Finding the Right Consumer: Optimizing for Conversion in Display Advertising Campaigns

Yandong Liu‡*, Sandeep Pandey†, Deepak Agarwal†, Vanja Josifovski†

‡ Carnegie Mellon, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA
† Yahoo! Research, 4301 Great America Parkway, Santa Clara, CA 95054, USA
yandongl@cs.cmu.edu | {spandey | dagarwal | vanjaj}@yahoo-inc.com

ABSTRACT

The ultimate goal of advertisers are *conversions* representing desired user actions on the advertisers' websites in the form of purchases and product information request. In this paper we address the problem of finding the right audience for display campaigns by finding the users that are most likely to convert. This challenging problem is at the heart of display campaign optimization and has to deal with several issues such as very small percentage of converters in the general population, high-dimensional representation of the user profiles, large churning rate of users and advertisers. To overcome these difficulties, in our approach we use two sources of information: a *seed* set of users that have converted for a campaign in the past; and a description of the campaign based on the advertiser's website. We explore the importance of the information provided by each of these two sources in a principled manner and then combine them to propose models for predicting converters. In particular, we show how seed set can be used to capture the campaign-specific targeting constraints, while the campaign metadata allows to share targeting knowledge across campaigns. We give methods for learning these models and perform experiments on real-world advertising campaigns. Our findings show that the seed set and the campaign metadata are complimentary to each other and both sources provide valuable information for conversion optimization.

Categories and Subject Descriptors

H.4.m [**Information Systems**]: Miscellaneous

General Terms

Algorithms, Performance, Experimentation

Keywords

conversions, modeling, advertising

*Work done while at Yahoo! Research

WSDM'12, February 8–12, 2012, Seattle, Washington, USA.

1. INTRODUCTION

Businesses, small and large alike, seek to expand by reaching out to users who can be their potential customers. Online advertising is becoming one of the main advertising channels, accounting for estimated $80 billion spending in 2011 and growing at the rate of 17% annually,[1] with display advertising being a big part of it. To make their campaign effective, display advertisers target specific users based on their historical pattern of activity or behavior, i.e., *behavioral targeting*. Different advertisers target different kind of users, e.g., a cellular company would be interested in users looking to subscribe to a cell phone plan or buying a handset, an online trading/investing company looks for finance-savvy users interested in buying/selling of shares, a travel agency wants to find customers for purchasing of flight tickets and booking of hotel rooms. While in some cases targeting criteria can be specified as a simple condition/function of the past user behavior, to get the best performance, increasingly sophisticated modeling techniques are being applied to detect the behavior patterns indicative of the user interest in a particular campaign/product.

Behavioral targeting aids advertisers in finding the right audience for their campaigns by characterizing and targeting users who fit their needs. Intermediaries such as ad networks, online exchanges and demand side platforms, perform behavioral targeting by bringing in the three parties involved, users, publishers and advertisers. While the complete details of advertising ecosystem is beyond the scope of this paper, a simplified view showing the interaction between advertisers and publishers is given in Figure 1. Here we use the term *ad broker* to refer to the intermediaries that (a) facilitate the collection of user data to build profiles and (b) select the best advertising campaign to display on a given Web page being viewed (impression) by a user on the publisher site. The ad broker constructs profiles for users based on their past online activities such as Web pages viewed at the participating publishers, Web search queries, vertical searches, etc. These profiles are then leveraged to learn advertiser-specific segment/model describing the desired target audience in terms of the pattern of user behavior. The focus of this paper is on producing such effective models for display advertising.

Most prior work on building behavioral targeting models focuses on maximizing clicks [6, 22], that is, construct models to identify those users who are most likely to click on ads when shown. While clicks serve as a natural proxy for

[1]According to a study by emarketer.com.

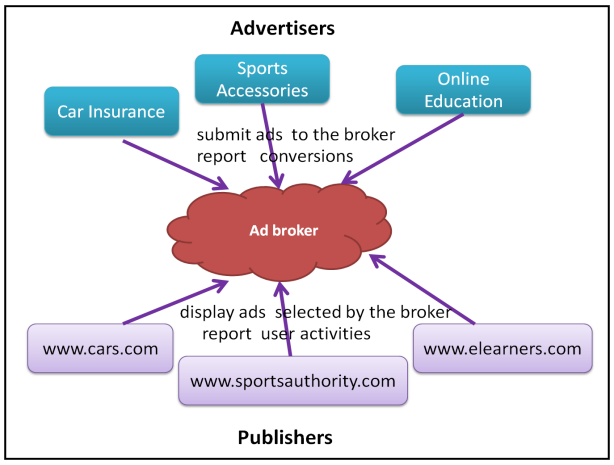

Figure 1: Behavioral Targeting.

user's interest in the advertised product, they can often be misleading, e.g., click fraud, bounce clicks [7, 12, 23]. Hence, in this work our goal is to target users for *conversions*, representing user activity on the advertisers website beyond the ad click [2, 4, 14]. Conversions are more tangible indication of user interest in the advertiser than clicks, separating the incidental and casual interest from purchasing intent.

Building conversion models is extremely challenging for many reasons. Usually, only a very small portion of the users that click eventually convert and thus, conversions are very rare events. This constrains the modeling techniques to parsimoniously work with the data. On the user side, user profiles are high dimensional consisting of several different kinds of activities, ranging from user demographics to search queries and page browsing. Dealing with such different activities in the presence of limited target/conversion information is non-trivial. To add to this, the data is highly volatile due to cookie churn, changes in campaigns, variability in user interests and other temporal effects that do not allow accumulating long-standing data and require the modeling approach to have a quick start and dynamically adapt over time as new data comes in.

Our Approach.

In view of these challenges, we propose a novel approach for conversion prediction that relies on two distinct sources of information: (a) the *metadata* associated with the advertising campaign such as ad creative, landing page etc.; and (b) *seed users* who have converted or viewed ads for the advertiser in the past. These two sources are complimentary in the way they guide the modeling process. For a new advertiser, since its ads have not been shown by the advertising network yet, the ad broker does not have any record of past converted users. As a result, initially, the ad broker must rely on using campaign's metadata to find right users for it. The campaign metadata quickly helps in understanding what the advertising campaign is about and thus identifying the potential targeting set. The metadata can be leveraged both in an unsupervised manner (e.g., target sports enthusiasts for sports related advertising) and supervised manner (e.g., pool the seed sets of campaigns using campaign metadata; more details in Section 3).

Subsequently, when the network has shown ads to enough users for the campaign, some of these users would have converted. This information can be used in refining the initial model: the converted users make the positive instances, while the unconverted users can be treated as the negatives. This can be modeled as a regression/classification task; predicting the conversion likelihood based on user profiles.

We give a principled approach to combine the two sources of information and propose a series of models. In particular, we show how seed set can be used to capture the campaign-specific targeting constraints (*local* component), while the campaign metadata allows to share targeting knowledge across campaigns (*global* component). For example, a "nike" campaign can learn/teach which users to target from/to an "adidas" campaign. Also, we give methods for learning these models in a joint manner that simultaneously optimizes for the local and global components and a two-step approach that performs this optimization in two stages.

In doing so we investigate several technical questions. For example, how do we represent users and campaign in a succinct manner? How useful are the two sources of information and how do they interact with each other? How can we combine them both in a principled manner? We answer these questions and make interesting observations through real advertising data collected from a large ad network. For example, we found that contrary to popular belief, learning models for large campaigns (in terms of number of conversions) can be more difficult compare to the smaller campaigns and the metadata can help in dealing with this issue.

Contributions.

In summary, we make the following contributions in this paper:

- We propose to predict conversions using: campaign metadata and seed users. We examine their relative value for campaigns with different characteristics (e.g., large and small, new and old).
- We give a set of modeling techniques that combine the two sources of information for optimal performance (through the local and global component). To the best of our knowledge, this is the first study on behavioral targeting to model the global component.
- Using the campaign metadata, we propose a method to bootstrap the prediction model for a campaign by exploiting information from related campaigns.
- We conduct extensive experiments and report the results from a real-life advertising dataset, to confirm the validity of our approach.

Finally, we note that, although the experiments are focused on conversion maximization for performance-based display advertisers, the principles described are applicable in a broader context as user profiles are the basis for audience selection in almost any setting of online advertising targeting and content personalization.

2. PROBLEM DEFINITION

As in traditional brand and performance advertising, display advertisers aim to *target* the users that might be interested in their products to promote their brand or get a direct response from the users. Depending on the *brand* or *performance* inclination, the advertisers can set up their

campaigns using different goals. While brand advertisers are primarily interested in number of ad views (impressions) by the targeted audience, performance advertisers usually set up either click or conversions goals. In this paper we focus on performance advertisers with **conversion optimization goals**. (More details on performance-advertising and the technical difficulties of it are given in Section 5.1).

Mathematically, the conversion optimization task can be formulated as the following. Let $c \in \mathcal{C}$ denote a campaign and $\boldsymbol{z}_c$ be the feature representation of its metadata. Let $seed_c$ denote the seed set for campaign c with labeled converters and non-converters. For each campaign $c \in \mathcal{C}$, the goal is to learn a model to differentiate between converters and non-converters. Let $\boldsymbol{x}_u$ denotes the feature profile of user u. In our example this is a high dimensional vector where each co-ordinate is binary. We want to learn a function $f(\boldsymbol{x}_u, \boldsymbol{z}_c, c)$ that helps us estimate the propensity of user u to convert on campaign c.

We can achieve this through a classification approach where we learn a function f that classifies a user u as a converter for campaign c if $f(\boldsymbol{x}_u, \boldsymbol{z}_c, c) > T$, T is a threshold (could be campaign specific) that is decided based on the cost of false positive and false negative. An alternate approach could learn f as odds of user u to convert on campaign c. The output scores can then be used to perform classification.

We discuss the class of functions f in the modeling section 3. Next we discuss how to derive the user and campaigns vectors ($\boldsymbol{x}_u$ and $\boldsymbol{z}_c$) that can be used in the subsequent modeling tasks.

2.1 User Representation

As in the offline advertising, to infer user interests, user profiles are constructed from known past user activity. User activity is tracked by the advertisers, publishers and third-parties through browser cookies that uniquely identify the user. For each user, the ad broker may store the history of page visits, ad views, and search queries, and based on the content of these events compose the profile used for targeting. Most of these events have textual content that can be analyzed using established text processing techniques. For example, the text of search queries issued, ids of ads viewed, the content of the pages viewed, etc. To represent event content we employ the bag of words method, which uses unigrams and bigrams, as well as nodes of a topical taxonomy that represent more general text categories. This gives us a feature vector representing a user, $\boldsymbol{x}_u$, for modeling purposes (see Figure 2). The weight of each feature can be binary or it can be computed based on its intensity in the user activities.

2.2 Campaign Representation

As mentioned before, we employ two sources of information to represent campaigns, as shown in Figure 2. First, we derive metadata from the campaign definition. Each campaign is composed of multiple *ad creatives*. An ad creative is an image or text snippet that is displayed to the user. Upon a click on the ad, the user is taken to a web page associated with this creative, also called a *landing page*. The creatives and the landing pages give a succinct characterization of the advertising campaign, and they can be useful to infer the domain of the campaign. In our approach, we construct a campaign metadata feature vector, $\boldsymbol{z}_c$, using the creatives and landing page content. One of the challenges in creating this feature vector is that campaigns can have multiple creatives and also creatives can be associated with multiple campaigns (see Figure 2). To be able to produce reasonable campaign metadata, we considered several variants and in our experiments we adopted the approach of merging the content of all landing pages connected to a campaign as its source of features. (More details are given in the experiment section.)

The second data source that characterizes the campaign is the set of *seed* users. The seed set is composed of *positive* and *negative* examples with regard to the given campaign. The positive set is composed of users that have converted for this campaign in the past, while the negative set represents the non-converted users. We will provide more details on this in our experimental evaluation section.

For new campaigns, there is no seed set available, and thus we must rely on campaign metadata to characterize the targeting requirements of the campaign. Over time the campaign expands as it is exposed to more and more users. Note that each user in the seed set comes at the cost of allocating one or more ad impressions to her.

3. MODELING APPROACHES

Recall from Section 2 that our goal is to learn function $f(\boldsymbol{x}_u, \boldsymbol{z}_c, c)$ that helps us estimate the propensity of user u to convert on campaign c. We confine ourselves to a class of functions f that can be decomposed additively as $f(\boldsymbol{x}_u, \boldsymbol{z}_c, c) = g(\boldsymbol{x}_u, \boldsymbol{z}_c) + f_c(\boldsymbol{x}_u)$, where g is a function of user features but depends on campaign c only through metadata $\boldsymbol{z}_c$, f_c is a campaign-specific function of user features.

Function Class. We consider three choices for function class f in this paper: a) Linear Support Vector Machine (L-SVM), b) Logistic Regression (LR), and c) Naive-Bayes (NB). Both SVM and logistic regression are known to provide good performance in advertising and many other applications, Naive Bayes was chosen due to its ability to substantially reduce variance at the expense of incurring bias due to the independence assumption. In noisy conversion data, this provides a good baseline to compare other more advanced methods like Linear SVM and logistic regression.

Linear SVM. Let $y_{u,c}$ denote a binary indicator that takes value $+1$ if user u converts on campaign c, -1 otherwise. Then SVM estimates function f to minimize the hinge loss $\sum_{u,c} \max(0, 1 - y_{u,c} \cdot f(\boldsymbol{x}_u, \boldsymbol{z}_c, c))$ (call L_1 SVM), or squared squared hinge loss $\sum_{u,c} \max(0, 1 - y_{u,c} \cdot f(\boldsymbol{x}_u, \boldsymbol{z}_c, c))^2$ (called L_2 SVM). If f is too flexible, it tends to overfit the data and additional penalty term to constrain f is often used. Hence, we assume f to be linear in the known variables ($\boldsymbol{x}_u$ and $\boldsymbol{z}_c$).

Logistic Regression. The hinge loss in this case is replaced by the sigmoid loss function $\sum_{u,c} \log(1 + exp(-y_{u,c} \cdot f(\boldsymbol{x}_u, \boldsymbol{z}_c, c)))$. As in SVM, we confine ourselves to a linear function and perform appropriate regularization. One can impose either the L_1 or L_2 norm constraint on the unknown coefficients. The former also ensures sparse solutions, i.e., many irrelevant variables have zero coefficients.

Naive Bayes. Naive Bayes builds f by estimating the joint density of features (user and/or campaigns) separately in the converting and non-converting classes. By Bayes theorem,

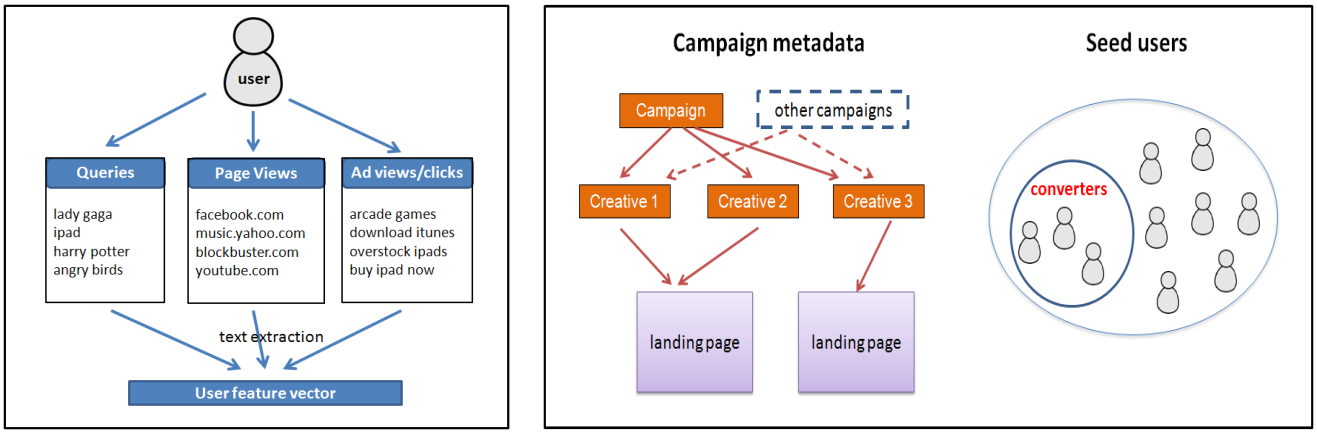

Figure 2: User and campaign representation.

the log-odds of probability of conversion is given by

$$log\frac{D_c(\boldsymbol{x}_u|y_{u,c}=1)}{D_c(\boldsymbol{x}_u|y_{u,c}=-1)}+log\frac{P(y_{u,c}=1)}{P(y_{u,c}=-1)}$$

where D_c denotes the appropriate joint density of user features in campaign c. The second term which is the prior log-odds of conversion for a campaign is constant when classifying a user as converter/non-converter on a given campaign, hence it can be ignored. Estimating the notoriously high-dimensional density is the crucial task here, Naive Bayes simplifies this through the independence assumption which ensures joint estimates as products of one dimensional marginals. Although the one dimensional marginals are estimated with precision and reduces variance, the bias incurred due to the independence assumption may degrade performance.

For a given function class f, next we describe how we use the seed set and the campaign metadata to build these models.

3.1 Local Models Using the Seed Set

Given that each campaign targets a different set of users, the first approach is to build a separate *local* model for each campaign. In particular, given the seed set, we exploit the positive and negative examples to learn a campaign-specific targeting function. In other words, we ignore $g(\boldsymbol{x}_u, \boldsymbol{z}_c)$ and assume $f(\boldsymbol{x}_u, \boldsymbol{z}_c, c) = f_c(\boldsymbol{x}_u)$. For training L-SVM and logistic regression, $f_c(\boldsymbol{x}_u) = \boldsymbol{x}'_u\,\boldsymbol{\beta}_c$, where $\boldsymbol{\beta}_c$ is an unknown vector that is to be estimated from training data. Formally stated, we obtain $\boldsymbol{\beta}_c$ as a solution to the following optimization problem.

$$\text{argmin}_{\boldsymbol{\beta}_c}\sum_{u\in\mathcal{C}}L(\boldsymbol{x}_u, y_{u,c}, \boldsymbol{\beta}_c)+\lambda_c\|\,\boldsymbol{\beta}_c\,\|_p \qquad (1)$$

where L is either the hinge $\max(0, 1-y_{u,c}\cdot\boldsymbol{x}'_u\,\boldsymbol{\beta}_c)$ or squared hinge $\max(0, 1-y_{u,c}\cdot\boldsymbol{x}'_u\,\boldsymbol{\beta}_c)^2$ for L-SVM, and sigmoid $log(1+exp(-y_{u,c}\cdot\boldsymbol{x}'_u\,\boldsymbol{\beta}_c))$ for logistic regression respectively. For L-SVM, we use $p=2$ but for logistic we use two values of p for the penalty: $p=1$ for L_1 regularization and $p=2$ gives L_2 regularization. The parameter $\lambda_c(\geq 0)$ determines the relative weight of the penalty term and is estimated through cross-validation. Larger values of λ_c implies more penalty on the parameters. We note that if the user vector $\boldsymbol{x}_u$ contains m features, the dimension of $\boldsymbol{\beta}_c$ is also m. This leads to too many parameters for large values of m, often the case in advertising applications. The lack of conversions per campaign further exacerbates the situation. In fact, in most settings the number of conversions could be much smaller than m. This makes penalization important to avoid overfitting. Hence the right choice of λ_c is essential for good performance (see Section 4.3). We select this parameter by extensive cross-validation.

Expanding $\boldsymbol{x}_u = (x_{u1}, \cdots, x_{um})$, for Naive Bayes $f_c(\boldsymbol{x}_u)$ is given by

$$f_c(\boldsymbol{x}_u)=\sum_{i=1}^{m}log\frac{D_i(x_{ui}|y_{uc}=1)}{D_i(x_{ui}|y_{uc}=-1)} \qquad (2)$$

where $D_i(x_{ui}|y_{uc})$ is the conditional marginal density of the i^{th} user feature in class y_{uc}. The estimation of marginal density $D_i(.|.)$ depends on the nature of the feature. In this paper, since all our user features are binary, this density is obtained simply by counting. For instance, $D_i(x_{ui}=1|y_{uc}=1)$ is simply the fraction of converters on campaign c who possess feature i. To avoid unreliable estimates for campaigns with small number of conversions and/or features that occur rarely, we perform mild smoothing. In particular, we estimate $D_i(x_{ui}=1|y_{uc}=1)$ as $\frac{|u:x_{ui}=1,y_{uc}=1|+(a\cdot p_{ic})}{|u:y_{uc}=1|+a}$, where a is a positive smoothing constant that could be interpreted as pseudo number of conversions, p_{ic} is the fraction of users who possess feature i in campaign c. We choose small values of a (e.g $a\in[1,5]$). Our experiments showed that Naive-Bayes is very robust and it showed little sensitivity in performance with respect to the choice of a.

We also note that learning local models is computationally efficient since the computation can be done separately for each campaign in parallel. All our computations with SVM and logistic regression were done with LIBLINEAR [19].

3.2 Global Models Using the Campaign Metadata

Per campaign local models work well for campaigns with large seed set, i.e., large number of conversions. For new campaigns or those with little training data, the performance is not satisfactory due to data sparsity. To mitigate this, we employ the campaign metadata information ($\boldsymbol{z}_c$). One option is to obtain user and campaign similarity

using their features as in traditional information retrieval. However, this does not work in our case since user and campaign features are not mapped to the same semantic space. The other idea is to exploit campaign metadata to correlate learning across campaigns and perform better prediction for those campaigns that lack enough data.

We explore several different ways of performing learning across campaigns in this paper. The first approach shares the model coefficients $\boldsymbol{\beta}_c$ for user features across campaigns. We call this the **Merge-based Global** model. The second approach extends the merge model to include an additional component that models user and campaign interaction through user features and campaign metadata. We call this the **Interaction-based Global** model. Finally, we explore our most complex model that extends the interaction model to include a campaign-specific local model, we call this the **Global+Local** model.

3.2.1 Merge-based Global Model

Here, we merge the seed users from all campaigns to learn a global model. In other words, $f_c(\boldsymbol{x}_u) = \boldsymbol{x}'\boldsymbol{\beta}$ for all campaigns c where $\boldsymbol{\beta}$ is the global weight vector. Thus, one single set of coefficients is estimated for all campaigns. This reduces the number of parameters dramatically and allows us to derive more precise estimates of the global coefficients. However, this model does not capture any user and campaign specific interactions. Instead, it captures the user's propensity to convert in general, which can be useful in many real scenarios. For example, the willingness of a user to conduct online transactions affects each campaign in the same way and can be learned/exploited globally.

To build this global model, we put together the seed sets of all campaigns. Even if the goal is to learn user's global propensity to convert, one has to be careful in training this model. For example, campaigns with large seed sets can easily dominate the global model and bias the model estimates. This is counter-effective since such a global model would not perform well on campaigns with small seed set, which are the ones that should benefit the most from such a collapsed model. We address this by weighing each campaign's seed set equally, both in the positive and negative classes.

3.2.2 Interaction-based Global Model

Merge model estimates the global propensity of a user to convert. However, it does not capture any affinity of campaigns to each other. For instance, if a kind of users has high propensity to convert on travel campaigns, we can perhaps use this information to recommend these users to a new travel campaign. One way to perform such cross campaign learning is by positing a linear model that is a function of both user and campaign features. We accomplish this through our **interaction** model by assuming $g(\boldsymbol{x}_u, \boldsymbol{z}_c) = \boldsymbol{x}'_u D \boldsymbol{z}_c$, where D is a matrix of unknowns to be estimated by pooling data across all campaigns. Hence, $f(\boldsymbol{x}_u, \boldsymbol{z}_c, c) = \boldsymbol{x}'_u D \boldsymbol{z}_c + \boldsymbol{x}'_u \boldsymbol{\beta}$. Mathematically, we obtain D by solving the following optimization problem.

$$\min_D \sum_{u,c} L(y_{u,c}, \boldsymbol{x}'_u D \boldsymbol{z}_c + \boldsymbol{x}'_u \boldsymbol{\beta}) + \lambda(\|D\|_p + \|\boldsymbol{\beta}\|_p) \quad (3)$$

Note that this optimization is performed by pooling data across all campaigns. Also, the number of parameters to be estimated in $(D,\boldsymbol{\beta})$ is $(m+1)\cdot n$, where m and n are the number of user and campaign features respectively. When m and/or n is large, this can lead to a high dimensional optimization problem that is difficult to solve. For instance, in our experiments we deal with $m = 70k$ and $n = 500$, this leads to an optimization problem consisting of $35M$ parameters. Also, the computations are not separable by campaigns (unlike Section 3.1). Further, several user and campaign features are non-informative and noisy. To keep the interaction matrix manageable, we use a simple a-priori feature filtering procedure that removes irrelevant user features (since the number of campaign features is relatively small, we do not perform any such filtering there). This filtering is performed through a Kullback-Liebler divergence measure, as described below.

Variable filtering. We describe a variable importance score for a user feature i. Let $q_{i,c}$ denote the conversion probability for feature i on campaign c and $q_{i\cdot}$ denote the overall conversion probability for feature i. We compute the values of $q_{i,c}$ and $q_{i\cdot}$ using their maximum-likelihood estimates, and to avoid unreliable estimates, we perform mild smoothing similar in spirit to that described in the context of Naive Bayes. The variable score for i is then given by the Kullback-Liebler divergence $\sum_c q_{i,c} \cdot log(\frac{q_{i,c}}{q_{i\cdot}})$. Note that a value of 0 implies no interaction of feature i with campaigns and hence this feature is not important to be included in our model.

3.3 Global + Local Model

Local models have the advantage that they capture the campaign-specific effect. Global models capture the global targeting constraints and have the advantage that they can generalize well even when there is lack of campaign-specific data. To get the best of both worlds, we build models that include both the global and local components. More specifically, we assume $f(\boldsymbol{x}_u, \boldsymbol{z}_c, c) = \boldsymbol{x}'_u D \boldsymbol{z}_c + \boldsymbol{x}'_u \boldsymbol{\beta}_g + \boldsymbol{x}'_u \boldsymbol{\beta}_c$ and solve for $(D, \boldsymbol{\beta}_g, \boldsymbol{\beta}_{c:c\in\mathcal{C}})$ by solving an appropriate L-SVM or logistic regression task. Mathematically, we obtain the parameters by solving the following optimization problem:

$$\min_D \sum_{u,c} L(y_{u,c}, \boldsymbol{X}'_u D \boldsymbol{z}_c + \boldsymbol{x}'_u \boldsymbol{\beta}_g + \boldsymbol{x}'_u \boldsymbol{\beta}_c) + \lambda\|D\|_p + \lambda\|\boldsymbol{\beta}_g\|_p + \sum_c \lambda_c \|\boldsymbol{\beta}_c\|_p$$

Here $\boldsymbol{X}_u$ are user features used for the interaction component after a-priori filtering, and so $\boldsymbol{x}_u$ can be different from $\boldsymbol{X}_u$. Note that this equation involves separate λs for each campaign and another λ for the transfer learning parameters. Obtaining so many tuning parameters when jointly optimizing $(D, \boldsymbol{\beta}_g, \boldsymbol{\beta}_{c:c\in\mathcal{C}})$ by pooling data across all campaigns is difficult in practice and computationally challenging.

We address the computational challenges by using two different approaches

- simultaneously learn the global and local components (i.e., *joint approach*) but assume $\lambda_c = \lambda$.
- learn the global component first, then fit a separate campaign-specific local model using $\boldsymbol{X}'_u \hat{D} \boldsymbol{z}_c + \boldsymbol{x}'_u \hat{\boldsymbol{\beta}}_g$ as a known constant where $\hat{D}$ and $\hat{\boldsymbol{\beta}}_g$ are the global component estimates (i.e., *offset approach*). This entails changing $L(\boldsymbol{x}_u, y_{u,c}, \boldsymbol{\beta}_c)$ to $L(\boldsymbol{x}_u, y_{u,c}, \boldsymbol{X}'_u \hat{D} \boldsymbol{z}_c + \boldsymbol{x}'_u \hat{\boldsymbol{\beta}}_g + \boldsymbol{\beta}_c)$ in Equation 1.

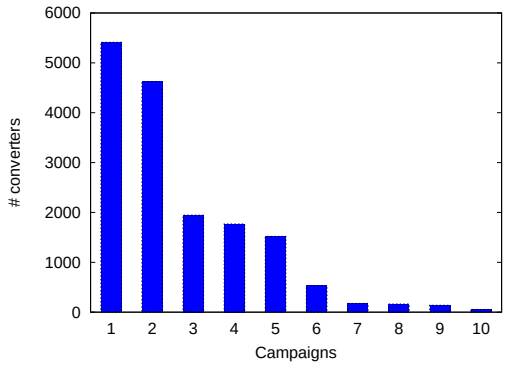

Figure 3: Campaign sizes in terms of the number of conversions.

This model performs well on both mature and new campaigns. For the latter, the transfer learning component based on campaign features plays an important role, while for the former the additional local term adds a correction that helps it converge to the local campaign models.

4. EXPERIMENTS

Next we evaluate our proposed models on a real dataset collected from a large advertising network.

4.1 Dataset

We constructed a dataset of user profiles, which are labeled as positive or negative depending on whether the user converted on a given campaign. Any potentially personally identifiable information was removed, and all the data was anonymized. All datasets were compliant with the company privacy policy. The users were drawn from 10 randomly selected display ad campaigns, which were registered on a major US advertising network in 2011. The dataset spans more than 300,000 users allowing us to draw meaningful conclusions from these experiments. All these campaigns are performance-based, i.e., advertisers only pay to the advertising network for actual conversions. Of the 10 campaigns, some are fairly small in terms of number of conversions (with 10 to 20 conversions per week, on average), while others are large and receive many thousands of conversions every week, as shown in Figure 3. For these campaigns we obtained a log of ad activity for the four weeks period from 03/18/2011 to 04/15/2011. The log contains fully anonymized ids of users who viewed, clicked, or converted on the ads from these campaigns.

For each campaign c we construct a dataset with those users that were shown one or more ads from the campaign during the study period. Using the conversion data described above we give binary labels to these user profiles as either positive or negative. Users who converted make the positive instances, while the rest make the negative examples. Since the number of negative examples can be huge (many millions) for large campaigns, we down-sample them to keep about 30,000 examples per campaign. We perform 3-fold cross validation on this dataset in our experiments and report the performance averaged over the 3 folds.

User profiles.

For each user observed in the period above, we take the four weeks of her online activity *preceding the conversion* to construct the user profile, as described in Section 2. These activities include page visits, ad views, and search queries. Note that while predicting a test instance, say on day t, we allow the prediction models to access user history up to day $t-1$. Hence, the prediction method is not using any future information.

Campaign metadata.

For the 10 campaigns in our experiments we collected the ad creatives associated with them (see Figure 2). This gives us about 15,000 creatives, of which most are images and do not have associated text. However, each creative contains a landing page that denotes the URL of the page which the user is directed to after clicking on the ad. We crawl each landing page, parse, and attribute the extracted content to its corresponding creative(s).

Since the relationship between campaigns and creatives is many-to-many, it is not clear how to propagate creative's content to a campaign. The experiments reported in this paper use a simple approach where, for a given campaign, we weigh all the connected creatives equally and put their content together. Another alternative is to differentiate the creatives and their contributions to a given campaign using the creative-campaign graph, e.g., give less weight to those creatives which are shared by more campaigns and vice-versa. We leave this exploration as a part of our future work. Finally, we perform some feature selection over campaign features using the number of landing pages a feature appears in.

Seed set.

We compose the seed set with positive and negative examples as described earlier in this section. Recall that the seed set denotes the labeled users that are available for training for a campaign at a given time. In our experiments we explore how the prediction power of the models depends on the size of the seed set. Thus, we simulate different sizes of the seed set by sampling the data from the full training seed set of a campaign. To explain further, say $local(c, x)$ denotes the local model built for campaign c using x number of positive examples (since negative examples are easy to acquire, even for a new campaign, we do not vary them). This allows us to simulate a campaign at different stages of its life. For example, to simulate a new campaign we set x to 0. To simulate slightly more mature campaigns, we can set x to 30 or 70 conversions. For the long-standing campaigns we set x to a large value which means that all the positives examples for this campaign in the training folds are used.

4.2 Evaluation Metric

We use the Receiver Operating Characteristic (ROC) curve to evaluate the ranked list of users produced by the different targeting models. A ROC curve plots true positives versus false positives for different classification thresholds. The area under the ROC curve is particularly interesting due to its probabilistic interpretation. The Area Under Curve (AUC) gives the probability that the audience selection method assigns a higher score to a random positive example than a random negative example (i.e., probability of concordance) [8, 10]. So, a purely random selection method will have an area under the curve of exactly 0.5. An algorithm that achieves AUC of 0.6 can distinguish a positive

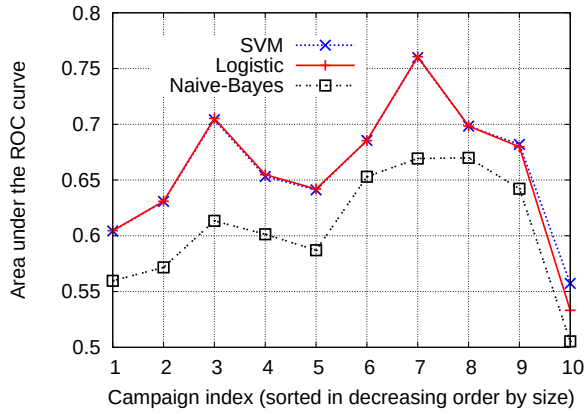

Figure 4: Performance comparison of SVM, Logistic and Naive-Bayes based local models.

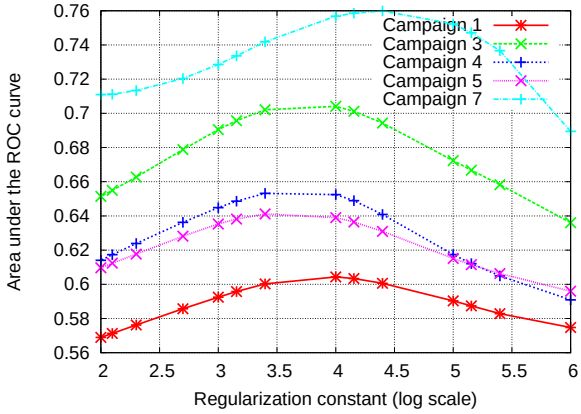

Figure 5: Effect of the regularization constant on SVM models.

user from a negative user with 60% probability, and is thus better than the random method by 20%.

An alternative metric could be to measure precision/recall at a certain rank in the list. Note that different campaigns may have different requirements in terms of precision and recall. For example, a small campaign whose reach is limited would prefer higher recall, while a large campaign that reaches out to many users might prefer higher precision). Consequently, selecting a rank at which to evaluate precision such that it would be suitable for all campaigns, is not possible. Instead, we use AUC since it combines the prediction performance over all ranks into a single number.

4.3 Local models using the seed set

For this experiment we build a separate model for each campaign using all the positive examples from the training folds. Hence, large campaigns will get to use more than 5000 conversions, while the small campaigns learn from some 500 conversions. First, in Figure 4 we show the performance comparison of SVM, Logistic (with L2 regularization) and Naive-Bayes models. The x-axis is the campaign index where the campaigns have been sorted in the decreasing order by the number of conversions (i.e., smaller indices denote larger campaigns). On the y-axis we plot the best AUC performance obtained for each campaign averaged over 3 folds after varying the model parameters. For SVM and Logistic we vary the regularization constant and for Naive-Bayes the smoothing constant was varied. We note that Logistic and SVM perform very similar, while Naive-bayes is slightly worse. However, we observed that Naive-Bayes was not too sensitive to any learning parameters (such as smoothing constant). The same is not true for SVM and Logistic. In Figure 5 we show the performance of the SVM models for 5 different campaigns (of the 10 campaigns in our dataset). Here the x-axis denotes the value of the regularization constant. As we can see that the performance depends quite significantly on the regularization constant. When the constant is too small, many useful features get eliminated due to severe penalization. On the other hand, when the constant is too large, the model starts overfitting and does not perform well on the test folds. For brevity we will use SVM to report the results for the remaining experiments.

Impact of the number of conversions in the training set.

We also note that the best performance for different campaigns can be quite different; AUC varies from 0.55 to 0.75 in Figure 4. To analyze this further we performed the following experiment. We varied the number of positive examples that were allowed for training while learning the model for a campaign. We call this $local(c, x)$ where c denotes the campaign and x denotes the *allowed* number of positive examples in the seed/training set. We note here that x is the upper-bound; we vary x up to 5000 but for small campaigns the seed set does not change beyond $x = 540$ since these are the maximum number of conversions we have for the small campaigns in our dataset. As mentioned earlier, in this experiment we aim to simulate a campaign at different stages of its life cycle. For new campaign x is close to 0, while for a mature campaign x can be in a range of a couple of hundreds depending on the size of the campaign. The results of this experiment are shown in in Figure 6. To avoid cluttering, we grouped campaigns by their sizes in the figure – the largest 2 campaigns in terms of the number of conversions were put into the "large" category, the middle 3 campaigns into the "medium" and the bottom 5 into "small" category.

On the first inspection the results look contrary to what one might expect: we see that the small campaigns not only outperform the large campaigns when x is small, but they also perform better for large values of x. For example, when $x = 5000$ large campaigns use all their 5000 conversions as positive examples for training, while small campaigns use $x = 540$ positive examples only and they still perform better. From a careful investigation of the data, we identified the following reason behind this. The small campaigns are quite restrictive in their definition of conversion (which is partly the reason behind them being small), e.g., they require email sign-up, click on the ad and order completion page for conversions. On the other hand, the large campaigns have more relaxed definition of conversion, e.g., conversions defined as a view of the ad followed by some kind of positive activity on the website. The converters for such large campaigns are very heterogeneous and noisy. As a result, even for large values of x (i.e., large number of positive examples in the training set) it is difficult to discriminate between converters and non-converters. Small campaigns, due

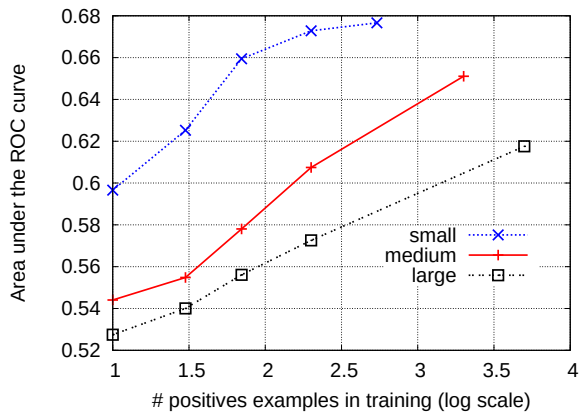

Figure 6: Performance evaluation of the local models for different number of positives (x) in the seed set. Note that for the small campaigns the maximum number of conversions is $x = 540$ in our dataset.

to their restrictive conversion definition and specific targeting segment, are easier to learn and have a quick start, i.e., even with a few positive examples the models can be learned significantly well.

4.4 Global Models

We start the evaluation of global models by studying the merge-based model. Recall from Section 3.2.1, merge model pools the seed sets from all campaigns while learning the model for a given campaign. In particular, let $merge(c, x)$ denote the merge model for campaign c when the seed set is of size x. To learn $merge(c, x)$, we employ all the data from the other campaigns, $\{\mathcal{C} - c\}$, and from campaign c we use x positive examples. This simulates the setting where campaign c is new and $\mathcal{C} - c$ are old/mature campaigns that the ad matching platform has run in the past.

The results are shown in Figure 7. We compare $merge(c, x)$ with $local(c, x)$ in the figure for small, medium and large campaign sizes. We note that for large campaigns the merge model performs better than the local model for small x values. In particular, till $x = 200$ conversions the merge model achieves higher AUC than the local model. This is expected in view of the previous section since we observed there that the large campaigns have slow start. Hence, using the seed sets of other campaigns allows the merge-based approach to learn more robust models. For small/medium campaigns the local models get better quite quickly (after 30 to 50 conversions). However, note that collecting 50 conversions may take a couple of days (if not week) for small vertically focused campaigns in a real-world environment where multiple advertisers compete for the attention of a relatively small set of targeted users.

Interaction-based global model.

Next we study the interaction-based global model from Section 3.2.2. We performed feature selection using KL-divergence to keep 3000 user features. On the campaign side, we kept 50 features per campaign by selecting the most frequent words in landing pages. In Table 1 we compare the interaction-based and merge-based global models. We note that overall the interaction approach is better, as expected.

Campaign size	Interaction-based model
Small	6.06%
Medium	0.4%
Large	3.93%

Table 1: Performance improvement achieved by the interaction-based model over merge-based.

Campaign size	Global	Global+Local
Small	-10.5%	**-8.5%**
Medium	**6.4%**	6.1%
Large	70%	**78.6%**

Table 2: Performance improvement of the global and global+local models over the local models for different campaign sizes.

We expect the improvement to increase with the number of campaigns pooled together, as that allows better learning of the weights for the user-campaign interaction features.

As shown in the table, the most gain comes for the small campaigns. Compare to the large campaigns which can have many thousand creatives and landing pages, the small campaigns have few creatives providing us with a high-quality homogeneous set of campaign features.

4.5 Global + Local Models

In this experiment we study the global+local models from Section 3.3. For brevity, we only focus on the joint optimization approach. In Table 2 we compare the joint model against the $local(c, x)$ and $merge(c, x)$ for $x = 30$. As discussed in Section 3.3, global+local model, in theory, subsumes both the local and global model and is a strict generalization over these models. In practice, this seems to hold true except for the small campaigns where the joint global+local model is slightly worse than the local model. We believe that this is due to the fact that the regularization constant is shared across all campaigns in the joint approach, while the local models have per-campaign tuning of regularization constant. In fact, we observed that if we force the local models to share the same regularization constant, then the performance difference between the local and joint models on the small campaigns is marginal.

More importantly, we observe that the global+local approach brings a large improvement (about 78%) over the local models on large campaigns. Recall that large campaigns suffer from slow start and so this is greatly beneficial to these campaigns. This demonstrates how the campaign metadata can be useful in sharing targeting knowledge across campaigns.

5. BACKGROUND AND RELATED WORK

First, we give some background on conversion optimization in display advertising. Then we compare our work against the existing literature.

5.1 Conversion Optimization in Display Advertising

Display advertising takes the form of graphical ads displayed on web pages alongside the original content. From

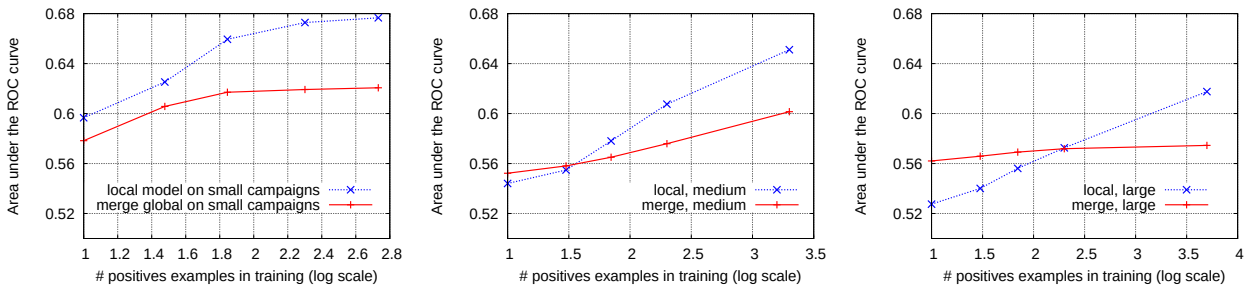

Figure 7: Performance comparison of local model with merge-based model for different number of positives (x) in the training set. The three figures are for the three different campaign groups (based on size).

its meager beginnings in the 1990's, display advertising has grown to an estimated $12.33 billion industry in 2011, according to the latest IDC report [21]. Display ads today are standardized in size, and created, sold, traded, and placed by an industry composed of hundreds of companies that include publishers, publisher optimizers, ad agencies, ad exchanges, ad networks, advertisers, and others.

Conversions are user actions that indicate the ad was successfully perceived by users such as making a purchase, requesting a price quote, signing up for an account, etc. Maximizing conversions is one of the key challenges for today's display ad brokers, posing several technical difficulties. First, conversions are very rare events. Only a small percentage of users who see an ad will click on the ad, and only a small percentage of those users convert. Conversions often occur in the range of one out of $10^5 - 10^6$ ad views. Furthermore, the conversions represent a diverse set of events and there is no single definition for conversions as it varies among the advertisers. For example, certain advertisers define conversion as the event when a user purchases a product, while others may call a subscription to mails/alerts or the filling of a form as conversion. Thus the conversion rate can vary significantly across different advertisers.

Lastly, conversions are tracked through *conversion pixel* which is a javascript code embedded in the advertiser's conversion page (e.g., order completion page). The code gets triggered to notify the advertising network when a user who was shown ads by the advertising network reaches the conversion page. A campaign can encompass multiple conversion pixels in the creative landing pages. Conceptually, the conversions pixels represent *goals* of the campaign while creatives are the *means* of achieving those goals. There could be multiple means to achieve the same goal as well as the same mean can lead to multiple goals. This further complicates the issue making the task of predicting conversions even harder.

5.2 Prior Work

Our work is related to modeling of user behavior and targeting based on observed past events. User behavior has been studied to understand user's querying pattern [20], news browsing behavior [11], interests inferencing [17] and personalized search [18]. In contrast, our focus is on online advertising and behavioral targeting in particular. Initial behavioral targeting studies mostly focused on predicting clicks on ads [6, 22]. Clicks are used simply because they are available and other information is not available at a large scale. While clicks do represent user intent, they are known to suffer with fraud issues [7, 12, 23]. Recently, however, advertisers have been willing to share feedback (through conversion pixeling) at the level of individual users, telling publishers which of the users who saw the ad have actually purchased the product [2, 3, 4, 14]. Since conversions are the ultimate goal of advertisers, we focus on conversion optimization in this study.

Previous work on conversion optimization uses only the seed set and learns local models [2, 3, 4], but as we showed in our experiments that the model performance depends significantly on the size of the seed set. Hence, we employ campaign metadata along with the seed set and propose a series of global models that pool campaigns together using their metadata. When the campaigns are new and the seed set is small or unavailable, we show that the metadata can be significantly useful. To the best of our knowledge, this is the first study in behavioral targeting framework which employs such collaborative modeling. Our work is also related to the cold-start problem in collaborative filtering literature where movies/items play the role of campaigns [1, 9, 15, 16]. However, there are a couple of major differences: (a) users interact with and give thumbs up to many movies/items, while they do not convert on that many campaigns, (b) the number of campaigns is not of the order of the number of items. Both these limitations restrict the factor models in our setting [1, 15].

Another area of online advertising is social targeting where the goal is to identify users having strong influence over others [5]. Since brand affinity is likely to be shared between socially connected users, Provost et al. [13] identified regions of the social network that may be more susceptible to a given brand message. This approach relies on the existence of a social network, although in their work [13] the network was approximated using co-visitations.

6. CONCLUSIONS

Advertisers want more bang per buck and as a result, conversion optimization in display campaigns is getting increasingly more attention. The task is very challenging for several reasons such as very low conversion rate, high variance in number of conversions across campaigns, and diverse set of events logged as conversions. In addition, on the user side we also have sparse and noisy profile activities. In this paper we proposed a two-pronged approach for conversion optimization whereby we use a seed set of

converters to capture the campaign-specific or local targeting criteria (e.g., interests in finance, shares, mortgage), and the campaign metadata to share targeting knowledge across campaigns (i.e., global component). To learn the local models we experimented with SVM, Logistic and Naive-Bayes models. We showed that SVM and Logistic perform better than Naive-Bayes, however they are sensitive with respect to the setting of model parameters while Naive-bayes is fairly resilient. We found it surprising that campaigns with higher number of conversions are actually harder to model, due to the heterogeneity among the converted users.

To investigate the global models we proposed merge-based and interaction-based models for pooling together the information from different campaigns. We showed how global models improve the prediction performance for large campaigns which suffer from slow learning rate during the initial phase. Next we showed how our global+local models capture the best of both worlds. They include the campaign-specific weight vector for user features (local), the shared weight vector for user features (merge-based) and the interaction of user and campaign features (interaction-based). Also, to learn these models we give a joint optimization approach which simultaneously accounts for both the local and global components. The offset approach, on the other hand, performs optimization in two steps, but it allows per-campaign regularization which is beneficial.

While in this work we focused on advertising, user profiling and targeting are required in a wide range of web applications beyond advertising, such as content recommendation and search personalization. The principles of user profile generation and two-pronged targeting described in this paper are not specific to advertising, and are therefore applicable to these other applications.

7. REFERENCES

[1] D. Agarwal and B. chung Chen. Regression-based latent factor models. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009.

[2] N. Archak, V. S. Mirrokni, and S. Muthukrishnan. Mining advertiser-specific user behavior using adfactors. In *Proceedings of the Nineteenth International World Wide Web Conference*, 2010.

[3] A. Bagherjeiran, A. O. Hatch, and A. Ratnaparkhi. Ranking for the conversion funnel. In *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 146–153, 2010.

[4] A. Bagherjeiran, A. O. Hatch, A. Ratnaparkhi, and R. Parekh. Large-scale customized models for advertisers. In *Proceedings of the IEEE International Conference on Data Mining Workshops*, 2010.

[5] R. Bhatt, V. Chaoji, and R. Parekh. Predicting product adoption in large-scale social networks. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, 2010.

[6] Y. Chen, D. Pavlov, and J. Canny. Large-scale behavioral targeting. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009.

[7] I. Click Forensics. Click fraud index. `http://www.clickforensics.com/resources/click-fraud-index.html`, 2010.

[8] M. Gonen. Receiver operating characteristic (ROC) curves. *SAS Users Group International (SUGI)*, 31:210–231, 2006.

[9] N. Good, J. B. Schafer, J. A. Konstan, A. Borchers, B. Sarwar, J. Herlocker, and J. Riedl. Combining collaborative filtering with personal agents for better recommendations. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, 1999.

[10] M. Greiner, D. Pfeiffer, and R. D. Smith. Receiver operating characteristic (roc) curves. *Preventive Veterinary Medicine*, 45:23–41, 2000.

[11] J. Liu, P. Dolan, and E. R. Pedersen. Personalized news recommendation based on click behavior. In *Proceedings of the 15th International Conference on Intelligent User Interfaces*, 2010.

[12] Y. Peng, L. Zhang, M. Chang, and Y. Guan. An effective method for combating malicious scripts clickbots. In *Proceedings of the 14th European Symposium on Research in Computer Security (ESORICS)*, 2009.

[13] F. Provost, B. Dalessandro, R. Hook, X. Zhang, and A. Murray. Audience selection for on-line brand advertising: privacy-friendly social network targeting. In *Proceedings KDD*, 2009.

[14] B. Rey and A. Kannan. Conversion rate based bid adjustment for sponsored search auctions. In *Proceedings of the Nineteenth International World Wide Web Conference*, 2010.

[15] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, 2008.

[16] A. I. Schein, A. Popescul, L. H., R. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2002.

[17] M. Shmueli-Scheuer, H. Roitman, D. Carmel, Y. Mass, and D. Konopnicki. Extracting user profiles from large scale data. In *Proceedings MDAC*, 2010.

[18] K. Sugiyama, K. Hatano, and M. Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *Proceedings WWW*, 2004.

[19] X.-R. Wang, K.-W. Chang, C.-J. Hsieh, R.-E. Fan, G.-X. Yuan, H.-F. Yu, F.-L. Huang, and C.-J. Lin. Liblinear – a library for large linear classification. `http://www.csie.ntu.edu.tw/~cjlin/liblinear/`.

[20] S. Wedig and O. Madani. A large-scale analysis of query logs for assessing personalization opportunities. In *Proceedings KDD*, 2006.

[21] K. Weide. Worldwide and U.S. internet ad spend report: Growth accelerates, but dark clouds gather. http://www.idc.com/getdoc.jsp?containerId=224593 (visited on 10/19/2010), 2010.

[22] J. Yan, N. Liu, G. Wang, W. Zhang, Y. Jiang, and Z. Chen. How much can behavioral targeting help online advertising? In *Proceedings of the 18th International Conference on World Wide Web*, 2009.

[23] L. Zhang and Y. Guan. Detecting click fraud in pay-per-click streams of online advertising networks. In *Proceedings of the 28th IEEE International Conference on Distributed Computing Systems*, 2008.