Information extraction

- **Introduction**
  - Task definition
  - Evaluation
  - IE system architecture
- **Acquiring extraction patterns**
  - Manually defined patterns
  - Learning approaches
    - Semi-automatic methods
    - Fully automatic methods
  - Finite-state methods
- **Named entity detection**

Issues…

- tension between **domain-independent** and **domain-dependent** language processing
  - treating task in a domain-independent way allows the use of general IR/NLP techniques and tools
  - treating task in a domain-dependent way allows for tailoring of techniques for better performance
- IE is generally handled as **domain-specific text understanding**
  - key system components need to be re-built for each new domain
  - difficult and time-consuming to build if constructed manually
    - Initially, ~6-12 months/system for IE from unstructured text
  - requires the expertise of computational linguists

Information extraction

- **Introduction**
  - Task definition
  - Evaluation
  - IE system architecture
- **Acquiring extraction patterns**
  - Manually defined patterns
  - Learning approaches
    - Semi-supervised methods
    - Fully supervised methods
  - Finite-state methods
- **Named entity detection**

Exercise: changes in management

post post
The company also said its president and former chairman both resigned.

IO-person:out
Evergreen said Barry Nelsen, who had a heart-bypass operation

post post
last week, resigned as president and chief executive. The board

IO-person:out
formally accepted the resignation of Thomas Casey, its former

post post
chairman, who stepped down effective Feb. 2.

Information extraction

- **Introduction**
  - Task definition
  - Evaluation
  - IE system architecture
- **Acquiring extraction patterns**
  - Manually defined patterns
  - Learning approaches
    - Semi-supervised methods
    - Fully supervised methods
  - Finite-state methods
- **Named entity detection**

Machine learning methods

- **acquire linguistic knowledge** by applying statistical and symbolic learning methods; derive training examples from the texts themselves
- **automate** the construction of each IE system component
- improve **robustness** of final systems while maintaining (or at least approaching) the accuracies of handcrafted systems

Learning IE patterns from examples

- **Goal**
  - Given a training set of ***annotated*** documents [answer keys],
  - Learn extraction patterns for each slot using an appropriate machine learning algorithm.
- **Options**
  - Memorize the fillers of each slot?
  - Generalize the fillers using context and
    - p-o-s tags?
    - phrase structure (NP, V) and grammatical roles (SUBJ, OBJ)?
    - semantic categories?

Learning IE patterns

- **Methods vary with respect to**
  - The **class of pattern** learned (e.g. lexically-based regular expression, syntactic-semantic pattern)
  - **Training corpus** requirements
  - Amount and type of **human feedback** required
  - Degree of **pre-processing** necessary
  - **Other resources**/knowledge bases required

Syntactico-semantic patterns

The twister occurred without warning at approximately 7:15p.m. and ***destroyed two mobile homes***.

Pattern:

Trigger: “destroyed”

condition: active voice verb?

Slot: Damaged-Object

Position: direct-object

condition: DO is a physical-object?

from Cardie [1997]

Pattern templates

Noun phrase extraction only

<subject> <passive-verb>	<victim> was **murdered**
<subject> <active-verb>	<perpetrator> **bombed**
<subject> <infinitival-verb>	<perpetrator> attempted to **kill**
<subject> <auxiliary-verb>+<noun>	<victim> was **victim**
***<passive-verb> <dobj>**	**killed** <victim>
<active-verb> <dobj>	**bombed** <target>
<infinitive> <dobj>	**to kill** <victim>
<verb>+<infinitive> <dobj>	threatened to **attack** <target>
<gerund> <obj>	**killing** <victim>
<noun>+ <auxiliary> <dobj>	**fatality** was <victim>
<noun>+<prep> <np>	**bomb** against <target>
<active-verb>+<prep> <np>	**killed** with <instrument>
<passive-verb>+<prep> <np>	was **aimed** at <target>

Autoslog algorithm

- **For each “string fill”, *s*, in the training data**
  - (Shallow) parse the sentence that contains *s.*
  - Apply the syntactic pattern templates in order. Execute the first one that applies to determine:
    - the *trigger* word
    - the triggering *constraints*
    - the *position* of phrase to be extracted
  - Determine *slot type*
    - The annotated slot type for *s* in the training corpus
  - Determine the *semantic constraints*
    - Defined a priori based on typical semantic class of fillers
  - Create and save the extraction pattern

Example

The twister occurred without warning at approximately 7:15p.m. and ***destroyed two mobile homes***.

damaged-object

Pattern:

Trigger: “<verb>”

condition: active voice

Slot: <slot-type> of <target-np>

Position: direct-object

condition: DO is <<semantic class> of <slot-type>>

Instantiation:

Trigger: “destroyed”

condition: active voice verb?

Slot: Damaged-Object

Position: direct-object

condition: DO is a physical-object?

Learned terrorism patterns

- **<victim> was murdered**
- **<perpetrator> bombed**
- **<perpetrator> attempted to kill**
- **was aimed at <target>**

Bad patterns are possible

- **took <victim>**

victim

They took 2-year-old Gilberto Molasco, son of Patricio Rodriquez, and 17-year-old Andres Argueta, son of Ernesto Argueta.

Natural disasters patterns

- Yesterday's earthquake registered 6.9 on the Richter scale.
  - <subject> = disaster-event (earthquake) registered (active)
  - registered (active) <direct obj> = magnitude
- measuring 6.9 …
  - measuring (gerund) <direct obj> = magnitude
- …sending medical aid to Afghanistan…
- …sending medical aid to earthquake victims
  - aid (noun)…in/to/for (prep) <obj> = disaster-event-location/victim

Autoslog algorithm

- **Domain-independent**
  - So require little modification when switching domains
- **Requires (minimally) a partial parser**
- **Assumes semantic category(ies) for each slot are known, and all potential slot fillers can be tested w.r.t. them**

Exercise: changes in management

post post
The company also said its president and former chairman both resigned.

IO-person:out
Evergreen said Barry Nelsen, who had a heart-bypass operation

post post
last week, resigned as president and chief executive. The board

IO-person:out
formally accepted the resignation of Thomas Casey, its former

post post
chairman, who stepped down effective Feb. 2.

Advantages and Disadvantages

- **Learns bad patterns as well as good patterns**
  - Too general (e.g. triggered by “is” or “are” or by verbs not tied to the domain)
  - Too specific
  - Just plain wrong
    - Parsing errors
    - Target NPs occur in a prepositional phrase and Autoslog can’t determine the trigger (e.g. is it the preceding verb or the preceding NP?)
- **Does not make good use of the training data**
  - Requires that a person review the proposed extraction patterns, discarding bad ones
- **No computational linguist needed (?)**
- **Reduced human effort from 1200-1500 hours to ~4.5 hours**

Results

- **1500 texts, 1258 answer keys**
- **4780 slots (6 types)**
- **Autoslog generated 1237 patterns**
- **After human filtering: 450 patterns**
- **Compare to manually built patterns**

System/Data Set	Recall	Precision	F-measure
Manual/TST3	46	56	50.51
Autoslog/TST3	43	56	48.65
Manual/TST4	44	40	41.90
Autoslog/TST4	39	45	41.79

Autoslog-TS

- **Largely unsupervised**
- **Two sets of documents: relevant, not relevant**
- **Apply pattern templates to extract every NP in the texts**
- **Compute *relevance rate* for each pattern *i* :**

Pr (relevant text | text contains i) =
freq of *i* in relevant texts / frequency of *i* in corpus

- **Sort patterns according to relevance rate and frequency**

relevance rate * log (freq)

Information extraction

- **Introduction**
  - Task definition
  - Evaluation
  - IE system architecture
- **Acquiring extraction patterns**
  - Learning approaches
    - Semi-supervised methods for extraction from unstructured text
    - Fully supervised methods for extraction from structured text
  - Finite-state methods
- **Named entity detection**

Covering algorithms

- **E.g. Crystal** [Soderland et al. 1995]
  - Allows for more complicated patterns
    - Can test target NP or any constituent in its context for
      - presence of any word or sequence of words
      - semantic class of heads or modifiers
- **Crystal is a “covering” algorithm**
- **Successively generalizes the patterns derived from input examples until the generalization produces errors**

Supervised Inductive Learning

Examples
[features + class]

ML Algorithm

(novel) examples
[features]

statistical model
(program)

class

Extracting Sources of Opinions

- **Supervised learning**
  - View as a sequence tagging task

<The Washington Post> reported **<Blair>**’s view on the oil crisis.

Machine Learning of Sources

Examples **of (Source) NPs in context**
[features + class]

ML Algorithm

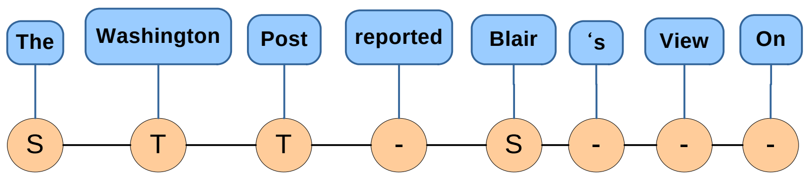
(novel) **NPs in context**
[features]

statistical model
(program)

Source?
Not Source?

Extracting Sources of Opinions

- **Supervised learning**
  - Sequence tagging
    - HMMs, MEMMs, CRFs



<The Washington Post> reported **<Blair>**’s view on the oil crisis.

Class Values

- **IOB representation**
  - B – *begins* an opinion holder phrase
  - I – *inside* an opinion holder phrase
  - O – *outside* an opinion holder phrase

Set fill extraction

- **If a slot has a fixed set of pre-specified possible fillers, text categorization methods can be used to fill the slot.**
  - Job category
  - Company type
- **Treat each of the possible values of the slot as a category, and classify the entire document or the sentence to determine the correct filler.**