LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-CONF-645771

Network Similarity via Multiple Social Theories

M. Berlingerio, D. Koutra, T. Eliassi-Rad, C. Faloutsos

November 7, 2013

IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining
Niagara Falls, Canada
August 25, 2013 through August 28, 2013

Disclaimer

Network Similarity via Multiple Social Theories

Michele Berlingerio
IBM Research Dublin
Email: mberling@ie.ibm.com

Danai Koutra
Carnegie Mellon University
Email: danai@cs.cmu.edu

Tina Eliassi-Rad
Rutgers University
Email: eliassi@cs.rutgers.edu

Christos Faloutsos
Carnegie Mellon University
Email: christos@cs.cmu.edu

***Abstract*—Given a set of k networks, possibly with different sizes and no overlaps in nodes or links, how can we quickly assess similarity between them? Analogously, are there a set of social theories which, when represented by a small number of descriptive, numerical features, effectively serve as a "signature" for the network? Having such signatures will enable a wealth of graph mining and social network analysis tasks, including clustering, outlier detection, visualization, etc. We propose a novel, effective, and scalable method for solving the above problem. Our approach has the following desirable properties: (a) It is supported by a set of social theories. (b) It gives similarity scores that are size-invariant. (c) It is scalable, being linear on the number of links for graph signature extraction. We present extensive experiments on numerous synthetic and real networks from disparate domains, and show how we outperform baseline competitors. We also show how our approach enables several mining tasks such as clustering, visualization, discontinuity detection, network transfer learning, and re-identification across networks.**

I. Introduction

We address the problem of *network similarity*. Specifically, given a set of k networks of potentially different sizes and without any assumptions on overlapping nodes or edges, how can we efficiently provide a meaningful measure of structural similarity (or distance)? For example, how structurally similar are the ASONAM and ICWSM co-authorship networks? How does their structural similarity compare with the similarity between the ASONAM and WSDM co-authorship networks? Such measures are extremely useful for numerous social network analysis and graph mining tasks. One such task is clustering: given a set of networks, find groups of similar ones; conversely, find anomalies or discontinuities – i.e., networks that stand out from the rest. Transfer learning is another application. If networks G_1 and G_2 are similar, we can transfer conclusions from one to the other to perform across-network classification with better predictive accuracy.

When considering the problem of network structural similarity, we need to make some choices. Should the comparison be at the local (node) level, at the global (network) level, or both? Should the comparison be based on the similarities (or distances) of the adjacency matrices or similarities (or distances) of structural features, or both? Should the approach be interpretable or is a black-box approach okay? Must the approach be scalable? Can the approach be readily extended to accommodate non-structural features? Clearly, these choices are not independent of each other. For example, comparisons at the local level tend to be more interpretable and scalable.

We present an approach, NetSimile, that has the following seven characteristics. (1) It can compare networks at the local (node and neighborhood) level. (2) It uses structural features supported by social theories. (3) It is scalable. (4) It is interpretable. (5) It is size-independent. (6) It can readily be extended to accommodate global-level features and non-structural features. (7) Its similarity values satisfy the *Identity*, *Symmetry*, and *Divergence* properties.

The core of NetSimile is a careful extraction, aggregation, and evaluation of structural features from nodes and their local neighborhoods. For every network G, we derive a small number of numerical features, which incorporate various social theories and capture the topology of the network as moments of distributions over its local structural features. The similarity score between two networks then is just the similarity of their "signature" vectors. Once we have the similarity function, we can do a wealth of interesting tasks, including clustering, visualization, anomaly detection, etc.

NetSimile incorporates four social theories when extracting the "signature" vector of a network: *Social Capital*, *Structural Hole*, *Balance*, and *Social Exchange*. These theories, respectively, capture connectivity of nodes and their neighborhoods, control of information flow, transitivity among the nodes, and reciprocity among the nodes. We selected these social theories because they are purely structural (as supposed say homophily which relies on a non-structural characteristic). Also, these theories can be applied to a wide range of networks as opposed to just social networks. Contractor, Wasserman, and Faust [1] provide a list of social theories.

Our empirical study includes experiments on more than 30 real-world networks and various synthetic networks generated by four different graph generators (namely, Erdös-Rényi, Forest Fire, Watts-Strogatz, and Barabási Preferential Attachment). We compare NetSimile with two baselines. The first baseline extracts frequent subgraphs from the given graphs and performs pairwise comparison on the intersection of the two sets of frequent patterns. The second baseline computes the k largest eigenvalues of each network's adjacency matrix and measures the distance between them.

Our experiments provide answers to the following questions: How do the various methods compare w.r.t. their similarity scores? Are their results intuitive (e.g., is a social network more similar to another social network than to a technological network)? How do they compare to null models? Are the methods just measuring the sizes of the networks in their comparisons? How scalable are the various methods?

The contributions of our work are:

- *Novelty*: By using moments of distribution as aggregators, NetSimile generates a single "signature" vector for each graph based on the local and neighborhood features of its nodes. Our features incorporate four

social theories that are endogenous to the network and are applicable to more than just social networks.
- *Effectiveness*: NETSIMILE produces similarity / distance measures that are size-independent, intuitive, and interpretable. The similarity values satisfy the identity, symmetry, and divergence properties.
- *Scalability*: The runtime complexity for generating NETSIMILE's "signature" vectors is linear on the number of edges.
- *Applicability:* NETSIMILE's "signature" vectors are useful in many social network analysis and graph mining tasks.

The paper is organized into the following sections: Proposed Method, Experiments, Related Work, and Conclusions.

II. PROPOSED METHOD

NETSIMILE has three steps: (1) *feature extraction*, (2) *feature aggregation*, and (3) *signature comparison*.

Feature extraction. Four social theories guide NETSIMILE's feature extraction: Coleman's *Social Capital* [2], Burt's *Structural Holes* [3], Heider's *Balance* [4], and Homan's *Social Exchange* [5]. We chose these social theories because they are purely structural and endogenous to the network (as opposed to, for example, Homophily which relies on a non-structural characteristic). Based on the aforementioned four theories, NETSIMILE extracts a small set of structural features for each node based on its local and egonet-based features.[1] We briefly describe the social theories used in NETSIMILE and the structural features associated with them next.

Theory of Social Capital operates at the node level and measures the connectivity of nodes and their neighborhoods. NETSIMILE captures Social Capital via these two features:

- $d_i = |N(i)|$: number of neighbors (i.e. degree) of node i; $N(i)$ denotes the set of neighbors for node i.
- $\bar{d}_{N(i)}$: average degree of N(i), computed as $\frac{1}{d_i}\sum_{\forall j \in N(i)} d_j$.

Theory of Structural Hole operates at the neighborhood level and measures control of information flow. NETSIMILE captures Structural Hole by extracting this feature:

- $|E_{ego(i)}|$: number of edges in node i's egonet; $ego(i)$ returns node i's egonet.

The degree of node i captured as part of Social Capital and the clustering-coefficient based features captured as part of Balance (see below) also serve to capture structural hole.

Theory of Balance operates at the triadic level and measures the transitivity among the nodes. NETSIMILE captures Balance by extracting these two features:

- c_i: clustering coefficient of node i, defined as the number of triangles connected to node i over the number of connected triples centered on node i.
- $\bar{c}_{N(i)}$: average clustering coefficient of N(i), node i's neighbors, calculated as $\frac{1}{d_i}\sum_{\forall j \in N(i)} c_j$.

[1] A node's egonet is the induced subgraph of its neighboring nodes.

Theory of Social Exchange operates at the dyadic level and measures reciprocity among the nodes. NETSIMILE captures Social Exchange at the egonet level via these two features:

- $|E^o_{ego(i)}|$: number of outgoing edges from $ego(i)$.
- $|N(ego(i))|$: number of neighbors of $ego(i)$.

Note that NETSIMILE is flexible enough to incorporate additional features, including global (network) level features and non-structural features, such as attributes on nodes. (See Remark 2 at the end of this section for a discussion on local- vs. global-level network comparison.) We choose the above local and egonet-based features for three reasons. First, they incorporate the pertinent local endogenous social theories (Social Capital, Structural Holes, Balance, and Social Exchange). Second, they satisfy our constraints in terms of effectiveness (namely, size-independence, intuitiveness, and interpretability) and scalability (see Section III). Third, empirically we observe that the above features are sufficient for measuring similarity across networks from various domains (see Section III).

Feature aggregation. After the feature extraction step, NETSIMILE has extracted a $node \times feature$ matrix, F_{G_j}, for each graph $G_j \in \{G_1, G_2, \cdots, G_k\}$. We can measure similarity between graphs by comparing their feature matrices (see discussion below under Remark 1). However, we discovered that generating a single "signature" vector for each graph produces more efficient and effective comparisons. To this end, NETSIMILE uses the following five aggregators on each feature (i.e., on each column of F_{G_j}): *median*, *mean*, *standard deviation*, *skewness*, and *kurtosis*. Note that the latter four of the five aggregators are moments of distribution of each feature. In other words, NETSIMILE feature aggregation captures the movements of distribution of the social theories that each feature represents. NETSIMILE is flexible enough to use other aggregators as well, though we found these five to be sufficient for the task of network comparison and satisfy our effectiveness and scalability constraints (see Section III).

Comparison. After the feature aggregation step, NETSIMILE has produced a "signature" vector $\vec{s}_{G_j}$ for every graph $G_j \in \{G_1, G_2, \cdots, G_k\}$. NETSIMILE now has the whole arsenal of clustering techniques and pairwise similarity / distance functions at its disposal. Amongst the collection of pairwise similarity / distance functions, we found Canberra Distance ($d_{Can}(P,Q) = \sum_{i=1}^{d} \frac{|P_i - Q_i|}{P_i + Q_i}$) to be very discriminative (a good property for a distance measure). This is because Canberra Distance is sensitive to small changes near zero; and it normalizes the absolute difference of the individual comparisons (see discussion in Section III).

Computational complexity. Let k = number of graphs given to NETSIMILE (i.e., $k = |\{G_1, \cdots, G_k\}|$), n_j = the number of nodes in G_j, m_j = the number of edges in G_j, f = number of structural features extracted, and r = number of aggregators used. Note that f, r, and k are small integers (in the teens).

Lemma 1: The runtime complexity for generating NETSIMILE's "signature" vectors is linear on the number of edges in $\{G_1, \cdots, G_k\}$, and specifically

$$O(\sum_{j=1}^{k}(fn_j + fn_j log(n_j))) \quad (1)$$

where $f \ll n_j \ll m_j$.

Proof: To generate NETSIMILE's "signature" vectors, features need to be extracted and then aggregated.

Feature Extraction: Recall that NETSIMILE is computing local and neighborhood-based structural features. As proved in [6], computation of neighborhood-based features is expected to take $O(n_j)$ for real-world graphs. Therefore to compute f neighborhood-based features on a graph G_j, it takes $O(fn_j)$.

Feature Aggregation: This is $O(fn_j log(n_j))$ for each graph G_j. Recall that NETSIMILE's aggregators are median, mean, standard deviation, skewness, and kurtosis. The latter four can be computed in one-pass through the f feature values. The most expensive computation is the median which cannot be done in one-pass. However, it can be computed in $O(nlog(n) + n)$ for n numbers. Basically, one needs $O(nlog(n))$ to sort the n numbers and then select the median with $O(n)$ operations. ■

Properties of NETSIMILE. NETSIMILE satisfies the properties of *Identity*, *Symmetry* and *Divergence*. Identity holds because for a given network G_1 and its "signature" vector $\vec{s}_{G_1}$, $\vec{s}_{G_1} - \vec{s}_{G_1} = 0$. Symmetry holds because for two networks G_1 and G_2 and their "signature" vectors $\vec{s}_{G_1}$ and $\vec{s}_{G_2}$, $|\vec{s}_{G_1} - \vec{s}_{G_2}| = |\vec{s}_{G_2} - \vec{s}_{G_1}|$. We show that NETSIMILE satisfies the Divergence property via an experiment. Figure 1 shows the NETSIMILE similarity scores between various networks (described in Section III-A) and their rewired versions. NETSIMILE similarity score (i.e. 1 minus the NETSIMILE Scaled Canberra Distance) is in $[0, 1]$, with 0 meaning no similarity at all and 1 meaning identical graphs. We rewire a graph by randomly reassigning a number of its edges. The rewiring parameter, $c \in [0, 1]$, determines the fraction of edges rewired. Edges are reassigned in a way that preserves the expected degree of each node in the graph. When c is 0, no rewiring takes place (i.e., the original and rewired graphs are identical) and as expected the NETSIMILE score is 1 between them. When c is 1, the rewired graph is the least similar to the original graph (because all the edges in the original graph have been randomly reassigned). As expected, comparing an Erdös-Rényi graph to its rewired version does not significantly change the NETSIMILE score (see the line with the black circles in Figure 1). However, for real-world graphs (like co-authorship networks and autonomous systems networks) as the rewiring parameter increases, the NETSIMILE score decreases.

Remark 1: Network comparison through statistical hypothesis testing. Given the $node \times feature$ matrices of two graphs, F_{G_1} and F_{G_2}, NETSIMILE can use statistical hypothesis testing to see if the two graphs are samples from the same underlying distribution. Specifically, NETSIMILE normalizes each column (i.e. feature) in F_{G_1} and F_{G_2} by its L_2 norm. Then, NETSIMILE does pairwise hypothesis testing across the features of the graphs. For example, it does hypothesis testing between the degree columns in G_1 and G_2; between the clustering coefficient columns in G_1 and G_2; and so on. This process produces seven $p\text{-}values$ (corresponding to the seven features extracted by NETSIMILE). To decide whether the two graphs are from the same underlying distribution, NETSIMILE uses the maximum $p\text{-}value$. (The average of the $p\text{-}values$ did not produce better results.)

For the statistical hypothesis tests, NETSIMILE can use any test available. We tried the Mann-Whitney (MW) Test [7] and the Kolmogorov-Smirnov (KS) Test [8]. The MW Test is nonparametric. It assumes two samples are independent and measures whether the two samples of observations have equally large values. The KS Test is also nonparametric. We used the two-sample KS Test which compares two samples w.r.t. the location and shape of the empirical cumulative distribution functions of the two samples. We found that in both MW and KS tests the maximum p-values were 0 for the overwhelming majority of the graph pairs, which indicated to us that neither test generated enough discriminative power to effectively capture differences between two graphs (though the MW Test was more discriminative than KS).

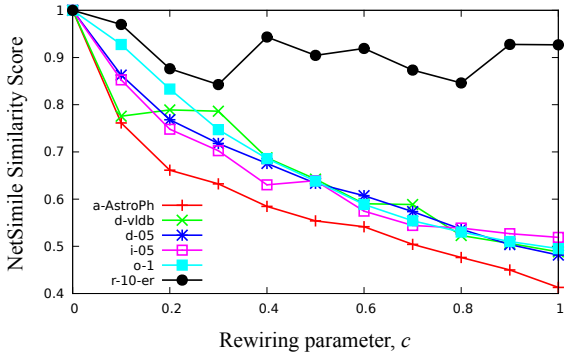


Fig. 1. NETSIMILE similarity scores for various graphs and their rewired versions. As the rewiring parameter increases, the NETSIMILE similarity score decreases in real-world networks (e.g., co-authorship networks from arXiv, DBLP, and IMDB and technological networks from Oregon AS). Unsurprisingly, increasing the rewiring parameter in an Erdös-Rényi graph (black circles) does not have the same pattern.

Remark 2: Network comparison at the local- vs. global-level. Whether one prefers local-level network similarity to global-level network similarity depends on the application for which the similarity is being used. NETSIMILE is designed such that it can take either local-level or global-level features. Here, we emphasis NETSIMILE's local-level network similarity. The advantages of local-level comparison is that node-level and egonet-level features are often more interpretable than global features – e.g., consider average degree of a node vs. the number of distinct eigenvalues of the adjacency matrix. Also, local-level features are computationally less expensive than global-level features – e.g., consider clustering coefficient of a node vs. diameter of the graph. Moreover, looking at local-level features that incorporate social theories answers the question: "are the given two networks from similar linking models?" For example, consider the Facebook and Google+ social networks. Even though Google+ is a smaller network than Facebook, are its users linking in a similar way to the users of the Facebook network? Is the smaller Google+ network following a similar underlying model as the lager Facebook network? Local-level features can capture any similarity present in the linking models of the two networks, but global-level features cannot.

III. EXPERIMENTS

This section is organized as follows. First, we outline the real and synthetic datasets used in our experiments, as well as our experimental setup. Second, we describe two baseline

Network		|V|	|E|	k	Net c	$\bar{c}$	|LCC|	#CC
arXiv	a-AstroPh	18,772	396,160	42.21	0.318	0.677	17,903	290
	a-CondMat	23,133	186,936	16.16	0.264	0.706	21,363	567
	a-GrQc	5,242	28,980	11.06	0.630	0.687	4,158	355
	a-HepPh	12,008	237,010	39.48	0.659	0.698	11,204	278
	a-HepTh	9,877	51,971	10.52	0.284	0.600	8,638	429
DBLP-C	d-vldb	1,306	3,224	4.94	0.597	0.870	769	112
	d-sigmod	1,545	4,191	5.43	0.601	0.856	1,092	116
	d-cikm	2,367	4,388	3.71	0.560	0.873	890	361
	d-sigkdd	1,529	3,158	4.13	0.505	0.879	743	189
	d-icdm	1,651	2,883	3.49	0.518	0.887	458	281
	d-sdm	915	1,501	3.28	0.540	0.870	243	165
DBLP-Y	d-05	39,357	79,114	4.02	0.415	0.642	29,458	3,229
	d-06	44,982	94,274	4.19	0.379	0.632	35,223	3,140
	d-07	47,465	103,957	4.38	0.373	0.628	38,048	3,078
	d-08	47,350	107,643	4.55	0.378	0.612	38,979	2,849
	d-09	45,173	102,072	4.52	0.331	0.595	36,767	2,920
IMDb	i-05	13,805	130,295	18.88	0.506	0.774	13,075	258
	i-06	14,228	142,955	20.09	0.480	0.760	13,458	269
	i-07	13,989	133,930	19.15	0.476	0.757	13,091	256
	i-08	14,055	132,007	18.78	0.469	0.750	13,313	273
	i-09	14,372	128,926	17.94	0.442	0.728	13,601	277
Query Log	ql-1	138,976	1,102,606	15.87	0.055	0.599	132,012	3,238
	ql-2	108,420	876,517	16.17	0.055	0.594	103,095	2,482
	ql-3	89,406	707,579	15.83	0.053	0.588	85,246	1,941
	ql-4	75,838	582,703	15.37	0.051	0.583	72,396	1,600
	ql-5	42,946	253,469	11.80	0.047	0.573	40,691	1,027
Oregon AS	o-1	10,900	31,181	5.72	0.039	0.501	10,900	1
	o-2	11,019	31,762	5.76	0.040	0.495	11,019	1
	o-3	11,113	31,435	5.66	0.034	0.490	11,113	1
	o-4	11,260	31,304	5.56	0.032	0.487	11,260	1
	o-5	11,461	32,731	5.71	0.037	0.494	11,461	1

TABLE I. REAL NETWORKS: #NODES, #EDGES, AVG DEGREE, NETWORK CLUSTERING COEFFICIENT (TRANSITIVITY), AVG NODE CLUSTERING COEFFICIENT, #NODES IN THE LARGEST CONNECTED COMPONENT, #CONNECTED COMPONENTS.

methods. Third, we present results that answer the following questions: How do the different approaches compare? Is there a particular method which clearly outperforms the others? If yes, to which extent? How can we interpret the results? Is NETSIMILE affected by the sizes of the networks? How do the proposed methods scale? How well does NETSIMILE perform in various graph mining applications?

A. Data and Experimental Setup

Real Networks. Table I lists the basic statistics of the real networks used in our experiments. Here is a short description. **arXiv** (http://arxiv.org) has 5 co-authorship networks corresponding to the following fields: Astro Physics, Condensed Matter, General Relativity, High Energy Physics and High Energy Physics Theory. **DBLP-C** (http://dblp.uni-trier.de) has 6 co-authorship networks from VLDB, SIGKDD, CIKM, ICDM, SIGMOD and WWW conferences, each spanning over 5 years (2005-2009). **DBLP-Y** has 5 co-authorship networks, each corresponding to one of the years from 2005 to 2009 and consisting of data from 31 conferences. **IMDb** (http://www.imdb.com) has 5 collaboration networks for movies issued from 2005 to 2009. Each node represents a person who took part in the movie (i.e., cast and crew). Edges connect people who collaborated on a movie. **QueryLog** (http://www.gregsadetsky.com/aol-data) has 5 word co-occurrence networks built from a query-log of approximately 20 millions web-search queries submitted by 650,000 users over 3 months. **Oregon AS** (http://snap.stanford.edu/data/) has 5 Autonomous Systems (AS) routing graphs between March 31st and May 26th 2001.

Synthetic Networks. We also produced several synthetic networks by using the following generators from the igraph library (http://igraph.sourceforge.net). **Barabási-Albert** [9]: With a non-assortative version of the generator, we created graphs with 1K, 10K, and 100K nodes, adding 4 edges in each iteration. **Forest-Fire** [10]: We generated graphs of size 1K, 10K, and 100K nodes, with 20% forward burning probability, 40% backward burning probability, and 4 ambassador vertices. **Erdös-Rényi** [11]: We used the $G(n, m)$ generator, where n is the number of nodes and m the number of edges, and produced graphs $G(n, 2n)$ with 1K, 10K, and 100K nodes. **Watts-Strogatz** [12]: We built graphs of size 200, 2K, and 20K nodes by setting the lattice dimension to 1, the degree to 4, and the rewiring probability to 0.3.

For each generator and for each node-set size, we built five networks. Our results report the average values obtained across the five networks per generator and node-set size.

Experimental Setup. We implemented our approach in C++ and Matlab, making use of the GNU Statistic Libraries and igraph. The code was run on a server equipped with 8 Intel Xeon processors at 3.0GHz, with 16GB of RAM, and running CentOS 5.2 Linux.

B. Baseline Methods

We compare NETSIMILE with (a) Frequent Subgraph Mining and (b) Eigenvalues Extraction. We chose these two methods because they are intuitive and widely applicable. Many methods discussed in Section IV are application-dependent.

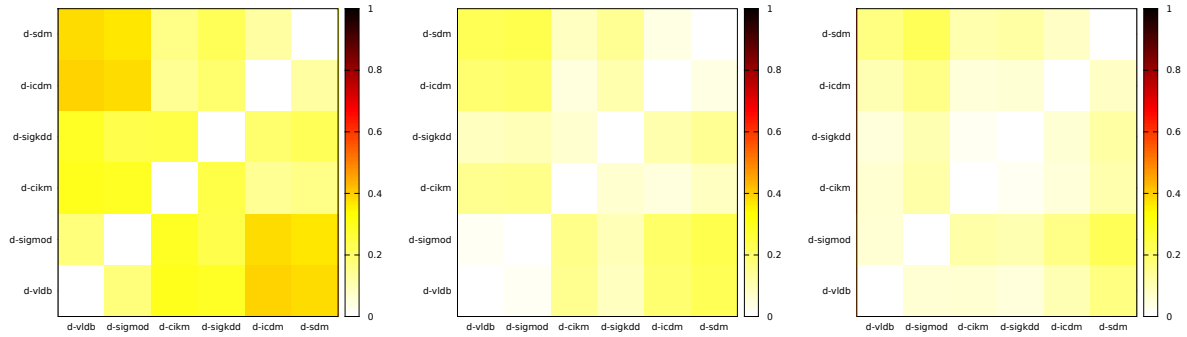
FSM (Frequent Subgraph Mining): Given two graphs, we take the intersection of their frequent pattern-sets and build two vectors (one per graph) of relative supports of their patterns [13]. We compare these FSM vectors with NETSIMILE's "signature" vectors using Canberra Distance. A clear drawback of FSM is its lack of scalability.

EIG (Eigenvalues Extraction): This is an intuitive measure of network similarity that is based on *global* feature extraction (as opposed to the *local* feature extraction of NETSIMILE). For each graph, we compute the k largest eigenvalues[2] of its adjacency matrix, and thus we obtain a vector of size k per graph. Then, we use the Canberra Distance in order to compare these vectors and find the pairwise similarities between the graphs. A disadvantage of EIG is that it is size dependent: larger networks - or ones with larger LCC (Largest Connected Component) - have higher eigenvalues. Thus, EIG will lead to higher similarity between networks with comparable sizes. Moreover, there is no global upper-bound for eigenvalues, making distance values hard to compare.

C. Comparative Results

For each method (NETSIMILE, FSM, and EIG), after extracting features from the graphs and obtaining one (aggregated) feature vector per graph, we apply the Canberra

[2]We tried a few values for k and saw no significant changes around 10; so we selected $k = 10$.



(a) NETSIMILE Scaled Canberra Distance on DBLP-C (b) FSM Scaled Canberra Distance on DBLP-C (c) EIG Scaled Canberra Distance on DBLP-C

Fig. 2. Illustration of NETSIMILE's success: Scaled Canberra Distance scores ($\in [0, 1]$) between the DBLP-C networks by NETSIMILE (a), FSM (b), and EIG (c). d-sigkdd, d-icdm, and d-sdm are data mining conferences. d-vldb and d-sigmod are databases conference; d-cikm is an information and knowledge management conference. NETSIMILE has more discriminative power than the baseline methods.

Distance. For brevity, we do not report comparative results on other similarity/distance measures that we tried.

Figure 2 depicts the results of a set of experiments involving the *scaled* Canberra Distance (where each value is in [0,1]) on the DBLP-C datasets. The columns in Figure 2 correspond to the heatmaps we obtained from NETSIMILE, FSM and EIG, respectively. Inspecting Figures 2, we observe that the results of NETSIMILE are similar to FSM. For instance, according to both, the d-vldb network is similar to d-sigmod (both are databases conferences), but d-vldb is not so similar to d-sdm (a data mining conference). However, the FSM results are much less discriminative. Also, we observe that the results from EIG differ from the ones from NETSIMILE and FSM. According to EIG, d-vldb has no significant differences with d-sigmod, d-cikm, and d-sigkdd; while NETSIMILE and FSM found differences. Since there is no global normalization for the EIG values,[3] global comparisons of a set of networks are harder to interpret with EIG than with NETSIMILE or FSM.

We also measure the entropy in feature vectors generated by NETSIMILE, FSM, and EIG on the DBLP-C co-authorship networks. As Figure 3 shows, NETSIMILE's feature vectors have higher entropy than FSM's or EIG's. Higher entropy means more uncertainty (i.e., we need more bits to store the desired information). So, NETSIMILE's feature vectors capture the nuances (i.e. uncertainty) in the graphs bettern than FSM or EIG, which then leads to more discriminative power when comparing graphs.

D. Interpretability of Results

To make sense of our results, we exploit the background knowledge about the networks used in our experiments. In the real networks, we have three sets of collaboration networks (DBLP-C, DBLP-Y and IMDb), one technological network (Oregon AS), and a word co-occurrence network (Query Log). In addition, we have different synthetic networks generated by various commonly used models. One would expect these networks to be "clustered" by their types. This idea was inspired by the considerations found in [14], where a large set of networks of different types are analyzed, together with their typical global and local features.

[3]It is possible to do pairwise normalization by the number of nodes, but this is not general for any set of networks.

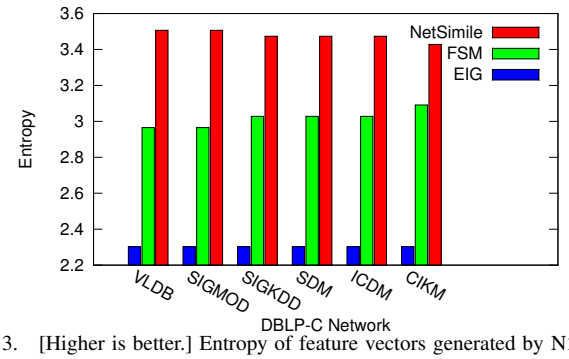


Fig. 3. [Higher is better.] Entropy of feature vectors generated by NETSIMILE, FSM, and EIG on the DBLP-C co-authorship networks. NETSIMILE's feature vectors have higher entropy than FSM's or EIG's, implying that they are capturing the nuances in the graphs better than FSM or EIG.

Figure 4(a) presents the dendrogram of all of our networks built by hierarchical agglomerative clustering with unweighted average linking and the Canberra Distance and using NETSIMILE's graph "signature" vectors. The network names are colored by data set. As evident in Figure 4(a), there is a clear distinction between the clusters. The collaboration networks appear all together, along with the forest fire synthetic networks. The Oregon AS forms a cluster that only at the height of 0.45 joins with the Query Log. The Erdös-Rényi and Watts-Strogatz form a separate cluster. This, in turns, reflects our aforementioned intuition about following our background knowledge of the data.

Figure 4(b) shows the dendrogram for the above experiment (hierarchical agglomerative clustering with unweighted average linking and the Canberra Distance) for graph vectors generated by EIG. This figure clearly shows a different picture, where the networks are grouped differently (see how the distribution of the colors is mixed). For example, in the leftmost cluster, two collaboration networks from arXiv are put together with four Query Log networks, while the missing Query Log network is placed together with the Oregon AS networks. The EIG results are not intuitive, thus making EIG not suitable for interpreting graph-similarity results.

Another way to visualize the similarity of graphs is to project NETSIMILE's $graph \times feature$ matrix into its principal component space through Singular Value Decomposition (SVD). Due to brevity, we have omitted these results.

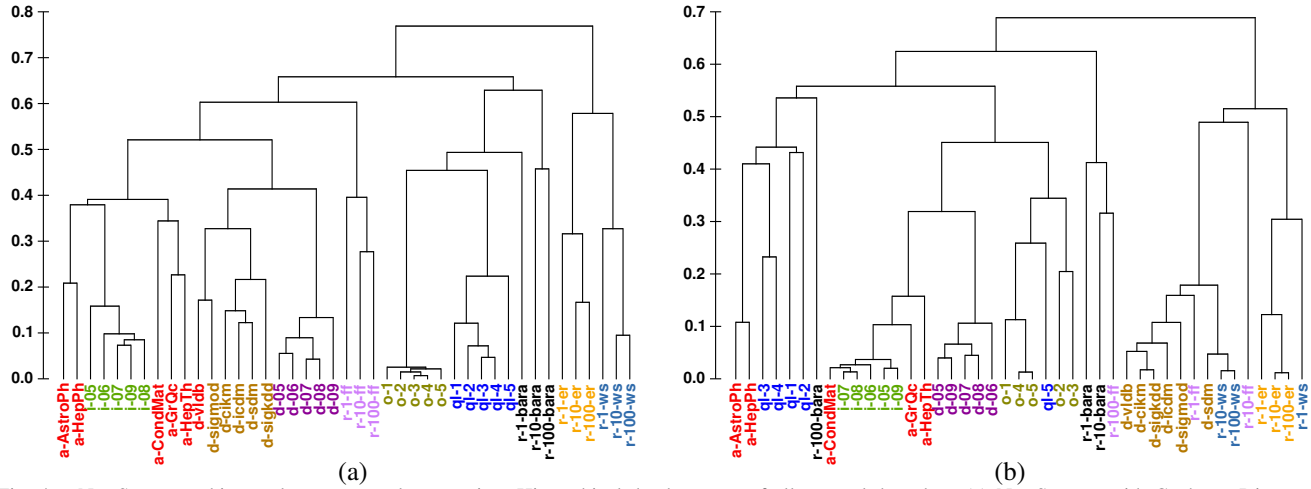


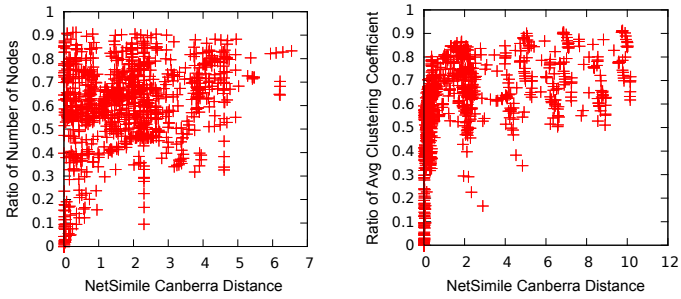
Fig. 4. NETSIMILE achieves cleaner network separation: Hierarchical dendrograms of all network based on (a) NETSIMILE with Canberra Distance, and (b) EIG with Canberra Distance. Network names are colored by data set. Homogeneity in colors (NETSIMILE's dendrogram) indicates better and more intuitive groupings (than EIG's dendrogram).

E. Similarity of Networks with Different Sizes

One question that may arise regarding NETSIMILE is whether its results are affected by the differences in sizes or other basic statistics of the two networks being compared. We do not want the size to play an important role in our solutions given that our interpretation of the question "are two networks similar?" leads to the question "do the two networks follow the same (or similar) underlying linking model?". To answer these questions, we compared the relationships between the NETSIMILE with Canberra Distance and some basic statistics of our real and synthetic networks. Specifically, we compared NETSIMILE values of two networks with the ratio between their (1) number of nodes, (2) number of edges, (3) average clustering coefficients of the nodes, (4) average degree, (5) maximum degree, and (6) network clustering coefficient. In all of them, we saw no correlation. For brevity, we only show the scatterplot for the NETSIMILE values and the ratio between the number of nodes of the two networks (see Figure 5(a)) and the scatterplot for the NETSIMILE values and the ratio between the average clustering coefficients of the nodes of the two networks (see Figure 5(b)). As evident in these scatterplots, NETSIMILE's results are not merely reflecting the difference in sizes of the networks. If they were, we would expect to observe correlations among the points in each scatterplot. This implies that we can generate two networks of the same kind, with different sizes (e.g., two Forest-Fire networks [10] with different node sizes) and NETSIMILE would find them similar.

F. Scalability

Table II reports the run times (in seconds) of NETSIMILE and the two baselines (FSM and EIG) applied to our real networks. Note that for NETSIMILE the run times refer to all the three steps described in Section II, with the comparison step constituted by the pairwise computation of both the Cosine Similarity and the Canberra Distance. For FSM we do not report running time longer than two days. NETSIMILE and EIG are able to compare graphs in a few of seconds, though EIG produces results that are size-dependent. FSM pays for its subgraph isomorphism, which considerably affects the performances. Note that FSM is affected not only by the size of the network, but also by its type. While DBLP is a set of collaboration networks (with sparsely connected cliques), the Oregon AS (being a technology network) is made of one single connected component, thus the cost for the isomorphism is much higher.



(a) NETSIMILE vs. ratio of # nodes

(b) NETSIMILE vs. ratio of avg. clustering coefficient

Fig. 5. NETSIMILE Canberra Distance is not measuring size, as there is no clear evidence of correlation between the two axes.

G. Applications

NETSIMILE can be used in numerous applications such measuring node overlap, classifying/labeling networks, and discontinuity detection. Due to brevity, we discuss two of them.

NETSIMILE as a Measure of Node-Overlap. Given three graphs G_A, G_B, and G_C of the same domain (e.g., co-authorship networks in *SIGMOD*, *VLDB* and *ICDE*), can we use *only* their NETSIMILE's "signature" vectors to gauge the amount of node-overlap between them? Our hypothesis is that if graph G_A is more similar to graph G_B than graph G_C, then G_A will have more overlap in terms of nodes with G_B than G_C. To test this hypothesis, we ran NETSIMILE with Canberra Distance on our real networks. Figure 6(a) depicts the scatterplot of NETSIMILE results on graphs within each comparable group (i.e., arXiv, DBLP-C, DBLP-Y, IMDb, Query Log, and Oregon AS graphs). The y-axis is the normalized node overlap and is equal to $\frac{|V_{G_A} \cap V_{G_B}|}{\sqrt{|V_{G_A}| \times |V_{G_B}|}}$. As the

Network		|V|	|E|	NETSIMILE	FSM	EIG
arXiv	a-AstroPh	18,772	396,160	9	> 2 days	6
	a-CondMat	23,133	186,936	2	> 2 days	4
	a-GrQc	5,242	28,980	1	> 2 days	1
	a-HepPh	12,008	237,010	6	> 2 days	3
	a-HepTh	9,877	51,971	1	> 2 days	2
DBLP-C	d-vldb	1,306	3,224	1	15	1
	d-sigmod	1,545	4,191	1	28	1
	d-cikm	2,367	4,388	1	11	1
	d-sigkdd	1,529	3,158	1	42	1
	d-icdm	1,651	2,883	1	17	1
	d-sdm	915	1,501	1	7	1
DBLP-Y	d-05	39,357	79,114	1	2231	2
	d-06	44,982	94,274	1	2856	2
	d-07	47,465	103,957	1	4603	2
	d-08	47,350	107,643	1	9859	3
	d-09	45,173	102,072	1	9209	2
IMDb	i-05	13,805	130,295	1	> 2 days	3
	i-06	14,228	142,955	1	> 2 days	3
	i-07	13,989	133,930	1	> 2 days	2
	i-08	14,055	132,007	1	> 2 days	3
	i-09	14,372	128,926	1	> 2 days	2
Oregon AS	o-1	10,900	31,181	2	> 2 days	1
	o-2	11,019	31,762	2	> 2 days	1
	o-3	11,113	31,435	2	> 2 days	1
	o-4	11,260	31,304	2	> 2 days	1
	o-5	11,461	32,731	2	> 2 days	1
Query Log	ql-1	138,976	1,102,606	209	> 2 days	14
	ql-2	108,420	876,517	119	> 2 days	11
	ql-3	89,406	707,579	107	> 2 days	9
	ql-4	75,838	582,703	68	> 2 days	8
	ql-5	42,946	253,469	11	> 2 days	5

TABLE II. RUN TIMES (IN SECONDS, UNLESS OTHERWISE NOTED) OF NETSIMILE, FSM, AND EIG ON OUR REAL NETWORKS

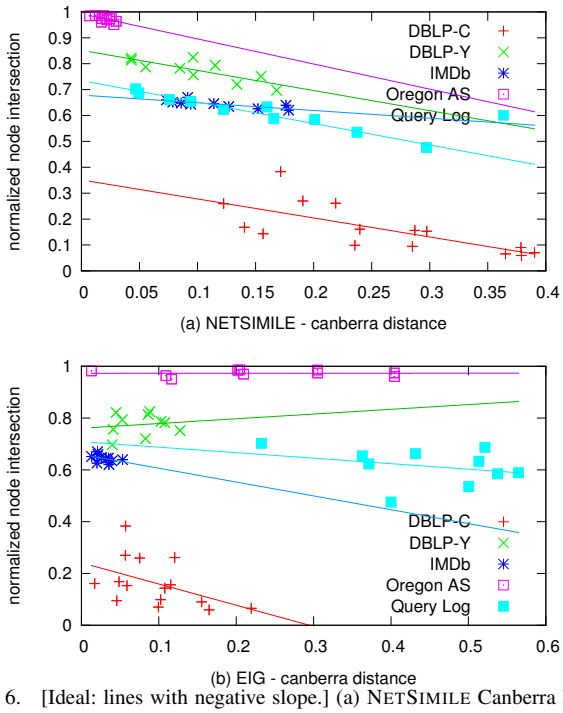


Fig. 6. [Ideal: lines with negative slope.] (a) NETSIMILE Canberra Distance on DBLP, IMDb, Oregon and QueryLog. (b) EIG Canberra Distance on the same networks. NETSIMILE is an effective measure for node overlap without any node-correspondence information. The lower the NETSIMILE Canberra Distance, the higher the normalized node intersection. This correlation does not hold for EIG. The points in both plots are along the fitted lines. For NETSIMILE (a), the root mean square (RMS) of residuals are $6.5E{-}2$ for DBLP-C, $2.6E{-}2$ for DBLP-Y, $9.0E{-}3$ for IMDb, $1.4E{-}2$ for Oregon AS, and $6.5E{-}2$ for Query Log. For EIG (b), the RMS of residuals are $8.2E{-}2$ for DBLP-C, $4.2E{-}2$ for DBLP-Y, $1.3E{-}3$ for IMDb, $1.2E{-}2$ for Oregon AS, and $6.7E{-}2$ for Query Log.

figure shows the lower the NETSIMILE Canberra Distance, the higher the normalized node intersection. This confirms our hypothesis that NETSIMILE can be used to gauge node-overlap between two graphs without node correspondence information. Figure 6(b) shows the same scatter plot, but computed using the EIG Canberra Distance approach. In this case, there is no correlation between node overlap and the distance. Due to its scalability issues, the FSM approach could not be computed on all the networks in Figure 6.

NETSIMILE as a Discontinuity Detector. Given a time-series of graphs $\{G_1, G_2, .., G_t\}$, can NETSIMILE detect any discontinuity (i.e. temporal outliers) in the data? To answer this, we utilize NETSIMILE Canberra Distance to compute the difference between graphs in a time series. For this experiment, we used data from Yahoo! IM and Twitter. For brevity, we only show the Yahoo! IM results; the other results were comparable.

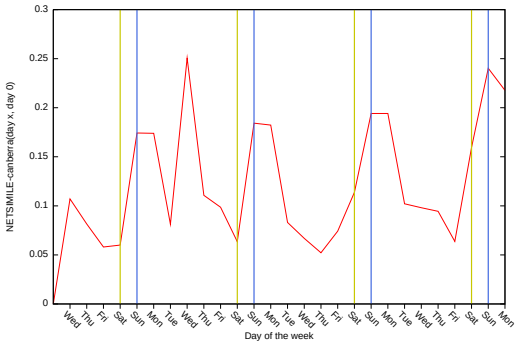
The Yahoo! IM communications dataset (http://sandbox.yahoo.com) spans 28 days and starts on Tuesday April 1, 2008. Each graph is a collection of instant messages (IMs) per day, with nodes representing IM users and links denoting communication events. The graphs are of varying sizes: number of nodes from 29K to 100K and number of edges from 80K to 280K. We computed the NETSIMILE normalized Canberra Distance between Day 0 (April 1, 2008) and the other 27 days. Figure 7(a) shows our results, with the x-axis representing days and the y-axis representing NETSIMILE (with normalized Canberra Distance) between Day 0 and the other 27 days. Figure 7(b) shows NETSIMILE (with normalized Canberra Distance) between Day 8 (April 9, 2008) and all the other days. As the figures illustrate, NETSIMILE detects the weekday vs. weekend discontinuities. It also detects a discontinuity on Wednesday April 9, 2008. The following event explains this discontinuity. Flickr announced that it will add video to its popular photo-sharing community[4] on April 8, 2008; but its news spread on April 9, 2008.[5] This event is reflected in the graph for April 9, 2008, where the number of connected components decreases by $4\times$ as the news about Flickr spreads among the IM users.

IV. RELATED WORK

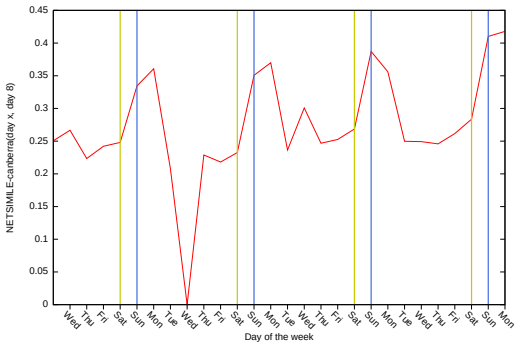
Assessing the similarity between two "objects" comes up in numerous settings, and so there exist similarity measures for various domains: distributions or multi-dimensional points [15], datacubes [16], and graphs, such as social [17], [18], information [19], and biological networks [20]. One aspect that separates NETSIMILE from most previous works (e.g. on graph isomorphism and graph matching) is that NETSIMILE does not require node correspondences nor does it attempt to match the graphs. A recent work, DELTACON [21] solves a much more restricted problem than NetSimile

[4] http://yhoo.client.shareholder.com/releasedetail.cfm?releaseid=303857

[5] http://searchengineland.com/flickr-launches-video-its-not-a-youtube-clone-13727



(a) NetSimile between each day and day 0 in Yahoo! IM



(b) NetSimile between each day and day 8 in Yahoo! IM

Fig. 7. NETSIMILE detects discontinuities in time-evolving graphs, and also captures weekly periodicities. (a) Distance of day 1, day 2, $\cdots$, day 27 IM graphs from day 0 (Tuesday April 1, 2008) IM graph. Weekdays are distinguished from weekends (yellow line= Saturday, blue line = Sunday). The peak on the 2nd Wednesday (April, 9, 2008) corresponds to a big Flicker announcement and a Microsoft offer to buy Yahoo!. (b) Distance of the other days from day 8 (April 9, 2008) IM graph. All the other days are distant from April 9, 2008.

since it assumes that the correspondences between the nodes of the networks are given (e.g., temporal who-emails-whom graph in a company). DELTACON uses belief propagation to measure network similarity. W.r.t. feature extraction, some of the features that have previously been used for network similarity are: frequent subgraphs [13], [22] and socially relevant features [18], [17]. Henderson et al. [6] propose a method for mining recursive structural features. NETSIMILE is easily extensible to incorporate these features. Li et al. [23] propose a classification approach of attributed graphs, which is based on global feature extraction. The weakness of that approach is that some features (e.g., eccentricity and shortest paths) are computationally expensive, and, thus, it is not scalable on large graphs. Also, the method is domain-specific and focuses on databases of graphs, such as chemical compounds, while our work is not domain-specific. NETSIMILE focuses on the extraction and aggregation of computationally inexpensive, *local* features supported by pertinent social theories that capture the nuances in the structural information of the given networks without assuming node-correspondences.

V. CONCLUSIONS

We introduced NETSIMILE, a novel, effective, size-independent, and scalable method for comparing large networks. NETSIMILE has three components: (1) feature extraction supported by relevant social theories, (2) feature aggregation, and (3) comparison. The heart of our contribution is in the first two components, where we discovered that (1) extracting structural features of the nodes and their egonets based on theories of Social Capital, Structural Hole, Balance, and Social Exchange, and (2) computing the moments of distributions of these structural features provides an excellent "signature" vector for a network. Through extensive empirical studies, we demonstrated that these "signature" vectors can be used to effectively and quickly assess the similarity of two or more networks without node-correspondence information or assumptions on node or edge overlaps.

REFERENCES

[1] N. Contractor, S. Wasserman, and K. Faust, "Testing multi theoretical, multilevel hypotheses about organizational networks: An analytic framework and empirical example," *Academy of Management Review*, vol. 31, no. 3, pp. 681–703, 2006.

[2] J. S. Coleman, *Individual interests and collective action: Selected essays*. Cambridge University Press, 1986.

[3] R. S. Burt, *Structural holes: The social structure of competition*. Harvard University Press, 1992.

[4] F. Heider, *The psychology of interpersonal relations*. Wiley, 1958.

[5] G. C. Homans, "Social behavior as exchange," *American J. of Sociology*, vol. 63, no. 6, pp. 597–606, 1958.

[6] K. Henderson, B. Gallagher, L. Li, L. Akoglu, T. Eliassi-Rad, H. Tong, and C. Faloutsos, "It's who you know: Graph mining using recursive structural features," in *KDD*, 2011, pp. 663–671.

[7] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables in stochastically larger than the other," *AMS*, vol. 18, no. 1, pp. 50–60, 1947.

[8] M. A. Stephens, "EDF statistics for goodness of fit and some comparisons," *JASA*, vol. 69, no. 347, pp. 730–737, 1974.

[9] A. Reka and Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, pp. 47–97, 2002.

[10] J. Leskovec, J. M. Kleinberg, and C. Faloutsos, "Graphs over time: densification laws, shrinking diameters and possible explanations," in *KDD*, 2005, pp. 177–187.

[11] P. Erdös and A. Rényi, "On random graphs I," *Publicationes Mathematicae Debrecen*, vol. 6, pp. 290–297, 1959.

[12] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[13] M. Berlingerio, F. Bonchi, B. Bringmann, and A. Gionis, "Mining graph evolution rules," in *ECMLL PKDD*, 2009, pp. 115–130.

[14] M. E. J. Newman, "The structure and function of complex networks," *SIAM Review*, vol. 45, pp. 167–256, 2003.

[15] S.-H. Cha, "Comprehensive survey on distance / similarity measures between probability density functions," *Int'l J. of Mathematical Models & Methods in Applied Sciences*, vol. 1, no. 4, pp. 300–307, 2007.

[16] E. Baikousi, G. Rogkakos, and P. Vassiliadis, "Similarity measures for multidimensional data," in *ICDE*, vol. 0, 2011, pp. 171–182.

[17] K. Faust, "Comparing social networks: Size, density and local structure," *Advances in Methodology and Statistics*, vol. 3, no. 2, pp. 185–216, 2006.

[18] O. Macindoe and W. Richards, "Graph comparison using fine structure analysis," in *IEEE SocialCom*, 2010, pp. 193–200.

[19] P. Papadimitriou, A. Dasdan, and H. Garcia-Molina, "Web graph similarity for anomaly detection," in *WWW*, 2008, pp. 1167–1168.

[20] H. Hu, X. Yan, Y. Huang, J. Han, and X. J. Zhou, "Mining coherent dense subgraphs across massive biological networks for functional discovery," *Bioinformatics*, vol. 21, pp. 213–221, 2005.

[21] D. Koutra, J. Vogelstein, and C. Faloutsos, "DeltaCon: A principled massive-graph similarity function," in *SDM*.

[22] M. Kuramochi and G. Karypis, "Finding frequent patterns in a large sparse graph," *DMKD*, vol. 11, no. 3, pp. 243–271, 2005.

[23] G. Li, M. Semerci, B. Yener, and M. J. Zaki, "Graph classification via topological and label attributes," in *MLG*, 2011.

Prepared by LLNL under Contract DE-AC52-07NA27344.