MODERN IMMUNIZATION APPROACHES ON PROPAGATION STYLE PROCESSES

[1]Megha Datta, [1] Harshita Negi, [1]Mansi Goel

[1]Department of Computer Applications
National Institute of Technology,Kurukshetra,India

[1]Mr.Vikas Goyal, [1]Dr. Kapil Gupta

[1]Department of Computer Applications
National Institute of Technology, Kurukshetra,India

ABSTRACT**: Given a graph, like a social/computer network in which an infection (or meme or virus) has been spreading for some time, how to select the best k nodes for immunization immediately? Similarly, given the follower followee network and some rumours spreading on it, which accounts should Twitter decide to suspend/delete/warn to stop the misinformation as fast as possible. For this, the Data-Aware Vaccination problem has been formulated which is NP-hard. Secondly, two effective polynomial-time heuristics DAVA and DAVA-fast are also presented. In this paper we present an algorithm to merge the infected nodes into one super infected node. Further we propose a Bayesian Probability analysis for a graph.**

Keywords: SIR, IC, DAVA, DAVA-Fast, DAVA Prune, Dominator tree, Minimum Spanning Tree, Benefit, Netshield, Shield Value.

I. INTRODUCTION

Propagation-style processes on graphs/networks are important tools to model situations of interest in real life like in epidemiology, in cyber-security and in social-systems. For instance, infectious diseases spreading over contact networks, malware propagating over computer networks, spam/ rumours spreading on Twitter, all are propagation-style processes.

Hence, controlling and stopping such malicious propagation is a natural and significant problem with a large number of applications. We use two well-known and popular virus propagation models to model the virus spreading on the network: the Susceptible-Infected-Recovered (SIR) model and the Independent Cascade (IC) model. The SIR model is an epidemiological model that computes the theoretical number of people infected with a contagious illness in a closed population overtime whereas the IC model is applicable on social media and is based on forming or breaking down of the information cascades. These baseline policies give us good guideline but they may not be ideal for making real-time decisions, given that an epidemic has already affected sets of people. Hence efficient algorithms for such a Data-Aware Vaccination problem can help in making a better choice.

II. LIST OF SYMBOLS

Table 1 lists the symbols with their description used in the paper.

TABLE 1:

TERMS AND SYMBOLS	DEFINITION AND DESCRIPTION
DAV	Data Aware Vaccination Problem.
SIR	Susceptible Infected Recovered Model
IC	Independent Cascade Model
G(V,E)	Graph G with node set V and edge set E.
P_{ij}	Propagation probability from node i to j.
Δ	Recovered probability for each node.
P_A	Probability that node A is infected by virus 1.
P_X	Probability that node B is infected by virus 1.
P_Y	Probability that node B is infected by virus 2.
P_C	Probability that node C is infected by virus 2.
P_B	Probability that node B is infected by super infected virus(merged virus)
K	The budget(i.e., nodes to give vaccine to)
S	Set of nodes selected for vaccination.
$\sigma'_{G,I}(S)$	The expected number of saved nodes after removing set S from graph G.
$\Upsilon S_{(J)}$	Benefit of adding additional node j into S
J	Additional node

III. LITERARTURE SURVEY

In [1], C. Chen, H. Tong, B. A. Prakash, C.Tsourakakis, T. Eliassi-Rad, C. Faloutsos and D. H.Chau, proposed a novel definition of ‘Shieldvalue’ score and a near-optimal and scalable algorithm (NetShield) to find a set of nodes with the highest ‘Shield value’ score.They further proposed its variant (NetShield+) to balance the optimization quality and speed.

In [3], R.Cohen, S.Havlin, and D. ben-Avraham, proposed novel efficient strategy for immunization, requiring no knowledge of the nodes, degrees or any other global information. This approach may be relevant to other networks, such as ecological networks of Predator prey, metabolic networks, networks of cellular proteins, and terrorist networks.

In [4], Y. Zhang, B. A. Prakash addresses the problem of immunizing healthy nodes in the presence of already infected nodes.The potential applications are broad: from distributing vaccines to control the epidemic, to stopping already present rumours in social media. The paper addresses the problem of immunizing healthy nodes in presence of already infected nodes, given a graph like a social/computer network. It also presents three polynomial-time heuristics DAVA, DAVA-prune and DAVA-fast for general graphs of varying degrees of performance and also the scalability of DAVA-prune and DAVA-fast on large-scale networks.

In [5], Y.Zhang, B. A. Prakash formulated the problem called Data Aware Vaccination Problem and presented two polynomial-time heuristics DAVA and DAVA-Fast for general graphs. The paper proves the effectiveness and efficiency of the algorithms through experiments on multiple datasets.

In [6], T. Lengauer and R. E. Tarjan presented fast algorithm for finding dominators in a flow graph. They presented an algorithm that runs in O (m log n) time, where m is the number of edges and n is the number of vertices in the problem graph.

IV. RESEARCH GAP

In the previously presented algorithm DAVA and DAVA-Fast the probability of a node infecting others was taken to be 1.

$$\boxed{P_{i,j} = 1}$$

Taking $P_{i,j} = 1$, means that the probability of node i infecting node j is 1 i.e., under all the circumstances the probability remains unchanged. However a more realistic approach would consider other factors like Time, the transmission capability of a virus, how worse the virus gets and other factors. For example a node A is in direct contact with node B and node According to DAVA algorithm the probability of node A getting infected by the two is same.However,if node B spends more time(or other factors)with node A then the probability that it might get the virus from it is more rather than getting it from node C.

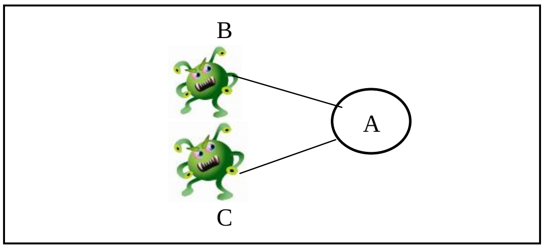

Figure 1: Example illustrating the research gap.

V.THEORETICAL CONCEPT, MODEL AND ALGORITHM

DATA-AWARE VACCINATION PROBLEM:

***Problem**: Given a set of infected nodes and a contact graph, how to distribute k vaccines (node removal) to minimize theexpected number of infected nodes at the end of the epidemic?*

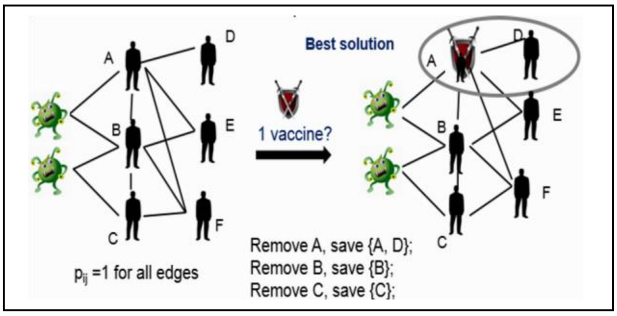

Figure 2: DAV Problem

DAVA algorithm (FOR TREES)

STEPS:

1: Simplify - Merging infected nodes

Merge all the infected nodes into a single ‘**super infected**’ node

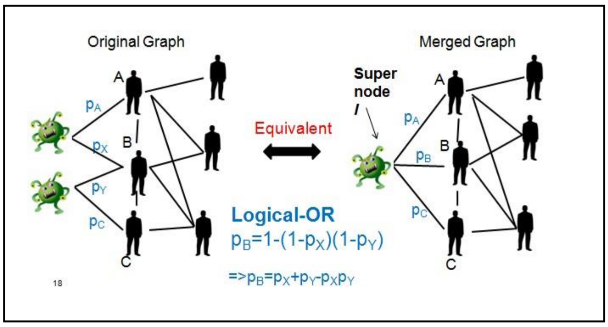

Figure 3: Merging the infected nodes.

2: RUN DAVA-Tree Algorithm

Select nodes with the largest “benefit”

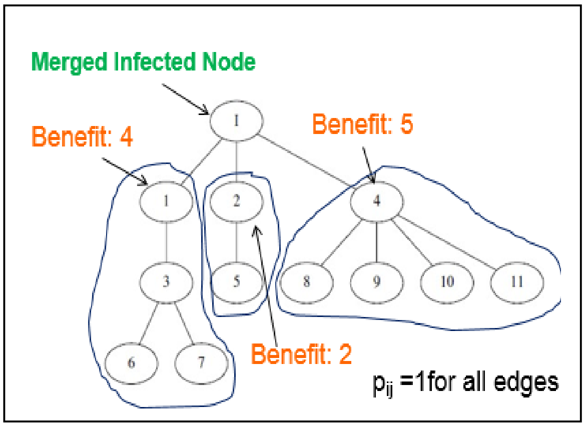

Figure 4: Calculating Maximum benefit

- σ’G,I(S): the expected number of saved nodes after removing set S on graph G
- Benefit of adding additional node j into S:

$$\gamma_S(j) = \underbrace{\underbrace{\sigma'_{G,I}(S\cup\{j\})}_{\text{\# of saved nodes after adding j to S.}} - \sigma'_{G,I}(S)}_{\text{Additional number of saved nodes after Adding j to S}}$$

FOR GRAPHS

STEPS:

1: Simplify - Merging infected nodes
2: Build a dominator tree.
3: Select the node (v) with the largest benefit with budget=1.
4: Remove v from G.
5: Go to Step 1 until |S|=k

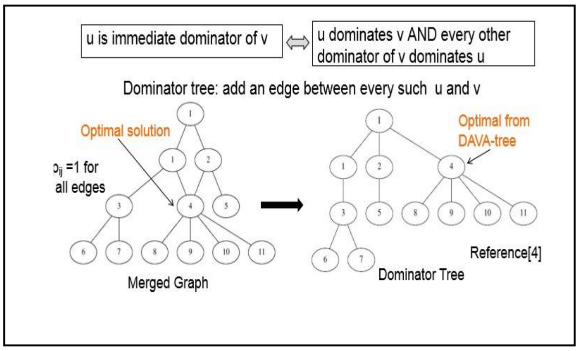
u is immediate dominator of v
⟺
u dominates v AND every other dominator of v dominates u
Dominator tree: add an edge between every such u and v
Optimal solution
p_{ij} =1 for all edges
Optimal from DAVA-tree
Merged Graph
Dominator Tree
Reference[4]

Figure 5: Building up a dominator tree.

After constructing the dominator tree the node with the maximum ‘benefit’ is chosen and is removed from the graph. Fig.5 illustrates the first pass(iteration) of the aforementioned algorithm.

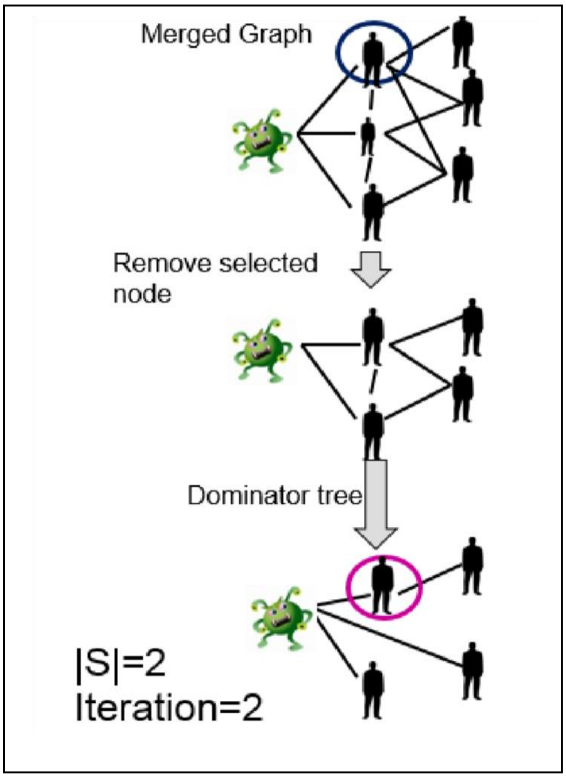

Figure 6: Running DAVA Tree algorithm

DAVA fast algorithm:

STEPS:

1: T = Build a dominator tree
2: S = Run DAVA-tree on T with budget=k

Time Complexity:

DAVA: The time complexity to run the DAVA algorithm is **O *((|E|+ |V|log|V|))* .**However, we build the dominator tree K times. Therefore, the time it takes is ***O (k (|E|+ |V|log|V|)).*** This algorithm works fine on small graphs, but can be slow on large graphs as it rebuilds the dominator tree k times.

DAVA fast: Since the budget for DAVA-Fast is taken to be as ‘k’ for a single pass therefore it reduces the time complexity by a factor of ‘k’. Thus the time complexity becomes ***O ((|E|+ |V|log|V|)).*** In practice, the performance of DAVA-fast is very close to DAVA.

VI. ALGORITHM TO MERGE THE INFECTED NODE:

```
merge_infected_node(G , Iset )
Step 1: Add node I in graph G.
Step 2: for node n in Iset
{
    for nbr in n.neighbours
    {
        w=weight(n,nbr)
        if(G.edge(I,nbr)==False
        {
            add G.edge(I,nbr)
            weight(I,nbr)=w
        }
        else
        {
            t=weight(I,nbr)
            weight(I,nbr)=t+(1-t)*w
        }
    }
}
Step 3: for n in Iset
{
    if n in G
    remove n from G
}
```

VII. PROPOSAL

We will be implementing the DAVA algorithm and will do the Bayesian probability analysis of the graph and will propose a suitable running algorithm with Time complexity analysis. We will then show a comparison between the given DAVA algorithm and our proposed algorithm.

VIII. BAYESIAN PROBABILITY ANALYSIS:

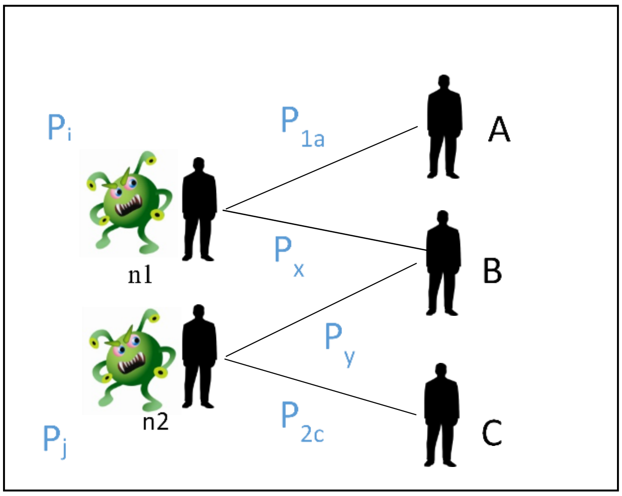

Figure 7:An example illustrating bayesian Analysis

Terminology:
Vertex set: {A,B,C,n1,n2}
Edge set: {(n1-A),(n1-B),(n2-B),(n2-C)}
$P_{i:}$ Probability that the node n1 is infected.
$P_{j:}$ Probability that the node n2 is infected.
$P_{1a:}$ Probability that node A is infected by n1.
$P_{x:}$ Probability that node B is infected by n1.
$P_{y:}$ Probability that node B is infected by n2.
$P_{2c:}$ Probability that node C is infected by n2.
P (n1|B): Probability of n1 infecting B given that B is infected.
P (B|n1): Probability that B has been infected by n1 given that n1 is infected.
P (B|n2): Probability that B has been infected by n2 given that n2 is infected.

TRUTH TABLE:

n1	n2	Probability of B
F	F	0
F	T	P_y
T	F	P_x
T	T	P_xP_y

Generalized Formula:

$$P(n1|B)=\frac{P(n_1)P(B|n_1)}{\sum_{i=1}^{n} P(ni)P(B|n_i)}$$

Deduced Formula:

$$\mathbf{P(n1|B)=}\frac{P_iP_x}{P_iP_x + PjPy - PiPjP_xP_y}$$

IX. CONCLUSION AND FUTURE SCOPE

This paper addresses the problem of immunizing healthy nodes in presence of already infected nodes given a graph like a social/computer network. Firstly, we identify the research gap and then present the problem definition and steps of DAVA algorithm. Further we propose the algorithm to merge the infected nodes into one and present the Bayesian probability analysis of a small graph.

X. REFERENCES

[1]. C. Chen, H. Tong, B. A. Prakash, C. Tsourakakis ,T.Eliassi-Rad, C.Faloutsos and D. H. Chau. ”Node Immunization on Large Graphs: Theory and Algorithms”**.**IEEE Transactions on knowledge and data engineering, VOL. 26, NO. 11, NOVEMBER 2014.

[2]. Y. Zhang, B. A. Prakash**.“**Scalable Vaccine Distribution in Large Graphs given Uncertain Data”. Proceedings of 23rd ACM International Conference on Information and Knowledge Management, CIKM ‘14, November 37, Shanghai, China.

[3]. R.Cohen, S. Havlin, and D. ben-Avraham.”Efficient Immunization Strategies for Computer Networks and Populations.” PHYSICAL REVIEW LETTERS , VOLUME 91, December 2003.

.

[4]. Y. Zhang, B. A. Prakash.”Data-Aware Vaccine Allocation over Large Networks”.ACM Transactions on Knowledge Discovery and Data Engineering, Series A, VOL 8, January 2015, Pages 1-32.

[5]. Y. Zhang, B. A. Prakash. “DAVA: Distributing Vaccines over Networks under Prior Information“.2014 SIAM International Conference on Data Mining, Philadelphia. April 24-26, 2014.

[6]. T. Lengauer and R. E. Tarjan.”A fast algorithm for finding dominators in a flow graph”. ACM Transactions on Programming Languages and Systems, VOL. 1,NO. 1, July 1979, pages 121-141.