Cross-Device User Linking: URL, Session, Visiting Time, and Device-log Embedding

Minh C. Phan
School of Computer Science and Engineering, Nanyang Technological University, Singapore
phan0050@e.ntu.edu.sg

Aixin Sun
School of Computer Science and Engineering, Nanyang Technological University, Singapore
axsun@ntu.edu.sg

Yi Tay
School of Computer Science and Engineering, Nanyang Technological University, Singapore
ytay017@e.ntu.edu.sg

ABSTRACT

Cross-Device User Linking is the task of detecting same users given their browsing logs on different devices (*e.g.,* tablet, mobile phone, PC, etc.). The problem was introduced in CIKM Cup 2016 together with a new dataset provided by Data-Centric Alliance (DCA). In this paper, we present insightful analysis on the dataset and propose a solution to link users based on their visited URLs, visiting time, and profile embeddings. We cast the problem as pairwise classification and use gradient boosting as the leaning-to-rank model. Our model works on a set of features exacted from URLs, titles, time and session data derived from user device-logs. The model outperforms the best solution in the CIKM Cup by a large margin.

KEYWORDS

Cross-Device User Linking; Entity Resolution

1 INTRODUCTION

"Consumers are media multitaskers - and cookies are not",[1] the utility of verifying user's identity across multiple devices lives at the heart of practical applications such as online advertising and user profiling. Intuitively, the large diversity of personal mobile devices would inevitably result in fragmentation of a user's online activities, *e.g.,* we check our phones on the go, read documents on iPads/tablets on the train and perhaps use PCs at home or work. It is clear how online behaviour is fragmented across a multitude of devices. As such, many brands and businesses have only weak user identities to work with, *i.e.,* the same user across multiple devices are considered different people. Naturally, we see the merits of automatically detecting the same person across multiple devices. Amongst the many benefits are richer, more accurate user profiles which could potentially lead to better advertisement targeting.

Recently, the significance of this problem motivated a data mining competition organized by the Conference on Information and Knowledge Management as the CIKM Cup[2] in 2016 which included a publicly released dataset provided by Data-Centric-Alliance (DCA). In this competition, the task is to identify all pairs of device-logs that belong to same users given their browsing logs on different devices. Figure 1 illustrates the task at hand. Within the competition setting, we noticed that learning-to-rank and pairwise classification are common approaches used by winning teams [2, 3]. Most of the winning approaches focus on feature engineering. Features are carefully designed based on the visited URLs and visiting time correlation between device-logs. However, we also note that some important aspects and features are missing such as session-based features which are clearly important in web usage mining. In addition, the semantic meaning of web page is not considered, because its URL is treated independently from its title. The prime contributions of our work are as follows:

- For the first time, we present important observations pertaining to the DCA cross-device dataset. Our analysis is used to derive insightful features that reflect the special characteristics of the dataset and the problem.
- We introduce a TF-IDF like *session-based weighting scheme*, and a neural network model, to compute the representations of device-logs. We also present effective probabilistic features and usage pattern related features for detecting user's identity. Our proposed model outperforms the winner of the CIKM Cup by a noticeable margin.

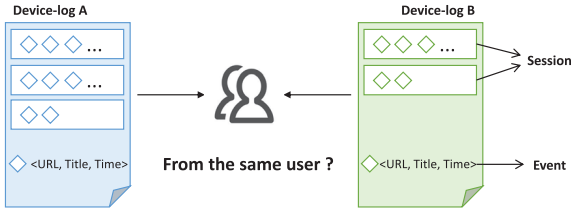


Figure 1: Overview of the Cross-Device User Linking problem. Events are grouped into sessions in preprocessing.

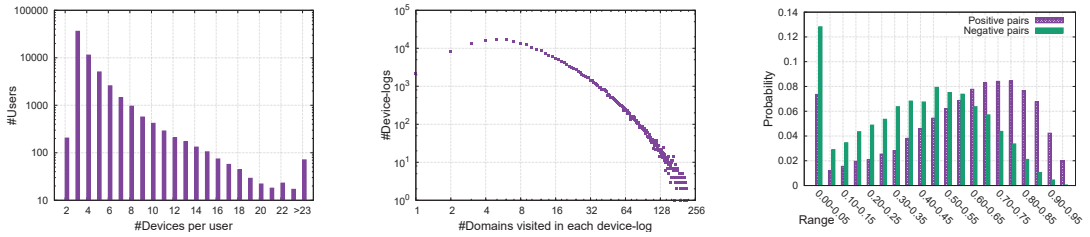
2 DATASET AND ANALYSIS

The cross-device user linking dataset contains anonymized browsing logs of 84,522 users on 338,990 different devices within 2 months (April 23rd to June 23rd, 2016). Each device-log is a list of events in the format of 3-tuple $\langle URL, title, time \rangle$: address, title of the website, and the visiting time. URL is in form of '$a/b/c/\ldots$' where a, b, and c are hash-codes of words in the address. Similarly, words in title are also obfuscated by a MD5-based hash function due to privacy

[1] https://adexchanger.com/data-exchanges/a-marketers-guide-to-cross-device-identity/
[2] http://cikm2016.cs.iupui.edu/cikm-cup/

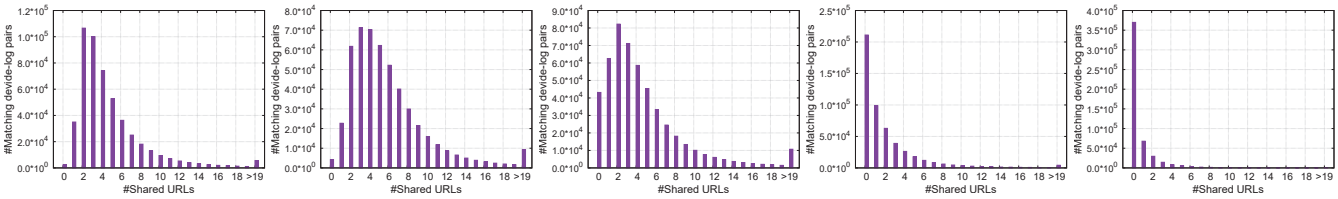
SIGIR'17, August 7–11, 2017, Shinjuku, Tokyo, Japan.

978-1-4503-5022-8/17/08…$15.00
DOI: http://dx.doi.org/10.1145/3077136.3080682



(a) Number of devices owned by each user (b) Number of domains visited by users (c) Similarity based on 24-hourly time slot

Figure 2: Distributions of user vs devices, domains visited by users, and temporal similarity. (Best viewed in color)



(a) Domain (or first-level 'a') (b) Second-level 'a/b' (c) Third-level '$a/b/c$' (d) Forth-level '$a/b/c/d$' (e) Fifth-level '$a/b/c/d/e$'

Figure 3: Number of common URLs in matching pairs of device-logs at each resolution.

Table 1: Dataset overview and statistics of dataset partitions.

(a) Overview of the dataset

Number of device-logs	338,990
Number of events	66,808,490
Average/Median number of events per device-log	197 / 106
Number of URLs	14,148,535
Number of URLs having title	1,796,480
Number of unique words in titles	8,485,859

(b) Statistics of dataset partitions

Statistics	**Training**	**Validation**	**Test**
Number of device-logs	240,732	50,146	48,112
Number of users	60,001	12,528	11,993
Number of golden links	506,136	107,654	107,653

concern. Golden labels are given for device-logs in training and validation sets, and matching predictions are required for the ones in test set. Tables 1(a) and 1(b) report the statistics of the dataset.

A golden link in Table 1(b) means a pair of matching device-logs from the same user. Figure 2(a) plots the number of users having 2, 3, and more devices based on the golden links in training data. It shows that most of users have 3 to 5 devices, and very few has more than 20. Next, we present three observations made from the device-logs in training set.

Observation 1. *Users generally visit a small set of websites (or domains) on each device.*

Given an encoded URL '$a/b/c/\ldots$', we call 'a' the domain of the website (and also the first-level address). Figure 2(b) plots the distribution of number of domains in device-logs. It shows that more than 50% users visit fewer than 10 domains on each device, and only 22.5% users visit more than 30 domains. On average, user visits about 15 different domains on each device.

Observation 2. *Same users do visit some same websites across different devices.*

It is not unexpected to observe that users do visit some same websites on different devices. From the 506,136 golden links (*i.e.,* pairs of device-logs from the same users) in the training set, we plot the number of common domains found in these matching pairs in Figure 3(a). Most users share at least two common domains when surfing on different devices. Furthermore, Figures 3(b) to 3(e) detail the distribution of common URLs at different resolutions. Observe that device-logs of same users generally have common URLs at first, second and third levels. Taking the full URL addresses does not help much in identifying same users.

Observation 3. *Users demonstrate similar temporal usage patterns on different devices.*

We presume that same users demonstrate similar usage patterns across devices, *i.e.,* users functioning at evening time using a device are more likely to active on other devices as well. We categorize events from each device-log into 24 hourly time slots, based on visiting time. Temporal similarity between a pair of device-logs is computed by cosine similarity of number of events in these time slots. Figure 2(c) plots the ratio of positive pairs (from the same user) and negative pairs (from different users) of device-logs, at each similarity interval. It shows that same users tend to have similar usage patterns between devices rather than different users.

3 PAIRWISE CLASSIFICATION APPROACH

We view each device-log as a '*document*' and regard the problem of cross-device user linking as a pairwise classification of documents. Two documents are matched, denoted by $d_i \equiv d_j$, if the two device-logs belong to same user. The key challenge here is to identify features for matching the two documents. To this end, we derive four types of similarity features for document pairs, to be detailed shortly. Two similarities are used to select candidate documents for a given document d_i, then a leaning-to-rank model trained on

all similarity features are used to predict the matching likelihood between d_i and each candidate document d_j.

3.1 Bag-of-URL Similarity

Based on Observations 1 and 2, common URLs are key evidences in cross-device user linking. A straightforward option is to regard each URL in a device-log as a term in a document, and then compute cosine similarity between the two documents using TF-IDF weighting scheme. There are two issues here.

First, Observation 2 indicates that forth-level, fifth-level, and more-detailed resolutions of URLs are not helpful in identifying matching users. To capture URL matching at different resolutions, we map each URL into three terms by taking its first-, second-, and third-level paths. For example, URL '$a/b/c/d/...$' is represented by three terms 'a', 'a/b', and '$a/b/c$'.

Second, users visit a website many times within a very short time period due to their temporal behaviour (or interestingness). However, this behaviour may unlikely present in other devices, hence it does not benefit the matching. For example, users may visit facebook.com on mobile more frequently than on laptop or vice versa. Therefore, it is more important to estimate the willingness of visiting some URLs when they use the device. Inspired by the studies in web usage mining [4], we split a device-log into sessions with a thirty-minute timeout. That is, if two consecutive URLs are distant by more than 30 minutes, they are separated into two sessions. We then introduce a TF-IDF like weighting scheme, namely *SF-ML* (Session Frequency - Matching Likelihood).

Session Frequency. In order to weaken the effect of term frequency (TF) in the traditional TF-IDF scheme, we propose session frequency (SF) weighting scheme. The SF counts the frequency of each term once even if it occurs multiple times in a session. In other words, SF is equal to the number of sessions containing term t within document d (*i.e.,* $s_t = |\{sec_i \in d : t \in sec_i\}|$). As a result, the local weight of term t is expressed as follows:

$$SF(t,d) = 1 + \log(s_t) \tag{1}$$

Matching Likelihood. Acting like the global term weight (IDF) component in the TF-IDF scheme, the matching likelihood (ML) reflects the discrimination of terms. Terms are more discriminative if they are more likely to be found in matching pairs of documents. We denote the matching likelihood of term t as $M(t)$. $M(t)$ reflects how likely the two documents are matched given that both documents contain t. $M(t)$ is pre-computed from the training data by the following formula:

$$M(t) = P(d_i \equiv d_j | t \in d_i \wedge t \in d_j) = \frac{N_t^* + 1}{N_t + 1} \tag{2}$$

- N_t is the number of document pairs containing term t. Let n_t be number of documents containing t, then $N_t = n_t \times (n_t - 1)/2$.
- N_t^* is the number of *matching* document pairs containing t, and $N_t^* = |\{(d_i, d_j) : t \in d_i \wedge t \in d_j \wedge d_i \equiv d_j\}|$. Note that N_t^* is zero for any new terms not appearing in training documents.

Obviously, the global term weight should be proportional to $M(t)$:

$$ML(t,D) = 1 + \log(N \times M(t)) \tag{3}$$

in which N is the number of documents in corpus D.

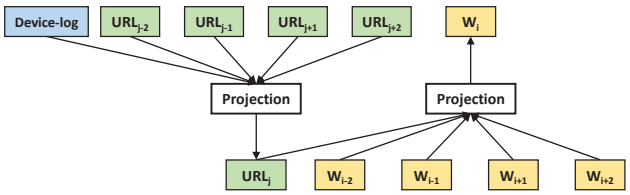


Figure 4: Framework for training device-log embeddings by using URLs and title tokens W_i. (Best viewed in color)

Finally, the ***SF-ML* weight** of term t in document d (given the corpus D) is defined as follows:

$$SF\text{-}ML(t,d,D) = SF(t,d) \times ML(t,D) \tag{4}$$

3.2 Semantic Device-log Embedding Feature

Previous sub-section describes the 'Bag-of-URL' model based on the websites visited in each device-log. In this sub-section, we introduce a neural network model used to learn the embedding of device-log by incorporating the words in title as well as the URLs visited in each session. Following Le *et al.* [1], we assume that two URLs to be 'semantically' similar if they frequently co-occur with same other URLs (*i.e.,* the context).

In the model architecture depicted in Figure 4, a URL not only is used to predict the surrounding URLs but also serves as the global context of the word sequence in its title. On the other hand, the context of device-log unit is shared among all the visited URLs. Overall, the framework contains two layers. One layer models the words' context and its dependency with the URLs, and the other layer models the relation between URLs and derives the representation of device-log. Similar to the Bag-of-URL model, we use the cosine similarity between two device-log's representations as a feature.

3.3 Probabilistic Matching Feature

As mentioned in Section 3.1, some URLs are more discriminative than others. If many of these URLs co-occur in two device-logs, the associated devices are likely to be owned by same user. Next, we define probabilistic matching features designed based on the discrimination of URLs. Note that 'term' in this subsection refers to a URL path at any level (*i.e.,* it is not restricted to the first three levels as in Section 3.1).

Term Matching. Let $T_{i,j} = \{t_1, t_2, ..., t_m\}$ be the set of common terms appear in a pair of documents (d_i, d_j), we compute the probability that the two documents are matching:

$$P_{TM}(d_i \equiv d_j | T_{i,j}) = \prod_{t_k \in T_{i,j}} M(t_k) \tag{5}$$

where $M(t_k)$ is the matching likelihood defined in Equation 2 and is calculated based on the training data.

Personal Domain Matching. Some terms (URLs) are absolutely discriminative, meaning that if they are found in a pair of device-logs, the logs belong to same user in the training data. We call such kind of URLs personal addresses. Furthermore, we define term t to be a *personal domain* if there exists term in the form of t/x (*i.e.,* next level URL), and t/x is *personal address*.

Taking the example of personal mailbox URL, the evidence of two device-logs containing common account address (*e.g.,* http:

Table 2: Performance of different models on test set; best scores are in boldface and second-best are underlined.

Model	Precision	Recall	$F1$
TF-IDF (baseline)	0.245	0.371	0.295
SF-ML (baseline)	0.258	0.383	0.308
CIKM Cup (1st place solution)	0.721	0.529	0.610
GBUL (all features)	<u>0.786</u>	**0.551**	**0.648**
GBUL (replace SF-ML with TF-IDF)	0.748	0.538	0.626
GBUL (without Embedding feature)	**0.803**	0.531	0.639
GBUL (without P_{TM} feature)	0.737	0.523	0.612
GBUL (without P_{PDM} feature)	0.781	<u>0.546</u>	<u>0.643</u>
GBUL (without Time feature)	0.736	0.533	0.618

//webmail.com/acc/user1) hints that the devices are owned by a same person. In this example, http://webmail.com/acc/user1 is a personal address and http://webmail.com/acc is a personal domain.

Different from personal address, personal domain can be shared among many users, therefore it can be used as an indicator for the matching task. We first define the likelihood of a URL t to be *personal domain* as follows:

$$PD(t) = \frac{|\{t' \in T : M(t') = 1 \wedge upper(t') = t\}|}{|\{t' \in T : upper(t') = t\}|} \quad (6)$$

where $upper(t')$ returns the URL at one-level upper of t' (*e.g.,* $upper(\text{`}a/b/c\text{'}) = \text{`}a/b\text{'}$). Consequently, given a set of common URLs C between two device-logs, the likelihood of matching is estimated based on the personal domain likelihoods of theirs antecedent URLs, expressed as follows:

$$P_{PDM}(d_i \equiv d_j | T_{i,j}) = \prod_{t_k \in T_{i,j}} PD\left(upper(t_k)\right) \quad (7)$$

In our model, P_{TM}, P_{PDM} and the number of commons URLs (*i.e.,* $|T_{i,j}|$) are included as features for matching.

3.4 Time-related Feature

Since same users tend to have similar usage patterns on their devices (see Observation 3), we include the similarities of usage patterns between a pair of device-logs as features. The usage patterns are calculated by aggregating the number of visits on hourly (24 bins) and weekly-hourly (24-hour for each day of the week, *i.e.,* 24×7 bins) basis.

4 EXPERIMENT

Setting. To learn profile embedding, we utilize the same technique for training paragraph vector in [1]. Specifically, we apply the continuous skip-gram model with dimension set to 300, window size is 10, and 5 training iterations. We use *Gradient Boosted Regression Tree* (GBRT) as the learning-to-rank model. The detailed settings of the model can be found in the source code.[3]

Following [3], we limit the candidate set for each device-log to the union of 18 nearest device-logs by cosine similarity based on SF-ML (Section 3.1) and semantic embedding (Section 3.2) representation. For each positive pair in training, we randomly select three other pairs from the candidate pairs to serve as negative samples.

Given the pairs of device-logs ranked by matching confidence from GBRT, pair selection is performed by taking the top ranked pairs until a threshold is met (predefined based on validation set).

Evaluation. We use the standard Precision (P), Recall (R) and F1 measures, where F1 is the main metric for evaluation. Different from the calculation used in CIKM Cup where 50% of the correct pairs are used to report the result in each phase of the competition, we use 100% golden labels in the test set to compute P, R and F1.

We name our proposed model as **GBUL** (for using Gradient Boosting on Cross-Device User Linking task). We compare GBUL with three baseline models. The first two baseline models use the traditional TF-IDF and the proposed SF-ML weighting scheme respectively to represent each device log. In the baseline models, cosine similarity is used for ranking and selecting matching pairs. The third baseline is the best solution in the CIKM Cup [3].

Results. With the newly designed features, our GBUL model outperforms the best solution in the CIKM Cup by a large margin on F1 (see Table 2). More than 100% improvement is achieved compared to the two baselines based on TF-IDF and SF-ML, respectively.

We analyse the effectiveness of the proposed SF-ML by replacing it with TF-IDF. As a result, the F1 drops from 0.648 to 0.626. Furthermore, our results show that the probabilistic term-matching feature (P_{TM}) and the time-related feature serve as significant feature. F1 drops by a large margin without either features. Moreover, it is not unexpected that including the neural embedding similarity as a feature will increase the recall of the model, with the price of precision degradation. Overall, using the embedding feature helps to increase F1 score from 0.639 to 0.648.

5 CONCLUSION

In this study, we analyse the new dataset about cross-device user linking and made three observations. We then carefully designed features from device-logs, to train learning-to-rank model for user identity prediction. Our model outperforms the best solution in the CIKM Cup. Based on our results, we will consider both neural network model and probabilistic model in our future study to improve the performance.

Acknowledgement: This work was supported by Singapore Ministry of Education Research Fund MOE2014-T2-2-066.

REFERENCES

[1] Quoc V. Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents. In *ICML, Beijing, China*. 1188–1196.
[2] Jianxun Lian and Xing Xie. 2016. Cross-Device User Matching Based on Massive Browse Logs: The Runner-Up Solution for the 2016 CIKM Cup. *CoRR* abs/1610.03928 (2016).
[3] Minh C. Phan, Yi Tay, and Tuan-Anh Nguyen Pham. 2016. Cross Device Matching for Online Advertising with Neural Feature Ensembles : First Place Solution at CIKM Cup 2016. *CoRR* abs/1610.07119 (2016).
[4] Jaideep Srivastava, Robert Cooley, Mukund Deshpande, and Pang-Ning Tan. 2000. Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data. *SIGKDD* 1, 2 (2000), 12–23.

[3] https://github.com/minhcp/GBUL