Semi-Supervised Feature Selection for Graph Classification

Xiangnan Kong, Philip S. Yu

Department of Computer Science
University of Illinois at Chicago

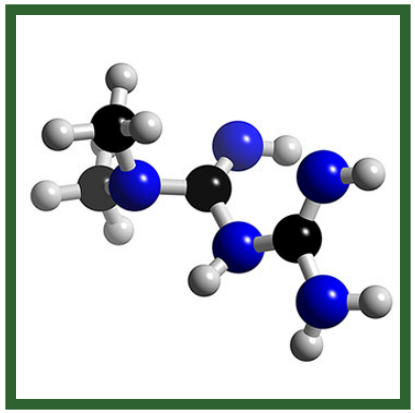
KDD 2010

Graph Classification
- why should we care?

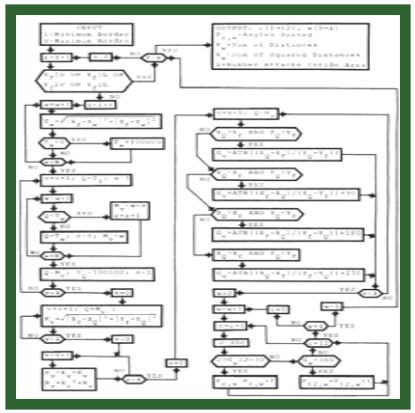
- Conventional data mining and machine learning approaches assume data are represented as feature vectors. E.g. $(x_1, x_2, \ldots, x_d) - y$

- In real apps, data are not directly represented as feature vectors, but **graphs** with complex structures.
  E.g. $G(V, E, l) - y$

Chemical Compounds



Program Flows

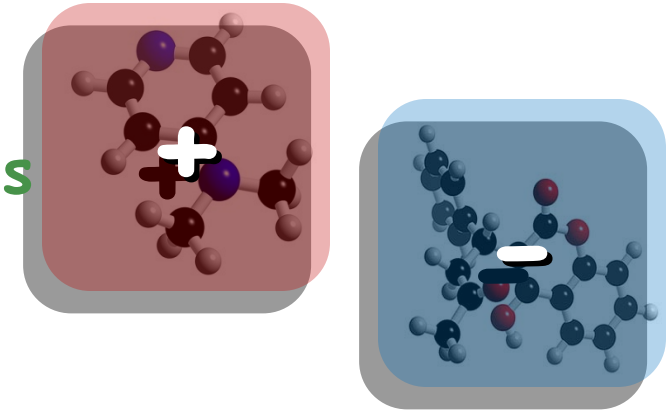


XML Docs

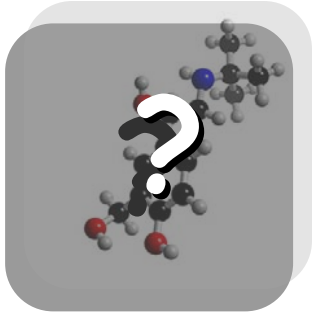
Example: Graph Classification

- **Drug activity prediction** problem
- Given a set of **chemical compounds** labeled with **activities**
- Predict the activities of testing molecules

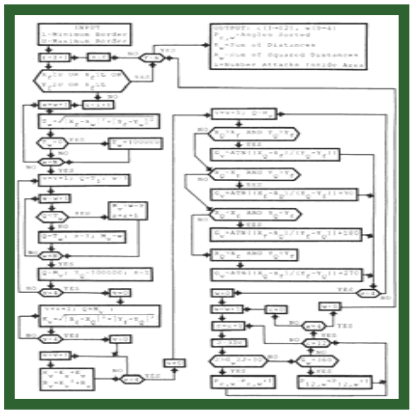
Training Graphs



Testing Graph

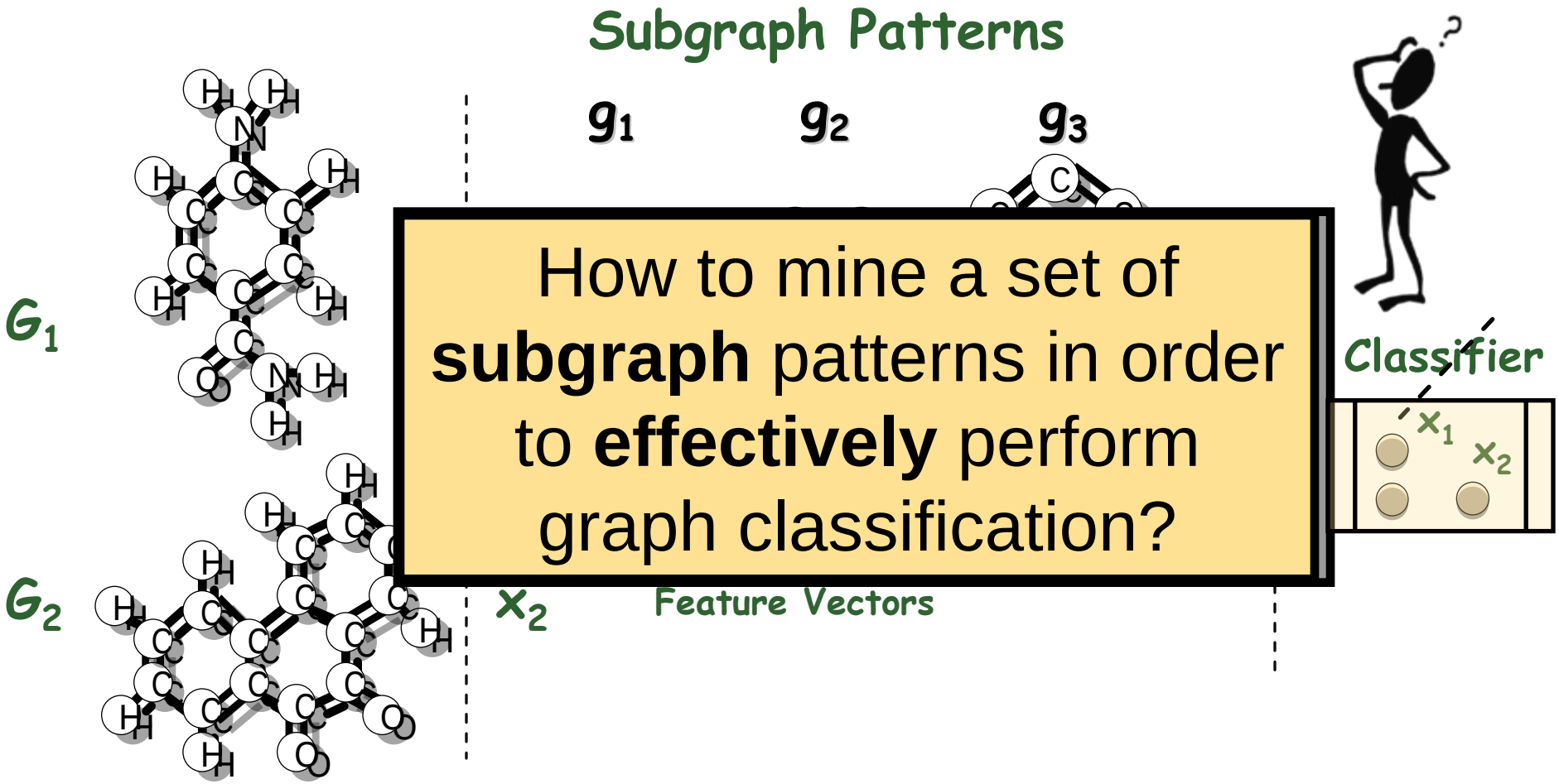


Program Flows



XML Docs

Subgraph-based Graph Classification



How to mine a set of **subgraph** patterns in order to **effectively** perform graph classification?

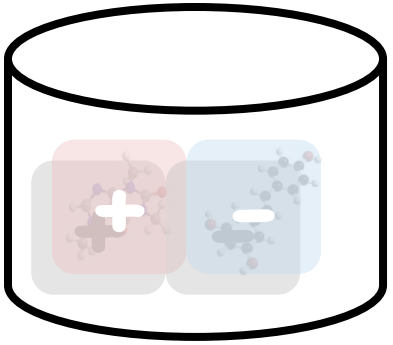
Graph Objects ⟶ Feature Vectors ⟶ Classifiers

Conventional Methods - Two Components

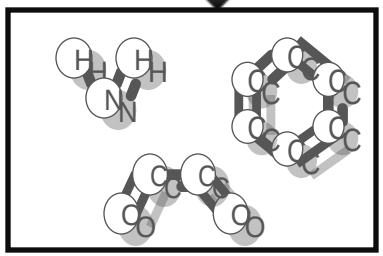
Two Components:

1. **Evaluation** (effective)
   whether a subgraph feature is relevant to graph classification?

2. **Search space pruning** (efficient)
   how to avoid enumerating all subgraph features?



Labeled Graphs

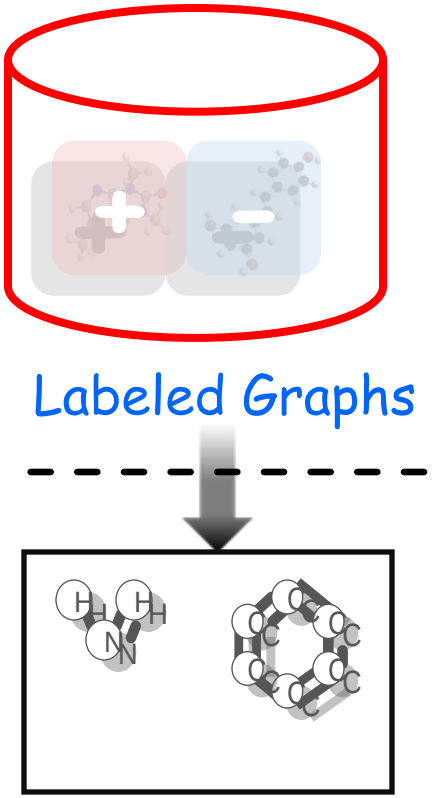


Discriminative Subgraphs

One Problem

- Supervised Settings
  - Require a **large** set of labeled training graphs

- However…

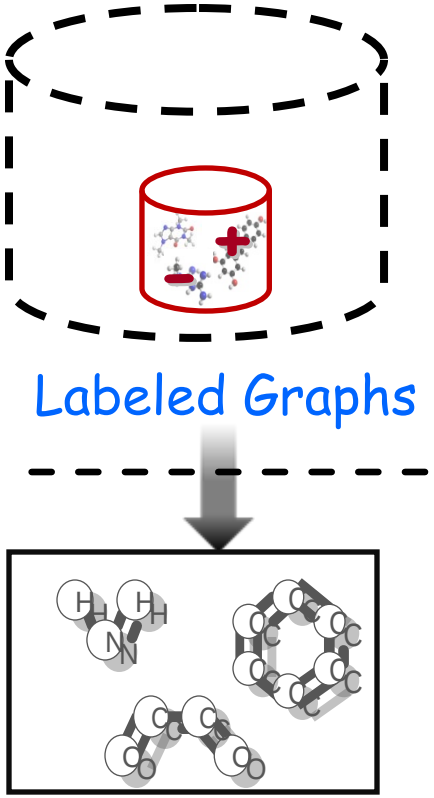


Lack of labels -> problems

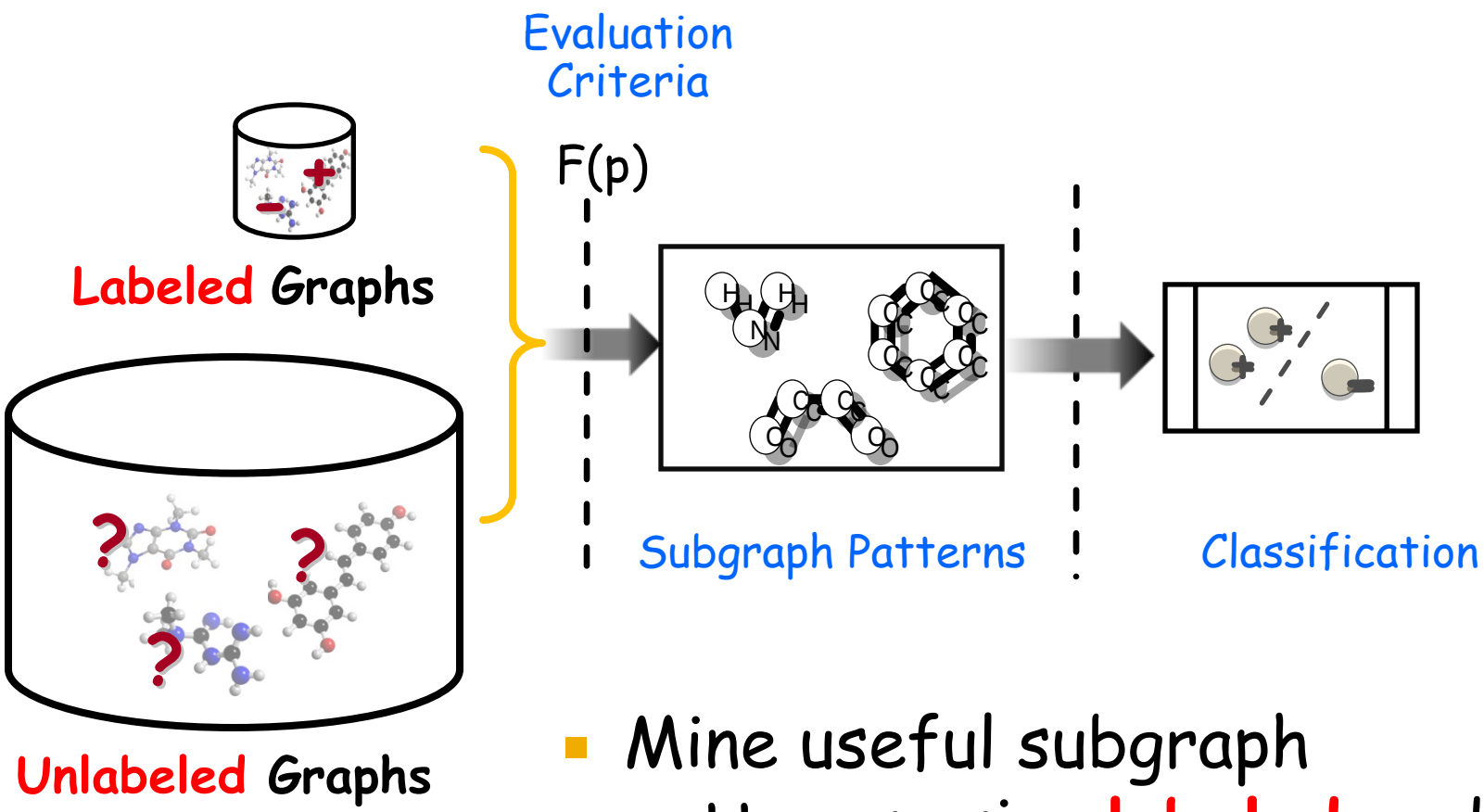
Supervised Methods:

1. Evaluation effective?
   require large amount of label information

2. Search space pruning efficient?
   pruning performances rely on large amount of label information



Semi-Supervised Feature Selection for Graph Classification



- Mine useful subgraph patterns using **labeled** and **unlabeled** graphs

Two Key Questions to Address

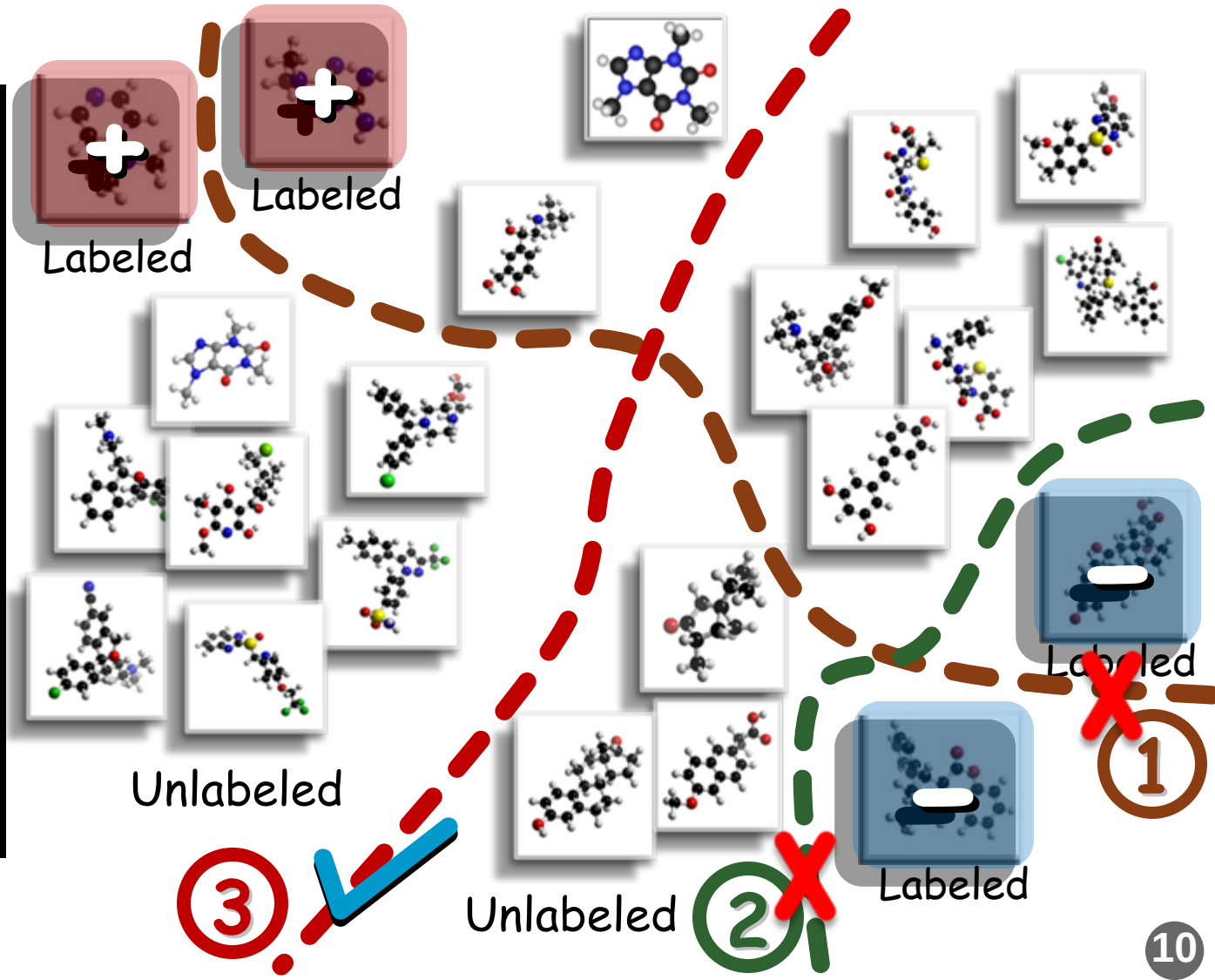
- **Evaluation**: How to evaluate a set of subgraph features with both labeled and unlabeled graphs? (effective)

- **Search Space Pruning**: How to prune the subgraph search space using both labeled and unlabeled graphs? (efficient)

What is a good feature?

- **Cannot-Link**
  Graphs in different classes should be far away
- **Must-Link**
  Graphs in the same class should be close

- **Separability**
  Unlabeled graphs are able to be separated from each other



Optimization

- **Cannot-Link**
  Graphs in different classes should be far away
- **Must-Link**
  Graphs in the same class should be close
- **Separability**
  Unlabeled graphs are able to be separated from each other

$$\mathcal{T}^* = \underset{\mathcal{T} \subseteq \mathcal{S}}{\operatorname{argmax}}\ J(\mathcal{T}) \qquad \text{s.t.} \quad |\mathcal{T}| \leq t$$

Evaluation Function:

$$\frac{\alpha}{2|\mathcal{C}|} \sum_{y_i y_j = -1} (D_{\mathcal{T}} \boldsymbol{x}_i - D_{\mathcal{T}} \boldsymbol{x}_j)^2$$

$$- \frac{\beta}{2|\mathcal{M}|} \sum_{y_i y_j = 1} (D_{\mathcal{T}} \boldsymbol{x}_i - D_{\mathcal{T}} \boldsymbol{x}_j)^2$$

$$+ \frac{1}{2|\mathcal{D}_u|^2} \sum_{G_i, G_j \in \mathcal{D}_u} (D_{\mathcal{T}} \boldsymbol{x}_i - D_{\mathcal{T}} \boldsymbol{x}_j)^2$$

Evaluation: gSemi Criterion

In matrix form:

$$J(\mathcal{T}) = \frac{1}{2}\sum_{i,j}(D_{\mathcal{T}}\boldsymbol{x}_i - D_{\mathcal{T}}\boldsymbol{x}_j)^2 W_{ij}$$

$$= \mathrm{tr}({D_{\mathcal{T}}}^\top X\,(D-W)\,X^\top D_{\mathcal{T}})$$

$$= \mathrm{tr}({D_{\mathcal{T}}}^\top X L X^\top D_{\mathcal{T}})$$

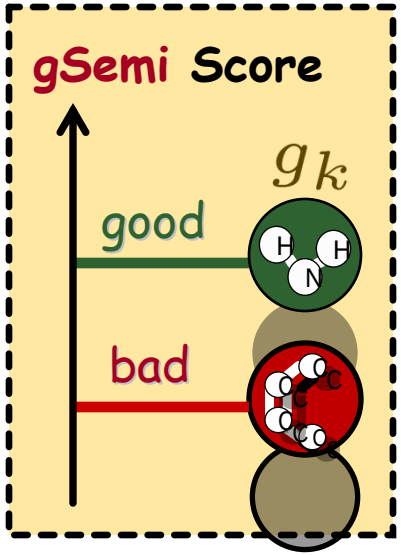
$$= \sum_{g_k \in \mathcal{T}} \underline{({\boldsymbol{f}_k}^\top L \boldsymbol{f}_k)}$$

(the sum over all selected features)

$$W_{ij} = \begin{cases} \frac{\alpha}{|\mathcal{C}|} & \text{if } y_i y_j = -1 \\ -\frac{\beta}{|\mathcal{M}|} & \text{if } y_i y_j = 1 \\ \frac{1}{|\mathcal{D}_u|^2} & \text{if } G_i, G_j \in \mathcal{D}_u \\ 0 & \text{otherwise} \end{cases}$$

$$L = D - W$$

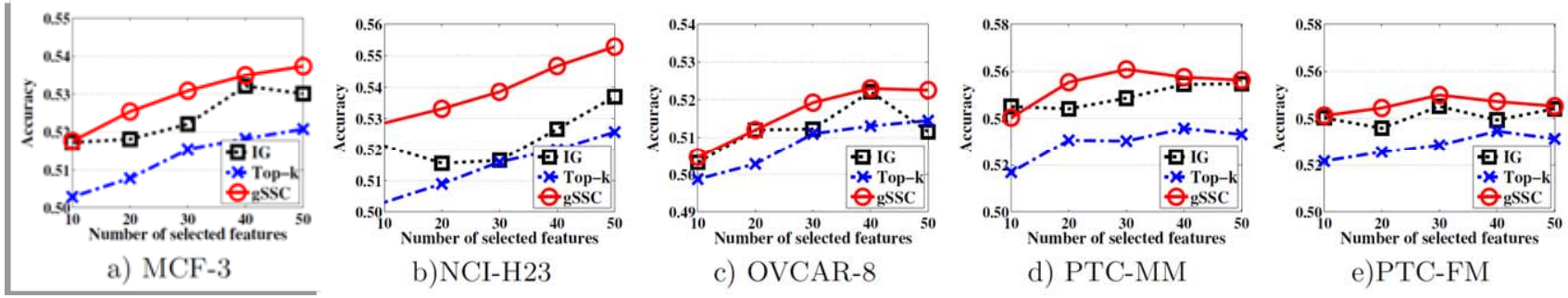
- **gSemi** Score: $\boxed{h(g_k, L) = {\boldsymbol{f}_k}^\top L \boldsymbol{f}_k}$



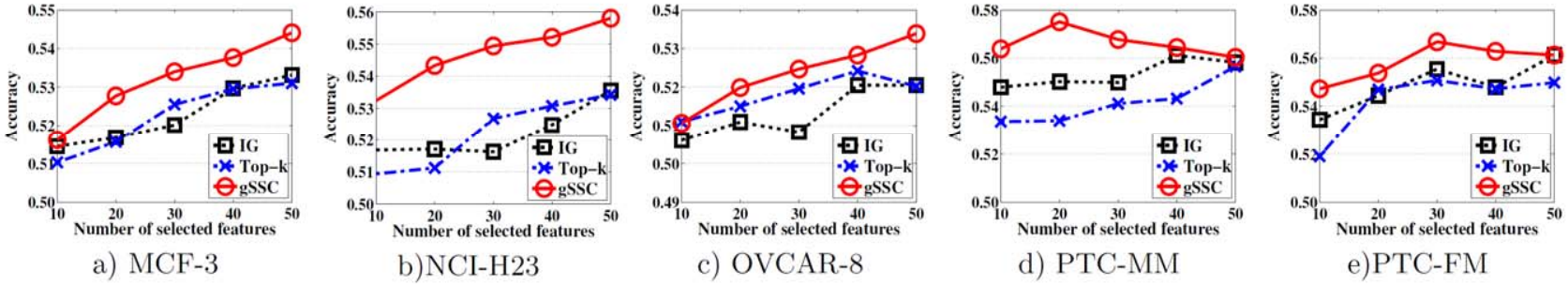
$\boldsymbol{f}_k \in \{0,1\}^n$ represents the *k*-th subgraph feature

Experiment Results

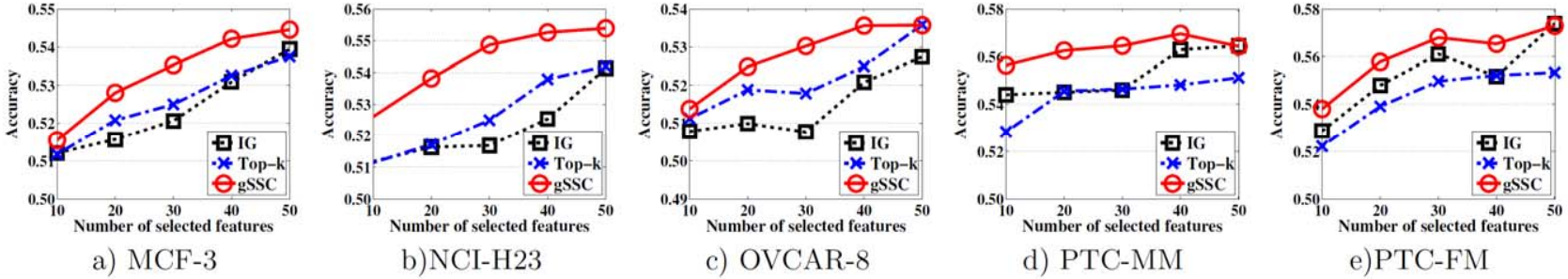
#labeled Graphs =30



#labeled Graphs =50

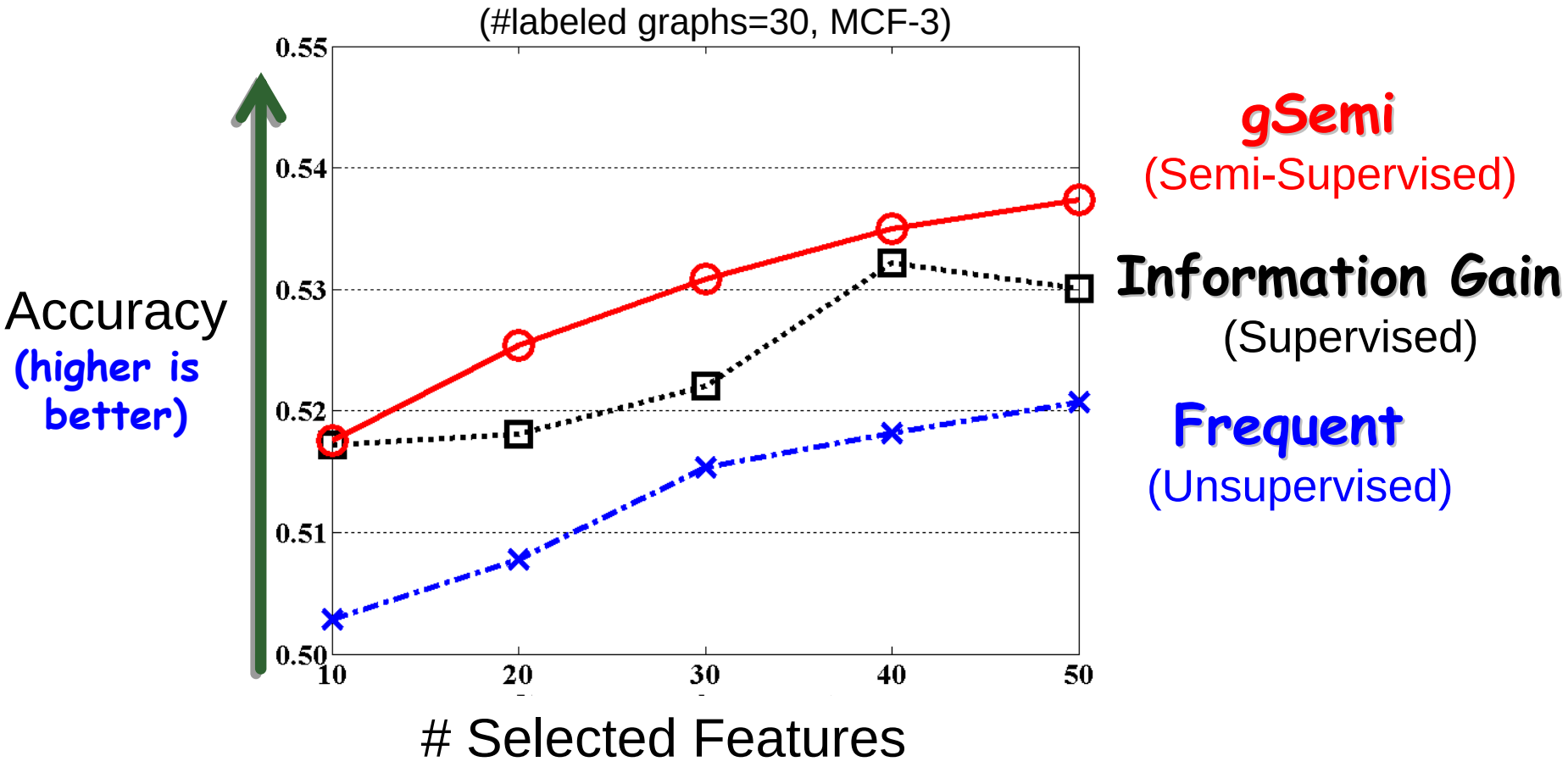


#labeled Graphs =70



MCF-3 dataset | NCI-H23 dataset | OVCAR-8 dataset | PTC-MM dataset | PTC-FM dataset

Experiment Results



- Our approach performed best at NCI and PTC datasets

Two Key Questions to Address

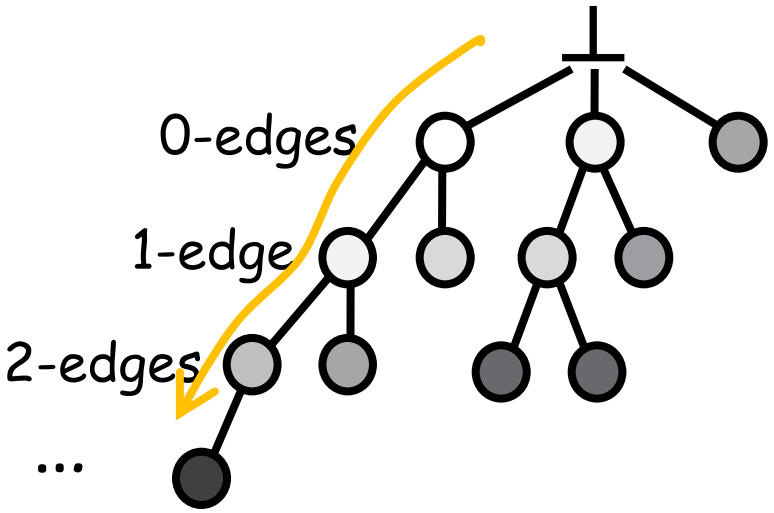
- How to evaluate a set of subgraph features with both labeled and unlabeled graphs? (effective)

- How to prune the subgraph search space using both labeled and unlabeled graphs? (efficient)

Finding a Needle in a Haystack

gSpan [Yan et. al ICDM'02]
An efficient algorithm to enumerate all frequent subgraph patterns
(frequency ≥ min_support)

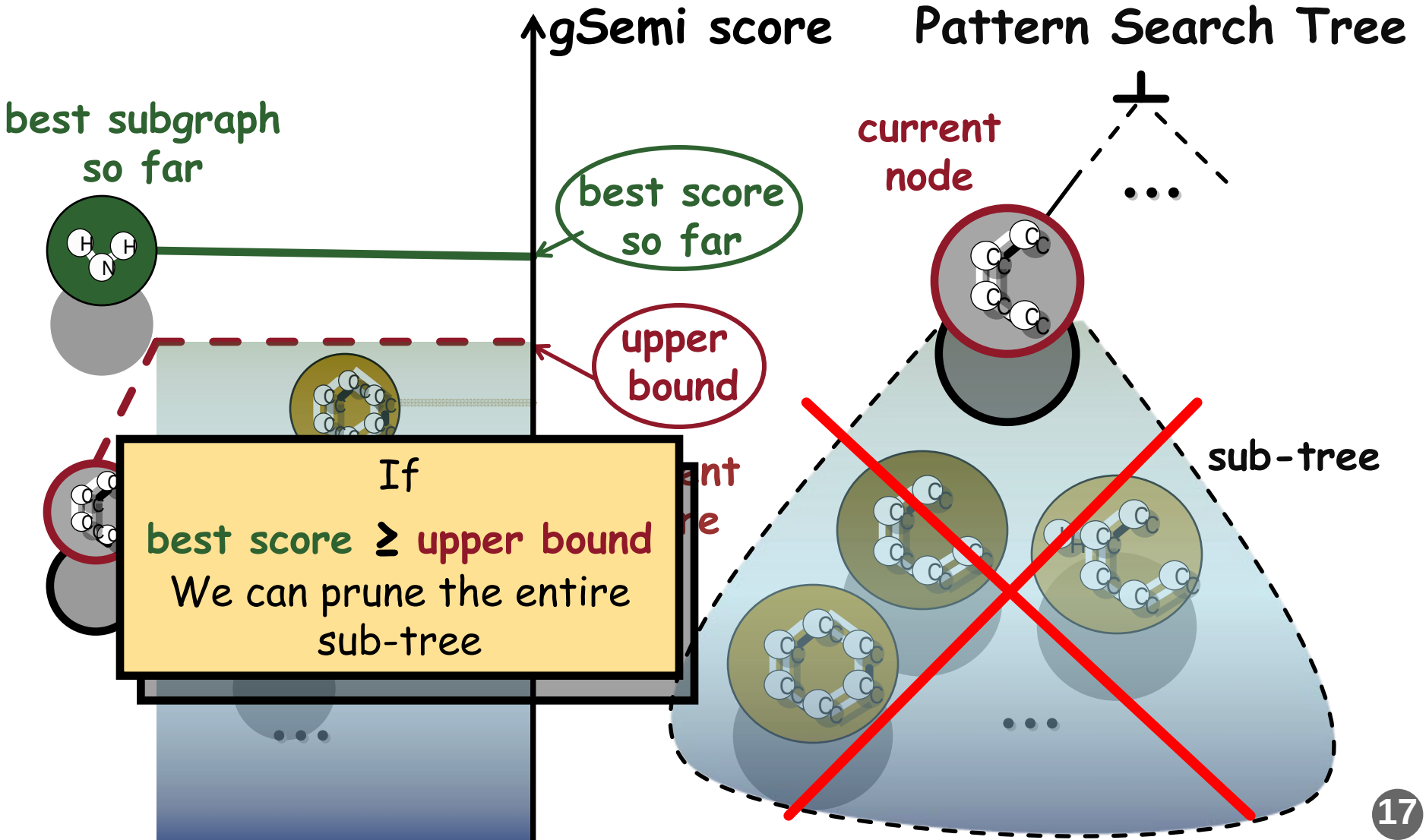
Pattern Search Tree



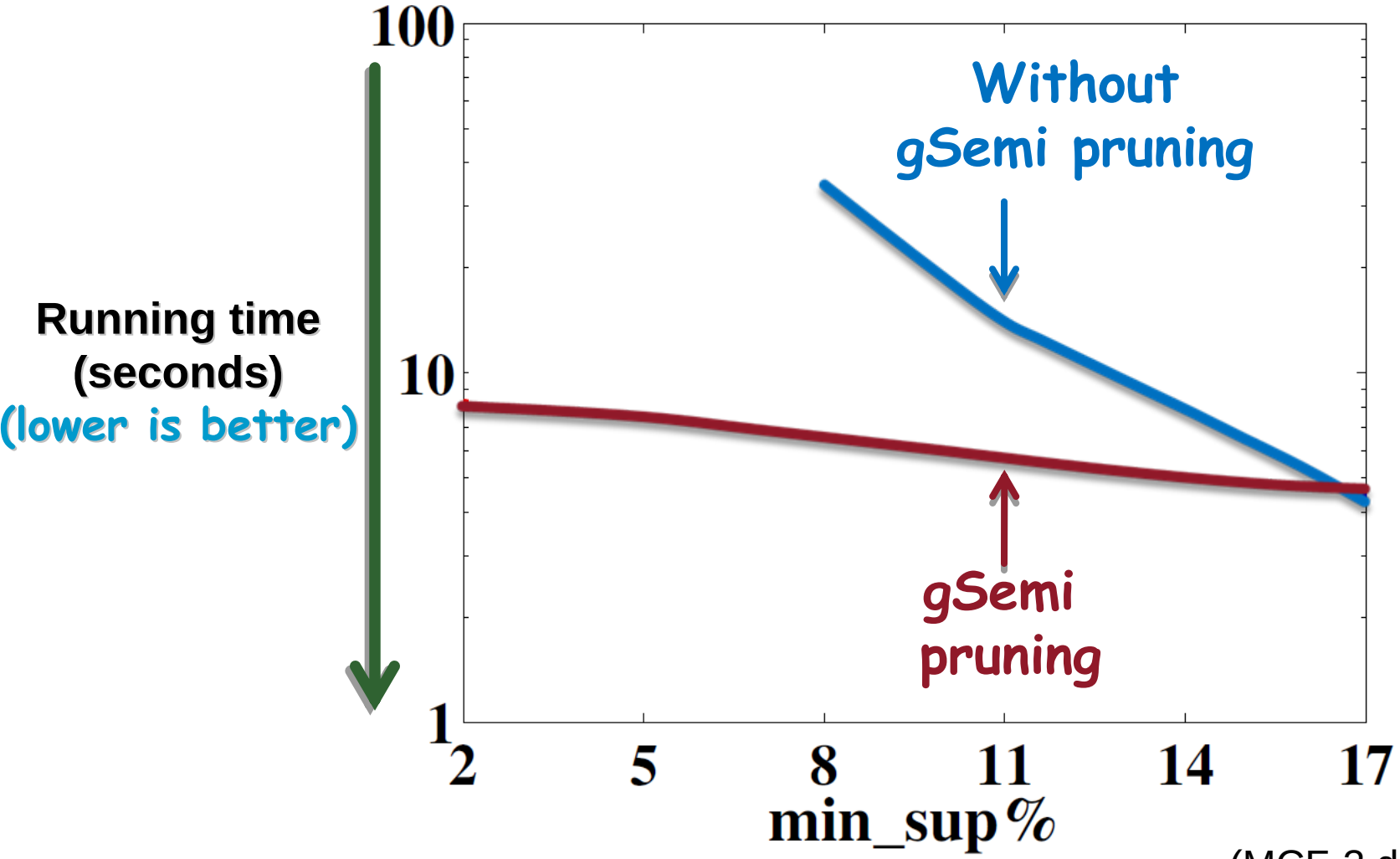
- Too many frequent subgraph patterns
- Find
  usefu

How to find the **Best node(s)** in this tree without searching all the nodes?
(Branch and Bound to prune the search space)

Pruning Principle



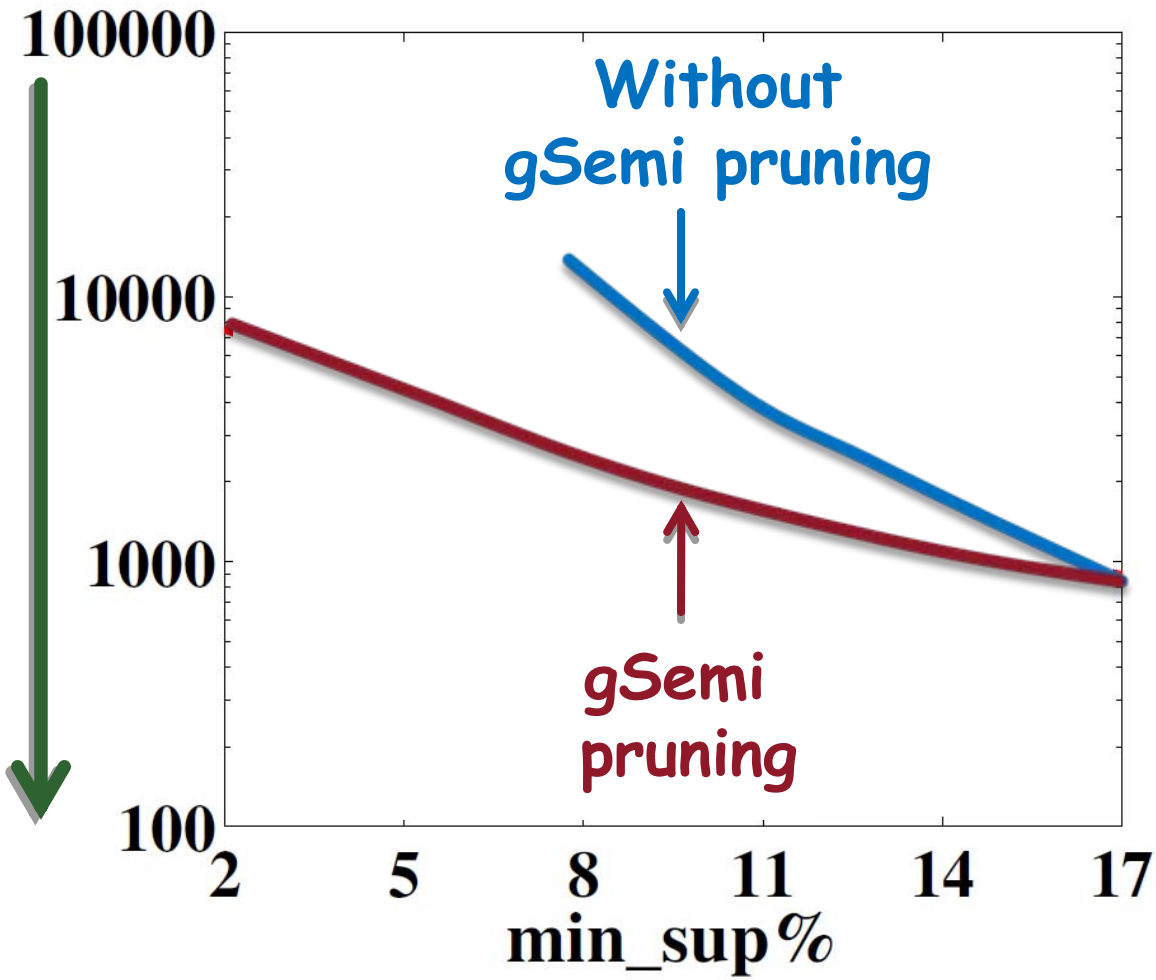
Pruning Results



(MCF-3 dataset)

Pruning Results

Subgraphs explored
(lower is better)



(MCF-3 dataset)

Conclusions

- **Semi-Supervised** Feature Selection for Graph Classification
  - **Evaluating** subgraph features using both labeled and unlabeled graphs (effective)
  - Branch&bound **pruning** the search space using labeled and unlabeled graphs (efficient)

Thank you!