Ontology Generation for Large Email Collections
Grace Hui Yang and Jamie Callan
Carnegie Mellon University

Roadmap

- Introduction
- Subtasks in Ontology Learning
- Supervised Hierarchical Clustering Framework
- Experimental Results
- User Study
Introduction

- Ontology is a data model that represents a set of concepts within a domain and the set of pair-wised relationships between those concepts.
  - Examples: WordNet, ODP
- Ontology Learning is the task to construct a well-defined ontology given
  - a text corpus or
  - a set of concept terms

Introduction

- In eRulemaking, there are large number of email comments sent to the agency every day
  - Ontology offers a nice way to summarize the important topics in the email comments
- In Information Retrieval, Natural Language Processing, there is need to know the relationships among the terms/phrases/concepts
  - Ontology offers relational associations between items
Subtasks in Ontology Learning

- Concept Extraction
- Synonym Detection
- Relationship Formulation by Clustering
- Cluster Labeling

Subtasks in Ontology Learning

- Concept Extraction
- Synonym Detection
- **Relationship Formulation by Clustering**
- Cluster Labeling
Concept Extraction

Noun N-gram Mining
- Each sentence is parsed by a part-of-speech (POS) tagger
- An n-gram generator then scans through to identify noun sequences
- Bigrams and trigrams are ranked by their frequencies of occurrences
- Longer Named Entities

Concept Filtering
- Web-based POS error detection
- Assumption:
  - Among the first 10 google snippets, a valid concept appears more than a threshold (4 in our case)
- Remove POS errors
  - protect/NN polar/NN bear/NN
- Remove Spelling errors
  - Pullution, polor bear

<table>
<thead>
<tr>
<th>Mercury Dataset</th>
<th>Polar Bear Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td># terms in 2gm vocab (unique): 462</td>
<td># terms in 2gm vocab (unique): 1389</td>
</tr>
<tr>
<td># tokens in 2g m vocab: 4510.</td>
<td># tokens in 2g m vocab: 3957.</td>
</tr>
<tr>
<td>power plants: 370</td>
<td>greenhouse gas: 248</td>
</tr>
<tr>
<td>mercury pollution: 260</td>
<td>gas pollution: 227</td>
</tr>
<tr>
<td>mercury emissions: 175</td>
<td>sea ice: 143</td>
</tr>
<tr>
<td>mercury levels: 110</td>
<td>ice habitat: 117</td>
</tr>
<tr>
<td>clean air: 70</td>
<td>endangered species: 115</td>
</tr>
<tr>
<td># terms in 3gm vocab (unique): 129</td>
<td># terms in 3gm vocab (unique): 361.</td>
</tr>
<tr>
<td># tokens in 3g m vocab: 900.</td>
<td># tokens in 3g m vocab: 731.</td>
</tr>
<tr>
<td>clean air act: 65</td>
<td>greenhouse gas pollution: 227</td>
</tr>
<tr>
<td>environmental protection agency: 35</td>
<td>sea ice habitat: 115</td>
</tr>
<tr>
<td>air quality rule: 30</td>
<td>endangered species act: 104</td>
</tr>
<tr>
<td>pollution control technology: 25</td>
<td>arctic sea ice: 62</td>
</tr>
<tr>
<td>interstate air quality: 25</td>
<td>greenhouse gas emissions: 19</td>
</tr>
</tbody>
</table>
Hierarchical Clustering

Different Strategies for Concepts at Different Abstraction Levels
- Concrete Concepts at the lower levels
  - Camp, basketball, car
- Abstract Concepts at the higher levels
  - Economy, math, study

Find Syntactic and Semantic Evidences for Concrete concepts
- concept candidates are organized into groups based on the 1st sense of the head noun in Wordnet
- one of their common head nouns will be selected as the parent concept for this group
  - pollution subsumes water pollution, air pollution.

Create a high accuracy concept forests at the lower level of the ontology

Bottom-Up Hierarchical Clustering
High Accuracy Ontology Fragments

- Two Problems in the previous step
  - Animal species and bear species are sisters
  - Different fragments need to be further grouped
- Solution: Use Wordnet Hypernyms to construct a higher level
  - Concepts at the leaf level are looked-up in Wordnet. If one is another's hypernym, the former is promoted as the parent of the latter's.
    - Species subsumes animal species subsumes bear species
    - Concepts in a Wordnet hypernym chain are connected
      - Their hypernym in Wordnet is used to label the group

Continue to be Bottom-Up
Different fragments are grouped

Ontology Fragments after Wordnet Refinement

- Problem
  - Still a forest
  - Many concepts at top level are not grouped
- In any clustering algorithm, we need a metric
  - Hard to know the metric to measure distance for those top level nodes
  - Learn it!

Continue to be Bottom-up
Supervised Hierarchical Clustering

- Learn for Whom?
  - Concepts at lower levels since they are highly accurate
  - User feedback
- Learn What?
  - A distance metric function
- After learning, then what?
  - Apply the distance metric function to high level to get distance scores for them
  - Then use whatever clustering algorithm to group them based on the distance scores

Training Data from Lower Levels

- A set of concepts $x^{(l)}$ on the $i$th level of the ontology hierarchy
- Distance matrix $y^{(l)}$
  - The Matrix entry which corresponding to concept $x^{(l)}_j$ and $x^{(l)}_k$ is $y^{(l)}_{jk} \in \{0,1\}$,
  - $y^{(l)}_{jk} = 0$, if $x^{(l)}_j$ and $x^{(l)}_k$ in the same group;
  - $y^{(l)}_{jk} = 1$, otherwise.
Training Data from Lower Levels

\[ y^{(i)} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \]

Distance metric represented as a Mahalanobis distance

\[ d(x_j^{(i)}, x_k^{(i)}) = \sqrt{\Phi(x_j^{(i)}, x_k^{(i)})^T A \Phi(x_j^{(i)}, x_k^{(i)})} \]

- \( \Phi(x_j^{(i)}, x_k^{(i)}) \) represents a set of pairwise underlying feature functions
- \( A \) is a positive semi-definite matrix, the parameter we need to learn

Parameter estimation by Minimize Squared Errors

\[
\min_A \sum_{i=1}^{||x^{(i)}||} \sum_{k=1}^{||x^{(i)}||} (y_{jk}^{(i)} - \sqrt{\Phi(x_j^{(i)}, x_k^{(i)})^T A \Phi(x_j^{(i)}, x_k^{(i)})})^2
\]
• Optimization can be done by
  ◦ Newton’s Method
  ◦ Interior-Point Method
  ◦ Any standard semi-definite programming (SDP) solvers
    • Sedumi, yalmip

---

**Table 1: Underlying Feature Functions**

The left half shows features for a single concept $x_j^0$, for the diagonal entries of the distance matrix. The right half shows features for two concepts $(x_j^0, x_k^0)$, for the off-diagonal entries of the distance matrix. The feature functions are represented in the form of $f^{-1}f$, where $f$ is a pairwise similarity function. “Normalized” means the data is normalized into the range of $[0,1]$ by divided by the maximum possible value. A verb predicate is in the form of verb(subject, object).

<table>
<thead>
<tr>
<th>Feature Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - normalized raw term frequency</td>
<td>1 - normalized raw frequency of term co-occurrences</td>
</tr>
<tr>
<td>1 - normalized log term frequency</td>
<td>1 - normalized log frequency of term co-occurrences</td>
</tr>
<tr>
<td>Constant (always 1)</td>
<td>1 - word overlap of Web definitions of $(x_j^0, x_k^0)$</td>
</tr>
<tr>
<td></td>
<td>1 - word overlap of children concepts of $(x_j^0, x_k^0)$</td>
</tr>
<tr>
<td></td>
<td>1 - object overlap of the same verb predicate when $x_j^0$ or $x_k^0$ as the subject</td>
</tr>
<tr>
<td></td>
<td>1 - subject overlap of the same verb predicate when $x_j^0$ or $x_k^0$ as the object</td>
</tr>
</tbody>
</table>
• We have learned A!
• For any pair of concepts at higher level \((x^{(i+1)}_l, x^{(i+1)}_m)\)
• The corresponding entry in the distance matrix \(y^{(i+1)}\) is

\[
y^{(i+1)}_{lm} = \sqrt{\Phi(x^{(i+1)}_l, x^{(i+1)}_m)^T A \Phi(x^{(i+1)}_l, x^{(i+1)}_m)}
\]

**Generate Distance Scores for Higher Level**

• A flat clustering at each level
• Use one of the concepts as the cluster center
• Estimate the number of clusters by Gap statistics [Tibshirani et al. 2000]

**K-medoids Clustering for Higher Level Concepts**
Supervised Hierarchical Clustering

- Repeat the learning process from each level
  - Learn parameter matrix A from lower level
  - Generate distance scores for higher level
  - Clustering higher level
  - Move one level up
    - Previous testing data now becomes training data!
    - Always trust groupings in the lower level since they are relatively more accurate

Cluster Labeling

- Problem:
  - Concepts are grouped together, but nameless
- Need to find a good name representing the meaning of entire group
- Solution:
  - A web-based approach
  - Send a query formed by concatenating the child concepts to Google
  - Parse top 10 snippets
  - Most frequent word is selected to be the parent of this group
Experimental Results

- Datasets

<table>
<thead>
<tr>
<th>Table 1: Statistics of Polar Bear Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statistic</td>
</tr>
<tr>
<td># documents</td>
</tr>
<tr>
<td># sentences</td>
</tr>
<tr>
<td># words</td>
</tr>
<tr>
<td># unique terms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2: Statistics of Mercury Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statistic</td>
</tr>
<tr>
<td># documents</td>
</tr>
<tr>
<td># sentences</td>
</tr>
<tr>
<td># words</td>
</tr>
<tr>
<td># unique terms</td>
</tr>
</tbody>
</table>

Component-based Performance Analysis

<table>
<thead>
<tr>
<th>Table 3: Component-based Precision, Recall and F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component(s)</td>
</tr>
<tr>
<td>add 2gm</td>
</tr>
<tr>
<td>add 3gm</td>
</tr>
<tr>
<td>add WN hypernym</td>
</tr>
<tr>
<td>add POS corrector</td>
</tr>
<tr>
<td>add Supervised clustering</td>
</tr>
<tr>
<td>overall</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 4: Component-based Precision, Recall and F1 (by turning off each component)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component(s)</td>
</tr>
<tr>
<td>w/o 2gm</td>
</tr>
<tr>
<td>w/o 3gm</td>
</tr>
<tr>
<td>w/o WN hypernym</td>
</tr>
<tr>
<td>w/o POS corrector</td>
</tr>
<tr>
<td>w/o Supervised clustering</td>
</tr>
<tr>
<td>overall</td>
</tr>
</tbody>
</table>
Component-based Performance Analysis

Error Analysis
Software

- Combine many techniques into a unified framework
  - pattern-based (concept mining)
  - knowledge-based (use of Wordnet)
  - Web-based (concept filtering and cluster naming)
  - Machine learning (supervised clustering)
- Effectively combine the strengths of automatic systems and human knowledge via relevance feedback
- Worked on harder datasets which do not contain broad, diverse concepts, hence require higher accuracy

Contributions
What is Next?

- Is bottom-up the best way to do?
  - Maybe not
  - Incremental clustering saves most efforts
- We have used different technologies for concepts at different levels, how to formally generalize it?
  - Model concept abstractness explicitly
- We have tested on domain-specific corpora, how about corpora for more general purpose?
  - Can we reconstruct Wordnet or ODP?