Weakly-Supervised Acquisition of Labeled Class Instances for Open-Domain Information Extraction

Partha Pratim Talukdar (UPenn)       Joseph Reisinger (UT Austin)
Marius Paşca (Google)       Deepak Ravichandran (Google)
Rahul Bhagat (USC)       Fernando Pereira (Google)

Work done at Google during Summer 2008.
Motivation

- (Class, Instance) pairs (e.g. (pain killer, aspirin)) can be useful in many applications e.g. web search.
Motivation

- (Class, Instance) pairs (e.g. \textit{(pain killer, aspirin)}) can be useful in many applications e.g. web search.

- Given an entity/instance, it is often desirable to know its type.
Motivation

- (Class, Instance) pairs (e.g. (pain killer, aspirin)) can be useful in many applications e.g. web search.

- Given an entity-instance, it is often desirable to know its type.

- A limited number of classes are not enough:
Motivation

- (Class, Instance) pairs (e.g. \textit{(pain killer, aspirin)}) can be useful in many applications e.g. web search.

- Given an entity/instance, it is often desirable to know its type.

- A limited number of classes are not enough:
  - Web search queries include \textit{active volcanoes} like \textit{Kilauea}, \textit{zoonotic diseases} like \textit{monkeypox} etc., demonstrating general user interest in them.
Motivation

- (Class, Instance) pairs (e.g. *(pain killer, aspirin)*) can be useful in many applications e.g. web search.

- Given an entity/instance, it is often desirable to know its type.

- A limited number of classes are not enough:
  - Web search queries include *active volcanoes* like *Kilauea*, *zoonotic diseases* like *monkeypox* etc., demonstrating general user interest in them.
  - Covering one class at a time (as in standard Named Entity Extraction) is resource intensive and not sufficient.
Motivation

- (Class, Instance) pairs (e.g. (pain killer, aspirin)) can be useful in many applications e.g. web search.

- Given an entity/instance, it is often desirable to know its type.

- A limited number of classes are not enough:
  - Web search queries include active volcanoes like Kilauea, zoonotic diseases like monkeypox etc., demonstrating general user interest in them.
  - Covering one class at a time (as in standard Named Entity Extraction) is resource intensive and not sufficient.
  - Need open domain extraction involving large number of classes and large number of instances.
Previous Work

- Named Entity Extraction: small number of classes, extensive supervision.
- (Van Durme and Pasca, AAAI 08): open domain extraction, high precision, low recall: precision drops fast with increasing recall.
- Our starting point: extractions from (Van Durme and Pasca, 2008).
Previous Work

- Named Entity Extraction: small number of classes, extensive supervision.
Previous Work

- Named Entity Extraction: small number of classes, extensive supervision.

- (Van Durme and Pasca, AAAI 08): open domain extraction, high precision, low recall: precision drops fast with increasing recall.
Previous Work

- Named Entity Extraction: small number of classes, extensive supervision.

- *(Van Durme and Pasca, AAAI 08)*: open domain extraction, high precision, low recall: precision drops fast with increasing recall.

- Our starting point: extractions from *(Van Durme and Pasca, 2008)*.

<table>
<thead>
<tr>
<th>Class</th>
<th>Size</th>
<th>Examples of Instances</th>
</tr>
</thead>
</table>
**OBJECTIVES**

Starting with such automatically extracted (class, instance) pairs:
OBJECTIVES

Starting with such automatically extracted (class, instance) pairs:

- Extract additional instances for existing classes.
OBJECTIVES

Starting with such automatically extracted (class, instance) pairs:

- Extract additional **instances** for existing **classes**.
- Identify additional **class labels** for existing **instances**.
Objectives

Starting with such automatically extracted (class, instance) pairs:

- Extract additional instances for existing classes.
- Identify additional class labels for existing instances.
- Handle initial pairs from diverse sources and methods.
OBJECTIVES

Starting with such automatically extracted (class, instance) pairs:

- Extract additional instances for existing classes.
- Identify additional class labels for existing instances.
- Handle initial pairs from diverse sources and methods.
- Require minimal human supervision.
OBJECTIVES

Starting with such automatically extracted (class, instance) pairs:

- Extract additional instances for existing classes.
- Identify additional class labels for existing instances.
- Handle initial pairs from diverse sources and methods.
- Require minimal human supervision.
- Do all these in a scalable manner.
OBJECTIVES

Starting with such automatically extracted (class, instance) pairs:

- Extract additional instances for existing classes.
- Identify additional class labels for existing instances.
- Handle initial pairs from diverse sources and methods.
- Require minimal human supervision.
- Do all these in a scalable manner.
- Increase coverage (recall) at comparable quality (precision)!
WHERE DO WE GET INSTANCES FROM?

- Extractions from unstructured text by (Van Durme and Pasca, AAAI 08).
- WebTables (Cafarella et al., VLDB 2008)
  - 154M HTML tables extracted from the web.
  - Rich source of instances, already segmented by webpage creators.
- Structured text.
WHERE DO WE GET Instances FROM?

• **A8**: Extractions from unstructured text by (Van Durme and Pasca, AAAI 08).
WHERE DO WE GET INSTANCES FROM?

- **A8**: Extractions from unstructured text by (Van Durme and Pasca, AAAI 08).
- **WebTables** (Cafarella et al., VLDB 2008)
WHERE DO WE GET INSTANCES FROM?

- **A8**: Extractions from unstructured text by (Van Durme and Pasca, AAAI 08).

- **WebTables** (Cafarella et al., VLDB 2008)
  - 154M HTML tables extracted from the web.
Where do we get instances from?

- **A8**: Extractions from unstructured text by (Van Durme and Pasca, AAAI 08).

- **WebTables** (Cafarella et al., VLDB 2008)
  - 154M HTML tables extracted from the web.
  - Rich source of instances, already segmented by webpage creators.
WHERE DO WE GET INSTANCES FROM?

- **A8**: Extractions from unstructured text by (Van Durme and Pasca, AAAI 08).

- **WebTables** (Cafarella et al., VLDB 2008)
  - 154M HTML tables extracted from the web.
  - Rich source of instances, already segmented by webpage creators.
  - Structured text.
Assigning class labels to **WebTable** instances

![Diagram showing WebTable and A8 instances]

**WebTable**

<table>
<thead>
<tr>
<th>Year</th>
<th>Artist</th>
<th>Albums</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Johnny Cash</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bob Dylan</td>
<td></td>
</tr>
</tbody>
</table>

**A8**

<table>
<thead>
<tr>
<th>musician</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bob Dylan</td>
</tr>
</tbody>
</table>

\[ \text{Score (musician, Johnny Cash)} = 0.87 \]
Putting together tuples from first phase extractors

A graph based representation is used: each tuple from A8 and WebTable is a weighted edge, with nodes representing classes and instances.

<table>
<thead>
<tr>
<th>Musician</th>
<th>Singer</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bob Dylan</td>
<td></td>
<td>0.95</td>
</tr>
<tr>
<td>Johnny Cash</td>
<td></td>
<td>0.87</td>
</tr>
<tr>
<td>Billy Joel</td>
<td></td>
<td>0.82</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.73</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.75</td>
</tr>
</tbody>
</table>
Putting together tuples from first phase extractors

- A graph based representation is used: each tuple from A8 and WebTable is a weighted edge, with nodes representing classes and instances.
**Initialization: Seed Labels Marked**

- **musician**
  - Bob Dylan: 0.95
  - Johnny Cash: 0.82
- **singer**
  - Billy Joel: 0.75

Seed Labels:
- **musician 1.0**
- **singer 1.0**
Label Propagation: Adsorption (Baluja et al., 2008)
Label Propagation: Adsorption (Baluja et al., 2008)

- After 1 iteration:

 Derived Labels

 Weakly-Supervised Acquisition of Labeled Class Instances for Open-Domain Information Extraction
Label Propagation: Adsorption (Baluja et al., 2008)

- After 2 iterations:
Label Propagation: Adsorption (Baluja et al., 2008)

- After 3 iterations:
Experimental Setup

- Dataset A8:
  - 924K (class, instance) pairs extracted from 100M web docs.
  - Extracted from *unstructured* text.
  - High precision, low recall.

- Dataset WT:
  - 74M unique additional pairs extracted from WebTables.
  - Source of new instances, extracted from *structured* text.
  - Low precision, high recall.

- Set of class labels in WT is the same as in A8.

- Graph constructed using A8 + WT had 1.4M nodes and 75M edges. This graph is used in all subsequent experiments.
Experimental Setup

• Dataset **A8**:
  • 924K (class, instance) pairs extracted from 100M web docs.
  • Extracted from *unstructured* text.
  • High precision, low recall.

• Dataset **WT**:
  • 74M unique additional pairs extracted from WebTables.
  • Source of new instances, extracted from *structured* text.
  • Low precision, high recall.
Experimental Setup

- Dataset **A8**:
  - 924K (class, instance) pairs extracted from 100M web docs.
  - Extracted from *unstructured* text.
  - High precision, low recall.

- Dataset **WT**:
  - 74M unique additional pairs extracted from WebTables.
  - Source of new instances, extracted from *structured* text.
  - Low precision, high recall.

- Set of class labels in WT is the same as in A8.
Experimental Setup

- **Dataset A8:**
  - 924K (class, instance) pairs extracted from 100M web docs.
  - Extracted from *unstructured* text.
  - High precision, low recall.

- **Dataset WT:**
  - 74M unique additional pairs extracted from WebTables.
  - Source of new instances, extracted from *structured* text.
  - Low precision, high recall.

- Set of class labels in WT is the same as in A8.

- Graph constructed using A8 + WT had 1.4M nodes and 75M edges. This graph is used in all subsequent experiments.
Experiments

• EXPT 1: Can we find new instances for fixed classes?
• EXPT 2: For a fixed set of instances, can we assign better class labels?
Experiments

- EXPT 1: Can we find new instances for fixed classes?
**Experiments**

- **EXPT 1**: Can we find new instances for fixed classes?
- **EXPT 2**: For a fixed set of instances, can we assign better class labels?
EXPT 1: Seed (Class, Instance) Pairs

<table>
<thead>
<tr>
<th>Seed Class</th>
<th>Seed Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>NFL Players</td>
<td>Ike Hilliard, Isaac Bruce, Torry Holt, Jon Kitna, Jamal Lewis</td>
</tr>
</tbody>
</table>

**Table:** Classes and seeds used to initialize Adsorption.
EXPT 1: Finding new instances for fixed classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Precision at 100 (non-A8 extractions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book Publishers</td>
<td>87.36</td>
</tr>
<tr>
<td>Federal Agencies</td>
<td>29.89</td>
</tr>
<tr>
<td>NFL Players</td>
<td>94.95</td>
</tr>
<tr>
<td>Scientific Journals</td>
<td>90.82</td>
</tr>
<tr>
<td>Mammal Species</td>
<td>84.27</td>
</tr>
</tbody>
</table>

Table: Precision of top 100 Adsorption extractions not present in A8.
EXPT 1: FINDING NEW INSTANCES FOR FIXED CLASSES

<table>
<thead>
<tr>
<th>Class</th>
<th>Precision at 100 (non-A8 extractions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book Publishers</td>
<td>87.36</td>
</tr>
<tr>
<td>Federal Agencies</td>
<td>29.89</td>
</tr>
<tr>
<td>NFL Players</td>
<td>94.95</td>
</tr>
<tr>
<td>Scientific Journals</td>
<td>90.82</td>
</tr>
<tr>
<td>Mammal Species</td>
<td>84.27</td>
</tr>
</tbody>
</table>

**Table:** Precision of top 100 Adsorption extractions **not** present in A8.

Coverage increased at precision level comparable to A8.
# New Extractions Found by Adsorption

<table>
<thead>
<tr>
<th>Seed Class</th>
<th>Top Ranked Instances Discovered by Adsorption</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NFL Players</strong></td>
<td>Tony Gonzales, Thabiti Davis, Taylor Stubblefield, Ron Dixon, Rodney Hannah</td>
</tr>
</tbody>
</table>
### SEMANTICALLY SIMILAR CLASS LABELS FOUND BY ADSORPTION: A BYPRODUCT

<table>
<thead>
<tr>
<th>Seed Class</th>
<th>Non-Seed Class Labels Discovered</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book Publishers</td>
<td>small presses, journal publishers, educational publishers, academic publishers, commercial publishers</td>
</tr>
<tr>
<td>NFL Players</td>
<td>sports figures, football greats, football players, backs, quarterbacks</td>
</tr>
<tr>
<td>Scientific Journals</td>
<td>prestigious journals, peer-reviewed journals, refereed journals, scholarly journals, academic journals</td>
</tr>
</tbody>
</table>

**Table:** Top class labels ranked by their similarity to a given seed class in Adsorption.
EXPT 2: Class assignment for fixed instances

Evaluation against WordNet Dataset (38 classes, 8910 instances)

- Adsorption (1 seed)
- Adsorption (5 seeds)
- Adsorption (10 seeds)
- Adsorption (25 seeds)
- Adsorption (25 seeds)

Adsorption is able to assign better class labels to more instances.
EXPT 2: Class assignment for fixed instances

Evaluation against WordNet Dataset (38 classes, 8910 instances)

Adsorption is able to assign better class labels to more instances.
CONCLUSION

• Demonstrated a scalable graph-based label propagation algorithm.

Future Work:
• Class label assignment in context.
• Scaling up further!
Conclusion

• Demonstrated a scalable graph-based label propagation algorithm.

• Improved coverage while maintaining adequate precision.
Conclusion

- Demonstrated a scalable graph-based label propagation algorithm.
- Improved coverage while maintaining adequate precision.
- Combined information from two different sources: unstructured and structured texts.

Future Work:
- Class label assignment in context.
- Scaling up further!
CONCLUSION

- Demonstrated a scalable graph-based label propagation algorithm.
- Improved coverage while maintaining adequate precision.
- Combined information from two different sources: unstructured and structured texts.
- Future Work:
CONCLUSION

- Demonstrated a scalable graph-based label propagation algorithm.
- Improved coverage while maintaining adequate precision.
- Combined information from two different sources: unstructured and structured texts.
- Future Work:
  - Class label assignment in context.
Conclusion

- Demonstrated a scalable graph-based label propagation algorithm.
- Improved coverage while maintaining adequate precision.
- Combined information from two different sources: unstructured and structured texts.
- Future Work:
  - Class label assignment in context.
  - Scaling up further!
Thank You!