Information Sciences 430–431 (2018) 314–330

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Finding the hottest item in data streams[☆]

Huaizhong Lin[a,*], Shanshan Wu[a], Leong Hou U[b], Ngai Meng Kou[b], Yunjun Gao[a], Dongming Lu[a]

[a] *College of Computer Science and Technology, Zhejiang University, Hangzhou, China*
[b] *Faculty of Science and Technology, University of Macau, Macau, China*

ARTICLE INFO

Article history:
Received 20 October 2016
Revised 5 October 2017
Accepted 8 November 2017
Available online 15 November 2017

Keywords:
Online algorithm
Hottest interval
Item stream

ABSTRACT

We study a problem of finding the hottest item interval in a data stream, where the hotness of an item over an interval is determined by its average frequency. Finding the hottest item interval is particularly helpful in business promotions, such as monitoring the peak sales records, finding the hottest period in an online game, digging the highest click rate of an online music, etc. Existing work focus on finding the most frequent item over a fixed length interval. However, these solutions cannot return the hottest interval since the best length (i.e., maximizing the average frequency) is unknown in advance. To discover the hottest item interval, a straightforward solution is to calculate the average frequencies of items for every possible interval length, which is too costly for stream applications. To efficiently compute the hottest item interval, we propose an algorithm that employs the arrival timestamps of items and reduce the search space by three pruning strategies. Extensive experiments show that the proposed algorithms can efficiently discover the hottest item interval on both real and synthetic datasets.

1. Introduction

A data stream is a sequence of events that can be read only once (or limited times) using constrained computing and storage capabilities. The data stream exists in many applications especially when the application itself continuously generates or collects data, such as sensor streams [2,13], financial monitoring streams [16,21], biomolecular streams [3,10], etc. Due to the stream volume, substantial analytical tasks have been developed to extract the underlying knowledge of the stream data, including clustering [8,9,14,22], classification [24], mining frequent patterns [7,19,23,26,27], estimating mutual information [17], etc.

The problem studied in this work is to find the hottest item in a data stream. Let A be the item set and S be a data stream of items and $S_{i,\,j}$ denote a sub-stream of the interval from i to j. Suppose t is the current timestamp, our task is to identify the hottest item a such that the average weight of a is the highest among any items in any sub-streams, i.e.,

$$\underset{a\in A\wedge S_{i,j}\subseteq S_{0,t}}{\operatorname{argmax}}\; f(a,[i,j])=\frac{\sum_{i\le t\le j} w_a[t]}{\ell(i,j)} \tag{1}$$

[☆] This manuscript is the authors' original work and has not been published nor has it been submitted simultaneously elsewhere.

[*] Corresponding author.

E-mail addresses: linhz@zju.edu.cn (H. Lin), wuss@zju.edu.cn (S. Wu), ryanlhu@umac.mo (L. Hou U), yb27406@umac.mo (N.M. Kou), gaoyj@zju.edu.cn (Y. Gao), ldm@zju.edu.cn (D. Lu).

https://doi.org/10.1016/j.ins.2017.11.012
0020-0255/© 2017 Elsevier Inc. All rights reserved.

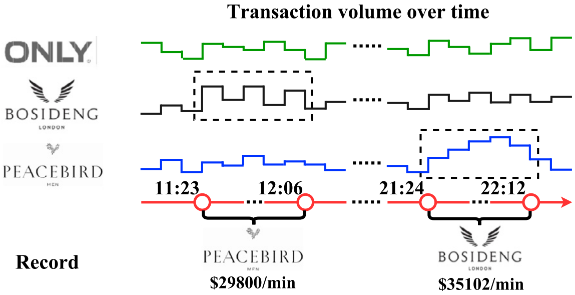
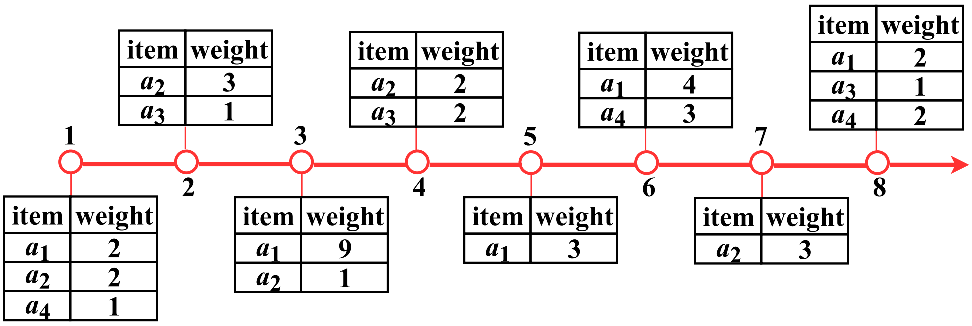


Fig. 1. Example of marketplace transactions.



Timestamp 1:

item	weight
a_1	2
a_2	2
a_4	1

Timestamp 2:

item	weight
a_2	3
a_3	1

Timestamp 3:

item	weight
a_1	9
a_2	1

Timestamp 4:

item	weight
a_2	2
a_3	2

Timestamp 5:

item	weight
a_1	3

Timestamp 6:

item	weight
a_1	4
a_4	3

Timestamp 7:

item	weight
a_2	3

Timestamp 8:

item	weight
a_1	2
a_3	1
a_4	2

Fig. 2. A data stream of 8 timestamps and more than one item may arrive at each timestamp.

where $w_a[t]$ indicates the weight of item a arriving at timestamp t and $[i, j]$ is interval from timestamp i to timestamp j. For fairly comparing the sub-streams of different lengths, we divide the sum of weights by its *window length* $\ell(i, j)$ where $\ell(i, j) = j - i + 1$.

The hottest item problem can be viewed as a monitoring problem that keeps tracking the best performing item over time. In the following we first discuss some potential applications and the challenges will be discussed shortly.

Application 1. To rev up the publicity of an online marketplace during a big promotion period (e.g., Singles' Day in China or Thanksgiving Day in United States), the marketing team may update the most representative brand on their web site in timely manner. As an example in Fig. 1, *PeaceBird* received 1.28 M USD in 43 min from 11:32 to 12:06 (i.e., 29,800 USD per minute) on 11 Nov. 2015. This record was broken by *BOSIDENG*, who received 1.68 M USD in 48 min from 21:24 to 22:12 (i.e., 35,102 USD per minute). To attract more customers, the marketing team can advertise on their web site that *BOSIDENG* surpasses *PeaceBird* and reaches to a new record in the history.

Application 2. Another example is to find the peak period of user logins in a game platform (e.g., steam). Suppose that *Team Fortress 2* (TF2) had been playing by 0.1 million users in 4 h (i.e., 25,000 users per hour) which is the recorded peak period in the history. However, this record was broken by *Counter Strike Global Offensive* (CSGO) over the period of the Season 5 playoff, who had been playing by 0.1855 million users in 7 h (i.e., 26,500 users per hour). Valve Corporation (the developer of CSGO) can announce this news that the number of login users reaches to a new record in the online game history of steam platform.

Application 3. Monitoring the most listened song in an online music store is particularly helpful for the online business. We may attract more users to download a popular song if it hits some historical records (the hottest item interval), e.g., the top-5 most listened songs per hour.

Challenges. Identifying the hottest item can improve the business of the above applications. However, this problem is challenging since (1) the data arrive rapidly in these applications (e.g., huge transactions during the peak period of the online marketplace example) and (2) the window length of the hottest interval is unknown in advance (e.g., CSGO may not find the historical record if the length is set to 6 h). We use the following concrete example to show how to identify the hottest item interval by a brute-force solution.

Fig. 2 presents a data stream of 8 timestamps and more than one item may arrive at each timestamp. Suppose the minimum window length $\delta = 3$, at timestamp 3 the hottest item is a_1 where $f(a_1, [1, 3]) = \frac{11}{3} = 3.67$ is the highest among any seen items and any seen intervals. This record is broken by the same item at timestamp 6, where $f(a_1, [3, 6]) = \frac{16}{4} = 4$. To identify the hottest item at timestamp t, a brute-force solution is to calculate function $f(\cdot)$ for every item and every possible interval. Assuming that $|S|$ is the size of the stream and m is the number of distinct items, the overall complexity of this brute-force solution is $O(|S|^2 m)$, which is too costly for stream applications.

In this work, we attempt to boost the process for the hottest item identification. Our main contributions can be summarized as follows: