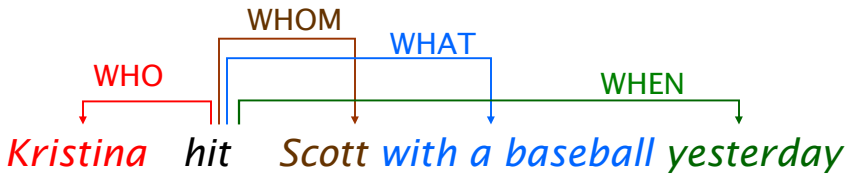
Automatic Semantic Role Labeling

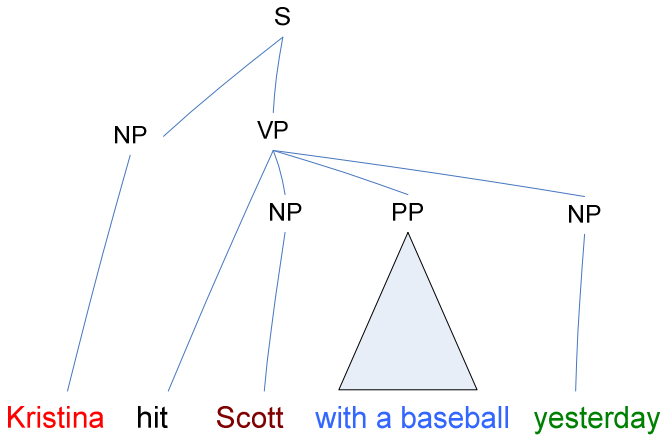
Scott Wen-tau Yih Kristina Toutanova
Microsoft Research

Natural Language Understanding
Question Answering

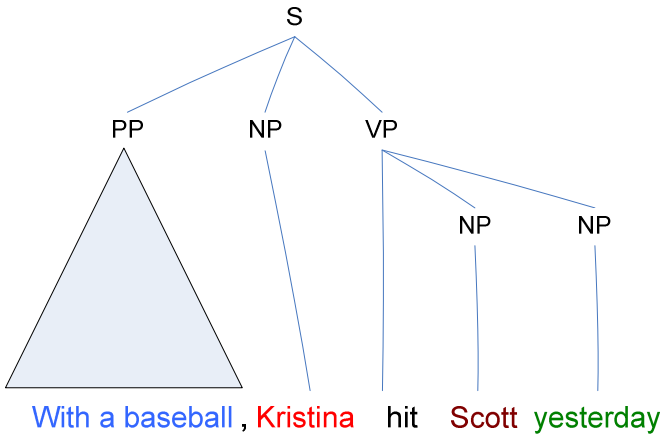


- Who hit Scott with a baseball?
- Whom did Kristina hit with a baseball?
- What did Kristina hit Scott with?
- When did Kristina hit Scott with a baseball?

Syntactic Analysis (1/2)



Syntactic Analysis (2/2)



Syntactic Variations

Yesterday, Kristina hit Scott with a baseball

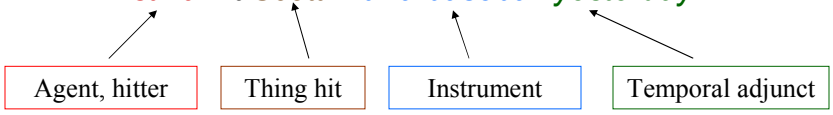
Scott was hit by Kristina yesterday with a baseball

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



Semantic Role Labeling – *Giving Semantic Labels to Phrases*

- [$_{AGENT}$ John] **broke** [$_{THEME}$ the window]
- [$_{THEME}$ The window] **broke**
- [$_{AGENT}$Sotheby's] .. **offered** [$_{RECIPIENT}$ the Dorrance heirs] [$_{THEME}$ a money-back guarantee]
- [$_{AGENT}$ Sotheby's] **offered** [$_{THEME}$ a money-back guarantee] to [$_{RECIPIENT}$ the Dorrance heirs]
- [$_{THEME}$ a money-back guarantee] **offered** by [$_{AGENT}$ Sotheby's]
- [$_{RECIPIENT}$ the Dorrance heirs] will [$_{ARM-NEG}$ not] be **offered** [$_{THEME}$ a money-back guarantee]

Why is SRL Important – *Applications*

- Question Answering
  - Q: When was Napoleon defeated?
  - Look for: [PATIENT Napoleon] [PRED **defeat-synset**] [ARGM-TMP *ANS*]
- Machine Translation

English (SVO)	Farsi (SOV)	
[AGENT The little boy]	[AGENT pesar koocholo]	boy-little
[PRED **kicked**]	[THEME toop germezi]	ball-red
[THEME the red ball]	[ARGM-MNR moqtam]	hard-adverb
[ARGM-MNR hard]	[PRED **zaad-e**]	hit-past

- Document Summarization
  - Predicates and Heads of Roles summarize content
- Information Extraction

Quick Overview

- Part I. Introduction
  - ✓ What is Semantic Role Labeling?
  - From manually created grammars to statistical approaches
    - Early Work
    - Corpora – FrameNet, PropBank, Chinese PropBank, NomBank
  - The relation between Semantic Role Labeling and other tasks
- Part II. General overview of SRL systems
  - System architectures
  - Machine learning models
- Part III. CoNLL-05 shared task on SRL
  - Details of top systems and interesting systems
  - Analysis of the results
  - Research directions on improving SRL systems
- Part IV. Applications of SRL

Moving toward Statistical Approaches

- Early work [Hirst 87]
- Available corpora
  - FrameNet [Fillmore et al. 01]
    - http://framenet.icsi.berkeley.edu
  - PropBank [Palmer et al. 05]
    - http://www.cis.upenn.edu/~mpalmer/project_pages/ACE.htm

Main Focus

- Corpora in development
  - Chinese PropBank
    - http://www.cis.upenn.edu/~chinese/cpb/
  - NomBank
    - http://nlp.cs.nyu.edu/meyers/NomBank.html

Early Work [Hirst 87]

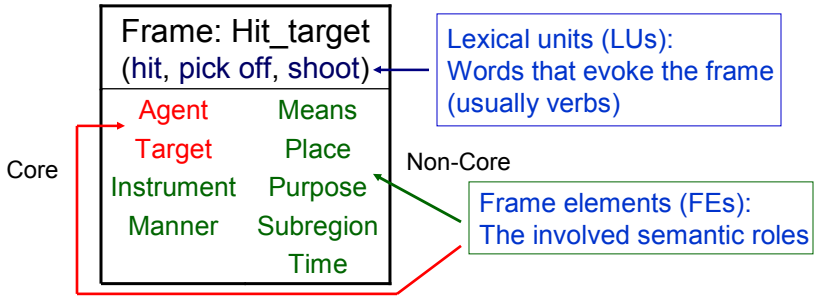
- Semantic Interpretation

  "The process of mapping a syntactically analyzed text of natural language to a representation of its meaning."
- *Absity* – semantic interpreter by Hirst
  - Based on manually created semantic rules
  - Input: *Nadia*$_{\text{subj}}$ *bought the book*$_{\text{obj}}$ *from a store in the mall.*
  - Output:

```
(a ?u
   (buy ?u
      (agent = (the ?x (person ?x
         (propername = "Nadia"))))
      (patient = (the ?y (book ?y)))
      (source = (a ?z (store ?z
         (location = (the ?w (mall ?w)))))))
```

FrameNet [Fillmore et al. 01]

- Sentences from the British National Corpus (BNC)
- Annotated with *frame-specific* semantic roles
  - Various participants, props, and other conceptual roles



[$_{\text{Agent}}$ *Kristina*] ***hit*** [$_{\text{Target}}$ *Scott*] [$_{\text{Instrument}}$ *with a baseball*] [$_{\text{Time}}$ *yesterday*].

FrameNet – Continued

- Methodology of constructing FrameNet
  - Define/discover/describe *frames*
  - Decide the participants (*frame elements*)
  - List *lexical units* that invoke the frame
  - Find example sentences in the corpus (BNC) and annotate them
- Corpora
  - Frame I – British National Corpus only
  - Frame II – LDC North American Newswire corpora
- Size
  - >8,900 lexical units, >625 frames, >135,000 sentences

http://framenet.icsi.berkeley.edu

Proposition Bank (PropBank)

- Transfer sentences to propositions
  - Kristina hit Scott → hit(Kristina,Scott)
- Penn TreeBank → PropBank [Palmer et al. 05]
  - Add a semantic layer on Penn TreeBank
  - Define a set of semantic roles for each verb
  - Each verb's roles are numbered

…[**A0** the company] to … *offer* [**A1** a 15% to 20% stake] [**A2** to the public]
…[**A0** Sotheby's] … *offered* [**A2** the Dorrance heirs] [**A1** a money-back guarantee]
…[**A1** an amendment] *offered* [**A0** by Rep. Peter DeFazio] …
…[**A2** Subcontractors] will be *offered* [**A1** a settlement] …

Proposition Bank (PropBank) Define the Set of Semantic Roles

- It's difficult to define a general set of semantic roles for all types of predicates (verbs).
- PropBank defines semantic roles for each verb and sense.
- The (core) arguments are labeled by numbers.
  - A0 – Agent; A1 – Patient or Theme
  - Other arguments – no consistent generalizations
- Adjunct-like arguments – ***universal*** to all verbs
  - AM-LOC, TMP, EXT, CAU, DIR, PNC, ADV, MNR, NEG, MOD, DIS

Proposition Bank (PropBank) Frame Files

- hit.01 “strike”
  - A0: agent, hitter; A1: thing hit; A2: instrument, thing hit by or with

    [$_{A0}$ *Kristina*] ***hit*** [$_{A1}$ *Scott*] [$_{A2}$ *with a baseball*] *yesterday*.

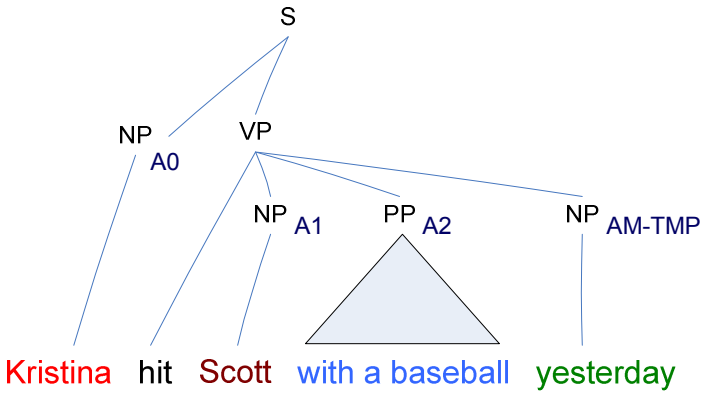
    AM-TMP *Time*
- look.02 “seeming”
  - A0: seemer; A1: seemed like; A2: seemed to

    [$_{A0}$ *It*] ***looked*** [$_{A2}$ *to her*] *like* [$_{A1}$ *he deserved this*].
- deserve.01 “deserve”
  - A0: deserving entity; A1: thing deserved; A2: in-exchange-for

    It looked to her like [$_{A0}$ *he*] ***deserved*** [$_{A1}$ *this*].

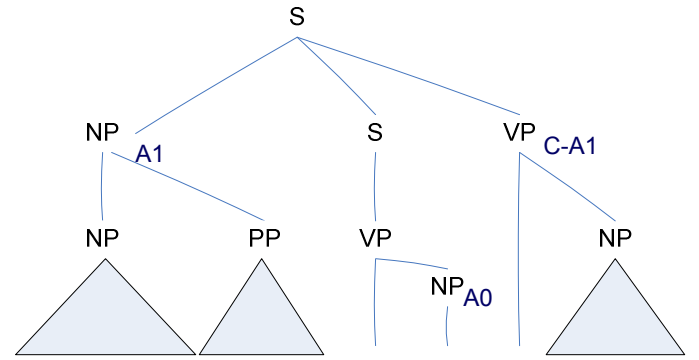
Proposition: A sentence and a target verb

Proposition Bank (PropBank) Add a Semantic Layer



[$_{A0}$ *Kristina*] ***hit*** [$_{A1}$ *Scott*] [$_{A2}$ *with a baseball*] [$_{AM-TMP}$ *yesterday*].

Proposition Bank (PropBank)
Add a Semantic Layer – Continued



[$_{A1}$ *The worst thing about him*] ***said*** [$_{A0}$ *Kristina*] [$_{C\text{-}A1}$ *is his laziness*].

Proposition Bank (PropBank)
Final Notes

- Current release (Mar 4, 2005): Proposition Bank I
  - Verb Lexicon: 3,324 frame files
  - Annotation: ~113,000 propositions
    http://www.cis.upenn.edu/~mpalmer/project_pages/ACE.htm
- Alternative format: CoNLL-04,05 shared task
  - Represented in table format
  - Has been used as standard data set for the shared tasks on semantic role labeling
    http://www.lsi.upc.es/~srlconll/soft.html

Corpora in Development

- Chinese PropBank http://www.cis.upenn.edu/~chinese/cpb/
  - Similar to PropBank, it adds a semantic layer on Penn Chinese Treebank
  - A pre-release version has 250K words and 10,364 sentences; ~55%
- NomBank http://nlp.cs.nyu.edu/meyers/NomBank.html
  - Label arguments that co-occur with nouns in PropBank
  - Current Release: Sep. 2005
    - 93,809 instances of nouns; 2,805 different words; ~80%
    - High frequency (>600) nouns have been completed

Quick Overview

- Part I. Introduction
  - ✓ What is Semantic Role Labeling?
  - ✓ From manually created grammars to statistical approaches
    - Early Work
    - Corpora – FrameNet, PropBank, Chinese PropBank, NomBank
  - The relation between Semantic Role Labeling and other tasks
- Part II. General overview of SRL systems
  - System architectures
  - Machine learning models
- Part III. CoNLL-05 shared task on SRL
  - Details of top systems and interesting systems
  - Analysis of the results
  - Research directions on improving SRL systems
- Part IV. Applications of SRL

Relation to Other Tasks

- Information extraction
- Semantic parsing for speech dialogues
  - The Air Travel Information Service (ATIS) task
- Deep semantic parsing
  - Limited domains
  - General domains
- Penn Treebank function tagging
- Predicting case markers

Related Task: Information Extraction

- Example (HUB Event-99 evaluations, [Hirschman et al. 99])
  - A set of domain dependent *templettes*, summarizing information about events from multiple sentences

<MARKET_CHANGE_1>:=	
INSTRUMENT	London [gold]
AMOUNT_CHANGE	fell [$4.70] cents
CURRENT_VALUE	$308.45
DATE:	daily

Time for our daily market report from NASDAQ.
London gold fell $4.70 cents to $308.45.

- Many other task specifications: extracting information about products, relations among proteins, authors of books, etc.

Information Extraction versus Semantic Role Labeling

Characteristic	IE	SRL
Coverage	narrow	broad
Depth of semantics	shallow	shallow
Directly connected to application	sometimes	no
Unit of processing	>sentence	sentence

- Approaches to task: diverse
  - Depends on the particular task and amount of available data
  - Hand written syntactic-semantic grammars compiled into FSA
  - Sequence labeling approaches (HMM, CRF, CMM)
  - Survey materials: http://scottyih.org/IE-survey3.htm
    [Appelt & Israel 99], [Muslea 99]

Related Task: Speech Dialogs

- Spoken Language Understanding: *extract the semantics from an utterance*
- Must deal with uncertainly and disfluencies in speech input
- Example: task setup in a narrow flight reservations domain (ATIS evaluations, [Price 90])

```
<ShowFlight>
  <Flight>
    <DCity filler="City"> Seattle </DCity>
    <ACity filler="City"> Boston </ACity>
  </Flight>
</ShowFlight>
```

Sentence: *"Show me all flights from Seattle to Boston"*

ATIS Parsing versus Semantic Role Labeling

Characteristic	ATIS	SRL
Coverage	narrow	broad
Depth of semantics	deeper	shallow
Directly connected to application	yes	no
Unit of processing	sentence	sentence

- Approaches to ATIS parsing (overview in [Wang et al. 05]):
  - Simultaneous syntactic/semantic parsing [Miller et al. 96], knowledge-based approach [Ward 94, Dowding et al. 93]
  - Current best: small semantic grammar and a sequence labeling model (no full syntactic parsing information) **Error 3.8%** ([Wang et al. 06]).

Related Task: Semantic Parsing for NL Interfaces to Databases

- Example: GeoQuery Domain (a domain of facts for US geography) [Zelle & Mooney 96]

  Sentence: *How many cities are there in the US?*

  Meaning Representation:

  answer(count(city(loc_2(countryid(usa)))))

- Characteristics:
  - A restricted domain for which we have a complete domain model
  - Sentences are usually short but could be ungrammatical
  - Syntax of target representation is more complex compared to the ATIS task
  - Need to represent quantifiers (the largest, the most populated, etc.)

Semantic Parsing for NL Interfaces to Databases versus Semantic Role Labeling

Characteristic	NL interfaces to DB	SRL
Coverage	narrow	broad
Depth of semantics	deep	shallow
Directly connected to application	yes	no
Unit of processing	sentence	sentence

- Approaches
  - Hand-built grammars [Androutsopoulos et al. 05] (overview)
  - Machine learning of symbolic grammars – e.g. [Zelle & Mooney 96]
  - Learned statistical syntactic/semantic grammar [Ge & Mooney 05] (supervised); [Zettlemoyer & Collins 05], [Wong & Mooney 06] (unsupervised)

Related Task: Deep Parsing

- Hand-built broad-coverage grammars create simultaneous syntactic and semantic analyses
  - The Core Language Engine [Alshawi 92]
  - Lexical Functional Grammar LFG ([Bresnan 01], [Maxwell & Kaplan 93])
  - Head Driven Phrase Structure Grammar ([Pollard & Sag 94], [Copestake & Flickinger 00])
- Model more complex phenomena
  - Quantifiers, quantifier scope, not just verb semantics, anaphora
- Difficult to create and expand

Deep Parsing versus Semantic Role Labeling

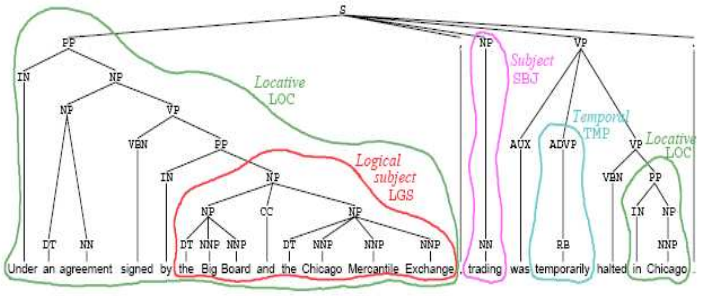
Characteristic	Deep Parsing	SRL
Coverage	broad	broad
Depth of semantics	deep	shallow
Directly connected to application	no	no
Unit of processing	sentence	sentence

- Approach
  - Hand-build grammar (possibly expand automatically)
  - Treated as a parsing problem (joint syntactic and semantic disambiguation)
  - For LFG ([Riezler et al. 02])
  - For HPSG ([Toutanova et al. 04], [Miyao & Tsujii 05])

Related Task: Prediction of Function Tags

[Blaheta&Charniak 00]

The Penn Treebank contains annotation of function tags for some phrases: *subject, logical subject, adjuncts (temporal, locative, etc.)*



Slide from Don Blaheta 03 thesis defense

Prediction of Function Tags versus Semantic Role Labeling

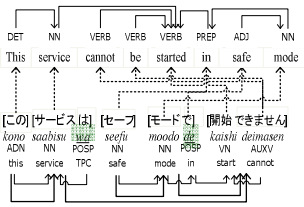
Characteristic	Predicting Function Tags	SRL
Coverage	broad	broad
Depth of semantics	shallower	shallow
Directly connected to application	no	no
Unit of processing	sentence	sentence

- Approach: a classifier based on voted perceptions and other ML techniques
  - Using rich syntactic information from Penn Treebank parse trees
  - Grammatical tags F1 96.4, other tags F1 83.8 [Blaheta 03]

Related Task: Predicting Case Markers

[Suzuki & Toutanova 06]

- Some languages have case markers
  - They indicate the syntactico-semantic relation between a phrase and the phrase it modifies
  - Needed for Machine Translation, foreign language learning



- In Japanese, case markers indicate e.g *subject, object, location.*
  - More similar to function tags than to semantic role labels
- Good news: no annotated data is required!
  - The case markers are part of the surface string

Predicting Case Markers versus Semantic Role Labeling

Characteristic	Predicting Case Markers	SRL
Coverage	broad	broad
Depth of semantics	shallower	shallow
Directly connected to application	yes	no
Unit of processing	sentence	sentence

- Approaches
  - Using content words from the target language only plus dependency information [Uchimoto et al. 02]
  - Using syntactic and word features from the source and target languages [Suzuki & Toutanova 06]; per case marker error using automatic parses: 8.4%

Summary of Part I – Introduction

- What is Semantic Role Labeling?
- Corpora for Semantic Role Labeling
  - We will discuss mainly PropBank.
- Related tasks to SRL
  - Information extraction
  - Deep semantic parsing
  - Penn Treebank function tagging
  - Predicting case markers
- **Next part: overview of SRL systems**

Quick Overview

- Part I. Introduction
  - ✓ What is Semantic Role Labeling?
  - ✓ From manually created grammars to statistical approaches
    - Early Work
    - Corpora – FrameNet, PropBank, Chinese PropBank, NomBank
  - ✓ The relation between Semantic Role Labeling and other tasks
- Part II. General overview of SRL systems
  - System architectures
  - Machine learning models
- Part III. CoNLL-05 shared task on SRL
  - Details of top systems and interesting systems
  - Analysis of the results
  - Research directions on improving SRL systems
- Part IV. Applications of SRL

Part II: Overview of SRL Systems

- Definition of the SRL task
  - Evaluation measures
- General system architectures
- Machine learning models
  - Features & models
  - Performance gains from different techniques

Development of SRL Systems

- Gildea & Jurafsky 2002
  - First statistical model on FrameNet
- 7+ papers in major conferences in 2003
- 19+ papers in major conferences 2004, 2005
- 3 shared tasks
  - Senseval 3 (FrameNet) – 8 teams participated
  - CoNLL 04 (PropBank) – 10 teams participated
  - CoNLL 05 (PropBank) – 19 teams participated

Task Formulation

- Most general formulation: determine a labeling on (usually but not always contiguous) *substrings* (*phrases*) of the sentence *s,* given a predicate *p*

[$_{A0}$ The queen] **broke** [$_{A1}$ the window].

[$_{A1}$ By working hard], [$_{A0}$ he] **said**, [$_{A1}$ you can get exhausted].

- Every substring *c* can be represented by a set of word indices $c \subseteq \{1, 2, \ldots, m\}$
- More formally, a semantic role labeling is a mapping from the set of substrings of s to the label set ***L***. ***L*** includes all argument labels and NONE.

$$2^{\{1,2,\ldots,m\}} \mapsto L$$

Subtasks

- Identification: $2^{\{1,2,\ldots,m\}} \mapsto \{NONE, ARG\}$
  - Very hard task: to separate the argument substrings from the rest in this exponentially sized set
  - Usually only 1 to 9 (avg. **2.7**) substrings have labels ARG and the rest have NONE
- Classification: $2^{\{1,2,\ldots,m\}} \mapsto L \setminus \{NONE\}$
  - Given the set of substrings that have an *ARG* label, decide the exact semantic label
- Core argument semantic role labeling: (easier)
  - Label phrases with core argument labels only. The modifier arguments are assumed to have label NONE.

Evaluation Measures

Correct: [$_{A0}$ The queen] **broke** [$_{A1}$ the window] [$_{AM\text{-}TMP}$ yesterday]
Guess: [$_{A0}$ The queen] broke the [$_{A1}$ window] [$_{AM\text{-}LOC}$ yesterday]

Correct	Guess
{The queen} →A0 {the window} →A1 {yesterday} ->AM-TMP all other → NONE	{The queen} →A0 {window} →A1 {yesterday} ->AM-LOC all other → NONE

- Precision ,Recall, F-Measure {*tp*=1,*fp*=2,*fn*=2} *p*=*r*=*f*=1/3
- Measures for subtasks
  - Identification (Precision, Recall, F-measure) {tp=2,fp=1,fn=1} *p*=*r*=*f*=2/3
  - Classification (Accuracy) acc = .5 (labeling of correctly identified phrases)
  - Core arguments (Precision, Recall, F-measure) {tp=1,fp=1,fn=1} *p*=*r*=*f*=1/2

Part II: Overview of SRL Systems

- ✓ Definition of the SRL task
  - ✓ Evaluation measures
- General system architectures
- Machine learning models
  - Features & models
  - Performance gains from different techniques

Terminology: Local and Joint Models

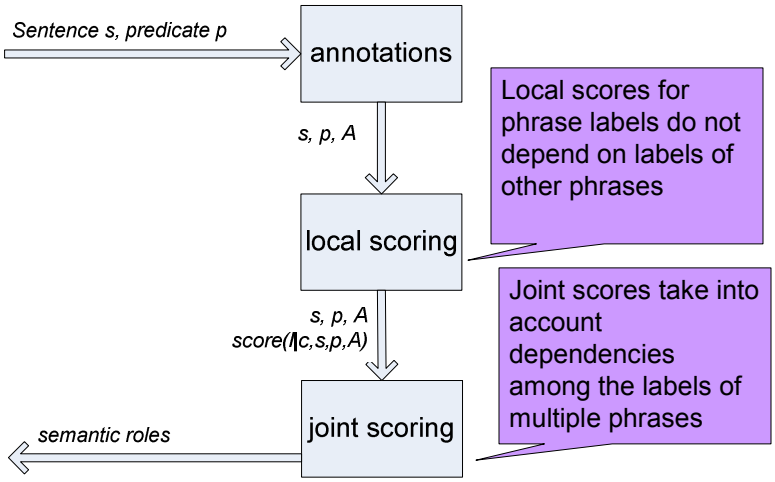
- Local models decide the label of each substring independently of the labels of other substrings
- This can lead to inconsistencies
  - overlapping argument strings

    *By [$_{A1}$ working [$_{A1}$ hard] , he] **said** , you can achieve a lot.*
  - repeated arguments

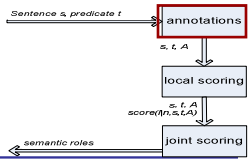
    *By [$_{A1}$ working] hard , [$_{A1}$ he] **said** , you can achieve a lot.*
  - missing arguments

    [$_{A0}$ By working hard , he] **said** , [$_{A0}$ you can achieve a lot].
- Joint models take into account the dependencies among labels of different substrings; they incorporate hard and/or soft constraints among these labels

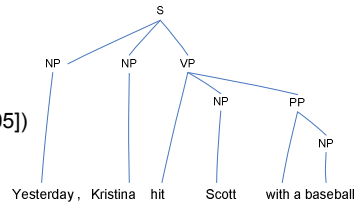
Basic Architecture of a Generic SRL System



Annotations Used



- Syntactic Parsers
  - Collins', Charniak's (most systems) CCG parses ([Gildea & Hockenmaier 03],[Pradhan et al. 05]) TAG parses ([Chen & Rambow 03])



- Shallow parsers

  [$_{NP}$Yesterday] , [$_{NP}$Kristina] [$_{VP}$hit] [$_{NP}$Scott] [$_{PP}$with] [$_{NP}$a baseball].

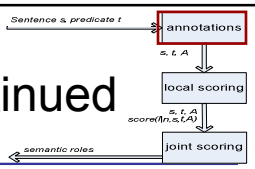
- Semantic ontologies (WordNet, automatically derived), and named entity classes

  (v) **hit** (cause to move by striking)

  WordNet hypernym

  → **propel, impel** *(*cause to move forward with force*)*

Annotations Used - Continued

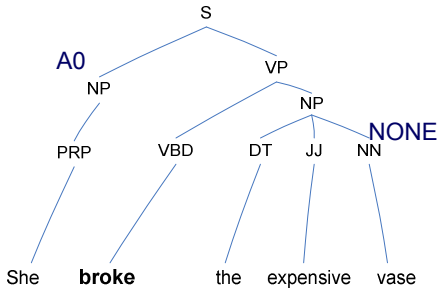


Most commonly, substrings that have argument labels correspond to syntactic constituents

- In Propbank, an argument phrase corresponds to exactly one parse tree constituent in the correct parse tree for **95.7**% of the arguments;
  - when more than one constituent correspond to a single argument (**4.3**%), simple rules can join constituents together (in 80% of these cases ,[Toutanova 05]);
- In Propbank, an argument phrase corresponds to exactly one parse tree constituent in Charniak's automatic parse tree for approx **90.0%** of the arguments.
  - Some cases (about 30% of the mismatches) are easily recoverable with simple rules that join constituents ([Toutanova 05])
- In FrameNet, an argument phrase corresponds to exactly one parse tree constituent in Collins' automatic parse tree for **87%** of the arguments.

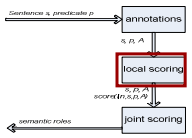
Labeling Parse Tree Nodes

- Given a parse tree *t*, label the nodes (phrases) in the tree with semantic labels
- To deal with discontiguous arguments
  - In a post-processing step, join some phrases using simple rules
  - Use a more powerful labeling scheme, i.e. C-A0 for continuation of A0



Another approach: labeling chunked sentences. Will not describe in this section.

Local Scoring Models



- Notation: a constituent node c, a tree t, a predicate node p, feature map for a constituent $\Phi(c,t,p)$
- Target labels $l \in \{A0,\ldots,A5,AM_{LOC},\ldots,NONE\}$

$Id(l) = NONE$ iff $l = NONE$

$Id(l) = ARG,$ otherwise

- Two (probabilistic) models
  - Identification model

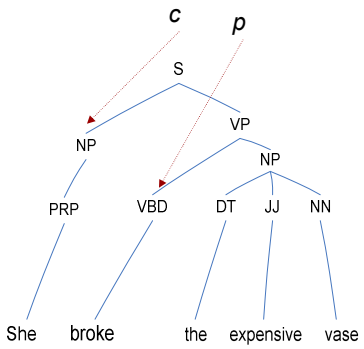
$$P(Id(l)|c,t,p) = P(Id(l)|\Phi(c,t,p))$$

  - Classification model

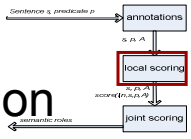
$$P(l|c,t,p) = P(l|Id(l),\Phi(c,t,p))$$

- Sometimes one model

$$P(l|c,t,p) = P(l|\Phi(c,t,p))$$

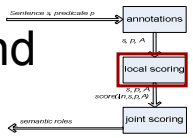


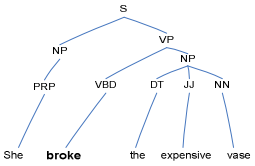
Why Split the Task into Identification and Classification



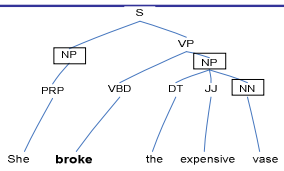
- Different features are helpful for each task
  - Syntactic features more helpful for identification, lexical features more helpful for classification
  - Example: the identity of the predicate, e.g. *p="hit"* is much more important for classification than for identification ([Pradhan et al. 04]):
    - Identification all features: 93.8 no predicate: 93.2
    - Classification all features: 91.0 no predicate: 82.4
  - Some features result in a performance decrease for one and an increase for the other task [Pradhan et al. 04]
- Splitting the task increases computational efficiency in training
  - In identification, every parse tree constituent is a candidate (linear in the size of the parse tree, avg. 40)
  - In classification, label a small number of candidates (avg. 2.7)

Combining Identification and Classification Models

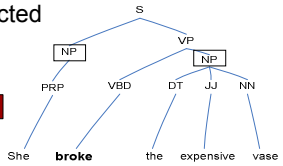




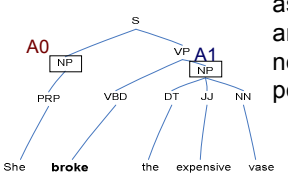
Step 1. *Pruning.* Using a hand-specified filter



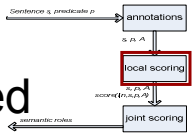
Step 2. *Identification.* Identification model (filters out candidates with high probability of NONE)



Step 3. *Classification.* Classification model assigns one of the argument labels to selected nodes (or sometimes possibly NONE)



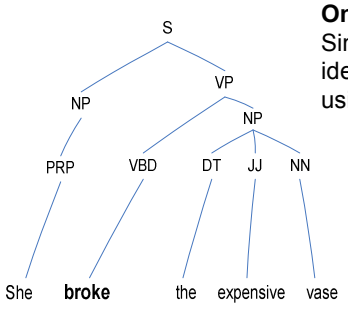
Combining Identification and Classification Models – Continued



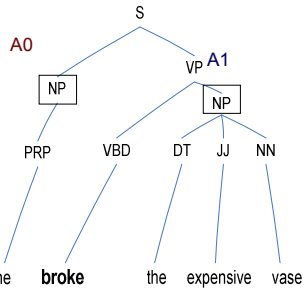
$$P(l|c,t,p) = P_{ID}(Id(l)|\Phi(c,t,p)) * P_{CLS}(l|Id(l),\Phi(c,t,p))$$

or

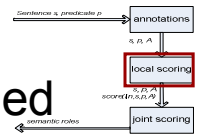
$$P(l|c,t,p) = P(l|\Phi(c,t,p))$$



One Step. Simultaneously identify and classify using $P(l|c,t,p)$

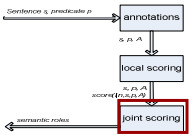


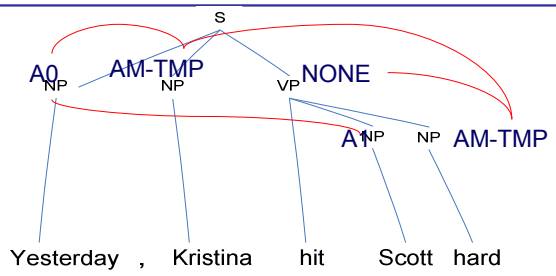
Combining Identification and Classification Models – Continued



- [Gildea&Jurafsky 02]
  - **Identification** + **Classification** for local scoring experiments
  - **One Step** for joint scoring experiments
- [Xue&Palmer 04] and [Punyakanok et al. 04, 05]
  - **Pruning** + **Identification** + **Classification**
- [Pradhan et al. 04]
  - **Identification** + **Classification**
- [Toutanova et al. 05]
  - **One Step**

Joint Scoring Models

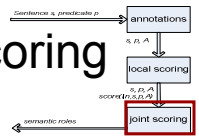




- These models have scores for a whole labeling of a tree (not just individual labels)
  - Encode some dependencies among the labels of different nodes

$$P_{JOINT}(l_1, \ldots, l_n | t, p) ! = \prod_i P(l_i | n_i, t, p)$$

Combining Local and Joint Scoring Models



- Re-ranking or approximate search to find the labeling which maximizes the product of scores [Gildea&Jurafsky 02] [Pradhan et al. 04] [Toutanova et al. 05]
  - Usually exponential search required to find the exact maximizer

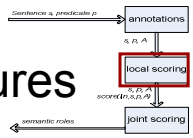
$$P_{COMBINED}(L|t,p) = P_{LOCAL}(L|t,p)P_{JOINT}(L|t,p)$$

- Tighter integration of local and joint scoring in a single probabilistic model and exact search [Cohn&Blunsom 05] [Màrquez et al. 05]
  - When the joint model makes strong independence assumptions
- Exact search for the maximizer using Integer Linear Programming [Punyakanok et al 04,05]
  - When the joint model enforces some natural hard constraints
- More details later

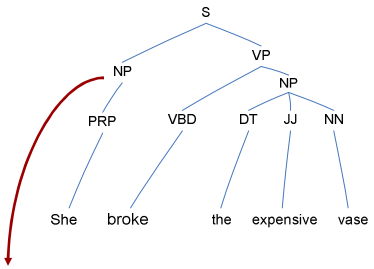
Part II: Overview of SRL Systems

- ✓ Definition of the SRL task
  - ✓ Evaluation measures
- ✓ General system architectures
- Machine learning models
  - Features & models
    - For Local Scoring
    - For Joint Scoring
  - Performance gains from different techniques

Gildea & Jurafsky (2002) Features

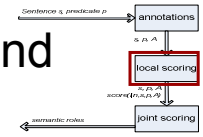


- Key early work
  - Future systems use these features as a baseline
- Constituent Independent
  - Target predicate (lemma)
  - Voice
  - Subcategorization
- Constituent Specific
  - Path
  - Position (*left, right*)
  - Phrase Type
  - Governing Category (*S* or *VP*)
  - Head Word



Target	*broke*
Voice	*active*
Subcategorization	*VP→VBD NP*
Path	*VBD↑VP↑S↓NP*
Position	*left*
Phrase Type	*NP*
Gov Cat	*S*
Head Word	*She*

Evaluation using Correct and Automatic Parses

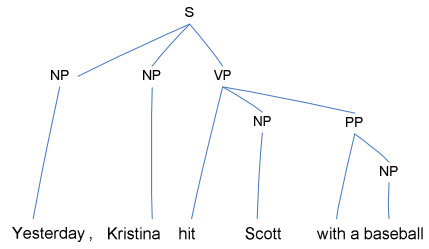


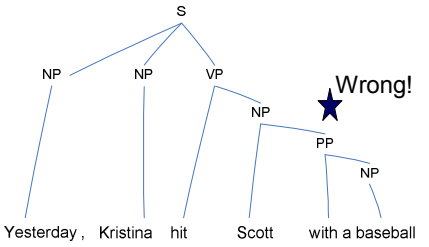
For a correct parse, 95.7% of arguments correspond to a single constituent and their boundaries are easy to consider

For an automatic parse (Charniak's parser), about 90% of the arguments correspond to a single constituent;

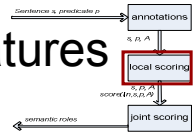
-the arguments for which the parser made a bracketing error are difficult to get

-additionally, attachment errors and labeling errors make the task much harder



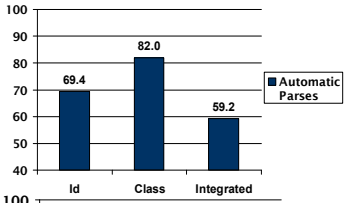


Performance with Baseline Features using the G&J Model

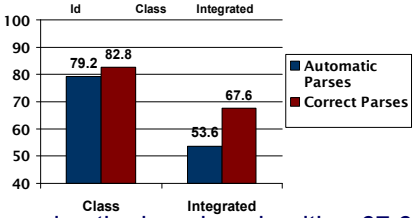


- **Machine learning algorithm:** interpolation of relative frequency estimates based on subsets of the 7 features introduced earlier

FrameNet Results

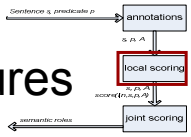


Propbank Results

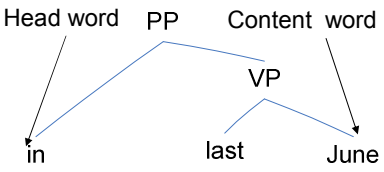


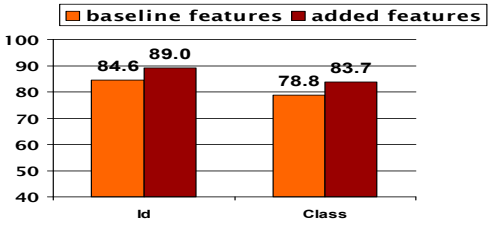
Just by changing the learning algorithm 67.6 → **80.8** using SVMs [Pradhan et al. 04]),

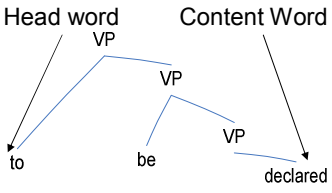
Surdeanu et al. (2003) Features



- Content Word (different from head word)
- Head Word and Content Word POS tags
- NE labels (Organization, Location, etc.)

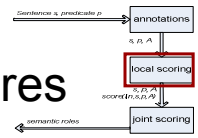




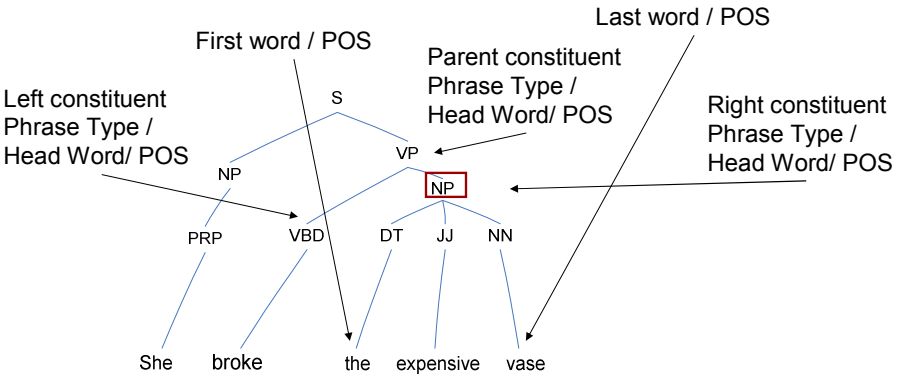


Gains from the new features using correct parses; **28%** error reduction for Identification and **23%** error reduction for Classification

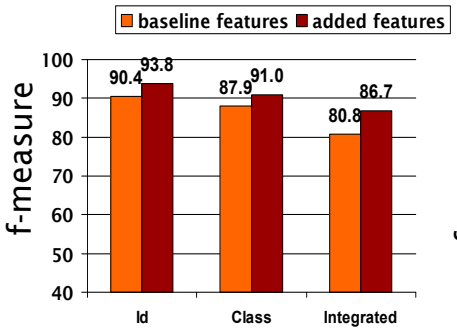
Pradhan et al. (2004) Features



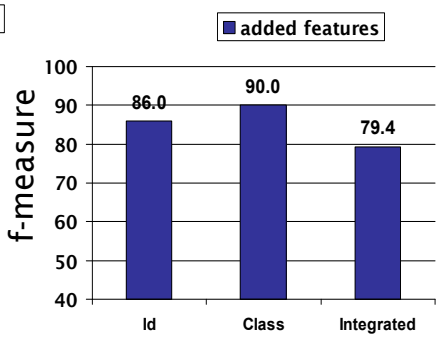
- More structural/lexical context (**31%** error reduction from baseline due to these + Surdeanu et al. features)



Pradhan et al. (2004) Results



Results on correct parse trees

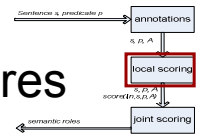


Results on automatic parse trees

Baseline results higher than Gildea and Jurafsky's due to a different classifier - SVM

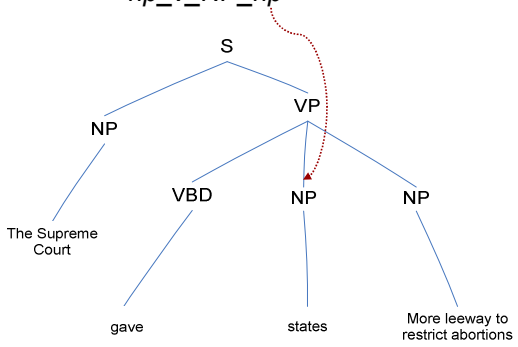
These are the highest numbers on Propbank version July 2002

Xue & Palmer (2004) Features

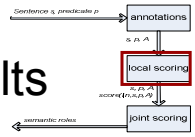


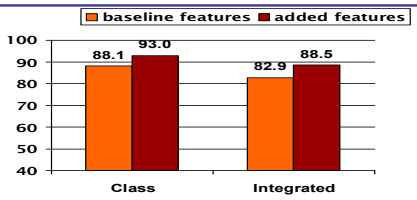
- Added explicit feature conjunctions in a MaxEnt model, e.g. predicate + phrase type
- Syntactic frame feature (**helps a lot**)
- Head of PP Parent (**helps a lot**)
  - If the parent of a constituent is a PP, the identity of the preposition (feature good for PropBank Feb 04)

np_give_NP_np
np_give_CURR_np
np_v_NP_np



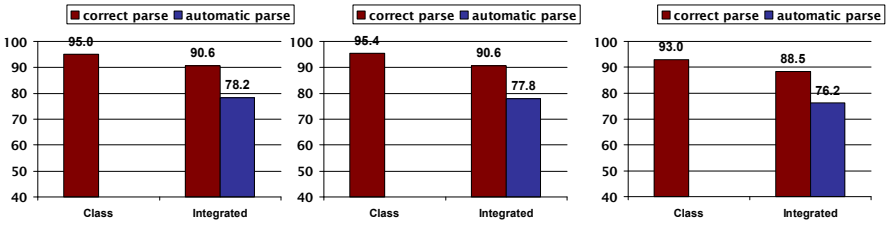
Xue & Palmer (2004) Results





All Arguments Correct Parse

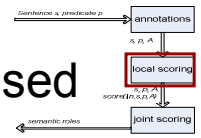
A newer version of Propbank – February 2004



Core Arguments | Coarse AM Arguments | All Arguments

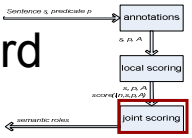
Results not better than [Pradhan et al. 04], but comparable.

Machine Learning Models Used



- Back-off lattice-based relative frequency models ([Gildea&Jurafsky 02], [Gildea& Palmer 02])
- Decision trees ([Surdeanu et al. 03])
- Support Vector Machines ([Pradhan et al. 04])
- Log-linear models ([Xue&Palmer 04][Toutanova et al. 05])
- SNoW ([Punyakanok et al. 04,05])
- AdaBoost, TBL, CRFs, …

Joint Scoring: Enforcing Hard Constraints

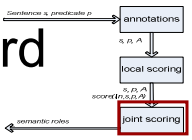


- Constraint 1: Argument phrases do not overlap

  *By [$_{A1}$ working [$_{A1}$ hard] , he] **said** , you can achieve a lot.*

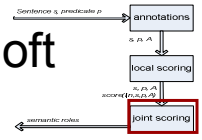
  - Pradhan et al. (04) – greedy search for a best set of non-overlapping arguments
  - Toutanova et al. (05) – exact search for the best set of non-overlapping arguments (dynamic programming, linear in the size of the tree)
  - Punyakanok et al. (05) – exact search for best non-overlapping arguments using linear programming (worst-case NP hard)
- Other constraints ([Punyakanok et al. 04, 05])
  - no repeated core arguments (good heuristic)
  - phrases do not overlap the predicate
  - (*more later*)

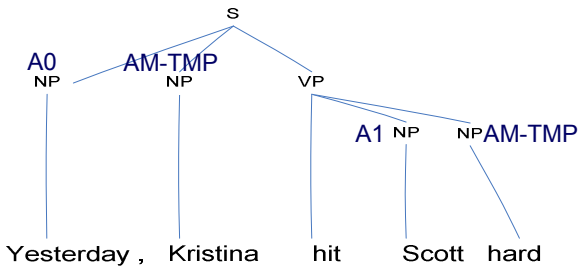
Gains from Enforcing Hard Constraints



- Argument phrases do not overlap
  - Pradhan et al. (04) good gains for a baseline system: 80.8 → 81.6 correct parses
  - Toutanova et al. (05) a small gain from non-overlapping for a model with many features 88.3 → 88.4 correct parses
- Other hard constraints (no repeating core arguments, set of labeled arguments allowable, etc.)
  - Punyakanok et al. (04) evaluation of this aspect only when using chunked sentences (not full parsing) 87.1 → 88.1 correct parses 67.1 → 68.2 automatic parses
  - Punyakanok et al. (05) non-overlapping must be extremely important when combining multiple systems (*more later*)

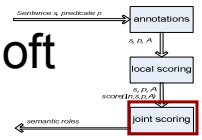
Joint Scoring: Integrating Soft Preferences





- There are many statistical tendencies for the sequence of roles and their syntactic realizations
  - When both are before the verb, AM-TMP is usually before A0
  - Usually, there aren't multiple temporal modifiers
  - Many others which can be learned automatically

Joint Scoring: Integrating Soft Preferences



- Gildea and Jurafsky (02) – a smoothed relative frequency estimate of the probability of frame element multi-sets:

$$P(\{A0, AM_{TMP}, A1, AM_{TMP}\}|hit)$$

  - Gains relative to local model 59.2 → 62.9 FrameNet automatic parses

- Pradhan et al. (04) – a language model on argument label sequences (with the predicate included)

$$P(A0, AM_{TMP}, hit, A1, AM_{TMP})$$

  - Small gains relative to local model for a baseline system 88.0 → 88.9 PropBank correct parses

- Toutanova et al. (05) – a joint model based on CRFs with a rich set of joint features of the sequence of labeled arguments (*more later)*
  - Gains relative to local model on PropBank correct parses 88.4 → 91.2 ; gains on automatic parses 78.2 → 80.0

- Sequence tagging for chunk-based representations (*more later*)

Combining Annotations and Combining Systems

- Punyakanok et al. (05) combine information from systems trained on top *n* parse trees produced by Charniak's parser and Collins' parser.
  - Effectively constituents from all trees can be selected as arguments
  - Constraints for non-overlap are enforced through ILP
  - Gains 74.8 → 77.3 on automatic parses (CoNLL 05 dev set)

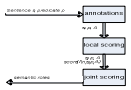
- Haghighi et al. (05) combine top *n* Charniak parse trees
  - This is achieved in a Bayesian way: sum over the parse trees approximated by max
  - Gains 79.7 → 80.3 on automatic parses (CoNLL 05 test set)

- Pradhan et al. (05) combine different syntactic views
  - Charniak syntactic parse, Combinatory Categorial Grammar parse

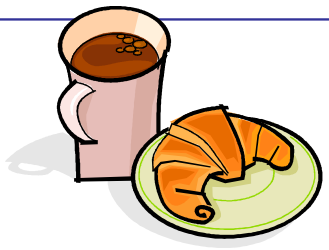
- Other systems in CoNLL 2005

- *More later on all of these*

Summary of Part II – System Overview



- Introduced SRL system architecture:
  - *annotations, local scoring, joint scoring*
- Described major features helpful to the task
  - showed that large gains can be achieved by improving the features
- Described methods for local scoring, combining i*dentification* and *classification* models
- Described methods for joint scoring
  - gains from incorporating *hard* constraints
  - gains from incorporating *soft* preferences
- Introduced the concept of combining systems and annotation
  - significant gains possible
- **Next part**: more details on the systems in CoNLL 2005



Break!!

[$_{A0}$ We] [$_{AM-MOD}$ will] see [$_{A1}$ you] [$_{AM-TMP}$ after the break].

Quick Overview

- Part I. Introduction
  - ✓ What is Semantic Role Labeling?
  - ✓ From manually created grammars to statistical approaches
    - Early Work
    - Corpora – FrameNet, PropBank, Chinese PropBank, NomBank
  - ✓ The relation between Semantic Role Labeling and other tasks
- ✓ Part II. General overview of SRL systems
  - ✓ System architectures
  - ✓ Machine learning models
- Part III. CoNLL-05 shared task on SRL
  - Details of top systems and interesting systems
  - Analysis of the results
  - Research directions on improving SRL systems
- Part IV. Applications of SRL

Part III: CoNLL-05 Shared Task on SRL

- Details of top systems and interesting systems
  - Introduce the top 4 systems
  - Describe 3 spotlight systems
- Analysis of the overall results
  - General performance
  - System properties
  - Per argument performance
- Directions to improving SRL systems

Details of CoNLL-05 Systems

- Top performing systems
  - #3 Màrquez et al. (Technical University of Catalonia)
  - #4 Pradhan et al. (University of Colorado at Boulder)
  - #1 Punyakanok et al. (U. of Illinois at Urbana-Champaign)
  - #2 Haghighi et al. (Stanford University)
- Spotlight systems
  - Yi & Palmer – *integrating syntactic and semantic parsing*
  - Cohn & Blunsorn – *SRL with Tree CRFs*
  - Carreras – *system combination*

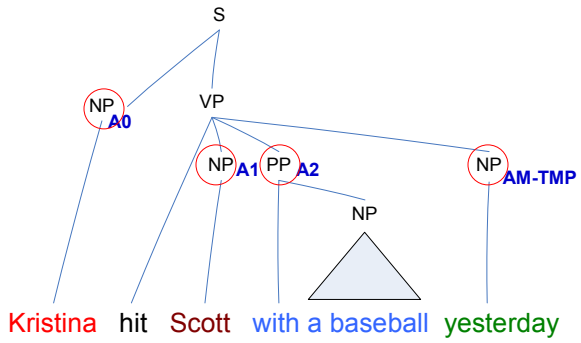
SRL as Sequential Tagging [Màrquez et al.]

- A conceptually simple but competitive system
- SRL is treated as a flat sequential labeling problem represented in the BIO format.
- System architecture
  - Pre-processing (sequentialization)
    - FP_{CHA}: full-parse, based on Charniak's parser
    - PP_{UPC}: partial-parse, based on UPC chunker & clauser
  - Learning using AdaBoost
  - Greedy combination of two systems

Sequentialization – Full Parse
[Màrquez et al.] – Continued

- Explore the sentence regions defined by the clause boundaries.
- The top-most constituents in the regions are selected as tokens.
- Equivalent to [Xue&Palmer 04] pruning process on full parse trees

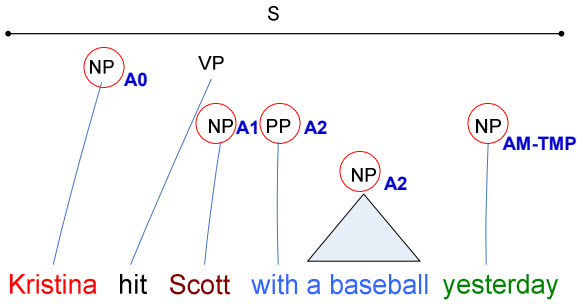
Kristina	B-A0
hit	O
Scott	B-A1
with a baseball	B-A2
yesterday	B-AM-TMP



Sequentialization – Partial Parse
[Màrquez et al.] – Continued

- Only clauses and base chunks are available.
- Chunks within the same clause are selected as tokens.

Kristina	B-A0
hit	O
Scott	B-A1
with	B-A2
a Baseball	I-A2
yesterday	B-AM-TMP



Greedy Combination

[Màrquez et al.] – Continued

- Join the maximum number of arguments from the output of both systems
  - More impact on Recall
- Different performance on different labels
  - FP_{CHA}: better for A0 and A1; PP_{UPC}: better for A2-A4
- Combining rule
  1. Adding arguments from FP_{CHA} that it's good at
  2. Adding arguments from PP_{UPC} that it's good at
  3. Repeat Step 1&2 for other arguments
  - *Drop overlapping/embedding arguments*

Results

[Màrquez et al.] – Continued

- Overall results on development set

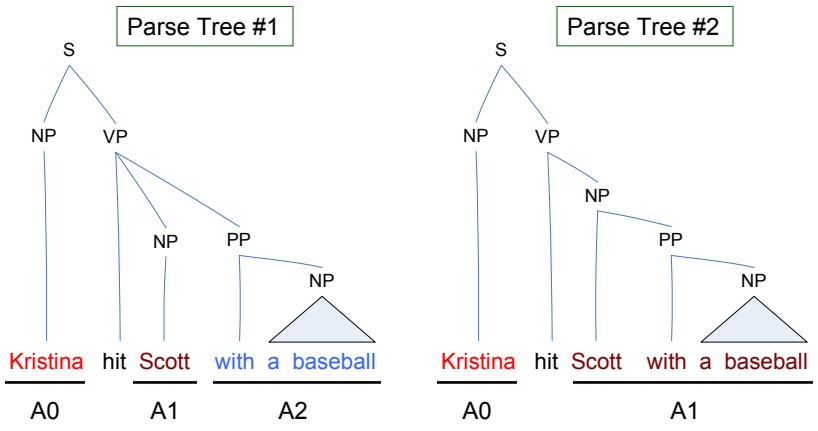
	F_1	Prec.	Rec.
PP_{UPC}	73.57	76.86	70.55
FP_{CHA}	75.75	78.08	73.54
Combined	76.93	78.39	75.53

- Final results on test sets
  - WSJ-23: 77.97 (F_1), 79.55 (Prec.), 76.45 (Rec.)
  - Brown: 67.42 (F_1), 70.79 (Prec.), 64.35 (Rec.)

Semantic Role Chunking Combining Complementary Syntactic Views [Pradhan et al.]

- Observations: the performance of an SRL system depends heavily on the syntactic ***view***
  - Syntactic parse trees generated by full parsers
    - Charniak's, Collins', …
  - Partial syntactic analysis by chunker, clauser, etc.
- Usage of syntactic information
  - Features (e.g., path, syntactic frame, etc.)
  - Argument candidates (mostly the constituents)
- Strategy to reduce the impact of incorrect syntactic info.
  - Build individual SRL systems based on different syntactic parse trees (Charniak's and Collins')
  - Use the predictions as additional features
  - Build a final SRL system in the sequential tagging representation

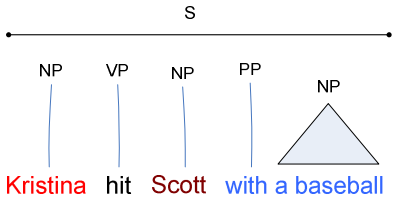
Constituent Views
[Pradhan et al.] – Continued



Chunk View

[Pradhan et al.] – Continued

- Sequentialization using base chunks [Hacioglu&Ward 03]
- Chunker: *Yamcha* [Kudo&Matsumoto 01]
  - http://chasen.org/~taku/software/yamcha/



Chunks	True Label	Pred #1	Pred #2
Kristina	B-A0	B-A0	B-A0
hit	O	O	O
Scott	B-A1	B-A1	B-A1
with	B-A2	B-A2	I-A1
a Baseball	I-A2	I-A2	I-A2

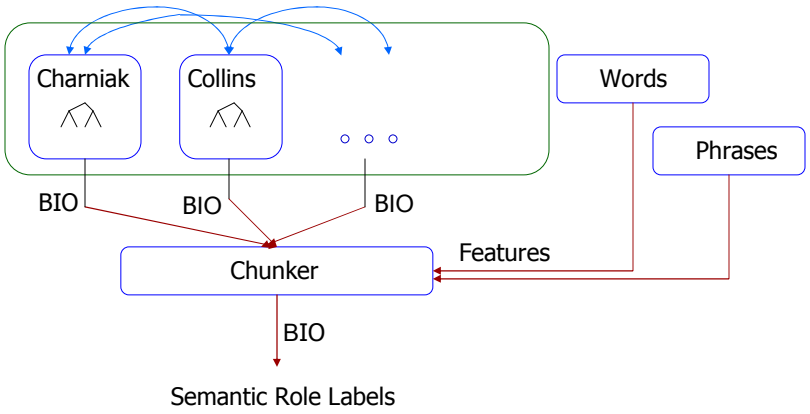
Algorithm

[Pradhan et al.] – Continued

- Generate features from Charniak's and Collins' parse trees
- Add a few features from one to the other, and construct two SRL systems
- Represent the output as semantic BIO tags, and use them as features
- Generate the final semantic role label set using a phrase-based chunking paradigm

Architecture

[Pradhan et al.] – Continued



Slide from Pradhan et al. (CoNLL 2005)

Results

[Pradhan et al.] – Continued

- Overall results on development set

System	F_1	Prec	Rec
Charniak	77	80	75
Collins	76	79	74
Combined	78	81	76

- Performance (F_1) on Test sets
  - Submitted system: WSJ-23 77.4, Brown 67.1
  - Bug-fixed system: WSJ-23 78.6, Brown 68.4

➢ Software: ASSERT (Automatic Statistical SEmantic Role Tagger)
http://oak.colorado.edu/assert

Generalized Inference [Punyakanok et al.]

- The output of the argument classifier often violates some constraints, especially when the sentence is long.
- Use the integer linear programming inference procedure [Roth&Yih 04]
  - Input: the local scores (by the argument classifier), and structural and linguistic constraints
  - Output: the best legitimate global predictions
  - Formulated as an optimization problem and solved via Integer Linear Programming.
  - Allows incorporating expressive (non-sequential) constraints on the variables (the arguments types).

Integer Linear Programming Inference
[Punyakanok et al.] – Continued

- For each argument a_i and label t
  - Set up a Boolean variable: $a_{i,t} \in \{0,1\}$
    - indicating if a_i is classified as t
- Goal is to maximize
  - $\sum_i$ score($a_i = t$) $a_{i,t}$
  - Subject to the (linear) constraints
    - Any Boolean constraint can be encoded this way.
- If score($a_i = t$) = P($a_i = t$), then the objective is
  - Find the assignment that maximizes the expected number of arguments that are correct
  - Subject to the constraints.

Examples of Constraints

[Punyakanok et al.] – Continued

- No duplicate argument classes

  $\Sigma_{a \in \text{POTARG}} x_{\{a = A0\}} \leq 1$

  Any Boolean rule can be encoded as a set of linear constraints.

- C-ARG

  If there is an C-*arg* phrase, there is an *arg* before it

  $\forall a' \in \text{POTARG}$,

  $\Sigma_{(a \in \text{POTARG}) \wedge (a \text{ is before } a')} x_{\{a = A0\}} \geq x_{\{a' = C\text{-}A0\}}$

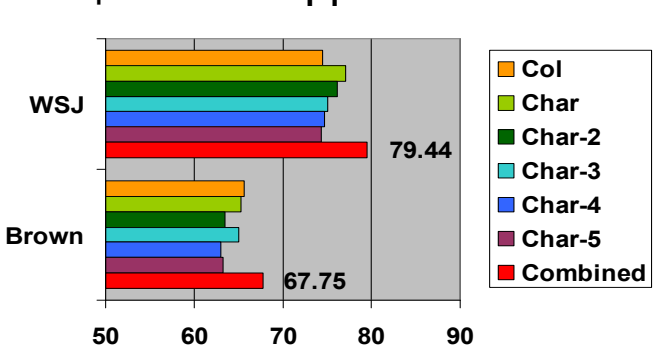
- Many other possible constraints:
  - Unique labels
  - No overlapping or embedding
  - Relations between number of arguments
  - If verb is of type A, no argument of type B

Joint inference can be used also to combine different SRL Systems.

Results

[Punyakanok et al.] – Continued

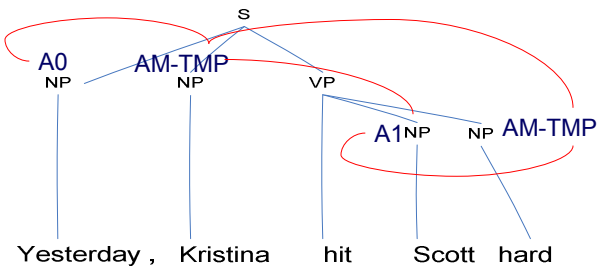
- Char: Charniak's parser (5-best trees)
- Col: Collins' parser



Online Demo: http://l2r.cs.uiuc.edu/~cogcomp/srl-demo.php

A Joint Model for SRL [Haghighi et al.]

- The main idea is to build a rich model for joint scoring, which takes into account the dependencies among the *labels* of argument phrases



Joint Discriminative Reranking

[Haghighi et al.] – Continued

- For computational reasons: start with local scoring model with strong independence assumptions

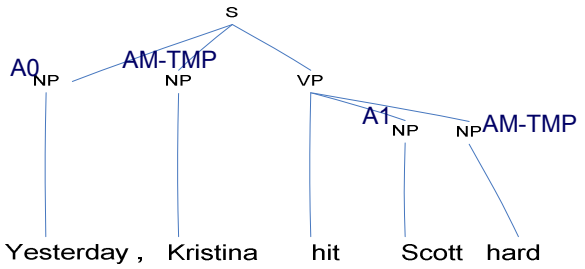
$$P(labels|tree) = \prod_{node_i \in tree} P(labels_i|node_i)$$

- Find top N non-overlapping assignments for local model using a simple dynamic program [Toutanova et al. 05]
- Use joint model to select best assignment among top N using a joint log-linear model [Collins 00]
- The resulting probability of a complete labeling *L* of the tree for a predicate *p* is given by:

$$P_{SRL}(L|tree, p) = log(P_{JOINT}(L|tree, p)) + \lambda log(P_{LOCAL}(L|tree, p))$$

Joint Model Features

[Haghighi et al.] – Continued



Repetition features: count of arguments with a given label c(AM-TMP)=2

Complete sequence syntactic-semantic features for the core arguments:

[NP_A0 hit NP_A1**] , [**NP_A0 VBD NP_A1**]** (backoff)

[NP_A0 hit] (left backoff)

[NP_ARG hit NP_ARG] (no specific labels)

[1 hit 1] (counts of left and right core arguments)

Using Multiple Trees

[Haghighi et al.] – Continued

- Using the best Charniak's parse, on development set
  - Local Model: 74.52(F_1); Joint Model: 76.71(F_1)
- Further enhanced by using Top K trees
  - For top *k* trees from Charniak's parser $t_1, t_2, \cdots, t_k$ find corresponding best SRL assignments $L_1, \cdots, L_k$ and choose the tree and assignment that maximize the score (approx. joint probability of tree and assignment)

$$score(L_i, t_i) = \alpha log(P(t_i)) + log(P_{SRL}(L_i | t_i))$$

- Final Results:
  - WSJ-23: 78.45 (F_1), 79.54 (Prec.), 77.39 (Rec.)
  - Brown: 67.71 (F_1), 70.24 (Prec.), 65.37 (Rec.)
  - Bug-fixed post-evaluation: WSJ-23 80.32 (F_1) Brown 68.81 (F_1)

Details of CoNLL-05 Systems

- ✓ Top performing systems
  - ✓ Màrquez et al. (Technical University of Catalonia)
  - ✓ Pradhan et al. (University of Colorado at Boulder)
  - ✓ Punyakanok et al. (U. of Illinois at Urbana-Champaign)
  - ✓ Haghighi et al. (Stanford University)
- Spotlight systems
  - Yi & Palmer – *integrating syntactic and semantic parsing*
  - Cohn & Blunsom – *SRL with Tree CRFs*
  - Carreras – *system combination*

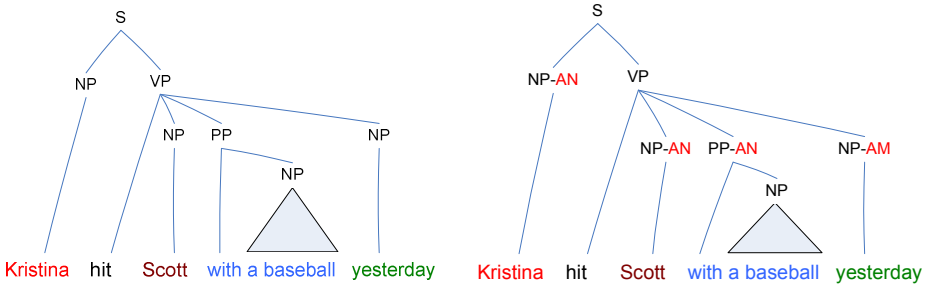
The Integration of Syntactic Parsing and Semantic Role Labeling [Yi & Palmer]

- The bottleneck of the SRL task: parsing
  - With [Xue&Palmer 04] pruning, given different parsers: 12%~18% arguments are lost (Development Set: WSJ-22)
- What do we want from syntactic parsing?
  - Correct constituent boundaries
  - Correct tree structures: expressing the dependency between the target verb and its arguments (e.g., the *path* feature)
- The proposed approach:
  - Combine syntactic parsing & argument identification (different cut of the task)
    - Train a new parser on the training data created by merging the Penn Treebank & the PropBank (sec 02-21)

Slide from Yi&Palmer (CoNLL 2005)

Data Preparation & Base Parser
[Yi & Palmer] – Continued

- Data preparation steps
  - Strip off the Penn Treebank function tags
  - 2 types of sub-labels to represent the PropBank arguments:
    - AN: core arguments
    - AM: adjunct-like arguments
- Train new maximum-entropy parsers [Ratnaparkhi 99]



Based on Yi&Palmer's slides (CoNLL 2005)

Results & Discussion
[Yi & Palmer] – Continued

- Overall results on development set

	F_1	Prec.	Rec.
AN-parser	67.28	71.31	63.68
AM-parser	69.31	74.03	65.11
Charniak	69.98	76.31	64.62
Combined	72.73	75.70	69.99

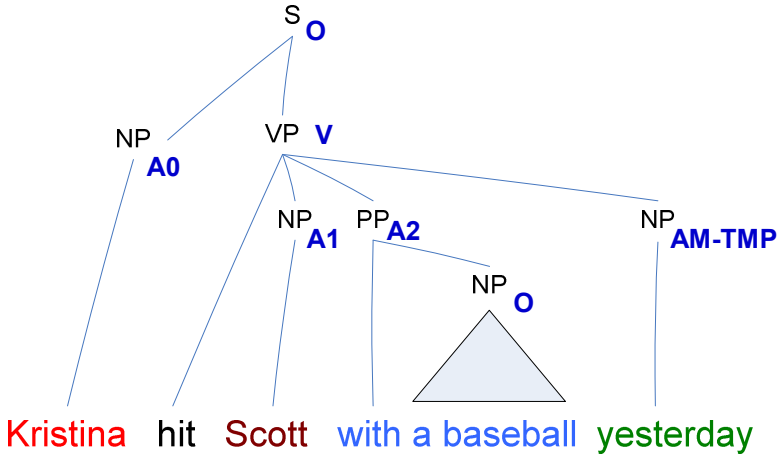
- Final F_1 – WSJ-23: 77.51, Brown: 67.88
- Worse than using Charniak's directly
  - Because of weaker base parser?
- Hurt both parsing and argument identification?

SRL with Tree CRFs [Cohn & Blunsom]

- A different joint model – apply tree CRFs
  - Generate the full parse tree using Collins' parser
  - Prune the tree using [Xue&Palmer 04]
  - Label each remaining constituent the **semantic role** or **O** (i.e., None)
  - Learn the CRFs model
- Efficient CRF inference methods exist for trees
  - Maximum Likelihood Training: sum-product algorithm
  - Finding the best in Testing: max-product algorithm

Tree Labeling

[Cohn & Blunsom] – Continued



Model and Results
[Cohn & Blunsom] – Continued

- Definition of CRFs $p(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp \sum_{c \in C} \sum_{k} \lambda_k f_k(c, \mathbf{y}_c, \mathbf{x})$
- Maximum log-likelihood training
  $E_{\tilde{p}(\mathbf{x},\mathbf{y})}[f_k] - E_{p(\mathbf{x},\mathbf{y})}[f_k] = 0$
  - Use sum-product to calculate marginal $E_{p(\mathbf{x},\mathbf{y})}[f_k]$
  - Use max-product to find the best labeling
- Results: Final F_1 – WSJ-23: 73.10, Brown: 63.63
- Findings [Cohn&Blunsom CoNLL-05 slides]:
  - CRFs improved over maxent classifier (+1%)
  - Charniak parses more useful (+3%)
  - Very few inconsistent ancestor/dependent labelings
  - Quite a number of duplicate argument predictions

Data from Cohn&Blunsom's slide (CoNLL 2005)

System Combination [Carreras et al.]

- How much can we gain from combining different participating systems at argument level?
  - Each system proposes arguments, scored according to overall F_1 on development
  - The final score for an argument is the sum of scores given by systems
- Greedy Selection
  - Repeat, until no more arguments in the candidate list
    - Select argument candidate with the best score
    - Removing overlapping arguments from candidate list

Results & Discussion

[Carreras et al.] – Continued

WSJ-23	F_1	Prec.	Rec.
punyakanok+haghighi+pradhan	80.21	79.10	81.36
punyakanok	79.44	82.28	76.78

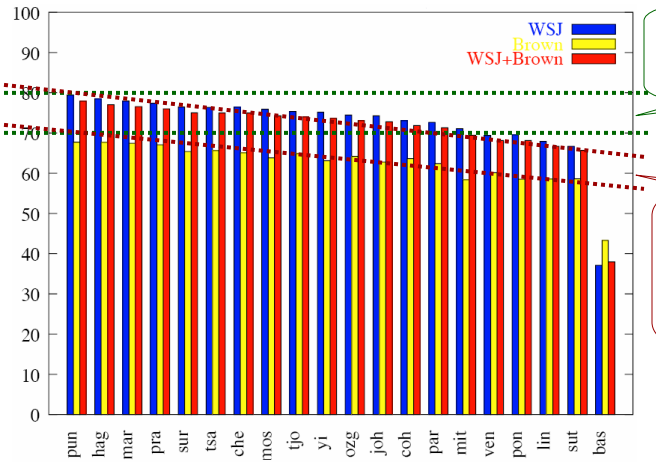
Brown	F_1	Prec.	Rec.
haghighi+marquez+pradhan+tsai	69.74	69.40	70.10
punyakanok	67.75	73.38	62.93

- The greedy method of combing systems increases recall but sacrifices precision.
- The gain on F_1 is not huge.

Part III: CoNLL-05 Shared Task on SRL

- ✓ Details of top systems and interesting systems
  - ✓ Introduce the top 4 systems
  - ✓ Describe 3 spotlight systems
- Analysis of the overall results
  - General performance
  - System properties
  - Per argument performance
- Directions for improving SRL systems

Results on WSJ and Brown Tests



F_1: 70% ~ 80%
Small differences

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

Figure from Carreras&Màrquez's slide (CoNLL 2005)

System Properties

- Learning Methods
  - SNoW, MaxEnt, AdaBoost, SVM, CRFs, etc.
  - ➢ *The choice of learning algorithms is less important.*
- Features
  - All teams implement more or less the standard features with some variations.
  - *A must-do for building a good system!*
  - *A clear feature study and more feature engineering will be helpful.*

System Properties – Continued

- Syntactic Information
  - Charniak's parser, Collins' parser, clauser, chunker, etc.
  - Top systems use Charniak's parser or some mixture
  - ➢ *Quality of syntactic information is very important!*
- System/Information Combination
  - 8 teams implement some level of combination
  - Greedy, Re-ranking, Stacking, ILP inference
  - ➢ *Combination of system or syntactic information is a good strategy to reduce the influence of incorrect syntactic information!*

Per Argument Performance

CoNLL-05 Results on WSJ-Test

- Core Arguments (Freq. ~70%)

	Best F_1	Freq.
A0	88.31	25.58%
A1	79.91	35.36%
A2	70.26	8.26%
A3	65.26	1.39%
A4	77.25	1.09%

- Adjuncts (Freq. ~30%)

	Best F_1	Freq.
TMP	78.21	6.86%
ADV	59.73	3.46%
DIS	80.45	2.05%
MNR	59.22	2.67%
LOC	60.99	2.48%
MOD	98.47	3.83%
CAU	64.62	0.50%
NEG	98.91	1.36%

Arguments that need to be improved

Data from Carreras&Màrquez's slides (CoNLL 2005)

Groups of Verbs in WSJ-Test

- By their frequencies in WSJ-Train

	0	1-20	21-100	101-500	501-1000
Verbs	34	418	359	149	18
Props	37	568	1098	1896	765
Args.	70	1049	2066	3559	1450

- CoNLL-05 Results on WSJ-Test – Core Arguments

	0	1-20	21-100	101-500	501-1000
Args. %	0.9	12.8	25.2	43.4	17.7
Best F_1	73.38	76.05	80.43	81.70	80.31

Arguments of low-frequency verbs need to be improved

Data from Carreras&Màrquez's slides (CoNLL 2005)

Part III: CoNLL-05 Shared Task on SRL

- ✓ Details of top systems and interesting systems
  - ✓ Introduce the top 4 systems
  - ✓ Describe 3 spotlight systems
- ✓ Analysis of the overall results
  - ✓ General performance
  - ✓ System properties
  - ✓ Per argument performance
- Directions for improving SRL systems

Directions for Improving SRL

- Better feature engineering
  - Maybe the most important issue in practice
- Joint modeling/inference
  - How to improve current approaches?
- Fine-tuned learning components
  - Can a more complicated system help?
- Cross domain robustness
  - Challenge to applying SRL systems

Better Feature Engineering

Gildea&Jurafsky '02
- Target predicate
- Voice
- Subcategorization
- Path
- Position (left, right)
- Phrase Type
- Governing Category
- Head Word

Surdeanu et al '03
- Content Word
- Head Word POS
- Content Word POS
- Named Entity

Xue&Palmer '04
- Feature conjunctions
- Syntactic frame
- Head of PP Parent

Pradhan et al '04
- Phrase Type / Head Word / POS of Left/Right/Parent constituent
- First/Last word/POS

- Individual feature contribution is not clear
  - Every set of features provide some improvement, but…
  - Different system, different corpus, different usage

Joint Model/Inference

- Unless pure local model reaches prefect results, joint model/inference often can improve the performance
- Greedy rules
  - ✓ Fast & Effective
  - × With no clear objective function
  - × Often increase recall by sacrificing precision
- Integer linear programming inference [Roth&Yih 04]
  - ✓ With clear objective function
  - ✓ Can represent fairly general *hard* constraints
  - × Statistical constraints are more expensive to integrate
- Joint Model [Toutanova et al. 05]
  - ✓ Capture statistical and hard constraints directly from the data
  - × Need re-ranking to avoid computational complexity problems

Fine-tuned Learning Components

- Separate core arguments and adjuncts
  - Adjuncts are independent of the target verb
  - Performance may be enhanced with specific features
- Train systems for different (groups of) verbs
  - Verbs (or senses) may have very different role sets
  - Example: stay.01(remain) vs. look.02 (seeming)

[$_{A1}$ Consumer confidence] **stayed** [$_{A3}$ strong] in October.

[$_{A0}$ The demand] **looked** [$_{A1}$ strong] in October.

Cross Domain Robustness

- The performance of SRL systems drops significantly when applied on a different corpus
  - ~10% F_1 from WSJ to Brown
  - The performance of all the syntactic taggers and parsers drops significantly
    - All trained on WSJ
- May not build a robust system without data
  - Semi-supervised learning
  - Active learning

Summary of Part III:
CoNLL-05 Shared Task on SRL

- Described the details of top performing SRL systems
  - Implement generally all *standard* features
  - Use good syntactic information – Charniak's parser & more
  - Deploy system/information combination schemes
  - Achieve ~80% F_1 on WSJ, ~70% F_1 on Brown
- Introduced some interesting systems
  - Train syntactic parser and argument identifier together
  - Apply Tree CRFs model
  - Investigate the performance of a large system combination

Summary of Part III:
CoNLL-05 Shared Task on SRL – Continued

- Analyzed the results of the CoNLL-05 systems
  - General performance
    - Performance on WSJ is between 70% and 80%
    - The differences among systems are small
    - Every system suffers from cross-domain test; ~10% F_1 drop on Brown corpus
  - Per argument performance
    - Core arguments A1 and A2 and some frequent adjunct arguments need to be improved
    - Arguments of low-frequency verbs need to be improved

Summary of Part III:
CoNLL-05 Shared Task on SRL – Continued

- Directions for improving SRL systems
  - Perform careful feature study
  - Design better features
  - Enhance current joint model/inference techniques
  - Separate models for different argument sets
  - Improve cross domain robustness
- **Next part**: Applications of SRL systems

Quick Overview

- ✓ Part I. Introduction
  - ✓ What is Semantic Role Labeling?
  - ✓ From manually created grammars to statistical approaches
    - ✓ Early Work
    - ✓ Corpora – FrameNet, PropBank, Chinese PropBank, NomBank
  - ✓ The relation between Semantic Role Labeling and other tasks
- ✓ Part II. General overview of SRL systems
  - ✓ System architectures
  - ✓ Machine learning models
- ✓ Part III. CoNLL-05 shared task on SRL
  - ✓ Details of top systems and interesting systems
  - ✓ Analysis of the results
  - ✓ Research directions on improving SRL systems
- Part IV. Applications of SRL

Part IV: Applications

- Information Extraction
  - Reduce development time
- Summarization
  - Sentence matching
- Question Answering
  - Understand questions better
- Textual Entailment
  - Deeper semantic representation

SRL in Information Extraction

[Surdeanu et al. 03]

- Information Extraction (HUB Event-99 evaluations, [Hirschman et al 99])
  - A set of domain dependent *templettes*, summarizing information about events from multiple sentences

<MARKET_CHANGE_1>:=	
INSTRUMENT	London [gold]
AMOUNT_CHANGE	fell [$4.70] cents
CURRENT_VALUE	$308.45
DATE:	daily

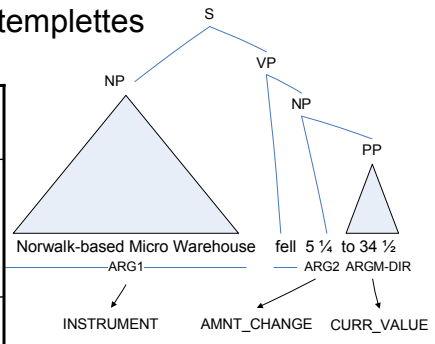
Time for our daily market report from NASDAQ.
London gold fell $4.70 cents to $308.45.

SRL in Information Extraction

[Surdeanu et al. 03]-Continued

- Find predicate argument relations and map resulting structures into templettes via hand-written simple rules

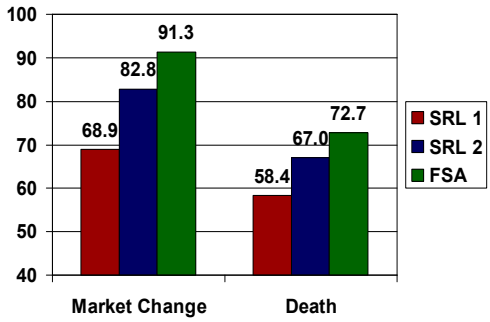
ARG1 and MARKET_CHANGE_VERB => INSTRUMENT
ARG2 and (MONEY or PERCENT or QAUNTITY) and MARKET_CHANGE_VERB => AMOUNT_CHANGE
(ARG4 or ARGM_DIR) and NUMBER and MARKET_CHANGE_VERB=> CURRENT_VALUE



SRL in Information Extraction

[Surdeanu et al. 03]-Continued

- Results
  - SRL 1
    - Identification 71.9
    - Classification 78.9
  - SRL 2
    - Identification 89.0
    - Classification 83.7
  - FSA is a traditional finite state approach



Better SRL leads to significantly better IE performance.

The FSA approach does better but requires intensive human effort (10 person days).

The systems using SRL require 2 hours of human effort.

SRL in Summarization (SQUASH, [Melli et al. 05])

- The task is to generate a 250-word summary
  - Given a specified topic and level of detail (specific, general)
- The system uses SRL extensively for:
  - Estimating a significance score for a sentence
    - which entities participate in which semantic relations
  - Estimating sentence similarity
    - which entities participating in which semantic relations are contained in two sentences

SRL in Summarization
(SQUASH, [Melli et al. 05]-Continued)

- It is not possible to remove just the SRL component from the system since SRL is used throughout
- Improving the SRL system improves Summarization performance (ROUGE-2 scores on the development set)
  - Naïve SRL **0.0699**
  - ASSERT SRL **0.0731**
- This is a pretty large improvement considering the impact of other successful features
  - Bias toward the first sentences 0.0714 → 0.0738
- The overall placement of an earlier version of SQUASH was 7th out of 25 systems in DUC 2005

SRL in Question Answering
[Narayanan & Harabagiu 04]

- Parsing Questions

***Q**: What kind of materials were stolen from the Russian navy?*

PAS(Q): What [$_{A1}$ kind of nuclear materials] were [Predicate:stolen] [$_{A2}$ from the Russian Navy]?

- Parsing Answers

***A(Q)**: Russia's Pacific Fleet has also fallen prey to nuclear theft; in 1/96, approximately 7 kg of HEU was reportedly stolen from a naval base in Sovetskaya Gavan.*

PAS(A(Q)): [$_{A1(P1)}$ Russia's Pacific Fleet] has [$_{AM-DIS(P1)}$ also] [P1: fallen] [$_{A1(P1)}$ prey to nuclear theft]; [$_{AM-TMP(P2)}$ in 1/96], [$_{A1(P2)}$ approximately 7 kg of HEU] was [$_{AM-ADV(P2)}$ reportedly] [P2: stolen] [$_{A2(P2)}$ from a naval base] [$_{A3(P2)}$in Sovetskawa Gavan]

- Result: exact answer= *"approximately 7 kg of HEU"*

Slide from Harabagiu and Narayanan (HLT 2004)

SRL in Question Answering

[Narayanan & Harabagiu 04]-Continued

- Parsing Questions

Q: What kind of materials were stolen from the Russian navy?

FS(Q): What [$_{GOODS}$ kind of nuclear materials] were [$_{Target\text{-}Predicate}$ stolen] [$_{VICTIM}$ from the Russian Navy]?

- Parsing Answers

***A(Q)**: Russia's Pacific Fleet has also fallen prey to nuclear theft; in 1/96, approximately 7 kg of HEU was reportedly stolen from a naval base in Sovetskaya Gavan.*

FS(A(Q)): [$_{VICTIM(P1)}$ Russia's Pacific Fleet] has also fallen prey to [$_{GOODS(P1)}$ nuclear] [$_{Target\text{-}Predicate(P1)}$ theft]; in 1/96, [$_{GOODS(P2)}$ approximately 7 kg of HEU] was reportedly [$_{Target\text{-}Predicate\ (P2)}$ stolen] [$_{VICTIM\ (P2)}$ from a naval base] [$_{SOURCE(P2)}$ in Sovetskawa Gavan]

- Result: exact answer= *"approximately 7 kg of HEU"*

Slide from Harabagiu and Narayanan (HLT 2004)

SRL in Question Answering

[Narayanan & Harabagiu 04]-Continued

- Evaluation of gains due to predicate-argument information.

Structure Used	Percent of Questions
Answer Hierarchy	12%
PropBank analyses	32%
FrameNet analyses	19%

Percent of questions for which the correct answer type was identified through using each structure.

- **Question:** What is the additional value compared to matching based on syntactic analyses?

SRL in Textual Entailment

[Braz et al. 05]

- Does a given text *S* entail a given sentence *T*
  - *S: The bombers had not managed to enter the building*
  - *T: The bombers entered the building*
- Evaluating subsumption by matching predicate argument structure
  - *S1: [$_{ARG0}$The bombers] had [$_{ARGM_NEG}$not] managed to [$_{PRED}$enter] [$_{ARG1}$ the building]*
  - *T1: [$_{ARG0}$The bombers] [$_{PRED}$entered] [$_{ARG1}$ the building]*

S does not entail *T* because they do not have the same set of arguments

SRL in Textual Entailment

[Braz et al. 05]-Continued

- SRL forms the basis of the algorithm for deciding entailment.
- It is also extensively used in *rewrite rules* which preserve semantic equivalence.
- Not possible to isolate the effect of SRL and unknown whether a syntactic parse approach can do similarly well.
- Results on the PASCAL RTE challenge 2005
  - Word based baseline: **54.7**
  - System using SRL and syntactic parsing: **65.9**
- The system placed 4th out of 28 runs by 16 teams

Summary of Part IV: Applications

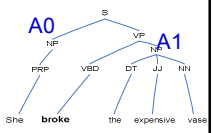
- Information Extraction
  - SRL has advantages in development time; good SRL → good IE
  - FSA systems are still about 10% better
- Summarization
  - Sophisticated sentence matching using SRL
  - Improving SRL improves summarization.
- Question Answering
  - Having more complex semantic structures increases the number of questions that can be handled about 3 times.
- Textual Entailment
  - SRL enables complex inferences which are not allowed using surface representations.
- **Action item**: evaluate contributions of SRL vs. syntactic parsing
  - None of the system performs a careful comparison

Conclusions

- Semantic Role Labeling is relatively new but has attracted a lot of interest
- Large corpora with annotated data are available
  - FrameNet, PropBank
- It provides a novel *broad-coverage* level of semantic interpretation
  - Shallower than some alternatives (Deep Parsing for limited and broad domains)
  - Deeper than others (Penn Treebank analyses with function tags)
- Tasks which profit from Penn Treebank syntactic analyses should profit from this semantic layer

Conclusions
Current State of the Art systems

- Achieve about **80%** per-argument F-measure (**60%** whole propositions correct)
  - Performance is respectable but still there is a lot of room for improvement
  - Inter-annotator agreement is **99%** for *all* nodes given *gold-standard* syntactic parses (chance agreement is **88%**); not comparable to system results



- Build on the strength of statistical parsing models
  - Perform poorly when the syntactic parsers do so
- Use syntactic information extensively
- Have mechanisms for increasing robustness to parser error
- Use powerful machine learning techniques
- Model dependencies among argument labels

Conclusions
Directions for Improving SRL

- Increase robustness to syntactic parser error
- Find ways to collect additional knowledge
  - Use unlabeled data
  - Share information across verbs
  - Can applications create more data for SRL automatically?
- Improve the statistical models
  - Other features, other dependencies
- Improve search/inference procedures

Conclusions
Major Challenges

- Need to connect SRL to natural language applications
  - Study the additional value of semantic labels compared to surface representations and syntactic analyses
  - Apply SRL to other applications
    - More Information Extraction applications
    - ATIS labeling and NL interfaces to databases
  - Have we defined the corpora well?
    - Validate the annotation standards through application domains
  - What level of accuracy is needed in order for SRL to be useful?

Final Remarks

- Semantic Role Labeling is an exciting area of research!
  - Progress is fast
  - There is still room for large contributions
- Provides robust broad-coverage semantic representations
- Easy integration with applications (Information Extraction, Question Answering, Summarization, Textual Entailment)
  - Good results in tasks
- Tools available online that produce SRL structures
  - ASSERT (Automatic Statistical SEmantic Role Tagger) http://oak.colorado.edu/assert
  - UIUC system (http://l2r.cs.uiuc.edu/~cogcomp/srl-demo.php)

References: Introduction

- Hiyan Alshawi, editor. The core language engine. *MIT Press, 1992.*
- Ion Androutsopoulos, Graeme Ritchie, and Peter Thanisch. Natural language interfaces to databases - an introduction. In *Journal of Natural Language Engineering 1(1), 1995.*
- Douglas Appelt and David Israel. Introduction to Information Extraction technology. *Tutorial at IJCAI 1999.*
- Don Blaheta and Eugene Charniak. Assigning function tags to parsed text. In *Proceedings of NAACL 2000.*
- Don Blaheta. Function tagging. *PhD Thesis, Brown CS Department, 2003.*
- Joan Bresnan. Lexical-functional syntax. *Blackwell, 2001.*
- Ann Copestake and Dan Flickinger. An open-source grammar development environment and broad-coverage English grammar using HPSG. In *Proceedings of LREC-2000.*
- John Dowding, Jean Gawron, Doug Appelt, John Bear, Lynn Cherny, Robert Moore, and Douglas Moran. Gemini: a natural language system for spoken-language understanding. In *Proceedings of ACL 1993.*

References: Introduction

- Charles J. Fillmore, Charles Wooters, and Collin F. Baker. Building a large lexical databank which provides deep semantics. In *Proceedings of the Pacific Asian Conference on Language, Information and Computation 2001.*
- Ruifang Ge and Raymond Mooney. A statistical semantic parser that integrates syntax and semantics. In *Proceedings of CoNLL 2005.*
- Graeme Hirst. Semantic interpretation and the resolution of ambiguity (Studies in natural language processing). *Cambridge University Press, 1987.*
- Lynette Hirschman, Patricia Robinson, Lisa Ferro, Nancy Chinchor, Erica Brown, Ralph Grishman, and Beth Sundheim. *Hub 4 Event99 general guidelines and templettes, 1999*.
- John T. Maxwell and Ronald M. Kaplan. The interface between phrasal and functional constraints. In *Computational Linguistics,19(4), 1993.*

References: Introduction

- Ion Muslea. Extraction patterns for Information Extraction tasks: a survey. *In Proceedings of the AAAI Workshop on Machine Learning for IE, 1999.*
- Yusuke Miyao and Jun'ichi Tsujii. Probabilistic disambiguation models for wide-coverage HPSG parsing. In *Proceedings of ACL 2005.*
- Scott Miller, Robert Bobrow, Robert Ingria, and Richard Schwartz. A Fully statistical approach to natural language interfaces. In *Proceedings of ACL 1996.*
- Martha Palmer, Dan Gildea, and Paul Kingsbury. The Proposition Bank: An annotated corpus of semantic roles. In *Computational Linguistics, 31(1), 2005.*
- Carl Pollard and Ivan A. Sag. Head-Driven Phrase Structure Grammar. *University of Chicago Press, 1994.*
- Patti Price. Evaluation of spoken language systems: the ATIS domain. *In Proceedings of the third DARPA Speech and Natural Language Workshop, 1990.*

References: Introduction

- Stefan Riezler, Tracy H. King, Ronald M. Kaplan, Richard Crouch, John T. Maxwell III, and Mark Johnson. Parsing the Wall Street Journal using a Lexical-Functional grammar and discriminative estimation techniques. *In Proceedings of ACL 2002.*
- Hisami Suzuki and Kristina Toutanova. Learning to predict case markers in Japanese. In *Proceedings of ACL-COLING* 2006.
- Kristina Toutanova, Penka Markova, and Christopher D. Manning. The leaf projection path view of parse trees: Exploring string kernels for HPSG parse selection. In *Proceedings of EMNLP 2004.*
- Kiyotaka Uchimoto, Satoshi Sekine and Hitoshi Isahara. Text generation from keywords. In *Proceedings of COLING 2002.*
- Ye-Yi Wang, John Lee, Milind Mahajan, and Alex Acero. Combining statistical and knowledge-based spoken language understanding in conditional models. In *Proceedings of ACL-COLING 2006.*

References: Introduction

- Ye-Yi Wang, Li Deng, and Alex Acero. Spoken language understanding: An introduction to the statistical framework. In *IEEE Signal Processing Magazine, Vol 27 No. 5. 2005.*
- Wayne Ward. Recent Improvements in the CMU spoken language understanding system. In *Proceedings of Human Language Technology Workshop, 1994.*
- Yuk Wah Wong and Raymond Mooney. Learning for semantic parsing with statistcial machine translation. In *Proceedings of HLT/NAACL 2006.*
- John Zelle and Raymond Mooney. Learning to parse database queries using inductive logic programming. In *Proceedings of AAAI 1996.*
- Luke Zettlemoyer and Michael Collins. Learning to map sentences to logical form: structured classification with probabilistic Categorial Grammars. In *Proceedings of UAI 2005.*

References: Overview of SRL Systems

- John Chen and Owen Rambow. Use of deep linguistic features for the recognition and labeling of semantic arguments. In *Proceedings of EMNLP 2003.*
- Xavier Carreras and Lluís Màrquez. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *Proceedings of CoNLL 2005.*
- Trevor Cohn and Philip Blunsom. Semantic role labelling with tree Conditional Random Fields. In *Proceedings of CoNLL 2005.*
- Daniel Gildea and Daniel Jurafsky. Automatic labeling of semantic roles. In *Computational Linguistics, 28(3), 2002.*
- Daniel Gildea and Martha Palmer. The necessity of parsing for predicate argument recognition. In *Proceedings of ACL 2002* .
- Daniel Gildea and Julia Hockenmaier. Identifying semantic roles using Combinatory Categorial Grammar. In *Proceedings of EMNLP 2003.*

References: Overview of SRL Systems

- Aria Haghighi, Kristina Toutanova, and Christopher Manning. A joint model for semantic role labeling. In *Proceedings of CoNLL 2005.*
- Lluís Màrquez, Pere Comas, Jesús Giménez, and Neus Català. Semantic role labeling as sequential tagging. In *Proceedings of CoNLL 2005.*
- Sameer Pradhan, Wayne Ward, Kadri Hacioglu, James H. Martin and Dan Jurafsky. Semantic role labeling using different syntactic views. In *Proceedings of ACL 2005.*
- Sameer Pradhan, Wayne Ward, Kadri Hacioglu, James Martin, and Dan Jurafsky. Shallow semantic parsing using Support Vector Machines. In *Proceedings of HLT 2004.*
- Vasin Punyakanok, Dan Roth, Wen-tau Yih and Dav Zimak. Semantic role labeling via Integer Linear Programming inference. In *Proceedings of COLING 2004.*

References: Overview of SRL Systems

- Vasin Punyakanok, Dan Roth, and Wen-tau Yih. The necessity of syntactic parsing for semantic role labeling. In *Proceedings of IJCAI 2005*.
- Mihai Surdeanu, Sanda Harabagiu, John Williams, and Paul Aarseth. Using predicate-argument structures for Information Extraction. In *Proceedings of ACL 2003*.
- Kristina Toutanova. Effective statistical models for syntactic and semantic disambiguation. *PhD Thesis, Stanford CS Department, 2005.*
- Kristina Toutanova, Aria Haghighi, and Christopher D. Manning. Joint learning improves semantic role labeling. In *Proceedings of ACL 2005*.
- Nianwen Xue and Martha Palmer. Calibrating features for semantic role labeling. In *Proceedings of EMNLP 2004*.

References: CoNLL-05 Shared Task on SRL

- Xavier Carreras and Lluís Màrquez. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *Proceedings of CoNLL 2005*.
- Trevor Cohn and Philip Blunsom. Semantic role labelling with tree Conditional Random Fields. In *Proceedings of CoNLL-2005*.
- Michael Collins and Terry Koo. Discriminative reranking for natural language parsing. In *Computational Linguistics 31(1), 2005.*
- Daniel Gildea and Daniel Jurafsky. Automatic labeling of semantic roles. In *Computational Linguistics, 28(3), 2002*.
- Kadri Hacioglu and Wayne Ward. Target word detection and semantic role chunking using Support Vector Machines. In *Proceedings of HLT-NACCL 2003.*
- Aria Haghighi, Kristina Toutanova, and Christopher Manning. A Joint model for semantic role labeling. In *Proceedings of CoNLL-2005.*

References: CoNLL-05 Shared Task on SRL

- Peter Koomen, Vasin Punyakanok, Dan Roth, and Wen-tau Yih. Generalized inference with multiple semantic role labeling systems. In *Proceedings of CoNLL-2005.*
- Taku Kudo and Yuji Matsumoto. Chunking with Support Vector Machines. In *Proceedings of NAACL 2001.*
- Lluís Màrquez, Pere Comas, Jesús Giménez, and Neus Català. Semantic role labeling as sequential tagging. In *Proceedings of CoNLL 2005.*
- Sameer Pradhan, Wayne Ward, Kadri Hacioglu, James Martin, and Dan Jurafsky. Shallow semantic parsing using Support Vector Machines. In *Proceedings of HLT 2004.*
- Sameer Pradhan, Kadri Hacioglu, Wayne Ward, James H. Martin, and Daniel Jurafsky. Semantic role chunking combining complementary syntactic views. In *Proceedings of CoNLL 2005.*
- Dan Roth and Wen-tau Yih. A Linear Programming formulation for global inference in natural language tasks. In *Proceedings of COLING 2004*.

References: CoNLL-05 Shared Task on SRL

- Mihai Surdeanu, Sanda Harabagiu, John Williams, and Paul Aarseth. Using predicate-argument structures for Information Extraction. In *Proceedings of ACL 2003*.
- Kristina Toutanova, Aria Haghighi, and Christopher D. Manning. Joint learning improves semantic role labeling. In *Proceedings of ACL 2005*.
- Szu-ting Yi and Martha Palmer. The integration of syntactic parsing and semantic role labeling. In *Proceedings of CoNLL 2005.*
- Nianwen Xue and Martha Palmer. Calibrating features for semantic role labeling. In *Proceedings of EMNLP 2004*.

References: Applications

- Rodrigo de Salvo Braz, Roxana Girju, Vasin Punyakanok, Dan Roth, and Mark Sammons. An inference model for semantic entailment in natural language. In *Proceedings of AAAI 2005*.
- Lynette Hirschman, Patricia Robinson, Lisa Ferro, Nancy Chinchor, Erica Brown, Ralph Grishman, and Beth Sundheim. *Hub 4 Event99 general guidelines and templettes, 1999*.
- Gabor Melli, Yang Wang, Yudong Liu, Mehdi M. Kashani, Zhongmin Shi, Baohua Gu, Anoop Sarkar and Fred Popowich. Description of SQUASH, the SFU question answering summary handler for the DUC-2005 summarization task. In *Proceedings of DUC 2005*.
- Srini Narayanan and Sanda Harabagiu. Question answering based on semantic structures. In *Proceedings of COLING 2004.*
- Mihai Surdeanu, Sanda Harabagiu, John Williams, and Paul Aarseth. Using predicate-argument structures for Information Extraction. In *Proceedings of ACL 2003*.