Wrocław University of Technology

Klasyfikacja relacyjna

metody klasyfikacji kolektywnej, modele złożone, fuzja informacji w sieciach złożonych, uczenie i wnioskowanie aktywne w sieciach złożonych

Tomasz Kajdanowicz

Wrocław, 18.03.2014

Outline

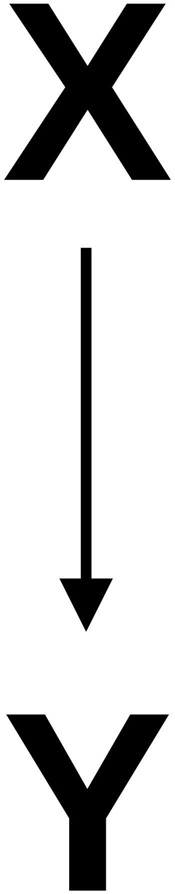
- Collective classification
- Ensemble methods
- Active learning and inference

Wrocław University of Technology

Collective classification

1. Kajdanowicz T., Kazienko P., Litwin K., Doskocz P.: Label-dependent Node Classification in the Social Network. Neurocomputing, 75(1), 2012, 199-209.
2. Kajdanowicz T., Kazienko P.: A Method of Label-dependent Feature Extraction in Social Networks. ICCCI'2010, LNAI 6422, Springer, 2010, 11-21.
3. Kajdanowicz T., Kazienko P., Doskocz P.: Label-dependent Feature Extraction in Social Networks for Node Classification. SocInfo'2010, LNAI 6430, Springer, 2010, 89-102.
4. Kajdanowicz T., Kazienko P., Doskocz P., Litwin K.: An Assessment of Node Classification Accuracy in Social Networks using Label-dependent Feature Extraction. WSKS 2010, CCIS 111, Springer, 2010, pp. 125-130.

The classification problem

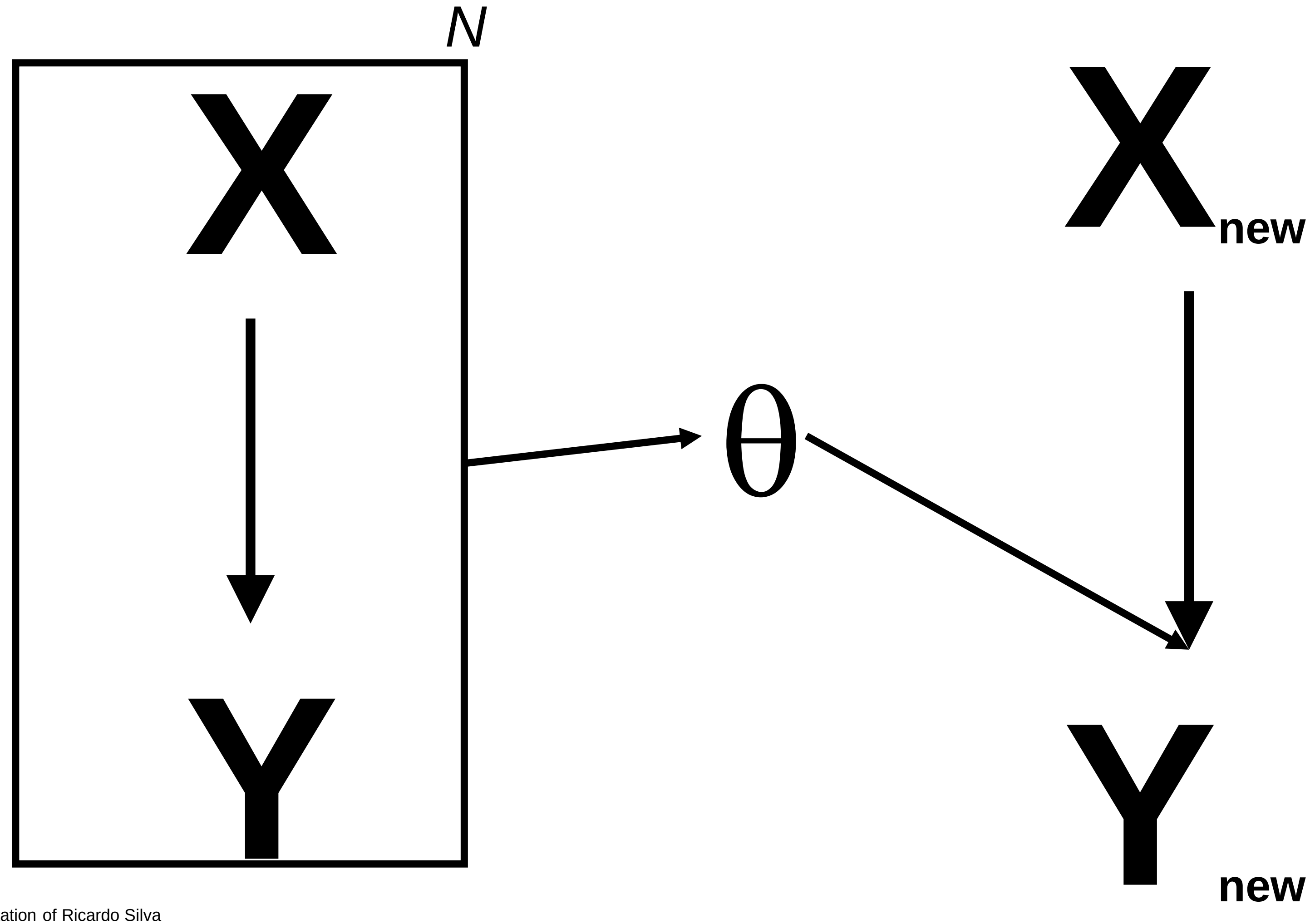


X

Y

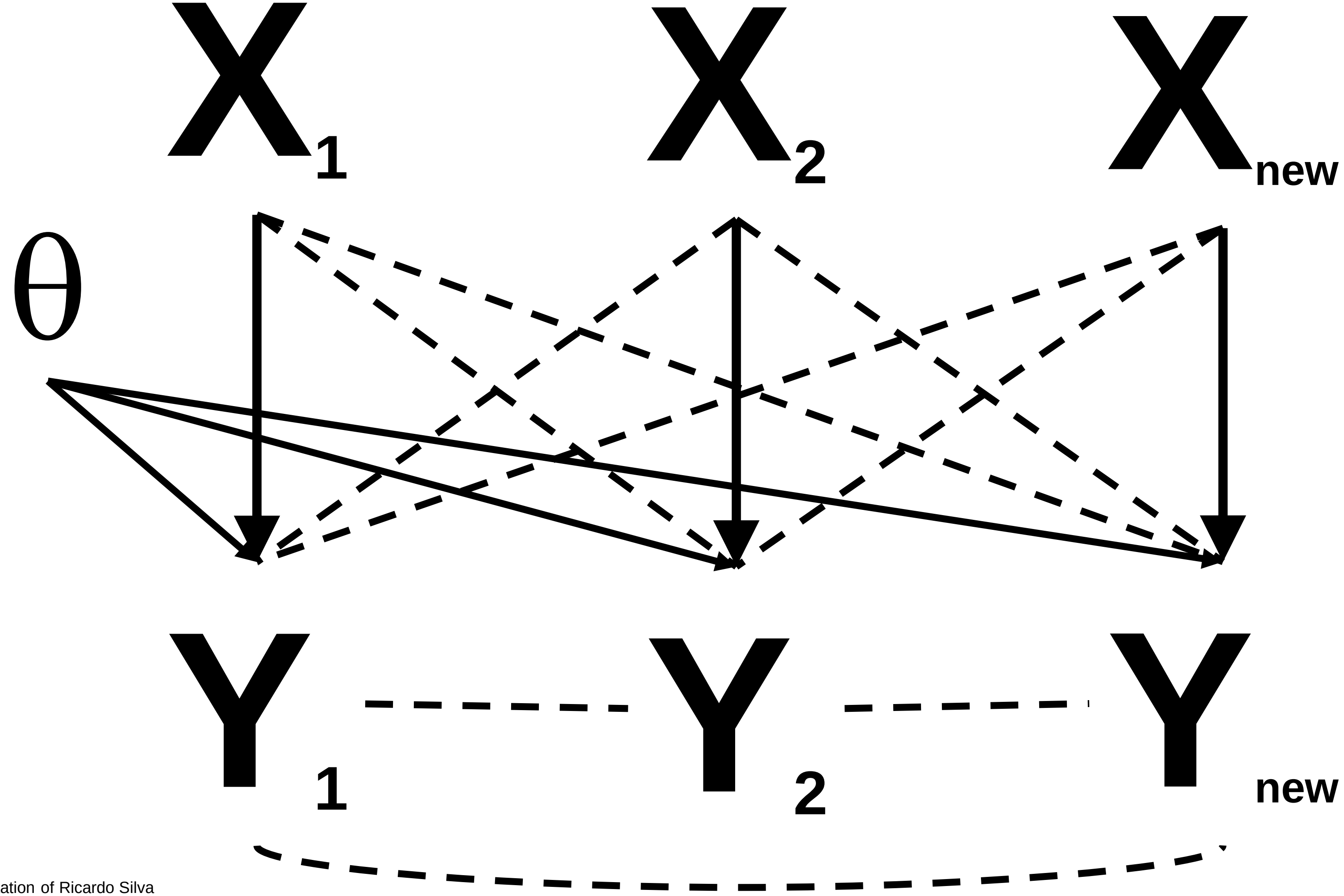
Based on presentation of Ricardo Silva

Standard setup



Based on presentation of Ricardo Silva

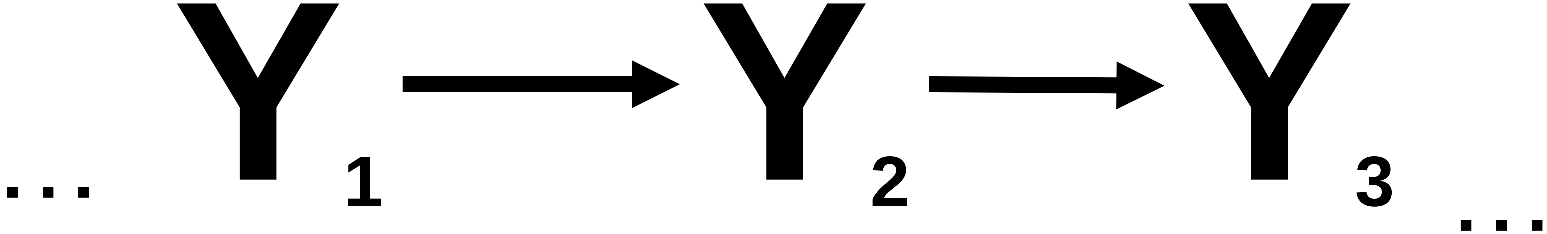
Prediction with non-iid data



Based on presentation of Ricardo Silva

Basic relational problem: time-series

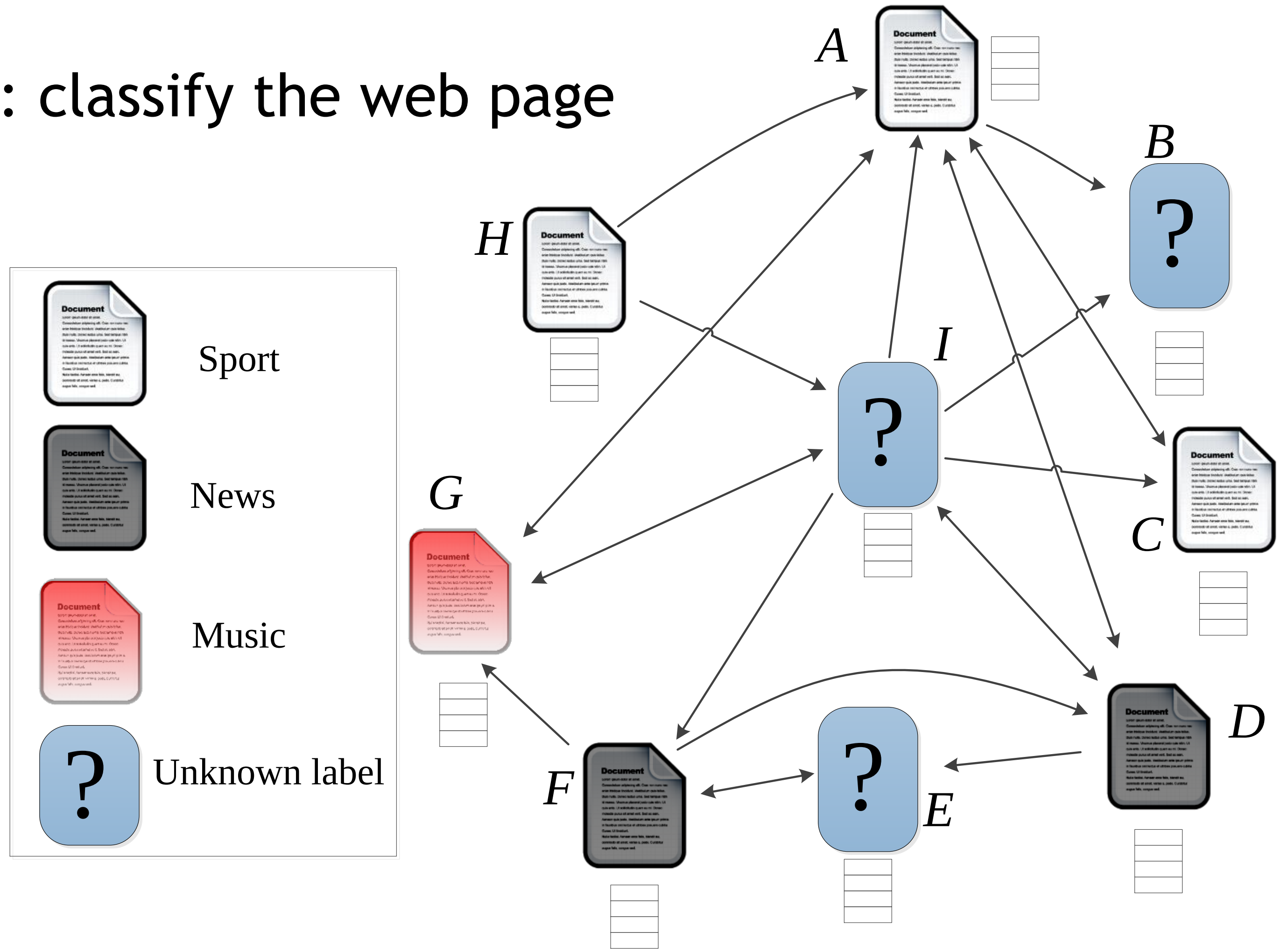
- Relations: “Y_i precedes Y_{i+k}”, $k > 0$
- Dependencies: “Markov structure *G*”



Based on presentation of Ricardo Silva

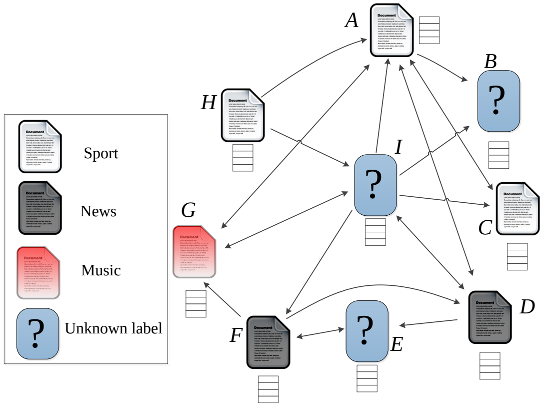
Introduction to Collective Classification

Task: classify the web page



Problem formulation

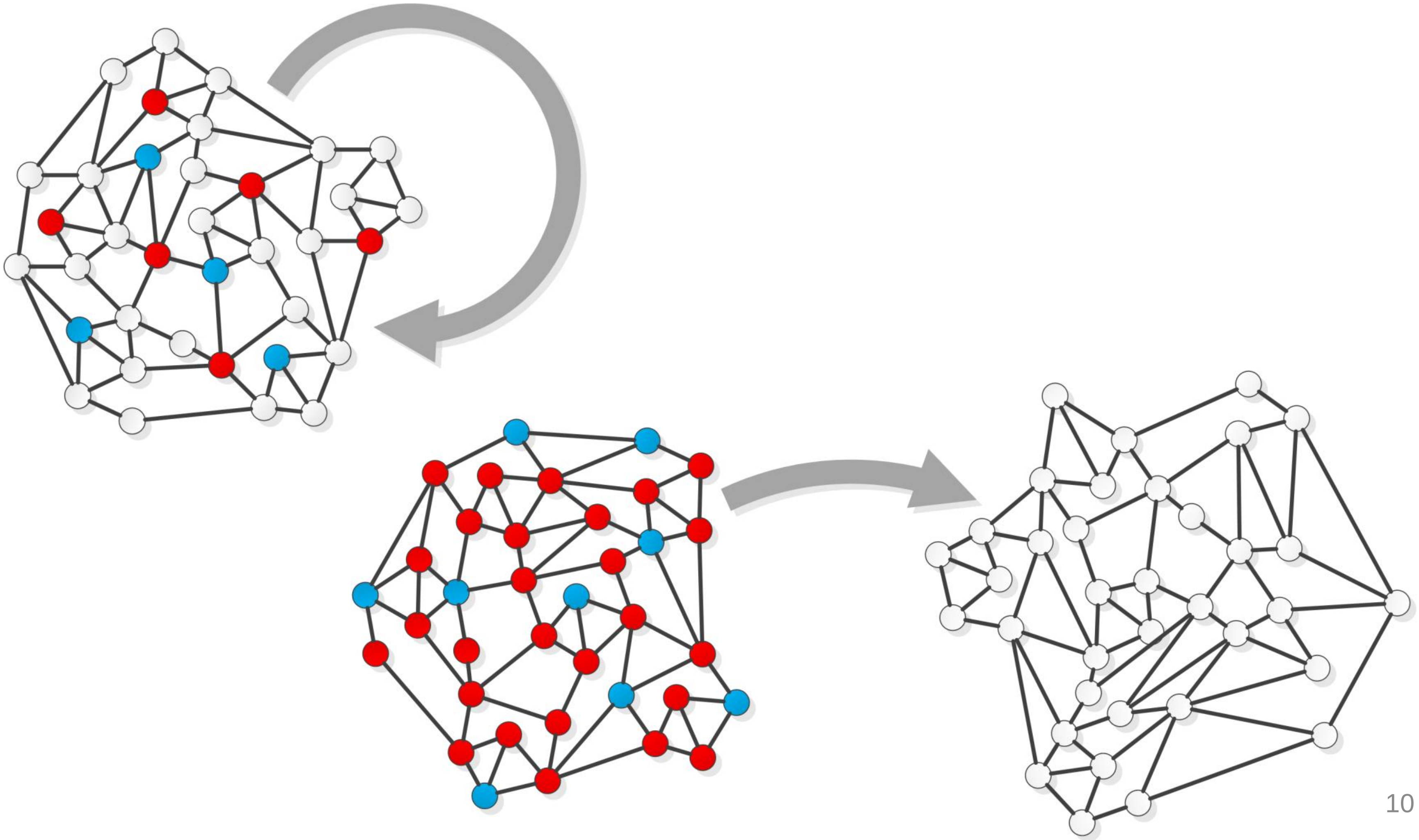
- Graph *G*=(*V*, *E*, *X*, *Y*)
  - *V* - vertices
  - *E* – edges
  - *X* – attribute space
  - *Y* – *label space*



- each vertix $v_i \in V$ is described by a feature vector $x_i \in$ X and his class label $y_i \in Y$

- *Find an inference function to assign labels*

Within vs. Across Network Classification



Basic assumptions

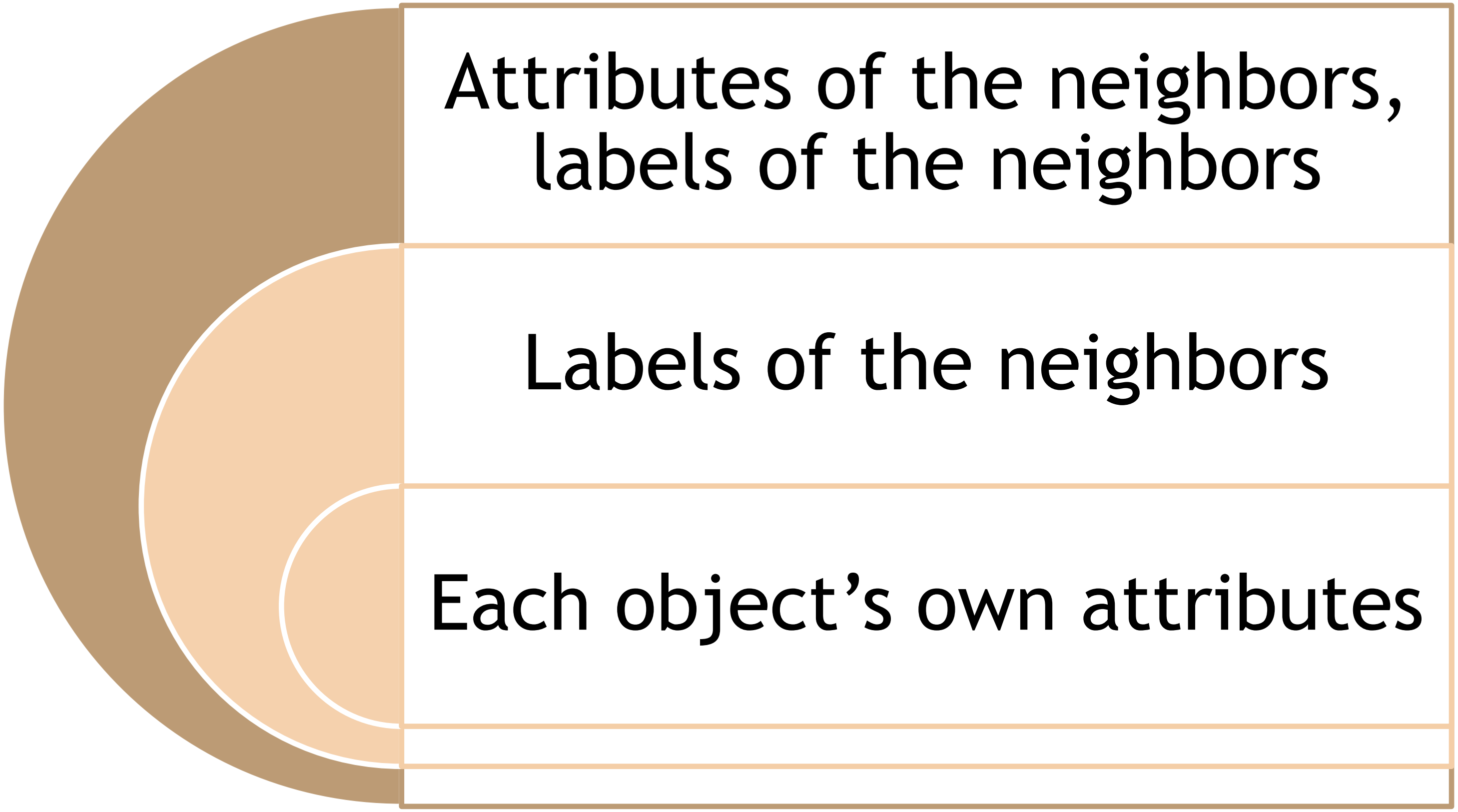
- TRADITIONALLY
  - classification algorithms considered the data do be drawn independently and identically from some distribution (i.i.d.)
  - algorithms treated the data as there were not any dependencies between vertices
- BUT
  - there exist dependencies between users which violates the i.i.d. assumption
  - label y_i does not depend on features x_i only
  - y_i of vertix v_i can depend on:
    - features x_i
    - labels y_j of all users v_j connected with v_i
    - attributes x_j of all users v_j connected with v_i

Non-iid, where from?

- Relations
  - Links between data points
    - Webpage A links to Webpage B
    - Movie A and Movie B are often rented together
- Relations as data
  - “Linked webpages are likely to present similar content”
  - “Movies that are rented together often have correlated personal ratings”

Taxonomy of techniques

Classification based on:



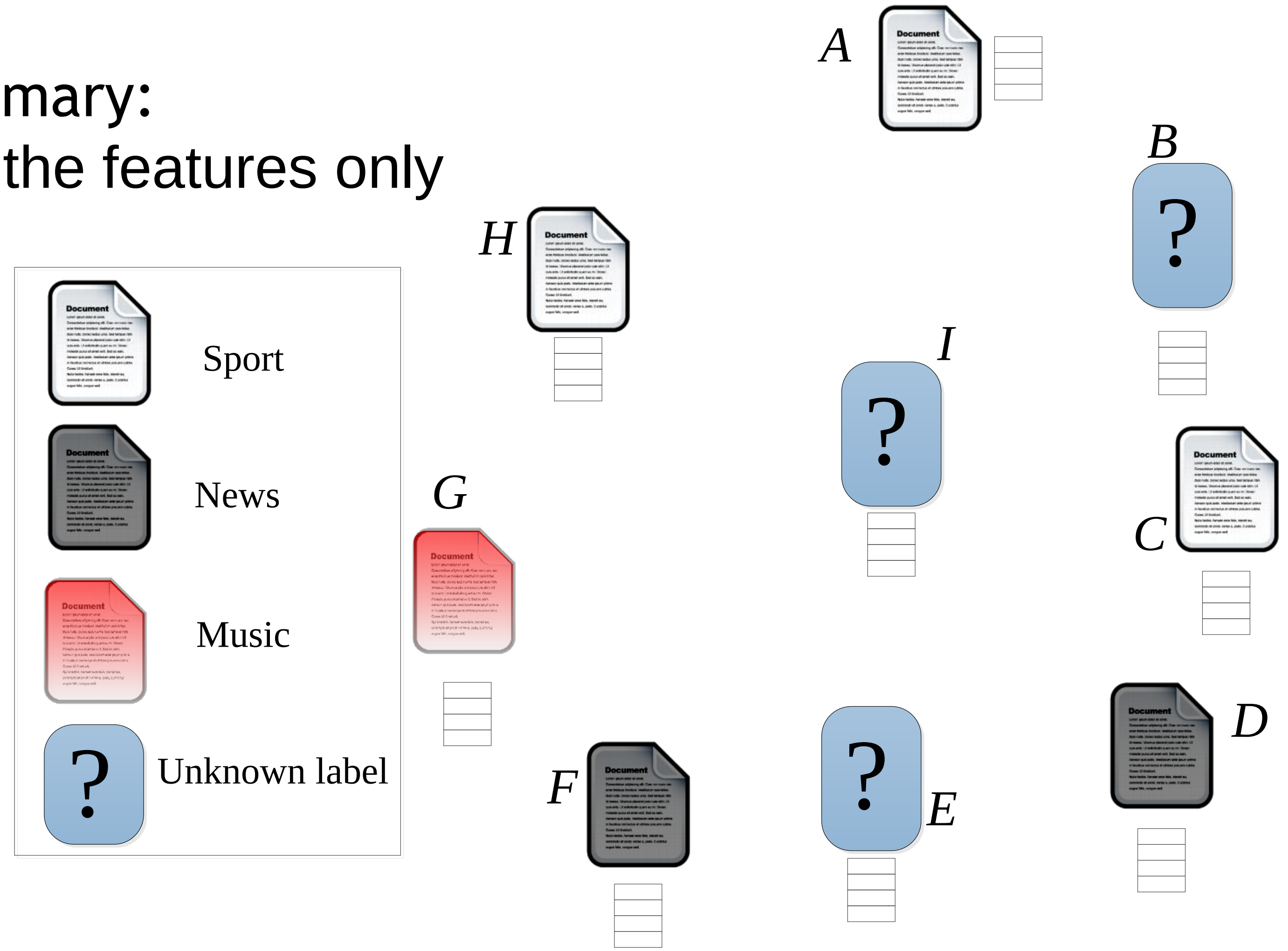
Attributes of the neighbors, labels of the neighbors

Labels of the neighbors

Each object’s own attributes

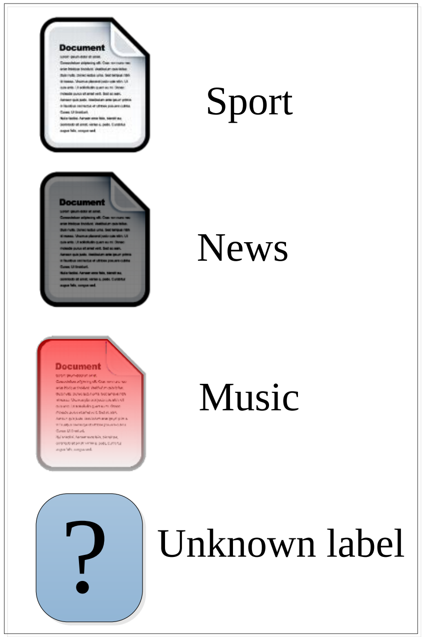
Content-based Classification

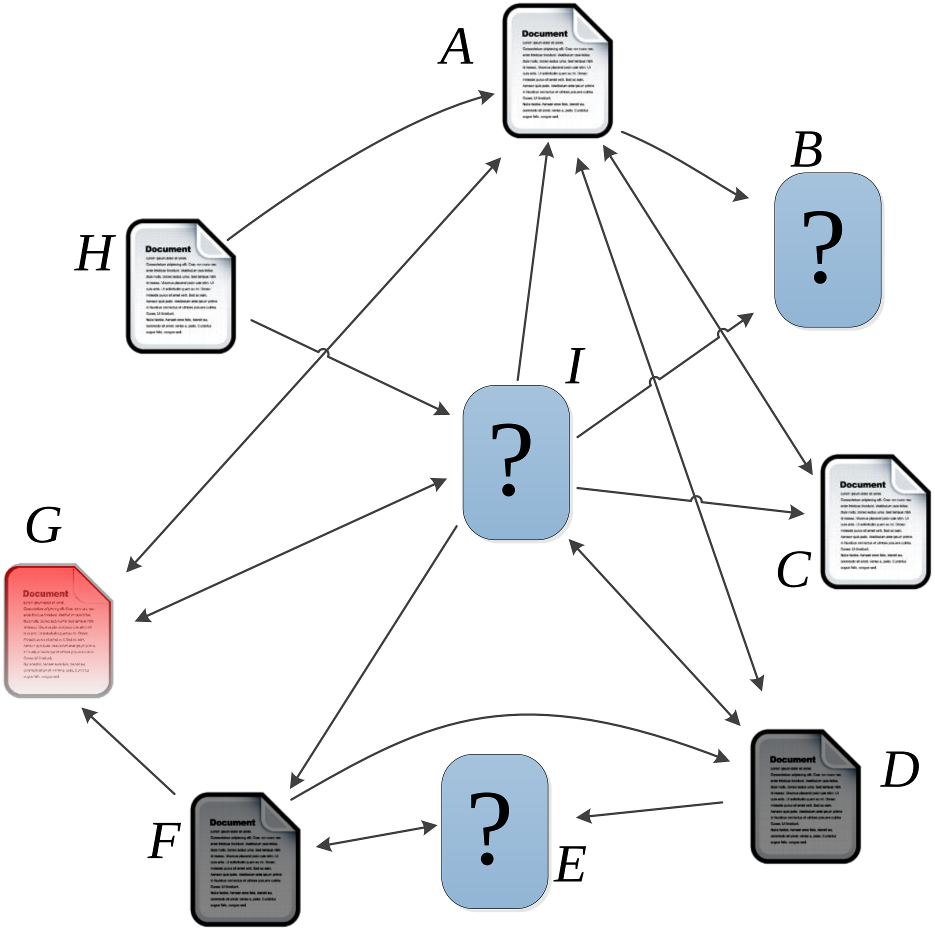
Summary:
use the features only



Relational Learning

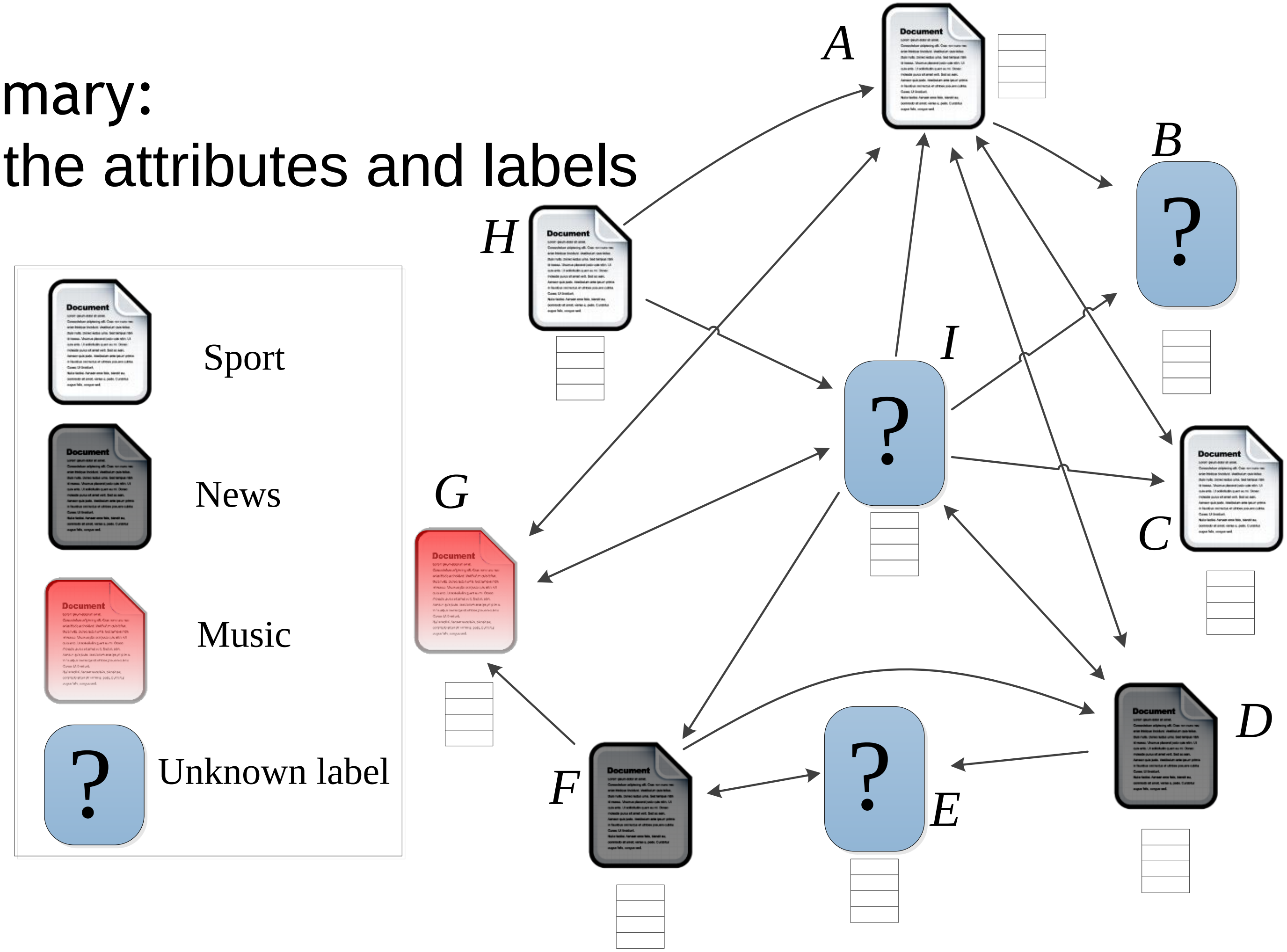
Summary:
use the labels only





Collective Classification

Summary:
use the attributes and labels

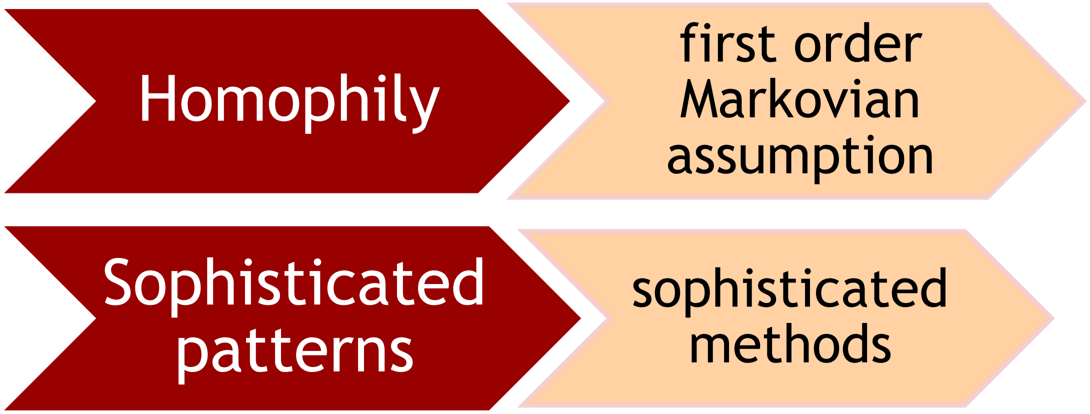


Applications

- Object labeling in images
- Part-of-speech tagging [Lafferty, 2001],
- Classification of maritime objects from video that are implicitly related spatially and/or temporally [Gupta, 2009]
- Trust Evaluation in Social Networks [Wang, 2011]
- Malicious software detection [Santos, 2011]
- Spam filtering [Laorden, 2012]
- Spam host discovery [Indyk, 2012]
- many more

Collective Classification: underlying phenomena

- **homophily** - the tendency of humans to connect with people with the same attitudes and beliefs [Wang11]
  - concentrated linkage [Jensen02] - clusters of objects linked to many common neighbours
  - autocorelation [Neville03] - values of given attribute are highly uniform among objects that share a common neighbour
- more sophisticated patterns

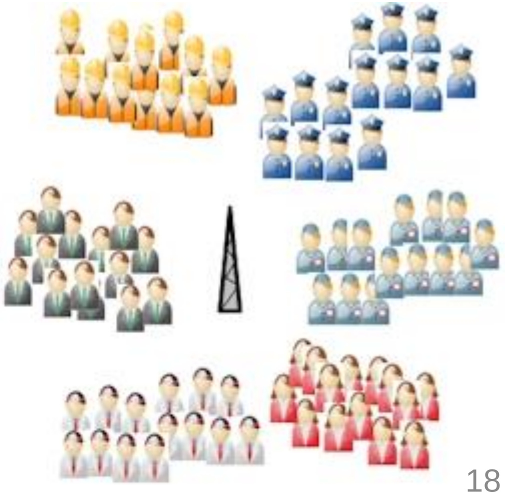


Homophily

first order Markovian assumption

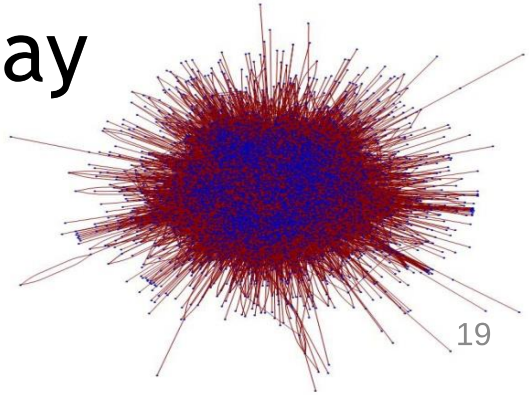
Sophisticated patterns

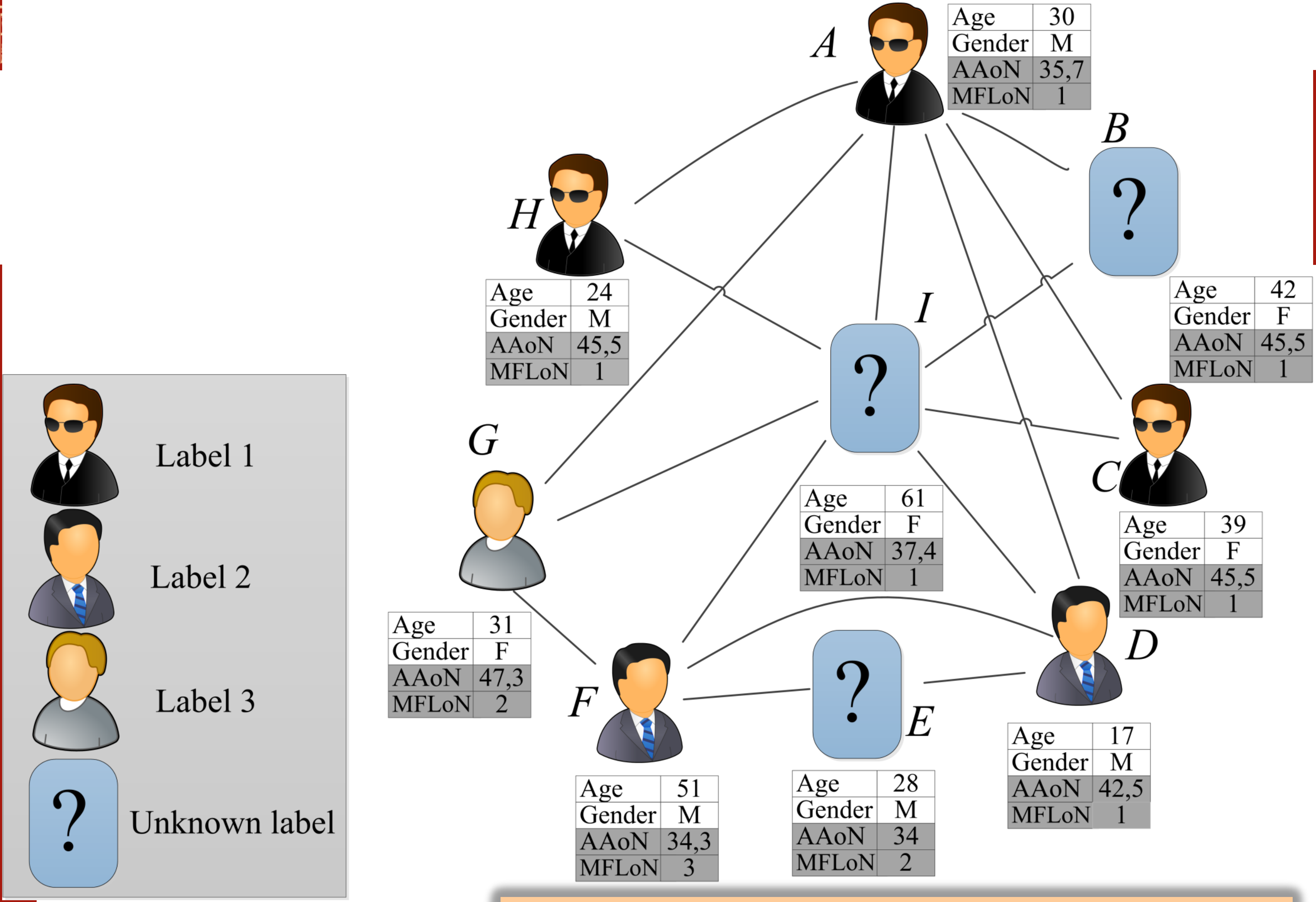
sophisticated methods



Collective Classification: underlying phenomena

- Basic realization of collective classication: first order Markovian assumption
  - v_i's class y_i depends on v_i's own attributes and the classes (labels) of v_i's immediate neighbours
- More sophisticated methods may utilize the whole graph





A

Age	30
Gender	M
AAoN	35,7
MFLoN	1

B

Age	42
Gender	F
AAoN	45,5
MFLoN	1

H

Age	24
Gender	M
AAoN	45,5
MFLoN	1

I

Age	61
Gender	F
AAoN	37,4
MFLoN	1

C

Age	39
Gender	F
AAoN	45,5
MFLoN	1

G

Age	31
Gender	F
AAoN	47,3
MFLoN	2

F

Age	51
Gender	M
AAoN	34,3
MFLoN	3

E

Age	28
Gender	M
AAoN	34
MFLoN	2

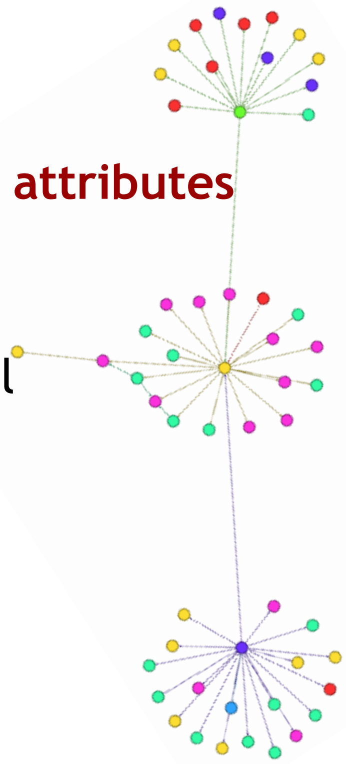
D

Age	17
Gender	M
AAoN	42,5
MFLoN	1

AAoN – average age of neighbours
MFLoN - most frequent label of neighbours

Local vs Global Models

- Methods are different in terms of **learning and inference**
- Local models
  - **vector space** - a representation of the network **structural attributes** (*centrality*, *prestige*, *betweenness*)
  - local models find the mapping *profiles -> class labels*, e.g. nodes with high degree tend to have a particular label more frequent than the others
- Global models
  - operate directly on the **whole graph** of related nodes
  - optimization of one global objective function
- Exact inference is **NP hard** for arbitrary networks



Collective Classification

- Models

Local Models

- operate on a **vector space representation of attributes** obtained by transforming a graph
- collection of local conditional classifiers successively applied to the unknown vertices

Global Models

- operate directly on a **whole graph** of related cases rather than attribute vectors
- defined as optimization of **one global objective function**

- Problems

- ⚠ which **features** should be used to maximize the classification accuracy (precise solution strongly depends on the application domain)
- ⚠ which **ordering** strategy should be used to determine, in which order to visit the nodes iteratively to re-label them
- ⚠ which global **objective function** should be implemented

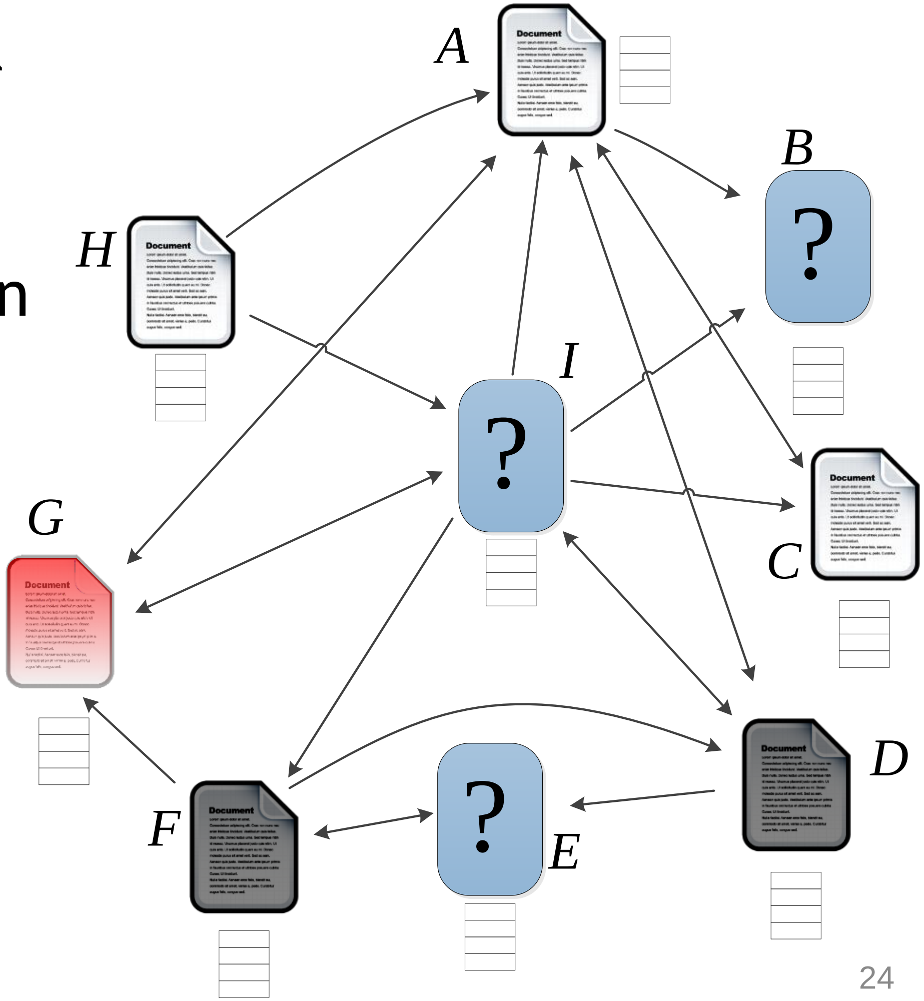
Collective Classification: Local Models

- Approximate local inference algorithms based on **local conditional classifiers**
- Example algorithms:
  - **Iterative Classification Algorithm (ICA)**
  - **Gibbs Sampling**
  - variations of above: ICAM [McDowell09] ICAMC [McDowell10] or Gradual Commit [Neville00]

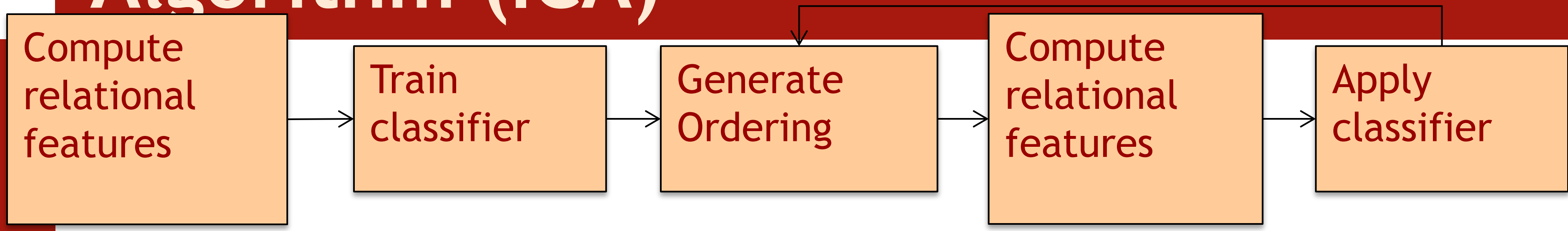
Iterative Classification

- Convert each object into a flat representation (aggregation)
- Train classifier using known labels
- Iterate until converges
  - Generate inference order
  - Reconstruct relational features
  - Update labels

- Convergence is not guaranteed



Iterative Classication Algorithm (ICA)



```
for each node $v_i \in V^K$ do
        compute relational features $x_i$
end for
train classier $\Phi$ using attributes of $V^K$ nodes
repeat
generate ordering $O$ over nodes in $V^{UK}$
for each following node $v_i$ from $O$ do
        compute relational features $x_i$ using current label assignments
        use classier $\Phi$ to infer label $l_i$ for node $v_i$
end for
until label stabilization
```

Gibbs Sampling

Bootstrapping

```
for each node $v_i \in V$ do
    compute relational features $x_i$
end for
train classifier $\Phi$ using attributes of $V^K$ nodes
for each node $v_i \in V^{UK}$ do
    use classifier $\Phi$ to infer label $l_i$ for node $v_i$
end for
```

Change vs. ICA

Burn-in

```
for $n$ = 1 to $s$ do
    generate ordering $O$ over nodes in $V^{UK}$
    for each node $v_i \in V^{UK}$ do
        recompute relational features $x_i$
        use classifier $\Phi$ to infer label $l_i$ for node $v_i$
    end for
end for
```

Gibbs Sampling

Initialize sample counts

```
for each node $v_i \in V^{UK}$ do
    for label $l \in L$ do
        $c[i; l] = 0$
    end for
end for
```

Collect samples

```
repeat
generate ordering $O$ over nodes in $V^{UK}$
for each following node $v_i$ from $O$ do
    recompute relational features $x_i$
    use classier $\Phi$ to infer label $l_i$ for node $v_i$
    $c[i; l] = c[i; l] + 1$
end for
```

Compute final labels

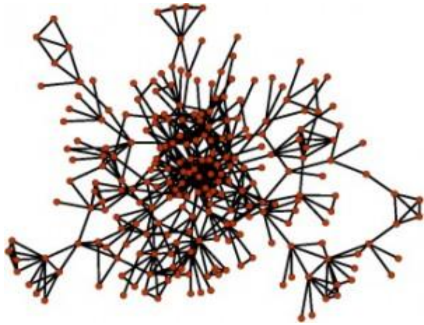
```
until stop condition
for each node $v_i \in V$ do
    $l_i \leftarrow \text{argmax}_{l \in L} c[i; l]$
end for
```

ICA and GS Challenges

- **Feature construction** for local classifier Φ
  - Φ often needs fixed-length vector
  - choice of aggregation (avg, mode, count, etc.)
  - choice of relations (in-, out-links, both)
  - choice of neighbour attributes (all?, top-k confident?)
- **Local classifier Φ**
  - requires training
  - choice of the classifier (NB, kNN, SVM, …)
- **Node ordering** for updates: random, diversity based
- **Convergence**
- **Run time** (many iterations for GS)

Collective Classication: Global Models

- Operates directly on the **whole graph**



- Optimization of **one** global **objective function**
- Probabilistic graphical models are convenient
- Markov Random Field – intractable to solve
- Algorithms:
  - **Loopy Belief Propagation** (LBP)[1,2]
  - Tree-reweighted Belief Propagation (TRBP)[3]

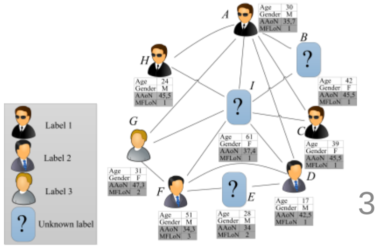
[1] Pearl J.: *Probabilistic reasoning in intelligent systems*. Morgan Kaufmann, 1988

[2] Yedidia J.S., Freeman W.T., Weiss Y.: *Generalized Belief Propagation, Neural Information Processing Systems (NIPS)*, 2000,vol. 13,689-695

[3] Wainwright M., Jaakkola T., Willsky A.: *A new class of upper bounds on the log partition function*. IEEE Trans. Info. Theory, 1(7), 2005, 2313-2335

Loopy Belief Propagation

- Iterative message-passing algorithm
- Messages transferred between **all** connected nodes v_i and v_j: $(v_i, v_j) \in E$
- Belief of what v_j's label should be based on v_i's label
- Originated from pairwise Markow Random Fields



Loopy Belief Propagation

- iterative message-passing algorithm (interpreted as belief of what label should be assigned based on neighboring label)
- global objective function: idea of pairwise Markov Random Field
  - message: $m_{i \to j}(l_j) = \alpha \sum_{l_i \in L} \Psi_{ij}(l_i, l_j) \phi(l_i) \prod_{v_k \in V^{UK} \setminus v_j} m_{k \to i}(l_i)$
  - believe: $b_i(l_i) = \alpha \phi(l_i) \prod_{v_j \in V^{UK}} m_{j \to i}(l_i)$

Loopy Belief Propagation

```
for each edge $(v_i, v_j) \in E$; $vi, vj \in V^{UK}$ do
    for each class label $l_j \in L$ do
        $m_{i \to j}(l_j) \leftarrow 1$
    end for
end for
  //perform message passing
repeat
   for each edge $(v_i, v_j) \in E$; $v_i, v_j \in V^{UK}$ do
        for each class label $l_j \in L$ do
            $m_{i \to j}(l_j) \leftarrow \alpha \Sigma_{lj \in L} \Psi_{ij}(l_i, l_j)\ \Phi(l_i)\ \Pi_{vk \in V^{UK} \backslash vj}\ m_{k \to i}(l_i)$
        end for
   end for
until stop condition
```

Loopy Belief Propagation, cont.

```
    //compute beliefs
for all $v_i \in V_{UK}$ do
    for all $l_j \in L$ do
        $b_i(l_j) \leftarrow \alpha \Phi(l_j) \prod_{v_j \in V^{UK}} m_{j \to i}(l_j)$
    end for
end for
```

Loopy Belief Propagation

- Advantages:
  - **Easy** to program & parallelize
  - Can be applied to any graphical model
- Challenges:
  - Convergence **not** guaranteed, especially if many closed loops

Progress to Date

- Probabilistic logic [Nilsson, 1986]
- Statistics and beliefs [Halpern, 1990]
- Knowledge-based model construction [Wellman et al., 1992]
- Stochastic logic programs [Muggleton, 1996]
- Probabilistic relational models [Friedman et al., 1999]
- Relational Markov networks [Taskar et al., 2002]
- Markov logic [Domingos & Lowd, 2009]
- Etc.

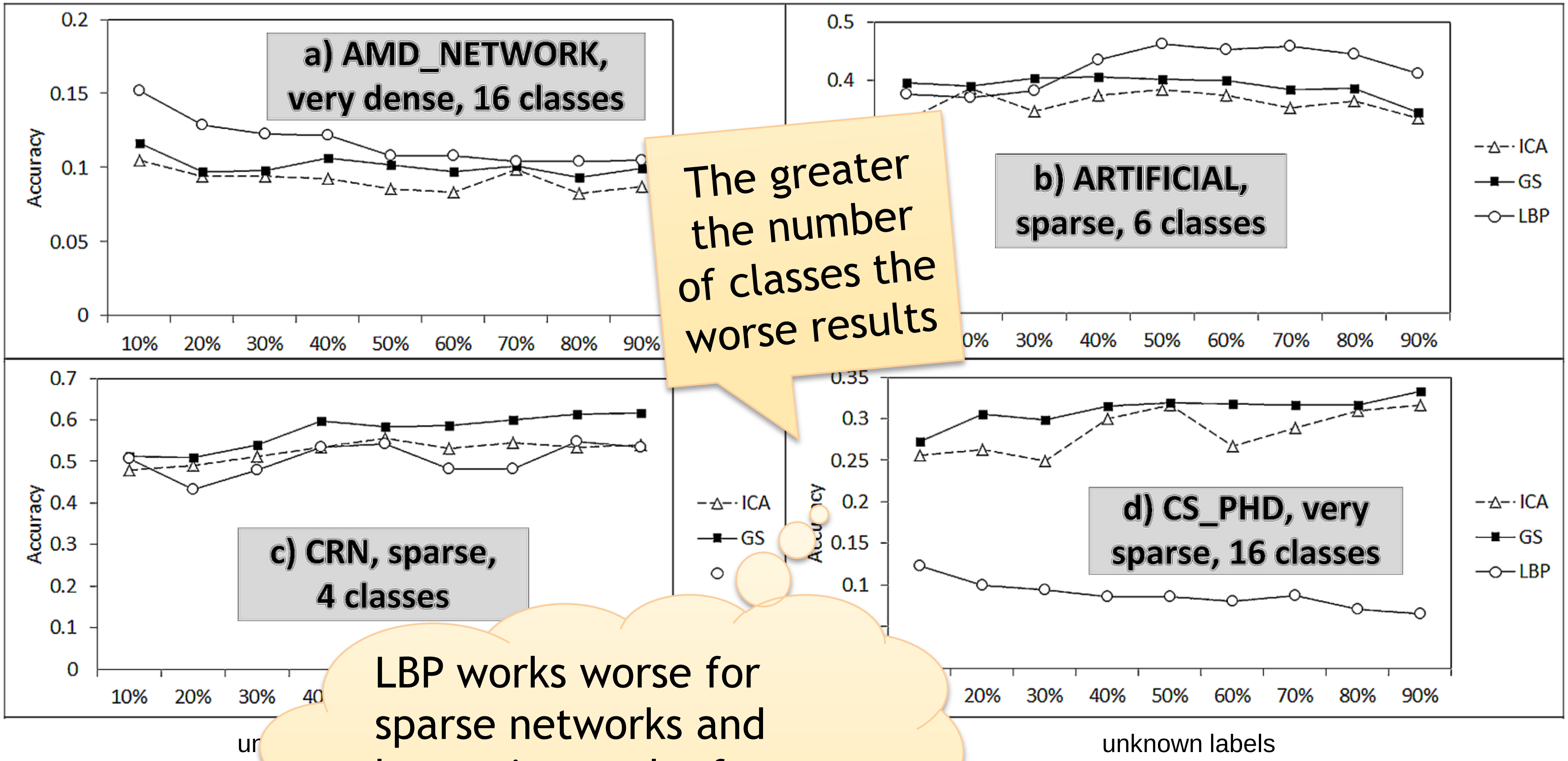
Experiments

- Compare predictive accuracy of:
  - Iterative Classification Algorithm (ICA)
  - Gibbs Sampling Algorithm (GSA)
  - Loopy Belief Propagation (LBP)
- Settings:
  - Local classifiers - C4.5
  - dataset splits between nodes known and unknown nodes - distinct proportions (from 10% to 90% unkown labels using uniform distribution)
  - 8 datasets

Datasets

Dataset	Vertices	Edges	Classes	Avg. Deg.	Type
AMD_NETWORK	332	69092	16	208,108	Attendance on conference
ARTIFICIAL	413	415	6	1,004	artificial
CRN	327	324	4	0,990	artificial
CS PHD	1451	924	16	0,636	PhD students -advisers
NET SCIENCE	1588	2742	26	1,726	co-authorship network
PAIRS FSG	4931	61449	3	12,461	word associationin dictionary
PAIRS FSG SMALL	1972	12213	3	6,193	word associationin dictionary
YEAST	2361	2353	13	0,996	protein - protein interaction network

Results

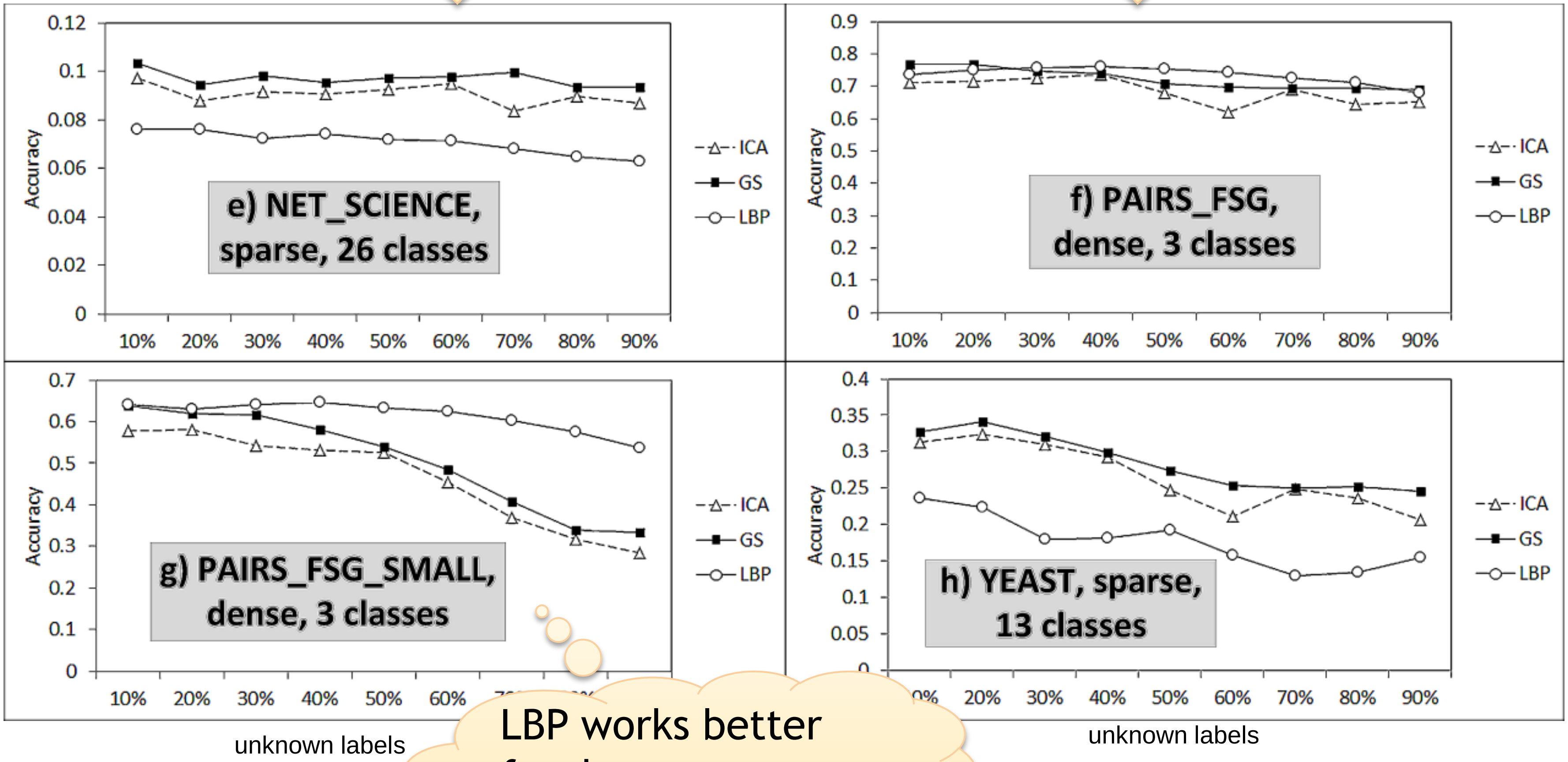


The greater the number of classes the worse results

LBP works worse for sparse networks and boosts its results for dense networks

Results

GS works a bit better than ICA



e) NET_SCIENCE, sparse, 26 classes

f) PAIRS_FSG, dense, 3 classes

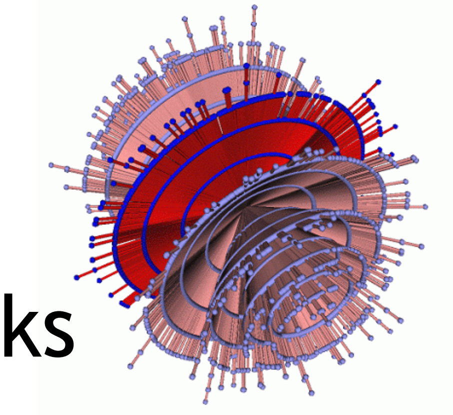
g) PAIRS_FSG_SMALL, dense, 3 classes

h) YEAST, sparse, 13 classes

LBP works better for dense networks

Problems in Collective Classication

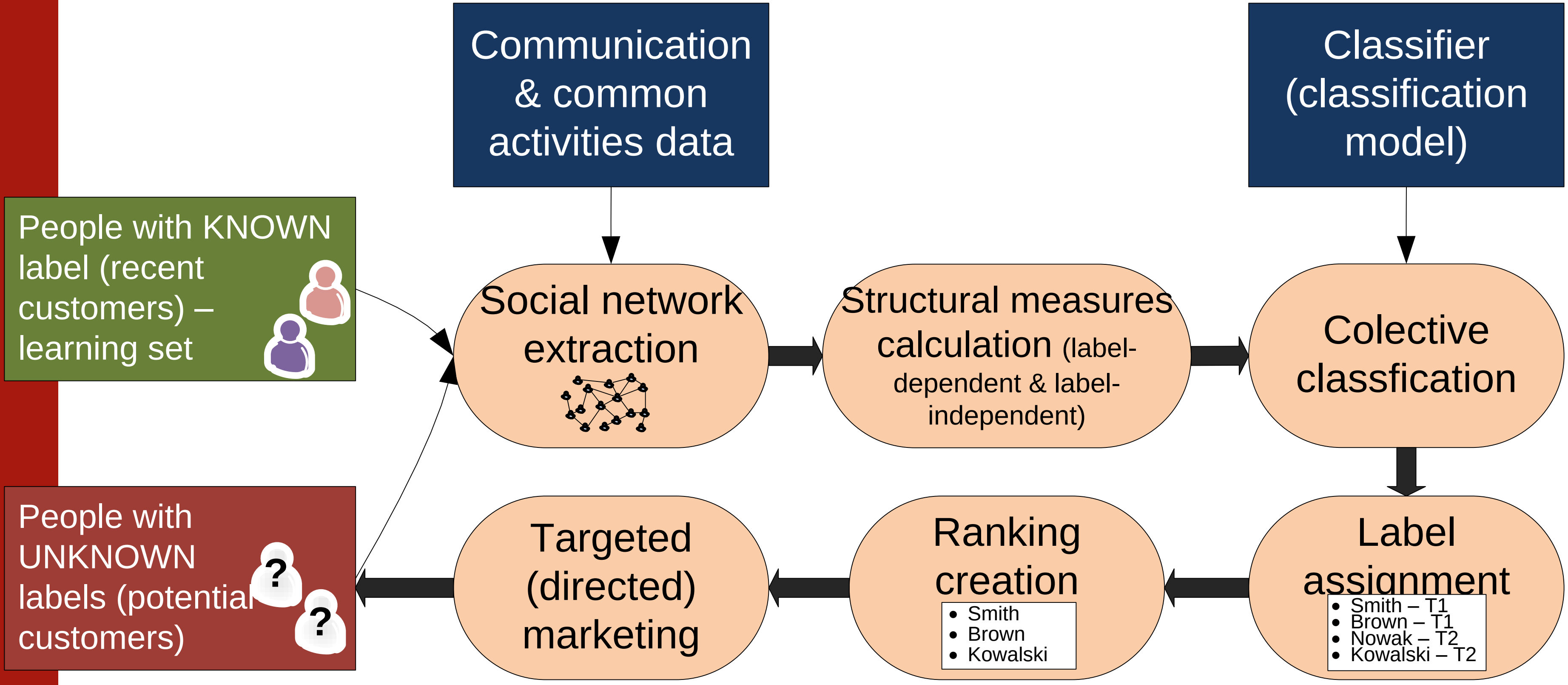
- Sparsely labelled networks
- Classication in Multiplex Networks
- Active learning and inference
- Inference for huge networks
- Classification in dynamic networks
- Generation of synthetic networked data



Problem Example: Telecom

- Telecommunication industry
- V - telecom customers
- E - relationships between customers extracted from phone calls performed by or to them
- X – regular: age, gender, relational: node degree
- Y – tarrifs, e.g. {T1=20Mbps, T2=1Gbps}
- Classication: who is most likely to buy T1 or T2

Classification Process



Communication & common activities data

Classifier (classification model)

People with KNOWN label (recent customers) – learning set

Social network extraction

Structural measures calculation (label-dependent & label-independent)

Colective classfication

People with UNKNOWN labels (potential customers)

Targeted (directed) marketing

Ranking creation

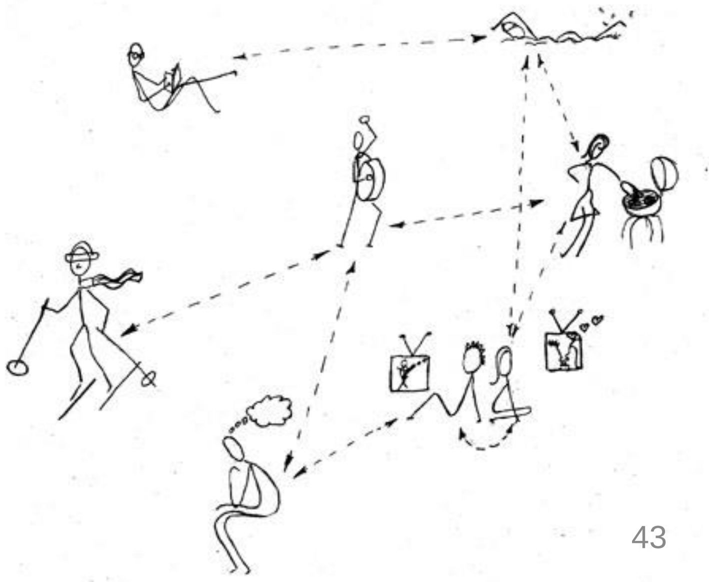
- Smith
- Brown
- Kowalski

Label assignment

- Smith – T1
- Brown – T1
- Nowak – T2
- Kowalski – T2

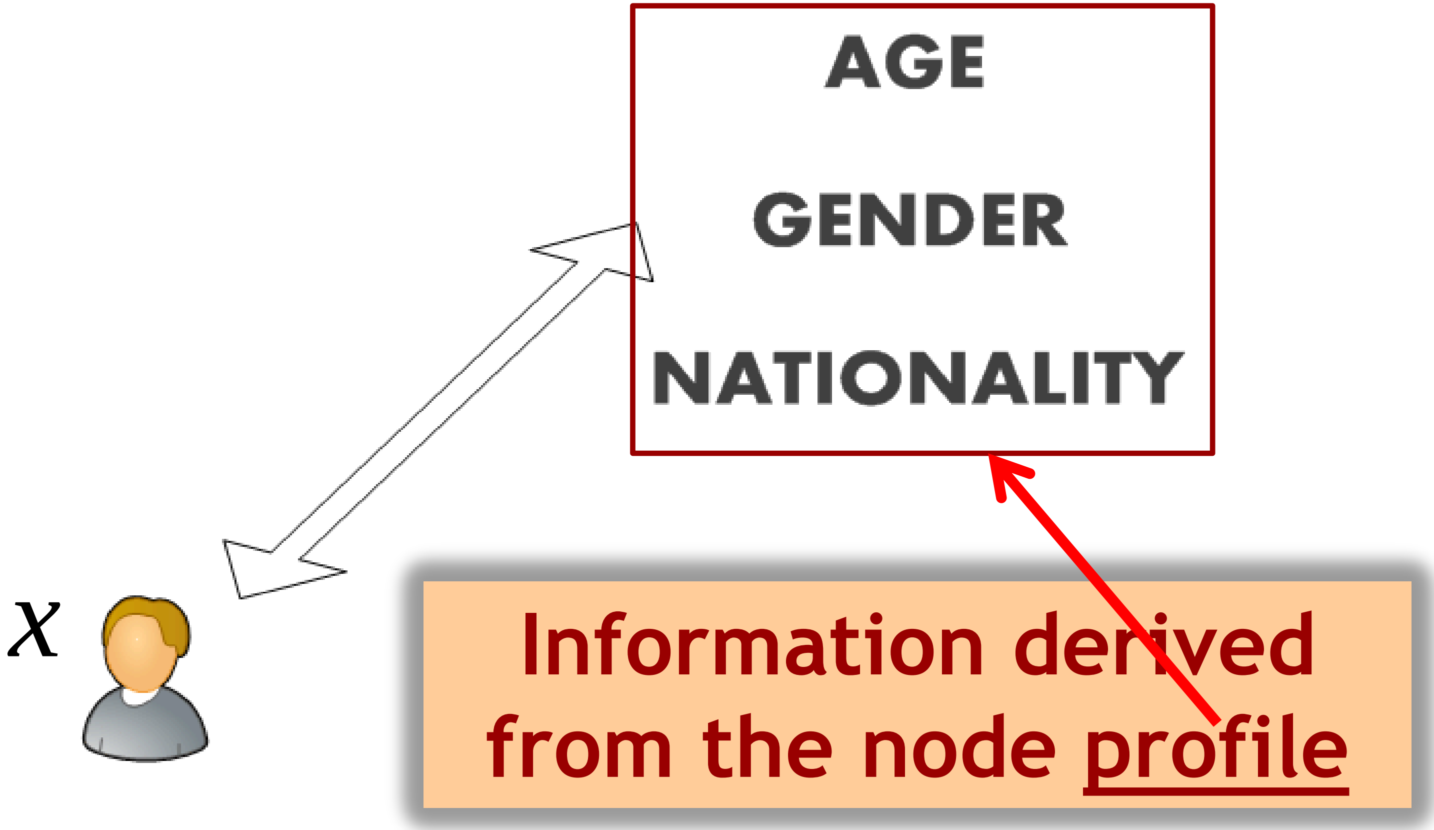
Source of Features

- Possible features (X)
  - Profiles
  - Label-independent (structural)
  - Label-dependent (structural)
  - Mixtures (hybrid)



Source of Features: Profiles

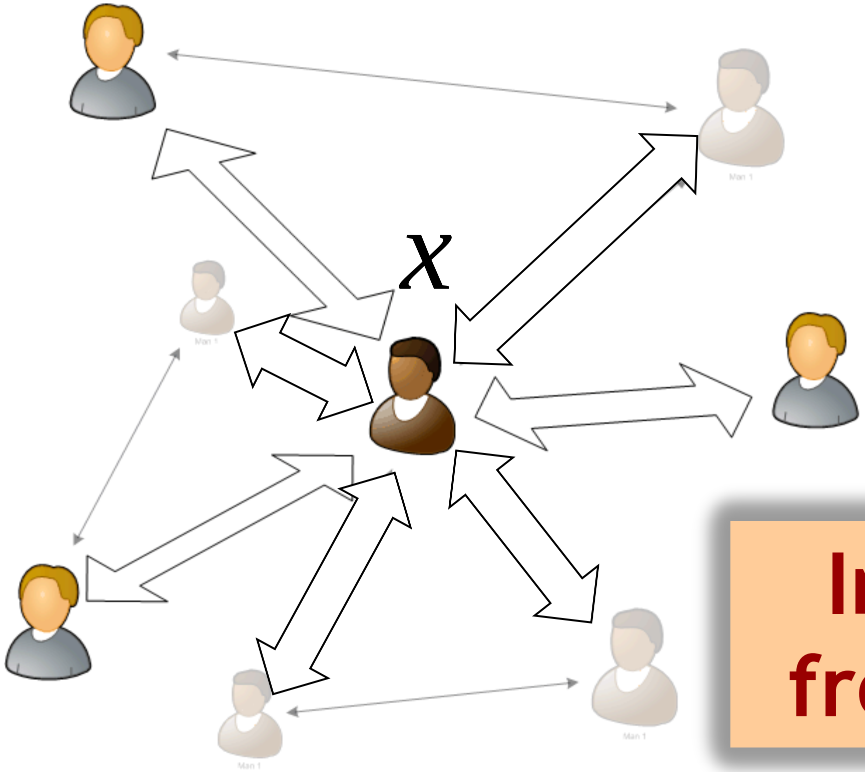
1. Correlation between *x*'s label (class) and *x*'s **attributes**



Information derived from the node profile

Source of Features: Label-independent

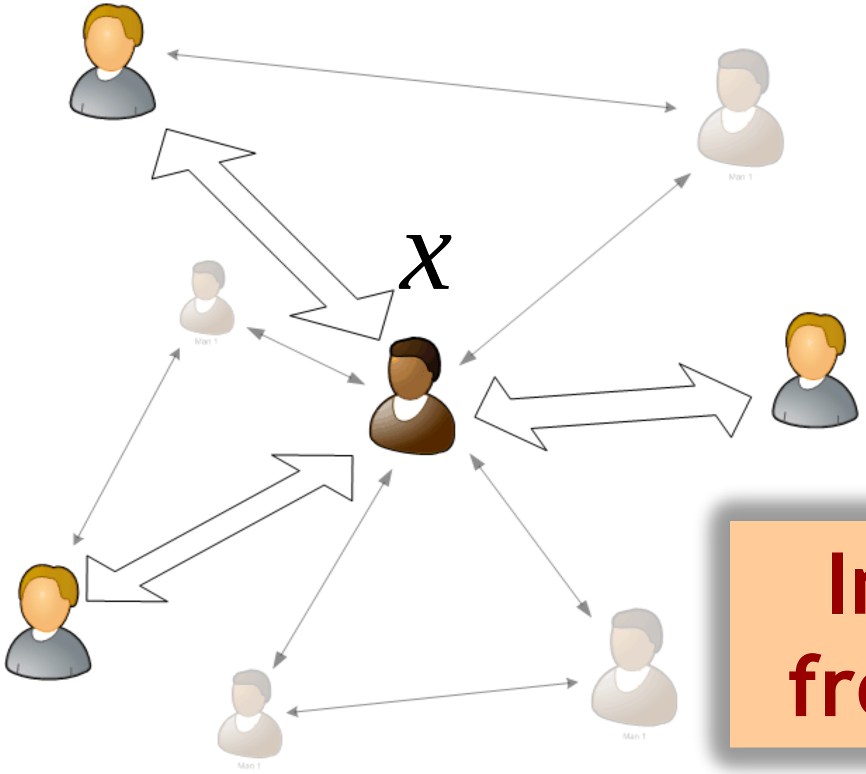
2. Correlation between *x*’s label and **all known labels** of nodes in *x*’s neighbourhood



Information derived from the SN <u>structure</u>

Source of Features: Label-dependent

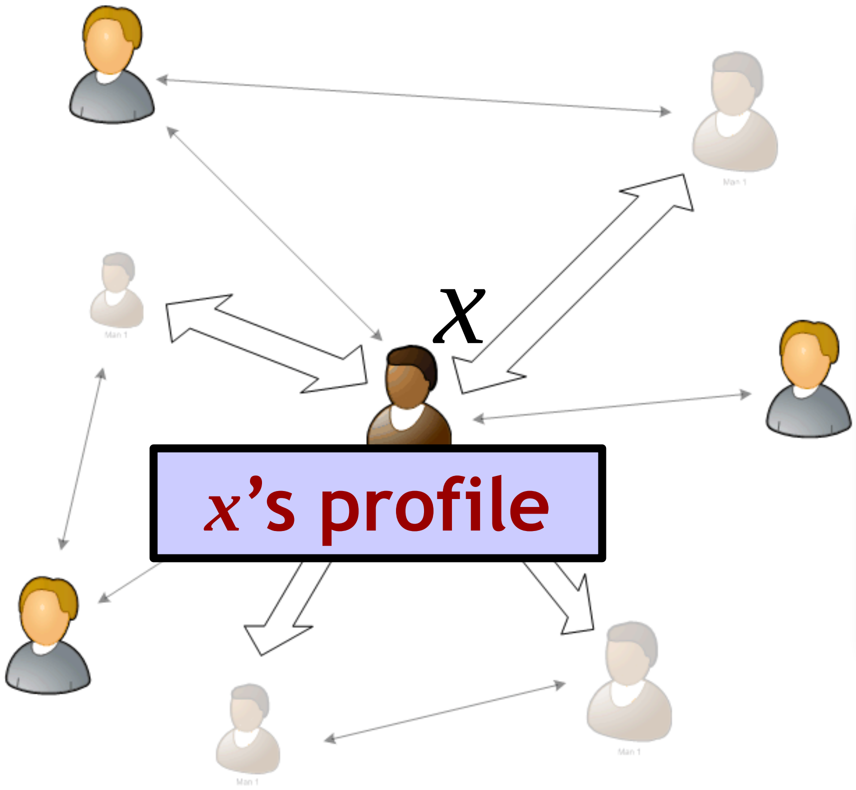
3. Correlation between x's label and **known labels** of **each type class separately** from x's neighbours



Information derived from the SN structure

Source of Features

4. Hybrid correlation between *x*'s label and **labels** of from the *x*'s neighbourhood and



Information derived from the SN structure & node profiles

Structural Measures

- Centrality
- Betweeness
- Prestige
- Reach
- Closeness
- Density
- Social Position – social contacts affect the productivity of individuals and groups
- Clustering coefficient
- etc.

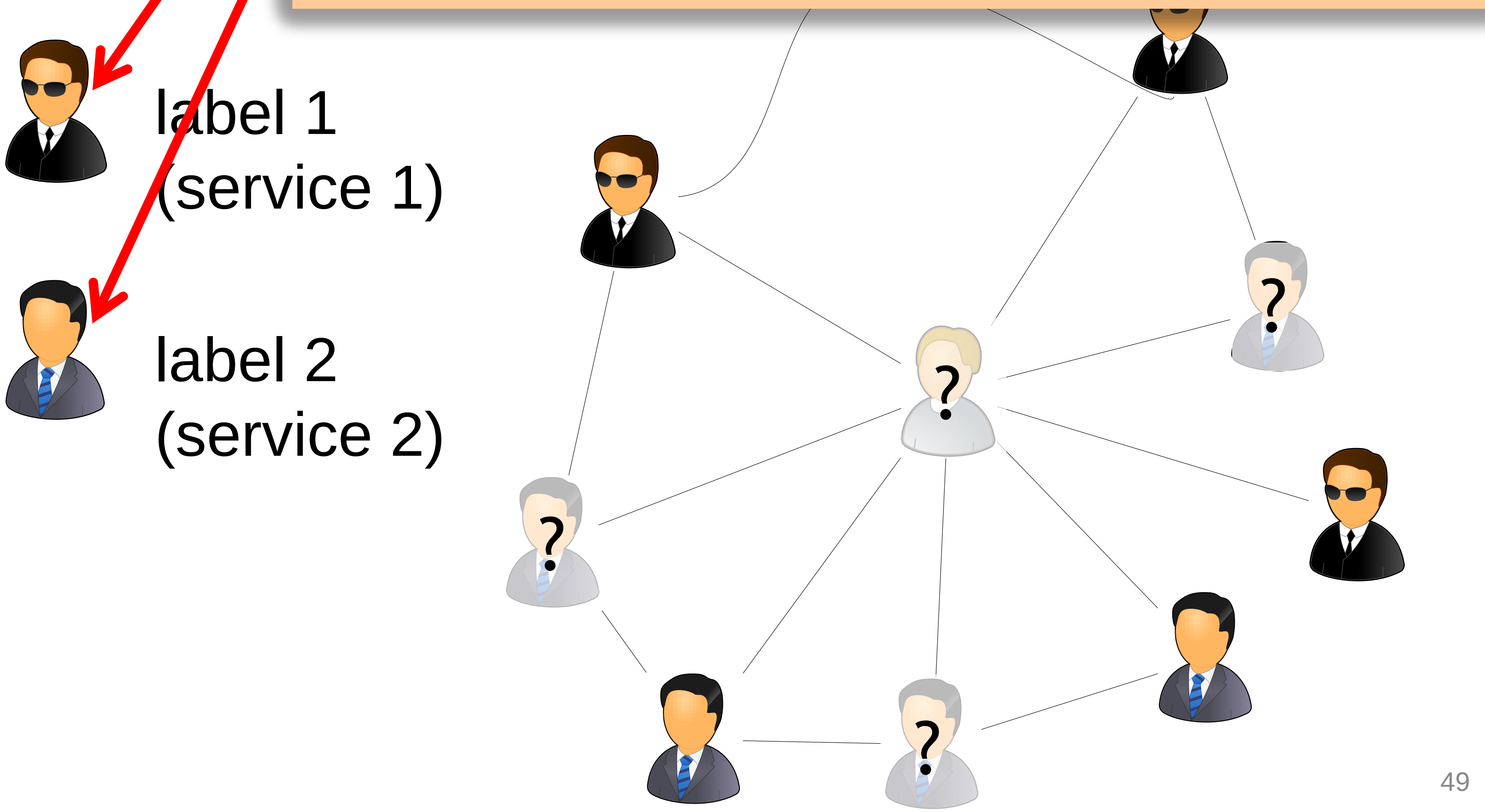
Node C
Label-o

Telecom business (ISP)

Service 1: Internet access 20 Mbps

Service 2: Internet access 1 Gbps

Service ? - for non customers



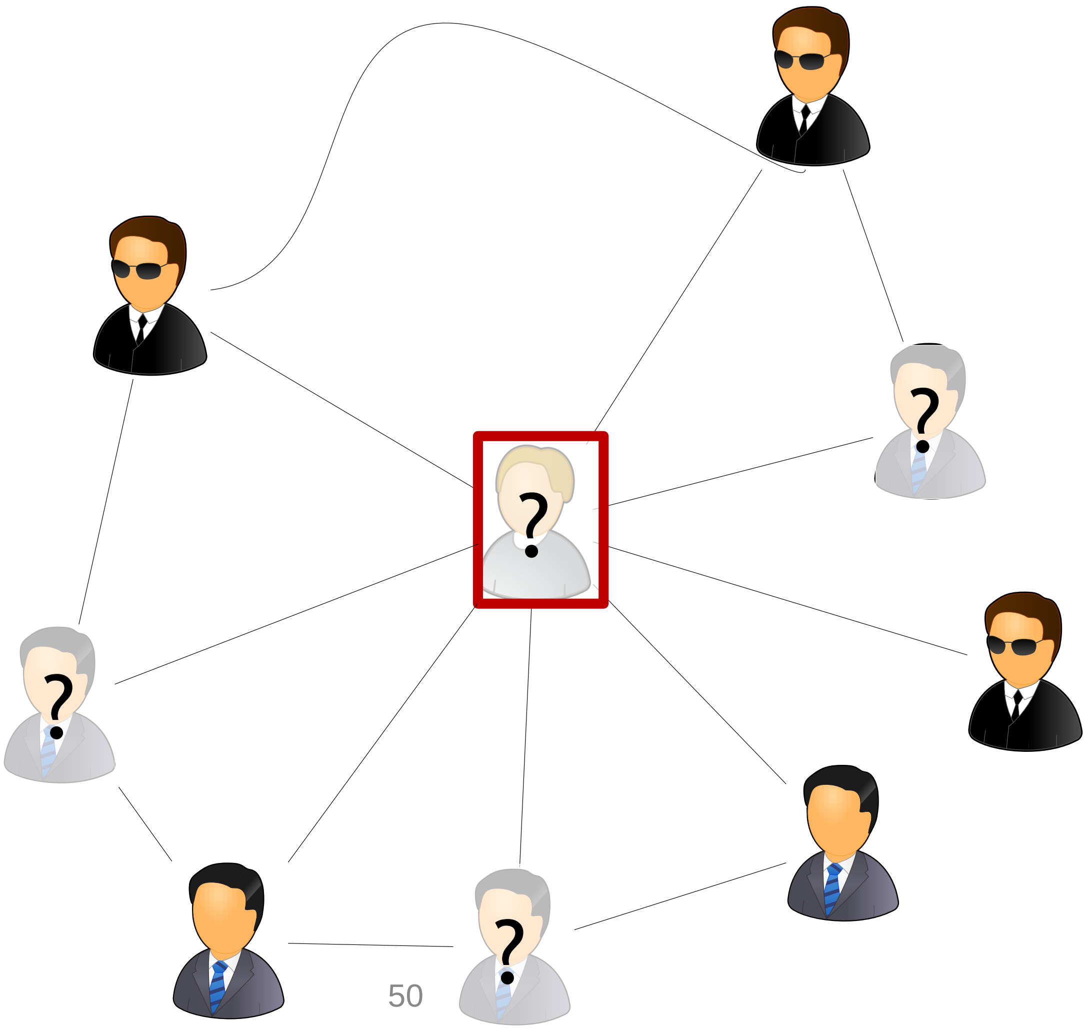
label 1
(service 1)

label 2
(service 2)

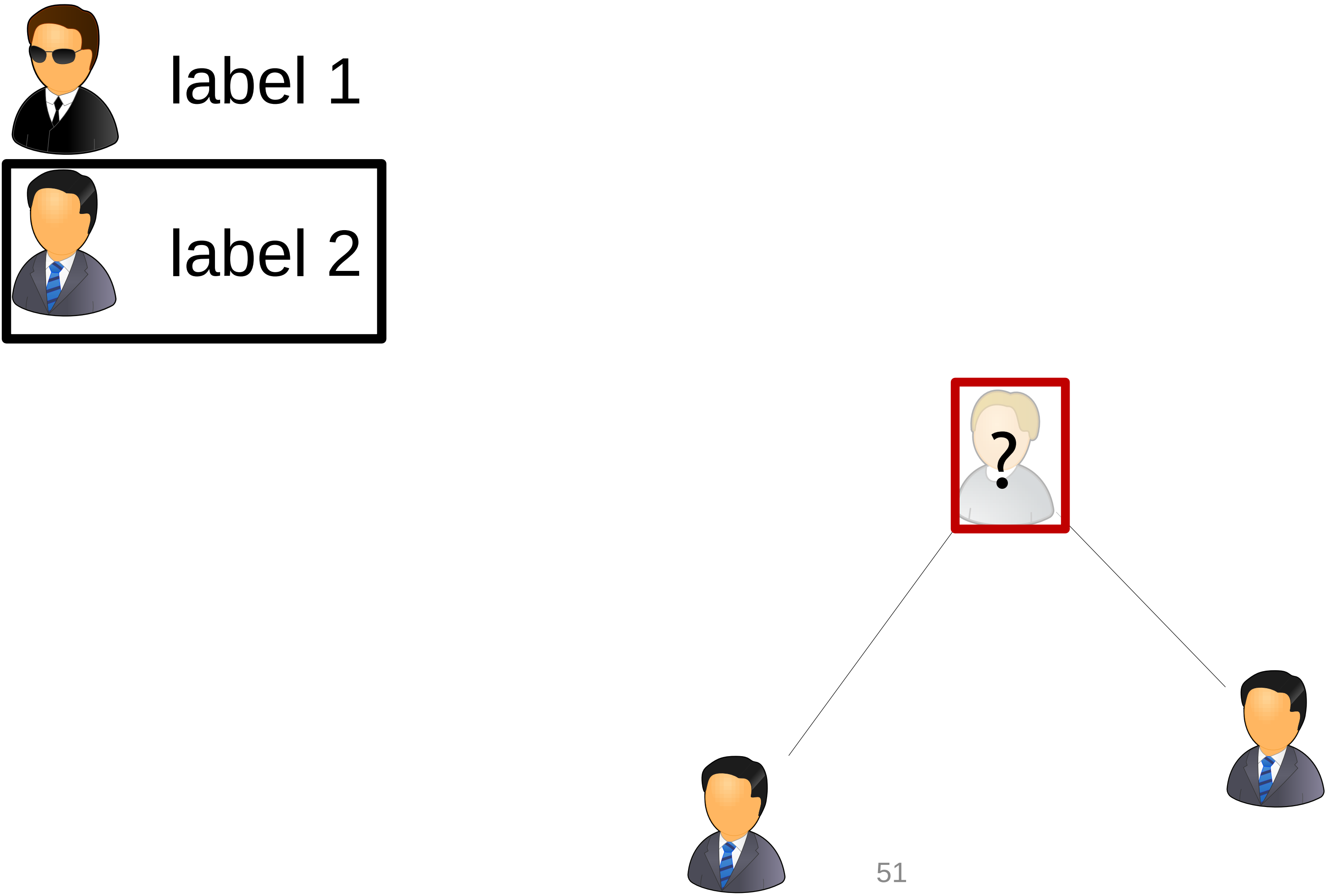
Node Classification using Label-dependent Features

label 1

label 2



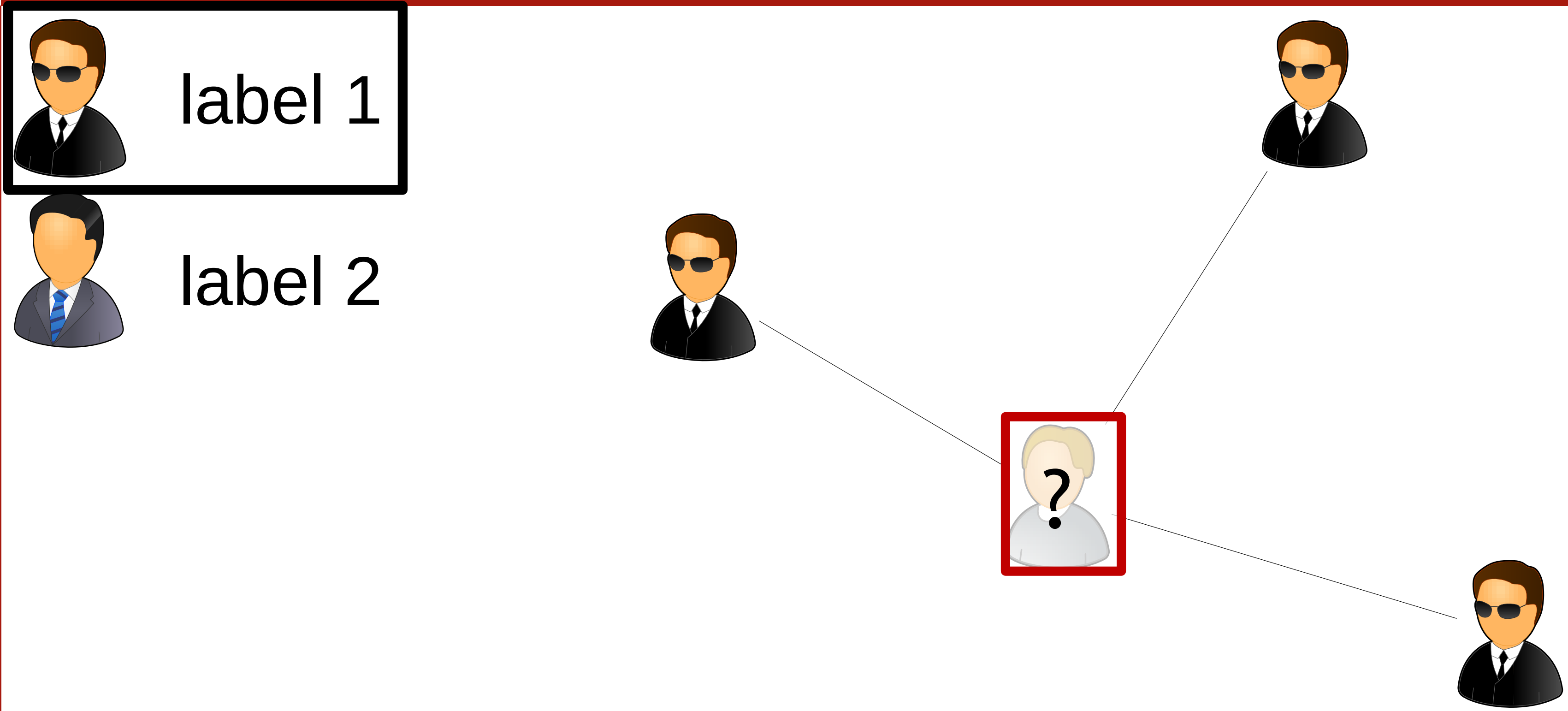
Node Classification using Label-dependent Features



label 1

label 2

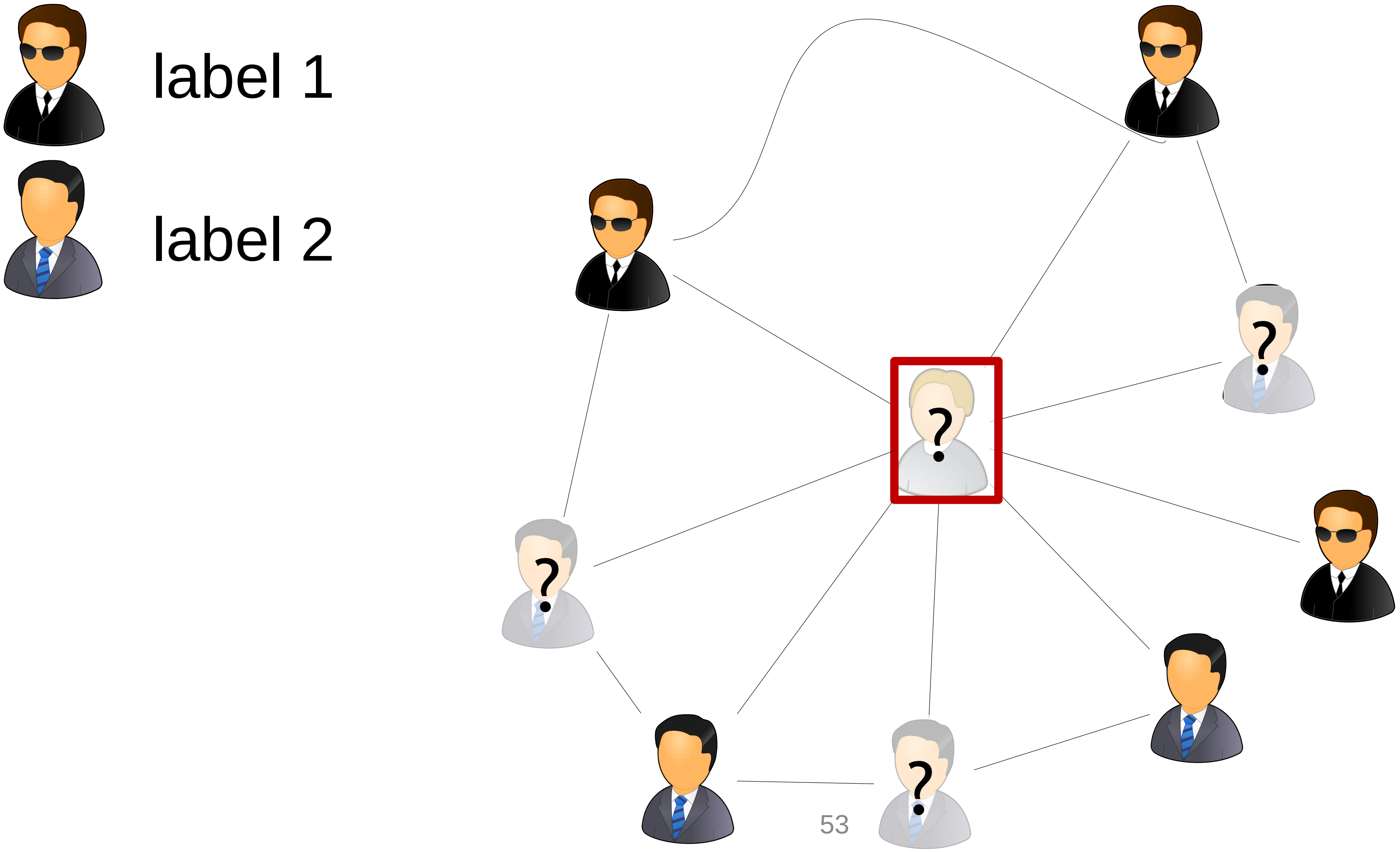
Node Classification using Label-dependent Features



label 1

label 2

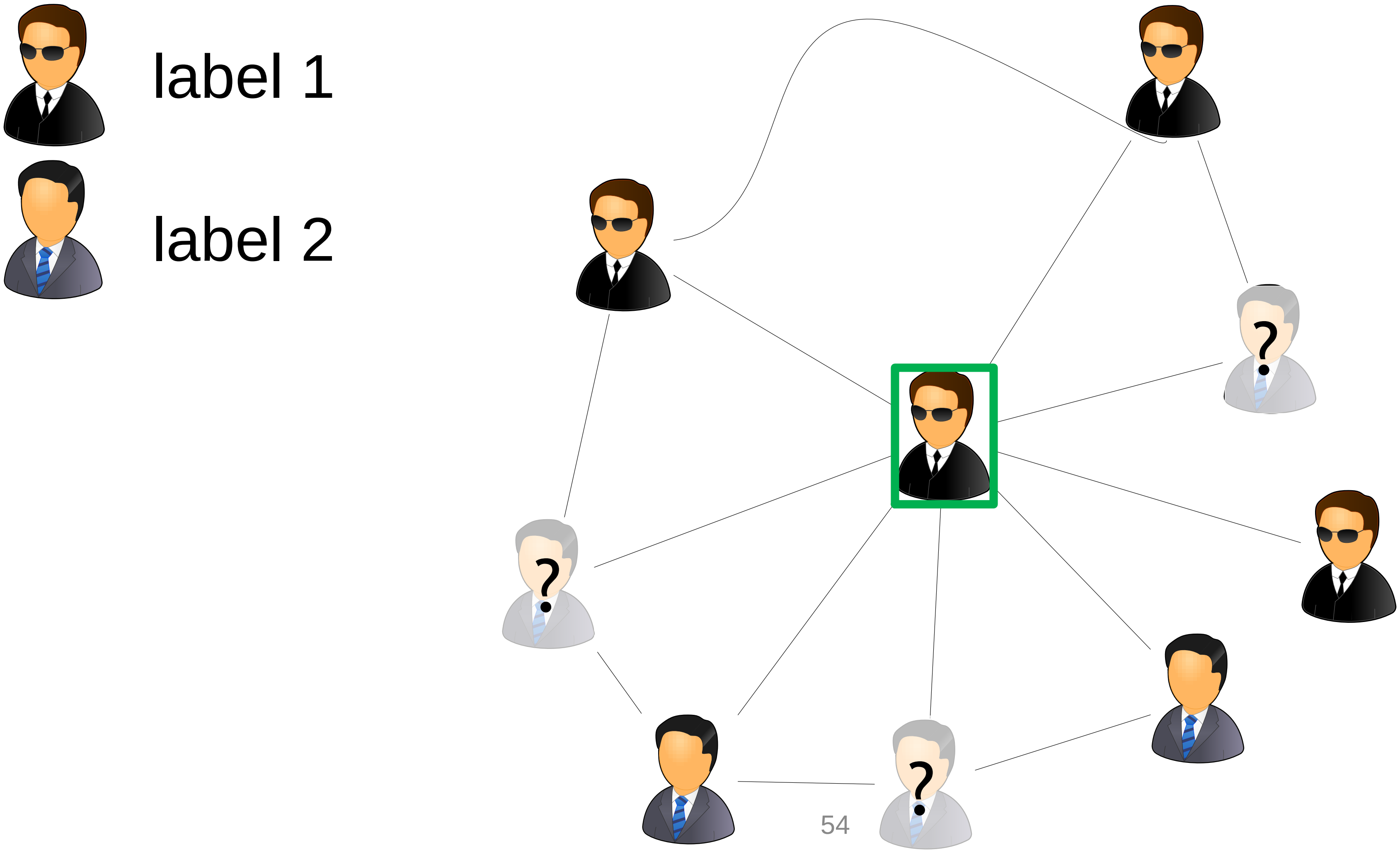
Node Classification using Label-dependent Features



label 1

label 2

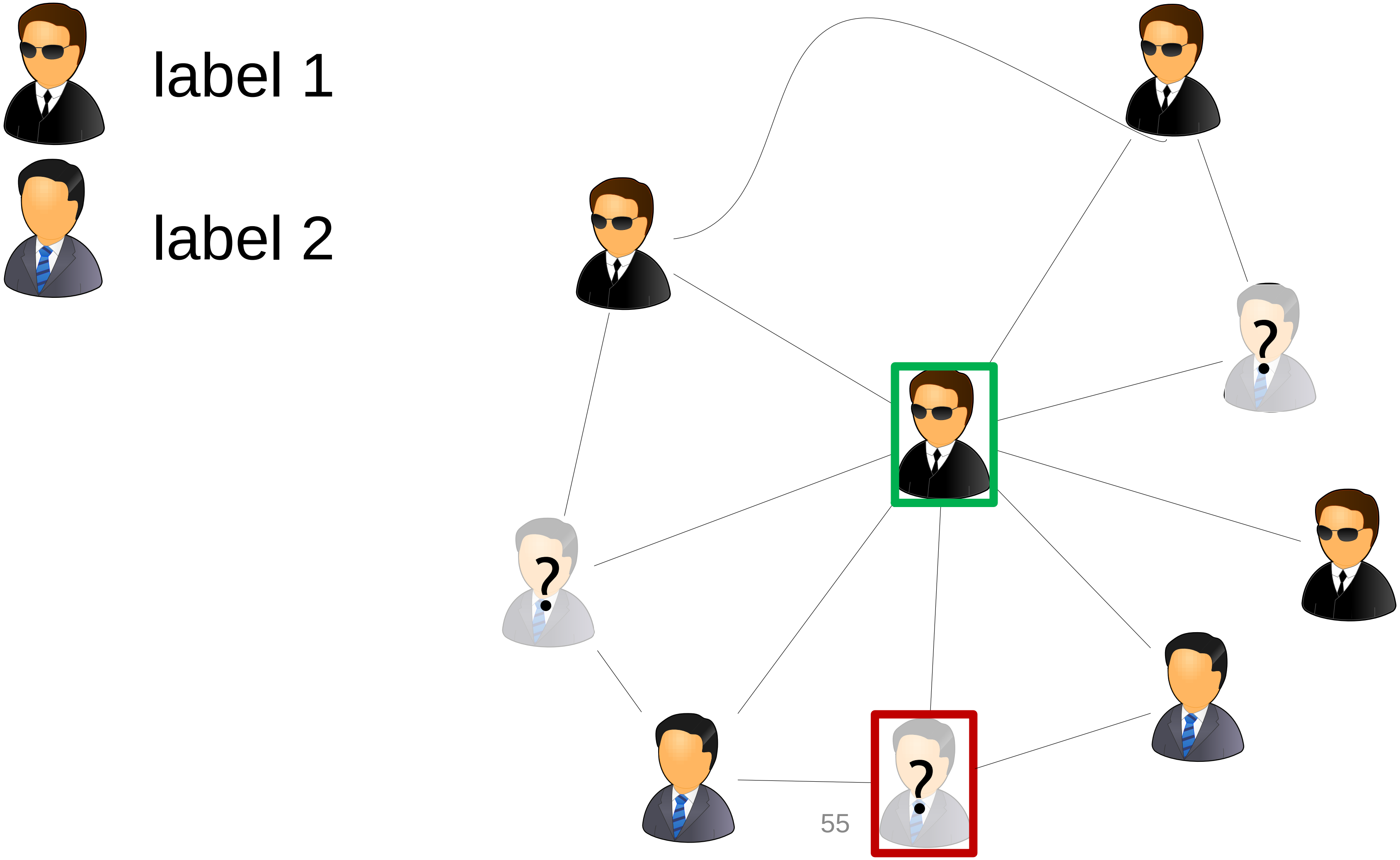
Node Classification using Label-dependent Features



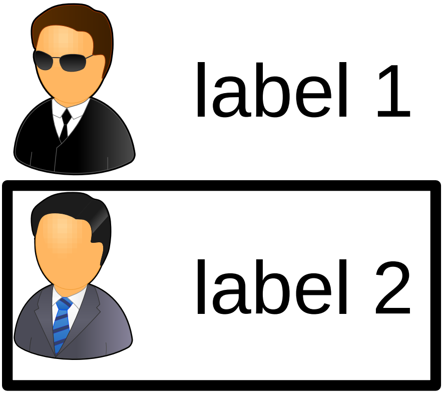
label 1

label 2

Node Classification using Label-dependent Features

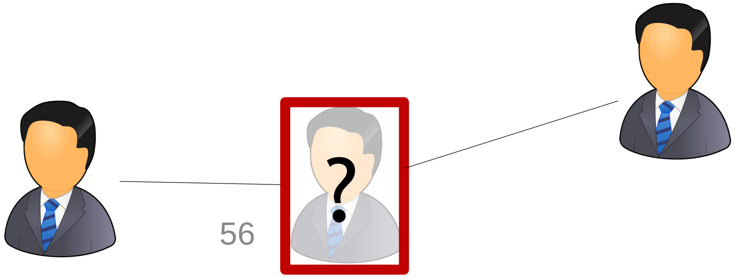


Node Classification using Label-dependent Features

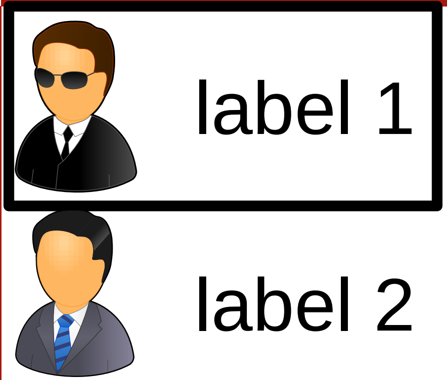


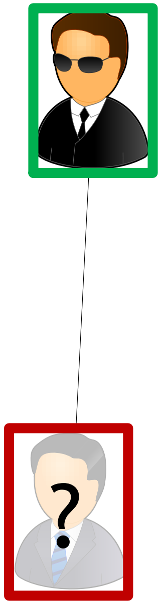
label 1

label 2

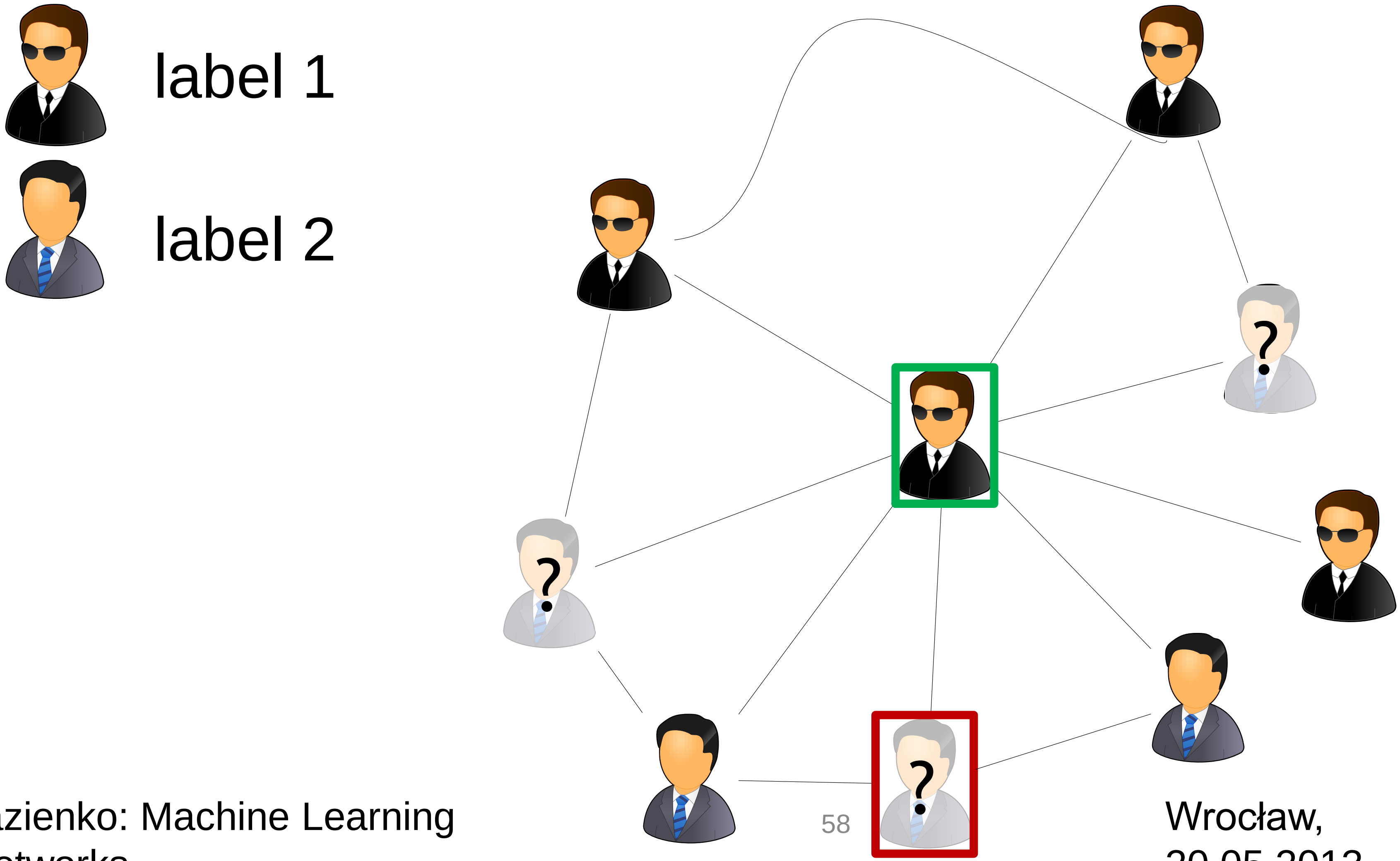


Node Classification using Label-dependent Features

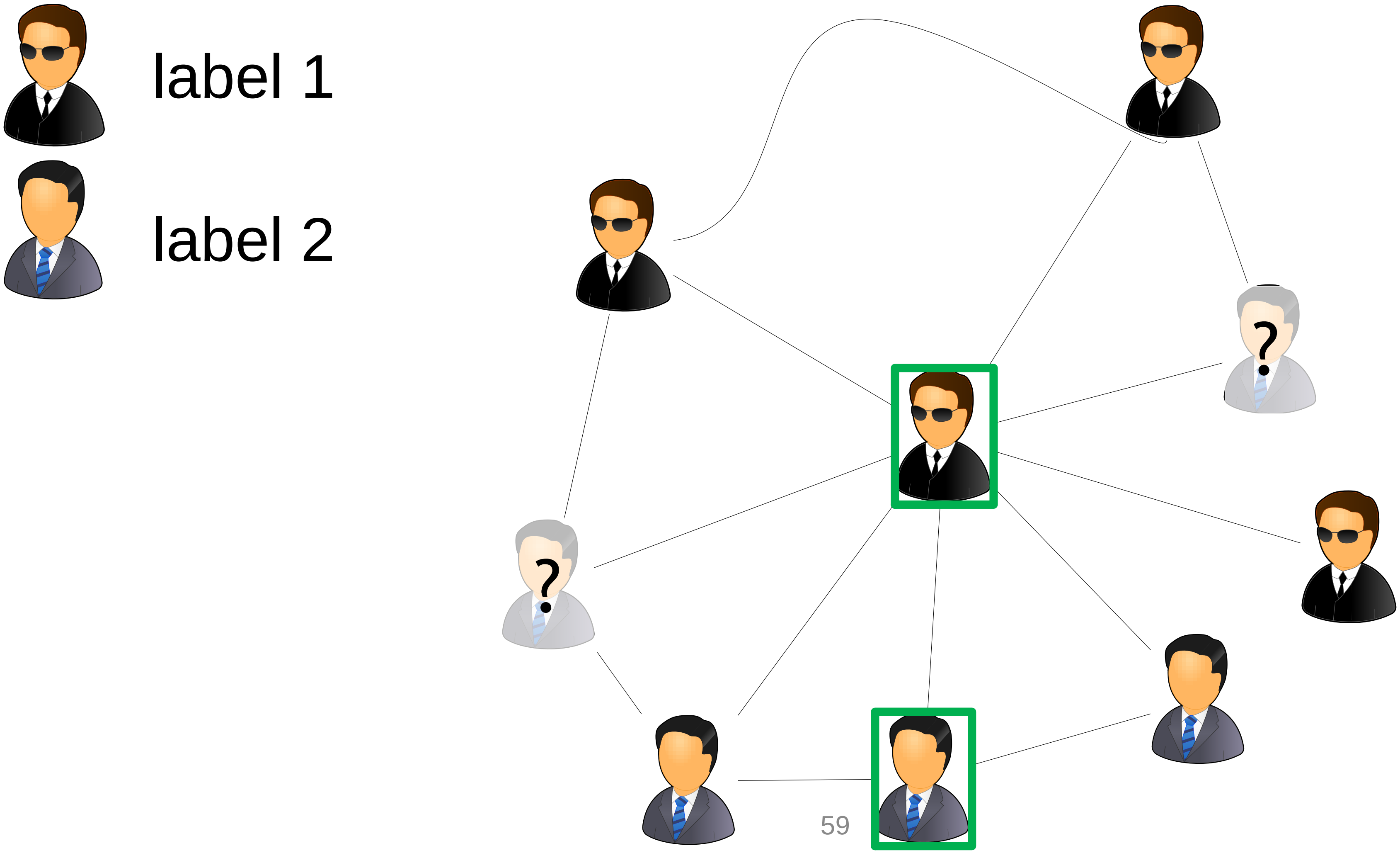




Node Classification using Label-dependent Features



Node Classification using Label-dependent Features

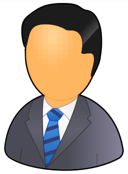


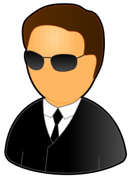
Node Classification using Label-dependent Features

An input vector consists of:

- Age
- Sex
- Nationality

Profile

- Degree for label 

- Degree for label 

Class Example

Telecom business (ISP)

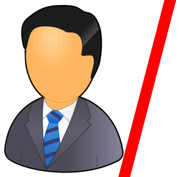
1. Internet access 20 Mbps
2. Internet access 1 Gbps
3. ? - for non customers

An input vector consists of:

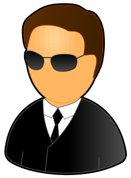
- Age
- Sex
- Nationality

Profile

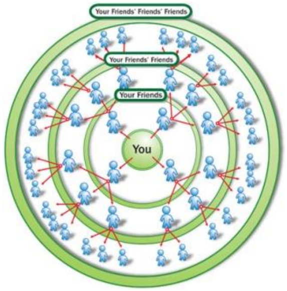
- **Degree** for label



- **Degree** for label



Experiments



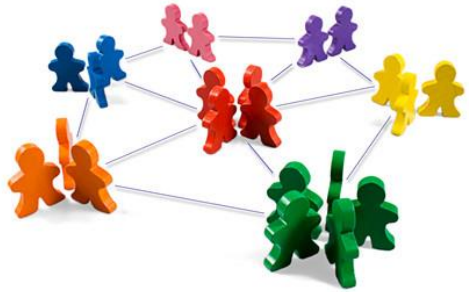
Datasets description:

- AMD dataset
  - Node – conference participator
  - Single connection – a fact that two participants were present on the same talks
  - Classification task: assign label of **participant's interest**
- CORA dataset
  - Node – paper
  - Single connection – citation between papers
  - Classification task: assign label of the **paper discipline**

- Used Label-dependent features: betweenness, degree, CC

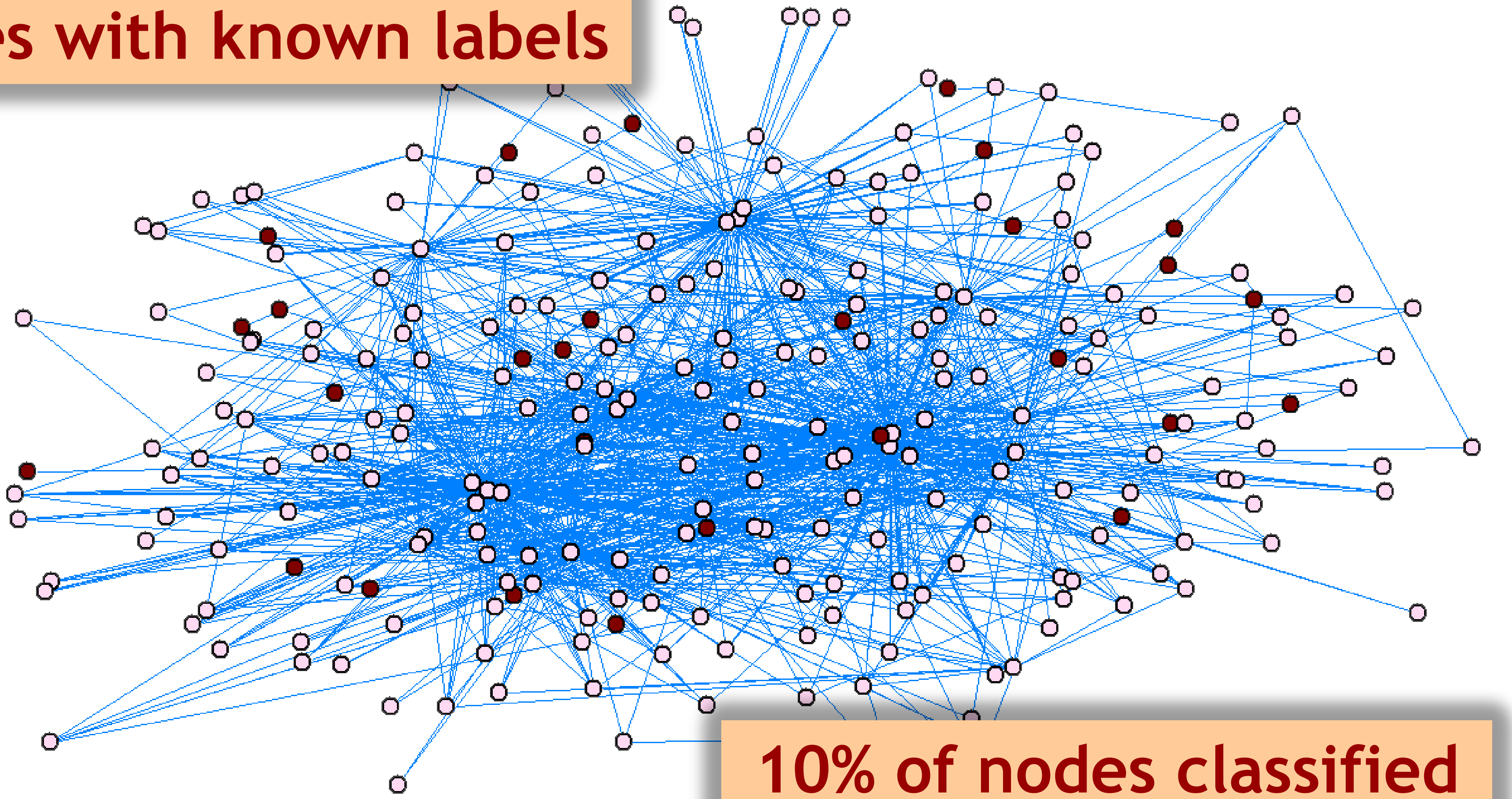
Experiments - Data Sets

Dataset	No. of node attributes (profile)	No. of nodes	No. of links	Directed links	Weighted links
AMD	4	334	68,770	☑	☑
CORA	4	6,527	10,394	☑	☒



Experiments - AMD

Different contribution of nodes with known labels



10% of nodes classified (the black ones)

Feature Sets

Set No.	Features
1	1. Age, 2. Gender,3. County 4. Phone provider
2	1. Betweenness 2. Degree 3. Clustering coefficient
3	1. Normalized sum of relation strengths to the neighbours with ‘0’ 2. Normalized sum of relation strengths to the neighbours with ‘1’ 3. Normalized no. of relations to the neighbours labeled with ‘0’ 4. Normalized no. of relations to the neighbours labeled with ‘1’ 5. Betweenness in the neighborhood for class ‘0’ 6. Betweenness in the neighborhood for class ‘1’ 7. Degree within the neighborhood for class ‘0’ 8. Degree within the neighborhood for class ‘1’ 9. Clustering coefficient within the neighborhood for class ‘0’ 10. Clustering coefficient within the neighborhood for class ‘1’
4	1+2+3 (all above features)

Profiles

Label-independent

Label-dependent

Feature Sets

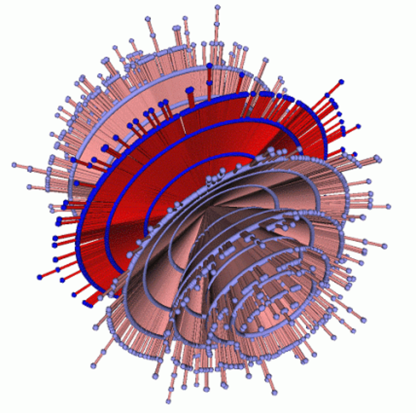
Set No.	Features	
1	1. Age, 2. Gender,3. County 4. Phone provider	Profiles
2	1. Betweenness 2. Degree 3. Clustering coefficient	Label-independent
3	10. Clustering coefficient within the neighborhood with class '1'	- ent
4	1+2+3 (all above)	

Verification of classification:

1. **Real interest** declared by conference participants
2. **Interest predicted** by the model

Classification Algorithms

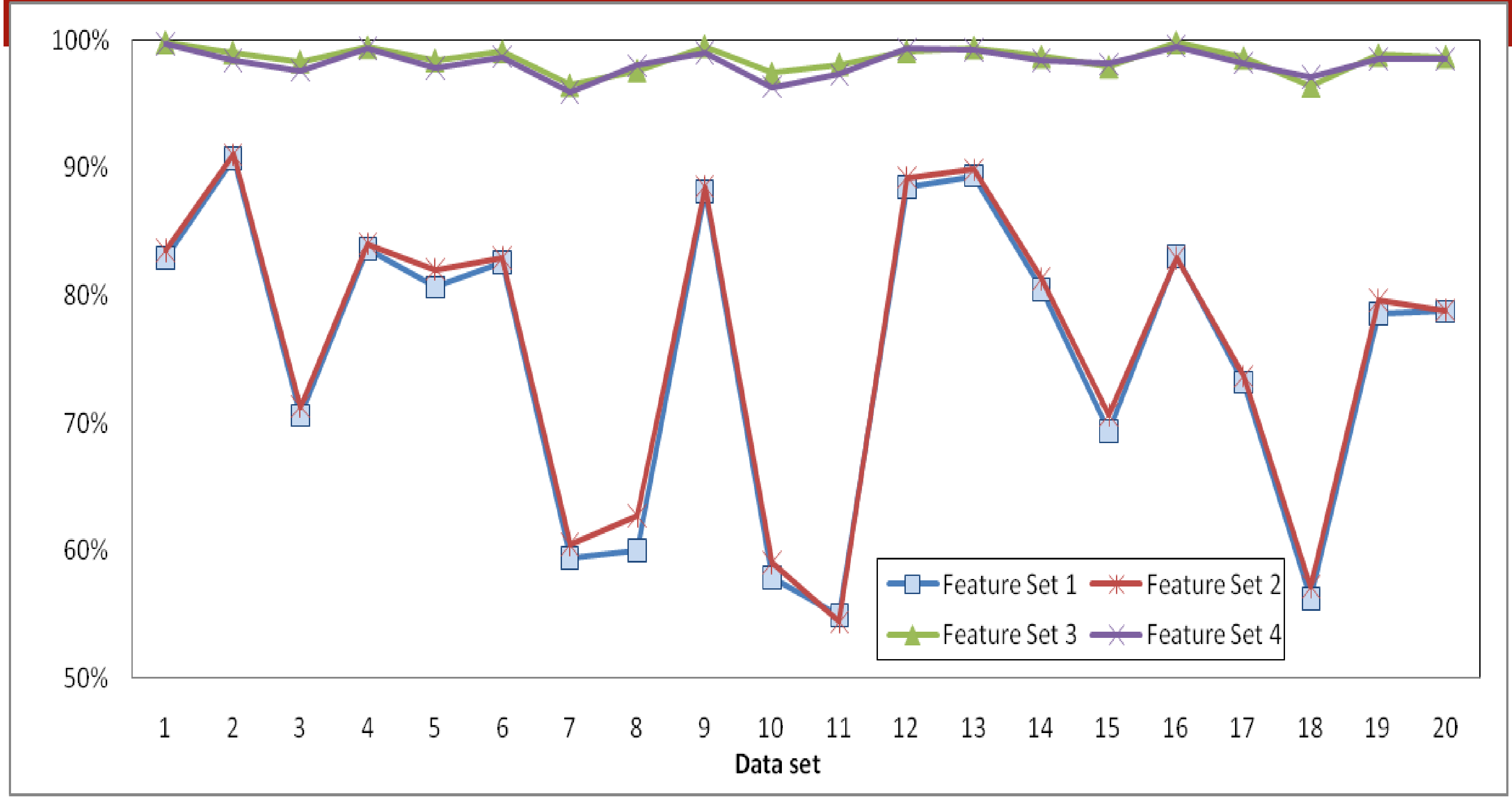
- Three algorithms (base models)
  - AdaBoost
  - Multilayer Perceptron
  - SVM



- 10-fold cross-validation
- 20 different contributions of known nodes (labels), 10%-90%

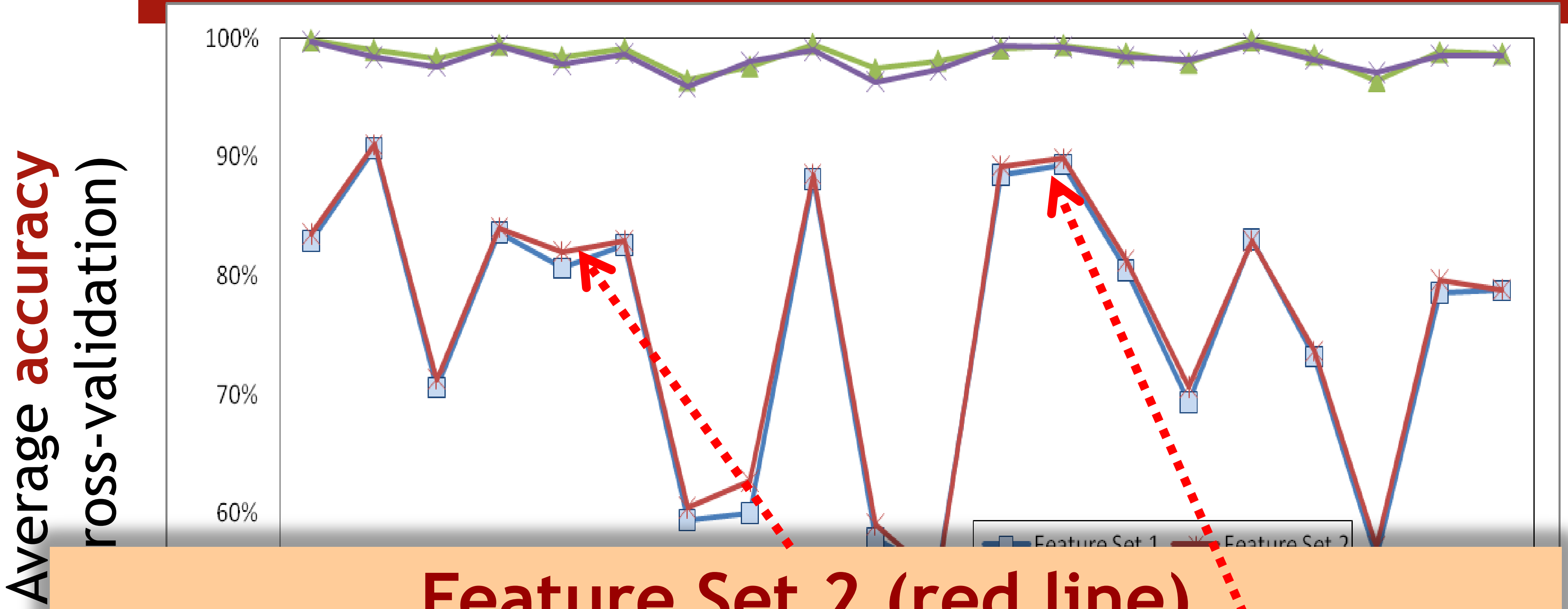
Results AMD

Average accuracy (cross-validation)



Data set: different participant interests

Results AMD



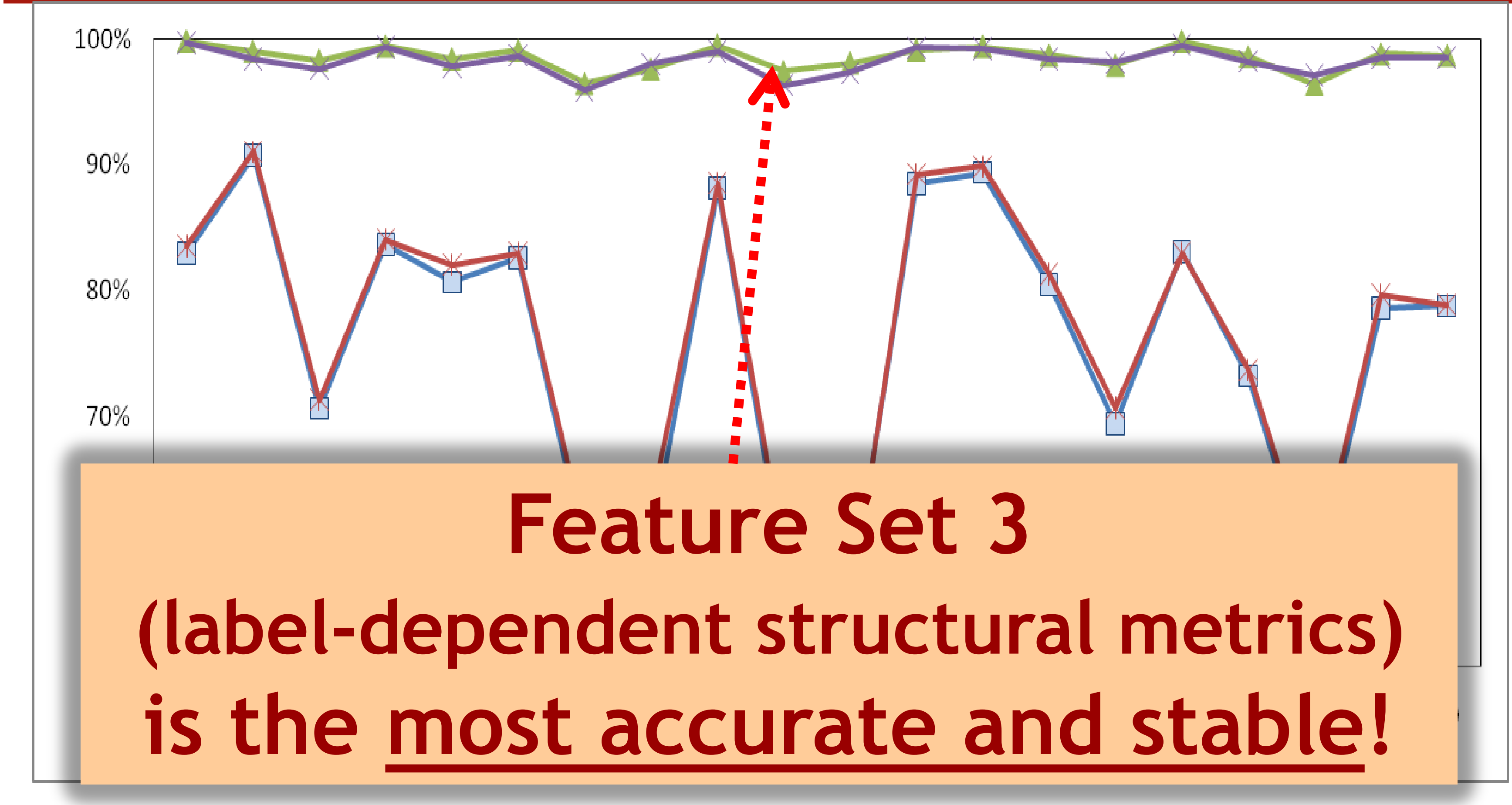
Feature Set 2 (red line)

(label-independent structural metrics)

performs <u>bad</u> but a bit better than profiles!

Results AMD

Average accuracy
(cross-validation)



Feature Set 3
(label-dependent structural metrics)
is the most accurate and stable!

Wrocław University of Technology

Ensemble Classification

Ensemble Classification

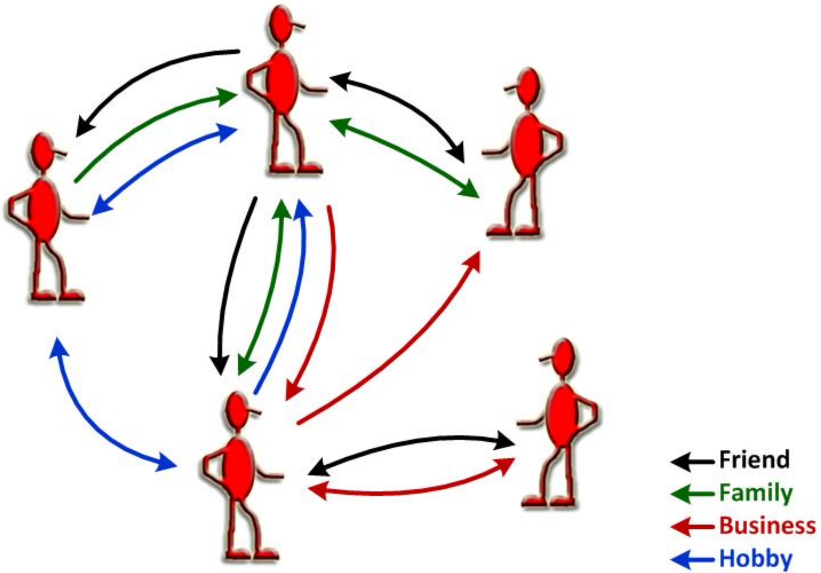
- Methods which combine different models
- Increases classification accuracy
- Usage
  - Combine results achieved by relational classification for different relations
  - Combine results of relational and local models
- Voting

$$P(c|x) = \frac{1}{L}\cdot\sum_{i=1}^{L} P(c|x)_i$$

- Stacking
  - Use Meta-classifier to learn a model on the results of different models
  - Build new instances $x_{new} = (P(c|x)_1, P(c|x)_2, \ldots, P(c|x)_L)$

Classication in Multi-layered Social Networks

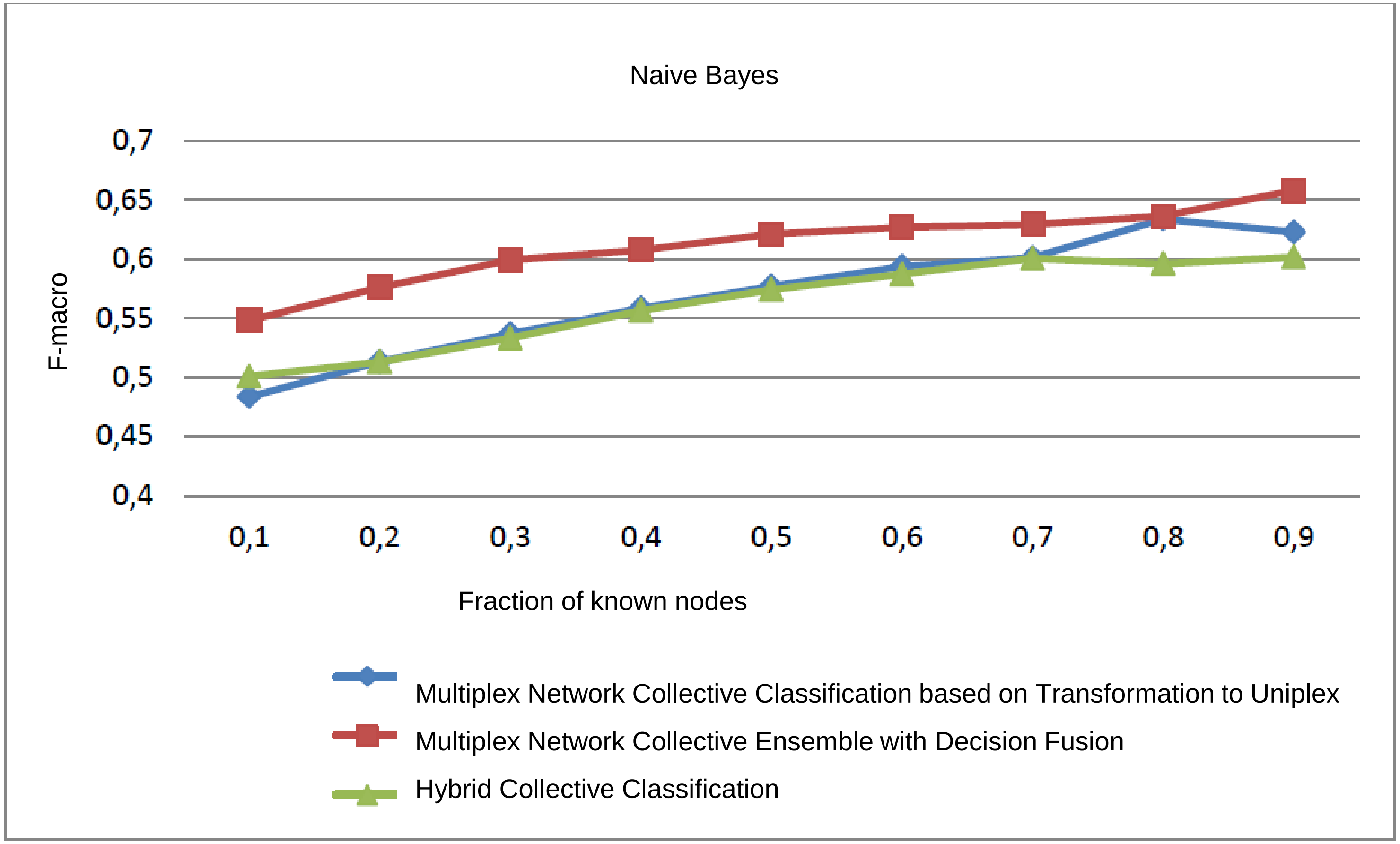
- Problem
  - standard collective classification models work only on unimodal networks (single type of relations between nodes)
- Solution
  - Ensemble collective classication methods
  - or collective fusion



Preliminary proposals

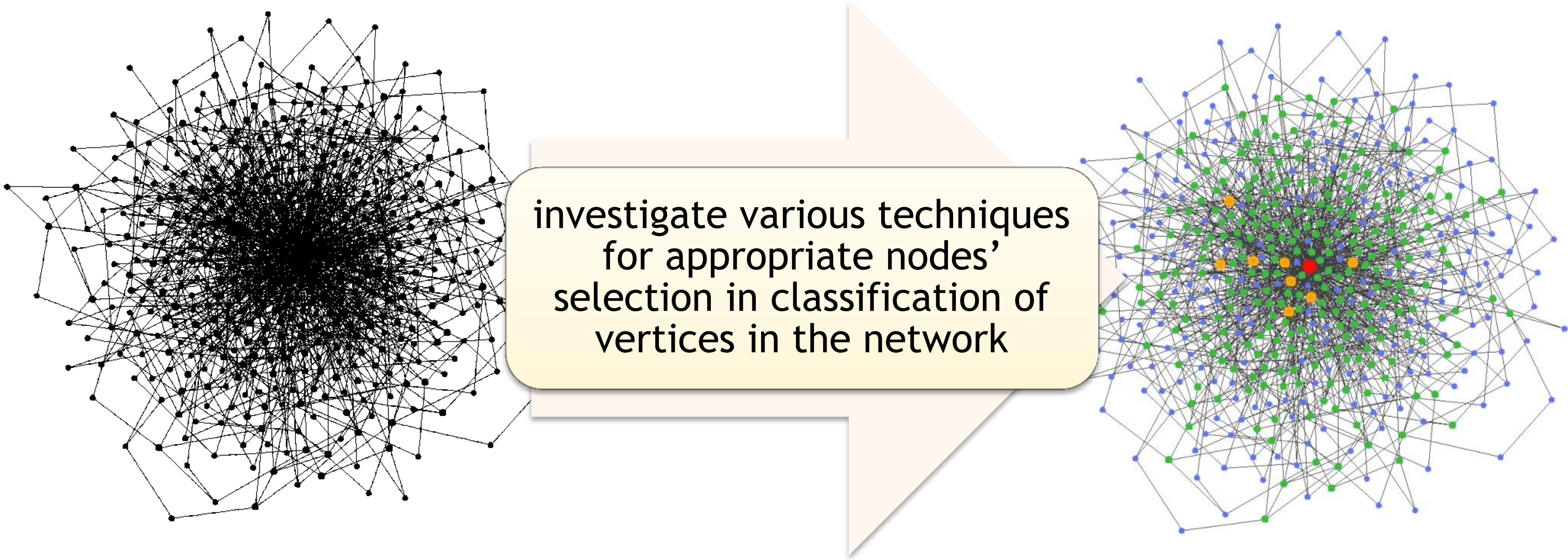
- Multiplex Network Collective Classification based on Transformation to Uniplex
- Multiplex Network Collective Ensemble with Decision Fusion
- Hybrid Collective Classification

First results



Active learning and inference

Goal



investigate various techniques for appropriate nodes' selection in classification of vertices in the network

Active learning

- Passive learning vs. active learning
  - Passive methods
    - all labels for an unlabelled dataset are obtained once
  - Active learning
    - learner has some role in determining on what data it will be trained
    - used when obtaining labeled data is expensive or time-consuming
    - Identifying which observations are most likely to be useful
- In some cases the number of nodes to be queried for labels is **logarithmic** when comparing to passive methods in order to achieve **similar accuracy** [Beygelzimer, 2009]

A. Beygelzimer, S. Dasgupta, and J. Langford. Importance weighted active learning. In Proceedings of the 26-th International Conference on Machine Learning, 2009.

Problem description

- $G = (V, E)$ graph with nodes and edges
- each $V_i \in V$ described by pair $\langle \vec{X}_i, Yi \rangle$ (attributes vector and class label)
- each edge $E_{ij} \in E$ describes some sort of relationship between $V_i$and V_j
- c_{kl} - misclassification cost of node’s label (wrongly assigned class y_k instead of correct y_l)

Problem description

- Across-network classification (active inference)
  - underlying collective model already learned
  - expected misclassification cost (EMC):

$$EMC(Y_i|X = x) = \min_{y_k} \sum_{y_l \neq yk} P(Y_i = y_l|X = x)\, c_{kl}$$

- objective: find optimal set A of labels to acquire, such that the total cost of acquisition $C(A)$ and EMC is minimized:

$$C(A) + \sum_{Y_i \in Y \setminus A} \sum_{a} P(A = a) EMC(Y_i|X = x, A = a)$$

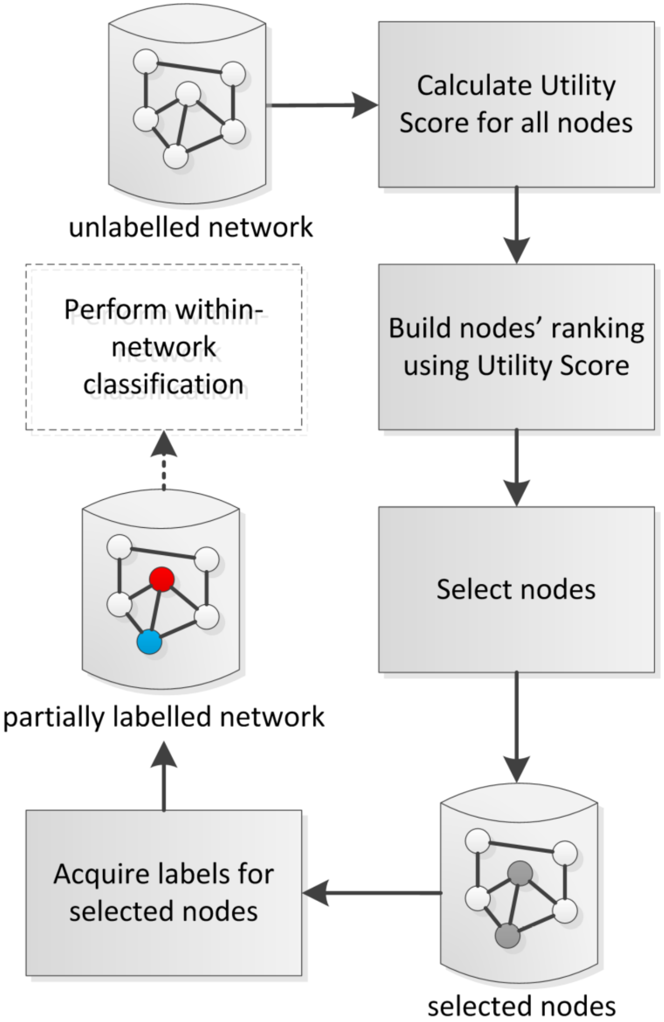
M. Bilgic and L. Getoor. Active inference for collective classification. In Proceedings of Twenty-Fourth Conference on Artificial Intelligence AAAI10. AAAI Press, 2010.

Within-network classification

- expected misclassification error depends additionally on abilities of relational classification algorithm Φ that is learnt on acquired labels
- objective:

$$C(A) + \sum_{Y_i \in Y \setminus A} \sum_{a} P(A = a) EMC(Y_i | X = x, A = a, \Phi(A))$$

Active learning and inference method



unlabelled network

Calculate Utility Score for all nodes

Build nodes' ranking using Utility Score

Select nodes

selected nodes

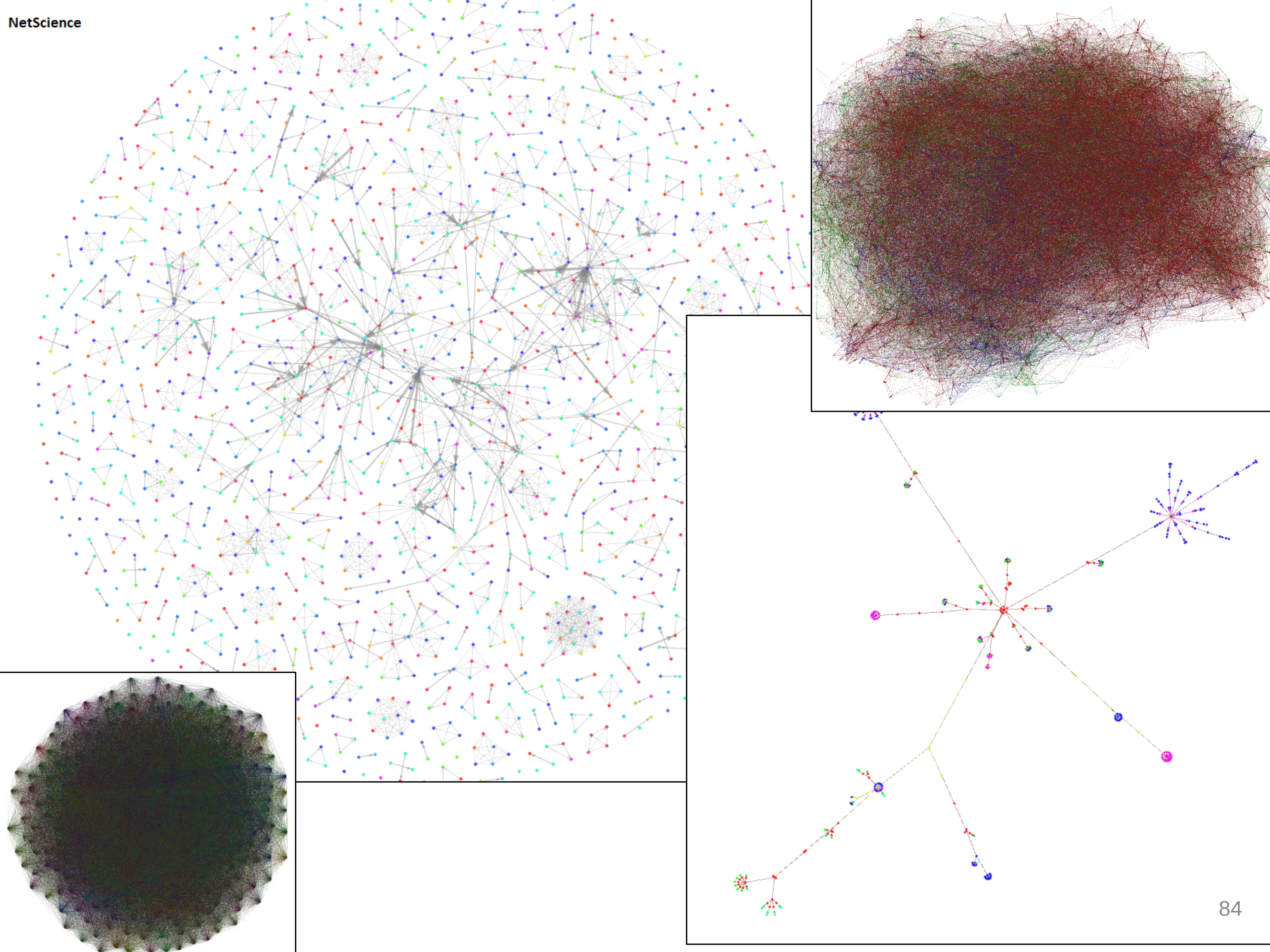
Acquire labels for selected nodes

partially labelled network

Perform within-network classification

Datasets

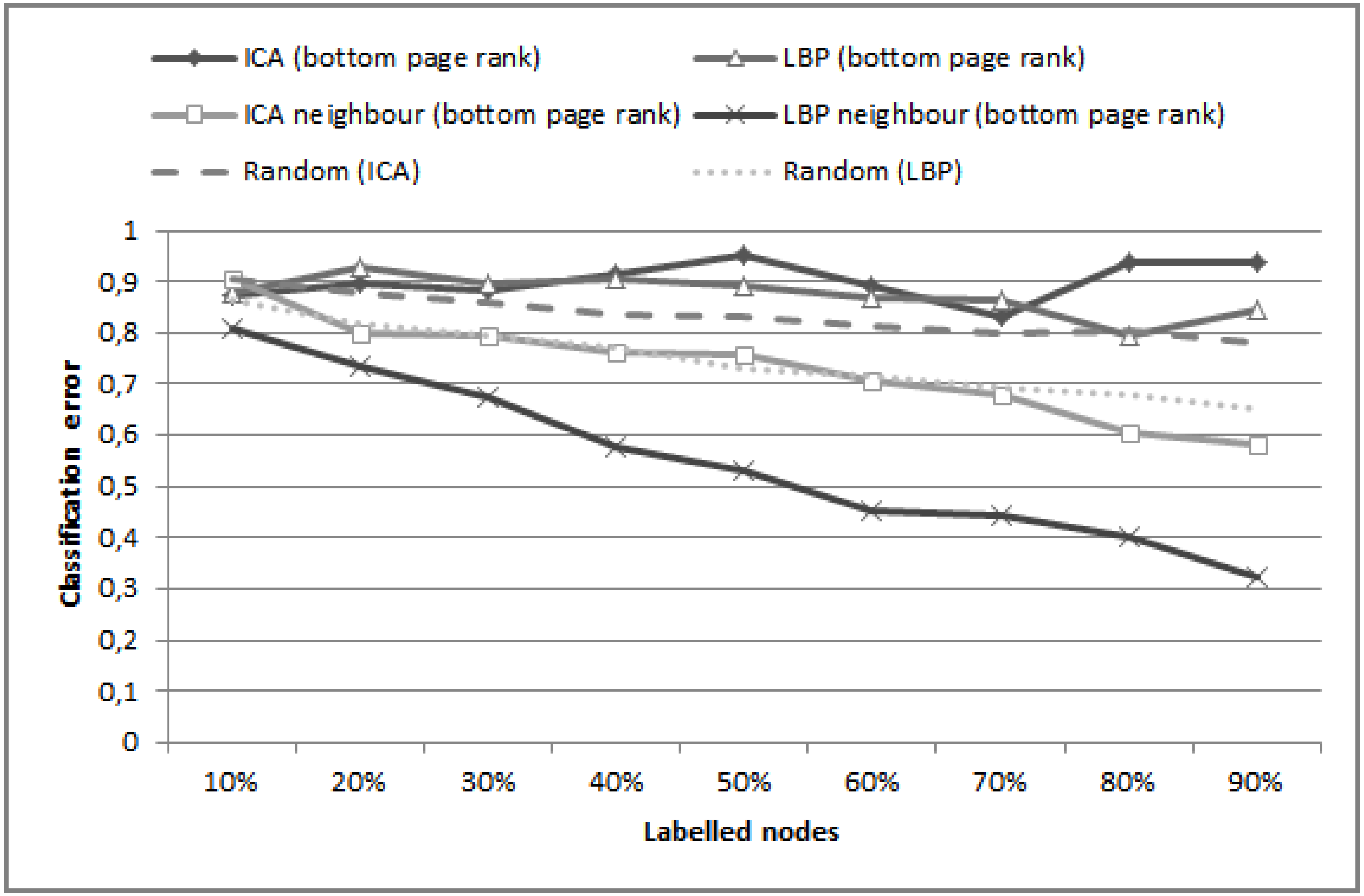
Dataset	Vertices	Edges	Classes	Avg. Deg.	Type
AMD_NETWORK	332	69092	16	208,108	Attendance on conference
ARTIFICIAL	413	415	6	1,004	artificial
CRN	327	324	4	0,990	artificial
CS PHD	1451	924	16	0,636	PhD students -advisers
NET SCIENCE	1588	2742	26	1,726	co-authorship network
PAIRS FSG	4931	61449	3	12,461	word association in dictionary
PAIRS FSG SMALL	1972	12213	3	6,193	word associationin dictionary
YEAST	2361	2353	13	0,996	protein - protein interaction network


NetScience

Experiments

- utility scores:
  - indegree centrality
  - outdegree centrality
  - betweenness centrality
  - clustering coefficient
  - hubness
  - authority
  - page rank
- measure-neighboorhood utility scores
- Iterative Classification (ICA) and Loopy Belief Propagation (LBP)

Experiments



AMD dataset

Conclusions

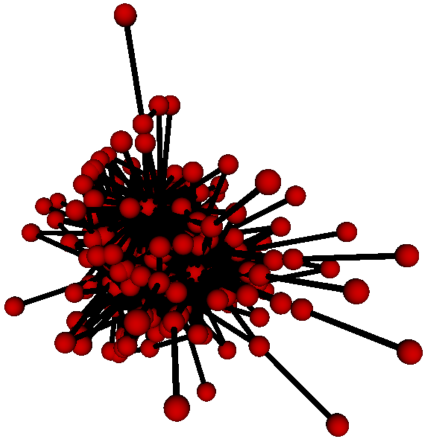
- small-world networks - good accuracy of measure-neighbour methods with LBP neighbourhood (outperforming other approaches)
- small modularity and density, greate clustering coefficient LBP neighbour approach outperforms others
- random networks with a very low connectivity, measure-neighbour methods are worse than original and random approaches

Publications

1. Kajdanowicz T., Popiel A., Kulisiewicz M., Kazienko P.: Ensamble Relational Classification in Multiplex Networks based on Decision Fusion, in reviews, 2014
2. Kajdanowicz T., Michalski R., Musial K., Kazienko P.: Learning in Unlabelled Networks - An Active Learning and Inference Approach. in reviews, 2014
3. Michalski R., Kajdanowicz T., Bródka P., Kazienko P.: Seed Selection for Spread of Influnece in Social Networks: Temporal vs. Static Approach. New Generation Computing, accepted, 2014.
4. Kajdanowicz T, Kazienko P and Indyk W (2014), "Parallel Processing of Large Graphs", Future Generation Computer Systems. Vol. 32, pp. 324-337.
5. Indyk W, Kajdanowicz T and Kazienko P (2013), "Relational large scale multi-label classification method for video categorization", Multimedia Tools and Applications. Vol. 65(1), pp. 63-74.
6. Kajdanowicz T and Kazienko P (2013), "Boosting-based Multi-label Classification", Journal of Universal Computer Science. Vol. 19(4), pp. 502-520.
7. Filipowski T, Kazienko P, Bródka P and Kajdanowicz T (2012), "Knowledge Exchange through Social Links in the Workplace", Behaviour and Information Technology. Vol. 31(8), pp. 779-790.
8. Kajdanowicz T and Kazienko P (2012), "Multi-label Classification Using Error Correcting Output Codes", International Journal of Applied Mathematics and Computer Science. Vol. 22(4), pp. 829-840.
9. Kajdanowicz T, Plamowski S and Kazienko P (2012), "New Entropy Based Distance for Training Set Selection in Debt Portfolio Valuation", International Journal of Information Technology and Web Engineering. Vol. 7(2), pp. 60-69.
10. Kajdanowicz T, Indyk W and Kazienko P (2012), "MapReduce Approach to Relational Influence Propagation in Complex Network", Pattern Analysis and Applications.
11. Kazienko P and Kajdanowicz T (2012), "Label-dependent node classification in the network", Neurocomputing. Vol. 75, pp. 199-209.
12. Kajdanowicz T and Kazienko P (2011), "Boosting-based sequential output prediction", New Generation Computing. Vol. 29(3), pp. 293-307.
13. Kazienko P, Musial K and Kajdanowicz T (2011), "Multidimensional social network in the social recommender system", IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans. Vol. 41(4), pp. 746-759.
14. Pasierb K, Kajdanowicz T and Kazienko P (2011), "Privacy-preserving Data Mining, Sharing and Publishing", Journal of Medical Informatics & Technologies. Vol. 18, pp. 69-76.
15. Kazienko P and Kajdanowicz T (2010), "Base classifiers in boosting-based classification of sequential structures", Neural Network World. Vol. 20(7), pp. 839-851.

Invitation to Summer School

- Advances in Machine Learning for Social Media Analysis
  - Sentiment Analysis
  - Relational Learning
  - Probabilistic Graphical Models
  - Knowledge extration from texts
- 25-27.09.2014 Wrocław
- 29-30.2014 The First European Network Intelligence Conference(ENIC.pwr.wroc.pl)



Thanks to Collaborators

- Prof. P. Kazienko
- P. Bródka
- R. Michalski

- P. Szymański
- W. Indyk
- Ł. Augustyniak
- M. Kulisiewicz
- W. Tuligłowicz
- A. Misiaszek
- A. Popiel

Thank you for attention