Targeted Advertising With Total User Privacy

Morgan Saville

CAUSE

morgan@cause.cx

Abstract

It is common for free platforms on the internet to use targeted advertising as a revenue stream, but this usually comes at the cost of user privacy. In this paper we propose a method of targeting advertising to users of an online platform without any of their data being collected or accessible. Broadly, this technique combines deep reinforcement learning (to learn user preferences) with cryptography (to prevent others, including the platform administrators, from accessing any private user data). Furthermore, we propose an algorithm which allows multiple platforms to benefit from these learned preferences while still maintaining user privacy.

1 Introduction

This paper solves two main problems. The first is maximising the revenue of an individual platform without sacrificing user privacy. The second is generalising this technique such that it can be used across multiple platforms.

With respect to the first problem, we will define our criteria for success.

(1) No private user data should be accessible to other users of the platform, attackers, advertisers, or the platform itself.

(2) Subject to criterion (1), platform revenue must be maximised.

(3) Skeptical users must be able to verify criterion (1) if they so wish.

The success criteria for the second problem are as follows:

(4) Targeting technqiues acquired for a user on one participating platform must be usable for targeting on any other participating platform.

(5) Criterion (1) must hold across and within all participating platforms.

(6) Skeptical users must be able to verify criterion (5) if they so wish.

(7) No malicious agent can interfere with the targeting techniques for a general user (e.g., an advertiser making their ad show up more frequently than it should).

It is important to define what is meant by "private user data" in criterion (1). For the remainder of this paper, this term refers to any and all of the following:

- Any information which indicates which advertisements a given user has seen or interacted with.

- Any information about a given user's activities on a given platform.

In addition to these, we will use the term "private user data" to capture the intuitive notion of any data which a particularly privacy-concerned user might not want/expect anybody else to have access to.

Note that it is up to the individual platforms to decide how and why they store data when it comes to other parts of their platform. In this paper we are simply ensuring that no private user data is collected by the targeted advertising system itself. This is to say that just because a platform participates in the targeted advertising scheme described herein, it does not provide a guarantee that the platform maintains the same level of privacy across its other features.

An example of data which *must* be collected when advertising is when a user clicks on an ad which redirects them to the advertiser's website, the advertiser does need to know the platform from which the click originated (or something to this effect). This is necessary for the purpose of billing. However, we do not consider this a violation of privacy so long as the advertiser can not connect that interaction to a specific user account on the platform.

Furthermore, such an interaction may well result in the advertiser being able to associate the click with the IP address of the user. Particularly cautious users might consider using a VPN while browsing, or indeed the platform might want to provide

ads in forms other than hyperlinks. Suffice it to say that workarounds for this problem exist, but are beyond the scope of this paper. Instead we will focus only on ensuring privacy in the process of *deciding* which ads to show.

2 Simulating Users

We will need to be able to simulate users and their preferences in order to quantify how well this algorithm is likely to work in production. We will make use of the ADS-16 Computational Advertising Dataset[1] for this task. This dataset contains (amongst other things) a set of answers given by 120 users to 10 personality questions. The answers are given on a scale from -2 to 2 (inclusive of both ends). Each user's age, income bracket, and gender are also included. In addition, the dataset contains a set of 300 advertisements split evenly into 20 categories. Crucially, this dataset also provides a rating from 1 to 5 (inclusive of both ends) that each user gave to each ad.

We propose a method for modelling the relationship between a user's answers to the personality questions and that user's rating for a given ad. We will then use this to generate a set of simulated users, and we will use this model to determine their interaction with ads.

Note that it is not important for our purposes what the personality questions specifically are — only that their answers hold some predictive power with respect to the ad ratings.

The technique laid out in this section can be easily adjusted to work with a different (perhaps custom) dataset, as long as it contains data from which a simulated user can be generated, and their interaction with ads modelled.

For the sake of clarity, we will define the following terms for the data we have available to us:

$$
\begin{aligned}
b_{i,j} &= \text{user } i\text{'s answer to personality question } j \\
&\in [-2,2] \cap \mathbb{Z}
\end{aligned}
$$

$$
\begin{aligned}
g_i &= \text{user } i\text{'s gender} \\
&\in [0,1]
\end{aligned}
$$

$$
\begin{aligned}
a_i &= \text{user } i\text{'s age in years} \\
&\in \mathbb{N}
\end{aligned}
$$

$$
\begin{aligned}
\lambda_i &= \text{user } i\text{'s income bracket} \\
&\in [0,3] \cap \mathbb{Z}
\end{aligned}
$$

$$
\begin{aligned}
r_{i,k} &= \text{user } i\text{'s rating for ad } k \\
&\in [1,5] \cap \mathbb{Z}
\end{aligned}
$$

$$
\begin{aligned}
c_k &= \text{category of ad } k \\
&\in [0,20) \cap \mathbb{Z}
\end{aligned}
$$

$$
\begin{aligned}
\bar{r}_{i,c} &= \text{mean rating from user } i \text{ across all ads with category } c \\
&= \text{mean}\left(\{r_{i,k} | c_k = k\}\right)
\end{aligned}
$$

With respect to g_i, we use a value of 1 to correspond to female, and 0 to male, though this is arbitrary. Furthermore, the dataset we are using only contains males and females, but this technique generalises seamlessly to non-binary identities corresponding to values in the range $(0,1)$.

For each user we construct a "raw" user feature vector f_i' to encapsulate all of the user data we might use to predict $r_{i,k}$.

$$
f_i' = \begin{bmatrix} b_{i,0} & \dots & b_{i,9} & g_i & a_i & \lambda_i & \bar{r}_{i,0} & \dots & \bar{r}_{i,19} \end{bmatrix}
$$

We refer to this as a "raw" vector because it requires some processing to become useful. If our hope is to eventually simulate users, it will involve generating one of these vectors. However, although the empirical distribution of each individual element can be found directly in our dataset, these features are not independent, and so cannot be sampled independently. This makes intuitive sense, as we might expect, for example,

age to correlate with income bracket, but this intuition can be confirmed by inspecting the correlation matrix of these features across all users.

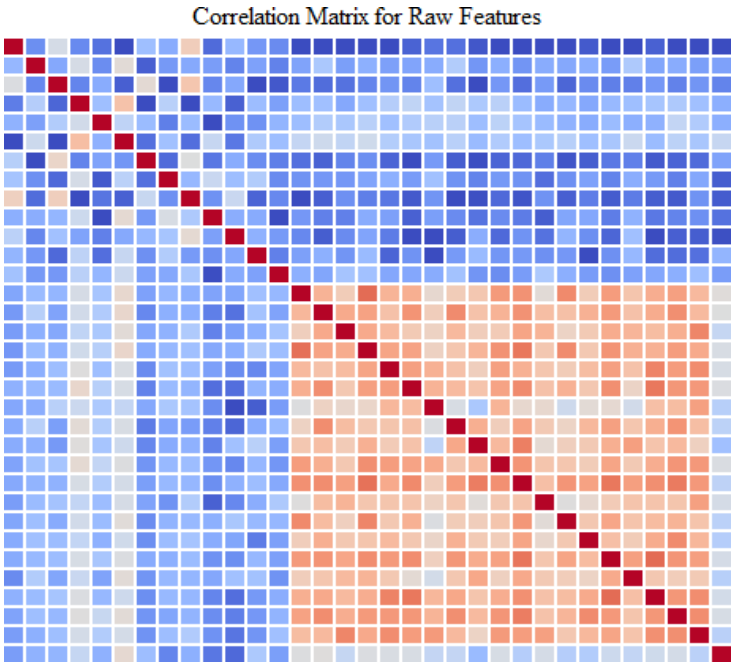

Figure 1: Correlation matrix for components of f_i'

We therefore perform principal component analysis on these features, yielding the user feature vector $f_i \in \mathbb{R}^{33}$, whose components are all independent.

We assume that each of the components are normally distributed, and we can estimate the parameters of the distributions across all of the users in the dataset.

In addition to the data above, we also require the text present in each ad. With our dataset, we must rely on optical character recognition to extract the text from the images of the ads we are given. With this, we use a pre-trained GloVe[2] embedding of dimension d_e.

This yields two tensors for the k^{th} ad, ${}_0v_k$ and ${}_1v_k$, which correspond to a one-hot vector encoding of c_k, and the ad's embedded word vectors respectively. The shape of ${}_0v_k$ is (20) and the shape of ${}_1v_k$ is (w_m, e_d) where w_m is the maximum number of words in an ad.

We use a deep neural network including a transformer block[3] generate an approximation $\hat{R}$ of the following function R:

$$R\left(f_i, {}_0v_k, {}_1v_k\right) = \text{onehot}\left(r_{i,k}\right)$$

such that

$$\text{argmax}\left(\hat{R}\left(f_i, {}_0v_k, {}_1v_k\right)\right) + 1 \approx r_{i,k}$$

The addition of 1 to the left hand side of the above is due to the fact that the indices of the one-hot vectors range from 0 to 4 (inclusive of both ends) whereas the values of $r_{i,k}$ range from 1 to 5 (inclusive of both ends).

The DNN has the following structure:

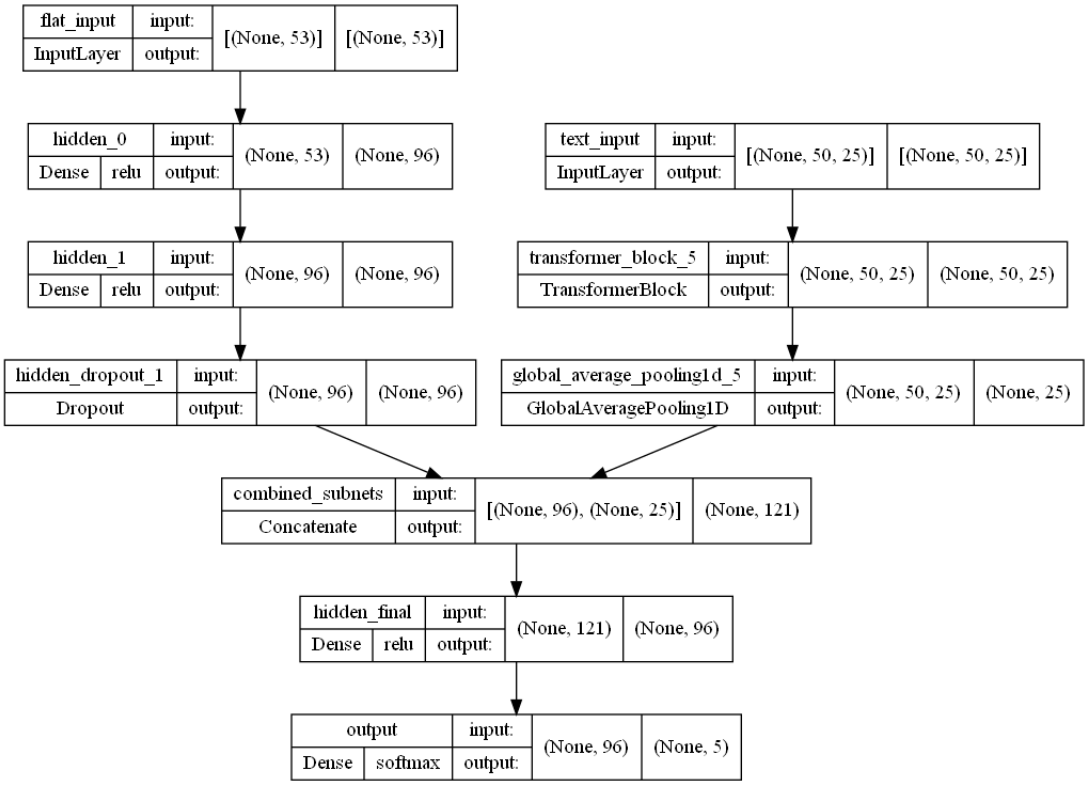

Figure 2: The structure of a DNN which approximates a given user's rating for a given ad, with $w_m = 50$, $e_d = 25$

The hyperparameters can be found in the appendix and were determined using the Keras[4] implementation of the Hyperband algorithm[5].

References

[1] G. Roffo and A. Vinciarelli, "Personality in computational advertising: A benchmark," in *4th Workshop on Emotions and Personality in Personalized Systems (EMPIRE)*, p. 18, 2016.

[2] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017.

[4] F. Chollet *et al.*, "Keras." `https://keras.io`, 2015.

[5] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6765–6816, 2017.