Graph Matching
CSE 373
Data Structures

Questions to ask:
1. Are G1 and G2 isomorphic?
2. Is G1 isomorphic to a subgraph of G2?
3. How similar is G1 to G2?
4. How similar is G1 to the most similar subgraph of G2?

Isomorphism for Digraphs
G1 is isomorphic to G2 if there is a 1-1, onto mapping h: V1 → V2 such that
[(vi, vj) ∈ E1] iff [(h(vi), h(vj)) ∈ E2]

Subgraph Isomorphism for Digraphs
G1 is isomorphic to a subgraph of G2 if there is a 1-1 mapping h: V1 → V2 such that
[(vi, vj) ∈ E1] ⇒ [(h(vi), h(vj)) ∈ E2]

Similar Digraphs
Sometimes two graphs are close to isomorphic, but have a few “errors.”
Let h(1)=b, h(2)=e, h(3)=c, h(4)=a, h(5)=d.
[(1,2) ∈ E1, but (b,e) ∉ E2]
[(2,5) ∈ E1, but (a,e) ∈ E2]
The mapping h has 2 errors.

Error of a Mapping
Intuitively, the error of mapping h tells us:
- how many edges of G1 have no corresponding edge in G2, and
- how many edges of G2 have no corresponding edge in G1.
Let G1=(V1,E1) and G2=(V2,E2), and let h:V1 → V2 be a 1-1, onto mapping.
Forward error:
|{(vi,vj) ∈ E1 | (h(vi),h(vj)) ∉ E2}|
Backward error:
|{(vi,vj) ∈ E2 | (h(vi),h(vj)) ∉ E1}|
Total error:
|{(vi,vj) ∈ E1 | (h(vi),h(vj)) ∉ E2}|
= Error(h) + Error(h)
Variations of Relational Distance

1. normalized relational distance:
   Divide by the sum of the number of edges in E1 and those in E2.

2. undirected graphs:
   Just modify the definitions of EF and EB to accommodate.

3. one way mappings:
   h is 1-1, but need not be onto.
   Only the forward error EF is used.

4. labeled graphs:
   When nodes and edges can have labels, each node should be mapped to a node with the same label, and each edge should be mapped to an edge with the same label.

Graph Matching Algorithms

1. graph isomorphism
2. subgraph isomorphism
3. relational distance
4. attributed relational distance (uses labels)

Subgraph Isomorphism

Given model graph M = (VM, EM)
Data graph D = (VD, ED)
Find 1-1 mapping h: VM → VD
satisfying (vij) ∈ EM ⇒ ((h(vi),h(vj)) ∈ ED).

Method: Backtracking Tree Search

Subgraph Isomorphism in Digraphs

Branch-and-Bound Tree Search

Keep track of the least-error mapping.