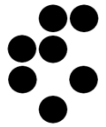
Identifying good patterns for relation extraction

Janez Starc
Blaž Fortuna



Motivation

- Extracting information from textual data and asserting it into the target ontology:
  - **Sentence**: Jim Whitehurst , CEO of Red Hat , just delivered a terrific opening keynote presentation for LinuxCon.
  - **Pattern:** *[PERSON]* , *[POSITION]* of *[ORGANIZATION]*
  - **Relation:** (positionOfPersonInOrganization JimWhitehurst Redhat CEO)
- **Good** patterns produce semantically useful relations with high percision and recall

Approach

- Our approach is similar to frequent n-gram extraction
  - Our algorithm can extract patterns from very large corpora because it does not exploit any syntactic analysis, such as, part-of-speech tags and phrase chunks

Types of patterns

- Arguments are only named entities (person, organization,...) provided by Enrycher [1]
  - Result is a relation
    - *[ORGANIZATION]* coach *[PERSON]*
      (… Boston coach Doc Rivers…)
  - Result is a concept (recursion!)
    - *[PERSON]* father
      (… Mary's father …)
- One variable length gap in the middle
  - *[PERSON]*, *[POSITION]* of *[ORGANIZATION]*
    (… Tim Brown is CEO and president of IDEO …)

Filtering and ranking of the patterns

- Statistics:
  - Frequency
  - Number of arguments
  - Number of stop words
  - Minimal token frequency
  - Normalized expectation [2]
    - Tells whether words frequently co-occur together
    - $NExp([w_1 w_2 \dots w_n]) = \frac{p([w_1 w_2 \dots w_n])}{\frac{1}{n}\sum_{i=1}^{n} p([w_1 w_2 \dots \widehat{w_i} \dots w_n])}$

Filtering and ranking of the patterns

Pattern	*Fq*	*Args*	*StopW*	*MinTokFq*	*NExp*
[PERSON] , executive director of *[ORGANIZATION]*	21	2	2	11699	0.484
's hospital in *[LOCATION]* , *[LOCATION]*	22	2	2	11917	0.571
to *[PERSON]* parents , *[PERSON]* was	40	2	3	12020	0.564
death by *[PERSON]* parents , *[PERSON]*	24	2	2	12020	0.282
[PERSON] parents , *[PERSON]* and *[PERSON]*	22	3	2	12020	0.506
[PERSON] have no idea what *[PERSON]*	20	2	3	12449	0.553
(*[ORGANIZATION]*) - *[PERSON]* scored	22	2	3	12514	0.735
victory over the *[ORGANIZATION]* on *[DATE]*	55	2	3	12626	0.653
, died *[DATE]* , at *[ORGANIZATION]*	61	2	3	12822	0.712
died *[DATE]* , at *[ORGANIZATION]* in	45	2	3	12822	0.732
[PERSON] was a member of *[ORGANIZATION]*	38	2	3	13399	0.623

Table 1 *Part of the table representing 6-gram patterns and their statistics*

Evaluation

- Translating news text to CycL and asserting it to CycKB
  - Pattern matching algorithm
  - Handcrafted translations

"I've been in this place before," Manuel said.

Pattern	*" [STRING] , " [PERSON] said*
CycL template	*(#$thereExists ?INFORMING (#$and (#$isa ?INFORMING #$Informing) (#$senderOfInfo ?INFORMING ?PERSON) (#$infoTransferred-NLString ?INFORMING ?STRING)))*
Recall	379
New terms	227
Recognized arguments	35
Total assertions	830
Matches with ambiguous assertions	37
Matches with a valid assertion	337
Precision	0.89

Future work

- More types of patterns
  - Variable length gaps in the beginning and the end of the pattern
  - Exploit part-of-speech tags and phrase chunks
- Better argument recognition
  - Word disambiguation
  - Resolving the type of argument
- Analyze patterns in different domains (business, sport, technology)
- Better ways to identify good patterns and translations with less human effort

Literature

- *[1]* Štajner, T. and Rusu, D. and Dali, L. and Fortuna, B. and Mladenić, D. and Grobelnik, M., "Enrycher: service oriented text enrichment," in *Proceedings of SiKDD*, 2009.
- *[2]* Dias, G. and Guilloré, S. and Lopes, J.G.P., "Mutual expectation: a measure for multiword lexical unit extraction," in *Proceedings of VExTAL Venezia per il Trattamento Automatico delle Lingue*, 1999.