

Spotting Suspicious Behaviors in Multimodal Data: A General Metric and Algorithms

Meng Jiang, Alex Beutel, Peng Cui, Bryan Hooi, Shiqiang Yang *Senior Member, IEEE*, Christos Faloutsos

Abstract—Many commercial products and academic research activities are embracing behavior analysis as a technique for improving detection of attacks of many sorts - from retweet boosting, hashtag hijacking to link advertising. Traditional approaches focus on detecting dense blocks in the adjacency matrix of graph data, and recently, the tensors of multimodal data. No method gives a principled way to score the suspiciousness of dense blocks with different numbers of modes and rank them to draw human attention accordingly. In this paper, we first given a list of axioms that any metric of suspiciousness should satisfy; we propose an intuitive, principled metric that satisfies the axioms, and is fast to compute; moreover, we propose CROSSSPOT, an algorithm to spot dense blocks that are worth inspecting, typically indicating fraud or some other noteworthy deviation from the usual, and sort them in the order of importance (“suspiciousness”). Finally, we apply CROSSSPOT to the real data, where it improves the F1 score over previous techniques by 68% and finds suspicious behavioral patterns in social datasets spanning 0.3 billion posts.

Index Terms—Suspicious Behavior, Fraud Detection, Multimodal Data, Dense Blocks

1 INTRODUCTION

Web applications such as Twitter, Amazon and PayPal have become important means of satisfying social, shopping and money-transfer needs, suspicious users such as spammers, fraudsters, and other types of attackers are increasingly attempting to engage in dishonest activity [1]. A common task is to detect when fraudsters are trying to manipulate the most popular tweets for a given trending topic (hashtag). Given time pressure, which is more worthy of 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? 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 IPs?

Figure 1 shows an example of these patterns from Weibo (the Chinese Twitter). Our method CROSSSPOT detected a block of 225 users, using 2 IP addresses (● and +), retweeting the same tweet 27,313 times within more than 200 minutes. Further, manual inspection shows that several of these users get activated every 5 minutes. This type of lockstep behavior is suspicious due to automated scripts, and it leads to the dense block. The block may span several modes such as user, IP, hashtag and timestamp. Although our main motivation is fraud detection in a Twitter-like setting, our proposed approach is suitable for numerous other settings, like

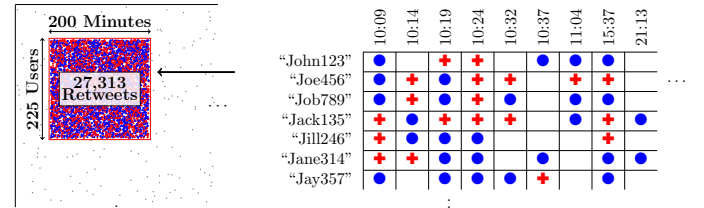


Fig. 1. Dense blocks in multiple modes are suspicious. Left: a dense block of 225 users on Weibo retweeting one message 27,313 times from 2 IPs over 200 minutes. Right: magnification of a subset of this block. ● and + indicate the two IPs. Note how lockstep the behavior is across several modes (user, IP, timestamp).

distributed-denial-of-service (DDoS) attacks, link fraud, click fraud and even health-insurance fraud.

In this paper, we raise the question that what is the right way to compare the severity, **suspiciousness**, or surprise of two **dense** blocks, that span 2 or more modes? Mathematically speaking, given a K -mode 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?

We would like to explain more about the two keywords in the above question: first, why do we explore the tensor, i.e., the multimodal dataset? Graphs and social networks have attracted huge interest, and they are perfectly modeled as $K=2$ mode data, that is, matrices. With the matrices, we can model Twitter’s “who-follows-whom” networks [2][3], Facebook’s “who-Likes-what” graphs [4], eBay’s “who-buys-from-whom” graph [5], and financial activities of “who-trades-what-stocks”. Several high-impact datasets make use of higher mode relations. With $K=3$ modes, we can consider how the above graphs evolve over time or what words are used in the product reviews. With $K=4$ modes, we can analyze the network traffic for intrusion detection and

- M. Jiang, P. Cui, and S. Yang are with the Department of Computer Science and Technology in Tsinghua University, Beijing 100084, China. E-mail: mjiang89@gmail.com, cuip@tsinghua.edu.cn, yangshq@tsinghua.edu.cn
- A. Beutel, B. Hooi, and C. Faloutsos are with the Computer Science Department in Carnegie Mellon University, Pittsburgh, PA 15217, US. E-mail: abeutel@cs.cmu.edu, bhooi@andrew.cmu.edu, christos@cs.cmu.edu

DDoS attacks by looking for patterns in the source IP, destination IP, destination port, and timestamp [6][7].

Second, why are the dense blocks worth inspecting? Dense regions are surprising in all of the examples above. Previous approaches have repeatedly found that dense blocks in the matrices correspond to suspicious, lockstep behavior: purchased Likes on Facebook result in a few users “Liking” the same “Pages” [4]; spammers paid to write deceptively five-star or one-star reviews for restaurants or hotels will reuse the same accounts and often even the same text [8][9]; zombie followers that are set up to build social links, will inflate the number of followers to make their customers seem more popular than they actually are [2][10]. This unexpected-density outcome shares a reason: fraudsters have constrained resources such as user, IP address and time, and they want to add as many edges to the graph/tensor as possible to maximize their profit while minimizing their costs. Intuitively, the denser the data is in a higher number of modes, the more worthy it is of further inspection.

A great line of work finds dense subgraphs [11][12][13] and local communities [14][15][16], including matrix algebra methods like singular value decompositions (SVD) [17][18], tensor decompositions like multi-way decomposition methods (CANDECOMP/PARAFAC) and high-order SVD (HOSVD) [19][20], and PageRank/TrustRank [21][22]; several more papers apply such methods for anomaly and fraud detection [6][23][24]. These methods find suspicious behaviors nearly always related to dense subgraphs. However, none of them answers the problem of our interest. The features that set our **target problem** apart are the following (also presented in Table 1).

Scoring blocks. How would you label an individual Like on Facebook or a single 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. Other methods either return no score (e.g., SVD and tensor decompositions) or return a score for each node (e.g., PageRank and belief propagation), but not for the whole group. These prior methods are harder to interpret and are more easily deceived through adversarial noise.

Working across modes. We look for suspicious density in all K modes, as well as any subset of the modes. In contrast, SVD and subgraph mining work only for 2 modes; tensor methods (PARAFAC and HOSVD) return blocks in all modes. How to evaluate the suspiciousness of a subset of the modes has not yet been studied.

To address the problem, we propose a series of novel “products” including the novel axioms to evaluate a good metric of unexpected density, the novel metric based on the principled probability theory that satisfy all the axioms, and a scalable algorithm to spot the suspicious behaviors in multimodal data.

The metric criteria (axioms). We propose a set of five novel axioms that a good metric must meet to detect dense blocks in sparse multimodal data. For example, if two blocks have the same size, the denser block is more

TABLE 1

Comparison of state-of-the-art metric/methods. While the methods have been successful in particular applications, they do not meet the general goals set out here.

	Method	Scoring Blocks	Axioms				
			Density	Size	Concentration	Contrast	Multimodal
			1	2	3	4	5
Metrics	SUSPICIOUSNESS	✓	✓	✓	✓	✓	✓
	Mass	✓	✓	✗	✗	✗	✓
	Density	✓	✓	✗	✓	✗	✗
	Average Degree [27]	✓	✓	✗	✗	✗	N/A
	Singular Value [12]	✓	✓	✓	✓	✗	✗
Methods	CROSSSPOT	✓	✓	✓	✓	✓	✓
	Subgraph [11][12]	✓	✓	✓	✓	✗	N/A
	CopyCatch [4]	✓	✓	✓	✓	✗	N/A
	EigenSpokes [23]	✗			N/A		
	CatchSync [2]	✗			N/A		
	fBox [3]	✗			N/A		
	TrustRank [24][25]	✗			N/A		
	SybilRank [28]	✗			N/A		
	CollusionRank [26]	✗			N/A		
	BP [5][9]	✗			N/A		

surprising; in practice, if we spot a group of 100 users from the same IP in 20 minutes, posting 10,000 messages is more suspicious than posting 100. The last axiom, which supports the cross-mode modeling, demonstrates that taking one more mode is always more suspicious. The axioms can be clearly observed from the blocks in real datasets. Table 1 shows that previous metrics satisfy at most three axioms. We demonstrate that while simple, meeting all of the criteria is non-trivial.

The suspiciousness metric. We introduce a novel metric to evaluate how suspicious a subvector, a submatrix or a subtensor is in 1-mode, 2-mode, and multimodal data. Our metric is derived from the principled probability theory and meets all the proposed axioms.

The CrossSpot algorithm. We develop a scalable algorithm based on the greedy search strategy to find suspicious blocks in multimodal datasets. The time complexity is quasi-linear in the number of nodes and linear in the number of non-zero entries. Extensive experiments have demonstrated the effectiveness in spotting (1) trending-topic manipulation, (2) tweet promotion via retweet boosting, (3) advertising with URL hijacking, and (4) attacks in network traffic. We generate synthetic datasets to evaluate the accuracy of our proposed algorithm. We find that optimizing the suspiciousness metric significantly improves the results over the SVD methods.

It is worthwhile to summarize our contributions: (1) we propose the novel problem of evaluating and detecting dense blocks in multimodal behavioral data, (2) we propose a set of axioms that a good metric of suspiciousness should obey, and a principled metric based on the probability theory, (3) we develop a scalable algorithm to spot the suspicious behaviors.

The remainder of this paper is organized as follows. In Section 2 we survey the related work. In Section 3

we propose the metric criteria and Section 4 presents the suspiciousness metric that meets all the axioms. Section 5 introduces a scalable algorithm to detect multimodal blocks. Section 6 reports empirical results on synthetic and real-world datasets. Section 7 concludes the paper.

2 RELATED WORK

In this section, we review related fields including suspicious behavior detection, decomposition methods and dense subgraph mining. We compare our work with prior methods in Table 1, and point out our uniqueness.

Suspicious behavior detection: A variety of research has found fraudulent behavior through mining multimodal relational data [1]. These patterns of fraud have been found to show up in eBay reviews [5], opinion spam [29][30][31], and false accounts [32][3][28]. Many methods have focused on labeling individual users, such as by using belief propagation (BP) [5][9] or TrustRank (PageRank-like) scores [24][28]. These methods label suspicious nodes/users, but do not return suspicious grouping behaviors themselves. Later work found that adding additional modes of information aided in detecting suspicious behavior. CopyCatch [4] found that suspicious patterns of Page Likes on Facebook correlated in time were good indicators of fraud. CatchSync [2] proposed the synchronicity and normality features to summarize the distributions of followers in a two-dimensional feature space, and thus caught the synchronized behaviors of zombie followers. Jindal et al. analyzed Amazon reviews, examining product, reviewer, rating, date, review title/body and feedbacks, to catch opinion spam [29]. 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. How can we label if an individual request from an IP is malicious?

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 of k modes. Rather, in this paper we study and quantify this pattern in a principled manner.

Decomposition methods: The singular value decompositions (SVD) and tensor decompositions have been widely used in subspace clustering [19], community detection [12], and pattern discovery [34][35]. Implicitly, the SVD focuses on dense regions of a matrix. Prakash et al. proposed EigenSpoke which reads scatter plots of pairs of singular vectors to find patterns and chip communities [23]. Chen et al. extracted dense subgraphs using a spectral cluster framework [12]. For multimodal data, tensor decompositions have been applied in many applications [20][33]. High-order singular value represented the importance of the cluster [19]. However, the decompositions do not offer the scoring function, and furthermore, we show in Section 3.3 that the decompositions has limitations to evaluate the cross-mode blocks.

Dense subgraph mining: Significant work has focused on looking for dense subgraphs with high av-

TABLE 2
Symbols and their definitions.

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}
ρ	The density, $c / \prod_k n_k$, of \mathcal{Y}
\hat{f}	Suspiciousness metric, parameterized by the masses
\tilde{f}	Suspiciousness metric, parameterized by the densities
$D_{KL}(\rho p)$	Directed KL-divergence of Poisson(p) & Poisson(ρ) $p - \rho + \rho \log \frac{\rho}{p}$

erage degree [13][15][16][36]. Charikar gave simple greedy approximation algorithms to find highly connected subgraphs of large average degree [27]. Quasi-cliques and K -cores introduced density-based metrics that were originally devised to measure dense components [11][12]. Specifically, Pei et al. adopted the density metric to evaluate the importance of the subgraphs. The method requires a manual setting of the threshold to select important subgraphs. Neither average degree nor density is applicable in evaluating multimodal datasets. Moreover, we are proposing to solve the problem of dense block detection in an automatic, multimodal way.

3 PROPOSED METRIC CRITERIA

Having given the high level intuition behind our perspective, now we give a precise problem definition. We focus on tensors where each cell contains a non-negative integer representing counts of events. We consider the mass of a subtensor to be the sum of entries in it, and the density to be the mass divided by its volume. The symbols and their definitions are listed in Table 2.

3.1 Problem Formulation

We formally give the definition of the problem of evaluating the suspiciousness of suspicious behaviors - mathematically, giving a suspiciousness score of dense blocks in the multimodal data.

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 the mass C , **define** a score function $f(\mathbf{n}, c, \mathbf{N}, C)$ to evaluate how suspicious a subtensor \mathcal{Y} of size $\mathbf{n} = [n_k]_{k=1}^K$ with the mass c .

We also consider an alternative parameterization using density: ρ is the density of \mathcal{Y} and p is that of \mathcal{X} :

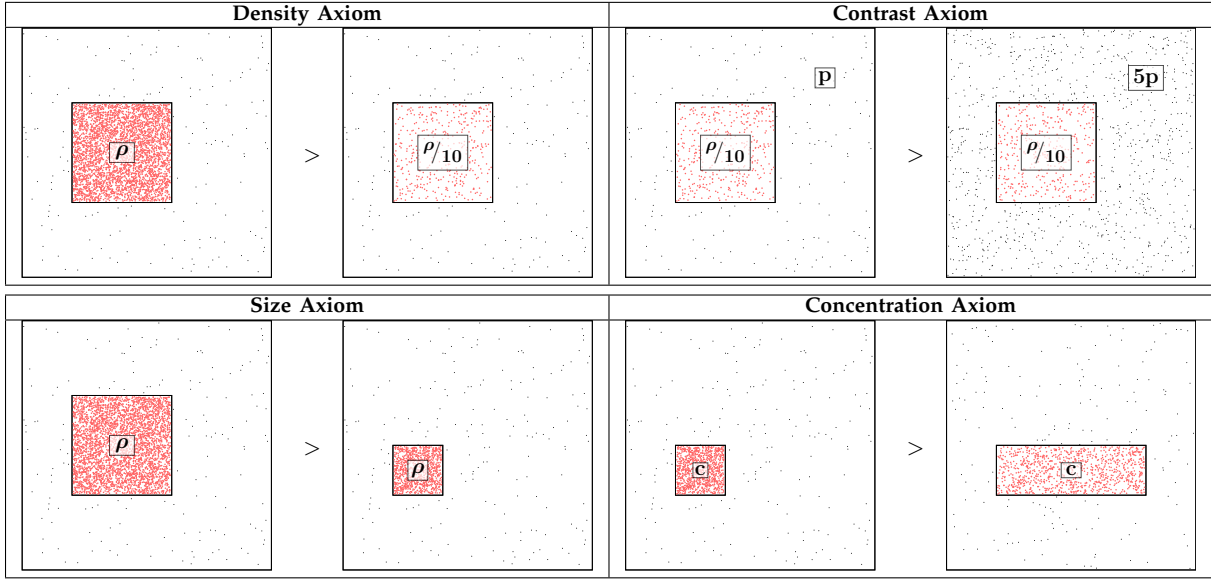
$$\hat{f}(\mathbf{n}, \rho, \mathbf{N}, p) = f\left(\mathbf{n}, \rho \prod_{k=1}^K n_k, \mathbf{N}, p \prod_{k=1}^K N_k\right) \quad (1)$$

In the rare case that the number of modes 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.

TABLE 3

A visual representation of the axioms: Density, Contrast, Size and Concentration. The blocks on the left are more suspicious (of higher “suspiciousness”) than those on the right. ($\rho = 0.1, c = 1000, p = 0.0008$)



3.2 Axioms

We now list five basic axioms that any suspiciousness metric f must meet. A pictorial representation can be found in Table 3.

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,

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

This axiom agrees with our intuition as well as the density axiom and size axiom. For example, 3 users post 100 tweets of 3 hashtags, which is more suspicious than a group of 10 users who post the same number of tweets of 10 hashtags.

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 ignored (was collapsed¹ into the remaining modes). Formally,

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

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

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

Proof:

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

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

In the experimental section, Table 8 and 11 show several multimodal blocks in real data, which demonstrates that our CROSSSPOT can capture the extensive attacks with Axiom 1-4 as well as the continuous attacks with Axiom 5. The block of the continuous attacks takes almost every timestamp value in the time mode.

3.3 Shortcomings of Competitors

While these axioms are simple and intuitive, they are non-trivial to meet. As shown in Table 1, 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.

1. *Collapsing* a tensor \mathcal{X} on mode K sums the values of \mathcal{X} across all indices in mode K [38], e.g., collapsing a tensor to a matrix: $\mathbf{X}_{i,j} = \sum_k \mathcal{X}_{i,j,k}$.

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 work on finding dense subgraphs focuses on the average degree of the subgraph [36][15], $f(\mathbf{n}, c, \mathbf{N}, C) = c/n_1$. 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.

Singular value: The SVD of a matrix \mathbf{A} is a factorization of the form $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^\top$. The singular values of \mathbf{A} correspond to $\Sigma_{r,r'}$ 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 [19][23]. As shown in [3], an independent block of size $n_1 \times n_2$ with mass c has a singular value σ 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, how can we extend this metric to more modes? 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. Later, we will observe in our proposed metric definition (Section 4) and experiments (Section 6) the drawbacks of the singular value in higher modes.

From the above, 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/suspicious behavior. We now offer our approach and demonstrate its effectiveness across all of these challenges.

4 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 Erdős-Rényi model [37], 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 Erdős-Rényi-Poisson model:

Definition 1: Erdős-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 ,

$$\mathcal{X}_i \sim \text{Poisson}(p).$$

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 Erdős-Rényi-Poisson model. Mathematically, given an $n_1 \times \dots \times n_K$ block of mass c in $N_1 \times \dots \times N_K$ data of total mass C , the suspiciousness score is

$$f(\mathbf{n}, c, \mathbf{N}, C) = -\log [\text{Pr}(Y_n = c)] \quad (2)$$

where Y_n is the sum of entries in the block.

4.1 Dense Subvector and Submatrix: 1-Mode and 2-Mode Suspiciousness

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:

$$\text{Pr}(X_i | p) = \frac{p^{X_i}}{X_i!} e^{-p}.$$

We are looking for an n -length subvector X_{i_1}, \dots, X_{i_n} that is unlikely and hence has a high suspiciousness.

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

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

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 $Y_n \sim \text{Poisson}(pn)$. The probability that Y_n equals a given number of retweets c is

$$\text{Pr}(Y_n = c) = \frac{(pn)^c e^{-pn}}{c!} = \frac{C^c}{c!} \left(\frac{n}{N} \right)^c e^{-\frac{cn}{N}}$$

Since the approximation for factorials (Stirling's formula) is as follows:

$$\log(c!) = c \log c - c + \mathcal{O}(\log c),$$

we obtain the suspiciousness score:

$$f(n, c, N, C) = -\log [\text{Pr}(Y_n = c)] = -\log \left[\frac{C^c}{c!} \left(\frac{n}{N} \right)^c e^{-\frac{cn}{N}} \right] \\ \approx c \left(\log \frac{c}{C} - 1 \right) + C \frac{n}{N} - c \log \frac{n}{N}.$$

Thus, we prove the formulas in the lemma. \square

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:

$$f([n_1, n_2], c, [N_1, N_2], C) = c(\log \frac{c}{C} - 1) + C \frac{n_1 n_2}{N_1 N_2} - c \log \frac{n_1 n_2}{N_1 N_2}$$

$$\hat{f}([n_1, n_2], \rho, [N_1, N_2], p) = n_1 n_2 D_{KL}(\rho || p)$$

4.2 Dense Subtensor: K-Mode Suspiciousness

We now extend the suspiciousness metric from low-order representations to a K -mode tensor.

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

$$f(\mathbf{n}, c, \mathbf{N}, C) = c(\log \frac{c}{C} - 1) + C \prod_{i=1}^K \frac{n_i}{N_i} - c \sum_{i=1}^K \log \frac{n_i}{N_i} \quad (3)$$

Using ρ as the block's density and p is the data's density, we have the simpler formulation

$$\hat{f}(\mathbf{n}, \rho, \mathbf{N}, p) = \left(\prod_{i=1}^K n_i \right) D_{KL}(\rho || p) \quad (4)$$

From the property of KL divergence, we have $f = \hat{f} \geq 0$. The non-negativity agrees with our intuition of the "suspiciousness".

4.3 Proofs: Satisfying the Axioms

Now that we have defined our suspiciousness metric, we prove that it meets all of the desired axioms proposed in Section 3.

Axiom 1: Density

Proof: Using Eq. (3), the derivative of the suspiciousness function with respect to the block's mass c is²

$$\frac{df}{dc} = \log \frac{c}{C} + \log \left(\prod_{i=1}^K \frac{N_i}{n_i} \right) = \log \frac{\rho}{p}$$

since $p = \frac{C}{\prod_{i=1}^K N_i}$ and $\rho = \frac{c}{\prod_{i=1}^K n_i}$. We are only considering blocks with higher density than the overall data, i.e. $\rho > p$, so $\frac{df}{dc} > 0$, i.e. suspiciousness increases with density. \square

Axiom 2: Size

Proof: Using Eq. (4), fixing n_k for $k \neq j$, the derivative of the suspiciousness function with respect to n_j is:

$$\frac{d\hat{f}}{dn_j} = \left(\prod_{k \neq j} n_k \right) D_{KL}(\rho || p) > 0$$

Thus, for fixed density ρ , as we increase any one dimension of the block with the remaining dimensions kept fixed, suspiciousness increases. \square

Axiom 3: Concentration

2. Formally, to take derivatives with respect to a discrete variable such as c , we extend Eq. (3) to take in c as real numbered input, then differentiate it with respect to c to show the desired monotonicity property. Then, the original, integer version of Eq. (3) agrees with the real numbered version whenever c is an integer, proving the desired monotonicity property for Eq. (3).

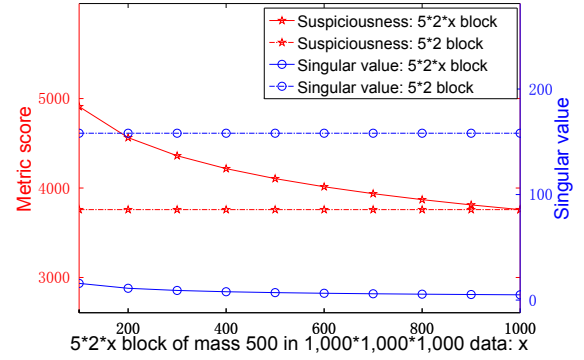


Fig. 2. Cross-mode comparisons: our suspiciousness metric obeys all the axioms, but the singular value breaks Axiom 5 (Multimodal).

Proof: Using Eq. (3), fixing n_k for $k \neq j$, the derivative of the suspiciousness function with respect to n_j is:

$$\frac{df}{dn_j} = \frac{C}{n_j} \prod_{i=1}^K \frac{n_i}{N_i} - \frac{c}{n_j} = \frac{c}{n_j} \left(\frac{p}{\rho} - 1 \right)$$

The last expression is negative since $\rho > p$. Thus, for a fixed mass, larger blocks are less suspicious. \square

Axiom 4: Contrast

Proof: Using Eq. (4), the derivative of suspiciousness with respect to the data density p is

$$\frac{d\hat{f}}{dp} = \left(\prod_{k=1}^K n_k \right) \left(1 - \frac{\rho}{p} \right)$$

Since $\rho > p$, we have $\frac{d\hat{f}}{dp} < 0$, so as the overall matrix gets denser, the block gets less suspicious. \square

Axiom 5: Multimodal

Proof: Using Eq. (3):

$$\begin{aligned} f_K([n_k]_{k=1}^{K-1}, N_K, c, [N_k]_{k=1}^K, C) \\ = c(\log \frac{c}{C} - 1) + C \frac{N_K}{N_K} \prod_{i=1}^{K-1} \frac{n_i}{N_i} - c \left(\log \frac{N_K}{N_K} + \sum_{i=1}^{K-1} \log \frac{n_i}{N_i} \right) \\ = c(\log \frac{c}{C} - 1) + C \prod_{i=1}^{K-1} \frac{n_i}{N_i} - c \sum_{i=1}^{K-1} \log \frac{n_i}{N_i} \\ = f_{K-1}([n_k]_{k=1}^{K-1}, c, [N_k]_{k=1}^{K-1}, C) \end{aligned}$$

The multimodal axiom indicates that taking one more mode cannot be less suspicious. \square

After the extensive proofs, we give an example to highlight the advantages of our "suspiciousness" over the singular value.

Example 1: Given a 3-mode tensor data of mass 10,000 and size $1,000 \times 1,000 \times 1,000$, suppose that we spot a 3-mode dense block of mass 500 and size $5 \times 2 \times x$; collapsing the 3rd mode of the 3-mode tensor, or directly given a 2-mode data of mass 10,000 and size $1,000 \times 1,000$, suppose that we spot a 2-mode dense block of mass 500 and size 5×2 . Figure 2 compares the suspiciousness scores and singular values of these dense blocks. Two observations are as follows.

- With the value of x decreasing, both metric scores of the 3-mode blocks increase. So, both the "suspiciousness" and the singular value obey Axiom 3 (Concentration).

- When $x < 1,000$, the suspiciousness of the higher-mode block is bigger than the lower-mode one, however, the singular value of the lower-mode block is bigger. So, the “suspiciousness” obeys Axiom 5 (Multimodal) and the singular value breaks it.

5 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.

5.1 The Detection Problem

Now we can formally give the definition of the problem of detecting suspicious behaviors - mathematically, detecting dense blocks in multimodal data.

Problem 2 (Suspicious block detection): **Given** dataset \mathcal{X} which is a $N_1 \times \dots \times N_K$ tensor of mass C , **find** a list of blocks in \mathcal{X} , in any subset of modes, with high suspiciousness scores, in descending order, based on Eq. (3) and (5).

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: $\mathcal{P}_j = \{p_1^{(j)}, \dots, p_{N_j}^{(j)}\}$. 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 the alternative parameterization for our suspiciousness function

$$\tilde{f}(\tilde{\mathcal{P}}, \mathcal{D}) = f([\tilde{\mathcal{P}}_j]_{j=1}^K, c(\tilde{\mathcal{P}}), [\mathcal{P}_j]_{j=1}^K, |\mathcal{X}|) \quad (5)$$

5.2 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 CrossSpot: Local Search with the Suspiciousness

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** $j = 1 \dots K$ **do**
 - 3: $\tilde{\mathcal{P}}_j \leftarrow \text{ADJUSTMODE}(j)$
 - 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 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 . The following Lemma demonstrates that the descending order of $\Delta c_{p_i^{(j)}}$ sorts the rows/values and including the top values will bring a more suspicious block. The general philosophy is a row that shares many events/points with the existing suspicious block is likely to be part of that block.

Algorithm 2 ADJUSTMODE(j)

- 1: $\tilde{\mathcal{P}}'_j \leftarrow \{\}$;
 - 2: $\mathbf{P}_j \leftarrow \{p_i^{(j)}\}_{i=1}^{N_j}$ sorted in descending order by $\Delta c_{p_i^{(j)}}$
 - 3: **for** $p_i^{(j)} \in \mathbf{P}_j$ **do**
 - 4: $\tilde{\mathcal{P}}'_j \leftarrow \tilde{\mathcal{P}}'_j \cup p_i^{(j)}$
 - 5: $\tilde{\mathcal{P}}' \leftarrow \{\tilde{\mathcal{P}}_{j'}\}_{j' \neq j} \cup \tilde{\mathcal{P}}'_j$
 - 6: **if** $\tilde{f}(\tilde{\mathcal{P}}, \mathcal{D}) \leq \tilde{f}(\tilde{\mathcal{P}}', \mathcal{D})$ **then**
 - 7: $\tilde{\mathcal{P}}_j \leftarrow \tilde{\mathcal{P}}'_j$
 - 8: **end if**
 - 9: **end for**
 - 10: **return** $\tilde{\mathcal{P}}_j$
-

Lemma 5: Holding constant $\tilde{\mathcal{P}}_{j'}$ for all $j' \neq j$, the optimal choice of values $\tilde{\mathcal{P}}_j \subseteq \mathcal{P}_j$ is the first n_j values in \mathbf{P}_j for some $n_j \leq N_j$.

Proof: We prove this by contradiction. Assume there is a subset $\tilde{\mathcal{P}}_j \subseteq \mathcal{P}_j$ that we believe to be the optimal choice of values but that $\tilde{\mathcal{P}}_j$ is not the first $|\tilde{\mathcal{P}}_j|$ values of \mathbf{P}_j . Therefore, there must exist a pair of values $p_i^{(j)}, p_{i'}^{(j)}$ where $p_i^{(j)} \in \tilde{\mathcal{P}}_j$ and $p_{i'}^{(j)} \notin \tilde{\mathcal{P}}_j$ but $\Delta c_{p_{i'}^{(j)}} > \Delta c_{p_i^{(j)}}$. By

Axiom 1, it is clear that removing $p_i^{(j)}$ and adding $p_{i'}^{(j)}$ to $\tilde{\mathcal{P}}_j$ results in a block with a higher suspiciousness score than the original, supposedly optimal block. From this contradiction, the optimal set of values for $\tilde{\mathcal{P}}_j$ must come from the top of \mathbf{P}_j . \square

Theorem 1: Holding constant $\tilde{\mathcal{P}}_{j'}$ for all $j' \neq j$, the $\tilde{\mathcal{P}}_j$ that maximizes $f(\mathbf{n}, c, \mathbf{N}, C)$ is found by ADJUSTMODE(j) in Algorithm 1.

Proof: Because ADJUSTMODE sorts \mathcal{P}_j and checks all possible values of n_j for mode j , Lemma 5 implies that ADJUSTMODE makes the optimal choice of values in each step. \square

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 6, even using randomly chosen seeds does well.

This starting point offers significant flexibility for CROSSSPOT to benefit from the findings of previous data mining work and side information. For example, we can use as a seed the dense regions found in each

TABLE 4
Data statistics: multi-modal datasets from social networks and network traffic.

Dataset	Mode #1	Mode #2	Mode #3	Mode #4	Mass
Retweeting	User id	Tweet id	IP address	Time (minute)	Retweet
	29,468,040	19,755,875	27,817,611	56,943	221,719,535
Tweeting hashtag	User id	Hashtag	IP address	Time (minute)	Tweet
	81,186,369	1,580,042	47,717,882	56,943	276,944,456
Tweeting from Los Angeles	User id	Hashtag	URL	Time (day)	Tweet
	14,949	24,711	76,950	113	402,036
Network traffic	Source IP	Destination IP	Port number	Time (second)	Packet
	2,345	2,355	6,055	3,610	230,836

rank of a singular value decomposition. Searching with multiple seeds is trivially parallelizable, so with more computational resources we can always choose additional random seeds or use additional prior methods as starting points for CROSSSPOT.

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 for real applications to catch suspicious behavior.

Convergence Guarantees: Our algorithm converges to a local optimum: by Theorem 1, we find that each time ADJUSTMODE is run, the value of $\tilde{f}(\hat{\mathcal{P}}, \mathcal{X})$ improves or stays constant. As such, since there are a finite number of possible subtensors and hence a finite number of possible (non-infinite) values of the objective, the algorithm must converge.

6 EXPERIMENTS

In this section, we conduct experiments to answer the following questions: (1) How effective is the proposed method CROSSSPOT in finding suspicious blocks? (2) Can CROSSSPOT discover suspicious behavioral patterns in real datasets? (3) How efficient is CROSSSPOT? The experimental results show that CROSSSPOT detects suspicious blocks more accurately and is more computationally efficient than competing baselines. We also use CROSSSPOT to identify large, dense blocks in a retweeting dataset, a hashtag promoting dataset and a network traffic dataset, and use side information to show that suspicious behavior is indeed identified.

6.1 Datasets

In our experiments we used extensive datasets including synthetically generated datasets, two large, new social networking datasets and a public network traffic dataset. A summary of the datasets can be found in Table 4.

Synthetic data: We adapt the Erdős-Rényi-Poisson model to generate multimodal data. The synthetic data is generated as a K -mode tensor of size $N_1 \times \dots \times N_K$ with mass C . Within the tensor we inject b dense blocks. Each block is assigned a size $n_1 \times \dots \times n_K$ and mass c . When an injected block falls in only a subset of modes \mathcal{I} , we set $n_i = N_i$.

Retweeting data: We use retweeting data from Tencent Weibo, one of the largest social networking platforms in China. These retweets consist of user id, tweet id, IP address, timestamp (from November 9 to December 20 in 2011) 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.

Tweeting hashtag data: As well as retweeting data, we use original tweets from Tencent Weibo that include hashtags in their content. The dataset consists of tuples of user id, hashtag, IP address, timestamp and tweet content. This dataset is interesting for hashtag hijacking and hashtag promotion, where purchased tweets will use popular hashtags to promote their own content or will tweet many times using a hashtag in an attempt to make it trend. By searching for dense, multimodal behavior, we hope to spot suspicious patterns of hashtag hijacking.

Tweeting from L.A. data: The dataset was crawled using Twitter Streaming API³ from August 1st to November 30th 2014. It consists of 0.4 million tweets from the Greater Los Angeles Area. We aim at detecting the online advertising campaigns that can be represented as dense blocks in the “user-hashtag-URL-time” 4-order tensor, i.e., a group of users post tweets of the same URLs with the same group of hashtags at the same time period.

Network traffic data: The network traffic log is public through a research effort to study Internet traffic of enterprises [39]. The data of thousands of packets was collected on servers within the Lawrence Berkeley National Lab (LBNL). Each packet trace includes source IP, destination IP, port number and a timestamp in seconds. We look for dense structures.

6.2 Experimental Setup

In this subsection, we introduce how we set up our experiments: baseline methods, parameter settings and evaluation methods.

Baselines: We compare our proposed method CROSSSPOT with the following baseline methods. All the methods utilize structured behavioral information in different ways.

- SVD and HOSVD (Higher-Order SVD) [17][23] compute the orthonormal spaces associated with the different modes of a tensor. The threshold value for

3. <https://dev.twitter.com/streaming/overview>

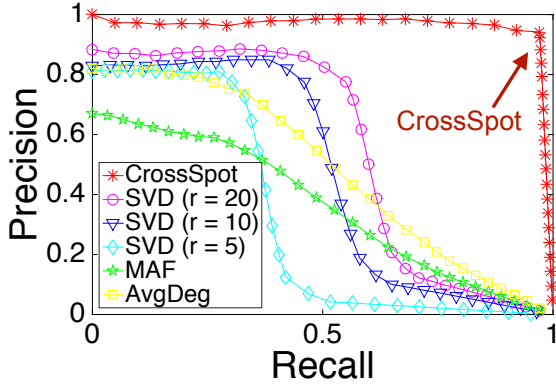


Fig. 3. Finding dense subgraphs: CROSSSPOT has nearly perfect performance on catching injected 2-mode blocks.

partitioning the decomposition vector is adaptively determined [23].

- MAF (MultiAspectForensics) [6] looks for spikes indicating the high-order subtensor (representing dense bipartite-core pattern) with eigenscore histogram vector and threshold parameters.
- AVGDeg (Dense SubGraph) [27] defines average degree as a metric of dense subgraph and develops a greedy method to find the dense components.

Parameter settings: We look for the best performance of every method. When running CROSSSPOT, we generate 1,000 random seeds to find their final blocks. We randomly decide the modes of a seed block and the set of values on each mode. We implement CROSSSPOT in Python. For the sake of efficiency in Algorithm 2 we prune out sparse values in each mode by stopping early if line 6 returns false. For SVD and HOSVD, we compare with different decomposition ranks such as 5, 10 and 20. We vary the threshold from 0 to 1 for every singular vector, considering rows (or columns) to be included in the block if their value in the singular vector is greater than the threshold. For other baselines, we use their standard implementations. We perform the experiments on a 2.40GHz×8 Intel Xeon CPU with 64GB RAM, running Windows Server 2008-64 bit.

Evaluation methods: To assess the effectiveness of our detection strategy in classifying suspicious and normal behaviors we use the standard information retrieval metrics of recall, precision and F1 score [40][2]. The recall is the ratio of the number of behaviors correctly classified to the number of suspicious behaviors. The precision is the ratio of the number of behaviors classified correctly to the total predicted suspicious behaviors. The F1 score is the harmonic mean of precision and recall.

6.3 Synthetic Experiments

We first evaluate CROSSSPOT on synthetic datasets. Overall, CROSSSPOT is effective: the tasks are to detect

- dense subgraphs in 2-mode graph data: extensive attacks such as ill-gotten Page Likes (“user-Page” links) and fake followers (“user-user” links),

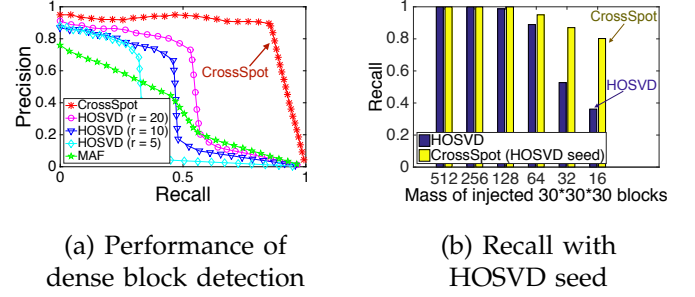


Fig. 4. Finding dense blocks: CROSSSPOT outperforms baselines in finding 3-mode blocks, and directly method improves the recall on top of HOSVD.

- dense k -mode blocks in k -mode tensor data: extensive attacks such as advertising campaigns (“user-phrase-URL-time” tuples),
- dense k' -mode blocks in k -mode tensor data ($k' < k$): continuous attacks that take every timestamp;

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

Effectiveness evaluations: We test the effectiveness of our proposed method CROSSSPOT in three tasks of finding suspicious behaviors in synthetic datasets represented by big, dense blocks in multimodal generated data, as well as the robustness of random seed number.

Finding dense subgraphs (2-mode blocks): We generate a random matrix under the ERP model with parameters as (1) the number of modes $k=2$, (2) the size of data $N_1=1000$ and $N_2=1000$, and (3) the mass of data $C=10,000$. We inject $b=6$ blocks of $k'=2$ modes into the random data, so, $\mathcal{I} = \{1, 2\}$. The size of every block is 30×30 and the block’s mass $c \in \{16, 32, 64, 128, 256, 512\}$. The task is to classify all the data entries into suspicious (injected) and normal classes. Figure 3 reports the classification performances of our proposed CROSSSPOT and the baselines of finding dense subgraphs. We observe:

- CROSSSPOT has nearly perfect precision: it only includes the entries that increase the suspiciousness because they belong to the dense blocks. It also has perfect recall: the local search does not miss any values in the block’s modes. The highest F1 of CROSSSPOT is 0.967, while the highest F1 scores of SVD, MAF, AVGDeg are 0.634, 0.439, and 0.511. MAF catches a big number of similar objects on some mode and thus it can catch very large blocks. AVGDeg catches very dense blocks but it will miss larger, less dense blocks.
- SVD has small recall but high precision. However, the SVD can hardly catch small, sparse injected blocks such as 30×30 submatrices of mass 16 and 32, even though they are denser than the background. Higher decomposition rank brings higher classification accuracy.

Note that the singular value $\sigma = \sqrt{\rho c}$ is relevant only with the information of the block including the density and mass but irrelevant with the data distribution. Our

TABLE 5

Our CROSSSPOT catches more lower-mode blocks: CROSSSPOT has high accuracy in finding the injected 4 blocks: (1) $30 \times 30 \times 30$, (2) $30 \times 30 \times 1,000$, (3) $30 \times 1,000 \times 30$, and (4) $1,000 \times 30 \times 30$, each of which has mass 1,000.

	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

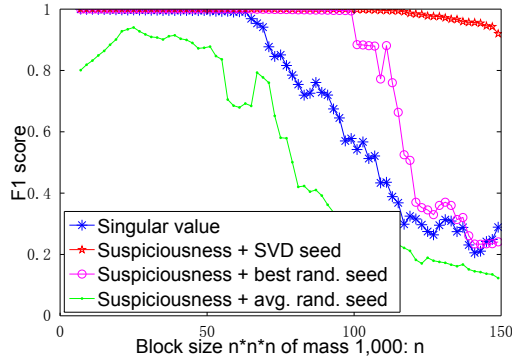


Fig. 5. Performance comparisons between different versions of CROSSSPOT and SVD. CROSSSPOT performs the best with SVD seeds.

metric evaluates the suspiciousness considering the distributions of both the block and the dataset. Even though the block is small and sparse but still looks suspicious in the data, CROSSSPOT can catch it with a high accuracy.

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 $N_1=1,000$, $N_2=1,000$ and $N_3=1,000$ and (3) the mass of data $C=10,000$. We inject $b=6$ blocks of $k'=3$ modes into the random data, so, $\mathcal{I} = \{1, 2, 3\}$. Each block has size $30 \times 30 \times 30$ and mass $c \in \{16, 32, 64, 128, 256, 512\}$. The task is again to classify the tensor entries into suspicious and normal classes. Figure 4(a) reports the performances of CROSSSPOT and baselines. We observe that in order to find all the six 3-mode injected blocks, our proposed CROSSSPOT has better performance in precision and recall 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 4(b) gives the recall value of every injected block. We observe that CROSSSPOT improves the recall over HOSVD, especially on slightly sparser blocks.

Figure 5 shows the performances of dense block detection: The different versions of CROSSSPOT include the performance with SVD seeds, the best performance with random seeds, and the average performance with random seeds. We inject a 3-mode dense block of mass 1,000 and size $n \times n \times n$ into the random tensor data of mass 10,000 and size $1,000 \times 1,000 \times 1,000$. We observe that

- CROSSSPOT performs better than high-order SVD when it uses SVD seeds or the best of random seeds.
- When the blocks become bigger, the F1 scores de-

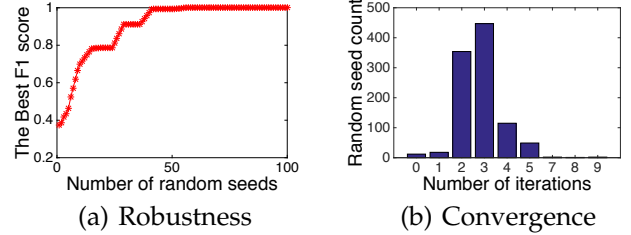


Fig. 6. CROSSSPOT is robust to the number of random seeds. In detecting the 4 low-order blocks, when we use 41 seeds, the best F1 score has reported the final result of as many as 1,000 seeds. CROSSSPOT converges very fast: the average number of iterations is 2.87.

crease, because the density of the dense blocks are smaller and the task becomes more difficult.

- CROSSSPOT with SVD seeds can perform almost perfectly: F1 scores are consistently more than 0.90. HOSVD considers only the block distribution. Especially when the blocks are very sparse in the tensor, it cannot catch the high-order blocks. Our metric evaluates the probability of the existence of a high-order block in the multimodal data. Therefore, CROSSSPOT can well catch the dense blocks.

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 dataset:

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

Note that blocks 2-4 are dense in only 2 modes and random in the third mode. Table 5 reports the classification performances of CROSSSPOT and baselines. We show the overall evaluations (precision, recall and F1 score) and recall value of every block. We 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.

TABLE 6
Big dense blocks with top metric values discovered in the retweeting dataset.

	Rank #	User×Tweet×IP×Minute	Mass c	Suspiciousness score
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

TABLE 7

Retweeting boosting: we spot a group of users retweet “Galaxy note dream project: Happy happy life travelling the world” in lockstep (every 5 minutes) on the same group of IP addresses. (Block 225×1×2×200 in Table 6)

User ID	Time	IP address (city, province)	Retweet comment (Google translator: from Simplified Chinese to English)
USER-A	11-26 10:08:54	IP-1 (Liaocheng Shandong)	Qi Xiao Qi: "unspoken rules count ass ah, the day listening...
USER-B	11-26 10:08:54	IP-1 (Liaocheng Shandong)	You gave me a promise, I will give you a result...
USER-C	11-26 10:09:07	IP-2 (Liaocheng Shandong)	Clouds have dispersed, the horse is already back to God...
USER-A	11-26 10:13:55	IP-1 (Liaocheng Shandong)	People always disgust smelly socks, it remains to his bed...
USER-B	11-26 10:13:57	IP-2 (Liaocheng Shandong)	Next life do koalas sleep 20 hours a day, eat two hours...
USER-C	11-26 10:14:03	IP-1 (Liaocheng Shandong)	all we really need to survive is one person who truly...
USER-A	11-26 10:18:57	IP-1 (Liaocheng Shandong)	Coins and flowers after the same amount of time...
USER-C	11-26 10:19:18	IP-2 (Liaocheng Shandong)	My computer is blue screen
USER-B	11-26 10:19:31	IP-1 (Liaocheng Shandong)	Finally believe that in real life there is no so-called...
USER-A	11-26 10:23:50	IP-1 (Liaocheng Shandong)	Do not be obsessed brother, only a prop.
USER-B	11-26 10:24:04	IP-1 (Liaocheng Shandong)	Life is like stationery, every day we loaded pen
USER-C	11-26 10:24:19	IP-1 (Liaocheng Shandong)	"The sentence: the annual party 1.25 Hidetoshi premature...

TABLE 8

Dense blocks discovered in hashtag data.

	Rank #	User×Hashtag×IP×Minute	Mass c	Suspiciousness score
CROSSSPOT	1	582×3×294×56,940	5,941,821	111,799,948
	2	188×1×313×56,943	2,344,614	47,013,868
	3	75×1×2×2,061	689,179	19,378,403
HOSVD	1	2,001×1×4×135	77,084	2,931,982
	2	327×1×2×401	212,519	8,599,843
	3	851×2×4×337	103,873	3,903,703

The F1 score of overall evaluation is as large as 0.972 with 68.8% improvement. None of the baseline methods consider the cross-mode setting, but our metric meets the multimodal axiom (Axiom 5). CROSSSPOT can detect the dense low-order blocks in the high-order tensor.

Testing robustness of the random seed number: We test how the performance of our CROSSSPOT improves when we use more seed blocks in the low-order block detection experiments. Figure 6(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 to the number of random seeds.

Efficiency analysis: CROSSSPOT can be parallelized into multiple machines to search dense blocks with different sets of random seeds. The time cost of every iteration is linear in the number of non-zero entries in the multimodal data as we have discussed in Section 5. Figure 6(b) reports the counts of iterations in the procedure of 1000 random seeds. We observe that usually CROSSSPOT takes 2 or 3 iterations to finish the local search. Each iteration takes only 5.6 seconds. Tensor decompositions such as HOSVD and PARAFAC used in MAF often take more time. On the same machine, HOSVD methods of rank $r=5$, 10 and 20 take 280,

1750, 34,510 seconds respectively. From Table 5 and Figure 6(a), even without parallelization, we know 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.

6.4 Retweeting Boosting

Table 6 shows big, dense block patterns of Tencent Weibo 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 more suspicious (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.

Table 7 shows an example of retweeting boosting from the big, dense 225×1×2×200 block reported by our proposed CROSSSPOT. A group of users (e.g., A, B, C) retweet the same message “Galaxy note dream project: Happy happy life travelling the world” in lockstep every 5 minutes on the same two IP addresses in the same city. We spot that their retweet comments are generated from some literature or art books. The periodicity of the retweets and the nonsensical comments are strong inde-

TABLE 9

Hashtag hijacking: we spot a group of users post tweets of multiple hashtags continuously on the same IP addresses. (Block $582 \times 3 \times 294 \times 56,940$ in Table 8)

User ID	Time	IP address (city, province)	Tweet text with hashtag
USER-D	11-18 12:12:51	IP-1 (Deyang, Shandong)	#Snow# the Samsung GALAXY SII QQ Service customized version...
USER-E	11-18 12:12:53	IP-1 (Deyang, Shandong)	#Snow# the Samsung GALAXY SII QQ Service customized version...
USER-F	11-18 12:12:54	IP-2 (Zaozhuang, Shandong)	#Snow# the Samsung GALAXY SII QQ Service customized version...
USER-E	11-18 12:17:55	IP-1 (Deyang, Shandong)	#Li Ning - a weapon with a hero# good support activities!
USER-F	11-18 12:17:56	IP-2 (Zaozhuang, Shandong)	#Li Ning - a weapon with a hero# good support activities!
USER-D	11-18 12:18:40	IP-1 (Deyang, Shandong)	#Toshiba Bright Daren# color personality test to find out your sense...
USER-E	11-18 17:00:31	IP-2 (Zaozhuang, Shandong)	#Snow# the Samsung GALAXY SII QQ Service customized version...
USER-D	11-18 17:00:49	IP-2 (Zaozhuang, Shandong)	#Toshiba Bright Daren# color personality test to find out your sense...
USER-F	11-18 17:00:56	IP-2 (Zaozhuang, Shandong)	#Li Ning - a weapon with a hero# good support activities!

TABLE 10

Advertising campaigns: we spot two accounts posting 2,605 tweets that promoted 75 URLs with 22 hashtags about job marketing at the Los Angeles from Sept. 17 to Sept. 24, 2014.

User ID	Time	Tweet content
USER-G	09-17 09:18:15	#TweetMyJobs #Job alert: General Marketing/Sales Six Flags #Valencia, CA http://t.co/jYgdKzbofq #Jobs
USER-G	09-17 09:28:30	#Valencia, CA #Job: Security at Six Flags http://t.co/DrUXy1GL4s #Jobs #TweetMyJobs
USER-H	09-17 09:47:47	#SupplyChain #Job in #SantaClarita, CA: Web Project Lead at Princess Cruises http://t.co/0DE6nK7rAf...
USER-H	09-17 10:09:01	Princess Cruises: Technical Manager, Electrical (#SantaClarita, CA) http://t.co/avHzdIyrpm #IT #Job...
USER-H	09-17 10:51:41	Princess Cruises: Specialist, Shore Excursions (#SantaClarita, CA) http://t.co/1hrY7uZn08 #BusinessMgmt...
USER-G	09-17 10:59:18	Six Flags #Job: Parking Attendants/Bus Drivers/Tram Drivers (#Valencia, CA) http://t.co/UG1XJRGYPQ...
USER-H	09-17 11:13:01	#SantaClarita, CA ...: Marketing & Sales Analyst - Consumer Sales at Princess Cruises http://t.co/l70oChgiWP
USER-H	09-17 11:34:19	#SantaClarita, CA ...: Business Development Manager ... Princess Cruises http://t.co/Muc0Kzb7NU #Jobs
USER-H	09-17 11:55:39	#Marketing #Job alert: Destination Marketing Specialist ... #SantaClarita, CA http://t.co/4bN1HoH4II #Jobs
USER-H	09-17 12:16:56	#SantaClarita, CA ...: Revenue Management Analyst at Princess Cruises http://t.co/vRkaUs2HPR #Jobs...
USER-G	09-17 12:30:40	Games - Six Flags: (#Valencia, CA) http://t.co/q6n4dMFqS1 #Job #Jobs #TweetMyJobs
USER-H	09-17 12:38:25	Princess Cruises #Hospitality #Job: Mgr, Prod Dev??? Creative & Guest Prog ... http://t.co/WaFqWRe7Pa #Jobs
USER-H	09-17 12:59:32	Princess Cruises: Data Analyst (#SantaClarita, CA) http://t.co/BERT4aOyX #BusinessMgmt #Job #Jobs...

TABLE 11

Big dense blocks in LBNL network data. The final suspicious blocks take all the time values indicating that suspicious traffic continuously happens in the hour.

	Rank #	Source IP×Destination IP×Port×Second	Mass <i>c</i>	Suspiciousness score
CROSSPOT	1	$411 \times 9 \times 6 \times 3,610$	47,449	552,465
	2	$533 \times 6 \times 1 \times 3,610$	30,476	400,391
	3	$5 \times 5 \times 2 \times 3,610$	18,881	317,529
	4	$11 \times 7 \times 7 \times 3,610$	20,382	295,869
HOSVD	1	$15 \times 1 \times 1 \times 1,336$	4,579	80,585
	2	$1 \times 2 \times 2 \times 1,035$	1,035	18,308
	3	$1 \times 1 \times 1 \times 1,825$	1,825	34,812
	4	$1 \times 13 \times 6 \times 181$	1,722	29,224

pendent evidence that the suspicious behavior found by CROSSPOT is actually fraudulent.

6.5 Hashtag Hijacking

Big, dense block patterns of tweeting hashtag data are illustrated in Table 8. CROSSPOT reports blocks of high mass and high density. We spot (1) continuous attacks: 582 users post as many as 5,941,821 tweets of the same 3 hashtags on 294 IP addresses for almost *every* minute in 43 days; (2) extensive attacks: 75 users post 689,179 tweets of the same hashtag on only 2 IPs in 35 hours. The *continuous attacks* in the top two blocks take almost all the timestamp values of the time mode. Our CROSSPOT can catch the continuous attacks based on the Axiom 5, which claims that taking one more mode is always more suspicious. HOSVD does not consider the cross-mode scenarios: it takes only the timestamps that have a “higher degree” value than a boundary. We can observe that it is not able to catch the continuous attacks. The blocks that CROSSPOT reports are more

suspicious than those blocks that HOSVD detects. The extensive attacks are caught based on Axiom 1-4, while the continuous attacks have to be caught with the cross-mode/multimodal axiom (Axiom 5).

Table 9 shows an example of hashtag hijacking from the big, dense $582 \times 3 \times 294 \times 56,940$ block. A group of users post tweets of advertising hashtags (e.g., #Snow#, #Li Ning - a weapon with a hero# and #Toshiba Bright Daren#) on multi-IPs of two cities in the same Province. This demonstrates that CROSSPOT catches the use of advertising hashtags to inflate popularity.

6.6 Advertising Campaigns

Most of the dense multimodal block patterns in the “tweeting from L.A.” data are “extensive attacks” that a small group of Twitter users post lots of tweet with the same small group of hashtags and URLs pointing to their external web platforms during a short time period. The most suspicious block is of the size $2 \times 22 \times 75 \times 8$ and as many as 2,605 events: 2 users posted 2,605 tweets that

promoted 75 URLs with 22 hashtags from September 17 to 24 in the year 2014. From Table 10, we spot that the major content of this group of tweets is about the job marketing at the Santa Clarita, CA. If we extract more modes from the data such as the locations and phrases, CROSSSPOT will be able to catch more types of meaningful block patterns.

6.7 Network Traffic

We illustrate big, dense block patterns of LBNL network traffic dataset in Table 11, comparing our proposed CROSSSPOT and HOSVD. CROSSSPOT reports blocks of high mass and high density. We spot (1) very big and dense blocks: 411 source IP addresses send a total of 47,449 packets (≥ 100 from each) to 9 destination IPs on 6 ports, or 533 source IPs send 30,476 packets to 6 destination IPs on the same port; (2) small but very dense blocks: 5 source IPs send 18,881 packets ($\geq 3,600$ from each, ≥ 1 for every second) to 5 destinations on 2 ports, or 11 source IPs send 20,382 packets to 7 destination IPs on 7 different port numbers. These subsets of events are extremely suspicious: the probability of their occurrence is smaller than 10^{-10^6} . We observe that the top four blocks that CROSSSPOT catches are all due to continuous attacks. HOSVD cannot catch these 3-mode blocks (collapsing the time mode). Therefore, it only catches extensive attacks that are less suspicious than the continuous attacks.

These subsets of events take all the values in time mode, forming 3-mode dense blocks in 4-mode data. The cross-mode results indicate that a group of source IP addresses continuously send packets to multiple destination servers with the same group of ports in every second of one hour.

6.8 Discussions

The suspiciousness metric is derived based on the ERP model of the multimodal data. It discovers the KL-divergence principle in evaluating the unexpected density. However, the model assumes each cell is independent in the normal data. In future work, we will derive the metric based on more sophisticated models. The KL-divergence principle directs us to the information-theoretically way: the Minimum Description Length (MDL) principle is a considerable option. It is straightforward to determine the importance of a block with how much bits can be saved in a compression manner.

7 CONCLUSION

In this paper, we addressed the novel problem of evaluating and detecting dense blocks that imply the suspicious behaviors in the multimodal behavioral data. Our main motivation was fraud detection, and more generally, attention routing. We proposed a set of five axioms that any good metric of suspiciousness should meet. Previous scoring function can only meet at most three of them. We proposed a novel metric based on

a principled probabilistic model, and we proved that it obeys all the axioms. Based on the metric, we develop a scalable algorithm called CROSSSPOT to catch the dense, suspicious blocks in the multimodal data. The extensive experimental results demonstrate that CROSSSPOT consistently improves the F1 score over the baseline methods, especially for the cross-mode scenarios.

REFERENCES

- [1] M. Jiang, P. Cui, and C. Faloutsos, "Suspicious Behavior Detection: Current Trends and Future Directions," in *IEEE Intelligent Systems*, no. 1, pp. 31–39, 2016.
- [2] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang, "Catchsync: catching synchronized behavior in large directed graphs," in *ACM SIGKDD*, pp. 941–950, 2014.
- [3] N. Shah, A. Beutel, B. Gallagher, and C. Faloutsos, "Spotting suspicious link behavior with fbox: An adversarial perspective," in *IEEE ICDM*, pp. 959–964, 2014.
- [4] A. Beutel, W. Xu, V. Guruswami, C. Palow, and C. Faloutsos, "Copycatch: stopping group attacks by spotting lockstep behavior in social networks," in *WWW*, pp. 119–130, 2013.
- [5] 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*, pp. 201–210, 2007.
- [6] K. Maruhashi, F. Guo, and C. Faloutsos, "Multiaspectforensics: Pattern mining on large-scale heterogeneous networks with tensor analysis," in *ASONAM*, pp. 203–210, 2011.
- [7] H.-H. Mao, C.-J. Wu, E. E. Papalexakis, C. Faloutsos, K.-C. Lee, and T.-C. Kao, "Malspot: Multi2 malicious network behavior patterns analysis," in *PAKDD*, pp. 1–14, 2014.
- [8] A. Mukherjee, B. Liu, and N. Glance, "Spotting fake reviewer groups in consumer reviews," in *WWW*, pp. 191–200, 2012.
- [9] L. Akoglu, R. Chandy, and C. Faloutsos, "Opinion fraud detection in online reviews by network effects," in *ICWSM*, 2013.
- [10] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang, "Inferring strange behavior from connectivity pattern in social networks," in *PAKDD*, pp. 126–138, 2014.
- [11] J. Pei, D. Jiang, and A. Zhang, "On mining cross-graph quasi-cliques," in *ACM SIGKDD*, pp. 228–238, 2005.
- [12] J. Chen and Y. Saad, "Dense subgraph extraction with application to community detection," in *IEEE TKDE*, vol. 24, no. 7, pp. 1216–1230, 2012.
- [13] O. D. Balalau, F. Bonchi, T. Chan, F. Gullo, and M. Sozio, "Finding subgraphs with maximum total density and limited overlap," in *WSDM*, pp. 379–388, 2015.
- [14] C. H. Ding, X. He, and H. Zha, "A spectral method to separate disconnected and nearly-disconnected web graph components," in *ACM SIGKDD*, pp. 275–280, 2001.
- [15] R. Andersen, "A local algorithm for finding dense subgraphs," in *Transaction on Algorithms*, vol. 6, no. 4, p. 60, 2010.
- [16] V. E. Lee, N. Ruan, R. Jin, and C. Aggarwal, "A survey of algorithms for dense subgraph discovery," in *Managing and Mining Graph Data*, pp. 303–336, 2010.
- [17] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," in *SIAM journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.
- [18] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," in *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.
- [19] 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 *ACM SIGKDD*, pp. 327–335, 2008.
- [20] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," in *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.
- [21] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web," in 1999.
- [22] J. Sun, H. Qu, D. Chakrabarti, and C. Faloutsos, "Neighborhood formation and anomaly detection in bipartite graphs," in *IEEE ICDM*, p. 8, 2005.
- [23] B. A. Prakash, A. Sridharan, M. Seshadri, S. Machiraju, and C. Faloutsos, "Eigenspokes: Surprising patterns and scalable community chipping in large graphs," in *PAKDD*, pp. 435–448, 2010.
- [24] Z. Gyögyi, H. Garcia-Molina, and J. Pedersen, "Combating web spam with trustrank," in *Vldb Endowment*, pp. 576–587, 2004.

- [25] B. N. Wu, V. Goel, and B.D. Davison, "Topical trustrank: Using topicality to combat web spam," in WWW, pp. 63-72, 2006.
- [26] S. Ghosh, B. Viswanath, and F. Kooti, "Understanding and combating link farming in the twitter social network," in WWW, pp. 61-70, 2012.
- [27] M. Charikar, "Greedy approximation algorithms for finding dense components in a graph," in Approximation Algorithms for Combinatorial Optimization, pp. 84-95, 2000.
- [28] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro, "Aiding the detection of fake accounts in large scale social online services," in NSDI, 2012.
- [29] N. Jindal and B. Liu, "Opinion spam and analysis," in WSDM, pp. 219-230, 2008.
- [30] F. Benevenuto, T. Rodrigues, V. Almeida, J. Almeida, and M. Goncalves, "Detecting spammers and content promoters in online video social networks," in ACM SIGIR, pp. 620-627, 2009.
- [31] X. Hu, J. Tang, Y. Zhang, and H. Liu, "Social spammer detection in microblogging," in IJCAI, pp. 2633-2639, 2013.
- [32] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang. "Detecting suspicious following behavior in multimillion-node social networks." in WWW companion, pp. 305-306, 2014.
- [33] J. Sun, D. Tao, S. Papadimitriou, P. S. Yu, and C. Faloutsos, "Incremental tensor analysis: Theory and applications," in ACM TKDD, vol. 2, no. 3, p. 11, 2008.
- [34] T. G. Kolda and J. Sun, "Scalable tensor decompositions for multi-aspect data mining," in IEEE ICDM, pp. 363-372, 2008.
- [35] L. Grasedyck, "Hierarchical singular value decomposition of tensors," in SIAM Journal on Matrix Analysis and Applications, vol. 31, no. 4, pp. 2029-2054, 2010.
- [36] Y. Asahiro, K. Iwama, H. Tamaki, and T. Tokuyama, "Greedy finding a dense subgraph," in Journal of Algorithms, vol. 34, no. 2, pp. 203-221, 2000.
- [37] M. EJ Newman, D. J. Watts, and S. H. Strogatz, "Random graph models of social networks." in PNAS, pp. 2566-2572, 2002.
- [38] J. Inah, E. E. Papalexakis, U. Kang, and C. Faloutsos, "Haten2: Billion-scale tensor decompositions," in ICDE, 2015.
- [39] R. Pang, M. Allman, M. Bennett, J. Lee, V. Paxson, and B. Tierney, "A first look at modern enterprise traffic," in SIGCOMM IMC, pp. 2-2, 2005.
- [40] Y. Yang, "An evaluation of statistical approaches to text categorization," in Information retrieval, vol. 1, no. 1-2, pp. 69-90, 1999.



Meng Jiang received his B.E. degree and Ph.D. degree in 2010 and 2015 at the Department of Computer Science and Technology in Tsinghua University. He is now a Postdoctoral Research Associate at University of Illinois at Urbana-Champaign. His main research interests include data mining, behavior analysis and social network analysis. He visited Carnegie Mellon University from 2012 to 2013. He has over 15 published papers on social recommendation and suspicious behavior detection in top conferences/journals of the relevant field such as IEEE TKDE, SIGKDD, TKDD and AAAI. He got the best paper finalist in ACM SIGKDD 2014.



Alex Beutel is a fifth year Ph.D. candidate at Carnegie Mellon University in the Computer Science Department. He previously received his B.S. from Duke University. His Ph.D. research focuses on large scale user behavior modeling, covering both recommendation systems and fraud detection systems. He has interned at Facebook on both the Site Integrity and News Feed Ranking teams, at Microsoft in the Cloud and Information Services Laboratory, and at Google Research. Alex's research is supported

by the National Science Foundation Graduate Research Fellowship Program and a Facebook Fellowship.



Peng Cui received his Ph.D. degree in computer science in 2010 from Tsinghua University and he is an Assistant Professor at Tsinghua. He has vast research interests in data mining, multimedia processing, and social network analysis. Until now, he has published more than 20 papers in conferences such as SIGIR, AAAI, ICDM, etc. and journals such as IEEE TMM, IEEE TIP, DMKD, etc. Now his research is sponsored by National Science Foundation of China, Samsung, Tencent, etc. He also serves as Guest Editor, Co-Chair, PC member, and Reviewer of several high-level international conferences, workshops, and journals.



Bryan Hooi is a Ph.D. student in Statistics and Machine Learning Joint PhD Program at Carnegie Mellon University. He received his B.S. degree and M.S. degree at Stanford University. His main research interests are high-dimensional inference, network analysis, biomedical and social science applications.



Shiqiang Yang received his B.E. and M.E. degrees from the Department of Computer Science and Technology of Tsinghua University in 1977 and 1983, respectively. He is now a Professor at Tsinghua University. His research interests include multimedia technology and systems, video compression and streaming, content-based retrieval for multimedia information, multimedia content security, and digital right management. He has published more than 100 papers and MPEG standard proposals. Professor Yang has

organized many conferences as program Chair or TPC member including PCM05, PCM06 Workshop On ACM Multimedia05, MMM06, ICME06, MMSP05, ASWC06, etc.



Christos Faloutsos is a Professor at Carnegie Mellon University. He has received the Presidential Young Investigator Award by the National Science Foundation (1989), the Research Contributions Award in ICDM 2006, the Innovations award in KDD10, 20 "best paper" awards, and several teaching awards. He has served as a member of the executive committee of SIGKDD; he has published over 200 refereed articles, 11 book chapters and one monograph. He holds five patents and he has given over 30 tutorials

and over 10 invited distinguished lectures. His research interests include data mining for graphs and streams, fractals, database performance, and indexing for multimedia and bio-informatics data.