Machine Learning applied to Natural Language Processing

Lluís Màrquez

amazon

Advanced Methods for Corpus Analysis

EM LCT – European Masters Program in Language and Communication Technologies

Donostia, June 6-8, 2018

amazon

Talk Overview

amazon

Natural Language Processing Applications

- Typical NLP applications:
  - ⇒ Machine Translation
  - ⇒ CLIR and document managment
  - ⇒ Information Extraction
  - ⇒ Modern Question Answering (e.g.,Watson, cQA, WebQA), virtual assistants and information search
  - ⇒ Machine Reading
  - ⇒ Document Summarization (multidocument, multilingual)
  - ⇒ Intelligent dialog systems; Conversational AI (Google Duplex)
- Different levels of linguistic knowledge and understanding are required
- Systems need to resolve a number of basic subproblems

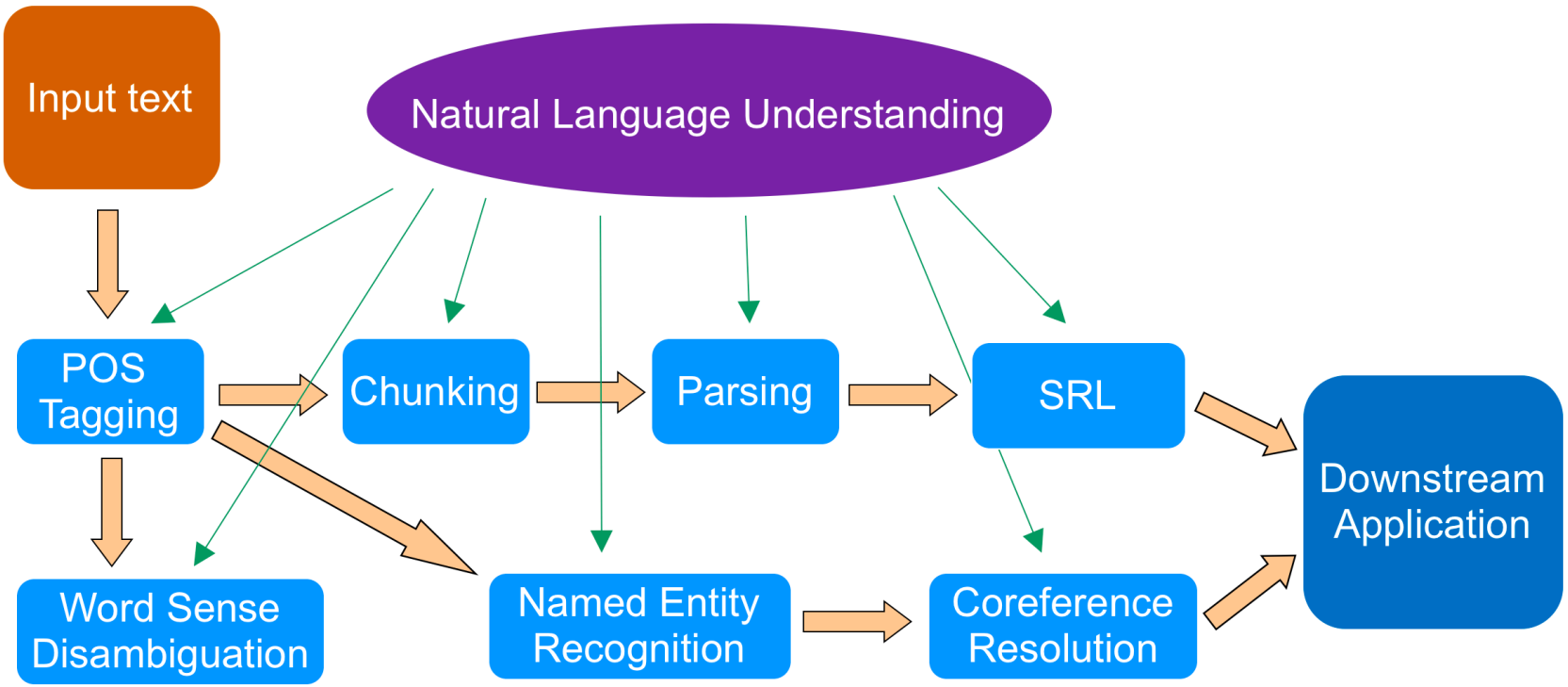
amazon

Natural Language Processing Applications

- Typical NLP applications:
  - ⇒ Machine Translation
  - ⇒ CLIR and document managment
  - ⇒ Information Extraction
  - ⇒ Modern Question Answering (e.g.,Watson, cQA, WebQA), virtual assistants and information search
  - ⇒ Machine Reading
  - ⇒ Document Summarization (multidocument, multilingual)
  - ⇒ Intelligent dialog systems; Conversational AI (Google Duplex)
- Different levels of linguistic knowledge and understanding are required
- Systems need to resolve a number of basic subproblems

amazon

The Pipeline Approach



amazon

Natural Language Processing Problems

Simple Idea:

- Mapping from an input to an output structure

  $\Rightarrow$ The input structure is typically a sequence of words, which might be enriched with some linguistic information

  $\Rightarrow$ Output structures are sequences, trees, graphs, etc.

Natural Language Processing Problems$_{(1)}$

Part–of–Speech Tagging

The San Francisco Examiner issued a special edition around noon yesterday that was filled entirely with earthquake new and information.

Natural Language Processing Problems$_{(1)}$

Part–of–Speech Tagging

The_DT San_NNP Francisco_NNP Examiner_NNP issued_VBD a_DT special_JJ edition_NN around_IN noon_NN yesterday_NN that_WDT was_VBD filled_VBN entirely_RB with_IN earthquake_NN news_NN and_CC information_NN ._.

POS tagging is a pure sequential labeling problem

(sequential learning paradigm)

But... are really words ambiguous with respect to POS?

Natural Language Processing Problems$_{(1)}$

Part–of–Speech Tagging

The_DT San_NNP Francisco_NNP Examiner_NNP issued_VBD a_DT special_JJ edition_NN around_IN noon_NN yesterday_NN that_WDT was_VBD filled_VBN entirely_RB with_IN earthquake_NN news_NN and_CC information_NN ._.

POS tagging is a pure sequential labeling problem

(sequential learning paradigm)

But... are really words ambiguous with respect to POS?

Natural Language Processing Problems$_{(1)}$

Part–of–Speech Tagging

The_DT San_NNP Francisco_NNP Examiner_NNP issued_VBD a_DT special_JJ edition_NN around_IN noon_NN yesterday_NN that_WDT was_VBD filled_VBN entirely_RB with_IN earthquake_NN news_NN and_CC information_NN ._.

But... are really words ambiguous with respect to POS?

YES! Let's take a look at a free on-line demo: FreeLing
http://nlp.lsi.upc.edu/freeling/demo/demo.php

Natural Language Processing Problems$_{(2)}$

Syntactic Analysis (Constituency parsing)

Pierre Vinken, 61 years old, will join the board as a nonexecutive director Nov. 29.

Natural Language Processing Problems$_{(2)}$

Syntactic Analysis (Constituency parsing)

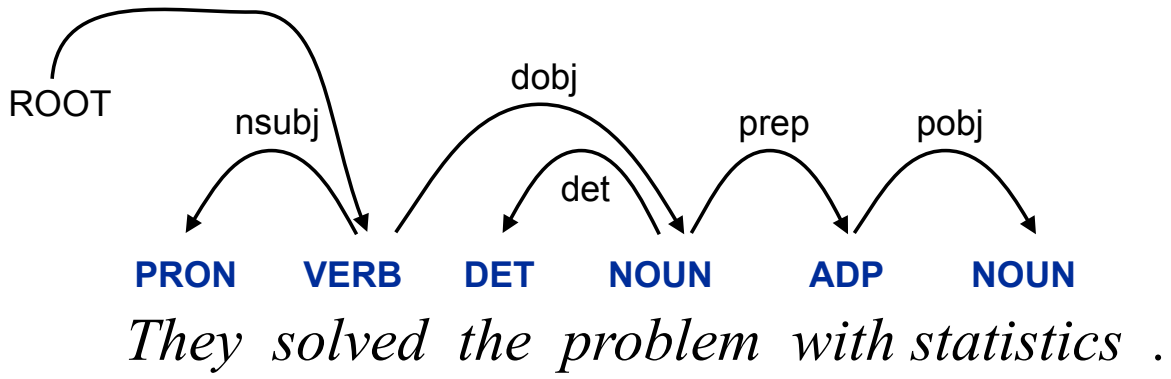
Pierre Vinken, 61 years old, will join the board as a nonexecutive director Nov. 29.

```
((S (NP-SBJ
        (NP (NNP Pierre) (NNP Vinken) )
        (, ,)
        (ADJP
            (NP (CD 61) (NNS years) )
            (JJ old) )
        (, ,) )
    (VP (MD will)
        (VP (VB join)
            (NP (DT the) (NN board) )
            (PP-CLR (IN as)
                (NP (DT a) (JJ nonexecutive) (NN director) ))
            (NP-TMP (NNP Nov.) (CD 29) )))
    (. .) ))
```

amazon

Natural Language Processing Problems$_{(2)}$

Dependency Parsing



amazon

Natural Language Processing Problems$_{(3)}$

Shallow Parsing (Chunking)

He reckons the current account deficit will narrow to only 1.8 billion in September.

Natural Language Processing Problems$_{(3)}$

Shallow Parsing (Chunking)

[NP He] [VP reckons] [NP the current account deficit] [VP will narrow] [PP o] [NP only 1.8 billion] [PP in] [NP September] .

Chunking is a sequential phrase recognition task

It can be seen as a sequential labeling problem (B-I-O encoding)

He_B-NP reckons_B-VP the_B-NP current_I-NP account_I-NP deficit_I-NP will_B-VP narrow_I-VP to_B-PP only_B-NP 1.8_I-NP billion_I-NP in_B-PP September_B-NP ._O

this is simple and usually effective

Natural Language Processing Problems$_{(3)}$

Shallow Parsing (Chunking)

[NP He] [VP reckons] [NP the current account deficit] [VP will narrow] [PP o] [NP only 1.8 billion] [PP in] [NP September] .

Chunking is a sequential phrase recognition task

It can be seen as a sequential labeling problem (B-I-O encoding)

He_B-NP reckons_B-VP the_B-NP current_I-NP account_I-NP deficit_I-NP will_B-VP narrow_I-VP to_B-PP only_B-NP 1.8_I-NP billion_I-NP in_B-PP September_B-NP ._O

this is simple and usually effective

Natural Language Processing Problems$_{(4)}$

Clause splitting (partial parsing)

The deregulation of railroads and trucking companies that began in 1980 enabled shippers to bargain for transportation.

Natural Language Processing Problems$_{(4)}$

Clause splitting (partial parsing)

(S The deregulation of railroads and trucking companies (SBAR that (S began in 1980)) enabled (S shippers to bargain for transportation) .)

Natural Language Processing Problems$_{(4)}$

Clause splitting (partial parsing)

(S The deregulation of railroads and trucking companies
(SBAR that
(S began in 1980))
enabled
(S shippers to bargain for transportation)
.)

Clauses may embed: they form a hierarchy

Clause splitting is a hierarchical prhase recognition problem

Not a good idea to treat it as a sequential problem...

amazon

Natural Language Processing Problems$_{(5)}$

Semantic Role Labeling (shallow semantic parsing)

He wouldn't accept anything of value from those he was writing about.

Natural Language Processing Problems$_{(5)}$

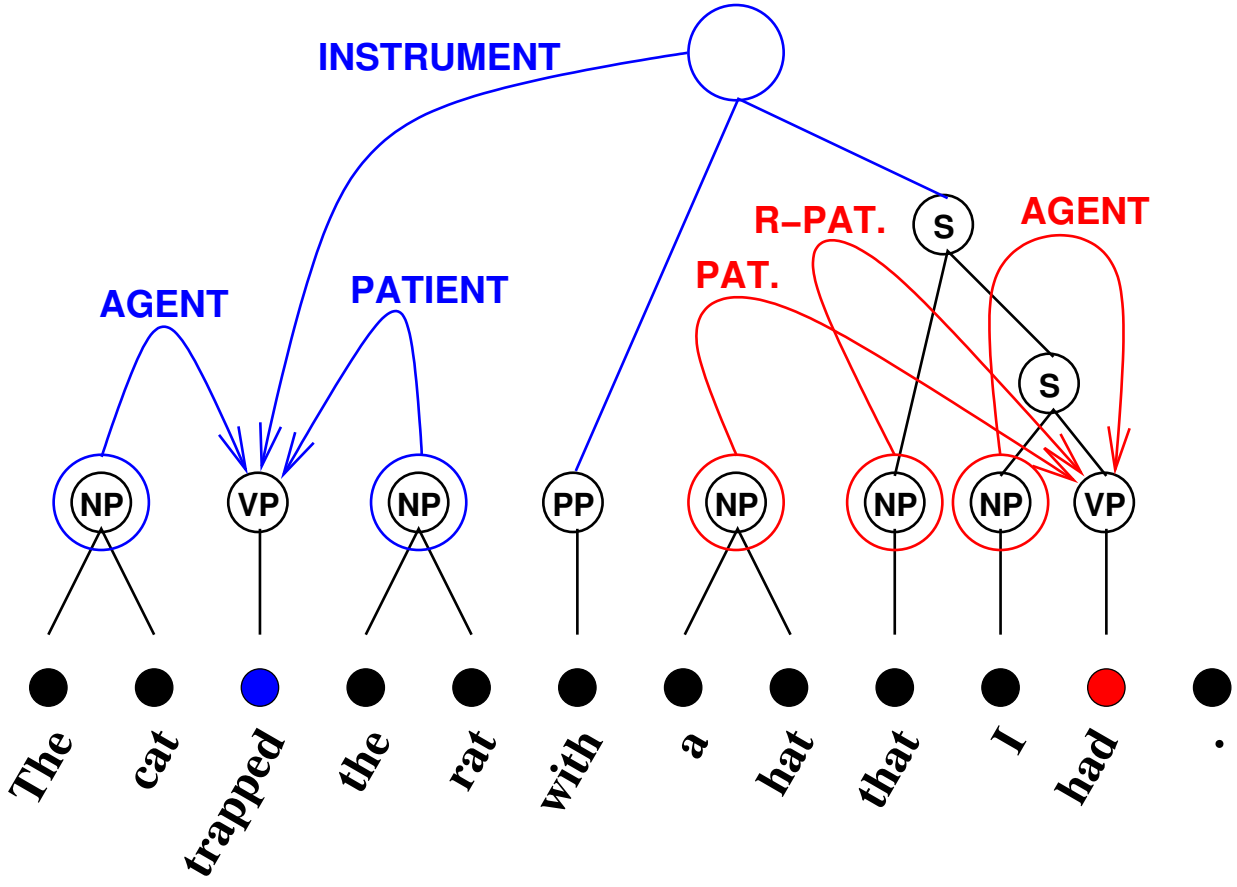
Semantic Role Labeling (shallow semantic parsing)

[A_0 He] [AM-MOD would] [AM-NEG n't] [V accept] [A_1 anything of value] from [A_2 those he was writing about] .

Roles for the predicate accept (PropBank Frames scheme):

V: verb; A_0: acceptor; A_1: thing accepted; A_2: accepted-from; A_3: attribute; AM-MOD: modal; AM-NEG: negation;

Natural Language Processing Problems(5)



amazon

Natural Language Processing Problems$_{(6)}$

Named Entity Extraction ("semantic chunking")

Wolff, currently a journalist in Argentina, played with Del Bosque in the final years of the seventies in Real Madrid.

Natural Language Processing Problems$_{(6)}$

Named Entity Extraction ("semantic chunking")

[PER Wolff] , currently a journalist in [LOC Argentina] , played with [PER Del Bosque] in the final years of the seventies in [ORG Real Madrid] .

- Named Entities may be embedded
- NE tracing: variants and co-reference resolution
- Relations between entities: event extraction

Natural Language Processing Problems(7)

Named Entities, relations, events, etc.

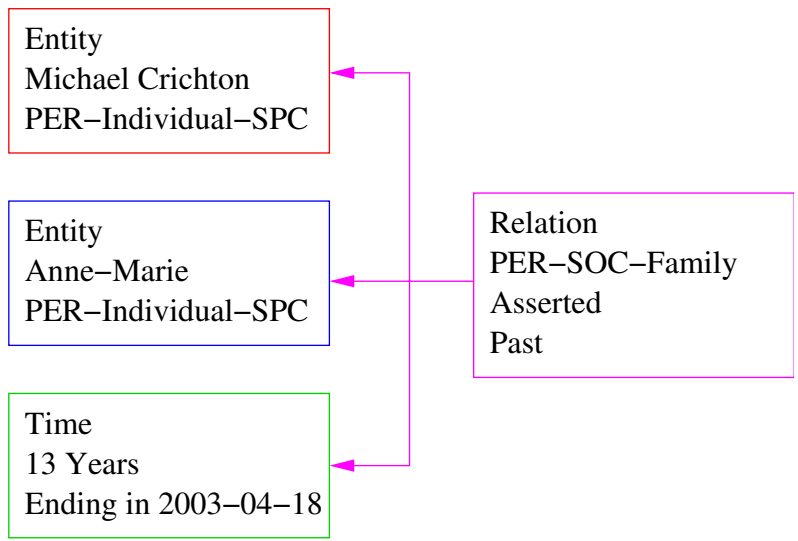
(example from the ACE corpus)

LOS ANGELES, April 18 (AFP)

Best–selling novelist and "Jurassic Park" creator Michael Crichton has agreed to pay his fourth wife 31 million dollars as part of their divorce settlement, court documents showed Friday.

Crichton, 60, is one of the world's wealthiest authors, and has had 12 of his novels made into major Hollywood movies.

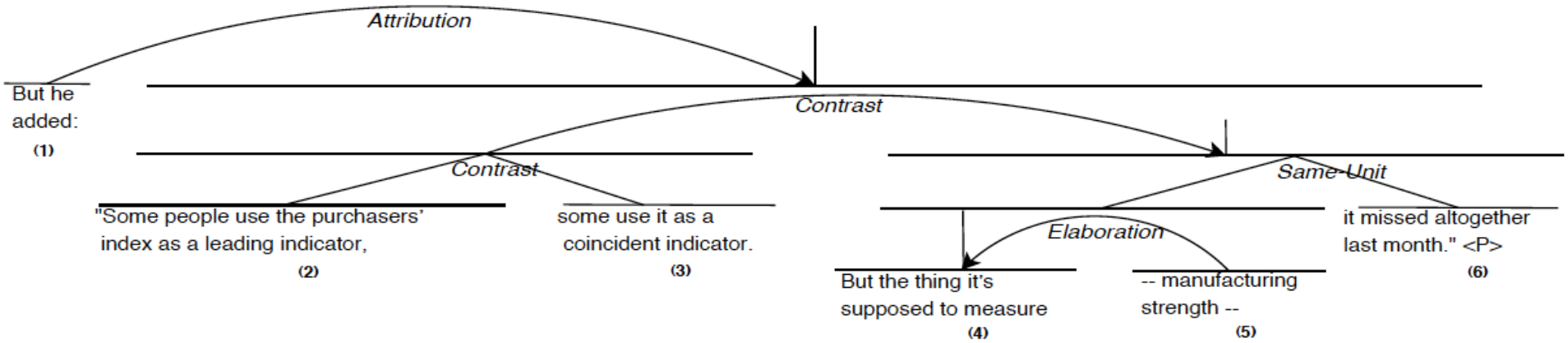
The writer will retain the rights to his books and films, although he has agreed to split a raft of other possessions with Anne Marie, his wife of 13 years, according to documents filed in Los Angeles Superior Court.



amazon

Natural Language Processing Problems(8)

Discourse Parsing



amazon

Natural Language Processing Problems

Recall the take away message:

- Mapping from an input to an output structure

  $\Rightarrow$ The input structure is typically a sequence of words, which might be enriched with some linguistic information

  $\Rightarrow$ Output structures are sequences, trees, graphs, etc.

- Machine Learning and Search (inference) are in between

NLP Meets Machine Learning

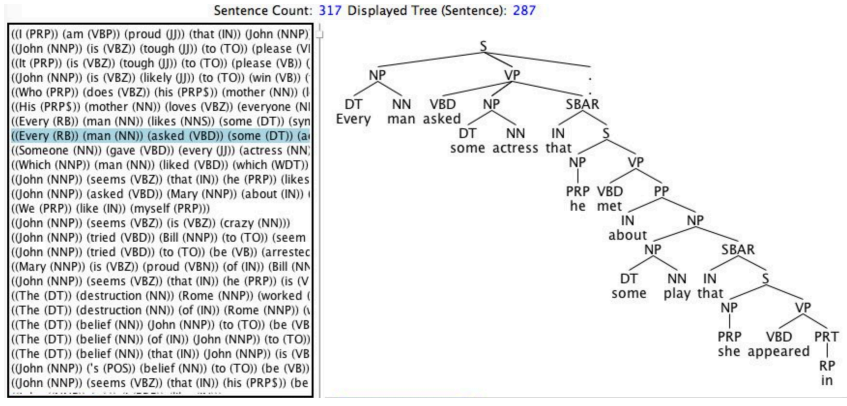
- 1980's resurgence of the empirical paradigm for NLP
- 1990's massive application of Machine Learning techniques
- Important factor (among others):

> Ambiguity resolution can be directly casted as classification

- NLP community learnt how to model and train local decisions
- Note 1: There is a big gap between classification and structure learning. Pure classification tasks don't really exist!
- Note 2: Search is strongly related to the generation of the output structure (*decoding*, *inference*, etc.)

Traditional Learning Approach

- Supervised Learning: data-driven, from annotated examples
  - ⇒ Need of manually annotating a large set of input-output examples (e.g., treebank)



- Vector representation based on explicit feature engineering
  - ⇒ counts, n-grams (e.g., words, POS), structure factors (e.g., dependencies, subtrees), linguistic properties, etc.
  - ⇒ sparse representation

amazon

Current Trend

End-to-end learning with neural networks

- Distributed representation of the input (dense embeddings)
- Avoids explicit feature engineering
- The intermediate representations are hidden (latent)

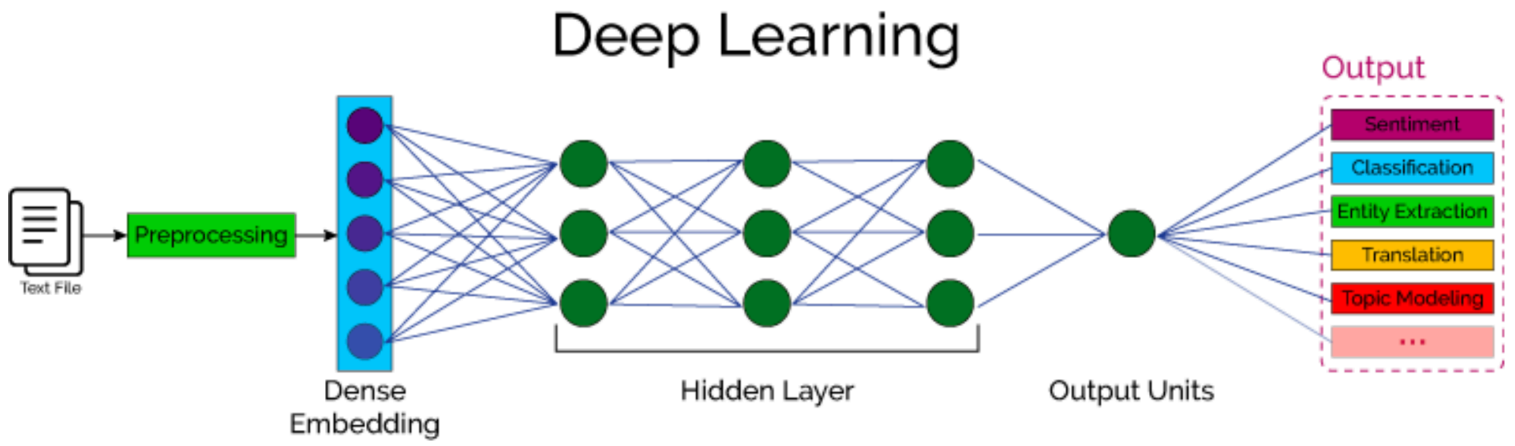


Figure from www.upwork.com

amazon

Other Learning Variants

- Unsupervised learning
- Semi-supervised learning
- Weakly-supervised learning
- Learning with distant supervision
- Transfer learning
- Multi-task learning
- Domain adaptation
- etc.

Why applying Machine Learning?

- Low cost development of NLP tools
- Language (quasi)independence: reusability
- Ability of acquiring/discovering knowledge from very large datasets
- Assist manual development of linguistic resources

amazon

On-line Demos and Software Suites

There are many available these days:

- FreeLing. Universitat Politè cnica de Catalunya. Basic syntactic processing. Catalan, Spanish, English and others. http://nlp.lsi.upc.edu/freeling/demo/demo.php
- CCG tools. University of Illinois at Urbana-Champaign. Multiple processors and applications. English. http://cogcomp.cs.illinois.edu/page/demos/
- Stanford NLP Group. Statistical NLP, deep learning NLP, and rule-based NLP tools for major computational linguistics problems. http://nlp.stanford.edu/software/
- Berkeley NLP Group. Statistical NLP tools for many problems, especially parsing. http://nlp.cs.berkeley.edu/software.shtml

amazon

Talk Overview

amazon

The Problem

Semantic Role Labeling

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

The Problem

IE point of view

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

[The luxury auto maker]$_{AGENT}$ [last year]$_{TEMP}$ sold$_{P}$ [1,214 cars]$_{OBJECT}$ [in the U.S.]$_{LOCATIVE}$

The Problem

IE point of view

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

[The luxury auto maker]$_{AGENT}$ [last year]$_{TEMP}$ sold$_{P}$ [1,214 cars]$_{OBJECT}$ [in the U.S.]$_{LOCATIVE}$

The Problem

Syntactic variations

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

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

$\Rightarrow$ All of them share the same semantic representation:

hit(Kristina,Scott,yesterday,with a baseball)

Example from (Yih & Toutanova, 2006)

amazon

The Problem

Syntactic variations

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

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

$\Rightarrow$ All of them share the same semantic representation:

hit(Kristina,Scott,yesterday,with a baseball)

Example from (Yih & Toutanova, 2006)

amazon

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$_{B-RECIP}$ me$_{I}$ this$_{B-TMP}$ morning$_{I}$.$_{O}$

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$_{B-RECIP}$ me$_{I}$ this$_{B-TMP}$ morning$_{I}$.$_{O}$

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

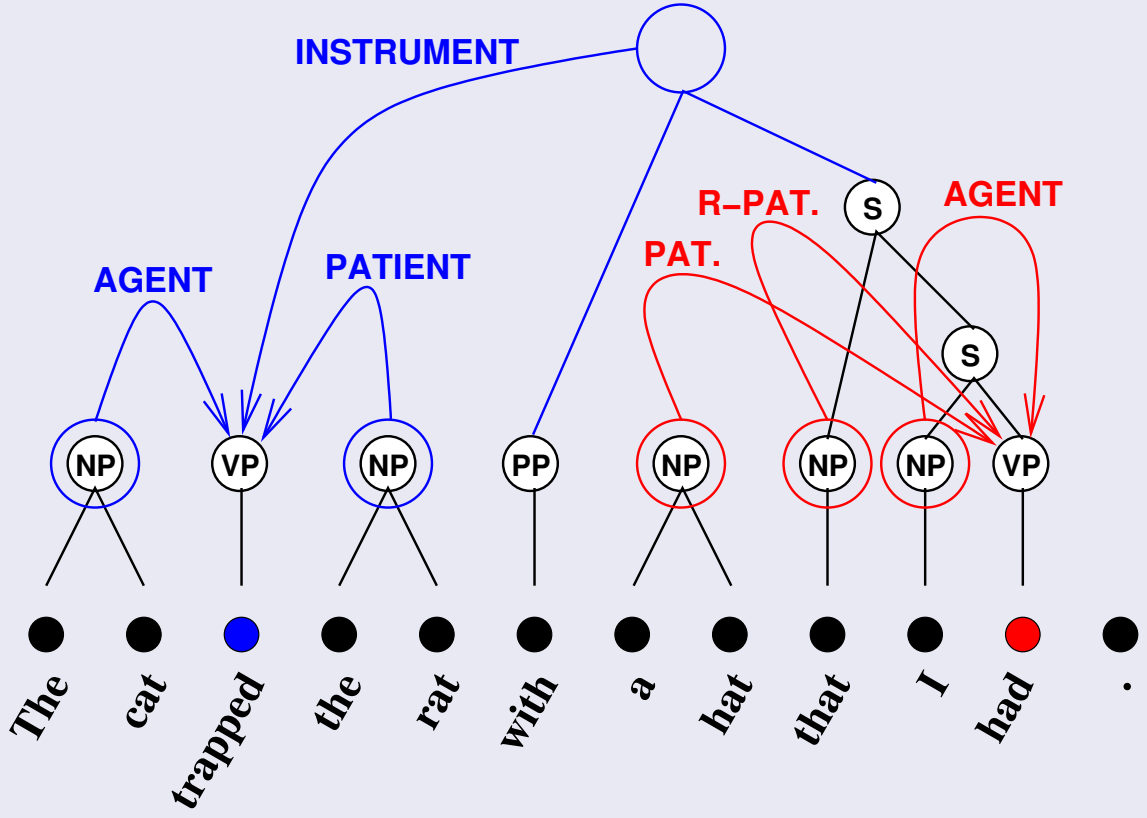
" Mr. Smith sent the report to me this morning . "

[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$_{B-RECIP}$ me$_{I}$ this$_{B-TMP}$ morning$_{I}$.$_{O}$

The Problem

Structural View

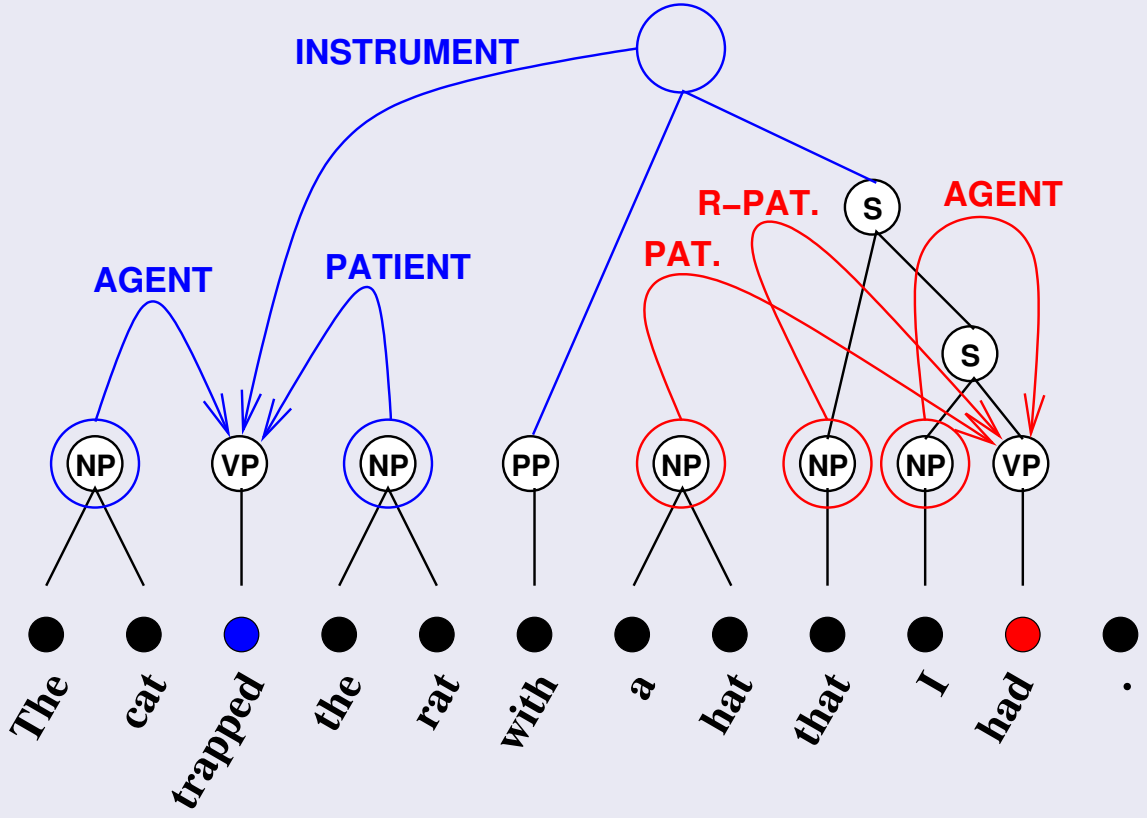


Output is a *hierarchy of labeled arguments*

amazon

The Problem

Structural View



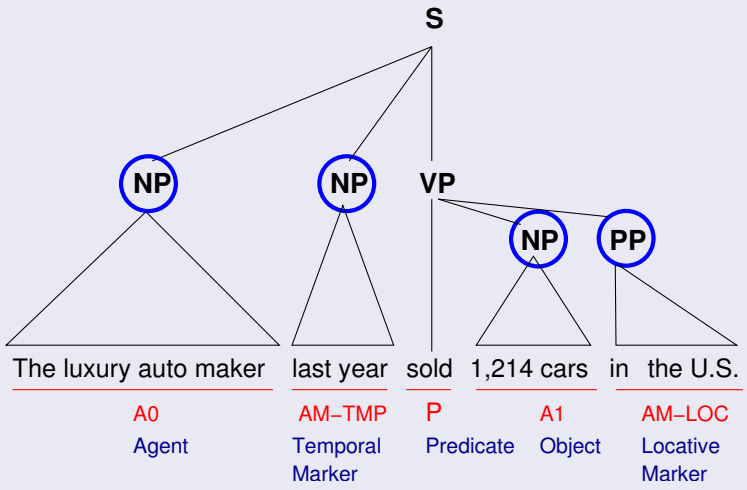
Output is a *hierarchy of labeled arguments*

amazon

The Problem

Linguistic nature of the problem

- Argument identification is strongly related to syntax

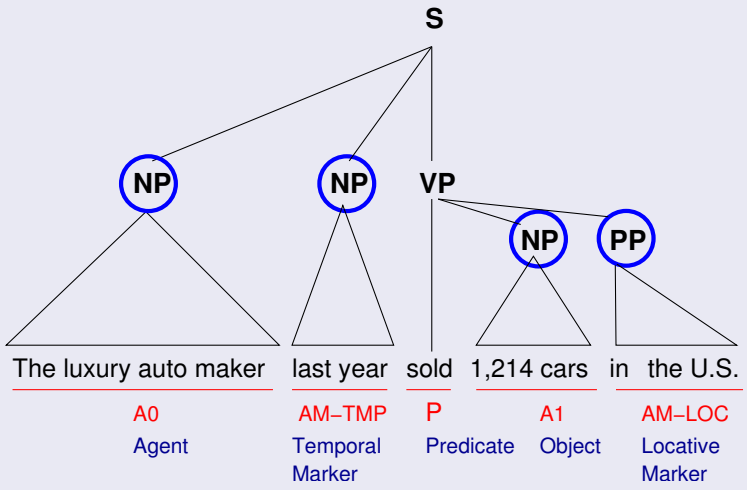


- Role labeling is a semantic task
  (e.g., selectional preferences could play an important role)

The Problem

Linguistic nature of the problem

- Argument identification is strongly related to syntax



- Role labeling is a semantic task

  (e.g., selectional preferences could play an important role)

SRL Systems Available

- ASSERT (Automatic Statistical SEmantic Role Tagger)
  http://cemantix.org/assert.html
- UIUC system demo
  http://l2r.cs.uiuc.edu/~cogcomp/srl-demo.php
- SwiRL: state-of-the-art system from CoNLL-2005
  http://www.surdeanu.name/mihai
- Shalmaneser: FrameNet-based system from SALSA project
  http://www.coli.uni-saarland.de/projects/salsa/shal/
- SEMAFOR: Probabilistic Frame(Net)-Semantic Parser
  http://www.ark.cs.cmu.edu/SEMAFOR/
- Brutus: A CCG-based Semantic Role Labeler
  http://www.ling.ohio-state.edu/~boxwell/software/brutus.html

Corpora Resources

- (English) PropBank
  http://verbs.colorado.edu/~mpalmer/projects/ace.html
- FrameNet
  http://framenet.icsi.berkeley.edu
- Korean PropBank
  http://www.ldc.upenn.edu/
- Chinese PropBank
  http://verbs.colorado.edu/chinese/cpb/
- 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/

amazon

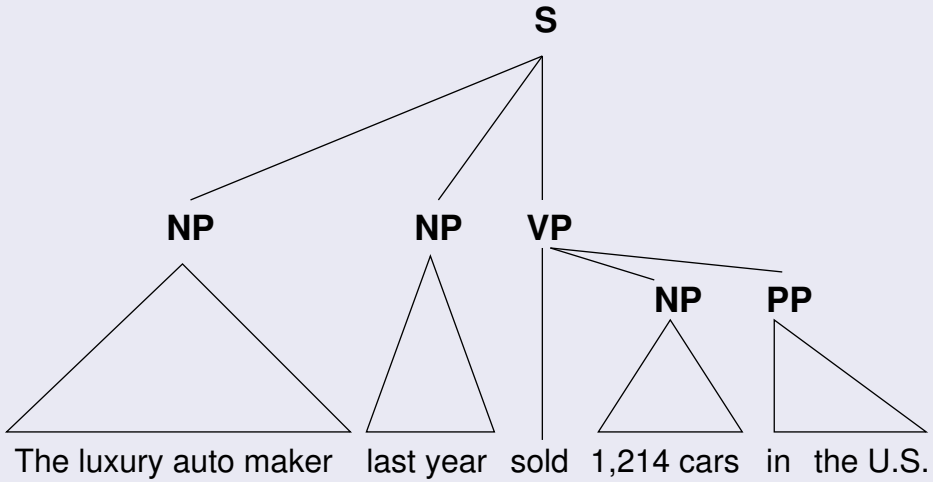
Corpora Resources

- (English) PropBank
  http://verbs.colorado.edu/~mpalmer/projects/ace.html
- FrameNet
  http://framenet.icsi.berkeley.edu
- Korean PropBank
  http://www.ldc.upenn.edu/
- Chinese PropBank
  http://verbs.colorado.edu/chinese/cpb/
- 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/

Corpora Resources

PropBank (Palmer et al., 2005)

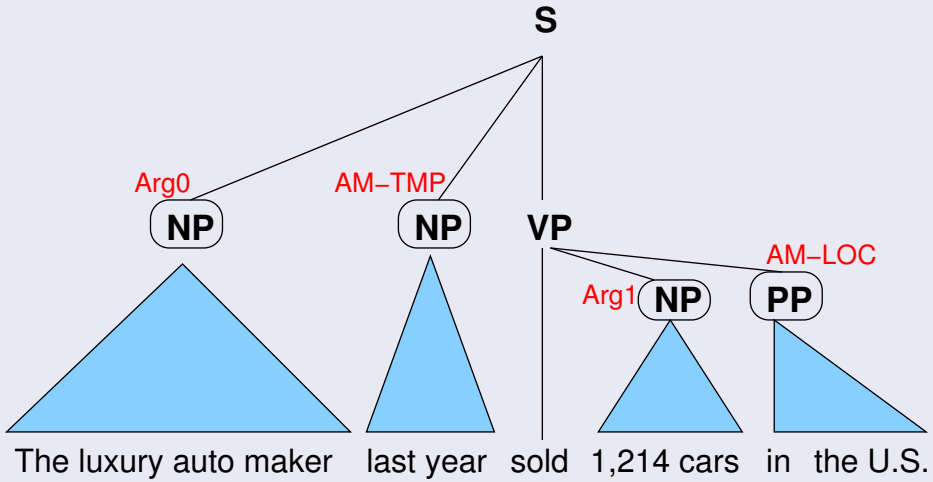
- Syntax-based approach: explaining the varied expression of verb arguments within syntactic positions
- Annotation of all verbal predicates in WSJ (Penn Treebank)
- http://verbs.colorado.edu/~mpalmer/projects/ace.html
- Add a semantic layer to the Syntactic Trees



Corpora Resources

PropBank (Palmer et al., 2005)

- Syntax-based approach: explaining the varied expression of verb arguments within syntactic positions
- 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



Corpora Resources

PropBank (Palmer et al., 2005)

- Theory neutral numbered 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.)

Corpora Resources

PropBank (Palmer et al., 2005)

- Theory neutral numbered 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.)

Corpora Resources

PropBank (Palmer et al., 2005)

- Theory neutral numbered 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.)

Corpora Resources

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]$_{ArgM-TMP}$

Corpora Resources

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]$_{Arg0}$ sell [it]$_{Arg1}$ [for $60 a bottle]$_{Arg3}$

- sell.02: give up
  Arg0="entity selling out"

  [John]$_{Arg0}$ sell out

- sell.03: sell until none is/are left
  Arg0="seller"; Arg1="thing sold"; ...

  [The new Harry Potter]$_{Arg1}$ sell out [within 20 minutes]$_{ArgM-TMP}$

Applications

Examples of applications of SRL (I)

- Information Extraction (Surdeanu et al., 2003)
- Question & Answering (Narayanan and Harabagiu, 2004; Frank et al., 2007; Shen and Lapata, 2007)
- Automatic Summarization (Melli et al., 2005)
- Coreference Resolution (Ponzetto and Strube, 2006)
- Text Categorization (Person et al., 2010)
- Opinion Expression Detection (Johansson and Moschitti, 2010)

Applications

Examples of applications of SRL (II)

- 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)

Empirical Evaluation of SRL Systems

Evaluation Exercises

- More than 10 evaluation exercises since 2004
  - ⇒ 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)
  - ⇒ SemEval-2010 (Ruppenhofer et al., 2010), etc.

Talk Overview

amazon

SRL Architecture: 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)
  - $\Rightarrow$ 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)
  - $\Rightarrow$ Matching is still ~90% when using automatic parsers

SRL Architecture: 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 Architecture: 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 Architecture: 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
  - $\Rightarrow$ Computational reasons but also modularity in feature engineering
- Many ML paradigms have been used: not big differences
- Features are more important

amazon

SRL Architecture: Step by Step

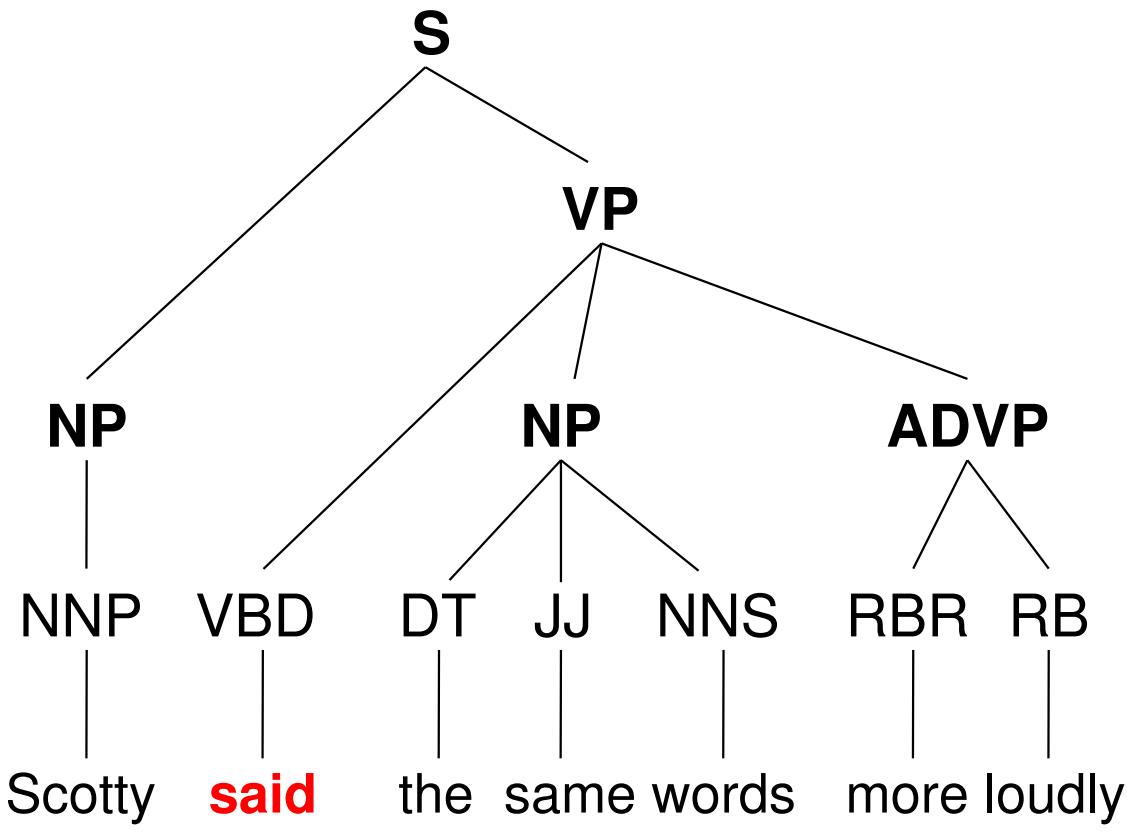
Step 2: Local scoring of candidates

- 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
  - $\Rightarrow$ Computational reasons but also modularity in feature engineering
- Many ML paradigms have been used: not big differences
- Features are more important

SRL Architecture: Steps 1 + 2

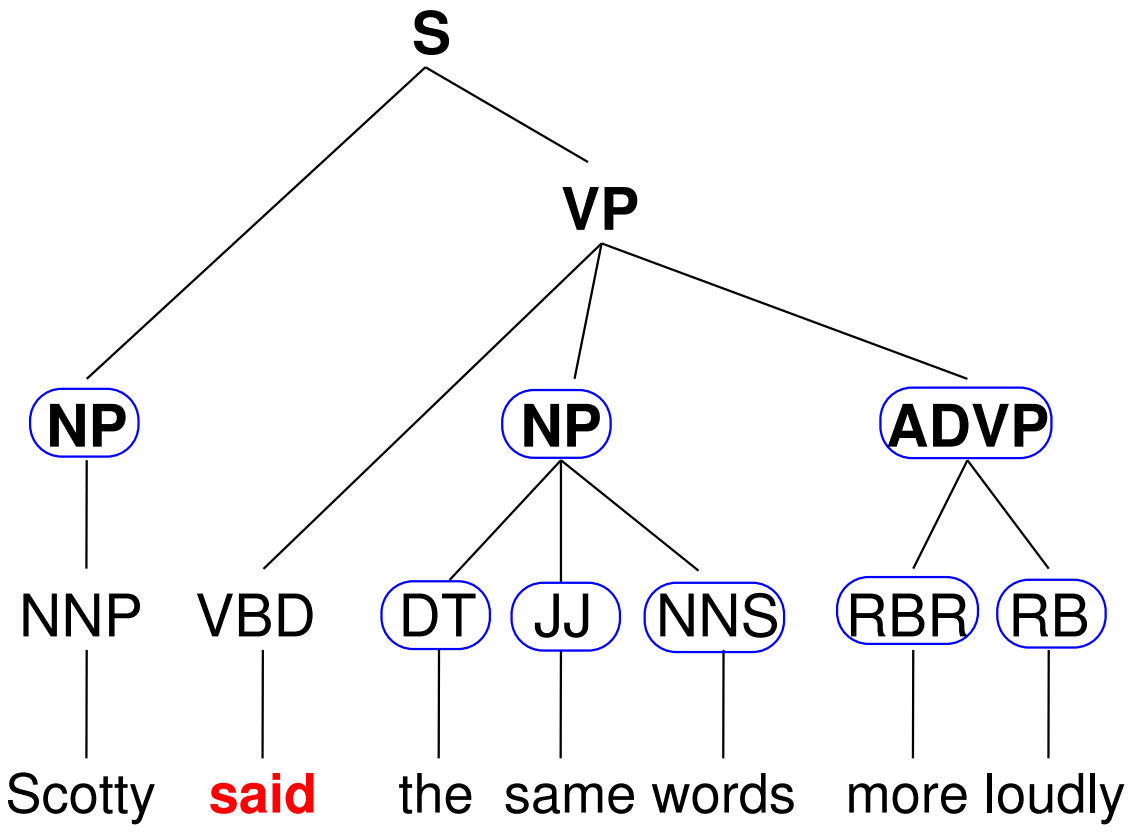
Scotty **said** the same words more loudly

SRL Architecture: Steps 1 + 2



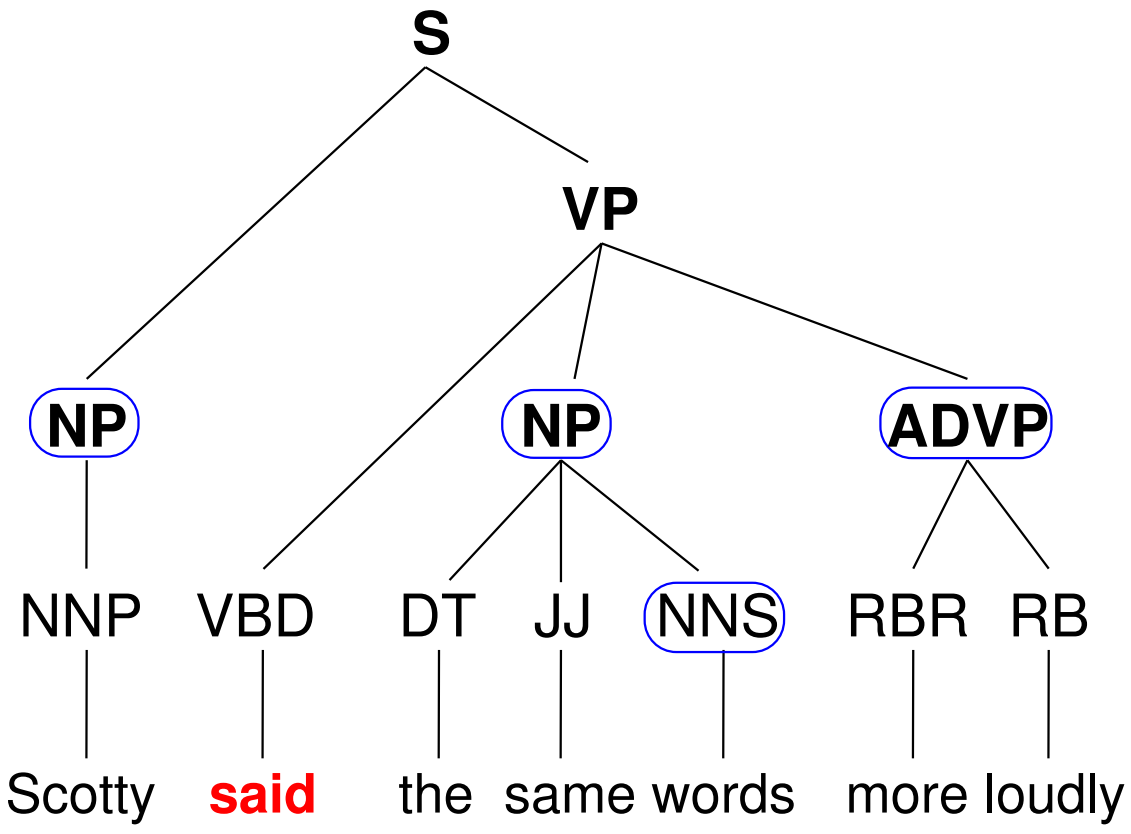
amazon

SRL Architecture: Steps 1 + 2



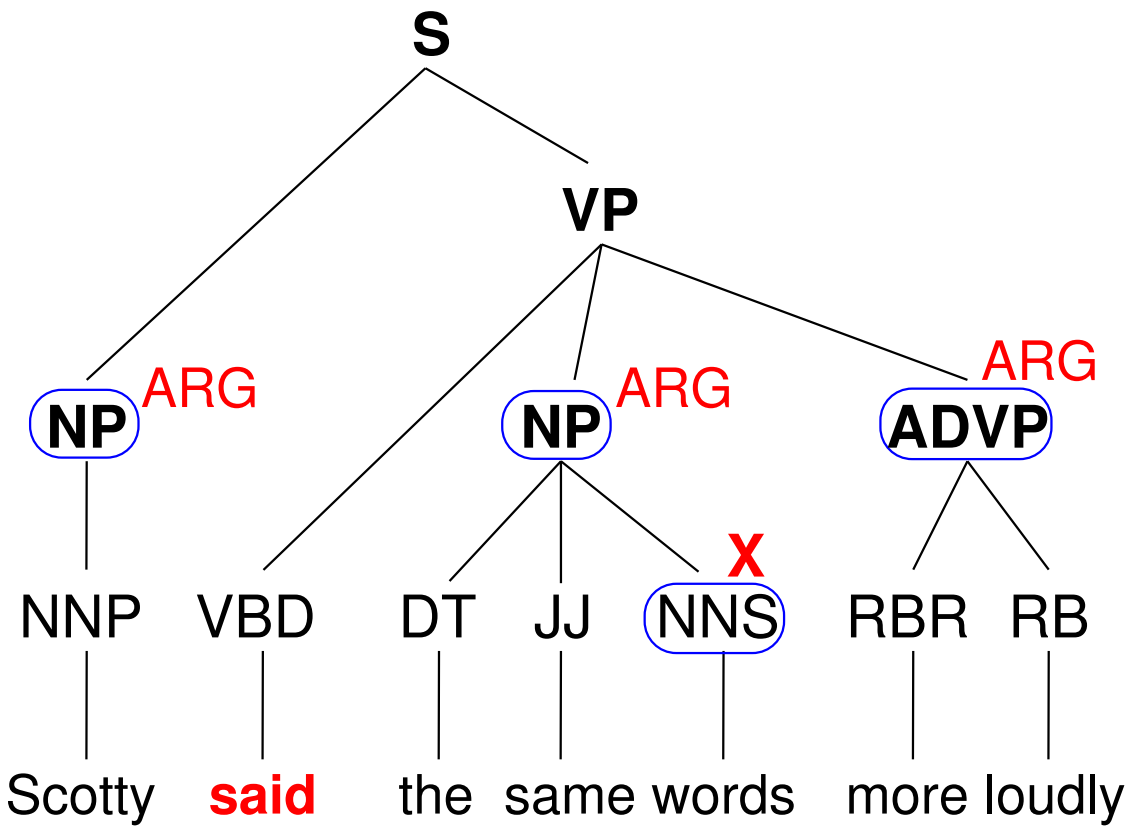
amazon

SRL Architecture: Steps 1 + 2



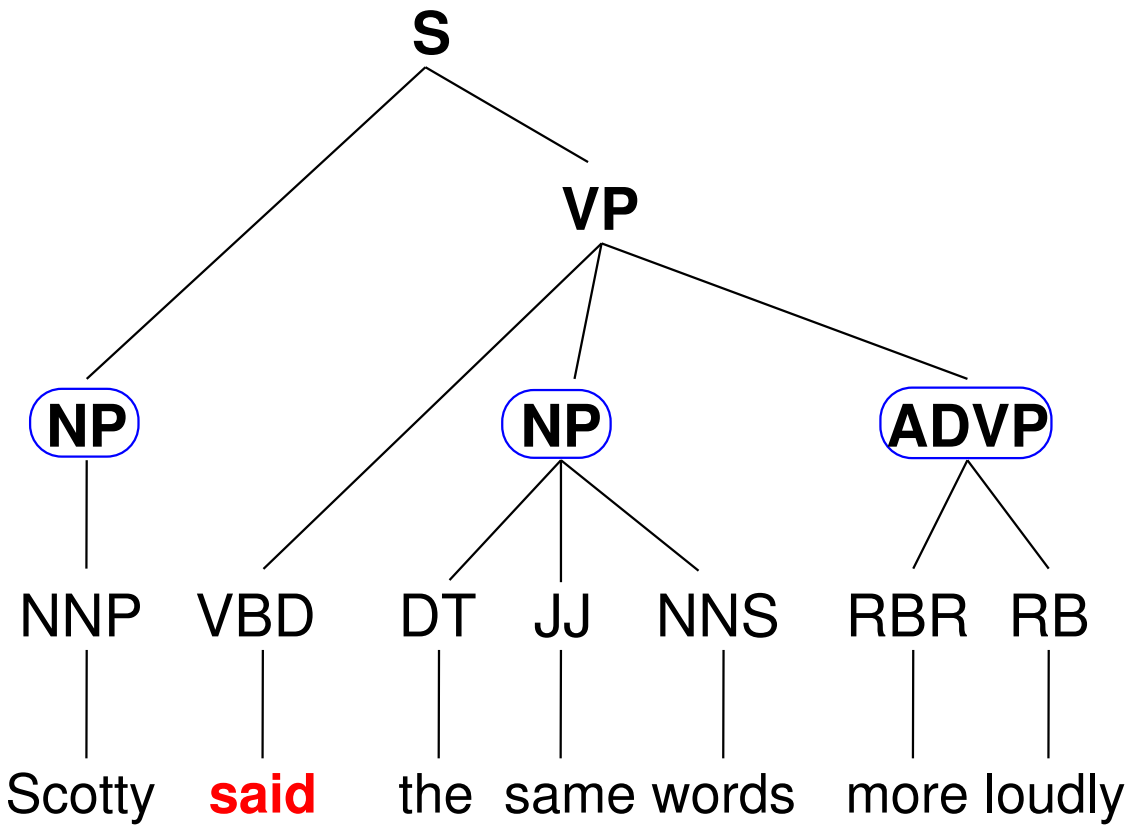
amazon

SRL Architecture: Steps 1 + 2



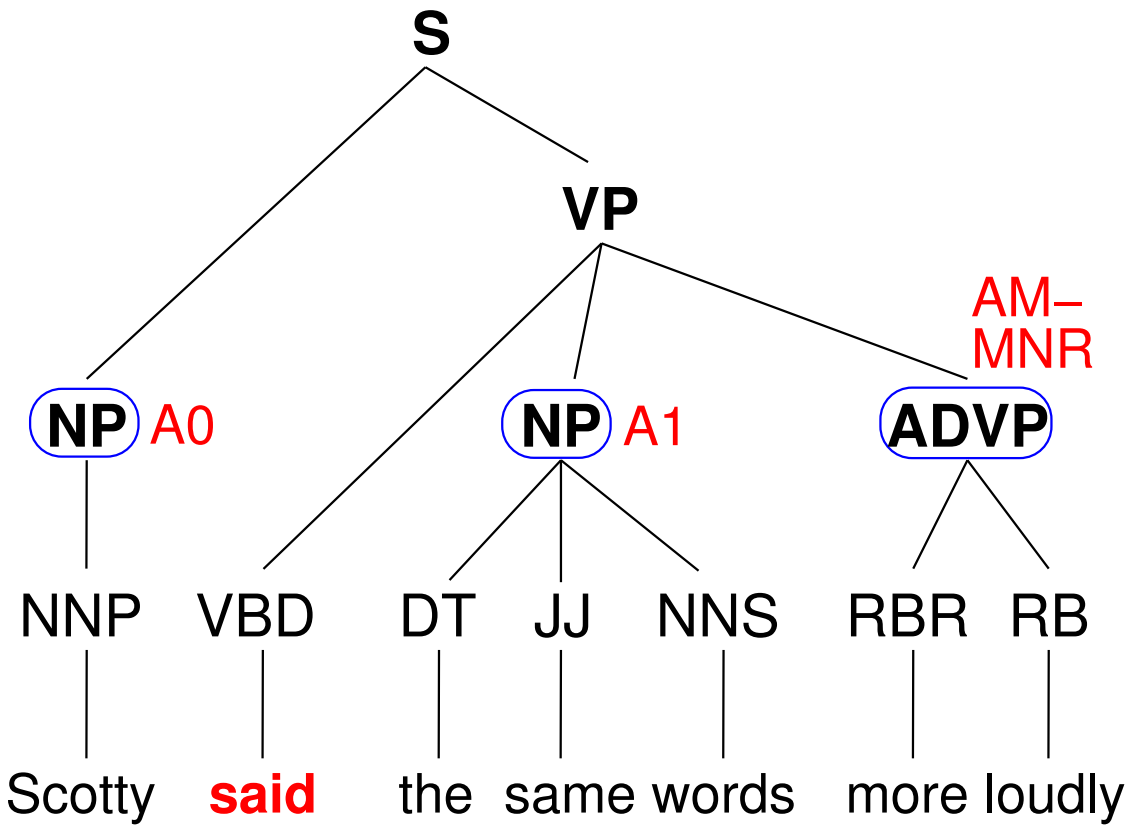
amazon

SRL Architecture: Steps 1 + 2



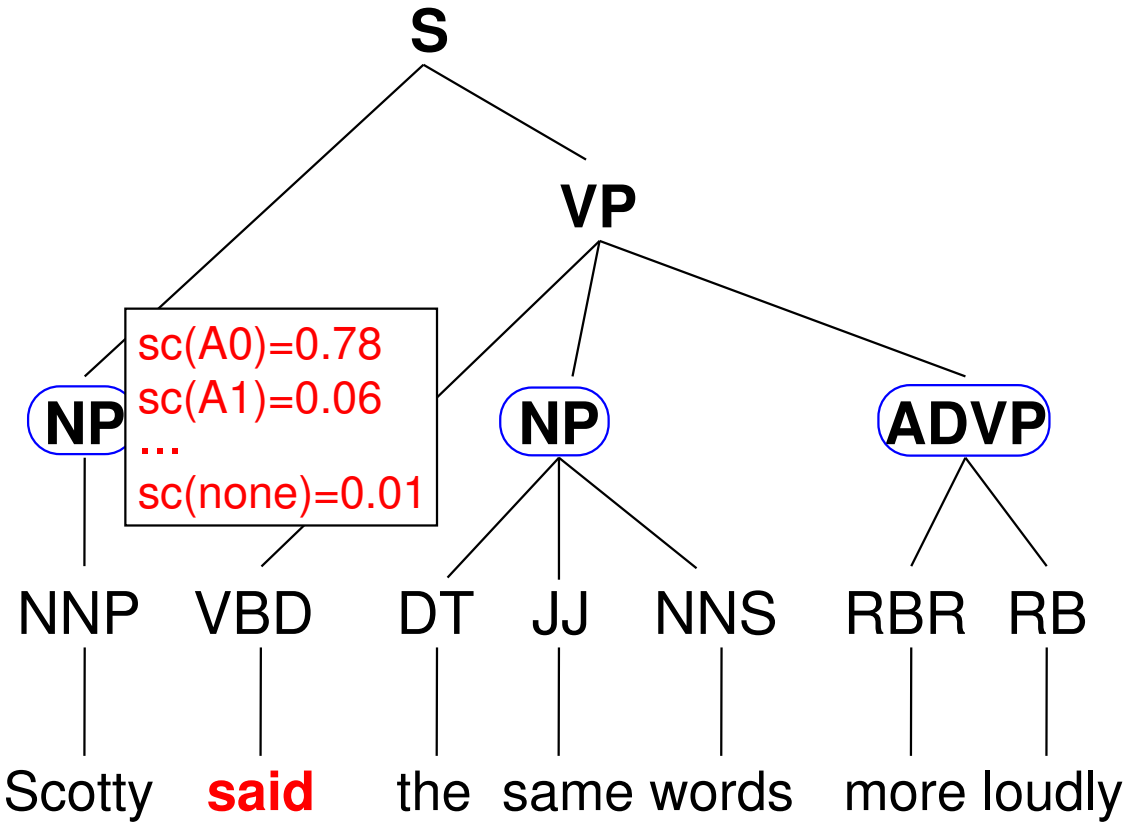
amazon

SRL Architecture: Steps 1 + 2



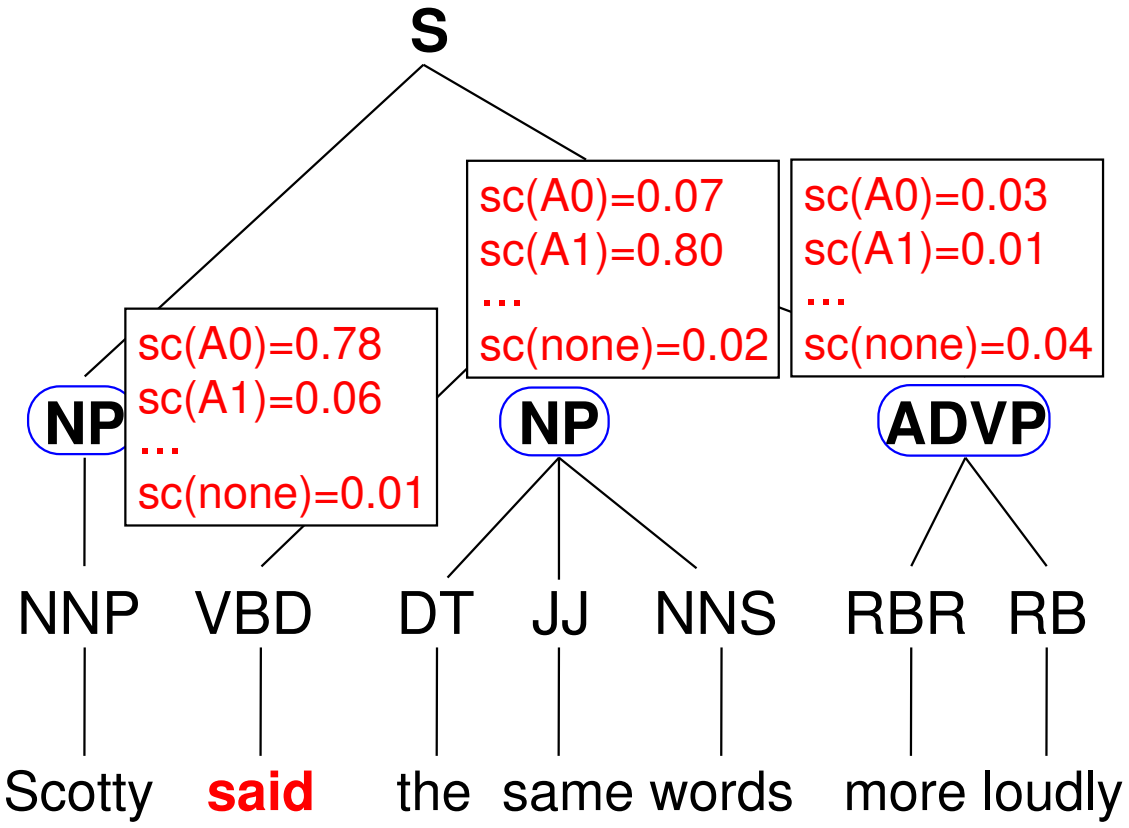
amazon

SRL Architecture: Motivating next step (joint scoring)



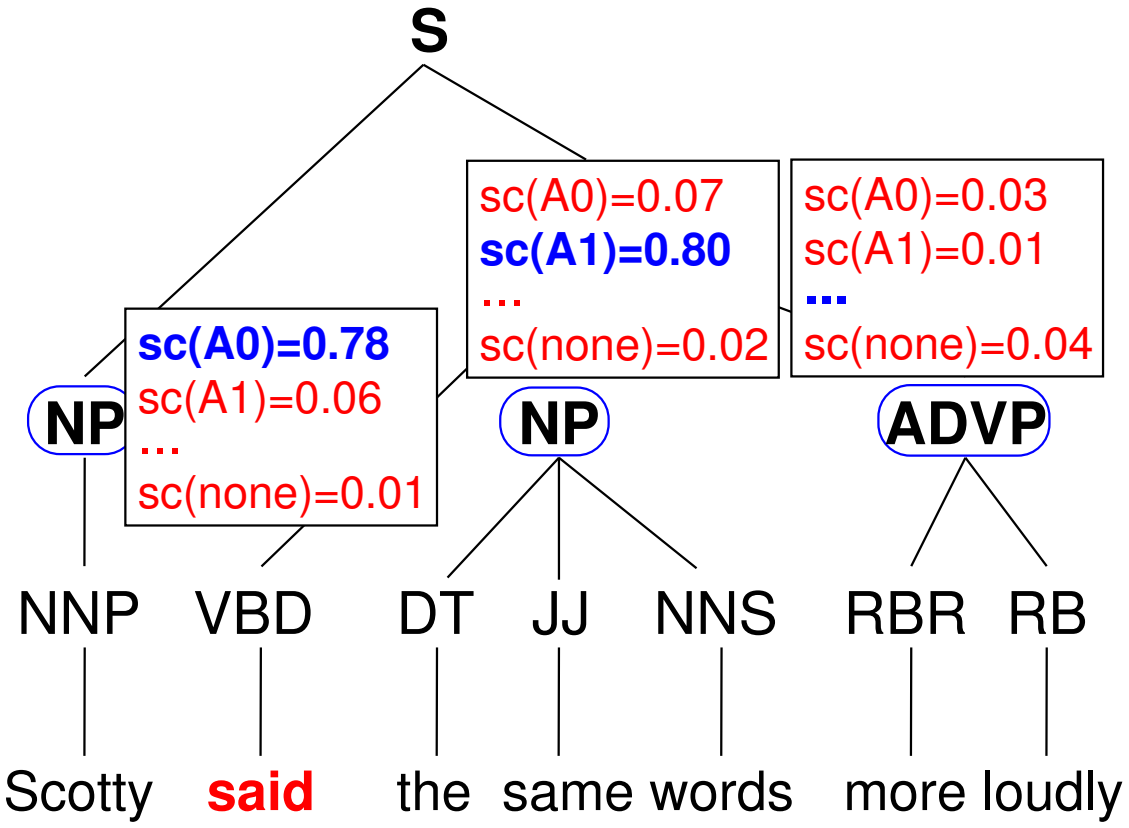
amazon

SRL Architecture: Motivating next step (joint scoring)



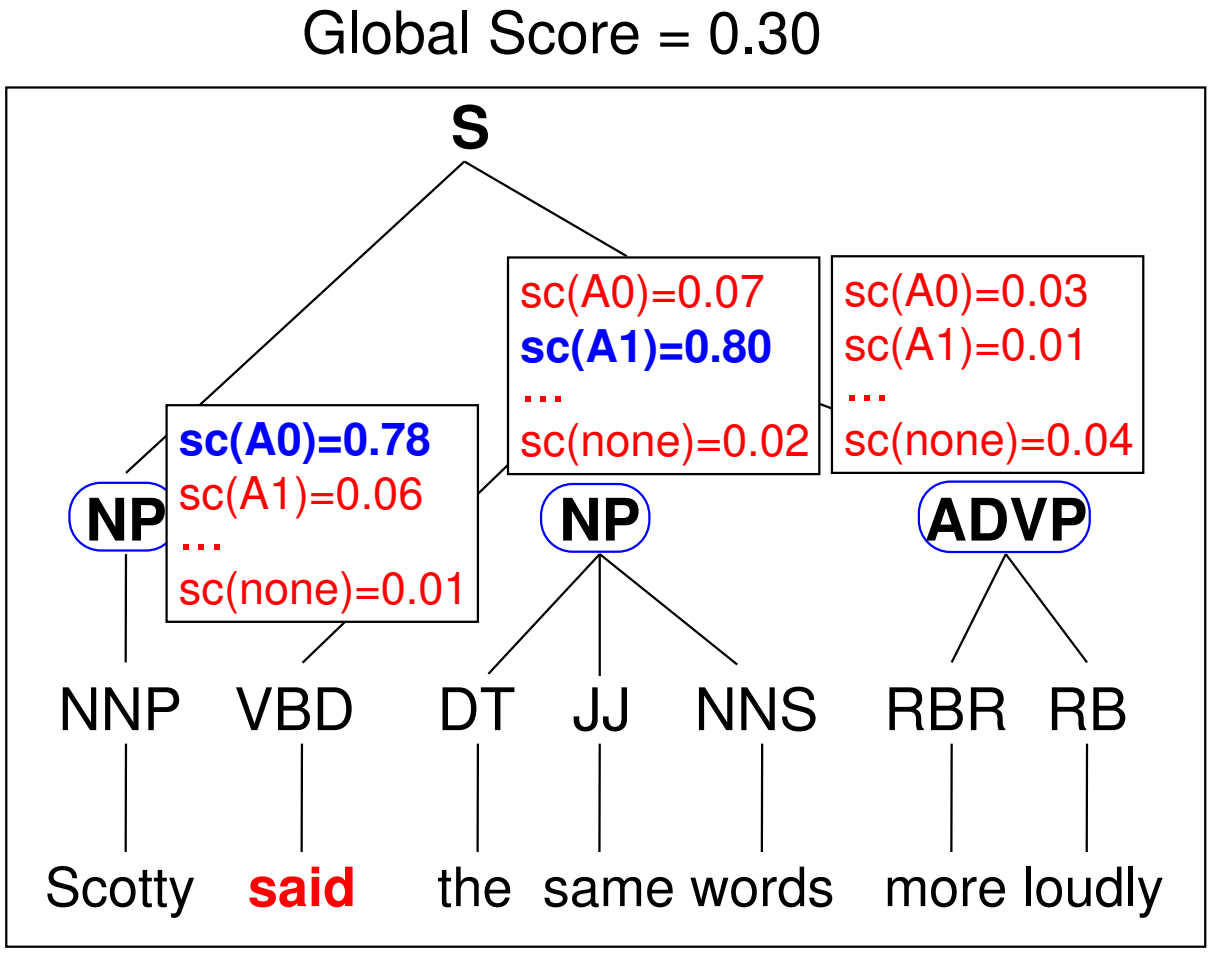
amazon

SRL Architecture: Motivating next step (joint scoring)

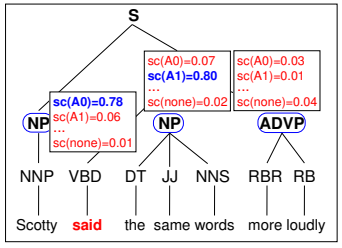


amazon

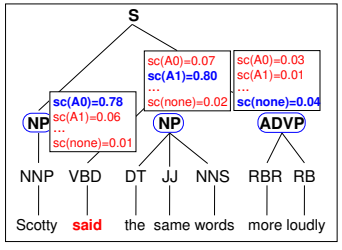
SRL Architecture: Motivating next step (joint scoring)



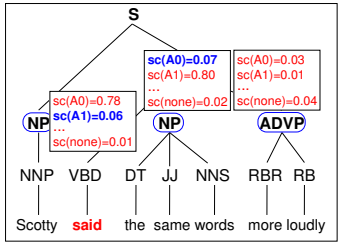
SRL Architecture: Motivating next step (joint scoring)



GS = 0.45

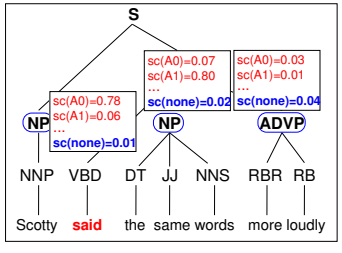


GS = 0.15



GS = 0.09

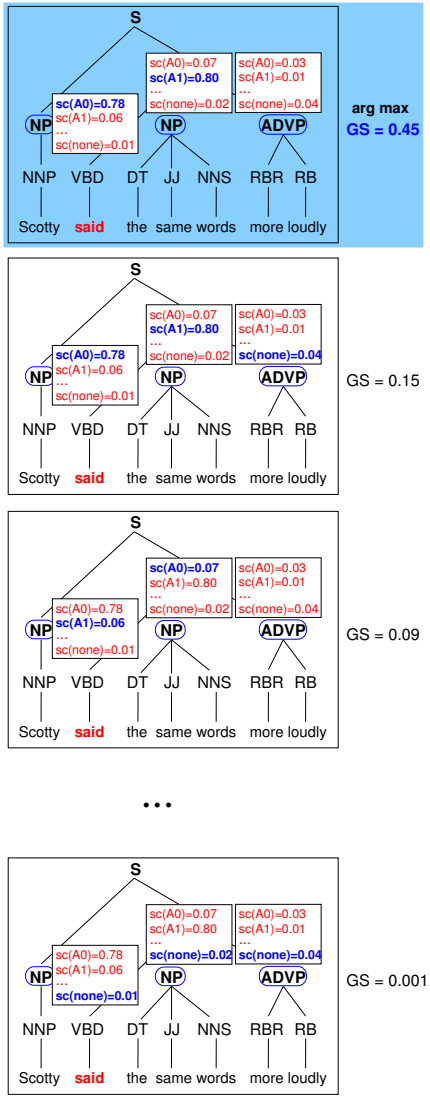
...



GS = 0.001

amazon

SRL Architecture: Motivating next step (joint scoring)



SRL Architecture: 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 +features +search
- Re-ranking of several candidate solutions
  (Haghighi et al., 2005; Toutanova et al., 2008)
  +learning +features –search
- Global search integrating joint scoring: Tree CRFs
  (Cohn & Blunsom, 2005)
  +learning +/–features +/–search

SRL Architecture: 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 +features +search
- Re-ranking of several candidate solutions
  (Haghighi et al., 2005; Toutanova et al., 2008)
  +learning +features –search
- Global search integrating joint scoring: Tree CRFs
  (Cohn & Blunsom, 2005)
  +learning +/–features +/–search

SRL Architecture: Step by Step

Step 4: Post-processing

- Application of a set of heuristic rules to:

  Correct frequent errors

  Enforce consistency in the solution

Detour to Machine Learning Concepts

What do we need from ML so far?

- Estimate functions to predict the local scores
  - Supervised machine learning for classification
  - Decision Trees, AdaBoost, MaxEnt, Perceptron, SVMs, NNs
- Mechanisms to implement a joint inference process (later...)

Talk Overview

amazon

Examples of SRL systems

- Generalized inference with local classifers and constraints ILP approach (Punyakanok et al., 2008)
- Joint System based on Reranking (Toutanova et al., 2008)
- SRL as sequential labeling (Màrquez et al., 2005)

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

Architecture

1. Identify argument candidates
   - ⇒ Pruning (Xue & Palmer, 2004)
   - ⇒ Argument identification: binary classification (using SNoW)
2. Classify argument candidates
   - ⇒ Argument Classifier: multi-class classification (SNoW)
3. Inference
   - ⇒ Use the estimated probability distribution given by the argument classifier
   - ⇒ Use structural and linguistic constraints
   - ⇒ Infer the optimal global output

amazon

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

Architecture

1. Identify argument candidates
   - ⇒ Pruning (Xue & Palmer, 2004)
   - ⇒ Argument identification: binary classification (using SNoW)
2. Classify argument candidates
   - ⇒ Argument Classifier: multi-class classification (SNoW)
3. Inference
   - ⇒ Use the estimated probability distribution given by the argument classifier
   - ⇒ Use structural and linguistic constraints
   - ⇒ Infer the optimal global output

amazon

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

Architecture

1. Identify argument candidates
   - ⇒ Pruning (Xue & Palmer, 2004)
   - ⇒ Argument identification: binary classification (using SNoW)
2. Classify argument candidates
   - ⇒ Argument Classifier: multi-class classification (SNoW)
3. Inference
   - ⇒ Use the estimated probability distribution given by the argument classifier
   - ⇒ Use structural and linguistic constraints
   - ⇒ Infer the optimal global output

amazon

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)

amazon

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)

amazon

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

amazon

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

Constraints: examples

- No duplicate argument classes: $\sum_{i=1}^{n} a_{i,Arg0} \leqslant 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i=1}^{j-1} a_{i,Arg0} \geqslant a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i \neq j} a_{i,Arg0} \geqslant a_{j,R-Arg0}$
- Many other possible constraints:
  - $\Rightarrow$ Unique labels
  - $\Rightarrow$ No overlapping or embedding
  - $\Rightarrow$ Relations between number of arguments; order constraints
  - $\Rightarrow$ 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: $\sum_{i=1}^{n} a_{i,Arg0} \leqslant 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i=1}^{j-1} a_{i,Arg0} \geqslant a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i \neq j} a_{i,Arg0} \geqslant a_{j,R-Arg0}$
- Many other possible constraints:
  - $\Rightarrow$ Unique labels
  - $\Rightarrow$ No overlapping or embedding
  - $\Rightarrow$ Relations between number of arguments; order constraints
  - $\Rightarrow$ 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: $\sum_{i=1}^{n} a_{i,Arg0} \leqslant 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i=1}^{j-1} a_{i,Arg0} \geqslant a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i \neq j} a_{i,Arg0} \geqslant a_{j,R-Arg0}$
- Many other possible constraints:
  - $\Rightarrow$ Unique labels
  - $\Rightarrow$ No overlapping or embedding
  - $\Rightarrow$ Relations between number of arguments; order constraints
  - $\Rightarrow$ 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: $\sum_{i=1}^{n} a_{i,Arg0} \leqslant 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leqslant j \leqslant n),\ \sum_{i=1}^{j-1} a_{i,Arg0} \geqslant a_{j,C-Arg0}$
- On reference arguments (R-ARG)

  [The deregulation]$_{Arg1}$ of railroads and trucking companies [that]$_{R-Arg1}$ began [in 1980]$_{AM-TMP}$ enabled ...
  $\forall j (1 \leqslant j \leqslant n),\ \sum_{i \neq j} a_{i,Arg0} \geqslant a_{j,R-Arg0}$
- 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: $\sum_{i=1}^{n} a_{i,Arg0} \leqslant 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i=1}^{j-1} a_{i,Arg0} \geqslant a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i \neq j} a_{i,Arg0} \geqslant a_{j,R-Arg0}$
- Many other possible constraints:
  - $\Rightarrow$ Unique labels
  - $\Rightarrow$ No overlapping or embedding
  - $\Rightarrow$ Relations between number of arguments; order constraints
  - $\Rightarrow$ 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

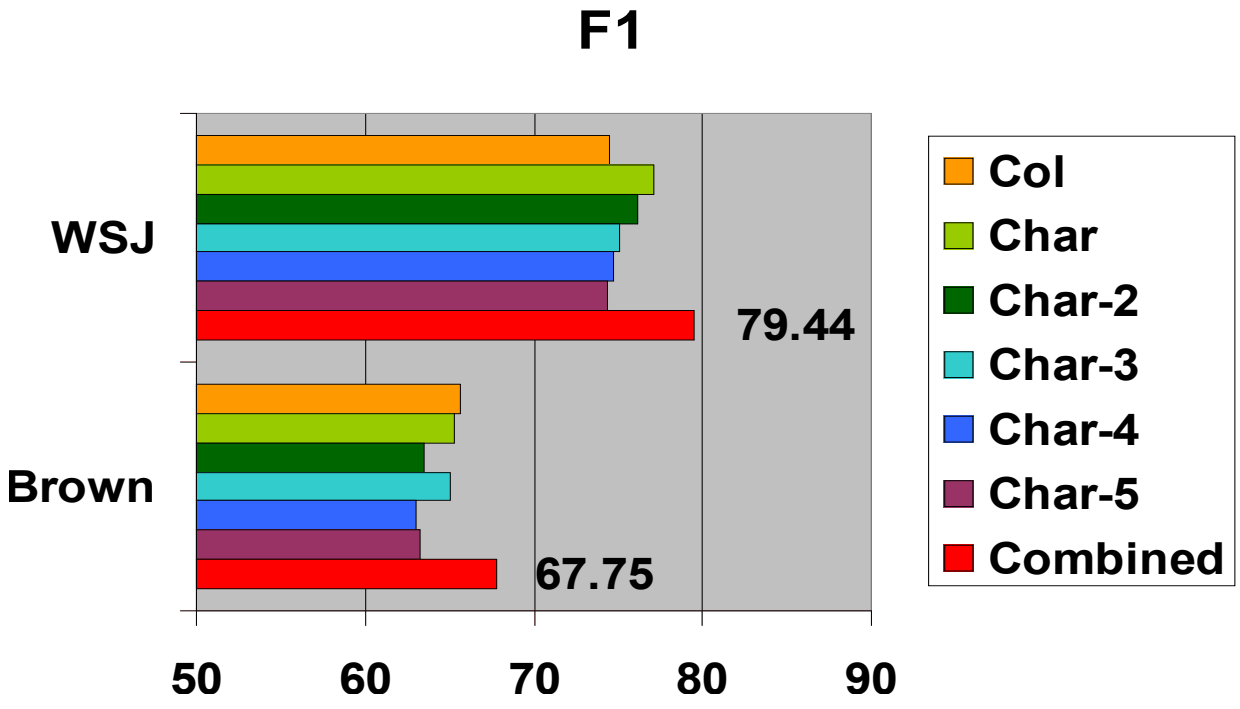
- No duplicate argument classes: $\sum_{i=1}^{n} a_{i,Arg0} \leqslant 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i=1}^{j-1} a_{i,Arg0} \geqslant a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i \neq j} a_{i,Arg0} \geqslant a_{j,R-Arg0}$
- Many other possible constraints:
  - $\Rightarrow$ Unique labels
  - $\Rightarrow$ No overlapping or embedding
  - $\Rightarrow$ Relations between number of arguments; order constraints
  - $\Rightarrow$ 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: $\sum_{i=1}^{n} a_{i,Arg0} \leqslant 1$
- On discontinuous arguments (C-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i=1}^{j-1} a_{i,Arg0} \geqslant a_{j,C-Arg0}$
- On reference arguments (R-ARG)
  $\forall j (1 \leqslant j \leqslant n), \sum_{i \neq j} a_{i,Arg0} \geqslant a_{j,R-Arg0}$
- Many other possible constraints:
  - $\Rightarrow$ Unique labels
  - $\Rightarrow$ No overlapping or embedding
  - $\Rightarrow$ Relations between number of arguments; order constraints
  - $\Rightarrow$ 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)



- Joint inference improves results > 2.0 F_1 points
- Inference with many parsers improves results ~ 2.6 F_1 points
- Best results at CoNLL-2005 shared task (Carreras & Màrquez, 2005)

amazon

Detour to Machine Learning Concepts (II)

What have we used from ML now?

- Inference with local classifiers under structural and problem-dependent constraints (CSP)
- Integer Linear Programming formulation
  - $\Rightarrow$ Efficient ILP (exact) solvers exist
  - $\Rightarrow$ Example: Joint learning of named entities and relations

amazon

Joint System based on Reranking (Toutanova et al., 2008)

Architecture

- Use a probabilistic local SRL model to produce multiple (n-best) candidate solutions for the predicate structure
- Use a feature–rich reranking model to select the best solution among them

Main goal: is to build a rich model for joint scoring, which takes into account the dependencies among the labels of argument phrases

amazon

Joint System based on Reranking (Toutanova et al., 2008)

Local Steps

i. Parse the sentence and apply pruning (Xue & Palmer, 2004) to filter argument candidates for a given predicate p

ii. Apply a simple local scoring model trained with log-linear classifiers (MaxEnt): $P(label_i|node, p)$ probability distribution

iii. Consider a simple global scoring scheme assuming independence of local assignments:
$P_{LOCAL}(L|tree, p) = \prod_{node_i \in tree} P(label_i|node_i, p)$

iv. Use dynamic programming to find the n–most probable non-overlapping complete labelings for predicate p

Joint System based on Reranking (Toutanova et al., 2008)

Reranking Step

i. Consider a reranking model trained to select the best among the n–most probable complete labelings; again a log-linear model: $P_{JOINT}(L_i|tree, p)$

ii. Consider the following combination of local and joint scoring models: $log(P_{SRL}(L|tree, p)) =$
$$log(P_{JOINT}(L|tree, p)) + \lambda log(P_{LOCAL}(L|tree, p))$$

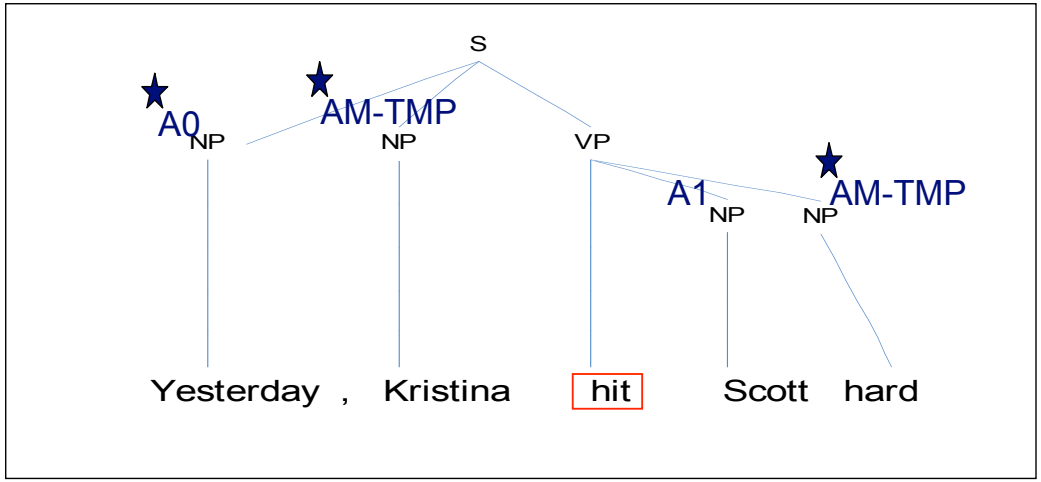
iii. Select the complete labeling ($L_i \in \{L_1, L_2, \ldots, L_n\}$) that maximizes the previous formula (reranking)

amazon

Joint System based on Reranking (Toutanova et al., 2008)

Features: joint scoring slide from (Yih & Toutanova, 2006)

Joint Model Features



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)

66

Joint System based on Reranking (Toutanova et al., 2008)

Enhancement by using multiple trees

- For top k trees from Charniak's parser, $t_1, t_2, \ldots, t_k$, find corresponding best SRL assignments $L_1, L_2, \ldots, L_k$ 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 (2nd best at CoNLL):
  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: 80.32 F_1 (WSJ) 68.81 F_1 (Brown)
- Improvement due to the joint model: >2 F_1 points

Joint System based on Reranking (Toutanova et al., 2008)

Enhancement by using multiple trees

- For top k trees from Charniak's parser, $t_1, t_2, \ldots, t_k$, find corresponding best SRL assignments $L_1, L_2, \ldots, L_k$ 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 (2nd best at CoNLL):
  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: 80.32 F_1 (WSJ) 68.81 F_1 (Brown)
- Improvement due to the joint model: >2 F_1 points

Joint System based on Reranking (Toutanova et al., 2008)

Enhancement by using multiple trees

- For top k trees from Charniak's parser, $t_1, t_2, \ldots, t_k$, find corresponding best SRL assignments $L_1, L_2, \ldots, L_k$ 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 (2nd best at CoNLL):
  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: 80.32 F_1 (WSJ) 68.81 F_1 (Brown)
- Improvement due to the joint model: >2 F_1 points

Detour to Machine Learning Concepts (III)

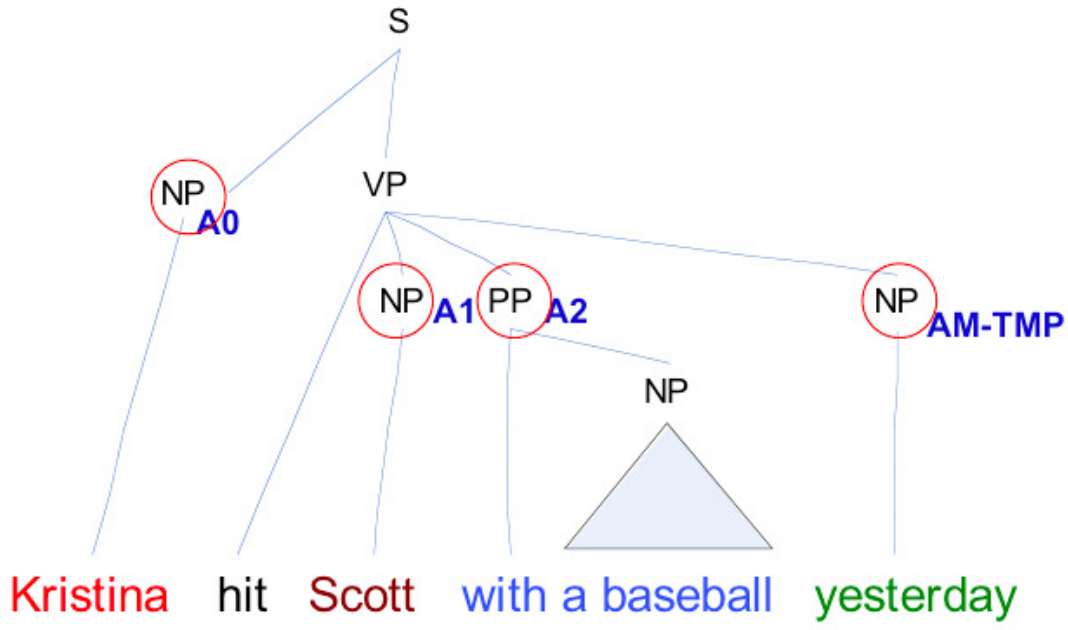
What else do we need from ML?

- Ranking and re-ranking algorithms (*learning to rank*)
  - $\Rightarrow$ A simple example: Ranking Perceptron

SRL as sequential tagging (Màrquez et al., 2005)

- 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



SRL as sequential tagging (Màrquez et al., 2005)

- 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 (2416 sentences)
    - 77.97 (F_1), 79.55 (Prec.), 76.45 (Rec.)
  - Brown (426 sentences; cross-domain test)
    - 67.42 (F_1), 70.79 (Prec.), 64.35 (Rec.)

Detour to Machine Learning Concepts (IV)

More things to learn from Machine Learning?

- Sequential tagging/segmentation paradigm
  - $\Rightarrow$ HMMs (generative models)
  - $\Rightarrow$ Chained local classifiers, MEMMs, CRFs, structure perceptron

SRL Architecture

Exceptions to the standard architecture

- Parsing variations for SRL
  - ⇒ Syntactic parser trained to predict argument candidates (Yi & Palmer, 2005)
  - ⇒ Joint parsing and SRL: semantic parsing (Musillo & Merlo, 2006; Merlo & Musillo, 2008)
  - ⇒ SRL based on dependency parsing (Johansson & Nugues, 2007)
  - ⇒ Systems from the CoNLL–2008 and 2009 shared tasks (Surdeanu et al., 2008; Hajič et al., 2009)
  - ⇒ CCG parser (Gildea and Hockenmaier, 2005; Boxwell et al., 2009)
  - ⇒ HPSG parsers with handcrafted grammars (Zhang et al., 2008; 2009)

SRL Architecture

Exceptions to the standard architecture (II)

- SRL as sequential tagging
  (Hacioglu et al., 2004; Màrquez et al., 2005; Surdeanu et al., 2007)
- Joint treatment of all predicates in the sentence
  (Carreras et al., 2004; Surdeanu et al., 2008)
- SRL using Markov Logic Networks
  (Meza-Ruiz & Riedel, 2008; 2009)

Talk Overview

amazon

Feature Engineering

Features: local scoring (Gildea & Jurafsky, 2002)

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

Feature Engineering

Features: local scoring — extensions

- "Brute force" features. Applied to the constituent and possibly to parent and siblings:
  - $\Rightarrow$ 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
  - $\Rightarrow$ 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

amazon

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

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

Feature Engineering

Features: joint scoring

- Richer features taking into account information from several arguments at a time
- Best example: when doing re-ranking one may codify patterns on the whole candidate argument structure
  (Hiaghighi et al., 2005; Toutanova et al., 2008)
- Good for capturing global preferences

amazon

(avoiding) Feature Engineering

The Kernel approach

- Knowledge poor approach
- Let the kernel function to compute the similarity/differences between examples by considering all possible substructures as features
- Motivation: avoid intense knowledge engineering
- Potentially useful for rapid system development and working with under resourced languages
- Mostly variants of Collins' all-subtrees convolution kernel (Moschitti et al., 2008; Pighin & Moschitti, 2009; 2010)

(avoiding) Feature Engineering

The Kernel approach

- Knowledge poor approach
- Let the kernel function to compute the similarity/differences between examples by considering all possible substructures as features
- Motivation: avoid intense knowledge engineering
- Potentially useful for rapid system development and working with under resourced languages
- Mostly variants of Collins' all-subtrees convolution kernel (Moschitti et al., 2008; Pighin & Moschitti, 2009; 2010)

amazon

(avoiding) Feature Engineering

The Kernel approach

- Knowledge poor approach
- Let the kernel function to compute the similarity/differences between examples by considering all possible substructures as features
- Motivation: avoid intense knowledge engineering
- Potentially useful for rapid system development and working with under resourced languages
- Mostly variants of Collins' all-subtrees convolution kernel (Moschitti et al., 2008; Pighin & Moschitti, 2009; 2010)

(avoiding) Feature Engineering

Features: the Kernel approach

Problems with the structural kernel approach

1. Uncontrolled explosion of features
2. Low efficiency
3. Difficulty of using linguistic knowledge

Some works in the previous directions

- Semantic Role Labeling Using a Grammar-Driven Convolution Tree Kernel. Includes approximate matching at substructure and node levels (Zhang et al., 2008)
- Feature selection in kernel space and lineariziation of Tree Kernel functions (Pighin & Moschitti, 2009)

(avoiding) Feature Engineering

Features: the Kernel approach

Problems with the structural kernel approach

1. Uncontrolled explosion of features
2. Low efficiency
3. Difficulty of using linguistic knowledge

Some works in the previous directions

- Semantic Role Labeling Using a Grammar-Driven Convolution Tree Kernel. Includes approximate matching at substructure and node levels (Zhang et al., 2008)
- Feature selection in kernel space and lineariziation of Tree Kernel functions (Pighin & Moschitti, 2009)

Other Approaches to Reduce Feature Engineering

- *Low-rank decomposition of high-order tensor models* (Lei et al., 2015; NAACL)
  Joint work with: *Tao Lei, Yuan Zhang, Alessandro Moschitti and Regina Barzilay*

- Summary:
  - ⇒ Automatically induce a compact feature representation for words and their relations, tailoring them to the task.
  - ⇒ Capture meaningful interactions between the argument, predicate, their syntactic path and the corresponding role label.
  - ⇒ Overall cross-product feature representation as a four-way low-rank tensor
  - ⇒ This approach provides a clear alternative to the traditional feature engineering.

amazon

Semantic Features for SRL

Selectional Preferences for Semantic Role Classification

Joint work with

Eneko Agirre, *Mihai Surdeanu* and *Beñat Zapirain*

(Zapirain et al. 2010) — ACL

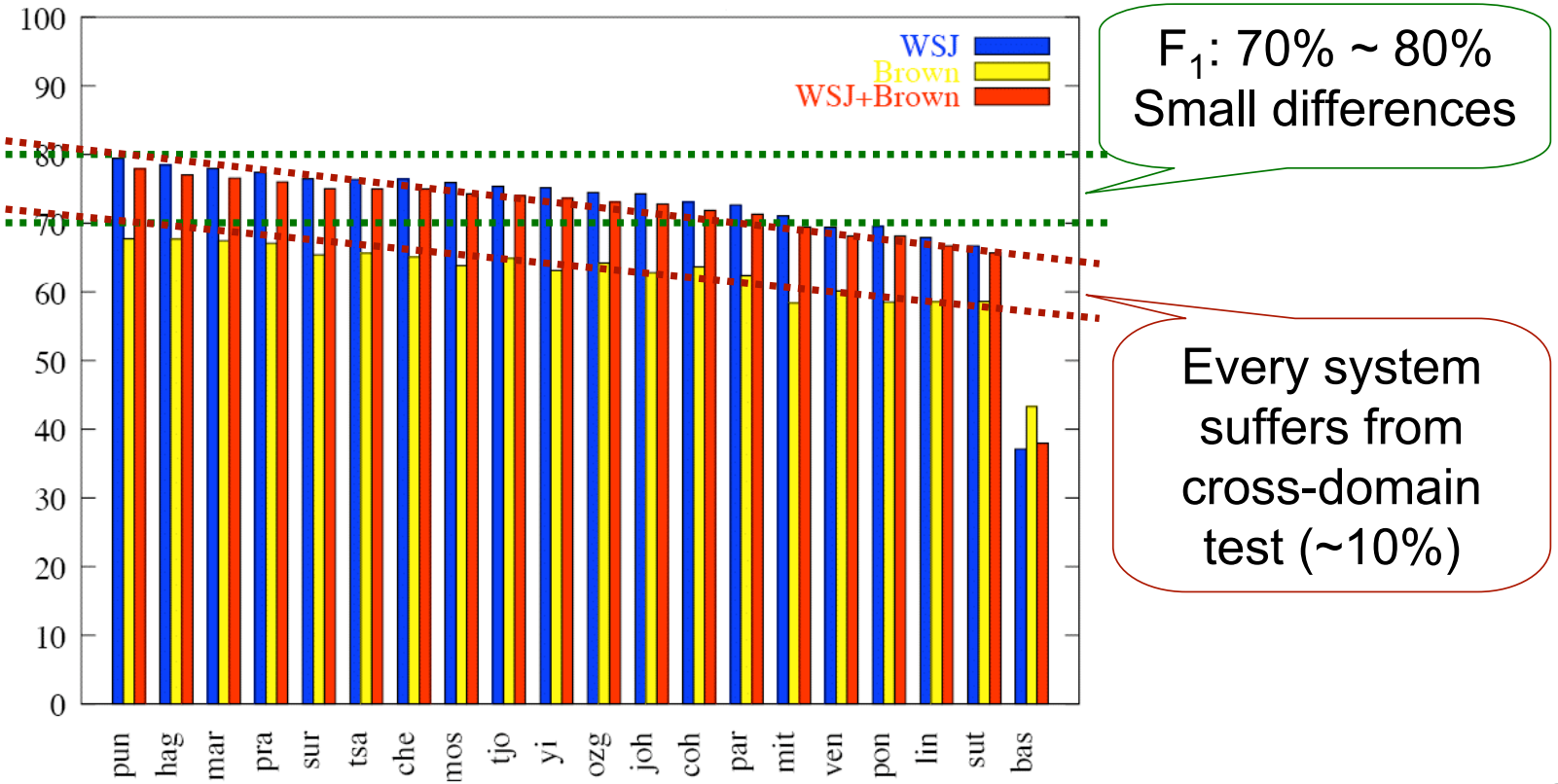
(Zapirain et al. 2011) — NAACL

(Zapirain et al. 2013) — Computational Linguistics 39(3)

amazon

Results from CoNLL-2005 shared task

Results on WSJ and Brown Tests



Results from CoNLL-2005 shared task

Reasons for the low generalization ability

- The training corpus is not representative and large enough (and it will never be)
- Taggers and syntactic parsers also experience a significant drop in performance
- The main loss in performance takes place in role classification, not identification — semantic explanation (Pradhan et al., 2008)

amazon

Semantic Features for SRL

Motivation

- Most current systems capture semantics through lexicalized features on the predicate and the head word of the argument to be classified
- But lexical features are sparse and generalize badly

  [JFK]$_{Patient}$ *was_assassinated* [in Dallas]$_{LOC}$

  [JFK]$_{Patient}$ *was_assassinated* [in November]$_{TMP}$
- [in Texas]$_{???}$, [in autumn]$_{???}$

Semantic Features for SRL

Motivation

- Most current systems capture semantics through lexicalized features on the predicate and the head word of the argument to be classified
- But lexical features are sparse and generalize badly

  [JFK]$_{Patient}$ *was_assassinated* [in Dallas]$_{LOC}$

  [JFK]$_{Patient}$ *was_assassinated* [in November]$_{TMP}$

- [in Texas]$_{???}$, [in autumn]$_{???}$

Semantic Features for SRL

Motivation

- Most current systems capture semantics through lexicalized features on the predicate and the head word of the argument to be classified
- But lexical features are sparse and generalize badly

  [JFK]$_{Patient}$ *was_assassinated* [in Dallas]$_{LOC}$

  [JFK]$_{Patient}$ *was_assassinated* [in November]$_{TMP}$
- [in Texas]$_{???}$, [in autumn]$_{???}$

Semantic Features for SRL

Motivation

Selectional Preferences and distributional similarity techniques should help us to classify arguments with low–frequency or unknown head words

$$[\text{Dallas} \approx \text{Texas}]_{\textit{Location}},\ [\text{November} \approx \text{autumn}]_{\textit{Temporal}}$$

Previous Work

Selectional Preferences

- Modeling semantic preferences that predicates impose on their arguments
- Long tradition of automatic acquisition of selectional preferences (SPs) from corpora. WordNet–based and distributional models of SPs
  (Resnik, 1993; Pantel and Lin, 2000; Brockmann and Lapata, 2003)
  (Erk 2007; Erk et al., 2011; etc.)

  $\Rightarrow$ e.g., estimate plausibility of triples:
  (verb, argument, head-word)

  $\Rightarrow$ useful for syntactic-semantic disambiguation

Previous Work

SPs applied to Semantic Role Labeling

- (Gildea and Jurafsky, 2002) – FrameNet
  - ⇒ First researchers to apply selectional preferences to SRL
  - ⇒ Distributional clustering and WordNet-based techniques to generalize argument heads
  - ⇒ Slight improvement in role classification (NP arguments)
- Zapirain et al. (2010; 2013) – PropBank
  - ⇒ Show that selectional preferences can improve semantic role classification in a state-of-the-art SRL system

amazon

Selectional Preferences for SRL (Zapirain et al., 2013)

Two types of selectional preferences (SP)

i. verb–*role*: list of heads of NP arguments of the predicate verb that are labeled with the role *role*

```
write-Arg0:  Angrist anyone baker ball bank Barlow Bates ...
write-Arg1:  abstract act analysis article asset bill book ...
write-Arg2:  bank commander hundred jaguar Kemp member ...
write-AM-LOC: paper space ...

...
```

ii. prep–*role*: list of nominal heads of PP arguments with preposition prep that are labeled with the role *role*

```
from-Arg2:  academy account acquisition activity ad ...
from-Arg3:  activity advertising agenda airport ...
from-Arg4:  europe Golenbock system Vizcaya west
from-AM-TMP: april august beginning bell day dec.  half ...
from-AM-LOC: agency area asia body bureau orlando ...

...
```

Selectional Preferences for SRL (Zapirain et al., 2013)

SP models: $SP_{sim}(p, r, w)$ compatibility score

- Discriminative approach: given a new argument of a predicate p, we compare its head (w) to the selectional preference of each possible role label r, i.e., we want to find the role with the selectional preference that fits the head best
- We compute the compatibility scores using two different methods
  - $\Rightarrow$ WordNet based —using (Resnik, 1993)
  - $\Rightarrow$ Based on distributional similarity —a la Erk (2007)

amazon

Selectional Preferences for SRL (Zapirain et al., 2013)

SP models: $SP_{sim}(p, r, w)$ compatibility score

- Discriminative approach: given a new argument of a predicate p, we compare its head (w) to the selectional preference of each possible role label r, i.e., we want to find the role with the selectional preference that fits the head best
- We compute the compatibility scores using two different methods
  - $\Rightarrow$ WordNet based —using (Resnik, 1993)
  - $\Rightarrow$ Based on distributional similarity —a la Erk (2007)

Selectional Preferences for SRL (Zapirain et al., 2013)

WordNet SP models

- Resnik formula (1993) is used to precalculate a weighted list of relevant synsets for the lists of words contained in the SPs

SP write–*Arg0*: Angrist anyone baker ball bank Barlow Bates ...

n#00002086 5.875 life form organism being living thing "any living entity"
n#00001740 5.737 entity something "anything having existence (living or nonliving)"
n#00009457 4.782 object physical object "a physical (tangible and visible) entity;"
n#00004123 4.351 person individual someone somebody mortal human soul "a human being;"
...

SP write–*Arg1*: abstract act analysis article asset bill book ...

n#00019671 7.956 communication "something that is communicated between people or groups"
n#04949838 4.257 message content subject matter substance "what a communication that ..."
n#00018916 3.848 relation "an abstraction belonging to or characteristic of two entities"
n#00013018 3.574 abstraction "a concept formed by extracting common features from examples"
...

amazon

Selectional Preferences for SRL (Zapirain et al., 2013)

WordNet SP models

- At test time, for a new argument of the predicate write with head word book:
  - $\Rightarrow$ consider $S = \{\texttt{<book>}\} \cup$ "all its hypernyms in WordNet" (for all senses of book)
  - $\Rightarrow$ $SP_{Res}(\textbf{write}, Arg1, book)$ returns the sum of the weights of the sysnsets in S matching the synsets in the list corresponding to the SP write–*Arg1*

amazon

Selectional Preferences for SRL (Zapirain et al., 2013)

Distributional SP models: based on Erk's (2007) setting

JFK was assassinated [in Texas]???

SP in–*TMP*: *November, century, month*

SP in–*LOC*: *Dallas, railway, city*

$$SP_{sim}(p,r,w) = \sum_{w_i \in Seen(p,r)} sim(w,w_i) \cdot weight(p,r,w_i)$$

$$\begin{aligned} SP(in, TMP, Texas) = \; & sim(Texas, November) \cdot weight(in, TMP, November) + \\ & sim(Texas, century) \cdot weight(in, TMP, century) + \\ & sim(Texas, month) \cdot weight(in, TMP, month) \end{aligned}$$

Selectional Preferences for SRL (Zapirain et al., 2013)

Distributional SP models: based on Erk's (2007) setting

JFK was assassinated [in Texas]???

SP in–*TMP*: *November, century, month*

SP in–*LOC*: *Dallas, railway, city*

$$SP_{sim}(p,r,w) = \sum_{w_i \in Seen(p,r)} sim(w,w_i) \cdot weight(p,r,w_i)$$

$$\begin{aligned} SP(in, TMP, Texas) = \; & sim(Texas, November) \cdot weight(in, TMP, November) + \\ & sim(Texas, century) \cdot weight(in, TMP, century) + \\ & sim(Texas, month) \cdot weight(in, TMP, month) \end{aligned}$$

Selectional Preferences for SRL (Zapirain et al., 2013)

Distributional SP models: based on Erk's (2007) setting

JFK was assassinated [in Texas]???

SP in–*TMP*: *November, century, month*

SP in–*LOC*: *Dallas, railway, city*

$$SP_{sim}(p,r,w) = \sum_{w_i \in Seen(p,r)} sim(w,w_i) \cdot weight(p,r,w_i)$$

$$\begin{aligned} SP(in, TMP, Texas) = \; & sim(Texas, November) \cdot freq(in, TMP, November) + \\ & sim(Texas, century) \cdot freq(in, TMP, century) + \\ & sim(Texas, month) \cdot freq(in, TMP, month) \end{aligned}$$

Selectional Preferences for SRL (Zapirain et al., 2013)

Distributional SP models: based on Erk's (2007) setting

JFK was assassinated [in Texas]???

SP in–*TMP*: *November, century, month*

SP in–*LOC*: *Dallas, railway, city*

$$SP_{sim}(p,r,w) = \sum_{w_i \in Seen(p,r)} sim(w,w_i) \cdot weight(p,r,w_i)$$

$$\begin{aligned} SP(in, LOC, Texas) = \; & sim(Texas, Dallas) \cdot freq(in, LOC, Dallas) + \\ & sim(Texas, railway) \cdot freq(in, LOC, railway) + \\ & sim(Texas, city) \cdot freq(in, LOC, city) \end{aligned}$$

SP(in,LOC,Texas) > SP(in,TMP,Texas)

Selectional Preferences for SRL (Zapirain et al., 2013)

Distributional SP models: based on Erk's (2007) setting

JFK was assassinated [in Texas]???

SP in–*TMP*: *November, century, month*

SP in–*LOC*: *Dallas, railway, city*

$$SP_{sim}(p,r,w) = \sum_{w_i \in Seen(p,r)} sim(w,w_i) \cdot weight(p,r,w_i)$$

$$\begin{aligned} SP(in, LOC, Texas) = \; & sim(Texas, Dallas) \cdot freq(in, LOC, Dallas) + \\ & sim(Texas, railway) \cdot freq(in, LOC, railway) + \\ & sim(Texas, city) \cdot freq(in, LOC, city) \end{aligned}$$

SP(in,LOC,Texas) > SP(in,TMP,Texas)

Selectional Preferences for SRL (Zapirain et al., 2013)

Distributional SP models: various instantiations for *sim*

- Using Padó and Lapata's software (2007) for computing distributional similarity measures
  - ⇒ Run on the British National Corpus
  - ⇒ Optimal parameterization as described in the paper
  - ⇒ Jaccard, cosine and Lin's similarity measures: sim_{Jac}, sim_{cos} and sim_{Lin}
- Using the already available Lin's thesaurus (Lin, 1998)
  - ⇒ Direct and second order similarity: sim^{th}_{Lin}, sim^{th2}_{Jac} and sim^{th2}_{cos}
  - ⇒ Average of both directions similarity

Evaluation of SPs in isolation (Zapirain et al., 2013)

Setting: Assign role labels to argument head words based solely on SP scores

$\Rightarrow$ For each head word (w), select the role (r) of the predicate or preposition (p) which fits best the head word:
$R_{sim}(p, w) = \arg\max_{r \in Roles(p)} SP_{sim}(p, r, w)$

$\Rightarrow$ SPs based on (p, r, w) triples from CoNLL-2005 data

$\Rightarrow$ In-domain (WSJ) and out-of-domain (Brown) test sets CoNLL-2005

$\Rightarrow$ Lexical baseline model: for a test pair (p, w), assign the role under which the head (w) occurred most often in the training data given the predicate (p)

amazon

Evaluation of SPs in isolation (Zapirain et al., 2013)

Setting: Assign role labels to argument head words based solely on SP scores

$\Rightarrow$ For each head word (w), select the role (r) of the predicate or preposition (p) which fits best the head word:
$R_{sim}(p,w) = \arg\max_{r \in Roles(p)} SP_{sim}(p,r,w)$

$\Rightarrow$ SPs based on (p,r,w) triples from CoNLL-2005 data

$\Rightarrow$ In-domain (WSJ) and out-of-domain (Brown) test sets CoNLL-2005

$\Rightarrow$ Lexical baseline model: for a test pair (p,w), assign the role under which the head (w) occurred most often in the training data given the predicate (p)

amazon

Evaluation of SPs in isolation (Zapirain et al., 2013)

Setting: Assign role labels to argument head words based solely on SP scores

$\Rightarrow$ For each head word (w), select the role (r) of the predicate or preposition (p) which fits best the head word:
$R_{sim}(p,w) = \arg\max_{r \in Roles(p)} SP_{sim}(p,r,w)$

$\Rightarrow$ SPs based on (p,r,w) triples from CoNLL-2005 data

$\Rightarrow$ In-domain (WSJ) and out-of-domain (Brown) test sets CoNLL-2005

$\Rightarrow$ Lexical baseline model: for a test pair (p,w), assign the role under which the head (w) occurred most often in the training data given the predicate (p)

amazon

Evaluation of SPs in isolation (Zapirain et al., 2013)

	WSJ-test			Brown		
	prec.	rec.	F_1	prec.	rec.	F_1
lexical	82.98	43.77	57.31	68.47	13.60	22.69
SP_{Res}	63.47	53.24	57.91	55.12	44.15	49.03
$SP_{sim_{Jac}}$	61.83	61.40	61.61	55.42	53.45	54.42
$SP_{sim_{cos}}$	64.67	64.22	64.44	56.56	54.54	55.53
$SP_{sim^{th2}_{Jac}}$	70.82	70.33	70.57	62.37	60.15	61.24
$SP_{sim^{th2}_{cos}}$	70.28	69.80	70.04	62.36	60.14	61.23

⇒ Lexical features have a high precision but very low recall

⇒ SPs are able to effectively generalize lexical features

⇒ SPs based on distributional similarity are better

⇒ Second-order similarity variants (Lin) attain the best results

Evaluation of SPs in isolation (Zapirain et al., 2013)

	WSJ-test			Brown		
	prec.	rec.	F_1	prec.	rec.	F_1
lexical	82.98	43.77	57.31	68.47	13.60	22.69
SP_{Res}	63.47	53.24	57.91	55.12	44.15	49.03
$SP_{sim_{Jac}}$	61.83	61.40	61.61	55.42	53.45	54.42
$SP_{sim_{cos}}$	64.67	64.22	64.44	56.56	54.54	55.53
$SP_{sim^{th2}_{Jac}}$	70.82	70.33	70.57	62.37	60.15	61.24
$SP_{sim^{th2}_{cos}}$	70.28	69.80	70.04	62.36	60.14	61.23

⇒ Lexical features have a high precision but very low recall

⇒ SPs are able to effectively generalize lexical features

⇒ SPs based on distributional similarity are better

⇒ Second-order similarity variants (Lin) attain the best results

SPs in a SRL System (Zapirain et al., 2013)

- *SwiRL* system for SRL (Surdeanu et al., 2007)
  - ⇒ System from CoNLL-2005 shared task (PropBank)
  - ⇒ Standard architecture (ML based on AdaBoost and SVMs)
  - ⇒ Best results from single (non-combined) systems at CoNLL-2005
- Simple approach: extending *SwiRL* features with SP predictions
  - ⇒ We train several extended *SwiRL*-SP_i models, one per selectional preferences model SP_i
  - ⇒ For each example (p, w) of *SwiRL*-SP_i, we add a single new feature whose value is the predicted role label $R_i(p, w)$

SPs in a SRL System (Zapirain et al., 2013)

Results

	WSJ-test			Brown		
	Core	Adj	All	Core	Adj	All
SwiRL	93.25	81.31	90.83	84.42	57.76	79.52
SwiRL+SP_{Res}	93.17	81.08	90.76	84.52	59.24	79.86
SwiRL+$SP_{sim_{Jac}}$	93.37	80.30	90.86	84.43	59.54	79.83
SwiRL+$SP_{sim_{cos}}$	93.33	80.92	90.87	85.14	60.16	80.50
SwiRL+$SP_{sim^{th2}_{Jac}}$	93.03	82.75	90.95	85.62	59.63	80.75
SwiRL+$SP_{sim^{th2}_{cos}}$	93.78	80.56	91.23	84.95	61.01	80.48

$\Rightarrow$ Slight improvements, especially noticeable on Brown corpus

$\Rightarrow$ Weak signal of a single feature?

SPs in a SRL System (Zapirain et al., 2013)

- Simple combinations of the individual *SwiRL+SP*$_i$ classifiers worked quite well (majority voting)
- We also trained a meta-classifier to combine the *SwiRL+SP*$_i$ classifiers and the stand-alone *SP*$_i$ models:
  - $\Rightarrow$ Binary classification approach: "is a proposed role correct or not?"
  - $\Rightarrow$ Features are based on the predictions of base *SP*$_i$ and *SwiRL+SP*$_i$ models
  - $\Rightarrow$ Trained with a SVM with a quadratic polynomial kernel

amazon

SPs in a SRL System (Zapirain et al., 2013)

Results (II)

	WSJ-test			Brown		
	Core	Adj	All	Core	Adj	All
SwiRL	93.25	81.31	90.83	84.42	57.76	79.52
$+SP_{sim^{th2}_{cos}}$	93.78	80.56	91.23	84.95	61.01	80.48
Meta	94.37	83.40	92.12	86.20	63.40	81.91

- Statistically significant improvements (99%) for both core and adjunct arguments, both in domain and out of domain

amazon

SPs in a SRL System (Zapirain et al., 2013)

Results (II)

	WSJ-test			Brown		
	Core	Adj	All	Core	Adj	All
SwiRL	93.25	81.31	90.83	84.42	57.76	79.52
$+SP_{sim^{th2}_{cos}}$	93.78	80.56	91.23	84.95	61.01	80.48
Meta	94.37	83.40	92.12	86.20	63.40	81.91

- Statistically significant improvements (99%) for both core and adjunct arguments, both in domain and out of domain

SPs in a SRL System (Zapirain et al., 2013)

Output analysis

- Manual inspection of 50 cases in which the meta classifier corrects SwiRL:
  - $\Rightarrow$ Usually cases with low frequency verbs or argument heads
  - $\Rightarrow$ In ~58% of the cases, syntax does not disambiguate, seems to suggest a wrong role label or it is confusing SwiRL because it is incorrect. However, most of the SP predictions are correct.
  - $\Rightarrow$ ~30% of the cases: unclear source of the SwiRL error but still several SP models suggest the correct role
  - $\Rightarrow$ ~12% of the cases: chance effect

amazon

SPs in a SRL System (Zapirain et al., 2013)

Output analysis

- Manual inspection of 50 cases in which the meta classifier corrects SwiRL:
  - $\Rightarrow$ Usually cases with low frequency verbs or argument heads
  - $\Rightarrow$ In ~58% of the cases, syntax does not disambiguate, seems to suggest a wrong role label or it is confusing SwiRL because it is incorrect. However, most of the SP predictions are correct.
  - $\Rightarrow$ ~30% of the cases: unclear source of the SwiRL error but still several SP models suggest the correct role
  - $\Rightarrow$ ~12% of the cases: chance effect

amazon

SPs in a SRL System (Zapirain et al., 2013)

Output analysis: example 1

		Several	JJ	(S1(S(NP*
		traders	NNS	*)
		could	MD	(VP*
		be	VB	(VP*
		seen	VBN	(VP*
		shaking	VBG	(S(VP*
		their	PRP$	(NP*
		heads	NNS	*)))
		when	WRB	(SBAR(WHADVP*)
A1	A0	the	DT	(S(NP*
A1	A0	news	NN	*)
	(P)	flashed	VBD	(VP*))))))
		.	.	*))

SPs in a SRL System (Zapirain et al., 2013)

Output analysis: example 2

		Italian	NNP	(S1(S(NP*
		President	NNP	*
		Francesco	NNP	*
		Cossiga	NNP	*)
	(P)	promised	VBD	(VP*
A2	A1	a	DT	(NP(NP*
A2	A1	quick	JJ	*
A2	A1	investigation	NN	*)
A2	A1	into	IN	(PP*
A2	A1	whether	IN	(SBAR*
A2	A1	Olivetti	NNP	(S(NP*)
A2	A1	broke	VBD	(VP*
A2	A1	Cocom	NNP	(NP*
A2	A1	rules	NNS	*))))))
		.	.	*))

SPs in a SRL System (Zapirain et al., 2013)

Output analysis: example 3

		Annual	JJ	(S(NP*
		payments	NNS	*)
		will	MD	(VP*
		more	RBR	(VP(ADVP*
		than	IN	*)
	(P)	double	VB	*
A3	TMP	from	IN	(PP*
A3	TMP	a	DT	(NP*
A3	TMP	year	NN	*
A3	TMP	ago	RB	*))
		to	TO	(PP*
		about	RB	(NP(QP*
		$240	CD	*
		million	CD	*)))
		...		

SPs in a SRL System (Zapirain et al., 2013)

Output analysis: example 4

		Procter	NNP	(S1(S(NP*
		&	CC	*
		Gamble	NNP	*
		Co.	NNP	*)
		plans	VBZ	(VP*
		to	TO	(S(VP*
		begin	VB	(VP*
	(P)	testing	VBG	(S(VP*
		next	JJ	(NP*
		month	NN	*)))
A1	A0	a	DT	(NP(NP*
A1	A0	superco.	JJ	*
A1	A0	detergent	NN	*)
A1	A0	that	WDT	(SBAR(WHNP*)
		...		
A1	A0	washload	NN	(NP*)))))))))))
		.	.	*))

SPs in a SRL System (Zapirain et al., 2013)

Some positive examples:

(a) Several traders could be seen shaking their heads when (([the **news**]$_{\text{Arg0}\Rightarrow\text{Arg1}}$)$^{\text{NP}}$ (*flashed*)$^{\text{VP}}$)$^{\text{S}}$.

(b) Italian President Francesco Cossiga (*promised* ([a quick **investigation** into whether Olivetti broke Cocom rules]$_{\text{Arg1}\Rightarrow\text{Arg2}}$)$^{\text{NP}}$)$^{\text{VP}}$.

(c) Annual payments (will more than *double* ([**from** (a year ago)$^{\text{NP}}$]$_{\text{TMP}\Rightarrow\text{Arg3}}$)$^{\text{PP}}$ to about \$240 million $\cdots$)$^{\text{VP}}$ $\cdots$

(d) Procter & Gamble Co. plans to (begin ((*testing* (next month)$^{\text{NP}}$)$^{\text{VP}}$)$^{\text{S}}$ ([a superco. **detergent** that $\cdots$ washload]$_{\text{Arg0}\Rightarrow\text{Arg1}}$)$^{\text{NP}}$)$^{\text{VP}}$.

amazon

SPs in a SRL System (Zapirain et al., 2013)

Some negative examples:

(a) Some "circuit breakers" installed after the October 1987 crash (*failed* ([their first **test**]$_{\text{Arg2}\Rightarrow\text{Arg1}}$)$^{\text{NP}}$)$^{\text{VP}}$...

(b) Many fund managers argue that now's ([the **time**]$_{\text{TMP}\Rightarrow\text{Arg1}}$)$^{\text{NP}}$ (*to buy*)$^{\text{VP}}$)$^{\text{S}}$.

(c) Telephone volume was up sharply, but it was still at just half the level of the weekend (*preceding* ([Black **Monday**]$_{\text{Arg1}\Rightarrow\text{TMP}}$)$^{\text{NP}}$)$^{\text{VP}}$.

Other Generalizations of Features

Embedded representations (learning with deep NNs)

- Collobert et al., JMLR 2011 (SENNA)
- Foland and Martin, NAACL 2015
- FitzGerald et al., EMNLP 2015
- Zhou and Xu, ACL 2015
- Roth and Lapata, ACL 2016

Talk Overview

amazon

SENNA

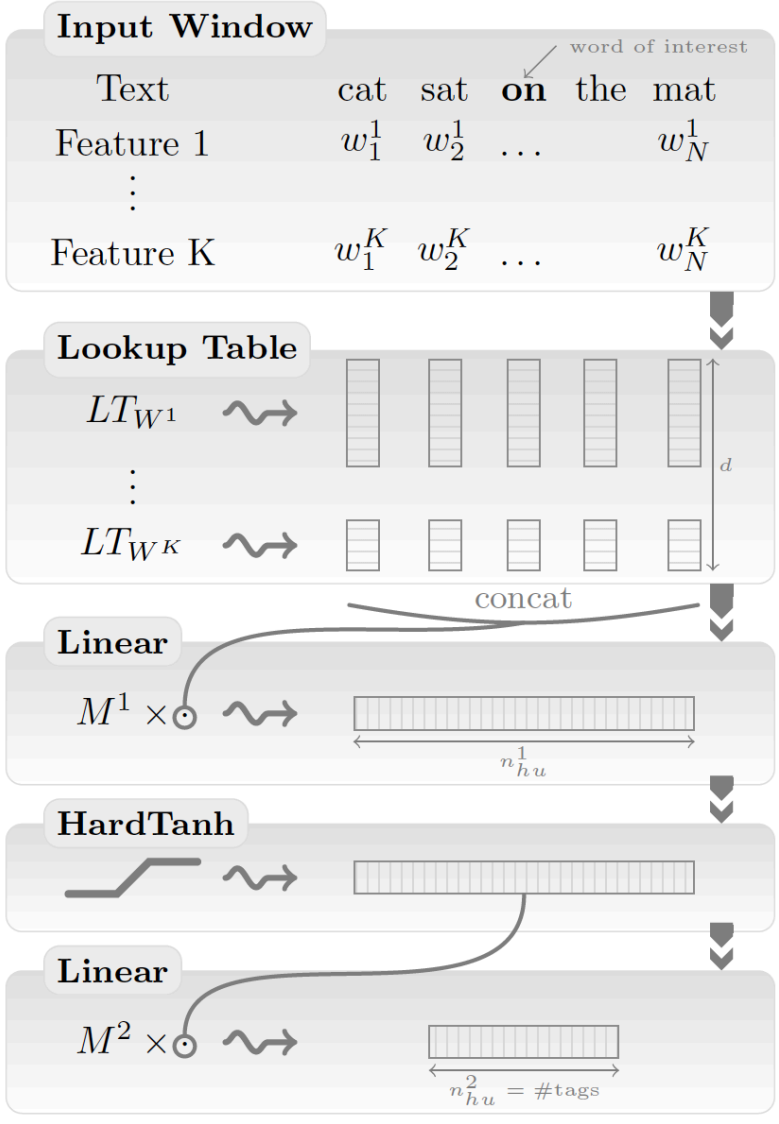
Natural Language Processing (Almost) from Scratch
(Collobert et al., JMLR 2011)

amazon

SENNA (Collobert et al., JMLR 2011)

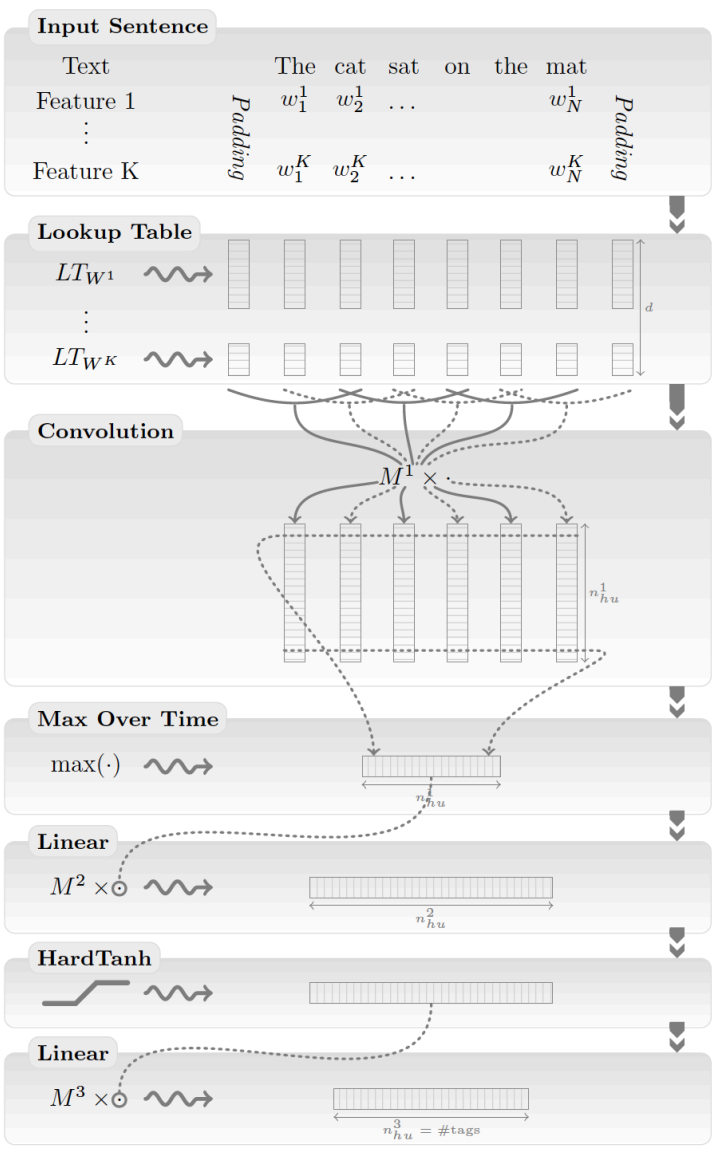
- "NLP from scratch" approach
- FFN (with convolutions) for multiple NLP tasks
- It does not rely on the output of existing NLP system
- Fast and with results close to the state-of-the-art

SENNA (Collobert et al., JMLR 2011)



amazon

SENNA (Collobert et al., JMLR 2011)



amazon

SENNA (Collobert et al., JMLR 2011)

Task		Benchmark	SENNA
Part of Speech (POS)	(Accuracy)	97.24 %	97.29 %
Chunking (CHUNK)	(F1)	94.29 %	94.32 %
Named Entity Recognition (NER)	(F1)	89.31 %	89.59 %
Parse Tree level 0 (PT0)	(F1)	91.94 %	92.25 %
Semantic Role Labeling (SRL)	(F1)	77.92 %	75.49 %

Table 15: Performance of the engineered sweet spot (SENNA) on various tagging tasks. The PT0 task replicates the sentence segmentation of the parse tree leaves. The corresponding benchmark score measures the quality of the Charniak parse tree leaves relative to the Penn Treebank gold parse trees.

SENNA (Collobert et al., JMLR 2011)

POS System	RAM (MB)	Time (s)
Toutanova et al. (2003)	800	64
Shen et al. (2007)	2200	833
SENNA	32	4

SRL System	RAM (MB)	Time (s)
Koomen et al. (2005)	3400	6253
SENNA	124	51

Table 16: Runtime speed and memory consumption comparison between state-of-the-art systems and our approach (SENNA). We give the runtime in seconds for running both the POS and SRL taggers on their respective testing sets. Memory usage is reported in megabytes.

Coding Syntactic Information for NN-based SRL

Dependency-Based Semantic Role Labeling using Convolutional Neural Networks

(Foland and Martin, NAACL 2015)

Global Inference with NN Factors

Semantic Role Labeling with Neural Network Factors
(FitzGerald et al., EMNLP 2015)

End-to-End SRL with LSTMs

End-to-end Learning of Semantic Role Labeling Using Recurrent Neural Networks
(Zhou and Xu, ACL 2015)

Task-specific NN Representations of Syntactic Paths

Neural Semantic Role Labeling with Dependency Path Embeddings
(Roth and Lapata, ACL 2016)

amazon

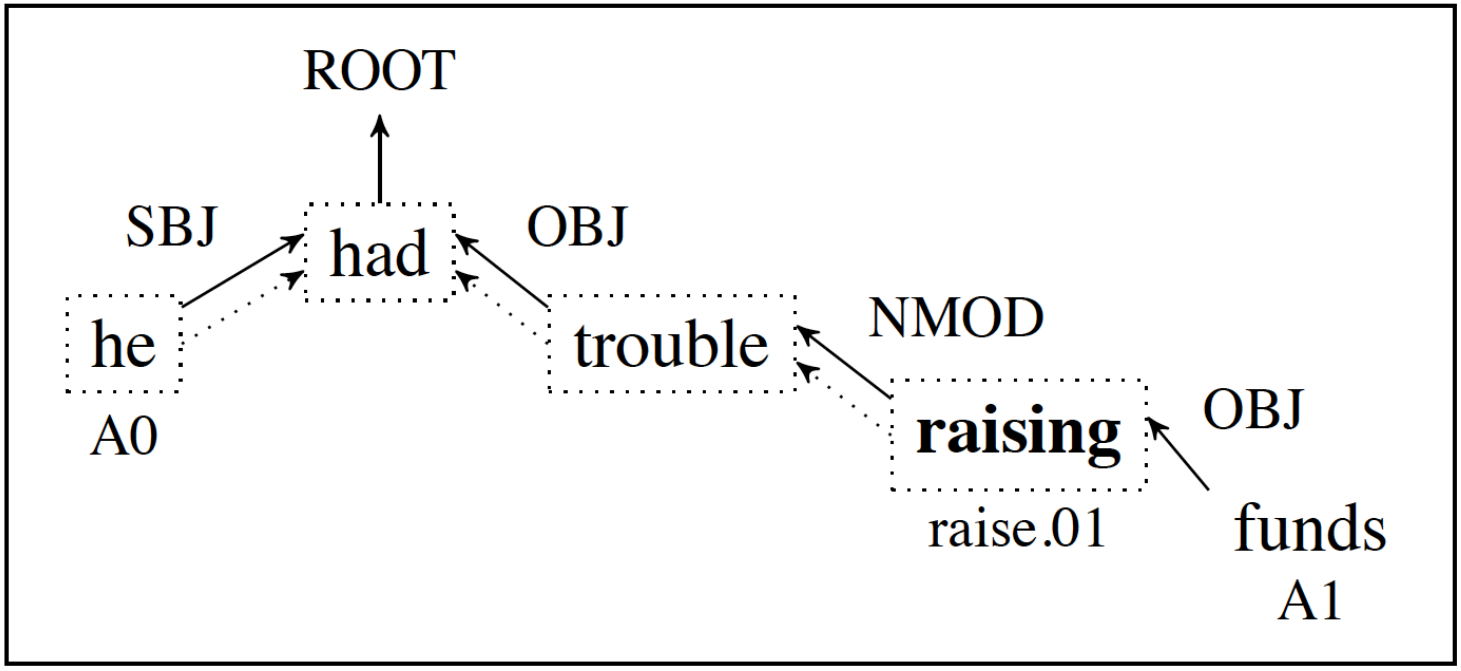
Task-specific NN Representations of Syntactic Paths

System	Analysis
mate-tools	*He had [trouble$_{A0}$] **raising** [funds$_{A1}$].
mateplus	*He had [trouble$_{A0}$] **raising** [funds$_{A1}$].
TensorSRL	*He had trouble **raising** [funds$_{A1}$].
easySRL	*He had trouble **raising** [funds$_{A1}$].
This work	[He$_{A0}$] had trouble **raising** [funds$_{A1}$].

Table 1: Outputs of SRL systems for the sentence *He had trouble raising funds*. Arguments of **raise** are shown with predicted roles as defined in PropBank (A0: getter of money; A1: money). Asterisks mark flawed analyses that miss the argument *He*.

amazon

Task-specific NN Representations of Syntactic Paths



amazon

Task-specific NN Representations of Syntactic Paths

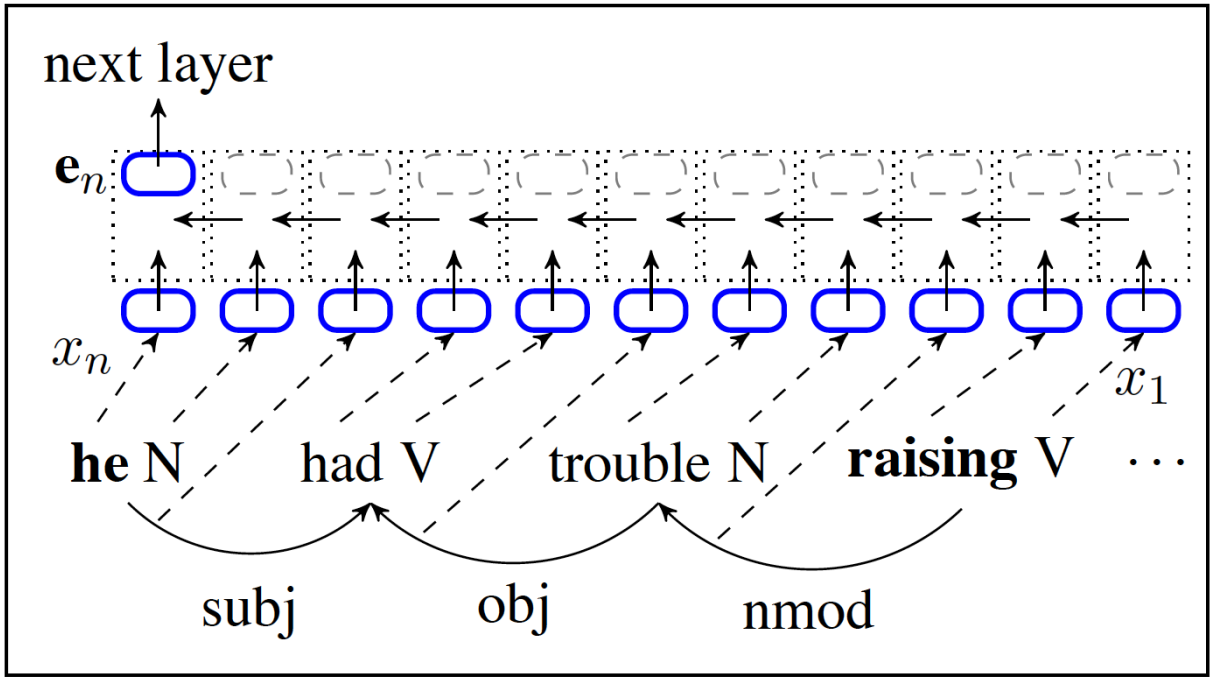
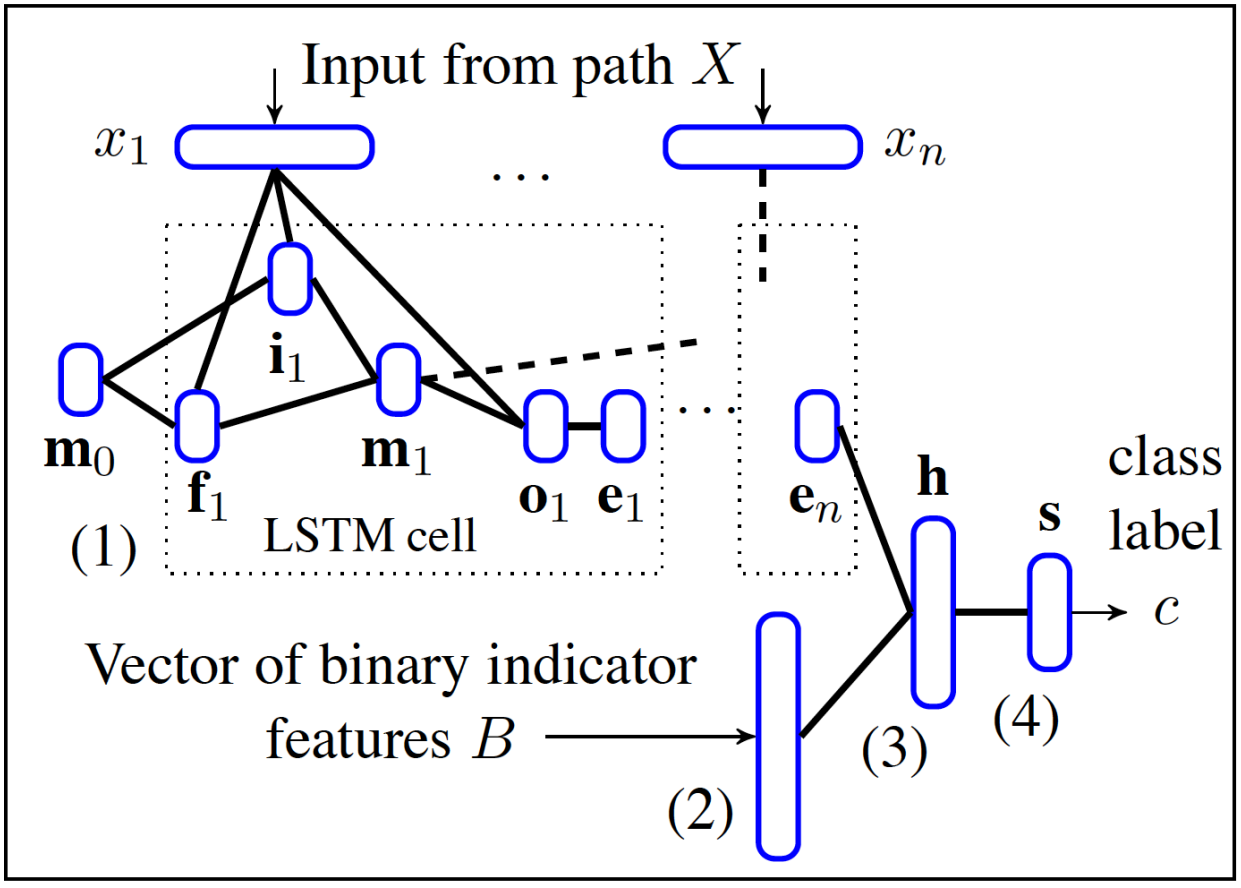


Figure 2: Example input and embedding computation for the path from *raising* to *he*, given the sentence *he had trouble raising funds*. LSTM time steps are displayed from right to left.

amazon

Task-specific NN Representations of Syntactic Paths



amazon

Task-specific NN Representations of Syntactic Paths

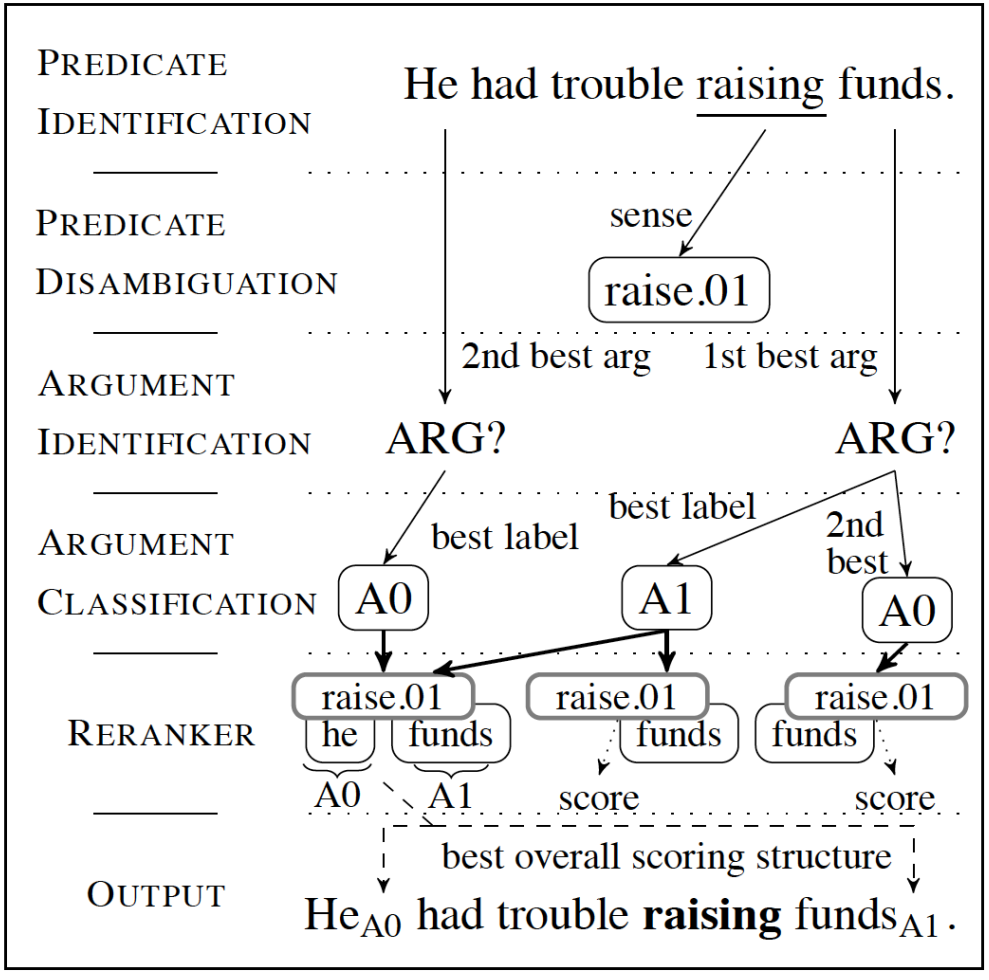


Figure 4: Pipeline architecture of our SRL system.

amazon

Task-specific NN Representations of Syntactic Paths

System (local, single)	P	R	F_1
Björkelund et al. (2010)	87.1	84.5	85.8
Lei et al. (2015)	–	–	86.6
FitzGerald et al. (2015)	–	–	86.7
PathLSTM w/o reranker	88.1	85.3	86.7
System (global, single)	**P**	**R**	**F_1**
Björkelund et al. (2010)	88.6	85.2	86.9
Roth and Woodsend (2014)[3]	–	–	86.3
FitzGerald et al. (2015)	–	–	87.3
PathLSTM	90.0	85.5	87.7
System (global, ensemble)	**P**	**R**	**F_1**
FitzGerald et al. 10 models	–	–	87.7
PathLSTM 3 models	**90.3**	**85.7**	**87.9**

Table 3: Results on the CoNLL-2009 in-domain test set. All numbers are in percent.

amazon

State-of-the-art Results

Deep semantic role labeling: What works and what's next
(He, Lee, Lewis and Zettlemoyer, ACL 2017)

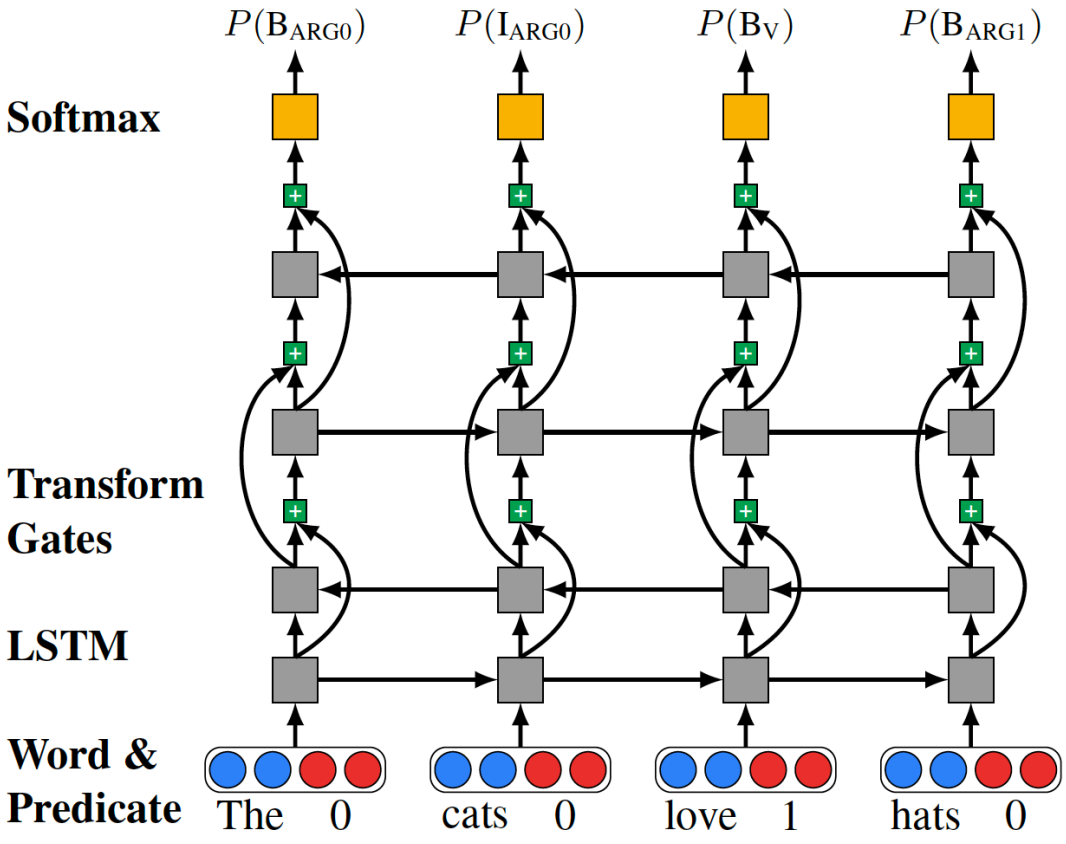
State-of-the-art Results (He, Lee, Lewis and Zettlemoyer, ACL 2017)

- End-to-end SRL system
- Deep highway BiLSTM architecture with constrained decoding
- No explicit syntactic information used
- Best results so far on the benchmark tasks

- Analysis:
  - ⇒ Deep models excel at recovering long-distance dependencies but can still make obvious errors,
  - ⇒ There is room for syntactic parsers to improve results

amazon

State-of-the-art Results (He, Lee, Lewis and Zettlemoyer, ACL 2017)



amazon

State-of-the-art Results (He, Lee, Lewis and Zettlemoyer, ACL 2017)

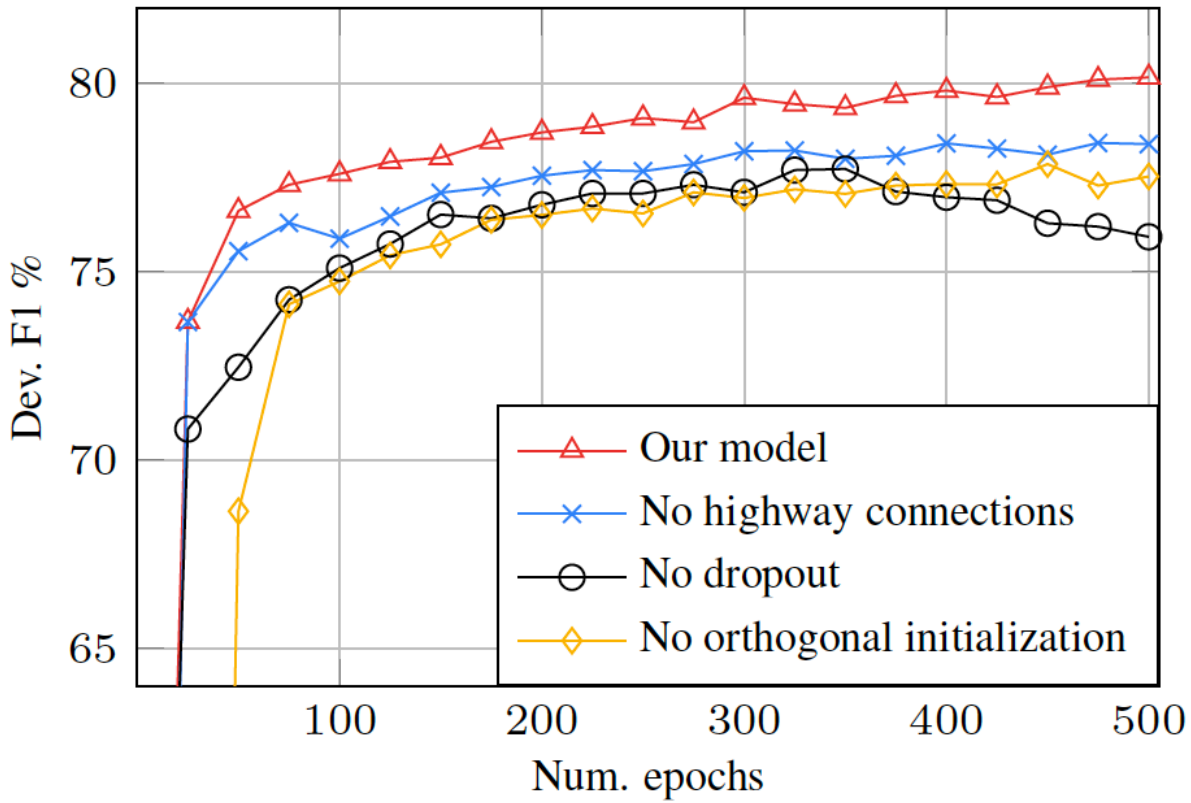
	Development				WSJ Test				Brown Test			
Method	P	R	F1	Comp.	P	R	F1	Comp.	P	R	F1	Comp.
Ours (PoE)	**83.1**	**82.4**	**82.7**	**64.1**	**85.0**	**84.3**	**84.6**	**66.5**	**74.9**	**72.4**	**73.6**	**46.5**
Ours	81.6	81.6	81.6	62.3	83.1	83.0	83.1	64.3	72.9	71.4	72.1	44.8
Zhou	79.7	79.4	79.6	-	82.9	82.8	82.8	-	70.7	68.2	69.4	-
FitzGerald (Struct.,PoE)	81.2	76.7	78.9	55.1	82.5	78.2	80.3	57.3	74.5	70.0	72.2	41.3
Täckström (Struct.)	81.2	76.2	78.6	54.4	82.3	77.6	79.9	56.0	74.3	68.6	71.3	39.8
Toutanova (Ensemble)	-	-	78.6	58.7	81.9	78.8	80.3	60.1	-	-	68.8	40.8
Punyakanok (Ensemble)	80.1	74.8	77.4	50.7	82.3	76.8	79.4	53.8	73.4	62.9	67.8	32.3

amazon

State-of-the-art Results (He, Lee, Lewis and Zettlemoyer, ACL 2017)

- "Tricks"
  - ⇒ Simplifying the input and output layers
  - ⇒ Introducing highway connections
  - ⇒ Using recurrent dropout
  - ⇒ Decoding with BIO constraints
  - ⇒ Ensembling with a product of experts

State-of-the-art Results (He, Lee, Lewis and Zettlemoyer, ACL 2017)



amazon

State-of-the-art Results (He, Lee, Lewis and Zettlemoyer, ACL 2017)

			SRL-Violations		
Model or Oracle	F1	Syn %	U	C	R
Gold	100.0	98.7	24	0	61
L8+PoE	82.7	94.3	37	3	68
L8	81.6	94.0	48	4	73
L6	81.4	93.7	39	3	85
L4	80.5	93.2	51	3	84
L2	77.2	91.3	96	5	72
L8+PoE+SRL	82.8	94.2	5	1	68
L8+PoE+AutoSyn	83.2	96.1	113	3	68
L8+PoE+GoldSyn	85.0	97.6	102	3	68
Punyakanok	77.4	95.3	0	0	0
Pradhan	78.3	93.0	84	3	58

amazon

Talk Overview

amazon

Joint work with

Xavier Lluís and *Xavier Carreras*

(Lluís et al. 2013) — TACL (presented at ACL)

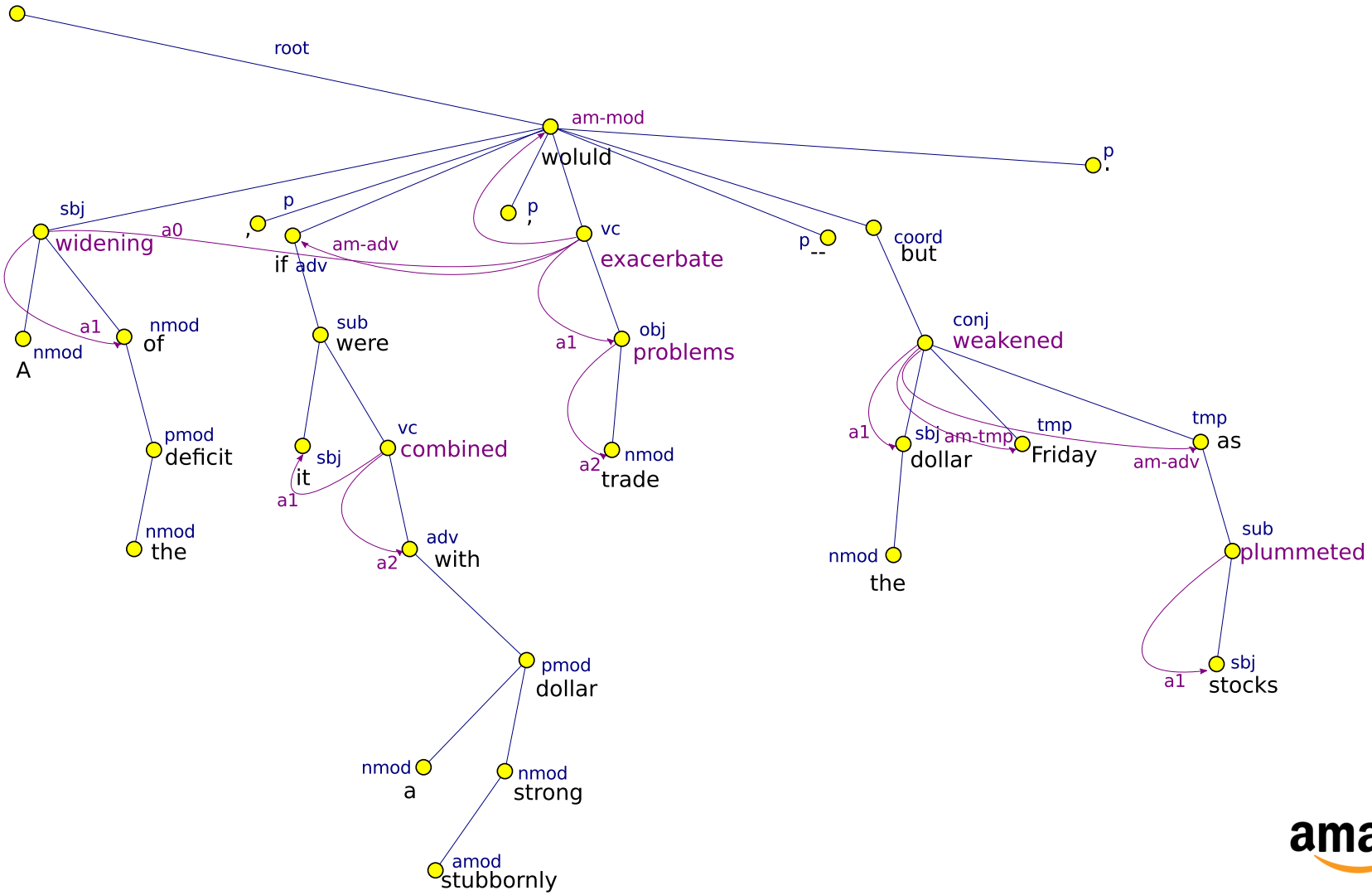
CoNLL-2008/2009 shared task

Joint parsing of syntactic and semantic dependencies

A widening of the deficit, if it were combined with a stubbornly strong dollar, would exacerbate trade problems – but the dollar weakened Friday as stocks plummeted.

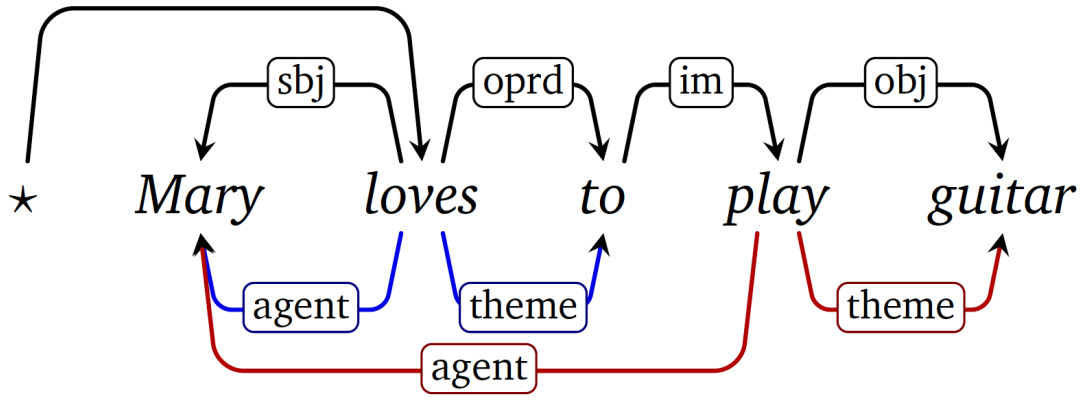
CoNLL-2008/2009 shared task

Joint parsing of syntactic and semantic dependencies



amazon

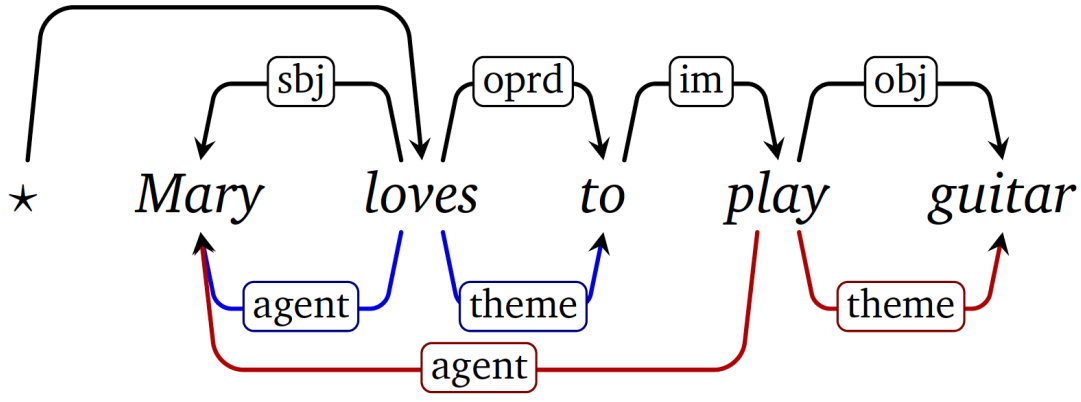
A Simplified Example



- Predicate-argument structures are naturally represented with dependencies

amazon

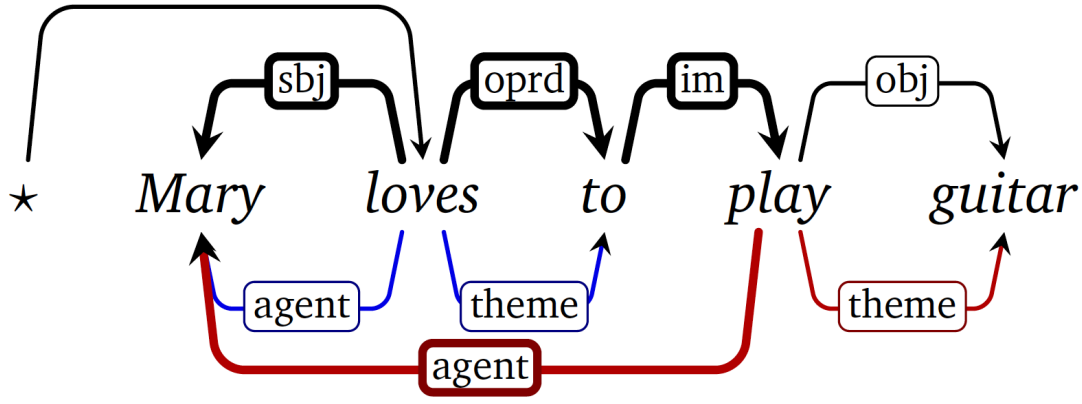
A Simplified Example



- Semantic roles are strongly related to syntactic structure
- Typical systems find semantic roles in a pipeline
  - ⇒ First obtain the syntactic tree
  - ⇒ Second obtain the semantic roles, using the syntactic tree
- Pipeline systems can not correct syntax based on semantic roles

amazon

A Simplified Example

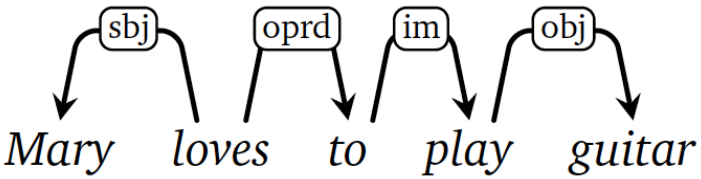


- We model the two structures jointly
  - ⇒ To capture interactions between syntactic and semantic dependencies
- Challenge:
  - ⇒ Some semantic dependencies are associated with a segment of syntactic dependencies
  - ⇒ Hard to factorize the two structures jointly

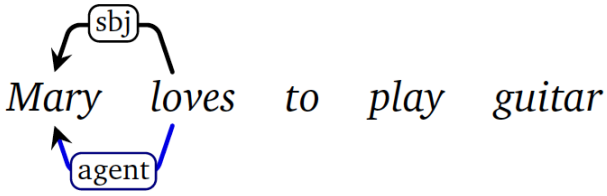
amazon

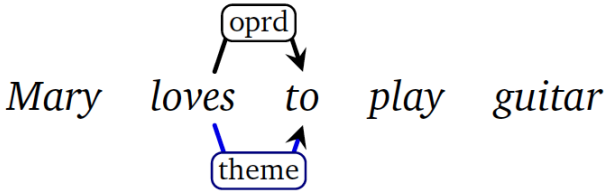
Decomposing Syntactic and Semantic Trees

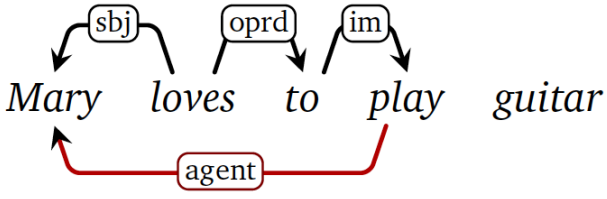
Syntactic Tree

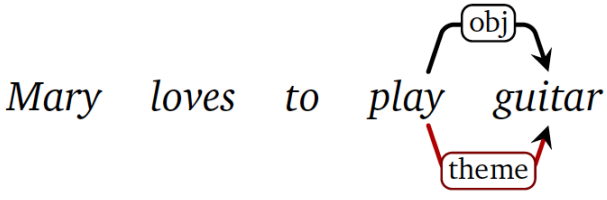


Semantic Trees with Local Syn.









Semantic trees need to agree with the syntactic tree.

Semantic features can conjoin

- any syntactic feature with
- a semantic role

amazon

Syntactic subproblem

$$syn(\mathbf{x}) = \operatorname*{argmax}_{\mathbf{y}} \ \text{score\_syn}(\mathbf{x}, \mathbf{y})$$
$$\text{subject to} \quad \text{cTree: } \mathbf{y} \text{ is a projective tree}$$

- Solved by a standard dependency parsing algorithm
- score_syn$(\mathbf{x}, \mathbf{y})$ is arc-factored: 1st and 2nd order models
- Graph-based parsing algorithms, reimplementing (McDonald, 2005; Carreras et al., 2007)
- Trained with (linear) average structure perceptron using state-of-the-art features

amazon

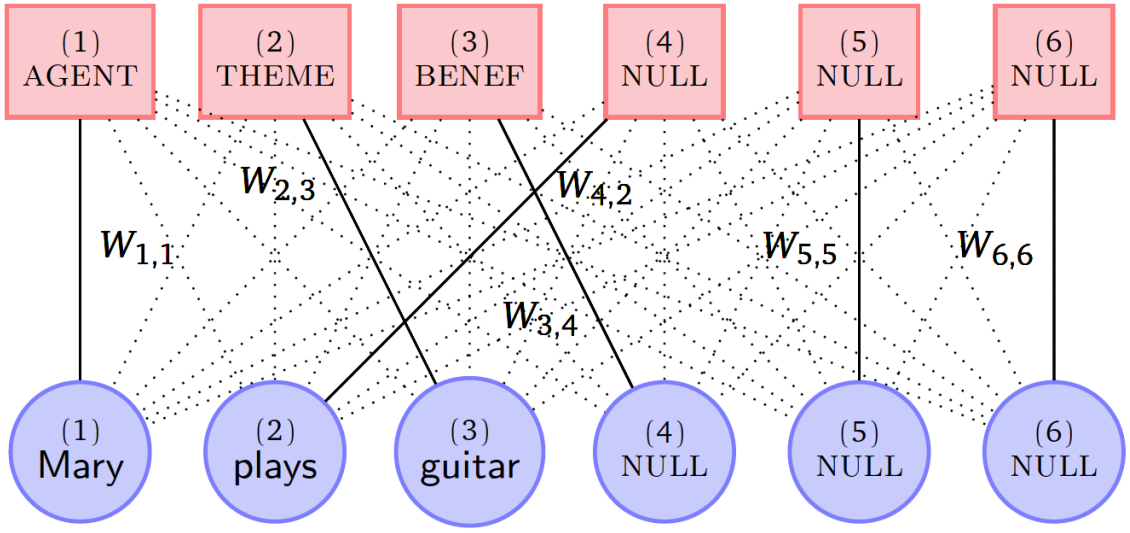
Semantic Subproblem

$$
\begin{aligned}
srl(\mathbf{x}) \quad = \quad & \operatorname*{argmax}_{\mathbf{z},\boldsymbol{\pi}} \ \text{score\_srl}(\mathbf{x}, \mathbf{z}, \boldsymbol{\pi}) \\
\text{subject to} \quad & \text{cRole: no repeated roles} \\
& \text{cArg: at most one role per token} \\
& \text{cPath: } \boldsymbol{\pi} \text{ codifies paths consistent with } \mathbf{z}
\end{aligned}
$$

- In a predicate:
  - $\Rightarrow$ A token appears at most once as argument
  - $\Rightarrow$ A semantic role appears at most once
- score_srl$(\mathbf{x}, \mathbf{z}, \boldsymbol{\pi})$ is factorized at the level of $\langle \mathbf{x}, p, a, r, \pi^{p,a,r} \rangle$
- local score_srl$(\mathbf{x}, p, a, r, \pi^{p,a,r})$ provided by linear classifiers
- We frame the *argmax* inference as a *linear assignment problem*

amazon

SRL as Assignment



- The Hungarian algorithm solves it in $\mathcal{O}(n^3)$
- $w_{i,j}$ are the previous local predictions score_srl$(\mathbf{x}, p, a, r, \pi^{p,a,r})$
- In practice, the list of most likely paths from p to a is pre-computed using syntactic models
- Learning is performed with structure perceptron, with feedback applied after solving the assignment problem

amazon

Joint Syntactic-Semantic Inference

$$\langle \mathbf{y}^*, \mathbf{z}^*, \boldsymbol{\pi}^* \rangle = \underset{\mathbf{y},\mathbf{z},\boldsymbol{\pi}}{\text{argmax}} \ \text{sc\_syn}(\mathbf{x}, \mathbf{y}) + \text{sc\_srl}(\mathbf{x}, \mathbf{z}, \boldsymbol{\pi})$$

subject to cTree, cRole, cArg, cPath

cSubtree: $\mathbf{y}$ is consistent with $\boldsymbol{\pi}$

amazon

Joint Syntactic-Semantic Inference

$$\langle \mathbf{y}^*, \mathbf{z}^*, \boldsymbol{\pi}^* \rangle = \underset{\mathbf{y},\mathbf{z},\boldsymbol{\pi}}{\text{argmax}} \ \text{sc\_syn}(\mathbf{x}, \mathbf{y}) + \text{sc\_srl}(\mathbf{x}, \mathbf{z}, \boldsymbol{\pi})$$

subject to cTree, cRole, cArg, cPath

cSubtree: $\mathbf{y}$ is consistent with $\boldsymbol{\pi}$

cSubtree constraints can be easily expressed as:

$$\forall d \in \mathbf{y}\,, \quad c \cdot \mathbf{y}_d \geqslant \sum_{p,a,r \in \mathbf{z}} \boldsymbol{\pi}_d^{p,a,r}$$

or, equivalently, as equality constraints

$$\forall d \in \mathbf{y}\,, \quad c \cdot \mathbf{y}_d - \sum_{p,a,r \in \mathbf{z}} \boldsymbol{\pi}_d^{p,a,r} - \xi_d = 0$$

Joint Syntactic-Semantic Inference

- We employed Dual Decomposition to solve the joint inference (Rush and Collins, 2011; Sontag et al 2010)
- Lagrangian relaxation-based method that iteratively solves decomposed sub-problems with agreement constraints:
  - $\Rightarrow$ Subtree constraints are relaxed by introducing Lagrange multipliers for every dependency λ_d
  - $\Rightarrow$ Subproblems now depend on the $\boldsymbol{\lambda}$ penalty variables but can be efficiently solved
  - $\Rightarrow$ Syntax: standard dependency parsing inference
  - $\Rightarrow$ Semantic: linear assignment
- Guaranteed optimal solution when it converges
- In experiments, convergence in $> 99.5\%$ of sentences

amazon

Experiments and Results

We ran experiments on the CoNLL-2009 datasets with the following configurations:

Pipeline	best *syn* then best *srl* enforcing cArg
+Assignment	enforces cRole, cArg over best *syn*
Forest	works with a forest of *syn* trees
DD	applies dual-decomposition

amazon

Experiments and Results

	syn	sem		
system	acc	prec	rec	F_1
Pipeline-1				
+Assignment-1				
Forest-1				
DD-1				

Results on WSJ development set

amazon

Experiments and Results

	syn	sem		
system	acc	prec	rec	F_1
Pipeline-1	85.32	86.23	67.67	75.83
+Assignment-1	85.32	84.08	71.82	77.47
Forest-1				
DD-1				

+Assignment improves over *Pipeline*

amazon

Experiments and Results

	syn	sem		
system	acc	prec	rec	F_1
Pipeline-1	85.32	86.23	67.67	75.83
+Assignment-1	85.32	84.08	71.82	77.47
Forest-1	85.32	80.67	73.60	76.97
DD-1				

Forests shows higher recall

amazon

Experiments and Results

	syn	sem		
system	acc	prec	rec	F_1
Pipeline-1	85.32	86.23	67.67	75.83
+Assignment-1	85.32	84.08	71.82	77.47
Forest-1	85.32	80.67	73.60	76.97
DD-1	85.48	83.99	72.69	77.94

DD-1 achieves better sem F_1

amazon

Experiments and Results

	syn	sem		
system	acc	prec	rec	F_1
Pipeline-1	85.32	86.23	67.67	75.83
+Assignment-1	85.32	84.08	71.82	77.47
Forest-1	85.32	80.67	73.60	76.97
DD-1	85.48	83.99	72.69	77.94
Pipeline-2	87.77	87.07	68.65	76.77
+Assignment-2	87.77	85.21	73.41	78.87
Forest-2	87.77	80.67	73.60	76.97
DD-2	87.84	85.20	73.23	78.79

Second-order paths are quite accurate

amazon

Experiments and Results

	syn	sem			
WSJ	acc	prec	rec	F_1	PP
Lluís09	87.48	73.87	67.40	70.49	39.68
Merlo09	88.79	81.00	76.45	78.66	54.80
DD-2	89.21	86.01	74.84	80.04	55.73

Results in WSJ corpus (in-domain) test set

amazon

Experiments and Results

	syn	sem			
WSJ	acc	prec	rec	F_1	PP
Lluís09	87.48	73.87	67.40	70.49	39.68
Merlo09	88.79	81.00	76.45	78.66	54.80
DD-2	89.21	86.01	74.84	80.04	55.73

Better results than *Merlo09*

Experiments and Results

	syn	sem			
Brown	acc	prec	rec	F_1	PP
Lluís09	80.92	62.29	59.22	60.71	29.79
Merlo09	80.84	68.97	63.06	65.89	38.92
DD-2	82.61	74.12	61.59	67.83	38.92

Results in Brown corpus (out-of-domain) test set

amazon

Talk Overview

amazon

Other Important Topics

1. Learning with *latent variables/structures*
   ⇒ Henderson et al., Computational Linguistics 39(4), 2013
2. Unsupervised models for SRL
   ⇒ Titov and Khoddam, NAACL 2015
3. Learning with *weak/distant* supervision
4. Cross-language approaches to SRL

Talk Overview

amazon

Some Random Comments

- NLP technology is very important for a number of current applications:
  - ⇒ MT, personal assistants, information search and analysis, Q&A, dialog systems, market study, trends, opinions, etc.
  - ⇒ The new Artificial Intelligence
- NLP current approaches are empirical
  - ⇒ based on data, statistics, and machine learning
  - ⇒ big data
- ML is at many stages of NLP state-of-the-art solutions
  - ⇒ and it is here to stay...
  - ⇒ "new" trend on distributed representations and deep NNs

Some Random Comments

- NLP technology is very important for a number of current applications:
  - ⇒ MT, personal assistants, information search and analysis, Q&A, dialog systems, market study, trends, opinions, etc.
  - ⇒ The new Artificial Intelligence
- NLP current approaches are empirical
  - ⇒ based on data, statistics, and machine learning
  - ⇒ big data
- ML is at many stages of NLP state-of-the-art solutions
  - ⇒ and it is here to stay...
  - ⇒ "new" trend on distributed representations and deep NNs

Some Random Comments

- NLP technology is very important for a number of current applications:
  - ⇒ MT, personal assistants, information search and analysis, Q&A, dialog systems, market study, trends, opinions, etc.
  - ⇒ The new Artificial Intelligence
- NLP current approaches are empirical
  - ⇒ based on data, statistics, and machine learning
  - ⇒ big data
- ML is at many stages of NLP state-of-the-art solutions
  - ⇒ and it is here to stay...
  - ⇒ "new" trend on distributed representations and deep NNs

Some Random Comments

- If you want to work for Google, Amazon, Facebook, Yahoo, Twitter, MSR, IBM Watson...
  But also Bloomberg, Goldman Sachs, Machine Zone, etc.
- You better learn about:
  $\Rightarrow$ Machine Learning, NLP, Statistics, Text mining, etc.
- ...and you conduct a PhD first
  (great work opportunities at the moment)

amazon

Some Random Comments

- If you want to work for Google, Amazon, Facebook, Yahoo, Twitter, MSR, IBM Watson...
  But also Bloomberg, Goldman Sachs, Machine Zone, etc.
- You better learn about:
  - $\Rightarrow$ Machine Learning, NLP, Statistics, Text mining, etc.
- ...and you conduct a PhD first
  (great work opportunities at the moment)

amazon

Some Random Comments

- If you want to work for Google, Amazon, Facebook, Yahoo, Twitter, MSR, IBM Watson...
  But also Bloomberg, Goldman Sachs, Machine Zone, etc.
- You better learn about:
  - $\Rightarrow$ Machine Learning, NLP, Statistics, Text mining, etc.
- ...and you conduct a PhD first
  (great work opportunities at the moment)

amazon

Some Random Comments

- If you want to work for Google, Amazon, Facebook, Yahoo, Twitter, MSR, IBM Watson...
  But also Bloomberg, Goldman Sachs, Machine Zone, etc.
- You better learn about:
  - $\Rightarrow$ Machine Learning, NLP, Statistics, Text mining, etc.
- ...and you conduct a PhD first
  (great work opportunities at the moment)

amazon

Some Random Comments

- I opted for taking a complex enough NLP task (SRL) as an excuse to cover many NLP-ML topics
- We have overviewed many important concepts and methods of Machine Learning for NLP (especially supervised)
- But there are MANY MORE that we left untouched
  $\Rightarrow$ some of them currently very TRENDY!

amazon

Some Random Comments

- I opted for taking a complex enough NLP task (SRL) as an excuse to cover many NLP-ML topics
- We have overviewed many important concepts and methods of Machine Learning for NLP (especially supervised)
- But there are MANY MORE that we left untouched
  $\Rightarrow$ some of them currently very TRENDY!

amazon

Some Random Comments

- I opted for taking a complex enough NLP task (SRL) as an excuse to cover many NLP-ML topics
- We have overviewed many important concepts and methods of Machine Learning for NLP (especially supervised)
- But there are MANY MORE that we left untouched
  - $\Rightarrow$ some of them currently very TRENDY!

amazon

on Semantic Role Labeling

- SRL is an important problem in NLP, strongly related to applications requiring some degree of semantic interpretation
- It is an active topic of research, which has generated an important body of work in the last 10 years
  $\Rightarrow$ techniques, resources, applications

Some news are good but...

$\Rightarrow$ SRL still has to resolve important problems before we see a spread usage in real open-domain applications

$\Rightarrow$ A jump is needed from the laboratory conditions to the real world.

on Semantic Role Labeling

- SRL is an important problem in NLP, strongly related to applications requiring some degree of semantic interpretation
- It is an active topic of research, which has generated an important body of work in the last 10 years
  $\Rightarrow$ techniques, resources, applications

Some news are good but...

$\Rightarrow$ SRL still has to resolve important problems before we see a spread usage in real open-domain applications

$\Rightarrow$ A jump is needed from the laboratory conditions to the real world.

Final Slide

I hope you enjoyed this part of the course!!!

amazon

Machine Learning applied to Natural Language Processing

Lluís Màrquez

amazon

Advanced Methods for Corpus Analysis

EM LCT – European Masters Program in Language and Communication Technologies

Donostia, June 6-8, 2018