ZOORANK: Ranking Suspicious Entities in Time-Evolving Tensors

Hemank Lamba[1], Bryan Hooi[1], Kijung Shin[1], Christos Faloutsos[1] and Jürgen Pfeffer[2]

[1]Carnegie Mellon University, USA, [2]TU Munich, Germany
{hlamba,kijungs,christos}@cs.cmu.edu, bhooi@andrew.cmu.edu,
juergen.pfeffer@tum.de

Abstract. Most user-based websites such as social networks (Twitter, Facebook) and e-commerce websites (Amazon) have been targets of group fraud (multiple users working together for malicious purposes). How can we better rank malicious entities in such cases of group-fraud? Most of the existing work in group anomaly detection detects lock-step behavior by detecting dense blocks in matrices, and recently, in tensors. However, there is no principled way of scoring the users based on their participation in these dense blocks. In addition, existing methods do not take into account temporal features while detecting dense blocks, which are crucial to uncover bot-like behaviors. In this paper (a) we propose a systematic way of handling temporal information; (b) we give a list of axioms that any individual suspiciousness metric should satisfy; (c) we propose ZOORANK, an algorithm that finds and ranks suspicious entities (users, targeted products, days, etc.) effectively in real-world datasets. Experimental results on multiple real-world datasets show that ZOORANK detected and ranked the suspicious entities with high accuracy, while outperforming the baseline approach.

1 Introduction

User-based systems, such as web-services like Amazon, Twitter or corporate IT networks, have become popular targets of fraud or attacks. A popular research problem is to detect the spammers/fraudsters/attackers that are trying to attack a given system [3,11,13,21]. Similarly, in the social networks setting, there are multiple websites where anyone can buy fake Facebook page-likes or Twitter followers. In all these cases, such fraudulent activities take the form of "lockstep" or highly synchronized behavior: such as, multiple users liking the same set of pages on Facebook, or multiple users following the same users almost at the same time on Twitter [3]. Such behavior results in dense blocks in matrices/ tensors. The reason behind these blocks is intuitive, as most of the fraudsters have constrained resources (accounts, IP addresses, time, etc.) and they reuse their resources to add as many fraudulent activities as possible to maximize their profits.

Various methods have been proposed to identify users exhibiting such behavior, which involve finding dense blocks in tensors [11,21] or clustering in subgraphs [3,23]. However, for security experts monitoring the systems, it is imperative to know which users are more suspicious than other users, since it directs their attention to such users for further analysis or actions. In this paper we propose a method that ranks entities effectively (see Figure 1) for a security analyst to view. Consider Figure 2; all three

users, A, B and C are participating in dense blocks (as they are part of the 2 rectangles), however their contribution towards the suspiciousness of each block is different. A core question we answer in our paper is as follows:

Informal Problem 1 (Individual Suspiciousness Metric) *Given multimodal temporal data in the form of (userId,productId,. . . ,timestamp), how can we find and score suspicious entities (e.g. users, activities, products, days,etc.)?*

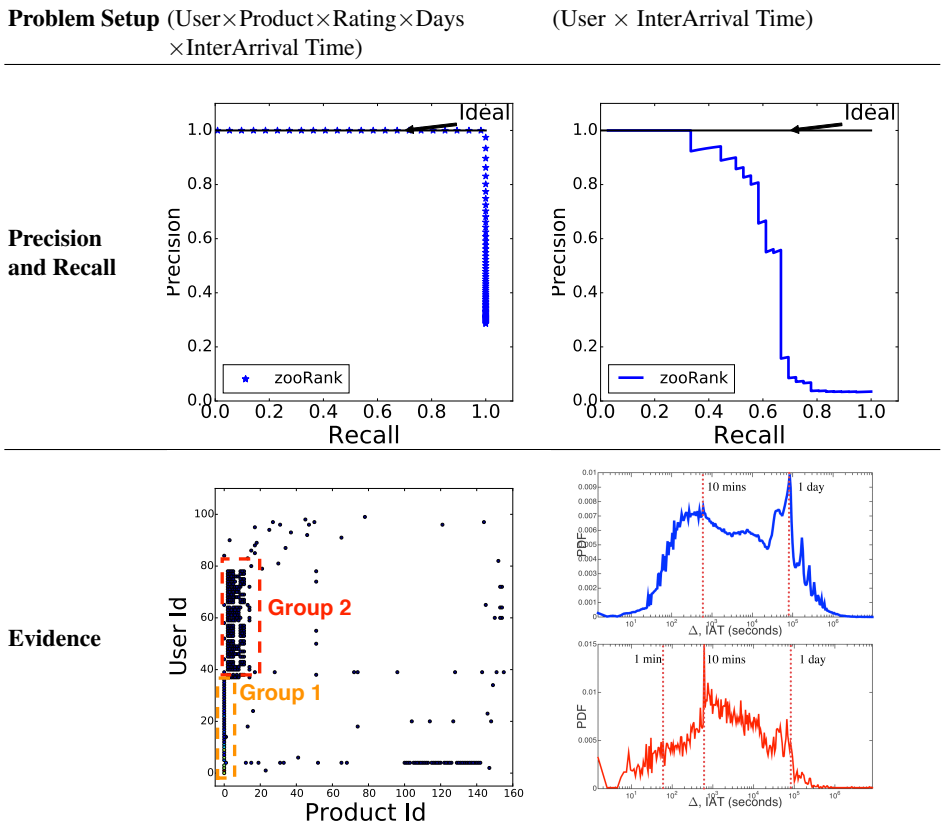


Fig. 1: **Effectiveness of ZOORANK on real world datasets.** (**Top Left**) Perfect precision-recall on software marketplace dataset. (**Top Right**) ZOORANK obtains good precision recall on Reddit dataset. (**Bottom Left**) Top 100 suspicious users found by ZOORANK show high synchronicity (formed groups) in rating and reviewing top suspicious products. (**Bottom Right**) The suspicious users (bottom; red) detected by ZOORANK for Reddit dataset show irregular spikes in inter-arrival time distribution, as compared to all the users (top; blue).

In addition, almost all the social networking websites and services have timestamps associated with every user activity. However, few approaches in the literature consider temporal features [3]. These timestamps can be useful for detecting fraudsters. However, it is not clear in dense block detection literature, in what ways we can incorporate the temporal information available to us. In this paper we answer the following question:

Informal Problem 2 (Temporal data handling) *Given data in the form of (cat 1, cat 2, . . . , timestamp), how can we generate features from timestamps useful for detecting fraudsters? Here cat 1,cat 2 are any categorical features (generally userId, productId, activityId, ratings, etc.)*

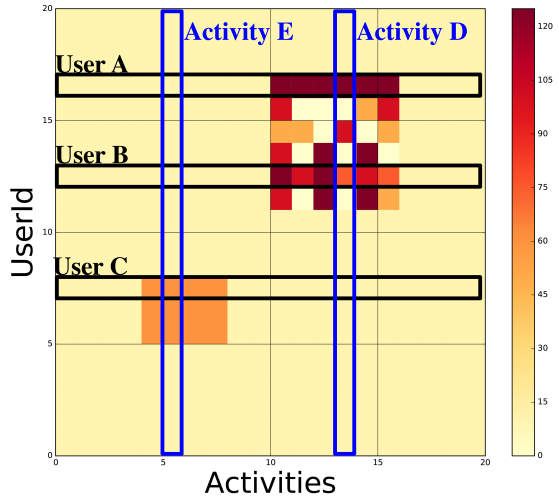


Fig. 2: How to rank users based on their suspiciousness, matching human intuition (A >B >C) ?

We propose ZOORANK, a novel approach for successfully scoring entities based on their participation in suspicious dense blocks. We introduce a set of axioms that any ideal individual scoring metric should satisfy. We show theoretically, that our proposed scoring function satisfies the proposed axioms. Additionally, ZOORANK also provides a framework to make good use of temporal information that generally exists in all the real-world datasets. As shown in Figure 1, ZOORANK successfully finds suspicious users in multiple real-world datasets (Software Marketplace data and Reddit data) with high accuracy. Additionally, the suspicious users found by our method showed clearly anomalous patterns. In Figure 1(Bottom Left), we see that multiple users are working in groups to target certain products. Similarly, in Figure 1(Bottom Right), the suspicious

users detected by our method show extremely regular and bot-like behavior resulting in spikes in the inter-arrival time distribution (difference in seconds between consecutive posts).

Our main contributions are as follows:

- **Theory**
  - *Axioms*: We propose a set of axioms that an individual scoring metric for measuring contribution of a user towards a suspicious block should follow.
  - *Metric*: We propose an individual suspiciousness scoring metric.
  - *Proofs*: We further prove that our proposed individual metric follows all the proposed axioms.
- **Temporal Features**: We provide a way of creating temporal features from the timestamp information present in the data.
- **Multimodality and Effectiveness**: The proposed approach ZOORANK can take into account various features, including temporal features. The approach detects suspicious entities in all modes of the data. We tested ZOORANK on various real-world datasets and were able to find suspicious entities with high accuracy, revealing interesting fraud patterns.

Reproducability: Our code and link to the datasets used is available at `https://goo.gl/rrvDTx`

2 Background and Related Work

A lot of work exists in the literature which aims at finding dense blocks, but none of the methods present a way of scoring the individual entities in dense blocks.

Detecting dense blocks: Densest-subgraph identification (i.e., the problem of finding a subgraph with maximum average degree) has been broadly studied in theory [5,8]. These theoretical results have been extended and applied to anomaly and fraud detection [11,20] since dense subgraphs (dense blocks) in real-world graph data tend to indicate fraudulent lock-step behavior, such as follower-buying services in Twitter. Spectral methods, which make use of eigen and singular value decomposition, also have been used for detecting dense subgraphs corresponding to 'cut-and-paste' bibliography in patent graphs [18], lock-step followers [13] and small-scale stealthy attacks [19] in social networks. Other approaches for dense-subgraph detection include co-clustering [3] and belief propagation [17]. Recently, dense-block detection in multi-aspect data also has been researched [12,21] for spotting groups synchronized in multiple aspects, such as IPs, review scores and review keywords. For our experiments, we use the best performing dense subgraph detection method M-Zoom [21]. The existing methods, however aim at only finding blocks, and do not provide a rank-list of users to inspect according to their suspiciousness.

Scoring Anomalies: Evaluating the anomalousness or suspiciousness of individuals is complementary to detecting dense blocks, which correspond to group activities. A widely-used approach is to detect outliers. Outlier detection methods are divided into parametric methods assuming underlying data distribution [10] and non-parametric methods using local features, such as distances to neighbors [14] and local

density [4, 15]. For graph data, on the other hand, various approaches, based on minimum description length [6, 9], neighborhood information [22], egonet features [2] have been proposed for scoring nodes. Many methods do exist in the literature, which use temporal information such as inter-arrival time [7, 16]. These features have been used to successfully detect bot-like behavior [7].

Our proposed method ZOORANK scores each entity (*individual-scoring*) in any of the dimensions (*multimodal*) of the tensor based on the entity's participation in the suspicious dense blocks (*dense-blocks*). It provides ways of transforming temporal data into useful features and thus handles both *numerical and categorical features*. A comparison between ZOORANK and other algorithms is summarized in Table 1. Our proposed method ZOORANK is the only one that matches all specifications.

	N-dim Outlier Methods			**Point Processes**		**Graph/Tensor based Methods**					
	GFADD /FADD [15]	**LoF** [4]	**Robust Random Cut** [9]	**RSC** [7]	**Self Feeding** [16]	**SPOKen** [18]	**CopyCatch** [3]	**CrossSpot** [12]	**M-Zoom** [21]	**FRAUDAR** [11]	ZOORANK
Dense Blocks						✓	✓	✓	✓	✓	✓
Individual Scoring	✓	✓	✓								✓
Numerical & Categorical Features							✓				✓
Multimodal and Extensible								✓	✓		✓
Temporal Features			✓	✓	✓		✓				✓

Table 1: Comparison of other methods and their features

3 Preliminaries and Problem Definition

3.1 Problem Definition

Definition 1 (K-way timed tensor) *A K-way timed tensor is a higher-order matrix containing entries of the form (category 1, category 2, ...,category K, timestamp).*

Many types of data including "like" data from Facebook (UserId, PageId, Timestamp), "follow" data from Twitter (UserId, FolloweeId, Timestamp), activity log from an organization (UserId, OperationId, Timestamp) or network data (Source IP, Source Port, Destination IP, Destination Port, Timestamp) all can be formulated as a K-way timed tensor. We now give a precise definition of the problem statements.

Problem 1 (Temporal Features Handling) *Given a K-way timed tensor $\mathcal{A}$, how can we effectively transform the temporal features associated with $\mathcal{A}$ to generate a categorical tensor $\mathcal{X}$?*

Symbol	Definition
$\mathcal{X}$	*Input categorical L-way tensor*
$\mathcal{Y}$	*Dense block within tensor $\mathcal{X}$*
$N_{\mathcal{Y}}^{i}$	*Size of ith mode of block $\mathcal{Y}$*
$m(i)$	*Mode of entity i*
$\rho_{\mathcal{Y}}$	*Density of block $\mathcal{Y}$*
$C_{\mathcal{X}}$	*Sum of the entries in $\mathcal{X}$*
$C_{\mathcal{Y}}$	*Sum of the entries in $\mathcal{Y}$*
$C_{\mathcal{Y}}(i)$	*Mass of entity i in $\mathcal{Y}$*
$V_{\mathcal{Y}}$	*Volume of the block $\mathcal{Y}$*
$g()$	*Block suspiciousness scoring function*
$f()$	*Individual-Suspiciousness scoring function*
$\delta_{\mathcal{Y}}(i)$	*Block level suspiciousness of entity i in block $\mathcal{Y}$*
$\mathcal{B}$	*List of suspicious dense blocks*
M	*Number of suspicious blocks to be considered*

Table 2: Symbols and Definitions

Problem 2 (Individual-Suspiciousness) *Given a L-way categorical tensor $\mathcal{X}$ of size $N_1 \times N_2 \times \cdots \times N_L$ with non-negative entries, compute a **score function** $f_{\mathcal{X}}(i)$, which defines the suspiciousness of entity i in the $m(i)^{th}$ mode of $\mathcal{X}$ with respect to the overall tensor $\mathcal{X}$.*

3.2 Block Level Suspiciousness Metrics

In this paper, we consider three block-level suspiciousness metrics although our proposed method is not restricted to them. The metrics are Arithmetic (g_{ari}), Geometric (g_{geom}) and Density (g_{susp}). Arithmetic computes the arithmetic average mass of a sub-block $\mathcal{Y}$ of a tensor $\mathcal{X}$. Similarly, Geometric metric is the geometric average mass of the block. The Density metric is the KL-divergence (Kullback Leibler) between the distribution of the mass in the sub-block with respect to the distribution of the mass in the tensor. These metrics are explained in the following sections.

3.3 Axioms

In this sub-section, we establish axioms that a good score function $f = f_{\mathcal{X}}(i)$ should satisfy. The suspiciousness of an entity should be based on its participation in dense blocks $\mathcal{B}$. Hence, our first two axioms govern the scores with respect to a single block $\mathcal{Y} \in \mathcal{B}$: our third axiom then governs how the single-block scores are combined to form $f_{\mathcal{X}}(i)$.

Let $\rho_{\mathcal{Y}}$ be the density (i.e. mass divided by volume) of $\mathcal{Y}$, and $\rho_{\mathcal{Y}}(i)$ be the density of the slice of $\mathcal{Y}$ defined by entity i. Similarly, let $\mathcal{C}_{\mathcal{Y}}(i)$ denote the mass of that same slice. The entire list of symbols is shown in Table 2.

Axiom 1 (Mass) *If an entity a has more mass than entity b in a block and given the fixed size of block in both the modes $m(a)$ and $m(b)$, then entity a is more suspicious. Formally*

$$\textit{IF } \mathcal{C}_{\mathcal{Y}}(a) > \mathcal{C}_{\mathcal{Y}}(b), \quad \textit{AND} \quad \mathcal{N}_{\mathcal{Y}}^{m(a)} = \mathcal{N}_{\mathcal{Y}}^{m(b)}, \textit{THEN } \delta_{\mathcal{Y}}(a) > \delta_{\mathcal{Y}}(b)$$

This is represented in Figure 2. See how entities are ranked by suspiciousness in the top right block (User A $>$ User B $>$ Activity D).

Axiom 2 (Concentration) *Given two entities a, b in different modes $m(a), m(b)$, where number of entities in one mode ($N_{\mathcal{Y}}^{(m(a))}$) is less than the number of entities in the second mode ($N_{\mathcal{Y}}^{(m(b))}$), then for fixed density, entity a is more suspicious than entity b.*

Formally,

$$\begin{aligned} &\textit{IF } N_{\mathcal{Y}}^{m(a)} < N_{\mathcal{Y}}^{m(b)} \quad \textit{AND} \quad \rho_{\mathcal{Y}} = \rho_{\mathcal{Y}}(a) = \rho_{\mathcal{Y}}(b), \\ &\qquad \textit{THEN} \quad \delta_{\mathcal{Y}}(a) > \delta_{\mathcal{Y}}(b) \end{aligned}$$

This is represented in Figure 2. See how entities are ranked by suspiciousness in the lower left block (User C $>$ Activity E).

Axiom 3 (Monotonocity) *If for every block, entity a has higher suspiciousness than entity b, then entity a has higher overall suspiciousness.*

Formally,

$$\textit{IF } \delta_{\mathcal{Y}}(a) > \delta_{\mathcal{Y}}(b) \quad \forall \mathcal{Y} \in \mathcal{B}, \textit{THEN } f_{\mathcal{X}}(a) > f_{\mathcal{X}}(b)$$

3.4 Shortcomings of Other Metrics

While these axioms are simple and intuitive, many other candidate metrics are not able to satisfy them. We consider some of them, and show why they fail.

Block Score: One simple metric to consider is the block suspiciousness score itself. The metric is to assign each individual the maximum block suspiciousness score out of all the blocks it is part of. The metric doesn't change if the two entities have different contributions to the block, and hence fails Axiom 1 (*Mass*) and Axiom 2 (*Concentration*).

SVD-score: Any matrix $\mathbf{A}$ can be decomposed using SVD decomposition as follows: $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathbf{T}}$. Each of the singular values in $\mathbf{\Sigma}$ represents the singular value related to a dense block that exists in the dataset. The metric here is the score of the maximum component for each user. This metric would again fail Axiom 1 (*Mass*) and Axiom 2 (*Concentration*).

Average δ-Block Score: Another proposed metric could be the average of all the contributions by the given entity to all the suspicious blocks. The contribution to a block

is computed as the difference in the suspiciousness between the block and the block after removing the specified entity. This statistic fails to satisfy Axiom 3 (*Monotonocity*) as if entity 1 has higher suspiciousness in 2 blocks than entity 2, but entity 2 exists only in one of the blocks, then the mean statistic is ambiguous.

As we show above, the metrics based on aggregation of block statistics do not work. In the following section, we propose ZOORANK, a scalable and effective method for finding and scoring suspicious individual entities in multimodal temporal data.

4 Proposed Approach: ZOORANK

4.1 Temporal Feature Handling

As mentioned, any data from a social networking website or a web service can be represented as a K-way timed tensor. We propose a way to handle such tensors by converting the numerical timestamp mode into interpretable categorical features. We propose to generate 0^{th}-order, 1^{st}-order, and temporal folding features.

- **0^{th}-order features:** The 0^{th} order features bucketize the timestamp into number of days, hours, minutes, etc. passed since the first observation was made. The temporal resolution can be chosen by practitioners based on the typical level of temporal variation present in their dataset.
- **1^{st}-order features:** Inter-arrival time is defined as the time interval between 2 consecutive timestamps of the same user. [7] found that bots tend to display regular inter-arrival time behavior such as performing an activity every exactly 5 minutes, due to automated scripts. To capture this pattern, we propose 1^{st}-order features, which is the log-bucketized inter-arrival time between 2 consecutive operations of a user (generalizable to any entity).
- **Temporal folding features:** We propose another way to detect fraudsters showing periodic behavior, which are common in bot-like behavior. For instance, a group of anomalous users might try to perform multiple login activities only from Wednesday 10 PM to 11 PM, or only on a specific day of the week. We work with 3 such features: 1) day of the week, 2) hour of the day and 3) hour of the week. We call these features temporal folding features.

4.2 Proposed Metric

Our metric is based on the δ-contribution of each entity towards the block suspiciousness score. We first define the δ-contribution for a given entity i in mode $m(i)$ of a specific block $\mathcal{Y} \in \mathcal{B}$, where $\mathcal{B}$ is a list of blocks. We denote this by $\delta_{\mathcal{Y}}(i)$

Definition 2 (Entity's Block-level Suspiciousness ($\delta_{\mathcal{Y}}(\mathbf{i})$)) *We define $\delta_{\mathcal{Y}}(i)$ as the difference between the suspiciousness score of block $\mathcal{Y}$ and block $\mathcal{Y}$ after removing entity i from the block i.e., $\delta_{\mathcal{Y}}(i) = g(\mathcal{Y}) - g(\mathcal{Y} \setminus i)$*

We need to aggregate the δ-metric over the entire list of blocks $\mathcal{B}$, in such a way that the given axioms are satisfied. We propose two metrics both of which satisfy the given axioms. The first metric is the sum of the δ-contributions, and the second is the maximum of the δ-contributions. We define the maximum metric as follows:

$$f_{\mathcal{X}}(i) = \max_{\mathcal{Y} \in \mathcal{B}} (\delta_{\mathcal{Y}}(i))$$

We empirically found that the maximum metric performs the best on the real-world datasets, and hence for the rest of the paper, all references to the proposed metric is for the maximum version of the metric.

4.3 Algorithm

After handling the temporal features, we produce a categorical tensor $\mathcal{X}$. Algorithm 1 defines the outline of ZOORANK. The first step is to compute suspicious blocks for the given tensor $\mathcal{X}$. To compute suspicious blocks, any existing method for block detection can be used.

We first find the M top suspicious dense blocks as determined by g (Line 1), where g is one of the metrics defined in Section 3.2. These top M suspicious blocks are stored in the list $\mathcal{B}$. For every entity i that has occurred at least once in any of the blocks in $\mathcal{B}$, we compute the individual suspiciousness score function f. This score function captures the contribution of a particular entity towards making the block suspicious. To do this, we compute the marginal contribution of each entity towards that block. This is equivalent to removing the entity i from the block, and re-computing the suspiciousness score (Lines 6-7). The difference between the new suspiciousness score and the original suspiciousness score is the marginal contribution of entity i. We compute the marginal contribution of each entity i over all the blocks (Lines 4-8). We define the individual suspiciousness score of the entity i as the maximum of the marginal contributions of entity i (Line 9). Another potential metric is to replace the maximization in Line 9 by the sum function. We conduct experiments with that metric as well.

This formulation of the scores $f_{\mathcal{X}}(i)$ satisfies intuitively reasonable properties, namely our axioms defined in Section 3.3:

Theorem 1. *The scores $f_{\mathcal{X}}(i)$ computed by Algorithm 1, using any of the metrics g_{ari}, g_{geo}, or g_{susp}, satisfies Axioms 1 to 3.*

Proof. :We first start by defining some of the standard block suspiciousness methods as follows:

$$g_{ari}(\mathcal{Y}, \mathcal{X}) = C_{\mathcal{Y}} / (\sum_{j} N_{\mathcal{Y}}^{j} / L)$$
$$g_{geo}(\mathcal{Y}, \mathcal{X}) = C_{\mathcal{Y}} / (V_{\mathcal{Y}}^{1/L})$$
$$g_{susp}(\mathcal{Y}, \mathcal{X}) = V_{\mathcal{Y}} \cdot \mathrm{D}(\rho_{\mathcal{Y}} || \rho_{\mathcal{X}})$$

where $\mathrm{D}(\rho_{\mathcal{Y}} || \rho_{\mathcal{X}}) = \rho_{\mathcal{X}} - \rho_{\mathcal{Y}} + \rho_{\mathcal{Y}} \log \frac{\rho_{\mathcal{Y}}}{\rho_{\mathcal{X}}}$.

Data: Tensor $\mathcal{X}$, block scoring function g, number of blocks to consider M, mode j to consider
Result: Individual scores for each entity i over the entire tensor : $f_{\mathcal{X}}(i)$

$\mathcal{B}$ = ComputeDenseBlocks($\mathcal{X}$, M, g)
for *each entity* $i \in N_j$ **do**
  $\delta_i = []$
  for $\mathcal{Y} \in \mathcal{B}$ **do**
    if $i \in \mathcal{Y}$ **then**
      Create new block $\mathcal{Y}'$ by removing the entries of entity i
      Append $(g(\mathcal{Y}) - g(\mathcal{Y}'))$ to δ_i
  end
  $f_{\mathcal{X}}(i) = \max(\delta_i)$
end
Sort and output $f_{\mathcal{X}}(i)$

Algorithm 1: ZOORANK: Detecting Suspiciousness Individuals

ZOORANK satisfies Axiom 1 (Mass)
If we fix the block's dimensions $N_{\mathcal{Y}}^1, \ldots, N_{\mathcal{Y}}^L$, all three metrics above strictly increase the mass of the block (i.e. $C_{\mathcal{Y}}$); this can be inferred directly from the form of g_{ari} and g_{geo}, and for g_{susp}.

As $\mathcal{C}_{\mathcal{Y}}(a) > \mathcal{C}_{\mathcal{Y}}(b)$, thus $\mathcal{Y} \setminus a$ has lower mass than $\mathcal{Y} \setminus b$, and since g is strictly increasing in mass (for fixed block dimensions), we get $g(\mathcal{Y} \setminus a) < g(\mathcal{Y} \setminus b)$. Therefore:

$$\begin{aligned}\delta_{\mathcal{Y}}(a) &= g(\mathcal{Y}) - g(\mathcal{Y} \setminus a) > g(\mathcal{Y}) - g(\mathcal{Y} \setminus b) \\ &= \delta_{\mathcal{Y}}(b)\end{aligned}$$

ZOORANK satisfies Axiom 2 (Concentration)
Using the same reasoning as above, it suffices to show $g(\mathcal{Y} \setminus a) < g(\mathcal{Y} \setminus b)$. Note that $N_{\mathcal{Y}}^{m(a)} < N_{\mathcal{Y}}^{m(b)} \Rightarrow V_{\mathcal{Y}\setminus a} < V_{\mathcal{Y}\setminus b}$ (since removing from a smaller mode decreases the volume more). Consider each metric g_{ari}, g_{geo}, and g_{susp} separately:

- **case 1:** g_{ari}.
  Here $\mathcal{Y} \setminus a$ and $\mathcal{Y} \setminus b$ have the same sum of block dimensions, and $\mathcal{C}_{\mathcal{Y}\setminus a} = \rho_{\mathcal{Y}} \cdot \mathcal{V}_{\mathcal{Y}\setminus a} < \rho_{\mathcal{Y}} \cdot \mathcal{V}_{\mathcal{Y}\setminus b} = \mathcal{C}_{\mathcal{Y}\setminus b}$ so that $g_{ari}(\mathcal{Y} \setminus a) < g_{ari}(\mathcal{Y} \setminus b)$.
- **case 2:** g_{geo}.
  Note that $g_{geo}(\mathcal{Y}) = C_{\mathcal{Y}}/(V_{\mathcal{Y}}^{1/L}) = \rho_{\mathcal{Y}} \cdot V_{\mathcal{Y}}/(V_{\mathcal{Y}}^{1/L}) = \rho_{\mathcal{Y}} \cdot V_{\mathcal{Y}}^{\frac{L-1}{L}}$. Thus:
  $$g_{geo}(\mathcal{Y} \setminus a) = \rho_{\mathcal{Y}} \cdot (\mathcal{V}_{\mathcal{Y}\setminus a})^{\frac{L-1}{L}} < \rho_{\mathcal{Y}} \cdot (\mathcal{V}_{\mathcal{Y}\setminus b})^{\frac{L-1}{L}} = g_{geo}(\mathcal{Y} \setminus b)$$
- **case 3:** g_{susp}.
  $g_{susp}(\mathcal{Y} \setminus a) = V_{\mathcal{Y}\setminus a} \cdot \mathrm{D}(\rho_{\mathcal{Y}} || \rho_{\mathcal{X}}) < V_{\mathcal{Y}\setminus b} \cdot \mathrm{D}(\rho_{\mathcal{Y}} || \rho_{\mathcal{X}}) = g_{susp}(\mathcal{Y} \setminus b)$

ZOORANK satisfies Axiom 3 (Monotonocity)

$$f_{\mathcal{X}}(a) = \max_{\mathcal{Y} \in \mathcal{B}} \delta_{\mathcal{Y}}(a) > \max_{\mathcal{Y} \in \mathcal{B}} \delta_{\mathcal{Y}}(b) = f_{\mathcal{X}}(b).$$

5 Experiments

In this section, we conducted experiments to answer the following questions:

- **Q1**: How effectively does ZOORANK find suspicious entities across all modes?
- **Q2**: How generalizable is ZOORANK over different datasets?
- **Q3**: Does ZOORANK scale linearly with size of the data ?

5.1 Datasets

We used various real-world datasets including a software marketplace dataset, a dataset from a popular social news aggregation website (Reddit), a dataset about Indian elections from Twitter, and a research lab's intrusion detection dataset.

- **Software Marketplace Dataset (SWM)**: We used the SWM dataset that was used previously by [1]. The dataset contains the reviews for all the products (software) under the entertainment category of the marketplace. The dataset contains 1,132,373 reviews from 966,839 unique users for 15,094 products. Each review has a rating from 1 to 5, and the timestamp on which the review was posted. The dataset, thus is in the format (UserId, ProductId, Rating, Timestamp). Previous studies [1, 23] manually annotated ground truth labels for suspicious users, which we considered as our ground truth.
- **Reddit Dataset**: Reddit is a social news aggregator website, which allows users to post, comment on, upvote and downvote stories. The dataset was collected and analyzed by [7]. The dataset contains 1,020,834 user comments for 1,036 users. The Reddit dataset is in the form (UserId, #Upvotes, #Downvotes, Length, Timestamp). The dataset has information about ground truth suspicious user accounts.
- **DARPA Intrusion Detection**: The DARPA intrusion detection dataset contains a sample of network data for the US Air Force laboratory. The dataset contains records in the format (Source IP, Destination IP, Timestamp). Further, it also contains labels for anomalous connections. For ground truth, we considered any source IP address that participates in at least 10 such anomalous connections, and any destination IP address that participates in at least 400 such connections. We altered this definition for ground truth thresholds and still achieved similar results as mentioned in the paper.
- **Indian Elections 2014 Dataset**: We collected tweets from 2014 Indian Elections. We crawled all the tweets from the 10% Sample API (Decahose). All the tweets contain the top 5 hashtags on Indian Elections per week. We further considered only those users who have at least 2 tweets in our dataset. This led us to a dataset of tweets from March, 2014 to May, 2014 consisting of 10,786 users.
- **Simulated Dataset**: We also tested our approach on a simulated dataset. For simulation, we used a realistic way of generating user-timestamps [7], then for each of the timestamp, we added activities based on a Poisson distribution. We simulated 3 blocks, comprising of 300, 400 and 200 genuine users respectively, where each block has different parameters for the activity Poisson distribution. For the suspicious blocks, we simulated three blocks for 50, 25 and 25 users respectively. The

first block does the most popular activity over the entire duration of the simulation and with random inter-arrival times. The second and third block do the second most and third most popular activities, respectively, at a steady inter-arrival time of 1 minute on a single day.

Experimental Settings: All our experiments were conducted on a machine on Intel(R) Xeon(R) CPU W3530 @ 2.80 GHz and 24 GB RAM. For all our experiments, we chose $M = 30$ and used M-Zoom [21] for dense block detection. We created multiple tensors based on different resolutions of time features (such as day of week, hour of the day, Inter-arrival time (in seconds, bucketized), etc.). However, we reported only the best accuracy obtained. The choice of what tensor to use, what block-level metric to use, and what value of M is appropriate, is for the practitioner to decide and depends on the type of data, on which the method is being applied.

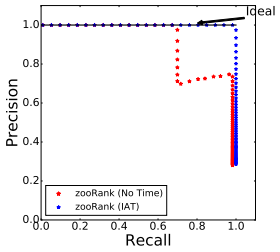
5.2 Q1. Effectiveness of ZOORANK

To test the effectiveness of ZOORANK, we compare our ranking of the suspicious entities with the ground truth suspicious users in our datasets. We further test the accuracy of our method on the SWM dataset. For software marketplace, we experimented with different versions of temporal features. Note that our algorithm achieves 100% accuracy in identifying suspicious users in the SWM dataset. From Figure 3a, we observed that adding the inter-arrival time feature increased the accuracy of the method. Our algorithm can rank entities in multiple modes; hence, we also tried to rank the products on basis of their suspiciousness. Though we do not have ground truth for which products were suspicious, we analyzed the top 5 suspicious products in Table 3b. We used the number of reviews by ground truth fraudsters as an indicator for suspiciousness. It can be observed that all the suspicious products are popular (high number of total reviews) and have also been targeted significantly from fraudsters (high number of fraud users). We also noticed that most of the reviews by fraudsters were highly synchronized and a large majority came on a single day (Figure 3c).

5.3 Q2. Generalizability of ZOORANK

We tested our method on multiple real-world datasets. In Table 3, we present our accuracy on each dataset. We observed that using maximum of the marginal contributions is better than using sum for all of the cases. Further, we also compared our method with a baseline approach. We define the following baseline: **Block Score**: defined as the maximum of all block suspiciousness scores a block is part of. From Figure 4, it can be observed that our approach clearly is better than the mentioned approach.

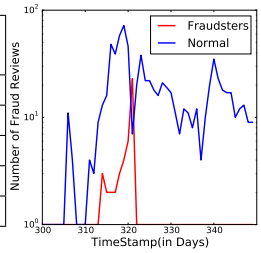
For Indian elections data, we did not have any ground truth. We extracted the top 100 suspicious users and evaluated them manually. The results for top 100 suspicious users are shown in Figure 5. The user ids are sorted by their suspiciousness score, and plotted on the scatter plot along with top suspicious hashtags. Figure 5 clearly shows groups of suspicious users. It is evident that the first two users are "hashtag hijackers". These two users tweeted spam messages with other hashtags but also focussed on generic hashtags related to the Indian elections. Both of these users have an identical behavior, which



(a) Effect of inter-arrival time: Performance on the user mode.

Index	Score	#Reviews Gen/Susp
1	44.625	34073 / 1371
2⋆	**2.349**	**2203 / 38**
3	1.35	222 / 66
4	1.33	5842 / 65
5	1.13	168 / 56

(b) List of Top 5 Suspicious Products.



(c) Performance on Product mode of SWM dataset.

Fig. 3: **ZOORANK is effective.** (a) It gives nearly 100% accuracy while identifying suspicious users in the SWM dataset. (b) ZOORANK marks products reviewed by known fraudsters as suspicious. (c) Product #2 received nearly all of it's reviews by fraud users on one single day.

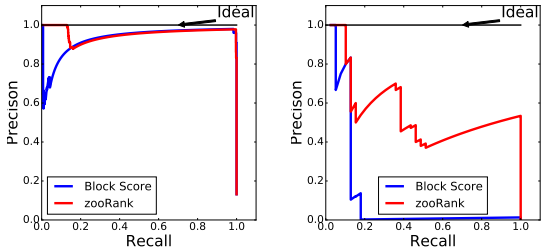
Dataset	F1-Score(SUM)	F1-Score(MAX)	Tensor
Reddit	0.62	**0.67**	User × Inter-Arrival Time (IAT)
SWM	0.98	**1.0**	User × Product × Rating ×Day × IAT
DARPA (SrcIP Mode)	0.97	**0.988**	SrcIP × DstIP × Day × IAT
DARPA (DstIP Mode)	0.29	**0.37**	SrcIP × DstIP × Hour × IAT

Table 3: ZOORANK is generalizable over multiple datasets, and multiple modes that exist in the datasets.

imply they do follow "lock-step" behavior. The second group of users were tweeting hashtags related to themselves and also generic hashtags related to the elections ("self-promoters"). We also spotted the user who tweets out all the trending topics at regular intervals, possibly through automated scripts ("trending topic aggregator"). We believe that the remaining users are users who were discussing indian elections a lot and were influencers in the political discussion. On further analysis, 20 users out of the 100 users were already suspended by Twitter. Thus, our algorithm was able to identify users that were considered spam by Twitter but also users that were missed by Twitter algorithm ("self-promoters") but were clearly malicious.

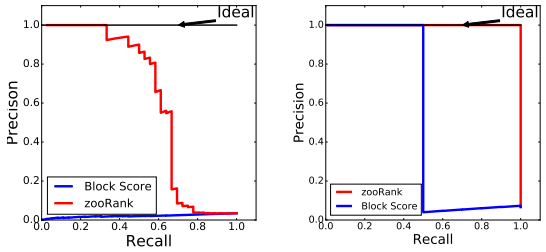
5.4 Q3. Scalability of ZOORANK

In this section, we evaluate the scalability of the ZOORANK. We measure the effects of the number of blocks and the number of records on the runtime of ZOORANK. To study the effect of the number of records, we generated the dataset with given number of entries in 3 dimensions, where cardinality of each dimension is 10^6. For all our results, we used arithmetic metric and operated on the most suspicious 30 blocks. The results



(a) Performance on DARPA (Source IP).

(b) Performance on DARPA (Target IP).



(c) Performance on Reddit Dataset.

(d) Performance on the Simulated Dataset

Fig. 4: **ZOORANK is generalizable**. ZOORANK outperforms the baseline across different modes (see (a) and (b)) and across multiple datasets (see (c) and (d))

are shown in Figure 6a, showing that our method scales linearly both in the data size and the number of blocks searched for. For the effect of the number of blocks, we generated a dataset with 10^4 records with a similar number of entries in each dimension.

6 Conclusions

In this paper, we proposed a set of axioms that a given individual suspiciousness scoring metric should follow. We presented such a metric that satisfies all the proposed axioms. Specifically, our contributions are as follows:

- **Individual-Suspiciousness Metric**: We propose a suspiciousness metric which scores each entity participating in dense blocks. The proposed criteria $f_{\mathcal{X}}(i)$ satisfies intuitive axioms.
- **Temporal Features**: The proposed method provides ways to transform the numerical timestamp mode to information rich categorical temporal features.
- **Effectiveness**: The proposed method ZOORANK was successfully tested on various real-world datasets. It scored the suspicious entities with high accuracy, and also uncovered interesting fraud patterns.

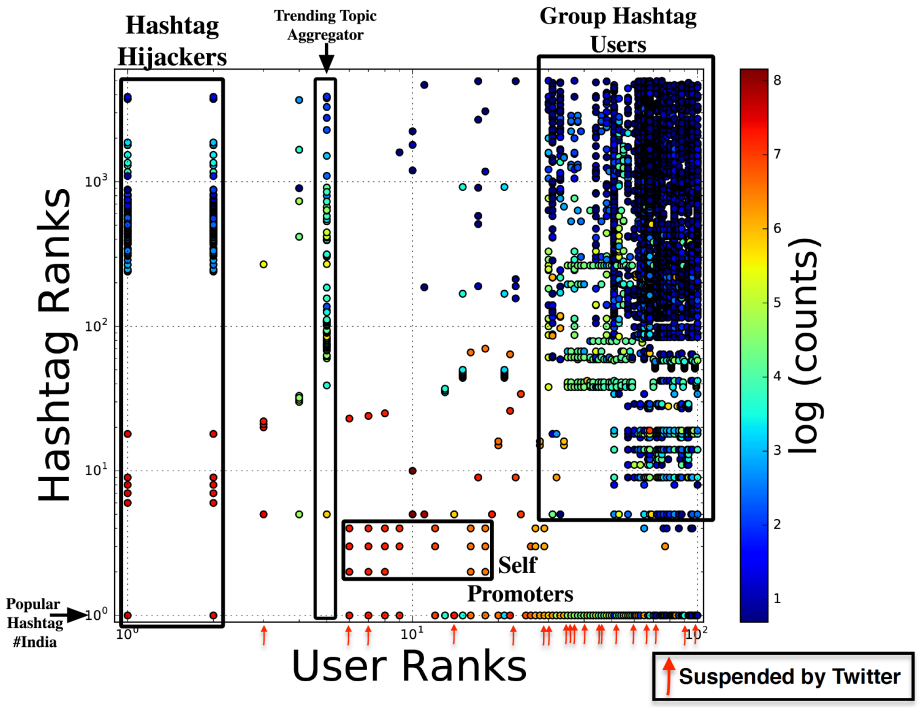
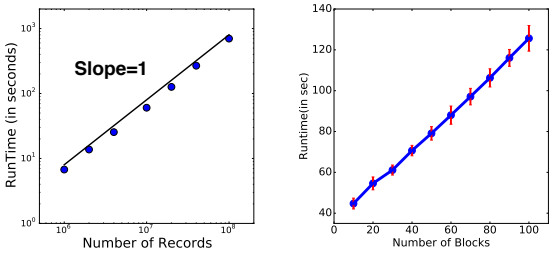


Fig. 5: **ZOORANK identifies fraudulent suspicious behavior in Twitter**: Top 100 suspicious users, and top hashtags as identified by ZOORANK.



(a) Number of Tuples. (b) Number of Blocks.

Fig. 6: **Scalability of ZOORANK** (a) ZOORANK scales linearly with the number of records. (b) ZOORANK scales linearly with the number of blocks we want to find.

– **Scalability**: The method is linearly scalable with the size of the data and can be used for *big-data* problems (see Figure 6).

Acknowledgement. This material is based upon work supported by the National Science Foundation under Grants No CNS-1314632, IIS-1408924, and by the Army

Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053. 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, 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. Akoglu, L., Chandy, R., Faloutsos, C.: Opinion fraud detection in online reviews by network effects. ICWSM (2013)
2. Akoglu, L., et al.: Oddball: Spotting anomalies in weighted graphs. In: PAKDD (2010)
3. Beutel, A., et al.: Copycatch: stopping group attacks by spotting lockstep behavior in social networks. In: WWW (2013)
4. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: Lof: identifying density-based local outliers. In: ACM sigmod record (2000)
5. Charikar, M.: Greedy approximation algorithms for finding dense components in a graph. In: APPROX (2000)
6. Eberle, W., et al.: Discovering structural anomalies in graph-based data. In: ICDMW (2007)
7. Ferraz Costa, A., et al.: Rsc: Mining and modeling temporal activity in social media. In: KDD (2015)
8. Goldberg, A.V.: Finding a maximum density subgraph. Technical Report (1984)
9. Guha, S., et al.: Robust random cut forest based anomaly detection on streams. In: ICML (2016)
10. Hawkins, D.M.: Identification of outliers, vol. 11. Springer (1980)
11. Hooi, B., Song, H.A., Beutel, A., Shah, N., Shin, K., Faloutsos, C.: Fraudar: Bounding graph fraud in the face of camouflage. In: KDD (2016)
12. Jiang, M., Beutel, A., Cui, P., Hooi, B., Yang, S., Faloutsos, C.: A general suspiciousness metric for dense blocks in multimodal data. In: ICDM (2015)
13. Jiang, M., et al.: Inferring strange behavior from connectivity pattern in social networks. In: PAKDD (2014)
14. Knox, E.M., Ng, R.T.: Algorithms for mining distancebased outliers in large datasets. In: PVLDB (1998)
15. Lee, J.Y., Kang, U., Koutra, D., Faloutsos, C.: Fast anomaly detection despite the duplicates. In: WWW Companion (2013)
16. Vaz de Melo, P.O.S., Faloutsos, C., Assunção, R., Loureiro, A.: The self-feeding process: A unifying model for communication dynamics in the web. In: WWW. pp. 1319–1330
17. Pandit, S., et al.: Netprobe: a fast and scalable system for fraud detection in online auction networks. In: WWW (2007)
18. Prakash, B., et al.: Eigenspokes: Surprising patterns and community structure in large graphs. PAKDD (2010)
19. Shah, N., Beutel, A., Gallagher, B., Faloutsos, C.: Spotting suspicious link behavior with fbox: An adversarial perspective. In: ICDM (2014)
20. Shin, K., Eliassi-Rad, T., Faloutsos, C.: Corescope: Graph mining using k-core analysis - patterns, anomalies and algorithms. In: ICDM (2016)
21. Shin, K., Hooi, B., Faloutsos, C.: M-zoom: Fast dense-block detection in tensors with quality guarantees. In: ECML/PKDD (2016)
22. Sun, J., Qu, H., Chakrabarti, D., Faloutsos, C.: Neighborhood formation and anomaly detection in bipartite graphs. In: ICDM (2005)
23. Ye, J., et al.: Discovering opinion spammer groups by network footprints. In: COSN (2015)