Probabilistic Deduplication of Anonymous Web Traffic

Rishiraj Saha Roy, Ritwik Sinha, Niyati Chhaya and Shiv Saini
Adobe Research Big Data Intelligence Lab, Bangalore, India - 560029.
{rroy, risinha, nchhaya, shsaini}@adobe.com

ABSTRACT

Cookies and log in-based authentication often provide incomplete data for stitching website visitors across multiple sources, necessitating probabilistic deduplication. We address this challenge by formulating the problem as a binary classification task for pairs of anonymous visitors. We compute visitor proximity vectors by converting categorical variables like IP addresses, product search keywords and URLs with very high cardinalities to continuous numeric variables using the Jaccard coefficient for each attribute. Our method achieves about 90% AUC and F-scores in identifying whether two cookies map to the same visitor, while providing insights on the relative importance of available features in Web analytics towards the deduplication process.

Categories and Subject Descriptors

[**H.3.5 Online Information Services**]: Web-based services

General Terms

Algorithms, Experimentation, Measurement

Keywords

Web analytics; Visitor deduplication; Binary classification

1. INTRODUCTION

Website cookies and log in authentication are fundamental in tracking user browsing behavior. However, users often switch between multiple browsers and devices, or do not log in to websites, making it difficult to bring together their browsing behavior. This makes content personalization difficult and often results in overestimation in several aggregate reporting use cases vital for Web analytics. In such cases, anonymous visitor identities cannot be mapped to existing customer records from cookies alone and there is a need to have smarter approaches for deduplicating website visitors.

Issues with past work. For a website, the cookie clustering approach in Dasgupta et al. [1] suffers from the difficulty of estimating the right granularity of the clustering process, such that each cluster would correspond to exactly one person. Large websites have millions of unique users and several of them might have very similar attributes, leading to possible ambiguities in deduplication with the fingerprinting approach as used in Panopticlick [2]. To gather ground truth data, organizations like *comSCORE* and *Nielsen* hire panelists to compute approximate ratios of cookies and known unique visitors to have an estimate of how many unique cookies belong to the same visitor. Google addresses this problem using the same single sign on credentials for Google Analytics and Google Chrome for every person and only provides authenticated access to its services. However, as far as we know, none of the other organizations providing similar services (like TAPAD, Janrain, and Drawbridge) and past research use our proposed pairwise visitor classification approach (Sec. 2). Our general framework will perform well for all cross-device and cross-channel situations where we can collect information about the attributes of the visit and the activities performed during the website visits.

WWW 2015 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2742750.

2. APPROACH

We formulate the visitor deduplication problem as a binary classification task. We wish to predict, with high confidence, whether a pair of visitors map to the same human user. For this purpose, we first aggregate information for each *visitor* (equivalently, *cookie*) (denoted by some visitor ID $v_i \in V$, the set of all visitor IDs) over the entire input log. This information consists of browsing attributes $a_k \in A$, the set of all *visit* (equivalently, a *click* or a *hit*) attributes, like IP address, URLs visited and browsing hours. Next, we build feature vectors for each pair of visitors (v_i, v_j) by computing overlaps between corresponding attributes. For example, let v_i and v_j have four and five distinct values each for attribute a_k = *product URLs visited*. Out of these nine a_k values, let two values be common for v_i and v_j. We plan to use a quantification of this overlap to build a feature vector for every pair of visitors. Since using raw counts may result in disproportionate importance for certain attributes, we use the Jaccard coefficient for normalizing the overlap values to the range $[0, 1]$. The Jaccard coefficient J between two sets a and b is computed as $J(a, b) = |a \cap b|/|a \cup b|$. So in our example, $J_{a_k}(v_i, v_j) = 2/7$. The vector of *Jaccard coefficients* for all the attributes A form the feature vector f for the pair (v_i, v_j). The feature vectors are computed for each pair of visitor IDs in the input log. A positive (visitors known to be the same person) or negative class label (visitors known to be different persons) can be associated with each visitor pair feature vector from the authentication information in the input log. Finally, the labeled data can be divided into training and test sections, for learning and evaluating our prediction model.

Classifier	Train	Test	Accuracy	AUC	F-Score
LR	$350k$	$150k$	0.862	0.902	0.875
SVM	$100k$	$30k$	0.845	**0.903**	**0.894**
RF	$350k$	$150k$	**0.877**	0.902	0.865

Table 1: Classification performance.

3. RESULTS AND INSIGHTS

Dataset. We used ten days of Web analytics data from one of Adobe Analytics' customer organizations, containing millions of clicks. We found about $250k$ visitor pairs with different visitor ID but referring to the same human user (known through authentication) – this was our positive class. We randomly sampled a similar number of visitor id pairs which are known to belong to different users, forming our negative examples. We use the following types of attributes, which we believed to be predictive of visitor uniqueness (short codes in parentheses): geo-location (IP address (*ip*) and city of origin *(city)*), product views (search terms (*search*), general site URLs visited (*url*) and product URLs visited (*produrl*)), browser properties (*useragent*), and time (browse hour of the day (*hour*)), thus using seven (out of an available 150) features in total. We built feature vectors for these $500k$ visitor pairs by computing attribute-wise Jaccard coefficients. We created a $70:30$ train-test split of the data into training and test sets with equal proportions of positive and negative classes in the training and test sets. We used three different classifiers for the binary classification - logistic regression (LR), support vector machine (SVM) and random forest (RF).

Experiments and results. We computed three popular metrics: AUC (area under ROC curve), F-Score (harmonic mean of precision and recall) and classification accuracy (fraction of correct predictions among all predictions) and report results in Table 1. The train and test columns represent the numbers of data points used, respectively. Performance was comparable across the three classifiers, and we achieved a very good AUC of about 90%, about $86-89\%$ F-Score, and about $84-87\%$ accuracy. Even though SVM produced the best results for two out of three metrics, the differences were not statistically significant. LR and RF scaled much better than SVM, which turned out to be computationally prohibitive on the entire dataset on a single node. LR required much less training time than RF on the entire data, and LR being a *parametric method* (unlike RF) will also be a better choice for storing prediction models for very large datasets.

We analyze the predictive power of the individual features for LR and RF and find that all features play *statistically significant* roles in the prediction process. The most predictive features towards visitor deduplication were found to be *IP address, search terms, useragent* and *browse hour*, determined using z-scores for LR and mean decrease in Gini index for RF. Making predictions using only these features in the classifier also show similar findings. We investigated the correlation between our features and plotted the correlogram shown in Fig. 1, where darker shades indicate higher correlations. We make interesting observations, like visitors coming from same IP addresses search for similar items, and cities show similar browsing hours.

Insights on reducing number of visitor pairs. We show Jaccard coefficient overlap distributions (red for negative class, blue for positive class; coefficients versus #pairs) for two of the features (IP address, city) in Fig. 2. The maximum y-value for both panels is 12000 pairs. IP address shows one of the highest contrasts between the two classes with overlap values only on extremes (0 and 1). This implies that visitors never sharing an IP are almost never the same person (negligible mass at 1.0 x-value on left panel for IP). However, significant probability mass at 0.0 Jaccard in the right panel for IP also shows that there are a number of visitor pairs who never share an IP address but are the same person, underlining the need for other features for prediction. Features like city show less drastic contrasts and have more probability mass on other Jaccard coefficient values. These plots can help us arrive at simple heuristics on reducing the total number of pairs to make predictions on, and how much error they are likely to induce.

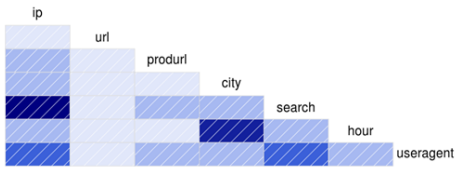
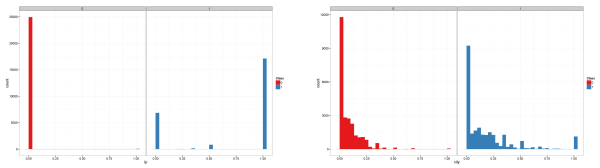


Figure 1: Correlogram of features.



(a) (b)

Figure 2: Jaccard distribution for positive (blue) and negative (red) classes: (a) IP (b) City.

4. CONCLUSIONS AND NEXT STEPS

We have presented a novel method for deduplicating website visitors using a binary classification approach. We have used location and browsing activity based features to build our predictive model, which are easily accessible to most Web analytics solutions. AUC values of 90% were obtained with multiple classifiers, highlighting the robustness and accuracy of our approach. Categorical variables with high cardinalities, like search terms or URLs, have been converted to numeric variables that can be used in the standard framework for learning algorithms. We find IP address, site search terms, browsing time, useragent and visit URLs to be highly predictive towards probabilistic deduplication. Our immediate next steps would be to use our model to reduce overestimation in aggregate reporting use cases like counting the number of unique visitors to a website.

5. REFERENCES

[1] A. Dasgupta, M. Gurevich, L. Zhang, B. Tseng, and A. O. Thomas. Overcoming browser cookie churn with clustering. In *WSDM '12*, pages 83–92, 2012.

[2] P. Eckersley. How unique is your web browser? In *Privacy Enhancing Technologies*, volume 6205 of *LNCS*, pages 1–18. Springer, 2010.