Information Extraction

Jordi Turmo

TALP Research Centre
turmo@lsi.upc.edu
Summary

- Introduction
- Architecture
- Examples
- Knowledge specific for IE
- Adaptability
Knowledge specific for IE

Knowledge resources

• Knowledge more or less stable
  – general lexicon
  – general grammar
  – basic NL processors: segmenters, taggers, parsers, ...

• Domain dependent knowledge
  – Domain specific vocabularies, terminology
  – gazetteers and patterns for NERC
  – **IE patterns**

Knowledge specifically used for IE
Knowledge specific for IE

Types of IE patterns

• Viewpoint 1: type of representation

  − rules

  X:subj(C-instrument) ... X=vp(go_off) ... X:mod(C-time) ... “in” X:mod(C-location)

  →

  INSTRUMENT := C-instrument
  TIME := C-time
  LOCATION := C-location
Knowledge specific for IE

Types of IE patterns

• Viewpoint 1: type of representation
  – rules
  – statistical models (ME, Hyperplanes, ...)

Jordi Turmo
Adaptive Information Extraction
Knowledge specific for IE

Types of IE patterns

– Viewpoint 2: type of values extracted

– event extraction patterns
  (the rule presented before)

\[
X:subj(\text{C-instrument}) \ldots X=vp(\text{go\_off}) \ldots X:mod(\text{C-time}) \ldots \text{“in”} \ X:mod(\text{C-location}) \rightarrow
\]

Event:INSTRUMENT := C-instrument
Event:DATE := C-time
Event:LOCATION := C-location
Types of IE patterns

– Viewpoint 2: type of values extracted

  – event extraction patterns
    (the rule presented before)

  – relation extraction patterns

  np(C-person) ... vp(is) pron(C-his) “wife”
  →
  Married_with:HUSBAND := C-his
  Married_with:WIFE := C-person
Summary

• Introduction
• Architecture
• Examples
• Knowledge specific for IE
• Adaptability
  • Introduction
  • Relation pattern learning
  • Event pattern learning
Introduction

- Use of ML techniques for ...
  - Low level tasks:
    - POS tagging
    - Segmentation
    - *Chunking*
    - Syntactic dependencies between chunks
    - NERC
    - Semantic role labeling
    - Coreference resolution
  - **IE pattern acquisition**
    - Corpus-based approaches (Cardie[97], Mooney and Cardie[99], Turmo et al.[06])
Summary

• Introduction
• Architecture
• Examples
• Knowledge specific for IE
• Adaptability
  • Introduction
  • Relation pattern learning
  • Event pattern learning
Relation pattern learning

- Supervised learning

Annotation of positive examples:
  - Identify participants (NE mentions), and
  - Associate a *slot* of the output structure to each participant to be extracted from the example

Employment relation

- Jordan Turmo works at UPC

Negative examples are the rest of entity pairs
Relation pattern learning

• Supervised learning

Examples of possible features, after document preprocess:

• Heads of both entities
• Semantic types of both entities (PER, ORG, LOC, ...)
• Distance in words between both entities
• Sequence of words between both entities
• Individual words between both entities
• Dependency tree between entities
• Shortest dependency path between entities
• ...

Jordi Turmo

Adaptive Information Extraction
Relation pattern learning

• Supervised learning
  – Examples of approaches:
    • ME (Kambhatla, 04)
    • SVM
      – Sequence kernel (Bunescu and Mooney, 05)
      – Tree kernel (Zelenko et al., 03)(Zhou and Grishman, 05) (Zhou et al., 07)
      – Composite kernel (Zhang et al., 06)
  – Require the annotation of large number of relations
Relation pattern learning

SVM (Zhang et al., 06)

- Composite kernel: entity kernel + tree kernel
- Entity kernel uses:
  - \( E_1, E_2 = \) Entities in the relation
- Convolution Tree kernel uses:
  - \( T = \) enclosed tree between the entities
Relation pattern learning

SVM (Zhang et al., 06)

- Composite kernel: entity kernel + tree kernel
  Example: relation work_for

Several students have jobs in their spare time, Henrik works as a system developer at IBM in USA
Relation pattern learning

SVM (Zhang et al., 06)

• Composite kernel: entity kernel + tree kernel
  Example: relation work_for

Several students have jobs in their spare time, Henrik works as a system developer at IBM in USA

E1

E2

Jordi Turmo
Adaptive Information Extraction
Relation pattern learning

SVM (Zhang et al., 06)

- Composite kernel: entity kernel + tree kernel

Example: relation work_for (in an incorrect parse tree)

Several students have jobs in their spare time, Henrik works as a system developer at IBM in USA.
Relation pattern learning

SVM (Zhang et al., 06)

- Composite kernel: entity kernel + tree kernel

Example: relation work_for (in an incorrect parse tree)
Relation pattern learning

SVM (Zhang et al., 06)

- Composite kernel: \( \alpha (K_1(R,R')+1)^2 + (1-\alpha) K_2(T,T') \)

- Entity kernel:
  - \( E_1, E_2 = \) Entities in the relation
  - \( K_1(R,R') = \sum C(E_1.f_i, E_1'.f_i) + \sum C(E_2.f_i, E_2'.f_i) \)
  - \( f_i = \) entity head, semantic type and subtype, mention type

- Convolution Tree kernel:
  - \( T = \) enclosed tree between the entities
  - \( \Phi(T) = (\#\text{subtree}_1(T), \ldots, \#\text{subtree}_K(T)) \)
  - \( K_2(R,R') = K(\Phi(T), \Phi(T')) = \sum \#\text{subtree}_i(T) \#\text{subtree}_i(T') \)
Relation pattern learning

SVM (Zhang et al., 06)

• $\Phi(T) = (#\text{subtree}_1(T), ..., #\text{subtree}_K(T))$

Ex:

```
DT   NN   DT   NN   DT   NN
|-----|-----|-----|-----|-----|-----|
The  man The  man The  man
```

```
NP   NP   NP   NP
|-----|-----|-----|-----|
DT   NN   DT   NN   DT   NN
|-----|-----|-----|-----|
The  man The  man
```

NP $\rightarrow$ DT NN
DT $\rightarrow$ the
NN $\rightarrow$ man

Jordi Turmo

Adaptive Information Extraction
Relation pattern learning

SVM (Zhang et al., 06)

- $\Phi(T) = (#\text{subtree}_1(T), ..., #\text{subtree}_K(T))$
- $K_2(R,R') = K(\Phi(T), \Phi(T')) = \sum #\text{subtree}_i(T) #\text{subtree}_i(T')$

Ex:

R1 = $w_1 w_2 w_3 ... w_{10} w_{11} ... w_{15} w_{16} w_{17}$

R2 = $w_1 w_2 w_3 ... w_{10} w_{11} w_{12} ... w_{15} w_{16} w_{17}$

(For simplicity, we only take subtrees a, b, c)

Jordi Turmo
Relation pattern learning

SVM (Zhang et al., 06)

- Results on ACE 2002/2003

<table>
<thead>
<tr>
<th>Kernel Type</th>
<th>P</th>
<th>R</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity kernel</td>
<td>75.1</td>
<td>42.7</td>
<td>54.4</td>
</tr>
<tr>
<td>Tree kernel</td>
<td>72.5</td>
<td>56.7</td>
<td>63.6</td>
</tr>
<tr>
<td>Composite kernel</td>
<td>76.1</td>
<td>68.4</td>
<td>72.1</td>
</tr>
</tbody>
</table>
Relation pattern learning

SVM (Zhang et al., 06)

- Comparison on ACE 2002/2003

<table>
<thead>
<tr>
<th>Method</th>
<th>P</th>
<th>R</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zhang et al. 06 (composite kernel)</td>
<td>77.3 (64.9)</td>
<td>65.6 (51.2)</td>
<td>70.9 (57.2)</td>
</tr>
<tr>
<td>Zhou et al. 05 (feature-based kernel)</td>
<td>72.5 (63.1)</td>
<td>56.7 (49.5)</td>
<td>63.6 (55.5)</td>
</tr>
<tr>
<td>Kambhatla, 04 (feature-based ME)</td>
<td>- (63.5)</td>
<td>- (45.2)</td>
<td>- (52.8)</td>
</tr>
<tr>
<td>Bunescu and Mooney 05 (dependency kernel)</td>
<td>65.5</td>
<td>43.8</td>
<td>52.5</td>
</tr>
<tr>
<td>Culotta and Sorensen, 04 (dependency kernel)</td>
<td>67.1</td>
<td>35.0</td>
<td>45.8</td>
</tr>
</tbody>
</table>
Relation pattern learning

• Semi-supervised learning
  – Examples of approaches:
    • Co-training variants (Sun & Grishman, 10)
    • Distant learning (Mintz et al., 09)
  – Require few supervision
Relation pattern learning

• Co-training variant (Sun & Grishman, 10)
  – Basic approach: a form of co-training for each relation type
  – Learn linguistic patterns for each relation type
  – Two views: both NEs + context of both NEs
Relation pattern learning

• Co-training variant (Sun & Grishman, 10)
  – Basic approach: a form of co-training for each relation type
  – Learn linguistic patterns for each relation type
  – Two views: both NEs + context of both NEs

1. Start with few seed patterns with confidence=1 (e.g. book-autor relation → “TITLE was written by PER”)
2. Collect examples (entity pairs) matching patterns (e.g. <“Animal Farm”, “George Orwell”>)
3. Compute the Confidence of the examples
4. Collect additional patterns using the new examples
5. Compute the Confidence of the new patterns
6. Select the K top patterns given their confidences
7. Repeat from 2 until stop condition (Most extended stop criterion: N iterations)
Relation pattern learning

• Co-training variant (Sun & Grishman, 10)
  – Basic approach: a form of co-training for each relation type
  – Learn linguistic patterns for each relation type
  – Two views: both NEs + context of both NEs

1. Start with few seed patterns with confidence=1 (eg. book-autor relation → “TITLE was written by PER”)
2. Collect examples (entity pairs) matching patterns (eg. <“Animal Farm”, “George Orwell”>)
3. **Compute the Confidence of the examples**
4. Collect additional patterns using the new examples
5. **Compute the Confidence of the new patterns**
6. Select the K top patterns given their confidences
7. Repeat from 2 until **stop condition** (Most extended stop criterion: N iterations)
Relation pattern learning

- Co-training variant (Sun & Grishman, 10)
  - Pattern representation
    Shortest dependency path between a NE pair

Ex: *LocationIn*

*Gates met with President Clinton in Seattle*

nsubj ← met → prep_in
Relation pattern learning

- Co-training variant (Sun & Grishman, 10)
  - Standard confidence of examples

\[
\text{Conf (e)} = 1 - \prod_{j=K}^{n} (1 - \text{Prec (p}_j)) \quad \text{initially,} \quad \text{Prec(p}_j) = 1
\]
Relation pattern learning

- Co-training variant (Sun & Grishman, 10)
  - Standard confidence of examples

\[
\text{Conf}(e) = 1 - \prod_{j \leq K} (1 - \text{Prec}(p_j)) \quad \text{initially, Prec}(p_j) = 1
\]

- Standard confidence of patterns

\[
\text{Conf}(p) = \frac{\text{Sup}(p)}{|H|} \log \text{Sup}(p) \quad \text{Sup}(p) = \sum_{e_i \in |H|} \text{Conf}(e_i)
\]

\[
\text{Prec}(p) = \frac{\text{Sup}(p)}{|H|}
\]

H is the set of examples matched by p
Relation pattern learning

- Distant learning (Mintz et al., 09)
  “any sentence that contains a pair of entities that participate in a known relation in a KB is likely to express that relation in some way”

Ex:

KB: date_of_birth <Michelle Obama, January 17, 1964>

Corpus: Michelle Obama was born in Chicago in January 17, 1964

- Build training examples from KB and corpus
- Use a supervised learning approach
Relation pattern learning

- Semi-supervised learning problems
  - Noisy examples
    - Ex: target relation *LocatedIn*
      - example = <Clinton, Arkansas>
      - mentions = Clinton visited Arkansas
      - Clinton has been elected Governor of Arkansas
      - Clinton was born in Arkansas
      - The example can match GovernorOfLocation and BornInLocation
  
  - Lack of positive examples for unfrequent relation types
  - Negative examples? Ex: mentions unrelated in the KB for distant learning. (false negative?)

Jordi Turmo
Adaptive Information Extraction
Summary

- Introduction
- Architecture
- Examples
- Knowledge specific for IE
- **Adaptability**
  - Introduction
  - Relation pattern learning
  - *Event pattern Learning*
Event pattern learning

- Supervised learning:
  Annotation of positive examples:
  - Identify triggers (commonly verbs), and
  - Associate a *slot* of the output structure to each slot-filler

incident-event
  type: natural-disaster
  cause
  time
  damage

Witnesses confirm that the twister occurred without warning at approximately 7:15 p.m. and destroyed two mobile homes.
Event pattern learning

• Supervised learning
  Divide the problem (Ahn,06)…:
  a) Identify which words in the document are triggers of the required event types (verbs or nominalizations)
     -> learn a trigger identifier/classifier
  b) Identify which pair <trigger, entity mention> defines a slot of an event
     -> learn an argument identifier/classifier
  c) Merge partial event extractions
     -> learn a classifier for coreferential event pairs
Event pattern learning

• Supervised learning

Learn a trigger identifier/classifier

e+ = trigger word   Ex: Natural-Disaster: e+ = destroy

Examples of possible local features:
• the word, its left/right contexts and some lexical information (POS, sense, ...)
• the syntactic dependences of the trigger
• other entities mentioned in the sentence of the trigger

Problem: trigger disambiguation may require larger context

Ex: *It destroyed two mobile homes yesterday in Paris and ...*

  bombing? natural disaster?

Examples of possibly useful global information:
• Sentence event classification, Event cooccurrences, Document topic classification
**Event pattern learning**

- **Supervised learning**
  
  **Learn a trigger identifier/classifier**

  
  $e^+ = \text{trigger word}$  
  
  Ex: Natural-Disaster: $e^+ = \text{destroy}$

  
  **Possible architectures:**

  - A binary trigger identifier (the word is/is_not a trigger word)
    
    - $e^+ = \text{trigger words, } e^- = \text{rest of words}$

  - A trigger multi-class classifier (it is a trigger for event-type)
    
    - $e^+ = \text{trigger words of the event-type, } e^- = \text{rest of trigger words}$

  - Directly, a trigger multi-class classifier
    
    - $e^+ = \text{trigger words of the event-type, } e^- = \text{rest of words}$

    - (particular no-event class?)

Jordi Turmo

Adaptive Information Extraction
Event pattern learning

• Supervised learning
  Learn a trigger identifier/classifier
  Example application:
  • Identifier: Trigger / no-trigger

Witnesses confirm that the twister occurred without warning at approximately 7:15 p.m. and destroyed two mobile homes.
Event pattern learning

• Supervised learning
  Learn a trigger identifier/classifier
  Example application:
  • Identifier: Trigger / no-trigger

Witnesses confirm that the twister occurred without warning at approximately 7:15 p.m. and destroyed two mobile homes.
Event pattern learning

• Supervised learning

  Learn a trigger identifier/classifier

  Example application:
  • Identifier: Trigger / no-trigger
  • Classifier: Natural-disaster / bombing / ...

Witnesses confirm that the twister occurred without warning at approximately 7:15 p.m and destroyed two mobile homes.

relation previously extracted with relation patterns
Event pattern learning

• Supervised learning
  
  Learn argument identifier/classifier
  
  $$e^+ = (\text{trigger, entity mention})$$  
  Example: damage: $$e^+ = (\text{destroy, two mobile homes})$$

  **Examples** of possible features:
  • the trigger word and the event type
  • the constituent head, the determiner (if any) and the type (name, pronoun, other) of the entity mention
  • 3 dependency paths (expressed as sequence of words, POS and labels)

  **Possible architectures:**
  • similar to the learning of trigger models
Event pattern learning

• Supervised learning
  Learn an argument identifier/classifier
  Example application:
  • Pairs to try:  <destroy, Witness>
    <destroy, twister> ...
    <destroy, two mobile homes>

Witnesses confirm that the twister occurred without warning at approximately 7:15 p.m and destroyed two mobile homes.
Event pattern learning

• Supervised learning

  Learn an argument identifier/classifier

  Example application:
  • Pairs to try:  
    \begin{itemize}
    \item \texttt{<destroy, Witness>}
    \item \texttt{<destroy, twister> \ldots}
    \item \texttt{<destroy, two mobile homes>}
    \end{itemize}
    \hfill <- CAUSE
    \hfill <- DAMAGE

Witnesses confirm that the \textit{twister} occurred without warning at approximately 7:15 p.m and destroyed \textit{two mobile homes}. 
Event pattern learning

• Supervised learning

  Learn a classifier for coreferential event pairs

  e+ = <event_mention_1, event_mention_2>

Examples of possible features:

• event type
• matching of both triggers, or their semantics
• distance between the event mentions
• overlapping of their arguments (ex: both subjects)

Learn a mention-pair classifier similar to those for coreference resolution.
**Event pattern learning**

- **Supervised learning**
  
  Learn a classifier for coreferential event pairs

  Example application:
  
  - Pairs: `<destroy, twister>` ... <- CAUSE
    `<destroy, two mobile homes>` <- DAMAGE

Witnesses confirm that the **twister** occurred without warning at approximately **7:15 p.m** and **destroyed two mobile homes**.

$$\text{Coref}(<\text{destroy, twister}>, <\text{destroy, two mobile homes}>) \ ?$$
Event pattern learning

- Supervised learning
  Learn a classifier for coreferential event pairs

Example application:
- Pairs: \(<\text{destroy, twister}> ... \) <- CAUSE
  \(<\text{destroy, two mobile homes}> \) <- DAMAGE

Witnesses confirm that the \textbf{twister} occurred without warning at approximately 7:15 p.m and \textit{destroyed} two mobile homes.

\( \text{Coref}(<\text{destroy, twister}>, <\text{destroy, two mobile homes}>) = \text{true} \)
Event pattern learning

- Semi-supervised learning
  Example: EXDISCO (Yangarber,02)

1. Initial set of patterns SVO, SV, VO
   Ex: SUBJ destroyed OBJ
2. Classify documents in \{\text{rel, unrel}\} with relevance weights \([0,1]\)
   \[ \text{Rel}_i(d) = \frac{|\text{patterns}(d)|}{|\text{patterns}|} \]
3. Acquire new patterns from the relevant texts
4. Select the relevant new patterns
   \[ \text{Prec}(p) = \frac{\text{freq}(p, \text{rel})}{\text{freq}(p)} \times \log \text{freq}(p) > \text{threshold} \]
5. Re-compute the weights of the documents
   \[ \text{Rel}_i(d) = \max (\text{Rel}_{i-1}(d), \text{Prec}_i(\text{patterns}(d))) \]
   \[ \text{Prec}_i(\text{patterns}(d)) = \sum_{d \in \text{docs}(\text{patterns}(d))} \frac{\text{Rel}_{i-1}(d)}{|\text{docs}(\text{patterns}(d))|} \]
6. Go to 3 until stop criterion
Bibliography


• M. Zhang, J. Zhang, J. Su, G. Zhou (2006) *A composite kernel to extract relations between entities with both flat and structured features*. ACL 2006


Bibliography
