Efficient Anomaly Detection in Dynamic, Attributed Graphs

Emerging Phenomena and Big Data

Benjamin A. Miller and Nicholas Arcolano
Lincoln Laboratory
Massachusetts Institute of Technology
Lexington, MA 02420
Email: {bamiller, arcolano}@ll.mit.edu

Nadya T. Bliss
ASURE
Scottsdale, AZ 85257
Email: nadya.bliss@asu.edu

***Abstract*—When working with large-scale network data, the interconnected entities often have additional descriptive information. This additional metadata may provide insight that can be exploited for detection of anomalous events. In this paper, we use a generalized linear model for random attributed graphs to model connection probabilities using vertex metadata. For a class of such models, we show that an approximation to the exact model yields an exploitable structure in the edge probabilities, allowing for efficient scaling of a spectral framework for anomaly detection through analysis of graph residuals, and a fast and simple procedure for estimating the model parameters. In simulation, we demonstrate that taking into account both attributes and dynamics in this analysis has a much more significant impact on the detection of an emerging anomaly than accounting for either dynamics or attributes alone. We also present an analysis of a large, dynamic citation graph, demonstrating that taking additional document metadata into account emphasizes parts of the graph that would not be considered significant otherwise.**

***Index Terms*—Subgraph detection, network modularity, signal detection theory, attributed graph modeling, generalized linear models**

I. Introduction

In numerous big data applications, relationships between entities are of interest. Connections between computers may be analyzed for a computer security application [1], for example, and large social networks are frequently analyzed to find communities and influential figures [2], [3]. In these applications, while the entities themselves may be of interest, it is their connections and relationships that provide real insight and situational awareness.

When working with relational data, a graph is a natural mathematical structure for data representation. A graph $G = (V, E)$ is a pair of sets: a set of vertices V representing the entities, and a set of edges E denoting their relationships or connections. Traditionally, graphs consist of only vertices and their connections, possibly with weights on the edges. In many application domains, relational datasets also contain contextual information on both edges and vertices. We receive raw collections not as a graph, but as a (potentially multi-source) set of heterogeneous data points. When a graph is constructed, a large fraction of the data is not considered. If these usually-ignored portions of the data are retained, they can be useful for many purposes. In particular, these metadata provide extra dimensions to the graph that can help determine which observed edges are likely to occur, and which are out of the ordinary. This enables the uncued detection of anomalies in the graph, which is the primary objective of many applications.

In earlier work, we developed a framework for uncued anomaly detection in large, dynamic graphs [4]. Detection of anomalous subgraphs is an important capability across a variety of applications, such as detection of malicious software in a computer network, or the detection of threat actors in a communication network [5]. This framework treats a graph as an instance drawn from a distribution of random graphs, and performs spectral analysis of graph residuals (i.e., the difference between the observed graph and its expected value) to determine the presence of anomalies. This analysis uses a given expected degree model, which has a low-rank expected value structure, to facilitate efficient residuals analysis. The parameters of this model have a simple, closed-form estimator that is approximately asymptotically unbiased, as shown in [6]. The residuals were analyzed over time, with a filter applied to emphasize certain behaviors. Steps toward optimizing such filters were later presented in [7].

In this paper, we demonstrate a new technique for detection of anomalies in large, attributed graphs. Leveraging recent work in subgraph detection, we show that this technique enables the detection of emergent activity that would not be detectable in a static graph, or one without attributes. This technique is based on a generalized linear model (GLM) framework for data modeling. Since this model has some properties that would make it intractable for real-time analysis

This work is sponsored by the Intelligence Advanced Research Projects Activity (IARPA) under Air Force Contract FA8721-05-C-0002. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

on large datasets, we provide an approximation that enables fast parameter estimation and efficient analysis of attributed graphs at large scales.

The remainder of this paper is organized as follows. In Section II, we outline basic graph properties, and the attributes considered in our analysis. Section III presents the anomaly detection problem model and our temporal integration technique. In Section IV, we present the analysis of an efficient method for incorporating the GLM into the subgraph detection framework, which allows for an eigendecomposition of the residuals matrix to be performed in linear time for a constant number of categorical attributes. Section V presents results of this technique in simulation, demonstrating the power of using both dynamics and attributes in the analysis of graph residuals, and in Section VI we present findings when this technique is applied to a large document corpus. In Section VII, we summarize and outline future research directions.

II. Graphs and Attributes

As mentioned previously, a graph is a pair of sets $G = (V, E)$. A graph may be either directed, where an edge goes from one vertex to another, or undirected, where the two vertices are connected with no notion of ordering. In the former case, $E \subset V \times V$, where ordering denotes the direction, whereas in the latter case E consists of 2-vertex subsets of V. For the purpose of this paper, we will consider only unweighted graphs, though the analysis can be extended to graphs with weights. For convenience, let $N = |V|$ and $M = |E|$. An important notion in our analysis is that of vertex degree. The degree of a vertex is the number of other vertices with which it shares an edge, i.e., the degree of $v \in V$ is $|\{u|\{u, v\} \in E\}|$ for undirected graphs, and in a directed setting, v has in- and out-degrees of $|\{u|(u, v) \in E\}|$ and $|\{u|(v, u \in E\}|$, respectively.

In our detection framework, we make use of matrix representations of a graph. The adjacency matrix A of the graph is a matrix in which the entry in row i and column j is nonzero only if there is an edge from vertex i to vertex j. (This requires an arbitrary labeling of the vertices with integers from 1 to N.) The value of an entry will be the edge weight in the case of a weighted graph, or 1 for an unweighted graph. Also, if the graph is undirected, the adjacency matrix will be symmetric. Finally, let $k \in \mathbb{Z}^N$ denote the vector of observed degrees, with the ith component k_i being the degree of vertex i. For directed graphs, there will be two vectors, k_{in} and k_{out}, for in- and out-degrees.

We will consider graph attributes that can be divided into the following cases:

- Vertex attributes: properties of the entities, which may be real-valued or categorical;
- Edge attributes: properties of the relationships, which may be real-valued or categorical;
- Vertex pair attributes: regardless of the presence of an edge, an attribute regarding a pair of vertices.

Within this context, edge attributes are typically part of the observed relationships, e.g., duration of a connection in a computer network or amount of money transferred in a financial graph. Thus, real-valued attributes can be expressed as edge weights, and categorical attributes can be encoded through the use of multi-graphs, i.e., graphs with multiple edges allowed between one pair of vertices, in this case denoting different kinds of relationships.

With this in mind, we will limit the scope of attributes we consider to those related to vertices and pairs of vertices. That is, the attributes of interest will be those that manifest themselves as parameters to the model (intrinsic vertex properties) rather than a specific network instance (observed relationships). For the purpose of the approximation introduced in Section IV, we will also only consider vertex pair attributes that are categorical. While this ignores some potentially interesting behavior—such as cases where past observations impact the probability of future ones—it enables a convenient, exploitable model that enables efficient residuals analysis in a broad space of random graphs.

III. Problem Model

We pose the subgraph detection problem as one of traditional signal detection theory, as first discussed in [8]. The objective, given an observed graph over time, $G(n)$, is to resolve the following binary hypothesis test:

$$\begin{cases} H_0: & \text{The observed graph is "noise" } G_B \\ H_1: & \text{The observed graph is "signal+noise" } G_B \cup G_S. \end{cases}$$

Here, $G_B = (V, E)$ is drawn from a distribution of random graphs, and $G_S = (V_S, E_S)$ is a small subgraph ($N_S = |V_S| \ll N$) that is embedded into the background. This subgraph will be unlikely to occur under the background distribution, and, therefore, is an anomaly and will serve as the signal. Only cases where the subgraph is embedded on vertices already existing in the background, i.e., $V_S \subset V$ and $G_B \cup G_S = (V, E \cup E_S)$, are considered.

The present work is focused on graphs with attributes on the vertices, and between pairs of vertices. For this purpose, we will assume that the metadata are not affected by the embedding procedure. That is, when embedding a subgraph into a background graph G_B, which has fixed attributes, the vertices of selected to comprise G_S will maintain their attribute values, and only the topology will change.

We resolve the hypothesis test through analysis of graph residuals. Filter coefficients h are given to integrate the residuals over a time window. An approximation of $\mathbb{E}[A(n)]$ is derived from the observed dynamic graph, and we consider the eigendecomposition of the integrated residuals matrix,

$$U \Lambda U^T = \sum_{\ell=0}^{L-1} h_\ell \left(A(n-\ell) - \mathbb{E}\left[A(n-\ell)\right] \right),$$

where L is the length of the time window. By considering only a few eigenvectors, we can reduce the dimensionality of the problem, and efficiently determine the presence of an anomalous subgraph.

As in [7], we will consider a known signal model, where the subgraph behavior of interest is known, but its position within the background is not. To compute these coefficients, a 3-way tensor of the subgraph adjacency matrix $A_S(n)$ is formed, where two dimensions correspond to vertices and one dimension corresponds to time, and a rank-1 approximation for the tensor is computed, as described in [9]. The factor of this approximation along the temporal dimension is used for the filter coefficients. As noted in [7], this will maximize the spectral norm of the integrated subgraph

$$\sum_{\ell=0}^{L-1} A_S(n-\ell)h_\ell,$$

thus improving its detectability in the eigenspace of the residuals matrix. In the subsequent sections, we will demonstrate that, while temporal integration provides a significant benefit in terms of subgraph detection ability, knowledge of vertex attributes allows for detection of much subtler anomalies.

IV. Residuals Analysis in Attributed Graphs

To model a random graph without edge weights, each possible edge will be modeled as a Bernoulli random variable. Let the probability of an edge occurring between vertices i and j be p_{ij}. For an attributed graph, we use a logistic regression model based on vertex and edge attributes. Let β_i be a vector of real-valued attributes for vertex i, and β_{ij} be the attributes of vertex pair (i,j). Using a logistic regression framework, we can use the same principle as linear regression to map a linear combination of attributes to edge probabilities, i.e.,

$$p_{ij} = \frac{1}{1+\exp\left(-x_{\text{out}}^T\beta_i - x_{\text{in}}^T\beta_j - x_{\text{pair}}^T\beta_{ij}\right)}. \tag{1}$$

Here, x_{out}, x_{in} and x_{pair} are the attribute weights for source vertices, destination vertices, and pairs of vertices, respectively. Note that, for undirected graphs, $x_{\text{out}} = x_{\text{in}}$. This sort of model has been used in link prediction [10], to quantify the impact of attributes on whether or not an edge occurs.

While this generalized linear model allows a direct mapping of attributes to probabilities, this comes at the expense of a relatively expensive fitting procedure. A maximum likelihood technique can be used to estimate the coefficients in x_{in}, x_{out} and x_{pair}, but this algorithm will cost $O(N^2)$ time per iteration, which is not tractable for large graphs. Also, to enable scaling to very large graphs, which are typically sparse, it is beneficial to have an exploitable structure in the expected value matrix that enables fast calculation of residuals without storing a large, dense matrix. For example, when using Newman's modularity matrix, as described in [11], the residuals matrix is the sum of a sparse matrix and a rank-1 matrix, which allows efficient computation of eigenvalues. No such exploitable structure exists for the GLM, meaning that the residuals matrix may be a dense matrix with no special structure, which will require $O(N^2)$ time and space simply for storage.

Since we are typically interested in sparse graphs, we use a log-linear model rather than a logistic-linear model for edge probabilities. That is, we allow the approximation

$$p_{ij} \approx \exp\left(x_{\text{out}}^T\beta_i + x_{\text{in}}^T\beta_j + x_{\text{pair}}^T\beta_{ij}\right),$$

since for small values of x, $e^x \approx (1+e^{-x})^{-1}$. The log-linear model is frequently used as a generalized linear model when a distribution can have an arbitrary positive expected value, such as a Poisson distribution. For small expected values, a Poisson distribution approaches a Bernoulli distribution, so this model is a good approximation for sparse graphs where any individual edge is fairly unlikely. As we demonstrate in the remainder of this section, this approximation has properties that can be exploited for computational efficiency.

In the case of categorical vertex pair attributes, i.e., cases in which the vertex pair attributes are dependent on the categories of the associated vertices, the rank of the expected value matrix will be equal to the number of categories. Let C be the number of categories, I_c be the indices of vertices in category c, and $I(i)$ be the indices of all vertices in the same category as vertex i. The probability matrix $P = \{p_{ij}\}$ has the form

$$P = \operatorname{diag}(\alpha_{\text{out}})\,\Theta X \Theta^T \operatorname{diag}(\alpha_{\text{in}}). \tag{2}$$

Here, Θ is an $N \times C$ assignment matrix, where each row contains a single entry of 1, with all other entries being 0. The position of the nonzero entry corresponds to the category of the vertex, i.e., if vertex i is in category c, the ith row of Θ will have a 1 only in column c. The matrix $X = \{x_{ij}\}$ consists of the terms from (1) dependent upon the categories of the vertex pairs, meaning that $x_{ij} = x_{\text{pair}}^T\beta_{I_iI_j}$, and the vectors α_{out} and α_{in} are defined such that $\alpha_{\text{out}}(i) = x_{\text{out}}^T\beta_i$ and $\alpha_{\text{in}}(i) = x_{\text{in}}^T\beta_i$. This matrix will have rank C.

The relatively low rank of the probability matrix allows for fast computation of the principal eigenvectors of $A-P$. As discussed in [4], matrix-vector multiplications are at the heart of algorithms for computing extreme eigenvectors and eigenvalues of matrices. With a probability matrix in the form of (2), eigenvalues of the residuals matrix can be computed without computing the full probability matrix, and the running time will scale tractably for large graphs. To compute the product $(A-P)z$ for an arbitrary vector $z \in \mathbb{R}^N$, we first compute Az, which is a sparse matrix-vector multiplication taking $O(M)$ time. Computing Pz requires two entry-wise scalings (by α_{in} and α_{out}, $O(N)$ time each), two multiplications by the sparse Θ matrices (also $O(N)$ time), and multiplication of a vector by X ($O(C^2)$ time). Finally, Pz is subtracted from Az, costing $O(N)$ time, yielding a total running time for the matrix-vector multiplication of $O(M+N+C^2)$. Using the implicitly-restarted Lanczos algorithm to compute m eigenvectors of the residuals matrix, the matrix-vector multiplication running time implies a total time complexity of $O((M+C^2)m+Nm^2+m^3)$ per restart, where the number of restarts depends on the relative distance between consecutive eigenvalues. Thus, to compute a fixed number of eigenvectors, if the number of categories remains fixed, the per-iteration cost of the algorithm will scale linearly in the number of edges.

In addition to efficient computation of the eigenspace, this model enables a fast parameter approximation scheme based

on moment matching. Consider the sum over the entries of the observed adjacency matrix corresponding to edges going from category a to category b. In expectation, we have

$$\mathbb{E}\left[\sum_{i\in I_a}\sum_{j\in I_b} a_{ij}\right] = \sum_{i\in I_a}\sum_{j\in I_b} \alpha_{\text{out}}(i)\alpha_{\text{in}}(j)X_{I_aI_b} = \|\alpha_{\text{out}}(I_a)\|_1\|\alpha_{\text{in}}(I_b)\|_1 X_{I_aI_b}.$$

Consider also the expected out-degree of vertex i. We have

$$\mathbb{E}\left[\sum_{j=1}^{N} a_{ij}\right] = \sum_{j=1}^{N} \alpha_{\text{out}}(i)\alpha_{\text{in}}(j)X_{I(i)I(j)} = \alpha_{\text{out}}(i)\sum_{c=1}^{C}\|\alpha_{\text{in}}(I_c)\|_1 X_{I(i)I_c}$$

Similarly, the expected in-degree is given by $\alpha_{\text{in}}(i)\sum_{c=1}^{C}\|\alpha_{\text{out}}(I_c)\|_1 X_{I_cI(i)}$. By assuming that each block of α_{in} and α_{out} corresponding to a distinct category has unit L_1 norm, we can estimate the probability matrix by letting the estimate of X be a matrix of the observed volumes of the portions of the graph corresponding to different category pairs, i.e., $\widehat{X} = \Theta^T A\Theta$, and the estimates of α_{in} and α_{out} be the observed in- and out-degree vectors, normalized in each category block to have an L_1 norm of 1. Thus, we have estimates

$$\widehat{\alpha_{\text{in}}}(i) = \frac{k_{\text{in}}(i)}{\sum_{j\in I(i)} k_{\text{in}}(j)} \quad \text{and} \quad \widehat{\alpha_{\text{out}}}(i) = \frac{k_{\text{out}}(i)}{\sum_{j\in I(i)} k_{\text{out}}(j)}.$$

The restriction of the α vectors having unit L_1 norm does not disallow any probability matrices of the form (2). Indeed, for arbitrary vectors α_{out} and α_{in}, the same P can be computed with block-normalized vectors using a new X matrix, $\widetilde{X} = \{\tilde{x}_{ij}\}$, such that

$$\tilde{x}_{ij} = x_{ij}\|\alpha_{\text{out}}(I_i)\|_1\|\alpha_{\text{out}}(I_j)\|_1.$$

Another interesting feature of this approximation is that, like the modularity matrix of [11], it creates a residuals matrix where the vector of all 1s, denoted $\mathbf{1}$, is in the nullspace. Multiplying the adjacency matrix on the right by $\mathbf{1}$, we get the vector of out degrees. The ith row of the estimated probability matrix times $\mathbf{1}$ will yield

$$\frac{k_{\text{out}}(i)}{\|k_{\text{out}}(I(i))\|_1}\sum_{j=1}^{N}\frac{k_{\text{in}}(j)}{\|k_{\text{in}}(I(j))\|_1}\widehat{X}_{I(i)I(j)} = \frac{k_{\text{out}}(i)}{\|k_{\text{out}}(I(i))\|_1}\sum_{j=1}^{C}\widehat{X}_{I(i)I_j} = k_{\text{out}}(i).$$

This holds similarly, yielding k_{in}^T, when multiplying by $\mathbf{1}^T$ on the left. Thus, any eigenvectors (for undirected graphs) or singular vectors (for directed) that are not in the nullspace will be orthogonal to $\mathbf{1}$, and the projection into the principal singular vectors will be centered at the origin.

V. Simulation Results

To demonstrate the impact of this additional information on subgraph detection, a Monte Carlo simulation was run, generating dynamic random graphs from a model based on (2). The graphs generated are undirected, with 3 categories, and 1000 vertices in each category. The α vector is created according to a power law, since many large, real-world graphs have powerlaw distributions. The categorical parameters create a homophily effect, causing higher probability of connectivity between vertices in the same category, similarly to the model for interaction rates proposed in [12]. In particular, we use

$$X = \begin{bmatrix} 10000 & 3500 & 4000 \\ 3500 & 10200 & 3100 \\ 4000 & 3100 & 9900 \end{bmatrix},$$

making connection within a category about 2.5–3 times as likely as between categories. The α vector was block-unit-normalized, so the values in X correspond to the expected volumes within the corresponding parts of the graph.

To incorporate dynamics, we generate 8 samples from this distribution independently. This is used for the H_0 case. For the H_1 case, 5 vertices are chosen uniformly at random from each category, and these 15 vertices comprise the signal subgraph. The signal starts with no edges, then adds edges at a constant rate over the course of the first 7 time steps, until it reaches a prescribed density. At the last time step, edges are removed, so that the subgraph has only 30% of the edges it had in the previous sample. This could represent an increase in communication leading up to an event, followed by dispersion to avoid detection. The filter coefficients are optimized using the Matlab Tensor Toolbox[1]. After integration according to this filter, a chi-squared test for independence is performed in the principal 2-dimensional subspace, as described in [8]. The result of this test is used as our detection statistic.

Four separate scenarios were run. In one case, neither dynamics nor attributes were considered. In this scenario, we take the graph at time step $n = 7$, i.e., the point in time where the subgraph is the densest. The residuals matrix used is the modularity matrix of [11], i.e., $A - (kk^T)/\|k\|_1$. Thus, the categories of the vertices are not considered. In the second case, we use dynamics, but not attributes, estimating k without considering categories, and integrating $(A(n) - (kk^T)/\|k\|_1)$ using the given filter coefficients. Third, we consider attributes, but not dynamics. Again, only time step $n = 7$ is used, but P is estimated using the method described in Section IV. Finally, we use both dynamics and attributes. In this case, P is estimated using measurements at all time steps, and we integrate the residuals over time using the filter.

Detection performance in a 2000-trial simulation is presented in Figure 1. When neither dynamics nor attributes are considered (top left), even when the subgraph is 100% dense, detection performance is fairly mediocre. For densities below 100%, performance is not much better than chance. When dynamics, but not attributes, are taken into account (top right), we achieve perfect detection performance when the subgraph grows to 90% density, with performance decreasing

[1] Available online at http://www.sandia.gov/∼tgkolda/TensorToolbox/.

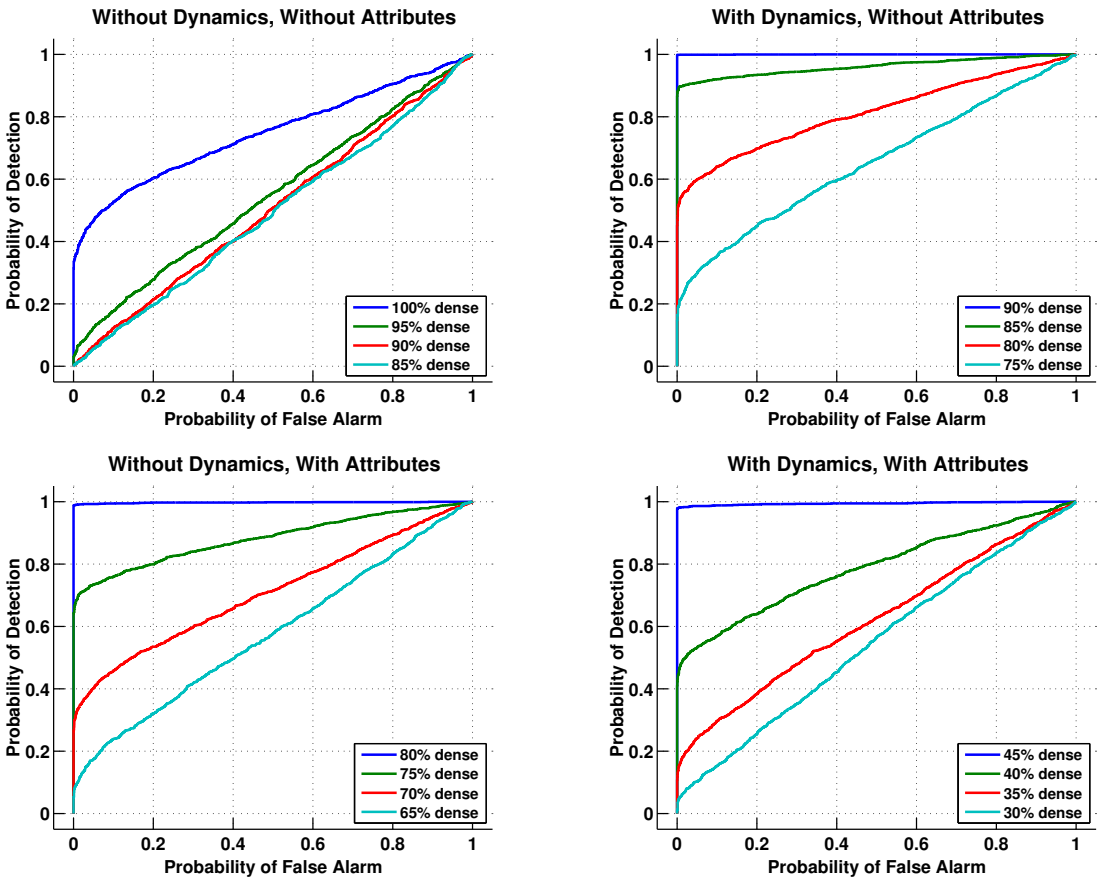

Figure 1. Detection performance in simulated graphs. Simulations were run in which the detection algorithm either includes (right column) or does not include (left column) dynamics, and includes (bottom row) or does not include (top row) vertex attributes. Densities listed are the greatest density of the signal subgraph over an 8-sample window. Detection performance increases as the subgraph gets denser, and as more features of the graph are considered.

as the density is reduced. With attributes and not dynamics (bottom left), even weaker subgraphs, with density of only 80%, are detected with near-perfect accuracy. Taking into account both attributes and dynamics (bottom right), however, provides a substantial benefit well beyond using either feature individually. We see near-perfect detection when the subgraph grows to only 45% density, which is undetectable in any of the other cases, demonstrating that accounting for dynamics and attributes has the potential to drastically increase performance.

VI. Data Analysis

Using the commercially available Thomson Reuters Web of Science® (WoS) database [13], we built a directed, dynamic graph based on document citations. This dataset is comprised of records, compiled for research purposes, representing scholarly publications of the international scientific community, published between 1900 and present in public commercial and open source journals and conference proceedings. Each record represents an individual document, and fields include document title and type, journal name, author names and institutional affiliations (as provided in publication), cited references, and publication date.

In the citation graph, each vertex represents a document, with a directed edge going from vertex i to vertex j if document i cites document j. We consider a citation graph for each year, since this is the most reliable time resolution. For each year, we fit the model in (2) to the graph at that time slice, using the procedure outlined in Section IV, with the vertex category determined by the "subject" field in the database. Over the years from 1937 to 1986, the yearly graph increases from about 1500 documents and 80,000 citations per year to nearly 500,000 documents and 7.8 million citations per year. At each year, after fitting the graph to the model, we integrate the residuals over a 6-year window, using a linear ramp filter to emphasize emergent behavior. The top 30 singular values are computed, which are shown in the lefthand plot in Figure 2. In 1976, there is a substantial uptick in several of the singular values, so we will investigate this period of time in detail.

Looking at the window from 1971 to 1976, the second and third right singular vectors (corresponding to cited documents) are shown in the scatterplot on the right hand side of Figure 2. (The first singular vector is dominated by a single, high-degree vertex, seen in the lower left in the plot.) The highlighted outliers toward the top of the plot all correspond to analytical chemistry papers written in the 1950s and '60s [14]–[18], all of which have accumulated thousands of citations over the years. Each of these documents has somewhat high degree

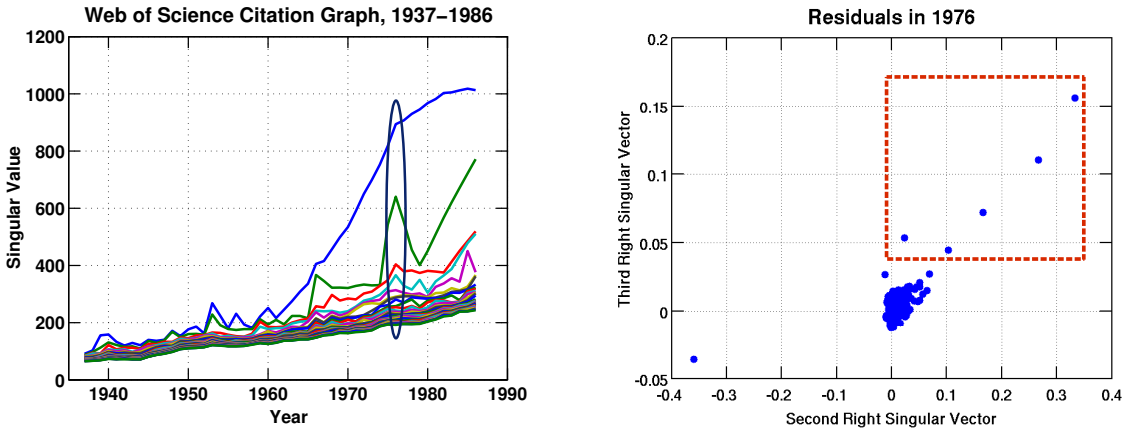

Figure 2. The Web of Science citation graph over 50 years. Singular values of the integrated residuals matrices grow over time, with a large spike in values highlighted in 1976 (left). The singular vectors in 1976 show 5 outliers corresponding to analytical chemistry papers with high cross-subject citation (right).

in 1976, but not as high as some other vertices that do not stand out in the residuals space. As it turns out, the dynamics and the attributes of the graph cause these papers to stand out over vertices that would be stronger otherwise. These 5 articles increase their annual citation count over the course of the 6-year time window, from less than 700 collective citations in 1971 to about 1000 in 1976. The ramp filter emphasizes this growth to make these vertices stand out more prominently. These documents also received citations from a broader range of subjects than other documents. The 5 outliers are cited by documents from between 63 and 76 of the 290 different subject labels over the 6 years, while other analytical chemistry papers with even higher degree were cited by documents from 47 to 55 different subjects over the same period. Without considering document subjects, these 5 documents are buried in the background noise, but their substantial cross-subject citation brings them into the front of the residuals space, demonstrating the power of this approach.

VII. Summary

In this paper, we describe a simple model for incorporating vertex attributes into spectral analysis of dynamic graph residuals. An approximation to the logistic regression model enables both efficient computation of residuals and simple fitting of model parameters. Simulations demonstrate the benefit to detection performance of taking both dynamics and vertex attributes into consideration, and an analysis of a dynamic citation network shows that accounting for attribute information emphasizes portions of the graph that would not be strong otherwise. It is clear that using vertex metadata is a powerful technique for subgraph detection, and future work will focus on extending the approximation used here to an even broader class of models, analyzing the statistics of the suggested estimator, and determining detectability when the observations are corrupted or obfuscated.

Acknowledgment

The authors wish to thank J. Kepner and M. S. Beard for their tremendous help in setting up the data ingestion and access architecture, enabling the analysis in Section VI, and A. Reuther, R. A. Bond and D. Martinez for their support of this research.

References

[1] T. Idé and H. Kashima, "Eigenspace-based anomaly detection in computer systems," in *KDD*, pp. 440–449, 2004.
[2] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E*, vol. 69, no. 2, 2004.
[3] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *J. ACM*, vol. 46, pp. 604–632, September 1999.
[4] B. A. Miller et al., "A scalable signal processing architecture for massive graph analysis," in *ICASSP*, pp. 5329–5332, 2012.
[5] C. Weinstein, W. Campbell, B. Delaney, and G. O'Leary, "Modeling and detection techniques for counter-terror social network analysis and intent recognition," in *Proc. IEEE Aerospace Conf.*, pp. 1–16, 2009.
[6] N. Arcolano et al., "Moments of parameter estimates for Chung–Lu random graph models," in *ICASSP*, pp. 3961–3964, 2012.
[7] B. A. Miller and N. T. Bliss, "Toward matched filter optimization for subgraph detection in dynamic networks," in *SSP*, pp. 113–116, 2012.
[8] B. A. Miller, N. T. Bliss, and P. J. Wolfe, "Toward signal processing theory for graphs and non-Euclidean data," in *ICASSP*, pp. 5414–5417, 2010.
[9] D. M. Dunlavy, T. G. Kolda, and E. Acar, "Temporal link prediction using matrix and tensor factorizations," *ACM Trans. Knowl. Discovery from Data*, vol. 5, Feb. 2011.
[10] K. T. Miller, T. L. Griffiths, and M. I. Jordan, "Nonparametric latent feature models for link prediction," in *Advances in Neural Inform. Process. Syst. 22* (Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, eds.), pp. 1276–1284, 2009.
[11] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys. Rev. E*, vol. 74, no. 3, 2006.
[12] S. T. Smith et al., "Network detection theory and performance." Preprint: arXiv:1303.5613v1, 2013.
[13] "Thomson Reuters Web of Science." http://thomsonreuters.com/ products_services/science/science_products/a-z/web_of_science.
[14] M. DuBois et al., "Colorimetric method for determination of sugars and related substances," *Anal. Chem.*, vol. 28, no. 3, pp. 350–356, 1956.
[15] P. S. Chen, T. Y. Toribara, and H. Warner, "Microdetermination of phosphorus," *Anal. Chem.*, vol. 28, no. 11, pp. 1756–1758, 1956.
[16] T. Bitter and H. M. Muir, "A modified uronic acid carbazole reaction," *Anal. Biochem.*, vol. 4, pp. 330–334, October 1962.
[17] R. S. Nicholson and I. Shain, "Theory of stationary electrode polarography: Single scan and cyclic methods applied to reversible, irreversible, and kinetic systems," *Anal. Chem.*, vol. 36, no. 4, pp. 706–723, 1964.
[18] M. S. Patterson and R. C. Greene, "Measurement of low energy beta-emitters in aqueous solution by liquid scintillation counting of emulsions," *Anal. Chem.*, vol. 37, no. 7, pp. 854–857, 1965.