through rate multiplied by 10^4 for different ad exchanges and user agents for the ad campaign 2. The x-tick labels of the left plot are the identification numbers of ad exchanges.

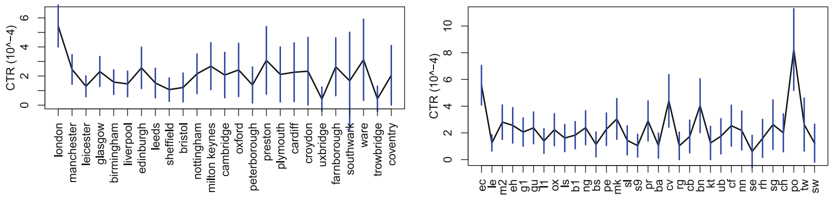

Fig. 6. The click through rate multiplied by 10^4 for 25 cites and 33 postcodes with largest numbers of requests for the ad campaign 2.

3.2 Multidimensional Scaling

In Fig. 7 we show the multidimensional scaling (MDS) performed by the SMACOF algorithm (see [10]) where we have used the Hamming distance for measuring the closeness between points $X, X' \in \mathbb{X}$. The MDS finds the association between the original high-dimensional points and points in a smaller dimension preserving the similarity of distances between points. Formally, the MDS for the dimension 2 is a solution of

$$\min_{z_1,\dots,z_n \in \mathbb{R}^2} \sum_{i=1}^{n} \sum_{j=1}^{n} (D_{ij} - ||z_j - z_i||_2)^2$$

where $D_{i,j}$ is the distance between the i-th request and the j-th request. We considered the set $\mathbb{X}$ with 7 factors: website, ad exchange, city, postcode, device type, user agent, user behaviour category. Since the multidimensional scaling is a hard computational problem we extract a subsample of the points from the data. We repeated the MDS for different subsamples and found that the pattern of 2D points very much repeats.

In Fig. 8 we show the supervised multidimensional scaling proposed in [18] with $\alpha = 0.2$, which is a solution of

$$\min_{z_1,\dots,z_n \in \mathbb{R}^2} \frac{1-\alpha}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(D_{ij} - ||z_j - z_i||_2\right)^2 + \alpha \sum_{i,j: y_j > y_i} (y_j - y_i) \sum_{s=1}^{2} \left(D_{ij} - (z_{js} - z_{is})\right)^2,$$

where $y_i = 1$ if the i-th request led to the click and $y_i = 0$ otherwise. Certainly, the supervised multidimensional scaling gives better separation between the

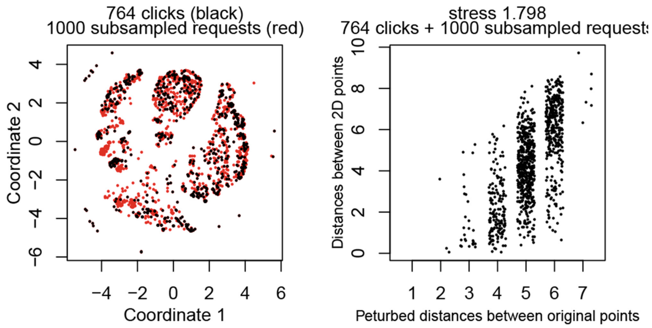

Fig. 7. The multidimensional scaling by SMACOF algorithm.

two groups: the users that have clicked on the ad and the users that haven't. However, the results of supervised scaling are hard to use in the adaptive strategy considered above. On the other hand, the classification obtained from unsupervised scaling are easy to use in such procedures.

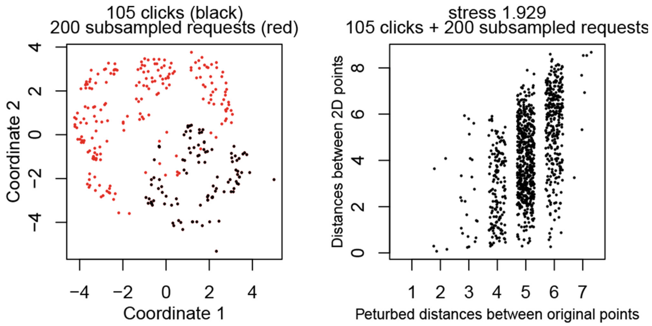

Fig. 8. The supervised multidimensional scaling.

3.3 Evaluation of the Adaptive Strategy

To investigate the performance of the adaptive strategy for the database of requests for ad campaign 2, we split the database of impressions into 2 sets: the training set $\mathbb{X}_p(T)$ of past records with dates until the certain time T (where T is interpreted as the present time) and the test set $\mathbb{X}_f(T)$ of future records with dates from the time T. We also define the set

$$L(r) = \{X_j \text{ from } \mathbb{X}_p(T)\text{: } \min_{clicked\ \tilde{X}_i \in \mathbb{X}_f(T)} d(X_j, \tilde{X}_i) \le r\};$$

that is, $L(r)$ is a set of requests where we have shown the ad and the minimal distance to the set of clicked requests from the set of past records is not greater than r. In other words, the set $L(r)$ is an intersection of the set of our requests with the union of balls of radius r centered around the clicked past requests. Here we also consider X_j with 7 factors: website, ad exchange, city, postcode, device type, user agent, user behaviour category.

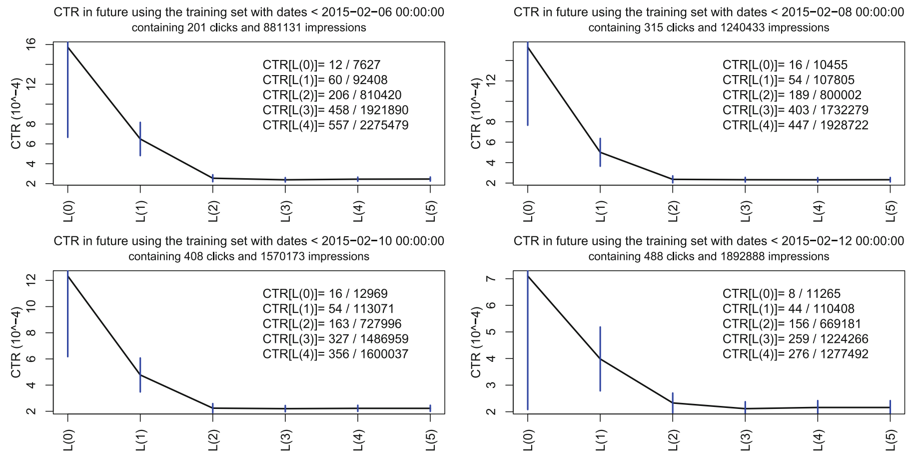

Fig. 9. The click through rate multiplied by 10^4 for the sets $L(r)$, $r = 0, 1, \ldots, 5$, for several values of T.

In Fig. 9 we show the click through rate for the sets $L(r)$, $r = 0, 1, \ldots, 5$, for several values of T. Recall that the ad campaign 2 starts on 2015-02-01 and finishes on 2015-02-17.

It is natural that the CTR for the set $L(r)$ decreases as r increases. We can observe that the CTR for $L(0)$ and $L(1)$ is very large but the number of impressions from $L(0)$ and $L(1)$ is small.

To be specific, for the time moment T=2015-02-08 the size of the set $L(r)$ is 10455 for $r = 0$, 107805 for $r = 1$, 800002 for $r = 2$, 1732279 for $r = 3$, and 1928722 for $r = 4$; and the number of clicked impressions in the set $L(r)$ is 16 for $r = 0$, 54 for $r = 1$, 189 for $r = 2$, 403 for $r = 3$ and 447 for $r = 4$.

Overall we can see that the CTR for $L(1)$ is significantly larger than the CTR for $L(2)$ at all times T.

3.4 The CTR for Different Choices of Factors

Let us perform the sensitivity analysis of the CTR for sets $L(r)$ for the ad campaign 2. In Table 1 we show the CTR for several sets $L(r)$ with T=2015-02-08 and different choices of factors. We can observe that the device type has no influence and the ad exchange has a small influence on the CTR for sets $L(0)$ and $L(1)$, consequently such factors can be removed from the model (and computations). The postcode has no influence on the CTR for the set $L(0)$ but has some influence on the CTR for the set $L(1)$.

Table 1. The CTR multiplied by 10^4 for several sets $L(r)$ with T=2015-02-08 and different choices of factors. Abbreviation of factors are Be:behaviour category, We:website, Ex:ad exchange, Ci:city, Po:postcode, De:device type, Ag:user agent.

Factors	CTR[$L(0)$]	CTR[$L(1)$]	CTR[$L(2)$]	CTR[$L(3)$]	CTR[$L(4)$]
Be,We,Ex,Ci,Po,De,Ag	15.3	5.01	2.36	2.33	2.32
We,Ex,Ci,Po,De,Ag	5.13	2.43	2.35	2.33	2.33
Be, Ex,Ci,Po,De,Ag	11.69	2.81	2.35	2.31	2.33
Be,We, Ci,Po,De,Ag	12.29	3.89	2.31	2.29	2.33
Be,We,Ex, Po,De,Ag	7.62	2.46	2.32	2.32	2.33
Be,We,Ex,Ci, De,Ag	14.96	2.45	2.32	2.32	2.33
Be,We,Ex,Ci,Po, Ag	15.27	5.09	2.38	2.33	2.32
Be,We,Ex,Ci,Po,De	4.87	3.37	2.20	2.33	2.33
Be,We,Ex,Ci, Ag	14.93	2.48	2.32	2.32	2.33
Be,We, Ci, Ag	11.99	2.38	2.29	2.33	2.33
Be,We, Ci,Po, Ag	12.27	3.88	2.34	2.29	2.33

In contrast, the user agent, the user behaviour category, and the city are very influential factors. It is very surprising that the postcode has no influence but the city has a big influence on the CTR for the set $L(0)$. However, the postcode is highly important to have the large value of the CTR for the set $L(1)$.

4 Conclusions

We have considered the problem of maximizing the CTR from the view-point of an advertising platform working with independent ad exchanges. We have discussed and studied an adaptive strategy which is based on the look-alike idea. We have tested the performance of the strategy. In particular, we have found out that the strategy of showing ads to requests from the set $L(1)$ yields the CTR which is 2.5 times larger than the CTR for the original ad campaign.

Acknowledgement. The paper is a result of collaboration of Crimtan, a provider of proprietary ad technology platform and University of Cardiff. Research of the third author was supported by the Russian Science Foundation, project No. 15-11-30022 “Global optimization, supercomputing computations, and application”.

References

1. Abello, J., Pardalos, P.M., Resende, M.G. (eds.): Handbook of Massive Data Sets, vol. 4. Springer Science and Business Media, New York (2002)
2. Aly, M., Hatch, A., Josifovski, V., Narayanan, V.K.: Web-scale user modeling for targeting. In: Proceedings of the 21st International Conference Companion on World Wide Web, pp. 3–12. ACM (2012)

3. Chakraborty, T., Even-Dar, E., Guha, S., Mansour, Y., Muthukrishnan, S.: Selective call out and real time bidding. In: Saberi, A. (ed.) WINE 2010. LNCS, vol. 6484, pp. 145–157. Springer, Heidelberg (2010)
4. Edelman, B., Ostrovsky, M., Schwarz, M.: Internet advertising and the generalized second price auction: selling billions of dollars worth of keywords. Am. Econ. Rev. **97**(1), 242–259 (2007)
5. eMarketer: US programmatic ad spend tops $10 Billion this year, to double by 2016 (2014). http://www.emarketer.com/Article/US-Programmatic-Ad-Spend-Tops-10-Billion-This-Year-Double-by-2016/1011312
6. Evans, D.S.: The online advertising industry: economics, evolution, and privacy. J. Econ. Perspect. **23**(3), 37–60 (2009)
7. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. **29**(5), 1189–1232 (2001)
8. Jansen, B.J., Mullen, T.: Sponsored search: an overview of the concept, history, and technology. Int. J. Electr. Bus. **6**(2), 114–131 (2008)
9. Google: the arrival of real-time bidding. Technical report (2011)
10. de Leeuw, J., Mair, P.: Multidimensional scaling using majorization: SMACOF in R. SMACOF. J. Stat. Softw. **31**(3), 1–30 (2009)
11. McMahan, H.B.: Follow-the-regularized-leader and mirror descent: equivalence theorems and L1 regularization. In: International Conference on Artificial Intelligence and Statistics, pp. 525–533 (2011)
12. McMahan, H.B., Holt, G., Sculley, D., et al.: Ad click prediction: a view from the trenches. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1222–1230 (2013)
13. Muthukrishnan, S.: Ad exchanges: research issues. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 1–12. Springer, Heidelberg (2009)
14. Nicholls, S., Malins, A., Horner, M.: Real-time bidding in online advertising (2014). http://www.gpbullhound.com/wp-content/uploads/2014/09/Real-Time-Bidding-in-Online-Advertising.pdf
15. Rendle, S.: Factorization machines. In: 2010 IEEE 10th International Conference on Data Mining (ICDM), pp. 995–1000. IEEE (2010)
16. Tu, S., Lu, C.: Topic-based user segmentation for online advertising with latent dirichlet allocation. In: Cao, L., Zhong, J., Feng, Y. (eds.) ADMA 2010, Part II. LNCS, vol. 6441, pp. 259–269. Springer, Heidelberg (2010)
17. Wang, J., Yuan, S., Shen, X., Seljan, S.: Real-time bidding: a new frontier of computational advertising research. In: CIKM Tutorial (2013)
18. Witten, D.M., Tibshirani, R.: Supervised multidimensional scaling for visualization, classification, and bipartite ranking. Comput. Stat. Data Anal. **55**(1), 789–801 (2011)
19. Yang, S., Ghose, A.: Analyzing the relationship between organic and sponsored search advertising: positive, negative or zero interdependence? Mark. Sci. **29**(4), 602–623 (2010)
20. Yuan, S., Wang, J., Zhao, X.: Real-time bidding for online advertising: measurement and analysis. In: ADKDD (2013)