MultiAspectForensics: Pattern Mining on Large-scale Heterogeneous Networks with Tensor Analysis

Koji Maruhashi
Fujitsu Laboratories Ltd.
Kanagawa 211-8588, Japan
Email: maruhashi.koji@jp.fujitsu.com

Fan Guo
Carnegie Mellon University
Pittsburgh, PA 15213, USA
Email: fanguo@cs.cmu.edu

Christos Faloutsos
Carnegie Mellon University
Pittsburgh, PA 15213, USA
Email: christos@cs.cmu.edu

***Abstract*—Modern applications such as web knowledge base, network traffic monitoring and online social networks have made available an unprecedented amount of network data with rich types of interactions carrying multiple attributes, for instance, port number and time tick in the case of network traffic. The design of algorithms to leverage this structured relationship with the power of computing to assist researchers and practitioners for better understanding, exploration and navigation of this space of information has become a challenging, albeit rewarding, topic in social network analysis and data mining. The constantly growing scale and enriching genres of network data always demand higher levels of efficiency, robustness and generalizability where existing approaches with successes on small, homogeneous network data are likely to fall short.**

We introduce *MultiAspectForensics*, a handy tool to automatically detect and visualize novel subgraph patterns within a local community of nodes in a heterogenous network, such as a set of vertices that form a dense bipartite graph whose edges share exactly the same set of attributes. We apply the proposed method on three data sets from distinct application domains, present empirical results and discuss insights derived from these patterns discovered. Our algorithm, built on scalable tensor analysis procedures, captures spectral properties of network data and reveals informative signals for subsequent domain-specific study and investigation, such as suspicious port-scanning activities in the scenario of cybersecurity monitoring.

I. Introduction

Modern applications in the Internet era, either data-informed or data-driven, has contributed to the boom of network data arising from a spectrum of domains, such as web knowledge base, network traffic monitoring and online social networks. A glowing trend in the accumulation and analysis of such data is the emergence of heterogenous interactions between nodes in the network, for which a vivid depiction is offered by the Facebook friendship page, with multiple page elements ranging from wall posts, comments, and photos, to mutual friends, shared interests and common networks between a pair of users. Browsing and navigation over such a space of information, despite its overwhelming scale and complexity, has been a challenging task common encountered in many fields. Yet the rather recent availability and popularity of these data, in addition to practical requirements over the efficiency, robustness and generalizability of the solution, has rendered the topic of pattern mining for heterogenous network data a relatively underexplored one, where even the definition of interesting or abnormal *patterns* could become a non-trivial problem itself.

Many of pioneering studies on pattern discovery for graph and network data focused on frequent substructure mining, with heuristics motivated by information theory [1], mathematical graph theory [2], [3], inductive logic programming [4], etc. An intimately related problem is the detection of rare event and anomalous behavior, which has attracted wide interests thanks to its many well-recognized applications concerned with security, risk assessment, and fraud analysis. Noble and Cook [5] were among the first to address this challenge on structured network data by providing solutions based on the minimal description length principle to search for abnormal subgraphs. And many alternative approaches are now available to spot anomalous nodes [6], edges [7], or both [8], with further elaboration adapted to bipartite graphs [9], and time-evolving graphs [10]. This piece of work, by revealing two classes of patterns in the context of heterogenous graphs, resembles a novel attempt to explore this relatively young realm of multi-aspect network data for state-of-the-art discoveries and developments.

We resort to a tensor-based representation for heterogenous network data and employ off-the-shelf decomposition algorithms [11] as a starting point of the analysis. Previous research along this line has paid a great deal of attention on individual nodes, which play a central role in similarity ranking [12], personalized recommendation [13], etc. The major finding in our study is that, for multiple heterogenous network data across diverse application domains, we could always observe groups of elements with similar connections along one or more data modes, as implied by nearly-identical decomposition scores, which transform to quite visible spikes in histogram plots. While algorithms in aforementioned studies mostly look for elements with top eigenscores, our heuristic distinguishes itself by being able to capture patterns formed by less well-connected nodes in the network, which do not necessarily stand out in the eigenspace and are often ignored by other extant techniques.

In summary, we propose *MultiAspectForensics*, which starts with a data decomposition step for input heterogeneous networks, features a spike detection heuristic to reveal non-trivial substructure patterns, and also includes programs to automatically visualize the findings. We demonstrate its effectiveness and efficiency by execut-

978-0-7695-4375-8/11 $26.00 © 2011 IEEE
DOI 10.1109/ASONAM.2011.80

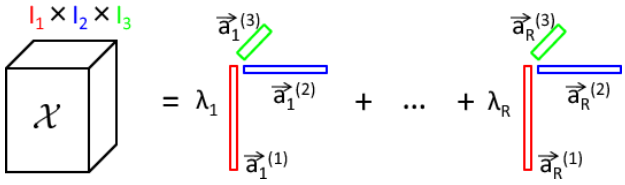

Figure 1. Illustration of the CP decomposition: the input 3-mode tensor on the left is decomposed into R triplets of vectors on the right, reminiscing of the rank-R singular value decomposition of a matrix.

ing *MultiAspectForensics* on three data sets from distinct application scenarios, present empirical results and investigate the discovered patterns, which could be leveraged to suggest suspicious activities from network traffic logs such as port-scanning and denial-of-service attack, extract interesting facts from a web knowledge base such as punk musicians or low-cost airline destinations, and report gene function groups in a developmental biology study consistent with established theories.

The remainder of this paper is organized as follows: we first elaborate on *MultiAspectForensics* procedures step-by-step in Section II, and then cover experimental studies in Section III. Related literatures are briefly sketched in Section IV. Lastly, Section V concludes the discussion and highlights future directions.

II. Algorithm

MultiAspectForensics, in a nutshell, consists of the following steps:

- *Data Decomposition*: take the input heterogenous network as a tensor and perform the CP decomposition to obtain an eigenscore vector along each data mode.
- *Spike Detection in Histograms*: iterate over all data modes to obtain histograms and apply the spike detection algorithm.
- *Substructure Discovery*: identify the induced subgraph for each spike and summarize patterns discovered.
- *Visualization*: create attribute plots and histogram plots with detected spikes highlighted.

The above procedure just makes use of the strongest component after data decomposition. If the contribution of the top one eigen-component is not as large, the latter three steps should be carried out over multiple strongest components in a similar fashion. For brevity, we subsequently elaborate on three algorithmic steps with only the first component taken into consideration, and the visualization step is illustrated by resulting figures intermixed with the rest of the discussion.

A. Data Decomposition

We first introduce a few definitions. A *tensor* can be represented as a multi-dimensional array of scalars. Its *order* is the dimensionality of the array, while each dimension is known as one *mode*, of which the value ranges over the set of *elements* for the specific mode. Thus, vectors are tensors of order one, and matrices are

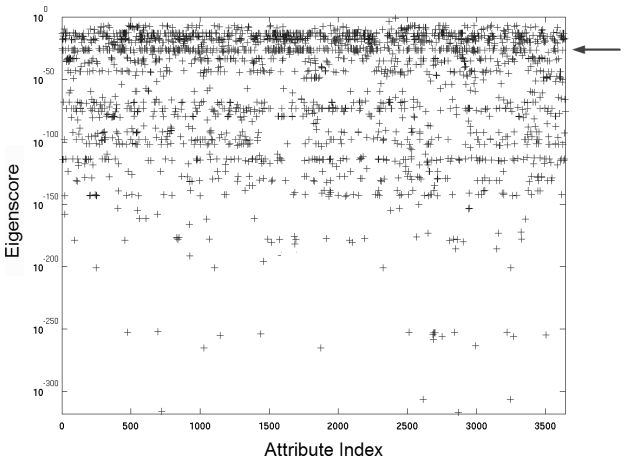

Figure 2. An attribute plot which displays absolute values of eigenscores (y-axis in log-scale) along its elements (indexed by the x-axis). The arrow on the right points to a common score value, illustrating an observation critical to the algorithmic design of *MultiAspectForensics*.

tensors with two modes. In Section III we will use *measure* to denote the unit of each *entry* in the multi-dimensional array.

To transform a heterogenous network into a tensor, every edge becomes a non-zero entry in the multi-dimensional array, where edge attributes, together with edge source and destination, make up different modes of the tensor. Edge weights naturally stay as entry values for weighted networks. Node attributes could also be incorporated by taking a Cartesian product over two end points of an edge, for instance, if a directed network contains nodes with 7 different colors, we could have an edge attribute whose arity is $7^2 = 49$.

Tensor decomposition leverages multi-linear algebra to the analysis of high-order data. The canonical polyadic (CP) decomposition we applied in this paper generalizes the singular value decomposition (SVD) for matrices. It factorizes a tensor to the weighted sum of outer products of mode-specific vectors, as illustrated in Figure 1 for a 3-order tensor. Formally, for an M-mode tensor $\mathcal{X}$ of size $I_1 \times I_2 \times \cdots \times I_M$, its CP decomposition of rank R yields

$$\begin{aligned}\mathcal{X}(i_1, \ldots, i_M) &\approx \sum_{r=1}^{R} \lambda_r \left(\overrightarrow{a_r^{(1)}} \times \ldots \times \overrightarrow{a_r^{(M)}} \right) \\ &= \sum_{r=1}^{R} \lambda_r \prod_{m=1}^{M} a_{r i_m}^{(m)} \end{aligned} \quad (1)$$

Similar to SVD, the approximation becomes closer as R enlarges, and would be exact if it equals the rank of the tensor (see [14] for details).

B. Spike Detection in Histograms

Now that we have transformed complex structured data into a set of more manageable vectors, the next step is to spot common patterns from these vectors. As a starting point, we visualize each vector by creating an attribute

Algorithm 1 *SDA* (Spike Detection Algorithm)

Require: Eigenscore histogram vector H of size N
Ensure: The set indicating spikes detected S

```
S = φ
sort the histogram to obtain an ordered vector H_o s.t. H_{o_1} ≥ H_{o_2} ≥ ··· ≥ H_{o_N}
Q_SUM ← Σ_{n=1}^{N} H_n^2
Q ← 0
for k = 1, ..., K do
    S ← S ∪ {o_k}
    Q ← Q + H_{o_k}^2
    if Q/Q_SUM ≥ s and H(o_k)/H(o_1) < r then
        break
    end if
end for
return S
```

plot, which displays absolute values of eigenscores (y-axis) along its elements (indexed by the x-axis). An example of such plots is given in Figure 2. Note that the y-axis should be in *log* scale to emphasize on the relative difference. The arrow on the right indicates a score value shared by many elements, which is not uncharacteristic in other dimensions and across different data sets. *This key observation* enables us to create effective heuristics to extract spikes from histograms and subsequently examine subgraph patterns they imply in the next subsection. And the fact that many spikes do not appear at the very top of the figure with most significant eigenscore values makes it more difficult for many alternative methods to be effective.

Prior to applying the spike detection heuristics, we obtain histogram data by equally dividing the range of eigenscores in log scale. The detection algorithm just needs to sort and traverse the histogram data until one of the following conditions is satisfied: (1) the energy as measured by sum of square values covered is equal or more than a fraction of s, and the magnitude of the spike is less than a fraction of r than the largest one; (2) there are already K spikes. Parameter values are empirically set to $s = 90\%, r = 50\%, K = 20$, where small variations lead to little perturbation of the output. The pseudo-code of the algorithm is listed in Algorithm 1 above. Application of this algorithm to the data vector in Figure 2 yields Figure 3, where we put attribute plot on the left side-by-side with histogram plot on the right, highlighting every spike in red.

C. Substructure Discovery

Having extracted sets of elements that form histogram spikes from each data mode, we head back to the input network data to examine corresponding local subnetworks to complete the final step of pattern discovery. The running example in this subsection comes from a snapshot of network traffic log which consists of packet traces in an enterprise network [15]. Each trace in the log is a triplet of (*source-IP*, *destination-IP*, *port-number*), which could be represented as a directed network of machine

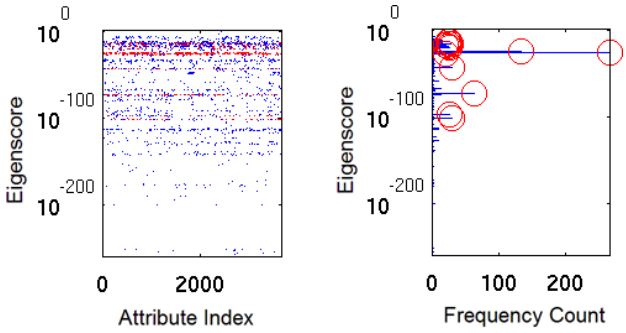

Figure 3. An attribute plot (adopted from Figure 2) on the left side-by-side with the corresponding histogram plot with spikes detected indicated by circles.

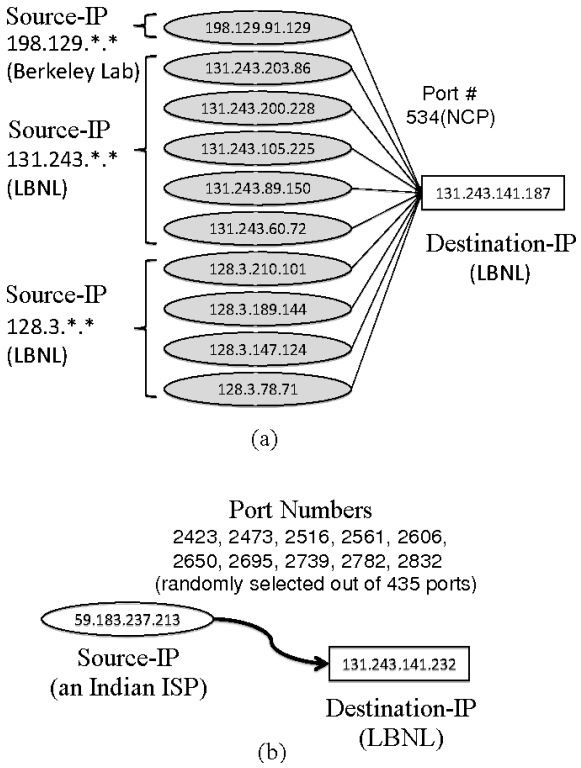

Figure 4. Examples of generalized star patterns discovered in the LBNL (Lawrence Berkeley National Lab) network traffic data set. Wavy arrows indicate multiple edges between the pair of nodes with a handful of distinct attribute values. (a) 10 source IP addresses (randomly selected out of 172 ones) are sending multiple packets to a server machine with Port# 524, which is a UDP port under the NCP protocol from a network OS for file sharing and printing services; (b) The source IP registered by an Indian ISP is sending packets to a host in LBNL via port numbers (ranging from 2,300 to 2,900) not usually intended for this type of communication, implying a suspicious activity.

IP addresses with the only edge attribute "port number" and number of packets as edge weights. Patterns derived from *MultiAspectForensics* could be summarized into the following two categories:

Pattern 1 (generalized star): A subnetwork which consists of conterminous edges that differ only in one data mode. For instance, a group of source IP addresses sending

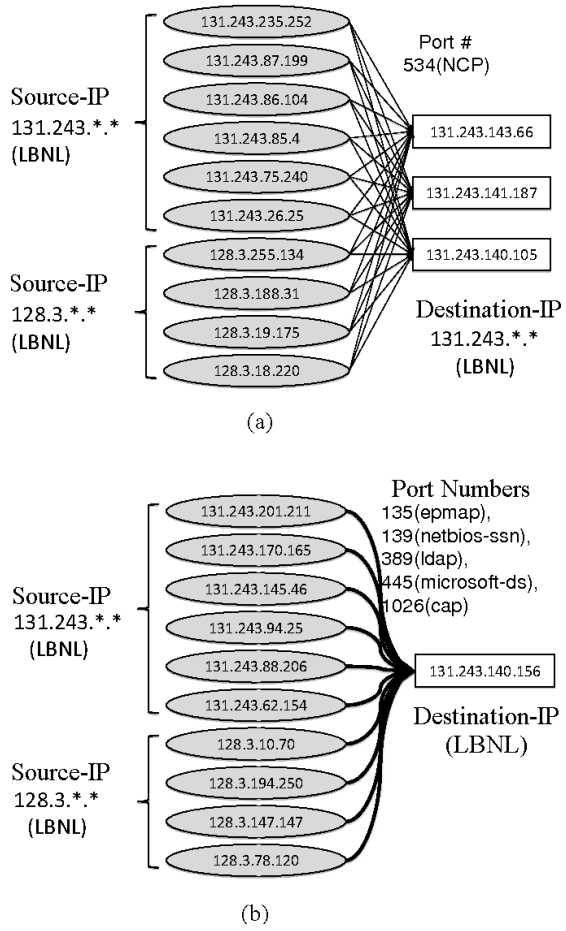

Figure 5. Examples of generalized bipartite-core patterns discovered in the LBNL (Lawrence Berkeley National Lab) network traffic data set. Wavy arrows indicate multiple edges between the pair of nodes with a handful of distinct attribute values. (a) 10 source IP addresses (randomly selected out of 119 ones) are sending multiple packets to an array of server machines over a port used for file sharing and printing services; (b) 10 source IP addresses (randomly selected out of 63 ones) are sending packets over different ports to a multi-purpose server machine.

packets to a single destination server using the same port. It generalizes the star pattern in two dimensional graphs, and makes up a continuous block along one dimension in the adjacency tensor, if elements along that dimension are order carefully. Note that in a heterogenous network, this category of patterns also includes multiple edges between one pair of nodes with differing attribute values, *e.g.*, a good many port numbers in our running example, in which case the source machine may be either an administrator performing port screening or a suspect trying to exploit a vulnerable port. Figure 4 provides an illustration of these patterns.

Pattern 2 (generalized bipartite-core): A subnetwork that represents a dense bipartite structure similar to the bipartite-core pattern in regular graphs. More generally, it can be viewed as a continuous block along two dimensions in higher-order tensors under specific element orders. For instance, a group of source IP addresses sending packets to multiple destination servers with the same port. Note that in a heterogenous network, this category of patterns also includes, written in the language of network forensics, multiple source IP addresses sending packets over different port numbers to the same server. This is likely to happen during a DDoS (Distributed Denial-of-Service) attack, a typical scenario of network intrusion, in which source IPs play the role of malicious hosts sending huge volumes of packets to the target server as the victim. Figure 5 provides an illustration of these patterns.

As a final remark, the statement that both patterns are belated to a block along one or two dimensions in the high-order tensor only holds when elements of their respective data modes are ordered in specific ways. And the complexity to search for such an order is generally exponential, which reflects, in some sense, the power of the proposed approach.

Data set	# modes	Dimensions	Measure	# non-zero elements
LBNL	4	2,345 source IPs, 2,355 dest IPs, 6,055 port #'s, 3,610 time ticks	# packets	281K
RTW	3	3,641 subjects, 3,929 objects, 98 verbs	binary	10K
BDGP	3	4,491 genes, 248 terms, 6 stages	binary	38K

Table I
A SUMMARY OF DATA SETS

III. EMPIRICAL RESULTS

We commence this section with the description of data sets as well as experimental environment. It is followed by the discussion of respective patterns discovered by *MultiAspectForensics* in each of the three data sets.

A. Data and Environment

Data sets are acquired from three dissimilar application domains: network traffic monitoring, knowledge networks, and bioinformatics. A summary is highlighted in Table I.

LBNL The network traffic log is made available through a research effort to study the characteristics of traffic for Internet enterprises [16]. The measurement was taken on servers within the Lawrence Berkeley National Lab (LBNL) from thousands of internal hosts over time, with millions of packet traces recorded. Each packet trace includes four data modes: source IP, destination IP, port number, and a time tick in second. With privacy in concern, lower 16 bits were randomly permuted to anonymize the host identity, whereas upper 16 bits were kept intact for proper identification of the location and service provider [17]. We borrowed a subset of this data set within 1-hour time span in this section.

RTW This online knowledge base is the outcome of the NELL (Never-Ending Language Learning)

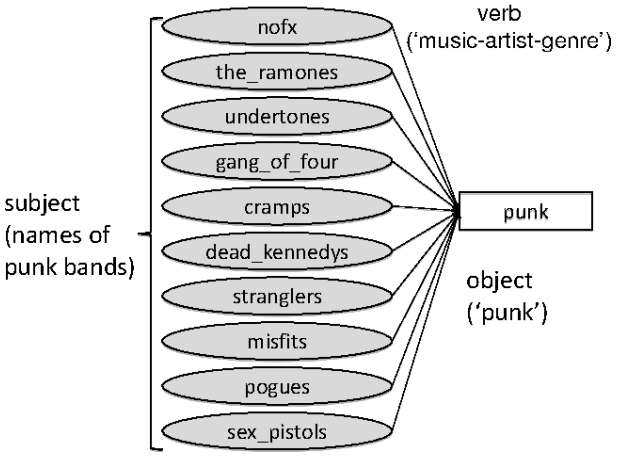

Figure 6. A generalized star pattern discovered from the RTW knowledge base about 49 punk music artists, of which a random selected set of 10 are listed. They are all specialized in punk or one of its sub-genres according to the knowledge base.

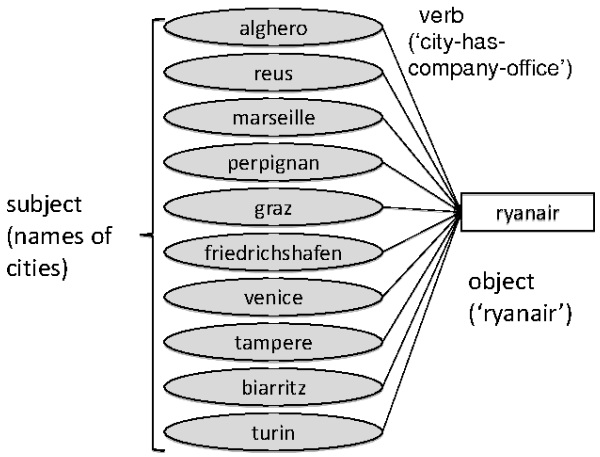

Figure 7. A generalized star pattern discovered from the RTW knowledge base about 36 European destinations of the Ryanair, an Irish low-cost airline, of which a random selected set of 10 are listed. Many of these cities have only sparse connections with other verbs.

system at Carnegie Mellon University [18]. It employs natural language processing and machine learning techniques to constantly and automatically crawl web pages and extract facts [19]. Each fact is a triplet of (subject, verb, object) such as (*pittsburgh*, *city-located-in-state*, *pennsylvania*), which could be represented as a directed graph made up of entities like *pittsburgh* or *pennsylvania*, edges with attributes like *city-located-in-state*. For better quality of results, we applied our algorithm on a preprocessed subset after noise removal (by courtesy of Dr. Byran Kisiel at Carnegie Mellon University).

BDGP The data set is collected from the Berkeley Drosophila Genome Project (BDGP) to study the spatial-temporal patterns of gene expression during the early development of fruit fly [20], [21]. We selected three data modes from the database dump available at [22], which consists of 4,491 genes, 248 functional annotation terms from a specialized vocabulary, and 6 different developmental stages.

MultiAspectForensics was implemented in the MATLAB language, and all following experiments were performed on a Unix machine with four 2.8GHz cores, and 16GB memories. For every of these data sets, the wall-clock time was no more than 2 minutes to carry out the computation and generate attribute plots and histogram plots along all modes.

B. LBNL Traffic Log

We have already discussed patterns discovered from a snapshot of this data set in Section II-C, illustrated in Figures 4, 5. With the additional mode of time tick, we found two dominating spikes in its histogram plot. Upon closer examination, we reported the following activities: the first spike is a generalized bipartite-core pattern related to the HTTP traffic on port 80 between four servers in LBNL and three remote hosts in Chinese academic institutions, possibly executing scripts to crawl/download web pages. The second spike represents a generalized star pattern between one of the local HTTP server and the same remote host at India aforementioned. We traced further in time and found that the remote host never sent packets back to acknowledge the connection, suggestive of suspicious activities to be reported to domain experts.

C. RTW Knowledge Base

Recall that each item in the knowledge database could be represented as a (subject, verb, object) triplet. *MultiAspectForensics* detected spikes mostly on data modes representing subjects and objects.

Figure 6 illustrates a subgraph discovered revealing a generalized star pattern. The music artists/bands listed here are specialized to punk music or its sub-genres (not shown in the figure) according to the knowledge base, whereas their more versatile peers will not be favorably selected by *MultiAspectForensics*.

Figure 7 displays another generalized star pattern between European cities and an Irish low-cost airline which flies to many regional or secondary airports to reduce cost, following a different business model and choice of destination from industrial giants.

The evidence here and many others alike could also be leveraged in a variety of graph mining tasks on this knowledge base such as clustering entities or creating an ontology between them, given the fact that nodes within the same spike tend to behave similarly and specifically. Moreover, as a sanity check, since node names are ordered alphabetically in this data set, the pattern does not make a continuous block in the tensor without non-trivial permutation.

D. BDGP Gene Annotation

In this data set *MultiAspectForensics* spots a set of genes known to be responsible for the *maternal effect* in the early

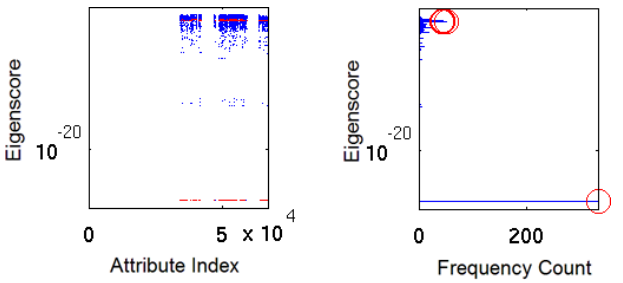

Figure 8. An attribute plot on the left side-by-side with the corresponding histogram plot for the "gene" mode of the BDGP data set. The largest spike appeared at the bottom is the set of *maternal genes*, a special class of genes that play a vital role in early embryo development such as the polarity of the egg, *i.e.*, which part will become the head and which other part turns into the tail later.

development of fruit fly (Figure 8), which also provides hints to study other higher organisms including *Homo sapiens*. Products of such maternal effect genes, in the form of either protein or mRNA, play a critical role in the very early stage of embryo development, such as the first few cell divisions. For instance, four of such genes, including *bicoid*, *caudal*, *hunchback*, and *nanos*, is mostly responsible for the determination of anterior-posterior axis – which side of the embryo will be the future head and which other side will be the future tail [23].

IV. Related Work

A. Anomaly Detection

Outlier detection, despite its wide interest across many application domains, is usually a challenging problem, as reflected in the fact that even a formal definition is not easy to make. A classical one was given by Hawkins in [24]: "an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism".

Outlier detection methods can be categorized into two sets: parametric, statistical-based approaches, and non-parametric, model-free approaches. A common characteristic of methods in the former category is the existence of statistical assumptions about the underlying data distribution [25]. The latter category usually makes the call by resorting to distance computation [26] or density estimation [27], [28]. Besides, projection-based methods [29] have been introduced for high-dimensional data. Moreover, clustering algorithms may output outlier labels as a by-product (*e.g.*, [30]).

Compared to outlier detection, anomaly detection in structured data has only gained recent attention [31], where we have reviewed relevant studies in the introductory section and claimed that there is no other attempt, to the extent of our knowledge, to discover similar patterns in heterogeneous network data as *MultiAspectForensics*.

B. Tensor Analysis and Graph Mining

Tensor decomposition has been a basic technique well studied and applied to a wide range of disciplines and scenarios. An informative survey on tensor decompositions is presented by Kolda and Bader [11] with many further references. Recent researches have further generalized the CP decomposition to handle incomplete data [32], or to produce non-negative components [33]. Tucker decomposition, as the other well-known approach, is more flexible, although its application is usually limited by its limited scalability and vulnerability to noise. Notably, recent work on scalable alternatives such as [34] may open up the venue to enhance the *MultiAspectForensics* methodology with more powerful decomposition algorithms.

Quite a few popular implementations of tensor decomposition algorithms for academic researchers have been made publicly available. Examples are the N-way toolbox by Andersson and Bro [35] and the more recent MATLAB Tensor Toolbox by Bader and Kolda [36].

Tensor analysis has also been applied to study the dynamics of graphs and networks [37]. They commonly start by analyzing graph/tensor snapshots within each timestamp, and take the output for subsequent time-series analysis. *MultiAspectForensics*, instead of focusing on the evolution between adjacent timestamps, treats timestamp as another data mode to allow better discovery of global patterns in this trade-off.

V. Conclusion

We presented *MultiAspectForensics*, a handy and effective tool to automatically detect and visualize a category of novel patterns, including generalized star and generalized bipartite-core patterns, within a local community of nodes in heterogenous networks, even if they exist among less-well connected nodes which are more likely to be ignored by many extant methods. Empirical results exhibited valuable insights derived from pattern discovered, across multiple application domains such as network traffic monitoring, knowledge networks, and bioinformatics. These successes could be attributed to the fact that we resorted to a tensor-based representation to facilitate data decomposition, reached a key observation leading to spike patterns in histogram plots, and revealed typical substructures reflecting spectral properties of heterogenous data. Hence *MultiAspectForensics* realizes an early attempt to research substructure patterns commonly existing in heterogenous network data, and a reasonable use case of tensor analysis, despite the simplicity of heuristics resided.

An important problem beyond the scope of this manuscript is the design of an objective and quantitative evaluation framework of discovered patterns, especially for large-scale networks for which it is prohibitive to label every interesting pattern. This would also shed lights on a principle way of optimizing parameters, though we found that results were usually not sensitive to parameter values when they vary within reasonable ranges. Meanwhile, it's our plan to open-source the *MultiAspectForensics* tool based on the generic boost graph library [38] to make it more accessible and usable by industrial practitioners and academic researchers, and collect feedbacks for possible future developments.

ACKNOWLEDGMENT

We are grateful to anonymous reviewers for their helpful and enjoyable comments.

This material is based upon work supported by the National Science Foundation under Grants No. DBI-0640543, IIS-0970179, the Defense Threat Reduction Agency under contract No. HDTRA1-10-1-0120, and a Google Focused Research Award. 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.

Research was also sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

REFERENCES

[1] D. J. Cook and L. B. Holder, "Substructure discovery using minimum description length and background knowledge," *Journal of Artificial Intelligence Research*, vol. 1, pp. 231–255, February 1994.

[2] X. Yan and J. Han, "gSpan: Graph-based substructure pattern mining," in *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM '02)*, 2002, p. 721.

[3] M. Kuramochi and G. Karypis, "An efficient algorithm for discovering frequent subgraphs," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, pp. 1038–1051, September 2004.

[4] L. Dehaspe and H. Toivonen, "Discovery of frequent datalog patterns," *Data Mining and Knowledge Discovery*, vol. 3, pp. 7–36, March 1999.

[5] C. C. Noble and D. J. Cook, "Graph-based anomaly detection," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '03)*, 2003, pp. 631–636.

[6] L. Akoglu, M. McGlohon, and C. Faloutsos, "OddBall: Spotting anomalies in weighted graphs," in *Proceedings of the 14th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD '10)*, 2010, pp. 410–421.

[7] D. Chakrabarti, "AutoPart: parameter-free graph partitioning and outlier detection," in *Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD '04)*, 2004, pp. 112–124.

[8] W. Eberle and L. Holder, "Discovering structural anomalies in graph-based data," in *Proceedings of the Seventh International Conference on Data Mining Workshops (ICDMW '07)*, 2007, pp. 393–398.

[9] J. Sun, H. Qu, D. Chakrabarti, and C. Faloutsos, "Neighborhood formation and anomaly detection in bipartite graphs," in *Proceedings of the Fifth IEEE International Conference on Data Mining (ICDM '05)*, 2005, pp. 418–425.

[10] H. Tong, S. Papadimitriou, J. Sun, P. S. Yu, and C. Faloutsos, "Colibri: fast mining of large static and dynamic graphs," in *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '08)*, 2008, pp. 686–694.

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

[12] T. Franz, A. Schultz, S. Sizov, and S. Staab, "TripleRank: Ranking semantic web data by tensor decomposition," in *Proceedings of the 8th International Semantic Web Conference (ISWC '09)*, 2009, pp. 213–228.

[13] N. Zheng, Q. Li, S. Liao, and L. Zhang, "Flickr group recommendation based on tensor decomposition," in *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval (SIGIR '10)*, 2010, pp. 737–738.

[14] J. Håstad, "Tensor rank is NP-complete," *Journal of Algorithms*, vol. 11, pp. 644–654, December 1990.

[15] Lawrence Berkeley National Laboratory and ICSI. LBNL/ICSI enterprise tracing project. [Online]. Available: http://www.icir.org/enterprise-tracing/

[16] R. Pang, M. Allman, M. Bennett, J. Lee, V. Paxson, and B. Tierney, "A first look at modern enterprise traffic," in *Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement (IMC '05)*, 2005, pp. 2–2.

[17] R. Pang, M. Allman, V. Paxson, and J. Lee, "The devil and packet trace anonymization," *SIGCOMM Computer Communication Review*, vol. 36, pp. 29–38, January 2006.

[18] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. H. Jr., and T. M. Mitchell, "Toward an architecture for never-ending language learning," in *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI 2010)*, 2010, pp. 1306–1313.

[19] A. Carlson, J. Betteridge, R. C. Wang, E. R. Hruschka, Jr., and T. M. Mitchell, "Coupled semi-supervised learning for information extraction," in *Proceedings of the third ACM international conference on Web search and data mining (WSDM '10)*, 2010, pp. 101–110.

[20] P. Tomancak, A. Beaton, R. Weiszmann, E. Kwan, S. Shu, S. Lewis, S. Richards, M. Ashburner, V. Hartenstein, S. Celniker, and G. Rubin, "Systematic determination of patterns of gene expression during Drosophila embryogenesis," *Genome Biology*, vol. 3, no. 12, pp. research0088.1–0088.14, 2002.

[21] P. Tomancak, B. Berman, A. Beaton, R. Weiszmann, E. Kwan, V. Hartenstein, S. Celniker, and G. Rubin, "Global analysis of patterns of gene expression during Drosophila embryogenesis," *Genome Biology*, vol. 8, no. 7, p. R145, 2007.

[22] Berkeley Drosophilia Genome Project. Patterns of gene expression in Drosophila embryogenesis. [Online]. Available: http://insitu.fruitfly.org/cgi-bin/ex/insitu.pl

[23] P. A. Lawrence, *The Making of a Fly: The Genetics of Animal Design*. Wiley-Blackwell, 1992.

[24] D. Hawkins, *Identification of outliers (Monographs on Statistics & Applied Probability)*. Chapman and Hall, 1980.

[25] V. Barnett and T. Lewis, *Outliers in Statistical Data*. John Wiley and Sons, 1994.

[26] E. M. Knorr, R. T. Ng, and V. Tucakov, "Distance-based outliers: algorithms and applications," *The VLDB Journal*, vol. 8, no. 3-4, pp. 237–253, 2000.

[27] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: identifying density-based local outliers," *SIGMOD Record*, vol. 29, pp. 93–104, May 2000.

[28] W. Jin, A. K. H. Tung, and J. Han, "Mining top-n local outliers in large databases," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '01)*, 2001, pp. 293–298.

[29] C. C. Aggarwal and P. S. Yu, "Outlier detection for high dimensional data," *SIGMOD Record*, vol. 30, pp. 37–46, May 2001.

[30] V. Chaoji, M. A. Hasan, S. Salem, and M. J. Zaki, "SPARCL: Efficient and effective shape-based clustering," in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM '08)*, 2008, pp. 93–102.

[31] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, pp. 15:1–15:58, July 2009.

[32] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, "Scalable tensor factorizations with missing data," in *Proceedings of the 2010 SIAM International Conference on Data Mining (SDM '10)*, 2010, pp. 701–712.

[33] A. Shashua and T. Hazan, "Non-negative tensor factorization with applications to statistics and computer vision," in *Proceedings of the 22nd international conference on Machine learning (ICML '05)*, 2005, pp. 792–799.

[34] C. E. Tsourakakis, "MACH: Fast randomized tensor decompositions," in *Proceedings of the 2010 SIAM International Conference on Data Mining (SDM '10)*, 2010, pp. 689–700.

[35] C. A. Andersson and R. Bro, "The N-way toolbox for MATLAB," *Chemometrics and Intelligent Laboratory Systems*, vol. 52, no. 1, pp. 1–4, 2000.

[36] B. W. Bader and T. G. Kolda. (2010, March) MATLAB Tensor Toolbox Version 2.4. [Online]. Available: http://csmr.ca.sandia.gov/~tgkolda/TensorToolbox/

[37] J. Sun, D. Tao, S. Papadimitriou, P. S. Yu, and C. Faloutsos, "Incremental tensor analysis: Theory and applications," *ACM Transactions on Knowledge Discovery from Data*, vol. 2, pp. 11:1–11:37, October 2008.

[38] J. Siek, L.-Q. Lee, and A. Lumsdaine. Boost Graph Library: a powerful C++ graph library. [Online]. Available: http://www.boost.org/libs/graph/