Semantic Role Labeling Tutorial Part 2
Neural Methods for Semantic Role Labeling

Diego Marcheggiani, Michael Roth, Ivan Titov, Benjamin Van Durme

University of Amsterdam
University of Edinburgh

EMNLP 2017
Copenhagen
Outline: the fall and rise of syntax in SRL

- Early SRL methods
- Symbolic approaches + Neural networks (syntax-aware models)
- Syntax-agnostic neural methods
- Syntax-aware neural methods
Recent papers which involve neural networks and SRL

- English language
- Skip predicate identification and disambiguation methods
- Focus on labeling of semantic roles
- PropBank [Palmer et al. 2005]
  - CoNLL 2005 dataset (span-based SRL)
  - CoNLL 2009 dataset (dependency-based SRL)
- F1 measure for role labeling and predicate disambiguation
Outline: the fall and rise of syntax in SRL

- **Early SRL methods**
  - Symbolic approaches + Neural networks (syntax-aware models)
  - Syntax-agnostic neural methods
  - Syntax-aware neural methods
General SRL Pipeline

Given a predicate:

Sequa makes and repairs jet engines
Given a predicate:
- Argument identification

Sequa makes and repairs jet engines
General SRL Pipeline

- Given a predicate:
  - Argument identification
  - Role labeling
General SRL Pipeline

- **Given a predicate:**
  - Argument identification
  - Role labeling
  - Global and/or constrained inference

```
Sequa          makes          and           repairs             jet          engines
```
Argument identification

- Hand-crafted rules on the full syntactic tree [Xue and Palmer, 2004]
- Binary classifier [Pradhan et al., 2005; Toutanova et al., 2008]
- Both [Punyakanok et al., 2008]
Role labeling

- Labeling is performed using a classifier (SVM, logistic regression)
- For each argument we get a label distribution
- Argmax over roles will result in a local assignment
- No guarantee the labeling is well formed
  - overlapping arguments, duplicate core roles, etc.
Inference

- Enforce linguistic and structural constraint (e.g., no overlaps, discontinuous arguments, reference arguments, …)
- Viterbi decoding (k-best list with constraints) [Täckström et al., 2015]
- Dynamic programming [Täckström et al., 2015; Toutanova et al., 2008]
- Integer linear programming [Punyakanok et al., 2008]
- Re-ranking [Toutanova et al., 2008; Björkelund et al., 2009]
Early symbolic models

- 3 steps pipeline
- Massive feature engineering
  - argument identification
  - role labeling
  - re-ranking
- Most of the features are syntactic [Gildea and Jurafsky, 2002]
Outline: the fall and rise of syntax in SRL

- Early SRL framework
- **Symbolic approaches + Neural networks (syntax-aware models)**
- Syntax-agnostic neural methods
- Syntax-Aware neural methods
Rule based argument identification
  as in [Xue and Palmer, 2004] but for dependency parsing

Neural network for local role labeling

Global structural inference based on dynamic programming
  [Täckström et al., 2015]
Fitzgerald et al., 2015: Architecture

Hidden layer

Embedding layer

Candidate argument features
Fitzgerald et al., 2015: Architecture

Hidden layer

Embedding layer

Candidate argument features
Fitzgerald et al., 2015: Architecture

Embedding layer

Hidden layer

Candidate argument features

Predicate embedding

Role embedding
Fitzgerald et al., 2015: Architecture

Hidden layer

Embedding layer

Candidate argument features

Predicate embedding

Role embedding

Predicate-specific role representation

Nonlinear transform
Fitzgerald et al., 2015: Architecture

Candidate argument features

Predicate embedding

Role embedding

Nonlinear transform

Predicate-specific role representation

Compatibility score

Dot product

Embedding layer

Hidden layer

\[ g_{NN}(s, r, \theta) \]
Fitzgerald et al., 2015: Span-based SRL results

CoNLL 2005 test

- Täckström et al. (2015) (global): 79.9
- Toutanova et al. (2008) (global): 79.7
- Surdenau et al. (2007) (global): 77.2
- FitzGerald et al. (2015) (global): 79.4
Fitzgerald et al., 2015: Span-based SRL results

Täckström et al. (2015) (global) 71.3
Toutanova et al. (2008) (global) 67.8
Surdenau et al. (2007) (global) 67.7
FitzGerald et al. (2015) (global) 71.2

CoNLL 2005 out of domain
Fitzgerald et al., 2015: Dependency-based SRL results

CoNLL 2009 test

- Lei et al. (2016) (local): 86.6
- Björkelund et al. (2010) (global): 86.9
- Täckström et al. (2015) (global): 87.3
- FitzGerald et al. (2015) (global): 87.3
Fitzgerald et al., 2015: Dependency-based SRL results

CoNLL 2009 out of domain

- Lei et al. (2016) (local) - 75.6
- Björkelund et al. (2010) (global) - 75.7
- Roth and Woodsend (2014) (global) - 75.9
- FitzGerald et al. (2015) (global) - 75.2
Predicate-role composition
- Predicate-specific role representation
- Learning distributed predicate representation across different formalisms
- State of the art on FrameNet dataset

Feature embeddings
- Use “simple” span features
- Let the network figure out how to compose them
- Reduced feature engineering
Dependency-based SRL

Neural network with dependency path embeddings as local classifier
  - Argument identification
  - Role labeling

Global re-ranking of k-best local assignments
Syntactic paths between predicates and arguments are an important feature
It may be extremely sparse
Creating a distributed representation can solve the problem
Use LSTM [Hochreiter and Schmidhuber, 1995] to encode paths
Sequa makes and repairs jet engines.
Roth and Lapata, 2016: Dependency path embeddings example
Roth and Lapata, 2016: Architecture

Candidate argument features
### Roth and Lapata, 2016: Dependency-based SRL results

**Bar Chart:** CoNLL 2009 test

<table>
<thead>
<tr>
<th>Researcher</th>
<th>Year</th>
<th>Methodology</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lei et al. (2016)</td>
<td>2016</td>
<td>Local</td>
<td>86.6</td>
</tr>
<tr>
<td>Björkelund et al. (2010)</td>
<td>2010</td>
<td>Global</td>
<td>86.9</td>
</tr>
<tr>
<td>Täckström et al. (2015)</td>
<td>2015</td>
<td>Global</td>
<td>87.3</td>
</tr>
<tr>
<td>FitzGerald et al. (2015)</td>
<td>2015</td>
<td>Global</td>
<td>87.3</td>
</tr>
<tr>
<td>Roth and Lapata (2016)</td>
<td>2016</td>
<td>Global</td>
<td>87.7</td>
</tr>
</tbody>
</table>
Roth and Lapata, 2016

- Encode syntactic paths with LSTMs
  - Overcome sparsity
- Combination of symbolic features and continuous syntactic paths
Outline: the fall and rise of syntax in SRL

- Early SRL framework
- Symbolic approaches + Neural networks
- Syntax-agnostic neural methods (the fall)
- Syntax-aware neural methods
Syntax-agnostic neural methods

- SRL as a sequence labeling task
Syntax-agnostic neural methods

- SRL as a sequence labeling task
  - Argument identification and role labeling in one step

Sequa makes and repairs jet engines

ARG 0

Sequa

ARG 1

jet engines

B-A0 makes and repair.01 repairs O O O

B-A1 I-A1
Syntax-agnostic neural methods

- General architecture
  - Word encoding
  - Sentence encoding (via LSTM)
  - Decoding
- No use of any kind of treebank syntax (not trivial to encode it)
- Differentiable end-to-end
  - [Collobert et al., (2011)]
Zhou and Xu, 2015: Word encoding

- Pretrained word embedding
Zhou and Xu, 2015: Word encoding

- Pretrained word embedding
- Distance from the predicate
Zhou and Xu, 2015: Word encoding

- Pretrained word embedding
- Distance from the predicate
- Predicate context (for disambiguation)
Zhou and Xu, 2015: Word encoding

- Pretrained word embedding
- Distance from the predicate
- Predicate context (for disambiguation)
- Predicate region mark
Zhou and Xu, 2015: Sentence encoding

- Bidirectional LSTM
  - Forward (left context)
Zhou and Xu, 2015: Sentence encoding

- **Bidirectional LSTM**
  - Forward (left context)
  - Backward (right context)
Zhou and Xu, 2015: Sentence encoding

- **Bidirectional LSTM**
  - Forward (left context)
  - Backward (right context)
  - Snake BiLSTM
Zhou and Xu, 2015: Decoder

- Conditional Random Field
  - [Lafferty et al., 2001]
  - Markov assumption between role labels
Zhou and Xu, 2015: Results

CoNLL 2005 test

<table>
<thead>
<tr>
<th>Study</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Täckström et al. (2015)</td>
<td>79.9</td>
</tr>
<tr>
<td>Toutanova et al. (2008)</td>
<td>79.7</td>
</tr>
<tr>
<td>Surdenau et al. (2007)</td>
<td>77.2</td>
</tr>
<tr>
<td>FitzGerald et al. (2015)</td>
<td>79.4</td>
</tr>
<tr>
<td>Zhou and Xu (2015) (CRF)</td>
<td>82.8</td>
</tr>
</tbody>
</table>
Zhou and Xu, 2015: Results

CoNLL 2005 out of domain

- Täckström et al. (2015) (global): 71.3
- Toutanova et al. (2008) (global): 67.8
- Surdeanu et al. (2007) (global): 67.7
- FitzGerald et al. (2015) (global): 71.2
- Zhou and Xu (2015) (CRF): 69.4
Zhou and Xu, 2015: Analysis
- No syntax
- Minimal word representation
- Sentence encoding with “Snake” BiLSTM
Pretrained word embedding
Predicate flag
He et al., 2017: Sentence encoding

- “Snake” Bi-LSTM
- Highway connections [Srivastava et al., 2015]
- Recurrent dropout [Gal and Ghahramani, 2016]
He et al., 2017: Highway connections [Srivastava et al., 2015]
He et al., 2017: Highway connections [Srivastava et al., 2015]

\[ r_{l,t} = \sigma(W^l(h_{l,t-1} \circ h_{l-1,t})) \]
He et al., 2017: Highway connections [Srivastava et al., 2015]

\[ r_{l,t} = \sigma(W^l(h_{l,t-1} \circ h_{l-1,t})) \]

\[ h_{l,t} = r_{l,t} \odot h'_{l,t} + (1 - r_{l,t}) \odot V h_{l-1,t} \]
He et al., 2017: Recurrent dropout [Gal and Ghahramani, 2016]

\[ \tilde{h}_{l,t} = z_l \odot h_{l,t} \]

Gated hidden state

Random binary mask

4 layers highway BiLSTM

word representation

Lane disputed those estimates
He et al., 2017: Recurrent dropout [Gal and Ghahramani, 2016]

\[ \tilde{h}_{l,t} = z_l \odot h_{l,t} \]

Gated hidden state

Random binary mask

4 layers highway BiLSTM

word representation

Lane disputed those estimates
He et al., 2017: Decoding

- A* decoding algorithm
  - BIO constraint
  - Continuation constraint
  - Uniqueness core roles
  - Reference constraint
  - Syntactic constraint

A1

Constrained A* Decoding

K layers highway BiLSTM

word representation

Lane disputed those estimates
He et al., 2017: Results

CoNLL 2005 test

<table>
<thead>
<tr>
<th>Study</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Täckström et al. (2015)</td>
<td>79.9</td>
</tr>
<tr>
<td>Toutanova et al. (2008)</td>
<td>79.7</td>
</tr>
<tr>
<td>Surdenau et al. (2007)</td>
<td>77.2</td>
</tr>
<tr>
<td>FitzGerald et al. (2015)</td>
<td>79.4</td>
</tr>
<tr>
<td>Zhou and Xu (2015) CRF</td>
<td>82.8</td>
</tr>
<tr>
<td>He et al. (2017) (global)</td>
<td>83.1</td>
</tr>
</tbody>
</table>
He et al., 2017: Results

CoNLL 2005 out of domain

- Höökström et al. (2015) (global): 71.3
- Toutanova et al. (2008) (global): 67.8
- Surdenau et al. (2007) (global): 67.7
- FitzGerald et al. (2015) (global): 71.2
- Zhou and Xu (2015) (CRF): 69.4
- He et al. (2017) (global): 72.1
He et al., 2017: Analysis syntactic constraints

![Graph showing the relationship between F1 score and penalty C for Gold, Choe, and Charniak models. The F1 score increases as the penalty C increases, with Gold showing the highest F1 score.]
He et al., 2017

- No syntax
- Super minimal word representation
- Exploit at best the representational power of NN
  - Highway networks
  - Recurrent dropout
Marcheggiani et al., 2017

- Dependency-based SRL
- Shallow syntactic information (POS tags)
- Intuitions from syntactic dependency parsing
- Local classifier
Marcheggiani et al., 2017: Word encoding

- Pretrained word embedding
- Randomly initialized embedding
- Randomly initialized embedding of POS tags
- Embeddings of the predicate lemmas
- Predicate flag
Marcheggiani et al., 2017: Sentence encoding

- **Standard (non-snake) BI-LSTM**
  - Forward LSTM encode left context
  - Backward LSTM encode right context
  - Forw. and Backw. states are concatenated

Diagram:

- Lane: disputed
- Those estimates
- K layers BiLSTM
- Word representation
Marcheggiani et al., 2017: Decoding

\[ p(r|t_i, t_p, l) \propto \exp(W_{l,r}(t_i \circ t_p)) \]

Concatenation of argument and predicate states [Kiperwasser and Goldberg, 2016]
Marcheggiani et al., 2017: Decoding

\[ p(r|t_i, t_p, l) \propto \exp(W_{l,r}(t_i \circ t_p)) \]

Concatenation of argument and predicate states
[Kiperwasser and Goldberg, 2016]

\[ W_{l,r} = ReLU(U(q_l \circ q_r)) \]

Predicate lemma embedding
Role embedding

Lane disputed those estimates

Fitzgerald et al. 2015

K layers BiLSTM

A1

Local classifier

0 1 0 0

word representation
Marcheggiani et al., 2017: Results

CoNLL 2009 test

- Lei et al. (2016) (local): 86.6%
- Björkelund et al. (2010) (global): 86.9%
- Täckström et al. (2015) (global): 87.3%
- FitzGerald et al. (2015) (global): 87.3%
- Roth and Lapata (2016) (global): 87.7%
- Marcheggiani et al. (2017) (local): 87.7%
Marcheggiani et al., 2017: Results

CoNLL 2009 out of domain

<table>
<thead>
<tr>
<th>Study</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lei et al. (2016) (local)</td>
<td>75.6</td>
</tr>
<tr>
<td>Björkelund et al. (2010) (global)</td>
<td>75.7</td>
</tr>
<tr>
<td>Roth and Woodsend (2014) (global)</td>
<td>75.9</td>
</tr>
<tr>
<td>FitzGerald et al. (2015) (global)</td>
<td>75.2</td>
</tr>
<tr>
<td>Roth and Lapata (2016) (global)</td>
<td>76.1</td>
</tr>
<tr>
<td>Marcheggiani et al. (2017) (local)</td>
<td>77.7</td>
</tr>
</tbody>
</table>
Marcheggiani et al., 2017: Ablation study

CoNLL 2009 development

- Full model: 86.6
- w/o POS tags: 85.9
Little bit of syntax (POS tags)
More sophisticated word representation
Fast local classifier conditioned on predicate representation
Outline: the fall and rise of syntax in SRL

- Early SRL framework
- Symbolic approaches + Neural networks
- Syntax-agnostic neural methods
- Syntax-aware neural methods (syntax strikes back!)
Is syntax important for semantics?

- POS tags are beneficial [Marcheggiani et al., 2017]
- Gold syntax is beneficial (but hard to encode) [He et al., 2017]
- Encoding syntax with Graph Convolutional Networks
  - [Marcheggiani and Titov, 2017]
Marcheggiani and Titov, 2017

- **Word encoding** [Marcheggiani et al., 2017]
- **Sentence encoding with BiLSTM** [Marcheggiani et al., 2017]
- **Syntax encoding with Graph Convolutional Networks (GCN)**
  - [Kipf and Welling, 2016]
  - Each word is enriched with the representation of its syntactic neighborhood
- **Local classifier** [Marcheggiani et al., 2017]
Marcheggiani and Titov, 2017: Syntactic GCN example

Lane disputed those estimates

SBJ OBJ NMOD
Marcheggiani and Titov, 2017: Syntactic GCN example

```
Lane 
\times W_{self}^{(1)}
\Rightarrow ReLU(\Sigma \cdot)
```

```
\times W_{self}^{(1)}
\Rightarrow ReLU(\Sigma \cdot)
```

```
\times W_{self}^{(1)}
\Rightarrow ReLU(\Sigma \cdot)
```

```
\times W_{self}^{(1)}
\Rightarrow ReLU(\Sigma \cdot)
```

Lane ----> disputed ----> those ----> estimates

SBJ ----> OBJ

NMOD
Marcheggiani and Titov, 2017: Syntactic GCN example

Lane disputed those estimates

ReLU(Σ·) × W(1)self
ReLU(Σ·) × W(1)self
ReLU(Σ·) × W(1)self
ReLU(Σ·) × W(1)self

SBJ OBJ

NMOD
Marcheggiani and Titov, 2017: Syntactic GCN example

ReLU(Σ·)

×W(1)_{self}

×W(1)_{subj}

×W(1)_{obj}

Lane

disputed

ReLU(Σ·)

×W(1)_{self}

×W(1)_{subj}

×W(1)_{obj}

those

estimates

ReLU(Σ·)

×W(1)_{self}

×W(1)_{nmod}

×W(1)_{nmod}

SBJ

OBJ

NMOD
Lane disputed those estimates

ReLU(\Sigma \cdot)

ReLU(\Sigma \cdot)

ReLU(\Sigma \cdot)

ReLU(\Sigma \cdot)

\times W(1)_{\text{self}}

\times W(1)_{\text{subj}}

\times W(1)_{\text{subj}}

\times W(1)_{\text{self}}

\times W(1)_{\text{obj}}

\times W(1)_{\text{obj}}

\times W(1)_{\text{nmod}}

\times W(1)_{\text{nmod}}

\times W(1)_{\text{nmod}}

SBJ

OBJ

NMOD
Marcheggiani and Titov, 2017: Syntactic GCN example
Marcheggiani and Titov, 2017: Syntactic GCN example

Stacking GCNs widens the syntactic neighborhood
Marcheggiani and Titov, 2017: Syntactic GCN

\[ h_{v}^{(k+1)} = ReLU \left( \sum_{u \in \mathcal{N}(v)} W_{L(u,v)}^{(k)} h_{u}^{(k)} + b_{L(u,v)}^{(k)} \right) \]

Sum over the syntactic neighborhood

Each node is transformed according to label and direction
Marcheggiani and Titov, 2017: Architecture

- Same architecture of [Marcheggiani et al., 2017]
- Syntactic GCN after BiLSTM encoder
  - Skip connections
  - Longer dependencies are captured
Marcheggiani and Titov, 2017: Results

CoNLL 2009 test

<table>
<thead>
<tr>
<th>Study</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lei et al. (2016) (local)</td>
<td>86.6</td>
</tr>
<tr>
<td>Björkelund et al. (2010) (global)</td>
<td>86.9</td>
</tr>
<tr>
<td>Täckström et al. (2015) (global)</td>
<td>87.3</td>
</tr>
<tr>
<td>FitzGerald et al. (2015) (global)</td>
<td>87.3</td>
</tr>
<tr>
<td>Roth and Lapata (2016) (global)</td>
<td>87.7</td>
</tr>
<tr>
<td>Marcheggiani et al. (2017) (local)</td>
<td>87.7</td>
</tr>
<tr>
<td>Marcheggiani and Titov (2017) (local)</td>
<td>88</td>
</tr>
</tbody>
</table>
Marcheggiani and Titov, 2017: Results

CoNLL 2009 out of domain

- Lei et al. (2016) (local): 75.6
- Björkelund et al. (2010) (global): 75.7
- Roth and Woodensen (2014) (global): 75.9
- FitzGerald et al. (2015) (global): 75.2
- Roth and Lapata (2016) (global): 76.1
- Marcheggiani et al. (2017) (local): 77.7
- Marcheggiani and Titov (2017) (local): 77.2
Marcheggiani and Titov, 2017: Analysis

CoNLL 2009 development

- No syntax: 82.7
- Syntactic GCN (predicted): 83.3
- Syntactic GCN (gold): 86.4
- Encoding structured prior linguistic knowledge in NN
  - Syntax
  - Semantics
  - Coreference
  - Discourse
- Complement LSTM with skip connections for long dependencies
Conclusion

- We can live without syntax (out of domain)
Conclusion

- We can live without syntax (out of domain)
- But life with syntax is better
Conclusion

- We can live without syntax (out of domain)
- But life with syntax is better
  - and the better the syntax (parsers) the better our semantic role labeler
Conclusion

- We can live without syntax (out of domain)
- But life with syntax is better
  - and the better the syntax (parsers) the better our semantic role labeler
- What’s the (present) future?
We can live without syntax (out of domain)

But life with syntax is better
  and the better the syntax (parsers) the better our semantic role labeler

What’s the (present) future?
  Multi-task learning
  Swayamdipta et al. (2017) frame-semantic parsing + syntax
  Peng et al. (2017) multi-task on different semantic formalisms
Conclusion

- We can live without syntax (out of domain)
- But life with syntax is better
  - and the better the syntax (parsers) the better our semantic role labeler

- What’s the (present) future?
  - Multi-task learning
  - Swayamdipta et al. (2017) frame-semantic parsing + syntax
  - Peng et al. (2017) multi-task on different semantic formalisms

- Neural networks work (I kid you not) …
Conclusion

- We can live without syntax (out of domain)
- But life with syntax is better
  - and the better the syntax (parsers) the better our semantic role labeler

- What’s the (present) future?
  - Multi-task learning
  - Swayamdipta et al. (2017) frame-semantic parsing + syntax
  - Peng et al. (2017) multi-task on different semantic formalisms

- Neural networks work (I kid you not) …
- … but we do have (a lot of) linguistic prior knowledge…
Conclusion

- We can live without (treebank) syntax (out of domain)
- But life with syntax is better
  - and the better the syntax (parsers) the better our semantic role labeler

- What’s the (present) future?
  - Multi-task learning
  - Swayamdipta et al. (2017) frame-semantic parsing + syntax
  - Peng et al. (2017) multi-task on different semantic formalisms

- Neural networks work (I kid you not) …
- … but we do have (a lot of) linguistic prior knowledge…
- … and it is time to use it again.
References

References


References

- Michael Roth and Mirella Lapata. 2016. Neural semantic role labeling with dependency path embeddings. In *Proceedings of ACL*.


References

References