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
Drug activity prediction problem

Given a set of chemical compounds labeled with activities

Predict the activities of testing molecules
Subgraph-based Graph Classification

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

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

- However...

  Labeling a graph is hard!
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
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
- **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

**Evaluation Function:**

\[ \mathcal{T}^* = \arg\max_{\mathcal{T} \subseteq S} J(\mathcal{T}) \quad \text{s.t.} \quad |\mathcal{T}| \leq t \]

\[
\frac{\alpha}{2|C|} \sum_{y_i y_j = -1} (D_T x_i - D_T x_j)^2 + \frac{\beta}{2|M|} \sum_{y_i y_j = 1} (D_T x_i - D_T x_j)^2 + \frac{1}{2|D_u|^2} \sum_{G_i, G_j \in D_u} (D_T x_i - D_T x_j)^2
\]
In matrix form:

\[
J(T) = \frac{1}{2} \sum_{i,j} (D_T x_i - D_T x_j)^2 W_{ij}
\]

\[
= \text{tr}(D_T^\top X (D - W) X^\top D_T)
\]

\[
= \text{tr}(D_T^\top X L X^\top D_T)
\]

\[
= \sum_{g_k \in T} (f_k^\top L f_k) \quad \text{(the sum over all selected features)}
\]

- **gSemi Score:**

\[
h(g_k, L) = f_k^\top L f_k
\]

\[
f_k \in \{0, 1\}^n \quad \text{represents the k-th subgraph feature}
\]
Experiment Results

- #labeled Graphs = 30
  - a) MCF-3
  - b) NCI-H23
  - c) OVCAR-8
  - d) PTC-MM
  - e) PTC-FM

- #labeled Graphs = 50
  - a) MCF-3
  - b) NCI-H23
  - c) OVCAR-8
  - d) PTC-MM
  - e) PTC-FM

- #labeled Graphs = 70
  - a) MCF-3
  - b) NCI-H23
  - c) OVCAR-8
  - d) PTC-MM
  - e) PTC-FM

Datasets:
- MCF-3 dataset
- NCI-H23 dataset
- OVCAR-8 dataset
- PTC-MM dataset
- PTC-FM dataset
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)

- Too many frequent subgraph patterns
- Find the most useful

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**

Best subgraph so far

If

\[ \text{best score} \geq \text{upper bound} \]

We can prune the entire sub-tree
**Pruning Results**

Without gSemi pruning

# Subgraphs explored
(lower is better)

**Graph:**
- Y-axis: # of Subgraphs explored (lower is better)
- X-axis: min_sup (%)
- Red line: Without gSemi pruning
- Blue line: gSemi pruning

(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!