On Detecting Frauds in Comparison-Shopping Services

Sang-Chul Lee
Carnegie Mellon University
Pittsburgh, PA, USA
sclee12@cs.cmu.edu

Christos Faloutsos
Carnegie Mellon University
Pittsburgh, PA, USA
christos@cs.cmu.edu

Dong-Kyu Chae
Hanyang University
Seoul, Korea
kyu899@hanyang.ac.kr

Sang-Wook Kim*
Hanyang University
Seoul, Korea
wook@hanyang.ac.kr

ABSTRACT

In *comparison-shopping services* (*CSS*), there exist *frauds* who perform *excessive clicks* on a target item in order to boost the *popularity* of it. In this paper, we introduce the problem of detecting frauds in CSS and propose three anomaly scores designed based on *click behaviors* of users in CSS.

Keywords

Fraud detection; User behavior analysis

1. INTRODUCTION

Recently, the number of shoppers using *comparison-shopping services*(*CSS*), such as Shopping.com, PriceGrabber.com, and Shopping.naver.com, is increasing rapidly due to their convenience. Given a query with some keywords, CSS provides a comprehensive comparison of items in terms of their prices and features, arranged in order of the items' popularity and relevance. This makes shoppers conveniently compare the items and decide what to buy among them just by clicking one in CSS, without visiting a number of e-commerce sites scattered over the Internet [3].

However, in CSS, the popularity of an item can be easily manipulated by some fraudulent sellers: CSS just redirects its shoppers to individual e-commerce sites by providing a link to the page built for buying the item (i.e., features, price, and transaction) maintained in those sites; it is unaware of whether the item has been purchased because the purchase happens in e-commerce sites rather than CSS. This makes CSS evaluate the popularity of an item only relying on the number of clicks on the item. Subsequently, this motivates fraudulent sellers to click their items excessively in CSS to manipulate the rankings of their items in search or recommendation results, rather than relying on traditional marketing solutions [6]. Such fraudulent actions may result in a significantly distorted quality of search and recommendation services in CSS.

This paper addresses the problem of detecting such frauds in CSS, in an unsupervised manner. Since their behaviors

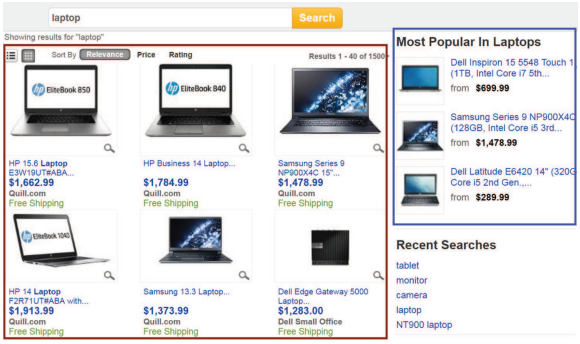

Figure 1: The search (shown in the left box) and the recommendation (shown in the right box) results of a query with "laptop", provided by Shopping.com.

are quite different from that those of the frauds in different domains, such as click frauds in advertisement networks [2], ranking frauds [6] or rating frauds [5] in online shops, and outlier detection in online social networks [1], a novel method specific to detect frauds in CSS is required.

2. ANOMALY SCORES

Given a user u, our goal is to quantify her degree of anomalous behavior compared to normal users in the range of 0 and 1. To this end, we propose and examine three anomaly scores: *inter-arrival time difference* (a_u^{IAT}), *diurnal activity difference* (a_u^{DA}), and *eigenscore difference* (a_u^{ES}).

Inter-arrival time (*IAT*) indicates *the time interval between a pair of successive clicks* conducted by an individual user. We expect that the *IAT* distribution of frauds is very different from that of normal users. For a user u, we first define her *IAT* vector I_u, where the dimension corresponds to the length of *IAT* and its value does the ratio of pairs of successive clicks having the corresponding *IAT* to all of u's pairs of successive clicks. Also, we set the dimensionality of I_u as 1,200, which indicates u's session (i.e., the sequence of her clicks during a single visit) is regarded to end if she does nothing for 20 minutes (i.e., 1,200 seconds). We then define a standard *IAT* vector for normal users, I_{normal}, by averaging the values obtained from all users' I_u for each dimension. Then, we compute the distance (i.e., difference) between the two vectors I_{normal} and I_u in order to compute a_u^{IAT}. We employ the Kullback-Leibler divergence as our distance function defined in the following:

$$D_{KL}(I_u||I_{normal}) = \sum_i I_u(i) log \frac{I_u(i)}{I_{normal}(i)} \quad (1)$$

*Corresponding author

WWW'17 Companion, April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4914-7/17/04.
http://dx.doi.org/10.1145/3041021.3054219

where $D_{KL}(I_u||I_{normal}$ indicates Kullback-Leibler divergence of I_u from I_{normal} and i does the index of elements for both vectors. Since D_{KL} is an asymmetric function, we take the average of $D_{KL}(I_u||I_{normal}$ and $D_{KL}(I_{normal}||I_u)$ to compute a_u^{IAT}. Finally, we normalize a_u^{IAT} of every user in the range of 0 and 1 by min-max normalization.

We also expect that *Diurnal activity* (DA) is quite useful to understand behavioral differences between frauds and normal users. For each user, we discretize the timestamp of her clicks in unit of hour and define her DA vector D_u, which is a 24-dimensional vector where a dimension indicates time in a day and its value does the ratio of u's clicks made in the corresponding time to u's all clicks. In the same way, we define a standard DA vector for normal users, D_{normal}, by averaging the values obtained from all users' D_u for each dimension. Then, we compute a_u^{DA} by taking the average of $D_{KL}(D_u||D_{normal})$ and $D_{KL}(D_{normal}||D_u)$, and finally do min-max normalization on a_u^{DA} in the range of 0 and 1.

In order to compute a_u^{ES}, we first conduct *singular vector decomposition* (SVD) on the clickstream dataset, which is represented as a $p \times q$ matrix where p is # of users, q is # of items $\times$ # of days, and each entry corresponds to a user's # of clicks on an item in a day. As a result, we obtain two decomposed, low-rank matrices U and V. Note that each column-vector in the left matrix U represents the *degree of relevance* of the corresponding user to each fraudulent pattern [4]. Among the values in each vector, we choose the highest value, which becomes the *eigenscore* for each user. We finally compute the distance between the average of all users' eigenscores ($es_{average}$) and user u's eigenscore (es_u) ($a_u^{ES} = |es_u - es_{average}|$).

3. EVALUATION

We now present the experimental results of evaluating the accuracy of our proposed anomaly scores. We used a click log dataset obtained from *Naver shopping*, one of the biggest CSS in Korea. It consists of 10K users, 301,840 items, and 422,610 click events traced in a period of eight months. Each click event is characterized by <userID, itemID, timestamp>, indicating userID clicked itemID at the time of the timestamp.

Since there is no ground truth data of frauds in CSS, we generated synthetic frauds and injected them into our clickstream dataset. Specifically, we considered three types of frauds, *bot*, *burst*, and *low temperature*, all of which may possibly exist in the real-world. We summarize the characteristics of three types of frauds in Table 2. Every fraud selects X target items and clicks Y times for each target item. The value of X was chosen randomly from [2, 4] for each fraud, and the value of Y was chosen randomly from [150, 250] for each fraud-item pair.

Table 1: Characteristics of created frauds

	Bot	Burst	Low temperature
click interval	1–30 seconds	2–5 seconds	5–15 seconds
duration of a session	*Having one session during its lifetime*	5–15 minutes	5–15 minutes
# of sessions per day		4–8 times	1 time
Interval between sessions		30–60 minutes	*Having one session in a day*
Starting time of 1st session in a day		12–2 PM	1–10 PM

We created 25 frauds for each fraud type (e.g., 25 *bursts*) and injected them into our dataset. Then, we measured every user's suspiciousness score and determined the top-k users as frauds. As accuracy metric, we employed *mean average precision* (MAP). To avoid the bias from randomness, we carried out 1,000 experiments with different sets of generated frauds and took the average of 1,000 MAPs. We compared the MAP of our proposed Φ_{or} and Φ_{and} with that of four baselines. The three of baselines are the solely-used anomaly scores, a_u^{IAT}, a_u^{DA}, and a_u^{ES}, and the other one is an intuitive anomaly score, denoted as a_u^{clicks}, which simply combines (1) the average number of clicks *per item* and (2) the average number of clicks *per day*. The results are shown in Figure 2.

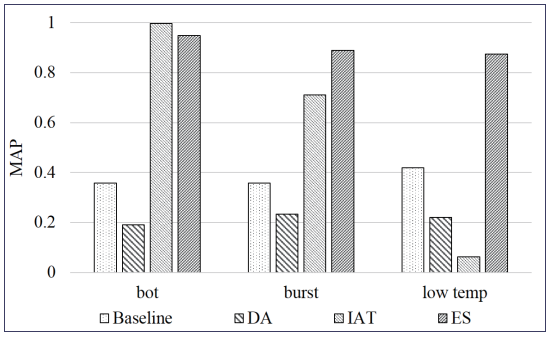

Figure 2: MAPs with different measures.

a_u^{clicks} is shown to provide low accuracy, which demonstrates that it is insufficient to determine whether a user is a fraud by considering just the number of clicks per item or per day. This is because there exist *hard shoppers*, who repeatedly click several items for comparison purpose but not with the fraudulent purpose. a_u^{IAT} is the best performer in detecting *bots*, but misses several *bursts* and most *low temperatures*. This is because the IAT of *low temperature* resembles that of normal users. a_u^{ES} captures more than 87.5% of frauds for all the fraud types, missing just a small portion of frauds. a_u^{DA} is shown to provide very poor accuracy when used alone. However, we observe that a_u^{DA} identifies several frauds that a_u^{IAT} and a_u^{ES} miss, which implies its potential usefulness.

4. ACKNOWLEDGEMENTS

This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. NRF-2014R1A2A1A10054151).

5. REFERENCES

[1] D.-H. Bae et al. Outlier detection using centrality and center-proximity. In *ACM CIKM*, pages 2251–2254, 2012.

[2] G. Cho et al. An empirical study of click fraud in mobile advertising networks. In *ARES*, 2015.

[3] M. Gupta et al. Characterizing comparison shopping behavior: A case study. In *ICDEW*, 2014.

[4] M. Jiang et al. A general suspiciousness metric for dense blocks in multimodal data. In *ICDM*, 2015.

[5] H.-K. Oh et al. Can you trust online ratings? a mutual reinforcement model for trustworthy online rating systems. *Systems, Man, and Cybernetics: Systems, IEEE Transactions on*, 2015.

[6] H. Zhu et al. Discovery of ranking fraud for mobile apps. *IEEE TKDE*, 2015.