Suspicion scoring of networked entities based on guilt-by-association, collective inference, and focused data access[1]

Sofus A. Macskassy
New York University
44 W. 4th Street
New York, NY 10012
smacskas@stern.nyu.edu

Foster Provost
New York University
44 W. 4th Street
New York, NY 10012
fprovost@stern.nyu.edu

Abstract

We describe a guilt-by-association system that can be used to rank networked entities by their suspiciousness. We demonstrate the algorithm on a suite of data sets generated by a terrorist-world simulator developed to support a DoD program. Each data set consists of thousands of entities and some known links between them. The system ranks truly malicious entities highly, even if only relatively few are known to be malicious ex ante. When used as a tool for identifying promising data-gathering opportunities, the system focuses on gathering more information about the most suspicious entities and thereby increases the density of linkage in appropriate parts of the network. We assess performance under conditions of noisy prior knowledge of maliciousness. Although the levels of performance reported here would not support direct action on all data sets, the results do recommend the consideration of network-scoring techniques as a new source of evidence for decision making. For example, the system can operate on networks far larger and more complex than could be processed by a human analyst. This is a follow-up study to a prior paper; although there is a considerable amount of overlap, here we focus on more data sets and improve the evaluation by identifying entities with high scores simply as an artifact of the data acquisition process.

Contact:
Sofus A. Macskassy
Stern School of Business
Department of Information, Operations & Management Sciences
New York University
New York, NY 10012-1126, USA

Tel: 1-212-998-0584
Fax: 1-212-995-4228
Email: smacskas@stern.nyu.edu

Keywords: relational learning, network learning, machine learning, collective inference, guilt-by-association

Acknowledgments

This research was sponsored in part by the Air Force Research Laboratory, Air Force Materiel Command, USAF, under Agreement number F30602-01-2-0585. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of AFRL or the U.S. Government. This work was also funded in part by a grant from the New York Software Industry Association.

[1] This is a follow-up study to the work reported in Macskassy & Provost (2005).

Suspicion scoring of networked entities based on guilt-by-association, collective inference, and focused data access

Sofus A. Macskassy and Foster Provost

1 Introduction

This paper studies suspicion scoring in networked data: ranking entities by their estimated likelihood of being malicious. Various applications, ranging from law enforcement and counterterrorism to commercial fraud detection, can benefit from accurate rankings of entities by suspicion. We address suspicion scoring in networks of people (entities), linked by communications, meetings, or other associations (e.g., being in the same vicinity at the same time). Our system makes use of the simple-yet-ubiquitous principle of homophily (Blau, 1977; McPherson et al., 2001); social science research has shown repeatedly that a person is more likely to associate with people who share similar interests or characteristics.

Suspicion scoring based on networked data has been used successfully, although typically in an ad hoc manner, for commercial fraud detection. The "dialed digits" monitors discussed by Fawcett and Provost score an account highly if it calls the same numbers called by known fraudulent accounts (Fawcett & Provost, 1997); the "communities of interest" of Cortes et al. explicitly represent the network neighborhoods around telephone accounts as a basis for suspicion scoring (Cortes et al., 2001). We extend such methods by propagating suspicion through the association network, and conducting suspicion-based acquisition of additional data.

Homophily is the basis of a very simple guilt-by-association technique: estimate suspicion level by counting malicious associates. One problem with using this simple homophily-based guilt-by-association technique in large networks is that prior knowledge of maliciousness may be sparse. For many entities, no associates will be known to be either malicious or benign. However, if the association graph is well connected, then following linkages of associations is likely eventually to lead to at least one entity who is known or is strongly suspected to be malicious. Based on this idea, we overcome the problem of sparse knowledge by propagating suspicion scores through the association network until all suspicion scores stabilize. In particular, we use an adaptation of a relaxation labeling method which has been shown to yield good performance for hypertext classification (Chakrabarti et al., 1998).

Relaxation labeling works well if the association graph is well-connected. For intelligence data, one must consider the difference between the true association network and the network of *known* associations. The true association network may be known only partially. We address this partially via suspicion-based data acquisition, using current suspicion scores to acquire additional connections to improve the suspicion propagation. In a realistic setting, acquiring association links (involving accessing databases of other organizations, obtaining subpoenas for transaction records, surveillance, interviews, phone taps, etc.) is costly in terms of money, resources, legal issues, and public perception. We attempt to minimize costs by acquiring such "secondary data" only for the entities with the highest estimated suspiciousness. This heuristic works well in the data we have studied.

2 Guilt-by-association, Collective inference, and data acquisition

Our scoring algorithm consists of three main components listed in Table 1. The first two components are part of a network learning toolkit called NetKit-SRL (Macskassy & Provost, 2004). This open-source toolkit, written in Java 1.5, is publicly available and contains methods for learning patterns more complicated than simple guilt-by-association. The third component is a data acquisition wrapper which uses this toolkit in its inner loop.

1. A *relational classifier* which generates a suspicion score for a particular entity, p_i, given the known associations of p_i and the strengths of those association links.
2. A *collective inference* technique to propagate scores throughout the network.
3. An adaptive technique for *acquiring data* to increase the density of connections in the network.

Table 1: Guilt-by-association main components.

2.1 Relational Classifier

The relational classifier used in the study is a simple "relational neighbor" model, based on the principle of homophily and a first-order Markov assumption (Macskassy & Provost, 2003; Macskassy & Provost, 2004). The model estimates suspicion as the weighted sum of the suspicions of the immediate neighbors in the association network. Specifically, the score of entity p_i is:

$$s(p_i) = \frac{1}{Z} \sum_{p_j \in N_i} w_{i,j} \cdot s(p_j), \tag{1}$$

where N_i is the set of known associates of entity p_i and $w_{i,j}$ is the strength of the association between entities p_i and p_j—in our application defined as the number of times p_i and p_j have been known to interact. The score, $s(p_j)$, is

1. Acquire information on all entities initially known to be malicious.
2. Generate suspicion scores for all entities with unknown scores.
3. Get information on the top k entities not yet queried ($k = 50$ for this paper).
4. Generate new suspicion scores.
5. Repeat steps 3 and 4 until some stopping criterion is met (in this paper: either when all entities in the data set have been queried against or when we have run at least 25 iterations *and* have queried against all entities who are only connected to "known" malicious entities.[a])

[a] This is different from the original study in (Macskassy & Provost, 2005), where we always stopped after the 25^{th} iteration. We explain the reason for this change in Section 3.1.

Table 2: Data Acquisition algorithm.

the current suspicion score of entity p_j (note the similarity of our method, paired with the updating method described below, to Hopfield Networks (Hopfield, 1982) and Boltzmann machines (Ackley et al., 1985)). For entities whose status is known (benign or malicious), this is static—viz., 1 for "malicious" and 0 for "benign". Z is the sum of weights $w_{i,j}, p_j \in N_i$, to keep all scores between 0 and 1.

2.2 Collective Inference

When only a few malicious entities are known, there will be neighbors who (initially) have no value for $s(p_j)$. To deal with this scenario, first recognize that if we had estimates of the unknown scores, then we could apply the relational classifier to estimate $s(p_i)$. Second, the scores of p_i and p_j are clearly interrelated and estimating one will have an influence on the other. We therefore estimate all unknown scores simultaneously or "collectively" (Jensen et al., 2004). As it is not tractable to perform exact inference to estimate the full joint probability distribution over a large network, we use an approximation technique. In particular, we use an adaptation of relaxation labeling, based on the work of Chakrabarti et al. (1998). Relaxation labeling "freezes" the current estimated scores and then updates all estimates pseudo-simultaneously to generate new estimates. It does so repeatedly until the estimates converge. Unfortunately, this often leads to oscillation between two or more distinct sets of world-estimates. Therefore, we apply simulated annealing to enforce convergence. More formally:

$$s(p_i)^{(t+1)} = \alpha^{(t+1)} \cdot s(p_i)^{(t)} + (1 - \alpha^{(t+1)}) \cdot \left(\frac{1}{Z} \sum_{p_j \in N_i} w_{i,j} \cdot s(p_j)^{(t)} \right), \tag{2}$$

where t is the iteration step and $\alpha^{(t)}$ the temperature, with

$$\begin{aligned} \alpha^{(0)} &= c & (3)\\ \alpha^{(t+1)} &= \beta \cdot \alpha^{(t)}, & (4) \end{aligned}$$

where c is a starting constant and β is a decay constant. We use the values 1 and 0.99 for c and β, respectively, and stop after 100 iterations.

Relaxation labeling and other collective inference techniques require initial estimates to bootstrap the inference. We initialize scores to 1 for initially "known" malicious entities (and freeze them) and 0.01 for the rest. If we had had knowledge of benign entities, we would have initialized (and frozen) those scores to 0.

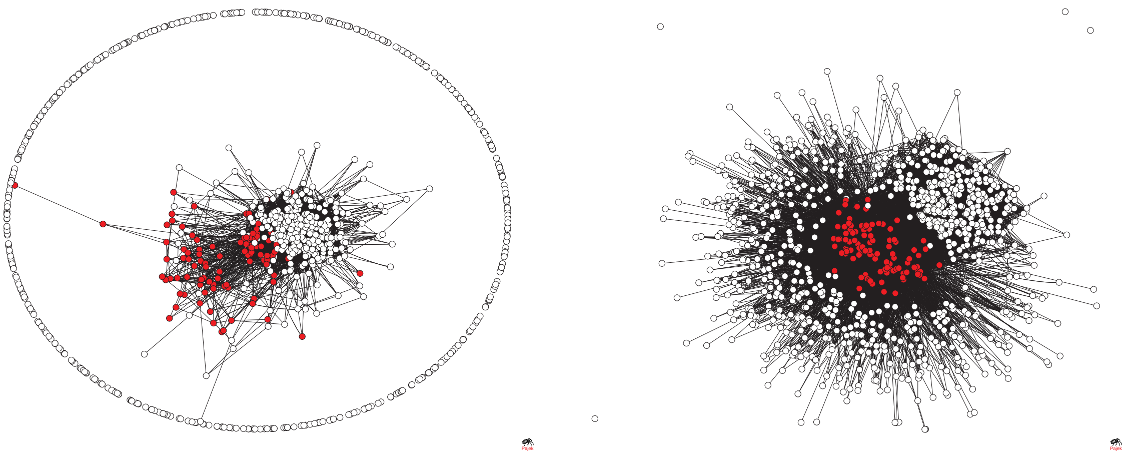
2.3 Data Acquisition

As discussed above, it may be possible (at a cost) to augment the association network incrementally. The strategy used in this paper is shown in Table 2, which acquires additional information (associations and possibly unknown associates) about the most suspicious entities.

3 Case Study

Using simulated data, we evaluate whether this method can produce accurate rankings of entities by suspicion scoring. Specifically: are the highest-scoring entities predominantly malicious? Our study is threefold, assessing: (1) the initial rankings, (2) the improvement as we acquire more information, and (3) how good the initial knowledge of maliciousness must be (i.e., how much noise can be tolerated).

This study, while considering the same problems as the earlier study in this domain, differs in two aspects: (1) in the original study, entities who had not yet been queried were included in all evaluations. This is unrealistic as it is unlikely that an action would be taken regarding someone before all available information on that person had been obtained (disregarding issues of timeliness). (2) Here we report results on more data sets (17, a superset of those used in the initial report).



(a) using only primary data. (b) after querying all “known” malicious entities.

Figure 1: 5057 linkage data. Red (shaded) dots are entities who are “known” to be malicious. We need to rank the white dots. The oval in (a) are entities who either are not yet known or for which we have no initial information (links). The graphs are showing 10K edges drawn at random from (a) 77K possible edges and (b) 142K possible edges.

3.1 Score-based Evaluation

Notice that the data acquisition methodology outlined in Table 2 keeps querying for more data until all entities that are only connected to the “known” malicious entitites have been queried against. The reason for this is due to a scoring-issue that was identified during an evaluation of the original methodology (Macskassy & Provost, 2005).

The issue is this: consider the case where querying the “known” entities results in links to K new entities. These new entities are by construction linked only to the “known” malicious entities and they therefore all have the same (maximal) suspicion score of 1. Figures 1(a)-(b) show this scenario on a small data set. Initially, as shown in Figure 1(a), all singletons (in the large oval) are either not yet known or have no information (links) associated with them. After querying all “known” entities, we have the graph shown in Figure 1(b). All entities are now connected (the few singletons present are due to the sub-sampling of edges), where all the singleton entities in 1(a) are now connected to all the “known” malicious entities—these are shown in the crescent of white dots, whereas the cluster of white dots in the upper right contains the dots that were already connected in 1(a).

Continuing with the example scenario the system will now, at each iteration, query 50 randomly chosen entities from this list of K newly connected entities. Although the system slowly chips away at these K entities, the data will at the first couple of iterations still contain a large number of entities who are still connected only to “known” entities because of the initial supply. These will all have the same maximal score (by construction) and cannot be distinguished by their score. This poses a serious problem for evaluation: clearly before taking any other action it is appropriate to obtain all available data on an entity who is suspicious by construction. Moreover, truly malicious entities may be unlikely to have the maximal score of 1, because they will interact with some non-malicious entities.

As with the original study, the system is evaluated using two metrics. The first metric used is the Area Under the ROC Curve (AUC), which is equivalent to the Mann-Whitney-Wilcoxon statistic and computes the probability that a randomly selected malicious entity would be given a higher suspicion score than a randomly selected benign entity. Therefore, an AUC of 0.5 means that a scoring is no better than random guessing (the ranking is well shuffled); a value of 1 indicates a perfect ranking—all the malicious entities get higher scores than all the benign entities. We include the “problematic” scores when evaluating the overall ranking performance of the system using AUC. Thus, the AUC results can be regarded as conservative, but do give an assessment of the overall ranking of all entities.

The second metric used is the fraction of the top-100 highest-scoring entities that truly are malicious. This evaluates the system under the assumption that an intelligence analyst will consider the highest scoring entities for possible further investigation. The analyst will have a processing capacity, for which we chose 100 cases. For this evaluation, we disregard the maximal-by-construction scores, as justified above. Specifically, we remove entities for which the secondary data has not been queried.

3.2 Data

There are many varieties of intelligence data—no single comparison of classified and synthetic data will be comprehensive. The data we use for this paper were generated by a flexible simulator as part of a DoD program to assess the feasibility of large-scale information systems to help identify terrorists. The synthetic data generated by this simulator are moderately sized examples of structured data representing terrorists and benign entities who are conducting activi-

	World Parameters		Primary Data			
Data set	Size	Number malicious	Size	True malicious	False malicious	Noise
5048	13756	2173	9601	226	0	0.000
5049	988	269	766	62	0	0.000
5055	9967	711	4916	166	0	0.000
5057	1011	274	497	101	0	0.000
5062	9897	2852	3745	500	0	0.000
5069	9907	2893	7374	520	0	0.000
5065	16046	7574	5907	1264	82	0.061
5066	16743	8002	5332	1284	173	0.119
5068	14209	5482	8352	1253	181	0.126
5058	100301	7350	40316	886	189	0.176
5063	9998	2823	4825	828	276	0.250
5067	9970	2860	7561	383	130	0.253
5046	13236	1484	4212	143	52	0.267
5053	1002	274	799	116	216	0.651
5056	1022	300	510	99	218	0.688
5052	986	278	763	82	210	0.719
5050	1008	316	692	103	336	0.765

Table 3: Characteristics of synthetic data sets, sorted and "grouped" by noise. In the World Parameters, Size refers to the number of entities in the true synthetic world and Number malicious refers to the total number of truly malicious entities. In the Primary Data, Size refers to the number of entities known initially; True malicious refers the number of entities tagged as "malicious" who truly were malicious in the world and False malicious refers to the number of entities falsely tagged as "malicious". The error rate (Noise) of these labelings ranges from none (0) to very high as seen in the bottom group of data sets. Note that step 1 in Table 2 above must query the secondary database for information on all "false malicious" as well as all "true malicious" entities.

ties over an extended period of time. The data are contained wholly in a single data source (although costly secondary access is simulated) and are self-consistent, neither of which is reliably true of classified data. However, the data do replicate a range of noisy and poorly observed activities, and the entities are intentionally obscured to simulate lack of knowledge, obfuscation, poor data-entry practices, multiple identities, etc. Nonetheless, we do not claim that the data fully replicate the limitations of actual data collection, aggregation and enrichment that the intelligence community routinely experiences. We use data sets generated for the purposes of DoD program evaluation. We do not create data sets ourselves for this paper.

One run of the simulator generates three databases:

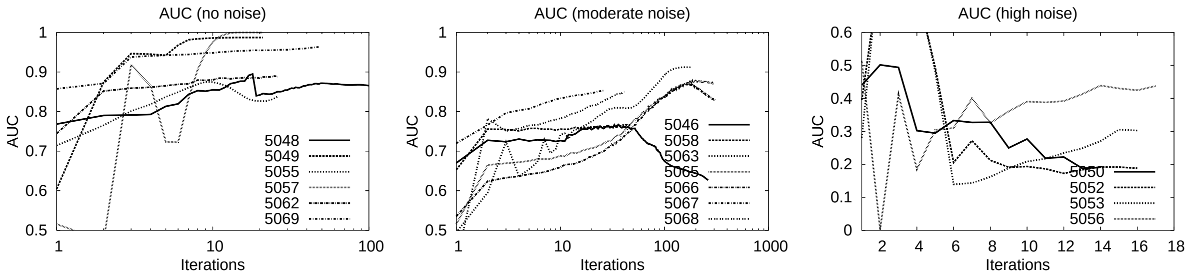
1. "primary" data that are known ex ante. These often are sparse and may contain partial (or no) information on any particular entity or group;
2. "secondary" data consisting of information that only can be acquired by querying (theoretically at a cost) to get information on a particular entity;
3. "truth" data, for evaluation, consisting of what really happened in the world.

The first two databases together reflect what possibly can be observed. They are potentially corrupt and contain only a subset of the complete truth. Further, the data never give hard evidence that an entity is benign, and therefore we only "know" about some malicious entities—those who are known to belong to one or more terrorist groups. Sometimes this knowledge is wrong. We evaluate the suspicion scoring on 17 data sets, whose characteristics are shown in Table 3.

3.3 Results

In order to address how noise affects performance, we group the data sets into three categories: no noise (5048, 5049, 5055, 5057, 5062, 5069), low to moderate noise (5046, 5058, 5063, 5065, 5066, 5067, 5068), and very high noise (5050, 5052, 5053, 5056).

Figures 2(a)-(c) show, for each category, how the system performed throughout its data acquisition run. Figure 2(a) shows a wide range of performances from perfect (5057) to just above AUC $= 0.8$ (5055). In all cases, we see that performance increases as we gather more data (except for dips on both 5057 and 5048). We also see that guilt-by-association is able to perform much better than random ranking (AUC $= 0.5$). Figure 2(b) shows the performance of the suspicion scores on the moderate-noise data sets. Again, as we query the secondary data the performance quickly improves to achieve AUC values in the range $0.8 - 0.9$, the only exception being 5046, which starts degrading around iteration 50. It is noteworthy that the system overcomes the initial noisy labels, showing that it is robust to even moderate noise where 1 out of 4 entities were mistakenly judged to be malicious. This is because "malicious" entities in these data communicate just as much as "benign" entities, but to far fewer entities. Hence their associative strength to other "malicious" entities is much stronger. Figure 2(c) shows the performance of the suspicion scoring on the high-noise data. In this case, more than half of the entities tagged as "malicious" are actually benign. Initially we



(a) no noise (b) moderate noise (c) very high noise

Figure 2: AUCs using active data acquisition. Notice the different scales between (a)-(b) and (c).

		Iteration						
	Data	3	4	5	10	15	20	Last
No Noise	5048	14	52	97	100	100	100	100
	5049	22	37	70	100	100	100	100
	5055	3	4	12	45	100	100	100
	5057	19	21	63	100	100	100	100
	5062	25	52	91	100	100	100	100
	5069	25	60	95	100	100	100	100
Moderate Noise	5046	8	22	43	87	98	100	100
	5058	6	29	51	97	96	94	91
	5063	21	21	38	77	80	80	99
	5065	42	57	76	97	96	95	95
	5066	44	58	72	98	94	94	95
	5067	26	54	81	100	100	100	100
	5068	32	49	62	97	96	97	98
High Noise	5050	37	33	37	10	12	-	12
	5052	27	43	34	8	7	-	7
	5053	23	40	48	15	15	-	14
	5056	28	27	42	16	28	-	32

Table 4: How many truly malicious entities were in the top 100 after iterations 3,4,5,10,15,20 and after the last iteration. We start at iteration 3 because that is the first iteration in which we have scores for queried entities that were not initially "known". Iteration 3 therefore only contains 50 entities. Further, all high noise data sets were so small that they stopped at iteration 16.

see high variability, but the scores deteriorate quickly, and all end up with ranking much worse than random (note the different scales, vertical and horizontal), because the algorithm is actually propagating knowledge non-maliciousness.

Figure 2 only tells part of the story. For an analyst, knowing that the system can achieve an AUC of 0.9 does not necessarily mean that the system is useful. Although the system, in general, will rank suspicious entities higher, when considering 10000 entities, the top 100 could potentially be primarily benign with the next 900 being primarily malicious. Albeit unusual, this would achieve a relatively high AUC, but not be very useful for analysts who only have time to look at a select few entities.

Usually for rankings such as suspicion scorings, the density of entities of interest is highest at the very top of the list, especially if the scores are estimated probabilities of membership in a class (e.g., malicious entity), and so the top of the list contains the entities with the highest estimated probabilities of being suspicious. For a given AUC value, how dense one expects the top of a ranking to be depends primarily on the marginal probability of entities of interest in the data (this and related issues are treated in detail elsewhere (Provost & Fawcett, 2001)). An AUC of 0.9 may have 99 truly malicious entities in the top 100 highest-suspicion entities, or it may have 10. In either case, the system may be useful to an analyst, depending on the application and how the ranking will be used (e.g., as a primary basis for action versus as an alternative source of evidence to augment existing practices[2]).

To illustrate the effectiveness of the scorings for these data sets for a particular threshold, we analyze the number of truly malicious entities at the top of the suspicion rankings. If we look across the 17 data sets, we can ask how many truly malicious entities are among the top 100 most suspicious.

Table 4 shows the results for the 17 data sets, grouped by their noise level. The evaluation starts at Iteration 3 because Iterations 1 and 2 do not yet contain scores on entities that have been queried and are not part of the "known" entities. Iteration 3 is therefore based only on the 50 entities which have been queried after the initial querying of all "known" entitites. Iteration 4 contains 100 queried entities and so on.

[2] For example, White and Fournelle (2005) show that an early version of our suspicion scores are an effective prefilter for their CADRE analysis system for link discovery.

The table shows quantitatively what Figure 2 was telling us: In the no-noise group, we get very high density already at iteration 5, and perfection at iteration 10 (5055 taking longer to get there). By iteration 15 and on, the system is 100% accurate on all no-noise data sets. We see a similar behavior for the moderate data sets, where iteration 10 already has very good accuracy for most of the data sets (5063 being the worst, with only 77 out of 100 truly being malicious), and nearly perfect accuracy on all data sets by the last iteration. Notice that we get 100% accuracy on 5046 although its AUC was only around 0.6 at this final iteration. Finally, we see in the very large noise group that by the final iteration the system has very few malicious entities in the top 100. Remember that the absolute numbers (e.g., "precision" of 80 out of 100) reflects the marginal probability of being malicious in a particular data set; the ROC curve is independent of this probability, which is one reason why 5063 and 5046, although having very similar ranking ability (AUC) at iteration 20 have very different precision for a fixed threshold (Provost & Fawcett, 2001).

4 Limitations

The system we described here has notable limitations. We have assumed substantial prior knowledge: of entities, of links between them, of maliciousness. We have shown some robustness to the knowledge of maliciousness, but have not systematically explored robustness along other dimensions. Moreover, collecting the data to build such a network is a considerable effort, and it would make sense to consider network construction in tandem with the system that would make use of the network.

Relatedly, we have considered network-based suspicion scoring in isolation. In reality, network-based scoring would be one source of evidence, combined for example with "profile"-based scoring. We conducted a preliminary investigation into augmenting the scoring by setting initial priors based on uncertain-but-better-than-random knowledge (as from a profiling system). We found that priors had little-to-no effect due to the algorithm's dominance by the scores propagated from the static labels (Macskassy & Provost, 2005). This is a problem which can affect many collective inference techniques. In retrospect, it appears necessary to integrate closely the use of profiling information with the network scoring (cf. (Macskassy & Provost, 2004)). This issue is likely to affect many collective inference techniques and needs to receive more attention.

5 Final Remarks

We described and evaluated a guilt-by-association system for generating suspicion scores based on entities' known associates. The system is notable for several reasons. First, it is able to generate remarkably good rankings even when very few entities are known to be malicious. Second, it can be relatively robust even to moderate noise in these prior labels. Third, it works remarkably well considering that it only uses prior labels and the network, but no profiling. Finally, it can be used as a data gathering tool not only to perform focused data acquisition of suspicious entities, but also to further improve its ranking—and in the process it often learns about suspicious entities that were not initially in the database.

References

Ackley, D. H., Hinton, G. E., & Sejnowski, T. J. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, *9*, 147–169.

Blau, P. M. (1977). *Inequality and Heterogeneity: A Primitive Theory of Social Structure*. New York: Free Press.

Chakrabarti, S., Dom, B., & Indyk, P. (1998). Enhanced Hypertext Categorization Using Hyperlinks. *Proceedings of the ACM SIGMOD International Conference on Management of Data* (pp. 307–318).

Cortes, C., Pregibon, D., & Volinsky, C. T. (2001). Communities of Interest. *Proceedings of Intelligent Data Analysis (IDA-2001)* (pp. 105–114).

Fawcett, T., & Provost, F. (1997). Adaptive fraud detection. *Data Mining and Knowledge Discovery*, *3*, 291–316.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *National Academy of Sciences*, *79*, 2554–2558.

Jensen, D., Neville, J., & Gallagher, B. (2004). Why Collective Inference Improves Relational Classification. *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 593–598).

Macskassy, S. A., & Provost, F. (2003). A Simple Relational Classifier. *Proceedings of the Second Workshop on Multi-Relational Data Mining (MRDM-2003) at KDD-2003* (pp. 64–76).

Macskassy, S. A., & Provost, F. (2004). *Classification in Networked Data: A toolkit and a univariate case study*. Technical Report CeDER Working Paper 04-08. Stern School of Business, New York University.

Macskassy, S. A., & Provost, F. (2005). Suspicion scoring based on guilt-by-association, collective inference, and focused data access. *International Conference on Intelligence Analysis*.

McPherson, M., Smith-Lovin, L., & Cook, J. M. (2001). Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology*, *27*, 415–444.

Provost, F., & Fawcett, T. (2001). Robust classification for imprecise environments. *Machine Learning*, *42*, 203–231.

White, J. V., & Fournelle, C. G. (2005). Threat detection for improved link discovery. *International Conference on Intelligence Analysis*.