EdgeCentric: Anomaly Detection in Edge-Attributed Networks

Neil Shah[1], Alex Beutel[1], Bryan Hooi[1], Leman Akoglu[1], Stephan Günnemann[2], Disha Makhija[3], Mohit Kumar[3], and Christos Faloutsos[1]

[1]Carnegie Mellon University, {neilshah,abeutel,bhooi,lakoglu,christos}@cs.cmu.edu
[2]Technische Universität München, guennemann@in.tum.de
[3]Flipkart, {k.mohit,disha.makhiji}@flipkart.com

***Abstract*—Given a network with attributed edges, how can we identify anomalous behavior? Networks with edge attributes are ubiquitous, and capture rich information about interactions between nodes. In this paper, we aim to utilize exactly this information to discern suspicious from typical behavior in an unsupervised fashion, lending well to the traditional scarcity of ground-truth labels in practical anomaly detection scenarios. Our work has a number of notable contributions, including (a) *formulation*: while most other graph-based anomaly detection works use structural graph connectivity or node information, we focus on the new problem of leveraging edge information, (b) *methodology*: we introduce EDGECENTRIC, an intuitive and scalable compression-based approach for detecting edge-attributed graph anomalies, and (c) *practicality*: we show that EDGECENTRIC successfully spots numerous such anomalies in several large, edge-attributed real-world graphs, including the Flipkart e-commerce graph with *over 3 million* product reviews between *1.1 million* users and *545 thousand* products, where it achieved 0.87 precision over the top 100 results.**

I. INTRODUCTION

Given a graph with attributed edges, what can we say about the behavior of the nodes? For example, in a user-product graph with a rating attribute (1-5 stars) on edges, how can we discern which users rate (and which products are rated) normally or abnormally? Furthermore, between two users with varying edge behavior, can we say which is more suspicious? These are exactly the questions we address in this paper – more specifically, we focus on the problem of leveraging edge-attributes in social and information graphs for anomaly detection and user behavior modeling purposes. For practitioners, learning about their data in an unsupervised fashion when ground-truth is scarce or unavailable is an important setting, particularly in fraud and anomaly detection usecases. Furthermore, answers to these questions are invaluable for routing attention to the most anomalous behaviors in given data. Informally, our problem is as follows:

Problem 1 (Informal)**. Given** *a static graph with (multiple) numerical or categorical edge attributes,* **rank** *the nodes with most irregular edge behavior in a* **scalable** *fashion.*

This problem has numerous applications – graphs with edge attributes are ubiquitous in the real-world. Typically, these attributes take the form of numerical or categorical features which describe details about node interactions. For example, edges in unipartite social graphs (e.g. Facebook, Twitter) may be attributed with temporal information indicating the beginning of a friendship, and those in e-commerce networks may be attributed with ratings or purchase information.

In this work, we propose EDGECENTRIC, an effective information theoretic approach for general node-based anomaly detection in edge-attributed graphs. Specifically, our method leverages MDL (Minimum Description Length) to rank abnormality of nodes based on patterns of edge-attribute behavior in an unsupervised fashion. Figure 1 shows one application of EDGECENTRIC on the Flipkart e-commerce network, where it is able to spot fraudulent users giving too many atypical rating values. Figure 1a shows a collapsed 2-dimensional subspace of users produced from the original 5-dimensional rating space (users rate products from 1-5 stars) which spectral algorithms or practitioners may examine in an effort to identify anomalous behavior. In this space, we do not find any apparent, suspicious microclusters of abnormal users. However, Figure 1b shows that our EDGECENTRIC approach successfully identifies (amongst others) highly abnormal behaviors of users who give many ratings of only 5 stars (red) or 1 stars (green). These behaviors deviate substantially from global user behavior, shown as the blue J-shape in Figure 1c.

The main contributions of our work are as follows:

1) **Formulation:** We formalize the problem of anomaly detection on edge-attributed graphs using an information-theoretic approach.
2) **Methodology:** We develop EDGECENTRIC, an effective and scalable algorithm for the same.
3) **Practicality:** We experiment with our EDGECENTRIC on multiple large, real-world graphs and demonstrate its effectiveness and generality.

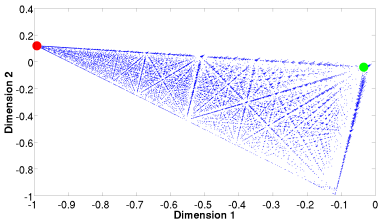
Reproducibility: Our code for EDGECENTRIC is open-sourced at www.cs.cmu.edu/~neilshah/code/edgecentric.tar.

II. RELATED WORK

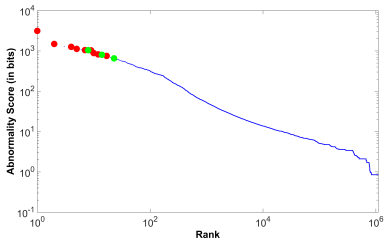
Prior work loosely falls into three categories: mining plain, node-attributed and edge-attributed graphs.

A. Mining unattributed graphs

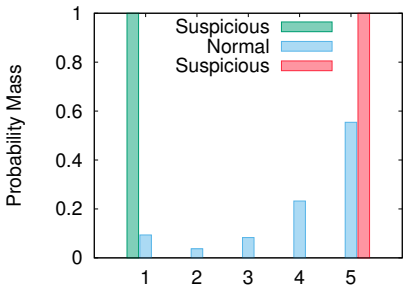
Akoglu et al. [3] identify power-law patterns in egonets and report deviating nodes as anomalous. Tong et al. [31] present a non-negative residual matrix factorization method to improve graph anomaly detection in low-rank subspaces. [29], [18], [17] propose spectral methods to spot fraudulent behavior in low-rank subspaces of social graphs. [12] proposes



(a) Two clusters (red and green) of hard-to-discern fraudsters shown in a collapsed 2D subspace, reduced from the original 5D subspace over user rating values (1-5).



(b) Our approach, EDGECENTRIC, identifies the users at the red and green clusters as highly abnormal.



(c) We find that the abnormal users in the red cluster give only 5 star ratings, whereas users in the green cluster give only 1 star ratings.

Fig. 1: **EDGECENTRIC spots abnormal users on real graphs.** Applied on a dataset of 3 million Flipkart user-product ratings, EDGECENTRIC finds users who greatly deviate from typical behavior – the red and green clusters contain single-mindedly "enthusiastic" and "disgusted" users who only give 5 star or 1 star reviews respectively, compared to the global (J-shape) behavior shown in blue.

a modified PageRank measure which penalizes users for social promiscuity. [25] and [2] use belief propagation to spot fraudsters in e-commerce graphs. [33] proposes the *network footprint score* to spot opinion spammers, exploiting self-similarity and neighborhood diversity.

Dense subgraph discovery is also relevant to the anomaly detection task. Numerous methods exist for graph partitioning, including METIS algorithm [20] and spectral methods [32], [28]. Several information-theoretic approaches automate parameter choices, including cross-associations [9], VOG [22] for static and TIMECRUNCH [30] for dynamic graphs.

B. Mining graphs with node attributes

[11] unifies structural and attribute similarity and infers communities using hidden Markov random fields. [27] introduces a "focused" clustering approach which identifies clusters and outliers given a set of seed nodes. [24] proposes an MDL formulation for identifying common graph substructures.

[23] uses spectral clustering to group homogeneous node-attributed relational data. [13] introduces a pruning-based algorithm to identify subspace clusters which also exhibit strong graph connectivity. [4] proposes an MDL formulation for jointly reordering connectivity and feature matrices to identify attributed clusters. [34] formulates a distance measure to weight the contributions of graph structure and node attribute similarity for clustering.

C. Mining graphs with edge attributes

There is less prior work in mining edge-attributed graphs. In some cases, an edge attribute is construed as a weight, which can be used by some cut-based [1] and spectral clustering [23] approaches. In our setting, we consider each edge as an interaction and each attribute as a descriptive *feature*.

The recommendation systems community has also focused on learning models of graphs with ratings [21] and in some cases find outliers [5]. [15] focuses on mining online reviews, which can be construed as textual attributes. [19] introduces linguistic indicators of fraud in online reviews.

[10] introduces a log-logistic model for call duration in phone-call networks. [6] uses local graph search on the Facebook user-likes-page graph with temporal edge features to find fraudulent subgraphs. [7], [8] propose methods for mining dense subgraphs with similar attribute subspaces. [16] proposes a related metric of suspiciousness for dense blocks but is based on Poisson distributions and limited to count data. [14] models the distribution of ratings and interarrival times from a Bayesian perspective and suggests priors. Our work differs in that it takes a frequentist approach based on MDL and is designed to handle any set of edge-attributes on complex heterogeneous graphs.

Summarily, unlike previous methods, our EDGECENTRIC approach (a) needs no priors or labels (b) extends naturally to heterogeneous networks, (c) supports multiple edge-attributes and (d) ranks anomalies.

III. PROBLEM FORMULATION

In this section, we outline the first core contribution of our work: specifically, we formalize the problem of detecting anomalous nodes using edge attributes by leveraging a compression paradigm based on MDL. For clarity, see Table I for an overview of the recurrent symbols used in future discourse.

A. Preliminaries

The MDL principle states that given a model family $\mathcal{M}$, the best model $M \in \mathcal{M}$ for data $\mathcal{D}$ is that which minimizes the $L(M)+L(\mathcal{D}|M)$, where $L(M)$ is the description length used to describe the model M, and $L(\mathcal{D}|M)$ is the same for encoding $\mathcal{D}$ using M. MDL enforces lossless encoding to fairly evaluate various models. The intuition behind our ranking approach is that data which fits the model well enjoys high compression, while poorly reprsented data is costly to encode.

In our problem setting, we are given a static directed or undirected multigraph $G(\mathcal{V},\mathcal{E},m)$ in which nodes are connected by edges. Technically, $m:\mathcal{E}\rightarrow\{\{u,v\}\,|\,u,v\in\mathcal{V}\}$ assigns each edge $e\in\mathcal{E}$ to a pair of nodes. Furthermore, we have object type and relation/edge type mapping functions $\Phi:\mathcal{V}\rightarrow\mathcal{B}$ and $\Psi:\mathcal{E}\rightarrow\mathcal{R}$, where each node $v\in\mathcal{V}$ is characterized by an object type $\Phi(v)\in\mathcal{B}$ and edge $e\in\mathcal{E}$ is characterized by a relation type $\Psi(e)\in\mathcal{R}$. Here, we define an object type to reflect a node "role," – for example, a user or product. A relation type reflects the relationship between two objects – for example, user-rates-product. R. When $|\mathcal{B}|=1$ and $|\mathcal{R}|=1$, the graph is *homogeneous*;

TABLE I: Frequently used symbols and definitions

Symbol	Definition
G	static input graph
$\mathcal{V}$,$\lvert\mathcal{V}\rvert$	node-set, # of nodes of G resp.
$\mathcal{E}$,$\lvert\mathcal{E}\rvert$	edge-set, # of edges of G resp.
$m(\cdot)$	function to realize the multi-graph
$\mathcal{A}$,$\lvert\mathcal{A}\rvert$	attribute-set, # of total attributes across edges in $\mathcal{E}$ resp.
$\mathcal{B}$,$\mathcal{R}$	set of object types and relation types resp.
$\Psi(\cdot)$	maps nodes in $\mathcal{V}$ to object types in $\mathcal{B}$
$\Phi(\cdot)$	maps edges in $\mathcal{E}$ to relation types in $\mathcal{R}$
$\Omega(\cdot)$	maps relation types in $\mathcal{R}$ to attribute sets in $2^{\mathcal{A}}$
$\delta(\cdot)$	unified abnormality function, defined on nodes in $\mathcal{V}$
U,P	user, product resp.
C,$C(i)$	global (model) dist. C, prob. mass of ith element resp.
$C_{u,r,w,j}$,$C_{p,r,w,k}$	jth (kth) U (P) model dist. on attr. w and rel. r resp.
$\rho_{u,r,w,j}$,$\rho_{p,r,w,k}$	jth (kth) U (P) cluster prop. on attr. w and rel. r resp.
$\hat{U}$,$\hat{P}$	discrete prob. dist. (of ratings) for U and P resp.
$f_{u,r}$,$f_{p,r}$	rating vectors for U and P on relation r resp.
$H(\cdot)$	Shannon entropy in bits, defined on discrete prob. dist.
$KL(\cdot \,\|\, \cdot)$	KL divergence in bits, defined on two discrete prob. dists.
M	data model M
$L(U,M)$	# of bits used to encode M and U's behavior given M
$L(M)$	# of bits to encode M

otherwise, it is *heterogeneous*. Furthermore, edges of each relation $r \in \mathcal{R}$ are labeled with values corresponding to the same finite subset of numerical or categorical attributes chosen from attribute set $\mathcal{A}$, given by the mapping $\Omega : \mathcal{R} \to 2^{\mathcal{A}}$, where $2^{\mathcal{A}}$ denotes the power set of $\mathcal{A}$. In other words, the graphs we consider can have numerous relation types, and edges of each relation type are characterized by a fixed number of the same attributes (at least 1). In the remainder of the problem formulation, let us consider a simple, undirected user-product graph, in which $|\mathcal{B}| = 2$ (users and products) and $|\mathcal{R}| = 1$ (user-rates-product) for ease of explanation. Let us also assume that we have only one attribute on the edges: product rating in stars (1-5).

Then, our formal problem definition is as follows:

Problem 2 (Formal)**. Given** *a static multigraph $G(\mathcal{V},\mathcal{E},m)$ with ≥ 1 numerical or categorical edge attributes chosen from $\mathcal{A}$,* **devise** *an abnormality function $\delta(\cdot)$ to score each node $v \in \mathcal{V}$ based on its edge attribute behavior, and* **identify** *the most irregular nodes in a* **scalable** *fashion.*

B. Intuition

In order to see how we can leverage MDL to inspire the formulation of δ, we must first consider our model and data representations. To encode each user node, we must store information about the user's interactions through edges. In our running example, because each edge simply contains information about a single categorical attribute value (1-5), we must encode the attribute value to losslessly reconstruct the vector which describes the user's rating behavior. Thus, for each user node, we will encode a vector of rating values, e.g. [5,5,1,2,5,3,...]. Likewise, to encode a product node, we store the product's rating vector: say, [1,2,1,3,2, 2,...].

To encode these individual user and product rating vectors, we first build a general model of rating behavior over *all* users and products, respectively. Note that this can be construed as an elementary user/product behavior model (we will relax the assumptions for a single model of behavior later in the section). For example, presume that the general pattern of rating behavior over all users follows the distribution [0.15,0.1,0.05,0.3,0.4] (total proportions of 1s, 2s, 3s, 4s and 5s, respectively). Then, we can describe this model distribution C as a trend that we expect a given user U's rating vector f_u to obey, and describe $|f_u|$ with respect to this model in our formulation. In doing so, our total encoding length in bits for each user U with distribution $\hat{U}$ is as follows:

$$L(U,M) = L(M) + L(U|M)$$

where

$$L(U|M) = |f_u| \cdot \big(H(\hat{U}) + KL(\hat{U} \,\|\, C)\big)$$

A similar cost could be written for each product. Following the earlier description of MDL, $L(M)$ is the cost to encode the overall model (in our case C), and $L(U|M)$ is the cost to encode a fixed user's data given the model. Here the cost for encoding the user based on the model includes the Shannon entropy H and the Kullback-Leibler divergence KL. While the Shannon entropy reflects the inherent information content of the distribution $\hat{U}$, the KL divergence captures the extra information content between the user distribution $\hat{U}$ and the model distribution C:

$$KL(U \,\|\, C) = \sum_i U(i) \log_2 \frac{U(i)}{C(i)}$$

Here both U and C are distributions over a discrete set of outcomes, and $U(i)$ and $C(i)$ denote the probability mass associated with outcome i. Given that $H(\hat{U}) + KL(\hat{U} \,\|\, C)$ describes the cost of encoding a single sample from the user distribution $\hat{U}$ according to the model distribution C, we multiply by $|f_u|$ to denote the cost of $|f_u|$ total samples from the user rating vector.

Although the general construction described above is required for fully encoding and reconstructing the rating vector given by a single user U (product P) according to MDL, our goal is to be able evaluate and compare the abnormality δ of two users (products) U_1 and U_2 according to our data model, rather than evaluate the model itself. In this regard, the last components of the above description, $|f_{u_1}| \cdot KL(\hat{U_1} \,\|\, C)$ and $|f_{u_2}| \cdot KL(\hat{U_2} \,\|\, C)$, are especially useful. Intuitively, these terms measure the total number of *extra* bits required to encode the attribute behavior of users U_1 and U_2 using a code optimized for the global model distribution C respectively. Note that our interest in abnormality comparison neither necessitates the use of the model cost $L(M)$, nor the entropy term. This is because the former is a fixed constant, and the latter is a cost associated with inherent information content rather than model fit. As a result, by excluding these terms, we are not measuring the *total* information content for a node, but rather the more desirable information content *with respect to the model*. Hence, we define our initial formulation δ_{base} as follows:

Definition 1 (Base)**.** *Given a single edge-attribute with model distribution C, the base abnormality scoring function δ_{base} for node $v \in \mathcal{V}$ is defined as*

$$\delta_{base}(v) = |f_v| \cdot KL(\hat{v} \,\|\, C)$$

where $|f_v|$ gives the cardinality of the edge-attribute value vector f_v produced from v's neighboring (outgoing) edges, $\hat{v}$ gives the discrete probability distribution associated with node v over the chosen attribute and C gives the global discrete probability distribution of the chosen attribute over all edges.

This formulation admits two especially desirable properties:

Observation 1. *Given two users U_1 and U_2 where $KL(\hat{U_1} \,\|\, C) = KL(\hat{U_2} \,\|\, C)$ and $KL(\hat{U_2} \,\|\, C) > 0$, if $|f_{u_1}| > |f_{u_2}| > 0$, then $\delta_{base}(U_1) > \delta_{base}(U_2)$.*

Observation 1 formalizes the intuition that given equal distributional deviation from the model, the user with more actions is more surprising.

Observation 2. *Given two users U_1 and U_2 such that $KL(\hat{U_1} \,\|\, C) > KL(\hat{U_2}) > 0$ and $|f_{u_1}| = |f_{u_2}|$ and $|f_{u_2}| > 0$, then $\delta_{base}(U_1) > \delta_{base}(U_2)$.*

Observation 2 formalizes the intuition that given an equal number of ratings, the user whose distribution is more unlike the model is more surprising.

Note that Definition 1 gives a base formulation δ_{base}, for the elementary case in which we have a relation with a single, global model distribution C for just a single edge-attribute. We next relax these assumptions and discuss how to extend this formulation to more complex scenarios. We first discuss extensions to scoring a multifaceted model in which we consider multiple model distributions for a single attribute, and next broach the topic of building a joint scoring function which can additionally incorporate multiple attributes. Finally, we touch upon expanding these definitions to a unified scoring scheme which can handle more complex, heterogeneous graph structures with multiple relation types. Our end goal is to devise a formulation of δ which accounts for all of these factors in ranking abnormality.

C. Handling multifaceted edge behavior

It is often the case that patterns in user behavior are more granular than singular, global trends – different users may rate products in different ways. One can consider that many such latent user (generally, node) behaviors may exist as a result of distinct preferences, response bias and a number of other factors. It can be useful to model these separately, as the single global distribution may actually be a mixture of behaviors of varying prevalence that nodes exhibit. For example, consider a global distribution of 50% ratings as 1-star and 50% as 5-star. This can actually be compromised of 3 latent user behaviors: only 1-star raters, only 5-star raters, and a small fraction of combined 1 and 5-star raters. The unified model will penalize the latter group the least as they perfectly match the global distribution, but in reality this could be the rarest group of users.

In fact, δ_{base} can be extended to incorporate such a multifaceted model without much complication. The base formulation assumes the existence of a single, global model C which describes the attribute distribution over all edges. To capture the notion of multiple models of attribute behavior, we introduce the notation $C_{u,j}$ and $C_{p,k}$ to denote the jth model distribution for user ratings and the kth model distribution for product ratings, where $j \in \{1...s\}$ and $k \in \{1...t\}$ given s user and t product rating distributions respectively. We can consider these as clusters which describe various modes of rating behavior. In addition to the cluster distributions, we also define their proportions $\rho_{u,j}$ and $\rho_{p,k}$ as the fraction of user and product nodes which belong to the jth and kth clusters respectively – we consider that a user U belongs to a cluster j if j yields minimum L^2 distance. The analagous definition applies to a product P and cluster k. Note that with the introduction of such a multifaceted model, our model distribution C is defined *separately* for user and product ratings – this is in contrast to the definition when we considered a single, global model. The distinguishing factor is that with multiple clusters, the patterns in how users rate and how products are rated can actually differ depending on G's edge structure.

In this case, we face the problem of identifying abnormality as a function of multiple clusters rather than just a single one. The abnormality of a node should also reflect to what extent its behavior fits with these various cluster distributions – for instance, even if there are two clusters of user rating behavior, if one cluster is more widespread and characteristic of general user rating behavior than the other, this factor should be intuitively accounted for in the scoring. To account for this concept, we introduce the following definition of the multifaceted abnormality scoring function δ_{mf}:

Definition 2 (Multifaceted)**.** *Given a single edge attribute and h cluster distributions of type $b \in \mathcal{B}$ indicated by $C_{b,g}$ where $g \in \{1...h\}$, the multifaceted abnormality scoring function δ_{mf} for a node $v \in \mathcal{V}$ with $\Psi(v) = b$ is defined as*

$$\delta_{mf}(v) = |f_v| \cdot \sum_{g=1}^{h} \big(\rho_{b,g} \cdot KL(\hat{v} \,\|\, C_{b,g})\big)$$

where $|f_v|$ gives the cardinality of the edge-attribute value vector f_v produced from v's neighboring (outgoing) edges, $\hat{v}$ gives the discrete probability distribution associated with node v over the chosen attribute, and $C_{b,g}$ and $\rho_{b,g}$ give the gth model distribution and proportion of the gth cluster respectively.

This scoring function intuitively gives the *expected* number of extra bits required to encode the behavior of v on a single edge attribute with respect to multiple cluster distributions. To see this, observe that δ_{mf} is in fact the expectation over random variable X with probability mass defined by the cluster proportions $\rho_{b,g}$ and outcomes defined by $\delta_{base}(v)$ and cluster distribution $C_{b,g}$. This extension to the base formulation admits yet another desirable property:

Observation 3. *Given two cluster distributions $C_{u,1}$ and $C_{u,2}$ with proportions such that $\rho_{u,1} > \rho_{u,2}$ and users U_1 and U_2 such that $\hat{U_1} = C_{u,1}$ and $\hat{U_2} = C_{u,2}$, if $KL(\hat{U_1} \,\|\, C_{u,2}) = KL(\hat{U_2} \,\|\, C_{u,1})$ and $KL(\hat{U_2} \,\|\, C_{u,1}) > 0$ and $|f_{u,1}| = |f_{u,2}|$ and $|f_{u,2}| > 0$, then $\delta_{mf}(U_1) < \delta_{mf}(U_2)$.*

Observation 3 formalizes the intuition that if two users have no deviation from their own cluster distributions and equal deviations from the other cluster's distribution, and otherwise give an equal number of ratings, then the user who belongs to the smaller cluster is more surprising.

Note that by incorporating multiple patterns of edge behavior in this way, the multifaceted model inherently allow for the possibility of capturing abnormal behavior as part of the model itself. In fact, we may find groups of users who form their own clusters based on abnormal rating patterns as a result of fraud or suspicious activity. However, by computing the expectation over clusters using the cluster proportions as probabilities, we can still robustly identify abnormal users assuming they make up a small fraction of all users, given that they will deviate substantially. The intuition is because although they may cost few bits with respect to their own cluster distribution, they will still cost many bits to store with respect to other cluster distributions with larger constituency.

D. Handling multiple edge-attributes

We now broach the topic of building a joint abnormality function which incorporates the presence of multiple edge attributes in addition to multifaceted models on each of the individual attributes. This is particularly useful in practical applications, where service providers collect a variety of information about each interaction. For example, in the user-rates-product scenario, practitioners may collect auxiliary information including timestamp, review text and purchase verification. For example, consider a user whose given rating distribution was not itself atypical, but had a consistent inter-arrival time (IAT) of 5 seconds between ratings – it is apparent in such a case that this reviewer's abnormality would not be well-indicated on the rating attribute, but would appear strongly on the temporal attribute.

There are a number of strategies we could employ for incorporating multiple attributes into the ranking context. One strategy is to consider ranking in a subspace formulation, where we consider abnormality with respect to various subspaces of edge attributes. However, this approach introduces an intractable number of subspaces along with sparsity issues, especially for high-dimensional data.

A second strategy is to consider abnormality additively over each of the attributes, assuming independence. In this approach, we compute the δ_{mf} score for each user over each attribute and simply sum the scores together. We find that this approach offers numerous comparative advantages over the previously mentioned joint subspace method. Firstly, instead of focusing on the combinatorial number of underlying subspaces, we focus on just a single space. This gives us a single abnormality ranking in which the top-ranking users are those who score highly in abnormality on many or all attributes. Furthermore, defining an additive measure of abnormality offers an attractive interpretation from the compression perspective – it is the expected number of extra bits to encode a node's actions with respect to independent edge attribute models.

We slightly modify our existing notation from the multifaceted (multiple clusters per attribute) model to distinguish cluster distributions between attributes $w \in \{1...y\}$ on a single relation. Now, instead of $C_{u,j}$ and $C_{p,k}$ to denote the jth cluster distribution for user ratings and kth cluster distribution for product ratings, we write $C_{u,w,j}$ and $C_{p,w,k}$ to denote the jth user cluster distribution and kth product cluster distribution for attribute w, respectively. Similarly, we write proportions as $\rho_{u,w,j}$ and $\rho_{p,w,k}$ for the proportion of the jth user cluster and kth product cluster for the wth attribute, respectively. Additionally, each attribute w may have a different number of user and product clusters so we write $j \in \{1...s_w\}$ and $k \in \{1...t_w\}$ where s_w and t_w denote the total number of user and product cluster distributions for the wth attribute, respectively. Thus, we define δ_{ma} as follows:

Definition 3 (Multi-attribute)**.** *Given multiple edge attributes $w \in \Omega(r)$ defined on a single relation $r \in \mathcal{R}$, with h_w cluster distributions of type $b \in \mathcal{B}$ respectively indicated by $C_{b,w,g}$ where $g \in \{1...h_w\}$, the multi-attribute abnormality scoring function δ_{ma} for node $v \in \mathcal{V}$ with $\Psi(v) = b$ is defined as*

$$\delta_{ma}(v) = |f_v| \cdot \sum_{w \in \Omega(r)} \Big(\sum_{g=1}^{h_w} \big(\rho_{b,w,g} \cdot KL(\hat{v_w} \,\|\, C_{b,w,g}) \big) \Big)$$

where $|f_v|$ gives the cardinality of the edge-attribute value vector f_v produced from v's neighboring (outgoing) edges, $\hat{v_w}$ gives the discrete probability distribution associated with node v over attribute w, and $C_{b,w,g}$ and $\rho_{b,w,g}$ give the gth model distribution and proportion of the gth cluster on the wth attribute respectively.

E. Handling multi-relational graphs

Thus far, we have built up δ_{ma} as an abnormality scoring function which handles multiple edge attributes with multifaceted models indicating various clusters of node behavior. Now, we briefly discuss how to extend this scoring function to more complex heterogeneous schemas with multiple relation types ($|\mathcal{R}| > 1$). Handling multiple relation types is yet another factor which can enable richer anomaly detection. For example, consider that in our running user-rates-product scenario, we additionally incorporate a new object type of seller, and introduce a new relation user-rates-seller. As motivation, consider a user who gives typical rating values for products but not for sellers. Thus, considering only the user-rates-product relation, we would not be able to identify a user as abnormal using the δ_{ma} score. However, incorporating the user-rates-seller relation, we are able to appropriately penalize the user's atypical behavior.

Fortunately, extending the formulation to handle multiple relations per object follows a very similar argument to the multi-attribute scenario where we consider handling multiple attributes per relation. We now define a joint model on the object type which incorporates multiple relations per object, and multiple attributes per relation. Given such a model, users who behave atypically on multiple types of interactions will be considered the most abnormal. We again propose an additive formulation with a minor modification to notation – given that a user may have rated a different number of products than sellers, we use the notation $f_{u,r}$ for user U's vector for relation r, and $|f_{u,r}|$ for the size of the attribute vector. Similarly, we write $f_{p,r}$ and $|f_{p,r}|$ for product P's vector and the associated size for relation type r. Then, we define the unified heteregeneous, multi-attribute and multifaceted abnormality scoring function δ as follows:

Definition 4 (Unified)**.** *Given multiple edge attributes $w \in \Omega(r)$ defined on multiple relations $r \in \mathcal{R}$, with h_w cluster distributions of type $b \in \mathcal{B}$ respectively indicated by $C_{b,r,w,g}$ where $g \in \{1...h_w\}$, the unified abnormality scoring function δ for a node $v \in \mathcal{V}$ with $\Psi(v) = b$ is defined as*

$$\delta(v) = \sum_{r \in R} \left(\sum_{w \in \Omega(r)} \Big(|f_{v,r}| \cdot \sum_{g=1}^{h_w} \big(\rho_{b,r,w,g} \cdot KL(\hat{v_w} \,\|\, C_{b,r,w,g}) \big) \Big) \right)$$

where $|f_{v,r}|$ gives the cardinality of the edge-attribute value vector $f_{v,r}$ produced from v's neighboring (outgoing) edges of type r. Formally, $f_{v,r} = \{e \in E \mid v \in m(e) \wedge \Psi(e) = r\}$. Furthermore, $\hat{v_w}$ gives the discrete probability distribution associated with v over attribute w, and $C_{b,r,w,g}$ and $\rho_{b,r,w,g}$ give the gth model distribution and proportion of the gth cluster on the rth relation type respectively.

Note that the definition of δ given in Definition 4 is the final formulation of the abnormality scoring function. From a compression perspective, it gives the expected number of extra bits required to encode a given node's edge-attribute vectors with respect to a model over multiple relations, multiple

attributes and multiple per-attribute clusters. The definition is general, and extends to various node types with various numbers of relations and attributes.

IV. Proposed Method: EdgeCentric

Thus far, we have built up both intuition and formalization for the use of δ as an abnormality score for nodes in edge-attributed graphs. We next describe the five key steps of our EdgeCentric algorithm, which draws the attention of a practitioner to the nodes with the most surprising behavior in the given network.

Step 1 – Aggregation: For each node-type in G, we aggregate the attribute values over the outgoing edges from each node for each associated relation-type. In our user-rates-products scenario, we have two node-types (users and products) connected by a relation with two attribute types: ratings (categorical) and timestamps (numerical). Since our relation is undirected, for each node we aggregate the attribute values for the adjacent edges, thereby collecting a vector of rating values for the user (product) as well as a vector of associated timestamps.

Step 2 – Discretization: Given the attribute types and ranges, we discretize the value space of each attribute in a principled manner. Categorical data is by definition discrete and thus does not need further processing. For numerical attributes, the discretization process requires more sophistication. We propose an adaptive binning approach as follows: if the maximum value of the attribute is an order of magnitude larger than the minimum, we space the bin markers logarithmically into d bins ($d=20$ in our experiments). Otherwise, we space the bins linearly. Logarithmic binning addresses issues associated with sparsity and scale insensitivity in large-ranged data. In smaller ranges, linear binning tends to be sufficient. For temporal data, instead of binning timestamps, we bin the inter-arrival times (IATs) instead to reflect time between actions.

Step 3 – Clustering: After binning the per-node attribute values and normalizing to construct the appropriate probability mass functions, we cluster the vectors describing the probability masses as a number of d-dimensional points. Though any clustering algorithm could be used for this purpose, we use X-means [26], as it automatically chooses the number of clusters in a principled manner by optimizing Bayesian Information Criterion (BIC). The centers of the resulting clusters are d-dimensional probability mass functions themselves, which we use as the cluster distributions. We can then compute cluster proportions by empirically assigning the input points to clusters by smallest L^2 distance.

Step 4 – Scoring: Given the cluster distributions across all attributes and node-types over the respective relations, we now compute the abnormality score $\delta(v)$ for each node $v \in \mathcal{V}$ according to Definition 4. For each node-type, and over each of the attributes on associated relations, we *additively* compute the abnormality score in terms of the expected cost in extra bits with respect to the attribute cluster distributions.

Step 5 – Ranking: Finally, we sort the scores for each node-type in a descending fashion and return the ranking with associated node indices to the practitioner. This effectively routes practitioner attention to the most abnormal nodes for each of the node-types in the graph (users, products, etc.), with respect to encoding cost over a joint model composed of independent edge-attribute models. This information can then be leveraged for further investigation.

Note that while EdgeCentric is motivated by the utilization of edge-attributes in interaction networks, it can also be applied in non-network settings given attribute distributions for each object in an object set are known.

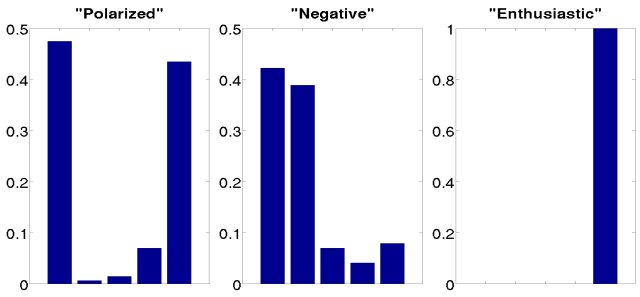


Fig. 2: **Discovered popular user-rating patterns.** Here, we show several cluster distributions and associated probability masses for user ratings on the `Flipkart` dataset – bins correspond to 1-5 stars.

V. Experimental Analysis

In this section, we evaluate EdgeCentric and aim to answer the following questions: what kinds of edge-attribute behavior do we observe in real graphs? Is EdgeCentric effective in finding abnormally-behaving nodes by leveraging this information? Finally, is EdgeCentric scalable?

A. Datasets

We apply EdgeCentric to 2 real-world graphs with various edge-attributes. The datasets are described below.

Flipkart: The `Flipkart` dataset contains information about reviews and ratings in the Flipkart e-commerce network which provides a platform for sellers to market products to customers. It contains roughly 3.3 million ratings given by 1.1 million users to 545 thousand products from Aug. 2011 to Jan. 2015.

Software Marketplace: The `SWM` dataset contains information about purchases in an online marketplace which allows customers to purchase software applications. The data for this marketplace was originally collected in [2]. It contains over 1.1 million ratings given by 964 thousand users to 15 thousand applications over the timespan of Apr. 2008 to June 2012.

B. Findings on `Flipkart`

In our analysis on the `Flipkart` dataset, we constructed a single relation (user-rates-product) on which we had one categorical attribute (rating from 1-5) and one temporal numerical attribute (UNIX timestamp). Thus, we ranked abnormality of users with respect to their rating and IAT behavior.

Figures 2 and 3 show the probability mass functions corresponding to the distributions that we found as a result of clustering the user edge-attribute data. Figure 2 shows several interesting rating patterns we discovered from the 17 total clusters produced from the X-means process: *polarized*, *negative* and *enthusiastic* users. *Polarized* users give mostly 1 star and 5 star ratings, with very few middle-ground ratings – this can correspond to the natural tendency to either love or hate a product, or result

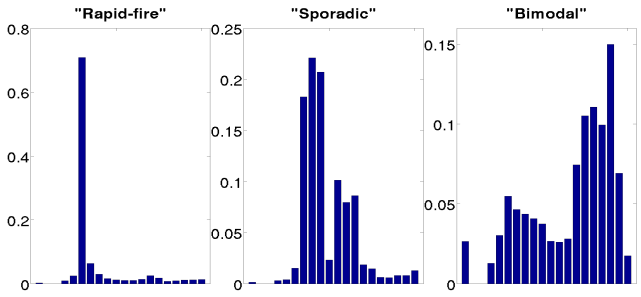


Fig. 3: **Discovered popular rating frequency (IAT) patterns.** Here, we show several cluster distributions and associated probability masses for interarrival times between ratings on the `Flipkart` dataset – bins correspond to logarithmically discretized interarrival times (the first 10 bins span IATs from several seconds to just a few minutes).

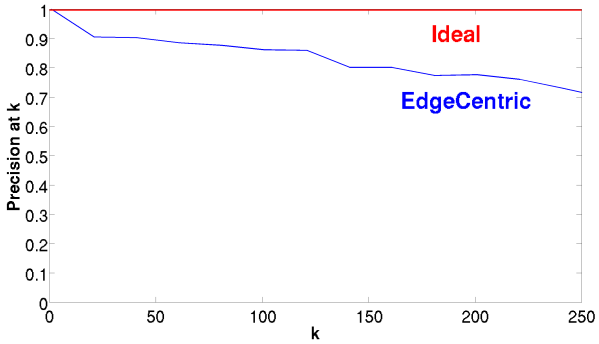


Fig. 4: **EDGECENTRIC finds fraudulent accounts on Flipkart with high precision.** Here, we show P@k for k ranging from 1 to 250, based on hand-labeled data from domain experts at Flipkart.

from fraudulent users who aim to popularize all the products of a single seller, and defame the competitors. *Negative* users give mostly 1 or 2 star ratings – we conjecture this is mostly a consequence of response bias, where users are sharing their opinions only because they are especially displeased with a product. Finally, *enthusiastic* users give only 5 star ratings and none others – this is suggestive of strong response bias or blatantly fraudulent behavior (especially when the user gives many such ratings). We additionally find isolated clusters for users who give only ratings of a single star outcome (1-5) – these *single-minded* are particularly prevalent in the data, given the large number of low-activity users who rated only one or a few products since inception. The presence of these behaviors in various proportions of the data then informs the computation of the abnormality scores and EDGECENTRIC rankings for individual users.

Figure 3 shows several IAT patterns (indicating rating frequency), selected from 17 total clusters produced from the X-means process: *rapid-fire*, *sporadic* and *bimodal* users. The bins are discretized logarithmically, so that the span of the first 10 bins corresponds to IATs between 0 seconds to roughly 10 minutes, whereas the latter 10 bins span from 10 minutes to several years (normally the case for users who rate only a few products in total, with a large gap between subsequent uses of the Flipkart platform). *Rapid-fire* users are the most blatantly suspicious – these users almost exclusively give ratings with just a few seconds between subsequent ones. This type of behavior is almost guaranteed to be fraudulent and does not correspond with any intuition of real human behavior. Conversely, *sporadic* users' behavior is far more in-line with human intuition. These users mostly give ratings several weeks to months apart. Very few ratings are given with shorter IATs, indicating that the users mostly rate single items upon purchase, and purchase only sporadically (grocery products, birthday presents, holiday gifts, etc.) Lastly, *bimodal* users behave bimodally, in that they occasionally spend weeks to months without rating a product, but often have periods of frequent activity on the order of multiple ratings (purchases) in days to weeks. Notice that the probability mass for the users in this cluster is distributed across almost all orders of IAT, with most of the mass concentrated in the days to weeks range, suggesting that the users are engaged with the Flipkart service and give ratings frequently (presumably because they also purchase products frequently). However, a non-trivial amount of the mass is distributed between shorter timeframes of seconds to minutes, indicating that the users rate multiple products in a single sitting (likely due to the purchase and resulting receipt of several products at the same time).

Upon applying EDGECENTRIC to this dataset, we provided a list of the 250 most abnormal accounts to domain-experts at Flipkart who investigated and labeled these users individually according to various criteria involving the user's review-text, rating distributions and frequencies. Figure 4 shows the precision at k (P@k) for a spread of k values over this range of 250 users, indicating positive results of 0.9 precision over the top 50 users, and over 0.7 precision over the top 250 users – recall results are incalculable given unbounded false negatives and lack of ground-truth labels. These are substantial findings for Flipkart – given previously unsophisticated fraud detection approaches, most fraudsters did not have to resort to distributed attacks (the fraudulent users had each committed hundreds to thousands of actions). One common pattern found by domain-experts was that most fraudsters either spammed 4/-5 star ratings to multiple products from a single seller (boosting seller ratings), or spamming 1/2-star ratings to products from another seller (defaming competition). We further found that the most abnormal user had given 3692 5-star ratings with an average IAT of just a few seconds.

C. Findings on `SWM`

On the `SWM` dataset, we constructed a single relation (user-rates-application) on which we had one categorical attribute (rating from 1-5). Thus, we ranked abnormality of users with respect to their rating behavior. We do not show the clustered rating behavior in interest of space, but note that similar behaviors can be observed in this dataset in terms of polarized raters, "single-minded" raters, etc. as in Figure 2.

We find that the users with the highest scores according to our EDGECENTRIC approach have spammy behavior. The most abnormal user in this dataset had given 186 5-star ratings to a single application. The accompanying reviews had very high textual similarity and included quotes like

- `"Awesome!!!,Get this app now and earn points for a $10 gift card."`
- `"Awesome App!!!! FREE money ,The app is great to earn points for FREE money. Get it today!"`

In fact, the top-ranked 20 users according to EDGECENTRIC often posted repetitive, spammy text in addition to highly skewed ratings. Usually, the review text promoted the

application, included personalized codes which the reviewers claimed would give customers free money/points, or were generally characteristic of information-free content. We additionally found correspondences between the codes reviewers asked customers to use and the reviewer's own usernames, suggesting that the code gave the reviewer an associated perk rather than the customer. It stands to reason that the associated applications incentivized existing customers to attract more potentials. Unfortunately, we are unable to check for ground-truth with service providers.

D. Scalability

The time-complexity of EDGECENTRIC is roughly $O(|\mathcal{E}|d + |\mathcal{V}|log|\mathcal{V}| + |\mathcal{V}|kdi)$ for a single attribute, where $|\mathcal{V}|$ and $|\mathcal{E}|$ are the node and edge count, d is attribute dimensionality, k is the cluster count over i clustering iterations. The terms reflect binning costs and kd-tree and clustering costs via X-means.

VI. CONCLUSION

In this work, we broach the issue of detecting anomalies in large, edge-attributed real-world graphs, which are commonplace in modern e-commerce platforms, social networks and other web services. Specifically, we first formalize the problem of detecting anomalous nodes in graphs as an unsupervised ranking problem, in which we aim to score nodes based on the abnormality of their edge behavior. To this end, we first build up the intuition of using information theoretic principles to quantify deviation from typical behavior in a data-driven fashion, and extend this formulation in the presence of multiple user behaviors, multiple edge-attributes and complex heterogeneous graphs. We then introduce the EDGECENTRIC approach to leverage this formulation. Finally, we show substantiating results including high precision (0.87 over the top 100 users) on the Flipkart e-commerce platform, practical scalability and interesting observations on atypical user behavior gleaned from applying our method to several large, real-world networks.

REFERENCES

[1] C. C. Aggarwal, H. Wang, et al. *Managing and mining graph data*, volume 40. Springer, 2010.

[2] L. Akoglu, R. Chandy, and C. Faloutsos. Opinion fraud detection in online reviews by network effects. *ICWSM*, 2013.

[3] L. Akoglu, M. McGlohon, and C. Faloutsos. Oddball: Spotting anomalies in weighted graphs. In *Advances in Knowledge Discovery and Data Mining*. Springer, 2010.

[4] L. Akoglu, H. Tong, B. Meeder, and C. Faloutsos. Pics: Parameter-free identification of cohesive subgroups in large attributed graphs. In *SDM*. SIAM, 2012.

[5] A. Beutel, K. Murray, C. Faloutsos, and A. J. Smola. CoBaFi: collaborative bayesian filtering. In *WWW*. ACM, 2014.

[6] A. Beutel, W. Xu, V. Guruswami, C. Palow, and C. Faloutsos. Copycatch: stopping group attacks by spotting lockstep behavior in social networks. In *WWW*. ACM, 2013.

[7] B. Boden, S. Günnemann, H. Hoffmann, and T. Seidl. Mining coherent subgraphs in multi-layer graphs with edge labels. In *KDD*. ACM, 2012.

[8] B. Boden, S. Günnemann, H. Hoffmann, and T. Seidl. Rmics: a robust approach for mining coherent subgraphs in edge-labeled multi-layer graphs. In *SSDBM*. ACM, 2013.

[9] D. Chakrabarti, S. Papadimitriou, D. S. Modha, and C. Faloutsos. Fully automatic cross-associations. In *KDD*. ACM, 2004.

[10] P. O. V. De Melo, L. Akoglu, C. Faloutsos, and A. A. Loureiro. Surprising patterns for the call duration distribution of mobile phone users. In *Machine learning and knowledge discovery in databases*. Springer, 2010.

[11] J. Gao, F. Liang, W. Fan, C. Wang, Y. Sun, and J. Han. On community outliers and their efficient detection in information networks. In *KDD*. ACM, 2010.

[12] S. Ghosh, B. Viswanath, F. Kooti, N. K. Sharma, G. Korlam, F. Benevenuto, N. Ganguly, and K. P. Gummadi. Understanding and combating link farming in the twitter social network. In *WWW*. ACM, 2012.

[13] S. Günnemann, I. Farber, B. Boden, and T. Seidl. Subspace clustering meets dense subgraph mining: A synthesis of two paradigms. In *ICDM*. IEEE, 2010.

[14] B. Hooi, N. Shah, A. Beutel, S. Gunnemann, L. Akoglu, M. Kumar, D. Makhija, and C. Faloutsos. Birdnest: Bayesian inference for ratings-fraud detection. 2016.

[15] M. Hu and B. Liu. Mining and summarizing customer reviews. In *KDD*. ACM, 2004.

[16] M. Jiang, A. Beutel, P. Cui, B. Hooi, S. Yang, and C. Faloutsos. A general suspiciousness metric for dense blocks in multimodal data. In *ICDM*. IEEE, 2015.

[17] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang. Catchsync: catching synchronized behavior in large directed graphs. In *KDD*. ACM, 2014.

[18] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang. Inferring strange behavior from connectivity pattern in social networks. In *Advances in Knowledge Discovery and Data Mining*. Springer, 2014.

[19] N. Jindal and B. Liu. Opinion spam and analysis. In *WSDM*. ACM, 2008.

[20] G. Karypis and V. Kumar. Metis-unstructured graph partitioning and sparse matrix ordering system. 1995.

[21] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8), 2009.

[22] D. Koutra, U. Kang, J. Vreeken, and C. Faloutsos. Vog: Summarizing and understanding large graphs. In *SDM*. SIAM, 2014.

[23] B. Long, Z. M. Zhang, X. Wu, and P. S. Yu. Spectral clustering for multi-type relational data. In *ICML*. ACM, 2006.

[24] C. C. Noble and D. J. Cook. Graph-based anomaly detection. In *KDD*. ACM, 2003.

[25] 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*. ACM, 2007.

[26] D. Pelleg, A. W. Moore, et al. X-means: Extending k-means with efficient estimation of the number of clusters. In *ICML*, 2000.

[27] B. Perozzi, L. Akoglu, P. Iglesias Sánchez, and E. Müller. Focused clustering and outlier detection in large attributed graphs. In *KDD*. ACM, 2014.

[28] B. A. Prakash, A. Sridharan, M. Seshadri, S. Machiraju, and C. Faloutsos. Eigenspokes: Surprising patterns and scalable community chipping in large graphs. In *Advances in Knowledge Discovery and Data Mining*. Springer, 2010.

[29] N. Shah, A. Beutel, B. Gallagher, and C. Faloutsos. Spotting suspicious link behavior with fbox: an adversarial perspective. In *ICDM*. IEEE, 2014.

[30] N. Shah, D. Koutra, T. Zou, B. Gallagher, and C. Faloutsos. Timecrunch: Interpretable dynamic graph summarization. In *KDD*. ACM, 2015.

[31] H. Tong and C.-Y. Lin. Non-negative residual matrix factorization with application to graph anomaly detection. In *SDM*. SIAM, 2011.

[32] S. White and P. Smyth. A spectral clustering approach to finding communities in graph. In *SDM*. SIAM, 2005.

[33] J. Ye and L. Akoglu. Discovering opinion spammer groups by network footprints. In *Machine Learning and Knowledge Discovery in Databases*. Springer, 2015.

[34] Y. Zhou, H. Cheng, and J. X. Yu. Graph clustering based on structural/attribute similarities. *VLDB*, 2009.