Microsoft Research

Large-Scale, Open Domain Semantic Mining: Basic Techniques

Shuming Shi
Microsoft Research Asia
August 2011

Outline

- Overview
- Semantic class mining
- Semantic hierarchy construction
- Mining attribute names and values
- General relation extraction
- Demo
- Summary

Microsoft Research

Semantic Mining: Introduction

- (Semi-)Automatically obtaining semantic knowledge
  - Semantic knowledge: Entities, concepts, relations
    - Similarity(significantly, substantially, 0.9)
    - Synonym(China, People’s Republic of China)
    - IsA(pear, fruit)
    - Peer(Beijing, Shanghai, Guangzhou…)
    - InClass(Beijing, C1)
    - Attribute(Capital, China, Beijing)
    - BornIn(Barack Obama, 1961)
    - DefeatedIn(Dallas Mavericks, Miami Heat, 2011 NBA Finals)
  - Data sources:
    - Web documents, query logs, web search results
    - Existing dictionaries & knowledge-bases

Semantic Mining: Introduction (cont.)

- Motivation
  - Build “smarter” computer systems with the semantic knowledge-base
  - Better fulfill the information needs of end users
    - Better web search
    - Better QA
    - Better machine translation
    - …

Outline

- Overview
- **Semantic class mining**
- Semantic hierarchy construction
- Mining attribute names and values
- General relation extraction
- Demo
- Summary

Semantic Class Mining

- Goal
  - Discover peer terms (or coordinate terms)
  - Sample: {C++, C#, Java, PHP, Perl, …}

- Main techniques
  - First-order co-occurrences
    - Standard co-occurrences
    - Patterns: Special first-order co-occurrences
  - Second-order co-occurrences
    - Distributional similarity

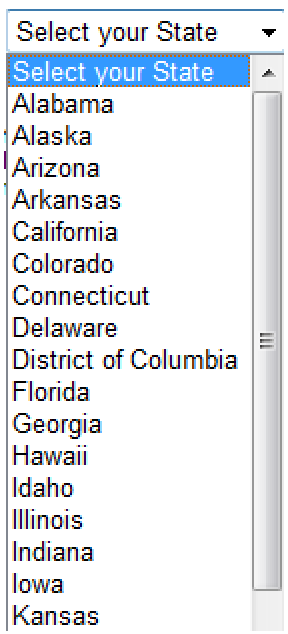
Pattern-Based (PB)

Hours may vary on holidays, such as Easter, Thanksgiving and Christmas.

Pattern: (such as | including) **T** {,**T**}* (and|,|.)

{Easter, Thanksgiving, Christmas}

Exploit
first-order
Co-occurrences



Pattern:

<select>
<option> **T** … <option> **T**
</select>

{Alabama, Alaska, Arizona…}

Microsoft Research

PB Implementation

- RASC mining
  - Employ predefined patterns to extract Raw Semantic Classes (RASCs)

<table>
<tr><th>Type</th><th>Pattern</th></tr>
<tr><td rowspan="3">Lexical</td><td>T {, T}* {,} (and|or) {other} T</td></tr>
<tr><td>(such as | including) T {,T}* (and|,|.)</td></tr>
<tr><td>T, T, T {,T}*</td></tr>
<tr><td rowspan="5">Tag</td><td> T … T </td></tr>
<tr><td> T … T </td></tr>
<tr><td><select> <option> T …<option> T </select></td></tr>
<tr><td><table> <tr> <td> T </td> … <td> T </td> </tr> ... </table></td></tr>
<tr><td>Other Html-tag repeat patterns</td></tr>
</table>

PB Implementation

- Compute Term Similarity
  - Based on the RASCs containing both terms

$$Sim(a,b) = \sum_{i=1}^{m} \log(1 + \sum_{j=1}^{k_i} w(P(C_{i,j})))$$

(Zhang et al., ACL'09)

$$Sim^*(a,b) = Sim(a,b) \cdot \sqrt{IDF(a) \cdot IDF(b)}$$

$$IDF(a) = \log(1 + N/N(a))$$

Microsoft Research

Distributional Similarity (DS)

- Distributional hypothesis (Harris, 1985): Terms occurring in analogous (lexical or syntactic) contexts tend to be similar

- Contexts shared by *Easter* and *Christmas*
  - the date _ is celebrated
  - | _ is a religious festival
  - history of the _ festival
  - …
- Contexts shared by *significantly* and *dr*
  - is _ improved by
  - unlikely to _ alter the
  - can _ increase health risks
  - …

Exploit **second-order** Co-occurrences

DS Implementation

- Define context
  - Syntactic context, lexical context…
- Represent each term by a feature vector
  - Feature: A context in which the term appears
  - Feature value: “Weight” of the context w.r.t. the term
- Compute term similarity
  - Term similarity = similarity between corresponding feature vectors

DS Implementation

Contexts	Text window (window size: 2, 4)
	Syntactic
Feature value	PMI
Similarity measure	Cosine, Jaccard

DS approaches implemented in the study

Pointwise mutual information:

$$f_{w,c} = \mathrm{PMI}_{w,c} = \log \frac{F(w,c) \cdot F(*,*)}{F(w,*) \cdot F(*,c)}$$

$$Cosine(\vec{x},\vec{y}) = \frac{\sum_i x_i y_i}{\sqrt{\sum_i {x_i}^2} \cdot \sqrt{\sum_i {y_i}^2}}$$

$$Jaccard(\vec{x},\vec{y}) = \frac{\sum_i \min(x_i,y_i)}{\sum_i x_i + \sum_i y_i - \sum_i \min(x_i,y_i)}$$

S
C

Se
Co

Microsoft Research

Compare DS and PB with Set Expansion

(Shi et al., COLING'2010)

- Set Expansion: Problem statement
  - Given a list of seed terms in a semantic class
    - $Q = \{s_1, s_2, \ldots, s_k\}$ (e.g. Q={*Lent*, *Epiphany*})
  - To find other members of the
    - E.g., {*Advent*, *Easter*, *Chri*

$R(t,s_i)$: the rank of term t among the neighbors of s_i

- Set Expansion with a similarity graph G
  - Select the terms most similar to the seeds

$$f(t,Q) = \sum_{i=1}^{k} w_i \cdot Sim(t, s_i)$$

$$Sim(t, s_i) = \frac{1}{\log(\lambda + r(t, s_i))}$$

Compare and Combine PB & DS (cont.)

- Corpus: ClueWeb (500 million English pages)
- Five term categories: **proper nouns, common nouns, verbs, adjectives, adverbs**
- Key observations: PB performs better for proper nouns; DS has better performance for other term categories

Samples (Query: significantly)

D:\Projects\NeedleSeek\bin\TermGraphClie...

1	significantly	187.419
2	and	54.8759
3	slightly	23.4412
4	but	21.8044
5	moderately	21.7083
6	english	20.4911
7	seriously	20.4479
8	yiddish	20.2321
9	hebrew	19.7871
10	too	19.6313
11	kigezi	18.4164
12	bunyoro	17.8679
13	specifically	17.8268
14	also	17.4519
15	mbale	17.3605
16	especially	17.2895
17	rich americans	17.1207
18	surely	16.7604
19	sharply	16.5638
20	it	15.6475

PB results

D:\Projects\NeedleSeek\bin\TermGraphClie...

1	significantly	0.121576
2	substantially	0.0162357
3	considerably	0.0154982
4	greatly	0.0138213
5	dramatically	0.013429
6	slightly	0.0100923
7	drastically	0.0089119
8	somewhat	0.00800886
9	vastly	0.0074269
10	steadily	0.00731532
11	severely	0.00688791
12	importantly	0.00640118
13	remarkably	0.0061907
14	inherently	0.00606039
15	comparatively	0.00604854
16	strongly	0.0060448
17	consistently	0.00603508
18	sufficiently	0.00602135
19	rapidly	0.00601235
20	gradually	0.00590148

DS results

Microsoft Research

Samples (Query: Apple)

D:\Projects\NeedleSeek\bin\TermGraphClie...

1	apple	1741.13
2	microsoft	639.909
3	ibm	617.503
4	sony	613.111
5	dell	601.909
6	hp	597.473
7	toshiba	546.464
8	orange	537.578
9	samsung	528.885
10	compaq	490.275
11	canon	476.098
12	cherry	472.247
13	pear	470.911
14	panasonic	467.727
15	peach	460.441
16	pineapple	444.158
17	intel	434.583
18	acer	433.825
19	lemon	424.788
20	strawberry	423.942

PB results

D:\Projects\NeedleSeek\bin\TermGraphClie...

1	apple	0.0808821
2	microsoft	0.00336825
3	the government	0.00237455
4	the company	0.00223547
5	google	0.00212872
6	sony	0.00193015
7	ibm	0.00185744
8	obama	0.00163117
9	dell	0.00161188
10	nintendo	0.00135578
11	bush	0.00129623
12	hp	0.00127199
13	banana	0.00126387
14	intel	0.00124417
15	someone	0.00123563
16	mccain	0.00119849
17	congress	0.00114992
18	israel	0.00111276
19	the team	0.00110751
20	adobe	0.00108731

DS results

Explain by Frequency

- Normalized frequency (F_{norm}) of term t

$$\frac{\text{Frequency in the RASCs}}{\text{Frequency in the sentences of the original documents}}$$

- Mean normalized frequency (MNF) of a query set S

$$MNF(S) = \frac{\sum_{t \in S} F_{norm}(t)}{|S|}$$

Seed Categories	Terms	MNF
Proper nouns	40	0.2333
Common nouns	40	0.0716
Verbs	40	0.0099
Adjectives	40	0.0126
Adverbs	40	0.0053

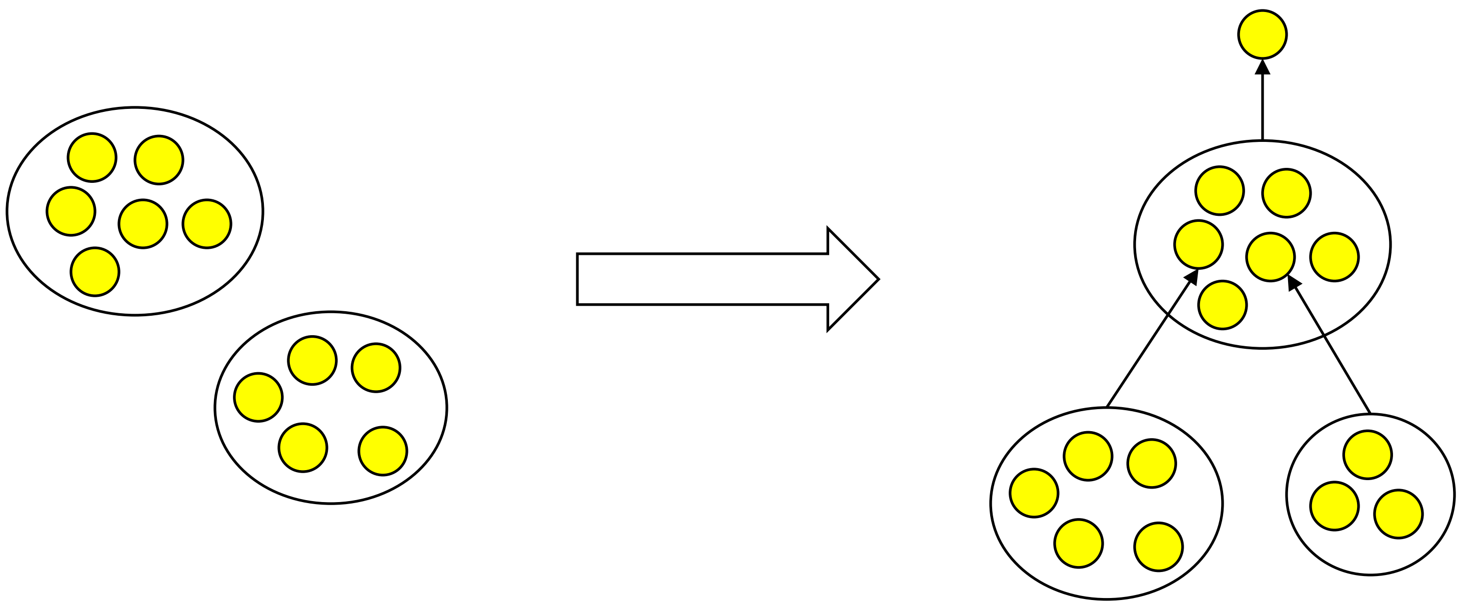
Related Papers

- Harris, 1985 (in The Philosophy of Linguistics)
  Distributional Structure
- Pantel & Lin, SIGKDD'2002
  Discovering Word Senses from Text
- Etzioni et al., WWW'2004
  Web-Scale Information Extraction in KnowItAll
- Wang & Cohen, ICDM'2008
  Itera-tive Set Expansion of Named Entities Using the Web
- Pantel, EMNLP'2009
  Web-Scale Distributional Similarity and Entity Set Expansion
- Agirre et al., NAACL'2009
  A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches
- Shi et al., COLING'2010
  Corpus-based Semantic Class Mining: Distributional vs. Pattern-Based Approaches

Outline

- Overview
- Semantic class mining
- ➢ **Semantic hierarchy construction**
- Mining attribute names and values
- General relation extraction
- Demo
- Summary

Semantic Hierarchy Construction



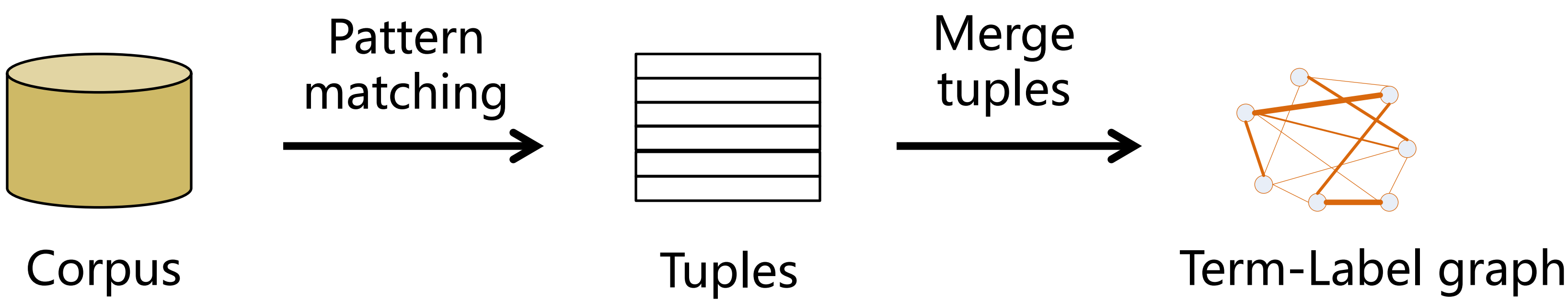
Semantic hierarchy construction

- Major subtasks
  - Assign category labels (hypernyms) to terms
    - Beijing → city, capital…
    - Apple → company, fruit…
    - Red → color…
    - Canon EOS 400D → digital camera, product…
  - Assign category labels to semantic classes
    - {Beijing, Shanghai, Dalian…} → cities, Chinese cities…
    - {Microsoft, IBM, Apple…} → companies, manufacturers…
  - Build the hierarchy

Microsoft Research

Subtask: Term→Label

- Approach: Pattern matching + counting



Tuple:

<term, label, pattern, source, weight>

<pear, fruit, P1, S1, 1.0>

<pear, shape, P2, S2, 1.0>

<pear, fruit, P3, S3, 1.0>

<New York, city, P1, S4, 1.0>

<New York, office, P2, S6, 1.0>

<New York, state, P4, S7, 1.0>

… …

Subtask: Term→Label (cont.)

- Pattern matching
  - Manually designed or automatically generated patterns
  - Text patterns or HTML tables

Label	Label	Label
Term	Term	Term
Term	Term	Term
...	...	...

Type	Pattern
Hearst-I	NP_L {,} (such as) {NP,}* {and|or} NP
Hearst-II	NP_L {,} (include(s) | including) {NP,}* {and|or} NP
Hearst-III	NP_L {,} (e.g.|e.g) {NP,}* {and|or} NP
IsA-I	NP (is|are|was|were|being) (a|an) NP_L
IsA-II	NP (is|are|was|were|being) {the, those} NP_L
IsA-III	NP (is|are|was|were|being) {another, any} NP_L

  - Output: <term, label, pattern, source, weight> tuples
  - Challenges
    - Boundary detection: term boundary, label boundary
    - Label selection

Subtask: Term→Label (cont.)

- Merge tuples
  - For each term *T* and label *L*, compute $w(T, L)$
- Methods
  - Simple counting
    - Count the number of <T, L, P, S, W> tuples for each (T, L) pair
    - Or TF-IDF
  - Nonlinear evidence fusion (Shi et al., ACL'2011)

$$Score(T, L) = \left(\sum_{i=1}^{K} \sqrt[p]{m_i} \right) \cdot \mathrm{IDF}(L)$$

$\boldsymbol{m_i}$: #tuples for pattern i
$\boldsymbol{x_{i,j}}$: Gain value given the j'th tuple for pattern i

Subtask: Class→Label

- Input
  - Class *C*: {orange, apple, pear, banana…}
- Output
  - Label list for *C*: fruit, tree, flavor…
- Method: Voting
  - orange: color, flavor, client, network, fruit, county, tree…
  - apple: company, brand, fruit, manufacturer, client, tree…
  - pear: fruit, tree, shape, flavor, juice, cut, wood…
  - banana: fruit, crop, flavor, tree, food, plant, vegetable…

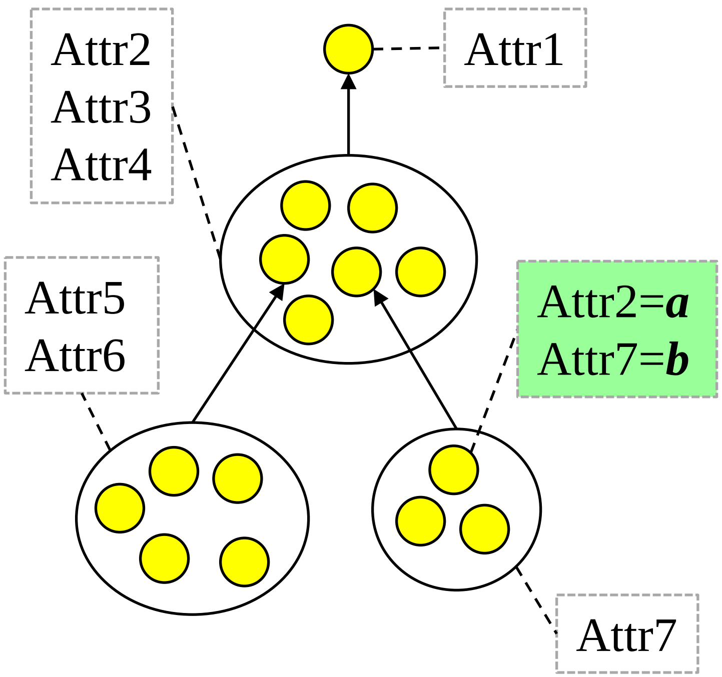
Related Papers

- Hearst, COLING'1992
  Automatic Acquisition of Hyponyms from Large Text Corpora
- Pantel & Ravichandran, HLT-NAACL'2004
  Automatically Labeling Semantic Classes
- Snow et al., COLING-ACL'2006
  Semantic Taxonomy Induction from Heterogenous Evidence
- Banko et al., IJCAI'2007
  Open Information Extraction from the Web
- Cafarella et al., VLDB'2008
  WebTables: Exploring the Power of Tables on the Web
- Durme & Pasca, AAAI'2008
  Finding cars, Goddesses and Enzymes: Parametrizable Acquisition of Labeled Instances for Open-Domain Information Extraction
- Zhang et al., ACL'2011
  Nonlinear Evidence Fusion and Propagation for Hyponymy Relation Mining

Outline

- Overview
- Semantic class mining
- Semantic hierarchy construction
- **Mining attribute names and values**
- General relation extraction
- Demo
- Summary

Semantic Attributes

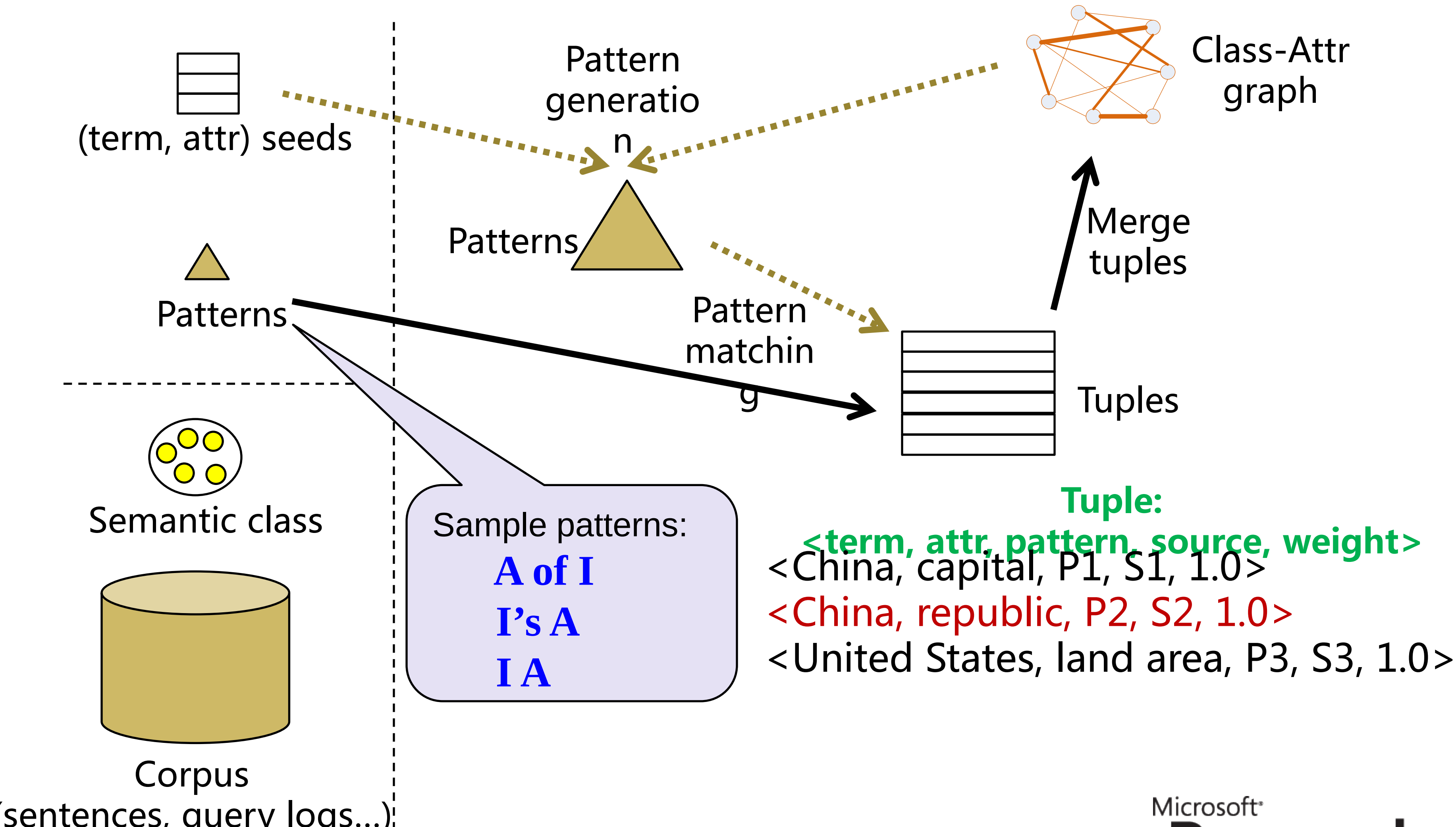


(city, population)
(country, flag)
(country, capital)
(company, CEO)

(China, capital, Beijing)
(Microsoft, CEO, Steve Ballmer)
(Barack Obama, Birth year, 1961)

Microsoft Research

Attribute Name Extraction from Unstructured Text



Attribute Name Extraction from Unstructured Text

- Major papers:
  - Pasca, WWW’2007
    Organizing and Searching the World Wide Web of Facts Step Two: Harnessing the Wisdom of the Crowds
  - Durme et al., COLING’2008
    Class-Driven Attribute Extraction
  - Pasca et al., CIKM’2007
    The Role of Documents vs. Queries in Extracting Class Attributes from Text
  - Bellare et al., NIPS’2007
    Lightly-Supervised Attribute Extraction
  - Reisinger & Pasca, 2009
    Low-Cost Supervision for Multiple-Source Attribute Extraction
  - Tokunaga et al., IJCNLP’2005 (Japanese data)
    Automatic Discovery of Attribute Words from Web Documents
  - …

Attribute Name & Value Extraction

- From Unstructured Text
  - Similar with extracting attribute names from un
  - C
  - C
- Fron
- Fron

Mountain Peak	Continent	Height
Mount Everest	Asia	8,850 m
Aconcagua	South America	6,959 m
Mount McKinley (Denali)	North America	6,194 m
Kilimanjaro	Africa	5,895 m
Mount Elbrus	Europe	5,642 m
Vinson Massif	Antarctica	4,897 m
Carstensz Pyramid Mount Kosciuszko (The highest point on the Australian landmass)	Australia - Oceania	4,884 m 2,228 m

http://woodlands-junior.kent.sch.uk/Homework/mountains/tallest.htm

Kinect for Xbox 360

KINECT for XBOX 360

Kinect sensor device

Product family	Xbox
Generation	Seventh generation era
Units Sold	10 million (as of March 9, 2011)[1]
Release date	NA November 4, 2010[2] EU November 10, 2010[3] CO November 14, 2010[4] AUS November 18, 2010[5] JP November 20, 2010[6]
Platform	Xbox 360, Microsoft Windows
Connectivity	USB 2.0 (type-A for original model; proprietary for Xbox 360 S)
Resolution	640×480 pixels @ 30 Hz (RGB camera) 640×480 pixels @ 30 Hz (IR depth-finding camera)[7]
Predecessor	Xbox Live Vision

Outline

- Overview
- Semantic class mining
- Semantic hierarchy construction
- Mining attribute names and values
- **General relation extraction**
- Demo
- Summary

General Relations

- Relations: Facts involving entities
  - [PER Susan Dumais] works for [ORG Microsoft Research], which is headquartered in [LOC Redmond, WA]
  - DefeatedIn(Dallas Mavericks, Miami Heat, 2011 NBA Finals)
- Relations vs. Events
  - Vague boundary
- History
  - Introduced in MUC-7 (1997) , extended by ACE, continued by KBP
  - Gain popularity in molecular biology, recent works including extracting protein-protein interaction

Type	Subtype
ART (artifact)	User-Owner-Inventor-Manufacturer
GEN-AFF (Gen-affiliation)	Citizen-Resident-Religion-Ethnicity, Org-Location
METONYMY*	*none*
ORG-AFF (Org-affiliation)	Employment, Founder, Ownership, Student-Alum, Sports-Affiliation, Investor-Shareholder, Membership
PART-WHOLE (part-whole)	Artifact, Geographical, Subsidiary
PER-SOC* (person-social)	Business, Family, Lasting-Personal
PHYS* (physical)	Located, Near

ACE' 05 relation types

Microsoft Research

Supervised Learning

- Treat relation mining as a classification problem
  - Use relational and non-relational mentions as positive and negative data, respectively
- Solve it with supervised Machine learning algorithms
  - Popular choices include SVM, MaxEnt, KNN
- Key: data representation
  - Feature based methods
  - Kernel based methods
- Evaluate metrics: Precision, Recall, F1 on relation mention level

Features

- List of common features (Kambhatla 2004)
  - **Words:** Words of both the entity mentions and all the words in between.
  - **Entity Type:** Entity type of both the mentions.
  - **Mention Level:** Mention level of both the mentions.
  - **Overlap:** Number of words separating the two mentions, number of other mentions in between, flags indicating whether the two mentions are in the same noun phrase, verb phrase or prepositional phrase.
  - **Dependency:** Words and PoS and chunk labels of the words on which the mentions are dependent in the dependency tree
  - **Parse Tree:** Path of non-terminals (removing duplicates) connecting the two mentions in the parse tree, and the path annotated with head words.
- Other features (Zhou et al. 2005)
  - **Based phrase chunking** chunk labels and chunk heads in between
  - **Semantic resources** (country list, etc)

Kernel based Methods

- Kernel (X, Y) defines similarity between X and Y
- X and Y can be
  - Vectors of features (as in previous slides)
  - Objects (string sequence, Parse trees)
- Kernel-based methods
  - Don't require extensive feature engineering
  - Maybe computational expensive
- Multiple Kernels can also be used in combination with a composite kernel (Zhao and Grishman, 2005)

Subsequence Kernel (Bunescu and Mooney, 2005)

- Implicit features are sequences of words anchored at the two entity names
  - $s =$ a word sequence

<e$_1$> … **bought** … <e$_2$> … **billion** … **deal**.

  - $x =$ an example sentence, containing s as a subsequence

$$\text{Google } \underbrace{\text{has}}_{g_1=1} \text{ bought } \underbrace{\text{video-sharing website}}_{g_2=3} \text{ YouTube } \underbrace{\text{in a controversial \$1.6}}_{g_3=4} \text{ billion } \underbrace{\vphantom{x}}_{g_4=0} \text{deal.}$$

  - $\varphi_s(x) =$ the value of feature s in example x

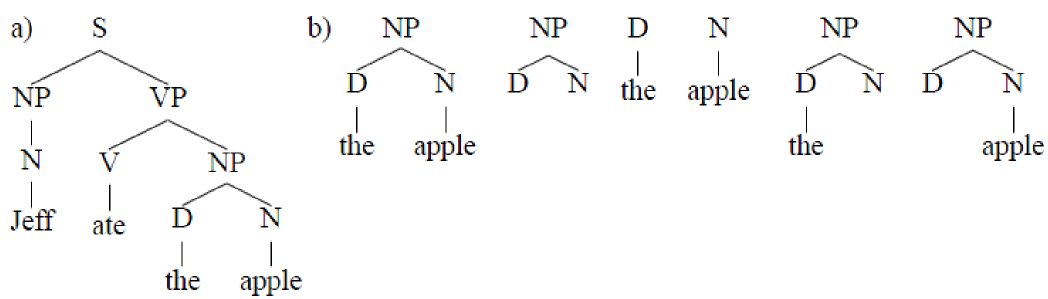
$$\varphi_s(x) = \lambda^{\sum g_i} = \lambda^{gap(s,x)} = \lambda^{1+3+4+0}$$

- $K(x_1,x_2) = \varphi(x_1)\varphi(x_2) =$ the number of common "anchored" subsequences between x_1 and x_2, weighted by their total gap

Microsoft Research

Tree Kernel for RDC

- Convolution kernels for NLP (Collins and Duffy. 2001)
  - K(T1, T2) defined over trees T1 and T2
  - Measured as number of overlapping fragments.

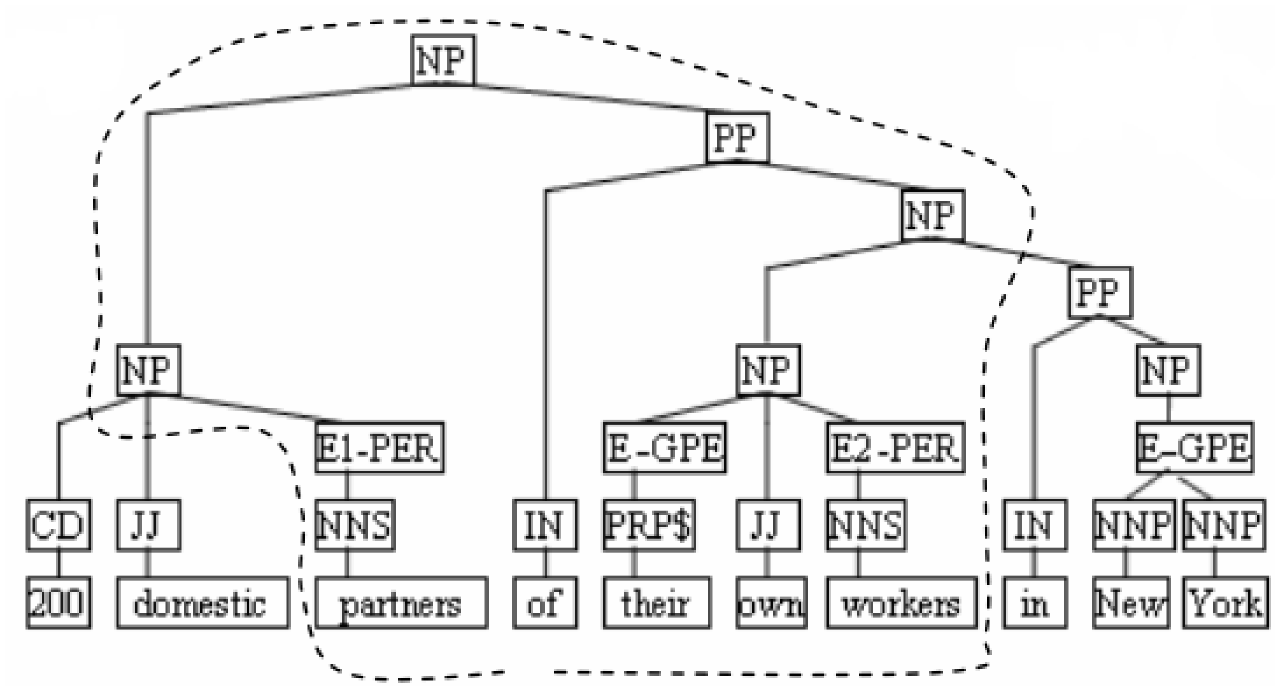


An example parse tree(a) and its sub-trees(b)

- Parse tree needs to be augmented before used for RDC
- Tree kernel for RDC differs in ways to augment/prune trees

Tree kernels for RDC

- An example of pruned parse tree augmented with entity types (Zhang et al. 2006)



Microsoft Research

Semi-Supervised Learning

- Supervised learning requires sufficient amount of annotated data
  - Expensive to obtain
  - Annotation error still occurs even dual annotated and adjudicated (ACE 2005)
- Semi-supervised learning (SSL) use a handful of seed tuples or patterns
- Bootstrapping alternates between finding pairs of arguments and contexts(pattern) of them

Bootstrapping

Initial Seed Tuples:

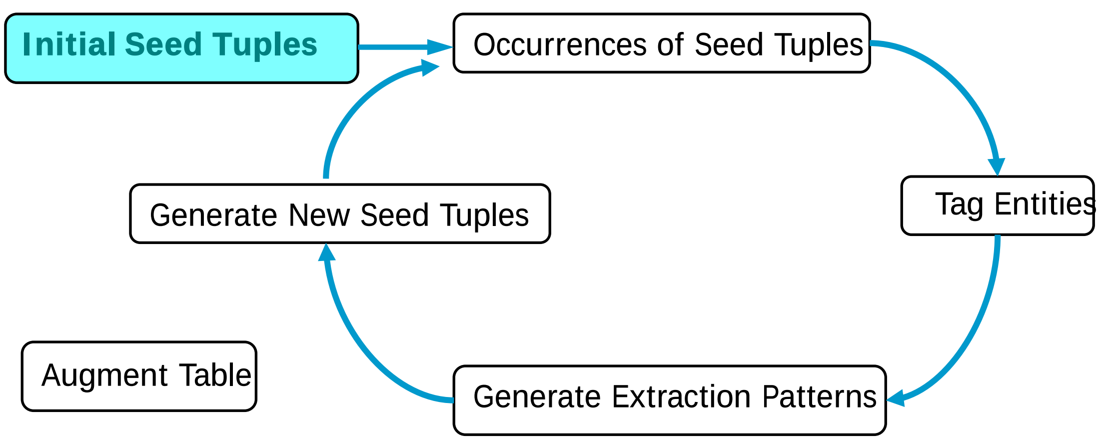
ORGANIZATION	*LOCATION*
MICROSOFT	REDMOND
IBM	ARMONK
BOEING	SEATTLE
INTEL	SANTA CLARA

DIRPRE (Brin 1998) patterns:

<*STRING1*>'s headquarters in <*STRING2*>

Snowball patterns:

<***left, NE tag1, middle, NE tag2, right***>,
left, middle, right are weighted terms



Evaluating Patterns and tuples
(Snowball)

$$Conf(Pat) = \frac{Positive}{Positive+Negative}$$

$$Conf(Tuple) = 1 - \prod(1 - Conf(P_i))$$

Weakly Supervision

- Handful of seeds for supervision

+/−	Arg a_1	Arg a_2
+	Google	YouTube
+	Adobe Systems	Macromedia
+	Viacom	DreamWorks
+	Novartis	Eon Labs
−	Yahoo	Microsoft
−	Pfizer	Teva

Table 1: Corporate Acquisition Pairs.

$+/S_1$: Search engine giant **Google** has bought video-sharing website **YouTube** in a controversial \$1.6 billion deal.

$-/S_2$: The companies will merge **Google**'s search expertise with **YouTube**'s video expertise, pushing what executives believe is a hot emerging market of video offered over the Internet.

$+/S_3$: **Google** has acquired social media company, **YouTube** for \$1.65 billion in a stock-for-stock transaction as announced by Google Inc. on October 9, 2006.

$+/S_4$: Drug giant **Pfizer Inc.** has reached an agreement to buy the private biotechnology firm **Rinat Neuroscience Corp.**, the companies announced Thursday.

$-/S_5$: He has also received consulting fees from Alpharma, Eli Lilly and Company, **Pfizer**, Wyeth Pharmaceuticals, **Rinat Neuroscience**, Elan Pharmaceuticals, and Forest Laboratories.

Figure 1: Sentence examples.

Bunescu and Mooney, 2007

Weakly Supervision (cont.)

- A SVM solution to tolerate noisy positive instances

minimize:

$$\mathbf{J}(w,b,\xi) = \frac{1}{2}\|w\|^2 + \frac{C}{L}\left(c_p\frac{L_n}{L}\Xi_p + c_n\frac{L_p}{L}\Xi_n\right)$$

$$\Xi_p = \sum_{X\in\mathcal{X}_p}\sum_{x\in X}\xi_x$$

$$\Xi_n = \sum_{X\in\mathcal{X}_n}\sum_{x\in X}\xi_x$$

Use a lower penalize factor for positive errors to tolerate noises from positive instances

subject to:

$$w\,\phi(x) + b \geq +1 - \xi_x,\ \ \forall x \in X \in \mathcal{X}_p$$

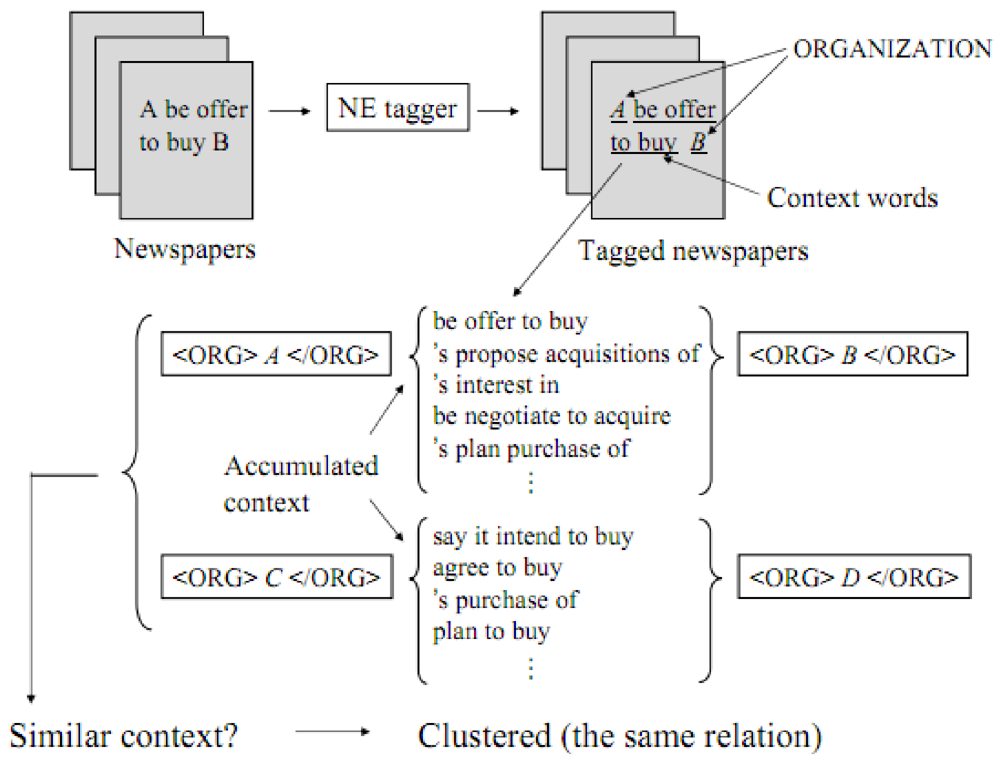
$$w\,\phi(x) + b \leq -1 + \xi_x,\ \ \forall x \in X \in \mathcal{X}_n$$

$$\xi_x \geq 0$$

Microsoft Research

Unsupervised Learning

- Automatically find major relations and respective arguments
- builds on the same duality of name pairs and contexts as relation bootstrapping methods



Hasegawa et al. 2004

- Uses Sekine's Extended NE tagger
- A domain is defined as a pair of name classes
- Bag-of-words features to model relational context
- hierarchical clustering

References for General Relation Mining

- MUC-7, http://www-nlpir.nist.gov/related_projects/muc/proceedings/muc_7_toc.html
- ACE, http://www.itl.nist.gov/iad/mig/tests/ace/
- KBP, http://nlp.cs.qc.cuny.edu/kbp/2011/
- Nanda Kambhatla. Combining Lexical, Syntactic, and Semantic Features with Maximum Entropy Models for Information Extraction. ACL 2004
- GuoDong Zhou, Jian Su, Jie Zhang, and Min Zhang. Exploring Various Knowledge in Relation Extraction. ACL 2005
- Shubin Zhao and Ralph Grishman. Extracting Relations withh Integrated Information Using Kernel Methods. ACL 2005
- Razvan Bunescu and Raymond J. Mooney . Subsequence Kernels for Relation Extraction . In Proceedings of the 19th Conference on Neural Information Processing Systems (NIPS), Vancouver, BC, December 2005

References for General Relation Mining (cont.)

- Michael Collins and Nigel Duffy. Convolution Kernels for Natural Language. NIPS 2001.
- Min ZHANG, Jie ZHANG, Jian SU, Exploring Syntactic Features for Relation Extraction using a Convolution Tree Kernel, In ACL 2006.
- Sergei Brin. Extracting Patterns and Relations from the World Wide Web. In Proc. World Wide Web and Databases International Workshop, pages 172-183. Number 1590 in LNCS, Springer, March 1998.
- Eugene Agichtein and Luis Gravano, Snowball: Extracting Relations from Large Plain-Text Collections, In Proc. 5th ACM International Conference on Digital Libraries (ACM DL), 2000
- Razvan Bunescu and Raymond J. Mooney. Learning to Extract Relations from the Web using Minimal Supervision. ACL 2007
- Takaaki Hasegawa, Satoshi Sekine, Ralph Grishman Discovering Relations among Named Entities from Large Corpora. ACL 2004.

Outline

- Overview
- Semantic class mining
- Semantic hierarchy construction
- Mining attribute names and values
- General relation extraction
- **Demo**
- **Summary**

Demo: NeedleSeek

Sempute NeedleSeek

- A sub-project of the Sempute (*Semantic Computing*) project in WSM group, MSRA
- URL: http://needleseek.msra.cn

Semantic Mining: **Summary**

- Semantic class mining
  - Sample: {C++, C#, Java, PHP, Perl, …}
  - Methods: Pattern matching (1st-order co-occurrences); distributional similarity (2nd-order co-occurrences)
- Semantic hierarchy construction
  - Key task: Hypernymy extraction (Beijing→city; pear→fruit; pear→shape)
  - Pattern matching; tuple aggregation; Label voting
- Mining attribute names and values
  - Samples: (company, CEO); (China, capital, Beijing)
  - Pattern learning; pattern matching; Table extraction; Wikipedia Infobox
- General relation extraction
  - Sample: WorkFor(Susan Dumais, Microsoft Research)
  - Supervised, semi-supervised, & unsupervised learning
  - Process contexts (especially middle contexts)