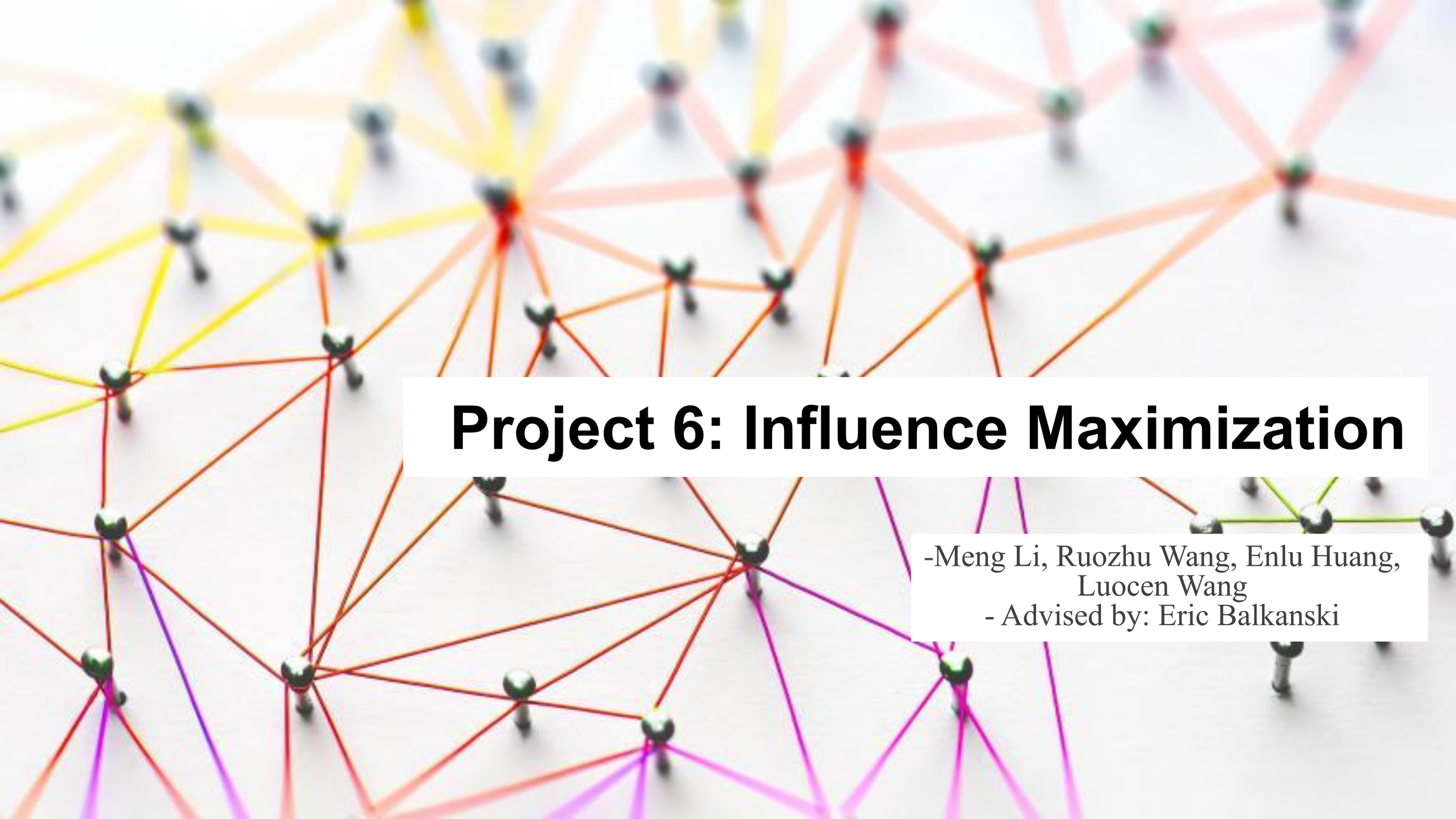
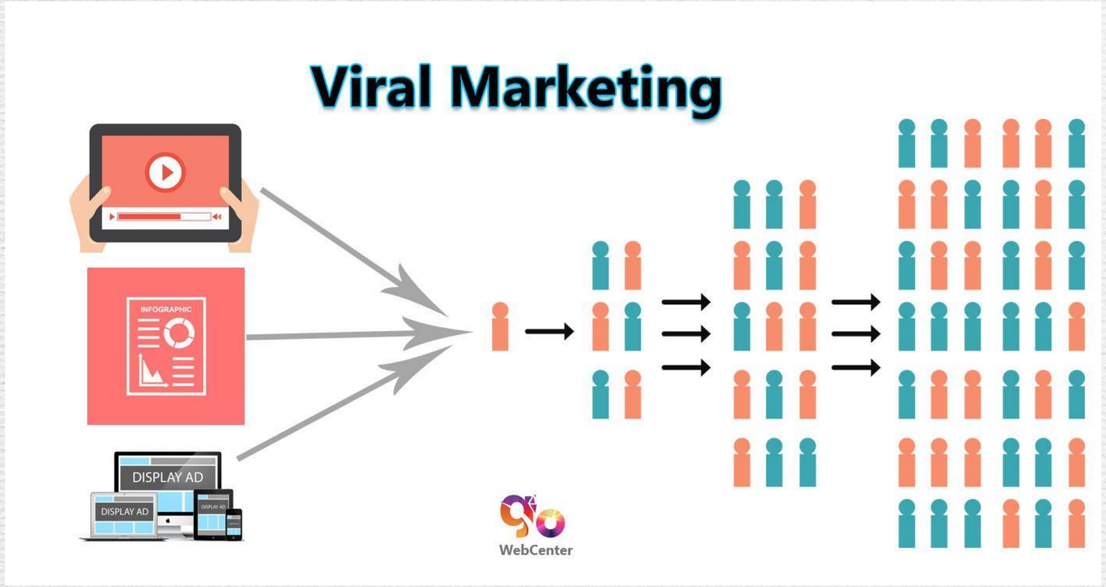
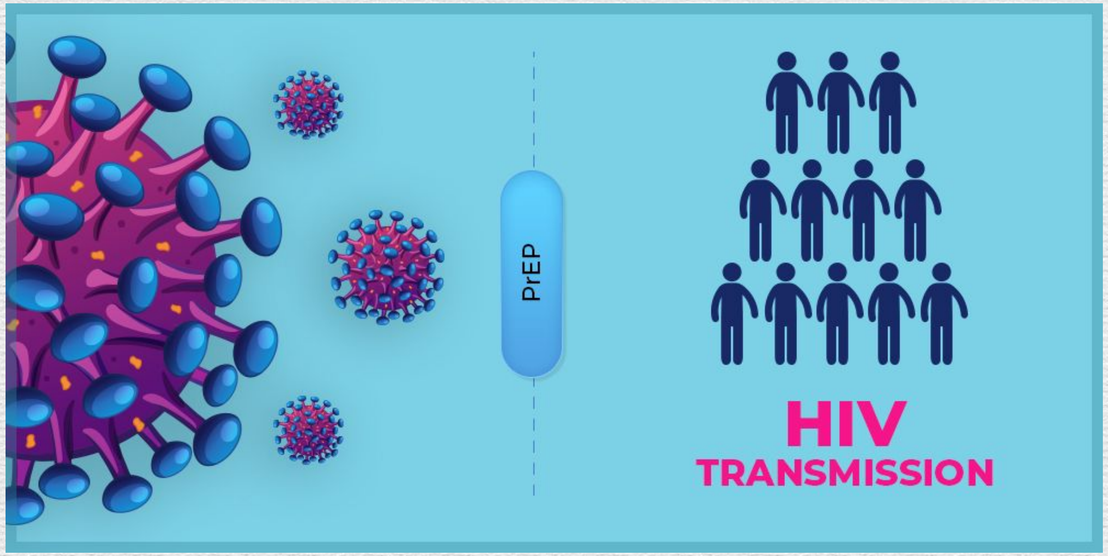
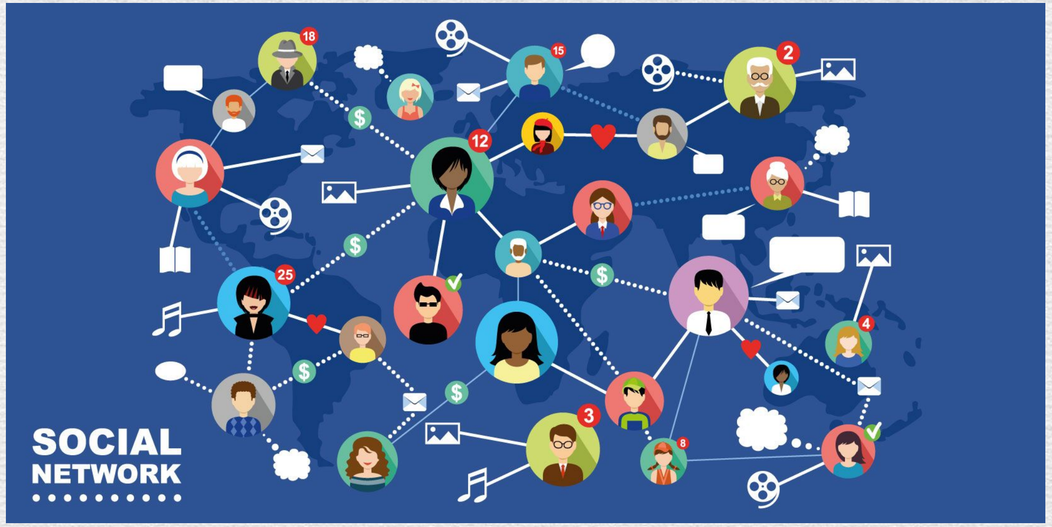


Project 6: Influence Maximization

-Meng Li, Ruozhu Wang, Enlu Huang, Luocen Wang
- Advised by: Eric Balkanski

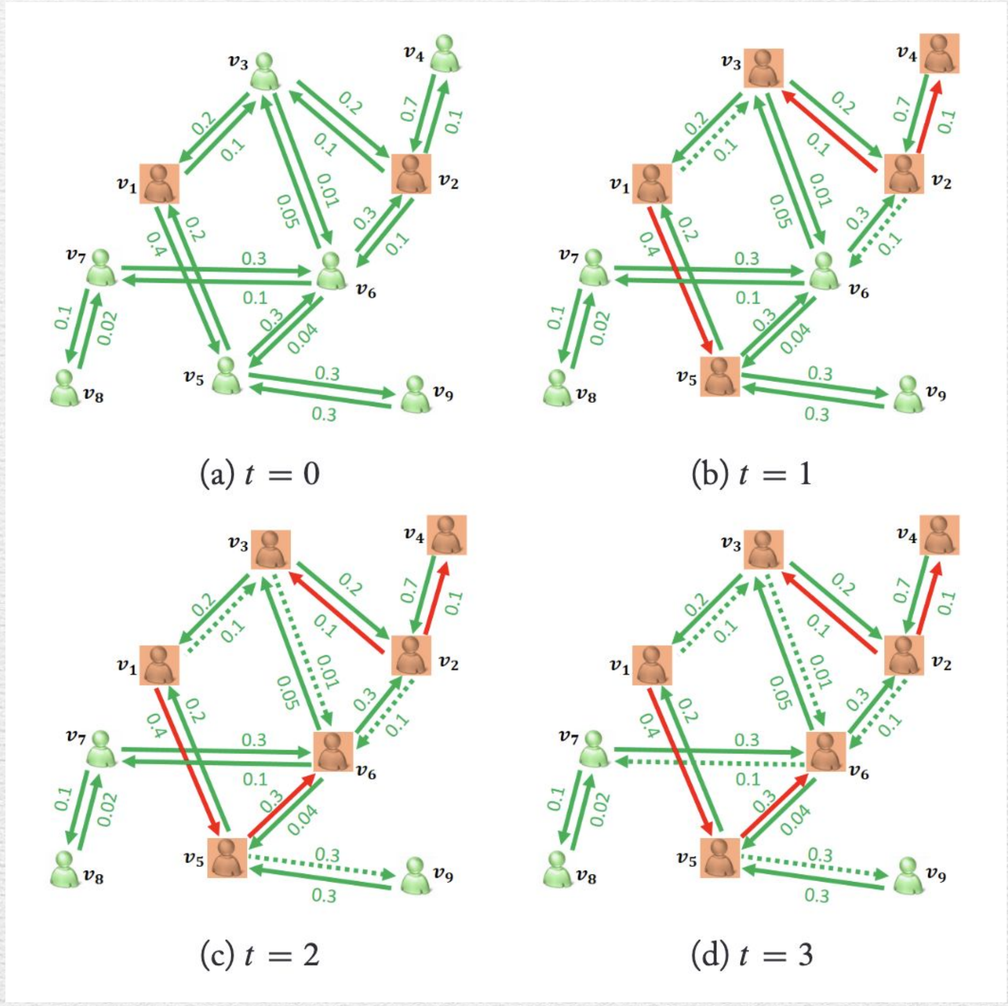

Viral Marketing
INFOGRAPHIC
DISPLAY AD
DISPLAY AD
WebCenter


PrEP
HIV
TRANSMISSION


SOCIAL
NETWORK

Independent Cascade Model

- The independent cascade model takes the social graph $G(V, E)$, the influence probability $p(\cdot)$ on all arcs, and the initial seed set S as the input, and generates the active sets S_t by the following randomized operation rule.(Goldenberg et al., 2001)
- Every arc $(u, v) \in E$ has an associated influence probability $p(u, v) \in [0, 1]$, corresponding to the probability that node u influences node v.
- Start with the initial active set S, every node has only one chance to be activated, once a node is activated, it stays active until no more influencing happens.



- **Influence Maximization Problem** (Kempe et al., 2003)
  - Given a graph $\boldsymbol{G = (V, E)}$, how to select $\boldsymbol{k}$ initial active nodes (seeds) that maximize the expected number of influenced nodes.

Research Question

Acquiring information about the network can be costly, or even impossible in many real cases, how should we select seeds with partial information?

Benchmark Algorithms

- **Random:** randomly choose k nodes from G as the seed set S. (**No info** required)
- **Greedy:** in each round, select the node that maximizes the expected gain of influence spread and add to the seed set S; repeat this until k nodes having been chosen. (**Full info** required)

*All expected influence spread are estimated by the independent cascade model.

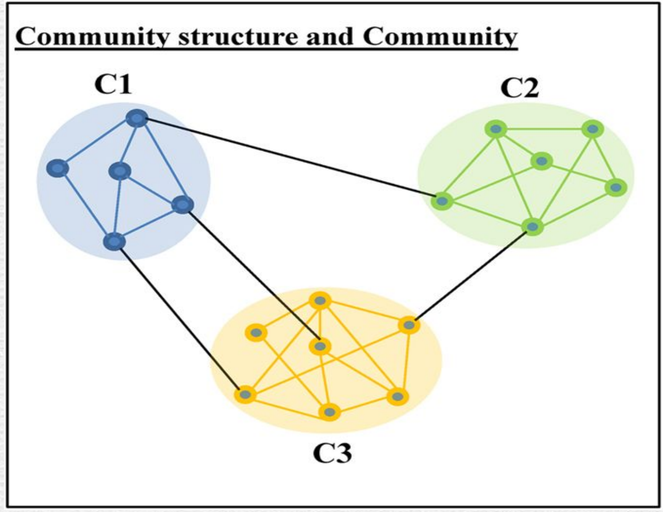
Algorithms with Only Partial Information Required

- **Edge Query (D.Eckles et al., 2019)**
- **ARISEN (Wilder et al., 2018)**
- **PV-IM (S.Eshghi et al., 2019)**
- **COPS (Balkanski et al., 2018)**

Edge Query

- **Required info**:
  - Estimated probability (higher than real probability)
  - Edge query: the i th neighbor of node u
- **Algorithm overview**:
  - Generating a set of samples by seeding a fixed set of nodes.
  - Greedily selecting the most connected nodes based on the generated samples.
- **Approximation guarantee:** $(1 - 1/e) * OPT - \epsilon n$

ARISEN



- **Required info:**
  - Total number of nodes in the network(n), influence probability within the community (Pw), influence probability between the community(Pb)
- **Algorithm idea:**
  - Using random walk to explore the network and carefully weight the candidate nodes by two weighting algorithms, then return the set of nodes according to the nodes' weights.
- **Algorithm Performance:**
  - After experiments, ARISEN works well on networks have strong community structure.

PV-IM Algorithm

- **Required Info:**
  - Visible part of graph, a list of subgraph (randomly selected nodes and their connections), a list of subgraph's weight, number of nodes(n), ε and δ
- **Algorithm Idea:**
  - Using the overlap fraction between set of nodes that can reach one selected node and the combined graph of visible graph with one subgraph to define the index 'realization probability', then utilizing greedy approximation algorithm to find the nodes with highest index.
  - This algorithm will give (1-1/e-ε) approximation performance with high probability.

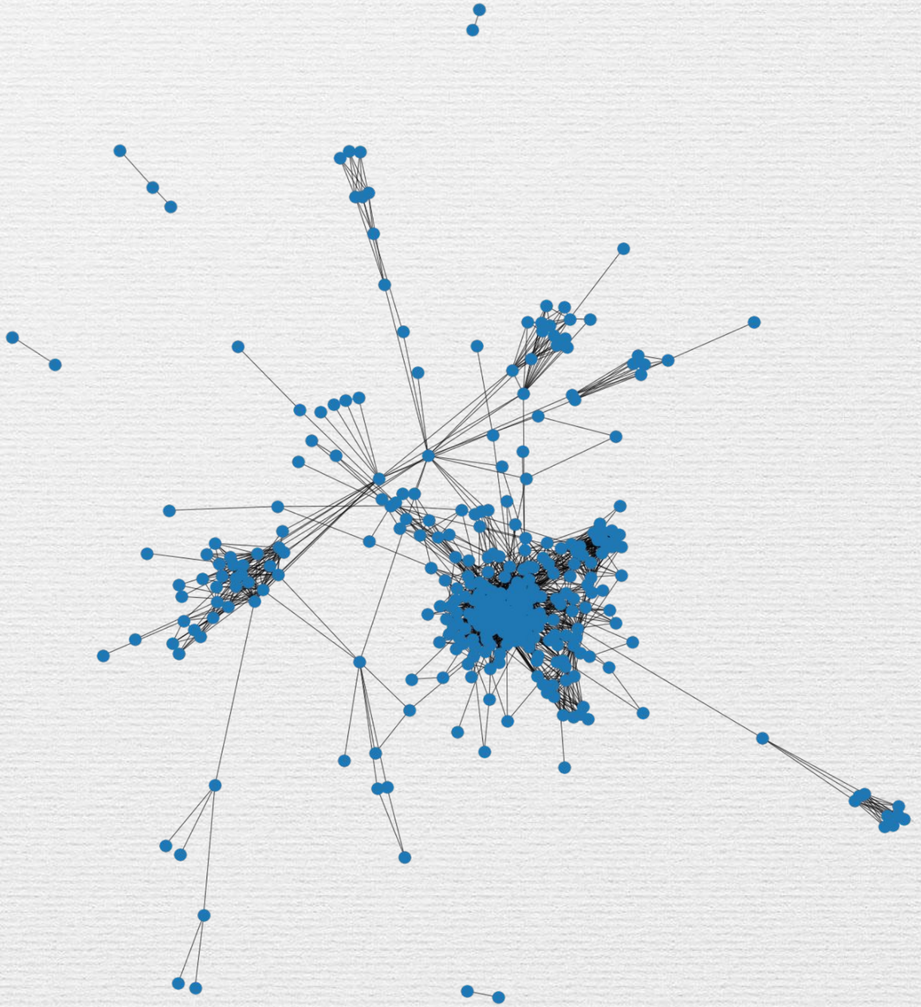
COPS

- **Required info:**
  - Observations of cascades based on randomly selected samples
- **Algorithm idea:**
  - The algorithm observes cascades and aims to select a set of nodes that are influential, but belong to different communities.
  - It picks the nodes that have the largest individual influence while avoid picking multiple nodes in the same community by pruning nodes with high influence overlap.

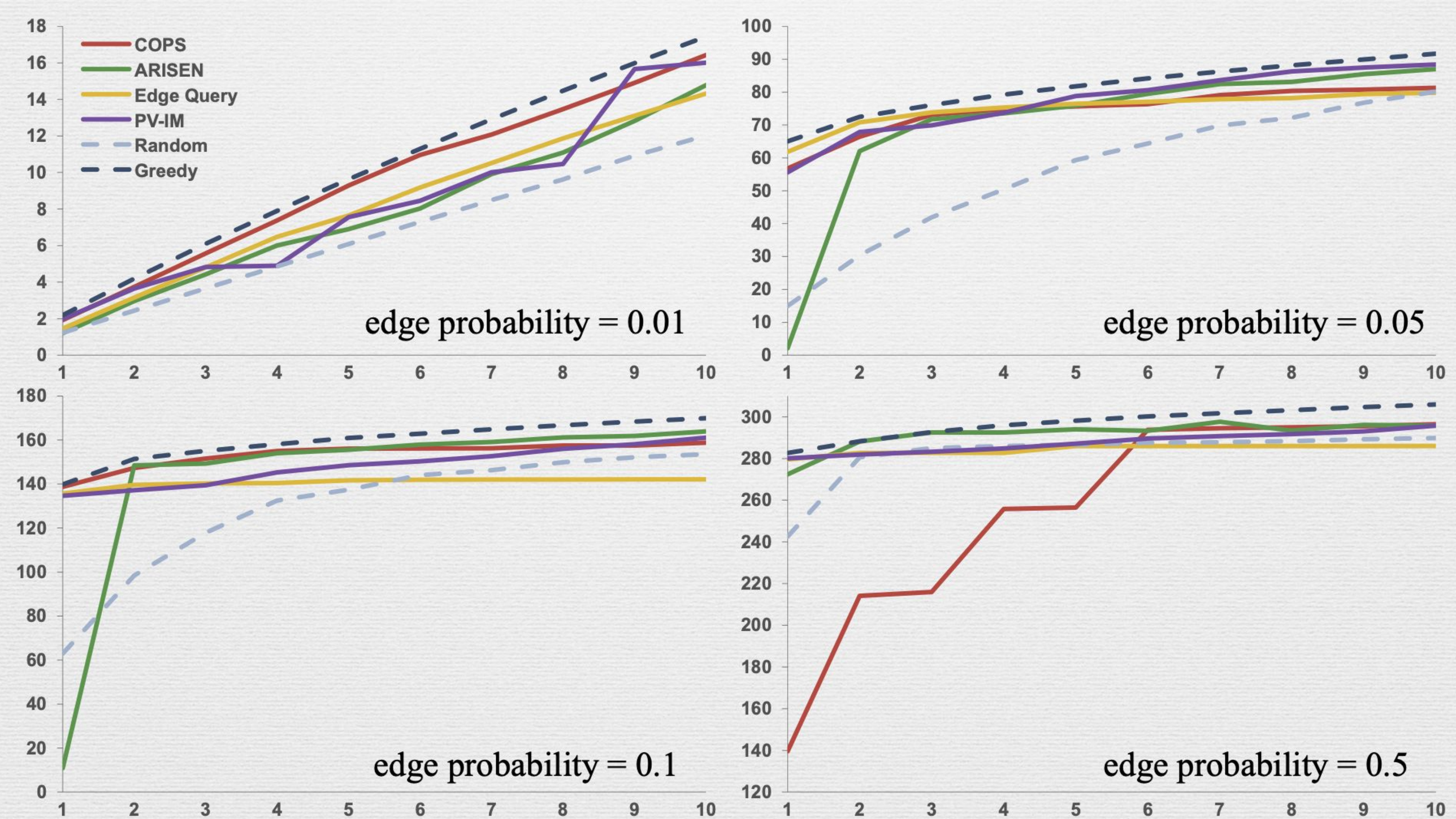
Experiments

Dataset

- **Introduction:** our dataset is from Facebook where nodes are users and edges are pairs of users who are friends on the network
- **Dataset statistics:** 333 nodes, 2519 edges



* *More details on https://snap.stanford.edu/data/ego-Facebook.html*


COPS
ARISEN
Edge Query
PV-IM
Random
Greedy
edge probability = 0.01
edge probability = 0.05
edge probability = 0.1
edge probability = 0.5

Analysis

- **COPS** and **ARISEN** consider the community structure of the networks. **COPS** aims to select nodes from different communities, while **ARISEN** gives nodes different weights based on their community size and their influence within the community. Their performance is not very stable when k is small but gets better as k increases.
- On the other hand, **Edge Query** does not consider about the community structure of the network, but repeatedly find the nodes that are most likely to have large spread size among the unselected nodes, which makes **Edge Query** performs well when k is small and edge probability is low.
- Different from the other three algorithms, **PV-IM** does not consider from community but from one totally-known part. **PV-IM** finds the most connected nodes in the defined intersection graph. So when edge probability is low, the intersection graph is small and relatively more random, **PV-IM** is not stable.

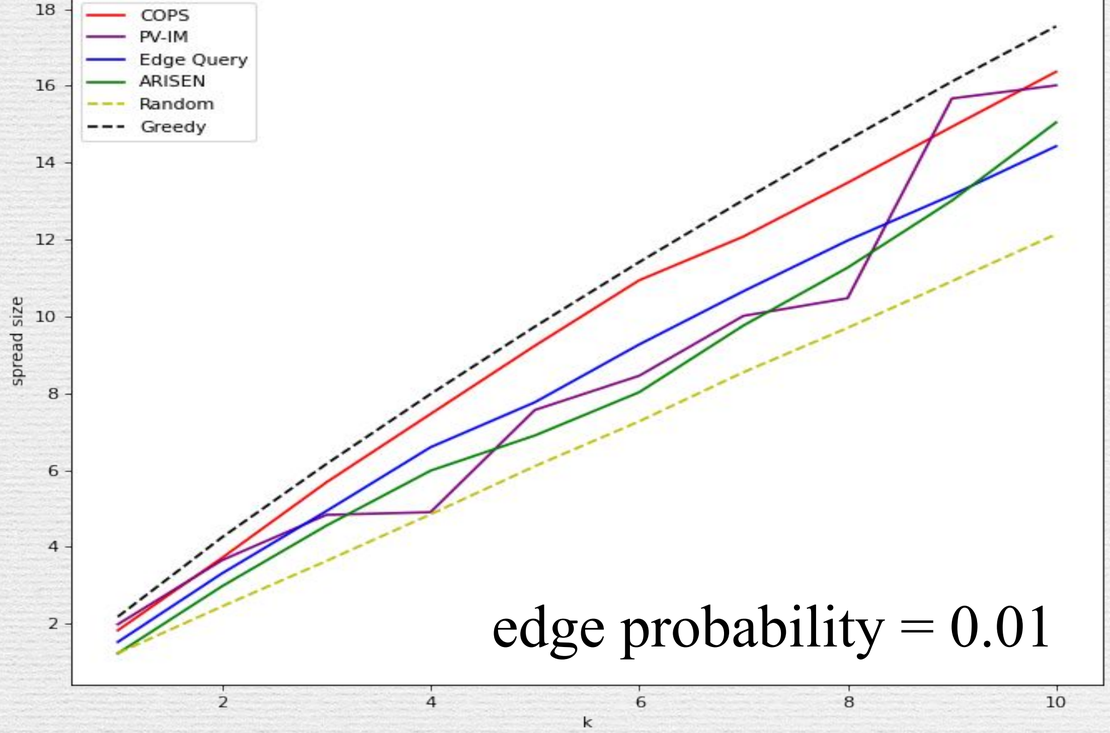
Conclusion

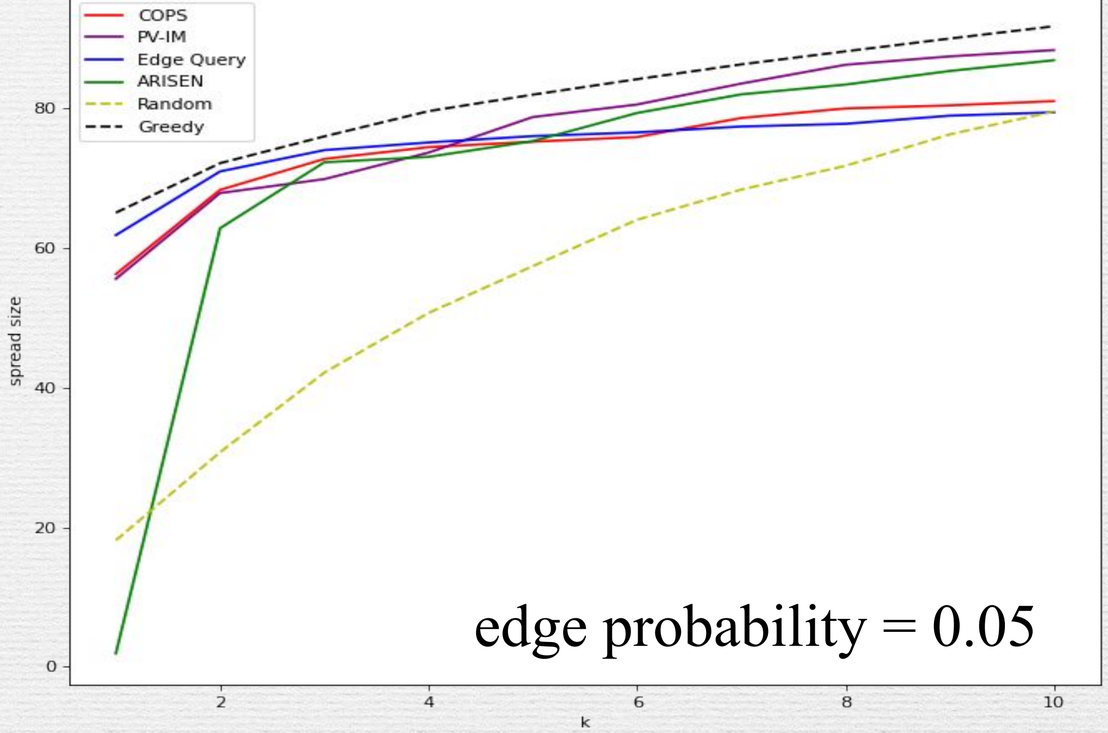
In this project, we investigated the influence maximization problem and examined four algorithms using only partial information to find the initial seed set with largest influence over the network.

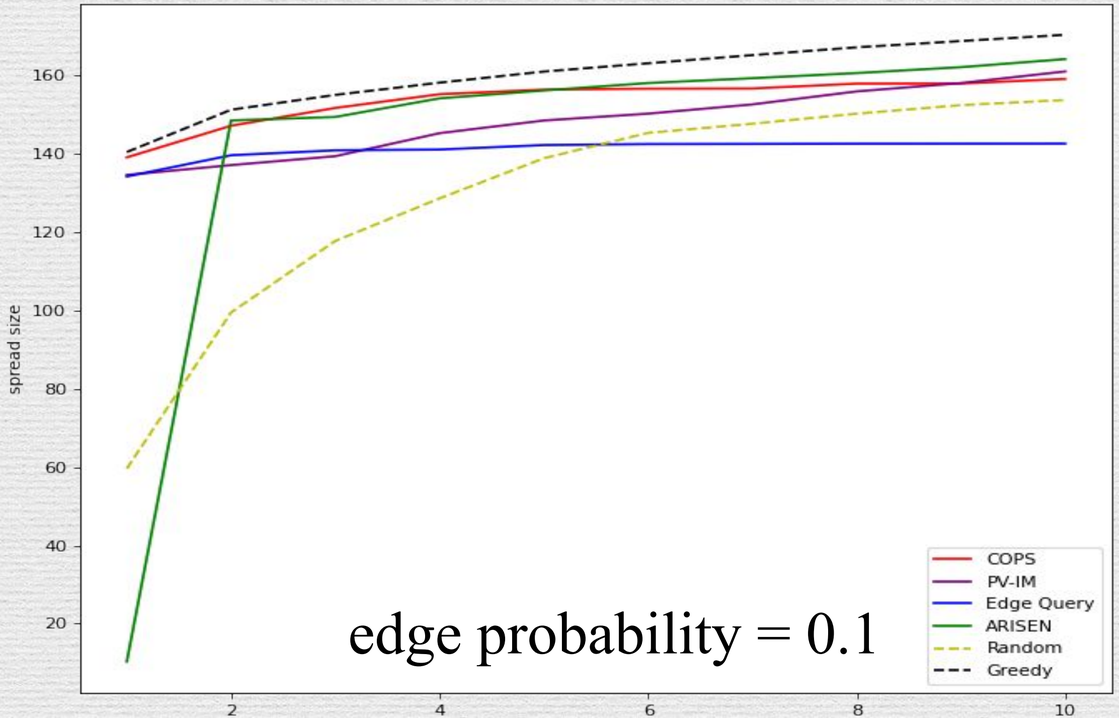
As for the experiments, we applied the algorithms on the Facebook dataset and used random and greedy algorithms as benchmarks. Results show that, overall, algorithms with partial information can achieve approximately good performance as greedy algorithm with full information.

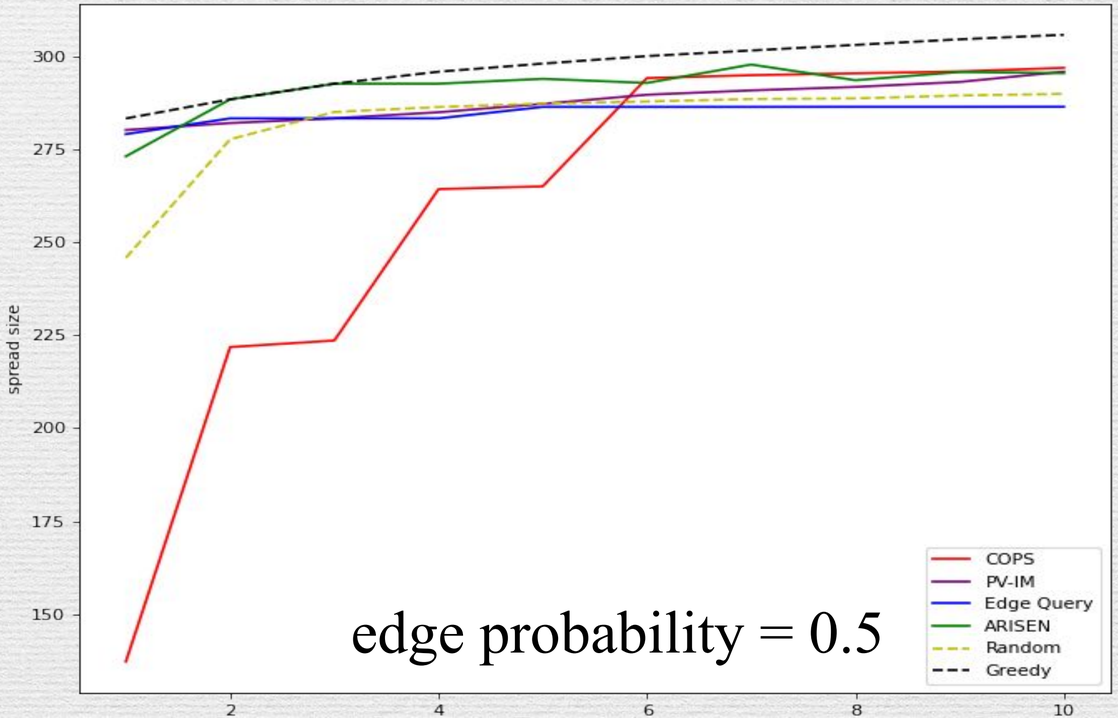
THANK YOU !!!

Graph Appendix


COPS
PV-IM
Edge Query
ARISEN
Random
Greedy
spread size
k
edge probability = 0.01


COPS
PV-IM
Edge Query
ARISEN
Random
Greedy
spread size
k
edge probability = 0.05


COPS
PV-IM
Edge Query
ARISEN
Random
Greedy
spread size
edge probability = 0.1


COPS
PV-IM
Edge Query
ARISEN
Random
Greedy
spread size
edge probability = 0.5