Automatically Generating Extraction Patterns from Untagged Text

By Ellen Riloff

Presented by: Joe Dillman

What is Autoslog-TS?

- A System that generates extraction patterns from untagged text.
- Requires two corpora:
  - Relevant
  - Irrelevant
- Based on Autoslog which requires tagging

Related Work - Autoslog

- Autoslog requires tagging of text to identify NP's for extraction
- Uses linguistic patterns:

PATTERN	EXAMPLE
<subj> passive-verb	<victim> was murdered
<subj> active-verb	<perp> bombed
<subj> verb infin.	<perp> attempted to kill
<subj> aux noun	<victim> was victim
passive-verb <dobj>[1]	killed <victim>
active-verb <dobj>	bombed <target>
infin. <dobj>	to kill <victim>
verb infin. <dobj>	tried to attack <target>
gerund <dobj>	killing <victim>
noun aux <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>

Related Work - Autoslog

- Linguistic patterns use CIRCUS to extract *concept nodes*

Id: DEV-MUC4-0657 **Slot filler:** "public buildings"
Sentence: (in la oroya, junin department, in the central peruvian mountain range, public buildings were bombed and a car-bomb was detonated.)

CONCEPT NODE

Name:	target-subject-passive-verb-bombed
Trigger:	bombed
Variable Slots:	(target (*S* 1))
Constraints:	(class phys-target *S*)
Constant Slots:	(type bombing)
Enabling Conditions:	((passive))

Figure 1: A good concept node definition

Id: DEV-MUC4-1192 **Slot filler:** "gilberto molasco"
Sentence: (they took 2-year-old gilberto molasco, son of patricio rodriguez, and 17-year-old andres argueta, son of emimesto argueta.)

CONCEPT NODE

Name:	victim-active-verb-dobj-took
Trigger:	took
Variable Slots:	(victim (*DOBJ* 1))
Constraints:	(class victim *DOBJ*)
Constant Slots:	(type kidnapping)
Enabling Conditions:	((active))

Figure 3: A bad concept node definition

- Generates MANY bad definitions!

Related Work – Autoslog

- Compared to manually created dictionary

System/Test Set	Recall	Precision	F-measure
MUC-4/TST3	46	56	50.51
AutoSlog/TST3	43	56	48.65
MUC-4/TST4	44	40	41.90
AutoSlog/TST4	39	45	41.79

- Autoslog: 5 hours for review
- Manual: 1500 hours (2 grad students)!

- Tagging corpus is time consuming

Autoslog-TS

- Does not require tagging
- Automatically generates extraction patterns for every NP
- Adds two more heuristic patterns:
  - <subj> active-verb dobj
  - Infinitive prep <np>
- Two stages of processing

Autoslog-TS: Stage 1

- Syntactic parse to identify noun phrases
- Generates concept nodes
- Can generate multiple rules:
  - Example: “terrorists bombed the US embassy”
    - <subj> bombed
    - <subj> bombed embassy

- Compare these in stage 2 to determine which to keep

Autoslog-TS: Stage 2

- Apply rules from Stage 1 to corpus
- Compute relevance statistics
  - Conditional Probability (relevance rate):

$$\Pr(relevant\ text \mid text\ contains\ pattern_i) = \frac{rel-freq_i}{total-freq_i}$$

    - rel – $freq_i$ = # of instances of $pattern_i$ in relevant texts
    - total – $freq_i$ = # of instances of $pattern_i$ in training corpus

Autoslog-TS: Conditional probability

- Motivation: Domain-specific will substantially appear more often in relevant than irrelevant
  - Question: Does this make sense? How different do corpora need to be?
- Used for ranking function
  - *relevance rate * log_2(frequency)*
    - Exception for negative correlations: if relevance rate<=0.5, return 0
- Manual review of patterns still required

Results

- Autoslog:

Slot	Corr.	Miss.	Mislab.	Dup.	Spur.
Perp	36	22	1	11	129
Victim	41	24	7	18	113
Target	39	19	8	18	108
Total	116	65	16	47	350

- Autoslog-TS:

Slot	Corr.	Miss.	Mislab.	Dup.	Spur.
Perp	30	27	2	12	97
Victim	40	25	7	19	85
Target	32	23	17	16	58
Total	102	75	26	47	240

- Autoslog-TS reduced spurious extractions

Results compared to Autoslog

	AutoSlog			AutoSlog-TS		
Slot	Recall	Prec.	F	Recall	Prec.	F
Perp	.62	.27	.38	.53	.30	.38
Victim	.63	.33	.43	.62	.39	.48
Target	.67	.33	.44	.58	.39	.47
Total	.64	.31	.42	.58	.36	.44

- R = corr. / (corr. + miss.)
- P = (corr. + dup.) / (corr. + dup. + mislab. + spur.)
- Autoslog better recall
- Autoslog-TS better precision

Results

- Many unused (bad) patterns:
  - Generated: 1970
  - Kept: 210, ~10.7% (85 min. to review)
- But, Autoslog used 450 patterns
- Autoslog-TS gets comparable results with less rules

Concluding remarks

- Relevant to Irrelevant corpus
  - How irrelevant? Paper suggest “near misses”
- Manual intervention
  - Must manually select which patterns make sense
  - Time-consuming
  - May miss valuable patterns deeper in list
- Improvements on ranking function
- From empirical use, Autoslog-TS enables a “quick start” on a topic