Feature Selection for Automatic Taxonomy Induction

Hui Yang and Jamie Callan

Language Technologies Institute, School of Computer Science, Carnegie Mellon University

{huiyang, callan}@cs.cmu.edu

Introduction

- Automatic taxonomy induction constructs taxonomies by exploiting simple lexical-syntactic patterns [1] and/or contextual information [2] in combination with a lexical resource such as WordNet.
- Most prior work on automatic taxonomy induction uses a single rule or feature function at all levels of abstraction.
- We study the effect of using *different features* for different types of relations and terms at different abstraction levels.

The Features

Input: Two terms (c_x, c_y)

Output: A numeric score $h(c_x, c_y) \in \mathbb{R}$, or $h(c_x, c_y) \in \mathbb{R}^n$.

Lexical-Syntactic Patterns

Hypernym Patterns	*Sibling Patterns*
NP_x (,)?and/or other NP_y	NP_x and/or NP_y
such NP_y as NP_x	*Part-of Patterns*
NP_y (,)? such as NP_x	NP_x of NP_y
NP_y (,)? including NP_x	NP_y's NP_x
NP_y (,)? especially NP_x	NP_y has/had/have NP_x
NP_y like NP_x	NP_y is made (up)? of NP_x
NP_y called NP_x	NP_y comprises NP_x
NP_x is a/an NP_y	NP_y consists of NP_x
NP_x , a/an NP_y	

Co-occurrence Features

$$pmi(c_x, c_y) = \log \frac{Count(c_x, c_y)}{Count(c_x)Count(c_y)},$$

Count(.) = # of sentences containing the term(s);

or # of documents containing the term(s);

or *n* as in "Results 1-10 of about *n* for …" in Google.

Contextual Features

- Global Context KL-Divergence =
  KL-Divergence(1000 Google Documents for C_x ,
  1000 Google Documents for C_y)
- Local Context KL-Divergence =
  KL-Divergence(Left two and Right two words for C_x ,
  Left two and Right two words for C_y)

Syntactic Dependency Features

- Minipar Syntactic Distance = Average length of syntactic paths in parse trees for sentences containing the terms
- Modifier Overlap = # of overlaps between modifiers of the terms; e.g., *red* apple, *red* pear.
- Object Overlap = # of overlaps between objects of the terms when the terms are subjects
  e.g., A dog eats *apple*; A cat eats *apple.*
- Subject Overlap = # of overlaps between subjects of the terms when the terms are objects
  e.g., A *dog* eats apple; A *dog* eats pear.
- Verb Overlap = # of overlaps between verbs of the terms when the terms are subjects/objects
  e.g., A dog *eats* apple; A cat *eats* pear.

Miscellaneous Features

- Word Length Difference
- Definition Overlap =
  # of non-stopword overlaps between definitions of two terms.

Our Related Work

H. Yang and J. Callan. 2009. "A Metric-based Framework for Automatic Taxonomy Induction." *ACL'09*.

Multi-criteria optimization based on minimization of taxonomy structures and modeling of term abstractness. The distance between 2 terms (c_x,c_y) is modeled by a weighted combination of features.

$$d(x, y) = \sum_j w_j h_j(c_x, c_y)$$

Experimental Methodology

- **Ground Truth:** 50 hypernym taxonomies from WordNet
  50 hypernym taxonomies from ODP
  50 meronym taxonomies from WordNet
- **Auxiliary Data:** 1000 Google documents per term/term pair
  100 Wikipedia documents per term
- **Metrics:** F1-measure (averaged by LOOCV).

Features vs. Relations

Table 2: F1-measure for Features vs. Relations: WordNet.

Feature Type	*is-a*	*sibling*	*part-of*	***Benefited Relations***
Contextual	0.21	**0.42**	0.12	*sibling*
Co-occur.	**0.48**	**0.41**	**0.28**	*All*
Patterns	**0.46**	**0.41**	**0.30**	*All*
Syntactic	0.22	**0.36**	0.12	*sibling*
Misc.	0.14	0.17	0.12	
All	***0.82***	***0.79***	***0.61***	***All***
Best Features	Co-occur., patterns	Contextual, co-occur., patterns	Co-occur., patterns	

Features vs. Abstractness

L_i represents depth of a concept in the taxonomy. Level L_2 contains children of the root. Level L_3 contains grandchildren of the root.

Table 3: F1-measure for Features vs. Abstractness: WordNet/*is-a*.

Feature Type	L_2	L_3	L_4	L_5	L_6
Contextual	0.29	0.31	**0.35**	**0.36**	**0.36**
Co-occurrence	**0.47**	**0.56**	**0.45**	**0.41**	**0.41**
Patterns	**0.47**	**0.44**	**0.42**	**0.39**	**0.40**
Syntactic	0.31	0.28	**0.36**	**0.38**	**0.39**
Misc.	0.14	0.14	0.14	0.14	0.14

Table 4: F1-measure for Features vs. Abstractness: ODP/*is-a*.

Feature Type	L_2	L_3	L_4	L_5	L_6
Contextual	**0.30**	**0.30**	**0.33**	**0.29**	**0.29**
Co-occurrence	**0.34**	**0.36**	**0.34**	**0.31**	**0.31**
Patterns	0.23	0.25	**0.30**	**0.28**	**0.28**
Syntactic	0.18	0.18	0.23	**0.27**	**0.27**
Misc.	0.14	0.14	0.14	0.13	0.13

Conclusions

- The first study using different features for different types of relations and terms at different abstraction levels.
- Co-occurrence and lexico-syntactic patterns are good features for common relations, including is-a, part-of, and sibling relation.
- Contextual and syntactic features are only good for the sibling relation.
- Contextual, co-occurrence, lexical-syntactic patterns, and syntactic features work well for concrete terms.
- Only co-occurrence works well for abstract terms.

References

[1] M. Hearst, 1992. Automatic acquisition of hyponyms from large text corpora. COLING'92.

[2] P. Pantel and D Lin, 2002. Discovering word senses from text. SIGKDD'02.