Seeing Red: Locating People of Interest in Networks

Pivithuru Wijegunawardana[1], Vatsal Ojha[2], Ralucca Gera[3]*, and Sucheta Soundarajan[1]

[1] Syracuse University, Department of Electrical Engineering & Computer Science, ppwijegu,susounda@syr.edu
[2] Dougherty Valley High School, San Ramon, California, vatsalojha@gmail.com
[3]Naval Postgraduate School, Department of Applied Mathematics, Monterey, CA, RGera@nps.edu

Abstract. The focus of the current research is to identify people of interest in social networks. We are especially interested in studying dark networks, which represent illegal or covert activity. In such networks, people are unlikely to disclose accurate information when queried. We present REDLEARN, an algorithm for sampling dark networks with the goal of identifying as many nodes of interest as possible. We consider two realistic lying scenarios, which describe how individuals in a dark network may attempt to conceal their connections. We test and present our results on several real-world multilayered networks, and show that REDLEARN achieves up to a 340% improvement over the next best strategy.

Keywords: multilayered networks, sampling, lying scenarios, nodes of interest

1 Introduction and Motivation

Today's complex environment requires decision makers to act in an overwhelmingly rich network environment, often based on partial information of that network. It is often desirable to locate "people of interest" (POI) residing in such networks while they conceal themselves or others. Our work was motivated by study of terrorist networks, which can be modeled multilayered networks where each layer is defined by a different relationship (e.g., relationships indicate organizations these terrorists belong to, the schools or trainings they went to, kinship, recruiting and so on.)

In this paper, we consider the goal of sampling a 'dark' network (i.e., a network representing illegal or covert activity) in such a way that we observe as many POIs as possible. We present REDLEARN, a novel learning-based algorithm for sampling networks with the goal of finding as many POIs as possible. We show that in cases where the POIs exhibit homophily (i.e., are likely to be connected to other POIs), a simple strategy of choosing the node with the most POI neighbors works well. However, in the more realistic scenario where POIs hide their connections with other POIs, REDLEARN shows outstanding performance, beating the next best strategy by up to 340%.

Problem Definition: We refer to nodes representing POIs as 'red' nodes, and other nodes as 'blue', giving us a purple network. We assume that there is an unobserved,

*Ralucca Gera would like to thank the DoD for partially sponsoring the current research.

underlying graph $G = (V,E)$, in which each node $v \in V$ has color $C_v \in \{red, blue\}$. We begin with having knowledge of only one red node in G.

To increase our observation of the network, we place *monitors* on nodes. A monitor tells us (1) the true color of the node being placed on, (2) the true neighbors of that node, and (3) the colors of the node's neighbors, possibly with inaccuracies. For example, placing a monitor on a suspected terrorist could represent determining whether that person is actually a terrorist, determining who his or her e-mail or phone contacts are, and questioning the individual about whether those neighbors are themselves terrorists. Naturally, some individuals may lie about the colors of those neighbors.[1]

We assume that we are given a budget of b monitors, and can place those monitors on any node that has been observed. In the first step, we must place a monitor on the initially observed node. We then place a monitor on any node that has been observed as a neighbor of a previously-monitored node.

Related Work: Our work is related to work on analyzing dark networks, a special type of social network [4]. A dark network is network that is illegal and covert [14], whose members are actively trying to conceal network information even at the expense of efficiency [4], and the existing connections are used infrequently [14]. Because a dark network is deceptive by nature, we examine the lying methodologies along with the discovery methods in looking for the POI.

There are a multitude of sampling techniques for network exploration, including random walks ([3], [11], [13]), biased random walks ([9]), or walks combined with reversible Markov Chains([2]), Bayesian methods([8]), or standard exhaustive search algorithms like depth-first or breadth-first searches, such as [1, 5, 6, 7, 12]. However, these methods generally do not use node attributes.

2 Proposed Method: REDLEARN

A monitor placement strategy is an incremental sampling strategy. A *monitored node* is a node with a monitor placed on it. At each step, the placement of the next monitor is determined based on the observed topology of the graph, known colors of nodes (observed by monitors placed directly on those nodes), and the stated colors of monitored nodes' neighbors (i.e., for each neighbor of a monitored node, whether the monitored node said that that neighbor was red or blue).

We now describe several natural monitor placement strategies as comparison algorithms in our experiments.

Smart Random Sampling (SR): In each step, the Smart Random Placement strategy places a monitor on a random unmonitored node.

Red Score (RS): If a node v reports its neighbor u as red, the score associated with node u is increased by one, making it more suspicious. This strategy selects the node with highest red score to place the next monitor.

Most Red Say Red (MRSR): The MRSR strategy places a monitor on the node with the greatest number of red neighbors who report it as a red node.

[1] We consider two realistic 'lying scenarios'; these are described in Section 2.1.

Most Red Neighbors (MRN): The MRN placement strategy places a monitor on the node with the most known red neighbors. This strategy would likely work best in a network with high homophily.

2.1 REDLEARN: A Learning Based Monitor Placement Strategy

When determining which node v to place the next monitor on the strategies above consider the colors of v's neighbors and/or the color that each of v's monitored neighbors reported, the presence of homophily, and the reported color of the neighbors.

To overcome these dependencies, we propose REDLEARN, a learning based monitor placement strategy. Our goal is to predict the probability of a node v being red ($P(v = R)$) based on the observed network structure and what v's neighbors say about v. We model this as a two class classification problem, but rather than looking at the assigned label (Red or Blue), we are more interested in finding $P(v = R)$. Once these probabilities are determined, REDLEARN places the next monitor on the node with the highest such probability.

Features: Table 1 describes the set of features used in our learning based monitor placement algorithm. There are two types of features: (*a*) Network structure-based features $(1,2,3)$, and (*b*) Neighbor answer-based features $(4,5,6,7,8)$.

Table 1: Classification features for REDLEARN. Consider a node v with neighbors $N(v)$

	Feature	Description
(1)	Number of Red Neighbors	$\lvert\{u \in N(v) \lvert c_u = R\}\rvert$
(2)	Number of Blue neighbors	$\lvert\{u \in N(v) \lvert c_u = B\}\rvert$
(3)	Number of Red triangles if v is red	$\left\lvert\{u,w \in N(v) \lvert u \in N(w) \cap w \in N(u) \cap c_u = c_w = R\}\right\rvert$
(4)	Red score	$\lvert\{u \in N(v) \lvert (u \; says \; R)\}\rvert$
(5)	Number of Red neighbors saying red	$\lvert\{u \in N(v) \lvert (u \; says \; R) \cap c_u = R\}\rvert$
(6)	Number of red neighbors saying blue	$\lvert\{u \in N(v) \lvert (u \; says \; B) \cap c_u = R\}\rvert$
(7)	Number of blue neighbors saying red	$\lvert\{u \in N(v) \lvert (u \; says \; R) \cap c_u = B\}\rvert$
(8)	Number of blue neighbors saying blue	$\lvert\{u \in N(v) \lvert (u \; says \; B) \cap c_u = B\}\rvert$
(9)	Inferred probability of being red	$P^I(v = R)$

Inferred probability of being red: We formulate four different probabilities to measure the trustworthiness of colors given by differently colored nodes (i.e., whether a monitored node lies or is honest about its neighbors' colors). Consider a node v which was discovered through a monitor placed on node u. Equation 1 shows how to calculate $P(v = R | color(u) \wedge color(u \; says \; v))$ when $v = R$, $u = R$ and u says v is red. Other probabilities can be calculated by changing components of this equation as appropriate.

$$P(v = R | (u = R) \wedge (u \; Says \; R)) = \frac{|\{(v = R) \cap (u = R) \cap (u \; says \; R)\}|}{|\{(u = R) \cap (u \; says \; R)\}|} \tag{1}$$

Given a node v, we calculate the inferred probability, $P^I(v = R)$ using equation 2.

$$P^I(v = R) \; = \; \frac{\sum_{u \in N(v)} P(v = R | color(u) \wedge color(u \; says \; v))}{|N(v)|} \tag{2}$$

The training data for this classification problem comes from the monitors placed so far and observed true colors. We predict $P(v = R)$ for each unmonitored node. We use logisitic regression as the classification algorithm in our experiments.

Algorithm 1 Learning based monitor placement

```
procedure LEARNING(start,budget)
    G ← Graph
    G.add(start), G.add(N(start))                    ▷ Starting node and neighbors
    while budget>0 do
        Monitors ← list of monitored nodes in G
        TrainingData ← feature vectors for Monitors
        Train classifier using TrainingData
        NotMonitors ← list of not yet monitored nodes in G
        for v ∈ NotMonitors do
            Get feature vector for v
            P(v=R) ← predict v's probability of Red using learning model
        Choose node v with maximum P(v = R) from NotMonitors
        budget ← (budget − 1)
        Use v as next monitor
```

3 Experimental Set Up

3.1 Datasets

PokeC Network: The PokeC network is part of a Slovenian online social network.[2] Each node has some number of associated user attributes (e.g., age, region, gender, interests, height etc.). We use a sample of this network containing all nodes in the region "kosicky kraj, michalovce". This sampled network contains $26,220$ nodes and $241,600$ edges. We assign node colors based on two different node attributes: *age* (a node with age in the range 28-32 is marked red, and blue otherwise, giving 1736 red nodes) and *height* (a user of height less than 160 *cm* is marked red, giving 1668 red nodes).

Noordin Top Network is a terrorist network with 139 nodes and 1042 edges depicting several types of relationships between them ('Noordin Top' is the name of the leader of this network).[3] [10]. In this network, every node is a terrorist , and POIs are those who communicate using some particular communication medium. We have identified five different communication mediums, and label nodes that use them as POIs: electronic (9 red nodes), print media (5 red nodes), support materials (9 red nodes), video (11 red nodes) and communication medium unknown (18 red nodes) .

Both networks have high homophily for red nodes (red nodes tend to be connected to each other). However, in a dark network where red nodes are actively trying to hide

[2] Obtained from `http://snap.stanford.edu/data/`.

[3] Obtained from `https://sites.google.com/site/sfeverton18/research/appendix-1`.

their presence, these nodes would conceal the existence of such connections (for example, instead of using their normal cell phone to make calls to other red nodes, a red node might use a burner phone for such calls). To account for this, we also consider versions of our datasets where all connections between red nodes are removed. Note that this type of network presents a much more challenging setting, as one cannot simply rely on homophily to find red nodes.

3.2 Lying Scenarios

In absence of ground truth, we formulate lying scenarios: we assume the existence of a hierarchy among the nodes, where nodes are more likely to lie to protect those above them in the hierarchy. We assume that the red nodes are fully aware of the hierarchy, blue nodes may or may not be aware, and that nodes may lie not only about the color of red nodes (i.e., lie to protect POIs), but also about the color of blue nodes (i.e., as a distraction).

Consider nodes u and v, where $u, v \in Edges$. The probability that u lies about v, $P(u\ lie\ v)$ depends on: (1) The color of u (C_u) and color of v (C_v),(2) The inherent honesty of u (H_u), where higher H values indicate that u is more predisposed to telling the truth and (3) The hierarchical position of u (L_u) relative to the position of v (L_v).

Table 2: The probability that node u lies about node v's color $P(u\ lie\ v)$ depending on u's and v's colors and lying scenarios

	LS1:Blue nodes know about red nodes		**LS2:Blue nodes don't know about red nodes**	
U/V	**Red**	**Blue**	**Red**	**Blue**
Red	Equation 3	Equation 4	Equation 3	Equation 4
Blue	Equation 3	Equation 4	1.0	0.0

The probability u will lie about a red node: where $\frac{L_v}{L_u}$ indicates how far above v is in the hierarchy compared to u and $1 - H_u$ is probability that u will lie.

$$P(u\ lie\ v|v = Red) = \min\{(1 - H_u) * \frac{L_v}{L_u}, 1\} \tag{3}$$

The probability u will lie about a blue node depends on u's honesty and is calculated as $(1 - H_u)$:

$$P(u\ lie\ v|v = Blue) = (1 - H_u) \tag{4}$$

We perform 25 runs of each monitor placement strategy, varying the honesty assignment and the colors that nodes say about neighbors between runs. In each run, we begin with a randomly selected red node and we consider budgets up to half the number of nodes in the network.

The honesty of each node is drawn from a normal distribution, $h \sim \mathcal{N}(0.5, 0.125)$. In the Noordin Top network, the ground truth hierarchy scores are Strategist (score 5),Commander; Religious Leader (score 4),Trainer/instructor; Bomb maker; Facilitator; Propagandist; Recruiter (score 3), Bomber/fighter; Suicide Bomber; Courier; Recon/Surveillance (score 2) and unknown (score 1). In the PokeC network, we set the hierarchy score to be the degree of the node.

Given a particular lying scenario, a monitored node u lies about a neighbor v's color with probability $P(u\ lie\ v)$ as shown in Table 2.

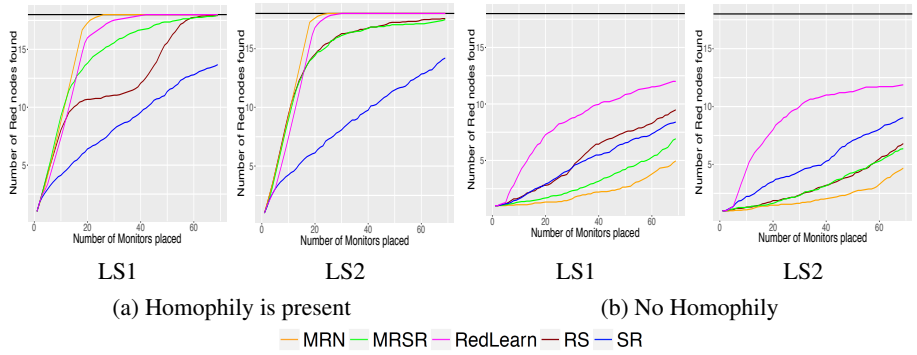

Fig. 1: Comparison of monitor placement strategies on the NoordinComs4 network. LS1:All nodes aware of red nodes. LS2: Only red nodes aware of red nodes. The black line indicates the total number of red nodes present in the network.

4 Results and Analysis

As an example, Figure 1 shows results on the NoordinComs4 network with edges between red nodes (left two plots) and without (right two plots). When there is homophily, the problem becomes easy, and the simple strategy of monitoring the node with the most red neighbors (MRN) is best. However, note that in both lying scenarios, REDLEARN is close behind the MRN strategy (because it needs time to train, it doesn't quite match the performance of MRN). However, we see from the right two figures that when edges between red nodes are removed, the MRN strategy performs very poorly. In this setting, REDLEARN performs much better than all comparison methods: it is able to learn the patterns and structural characteristics of red nodes, and by incorporating what neighbors say about a node, achieves strong performance.

Due to space constraints, we summarize results by showing the percentage of red nodes found from each monitor placement strategy for other networks in Table 3. We see similar patterns across all networks: when there are edges between red nodes, it is enough to select the node with the most red neighbors; but when these edges are concealed, REDLEARN is the clear winner. Even when there are edges between red nodes, REDLEARN usually achieves performance close to the MRN strategy.

5 Conclusions and Further Directions

By nature, members of dark networks conceal information, but while deceptive and sparse, these networks are still structured. To exploit these properties, we created REDLEARN, a learning-based method for locating People of Interest in dark networks. REDLEARN uses features from simpler methods and learns how to identify red nodes in networks. We showed that REDLEARN outperforms the other methods in cases where one cannot rely on homophily to identify red nodes.

In our future work, one interesting direction is to consider the dynamicity of the network (both on the edge and node rate of birth and retirement), as well as a more sophisticated model of the concealed nodes and relationships.

Table 3: Comparison of the percentage of red nodes found from each monitor placement strategy. Budgets include Low (10% of the nodes), Medium (25% of the nodes), and High (50% of the nodes). These networks exhibit homophily: edges between red nodes have not been removed.

(a) Lying Scenario 1, original data (with homophily)

	Low Budget					Medium Budget					High Budget				
Network/ Strategy	RS	RDLRN	MRN	MRSR	SR	RS	RDLRN	MRN	MRSR	SR	RS	RDLRN	MRN	MRSR	SR
NrdnComs1	28	74	**97**	52	32	43	97	**100**	77	51	97	**100**	**100**	92	75
NrdnComs2	42	37	**62**	48	42	61	72	**100**	72	55	99	93	**100**	91	81
NrdnComs3	33	63	**83**	52	27	59	89	**100**	77	46	**100**	**100**	**100**	97	73
NrdnComs4	54	60	**70**	66	28	63	98	**100**	90	48	**100**	**100**	**100**	**100**	76
NrdnComs5	34	67	**84**	52	32	43	**91**	**91**	75	46	88	**91**	**91**	86	67
PokeC age	5	14	**22**	20	7	15	43	**47**	39	21	48	**73**	68	62	47
PokeC height	14	14	21	**23**	11	36	32	**48**	47	28	**74**	64	73	69	54

(b) Lying Scenario 2, original data (with homophily)

	Low Budget					Medium Budget					High Budget				
Network/ Strategy	RS	RDLRN	MRN	MRSR	SR	RS	RDLRN	MRN	MRSR	SR	RS	RDLRN	MRN	MRSR	SR
NrdnComs1	57	78	**89**	57	33	82	**100**	**100**	79	47	96	**100**	**100**	95	73
NrdnComs2	56	54	**83**	55	38	83	66	**99**	82	52	94	89	**100**	94	74
NrdnComs3	50	70	**76**	52	34	77	85	**100**	80	50	99	97	**100**	99	77
NrdnComs4	68	62	**74**	67	28	92	**100**	**100**	91	50	98	**100**	**100**	97	79
NrdnComs5	59	64	**88**	59	32	79	**91**	**91**	79	50	89	**91**	**91**	90	74
PokeC age	20	12	**22**	19	7	39	33	**47**	39	21	62	60	**68**	62	46
PokeC height	**23**	12	21	**23**	12	46	29	**48**	46	28	69	62	**73**	69	54

(c) Lying Scenario 1, no homophily

	Low Budget					Medium Budget					High Budget				
Network/ Strategy	RS	RDLRN	MRN	MRSR	SR	RS	RDLRN	MRN	MRSR	SR	RS	RDLRN	MRN	MRSR	SR
NrdnComs1	16	**33**	12	14	22	38	**46**	20	27	36	**72**	64	40	48	58
NrdnComs2	30	**70**	21	26	35	50	**82**	26	41	55	86	**94**	52	63	84
NrdnComs3	21	**59**	12	14	22	57	**82**	16	28	44	76	**99**	41	57	76
NrdnComs4	12	**30**	6	8	12	31	**52**	11	15	28	53	**67**	28	38	47
NrdnComs5	13	**35**	10	11	16	30	**51**	12	18	26	52	**55**	28	34	40
PokeC age	5	**13**	5	6	7	14	**34**	16	18	20	43	**59**	39	41	44
PokeC height	13	**14**	5	7	11	**33**	**33**	15	19	27	**69**	59	37	44	52

(d) Lying Scenario 2, no homophily

	Low Budget					Medium Budget					High Budget				
Network/ Strategy	RS	RDLRN	MRN	MRSR	SR	RS	RDLRN	MRN	MRSR	SR	RS	RDLRN	MRN	MRSR	SR
NrdnComs1	14	**33**	12	14	22	21	**53**	19	24	41	53	**64**	40	48	63
NrdnComs2	26	**58**	22	25	37	40	**70**	26	35	54	68	83	54	70	**84**
NrdnComs3	15	**64**	12	16	23	26	**85**	17	23	38	54	**98**	41	57	70
NrdnComs4	8	**35**	7	8	15	15	**59**	10	15	27	38	**66**	26	35	50
NrdnComs5	9	**39**	9	11	18	16	**47**	13	17	27	33	**53**	27	35	42
PokeC age	6	**10**	5	6	7	16	**26**	14	16	18	42	**59**	39	41	44
PokeC height	6	**12**	5	7	10	19	**28**	15	19	26	43	**58**	37	43	52

Bibliography

[1] Lada A. Adamic, Rajan M. Lukose, Amit R. Puniyani, and Bernardo A. Huberman. Search in power-law networks. Physical review E, 64(4):046135, 2001.

[2] David Aldous and Jim Fill. Reversible markov chains and random walks on graphs, 2002.

[3] A. Asztalos and Z. Toroczkai. Network discovery by generalized random walks. EPL (Europhysics Letters), 92(5):50008, 2010.

[4] Wayne E. Baker and Robert R. Faulkner. The social organization of conspiracy: Illegal networks in the heavy electrical equipment industry. American sociological review, pages 837–860, 1993.

[5] Patrick Biernacki and Dan Waldorf. Snowball sampling: Problems and techniques of chain referral sampling. Soc. methods & research, 10(2):141–163, 1981.

[6] Catherine A. Bliss, Christopher M. Danforth, and Peter Sheridan Dodds. Estimation of global network statistics from incomplete data. PloS one, 9(10):e108471, 2014.

[7] Benjamin Davis, Ralucca Gera, Gary Lazzaro, Bing Yong Lim, and Erik C. Rye. The marginal benefit of monitor placement on networks. In Complex Networks VII, pages 93–104. Springer, 2016.

[8] Nir Friedman and Daphne Koller. Being bayesian about network structure. a bayesian approach to structure discovery in bayesian networks. Machine learning, 50(1-2):95–125, 2003.

[9] Agata Fronczak and Piotr Fronczak. Biased random walks in complex networks: The role of local navigation rules. Physical Review E, 80(1):016107, 2009.

[10] Ralucca Gera, Ryan Miller, Miguel MirandaLopez, and Scott Warnke. Developing multilayered dark networks to enhance community identification. Submitted for publication (2016).

[11] Barry D. Hughes. Random walks and random environments. 1996.

[12] Jure Leskovec and Christos Faloutsos. Sampling from large graphs. In SIGKDD, pages 631–636. ACM, 2006.

[13] Jae Dong Noh and Heiko Rieger. Random walks on complex networks. Physical review letters, 92(11):118701, 2004.

[14] Jörg Raab and H. Brinton Milward. Dark networks as problems. Journal of public administration research and theory, 13(4):413–439, 2003.