# A General Suspiciousness Metric for Dense Blocks in Multimodal Data


Meng Jiang
Tsinghua University
mjiang89@gmail.com

Alex Beutel
Carnegie Mellon University
abeutel@cs.cmu.edu

Peng Cui
Tsinghua University
cuip@tsinghua.edu.cn

Bryan Hooi
Carnegie Mellon University
bhooi@andrew.cmu.edu

Shiqiang Yang
Tsinghua University
yangshq@tsinghua.edu.cn

Christos Faloutsos
Carnegie Mellon University
christos@cs.cmu.edu



***Abstract*—Which seems more suspicious: 5,000 tweets from 200 users on 5 IP addresses, or 10,000 tweets from 500 users on 500 IP addresses but all with the same trending topic and all in 10 minutes? The literature has many methods that try to find dense blocks in matrices, and, recently, tensors, but no method gives a principled way to score the suspiciouness of dense blocks with different numbers of modes and rank them to draw human attention accordingly. Dense blocks are worth inspecting, typically indicating fraud, emerging trends, or some other noteworthy deviation from the usual. Our main contribution is that we show how to unify these methods and how to give a *principled* answer to questions like the above. Specifically, (a) we give a list of axioms that any metric of suspicousness should satisfy; (b) we propose an intuitive, principled metric that satisfies the axioms, and is fast to compute; (c) we propose CROSSSPOT, an algorithm to spot dense regions, and sort them in importance ("suspiciousness") order. Finally, we apply CROSSSPOT to real data, where it improves the F1 score over previous techniques by *68%* and finds retweet-boosting in a real social dataset spanning *0.3 billion* posts.**


## I. INTRODUCTION

Imagine your job at Twitter is to detect when fraudsters are trying to manipulate the most popular tweets for a given trending topic. Given time pressure, which is more worthy of your investigation: 2,000 Twitter users, all retweeting the same 20 tweets, 4 to 6 times each; or 225 Twitter users, retweeting the same 1 tweet, 10 to 15 times each? Now, what if the latter batch of activity happened within 3 hours, while the former spanned 10 hours? What if all 225 users of the latter group used the same 2 IP addresses?

Figure 1 shows an example of these patterns from Tencent Weibo, one of the largest microblogging platforms in China; our method CROSSSPOT detected a block of 225 users, using 2 IP addresses ("blue circle" and "red cross"), retweeting the same tweet 27K times, within 200 minutes. Further, manual inspection shows that several of these users get activated every 5 minutes. This type of lockstep behavior is suspicious (say, due to automated scripts), and it leads to dense blocks, as in Figure 1. These blocks may span several modes (user-id, timestamp, hashtag, etc.). Although our main motivation is fraud detection in a Twitter-like setting, our proposed approach is suitable for numerous other settings, like distributed-denial-of-service (DDoS) attacks, link fraud, click fraud, even health-insurance fraud, as we discuss next.

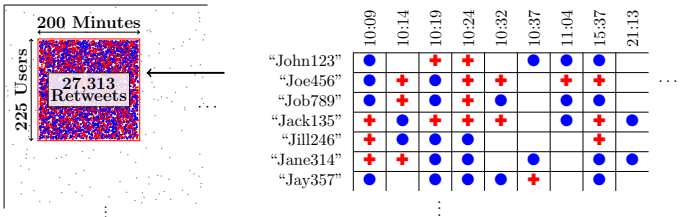



Fig. 1. **Density in multiple modes is suspicious.** Left: A dense block of 225 users on Tencent Weibo (Chinese Twitter) retweeting one tweet 27,313 times from 2 IP addresses over 200 minutes. Right: magnification of a subset of this block. ● and ✚ indicate the two IP addresses used. Notice how synchronized the behavior is, across several modes (IP-address, user-id, timestamp).

Thus, the core question we ask in this paper is: *what is the right way to compare the severity/suspiciousness/surprise of two dense blocks, that span 2 or more modes?* Informally, the problem is:

**Informal Problem 1 (Suspiciousness score)** *Given a $K$-mode dataset (tensor) $\mathcal{X}$, with counts of events (that are non-negative integer values), and two subtensors $\mathcal{Y}_1$ and $\mathcal{Y}_2$, which is more suspicious and worthy of further investigation?*

**Why multimodal data (tensor):** Graphs and social networks have attracted huge interest - and they are perfectly modeled as $K$=2 mode datasets, that is, matrices. With $K$=2 modes we can model Twitter's "who-follows-whom" network [1][2], Facebook's "who-friends-whom" and "who-Likes-what" graphs [3], eBay's "who-buys-from-whom" graph [4], financial activities of "who-trades-what-stocks", and scientific relations of "who-cites-whom." Several high-impact datasets make use of higher mode relations. With $K$=3 modes, we can consider how all of the above graphs change over time or what words are used in product reviews on eBay or Amazon. With $K$=4 modes, we can analyze network traces for intrusion detection and distributed denial of service (DDoS) attacks by looking for patterns in the source IP, destination IP, destination port, and timestamp [5].

**Why are dense regions worth inspecting:** Dense regions are surprising in all of the examples above. Past work has repeatedly found that dense regions in these tensors correspond to suspicious, lockstep behavior: Purchased Page Likes on Facebook result in a few users "Liking" the same "Pages" always at the same time (when the order for the Page Likes is placed) [3]. Zombie followers, botnets who are set up to

build social links, will inflate the number of followers to make their customers seem more popular than they actually are [1][6]. This high-density outcome has a reason: Spammers have constrained resources (users, IP addresses, time, etc.) and they want to add as many edges to the graph/tensor as possible, to maximize their profit while minimizing their costs. Intuitively, the more synchronized the data is, in higher number of modes, the more worthy it is of further inspection.

**Our new perspective:** There are numerous papers on finding dense subgraphs, blocks and communities, including matrix algebra methods (SVD [7], tensor decompositions like PARAFAC and HOSVD [8], and PageRank/TrustRank [4][9]; several more papers apply such methods for anomaly and fraud detection [5]. These methods *do* effectively find suspicious behavior, nearly always related to dense subgraphs. However, none of them answers the problem of interest (Problem 1). The features that set this work apart are the following (also presented in Table I):

- **Block score:** How would you label an individual Like on Facebook or follower on Twitter? These actions are impossible to evaluate in isolation but can be understood in the aggregate. Therefore, we focus on finding and measuring the *suspiciousness* of *blocks* of data. Other methods either return no score (like SVD/eigenspokes, and PARAFAC/Tucker tensor decomposition) or they return a score for each node (like PageRank, TrustRank, and belief propagation), but not for the whole group. These prior methods are harder to interpret and are more easily deceived through adversarial noise.
- **Cross modes:** We look for suspicious density in all $K$ modes, as well as *any subset* of the modes. In contrast, SVD and dense subgraph mining methods work only for $K$=2 modes; (sparse) PARAFAC, HOSVD and related tensor analysis return blocks in *all* modes.

In this paper, we offer the following contributions:

1) **Metric criteria:** We propose a set of basic axioms that a good metric must meet to detect dense subregions in sparse multimodal data (e.g. if two blocks are the same size, the denser one is more surprising). We demonstrate that while simple, meeting all of the criteria is non-trivial.
2) **Novel metric:** We introduce a novel *suspiciousness* metric to evaluate how suspicious a subvector, a submatrix or a subtensor is in multimodal data. Our metric is derived from basic probability and meets the specified criteria.
3) **The CROSSSPOT algorithm:** We design a scalable search algorithm to find suspicious regions of a tensor.
4) **Validation:** Extensive experiments have demonstrated the effectiveness in detecting tweet promotion through retweets. We find that directly optimizing our metric significantly improves the results over just applying computationally-convenient methods like the SVD.

## II. RELATED WORK

In this section, we review related fields including suspicious behavior detection and decomposition methods. We compare our work with baseline methods in Table I, and point out the uniqueness of ours.

**Suspicious behavior detection:** A variety of research has found fraudulent behavior through mining multimodal relational data. These patterns of fraud have been found to show up in eBay reviews [4], opinion spam [12], false accounts [1][2], among many others. Many methods have focused on labeling individual users, such as by using belief propagation (BP) [4] or TrustRank scores [9]. These methods label suspicious nodes/users, but do not return suspicious grouping behaviors themselves. Later works found that adding additional modes of information aided in detecting suspicious behavior. CopyCatch [3] found that suspicious patterns of Page Likes on Facebook correlated in time were good indicators of fraud. Many of the above methods return labels or scores for individual users or IP addresses but not blocks. Even a human evaluation of the results is difficult. Finally, because they are operating on independent formulations, it is impossible to compare their effectiveness and measure progress in the field as a whole. However, none of them gives a "surprise" scoring function for a dense sub-tensor. Rather, in this paper we study and quantify this pattern in a principled manner.

TABLE I. COMPARISON OF STATE-OF-THE-ART METRICS/METHODS.

| | | | Axioms | | | | |
|---|---|---|---|---|---|---|---|
| | | | Density | Size | Concentration | Contrast | Multimodal |
| | Method | Scores Blocks | 1 | 2 | 3 | 4 | 5 |
| Metrics | **SUSPICIOUSNESS** | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
| | Mass | ✓ | ✓ | ✗ | ✗ | ✗ | ✓ |
| | Density | ✓ | ✓ | ✗ | ✓ | ✗ | ✗ |
| | Average Degree [10] | ✓ | ✓ | ✗ | ✗ | ✗ | N/A |
| | Singular Value [11] | ✓ | ✓ | ✓ | ✓ | ✗ | ✗ |
| Methods | **CROSSSPOT** | ✔ | ✔ | ✔ | ✔ | ✔ | ✔ |
| | Subgraph [11] | ✓ | ✓ | ✓ | ✓ | ✗ | N/A |
| | CopyCatch [3] | ✓ | ✓ | ✓ | ✓ | ✗ | N/A |
| | TrustRank [9] | ✗ | N/A | | | | |
| | BP [4] | ✗ | N/A | | | | |

**SVD-based methods:** Decomposition methods have been widely used in subspace clustering [13], community detection [11], and pattern discovery [8]. Implicitly, the SVD focuses on dense regions of a matrix. Chen et al. extracted dense subgraphs using a spectral cluster framework [11]. For multimodal data, tensor decompositions have been applied in many applications [8].High-order singular value represented the importance of the cluster [13]. However, later we show that the SVD has limitations to evaluate cross-mode blocks.

## III. PROPOSED METRIC CRITERIA

We now give a precise definition of the problem. We consider the mass of a subtensor to be the sum of entries in that subtensor, and the density to be the mass divided by the volume of the subtensor. A full list of our notation can be found in Table II.

**Formal Problem 1 (Suspiciousness score)** ***Given*** *a $K$-mode tensor $\mathcal{X}$ with non-negative entries, of size $\mathbf{N} = [N_k]_{k=1}^K$ and with mass $C$ (describing $C$ events by summing entries of the tensor),* ***define*** *a score function $f(\mathbf{n}, c, \mathbf{N}, C)$ for how suspicious a subtensor $\mathcal{Y}$ of size $\mathbf{n} = [n_k]_{k=1}^K$ with mass $c$.*

We consider an alternative parameterization using density. Here $\rho$ is the density of $\mathcal{Y}$ and $p$ is the density of $\mathcal{X}$:

$$\hat{f}(\mathbf{n}, \rho, \mathbf{N}, p) = f\left(\mathbf{n}, \rho \prod_{k=1}^{K} n_k, \mathbf{N}, p \prod_{k=1}^{K} N_k\right)$$

TABLE II. THE NOTATION USED THROUGHOUT THIS PAPER.

| Symbol | Definition |
|---|---|
| $K$ | Number of modes in our dataset |
| $\mathcal{X}$ | $K$-mode tensor dataset |
| $\mathcal{Y}$ | Subtensor within $\mathcal{X}$ |
| $\mathbf{N}$ | $K$-length vector for the size of each mode of $\mathcal{X}$ |
| $C$ | The mass of $\mathcal{X}$ (summing the entries of $\mathcal{X}$) |
| $\mathbf{n}$ | $K$-length vector for the size of each mode of $\mathcal{Y}$ |
| $c$ | The mass of $\mathcal{Y}$ |
| $p$ | The density, $C/\prod_k N_k$ of $\mathcal{X}$ |
| $\rho$ | The density, $c/\prod_k n_k$, of $\mathcal{Y}$ |
| $f$ | Suspiciousness metric, parameterized by the masses |
| $\hat{f}$ | Suspiciousness metric, parameterized by the densities |
| $D_{KL}(\rho\|p)$ | Directed KL-divergence of Poisson($p$) & Poisson($\rho$) $p-\rho+\rho\log\frac{\rho}{p}$ |

In the rare case that the number of modes being considered is unclear, we will refer to the functions by $f_K$ and $\hat{f}_K$.

Note that we restrict $f$ to only focus on blocks for which $\rho > p$, that is the density inside the block is greater than the density in the general tensor. While extremely sparse regions are also unusual, they are not the focus of this work.

### A. Axioms

We now list five basic axioms that any suspiciousness metric $f$ must meet.

**Axiom 1** ***Density*** *If there are two blocks of the same size in the same number of modes, the block of bigger mass is more suspicious than the block of less mass. Formally,*

$$c_1 > c_2 \iff f(\mathbf{n}, c_1, \mathbf{N}, C) > f(\mathbf{n}, c_2, \mathbf{N}, C)$$

**Axiom 2** ***Size*** *If there are two blocks of the same density in the same number of modes, the bigger block is more suspicious than smaller block. Formally,*

$$n_j > n'_j \wedge n_k \geq n'_k \ \forall k \implies \hat{f}(\mathbf{n}, \rho, \mathbf{N}, p) > \hat{f}(\mathbf{n}', \rho, \mathbf{N}, p)$$

**Axiom 3** ***Concentration*** *If there are two blocks of the same mass in the same number of modes, the smaller block is more suspicious than bigger block. Formally,*

$$n_j < n'_j \wedge n_k \leq n'_k \ \forall k \implies f(\mathbf{n}, c, \mathbf{N}, C) > f(\mathbf{n}', c, \mathbf{N}, C)$$

**Axiom 4** ***Contrast*** *If two identical blocks lie in two tensors each of the same size but one is sparser, then the block in the sparser tensor is more suspicious. Formally,*

$$p_1 < p_2 \iff \hat{f}(\mathbf{n}, \rho, \mathbf{N}, p_1) > \hat{f}(\mathbf{n}, \rho, \mathbf{N}, p_2)$$

**Axiom 5** ***Multimodal*** *A block which contains all possible values within a mode is just as suspicious as if that mode was collapsed* [1] *into the remaining modes. Formally,*

$$f_{K-1}\big([n_k]_{k=1}^{K-1}, c, [N_k]_{k=1}^{K-1}, C\big) = f_K\left(([n_k]_{k=1}^{K-1}, N_K), c, [N_k]_{k=1}^{K}, C\right)$$

**Lemma 1** ***Cross-mode comparisons*** *Learning of a new mode about our data can only make blocks in that data more suspicious. Formally,*

$$f_{K-1}\big([n_k]_{k=1}^{K-1}, c, [N_k]_{k=1}^{K-1}, C\big) \leq f_K\big([n_k]_{k=1}^{K}, c, [N_k]_{k=1}^{K}, C\big)$$

*Proof:*

$$\begin{aligned} f_{K-1}\left([n_k]_{k=1}^{K-1}, c, [N_k]_{k=1}^{K-1}, C\right) &= f_K\left(([n_k]_{k=1}^{k-1}, N_K), c, [N_k]_{k=1}^{K}, C\right) \\ &\leq f_K\left(([n_k]_{k=1}^{k-1}, n_K), c, [N_k]_{k=1}^{K}, C\right) \end{aligned}$$

Above we find that the first equality is given by Axiom 5 and the second by Axiom 3. ■

[1] *Collapsing* a tensor $\mathcal{X}$ on mode $K$ sums the values of $\mathcal{X}$ across all indices in mode $K$ [14], e.g. collapsing a tensor to a matrix: $\mathbf{X}_{i,j} = \sum_k \mathcal{X}_{i,j,k}$.

### B. Shortcomings of Competitors

While these axioms are simple and intuitive, they are non-trivial to meet. As shown in Table I, simple metrics fail a number of the axioms.

**Mass:** One possible metric is the mass $f(\mathbf{n}, c, \mathbf{N}, C) = c$. This does not change if the same mass is concentrated in a smaller region, and hence fails Axiom 3 (Concentration); it does not consider the background density $p$, and so fails Axiom 4 (Contrast) as well.

**Density:** Another possible metric is the density of the block $\hat{f}(\mathbf{n}, \rho, \mathbf{N}, p) = \rho$. However, this does not consider the size of the dense block, and hence fails Axiom 2 (Size). It also does not consider the background density, and fails Axiom 4 (Contrast). Since density in general decreases with more modes, Axiom 5 (Multimodal) is also broken.

**Average degree:** Much of the research on finding dense subgraphs focuses on the average degree of the subgraph [15], [16], $f(\mathbf{n}, c, \mathbf{N}, C) = c/n_1$. This metric breaks both Axioms 2 and 3 by not considering $n_2$ and breaks Axiom 4 by not considering $C$ and $\mathbf{N}$. Additionally it is unclear how we would define the average degree for $K > 2$, making it unsuitable for multi-modal data.

**SVD:** The SVD of a matrix $\mathbf{A}$ is a factorization of the form $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$. The singular values of $\mathbf{A}$ correspond to $\mathbf{\Sigma}_{r,r}$, and $\mathbf{U}, \mathbf{V}$ are the singular vectors. The top singular values and vectors indicate big, dense blocks/clusters in the multi-mode data and have been used to find suspicious behavior [13]. As shown in [2], an independent block of size $n_1 \times n_2$ with mass $c$ has a singular value $\sigma$ corresponding to that block of $\sigma = \frac{c}{\sqrt{n_1 n_2}} = \sqrt{\rho c}$. Given the SVD prioritizes the parts of the data with higher singular values, we can view this as a competing metric of suspiciousness. While this metric now meets Axioms 1 through 3, it has a challenge generalizing. First, it is clear that this metric ignores the density of the background data. As a result, Axiom 4 is broken. Second, HOSVD does not have the same provable guarantees as SVD and thus does not necessarily find the largest, densest blocks. Even if we consider density in higher modes, what we find is that with each additional mode added, the volume of a block becomes greater and thus the density lower. This breaks Axiom 5 and would make an algorithm collapse all data down to one mode rather than consider the correlation across all $K$ modes.

From this, we see that methods building on average degree and SVD meet the requirements for many cases, but break down on certain corner cases, limiting their path toward a general approach to finding surprising behavior.

## IV. PROPOSED SUSPICIOUSNESS METRIC

Our metric is based on a model of the data in which the $C$ events are randomly distributed across the tensor data $\mathcal{X}$. For binary data this corresponds to a multi-mode Erdös-Rényi model, where the value in each cell follows a binomial distribution. Because each cell in the tensor can contain more than one occurrence, we instead use a Poisson distribution, resulting in the Erdös-Rényi-Poisson model:

**Definition 1** ***Erdös-Rényi-Poisson (ERP) model*** *A tensor $\mathcal{X}$ generated by the ERP model, has each value in the tensor sampled from a Poisson distribution parameterized by $p$.*

In general, we set $p$ to be the density of the overall tensor. Using this model we define our metric:

**Definition 2** ***The suspiciousness metric*** *The suspiciousness score of a multimodal block is the negative log likelihood of block's mass under an Erdös-Rényi-Poisson model. Mathematically, given an $n_1 \times \cdots \times n_K$ block of mass $c$ in $N_1 \times \cdots \times N_K$ data of total mass $C$, the suspiciousness score is*

$$\boxed{f(\mathbf{n}, c, \mathbf{N}, C) = -\log\left[Pr(Y_n = c)\right]} \tag{1}$$

*where $Y_n$ is the sum of entries in the block.*

### A. Dense Subvector and Submatrix

Consider an $N$-length vector $\mathbf{X}$, which we believe to be generated by the ERP model defined above. We can think of this vector as the number of tweets per IP address. If there are $C$ tweets total, then the density is $p=\frac{C}{N}$ and each $X_i$ has a Poisson distribution. We are searching for an $n$-length subvector $X_{i_1}, \ldots, X_{i_n}$ that is unlikely and hence has a high suspiciousness score.

**Lemma 2** *The suspiciousness of an $n$-length subvector $[X_{i_1}, \ldots, X_{i_n}]$ in the $N$-length vector data $[X_1, \ldots, X_N]$ is*

$$f(n, c, N, C) = c\Big(\log\frac{c}{C} - 1\Big) + C\frac{n}{N} - c\log\frac{n}{N}$$
$$\hat{f}(n, \rho, N, p) = n\left(p - \rho + \rho\log\frac{\rho}{p}\right) = nD_{KL}(\rho||p)$$

Here $c = \sum_{j=1}^{n} X_{i_j}$ and $D_{KL}(\rho||p)$ is the Kullback-Leibler (KL) divergence of $\text{Poisson}(p)$ from $\text{Poisson}(\rho)$.

*Proof:* We denote the sum of $n$ variables by $Y_n = \sum_{j=1}^{n} X_{i_j}$. From the Poisson property, we know $Y_n \sim \text{Poisson}(pn)$. Then we can have the probability that $Y_n$ equals a given number of retweets $c$. With the Stirling's formula, we obtain the suspiciousness score:

$$\begin{aligned} f(n, c, N, C) &= -\log\left[Pr(Y_n = c)\right] = -\log\left[\tfrac{C^c}{c!}\big(\tfrac{n}{N}\big)^c e^{-\frac{Cn}{N}}\right] \\ &\approx c\big(\log\tfrac{c}{C} - 1\big) + C\tfrac{n}{N} - c\log\tfrac{n}{N}. \end{aligned}$$

■

We now extend suspiciousness to a 2-mode matrix.

**Lemma 3** *The suspiciousness of an $n_1 \times n_2$ block of mass $c$ in $N_1 \times N_2$ data of total mass $C$ is:*

$$f([n_1, n_2], c, [N_1, N_2], C) = c\big(\log\tfrac{c}{C} - 1\big) + C\tfrac{n_1 n_2}{N_1 N_2} - c\log\tfrac{n_1 n_2}{N_1 N_2}$$
$$\hat{f}([n_1, n_2], \rho, [N_1, N_2], p) = n_1 n_2 D_{KL}(\rho||p)$$

### B. Dense Subtensor: K-Mode Suspiciousness

We now extend suspiciousness to a $K$-mode tensors.

**Lemma 4** *Given an $n_1 \times \cdots \times n_K$ block of mass $c$ in $N_1 \times \cdots \times N_K$ data of total mass $C$, the suspiciousness function is*

$$\boxed{f(\mathbf{n}, c, \mathbf{N}, C) = c(\log\frac{c}{C} - 1) + C\prod_{i=1}^{K}\frac{n_i}{N_i} - c\sum_{i=1}^{K}\log\frac{n_i}{N_i}} \tag{2}$$

*Using $\rho$ as the block's density and $p$ is the data's density, we have the simpler formulation*

$$\boxed{\hat{f}(\mathbf{n}, \rho, \mathbf{N}, p) = \left(\prod_{i=1}^{K} n_i\right) D_{KL}(\rho||p)} \tag{3}$$

From the nonnegativity of KL divergence, we have $f = \hat{f} \geq 0$.

### C. Proofs: Satisfying the Axioms. Omitted for brevity.

## V. Suspicious Block Detection

Having defined a metric for measuring the suspiciousness of a block, in this section we formally define the problem of detecting suspicious blocks across modes, and give a scalable algorithm based on our proposed metric to identify the blocks.

**Problem 1 (Suspicious block detection)** ***Given*** *dataset $\mathcal{X}$ which is a $N_1 \times \cdots \times N_K$ tensor of mass $C$,* ***find*** *a list of blocks in $\mathcal{X}$, in any subset of modes, with high suspiciousness, in descending order, based on Eq.* (2) *and* (4)*.*

As before, we have a $K$-mode tensor $\mathcal{X}$ and a $k$-mode subtensor $\mathcal{Y}$ to represent the suspicious block. Mode $j$ of the tensor has $N_j$ possible values: $\mathcal{P}_j = \{p_1^{(j)}, \ldots, p_{N_j}^{(j)}\}$. Subtensor $\mathcal{Y}$ covers a subset of values in each mode: $\tilde{\mathcal{P}}_j \subseteq \mathcal{P}_j, \forall j$. Define $\tilde{\mathcal{P}} = \{\tilde{\mathcal{P}}_j\}_{j=1}^{K}$. Let $c(\tilde{\mathcal{P}})$ be the number of events in the subtensor defined by $\tilde{\mathcal{P}}$.

The dimensions of our block $\mathbf{n}$ are $n_j = |\tilde{\mathcal{P}}_j|$. If a mode $j$ is not included, we consider $\tilde{\mathcal{P}}_j = \mathcal{P}_j$, based on Axiom 5 and the properties of collapse operation. For the sake of notational simplicity we define one last alternative parameterization for our suspiciousness function

$$\tilde{f}(\tilde{\mathcal{P}}, \mathcal{D}) = f([|\tilde{\mathcal{P}}_j|]_{j=1}^{K}, c(\tilde{\mathcal{P}}), [|\mathcal{P}_j|]_{j=1}^{K}, |\mathcal{X}|) \tag{4}$$

### A. Proposed Algorithm CrossSpot

We define here a local search algorithm to search for suspicious blocks in the dataset. We start with a seed suspicious block, then perform an iterative alternating optimization, where we find the optimal set of values in mode $j$ while holding constant the included values in all other modes. We run this sequence of updates until convergence. The complete algorithm is shown in Algorithm 1.

**Algorithm 1** Local Search

**Require:** Data $\mathcal{X}$, seed region $\mathcal{Y}$ with $\tilde{\mathcal{P}} = \{\tilde{\mathcal{P}}_j\}_{j=1}^{K}$
1: **while** not converged **do**
2: **for** $j = 1 \ldots K$ **do**
3: $\tilde{\mathcal{P}}_j \leftarrow$ AdjustMode$(j)$
4: **end for**
5: **end while**
6: **return** $\tilde{\mathcal{P}}$

**Adjusting a Mode:** During each iteration of AdjustMode, we optimally choose a subset of values from $\mathcal{P}_j$ holding constant the values in other modes, i.e. fixing $\tilde{\mathcal{P}}_{j'}$ for $j' \neq j$. Denote $\Delta c_{p_i^{(j)}}$ as the number of events in the intersection of row $i$ (in mode $j$) and the currently fixed values in the other

**Algorithm 2** ADJUSTMODE($j$)

$\tilde{\mathcal{P}}'_j \leftarrow \{\}$;
$\mathbf{P}_j \leftarrow \{p_i^{(j)}\}_{i=1}^{N_j}$ sorted in descending order by $\Delta c_{p_i^{(j)}}$
**for** $p_i^{(j)} \in \mathbf{P}_j$ **do**
  $\tilde{\mathcal{P}}'_j \leftarrow \tilde{\mathcal{P}}'_j \cup p_i^{(j)}$
  $\tilde{\mathcal{P}}' \leftarrow \{\tilde{\mathcal{P}}_{j'}\}_{j' \neq j} \cup \tilde{\mathcal{P}}'_j$
  **if** $\tilde{f}(\tilde{\mathcal{P}}, \mathcal{D}) \leq \tilde{f}(\tilde{\mathcal{P}}', \mathcal{D})$ **then**
    $\tilde{\mathcal{P}}_j \leftarrow \tilde{\mathcal{P}}'_j$
  **end if**
**end for**
**return** $\tilde{\mathcal{P}}_j$

modes, i.e. $\tilde{\mathcal{P}}_{j'}$ for $j' \neq j$. We refer to $\Delta c_{p_i^{(j)}}$ as the "benefit" of $p_i^{(j)}$. In Algorithm 2 we use these benefit scores to order the values in $\mathcal{P}_j$, from greatest to least benefit. We will refer to this ordered list as $\mathbf{P}_j$.

**Seeds:** In Algorithm 1, we start from a seed subtensor $\mathcal{Y}$. In the simplest case, we start from a randomly chosen seed, containing an individual cell of the tensor or a larger randomly chosen block. As we will show in Section VI, even using randomly chosen seeds does well.

**Complexity:** The time complexity of Algorithm 1 is $\mathcal{O}(T \times K \times (E + N \log N))$, where $T$ is the number of iterations, $K$ is the number of modes, $E$ is the number of non-zero entries in the data, and $N = \max_j N_j$ is the maximum size of any mode. Because $T$ and $K$ are often set to constant values, the complexity is quasi-linear in $N$ and linear in the number of non-zero entries. Thus, Algorithm 1 is scalable.

## VI. EXPERIMENTS

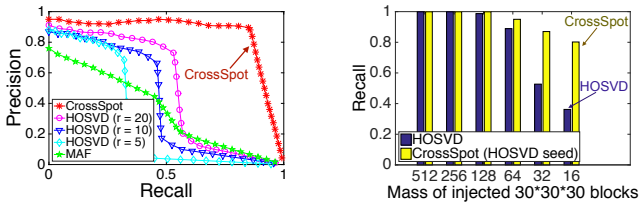
### A. Datasets and Experimental Setup

We used extensive datasets including synthetically generated datasets and one large, new social networking dataset. The synthetic data is generated as a $K$-mode tensor of size $N_1 \times \cdots \times N_K$ with mass $C$. Within the tensor we inject $b$ dense blocks. Each block is assigned a size $n_1 \times \cdots \times n_K$ and mass $c$. When an injected block falls in only a subset of modes $\mathcal{I}$, we set $n_i = N_i$. We use retweeting data from Tencent Weibo. These retweets consist of *user id*, *tweet id*, *IP address*, *timestamp* and *retweeting comment*. On Weibo, *retweet boosting* is common, where retweets can be purchased to make a particular tweet seem more popular than it actually is. This results in a distorted user experience. The dataset has 29.5M users, 19.8M tweets, 27.8M IP addresses and 221.7M retweets, spanning 56,943 minutes.

We compare CROSSSPOT with the following baselines: SVD [7] and HOSVD (Higher-Order SVD) [8], MAF (MultiAspectForensics) [5], and AVGDEG (Average Degree) [10].

### B. Synthetic Experiments

We first evaluate CROSSSPOT on synthetic datasets. Overall, CROSSSPOT is effective: it detects dense subgraphs in 2-mode data, dense $k$-mode blocks in $k$-mode tensor data, and even dense $k'$-mode blocks in $k$-mode tensor data ($k' < k$)




(a) Performance (b) Recall with HOSVD seed

Fig. 2. Finding dense blocks: CROSSSPOT outperforms baselines in finding 3-mode blocks, and directly method improves the recall on top of HOSVD.

with very high precision and recall. It is also efficient: it has faster execution time than complex traditional methods.

**Finding dense high-order blocks in multimodal data:** We generate random tensor data with parameters as (1) the number of modes $k$=3, (2) the size of data $N_1$=$N_2$=$N_3$=1,000 and (3) the mass of data $C$=10,000. We inject $b$=6 blocks of $k'$=3 modes into the random data. Each block has size 30×30×30 and mass $c \in \{16, 32, 64, 128, 256, 512\}$. The task is to classify the entries into suspicious and normal classes. Figure 2(a) reports the performances of CROSSSPOT and baselines. We observe that in order to find all the 6 injected blocks, our proposed CROSSSPOT has better performance than baselines. The best F1 score CROSSSPOT gives is 0.891, which is 46.0% higher than the F1 score given by the best of HOSVD (0.610). If we use the results of HOSVD as seeds to CROSSSPOT, the best F1 score of CROSSSPOT reaches 0.979. Figure 2(b) gives the recall value of every injected block. We observe that CROSSSPOT improves the recall over HOSVD.

**Finding dense low-order blocks in multimodal data:** We generate random tensor data with parameters as (1) the number of modes $k$=3, (2) the size of data $N_1$=1,000, $N_2$=1,000 and $N_3$=1,000 and (3) the mass of data $C$=10,000. We inject $b$=4 blocks into the random data:

- Block #1: The number of modes is $k'_1$=3 and $\mathcal{I}_1$={1,2,3}. The size is 30×30×30 and the block's mass is $c_1$=512.
- Block #2: $k'_2$=2, $\mathcal{I}_2$={1,2}; 30×30×1,000 of $c_2$=512.
- Block #3: $k'_3$=2, $\mathcal{I}_3$={1,3}; 30×1,000×30 of $c_3$=512.
- Block #4: $k'_4$=2, $\mathcal{I}_4$={2,3}; 1,000×30×30 of $c_4$=512.

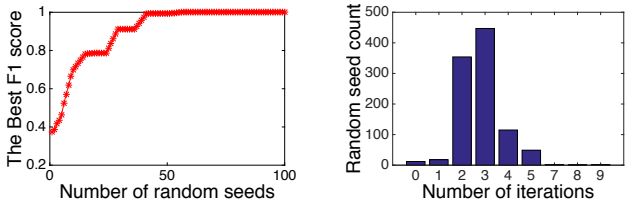
Note, blocks 2-4 are dense in only 2 modes and random in the third mode. From Table III we show the overall evaluations and observe that CROSSSPOT has 100% recall in catching the 3-mode block #1, while the baselines have 85-95% recall. More impressively, CROSSSPOT successfully catches the 2-mode blocks, where HOSVD has difficulty and low recall. The F1 score of overall evaluation is as large as 0.972 with 68.8% improvement.

**Testing robustness of the random seed number:** Figure 3(a) shows the best F1 score for different numbers of random seeds. We find that when we use 41 random seeds, the best F1 score is close to the results when we use as many as 1,000 random seeds. Thus, once we exceed a moderate number of random seeds, the performance is fairly robust.

**Efficiency analysis:** CROSSSPOT can be parallelized into multiple machines to search dense blocks with different sets of random seeds. Figure 3(b) reports the counts of iterations in the procedure of 1,000 random seeds. Each iteration takes only 5.6 seconds. From Table III and Figure 3(a), we know

TABLE III. OUR CROSSSPOT CATCHES MORE LOWER-MODE BLOCKS: CROSSSPOT HAS HIGH ACCURACY IN FINDING THE INJECTED 4 BLOCKS.

| | Recall | | | | Overall Evaluation | | |
|---|---|---|---|---|---|---|---|
| | Block #1 | Block #2 | Block #3 | Block #4 | Precision | Recall | F1 score |
| HOSVD ($r$=20) | 93.7% | 29.5% | 23.7% | 21.3% | **0.983** | 0.407 | 0.576 |
| HOSVD ($r$=10) | 91.3% | 24.4% | 18.5% | 19.2% | 0.972 | 0.317 | 0.478 |
| HOSVD ($r$=5) | 85.7% | 10.0% | 9.5% | 11.4% | 0.952 | 0.195 | 0.324 |
| CROSSSPOT | **100%** | **99.9%** | **94.9%** | **95.4%** | 0.978 | **0.967** | **0.972** |




(a) Robustness (b) Convergence

Fig. 3. CROSSSPOT is robust to the number of random seeds. In detecting the 4 low-order blocks, when we use 41 seeds, the best F1 score has reported the final result of as many as 1,000 seeds. CROSSSPOT converges very fast: the average number of iterations is 2.87.

TABLE IV. BIG DENSE BLOCKS WITH TOP METRIC VALUES DISCOVERED IN THE RETWEETING DATASET.

| | # | User×tweet×IP×minute | Mass $c$ | Suspiciousness |
|---|---|---|---|---|
| **CROSSSPOT** | 1 | 14×1×2×1,114 | 41,396 | 1,239,865 |
| | 2 | 225×1×2×200 | 27,313 | 777,781 |
| | 3 | 8×2×4×1,872 | 17,701 | 491,323 |
| HOSVD | 1 | 24×6×11×439 | 3,582 | 131,113 |
| | 2 | 18×4×5×223 | 1,942 | 74,087 |
| | 3 | 14×2×1×265 | 9,061 | 381,211 |

that CROSSSPOT takes only 230 seconds to have the best F1 score 0.972, while HOSVD needs more time (280 seconds if $r$=5) to have a much smaller F1 score 0.324.

### C. Retweeting Boosting

Table IV shows big, dense block patterns of retweeting dataset. CROSSSPOT reports blocks of high mass and high density. For example, we spot that 14 users retweet the same content for 41,396 times on 2 IP addresses in 19 hours. Their coordinated, suspicious behaviors result in a few tweets that seem extremely popular. We observe that CROSSSPOT catches bigger and denser blocks than HOSVD does: HOSVD evaluates the number of retweets per user, item, IP, or minute, but does not consider the block's density, mass nor the background.

## VII. CONCLUSION

We provide a metric of suspiciousness for a dense block, in arbitrary number of modes. Our contributions are:

- **Metric criteria:** We propose a set of axioms that any metric of suspicious dense behavior should meet.
- **Novel metric:** We propose a suspiciousness metric, that is based on a principled, probabilistic model; and we prove that it obeys our axioms.
- **CROSSSPOT algorithm:** We propose a scalable algorithm to find dense, suspicious blocks in multi-modal data.
- **Empirical results:** We demonstrate the effectiveness of our approach on synthetic as well as on real world data.

## ACKNOWLEDGEMENT

This work was supported by National Program on Key Basic Research Project, No. 2015CB352300; National Science Foundation of China, No. 61370022 and No. 61210008; International Science and Technology Cooperation Program of China, No. 2013DFG12870. Thanks for the support of NExT Research Center under the research grant, WBS:R-252-300-001-490 and the research fund of Tsinghua-Tencent Joint Laboratory. Thanks for the support of National Science Foundation, No. CNS-1314632, Nos. IIS-1217559 and Grant No. DGE-1252522 as well as a Facebook Fellowship; Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053. Prepared by LLNL under Contract DE-AC52-07NA27344.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation, DARPA, or other funding parties. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

## REFERENCES

[1] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang, "Catchsync: catching synchronized behavior in large directed graphs," in *SIGKDD*, 2014, pp. 941–950.

[2] N. Shah, A. Beutel, B. Gallagher, and C. Faloutsos, "Spotting suspicious link behavior with fbox: An adversarial perspective," in *ICDM*, 2014.

[3] A. Beutel, W. Xu, V. Guruswami, C. Palow, and C. Faloutsos, "Copycatch: stopping group attacks by spotting lockstep behavior in social networks," in *WWW*, 2013, pp. 119–130.

[4] S. Pandit, D. H. Chau, S. Wang, and C. Faloutsos, "Netprobe: a fast and scalable system for fraud detection in online auction networks," in *WWW*, 2007, pp. 201–210.

[5] K. Maruhashi, F. Guo, and C. Faloutsos, "Multiaspectforensics: Pattern mining on large-scale heterogeneous networks with tensor analysis," in *ASONAM*, 2011, pp. 203–210.

[6] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang, "Inferring strange behavior from connectivity pattern in social networks," in *PAKDD*, 2014, pp. 126–138.

[7] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[8] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[9] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen, "Combating web spam with trustrank," in *VLDB Endowment*, 2004, pp. 576–587.

[10] M. Charikar, "Greedy approximation algorithms for finding dense components in a graph," in *Approximation Algorithms for Combinatorial Optimization*, 2000, pp. 84–95.

[11] J. Chen and Y. Saad, "Dense subgraph extraction with application to community detection," *TKDE*, vol. 24, no. 7, pp. 1216–1230, 2012.

[12] X. Hu, J. Tang, Y. Zhang, and H. Liu, "Social spammer detection in microblogging," in *IJCAI*, 2013, pp. 2633–2639.

[13] H. Huang, C. Ding, D. Luo, and T. Li, "Simultaneous tensor subspace selection and clustering: the equivalence of high order svd and k-means clustering," in *SIGKDD*, 2008, pp. 327–335.

[14] J. Inah, E. E. Papalexakis, U. Kang, and C. Faloutsos, "Haten2: Billion-scale tensor decompositions," in *ICDE*, 2015.

[15] Y. Asahiro, K. Iwama, H. Tamaki, and T. Tokuyama, "Greedily finding a dense subgraph," *Journal of Algorithms*, vol. 34, no. 2, 2000.

[16] R. Andersen, "A local algorithm for finding dense subgraphs," *Transaction on Algorithms*, vol. 6, no. 4, p. 60, 2010.