Semantic Role Labeling

车万翔
社会计算与信息检索研究中心
2017年春季学期

1. Introduction
2. State-of-the-art
3. Empirical evaluation and lessons learned
4. Problems and challenges
5. Conclusions

Tutorial Overview

1. Introduction
   - Problem definition and properties
   - Main Computational Resources and Systems
2. State-of-the-art
3. Empirical evaluation and lessons learned
4. Problems and challenges
5. Conclusions

Tutorial Overview

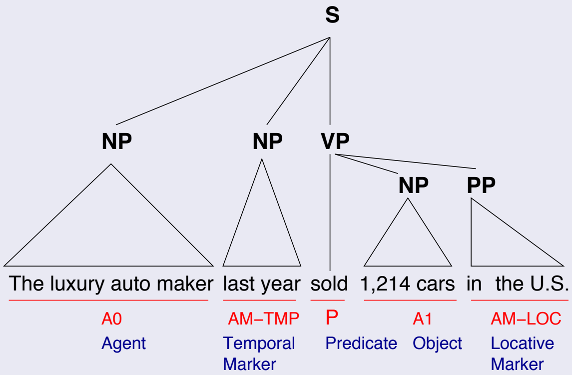
1. Introduction
   - Problem definition and properties
   - Main Computational Resources and Systems
2. State-of-the-art
3. Empirical evaluation and lessons learned
4. Problems and challenges
5. Conclusions

Semantic Role Labeling: The Problem

SRL $\stackrel{def}{=}$ detecting basic event structures such as *who* did *what* to *whom*, *when* and *where* [**IE** point of view]

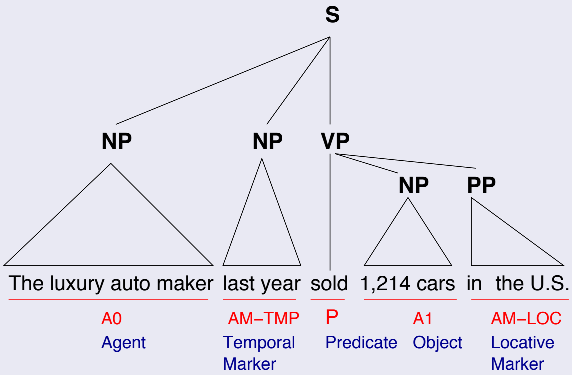
Semantic Role Labeling: The Problem

SRL $\stackrel{def}{=}$ detecting basic event structures such as *who* did *what* to *whom*, *when* and *where* [**IE** point of view]



Semantic Role Labeling: The Problem

SRL $\stackrel{def}{=}$ identify the *arguments* of a given verb and assign them semantic labels describing the *roles* they play in the predicate (i.e., identify predicate argument structures) [**CL** point of view]



Semantic Role Labeling: The Problem

Syntactic variations

$\overbrace{\text{Yesterday}}^{\text{TEMP}}$, $\overbrace{\text{Kristina}}^{\text{HITTER}}$ hit $\overbrace{\text{Scott}}^{\text{THING HIT}}$ $\overbrace{\text{with a baseball}}^{\text{INSTRUMENT}}$

- Scott was hit by Kristina yesterday with a baseball
- Yesterday, Scott was hit with a baseball by Kristina
- With a baseball, Kristina hit Scott yesterday
- Yesterday Scott was hit by Kristina with a baseball
- Kristina hit Scott with a baseball yesterday

Example from (Yih & Toutanova, 2006)

Semantic Role Labeling: The Problem

Syntactic variations

$\overbrace{\text{Yesterday}}^{\text{TEMP}}$, $\overbrace{\text{Kristina}}^{\text{HITTER}}$ hit $\overbrace{\text{Scott}}^{\text{THING HIT}}$ $\overbrace{\text{with a baseball}}^{\text{INSTRUMENT}}$

- Scott was hit by Kristina yesterday with a baseball
- Yesterday, Scott was hit with a baseball by Kristina
- With a baseball, Kristina hit Scott yesterday
- Yesterday Scott was hit by Kristina with a baseball
- Kristina hit Scott with a baseball yesterday

Example from (Yih & Toutanova, 2006)

Semantic Role Labeling: The Problem

Structural view

Mapping from input to output structures:

- **Input** is *text* (enriched with morpho-syntactic information)
- **Output** is a *sequence of labeled arguments*
- **Sequential** segmenting/labeling problem

" Mr. Smith sent the report to me this morning . "

[Mr. Smith]$_{AGENT}$ sent [the report]$_{OBJ}$ to [me]$_{RECIP}$ [this morning]$_{TMP}$.

Mr.$_{B-AGENT}$ Smith$_{I}$ sent the$_{B-OBJ}$ report$_{I}$ to$_{O}$ me$_{B-RECIP}$ this$_{B-TMP}$ morning$_{I}$.$_{O}$

Semantic Role Labeling: The Problem

Structural view

Mapping from input to output structures:

- **Input** is *text* (enriched with morpho-syntactic information)
- **Output** is a *sequence of labeled arguments*
- **Sequential** segmenting/labeling problem

" Mr. Smith sent the report to me this morning . "

[Mr. Smith]$_{AGENT}$ sent [the report]$_{OBJ}$ to [me]$_{RECIP}$ [this morning]$_{TMP}$.

Mr.$_{B-AGENT}$ Smith$_{I}$ sent the$_{B-OBJ}$ report$_{I}$ to$_{O}$ me$_{B-RECIP}$ this$_{B-TMP}$ morning$_{I}$.$_{O}$

Semantic Role Labeling: The Problem

Structural view

Mapping from input to output structures:

- **Input** is *text* (enriched with morpho-syntactic information)
- **Output** is a *sequence of labeled arguments*
- **Sequential** segmenting/labeling problem

" Mr. Smith sent the report to me this morning . "

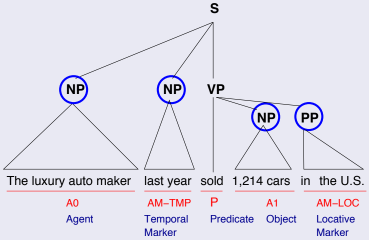
[Mr. Smith]$_{AGENT}$ sent [the report]$_{OBJ}$ to [me]$_{RECIP}$ [this morning]$_{TMP}$.

Mr.$_{B-AGENT}$ Smith$_{I}$ sent the$_{B-OBJ}$ report$_{I}$ to$_{O}$ me$_{B-RECIP}$ this$_{B-TMP}$ morning$_{I}$.$_{O}$

Semantic Role Labeling: The Problem

Linguistic nature of the problem

- Argument identification is strongly related to syntax

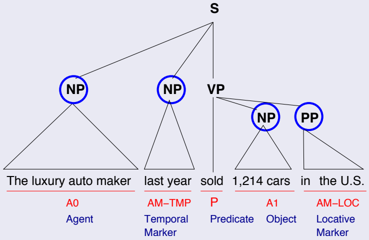


- Role labeling is a semantic task
  - e.g., selectional preferences should play an important role

Semantic Role Labeling: The Problem

Linguistic nature of the problem

- Argument identification is strongly related to syntax



- Role labeling is a semantic task
  - e.g., selectional preferences should play an important role

Semantic Role Labeling: Applications

Is SRL really useful for NLP applications?

1. Information Extraction (Surdeanu et al., 2003; Frank et al., 2007)
2. Question & Answering (Narayanan and Harabagiu, 2004)
3. Automatic Summarization (Melli et al., 2005)
4. Coreference Resolution (Ponzetto and Strube, 2006)
5. Machine Translation (Boas, 2002; Giménez and Màrquez, 2007; Wu and Fung, 2009a;2009b)
6. etc. [more on SRL and applications in the last section]

Semantic Role Labeling: Applications

Is SRL really useful for NLP applications?

1. Information Extraction (Surdeanu et al., 2003; Frank et al., 2007)
2. Question & Answering (Narayanan and Harabagiu, 2004)
3. Automatic Summarization (Melli et al., 2005)
4. Coreference Resolution (Ponzetto and Strube, 2006)
5. Machine Translation (Boas, 2002; Giménez and Màrquez, 2007; Wu and Fung, 2009a;2009b)
6. etc. [more on SRL and applications in the last section]

Tutorial Overview

1. Introduction
   - Problem definition and properties
   - Main Computational Resources and Systems
2. State-of-the-art
3. Empirical evaluation and lessons learned
4. Problems and challenges
5. Conclusions

SRL: Computational Resources

From theory to computational resources

- Since (Fillmore, 1968), considerable linguistic research has been devoted to the nature of semantic roles
- Two broad families exist:
  1. Syntax-based approach : explaining the varied expression of verb arguments within syntactic positions : Levin (1993) verb classes ⟹ VerbNet (Kipper et al., 2000) ⟹ PropBank (Palmer et al., 2005) : Focused on verbs
  2. Situation-based approach (a word activates/invokes a frame of semantic knowledge that relates linguistic semantics to encyclopedic knowledge) : Frame semantics (Fillmore, 1976) ⟹ FrameNet (Fillmore et al., 2004) : Words with other POS can invoke frames too (e.g., nouns, adjectives)

SRL: Computational Resources

From theory to computational resources

- Since (Fillmore, 1968), considerable linguistic research has been devoted to the nature of semantic roles
- Two broad families exist:

  1. Syntax-based approach : explaining the varied expression of verb arguments within syntactic positions : Levin (1993) verb classes ⟹ VerbNet (Kipper et al., 2000) ⟹ PropBank (Palmer et al., 2005) : Focused on verbs

  2. Situation-based approach (a word activates/invokes a frame of semantic knowledge that relates linguistic semantics to encyclopedic knowledge) : Frame semantics (Fillmore, 1976) ⟹ FrameNet (Fillmore et al., 2004) : Words with other POS can invoke frames too (e.g., nouns, adjectives)

SRL: Computational Resources

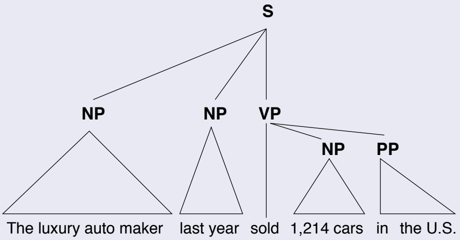
From theory to computational resources

- Since (Fillmore, 1968), considerable linguistic research has been devoted to the nature of semantic roles
- Two broad families exist:
  1. Syntax-based approach : explaining the varied expression of verb arguments within syntactic positions : Levin (1993) verb classes $\Longrightarrow$ VerbNet (Kipper et al., 2000) $\Longrightarrow$ PropBank (Palmer et al., 2005) : Focused on verbs
  2. Situation-based approach (a word activates/invokes a frame of semantic knowledge that relates linguistic semantics to encyclopedic knowledge) : Frame semantics (Fillmore, 1976) $\Longrightarrow$ FrameNet (Fillmore et al., 2004) : Words with other POS can invoke frames too (e.g., nouns, adjectives)

Semantic Role Labeling: Corpora

PropBank (Palmer et al., 2005)

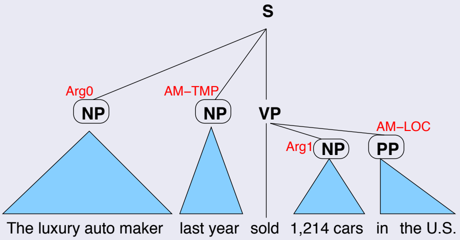
- Annotation of all verbal predicates in WSJ (Penn Treebank)
- http://verbs.colorado.edu/∼mpalmer/projects/ace.html
- Add a semantic layer to the Syntactic Trees



Semantic Role Labeling: Corpora

PropBank (Palmer et al., 2005)

- Annotation of all verbal predicates in WSJ (Penn Treebank)
- http://verbs.colorado.edu/∼mpalmer/projects/ace.html
- Add a semantic layer to the Syntactic Trees



Semantic Role Labeling: Corpora

PropBank (Palmer et al., 2005)

- Theory neutral numeric core roles (Arg0, Arg1, etc.)
  - Interpretation of roles: verb-specific framesets
  - Arg0 and Arg1 usually correspond to prototypical Agent and Patient/Theme roles. Other arguments do not consistently generalize across verbs
  - Different senses have different framesets
  - Syntactic alternations that preserve meaning are kept toghether in a single frameset
- Closed set of 13 general labels for Adjuncts (e.g., Temporal, Manner, Location, etc.)

Semantic Role Labeling: Corpora

PropBank (Palmer et al., 2005)

- Theory neutral numeric core roles (Arg0, Arg1, etc.)
  - Interpretation of roles: verb-specific framesets
  - Arg0 and Arg1 usually correspond to prototypical Agent and Patient/Theme roles. Other arguments do not consistently generalize across verbs
  - Different senses have different framesets
  - Syntactic alternations that preserve meaning are kept toghether in a single frameset
- Closed set of 13 general labels for Adjuncts (e.g., Temporal, Manner, Location, etc.)

Semantic Role Labeling: Corpora

PropBank (Palmer et al., 2005)

- Theory neutral numeric core roles (Arg0, Arg1, etc.)
  - Interpretation of roles: verb-specific framesets
  - Arg0 and Arg1 usually correspond to prototypical Agent and Patient/Theme roles. Other arguments do not consistently generalize across verbs
  - Different senses have different framesets
  - Syntactic alternations that preserve meaning are kept toghether in a single frameset
- Closed set of 13 general labels for Adjuncts (e.g., Temporal, Manner, Location, etc.)

Semantic Role Labeling: Corpora

PropBank: Frame files (Palmer et al., 2005)

- sell.01: commerce: seller
  Arg0="seller" (*agent*); Arg1="thing sold" (*theme*); Arg2="buyer" (*recipient*); Arg3="price paid"; Arg4="benefactive"
  [Al Brownstein]$_{Arg0}$ sold [it]$_{Arg1}$ [for $60 a bottle]$_{Arg3}$
- sell.02: give up
  Arg0="entity selling out"
  [John]$_{Arg0}$ sold out
- sell.03: sell until none is/are left
  Arg0="seller"; Arg1="thing sold"; ...
  [The new Harry Potter]$_{Arg1}$ sold out [within 20 minutes]$_{ArgM-TMP}$

Semantic Role Labeling: Corpora

PropBank: Frame files (Palmer et al., 2005)

- sell.01: commerce: seller
  Arg0="seller" (*agent*); Arg1="thing sold" (*theme*); Arg2="buyer" (*recipient*); Arg3="price paid"; Arg4="benefactive"
  [Al Brownstein]$_{Arg0}$ sold [it]$_{Arg1}$ [for $60 a bottle]$_{Arg3}$
- sell.02: give up
  Arg0="entity selling out"
  [John]$_{Arg0}$ sold out
- sell.03: sell until none is/are left
  Arg0="seller"; Arg1="thing sold"; ...
  [The new Harry Potter]$_{Arg1}$ sold out [within 20 minutes]$_{ArgM-TMP}$

Semantic Role Labeling: Corpora

PropBank: Frame files (Palmer et al., 2005)

- sell.01: commerce: seller
  Arg0="seller" (*agent*); Arg1="thing sold" (*theme*); Arg2="buyer" (*recipient*); Arg3="price paid"; Arg4="benefactive"
  [Al Brownstein]$_{Arg0}$ sold [it]$_{Arg1}$ [for $60 a bottle]$_{Arg3}$
- sell.02: give up
  Arg0="entity selling out"
  [John]$_{Arg0}$ sold out
- sell.03: sell until none is/are left
  Arg0="seller"; Arg1="thing sold"; ...
  [The new Harry Potter]$_{Arg1}$ sold out [within 20 minutes]$_{ArgM-TMP}$

Semantic Role Labeling: Corpora

PropBank: Frame files (Palmer et al., 2005)

- sell.01: commerce: seller
  Arg0="seller" (*agent*); Arg1="thing sold" (*theme*); Arg2="buyer" (*recipient*); Arg3="price paid"; Arg4="benefactive"
  [Al Brownstein]$_{Arg0}$ sold [it]$_{Arg1}$ [for $60 a bottle]$_{Arg3}$
- sell.02: give up
  Arg0="entity selling out"
  [John]$_{Arg0}$ sold out
- sell.03: sell until none is/are left
  Arg0="seller"; Arg1="thing sold"; ...
  [The new Harry Potter]$_{Arg1}$ sold out [within 20 minutes]$_{ArgM-TMP}$

Semantic Role Labeling: Corpora

NomBank (Meyers et al., 2004)

- NomBank Project: http://nlp.cs.nyu.edu/meyers/NomBank.html
- Annotation of the nominal predicates in WSJ–PennTreeBank

  IBM appointed John
  John was appointed by IBM
  IBM's appointment of John
  The appointment of John by IBM
  John is the current IBM appointee

- Annotation similar to PropBank

  [Her]$_{Arg0}$ gift of [a book]$_{Arg1}$ [to John]$_{Arg2}$

Semantic Role Labeling: Corpora

NomBank (Meyers et al., 2004)

- NomBank Project: http://nlp.cs.nyu.edu/meyers/NomBank.html
- Annotation of the nominal predicates in WSJ–PennTreeBank

  IBM appointed John
  John was appointed by IBM
  IBM's appointment of John
  The appointment of John by IBM
  John is the current IBM appointee

- Annotation similar to PropBank

  [Her]$_{Arg0}$ gift of [a book]$_{Arg1}$ [to John]$_{Arg2}$

Semantic Role Labeling: Corpora

Languages other than English

- Chinese PropBank
  http://verbs.colorado.edu/chinese/cpb/
- Korean PropBank
  http://www.ldc.upenn.edu/
- AnCora corpus: Spanish and Catalan
  http://http://clic.ub.edu/ancora/
- Prague Dependency Treebank: Czech
  http://ufal.mff.cuni.cz/pdt2.0/
- Penn Arabic TreeBank: Arabic
  http://www.ircs.upenn.edu/arabic/
- Others are under development, e.g., Scandinavian and Baltic languages

Semantic Role Labeling: Corpora

Other extensions

- FrameNet for German (SALSA corpus), Spanish and Japanese
- OntoNotes corpus: TreeBank + PropBank + word senses + coreference annotation
  http://www.bbn.com/NLP/OntoNotes
- CoNLL–2008 shared task: joint representation for syntactic and semantic dependencies
  http://www.yr-bcn.es/conll2008/
- CoNLL–2009 shared task: extension to multiple languages (Catalan, Chinese, Czech, English, German, Japanese, Spanish)
  http://ufal.mff.cuni.cz/conll2009-st/

Semantic Role Labeling: Systems Available

Tools available online that produce SRL structures

- ASSERT (Automatic Statistical SEmantic Role Tagger)
  http://cemantix.org/assert
- UIUC system
  `http://l2r.cs.uiuc.edu/~cogcomp/srl-demo.php`
- SwiRL
  http://www.surdeanu.name/mihai
- Shalmaneser: FrameNet-based system from SALSA project
  http://www.coli.uni-saarland.de/projects/salsa/shal/

Tutorial Overview

1 Introduction

2 State-of-the-art
- Architecture
- Feature engineering
- SRL systems in detail

3 Empirical evaluation and lessons learned

4 Problems and challenges

5 Conclusions

Tutorial Overview

1. Introduction
2. State-of-the-art
   - Architecture
   - Feature engineering
   - SRL systems in detail
3. Empirical evaluation and lessons learned
4. Problems and challenges
5. Conclusions

SRL: Step by Step

The Problem

- Given a sentence and a designated predicate p
- Every subsequence of words (not necessarily contiguous) is a potential argument of p
- Arguments can be discontinuous:
- SRL can be formalized as a mapping from word substrings to the set of argument labels plus 'non-argument'
- This is clearly impractical. We need to filter the set of candidates...

SRL: Step by Step

The Problem

- Given a sentence and a designated predicate p
- Every subsequence of words (not necessarily contiguous) is a potential argument of p
- Arguments can be discontinuous:
  "One troubling aspect of DEC's results, analysts said, was its performance in Europe"
- SRL can be formalized as a mapping from word substrings to the set of argument labels plus 'non-argument'
- This is clearly impractical. We need to filter the set of candidates...

SRL: Step by Step

The Problem

- Given a sentence and a designated predicate p
- Every subsequence of words (not necessarily contiguous) is a potential argument of p
- Arguments can be discontinuous:
  [One troubling aspect of DEC's results]$_{Arg1}$, [analysts]$_{Arg0}$ said, [was its performance in Europe]$_{C-Arg1}$.
- SRL can be formalized as a mapping from word substrings to the set of argument labels plus 'non-argument'
- This is clearly impractical. We need to filter the set of candidates...

SRL: Step by Step

The Problem

- Given a sentence and a designated predicate *p*
- Every subsequence of words (not necessarily contiguous) is a potential argument of *p*
- Arguments can be discontinuous:
  [One troubling aspect of DEC's results]$_{Arg1}$, [analysts]$_{Arg0}$ said, [was its performance in Europe]$_{C-Arg1}$.
- SRL can be formalized as a mapping from word substrings to the set of argument labels plus 'non-argument'
- This is clearly impractical. We need to filter the set of candidates...

SRL: Step by Step

Step 1: Select argument candidates

- Given a sentence and a designated predicate
- Parse the sentence
- Identify candidates in tree constituents (filtering/pruning)
  - Simple heuristic rules can be used, which maintain a high recall (Xue & Palmer, 2004)
- Key point: 95% of semantic arguments coincide with unique syntactic constituents in the gold parse tree (PropBank)
  - Matching is still ∼90% when using automatic parsers

SRL: Step by Step

Step 1: Select argument candidates

- Given a sentence and a designated predicate
- Parse the sentence
- Identify candidates in tree constituents (filtering/pruning)
    - Simple heuristic rules can be used, which maintain a high recall (Xue & Palmer, 2004)
- Key point: 95% of semantic arguments coincide with unique syntactic constituents in the gold parse tree (PropBank)
    - Matching is still ∼90% when using automatic parsers

SRL: Step by Step

Step 2: Local scoring of candidates

- Apply classifiers to assign confidence scores to argument candidates (all labels + 'non-argument')
- Candidates are treated independently of each other
- *Identification* and *Classification* may be performed separately
  - Computational reasons but also modularity in feature engineering
- Many ML paradigms have been used: not big differences
- Features are more important

SRL: Step by Step

Step 2: Local scoring of candidates

- Apply classifiers to assign confidence scores to argument candidates (all labels + ‘non-argument’)
- Candidates are treated independently of each other
- *Identification* and *Classification* may be performed separately
  - Computational reasons but also modularity in feature engineering
- Many ML paradigms have been used: not big differences
- Features are more important

SRL: Step by Step

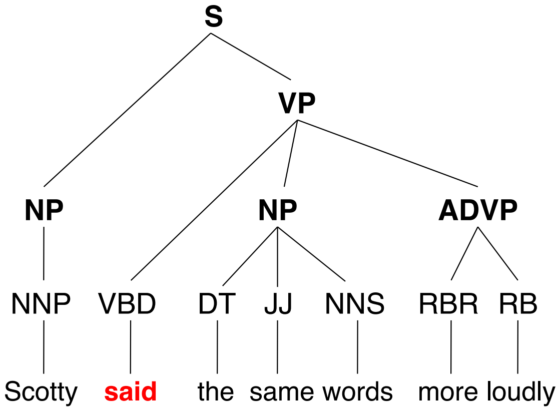
Step 2: Local scoring of candidates

- Apply classifiers to assign confidence scores to argument candidates (all labels + 'non-argument')
- Candidates are treated independently of each other
- *Identification* and *Classification* may be performed separately
  - Computational reasons but also modularity in feature engineering
- Many ML paradigms have been used: not big differences
- Features are more important

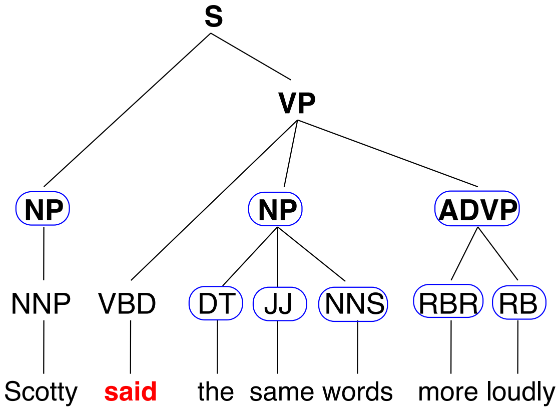
SRL: Steps 1 + 2

Scotty **said** the same words more loudly

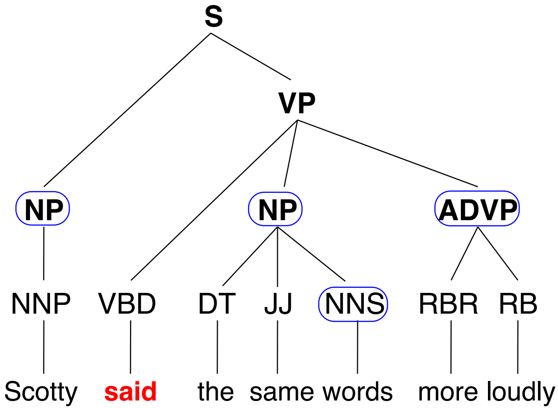
SRL: Steps 1 + 2


S
VP
NP
NP
ADVP
NNP
VBD
DT
JJ
NNS
RBR
RB
Scotty
said
the
same
words
more
loudly

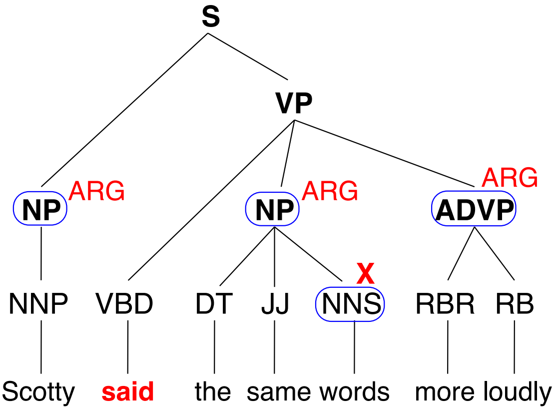
SRL: Steps 1 + 2


S
VP
NP
NP
ADVP
NNP
VBD
DT
JJ
NNS
RBR
RB
Scotty
said
the
same
words
more
loudly

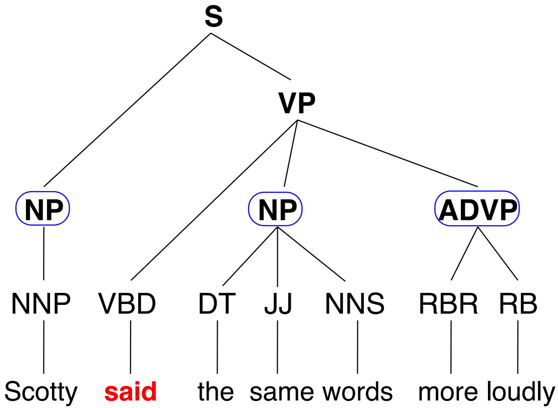
SRL: Steps 1 + 2


S
VP
NP
NP
ADVP
NNP
VBD
DT
JJ
NNS
RBR
RB
Scotty
said
the
same
words
more
loudly

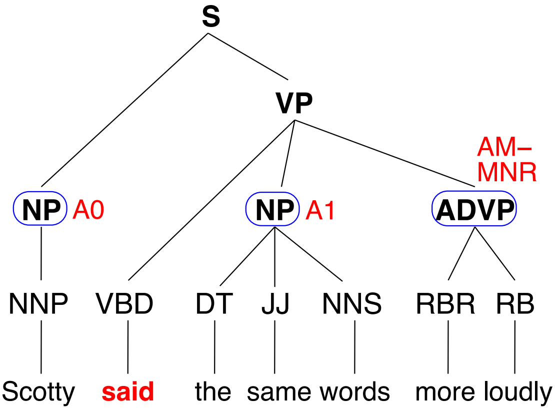
SRL: Steps 1 + 2


S
VP
NP
ARG
NP
ARG
ARG
ADVP
X
NNP
VBD
DT
JJ
NNS
RBR
RB
Scotty
said
the
same
words
more
loudly

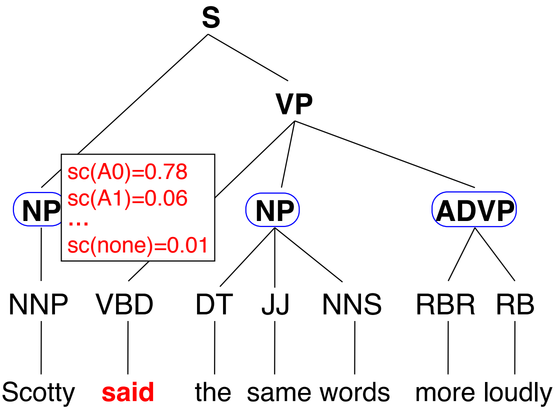
SRL: Steps 1 + 2


S
VP
NP
NP
ADVP
NNP
VBD
DT
JJ
NNS
RBR
RB
Scotty
said
the
same
words
more
loudly

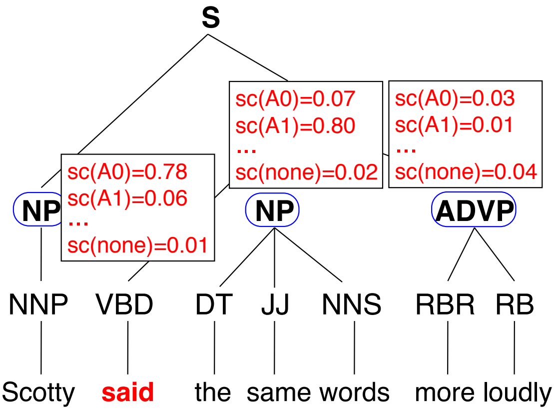
SRL: Steps 1 + 2


S
VP
NP
A0
NP
A1
AM–MNR
ADVP
NNP
VBD
DT
JJ
NNS
RBR
RB
Scotty
said
the
same
words
more
loudly

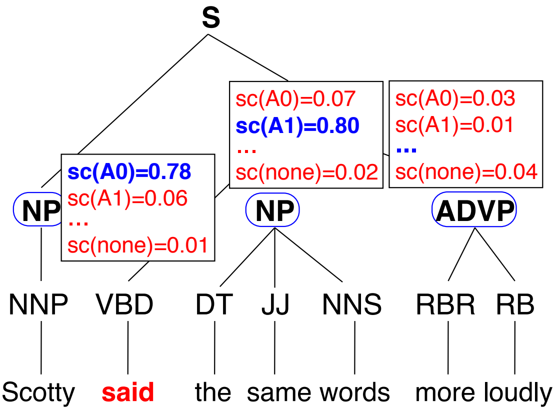
SRL: Motivating next step (joint scoring)



SRL: Motivating next step (joint scoring)

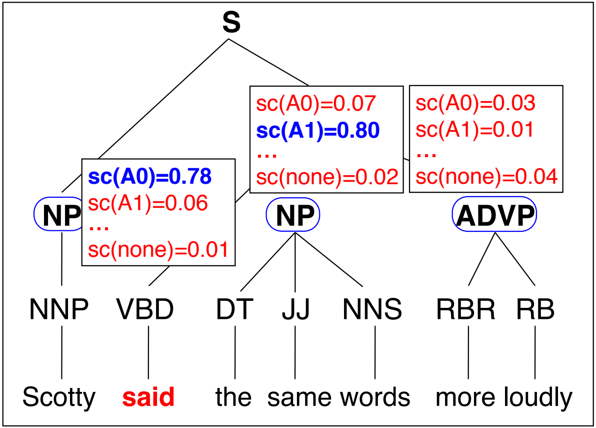


SRL: Motivating next step (joint scoring)

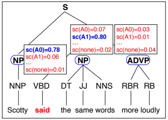


SRL: Motivating next step (joint scoring)

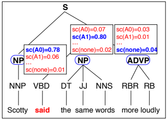
Global Score = 0.30



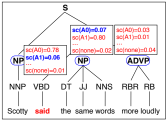
SRL: Motivating next step (joint scoring)



GS = 0.45

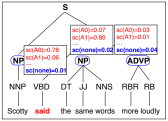


GS = 0.15



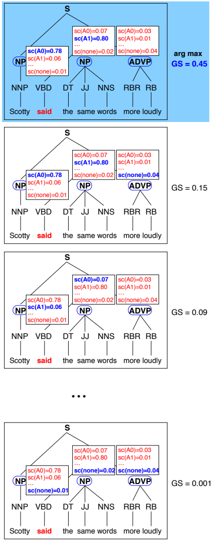
GS = 0.09

...



GS = 0.001

SRL: Motivating next step (joint scoring)



SRL: Step by Step

Step 3: Joint scoring — Paradigmatic examples

- Combine local predictions through ILP to find the best solution according to structural and linguistic constraints
  (Koomen et al., 2005; Punyakanok et al., 2008)
  −learning +dependencies +search
- Re-ranking of several candidate solutions
  (Haghighi et al., 2005; Toutanova et al., 2008)
  +learning +dependencies −search
- Global search integrating joint scoring: Tree CRFs
  (Cohn & Blunsom, 2005)
  +learning +/−dependencies +/−search

SRL: Step by Step

Step 3: Joint scoring — Paradigmatic examples

- Combine local predictions through ILP to find the best solution according to structural and linguistic constraints (Koomen et al., 2005; Punyakanok et al., 2008)

  –learning +dependencies +search

- Re-ranking of several candidate solutions (Haghighi et al., 2005; Toutanova et al., 2008)

  +learning +dependencies –search

- Global search integrating joint scoring: Tree CRFs (Cohn & Blunsom, 2005)

  +learning +/–dependencies +/–search

SRL: Step by Step

Step 4: Post-processing

- Application of a set of heuristic rules to:

  Correct frequent errors

  Enforce consistency in the solution

Tutorial Overview

1 Introduction

2 State-of-the-art
- Architecture
- Feature engineering
- SRL systems in detail

3 Empirical evaluation and lessons learned

4 Problems and challenges

5 Conclusions

SRL: Feature Engineering

Features: local scoring (Gildea & Jurafsky, 2002)

- Highly influential for the SRL work.
  They characterize:
  1. The candidate argument (constituent) and its context: phrase type, head word, governing category of the constituent
  2. The verb predicate and its context: lemma, voice, subcategorization pattern of the verb
  3. The relation between the consituent and the predicate: position of the constituent with respect to the verb, category path between them.

SRL: Feature Engineering

Features: local scoring (Gildea & Jurafsky, 2002)

- Highly influential for the SRL work.
  They characterize:
  1. The candidate argument (constituent) and its context: phrase type, head word, governing category of the constituent
  2. The verb predicate and its context: lemma, voice, subcategorization pattern of the verb
  3. The relation between the consituent and the predicate: position of the constituent with respect to the verb, category path between them.

SRL: Feature Engineering

Features: local scoring (Gildea & Jurafsky, 2002)

- Highly influential for the SRL work.
  They characterize:
  1. The candidate argument (constituent) and its context: phrase type, head word, governing category of the constituent
  2. The verb predicate and its context: lemma, voice, subcategorization pattern of the verb
  3. The relation between the consituent and the predicate: position of the constituent with respect to the verb, category path between them.

SRL: Feature Engineering

Features: local scoring (Gildea & Jurafsky, 2002)

- Highly influential for the SRL work.
  They characterize:
  1. The candidate argument (constituent) and its context: phrase type, head word, governing category of the constituent
  2. The verb predicate and its context: lemma, voice, subcategorization pattern of the verb
  3. The relation between the consituent and the predicate: position of the constituent with respect to the verb, category path between them.

SRL: Feature Engineering

Features: local scoring — extensions

- "Brute force" features. Applied to the constituent and possibly to parent and siblings:
  - First and last words/POS in the constituent, bag-of-words, *n*-grams of POS, and sequence of top syntactic elements in the constituent.
- Linguistically–inspired features
  - Content word, named entities (Surdeanu et al., 2003), syntactic frame (Xue & Palmer, 2004), path variations, semantic compatibility between constituent head and predicate (Zapirain et al., 2007;2009), etc.
- Significant (and cumulative) increase in performance

SRL: Feature Engineering

Features: local scoring — extensions

- "Brute force" features. Applied to the constituent and possibly to parent and siblings:
  - First and last words/POS in the constituent, bag-of-words, *n*-grams of POS, and sequence of top syntactic elements in the constituent.
- Linguistically–inspired features
  - Content word, named entities (Surdeanu et al., 2003), syntactic frame (Xue & Palmer, 2004), path variations, semantic compatibility between constituent head and predicate (Zapirain et al., 2007;2009), etc.
- Significant (and cumulative) increase in performance

SRL: Feature Engineering

Features: local scoring — extensions

- "Brute force" features. Applied to the constituent and possibly to parent and siblings:
  - First and last words/POS in the constituent, bag-of-words, *n*-grams of POS, and sequence of top syntactic elements in the constituent.
- Linguistically–inspired features
  - Content word, named entities (Surdeanu et al., 2003), syntactic frame (Xue & Palmer, 2004), path variations, semantic compatibility between constituent head and predicate (Zapirain et al., 2007;2009), etc.
- Significant (and cumulative) increase in performance

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Inference

- The output of the argument classifier often violates some constraints, especially when the sentence is long
- Finding the best legitimate output is formalized as an optimization problem and solved via Integer Linear Programming (Roth & Yih, 2004)
- Input formed by:
  - The probability estimation (by the argument classifier)
  - Structural and linguistic constraints
- Allows incorporating expressive constraints (non-sequential) on the variables (the arguments types)

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Inference

- The output of the argument classifier often violates some constraints, especially when the sentence is long
- Finding the best legitimate output is formalized as an optimization problem and solved via Integer Linear Programming (Roth & Yih, 2004)
- Input formed by:
  - The probability estimation (by the argument classifier)
  - Structural and linguistic constraints
- Allows incorporating expressive constraints (non-sequential) on the variables (the arguments types)

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Inference

- The output of the argument classifier often violates some constraints, especially when the sentence is long
- Finding the best legitimate output is formalized as an optimization problem and solved via Integer Linear Programming (Roth & Yih, 2004)
- Input formed by:
  - The probability estimation (by the argument classifier)
  - Structural and linguistic constraints
- Allows incorporating expressive constraints (non-sequential) on the variables (the arguments types)

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Inference

- The output of the argument classifier often violates some constraints, especially when the sentence is long
- Finding the best legitimate output is formalized as an optimization problem and solved via Integer Linear Programming (Roth & Yih, 2004)
- Input formed by:
    - The probability estimation (by the argument classifier)
    - Structural and linguistic constraints
- Allows incorporating expressive constraints (non-sequential) on the variables (the arguments types)

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Integer Linear Programming Inference

- For each candidate argument a_i $(1 \leq i \leq n)$,
  Set up a Boolean variable: $a_{i,t}$ indicating whether a_i is classified as argument type t
- Goal is to maximize: $\sum_i \mathrm{score}(a_i = t) \cdot a_{i,t}$
  Subject to the (linear) constraints
- If $\mathrm{score}(a_i = t) = P(a_i = t)$, the objective is to find the assignment that maximizes the expected number of arguments that are correct and satisfies the constraints

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Constraints: examples

- No duplicate argument classes: $\sum_{i=1}^{n} a_{i,Arg0} \leq 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leq j \leq n), \sum_{i=1}^{j-1} a_{i,Arg0} \geq a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j (1 \leq j \leq n), \sum_{i \neq j} a_{i,Arg0} \geq a_{j,R-Arg0}$
- Many other possible constraints:
  - Unique labels
  - No overlapping or embedding
  - Relations between number of arguments; order constraints
  - If verb is of type A, no argument of type B
- ILP inference can be used to combine different SRL systems

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Constraints: examples

- No duplicate argument classes: $\sum_{i=1}^{n} a_{i,Arg0} \leq 1$
- On discontinuous arguments (C-ARG)
  $\forall j(1 \leq j \leq n), \sum_{i=1}^{j-1} a_{i,Arg0} \geq a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j(1 \leq j \leq n), \sum_{i \neq j} a_{i,Arg0} \geq a_{j,R-Arg0}$
- Many other possible constraints:
  - Unique labels
  - No overlapping or embedding
  - Relations between number of arguments; order constraints
  - If verb is of type A, no argument of type B
- ILP inference can be used to combine different SRL systems

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Constraints: examples

- No duplicate argument classes: $\sum_{i=1}^{n} a_{i,Arg0} \leq 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leq j \leq n),\ \sum_{i=1}^{j-1} a_{i,Arg0} \geq a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j (1 \leq j \leq n),\ \sum_{i \neq j} a_{i,Arg0} \geq a_{j,R-Arg0}$
- Many other possible constraints:
  - Unique labels
  - No overlapping or embedding
  - Relations between number of arguments; order constraints
  - If verb is of type A, no argument of type B
- ILP inference can be used to combine different SRL systems

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Constraints: examples

- No duplicate argument classes: $\sum_{i=1}^{n} a_{i,Arg0} \leq 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leq j \leq n), \sum_{i=1}^{j-1} a_{i,Arg0} \geq a_{j,C-Arg0}$
- On reference arguments (R-ARG)

  [The deregulation]$_{Arg1}$ of railroads and trucking companies [that]$_{R-Arg1}$ began [in 1980]$_{AM-TMP}$ enabled ...

  $\forall j (1 \leq j \leq n), \sum_{i \neq j} a_{i,Arg0} \geq a_{j,R-Arg0}$
- Many other possible constraints:
  - Unique labels
  - No overlapping or embedding
  - Relations between number of arguments; order constraints
  - If verb is of type A, no argument of type B
- ILP inference can be used to combine different SRL systems

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Constraints: examples

- No duplicate argument classes: $\sum_{i=1}^{n} a_{i,Arg0} \leq 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leq j \leq n),\ \sum_{i=1}^{j-1} a_{i,Arg0} \geq a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j (1 \leq j \leq n),\ \sum_{i \neq j} a_{i,Arg0} \geq a_{j,R-Arg0}$
- Many other possible constraints:
  - Unique labels
  - No overlapping or embedding
  - Relations between number of arguments; order constraints
  - If verb is of type A, no argument of type B
- ILP inference can be used to combine different SRL systems

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Constraints: examples

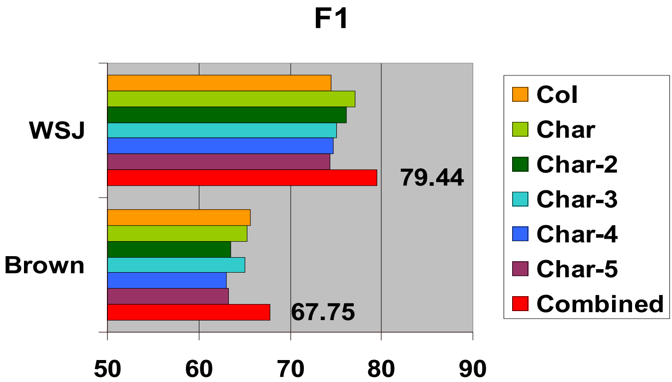
- No duplicate argument classes: $\sum_{i=1}^{n} a_{i,Arg0} \leq 1$
- On discontinuous arguments (C-ARG)
  $\forall j(1 \leq j \leq n),\ \sum_{i=1}^{j-1} a_{i,Arg0} \geq a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j(1 \leq j \leq n),\ \sum_{i \neq j} a_{i,Arg0} \geq a_{j,R-Arg0}$
- Many other possible constraints:
  - Unique labels
  - No overlapping or embedding
  - Relations between number of arguments; order constraints
  - If verb is of type A, no argument of type B
- ILP inference can be used to combine different SRL systems

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)

Constraints: examples

- No duplicate argument classes: $\sum_{i=1}^{n} a_{i,Arg0} \leq 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leq j \leq n),\ \sum_{i=1}^{j-1} a_{i,Arg0} \geq a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j (1 \leq j \leq n),\ \sum_{i \neq j} a_{i,Arg0} \geq a_{j,R-Arg0}$
- Many other possible constraints:
  - Unique labels
  - No overlapping or embedding
  - Relations between number of arguments; order constraints
  - If verb is of type A, no argument of type B
- ILP inference can be used to combine different SRL systems

Generalized Inference – ILP (Koomen et al., 2005; Punyakanok et al., 2008)



- Inference with many parsers improves results ~ 2.6 F_1 points
- Best results at CoNLL-2005 shared task (Carreras & Màrquez, 2005)

Tutorial Overview

1. Introduction
2. State-of-the-art
3. Empirical evaluation and lessons learned
4. Problems and challenges
5. Conclusions

Empirical Evaluation of SRL Systems

Evaluation Exercises

- Up to 9 evaluation exercises in the last 5 years
  - CoNLL-2004/2005 shared tasks (Carreras & Màrquez, 2004; 2005)
  - Senseval–3 (Litkowski, 2004)
  - SemEval-2007 (Pradhan et al., 2007; Màrquez et al., 2007) (Baker et al., 2007; Litkowski & Hargraves, 2007)
  - CoNLL-2008 shared task (Surdeanu et al., 2008)
  - CoNLL-2009 shared task (Hajič et al., 2009)

Tutorial Overview

1. Introduction
2. State-of-the-art
3. Empirical evaluation and lessons learned
4. Problems and challenges
   - Generalization to new Domains
   - Dependence on Syntax
   - SRL systems in applications
5. Conclusions

Tutorial Overview

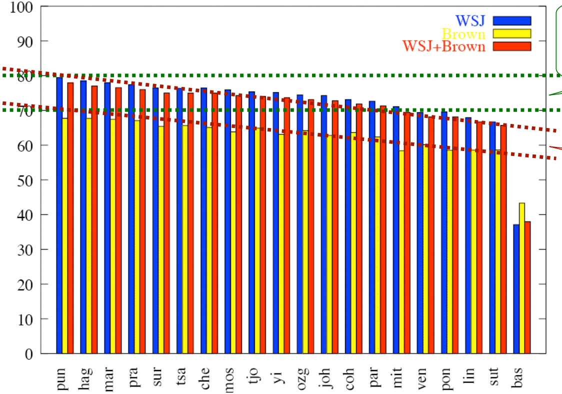
1. Introduction
2. State-of-the-art
3. Empirical evaluation and lessons learned
4. Problems and challenges
   - Generalization to new Domains
   - Dependence on Syntax
   - SRL systems in applications
5. Conclusions

Domain Dependence

- All statistical ML systems suffer from domain dependence
- How large is this dependence in the case of SRL?
- CoNLL-2005 evaluation: out-of-domain test corpus (Brown) $\Longrightarrow \sim$10 F_1 point drop in performance
- Similar evaluations at CoNLL-2008/2009 shared tasks

Domain Dependence: CoNLL-2005

Results on WSJ and Brown Tests



F_1: 70% ~ 80%
Small differences

Every system suffers from cross-domain test (~10%)

Tutorial Overview

1. Introduction
2. State-of-the-art
3. Empirical evaluation and lessons learned
4. Problems and challenges
   - Generalization to new Domains
   - Dependence on Syntax
   - SRL systems in applications
5. Conclusions

Impact of Syntactic Processing in SRL

- SRL results strongly depend on syntax (bottleneck)
- Gold vs. automatic parses: ∼90% vs. ∼80% F_1
- Drop in performance occurs in identifying argument boundaries

Tutorial Overview

1. Introduction
2. State-of-the-art
3. Empirical evaluation and lessons learned
4. Problems and challenges
   - Generalization to new Domains
   - Dependence on Syntax
   - SRL systems in applications
5. Conclusions

SRL in Applications

Examples of applications of SRL

- Information Extraction (Surdeanu et al., 2003)
- Question & Answering (Narayanan and Harabagiu, 2004; Frank et al., 2007)
- Automatic Summarization (Melli et al., 2005)
- Coreference Resolution (Ponzetto and Strube, 2006)
- See (Yih & Toutanova, 2006) tutorial for a discussion on all previous works

SRL in Applications

Other applications of SRL

- Machine Translation Evaluation
  (Giménez and Màrquez, 2007)
- Machine Translation
  (Boas, 2002; Wu and Fung, 2009a;2009b)
- Textual Entailment
  (Tatu & Moldovan, 2005; Burchardt et al., 2007)
- Modeling Early Language Acquisition (Connor et al., 2008;2009)
- Pictorial Communication Systems (Goldberg, et al., 2008)
- ...
- We will concentrate on Machine Translation

Tutorial Overview

1 Introduction

2 State-of-the-art

3 Empirical evaluation and lessons learned

4 Problems and challenges

5 Conclusions

General Conclusions

- SRL is an important problem in NLP with strong connections to applications requiring some degree of semantic interpretation
- It is a very active topic of research, which has generated an important body of work in the last 6 years
- Some news are good but...
- SRL still has to face important challenges before we see systems in real open-domain applications
- Good opportunities for future research on the topic