

# Growing a Tree in the Forest: Constructing Folksonomies by Integrating Structured Metadata

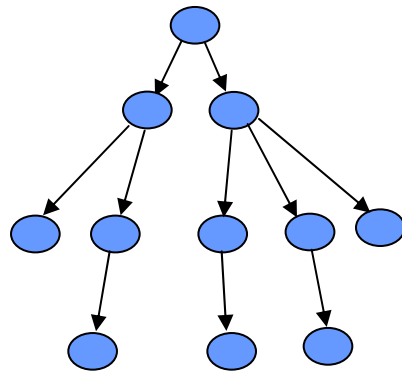
**Anon Plangprasopchok (USC/ISI)**  
**Kristina Lerman (USC/ISI)**  
**Lise Getoor (UMD)**



- **Explosion of user-generated content**
  - *Images: Flickr, Picasa,..*
  - *Videos: YouTube, Vimeo,..*
  - *Maps: WikiMapia,..*
  - *Stories: Blogs, Twitter,..*
  - *Relational Data: Metaweb, Google Base,..*
  
- **User-generated semantics: annotation/metadata**
  - *Tags, Geotags*
  - *Personal Hierarchies*

**Goal: extract users' knowledge (folk knowledge) from this metadata**

Folksonomy that users commonly have in their *mind* (*hidden*)



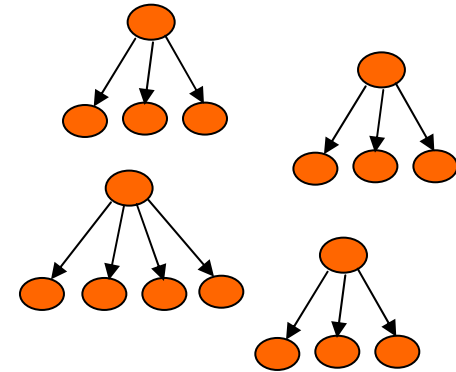
...

[deep & bushy]

*Users select a portion of the hierarchy to organize their content.*



Personal hierarchies from various users (*observed*) such as users' folder-sub folders



[shallow, noisy, sparse(incomplete) & inconsistent]

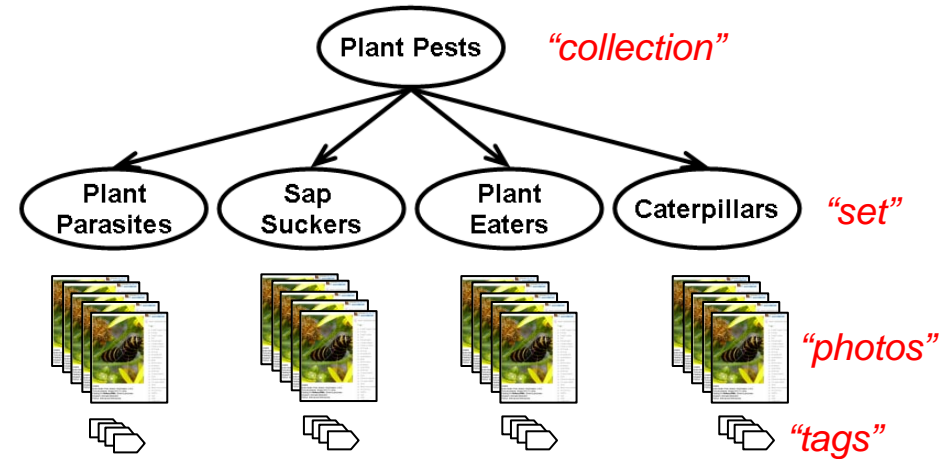


*Can we recover the folksonomy back from many observed hierarchies? → folksonomy learning*

- **Personal Hierarchies in Social Web**
- **Challenges in Folksonomy Learning**
- **Integrate Personal Hierarchies to Folksonomy**
  - *Relational Clustering for Learning Folksonomy*
- **Evaluations**
  - *Metrics*
  - *Results*
- **Related Work**
- **Discussion & Conclusions**



## Shallow Hierarchy by user "maxi millipede"

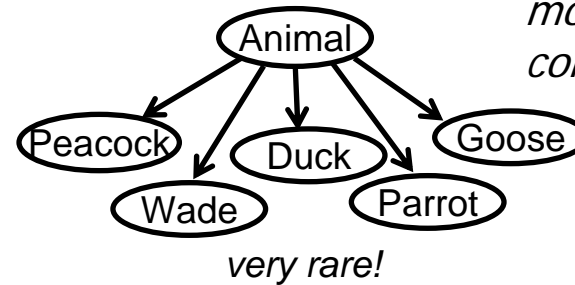
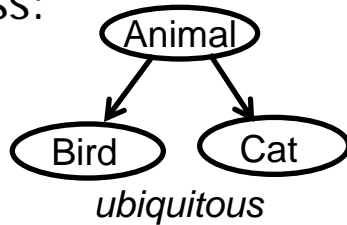


## Tags on each photo

### Assume:

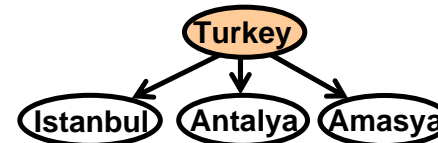
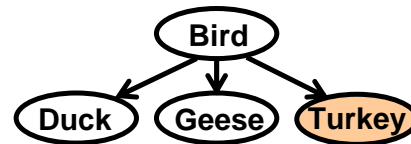
- 1) A set of tags on a set is an aggregation of all tags of all photos in the set
- 2) A set of tags on a collection is an aggregation of all tags of all sets in the collection

1.) Sparseness:

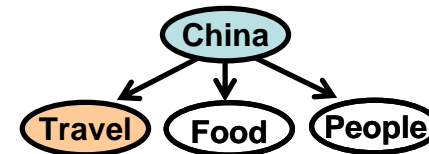
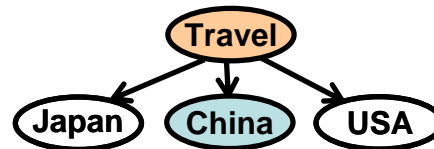


*mostly personal hierarchies  
contain very few child nodes*

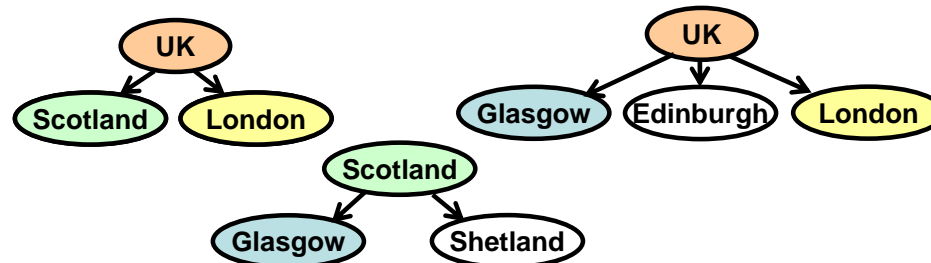
2.) Ambiguity:



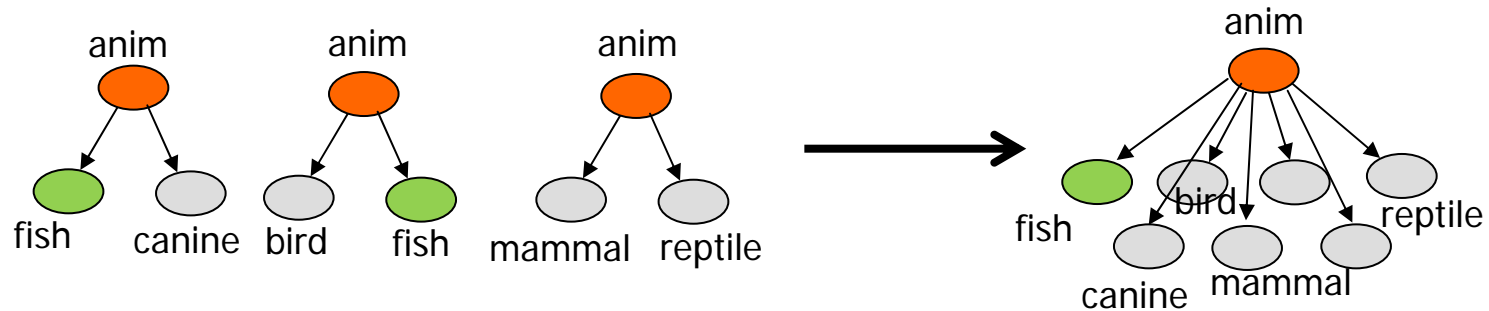
3.) Conflict:



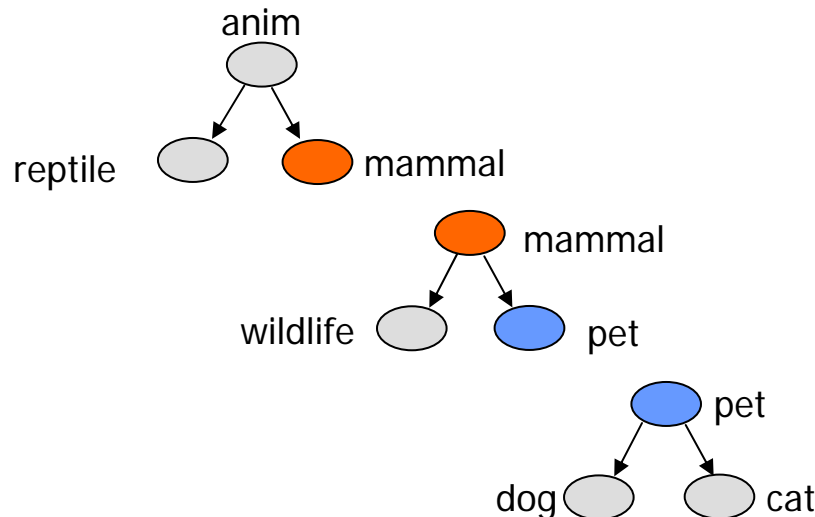
4.) Varying Granularity:



Sketched idea: *combine/aggregate personal hierarchies together in both horizontal and vertical directions.*

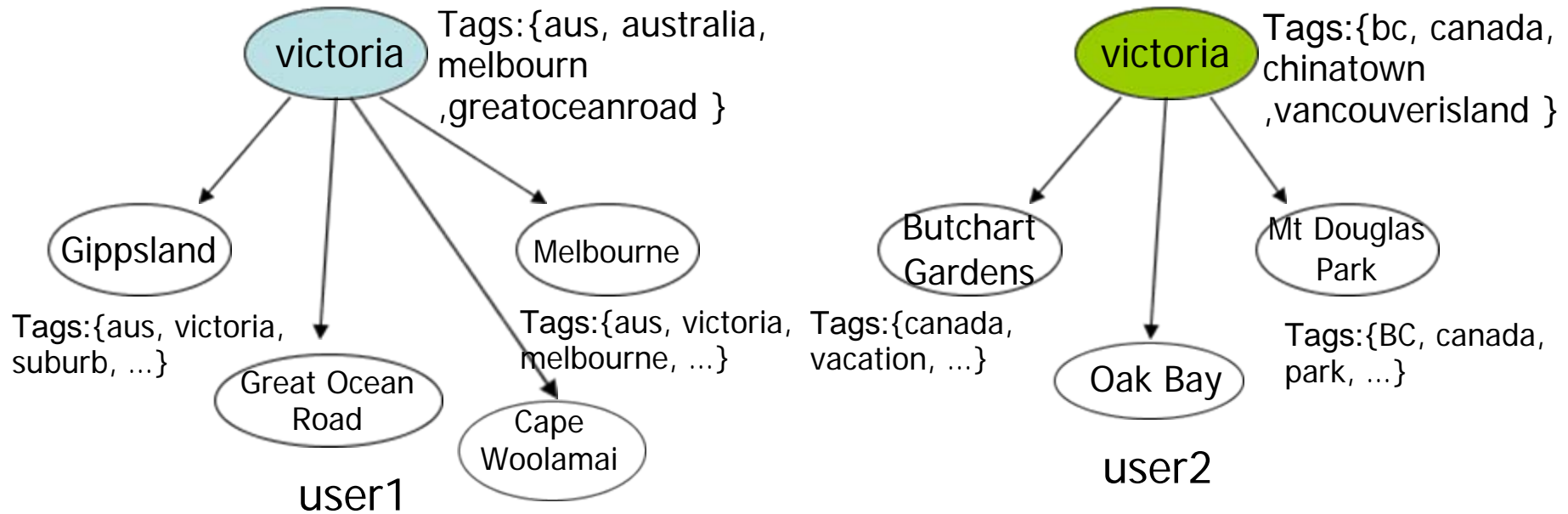


*Horizontal aggregation: expanding folksonomy's width*



*Vertical aggregation: extending folksonomy's depth*

**Sketched idea:** 2 nodes should be clustered if they are similar enough  
– similarity is computed using *contextual & relational information*

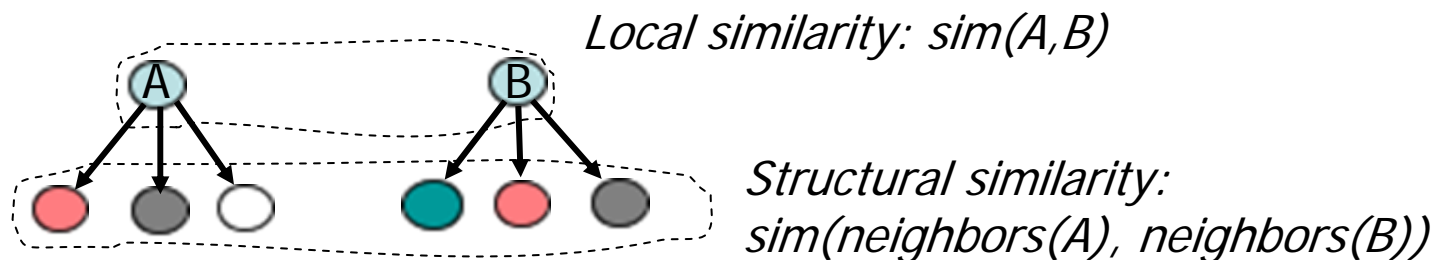


*Check common tags & child nodes*



Formally, two nodes are considered similar if:

- (1) their features are similar, i.e., have similar names, have many common tags – *local similarity*
- (2) their neighbors are similar – *structural similarity*



$$Sim(A,B) = (1-\alpha)*localsim(A,B) + \alpha*structuralSim(A,B)$$

$\alpha$  is a weight on how much we rely on structural information

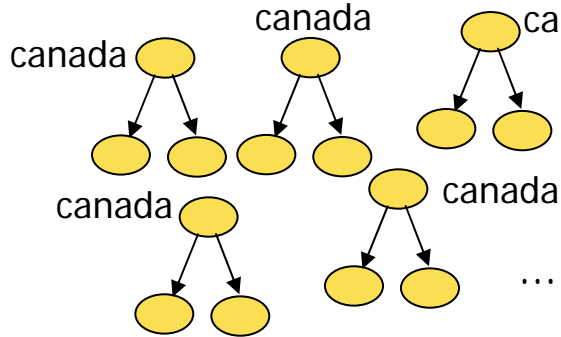
We then merge nodes together if they are similar enough.

Note that we use naïve version of relational clustering by simply using neighbors' local features to compute structural similarity, instead of neighbors' class labels\*.

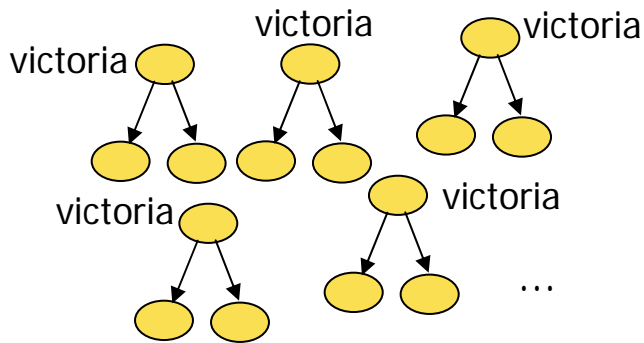
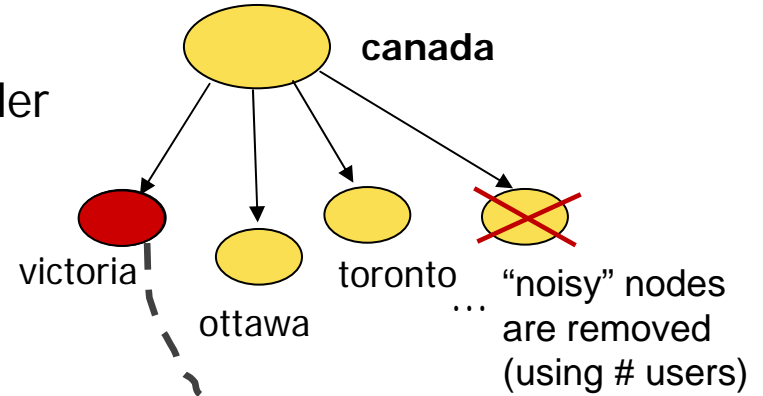
# Incremental Relational Clustering for Learning Folksonomy



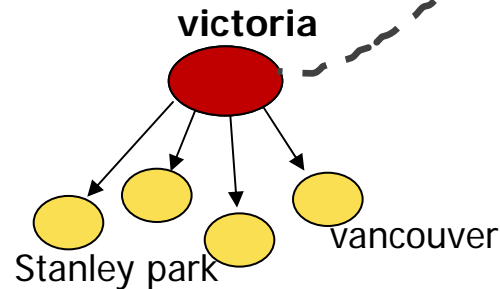
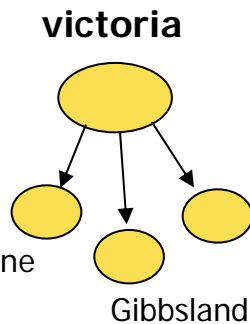
- Pick a seed(root) term, e.g., "canada"



Making it broader

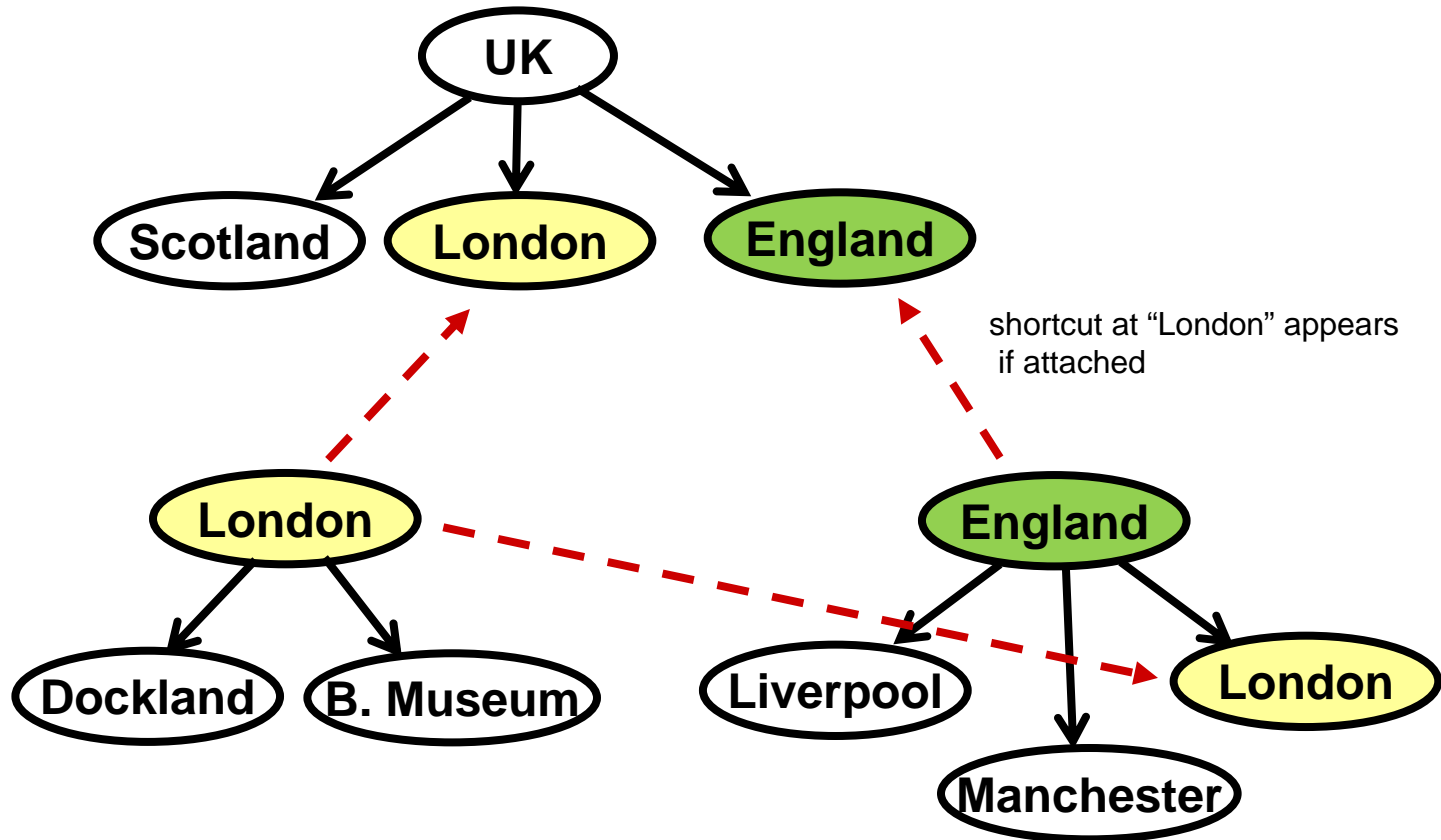


Making it deeper

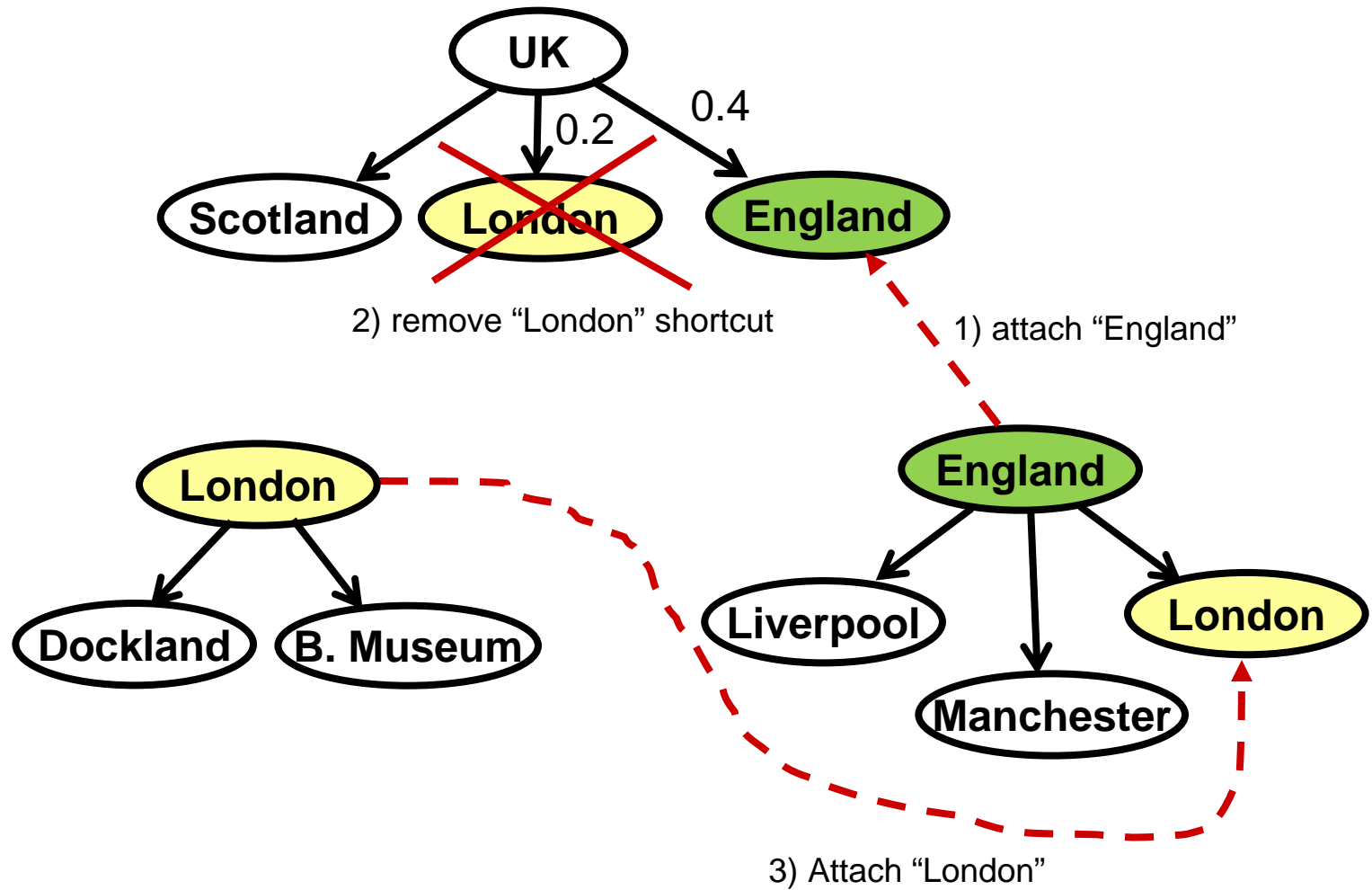


& remove **structural inconsistencies** that may appear

Suppose we have the following clusters of hierarchies:



- Shortcuts have to be removed to make the learned hierarchy consistent



- Keep the longer path since it captures more specific knowledge

# Evaluations

- Growing trees from 32 seed terms & uses personal hierarchies from Flickr as in the previous work.\*

## Evaluation Methodologies:

- 1) *Against the reference hierarchy* (DMOZ)
- 2) *Structural evaluation*
- 3) *Manual evaluation*

## Baseline Approach\*

- Assume nodes having the same name refer to the same concept
- Keep the relations between node pairs if they are not generated at random (using significance test)
- Then, combine all relations into a tree

# Evaluation Metrics

## 1.) an automatic comparison to the reference hierarchy

- **Taxonomic Overlap** [adapted from Maedche & Staab] measuring structure similarity between two trees. For each node, determining how many ancestor and descendant nodes overlap to those in the reference tree.

- **Lexical Recall** measuring how well an approach can discover concepts, existing in the reference hierarchy (coverage)

## 2.) Structural evaluation

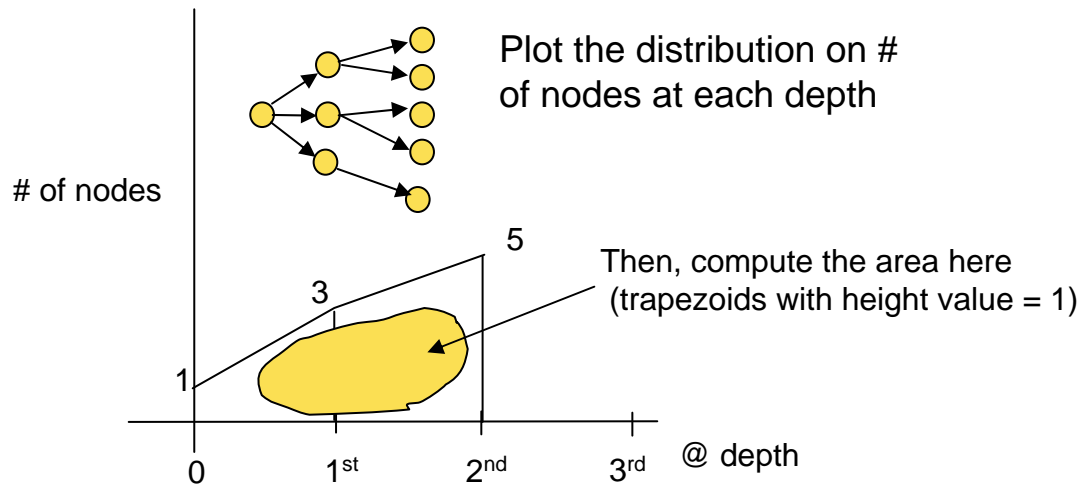
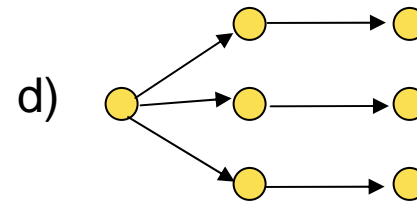
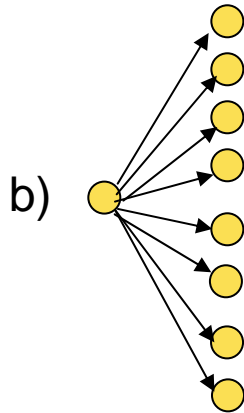
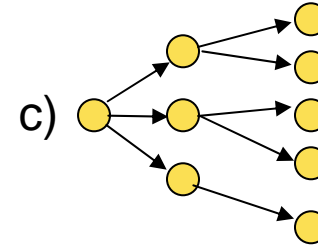
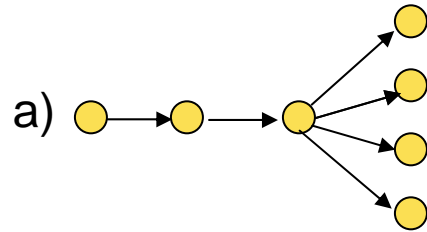
- **Area Under Tree (AUT)** combining bushiness and depth of the tree into a single number: the higher value, the bushier and deeper tree.

## 3.) Manual evaluation

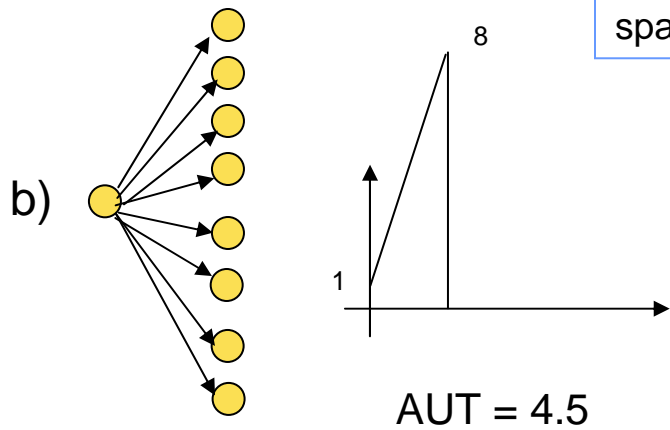
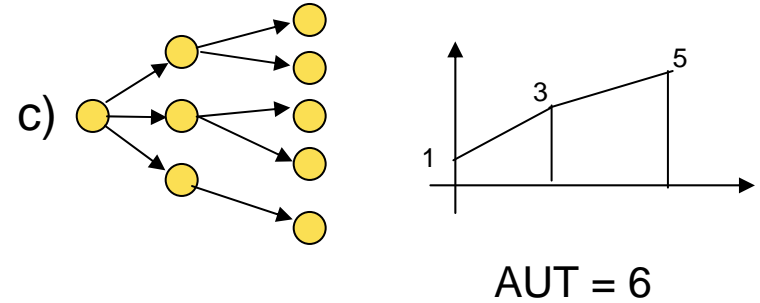
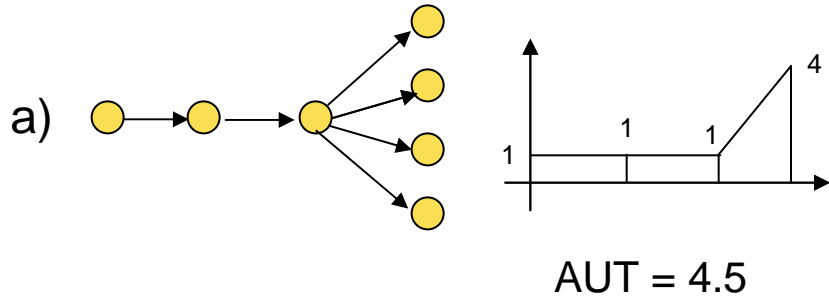
- **Accuracy**: simply asking users whether a path from root to leaf of is correct: if there are some nodes misplaced in the wrong order, users will judge the whole path incorrect

# Area Under Tree (AUT)

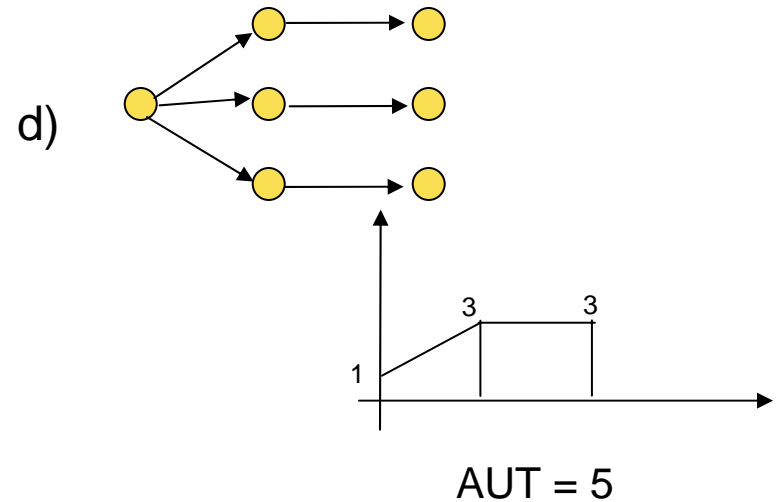
Which structures are the best in term of “bushiness” and “depth”?



# Area Under Tree (AUT)



The largest area we can get is from the tree that keeps spanning at each level



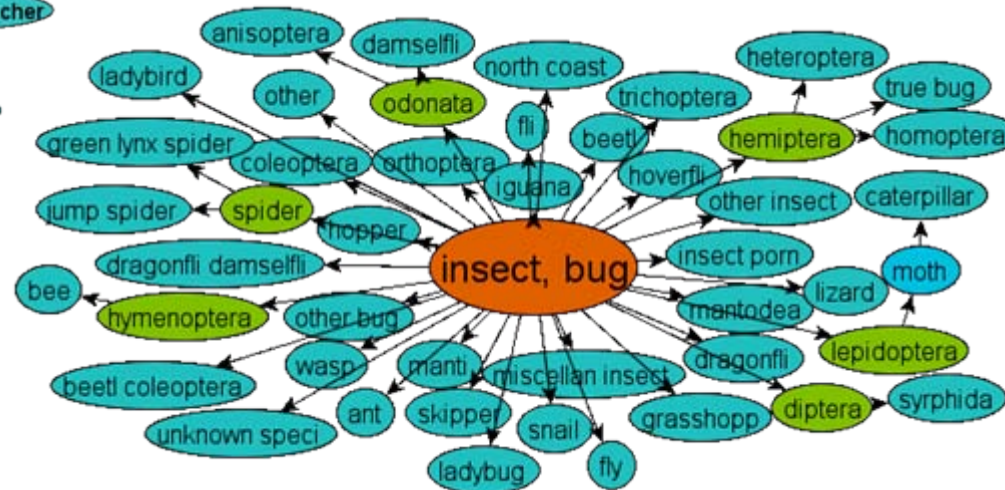
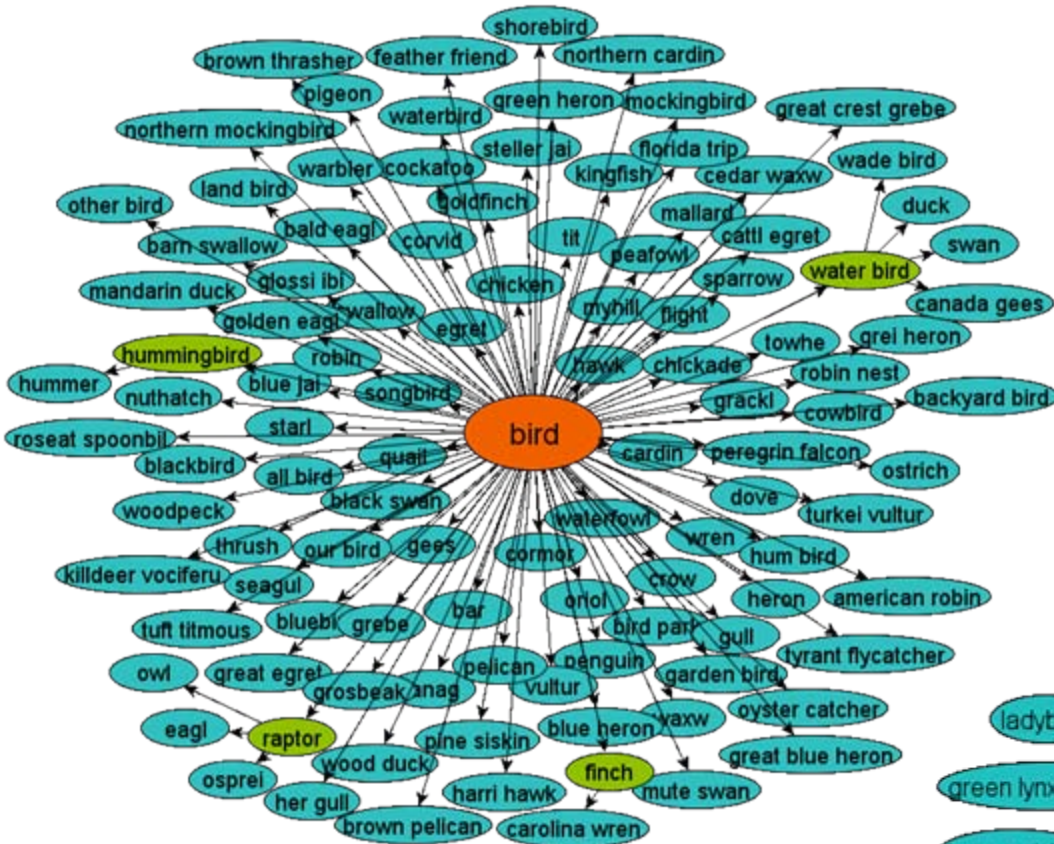


# Experimental Results

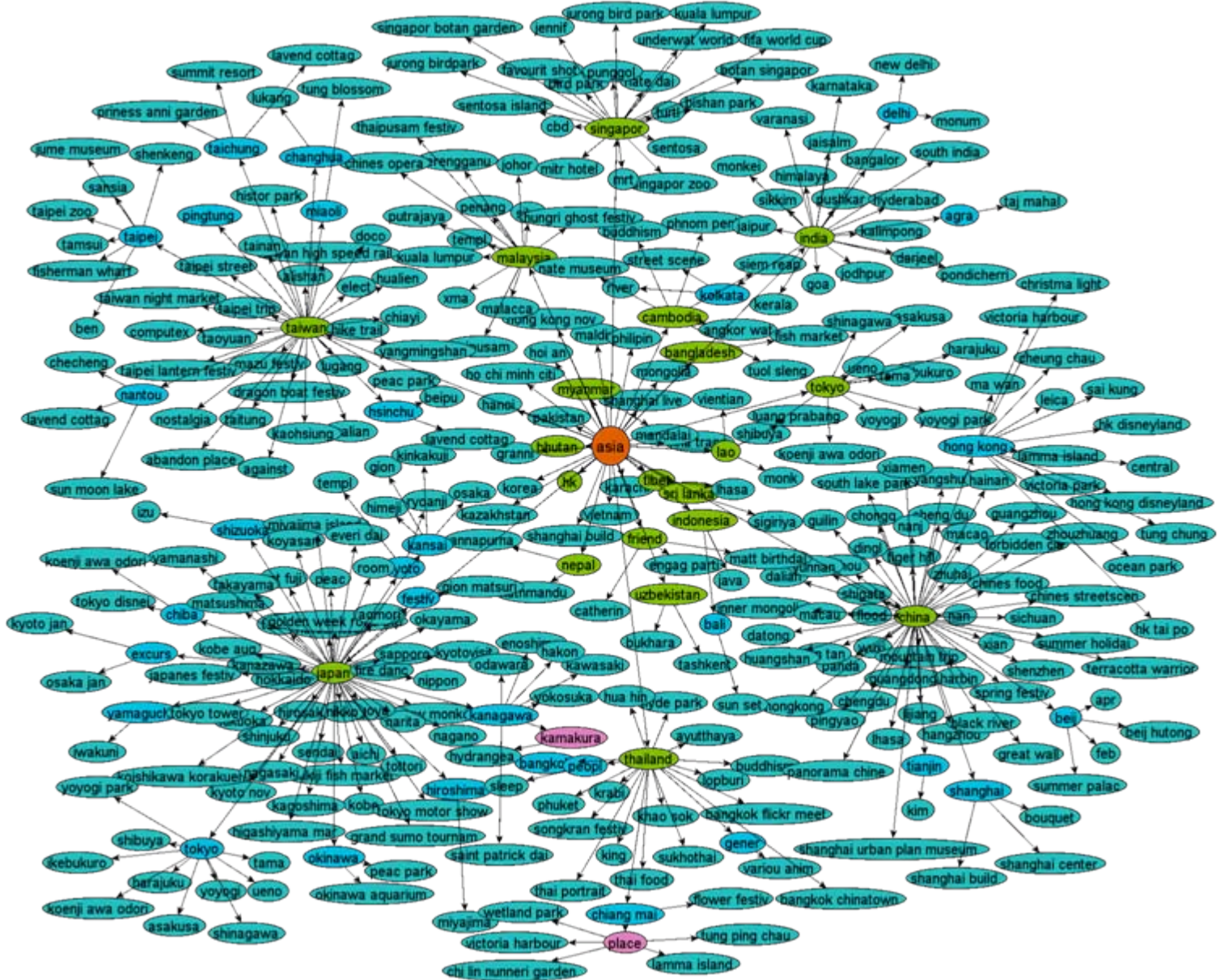
Evaluate on 32 cases

Metrics	# of cases that are superior to the other approach	
	Baseline	The present work
Taxonomic Overlap	7	15
Lexical Recall	6	19
AUT	3	18
<b>TO+LR+AUT</b>	0	11
Accuracy (Manual)	5	5

# Some of the Learned Folkonomies



# Some of the Learned Folkonomies



- **Learning concept hierarchy from text data**
  - Syntactic based [[Hearst92](#), [Caraballo99](#), [Pasca04](#), [Cimiano+05](#), [Snow+06](#)]
  - Word clustering [e.g., [Segal+02](#), [Blei+03](#)]
- **Induce concept hierarchy from tags**
  - Graph-based & clustering based [[Mika05](#), [Brooks+06](#), [Heymann+06](#), [Zhou07+](#)]
  - Probabilistic subsumption [[Schmitz06](#)]
- **Ontology alignment** [e.g., [Udrea+07](#)]
- **Exploit user-specified hierarchy for recommendation**
  - GiveALink [[Markines06+](#)]

- The present work can create **more accurate** and **more detailed** folksonomies than the current state-of-the-art approach, since it exploits structural information during the merging process
- The present work is **more scalable**: incrementally growing the folksonomies rather than using on an exhaustive search
- Future work:
  - *Automatically separate broader/narrower from related-to relations (facets)*
  - *combining more sources of evidence such as geographical information*
  - *Apply on different data sets: e.g., personal workspaces, semantic network*



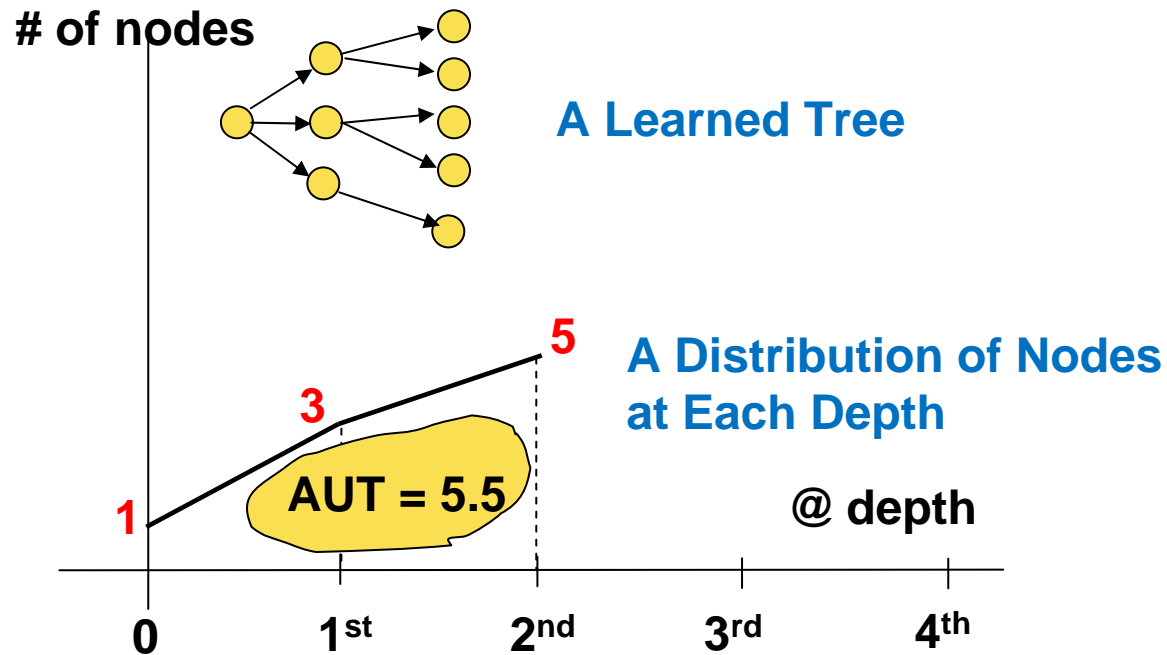
- **Thank you..**
- **Our poster is #21**

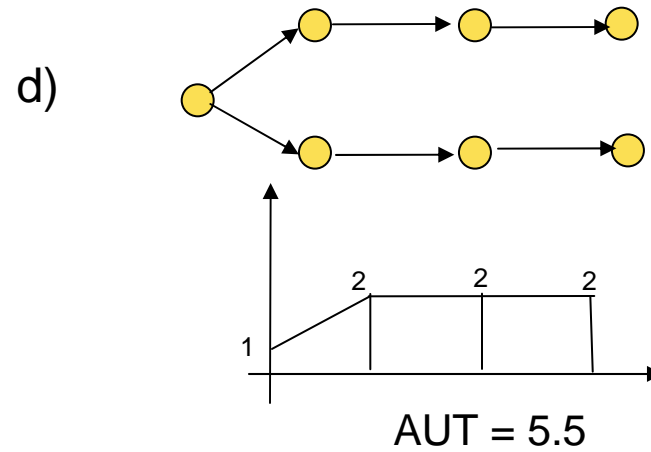
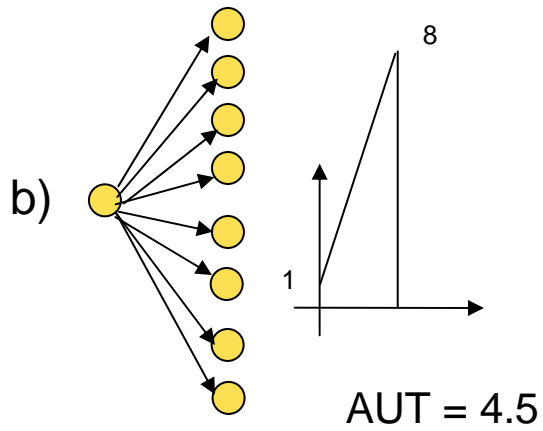
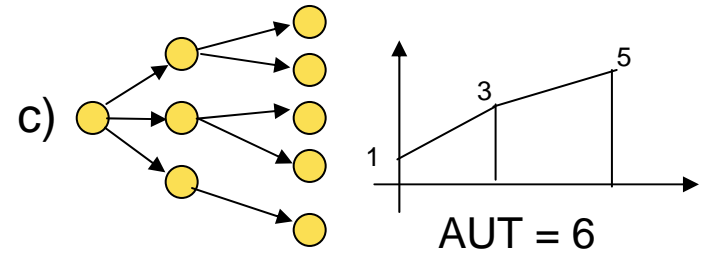
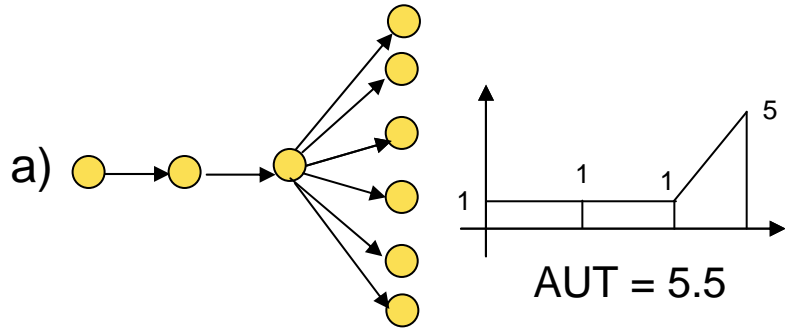
# Spares



- Personal Hierarchies in Social Web
- **Challenges in Folksonomy Learning**
- Integrate Personal Hierarchies to Folksonomy
  - *Relational Clustering for Learning Folksonomy*
- Evaluations
  - *Metrics*
  - *Results*
- Discussion & Conclusions
- Related Work







# Structural Similarity

Depends on the roles (root or leaf) of two nodes to be compared:

Root - Root:

$$\text{Let } K_{A,B} = \frac{|\text{name}(\text{leaves}(A)) \cap \text{name}(\text{leaves}(B))|}{\min(|\text{leaves}(A)|, |\text{leaves}(B)|)}$$

# of common leaf node names

for normalizing K

$$\text{structuralSim}(R1, R2) = K_{r1, r2} + (1 - K_{r1, r2}) \times \text{tagsim}(\text{leaf nodes of A, B that do not have common name})$$

Leaf - Root:

$$\text{structuralSim}(L1, R2) = \text{structuralSim}(\text{root}(L1), R2)$$

Leaf - Leaf:

*If the roots of node A and B are considered similar, we simply say A and B are similar if they have the same name.*

Plot the distribution on # of nodes at each depth

