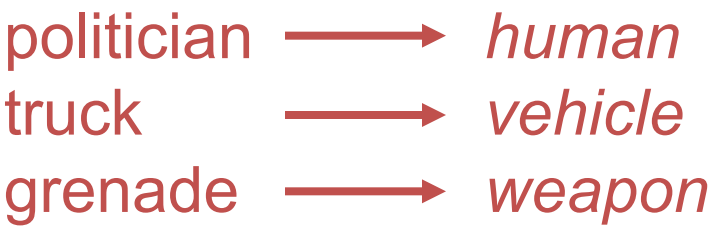
Semantic Class Induction

Some slides were adapted from the ones created by Ellen Riloff

Motivation

- A semantic lexicon assigns semantic categories to words.



- Domain-specific vocabulary is often not found in general purpose resources, such as WordNet.
- Automatic methods could be used to enhance these resources or create domain-specific lexicons.

Syntactic Heuristics for Learning Semantic Labels

Conjunctions	lions and tigers and bears
Lists	lions, tigers, bears
Appositives	the horse, a stallion
Predicate Nominals	the wolf is a mammal
Compound nouns	tuna fish
	Honda Sedan

[Riloff & Shepherd 97; Roark & Charniak 98; Phillips & Riloff 02; etc.]

Hyponym patterns	dogs such as beagles and boxers
	dogs, including beagles and boxers

[Hearst 92; KnowItAll (U.Washington), Kozareva et al. 2008; etc.]

Bootstrapping Semantic Lexicons

Unannotated Texts

Ex: dog, cat, lion, lizard, snake

N best words

Ex: terrier, poodle, tiger, frog, iguana

Co-occurrence Statistics

prospective category words

Extraction Patterns

- Represent syntactic context that often reveals the semantic class of a word.
- AutoSlog: each pattern extracts an NP from one of 3 syntactic positions: *subject, direct object, pp obj*.

Some patterns to extract locations:

<subject> was inhabited	**the locality was inhabited…**
patrolling <direct object>	**…patrolling Zacamil neighborhood**
lives in <pp obj>	**…lives in Argentina**

EXTRACTION PATTERN TYPES

<subject> passive-vp	*<target> was bombed*
<subject> active-vp	*<perpetrator> bombed*
<subject> active-vp dobj	*<perpetrator> threw dynamite*
<subject> active-vp infinitive	*<perpetrator> tried to kill*
<subject> passive-vp infinitive	*<perpetrator> was hired to kill*
<subject> auxiliary dobj	*<victim> was fatality*

---

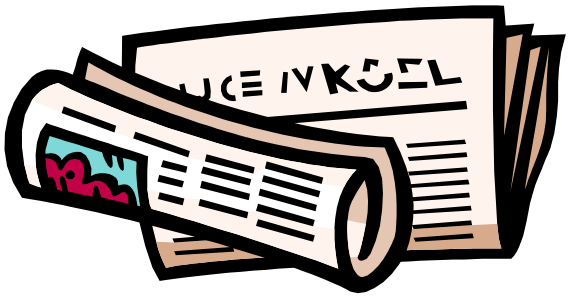
active-vp <dobj>	*bombed <target>*
infinitive <dobj>	*to kill <victim>*
active-vp infinitive <dobj>	*tried to kill <victim>*
passive-vp infinitive <dobj>	*was hired to kill <victim>*
subject auxiliary <dobj>	*fatality was <victim>*

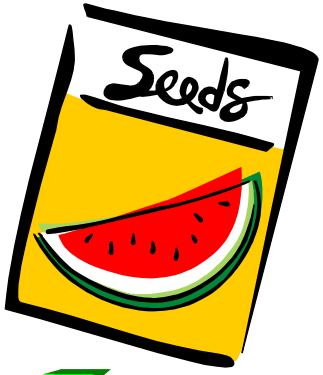
---

passive-vp prep <np>	*was killed by <perpetrator>*
active-vp prep <np>	*exploded in <target>*
infinitive prep <np>	*to kill with <weapon>*
noun prep <np>	*assassination of <victim>*

Mutual Bootstrapping [Riloff & Jones 99]

Unannotated Texts





Ex: dog, cat, lion, lizard, snake

Ex: <NP> growled

Best Extraction Pattern

Ex: Rottweiler, terrier, cougar

Extractions (Nouns)

Mutual Bootstrapping Example

SEEDS: Nicaragua, city, region, town

Best pattern: headquartered in <NP>
Extractions: Nicaragua, city, Chapare region, San Miguel

Best pattern: downed in <NP>
Extractions: Nicaragua, city, Usulutan region,
San Miguel, area, Soyapango

Best pattern: to occupy <NP>
Extractions: Nicaragua, town, this northern area,
small country, San Sebastian neighborhood,
private property

Examples of Learned Patterns

Location Patterns (Web)	Location Patterns (Terrorism)
offices in <np>	living in <np>
facilities in <np>	traveled in <np>
operations in <np>	become in <np>
loans in <np>	sought in <np>
operates in <np>	presidents in <np>
locations in <np>	parts of <np>
producer in <np>	to enter <np>
states of <np>	condemned in <np>
seminars in <np>	relations between <np>
activities in <np>	ministers of <np>
consulting in <np>	part in <np>
countries of <np>	taken in <np>

Bootstrapping Procedure

1. **Start from several seed words for a semantic class and an unlabeled text corpus.**

2. **Score patterns and keep the top N patterns.**

3. **Score pattern extractions (candidate lexicon words) and select the top M new words as new lexicon words.**

4. **Increase N by 1 and go back to step 1.**

Pattern Scoring

Every extraction pattern is scored and the best patterns are kept.

The scoring function is:

$$\text{RlogF}(\text{pattern}_i) = \frac{F_i}{N_i} * \log_2(F_i)$$

where:

F_i is the number of unique category members extracted by pattern_i

N_i is the total number of unique nouns extracted by pattern_i

Selecting Words for the Lexicon

Score: the average number of category members extracted by each pattern (*while the original algorithm considers all patterns, we'll only consider patterns selected in the previous iteration*) that extracted the candidate word.

$$\text{score}(\text{word}_i) = \frac{\sum_{j=1}^{N_i} F_j}{N_i}$$

$$\text{AvgLog}(\text{word}_i) = \frac{\sum_{j=1}^{N_i} \log_2 (F_j + 1)}{N_i}$$

where:

F_j **is the number of unique category members extracted by pattern**$_j$

N_i **is the total number of patterns that extract word**$_i$

More notes

- The mutual bootstrapping approach can be extend to learn semantic lexicon in multiple categories (the Basilisk system).
- Very often, manual review is still necessary to use the learned dictionaries.
- Performance for some categories is beginning to approach levels for which manual review may not be necessary.