Semantic Role Labeling Tutorial: Part 2
Supervised Machine Learning methods

Shumin Wu
The bed on which I slept broke.
The bed broke on which I slept.
SRL Supervised ML Pipeline

Syntactic Parse

NP₁
He

S

NP₁
VP
Walked

PP
in

NP₂
the park

Prune Constituents

supervised ML

NP₁
V
Yes
Given

PP
Yes

NP₂
No

Argument Identification

Arguments

supervised ML

NP₁
ARG0/ARG1
V
Predicate

PP
ARG1/AM-LOC

Structural Inference

heuristic or ML optimization

Semantic Roles

Candidates

NP₁
ARG0/ARG1
V
Predicate

PP
ARG1/AM-LOC
Pruning Algorithm [Xue, Palmer 2004]

- For the predicate and each of its ancestors, collect their sisters unless the sister is *coordinated* with the predicate.
- If a sister is a PP also collect its immediate children.
ML for Argument Identification/Labeling

1. Extract features from sentence, syntactic parse, and other sources for each candidate constituent
2. Train statistical ML classifier to identify arguments
3. Extract features same as or similar to those in step 1
4. Train statistical ML classifier to select appropriate label for arguments
   - SVM, Linear (MaxEnt, LibLinear, etc), structured (CRF) classifiers
   - All vs one, pairwise, structured multi-label classification
Commonly Used Features: Phrase Type

- Intuition: different roles tend to be realized by different syntactic categories
- For dependency parse, the dependency label can serve similar function
- Phrase Type indicates the syntactic category of the phrase expressing the semantic roles
- Syntactic categories from the Penn Treebank
- FrameNet distributions:
  - NP (47%) – noun phrase
  - PP (22%) – prepositional phrase
  - ADVP (4%) – adverbial phrase
  - PRT (2%) – particles (e.g. make something up)
  - SBAR (2%), S (2%) - clauses
He heard the sound of liquid slurping in a metal container as Farell approached him from behind.
Features: Governing Category

- Intuition: There is often a link between semantic roles and their syntactic realization as subject or direct object
- *He drove the car over the cliff*
  - Subject NP more likely to fill the agent role
- Approximating grammatical function from parse
  - Function tags in constituent parses (typically not recovered in automatic parses)
  - Dependency labels in dependency parses
Features: Governing Category

can you blame the dealer for being late?
Features: Parse Tree Path

- Intuition: need a feature that factors in relation to the target word.
- Feature representation: string of symbols indicating the up and down traversal to go from the target word to the constituent of interest.
- For dependency parses, use dependency path.
### Features: Parse Tree Path

<table>
<thead>
<tr>
<th>Frequency</th>
<th>Path</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>14.2%</td>
<td>VB↑VP↓PP</td>
<td>PP argument/adjunct</td>
</tr>
<tr>
<td>11.8</td>
<td>VB↑VP↑S↓NP</td>
<td>subject</td>
</tr>
<tr>
<td>10.1</td>
<td>VB↑VP↓NP</td>
<td>object</td>
</tr>
<tr>
<td>7.9</td>
<td>VB↑VP↑VP↑S↓NP</td>
<td>subject (embedded VP)</td>
</tr>
<tr>
<td>4.1</td>
<td>VB↑VP↓ADVP</td>
<td>adverbial adjunct</td>
</tr>
<tr>
<td>3.0</td>
<td>NN↑NP↑NP↓PP</td>
<td>prepositional complement of noun</td>
</tr>
<tr>
<td>1.7</td>
<td>VB↑VP↓PRT</td>
<td>adverbial particle</td>
</tr>
<tr>
<td>1.6</td>
<td>VB↑VP↑VP↑VP↑S↓NP</td>
<td>subject (embedded VP)</td>
</tr>
<tr>
<td>14.2</td>
<td>no matching parse constituent</td>
<td></td>
</tr>
<tr>
<td>31.4</td>
<td>Other</td>
<td>none</td>
</tr>
</tbody>
</table>
Features: Parse Tree Path

- Issues:
  - Parser quality (error rate)
  - Data sparseness
    - 2978 possible values excluding frame elements with no matching parse constituent
      - Compress path by removing consecutive phrases of the same type, retain only clauses in path, etc
    - 4086 possible values including total of 35,138 frame elements identifies as NP, only 4% have path feature without VP or S ancestor [Gildea and Jurafsky, 2002]
Features: Subcategorization

- List of child phrase types of the VP
  - highlight the constituent in consideration
- Intuition: Knowing the number of arguments to the verb constrains the possible set of semantic roles
- For dependency parse, collect dependents of predicate
Intuition: grammatical function is highly correlated with position in the sentence
- Subjects appear before a verb
- Objects appear after a verb

Representation:
- Binary value – does node appear before or after the predicate

Can you blame the dealer for being late?

before after after
Features: Voice

- Intuition: Grammatical function varies with voice
  - Direct objects in active ⇔ Subject in passive
    - He slammed the door.
    - The door was slammed by him.

- Approach:
  - Use passive identifying patterns / templates (language dependent)
    - Passive auxiliary (to be, to get), past participle
    - bei construction in Chinese
Features: Tree kernel

- Compute sub-trees and partial-trees similarities between training parses and decoding parse.
Features: Tree kernel

- Does not require exact feature match
  - Advantage when training data is small (less likely to have exact feature match)
- Well suited for kernel space classifiers (SVM)
  - All possible sub-trees and partial trees do not have to be enumerated as individual features
  - Tree comparison can be made in polynomial time even when the number of possible sub/partial trees are exponential
More Features

- **Head word**
  - Head of constituent
- **Name entities**
- **Verb cluster**
  - Similar verbs share similar argument sets
- **First/last word of constituent**
- **Constituent order/distance**
  - Whether certain phrase types appear before the argument
- **Argument set**
  - Possible arguments in frame file
- **Previous role**
  - Last found argument type
- **Argument order**
  - Order of arguments from left to right
Verb predicate annotation doesn’t always capture fine semantic details:

The fed is *considering* interest rate *reduction* of a quarter point at the next meeting.
Arguments of Nominal Predicates

- Can be harder to classify because arguments are not as well constrained by syntax

- Find the “supporting” verb predicate and its argument candidates
  - Usually under the VP headed by the verb predicate and is part of an argument to the verb
Structural Inference

- Take advantage of predicate-argument structures to re-rank argument label set
  - Arguments should not overlap
    - Can you blame the dealer for being late?
    - ARG0
    - ARG1: 0.5
    - ARG2: 0.8
    - ARG1: 0.6
  - Numbered arguments (arg0-5) should not repeat
    - John sold Mary the book
    - ARG0
    - ARG1: 0.6
    - ARG2: 0.4
    - ARG1: 0.8
    - ARG2: 0.2
  - R-arg[type] and C-arg[type] should have an associated arg[type]
    - The bed on which I slept broke
    - not arg: 0.6
    - AM-loc: 0.4
    - R-AM-loc
    - ARG0
Structural Inference Methods

- Optimize log probability of label set \( \left( \sum_{i=1}^{n} \log(p(A_i)) / n \right) \)
  - Beam search
  - Formulate into integer linear programming (ILP) problem
- Re-rank top label sets that conform to constraints
  - Choose n-best label sets
  - Train structural classifier (CRF, etc)
Syntactic parse input

Training parse accuracy needs to match decoding parse accuracy
- Generate parses via cross-validation
- Cross-validation folds need to be selected with low correlation
  - Training data from the same document source needs to be in the same fold

Separate stages of constituent pruning, argument identification and argument labeling
- Constituent pruning and argument identification reduce training/decoding complexity, but usually incur a slight accuracy penalty
Linear Classifier Notes

- Popular choices: LibLinear, MaxEnt, RRM
- Perceptron model in feature space
  - each feature \( j \) contributes positively or negatively to a label \( i \)
    \[
    L_i = \text{sign}(w_{i,0} + \sum_j f_j w_{i,j})
    \]
- How about position and voice features for classifying the agent?
  - \textbf{He} slammed the door.
  - \textit{The door was slammed by} \textbf{him}.
  - Position (\textit{left}): positive indicator since active construction is more frequent
  - Voice (\textit{active}): weak positive indicator by itself (agent can be omitted in passive construction)
- Combine the 2 features as a single feature
  - \textit{left-active} and \textit{right-passive} are strong positive indicators
  - \textit{left-passive} and \textit{right-active} are strong negative indicators
Popular choices: LibSVM, SVM\textsuperscript{light}

Kernel space classification (linear kernel example)

- The correlation ($c_j$) of the features of the input sample with each training sample $j$ contributes positively or negatively to a label $i$

$$L_i = sign(w_{i,0} + \sum_j c_j w_{i,j})$$

- Creates $n \times n$ dense correlation matrix during training ($n$ is the size of training samples)
- Requires a lot of memory during training for large corpus
  - Use a linear classifier for argument identification
  - Train base model with a small subset of samples, iteratively add a portion of incorrectly classified training samples and retrain
- Decoding speed not as adversely affected
  - Trained model typically only has a small number of “support vectors”
- Tend to perform better when training data is limited
Evaluation

- Precision – percentage of labels output by the system which are correct
- Recall – recall percentage of true labels correctly identified by the system
- F-measure, F_\beta – harmonic mean of precision and recall

\[
F = \frac{2PR}{P + R}
\]
\[
F_\beta = \frac{(1 + \beta^2)PR}{\beta^2 P + R}
\]
Evaluation

Lots of choices when evaluating in SRL:

- **Arguments**
  - Full span (CoNLL-2005)
  - Headword only (CoNLL-2008)

- **Predicates**
  - Given (CoNLL-2005)
  - System Identifies (CoNLL-2008)
  - Verb and nominal predicates (CoNLL-2008)
### Evaluation

<table>
<thead>
<tr>
<th>Gold Standard Labels</th>
<th>SRL Output</th>
<th>Full</th>
<th>Head</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arg0: John</td>
<td>Arg0: John</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Rel: mopped</td>
<td>Rel: mopped</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Arg1: the floor</td>
<td>Arg1: the floor</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Arg2: with the dress</td>
<td>Arg2: with the dress</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>Thailand</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arg0: Mary</td>
<td>Arg0: Mary</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Rel: bought</td>
<td>Rel: bought</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Arg1: the dress</td>
<td>Arg1: the dress</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Arg0: Mary</td>
<td>Arg0: Mary</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>rel: studying</td>
<td></td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Argm-LOC: in Thailand</td>
<td></td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Arg0: Mary</td>
<td>Arg0: Mary</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Rel: traveling</td>
<td>Rel: traveling</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Argm-LOC: in Thailand</td>
<td></td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

John mopped the floor with the dress Mary bought while studying and traveling in Thailand.

**Evaluated on Full Arg Span**
- Precision: $P = 8$ correct / $10$ labeled = $80.0\%$
- Recall: $R = 8$ correct / $13$ possible = $61.5\%$
- F-Measure: $F = P \times R = 49.2\%$

**Evaluated on Head word Arg**
- Precision: $P = 9$ correct / $10$ labeled = $90.0\%$
- Recall: $R = 9$ correct / $13$ possible = $69.2\%$
- F-Measure: $F = P \times R = 62.3\%$
Applications

- Question & answer systems

The police officer detained the suspect at the scene of the crime

- ARG0
- V
- ARG2
- AM-loc

Who did what to whom at where?
Multilingual Applications

- Machine translation generation/evaluation

**src:**

Democratic party blame Bush point he create le new evil shaft

**ref:**

democrats criticized bush for creating a new axis of evil

**MT1:**

the democratic party criticized bush that he created a new evil axis

**MT2:**

the democratic party group george w. bush that he created a new axis of evil

**BLEU score**

0.0 4-tuple BLEU score

0.32 4-tuple BLEU score

**SRL match**

Much better ref SRL match

Good src SRL match

Missing verb, ungrammatical sentence
Multilingual Applications

- Identifying/recovering implicit arguments across language
  - Chinese dropped pronoun

They were afraid that once they go to the city, they will not be able to assimilate into the schools and things like that. etc.
SRL Training Data, Parsers

Training Data (Treebank and PropBank):
- LDC
  [http://www.ldc.upenn.edu/](http://www.ldc.upenn.edu/)

Parsers:
- Collins Parser
- Charniak Parser
  [http://cs.brown.edu/people/ec/#software](http://cs.brown.edu/people/ec/#software)
- Berkeley Parser
- Stanford Parser (includes dependency conversion tools)
- ClearNLP (dependency parser and labeler, Apache license)
  [https://code.google.com/p/clearnlp/](https://code.google.com/p/clearnlp/)
Some SRL systems on the Web

Constituent Based SRL:

- **ASSERT**
  - one of the top CoNLL-2005 system, extended to C-ASSERT for Chinese SRL)
  - [http://cemantix.org/software/assert.html](http://cemantix.org/software/assert.html)
- **Senna (GPL license)**
  - fast implementation in C
- **SwiRL**
  - one of the top CoNLL-2005 system
  - [http://www.surdeanu.info/mihai/swirl/](http://www.surdeanu.info/mihai/swirl/)
- **UIUC SRL Demo**
  - based on the top CoNLL-2005 system w/ ILP argument set inference
  - [http://cogcomp.cs.illinois.edu/demo/srl/](http://cogcomp.cs.illinois.edu/demo/srl/)

Dependency Based SRL:

- **ClearNLP (dependency parser and labeler, Apache license)**
  - state-of-the-art dependency based SRL (comparable to top CoNLL-2008 system)
  - models for OntoNotes and medical data, actively maintained
  - [https://code.google.com/p/clearnlp/](https://code.google.com/p/clearnlp/)
References


J. D. Choi, M. Palmer, and Ni Xue. *Using parallel propbanks to enhance word-alignments*. ACL-LAW, 2009


