Mach Learn (2014) 95:103–127
DOI 10.1007/s10994-013-5375-2

Machine learning for targeted display advertising: transfer learning in action

C. Perlich · B. Dalessandro · T. Raeder · O. Stitelman · F. Provost

Received: 19 November 2012 / Accepted: 29 April 2013 / Published online: 30 May 2013

Abstract This paper presents the design of a fully deployed multistage transfer learning system for targeted display advertising, highlighting the important role of problem formulation and the sampling of data from distributions different from that of the target environment. Notably, the machine learning system itself is deployed and has been in continual use for years for thousands of advertising campaigns—in contrast to the more common case where predictive models are built outside the system, curated, and then deployed. In this domain, acquiring sufficient data for training from the ideal sampling distribution is prohibitively expensive. Instead, data are drawn from surrogate distributions and learning tasks, and then transferred to the target task. We present the design of the transfer learning system We then present a detailed experimental evaluation, showing that the different transfer stages indeed each add value. We also present production results across a variety of advertising clients from a variety of industries, illustrating the performance of the system in use. We close the paper with a collection of lessons learned from over half a decade of research and development on this complex, deployed, and intensely used machine learning system.

Keywords Transfer learning · Display advertising · Predictive modeling

Editors: Kiri Wagstaff and Cynthia Rudin.

C. Perlich (✉) · B. Dalessandro · T. Raeder · O. Stitelman · F. Provost
M6D Research, 37 E. 18th St., New York, NY, USA
e-mail: claudia@m6d.com

B. Dalessandro
e-mail: briand@m6d.com

T. Raeder
e-mail: troy@m6d.com

O. Stitelman
e-mail: ori@m6d.com

F. Provost
Leonard N. Stern School of Business, New York University, 44 W. 4th St., New York, NY, USA
e-mail: fprovost@stern.nyu.edu

1 Introduction

Advertising is a huge industry (around 2 % of U.S. GDP), and advertisers are keenly interested in well-targeted ads. Online display advertising is a large subfield of the industry where ad targeting holds both promise and challenges. It is promising because of the wealth of data that can be brought to bear to target ads. It is challenging because the display advertising ecosystem is an extremely complicated system where accessing the data and delivering the ads can involve dozens of different corporate players. This is in contrast to search advertising for example. This paper deals with a particularly challenging segment of the online display advertising market: customer prospecting. Customer prospecting involves delivering advertisements to consumers who have no previously observed interactions with the brand, but are good prospects—i.e., are likely to become customers after having been shown an advertisement.

Display advertising has matured rapidly over the past several years, with the proliferation of real-time bidding exchanges (RTBs) that auction off website real estate for placing online display ads. This has created an efficient platform for advertisers to target advertisements to particular consumers (see e.g. Singer 2012). As is standard in the industry, we call the presentation of a display ad to a particular consumer an "impression." In each RTB the good being auctioned is an impression opportunity—a particular space or "slot" on a particular webpage at a particular instant with a particular consumer viewing it. The auctions are run in real-time, being triggered the instant a consumer navigates to the page and taking place during the time the page is being rendered in the consumer's browser. At the time of auction, information about the location of the potential advertisement and an identifying random number of the particular internet user are passed to all potential bidders in the form of a bid request. Advertisers often supplement these data with information previously collected or purchased about the specific consumer and website. When an auction is initiated, a potential advertiser must determine if it wants to bid on this impression, how much it would like to bid, and what advertisement it would like to display if it wins the auction. There are billions of such real-time auctions daily and advertisers require large-scale and efficient systems to make these decisions in milliseconds.

This complicated ecosystem invites machine learning to play a key role in the ad optimization process, particularly because of the simultaneous availability of (i) massive, very fine-grained data on consumer behavior, (ii) data on the brand-oriented actions of consumers, via instrumentation of purchase systems, and (iii) the ability to make advertising decisions and deliver advertisements in real time. The work we describe in this paper is one such massive-scale machine learning system that is deployed and in regular use by M6D, a company that finds prospective customers for targeted display advertising campaigns and executes those campaigns on the many advertising exchanges. Notably, the *learning system itself* is deployed, in contrast to the much more common case of deploying the models resulting from machine learning plus human model curation. Each week, this learning system builds thousands of models totally automatically, driving the advertising campaigns for major marketers across many industries.

This paper's main contribution to the machine learning literature is to use this application domain to demonstrate how data characteristics and availability constraints are translated and integrated into a complex problem formulation and finally implemented successfully as a robust learning system. We cover some seldomly discussed aspects of problem formulation for machine learning applications, focusing much of the discussion on the fact that for pragmatic reasons, the system draws data from multiple different sampling distributions to compose the machine learning solution.

As mentioned at the outset, our task is to identify prospective customers—online consumers who are most likely to purchase a specific product for the first time in the near future after seeing an advertisement. The ultimate goal of the system is to build predictive models automatically for hundreds of different and concurrent display ad targeting campaigns. A challenge for the system is that each campaign may have a different performance criterion. Each of these criteria may be approximated with a good ranking of potential purchasers in terms of their likelihood of purchasing. These problems have been described in detail previously (Provost et al. 2009; Perlich et al. 2012; Raeder et al. 2012). A primary source of feature data is a consumer's browsing history, captured as a collection of anonymized URLs that the consumer has visited in the past. The class label for each campaign is based on the observation of actual purchases of the campaign product. At first blush, this looks like an instance of a fairly straightforward predictive modeling problem. However, from a practical standpoint, it is impossible (both in terms of time and money) to obtain adequate training data directly for this problem. The dimensionality of the problem is already far beyond one million features in the simplest case of considering the browsing history of hashed URLs as an unordered set. The typical probability of purchasing the product within the next 7 days after seeing the ad varies between 0.001 and 0.0000001, depending on the product and the targeting. Collecting an ideal training set is often (if not always) prohibitively expensive. It also is time consuming, which has "cold start" implications for new campaigns.

Thus, at a high level the problem faced in building models automatically at scale for display advertising is twofold:

- The ideal training data are very costly to obtain, which is driven by many factors. Before a campaign is started there is no data at all on whether or not consumers purchase the product **after** having been shown the advertisement (no one has yet been shown the advertisement!). Once a campaign starts, there are severe selection bias problems with almost any available data—unless very careful and costly randomized experimentation is conducted. The base purchase rate is generally very low, so a tremendous number of randomly targeted ads must be shown to collect a data set with a significant number of positive examples. The result is that data from the ideal distribution for learning are scarce. However, related data from other distributions can be acquired at substantially lower cost.
- The system needs to learn models automatically for each new campaign. However, each campaign can vary widely in terms of the data available to it for learning, including the number of positive labels collected and the optimal feature engineering that can be applied. To operate as a robust and scalable learning system, the system needs to be flexible enough so that it can learn and exploit the idiosyncrasies of a specific task (i.e., campaign), but also run with minimal human intervention.

The system presented here solves these two problems with a two-level modeling approach. The first-level modeling deals with the sparseness, high-dimensionality, and model variety by drawing from sources of abundant, cheap (but notably biased) data. This step aggregates data from various data providers, engineers features and learns relationships from a variety of related processes that can be transferred to the main task we are trying to learn. The second-level modeling uses a stacked ensemble to combine, weight and recalibrate the outputs of the first-level process, and this learning uses data drawn from the actual "target" distribution.

We are aware of only few existing papers in the machine learning literature that look critically under the hood at the anatomy and design choices of real, deployed, massive-scale learning systems, some of them for targeting advertisements (cf., Perlich et al. 2012;

Raeder et al. 2012). Examining real, deployed learning systems can help to keep machine learning researchers aware of issues that are critical to the actual use of machine learning, and thereby can help to keep the science of machine learning vital (Provost and Kohavi 1998). Based on our experience in many real applications of machine learning, the issues that we present in this paper are more common in practice than the research literature would suggest. Specifically, this paper focuses on crucial issues of dealing with constraints of data availability including having data drawn from less-than-ideal distributions, and extremely rare outcomes. To address these issues, M6D's system incorporates and composes techniques from transfer learning and stacked ensemble classification. But more generally, we assert that most deployed applications of machine learning are actually instances of transfer learning—with at least some aspects of the learning data different from the true target task. If this indeed is true, then as a community we should examine applications more broadly as applications of transfer learning.

Other aspects of the system have been presented previously in conference publications (Perlich et al. 2012; Provost et al. 2009; Raeder et al. 2012), and with few exceptions these will be treated quickly in this paper.

2 Background on M6D display advertising and related work

M6D (Media6Degrees) delivers targeted display advertisements to online consumers, primarily focusing on prospecting, as described above. M6D finds and targets consumers online for over 100 marketers/brands at any given time, delivering many millions of ad impressions daily. The targeting system uses cookies to maintain a unique identifier for an internet user (until the cookie is deleted or the consumer chooses 'do-not-track') and this allows the system to associate different events with the same consumer as it interacts with her (technically, with her cookie). M6D works with a number of data partners that allow it to observe consumers' (partial) browsing history. In addition, at the beginning of a campaign each marketer places a number of "tracking pixels" on its brand website that allow the system to record visits to the site, purchases, and other actions that the marketer has deemed interesting. This also enables marketers and advertisers to measure meaningful metrics such as *post-view* conversions (important actions that occur subsequent to the showing of an ad) rather than just (mostly meaningless (Dalessandro et al. 2012)) clicks. Specifically, after a consumer is shown an impression, there is a time period established by the marketer within which some relevant action (e.g., visiting the marketer's website, downloading information, subscribing to some service, or purchasing a product) is considered a **conversion**. These conversions are the basic unit on which M6D's customers evaluate the success of the campaigns. Throughout the paper, we use the term **brand action** to refer to any relevant interaction between a consumer and a brand's web site. Most notably, we define **site visits** as visits to the brand home page or other selected pages. These are used to train many of our models where purchase information is too sparse or not available at all.

M6D delivers the majority of its ad impressions through ad exchanges (mostly in form of banner ads, but also video and mobile). After receiving a bid request from the exchange, M6D evaluates whether the consumer associated with the impression opportunity is a good prospect for a particular campaign, and if so, M6D will submit a bid. The bid price is determined by a separate machine learning process, which is described elsewhere (Perlich et al. 2012).[1] If the M6D bid was the highest across all bids for this auction, M6D delivers an ad for the campaign.

[1] In this paper, we will ignore issues of contention between multiple campaigns.

The system described here is not the only machine learning system to evaluate and optimize online display ad opportunities, and some prior efforts have been written about. Prior papers on the use of machine learning have focused on challenges associated with the applying machine learning in online advertising, such as the modeling of very rare outcomes using high-dimensional feature vectors and the "cold start" problem of having no training data from the target modeling task at the outset of the campaign. The use of high-dimensional, raw user log data as features in classification models was introduced in earlier work (Provost et al. 2009; Chen et al. 2009). Further work describes in finer detail how features can be constructed from raw user event data (e.g., Pandey et al. 2011; Liu et al. 2012).

To address the rare event/high dimensionality problem, various solutions have been proposed. Agarwal et al. (2010) use hierarchical relationships within the feature set to smooth probability estimates across levels in the hierarchy. Chen et al. (2009) incorporate Laplace smoothing into Poisson regression estimates, and Pandey et al. (2011) and Dalessandro et al. (2012) augment the rare outcome with a correlated outcome that has higher rates of occurrence. Though neither Pandey et al. (2011) nor Dalessandro et al. (2012) mention it, the use of alternative outcomes in a classification model is an instance of transfer learning. Liu et al. (2012) directly approach the challenges of modeling in the online display advertising setting with transfer learning in mind. Specifically, they propose a method, often referred to as "multi-task learning," where data from multiple tasks (campaigns) are pooled, a joint feature space is defined, and parameters are estimated across the joint feature space. The joint feature space is constructed such that some features can be thought of as global across all campaigns and others are specific to individual campaigns. This method is similar to the multi-task learning approach to spam filtering presented by Weinberger et al. (2009). Transfer learning across campaigns, however, is not the focus of this paper. M6D generally does not apply cross-campaign transfer because that may involve using one brand's data to optimize a competing brand's campaign, which is undesireable to M6D's clients.

The transfer learning conducted by the system described in this paper involves the use of data directly relevant to the current campaign, but drawn from a variety of different sources. Importantly, these sources represent distributions and tasks that are different from the distribution and task to which the learned models will be applied. We describe that more precisely next. To our knowledge, this is the first paper: to describe this sort of transfer learning in advertising, to describe in detail an actual production learning system that scalably conducts transfer learning across a multitude of source tasks, and to describe and carefully evaluate the inner workings of a fully deployed display advertising system, in this case a system that combines multiple models via (stacked) ensemble learning.

3 Transfer learning for display advertising

The focus of this paper is on transfer learning across different tasks, so let us next introduce formal definitions that will allow us to discuss the transfer precisely. The definitions here are somewhat more elaborate than in many machine learning papers because we need to decouple the various elements of the data that come from different sources. The interested reader is directed to Pan and Yang (2010) for a comprehensive survey of transfer learning, or to Weinberger et al. (2009), Xue et al. (2007), Heskes (1998), Evgeniou and Pontil (2004) for work most closely related to the sort of transfer learning described here. At a high level, the idea of transfer learning is that (i) the learning (in part) is conducted for a task that differs from the real target task in either the sampling distribution of the examples, the features

describing the examples, the exact quantity being modeled (the "label"), or the functional dependence between the features and the label, and (ii) that the knowledge obtained from this alternative learning task is then transferred to the real task—i.e., it is somehow used to improve the learning for the target task.

Specifically, let a **task** consist of a domain and a mapping. The **domain**, D, consists of an **example/instance space**, E, a **sampling distribution**, $P(E)$, on E, and a feature representation $X(e)$ for any $e \in E$, which provides a feature set for the example. The separation of the example space and the feature representation is important—examples (online consumers, in our case) are sampled from different distributions and the features can be engineered in different ways. Crucial to understanding the transfer is to understand that users may be sampled from distributions other than the target distribution, in order to augment the training data. Given the domain, a **mapping**, M, consists of a **label/outcome** space, Y, and a **function** $f(\cdot)$ that maps from a feature space X to a label space Y.

Any labeled dataset, whether used for training or testing, represents such a learning task with an implicit example space, some implicit sampling distribution, an explicit feature representation, and some implicit mapping. A **target task** includes a target domain $D_T = \{E_T, P_T(E_T), X_T(E_T)\}$ and a target mapping $M_T = \{Y_T, f_T(\cdot)\}$. The ultimate goal is to build models that predict well for the target task, i.e., have as good as possible an estimate of $f_T(\cdot)$. What it means for an estimate to be good is specific to the target task. Each potential **source task** includes a source domain $D_S = \{E_S, P_S(E_S), X_S(E_S)\}$ and source mapping $M_S = \{Y_S, f_S(\cdot)\}$, where $D_S \neq D_T$ and/or $M_S \neq M_T$. Transfer learning aims to improve the learning of $f_T(\cdot)$ (in D_T) by transferring knowledge of D_S and M_S into the estimation of $f_T(\cdot)$. Note that, as with machine learning problems generally, each function $f(\cdot)$ is not observed but can be learned approximately from the data. We can use this characterization to define precisely the different datasets used by the M6D system.

Recall the ultimate goal of our targeting system: identify internet users who are likely to *purchase* a particular product for the first time shortly after *seeing an advertisement*. This ultimate goal is the target task. Under this premise, the 'correct' target sampling distribution $P_T(E)$ for our models is the distribution of those users for whom we can win an ad impression given our pricing in a real-time auction and for whom there are no prior brand actions (purchases or site visits) related to the campaign in question. The target feature representation $X_T(E)$ is chosen by M6D to be most broadly a consumer's associated browsing history and characteristics inferrable from the consumer's browser. The ultimate target outcome Y_T is binary: did/will the user purchase after seeing an ad?

As pointed out previously, actually drawing sufficient data from the target task is prohibitively expensive, and we can now state precisely why. There are two reasons: drawing from P_T involves actually winning—purchasing—*randomly selected* impressions in the bidding systems (Perlich et al. 2012). Moreover, these impressions must be purchased in a quantity that provides sufficient positive instances to estimate a reliable, high-quality model given the feature space X_T. Three factors make this impracticable:

1. Most of the relevant information about purchase propensity is hidden in the complex browsing history. As a result, the potentially informative feature space X is extremely large (in our case tens of millions of URLs even after massive reduction).
2. Y_T has very, very few positive examples. The basic purchase rate for most of our advertisers' products, in particular for prospects (consumers who have not bought or interacted with the brand before) is typically below 0.001 % (sometimes well below) even when well targeted.
3. Showing ads *randomly* to (non-targeted) browsers and then waiting for them to purchase leads to many times fewer individuals that purchase, as compared to targeting intelli-

gently, and thus it is difficult to convince advertisers to spend money on random ad targeting.

Ultimately, the performance expectations of the advertisers do not allow for an expensive and lengthy "data gathering" period. Campaigns need to meet client goals in a relatively short period of time, often within the first week of launch. M6D's system solves this problem by using data it has gathered already. This data often goes back months to years and involves different $P(E)$, as well as actions similar to and relevant to Y_T but in much greater supply. Sampling from these alternative processes forms the basis for transfer learning within the system.

3.1 Possible alternative mappings/labels for targeted advertising

There are a number of more liberal definitions of labels Y that can be utilized to increase the number of positives for model learning. As mentioned, the primary target label of "purchase after being exposed to an ad" is a very rare event that requires delivering costly impressions. Alternative labels, candidates for Y_S, include: (1) clicking on an ad (still requires showing impressions), (2) any purchase (not just first time) after an ad, (3) any purchase with or without an ad, and (4) any other brand action with or without an ad. The number of positively labeled internet users is larger for the alternative actions. In fact, 4 is a superset of 3 is a superset of 2 is a superset of our target label. In addition to having a larger supply of Y_S than Y_T, for the transfer of knowledge to be effective, the estimated function $f_S(\cdot)$ should be (closely) related to the function of interest $f_T(\cdot)$. Consequently, the outcomes Y_S and Y_T should be (strongly) related. Intuitively, one interpretation is that whatever are the fundamental behavioral drivers for Y_T, they also should drive Y_S to some reasonable extent.

3.2 Domains and features of a users' online activity

As defined above a domain D has three main parts: the example space E, the sampling distribution $P(E)$, and the feature representation $X(E)$. For the M6D system, the example space most generally is the space of internet users/online consumers. This is globally true throughout the system. However, users are sampled in several different ways, specifically based on a set of events where M6D can interact with them. Individual internet users may be more or less likely to engage in specific sampling events (affecting $P(E)$), and this induces substantial heterogeneity across potential source and target tasks.

Specifically, the events during which M6D encounters users are based on several different interaction scenarios:

1. general internet activity—a user visiting a site/URL with which M6D has a data partnership,
2. bid requests from the exchanges/bidding systems,
3. showing an ad impression, whether targeted or untargeted,
4. clicking on an ad,
5. making a purchase at a campaign's brand's site,
6. taking some other related online brand action that can be tracked online, including visiting the brand's homepage, store locator page, etc.

For this paper, the main differences between populations collected through the different sampling events are differences in the sampling distributions $P(E)$. Thus, the data for a specific domain D comprise the set of instances collected via one of the above interaction scenarios, based on its corresponding distribution $P(E)$. In our stage-1 experiments below,

we use the union of all these events as the basis for the source domain. In practice, M6D builds different source-domain models based on the different events.

Making things more complicated, sampling events can be used to label the examples—some of these events reflect brand interaction, others do not. Thus the system can build modeling data sets by sampling a population from one event and assigning labels from a different event. For instance, those consumers who were shown an ad might be the population and the subset of those consumers who subsequently purchase from the brand's website might be the positively labeled consumers. Alternatively, a set of consumers active within the system who may or may not have seen an ad could be the population, and the subset of them who have also visited the brand's website could be the positively labeled set.

We would ideally like to use all available information to make the best decision. So as described before, the target feature representation $X_T(E)$ includes a consumer's associated browsing history as well as other user information. In any of the above domains and event samples, we can characterize a user i by a large set of K features $\{x_{1i}, x_{2i}, \ldots, x_{Ki}\}$. These features capture various aspects of the event, the user, and the user's internet browsing history. Also, this representation is general to both target and source domains. We generally assume that the feature representation $\{x_{1i}, x_{2i}, \ldots, x_{Ki}\}$ is universal in the system, and that only $P(\{x_{1i}, x_{2i}, \ldots, x_{Ki}\})$ is different between source and target domains. An example feature representation might be based on the collection of URLs that the browser has visited in some pre-specified history. For example, some x_{ki} may be a binary indicator of the user having visited a specific URL in the past. This particular feature representation will be denoted as X_{binary}. Alternatively, each x_{ki} may be some real numbered value that reflects some weighting of recency and frequency of the browser visiting a particular URL. In the M6D production system when URLs are employed as features, the system hashes the URL string into a string token that is devoid of any semantic context or meaning. The actual URL is not saved. This is done in an effort to maintain the privacy and anonymity of the user. Additionally, features describing various characteristics of the user can be used, such as for how long the user has been observed, some measure of the amount of internet activity, coarse geographic location, and descriptive elements of the browser application the user is using (X_{info}). Appendix B uses this formal characterization to present the specific definitions of the target and source tasks we use in the experiments in Sect. 4. Figure 1 highlights the main relationships between the user events (different colors identify different events and correspond in color to the sampling distribution of the tasks) on the left, the target task drawn from randomly targeted impressions and the two-staged transfer learning tasks.

3.3 Two-stage transfer learning

So far we have identified many possible source learning tasks (domains and mappings) that can be utilized for the ultimate goal of predicting the users who will most likely purchase a product after being exposed to an ad. Rather than selecting one, we use a two-stage transfer learning approach to learn multiple candidate mappings and then weight and combine them. Intuitively, the goal of the first stage is to dramatically reduce the massive target feature set X_T so that in the second step, we can actually learn based on the target sampling distribution P_T. The first stage considers multiple, parallel source learning tasks. Each task has its own learning problem that estimates a function $f_s(X)$ to approximate the label Y_S. In the second step, we learn how to transfer the set of predictions from the first stage by weighting the individual inputs via a learned linear classifier. The details of this are presented below. The key to understanding the different source and target tasks is to understand the different events; these lead to different sampling distributions and labels, as depicted in Fig. 1.

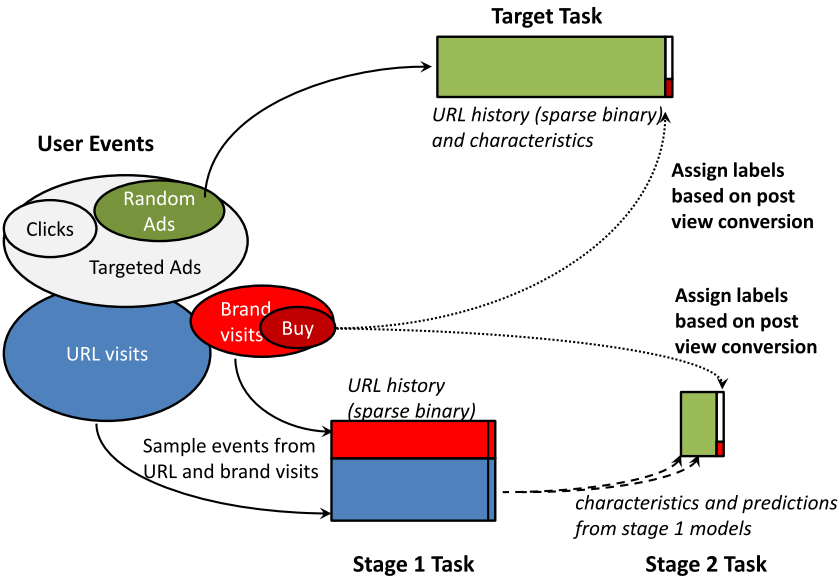

Fig. 1 Conceptual overview of the different events and tasks, their feature sets and how they are sampled from browser events. The *colors* identify different event types; the italic text describes the feature representation for the different tasks; *dotted arrows* and bold text specifies the process of assigning labels; *regular arrows* indicate the event sampling; the *dashed arrows* show the stage 1 to stage 2 transition where the predictions of models from stage one form the new feature set for stage 2

An interesting caveat about the system is that the 'correct' target learning task, which is whether or not a consumer purchases following an ad impression, is sometimes not used at all in our production system to build models for a particular campaign. The reason is that for some campaigns, budgets make it unrealistic to serve enough impressions to observe a sufficient number of purchases. In extreme cases, the purchase event of interest isn't observed at all due to issues with instrumenting the tracking pixels on the brand's post-conversion webpage. In such cases, the closest outcome observed is used as the target learning task. In practice, the next best outcome is usually a visit to the brand's website following an ad impression. A detailed analysis of such proxy outcomes (Dalessandro et al. 2012) shows that when attempting to predict purchases, using the site visit as the training outcome can significantly outperform using a purchase as the training outcome. In light of these findings we will move on, no longer making a formal distinction between a purchase and a site visit; the union of the two will be considered as the target label. This distinction is secondary for the transfer learning application in this paper, as we focus primarily on the sampling distributions $P(E)$ and how site visits/purchases are incorporated as labels.

3.3.1 Step 1—reducing the high-dimensional feature space

The prime motivation of the first stage in the transfer process is that the number of features in X_T is very large while the target label Y_T is very rare. The cold-start problem can be thought of as an extreme case of not having sufficient labeled data to learn a model. Given millions of ad impressions served, such a low base rate may or may not be a problem if X were of low dimensionality. Thus, the goal of the first step is to reduce the feature space in order to enable (subsequent) learning from a sample drawn from the actual target sampling distribution P_T.

The first stage involves learning multiple functional mappings in parallel, from different source tasks. The system creates multiple modeling datasets by sampling users from events that occur often, are relatively inexpensive, and are available prior to the start of the campaign. One of the most effective approaches is simply to assign every user who visits an advertiser's website a positive label, and then add a random sample of all other users that were active during the same time period as negative instances (as shown in Fig. 1). Call this alternate outcome Y_S. Note that although superficially similar, this setup is in fact quite different from sampling a cohort of consumers that could be "won" in the auctions and waiting for them subsequently to purchase. We can learn a function $f_S(X)$ that estimates/predicts Y_S. Any learning algorithm that is appropriate for the given feature space X_{binary} and is also scalable to tens of millions of features and tens of millions of examples may be used to learn $f_S(X_{binary})$. The production system at M6D currently uses both modified naive Bayes and logistic regression trained with stochastic gradient descent to estimate the multiple $f_S(X_{binary})$, due to these methods' scalability and robustness in production.

3.3.2 Stage 2—adjust to the target domain

As discussed above, in the first stage we estimate a number of models for each campaign, one for each suitable source task. More formally, let $\mathcal{S}$ be the set of source tasks that are suitable to be used in transfer learning. These source tasks are different combinations of domains and mappings as described above (for example, different sampling distributions and different label definitions). Given a user sampled in the target domain, we create a new feature representation $X_{\mathcal{S}} = [f_1(X_{binary}), f_2(X_{binary}), \ldots, f_n(X_{binary})]$ where each $f_s(X_{binary})$ is the result of applying the learned function $f_s(\cdot)$ for one of the source tasks. This feature representation can be appended to the mostly numeric feature set X_{info}, capturing other user characteristics.

In essence, we replace the original high-dimensional binary feature space with a much-condensed set of predictions (each of which again is a mapping from the binary feature space X_{binary} to a single numeric predictor) from the first stage models that have captured discriminative information. However, these predictions potentially have incurred bias due to domain differences. The second stage uses a stacked ensemble to estimate a low-dimensional function $Y_T = f(X_{\mathcal{S}}, X_{info})$ in this compressed feature space. Having a much smaller (but still informative) feature space allows us to learn this second-stage model (i) using the target sampling distribution $P_T(E)$ (a "control" set of users without prior brand interactions who had been selected randomly and targeted with advertisements in keeping with M6D's standard operating procedure) and (ii) using a target that actually tracks outcomes forward in time and after the ad event. As discussed above, in certain cases even the true campaign target label of purchase isn't available in sufficient supply for this step. So ultimately, even the second stage can yet again be an instance of transfer learning (albeit one that we have studied in depth to convince ourselves that it indeed is appropriate (Dalessandro et al. 2012)). Now we can move on to assess this transfer learning empirically, addressing two main questions:

1. Does the increase in positive labels in stage 1 indeed improve performance on the target task, considering the large bias resulting from the use of different sampling distributions and different definitions of labels? And as a side question: can we measure how different the tasks actually are?
2. Does the adjustment of the domain and the labels to the target task in stage 2 improve over the performance of the stage-1 models? Note that the models from the first stage can be used directly in production to target browsers with high purchase propensity.

In production, this two-step process produces two sets of models that are applied in sequence to identify the best browsers for targeting. Section 5 will speak to the production performance of the entire transfer system.

4 Transfer learning results

We now present results examining the different stages of the system's transfer learning and provide empirical answers the two questions presented at the end of the last section. The two experiments use the production system to create training sets and we provide two evaluation/test sets to isolate the impact of training on different source tasks in stage 1 and of combining and weighting the resultant models in stage 2.

The tasks on which we will evaluate the models in both stages have the correct sampling distribution $P_T(E_T)$ of the target task (the above-mentioned population of a randomly selected set of users to whom we can show an ad and who have not taken a brand action previously) and use the feature set that was used for training. Note that in sequence, the models of stage 1 and stage 2 provide a mapping of the complete feature set of the target feature representation $X_T = (X_{binary}, X_{info})$ from both browsing history and user characteristics. Positive browsers (in Y_T) are those who take a brand action within seven days of seeing the ad.

4.1 The benefits of stage-1 transfer

This section shows that using a convenient sampling distribution $P_S(E)$ and labeling scheme that maximizes positives but does not reflect the actual target task often provides better results than if we were to just always use the sample $P_T(E)$ of the target task. From the transfer learning perspective, we will show that the estimation of function $f_S(\cdot)$ is often a better predictor of Y_T than the estimation of $f_T(\cdot)$.

To demonstrate that the source and target tasks are indeed significantly different, we empirically test the difference in $P_T(E)$ between our target domain and the source domain $P_S(E)$ comprising all active internet consumers observed in our system. Specifically, we build a classifier to distinguish users sampled from the two distributions using binary URL indicators as the feature set and training a linear logistic regression model with the class variable representing the domain from which the user was drawn. If this model has predictive power, then the domains indeed are measurably different with respect to the feature set. Indeed, this model achieves an out-of-sample AUC of 0.8 at predicting the domain, which means that we can separate reasonably well the consumers sampled from the target domain and those sampled from the source domains. This implies specifically that *there are considerable differences between the population of all active internet users and the population of those for which it is possible to bid and to win impression opportunities within the bidding systems*.

4.1.1 Data details

Proceeding to our main analysis, for the experiments that follow, we define the source population E_S as all active internet users who are observable within the system, and the sampling distribution $P_S(E_S)$ as the empirical convenience distribution represented by taking the union of all the sampling events discussed in detail in the previous section. The source label Y_S indicates whether the user has visited the marketer's website at any point in the past. We

compare models trained on this source task against models trained directly using the target task. Specifically, the target population E_T is all users who appear in the ad exchanges and for whom we could win the auction at our desired bid price, along with the corresponding $P_T(E_T)$. The target label Y_T is whether the browser has taken a brand action within seven days following the ad impression. We estimated the functions $f_T(X_{binary})$ and $f_S(X_{binary})$ across 28 active campaigns. The evaluations are made on the distribution $P_T(E_T)$ of the target task using an out-of-sample and out-of-time holdout sample. For the experiments of this paper, each model is trained on data from the same 14-day time period, with identical feature sets. The only difference between the source and target tasks is the sampling distribution P that determines *which* users make it into the training sample. The (out-of-sample/out-of-time) evaluation holdout set comprises a random sample of target-task users from the following seven days. The results, presented below, show that this biased initial sampling scheme is better for learning from the massive, fine-grained URL features.

4.1.2 Modeling details

The stage-1 models presented here are trained with logistic regression based on the URL feature representation (binary features). The logistic regression is trained using stochastic gradient descent (SGD) (Bottou 2010), because it scales easily to millions of features and millions of examples. To specify the exact parameter settings, recall that SGD trains $F(\mathbf{x}) = \beta^T \mathbf{x}$ by processing each instance in the training set individually and making small incremental updates to each coefficient along the gradient of the loss function. The loss function is regularized maximum likelihood, using either L1- or L2-regularized loss (using the standard formulations). The SGD training process takes two parameters, hereafter called *learning parameters*. The first, the *learning rate*, controls how aggressively the model coefficients are updated with each instance and the second, the *regularization parameter* penalizes large coefficients to reduce overfitting. Through careful experimentation (done previously over several years, on completely separate prior data), we have found a set of default learning parameters that works well across a wide variety of campaigns. Although we examine parameter tuning below for the purposes of our experiments, M6D typically does not conduct expensive optimization of the learning parameters on-the-fly for each production model, as doing so increases training time dramatically and generally reduces the robustness of the production system (Raeder et al. 2012). Production parameter changes are done offline.

4.1.3 Experimental results

The performance metric for these experiments is *lift* within the top 2 % of the population: the number of positive examples in the set of examples with the top 2 % of model scores divided by the number that would be expected from a random classifier (i.e., 2 % of all positives). This choice of metric reflects a typical campaign setup, where given a fixed budget that allows targeting only a small proportion of browsers, M6D is expected to return as many conversions as possible. Each of the reported lift numbers is the average of 100 bootstrap estimates, in order to produce a low-variance estimate.

Figure 2 plots the performance of the model trained on the stage-1 source task against the performance of the model trained on the target task, as evaluated on the same target-task holdout data. Each point is one campaign; the x-axis plots the target-trained lift; the y-axis plots the source-trained lift. Models for which source training is better are above the diagonal line; models for which target training is better are below. Clearly, the stage-1 source-trained models usually outperform the target-trained models (20-3-5 win-tie-loss

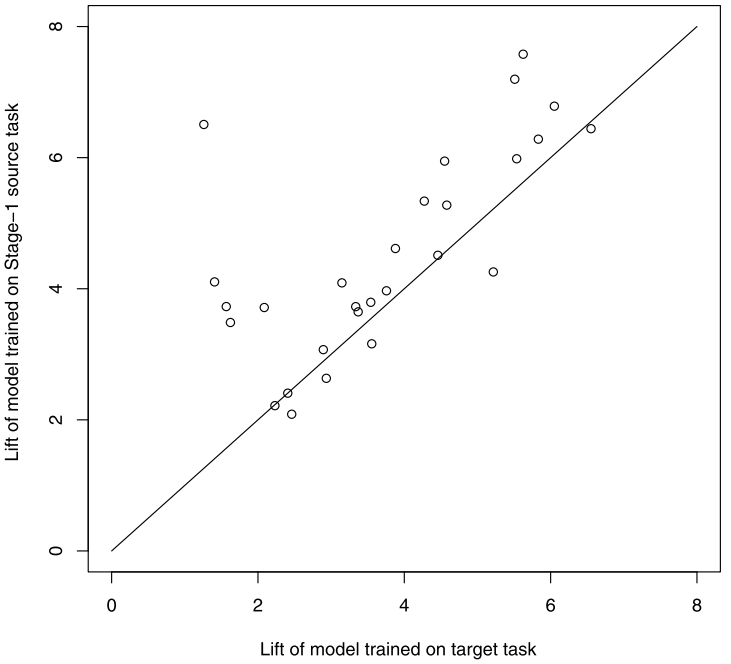

Fig. 2 Comparison of model performance between learning on the stage-1 source task and learning on the target task (default learning parameters). Both are evaluated on the target task. Every *point* is a campaign. *Points* above the identity *line* indicate that the models trained on the source perform better on the target task than the models trained on the target task

record, highly significant by a sign test). Moreover, the source-trained models sometimes are much better, whereas in the few cases where the target-trained models are better, the lift difference is small.

In order to address the possibility that the default learning parameters are especially well-tuned to the stage-1 population, we conducted an extensive, expensive parameter search on a smaller set of seven campaigns. Specifically, we built models on both the stage-1 task and the target task using 25 different combinations of learning rate and regularization parameter (including instances of both L1 and L2 regularization). For each campaign, we choose the set of learning parameters with the best two-fold cross-validation (cv) performance on the training set, and we report the performance of the model learned with these chosen parameters, as well as the default-parameter model, on the same out-of-time test sets that we used for the prior experiments.

Detailed results for these seven campaigns appear in Fig. 3. Each cluster of bars represents the prediction performance of four different models at target-task prediction on the test set. From left to right, the bars represent the performance of: (1) the chosen-parameter model trained directly on the target task, (2) the default-parameter model trained directly on the target task, (3) the chosen-parameter model trained on the stage-1 source task and (4) the default-parameter model trained on the stage-1 source task. In six of the seven cases, the model chosen by cross-validation on the stage-1 source task outperforms the cv-chosen "best" model built on the target task directly. In one case, the model trained with default learning parameters on the target task outperforms everything, but in general we see that

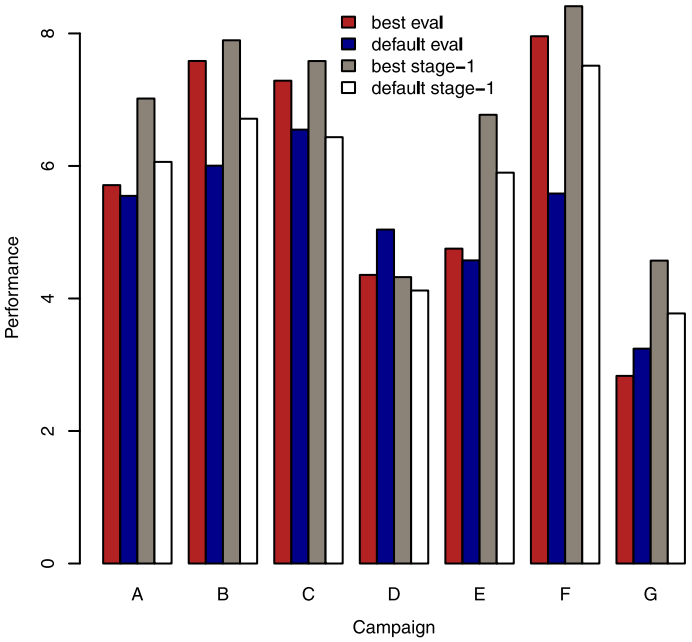

Fig. 3 Comparison of the performance (lift at 5 %) on the target task of stage-1 and target-task (eval) models for seven campaigns (A-G) after tuning the learning parameters. The "best" model is the one that performed best in cross-validation on the training set

even when tuning the learning parameters, training on the stage-1 source task remains the most effective strategy.

These results are not necessarily intuitive—i.e., that learning from a completely different (source) training distribution and with a different definition of positive vs. negative label would yield consistently better results in the target domain. However but it is worth noting that the training population consistently provides more positive examples by at least an order of magnitude. For the campaigns in Fig. 2, our stage-1 source population contained anywhere from 15 to 98 times as many positives as the target population. This suggests that the reduction in variance afforded by the increased positive-class signal is more than sufficient to compensate for the bias introduced by training on a different sampling distribution and different labels. It seems likely that this result would hold in many real-word applications where positive-class data are scarce or expensive.

4.2 Stage-2 ensemble model

As mentioned earlier, our production system builds several stage-1 source models for each campaign. The second stage of our transfer process uses the output scores produced by these models as components of a low-dimensional feature representation of the target-task sample of users (those who have seen a randomly targeted impression). In this section, we compare the performance of each "stacked" (Breiman 1996) stage-2 model to the performance of its constituent stage-1 models. This will provide evidence that the explicit adjustment to the target task, via the retrained ensemble, improves over simply using one of the source models without target-task adjustment. (Recall that the high-dimensional stage-1 source-trained

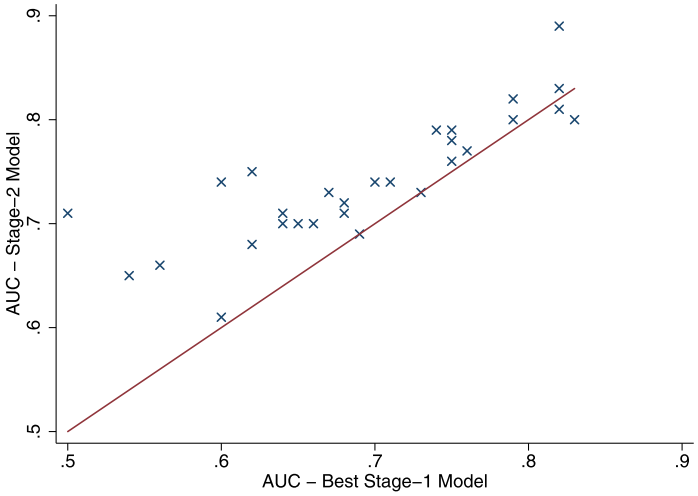

Fig. 4 Performance of the stage-2 stacked ensemble trained on the target task data compared to the best stage-1 model for each campaign. Performance is reported as the areas under the ROC curve (AUC) for each campaign and model. The stage-2 models consistently outperform even the best stage-1 models. (Here, to be conservative, the "best" stage-1 models are chosen based on their performance on the test data)

models outperform the high-dimensional target-trained models, as shown in the previous section.)

4.2.1 Data details

For these experiments, as the basis for the target distribution we collected 30 days of randomly targeted users from $P_T(E_T)$. The base rates of the campaigns vary dramatically; these data sets had anywhere from 50 to 10,000 positive examples. In contrast, each campaign has a huge number of negative examples. For these experiments, we selected a random sample of 50,000 negatives.[2] The stage-2 feature-generating process produces approximately 50 features, including all of the stage-1 model scores for the particular campaign and user, and a number of user characteristic features (X_{info}) such as browser type, age of the cookie, and geo-location information.

4.2.2 Modeling details

The stage-2 model is a logistic regression classifier trained using elastic net (Zou and Hastie 2005) regularization. Elastic net regularization combines L_1 and L_2 regularization and will usually send some model coefficients to zero, and thus also serves as feature selection (as with straight L_1 regularization).

4.2.3 Experimental results

Figure 4 shows a performance comparison across 29 different campaigns, a representative sample of target tasks that are part of recurring advertising campaigns. We plot the area

[2]The AUC results below are not affected by the arbitrary change to the base rate. The training may be affected, but generally if a fixed number of examples is to be used for training, a more-balanced sample generally leads to better predictive performance, as measured by AUC (Weiss and Provost 2003).

under the ROC curve (AUC) of the stage-2 model on the y-axis against the AUC of the best-performing stage-1 model on the x-axis. All performance estimates were calculated on an out-of-time hold-out set similar to stage 1 methodology. An important concern in stacking is ensuring out-of-time evaluation across both stages. The evaluation period was not part of the stage-1 training period.

The performance differences between the stage-2 learner and the best stage-1 learner show that combining source models and incorporating information about the target task significantly improves the effectiveness of the models. In our experience, the majority of this improvement is due to combining the various stage-1 models. We attempted to build assembles of stage 1 models an user characteristic features on stage 1 domain and observed significant deterioration of performance compared to the best stage 1 model. This implies that the performance improvements in stage 2 are indeed primarily due to the adjustment to the target domain and not due to the ensemble effect or the additional features introduced in stage 2.

The median and average improvements in AUC across the different campaigns were 0.038 and 0.041, indicating substantial improvements in predictive performance. The improvements are even more dramatic for cases where the best stage-1 model has relatively poor performance. For cases where the best stage-1 model is in the bottom 50 % of campaigns, the median and average improvements due to the stage-2 ensemble are 0.056 and 0.061 respectively. Any "negative transfer" is implicitly handled by the learning procedure, meaning that the logistic model will assign a very low or negative parameter if one of the stage-1 models is not predictive of the target task or would make the performance worse. Any stage-1 model with a meaningful relationship to the label, positive or negative, should receive an appropriate weight in the ensemble.

There are two cases where the ensemble learner underperforms the best stage-1 learner. These are cases where the best stage-1 model already has very high performance. Note that for illustration (and to be conservative) we compare the stacked-ensemble performance to the *best* stage-1 learner as selected on the test (holdout) data, which involves a multiple comparisons problem (Jensen and Cohen 2000). Choosing the best stage-1 learner via cross-validation instead should only increase the perceived advantage for the stacked ensemble, but does not illustrate fully the advantage of the stage-2 learning.

The reader might notice the large variance in AUC across campaigns in both stages of the transfer process. Such variance is common in this application due to the diversity of brands. Some are mass market brands, such as well known banks, telephone service providers and soft drink makers; others are niche brands, such as high-end fashion retailers, luxury travel destinations and specialized hobbies. Some of the latter yield much greater discrimination ability. As a brand gets more popular, its resulting customer base becomes more heterogenous and this makes building discriminative models more difficult. Therefore, the absolute values are less important than the comparisons across methods.

5 Production results

The systematic experiments in the prior section showed the contributions of the two transfer stages in lab experiments (on real data). We now present two sets of results demonstrating the system in action. It is worthwhile to consider who are the "customers" of the deployed machine learning system. The immediate customers of the system are the company officers (e.g., the CEO) and the Board of Directors. The machine learning system is integral to the performance of the company, for which they are responsible. Therefore, we first will show

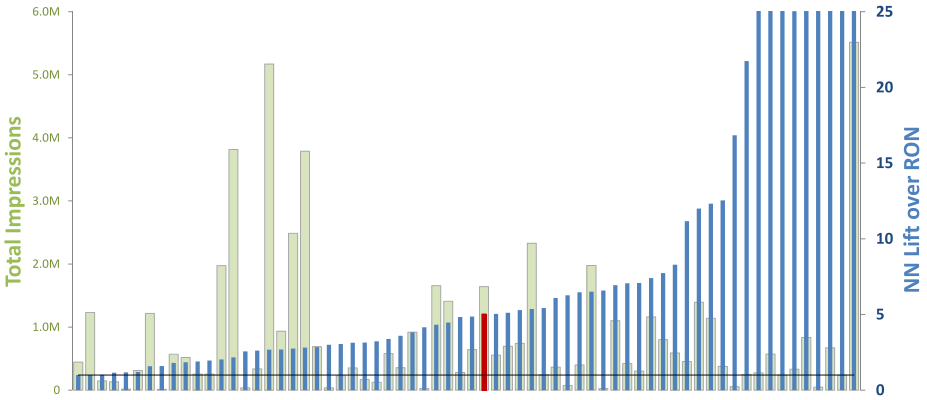

Fig. 5 Actual performance of the production system across 66 representative campaigns for a representative time period. The *left axis* (*with wider light colored bars*) shows the impression volume in millions, the *right axis* the predictive performance in terms of lift on post-view conversions of the targeted campaign over the (random) control group. The campaigns in the graph are sorted by performance (better to the *right*). Every campaign has two *bars*, a wider but lighter one for volume and a narrower darker one for predictive performance. The median performance of about 5x is highlighted in the *middle*. To keep the left side legible, the performance bars for the right most 9 campaigns are truncated at a lift of 25x

the overall performance results as viewed by these stakeholders. The ultimate customers of the decisions made by the system are the company's advertising clients. Therefore, we also provide testimonials from these clients describing their individual experiences with the system.

5.1 Results for the CEO and board

M6D tracks a number of key performance indicators (KPI). Typically there is a notable variation in goals between different campaigns (see cases in Appendix A for examples). To allow for consistent performance tracking and statistical reliability, M6D computes a lift measure: the ratio of the number of site-visit conversions on all targeted prospects to the number of site-visit conversions on the randomly targeted control group that also saw ads for the campaign. Note that we again use site-visit instead of purchase for evaluation (Dalessandro et al. 2012). This is necessary for two reasons: (1) some campaigns have no purchase events for various reasons, and (2) even if they do, we often have too few post-impression conversions in the randomly targeted control group to estimate the baseline (random) conversion rate reliably. In the lab results presented above, we used a targeting threshold of 2 % to compute the lift. Here we use whatever the targeting threshold actually was for that campaign, which is determined by a complex constellation of business factors including the targeting budget. The results in Fig. 5 show that for all the campaigns, model-based targeting outperforms random targeting; the median lift is close to 5 and for some 15 % of campaigns the lift easily exceeds a factor of 25.

5.2 Individual client cases

We have included in Appendix A a number of detailed cases reporting on actual reactions from customers in a variety of industries including retail, travel, automotive, consumer packaged goods, and education. Generally large advertisers such as these engage multiple targeting firms, and compare the targeters' performances using their own internal evaluations.

Note that we cannot reject the concern that there is a positive bias in the reported results, as they are not subject to the scientific rigor we might desire and are more likely to be published if the customer was satisfied. This being said, the fact that the presented approach *can* achieve as impressive results as those reported in the cases is nevertheless testimony to the value of the machine learning system.

Brands use a number of techniques to assess and compare performance: (1) use multiple targeting firms ("on the plan") at the same time and compare for instance the conversion results (subject to attribution) or the ("effective") Cost Per Acquisition (CPA or eCPA), (2) involve a firm like Nielsen to assess the performance of the targeting as well as the causal impact of the advertisement, and (3) have some internal process to measure ROI based on the estimated campaign-induced increase in revenue.

These case study results show that on a number of campaigns the machine learning system's targeting performs extremely well and, in particular, notably better than the competitors. Ultimately M6D has able to achieve very high customer satisfaction in terms of ad impact, ROI, cost per customer acquisition (CPA), and ultimately consistent campaign renewals for M6D.

6 Conclusion

This paper presents the detailed problem formulation incorporated by a massive-scale, real-world, production machine learning system for targeted display advertising. The problem formulation involves the use of labels and sampling distributions different from the target task, in order to deal with the practical inability to acquire sufficient data from the target environment. The system learns models with different "source" sampling distributions and training labels, and then transfers that knowledge to the target task. Experimental results show that the *conscious* use of biased proxy populations for training can improve model performance in situations where data are scarce.

More specifically, sampling data from the target task is expensive and positive outcomes are quite scarce, especially considering the very high dimensionality of the feature representation (specifically, comprising data on visitation to a massive set of anonymized URLs). The system trains the full-fledged high-dimensional models on proxy populations where examples are cheaper and labels are more abundant. Then, it trains a much lower-dimensional, stacked ensemble model, with the previous model scores as features, on a smaller sample from the actual target sampling distribution. The experimental results show that each component of the multi-stage transfer learning system improves over not using that component.

The results and approaches presented in this paper represent the result of years of development of the massive-scale machine learning system that is deployed and in regular use by M6D. Some of the most important lessons learned from this experience are presented above and can be summarized and synthesized as follows:

1. In building machine learning applications, thinking explicitly about the subtleties in the definitions of E, $P(E)$, and Y can allow significant improvements in the result of machine learning. For example, in this application we see that drawing data from distributions other than the target distribution as well as using labels different from the target label both can improve performance. Combining source models and adjusting to the target distribution can improve performance further.
2. This transfer learning approach has additional practical benefits that we have not emphasized. It addresses directly the "cold-start" problem: in cases where no advertisements have been shown, one cannot learn models directly for the target task. However, the

source models do not depend on advertisements being shown, and they can be used directly when there are too few training data from the target domain.[3] In addition, a multi-stage approach like that applied here—learning multiple, biased, source models and then combining and correcting them—is especially attractive in production settings where new modeling methods can be added more easily than existing production procedures can be changed. New machine learning methods operating on the super-high-dimensional data can be added in easily, and they immediately become additional features in the stage-2 stacking procedure. The system seamlessly evaluates them, via the stage-2 learning, and if they add value (as determined empirically) then they get significant weights in the resultant stage-2 model.

3. Acquiring a large amount of data that is not from the optimal data generating distribution can be better than acquiring only a small amount of data from the optimal data generating distribution. It is always important to weigh the cost of getting data from other source tasks versus the benefit of having more of that data. M6D explicitly conducts evaluations of the cost/benefit tradeoff of acquiring data from each data partner, based on the improvement (or lack thereof) of adding it in as a new source domain.
4. When building an automated system that (i) learns many models simultaneously and automatically, (ii) updates those models on an ongoing basis, and (iii) needs to be scalable, it is important to make design decisions that benefit the majority of the models without severely damaging any of them.
5. The idea of progressive dimensionality reduction, building successively lower-dimensional models, is useful in its own right (separate from these production constraints) whenever data are available at multiple resolutions. It is ineffective to mix features such as browser characteristics (or demographics, if a targeter wished to acquire such data) directly with millions of binary URL indicators. On the other hand, such a low-dimensional feature set integrates smoothly with the low-dimensional stage-2 feature set of stage-1 model outputs.

We hope that these lessons will be useful to other researchers in their efforts to study important problems in Machine Learning, and to other practitioners in their efforts to apply machine learning, especially in situations where they seek to build an automated machine system that regulates itself with minimal human intervention. We believe that in real applications, learning from distributions and/or with labels that do not match the target task exactly is much more common than is apparent in the machine learning research literature. This has been noted explicitly for sales targeting (Rosset and Lawrence 2006). For credit scoring, models often are built at least in part from data on a credit issuer's customers, which is a significantly different distribution from that of the target task (credit applicants); similarly, in targeted marketing, predicting response *level* generally is done based on those customers who have responded, even when the target task is to estimate the response level across the prospect population. Both of these are instances of learning under selection bias, which has been studied extensively (Zadrozny 2004). In fraud detection data often are drawn from different sources and composed into training data, without (necessarily) careful consideration of the target sampling distribution (Fawcett and Provost 1997). For webpage classification for safe advertising, data on the very rare classes of interest are drawn from all manner of sources different from the actual target task (Attenberg and Provost 2010;

[3] In cases where source task predictions are not combined via learning with an appropriate target task, M6D will run multiple prediction algorithms simultaneously in a champion/challenger framework. Budget allocations will be reset according to which algorithms show better in campaign performance.

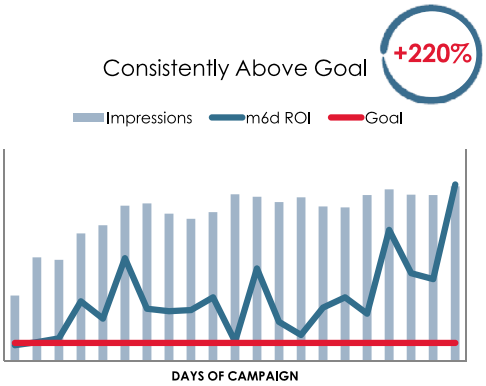

Fig. 6 Sneaker campaign performance

Attenberg et al. 2011; Ipeirotis et al. 2010). In some of these applications, such as the aforementioned credit scoring and traditional targeted marketing, long experience has led some practitioners to think about certain issues of transfer learning explicitly (even if they don't call it that). In other domains practitioners proceed without carefully thinking about issues of transfer. Further, as we have illustrated, often the transfer is much more nuanced than is accounted for by the common corrections for selection bias.

This paper illustrates how transfer learning can lead to measurable improvements, and we assert that, at least in our case, thinking explicitly about the transfer aspects of such applications has led to more improvement than if such transfer were simply done based on modeling intuition.

Acknowledgement Foster Provost thanks NEC for a Faculty Fellowship.

Appendix A: Performance case studies

In this appendix we present case studies for particular campaigns that we ran for individual customers using the system presented in the body of this article. We have run thousands of these campaigns for major marketers and these case studies present a few campaigns where marketers were willing to go on the record and present their results along with some of the metrics that they used to evaluate campaign success. Additional cases are available at the M6D website. The data and analyses presented in this appendix were made by M6D clients without the consultation of the authors or any employee of M6D. As a result, specific details on methodology are unavailable.

A.1 Online retail: Sneakers

Over the decades, this casual footwear brand has become the true American staple for canvas sneakers. The brand's goal was to drive a large volume of revenue while delivering an efficient ROI. As a result, M6D was a top performer on the plan, exceeding the client's ROI goal by 220 % for the entirety of the campaign. M6D was given a 50 % increase in spend per month for the remainder of 2012, and at its peak was exceeding the ROI goal by 10x.

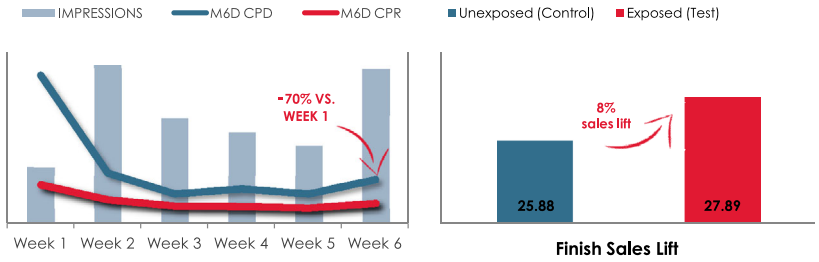

Fig. 7 Finish campaign performance

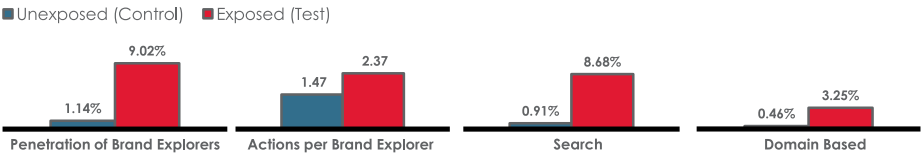

Fig. 8 Cosmopolitan campaign performance

A.2 Customer packaged goods: Finish

Finish, (Dishwashing Detergent) was looking to drive offline sales and online coupon downloads at scale without wasting impressions. M6D was evaluated on two performance criteria for this campaign: coupon downloads and in-store sales. To measure in-store sales, measurement leader Nielsen was commissioned to perform their Sales Effect study. By analyzing the implicit site visitation patterns that emerged for Finish customers, M6D was able to improve performance and find more new prospects as the campaign progressed. From week 1 to week 6, M6D was able to drive down weekly CPD (cost per download) by 70 %. In the same time period, CPR (cost per registration) decreased by 48 %. Households that were in the exposed group purchased more total Finish detergent than those that were not exposed, driving an 8 % sales lift and resulting in an estimated $1.7 Million in offline incremental sales. M6D drove the highest incremental revenue per thousand impressions that Nielsen has seen among its Sales Effect studies for Reckitt Benckiser, and drove the highest ROI among all vendors.

> **Quote:** *"With RB being a flagship branding partner with M6D, it was very refreshing to find such a new, innovative company truly engaged and eager to figure out, while working in lockstep with a client, how to crack the branding code by re-jiggering what was considered to be an industry leading DR-platform and technology. The willingness and drive of M6D to re-work their toolkit for custom needs of RB as a representative branding client was an exciting and very rewarding undertaking."*
> Media Director, North America, Reckitt Benckiser

A.3 Travel: Cosmopolitan of Las Vegas

The Cosmopolitan collaborated with Nielsen to measure the ad effectiveness of the M6D campaign along a number of dimensions. **Research actions** are user visit (page view) to cosmopolitanlasvegas.com or a brand-related search term. A **brand explorer** is a person who engages in at least one Research Action. As measured by a Nielsen Response Effect

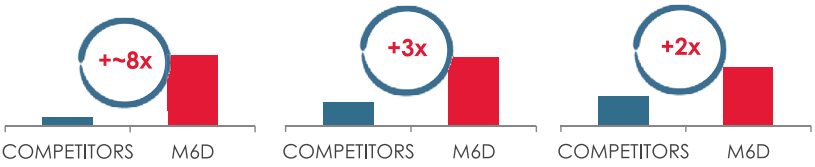

Fig. 9 Infiniti campaign performance. The three measures are, *left* to *right*, the numbers of lead requests, the click-through-rate, and a compilation of brand-specific performance indicators

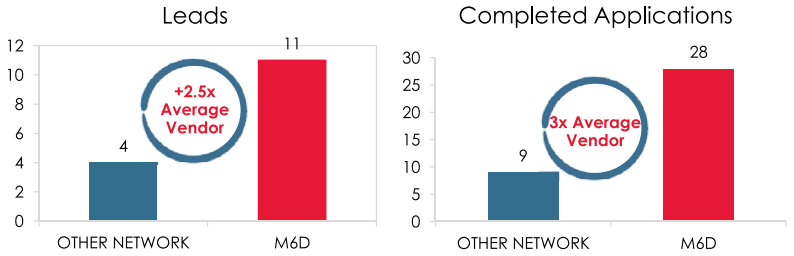

Fig. 10 New School campaign performance

study, M6D increased Cosmopolitan Hotel online brand explorers by 690 % and the number of research actions per explorer by 61 %.

A.4 Automotive: Infiniti

By using M6D's targeting rather than more traditional targeting methods, Infinity was able to reach audiences that are more receptive to their brand's message. The marketing network for Infiniti-branded vehicles now includes over 230 dealers in over 15 countries. Infiniti pitted M6D against two ad networks measuring three main performance criteria, presented in Fig. 9. Left to right the figure shows the comparison with the competitors in terms of: **Lead requests** include online events such as dealer locator, vehicle configurator, contact retailer, contact dealer pre-order, CPO contact dealer confirm, quote requests, and test drives. **Click-thru rate (CTR)** is the traditional percentage of clicks on the ads. **Other performance indicators (OPI)** are directed at in-markets users and includes certified pre-owned (CPO) search results, credit inquiries, CPO dealer locator and retail locator. In terms of OPI, M6D performed better than the other two video networks. OPI conversions were 2x better than the competition. Per DART reporting, M6D generated a CTR that was 3x better than the competition. In terms of Lead Requests, M6D was the top performer with almost 8x more conversions than the closest competitor.

> **Quote:** *"By using M6D's targeting rather than more traditional targeting methods, Infiniti was able to reach audiences that are more receptive to their brand's message."* Strategist, OMD Digital

A.5 Education: The New School

Parsons The New School for Design is known for cultivating outstanding artists, designers, scholars, business people, and community leaders for over a century. Looking to reach a large audience of qualified new prospects without sacrificing ROI, The New School was

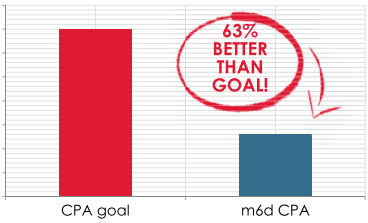

Fig. 11 Bonobos campaign performance

looking to M6D to drive efficient leads and applications. The results showed not only increased leads, but also an increase in delivering qualified applicants to The New School. M6D delivered 11 leads (information requests)—more than 2.5 times the number delivered by the competing network. M6D delivered 28 completed applications, more than three times the number delivered by the competing network.

> **Customer Quote:** *"M6D helped us reach beyond the audience that was already on our site and find new prospects who were interested in the university; the results speak for themselves."*
> Digital Marketing Manager, The New School

A.6 Online retail: Bonobos

Bonobos has built a reputation for service, quality, style, and most of all, fit. They were asking how can an innovative retailer with extensive knowledge of its customer life cycle use prospecting to grow its client base and drive sales? M6D drove a CPA 63 % better than the client's goal for finding new prospects. Due to M6D's strong performance, the client renewed for the following quarter, doubling its monthly spending. We also presented the client with our proprietary Non-Invasive Causal Estimation (NICE) analysis, demonstrating that M6D prospecting could drive a significant lift in conversions for the brand (cf., Stitelman et al. 2011). The team also used the NICE analysis to highlight the portion of conversions attributable to our targeting (vs. those that would have happened organically).

> **Quote:** *"M6D provided a unique combination of avant-garde technology and insight that helped us to effectively use their prospecting engine to expand our customer base without over-spending on wasted impressions. With the correct attribution strategy in place, they quickly became our top performer for our holiday campaign."*
> Head of Customer Acquisition, Bonobos

Appendix B: Definition of the tasks and datasets

Stage 1 (high-dimensional training) The objective of this first task is to provide as many positives as possible to allow estimation of complex and high-dimensional models on a representation of the user's browsing history, with each URL hashed into its own binary feature. The resulting feature space is very large (≈10 Million) and extremely sparse. For the experiments, we use all brand action events from a given time period as positive examples and include a random sample of other browsing events as negative.

- Sample: Union of all online activity including general browsing events and all brand action events. Users who are in both are assigned to action events.
- Features X_{binary}: binary URL indicators for approximately 10 million URLs.
- Label: $Y_{Stage1} = [0, 1]$; 0 for browsing events and 1 for action events.

Stage 2 (low-dimensional training) Stage-1 modeling learns an accurate high-dimensional model on a biased sample. The stage-2 task uses the output of our stage-1 models as part of a much-lower-dimensional model on the target sample $P_T(E_T)$. Like our final target task, it uses future brand actions as the label Y_T. The lower dimensionality allows us to learn effectively with the small number of positives in the correct population.

- Sample: A set of users who have seen a random (untargeted) ad. All users who have previously taken a brand action are removed. This sample $P_T(E_T)$ is consistent with the ultimate target task.
- Features $X = (X_{info}, X_{Stage1})$: Some basic cookie statistics X_{info} (cookie age, number of interactions, user agent) along with the predictions of the stage-1 models $X_{Stage1} = [f_1(X_{binary}), f_2(X_{binary}), \ldots]$ where each $f_s(X_*)$ is the application of learned function of the first stage.
- Label: $Y_T = [0, 1]$: Did the consumer visit the brand site/make a purchase within seven days of being exposed to an ad? This is consistent with the target label.

Target task The ultimate goal of our learning system is to predict, using ALL the information that we have on a user, whether or not that user will buy something within seven days. We obtain far too few positives in this space to model it directly, but it is useful for evaluation.

- Sample $P_T(E_T)$: A set of users who have seen a random (untargeted) ad. All users who have previously taken a brand action are removed.
- Features $X_T = (X_{info}, X_{binary})$: Union of all features including basic browser/cookie stats and binary URL indicators that are transformed into second-stage features using the models estimated in stage 1.
- Target: $Y_T = [0, 1]$ Did the user visit the brand site/make a purchase within seven days of the ad impression?

References

Agarwal, D., Agrawal, R., Khanna, R., & Kota, N. (2010). Estimating rates of rare events with multiple hierarchies through scalable log-linear models. In *Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 213–222).

Attenberg, J., & Provost, F. (2010). Why label when you can search? Strategies for applying human resources to build classification models under extreme class imbalance. In *Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining*.

Attenberg, J., Ipeirotis, P., & Provost, F. (2011). Beat the machine: challenging workers to find the unknown unknowns. In *Workshops at the 25th AAAI conference on artificial intelligence*.

Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *Proceedings of the 19th COMPSTAT international conference on computational statistics* (pp. 177–187). Berlin: Springer.

Breiman, L. (1996). Stacked regressions. *Machine Learning*, *24*(1), 49–64.

Chen, Y., Pavlov, D., & Canny, J. (2009). Large-scale behavioral targeting. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 209–218).

Dalessandro, B., Hook, R., Perlich, C., & Provost, F. (2012). Evaluating and optimizing online advertising: forget the click, but there are good proxies. NYU Working Paper CBA-12-02.

Evgeniou, T., & Pontil, M. (2004). Regularized multi–task learning. In *Proceedings of the 10th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 109–117). New York: ACM.

Fawcett, T., & Provost, F. (1997). Adaptive fraud detection. *Data Mining and Knowledge Discovery*, *1*(3), 291–316.

Heskes, T. (1998). Solving a huge number of similar tasks: a combination of multi-task learning and a hierarchical Bayesian approach. In *Proceedings of the 15th ICML international conference on machine learning* (pp. 233–241).

Ipeirotis, P., Provost, F., & Wang, J. (2010). Quality management on amazon mechanical turk. In *Proceedings of the ACM SIGKDD workshop on human computation* (pp. 64–67).

Jensen, D., & Cohen, P. (2000). Multiple comparisons in induction algorithms. *Machine Learning*, *38*(3), 309–338.

Liu, Y., Pandey, S., Agarwal, D., & Josifovski, V. (2012). Finding the right consumer: optimizing for conversion in display advertising campaigns. In *Proceedings of the 5th ACM international conference on web search and data mining* (pp. 473–482).

Pan, S., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345–1359.

Pandey, S., Aly, M., Bagherjeiran, A., Hatch, A., Ciccolo, P., Ratnaparkhi, A., & Zinkevich, M. (2011). Learning to target: what works for behavioral targeting. In *Proceedings of 20th ACM CIKM conference on information and knowledge management* (pp. 1805–1814).

Perlich, C., Dalessandro, B., Hook, R., Stitelman, O., Raeder, T., & Provost, F. (2012). Bid optimizing and inventory scoring in targeted online advertising. In *Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 804–812). New York: ACM.

Provost, F., & Kohavi, R. (1998). Guest editors' introduction: on applied research in machine learning. *Machine Learning*, *30*(2), 127–132.

Provost, F., Dalessandro, B., Hook, R., Zhang, X., & Murray, A. (2009). Audience selection for on-line brand advertising: privacy-friendly social network targeting. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 707–716).

Raeder, T., Dalessandro, B., Stitelman, O., Perlich, C., & Provost, F. (2012). Design principles of massive, robust prediction systems. In *Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining*.

Rosset, S., & Lawrence, R. (2006). Data enhanced predictive modeling for sales targeting. In *Proceedings of SIAM conference on data mining*.

Singer, N. (2012). Your online attention, bought in an instant. *The New York Times*. November 17, 2012

Stitelman, O., Dalessandro, B., Perlich, C., & Provost, F. (2011). Estimating the effect of online display advertising on browser conversion. In *Proceedings of the workshop on data mining and audience intelligence for online advertising at ACM SIGKDD*.

Weinberger, K., Dasgupta, A., Langford, J., Smola, A., & Attenberg, J. (2009). Feature hashing for large scale multitask learning. In *Proceedings of the 26th ICML international conference on machine learning* (pp. 1113–1120). New York: ACM.

Weiss, G., & Provost, F. (2003). Learning when training data are costly: the effect of class distribution on tree induction. *Journal of Artificial Intelligence Research*, *19*, 315–354.

Xue, Y., Liao, X., Carin, L., & Krishnapuram, B. (2007). Multi-task learning for classification with Dirichlet process priors. *Journal of Machine Learning Research*, *8*, 35–63.

Zadrozny, B. (2004). Learning and evaluating classifiers under sample selection bias. In *Proceedings of the 25th ICML international conference on machine learning* (p. 114). New York: ACM.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, *67*(2), 301–320.