Online Social Networks and Media

Diffusion:
Epidemic Spread
Influence Maximization
Introduction

**Diffusion:** process by which a *piece of information* is spread and reaches *individuals* through *interactions*
Why do we care?
Why do we care?

Modeling epidemics
Why do we care?

Viral marketing
Why do we care?

Viral video marketing network effect
Why do we care?

Spread of innovation
Outline

- Epidemic models
- Influence maximization
EPIDEMIC SPREAD
Epidemics

Understanding the spread of viruses and epidemics is of great interest to

- Health officials
- Sociologists
- Mathematicians
- Hollywood

The underlying contact network clearly affects the spread of an epidemic
Epidemics

• Model epidemic spread as a random process on the graph and study its properties

• Questions that we can answer:
  – What is the projected growth of the infected population?
  – Will the epidemic take over most of the network?
  – How can we contain the epidemic spread?

Diffusion of ideas and the spread of influence can also be modeled as epidemics
Basic Reproductive Number $R_0$

- Basic Reproductive Number ($R_0$): the expected number of new cases of the disease caused by a single individual.
- This is a dimensionless number (it does not have units) and it characterizes the spread of the virus.
- General computation:

\[ R_0 \propto \left( \frac{\text{infection}}{\text{contact}} \right) \left( \frac{\text{contact}}{\text{time}} \right) \left( \frac{\text{time}}{\text{infection}} \right) \]

\[ R_0 = \tau \bar{c} d \]

- In general, we want $R_0 < 1$ since this usually (but not always) implies that the infection will die out.
$R_0$ and $R_t$

- The computation of $R_0$ assumes that everyone is susceptible to infection.
- For monitoring the real-time development of an infection the real-time or effective $R_t$ is used.
- It takes into account the current state of the disease, who is sick, and who is immune.
- We definitely want $R_t < 1$.
- It is very hard to compute and depends on multiple factors.
A simple model

- **Branching process**: A person transmits the disease to each person she meets independently with a probability $p$.
- An infected person meets $k$ (new) people while she is contagious.
- Infection proceeds in waves.

Contact network is a tree with branching factor $k$. 
Infection Spread

• We are interested in the number of people infected (spread) and the duration of the infection
• This depends on the infection probability $p$ and the branching factor $k$

An aggressive epidemic with high infection probability

The epidemic survives after three steps
Infection Spread

• We are interested in the number of people infected (spread) and the duration of the infection

• This depends on the infection probability $p$ and the branching factor $k$

A mild epidemic with low infection probability

The epidemic dies out after two steps
Basic Reproductive Number

• Basic Reproductive Number \((R_0)\): the expected number of new cases of the disease caused by a single individual

\[ R_0 = kp \]

• Claim:
  a) If \(R_0 < 1\), then with probability 1, the disease dies out after a finite number of waves. In this case each person infects less than one person in expectation. The infection eventually \textit{dies out}.

  b) If \(R_0 > 1\), then with probability greater than 0 the disease persists by infecting at least one person in each wave. In this case each person infects more than one person in expectation. The infection \textit{persists}.

Application: Reduce \(k\), or \(p\) to combat an epidemic
Analysis

• $X_n$: random variable indicating the number of infected nodes at level $n$ (after $n$ steps)
• $q_n = \Pr[X_n \geq 1]$: probability that there exists at least 1 infected node after $n$ steps
• $q^* = \lim q_n$: the probability of having infected nodes as $n \to \infty$

We want to show that

(a) $R_0 < 1 \Rightarrow q^* = 0$
(b) $R_0 > 1 \Rightarrow q^* > 0$. 
Proof

- At level \( n \), \( k^n \) nodes
- \( Y_{nj} \): 1 if node \( j \) at level \( n \) is infected, 0 otherwise
  \[ E[Y_{nj}] = p^n \]
- \( E[X_n] = R_0^n \)
- \( E[X_n] \geq \Pr[X_n \geq 1] \Rightarrow q_n \leq R_0^n \)

This proves (a) but not (b)
Proof

Each child of the root starts a branching process of length $n-1$

$$q_n = 1 - (1 - pq_{n-1})^k$$

if

$$f(x) = 1 - (1 - px)^k$$

then

$$q_n = f(q_{n-1})$$

We also have: $q_0 = 1$.

So we obtain a series of values: $1, f(1), f(f(1)), ...$

We want to find where this series converges
Proof

• Properties of the function $f(x)$:

1. $f(0) = 0$ and $f(1) = 1 - (1 - p)^k < 1$.
   
   passes through $(0, 0)$; below $y = x$, once $x = 1$

2. $f'(x) = pk(1 - px)^{k-1} > 0$, in the interval $[0,1]$ but decreasing. Our function is increasing and concave.

3. $f'(0) = pk = R_0$

   Slope at $x = 0$
Proof

• Case 1: $R_0 = pk > 1$. The function starts above the line $y = x$ but then drops below the line.

$f(x)$ crosses the line $y = x$ at some point
Proof

• Starting from the value 1, repeated applications of the function $f(x)$ will converge to the value $q^* = q_n = f(q_n)$
Proof

• Case 2: $R_0 = pk < 1$. The function starts with below the line $y = x$. Repeated applications of $f(x)$ converge to zero.
Branching process

- Assumes no network structure, no triangles or shared neighbors
The SIR model

• Each node may be in the following states
  – **Susceptible**: healthy but not immune
  – **Infected**: has the virus and can actively propagate it
  – **Removed**: (Immune or Dead) had the virus but it is no longer active

• Parameter $p$: the probability of an Infected node to infect a Susceptible neighbor
The SIR process

• Initially all nodes are in state $S$ (susceptible), except for a few nodes in state $I$ (infected).

• An infected node stays infected for $t_I$ steps.
  – Simplest case: $t_I = 1$

• At each of the $t_I$ steps the infected node has probability $p$ of infecting any of its susceptible neighbors
  – $p$: Infection probability

• After $t_I$ steps the node is Removed
Example
Example
Example
Example
Figure 21.2: The course of an SIR epidemic in which each node remains infectious for a number of steps equal to $t_I = 1$. Starting with nodes $y$ and $z$ initially infected, the epidemic spreads to some but not all of the remaining nodes. In each step, shaded nodes with dark borders are in the Infectious ($I$) state and shaded nodes with thin borders are in the Removed ($R$) state.
Extensions

- Probability per pair of nodes
- Sequence of several states (e.g. early, middle, and late periods of the infection), and allowing the contagion probabilities to vary across these states
- Mutating, change the characteristics
Continuous case

• We can analyze the SIR model assuming a continuous change in the number of Susceptible (S), Infected (I), and Removed (R) nodes.
• In the continuous model the infection probability is replaced by the infection rate $\beta$
• We also have the recovery (or removal) rate $\gamma = 1/t_I$ which is the rate by which nodes recover (or die)
• Let $s = \frac{S}{N}, i = \frac{I}{N}, r = \frac{R}{N}$, the fraction of S, I, R nodes, where $N$ the size of the population
• We assumed that initially $s \approx 1$
• We assume that we have $si$ contacts (random contacts)
Continuous case

- We can describe SIR with the following system of differential equations:

\[
\frac{ds}{dt} = -\beta si \\
\frac{di}{dt} = \beta si - \gamma i \\
\frac{dr}{dt} = \gamma i
\]

- The epidemic persists if

\[
\frac{di}{dt} > 0 \Rightarrow \frac{\beta}{\gamma} > 1
\]

\[
R_0 = \frac{\beta}{\gamma}
\]
SIR and the Branching process

• The branching process is a special case where the graph is a tree (and the infected node is the root)
  – The existence of triangles shared neighbors makes a big difference

• The basic reproductive number is not necessarily informative in the general case
SIR and the Branching process

Example

$R_0$ the expected number of new cases caused by a single node

Assume

$p = 2/3,$

$R_0 = 4/3 > 1$

Probability to fail at each level and stop $(1/3)^4 = 1/81$

Figure 21.3: In this network, the epidemic is forced to pass through a narrow “channel” of nodes. In such a structure, even a highly contagious disease will tend to die out relatively quickly.
Percolation

• **Percolation**: we have a network of “pipes” which can carry liquids, and they can be either **open**, or **closed** with some probability
  – The pipes can be pathways within a material

• If liquid enters the network from some nodes, does it reach most of the network?
  – The network **percolates**
SIR and Percolation

• There is a connection between SIR model and percolation
• When a virus is transmitted from $u$ to $v$, the edge $(u, v)$ is activated with probability $p$
• We can assume that all edge activations have happened in advance, and the input graph has only the active edges.
• Which nodes will be infected?
  – The nodes reachable from the initial infected nodes
• In this way we transformed the dynamic SIR process into a static one.
  – This is essentially percolation in the graph.
Figure 21.4: An equivalent way to view an SIR epidemic is in terms of *percolation*, where we decide in advance which edges will transmit infection (should the opportunity arise) and which will not.
The SIS model

- **Susceptible-Infected-Susceptible**
  - Susceptible: healthy but not immune
  - Infected: has the virus and can actively propagate it
- An **Infected** node infects a **Susceptible** neighbor with probability $p$
- An **Infected** node becomes **Susceptible** again with probability $q$ (or after $t_I$ steps)
  - In a **simplified** version of the model $q = 1$
- Nodes **alternate** between **Susceptible** and **Infected** status
Example

• When no **Infected** nodes, virus dies out
• Question: will the virus die out?

Figure 21.5: In an SIS epidemic, nodes can be infected, recover, and then be infected again. In each step, the nodes in the Infectious state are shaded.
An eigenvalue point of view

- If $A$ is the adjacency matrix of the network, then the virus dies out if
  \[ \lambda_1(A) \leq \frac{q}{p} \]
- Where $\lambda_1(A)$ is the first eigenvalue of $A$

SIS and SIR

Time expanded network
Including time

- Infection can only happen within the **active** window

---

(a) In a contact network, we can annotate the edges with time windows during which they existed.

(b) The same network as in (a), except that the timing of the $w-v$ and $w-y$ partnerships have been reversed.

Figure 21.8: Different timings for the edges in a contact network can affect the potential for a disease to spread among individuals. For example, in (a) the disease can potentially pass all the way from $u$ to $y$, while in (b) it cannot.
Concurrenciy

• Importance of concurrency – enables branching

(a) No node is involved in any concurrent partnerships

(b) All partnerships overlap in time

Figure 21.10: In larger networks, the effects of concurrency on disease spreading can become particularly pronounced.
SIRS

• Initially, some nodes are in the \( I \) state and all others in the \( S \) state.

• Each node \( u \) that enters the \( I \) state remains infectious for a fixed number of steps \( t_I \). During each of these \( t_I \) steps, \( u \) has a probability \( p \) of infected each of its susceptible neighbors.

• After \( t_I \) steps, \( u \) is no longer infectious. Enters the \( R \) state for a fixed number of steps \( t_R \). During each of these \( t_R \) steps, \( u \) cannot be infected nor transmit the disease.

• After \( t_R \) steps in the \( R \) state, node \( u \) returns to the \( S \) state.
References

• D. Easley, J. Kleinberg. *Networks, Crowds and Markets: Reasoning about a highly connected world*. Cambridge University Press, 2010 – Chapter 21

• James Holland Jones, Notes on $R_0$

INFLUENCE MAXIMIZATION
Maximizing spread

• Suppose that instead of a virus we have an item (product, idea, video) that propagates through contact
  – Word of mouth propagation.

• An advertiser is interested in maximizing the spread of the item in the network
  – The holy grail of “viral marketing”

• Question: which nodes should we “infect” so that we maximize the spread? [KKT2003]
Independent cascade model

• Each node may be **active** (has the item) or **inactive** (does not have the item)

• Time proceeds at discrete time-steps.

• At time $t$, every node $v$ that became active in time $t-1$ activates a non-active neighbor $w$ with probability $p_{uw}$. If it fails, it does not try again.

• The same as the simple **SIR model**
Independent cascade

Step 0

Step 1

Step 2

Step 3

Step 4

Final Stage
Influence maximization

• **Influence function**: for a set of nodes $A$ (target set) the influence $s(A)$ (spread) is the expected number of active nodes at the end of the diffusion process if the item is originally placed in the nodes in $A$.

• **Influence maximization problem** [KKT03]: Given a network, a diffusion model, and a value $k$, identify a set $A$ of $k$ nodes in the network that maximizes $s(A)$.

• The problem is NP-hard
A Greedy algorithm

• What is a simple algorithm for selecting the set $A$?

**Greedy algorithm**
- Start with an empty set $A$
- Proceed in $k$ steps
  - At each step add the node $u$ to the set $A$ the maximizes the increase in function $s(A)$
    - The node that activates the most additional nodes

• Computing $s(A)$: perform multiple Monte-Carlo simulations of the process and take the average.
• How good is the solution of this algorithm compared to the optimal solution?
Approximation Algorithms

• Suppose we have a (combinatorial) optimization problem, and $X$ is an instance of the problem, $\text{OPT}(X)$ is the value of the optimal solution for $X$, and $\text{ALG}(X)$ is the value of the solution of an algorithm $\text{ALG}$ for $X$
  
  — In our case: $X = (G, k)$ is the input instance, $\text{OPT}(X)$ is the spread $s(A^*)$ of the optimal solution, $\text{GREEDY}(X)$ is the spread $s(A)$ of the solution of the Greedy algorithm

• $\text{ALG}$ is a good approximation algorithm if the ratio of $\text{OPT}$ and $\text{ALG}$ is bounded.
Approximation Ratio

- For a maximization problem, the algorithm ALG is an $\alpha$-approximation algorithm, for $\alpha < 1$, if for all input instances $X$,
  $$\text{ALG}(X) \geq \alpha \text{OPT}(X)$$

- The solution of ALG($X$) has value at least $\alpha\%$ that of the optimal

- $\alpha$ is the approximation ratio of the algorithm
  - Ideally, we would like $\alpha$ to be a constant close to 1
Approximation Ratio for Influence Maximization

- The **GREEDY** algorithm has approximation ratio $\alpha = 1 - \frac{1}{e}$

$$\text{GREEDY}(X) \geq \left(1 - \frac{1}{e}\right) \text{OPT}(X), \text{ for all } X$$
Proof of approximation ratio

• The spread function $s$ has two properties:

  • $s$ is monotone:
    $$s(A) \leq s(B) \text{ if } A \subseteq B$$

  • $s$ is submodular:
    $$s(A \cup \{x\}) - s(A) \geq s(B \cup \{x\}) - s(B) \text{ if } A \subseteq B$$

• The addition of node $x$ to a set of nodes has greater effect (more activations) for a smaller set.
  – The diminishing returns property
Optimizing submodular functions

• **Theorem**: A greedy algorithm that optimizes a monotone and submodular function $s$, each time adding to the solution $A$, the node $x$ that maximizes the gain $s(A \cup \{x\}) - s(A)$ has approximation ratio $\alpha = \left(1 - \frac{1}{e}\right)$

• The spread of the Greedy solution is at least 63% that of the optimal
Submodularity of influence

• Why is \( s(A) \) submodular?
  – How do we deal with the fact that influence is defined as an expectation?

• We will use the fact that probabilistic propagation on a fixed graph can be viewed as deterministic propagation over a randomized graph
  – Express \( s(A) \) as an expectation over the input graph rather than the choices of the algorithm
Independent cascade model

• Each edge \((u, v)\) is considered only once, and it is “activated” with probability \(p_{uv}\).

• We can assume that all random choices have been made in advance
  – generate a sample subgraph of the input graph where edge \((u, v)\) is included with probability \(p_{uv}\)
  – propagate the item deterministically on the input graph
  – the active nodes at the end of the process are the nodes reachable from the target set \(A\)

• The influence function is obviously (?) submodular when propagation is deterministic

• The linear combination of submodular functions is also a submodular function
Computation of Expected Spread

Computing $s(A)$: perform multiple Monte-Carlo simulations of the process and take the average.

To estimate the influence spread of $S \cup \{u\}$, $R$ repeated simulations of $RanCas(S \cup \{u\})$ are used.

Each run takes $O(m)$.

Complexity for computing the marginal gain of adding $u$: $O(Rm)$

For each $k$, all $n$ nodes are tested, thus $O(knRm)$
Computation of Expected Spread

• Performing simulations for estimating the spread on multiple instances is very slow. Several techniques have been developed for speeding up the process.
  – **CELF**: exploiting the submodularity property:
    • the marginal gain of a node in the current iteration cannot be better than its marginal gain in the previous iteration
  – **Maximum Influence Paths**: store paths for computation
  – **Sketches**: compute sketches for each node for approximate estimation of spread
Degree discount

General idea

- Select seed nodes based on their degree
- If node $v$ is selected, decrease the degree of all its neighbors

Maximum influence path

General idea

- For each node, use the maximum influence paths (paths with the largest probability) to all other nodes
  - Shortest weighted path
- Assumption: influence propagates through these paths
- Given this assumption, estimate the probability that a node is activated

Wei Chen, Chi Wang, Yajun Wang: Scalable influence maximization for prevalent viral marketing in large-scale social networks. KDD 2010: 1029-1038
Reverse Reachable Sets

Construct graph $X$ from G by activating edges with probability $p(e)$.

Let $v$ be a node in G, the reverse reachable (RR) set for $v$ in $X$ is the set of nodes in $X$ that can reach $v$.

That is, for each node $u$ in the RR set, there is a directed path from $u$ to $v$ in $X$.

Youze Tang, Xiaokui Xiao, Yanchen Shi: Influence maximization: near-optimal time complexity meets practical efficiency. SIGMOD Conference 2014: 75-86
Reverse Reachable Sets

Let $p$ be the probability for an RR set generated for $v$ to overlap with a node set $A$, then when we use $A$ as the seed set to run an influence propagation process on $G$, we have probability $p$ to activate $v$.

A random RR set is an RR set generated on an instance of $X$ randomly sampled from $G$, for a node selected uniformly at random from $X$. 
Reverse Reachable Sets

1. Generate a certain number of random RR sets from G.

2. Select k nodes to cover the maximum number of RR sets generated. (maximum coverage)

3. Return the k nodes as seed
Linear threshold model

- Again, each node may be active or inactive.
- Every directed edge \((v,u)\) in the graph has a weight \(b_{vu}\), such that
  \[
  \sum_{v \text{ is a neighbor of } u} b_{vu} \leq 1
  \]
- Each node \(u\) has a randomly generated threshold value \(T_u\).
- Time proceeds in discrete time-steps. At time \(t\) an inactive node \(u\) becomes active if
  \[
  \sum_{v \text{ is an active neighbor of } u} b_{vu} \geq T_u
  \]
- Related to the game-theoretic model of adoption.
Linear threshold model
Influence Maximization

- KKT03 showed that in this case the influence $s(A)$ is still a submodular function, using a similar technique
  - Assumes uniform random thresholds
- The Greedy algorithm achieves a $(1-1/e)$ approximation
Proof idea

• For each node $u$, pick one of the edges $(v, u)$ incoming to $u$ with probability $b_{vu}$ and make it live. With probability $1 - \sum b_{vu}$ it picks no edge to make live

• Claim: Given a set of seed nodes $A$, the following two distributions are the same:
  – The distribution over the set of activated nodes using the Linear Threshold model and seed set $A$
  – The distribution over the set of reachable nodes from $A$ using live edges.
Proof idea (submodularity LT model)

• Consider the special case of a DAG (Directed Acyclic Graph)
  – There is a topological ordering of the nodes \(v_0, v_1, ..., v_n\) such that edges go from left to right

• Consider node \(v_i\) in this ordering and assume that \(S_i\) is the set of neighbors of \(v_i\) that are active.

• What is the probability that node \(v_i\) becomes active in either of the two models?
  – In the Linear Threshold model the random threshold \(\theta_i\) must be \(\sum_{u \in S_i} b_{ui} \geq \theta_i\)
  – In the live-edge model we should pick one of the edges in \(S_i\)

• This proof idea generalizes to general graphs
  – Note: if we know the thresholds in advance submodularity does not hold!
Example

Assume that all edge weights incoming to any node sum to 1
The nodes select a single incoming edge with probability equal to the weight (uniformly at random in this case)
Example

Node $v_1$ is the seed
Node $v_3$ has a single incoming neighbor, therefore for any threshold it will be activated
The probability that node $v_4$ gets activated is $2/3$ since it has incoming edges from two active nodes.
The probability that node $v_4$ picks one of the two edges to these nodes is also $2/3$
Similarly the probability that node \( v_6 \) gets activated is 2/3 since it has incoming edges from two active nodes. The probability that node \( v_6 \) picks one of the two edges to these nodes is also 2/3.
The set of active nodes is the set of nodes reachable from \( v_1 \) with live edges (orange).
One-slide summary

- **Influence maximization:** Given a graph $G$ and a budget $k$, for some diffusion model, find a subset of $k$ nodes $A$, such that when activating these nodes, the spread of the diffusion $s(A)$ in the network is maximized.

- **Diffusion models:**
  - Independent Cascade model
  - Linear Threshold model

- **Algorithm:** *Greedy* algorithm that adds to the set each time the node with the maximum marginal gain, i.e., the node that causes the maximum increase in the diffusion spread.

- The Greedy algorithm gives a $\left(1 - \frac{1}{e}\right)$ approximation of the optimal solution
  - Follows from the fact that the spread function $s(A)$ is
    - Monotone: $s(A) \leq s(B)$, if $A \subseteq B$
    - Submodular: $s(A \cup \{x\}) - s(A) \geq s(B \cup \{x\}) - s(B)$, $\forall x$ if $A \subseteq B$
Evolving network

• Consider a network that changes over time
  – Edges and nodes can appear and disappear at discrete time steps

• Model:
  – The evolving network is a sequence of graphs \( \{G_1, G_2, \ldots, G_n\} \) defined over the same set of vertices \( V \), with different edge sets \( E_1, E_2, \ldots, E_n \)
  • Graph snapshot \( G_i \) is the graph at time-step \( i \).

ACM COSN 2015
Example

$G^1$

$G^2$

$G^3$
Time

• How does the evolution of the network relates to the evolution of the diffusion?
  – How much physical time does a diffusion step last?
• Assumption: The two processes are in sync. One diffusion step happens in on one graph snapshot
• Evolving IC model: at time-step $t$, the infectious nodes try to infect their neighbors in the graph $G_t$.
• Evolving LT model: at time-step $t$ if the weight of the active neighbors of node $v$ in graph $G_t$ is greater than the threshold the nodes gets activated.
Submodularity

• Will the spread function remain monotone and submodular?

• No!
Monotonicity for the EIC model
Monotonicity for the EIC model

The spread is not monotone in the case of the Evolving IC model.
Submodularity for the EIC model

\[ \begin{align*}
G^1 & : \quad v_1 \leftrightarrow v_2 \leftrightarrow v_3 \leftrightarrow v_4 \leftrightarrow u_1 \leftrightarrow u_2 \\
G^2 & : \quad v_1 \leftrightarrow v_2 \leftrightarrow v_3 \leftrightarrow v_4 \leftrightarrow u_1 \leftrightarrow u_2 \\
G^3 & : \quad v_1 \leftrightarrow v_2 \leftrightarrow v_3 \leftrightarrow v_4 \leftrightarrow v_5 \leftrightarrow v_6 \\
G^4 & : \quad v_1 \leftrightarrow v_2 \leftrightarrow v_3 \leftrightarrow v_4 \leftrightarrow v_5 \leftrightarrow v_6 
\end{align*} \]
Submodularity for the EIC model

Activating node $v_1$ at time $t = 0$ has spread 7
Submodularity for the EIC model

Activating node $v_1$ at time $t = 0$ has spread 7

Adding node $v_6$ at time $t = 3$ does not increase the spread
Submodularity for the EIC model

Activating nodes $v_1$ and $v_5$ at time $t = 0$ has spread 4
Submodularity for the EIC model

Activating nodes $v_1$ and $v_5$ at time $t = 0$ has spread 4.
Adding node $v_6$ at time $t = 3$ increases the spread to 9.
Evolving LT model

- The evolving LT model is monotone but it is not submodular

\[ G_U \quad G^1 \quad G^2 \]

- **Expected Spread**: the probability that \( u \) gets infected
  - Adding node \( v_3 \) has a larger effect if added to the set \( \{v_1, v_2\} \) than to set \( \{v_1\} \).
Extensions

• Other models for diffusion
  – **Deadline model**: There is a deadline by which a node can be infected
  – **Time-decay model**: The probability of an infected node to infect its neighbors decays over time
  – **Timed influence**: Each edge has a speed of infection, and you want to maximize the speed by which nodes are infected.

• Competing diffusions
  – Maximize the spread while competing with other products that are being diffused.
Extensions

• Reverse problems:
  – **Initiator discovery**: Given the state of the diffusion, find the nodes most likely to have initiated the diffusion
  

  – **Diffusion trees**: Identify the most likely tree of diffusion tree given the output


  – **Infection probabilities**: estimate the true infection probabilities

References

• N. Gayraud, E. Pitoura, P. Tsaparas. Maximizing Diffusion in Evolving Networks. ICCSS 2015
• H. Mannila, E. Terzi. Finding Links and Initiators: A Graph-Reconstruction Problem. SDM 2009
EXTRA SLIDES
Multiple copies model

• Each node may have **multiple copies** of the same virus
  
  – \( \mathbf{v} \): state vector : \( v_i \) : number of virus copies at node \( i \)

• At time \( t = 0 \), the state vector is initialized to \( \mathbf{v}^0 \)
• At time \( t \),
  
  For each node \( i \)
    
    For each of the \( v_i^t \) virus copies at node \( i \)
      
      the copy is copied to a neighbor \( j \) with prob \( p \)
    
    the copy dies with probability \( q \)

Analysis

• The expected state of the system at time $t$ is given by

$$\mathbf{v}^t = (pA + (1 - q)I)\mathbf{v}^{t-1} = M\mathbf{v}^{t-1}$$

$$M = \begin{bmatrix}
1 - q & p & p & 0 \\
0 & 1 - q & p & p \\
0 & 0 & 1 - q & p \\
p & 0 & 0 & 1 - q
\end{bmatrix}$$

Probability that the copy from node $v_4$ is copied to node $v_1$

Probability that the copy from node $v_4$ survives at $v_4$
Analysis

• As $t \to \infty$
  
  – if $\lambda_1(M) < 1 \iff \lambda_1(A) < q/p$ then $\sqrt{v^t} \to 0$
    • the probability that all copies die converges to 1
  
  – if $\lambda_1(M) = 1 \iff \lambda_1(A) = q/p$ then $\sqrt{v^t} \to c$
    • the probability that all copies die converges to 1
  
  – if $\lambda_1(M) > 1 \iff \lambda_1(A) > q/p$ then $\sqrt{v^t} \to \infty$
    • the probability that all copies die converges to a constant $< 1$
Another example

• What is the spread from the red node?

(a) In a contact network, we can annotate the edges with time windows during which they existed.

(b) The same network as in (a), except that the timing of the w-v and w-y partnerships have been reversed.

• Inclusion of time changes the problem of influence maximization
  – N. Gayraud, E. Pitoura, P. Tsaparas, Diffusion Maximization on Evolving networks