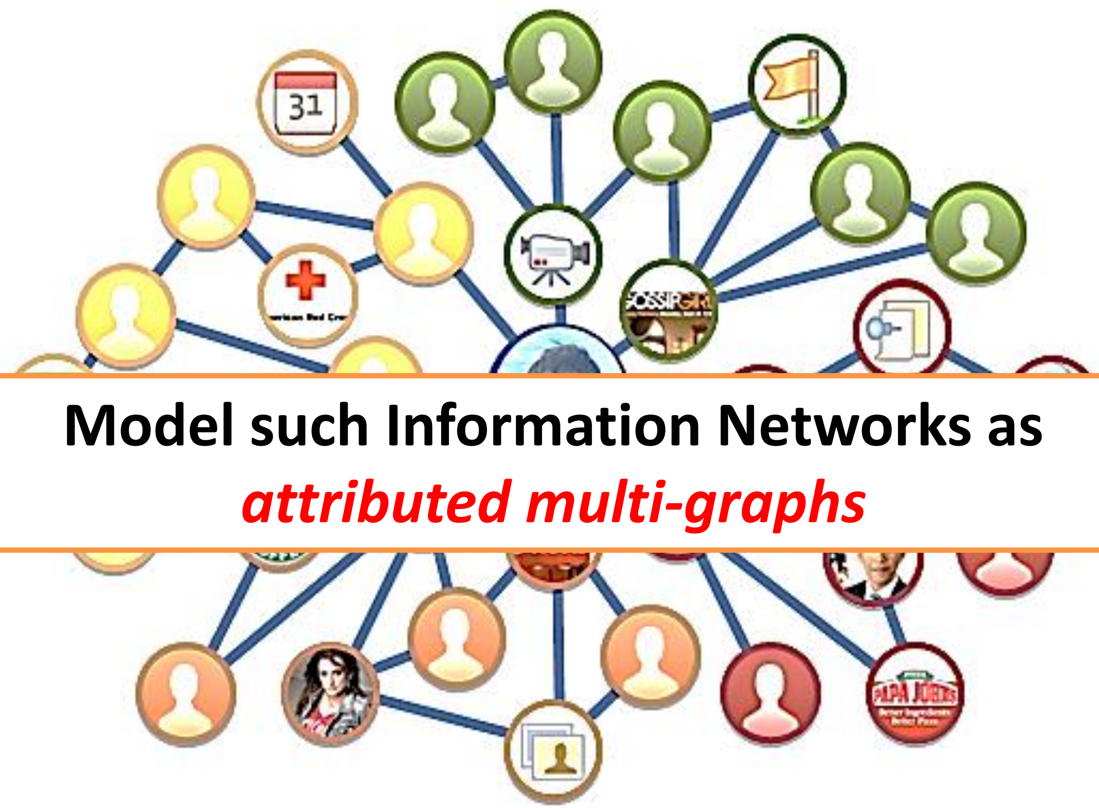
A. Papadopoulos, G. Pallis, M. D. Dikaiakos

Identifying Clusters with Attribute Homogeneity and Similar Connectivity in Information Networks

IEEE/WIC/ACM International Conference on Web Intelligence
Nov. 17-20, 2013
Atlanta, GA
USA

The Real World: Information Networks



Model such Information Networks as
attributed multi-graphs

An Online Social Network

Clustering

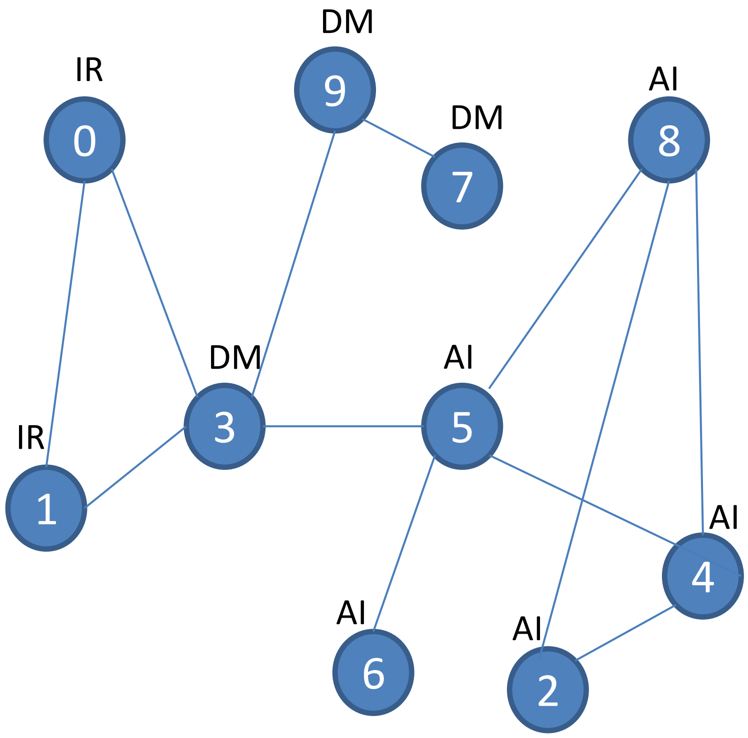
- The process of identifying groups of related data/objects in a dataset/information network
- Why? Discover hidden knowledge!

Network		**Applications**
Co-author Networks	Clustering →	**Recommending new collaborations**
Social Networks		**Recommend friendships, group targeted advertisement**

Challenges

- A vertex may belong to more than one cluster
  - Fuzzy clustering

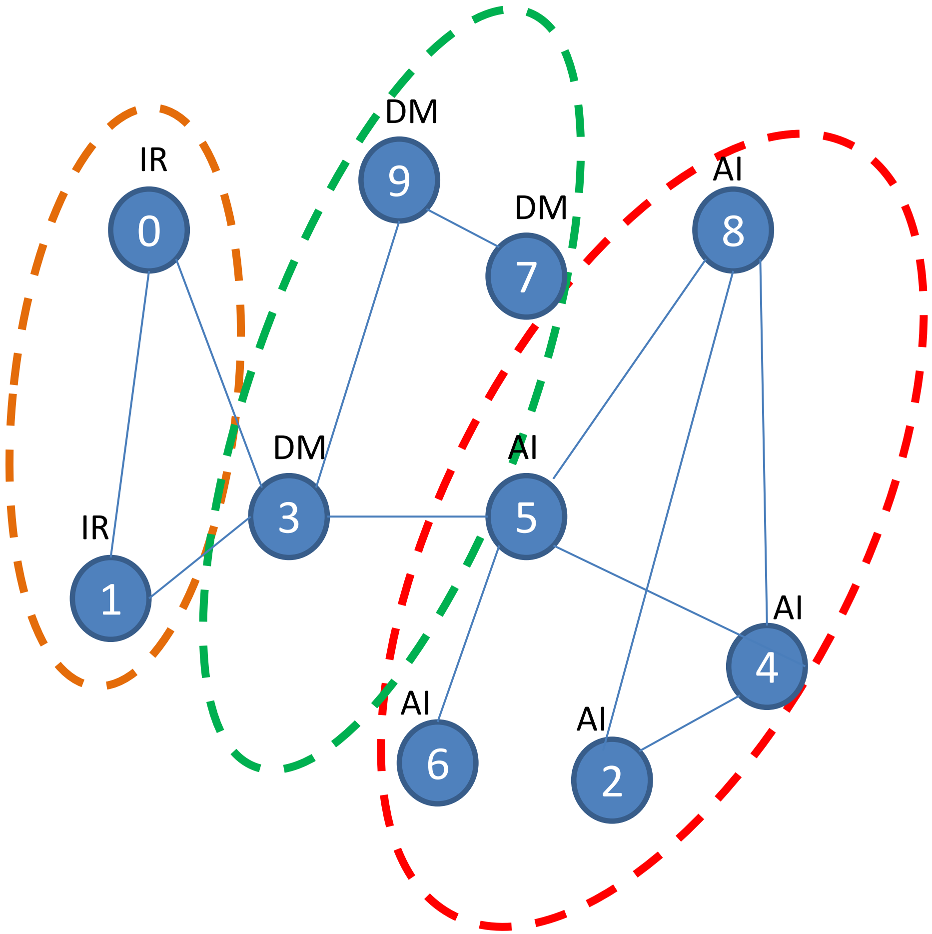
- Cluster based on:
  - Structure
  - Attributes



Challenges

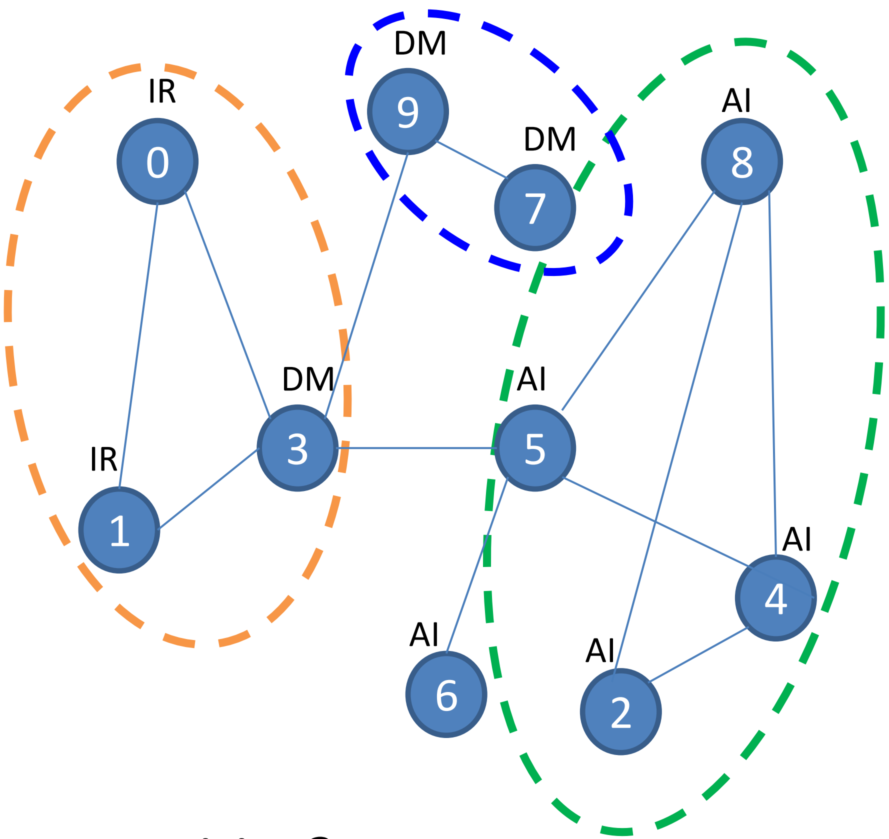
Cluster based on:

- Attributes
- Structure



Challenges

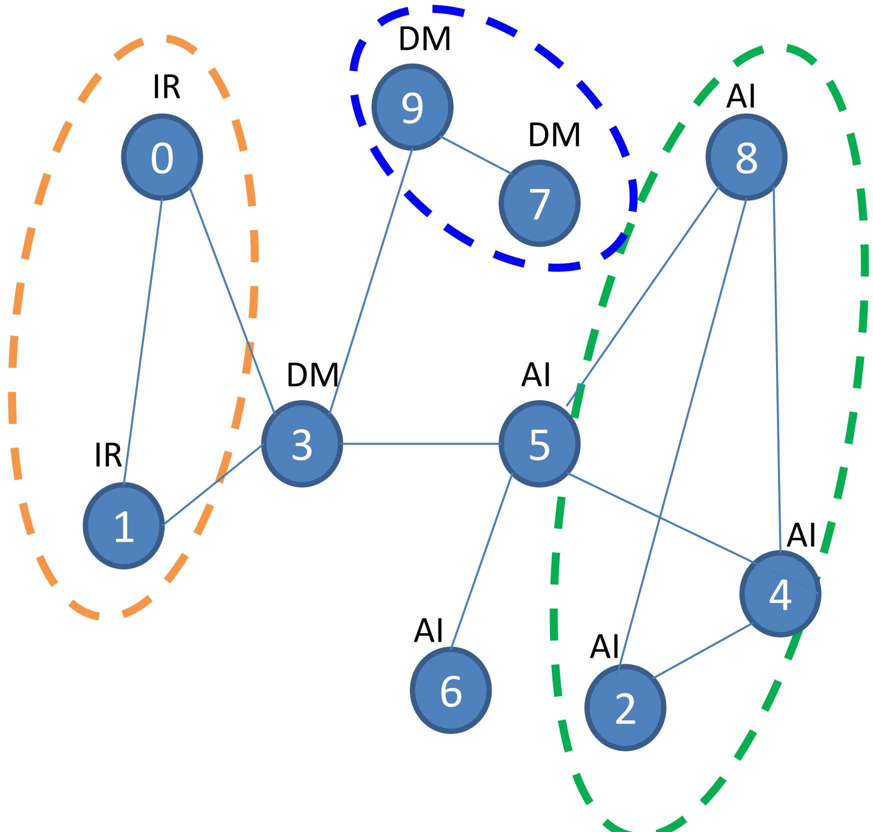
Cluster based on:
- Attributes
- Structure



Communities?

Challenges

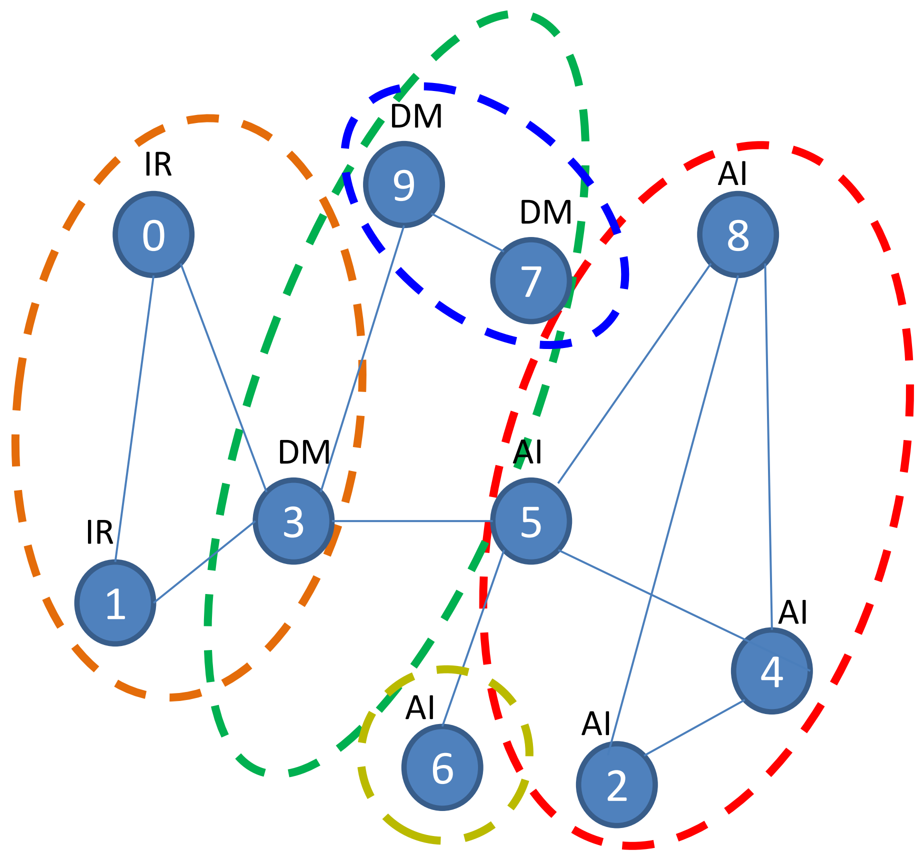
Cluster based on:
- Attributes
- Structure



Similar Connectivity?

Challenges

Cluster based on:
- Structure
- Attributes



Challenges

- How to balance the attribute and structural properties of the vertices?
- How to identify which link type is more important?
  - A request to join a political group is more important than sharing a funny video
- How to identify which attribute is more important?
  - The attribute political views of a person is clearly more important than its name or gender

Related Work

Distance Based

- **SA-Cluster (ACM TKDD 2011)**
  - Graph augmentation with attributes and random walks
  - Different attributes importance
- **PICS (SIAM SDM 2012)**
  - MDL Compression
  - Similar connectivity
  - Parameter Free

Model Based

- **BAGC (SIGMOD 2012)**
  - Bayesian Inference Model
  - Directed graphs
- **GenClus (VLDB 2012)**
  - EM algorithm
  - Multi-graphs
  - Different link types importance

HASCOP

HASCOP

Objective Function
Similar Connectivity
Attribute Coherence
Weight Adjustment Mechanism
Clustering Process

HASCOP

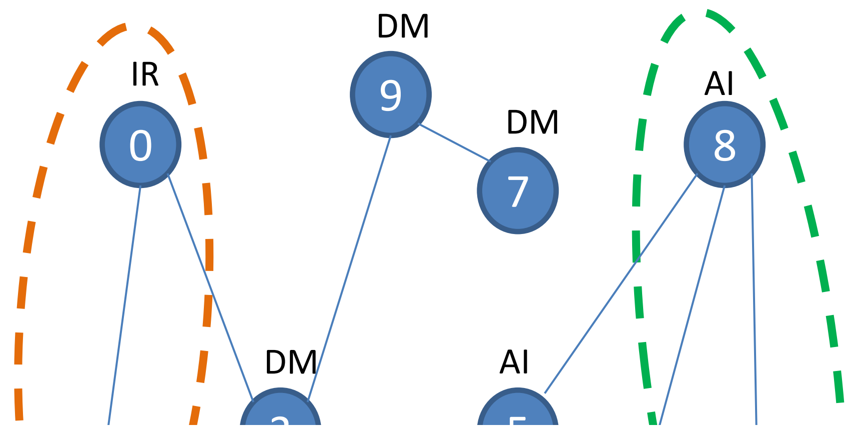
Assigns vertices in the same cluster so as to exhibit **both similar connectivity** and **attribute coherence**

- *Given function $s(v_i, c_j)$ the clustering objective function is:*

$$O(\Theta, \vec{\omega_t}, \vec{w_\alpha}) = \sum_{i=1}^{|V|} \sum_{j=1}^{k} \Theta_{i,j} \cdot s(v_i, c_j, \vec{\omega_t}, \vec{w_a})$$

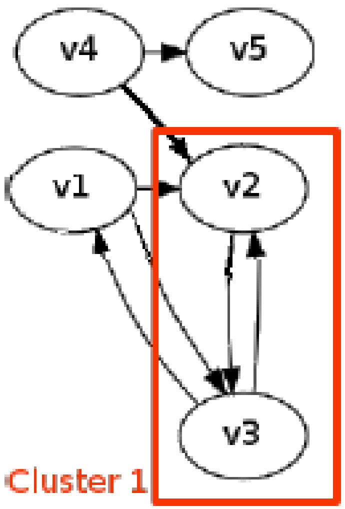
Similar Connectivity

- Two vertices v_i, v_j have similar connectivity pattern if $S(v_i)$ and $S(v_j)$ highly



Similar Connectivity represents how similar two vertices are based on their outgoing links

Similar Connectivity



(a) Example graph

L^0	v_1	v_2	v_3	v_4	v_5
c_1	1	1	1	0	0

v_1	1	1	1	0	0
v_2	0	1	1	0	0
v_3	1	1	1	0	0
v_4	0	1	0	1	1
v_5	0	0	0	0	1

(b) Cluster c_1 properties and adjacency matrix.

$$link\_sim(v_1, c_1) = 1$$

$$link\_sim(v_5, c_1) = \frac{1}{3}$$

(c) Similar Connectivity

$$link\_sim(v_i, c_j) = \frac{}{1 + \sqrt{\sum_{x=1}^{|V|} \left(L_{i,x} - C_{j,x}^{links}\right)^2}}$$

Attribute Coherence

- Weighted Euclidean distance
- It is close to one if the attribute vector of v_i is very close to the attribute centroid of c_j

$$attr\_sim(v_i, c_j, \vec{w_\alpha}) = \frac{1}{1 + \sqrt{\sum_{l=1}^{p} w_{\alpha_l} \cdot \left(A_{i,l} - C_{j,l}^{attr}\right)^2}}$$

HASCOP: Approach

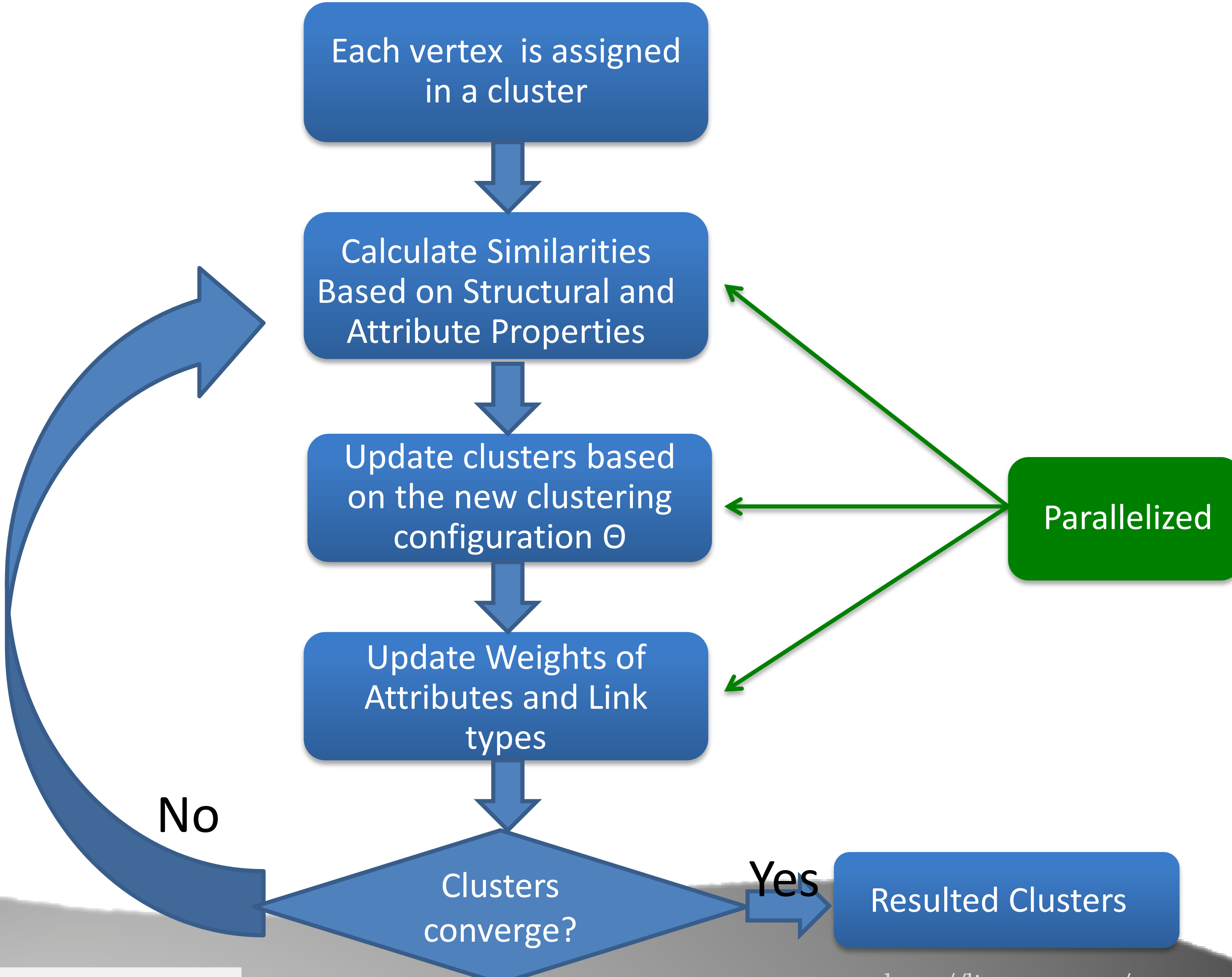
- A vertex has high similarity with a cluster if **both** their similar connectivity and attribute coherence are high.

$$\boxed{s(c_j, v_i, \vec{w_a}) = link\_sim(v_i, c_j) \cdot attr\_sim(v_i, c_j, \vec{w_\alpha})}$$

Weight Adjustment

- Voting mechanism
- The weights are adjusted towards the direction of increasing the clustering objective function:
  - If vertices in the same cluster are connected by link-type A then the weight of link-type A is increased
  - If vertices in the same cluster share the same value for an attribute X then the weight of attribute X is increased

Clustering Process



Evaluation

Datasets

Evaluation Measures

Evaluations

Datasets

GoogleSP-23: Google Software Packages

- Built from software files installed on Cloud
- Software files are **not** densely connected components

- Vertex: software file
- Attributes:
  - File Size
  - File Type
  - Last Access Time
  - Last Content Modified Time
  - Time of the most recent metadata change
- Link-types:
  - File name similarities
  - File path similarities

Datasets

DBLP: Bibliography Network

- Vertex: author
- Attributes:
  - Number of publications
  - Research area
- Link-types:
  - Co-author relationship

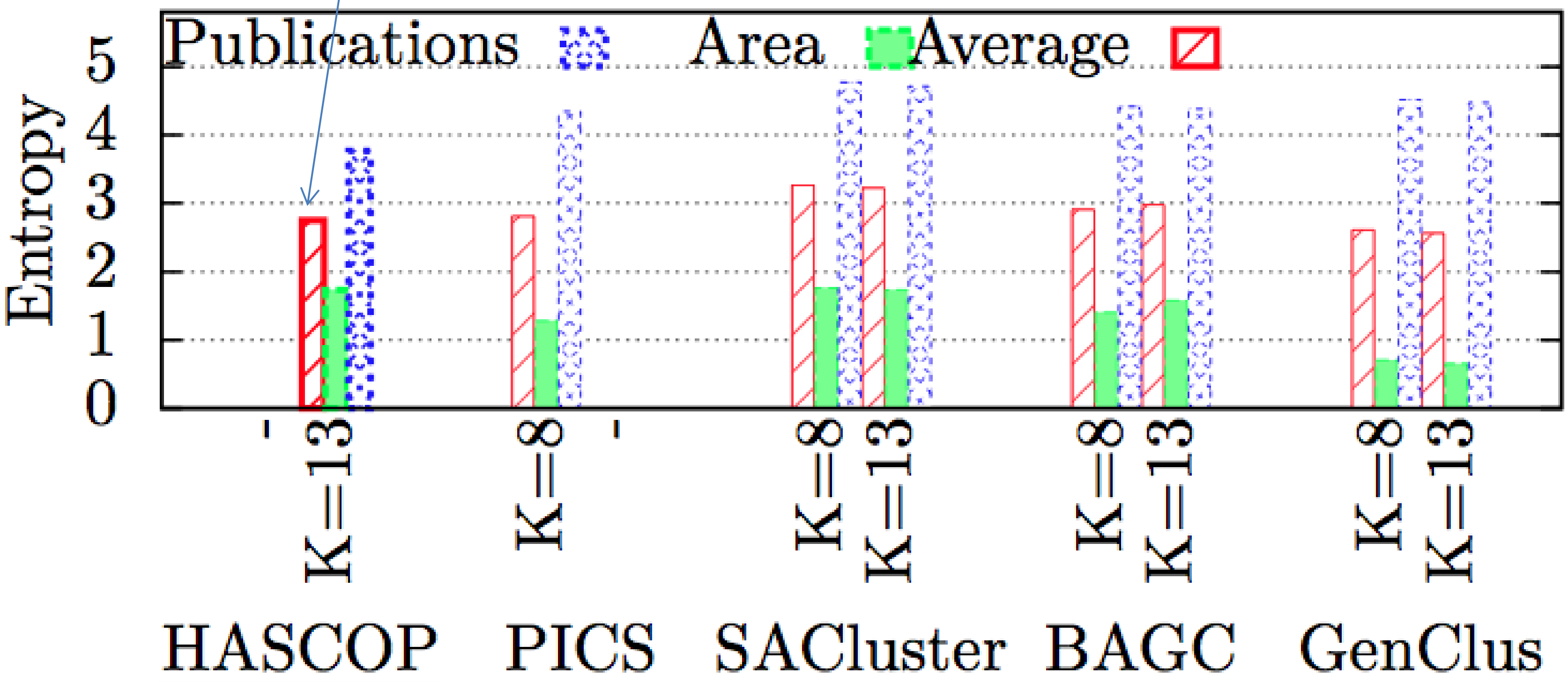
Dataset	DBLP-1000	GoogleSP-23
Nodes	1000	1297
Edges	17128	24153
Attributes	2	5
Link Types	1	2
Type of Graph	Undirected	Undirected

Evaluation Measures

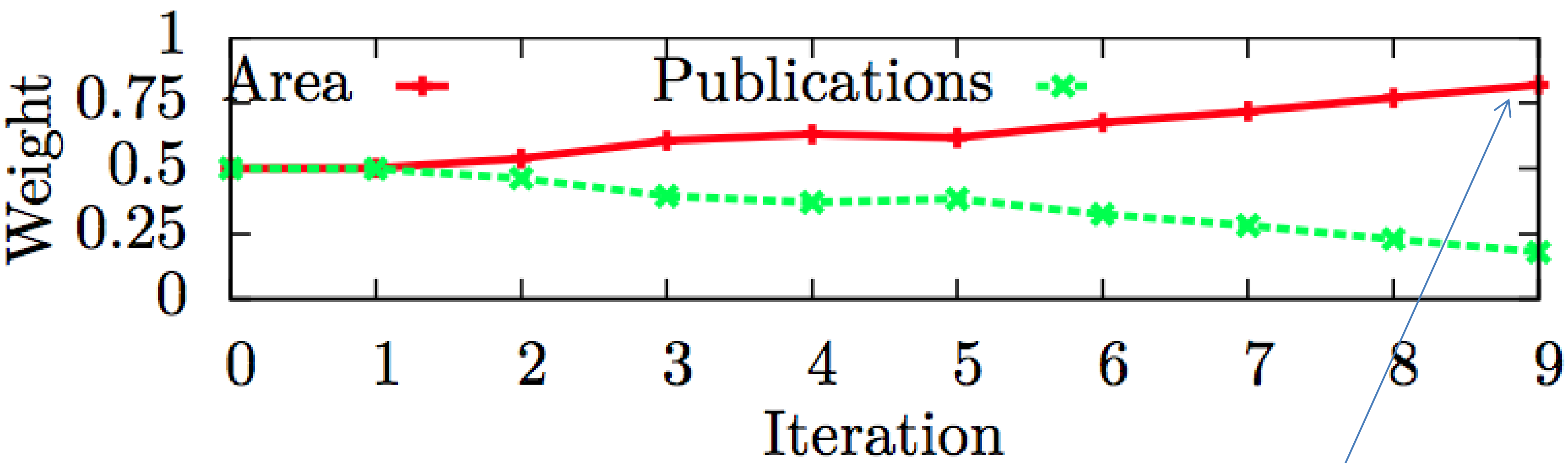
- Entropy
  - Attribute properties
  - Close to zero for attribute cohesive clusters
- For GoogleSP-23 dataset we measure:
  - The percentage of clusters overlapping with a software package
  - The percentage of software packages that were actually identified

Evaluation – DBLP-1000

Very close to the lowest average entropy



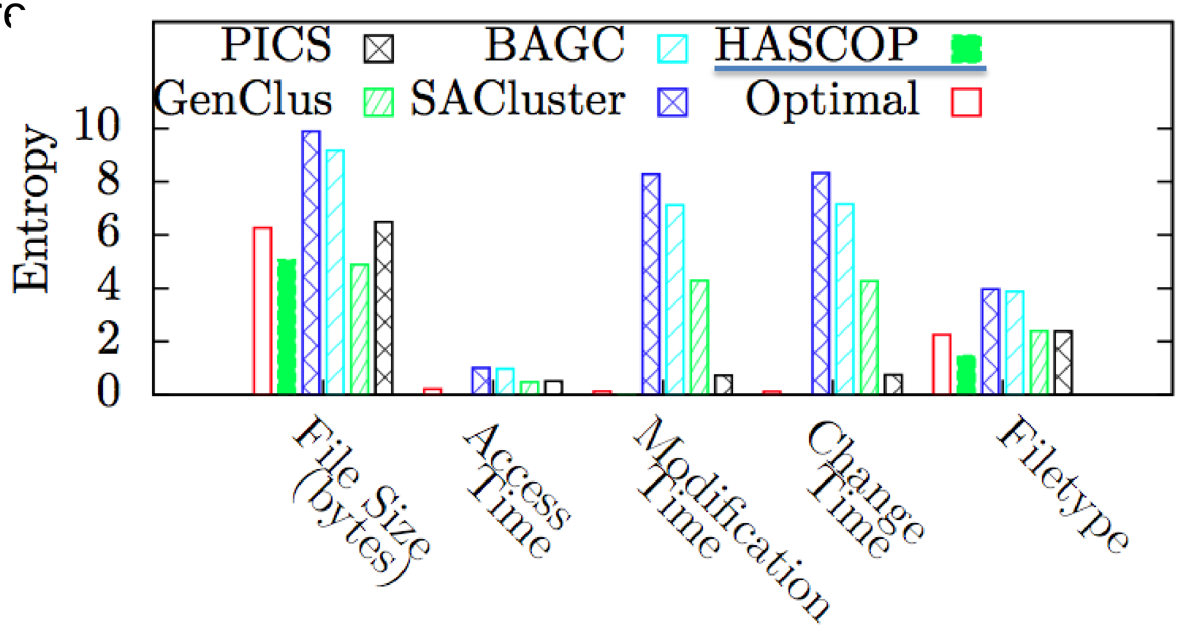
Evaluation – DBLP-1000



Successfully identified the importance of "Area of interest" against "No of publications"

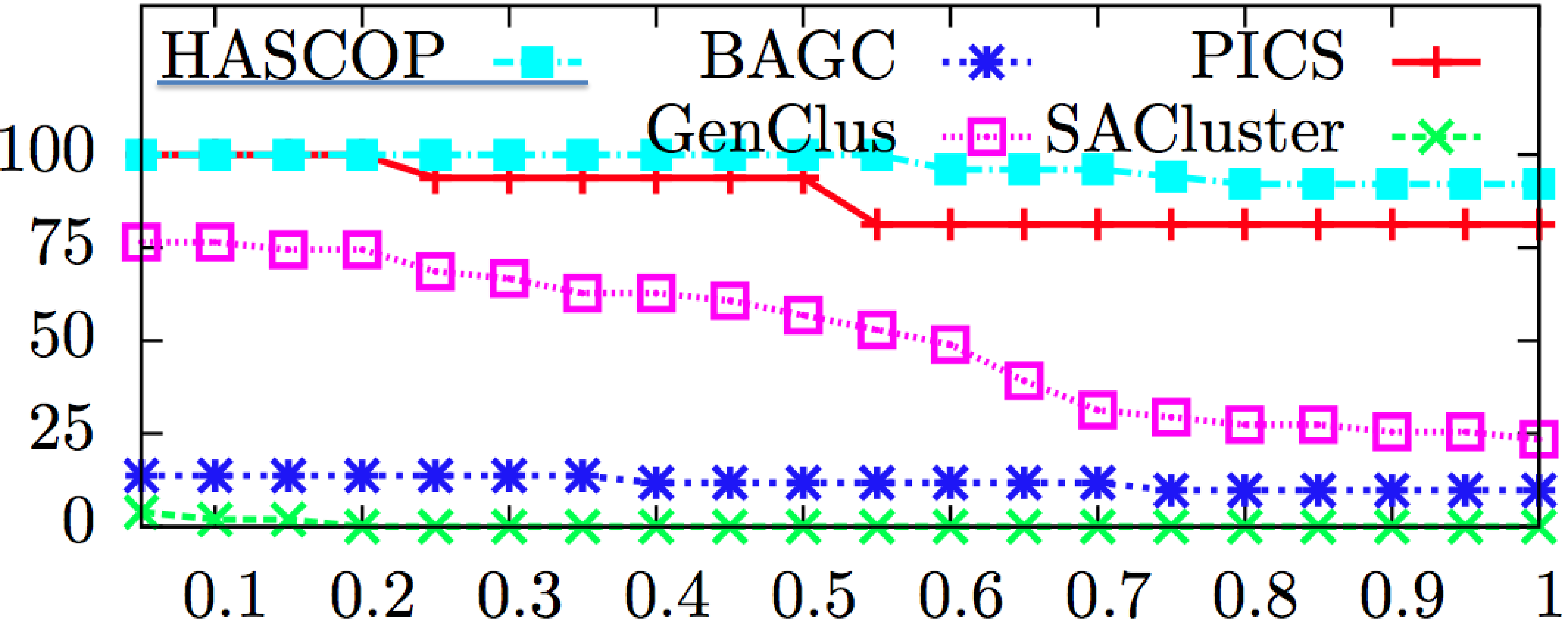
Evaluation – GoogleSP-23

- Comparison to the "ground truth"
- Must identify the software packages
- HASCOP is **closest to the "optimal"** entropy



Evaluation – GoogleSP-23

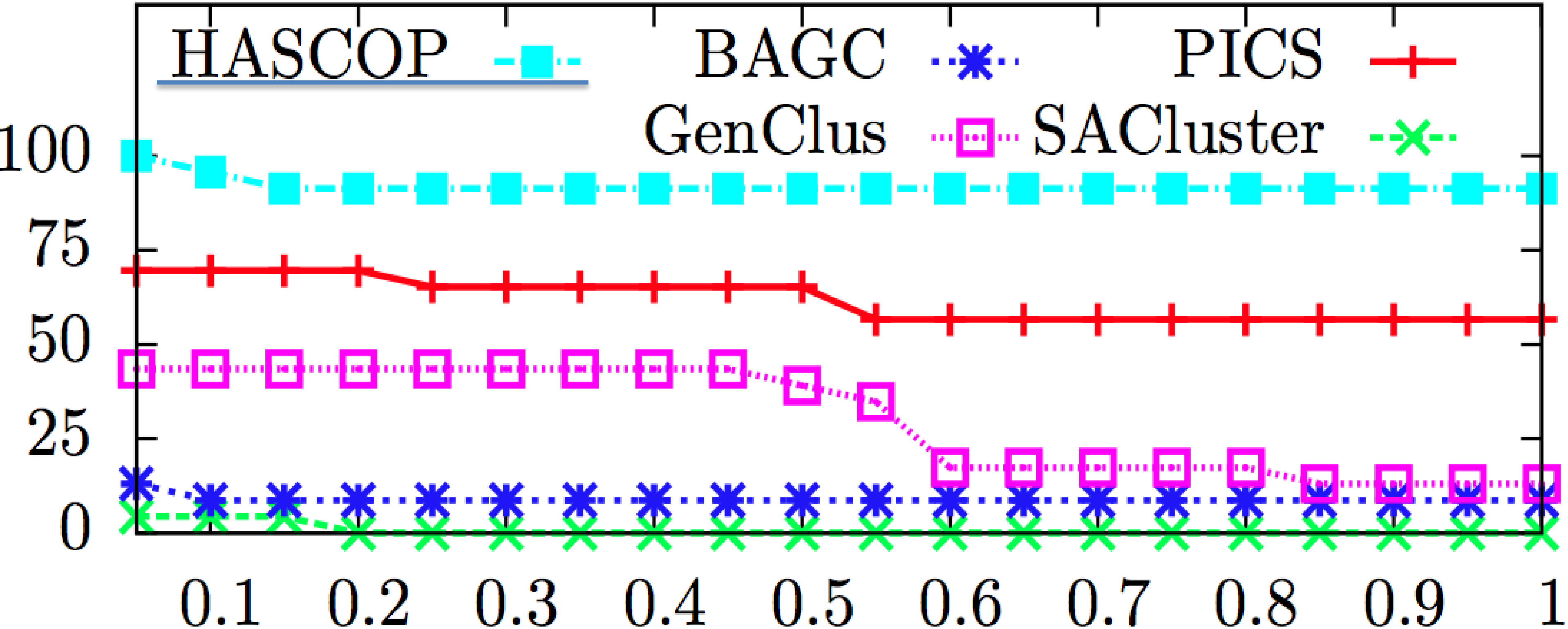
- HASCOP found 51 clusters
- More than **80%** of returned clusters by HASCOP and PICS are consisted of files from the same software packages



Evaluation – GoogleSP-23

- Almost all clusters (**>90%**) returned by HASCOP have full overlap with a software package

- Almost all (**21 of 23**) software packages have been identified



Conclusions

Conclusions

Future Work

Conclusions

- HASCOP succeeded in returning clusters useful to many applications studying such information networks
  - Correctly identified software packages installed on a Cloud infrastructure
- Experiments confirmed that HASCOP finds clusters characterized by **attribute homogeneity**
- **Similar Connectivity** is important

Future Work

- Integrate into MinerSoft[1] (a software file search engine)
- Extend HASCOP to handle:
  - Weighted multi-graphs
  - Heterogeneous information networks
  - Deploy to a large scale Hadoop cluster

1: Minersoft is available at: http://euclid.grid.ucy.ac.cy:1997/MinerSoft/SimpSearch

A. Papadopoulos, G. Pallis, M. D. Dikaiakos

{ andpapad, gpallis, mdd } @ cs.ucy.ac.cy

Identifying Clusters with Attribute Homogeneity and Similar Connectivity in Information Networks

Thank You!

Laboratory for Internet Computing
Department of Computer Science
University of Cyprus
http://linc.ucy.ac.cy

References

[1] L. Akoglu, H. Tong, B. Meeder, and C. Faloutsos. Pics: Parameter-free identification of cohesive subgroups in large attributed graphs. In *Proceedings of the 12th SIAM International Conference on Data Mining, SDM 2012,* Anaheim, CA, April 2012.

[2] H. Cheng, Y. Zhou, and J. X. Yu. Clustering large attributed graphs: A balance between structural and attribute similarities. *ACM Trans. Knowl. Discov. Data,* 5(2):12:1–12:33, Feb. 2011.

[3] I. Choi, B. Moon, and H.-J. Kim. A clustering method based on path similarities of xml data. *Data Knowl. Eng.,* 60(2):361–376, Feb. 2007.

[4] M. D. Dikaiakos, A. Katsifodimos, and G. Pallis. Minersoft: Software retrieval in grid and cloud computing infrastructures. *ACM Trans. Internet Technol.,* 12(1):2:1–2:34, July 2012.

[5] S. Jenkins and S. Kirk. Software architecture graphs as complex networks: A novel partitioning scheme to measure stability and evolution. *Information Sciences,* 177(12):2587 – 2601, 2007.

References

[6] S. E. Schaeffer. Graph clustering. *Computer Science Review,* 1(1):27 – 64, 2007.

[7] Y. Sun, C. C. Aggarwal, and J. Han. Relation strength-aware clustering of heterogeneous information networks with incomplete attributes. *Proc. VLDB Endow.,* 5(5):394–405, Jan. 2012.

[8] Z. Xu, Y. Ke, Y. Wang, H. Cheng, and J. Cheng. A model-based approach to attributed graph clustering. In *Proceedings of the 2012 international conference on Management of Data,* SIGMOD ’12, pages 505–516, New York, NY, USA, 2012. ACM.

[9] X. Zheng, D. Zeng, H. Li, and F. Wang. Analyzing open-source software systems as complex networks. *Physica A: Statistical Mechanics and its Applications,* 387(24):6190– 6200, 2008.

[10] Y. Zhou, H. Cheng, and J. Yu. Clustering large attributed graphs: An efficient incremental approach. In *Data Mining (ICDM), IEEE 10th International Conference on,* pages 689–698, Dec. 2010.

[11] Y. Zhou, H. Cheng, and J. X. Yu. Graph clustering based on structural/attribute similarities. *Proc. VLDB Endow.,* 2(1):718–729, Aug. 2009.