CS345

Compact Skeletons
Compact Skeletons

- Assume tuples components are scattered over website
- We have a tagger that can tag all tuple components on website
  - Assume no noise for now
- Reconstruct relation
Compact Skeletons

Relation

Skeleton

Data Graph

Website
Welcome to Big Corp! Join our team.

Jobs are available in these departments:
- R&D
- Corporate

The following jobs are open:
- Job #12345
- Job #12346

Send resumes to:
1200 Jose Blvd, CA

Job Title: Programmer
Salary: 100K
Must know Java….
Welcome to Big Corp! Join our team.

Jobs are available in these departments:

- R&D
- Corporate

The following jobs are open:

- Job #12345
- Job #12346

Send resumes to:

1200 Jose Blvd, CA

Job Title: Programmer
Salary: 100K
Must know Java....
<table>
<thead>
<tr>
<th>T</th>
<th>S</th>
<th>D</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programmer</td>
<td>100K</td>
<td>R &amp; D</td>
<td>1200 Jose Blvd</td>
</tr>
<tr>
<td>CTO</td>
<td>150K</td>
<td>R &amp; D</td>
<td>1200 Jose Blvd</td>
</tr>
<tr>
<td>Admin Asst</td>
<td>60K</td>
<td>Corporate</td>
<td>400 7th Ave</td>
</tr>
<tr>
<td>CEO</td>
<td>(null)</td>
<td>Corporate</td>
<td>400 7th Ave</td>
</tr>
<tr>
<td>T</td>
<td>S</td>
<td>D</td>
<td>A</td>
</tr>
<tr>
<td>--------------</td>
<td>-------</td>
<td>---------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Programmer</td>
<td>100K</td>
<td>R &amp; D</td>
<td>1200 Jose Blvd</td>
</tr>
<tr>
<td>CTO</td>
<td>150K</td>
<td>R &amp; D</td>
<td>1200 Jose Blvd</td>
</tr>
<tr>
<td>Admin Asst</td>
<td>60K</td>
<td>Corporate</td>
<td>1200 Jose Blvd</td>
</tr>
<tr>
<td>CEO</td>
<td>(null)</td>
<td>Corporate</td>
<td>1200 Jose Blvd</td>
</tr>
</tbody>
</table>
Skeletons

- Labeled trees
- Transformation from data graphs to relations
Overlays

R & D

Programmer 100K

CTO 150K

1200 Jose Blvd

D

A

T S
Overlays

<table>
<thead>
<tr>
<th>T</th>
<th>S</th>
<th>D</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programmer</td>
<td>100K</td>
<td>R &amp; D</td>
<td>1200 Jose Blvd</td>
</tr>
</tbody>
</table>
Overlays

<table>
<thead>
<tr>
<th>T</th>
<th>S</th>
<th>D</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programmer</td>
<td>100K</td>
<td>R &amp;D</td>
<td>1200 Jose Blvd</td>
</tr>
<tr>
<td>CTO</td>
<td>150K</td>
<td>R &amp;D</td>
<td>1200 Jose Blvd</td>
</tr>
</tbody>
</table>
Overlays

R & D

1200 Jose Blvd

Programmer 100K  CTO  150K

D

T S A
Overlays

<table>
<thead>
<tr>
<th>T</th>
<th>S</th>
<th>D</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programmer</td>
<td>150K</td>
<td>R &amp; D</td>
<td>1200 Jose Blvd</td>
</tr>
</tbody>
</table>
Overlays

<table>
<thead>
<tr>
<th>T</th>
<th>S</th>
<th>D</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programmer</td>
<td>150K</td>
<td>R &amp; D</td>
<td>1200 Jose Blvd</td>
</tr>
<tr>
<td>CTO</td>
<td>100K</td>
<td>R &amp; D</td>
<td>1200 Jose Blvd</td>
</tr>
</tbody>
</table>
Inconsistent Overlays

R & D

Programmer 100K

CTO 150K

1200 Jose Blvd

D

T S

A
Inconsistent Overlays

R & D

Programmer  100K  CTO  150K

1200 Jose Blvd

D

A

T  S
Compact Skeletons

- A skeleton is **compact** if all overlays are consistent
- **Perfect** if each node and edge of data graph is covered by at least one overlay
- Given a data graph G, does G have a Perfect Compact Skeleton (PCS)?
  - Not always
  - But if it exists it is unique
PCS Algorithm

R & D

1200 Jose Blvd

Programmer 100K

CTO 150K
PCS Algorithm

Work bottom-up:
Compute node signatures
Place nodes in equivalence classes based on signature
Construct skeleton from equivalence classes
PCS Algorithm
Incomplete information

Corporate

Admin Asst

60K

CEO

400 7th Ave

D

TS

A
Incomplete information

<table>
<thead>
<tr>
<th>T</th>
<th>S</th>
<th>D</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admin Asst</td>
<td>60K</td>
<td>Corporate</td>
<td>400 7th Ave</td>
</tr>
</tbody>
</table>
Incomplete information

<table>
<thead>
<tr>
<th>T</th>
<th>S</th>
<th>D</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admin Asst</td>
<td>60K</td>
<td>Corporate</td>
<td>400 7th Ave</td>
</tr>
<tr>
<td>CEO</td>
<td></td>
<td>Corporate</td>
<td>400 7th Ave</td>
</tr>
</tbody>
</table>
Partial Compact Skeletons

- For data graphs with incomplete information, we allow partial overlays
  - Results in nulls in relation
- If we can use consistent partial overlays to cover every node and edge of the graph, we have a partially perfect compact skeleton (PPCS)
Tuple subsumption

- Tuple $t$ subsumes tuple $u$ if $t$ and $u$ agree on every component of $u$ that is not null
Real-life websites are *noisy*
- False positives e.g., MS = degree, state or Microsoft?
- Non-skeleton links e.g., featured products
Data graph for a retail website

For simplicity: assume all nodes have a label

Skeleton K1

C: Category
I: Item
P: Price
A: Availability
Coverage of a skeleton

Skeleton K1

C
I
P
A

i1
i2
i3
a4
i4
p1
a1
p3
p4
a2
a3

C1 C2 C3
P1 P3 P4 A3
Coverage of a skeleton

Skeleton K1
Coverage = 28
Coverage of a skeleton

Skeleton K1
Coverage = 28

Skeleton K2
Coverage = 12
Problem:
- Find skeleton K with optimal coverage, called the best-fit skeleton (BFS)

NP-complete
Greedy Heuristic for BFS
Greedy Heuristic for BFS

```
<table>
<thead>
<tr>
<th>Label</th>
<th>Parent</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>I</td>
<td>3</td>
</tr>
<tr>
<td>A</td>
<td>I</td>
<td>3</td>
</tr>
<tr>
<td>C</td>
<td>I</td>
<td>1</td>
</tr>
<tr>
<td>I</td>
<td>C</td>
<td>4</td>
</tr>
<tr>
<td>R</td>
<td>R</td>
<td>1</td>
</tr>
</tbody>
</table>
```

Diagram:
- Node R has a parent P and children I, A, C.
- Node P has a parent A and children I, A, P, A.
- Node A has a parent I and children I, C, A.
- Node C has a parent I and children I, R.
- Node I has a parent P and children R, C.
- Node P has a parent A and children I, A, P, A.
- Node A has a parent I and children I, C, A.
- Node C has a parent I and children R, C.
Greedy skeleton

<table>
<thead>
<tr>
<th>Label</th>
<th>Parent</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>D</td>
<td>C</td>
<td>4</td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>1</td>
</tr>
<tr>
<td>A</td>
<td>R</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>R</td>
<td>1</td>
</tr>
</tbody>
</table>
Greedy skeleton
Coverage = 9
Greedy skeleton
Coverage = 9

Optimal skeleton
Coverage = 15
Weighted Greedy Heuristic

- **Simple Greedy** heuristic uses parent counts
  - “Memory-less”

- **Weighted Greedy** heuristic takes into account past selections to improve simple greedy selection
  - Computes “benefit” of each decision at every stage
Weighted Greedy

Greedy skeleton
Coverage = 9
Weighted Greedy

\[ \text{benefit}(A \rightarrow C) = 4 \]

Greedy skeleton
Coverage = 9
Greedy skeleton
Coverage = 9

Weighted Greedy

\[ \text{benefit}(A \rightarrow C) = 4 \]
\[ \text{benefit}(B \rightarrow C) = 10 \]
Weighted Greedy

Greedy skeleton
Coverage = 9
Greedy skeleton
Coverage = 9

Weighted greedy skeleton
Coverage = 15