Supplementary Information: Multiple Dense Subtensor Estimation with High Density Guarantee

Anonymous Author(s)

1 REPRODUCIBLE SUPPLEMENT

In this supplement, we present additional information in more detail for reproducibility the experimental results in the original paper. Source code and data used in the paper are provided and made publicly at https://bitbucket.org/l33ts/mtensor.

1.1 Example of dense subtensor detection

This subsection presents an example of tensor data and a dense subtensor in the input data with its density.

Example 1.1. Let's consider an example of 3-way tensor T as in Figure 1. The value in each cell is the number of visits that a user (mode User) visit a web page (mode Page) on a date (mode Date). The values of hidden cells are all zeros. The set of slices of tensor T is {(1,1), (1,2), (1,3), (2,1), (2,2), (2,3), (3,1), (3,2)}. A subtensor Q formed by the following slices {(1,2), (1,3), (2,1), (2,2), (3,1)} is the densest subtensor and the density of Q is (5+5+7+2)/5 = 3.8.

Meanwhile in Figure 2, it is an example of 2-way tensor. The tensor in Figure 2, T, can be represented as a matrix. The yellow region in the tensor is the densest subtensor in T, and its density is $\frac{8+5+5}{4} = 4.5$. The density of the subtensor Q formed by the first three columns and the first three rows is $\frac{(8+5+1+5+1+3)}{6} = \frac{23}{6}$.

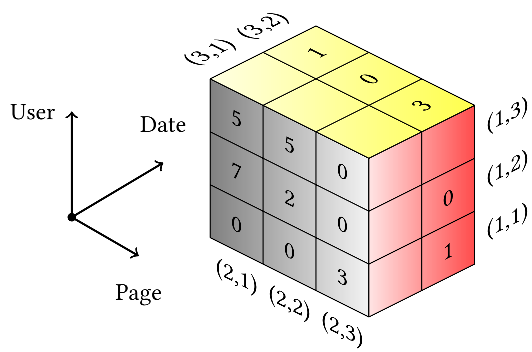

Figure 1: An example of 3-way tensor.

8	5	1	1	1	...
5	0	0	0	0	...
1	0	3	5	0	...
1	0	5	0	5	...
1	0	0	5	0	...
...	...	...	...	...	...

Figure 2: An example of 2-way tensor.

1.2 Datasets

In our experiments, we used 10 real-world datasets, and the datasets are following.

- *Air Force* dataset, which contains TCP dump data for a typical U.S. Air Force LAN. The dataset was modified from the KDD Cup 1999 Data and provided by [7].
- *Android* dataset, which contains product reviews and rating metadata of applications for Android from Amazon [2].
- *Enron Emails* dataset provided by the Federal Energy Regulatory Commission to analyze the social network of employees during its investigation of fraud detection and counter terrorism.
- *StackO* dataset, which represents data of users and posts on the Stack Overflow. Each instance contains the information of a user marked a post as favorite at a timestamp [3].
- *Enwiki* and *Kowiki* datasets provided by Wikipedia[1]. Enwiki and Kowiki are metadata presenting respectively number of user revisions on a page at time (in hour units) in English Wikipedia and Korean Wikipedia. Enwiki and Kowiki are up to the date of 20th Nov 2018, and we chose the largest dump files of Enwiki.
- *YouTube* dataset, which consists of the friendship connections between YouTube users [5].
- *Darpa* dataset, which is a dataset collected by MIT Lincoln Lab to evaluate the performance of intrusion detection systems (IDSs) in corporation with DARPA [4].
- *NIPS Pubs* dataset, which contains papers published in NIPS[2] from 1987 to 2003 [1].
- *LBNL-Network* dataset, which consists of internal network traffic captured by Lawrence Berkeley National Laboratory and ICSI [6]. Each instance contains the packet size that a source (ip, port) send to a destination (ip, port) at a time.

We selected these datasets because of their diversity, and because they are widely-used as benchmark datasets in the literature [7, 8].

The *Darpa* dataset was provided in the prior work, DenseAlert [8], and we downloaded the dataset at this address[3]. The *StackO* and *YouTube* were directly downloaded from The Koblenz Network Collection repository [3], and we got the datasets at this

[1] https://dumps.wikimedia.org/
[2] https://nips.cc/
[3] http://www.cs.cmu.edu/ kijungs/codes/alert/data/darpa.zip

address[4]. The *Enron Emails*, *NIPS Pubs*, and *LBNL-Network* were directly downloaded from an open source project, The Formidable Repository of Open Sparse Tensors and Tools (FROSTT) [9], at this address[5]. The *Android* dataset was obtained from Stanford Network Analysis Project at this address[6]. The Kowiki and Enwiki were downloaded from Wikipedia. The *Air Force* was modified from the KDD Cup 1999 Data[7]. We kept fields which are described in Table 3 in the main paper, while other fields were removed. Detail of our tool is provided later in the next paragraphs.

The datasets in the experiments will be provided by authors upon request.

1.3 Implementation

Our implementation has two parts. The first part is our tool for preprocessing Kowiki and Enkiwi datasets. The second part of our implementation is source code of the algorithms, MUST, M-Zoom, and M-Biz. Source code of the M-Zoom and M-Biz were provided by the authors at this address[8]. All the experiments were carried out on a personal computer running Windows 10 as operating system, having a 64 bit Intel i7 2.6 GHz processor and 16GB of RAM.

Preprocessing Tool. The tool is implemented in Python[9], and we run the tool with Python version 3.6.

(1) The tool is designed to convert a metadata XML struture file to a tensor format, e.g. *(user, page, time, #revisions)*.
(2) run *python wiki-preprocess.py*.

Configuration Settings:

(1) Input Arguments:
  (a) *input_file*: file name of the dataset being processed.
  (b) *output_file*: file name of the result dataset.
(2) DateTime setting format: %Y-%m-%d %H:%M:%S.
(3) Output: A file with tensor format *(user, page, time, #revisions)*.

Algorithms. MUST, M-Zoom, and M-Biz were implemented in Java with Eclipse IDE.

(1) Usage: The package is implemented to maintain multiple subtensors in an input tensor to maximize density of the estimated subtensors.
(2) run *Must.java* with arguments, etc. run *'Must.java enwiki.csv output 3 ARI 10'*, this command will return top ten highest density subtensors in Enwiki dataset (which the order of tensor is three) with arithmetic mass by MUST algorithm.

Configuration Settings of Input Arguments: there are five main input arguments of the proposed algorithm. The arguments are as follows:

(1) *input_path*: the path of input tensor file.
(2) *output_path*: the directory for saving estimated subtensors.
(3) *num_of_way*: number of ways of the input tensor.
(4) *density_measure*: there are four values corresponding to four types of density. These values are as follows:
  (a) *'ARI'*: arithmetic mass
  (b) *'GEO'*: geometric mass
  (c) *'SUSP'*: suspiciousness mass
  (d) *'ES_{value of α}'*: entry surplus mass with value of α
(5) *num_of_subtensors*: Number of estimated subtensors.

Output: the algorithm returns *num_of_subtensors* subtensors with theirs density in the input tensor.

In order to load the input tensor from a file, the package uses a function $importTensor$. Parameters of the $importTensor$ function are:

(1) *input_path*: the path of input tensor file.
(2) *delimiter*: is a character to separate fields in the tensor file.
(3) *num_of_way*: number of way of the input tensor.
(4) *entry_field*: position of the entry field of the input tensor.

[4]http://konect.uni-koblenz.de/downloads/tsv
[5]http://frostt.io/tensors/
[6]http://snap.stanford.edu/data/amazon/productGraph/categoryFiles/ratings_Apps_for_Android.csv
[7]http://kdd.ics.uci.edu/databases/kddcup99/kddcup.data.gz
[8]https://github.com/kijungs/mzoom
[9]https://www.python.org/

2 ADDITIONAL PROOF OF THEOREMS AND PROPERTIES IN SECTION 3

This section provides detail proofs of theorems and properties presented in the main paper.

2.1 Proof of Theorem 3.2

Proof. We denote $w_0 = w_{\pi(z_0)}(Z)$. Note that, because T^* is the densest subtensor, so:

$$\forall q \in T^*, w_q(T^*) \geq \rho^* \Rightarrow w_0 \geq \rho^*.$$

Due to the characteristic of D-Ordering, then we have:

$$w_q(Z) \geq w_{\pi(z_0)}(Z) = w_0, \forall q \in Z.$$

Consider a way I_i among the N ways of the tensor T. Then, we have:

$$f(Z) = \sum_{q \in T^* \wedge q \in I_i} w_q(Z) + \sum_{q \notin T^* \wedge q \in I_i} w_q(Z).$$

Furthermore, regarding the way we choose Z, we have:

$$T^* \subseteq Z \Rightarrow \sum_{q \in T^* \wedge q \in I_i} w_q(Z) \geq \sum_{q \in T^* \wedge q \in I_i} w_q(T^*) = f(T^*).$$

Therefore,

$$f(Z) \geq f(T^*) + \sum_{q \notin T^* \wedge q \in I_i} w_q(Z) \geq f(T^*) + b_{I_i} w_0, \qquad (1)$$

where b_{I_i} is the number of slices in Z on dimension I_i that are not in T^*. Let $b = \sum_{i=1}^{N} b_{I_i}$, Exploiting the formula (1) on N ways, we get:

$$\begin{aligned}
Nf(Z) &\geq Nf(T^*) + w_0 \sum b_{I_i} \\
\Rightarrow N(a+b)\rho(Z) &\geq Na\rho^* + w_0 b \\
\Rightarrow N(a+b)\rho(Z) &\geq Na\rho^* + b\rho^* \\
\Rightarrow \rho(Z) &\geq \frac{Na+b}{N(a+b)}\rho^* \qquad \square
\end{aligned}$$

2.2 Proof of Property 1

Proof. We denote the fraction of the density of the estimated subtensor Z and the densest subtensor as R. We have the following properties about this lower bound fraction:

(1) In the simple case, when $N = 1$, the lower bound rate values both in the previous proof and in this proof are 1, it means that the estimated subtensor Z is the densest one and having highest density. Otherwise,
$R \geq \frac{Na+b}{N(a+b)} = \frac{a+b}{N(a+b)} + \frac{(N-1)a}{N(a+b)} > \frac{1}{N}, \forall N > 1.$

Furthermore, since the size of Z is not greater than n, we have:
$R \geq \frac{1}{N}(1 + \frac{(N-1)a}{(a+b)}) \geq \frac{1}{N}(1 + \frac{a(N-1)}{n}).$

(2) In conclusion, we have the following boundary of the density of estimated Zero subtensor Z:
$\rho(Z) = \begin{cases} \rho^*, & \text{if } N = 1 \vee b = 0 \\ \frac{1}{N}(1 + \frac{a(N-1)}{n})\rho^*, & \text{if } a + b = n. \end{cases}$

In an ideal case, when the value of b goes to zero, the estimated subtensor is the densest subtensor, and its density is the highest. □

2.3 Proof of Theorem 3.3

Proof. Because q is a slice having the minimum cut, so we have $w_q(T) \leq w_p(T), \forall p \in T$. Sum up of all the slices in the tensor, we get:

$$|T|w_q(T) \leq \sum_{p \in T} w_p(T) = Nf(T)$$
$$\Rightarrow w_q(T) \leq \frac{Nf(T)}{|T|} = N\rho(T) \qquad \square$$

2.4 Proof of Property 2

Proof. Let M_{I_i}, M_{I_j} be the indexes of the last removed slices of any two ways I_i and I_j, the difference between M_{I_i}, M_{I_j} is $\Delta(M_{I_i}, M_{I_j}) = |M_{I_i} - M_{I_j}| \geq 1$, we have N numbers (N ways) and the maximum (the last index) is a, so we get:

$$\text{Max}(M_{I_i}) - \text{Min}(M_{I_i}) \geq N - 1$$
$$\Rightarrow M = \text{Min}(M_{I_i}) \leq a - N + 1 \qquad \square$$

2.5 Proof of Theorem 3.4

Proof. Let E be any entry of the densest subtensor T^* and E is composed by the intersection of N slices, $q_{I_x} (1 \leq x \leq N)$, q_{I_x} is on way I_x. Assume that the first removed index of all slices composing E is at index i. Definitely, this index is not greater than M, so the entry E is in T_i, and its value is counted in $w_{q_x}(T_x)$. Therefore, we have: $\sum_{i=1}^{M} w_{q_i}(T_i) \geq f(T^*)$ □

2.6 Proof of Theorem 3.5

Proof. According to Theorem 3.2 and Property 1, we have:

$$\rho_{max} \geq \rho(T_1) \geq \frac{1}{N}(1 + \frac{a(N-1)}{n})\rho^* \tag{2}$$

Furthermore, we already have $\rho_{max} \geq \frac{\rho^*}{N}\frac{a}{a-N+1}$, so we also have:

$$\rho_{max} \geq \frac{\rho^*}{N}\frac{a}{a-N+1} \geq \frac{1}{N}(1 + \frac{N-1}{a})\rho^* \tag{3}$$

We combine together two inequations 2-3, we get:

$$\rho_{max} \geq \frac{1}{N}(1 + \frac{1}{2}(\frac{a(N-1)}{n} + \frac{N-1}{a}))\rho^*$$
$$\Rightarrow \rho_{max} \geq \frac{1}{N}(1 + \frac{N-1}{\sqrt{n}})\rho^*$$

Note that $\rho_{max} \geq \frac{1}{N}(1 + \frac{N-1}{a})\rho^*$, so finally we have:

$$\rho_{max} \geq \frac{1}{N}(1 + \frac{N-1}{min(a, \sqrt{n})})\rho^* \qquad \square$$

3 ADDITIONAL PROOF OF THEOREMS AND PROPERTIES IN SECTION 4

3.1 Proof of Theorem 4.2

Proof. From the inequality of density of forward subensor, $\rho(F_i) \geq \frac{Na+b}{N(a+b+i)}\rho^*$.
If we have $i \leq N(N-1)$, hence:

$$\begin{aligned}
\Rightarrow a + b + i &\leq a + b + N(N-1) \\
\Rightarrow \frac{Na+b}{N(a+b+i)}\rho^* &\geq \frac{Na+b}{N(a+b+N(N-1))}\rho^* \\
\Rightarrow \frac{Na+b}{N(a+b+i)}\rho^* &\geq \frac{a+b+a(N-1)}{N(a+b+N(N-1))}\rho^* \\
\Rightarrow \frac{Na+b}{N(a+b+i)}\rho^* &\geq \frac{a+b+N(N-1)}{N(a+b+N(N-1))}\rho^* \\
\Rightarrow \rho(F_i) &\geq \frac{Na+b}{N(a+b+i)}\rho^* \geq \frac{1}{N}\rho^* \qquad \square
\end{aligned}$$

3.2 Proof of Property 4

Proof. We denote $w_0 = w_{\pi(z_0)}(Z)$. Because we assume that $T(\pi, z_0 + 1)$ can form a subtensor with size $(a + b - 1)$, then this size is at least N, i.e., $a + b - 1 \geq N$.

Further more, $T^* \subseteq Z$, then:

$$w_0 \geq w_{\pi(z_0)}(T^*) \geq \rho^*, \text{ and } f(Z) = f(B_1) + w_0.$$

Let us consider the ways $I_i (i \in [1, N])$. We denote by a_i the number of slices in both Z and T^* on way I_i, and b_i the number of slices in Z but are not in T^* on way I_i.

We let I^+ be the set of ways such that $b_i > 0$, $I^+ = \{I_i | \, b_i > 0\}$. Then, on each way I_i in the I^+, we have:

$$f(Z) = \sum_{q \in T^* \wedge q \in I_i} w_q(Z) + \sum_{q \notin T^* \wedge q \in I_i} w_q(Z)$$
$$\Rightarrow f(Z) \geq f(T^*) + b_i \times w_0.$$

Therefore, if we sum up for all ways in $I+$, we get

$$\sum_{I_i \in I^+} f(Z) = \sum_{I_i \in I^+} (f(T^*) + b_i \times w_0)$$
$$\Rightarrow |I^+|f(Z) \geq |I^+|f(T^*) + w_0 \times \sum b_i$$
$$\geq |I^+|f(T^*) + w_0 \times b.$$

Case 1: When I^+ is not empty, $I^+ \neq \emptyset$.

Let $K = |I^+|$, then we have $1 \le K \le N$. The above inequality becomes:

$$\begin{aligned}
&Kf(Z) && \ge Kf(T^*) + w_0 \times b\\
\Rightarrow\ &K(f(B_1) + w_0) && \ge Kf(T^*) + w_0 \times b\\
\Rightarrow\ &Kf(B_1) && \ge Kf(T^*) + w_0(b-K).
\end{aligned}$$

Since $b_i \ge 1, \forall I_i \in I^+ \Rightarrow b = \sum b_i \ge |I^+| = K$, $(b-K) \ge 0$. Plugging this into the above inequality, we have

$$\begin{aligned}
&Kf(B_1) && \ge Kf(T^*) + (b-K)\rho^*\\
\Rightarrow\ &K(a+b-1)\rho(B_1) && \ge Ka\rho^* + (b-K)\rho^*\\
\Rightarrow\ &\rho(B_1) && \ge \frac{K(a-1)+b}{K(a-1+b)}\rho^* \ge \frac{1}{K}\rho^*,
\end{aligned}$$

because of the following reason

$$\frac{K(a-1)+b}{K(a+b-1)} \ge \frac{1}{K} \ge \frac{1}{N}, \forall K \ge 1 \wedge a \ge 1$$

Case 2: When I^+ is empty, $I^+ = \emptyset$.

It means that $Z = T^*$ and $b = 0 \Rightarrow a > N$. Hence, the estimated density of the algorithm can be shown to be equal to the highest density in the tensor, $\rho(Z) = \rho^*$. In this case, $f(B_1) + w_0 = f(Z) = f(T^*)$.

Because w_0 is the smallest weight value of all slices in T^*, $\forall q \in T^*, w_0 \le w_q(T^*)$.

Consider an any-way $I_i, \forall i \in [1, N]$, of the tensor. Then, we have

$$\begin{aligned}
&f(T^*) && = \sum_{q \in I_i \wedge q \in T^*} w_q(T^*) \quad \ge |I_i| \times w_0\\
\Rightarrow\ &w_0 && \le \frac{f(T^*)}{x},
\end{aligned}$$

where $|I_i|$ is the number of slices on way I_i of subtensor T^*, and $x = max(|I_i|)$. Because that

$$\sum_{i=1}^{N} |I_i| = a \Rightarrow x \ge \frac{a}{N} > 1.$$

so we get

$$\begin{aligned}
&f(T^*) && = f(B_1) + w_0\\
\Rightarrow\ &f(T^*) && \le f(B_1) + \frac{f(T^*)}{x}\\
\Rightarrow\ &a\rho^* && \le (a-1)\rho(B_1) + \frac{a\rho^*}{x}\\
\Rightarrow\ &\rho(B_1) && \ge \frac{ax-a}{ax-x}\rho^*
\end{aligned}$$

Moreover, we have:

$$\begin{aligned}
\frac{a(x-1)}{(a-1)x} &= \frac{a}{(a-1)} \times \frac{x-1}{x}\\
&\ge \frac{a}{(a-1)} \times \frac{\frac{a}{N}-1}{a/N}\\
&\ge \frac{a-N}{a-1} \ge \frac{1}{N}, \forall x, N > 1\\
\Rightarrow \rho(B_1) &\ge \frac{1}{N}\rho^*
\end{aligned}$$

□

3.3 Proof of Theorem 4.4

Proof. Note that $f(B_i) = f(B_{i+1}) + w_{\pi(z_0+i)}(B_i)$. Let $B_0 = Z$, and in the following we denote $w_i(B_i) = w_{\pi(z_0+i)}(B_i)$ for short. Then, we have

$$\begin{aligned}
Kf(Z) &= K(f(B_1) + w_0(B_0))\\
&= K(f(B_2) + w_0(B_0) + w_1(B_1))\\
&= Kf(B_k) + K\sum_{i=0}^{k-1} w_i(B_i).
\end{aligned}$$

Because $T^* \subseteq Z$, then:

$$Kf(Z) \ge Kf(T^*) + \sum_{q \in Z \wedge q \notin T^*} w_q(Z), \tag{4}$$

From the above Inequation, we get

$$\begin{aligned}
&Kf(B_k) + K\sum_{i=0}^{k-1} w_i(B_i) \ge Kf(T^*) + \sum_{q \in Z \wedge q \notin T^*} w_q(Z)\\
\Rightarrow\ &Kf(B_k) \ge Kf(T^*) + \sum_{q \in Z \wedge q \notin T^*} w_q(Z) - K\sum_{i=0}^{k-1} w_i(B_i).
\end{aligned}$$

We denote the set $Q = \{q |\ q \in Z \wedge q \notin T^*\}$ by $\{q_1, q_2, \ldots, q_b\}$. Note that $B_i \subseteq Z$, then: $\forall j, i, w_{q_j}(Z) \ge w_{q_j}(B_i) \ge w_i(B_i)$, and $w_{\pi(z_0)}(Z) \ge w_{\pi(z_0)}(T^*) \ge \rho^*$.

On the other hand, we have the condition of k: $b - k \times K \ge b - k \times N \ge 0$. In conclusion, we have the following inequality:

$$\begin{aligned}
&Kf(B_k) - Kf(T^*) \ge \sum_{q \in Z \wedge q \notin T^*} w_q(Z) - K\sum_{i=0}^{k-1} w_i(B_i)\\
&\quad \ge \sum_{i=0}^{k-1}\sum_{j=1}^{K} w_{q_{(i \times K + j)}}(Z) - Kw_i(B_i) + \sum_{i=k \times K+1}^{b} w_{q_i}(Z)\\
&\quad \ge (b - k \times K) \times E_{\pi(z_0)}(Z)\\
&\quad \ge (b - k \times K)\rho^*\\
&\Rightarrow K\rho(B_k)(a+b-k) \ge Ka\rho^* + (b - k \times K)\rho^*\\
&\Rightarrow \rho(B_k) \ge \frac{Ka + b - k \times K}{K(a+b-k)}\rho^*\\
&\Rightarrow \rho(B_k) \ge \frac{K(a-k)+b}{K(a+b-k)}\rho^*\\
&\Rightarrow \rho(B_k) \ge \frac{1}{K}\rho^* \ge \frac{1}{N}\rho^*.
\end{aligned}$$

□

3.4 Proof of Theorem 4.5

Proof. Assume I_x is the way that has the smallest number of slices in T^*, with a number of slices s. Then, $s \le a/N$.

Let's denote the set $Q = \{q \in Z\} = \{q_1, \ldots, q_s, \ldots, q_a, \ldots, q_{a+b}\}$, it is the set of slices in Z. $(a+b)$ is the size of the Zero subtensor.

Let B_k be a k-Backward Subtensor of T, with $1 \le k \le \frac{(a+b)}{N}$. Then,

$$Nf(Z) = \sum_{i=1}^{s} w_{q_i}(Z) + \sum_{i=s+1}^{a+b} w_{q_i}(Z) \ge f(T^*) + \sum_{i=s+1}^{a+b} w_{q_i}(Z).$$

Because $Nf(Z) = N(f(B_k) + \sum_{i=0}^{k-1} w_i(B_i))$, the above inequality can be rewritten as

$$\Rightarrow N(f(B_k) + \sum_{i=0}^{k-1} w_i(B_i)) \geq f(T^*) + \sum_{i=s+1}^{a+b} w_{q_i}(Z).$$

Subtensor B_i is a backward subtensor of Z by removing i slices in Z, i.e., $B_i \subseteq Z$ and $\forall j, i, E_{q_j}(Z) \geq E_{q_j}(B_i) \geq E_{\pi(z_0+i)}(B_i)$. Hence,

$$\begin{aligned}
Nf(B_k) &\geq f(T^*) + \sum_{i=s+1}^{a+b} w_{q_i}(Z) - N\sum_{i=0}^{k-1} w_i(B_i) \\
&= f(T^*) + \sum_{i=0}^{k-1}\sum_{j=1}^{N} w_{q_{(s+i\times N+j)}}(Z) - Nw_i(B_i) \\
&\quad + \sum_{i=s+k\times N+1}^{a+b} w_{q_i}(Z) \\
&\geq f(T^*) + (a+b-kN-s)w_{\pi(z_0)}(Z).
\end{aligned}$$

Because:

$$\begin{aligned}
a+b-kN-s &\geq a+b-kN-\frac{a}{N} \\
&\geq \frac{(a+b)(N-1)+b}{N} - kN \\
&\geq 0, \forall k \leq \frac{(a+b)(N-1)}{N^2}.
\end{aligned}$$

So we have

$$\begin{aligned}
&Nf(B_k) \geq a\rho^* + (a+b-kN-s)\rho^* \\
&Nf(B_k) \geq (2a+b-kN-s)\rho^* \\
&\Rightarrow \rho(B_k) \geq \frac{(2a+b-kN-s)}{N(a+b-k)}\rho^* \\
&\Rightarrow \rho(B_k) \geq \frac{1}{N}\frac{2a+b-kN-s}{a+b-k}\rho^* \\
&\Rightarrow \rho(B_k) \geq \frac{1}{N}\frac{(a+b-k)+(a-k(N-1)-a/N)}{a+b-k}\rho^* \\
&\Rightarrow \rho(B_k) \geq \frac{1}{N}(1+\frac{(a-kN)(N-1)}{N(a+b-k)})\rho^* \\
&\Rightarrow \rho(B_k) \geq \frac{\rho^*}{N}, \forall k \leq \frac{a}{N}.
\end{aligned}$$

The theorem is proved. □

3.5 Proof of Theorem 4.6

Proof. Let Z denote the Zero subtensor of T on π by Algorithm 1, and the zero index is z_0, such that $N \leq n - z_0$. Then, we have the following:

(1) By Theorem 4.2, there are at least $Min(N(N-1), z_0)$ forward Subtensors $F_1, F_2, \ldots,$ having density higher than $\frac{1}{N}\rho^*$.
(2) By Theorems 4.3-4.4, there are backward Subtensors $B_1, B_2, \ldots,$ having density higher than $\frac{1}{N}\rho^*$. The principle of the number of backward subtensors having density greater than $\frac{1}{N}$ of the highest density is as follows:

$$\begin{cases} \frac{b}{N}, & \text{by Theorem 4.3.} \\ min(\frac{a}{N}, \frac{(a+b)(N-1)}{N^2}), & \text{by Theorem 4.4.} \end{cases} \tag{5}$$

From (5), there is at least $max(\frac{b}{N}, min(\frac{a}{N}, \frac{(a+b)(N-1)}{N^2}))$ backward subtensors having density greater than the lower bound.

If $\frac{a}{N} \leq \frac{(a+b)(N-1)}{N^2}$, then number of backward subtensors having density greater than the lower bound is at least $max(\frac{a}{N}, \frac{b}{N}) \geq \frac{a+b}{2N}$.

Otherwise, we have

$$min(\frac{a}{N}, \frac{(a+b)(N-1)}{N^2}) = \frac{(a+b)(N-1)}{N^2} \geq \frac{a+b}{2N}$$

So we have the number of backward subtensors is at least $\frac{a+b}{2N}$.

If we combine with the number of forward subtensors, then there is at least $min(1+\frac{n}{2N}, 1+N(N-1))$ subtensors in the tensor having density greater than a lower bound. This can be proved as follows.

According to Theorem 4.4, we have number of backward subtensors having density greater than the lower bound, denoted as bw, and $bw \geq \frac{(a+b)}{2N}$. By Theorem 4.2, number of forward subtensors having density greater than the lower bound, we denote it as fw, and $fw \geq \mathrm{Min}(N(N-1), z_0)$.

If $z_0 \geq N(N-1)$, then we have a number of subtensors having density greater than a lower bound is $1 + fw + bw \geq 1 + N(N-1)$, where 1 is counted for the zero subtensor.

Otherwise, if $z_0 \leq N(N-1)$, note that we have $a + b + z_0 = n$, so we get

$$\begin{aligned}
&1 + fw + bw \geq 1 + \frac{(a+b)}{2N} + z_0 \\
\Rightarrow &1 + fw + bw \geq 1 + \frac{(n-z_0)}{2N} + z_0 \\
\Rightarrow &1 + fw + bw \geq 1 + \frac{n}{2N} + \frac{z_0(2N-1)}{2N} \\
\Rightarrow &1 + fw + bw \geq 1 + \frac{n}{2N}.
\end{aligned}$$

Therefore, we have number of subtensors having density greater than the lower bound is $1 + fw + bw \geq min(1+\frac{n}{2N}, 1+N(N-1))$.

If $(a+b) \leq n - N(N-1)$, we have at least $N(N-1)$ forward subtensors having density greater than $\frac{1}{N}$ of the highest density.

Otherwise, if $n >> N$ such that

$$(a+b) \geq n - N(N-1) \geq 2N^3$$

$$\Rightarrow \text{then we get } \frac{(a+b)}{2N} \geq N(N-1)$$

Then we have at least $N(N-1)$ backward subtensors, each having density greater than $\frac{1}{N}$ of the highest density. By adding the zero subtensors, we have at least $(1 + N(N-1))$ subtensors having density greater than $\frac{1}{N}$ of the highest density each. □

REFERENCES

[1] A. Globerson, G. Chechik, F. Pereira, and N. Tishby. 2007. Euclidean Embedding of Co-occurrence Data. *The Journal of Machine Learning Research* 8 (2007), 2265–2295.
[2] Ruining He and Julian McAuley. 2016. Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. In *Proceedings of the 25th International Conference on World Wide Web, WWW*. 507–517.
[3] Jérôme Kunegis. 2013. KONECT: The Koblenz Network Collection. In *Proceedings of the 22nd International Conference on World Wide Web, WWW*. 1343–1350.
[4] Richard Lippmann, Joshua W. Haines, David J. Fried, Jonathan Korba, and Kumar Das. 2000. The 1999 DARPA Off-line Intrusion Detection Evaluation. *Computer Networks* 34, 4 (2000), 579–595.
[5] Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. 2007. Measurement and Analysis of Online Social Networks. In *Proceedings of the 7th ACM Conference on Internet Measurement, SIGCOMM*. 29–42.

[6] Ruoming Pang, Mark Allman, Vern Paxson, and Jason Lee. 2006. The Devil and Packet Trace Anonymization. *ACM SIGCOMM Computer Communication Review* 36, 1 (2006), 29–38.

[7] Kijung Shin, Bryan Hooi, and Christos Faloutsos. 2018. Fast, Accurate, and Flexible Algorithms for Dense Subtensor Mining. *ACM Transactions on Knowledge Discovery from Data* 12, 3 (2018), 28:1–28:30.

[8] Kijung Shin, Bryan Hooi, Jisu Kim, and Christos Faloutsos. 2017. DenseAlert: Incremental Dense-Subtensor Detection in Tensor Streams. In *Proceedings of the 23rd ACM International Conference on Knowledge Discovery and Data Mining, SIGKDD*. 1057–1066.

[9] Shaden Smith, Jee W. Choi, Jiajia Li, Richard Vuduc, Jongsoo Park, Xing Liu, and George Karypis. 2017. FROSTT: The Formidable Repository of Open Sparse Tensors and Tools. http://frostt.io/