Delexicalized Parsing

Daniel Zeman, Rudolf Rosa

April 3, 2020
What if we feed the parser with tags instead of words?

- Ændringer i listen i bilaget offentliggøres og meddeles på samme måde.
- NNS IN NN IN NN VB CC VB IN DT NN
- NNS IN NN MD VB CC VB IN DT NN
- Förändringar i förteckningen skall offentliggöras och meddelas på samma sätt.
• What if we feed the parser with tags instead of words?

  • Ændringer i listen i bilaget offentliggøres og meddeles på samme måde.
  • (((NNS (IN NN (IN NN)))) ((VB CC VB) (IN (DT NN)))))
  • (((NNS (IN NN)) ((MD (VB CC VB)) (IN (DT NN)))))
  • Förändringar i förteckningen skall offentliggöras och meddelas på samma sätt.
Danish – Swedish Setup

  - In *IJCNLP 2008 Workshop on NLP for Less Privileged Languages*, pp. 35–42, Hyderabad, India

- CoNLL 2006 treebanks (dependencies)
  - Danish Dependency Treebank
  - Swedish Talbanken05

- Two constituency parsers:
  - "Charniak"
  - "Brown" (Charniak N-best parser + Johnson reranker)

- Other resources
  - JRC-Acquis parallel corpus
  - Hajič tagger for Swedish (PAROLE tagset)
Danish – Swedish Setup

  - In *IJCNLP 2008 Workshop on NLP for Less Privileged Languages*, pp. 35–42, Hyderabad, India

- CoNLL 2006 treebanks (*dependencies*)
  - Danish Dependency Treebank
  - Swedish Talbanken05

- Two constituency parsers:
  - “Charniak”
  - “Brown” (Charniak N-best parser + Johnson reranker)

- Other resources
  - (JRC-Acquis *parallel* corpus)
  - Hajič tagger for Swedish (*PAROLE* tagset)
Danish – Swedish Setup

  - In *IJCNLP 2008 Workshop on NLP for Less Privileged Languages*, pp. 35–42, Hyderabad, India

- CoNLL 2006 treebanks *(dependencies)*
  - Danish Dependency Treebank
  - Swedish Talbanken05

- Two constituency parsers:
  - “Charniak”
  - “Brown” (Charniak N-best parser + Johnson reranker)

- Other resources
  - Hajič tagger for Swedish *(PAROLE tagset)*
Treebank Normalization

Danish
- DET governs ADJ
  ADJ governs NOUN

Swedish
- NOUN governs both DET and ADJ
Treebank Normalization

Danish

- DET governs ADJ
  ADJ governs NOUN
- NUM governs NOUN

Swedish

- NOUN governs both DET and ADJ
- NOUN governs NUM
Treebank Normalization

Danish

- DET governs ADJ
  ADJ governs NOUN
- NUM governs NOUN
- GEN governs NOM
  Ruslands vej
  Russia’s way

Swedish

- NOUN governs both DET and ADJ
- NOUN governs NUM
- NOM governs GEN
  års inkomster
  year’s income
Treebank Normalization

Danish

- DET governs ADJ
  ADJ governs NOUN
- NUM governs NOUN
- GEN governs NOM
  *Ruslands vej*
  *Russia’s way*
- COORD: last member on conjunction, everything else on first member

Swedish

- NOUN governs both DET and ADJ
- NOUN governs NUM
- NOM governs GEN
  *års inkomster*
  *year’s income*
- COORD: member on previous member, commas and conjs on next member
Treebank Preparation

- Transform Danish to Swedish tree style
  - A few heuristics
  - Only for evaluation! Not needed in real world.
Treebank Preparation

- Transform Danish to Swedish tree style
  - A few heuristics
  - Only for evaluation! Not needed in real world.

- Convert dependencies to constituents
  - Flattest possible structure
Treebank Preparation

- Transform Danish to Swedish tree style
  - A few heuristics
  - Only for evaluation! Not needed in real world.
- Convert dependencies to constituents
  - Flattest possible structure
- DA/SV tagset converted to Penn Treebank tags
Treebank Preparation

- Transform Danish to Swedish tree style
  - A few heuristics
  - Only for evaluation! Not needed in real world.
- Convert dependencies to constituents
  - Flattest possible structure
- DA/SV tagset converted to Penn Treebank tags
- Nonterminal labels:
  - derived from POS tags
  - then translated to the Penn set of nonterminals
- Make the parser feel it works with the Penn Treebank
  - (Although it could have been configured to use other sets of labels.)
Unlabeled F Scores

- da-da lexicalized: Charniak = 78.16, Brown = 78.24
  - (CoNLL train 94K words, test 5852 words)
- sv-sv lexicalized: Charniak = 77.81, Brown = 78.74
  - (CoNLL train 191K words, test 5656 words)
- da-sv lexicalized: Charniak = 43.28, Brown = 41.84
  - (no morphology tweaking)
- da-da delexicalized: Charniak = 79.62, Brown = 80.20 (!)
  - (hybrid sv-da Hajič-like tagset = "words", Penn POS = "tags")
- sv-sv delexicalized: Charniak = 76.07, Brown = 77.01
- da-sv delexicalized: Charniak = 65.50, Brown = 66.40
Unlabeled F Scores

- **da-da lexicalized**: Charniak = 78.16, Brown = 78.24
  - (CoNLL train 94K words, test 5852 words)
- **sv-sv lexicalized**: Charniak = 77.81, Brown = 78.74
  - (CoNLL train 191K words, test 5656 words)
- **da-sv lexicalized**: Charniak = 43.28, Brown = 41.84
  - (no morphology tweaking)

Delexicalized Parsing
Unlabeled F Scores

- **da-da lexicalized**: Charniak = 78.16, Brown = 78.24
  - (CoNLL train 94K words, test 5852 words)

- **sv-sv lexicalized**: Charniak = 77.81, Brown = 78.74
  - (CoNLL train 191K words, test 5656 words)

- **da-sv lexicalized**: Charniak = 43.28, Brown = 41.84
  - (no morphology tweaking)

- **da-da delexicalized**: Charniak = 79.62, Brown = 80.20 (!)
  - (hybrid sv-da Hajič-like tagset = “words”, Penn POS = “tags”)

- **sv-sv delexicalized**: Charniak = 76.07, Brown = 77.01
  - da-sv delexicalized: Charniak = 65.50, Brown = 66.40
Unlabeled F Scores

- da-da lexicalized: Charniak = 78.16, Brown = 78.24
  - (CoNLL train 94K words, test 5852 words)
- sv-sv lexicalized: Charniak = 77.81, Brown = 78.74
  - (CoNLL train 191K words, test 5656 words)
- da-sv lexicalized: Charniak = 43.28, Brown = 41.84
  - (no morphology tweaking)
- da-da delexicalized: Charniak = 79.62, Brown = 80.20 (!)
  - (hybrid sv-da Hajič-like tagset = “words”, Penn POS = “tags”)
- sv-sv delexicalized: Charniak = 76.07, Brown = 77.01
Unlabeled F Scores

- da-da lexicalized: Charniak = 78.16, Brown = 78.24
  - (CoNLL train 94K words, test 5852 words)
- sv-sv lexicalized: Charniak = 77.81, Brown = 78.74
  - (CoNLL train 191K words, test 5656 words)
- da-sv lexicalized: Charniak = 43.28, Brown = 41.84
  - (no morphology tweaking)
- da-da delexicalized: Charniak = 79.62, Brown = 80.20 (!)
  - (hybrid sv-da Hajič-like tagset = “words”, Penn POS = “tags”)
- sv-sv delexicalized: Charniak = 76.07, Brown = 77.01
- da-sv delexicalized: Charniak = 65.50, Brown = 66.40
How Big Swedish Treebank Yields Similar Results?

Unlabeled $F_1$-score

Graph showing the relationship between training sentences and unlabeled $F_1$-score. The graph indicates that as the number of training sentences increases, the unlabeled $F_1$-score also increases. A specific point at 1546 sentences has a score of 66.40.
Ryan McDonald, Slav Petrov, Keith Hall (2011). Multi-Source Transfer of Delexicalized Dependency Parsers

Delexicalized Dependency Parsing

- Ryan McDonald, Slav Petrov, Keith Hall (2011). Multi-Source Transfer of Delexicalized Dependency Parsers

- Transition-based parser, arc-eager algorithm, averaged perceptron, pseudo-projective technique on non-projective treebanks
Delexicalized Dependency Parsing

- Ryan McDonald, Slav Petrov, Keith Hall (2011). Multi-Source Transfer of Delexicalized Dependency Parsers

- Transition-based parser, arc-eager algorithm, averaged perceptron, pseudo-projective technique on non-projective treebanks

- Google universal POS tags, two scenarios:
  - Gold-standard (just converted)
  - Projected across parallel corpus from English

"Danish is the worst possible source language for Swedish."
Delexicalized Dependency Parsing

- Ryan McDonald, Slav Petrov, Keith Hall (2011). Multi-Source Transfer of Delexicalized Dependency Parsers

- Transition-based parser, arc-eager algorithm, averaged perceptron, pseudo-projective technique on non-projective treebanks

- Google universal POS tags, two scenarios:
  - Gold-standard (just converted)
  - Projected across parallel corpus from English

- UAS (unlabeled attachment score)
- No tree structure harmonization
Delexicalized Dependency Parsing

- Ryan McDonald, Slav Petrov, Keith Hall (2011). Multi-Source Transfer of Delexicalized Dependency Parsers

- Transition-based parser, arc-eager algorithm, averaged perceptron, pseudo-projective technique on non-projective treebanks

- Google universal POS tags, two scenarios:
  - Gold-standard (just converted)
  - Projected across parallel corpus from English

- UAS (unlabeled attachment score)

- No tree structure harmonization
  - “Danish is the worst possible source language for Swedish.”
### Multi-Source Transfer (McDonald et al., 2011)

#### Delexicalized Parsing

<table>
<thead>
<tr>
<th>Target Test Language</th>
<th>da</th>
<th>de</th>
<th>el</th>
<th>en</th>
<th>es</th>
<th>it</th>
<th>nl</th>
<th>pt</th>
<th>sv</th>
</tr>
</thead>
<tbody>
<tr>
<td>da</td>
<td>79.2</td>
<td>45.2</td>
<td>44.0</td>
<td>45.9</td>
<td>45.0</td>
<td>48.6</td>
<td>46.1</td>
<td>48.1</td>
<td>47.8</td>
</tr>
<tr>
<td>de</td>
<td>34.3</td>
<td>83.9</td>
<td>53.2</td>
<td>47.2</td>
<td>45.8</td>
<td>53.4</td>
<td>55.8</td>
<td>55.5</td>
<td>46.2</td>
</tr>
<tr>
<td>el</td>
<td>33.3</td>
<td>52.5</td>
<td>77.5</td>
<td>63.9</td>
<td>41.6</td>
<td>59.3</td>
<td>57.3</td>
<td>58.6</td>
<td>47.5</td>
</tr>
<tr>
<td>en</td>
<td>34.4</td>
<td>37.9</td>
<td>45.7</td>
<td>82.5</td>
<td>28.5</td>
<td>38.6</td>
<td>43.7</td>
<td>42.3</td>
<td>43.7</td>
</tr>
<tr>
<td>es</td>
<td>38.1</td>
<td>49.4</td>
<td>57.3</td>
<td>53.3</td>
<td>79.7</td>
<td>68.4</td>
<td>51.2</td>
<td>66.7</td>
<td>41.4</td>
</tr>
<tr>
<td>it</td>
<td>44.8</td>
<td>56.7</td>
<td>66.8</td>
<td>57.7</td>
<td>64.7</td>
<td>79.3</td>
<td>57.6</td>
<td>69.1</td>
<td>50.9</td>
</tr>
<tr>
<td>nl</td>
<td>38.7</td>
<td>43.7</td>
<td>62.1</td>
<td>60.8</td>
<td>40.9</td>
<td>50.4</td>
<td>73.6</td>
<td>58.5</td>
<td>44.2</td>
</tr>
<tr>
<td>pt</td>
<td>42.5</td>
<td>52.0</td>
<td>66.6</td>
<td>69.2</td>
<td>68.5</td>
<td>74.7</td>
<td>67.1</td>
<td>84.6</td>
<td>52.1</td>
</tr>
<tr>
<td>sv</td>
<td>44.5</td>
<td>57.0</td>
<td>57.8</td>
<td>58.3</td>
<td>46.3</td>
<td>53.4</td>
<td>54.5</td>
<td>66.8</td>
<td>84.8</td>
</tr>
</tbody>
</table>

*Note: The table shows the accuracy scores for different source training languages and target test languages.*
Single-Source, Harmonized (DZ, summer 2015)

- Malt Parser, stack-lazy algorithm (nonprojective)
  - Same algorithm for all, no optimization
  - Same selection of training features for all treebanks

- Trained on the first 1000 sentences only
- Tested on the whole test set
- Default score: UAS (unlabeled attachment)
- Only harmonized data used (HamleDT 3.0 = UD v1 style)
- Single source language for every target
Delexicalized Dependency Parsing with Harmonized Data
Who Helps Whom?

- Czech (62.44) ⇐ Croatian (63.27), Slovenian (62.87)
- Slovak (59.47) ⇐ Croatian (60.28), Slovenian (59.32)
- Polish (77.92) ⇐ Croatian (66.42), Slovenian (64.31)
- Russian (66.86) ⇐ Croatian (57.35), Slovak (55.01)
- Croatian (75.52) ⇐ Slovenian (58.96), Polish (55.42)
- Slovenian (76.17) ⇐ Croatian (62.92), Finnish (59.79)
- Bulgarian (78.44) ⇐ Croatian (74.39), Slovenian (71.52)
Who Helps Whom?

- Catalan (75.28) ⇐ Italian (71.07), French (68.30)
- Italian (76.66) ⇐ French (70.37), Catalan (68.66)
- French (69.93) ⇐ Spanish (64.28), Italian (63.33)
- Spanish (67.76) ⇐ French (67.61), Catalan (64.54)
- Portuguese (69.89) ⇐ Italian (69.48), French (66.12)
- Romanian (79.74) ⇐ Croatian (67.01), Latin (66.75)
Who Helps Whom?

- Swedish (75.73) ⇐ Danish (66.17), English (65.41)
- Danish (75.19) ⇐ Swedish (59.23), Croatian (56.89)
- English (72.68) ⇐ German (57.95), French (56.70)
- German (67.04) ⇐ Croatian (58.68), Swedish (57.48)
- Dutch (60.76) ⇐ Hungarian (41.90), Finnish (37.89)
How Big Swedish Treebank Yields Similar Results as Delex from Danish?

66.17
(delex)
~ 75
sentences
Multiple Source Treebanks

- So far: select one source at a time
  - How to select the best possible source?

  - Alternative 1: train on all sources concatenated
    - Possibly with "weights" – take only part of a treebank, or take multiple copies of a treebank, or omit some treebanks
  - Alternative 2: train on each source separately, then vote
    - Separate voting about every node's incoming edge
      - Weights – how much do we trust each source?

  The result should be a tree!

Chu-Liu-Edmonds MST algorithm, as in graph-based parsing.
Multiple Source Treebanks

- So far: select one source at a time
  - How to select the best possible source?

- Alternative 1: train on all sources concatenated
  - Possibly with “weights” – take only part of a treebank, or take multiple copies of a treebank, or omit some treebanks
Multiple Source Treebanks

- So far: select one source at a time
  - How to select the best possible source?

- Alternative 1: train on all sources concatenated
  - Possibly with “weights” – take only part of a treebank, or take multiple copies of a treebank, or omit some treebanks

- Alternative 2: train on each source separately, then vote
  - Separate voting about every node’s incoming edge
  - Weights – how much do we trust each source?
Multiple Source Treebanks

- So far: select one source at a time
  - How to select the best possible source?

- Alternative 1: train on all sources concatenated
  - Possibly with “weights” – take only part of a treebank, or take multiple copies of a treebank, or omit some treebanks

- Alternative 2: train on each source separately, then vote
  - Separate voting about every node’s incoming edge
  - Weights – how much do we trust each source?
  - The result should be a tree!
    - Chu-Liu-Edmonds MST algorithm, as in graph-based parsing
Multiple Source Treebanks

- So far: select one source at a time
  - How to select the best possible source?

- Alternative 1: train on all sources concatenated
  - Possibly with “weights” – take only part of a treebank, or take multiple copies of a treebank, or omit some treebanks

- Alternative 2: train on each source separately, then vote
  - Separate voting about every node’s incoming edge
  - Weights – how much do we trust each source?
  - The result should be a tree!
    - Chu-Liu-Edmonds MST algorithm, as in graph-based parsing
Syntactic Similarity of Languages

- Observation: We cannot compare trees!
  - In real-world applications, target trees will not be available

- Language genealogy
  - Targeting a Slavic language? Use Slavic sources!

- Problem 1: What if no relative is available? (Buryat…)
- Problem 2: The important characteristics may differ significantly

- English is isolating, rigid word order
- German uses morphology, freer but peculiar word order
- Icelandic has even more morphology

- WALS features (recall the first week)
- Language recognition tool
  - But it relies on orthography!

- cs: Generál přeskupil síly ve Varšavě.
- pl: Generał przegrupował siły w Warszawie.
- ru: Генерал перегруппировал войска в Варшаве.
- en: The general regrouped forces in Warsaw.
Syntactic Similarity of Languages

- Observation: We cannot compare trees!
  - In real-world applications, target trees will not be available
- Language genealogy
  - Targeting a Slavic language? Use Slavic sources!
Observation: We cannot compare trees!
  - In real-world applications, target trees will not be available

Language genealogy
  - Targeting a Slavic language? Use Slavic sources!
  - Problem 1: What if no relative is available? (Buryat...)
Example: CoNLL 2018 Parsing Shared Task

- Low-resource languages:
  - IE: Breton, Faroese, Naija, Upper Sorbian, Armenian, Kurmanji
  - Other: Kazakh, Buryat, Thai
- **Low-resource languages:**
  - IE: Breton, Faroese, Naija, Upper Sorbian, Armenian, Kurmanji
  - Other: Kazakh, Buryat, Thai

- **High(er)-resource languages (selected groups only):**
  - 1 Celtic (Irish)
  - 8 Germanic
  - 10 Slavic
  - 1 Iranian
  - 2 Turkic
Syntactic Similarity of Languages

- Observation: We cannot compare trees!
  - In real-world applications, target trees will not be available
- Language genealogy
  - Targeting a Slavic language? Use Slavic sources!
  - Problem 1: What if no relative is available? (Buryat...)
  - Problem 2: The important characteristics may differ significantly
    - English is isolating, rigid word order
    - German uses morphology, freer but peculiar word order
    - Icelandic has even more morphology
  - WALS features (recall the first week)
  - Language recognition tool
    - But it relies on orthography!
  - cs: Generál přeskupil síly ve Varšavě.
  - pl: Generał przegrupował siły w Warszawie.
  - ru: Генерал перегруппировал войска в Варшаве.
  - en: The general regrouped forces in Warsaw.
Syntactic Similarity of Languages

- Observation: We cannot compare trees!
  - In real-world applications, target trees will not be available
- Language genealogy
  - Targeting a Slavic language? Use Slavic sources!
  - Problem 1: What if no relative is available? (Buryat...)
  - Problem 2: The important characteristics may differ significantly

Delexicalized Parsing
Syntactic Similarity of Languages

- Observation: We cannot compare trees!
  - In real-world applications, target trees will not be available
- Language genealogy
  - Targeting a Slavic language? Use Slavic sources!
  - Problem 1: What if no relative is available? (Buryat...)
  - Problem 2: The important characteristics may differ significantly
    - English is isolating, rigid word order
Observation: We cannot compare trees!
- In real-world applications, target trees will not be available

Language genealogy
- Targeting a Slavic language? Use Slavic sources!
- Problem 1: What if no relative is available? (Buryat...)
- Problem 2: The important characteristics may differ significantly
  - English is isolating, rigid word order
  - German uses morphology, freer but peculiar word order
  - Icelandic has even more morphology
Observation: We cannot compare trees!
- In real-world applications, target trees will not be available

Language genealogy
- Targeting a Slavic language? Use Slavic sources!
- Problem 1: What if no relative is available? (Buryat...)
- Problem 2: The important characteristics may differ significantly
  - English is isolating, rigid word order
  - German uses morphology, freer but peculiar word order
  - Icelandic has even more morphology

WALS features (recall the first week)
Syntactic Similarity of Languages

- Observation: We cannot compare trees!
  - In real-world applications, target trees will not be available

- Language genealogy
  - Targeting a Slavic language? Use Slavic sources!
  - Problem 1: What if no relative is available? (Buryat...)
  - Problem 2: The important characteristics may differ significantly
    - English is isolating, rigid word order
    - German uses morphology, freer but peculiar word order
    - Icelandic has even more morphology

- WALS features (recall the first week)

- Language recognition tool
  - But it relies on orthography!
  - cs: Generál přeskupil síly ve Varšavě.
  - pl: Generał przegrupował siły w Warszawie.
  - ru: Генерал перегруппировал войска в Варшаве.
  - en: The general regrouped forces in Warsaw.
## Measuring Treebank Similarity: POS Tag N-grams

<table>
<thead>
<tr>
<th></th>
<th>en</th>
<th>de</th>
<th>it</th>
<th>cs</th>
</tr>
</thead>
<tbody>
<tr>
<td>DET ADJ NOUN</td>
<td>1.51</td>
<td>1.99</td>
<td>0.96</td>
<td>0.40</td>
</tr>
<tr>
<td>DET NOUN ADJ</td>
<td>0.05</td>
<td>0.26</td>
<td>1.77</td>
<td>0.10</td>
</tr>
<tr>
<td>#sent ADJ NOUN</td>
<td>0.13</td>
<td>0.09</td>
<td>0.02</td>
<td>0.52</td>
</tr>
<tr>
<td>NOUN PUNCT #sent</td>
<td>2.44</td>
<td>1.18</td>
<td>1.41</td>
<td>2.73</td>
</tr>
<tr>
<td>VERB PUNCT #sent</td>
<td>0.48</td>
<td>1.48</td>
<td>0.23</td>
<td>0.58</td>
</tr>
</tbody>
</table>
Kullback-Leibler Divergence

- \( UPOS \) ... universal set of 17 coarse-grained tags (from UD)
- \( UPOS' = UPOS \cup \{\#sent\} \) ... added sentence boundaries
- \((t_{i-2}, t_{i-1}, t_i)\) where \(t_{i-2}, t_{i-1}, t_i \in UPOS'\) ... trigram of tags at positions \(i - 2 \ldots i\) of the corpus
Kullback-Leibler Divergence

- \textit{UPOS} ... universal set of 17 coarse-grained tags (from UD)
- \textit{UPOS}' = \textit{UPOS} \cup \{\#\text{sent}\} ... added sentence boundaries
- \((t_{i-2}, t_{i-1}, t_i)\) where \(t_{i-2}, t_{i-1}, t_i \in \textit{UPOS}'\) ... trigram of tags at positions \(i-2 \ldots i\) of the corpus
- \(P_{\text{Corpus}}(x, y, z) = \frac{\text{count}_{\text{Corpus}}(x, y, z)}{\sum_{a, b, c \in \textit{UPOS}'} \text{count}_{\text{Corpus}}(a, b, c)} = \frac{\text{count}_{\text{Corpus}}(x, y, z)}{|\text{Corpus}|}\)
  - \(x, y, z \in \textit{UPOS}'\)
  - Smoothing: need non-zero probability of every possible trigram
- Smooth: need non-zero probability of every possible trigram

  - In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Short Papers
Kullback-Leibler Divergence

- $UPOS$ ... universal set of 17 coarse-grained tags (from UD)
- $UPOS' = UPOS \cup \{\#sent\}$ ... added sentence boundaries
- $(t_{i-2}, t_{i-1}, t_i)$ where $t_{i-2}, t_{i-1}, t_i \in UPOS'$ ... trigram of tags at positions $i - 2$ ... $i$ of the corpus
- $P_{Corpus}(x, y, z) = \frac{count_{Corpus}(x, y, z)}{\sum_{a, b, c \in UPOS'} count_{Corpus}(a, b, c)} = \frac{count_{Corpus}(x, y, z)}{|Corpus|}$
  - $x, y, z \in UPOS'$
  - Smoothing: need non-zero probability of every possible trigram
- $D_{KL}(P_A||P_B) = \sum_{x, y, z} P_A(x, y, z) \cdot \log \frac{P_A(x, y, z)}{P_B(x, y, z)}$
Kullback-Leibler Divergence

- **UPOS** ... universal set of 17 coarse-grained tags (from UD)
- **UPOS' = UPOS ∪ {#sent}** ... added sentence boundaries
- \((t_{i-2}, t_{i-1}, t_i)\) where \(t_{i-2}, t_{i-1}, t_i \in UPOS'\) ... trigram of tags at positions \(i - 2 \ldots i\) of the corpus
- \(P_{\text{Corpus}}(x, y, z) = \frac{\text{count}_{\text{Corpus}}(x, y, z)}{\sum_{a, b, c \in UPOS'} \text{count}_{\text{Corpus}}(a, b, c)} = \frac{\text{count}_{\text{Corpus}}(x, y, z)}{|\text{Corpus}|}\)
  - \(x, y, z \in UPOS'\)
  - Smoothing: need non-zero probability of every possible trigram
- \(D_{KL}(P_A || P_B) = \sum_{x, y, z} P_A(x, y, z) \cdot \log \frac{P_A(x, y, z)}{P_B(x, y, z)}\)
- \(KL_{cpos^3}(tgt, src) = D_{KL}(P_{tgt} || P_{src})\)
Kullback-Leibler Divergence

- **$UPOS$** ... universal set of 17 coarse-grained tags (from UD)
- **$UPOS' = UPOS \cup \{\#sent\}$** ... added sentence boundaries
- $(t_{i-2}, t_{i-1}, t_i)$ where $t_{i-2}, t_{i-1}, t_i \in UPOS'$ ... trigram of tags at positions $i - 2 \ldots i$ of the corpus

$$P_{\text{Corpus}}(x, y, z) = \frac{\text{count}_{\text{Corpus}}(x, y, z)}{\sum_{a,b,c \in UPOS'} \text{count}_{\text{Corpus}}(a,b,c)} = \frac{\text{count}_{\text{Corpus}}(x, y, z)}{|\text{Corpus}|}$$

- $x, y, z \in UPOS'$
- Smoothing: need non-zero probability of every possible trigram

$$D_{KL}(P_A \| P_B) = \sum_{x,y,z} P_A(x, y, z) \cdot \log \frac{P_A(x, y, z)}{P_B(x, y, z)}$$

$$KL_{cpos^3}(tgt, src) = D_{KL}(P_{tgt} \| P_{src})$$

- Asymmetric: amount of info lost when using the source distribution to approximate the true target distribution

  - In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Short Papers*
How to Make the Languages More Similar?


- Transition-based parsers rely on word order
  - en: *the following question* (features: s0=ADJ, b0=NOUN)
  - fr: *la question suivante* (features: s0=NOUN, b0=ADJ)
How to Make the Languages More Similar?


- Transition-based parsers rely on word order
  - en: *the following question* (features: s0=ADJ, b0=NOUN)
  - fr: *la question suivante* (features: s0=NOUN, b0=ADJ)

- Preprocess training data
  - Reorder words
  - Remove words
How to Make the Languages More Similar?


- Transition-based parsers rely on word order
  - en: the *following* question (features: s0=ADJ, b0=NOUN)
  - fr: la question *suivante* (features: s0=NOUN, b0=ADJ)

- Preprocess training data
  - Reorder words
  - Remove words

- How do we know?
  - Heuristics based on WALS
How to Make the Languages More Similar?


- Transition-based parsers rely on word order
  - en: the following question (features: s0=ADJ, b0=NOUN)
  - fr: la question suivante (features: s0=NOUN, b0=ADJ)

- Preprocess training data
  - Reorder words
  - Remove words

- How do we know?
  - Heuristics based on WALS
  - UPOS language model
    - Generate all permutations in window of 3 words
How to Make the Languages More Similar?


- Transition-based parsers rely on word order
  - en: *the following question* (features: s0=ADJ, b0=NOUN)
  - fr: *la question suivante* (features: s0=NOUN, b0=ADJ)

- Preprocess training data
  - Reorder words
  - Remove words

- How do we know?
  - Heuristics based on WALS
  - UPOS language model
    - Generate all permutations in window of 3 words
    - Discard non-projective subtrees; if nothing left, retain source sequence
How to Make the Languages More Similar?


- Transition-based parsers rely on word order
  - en: *the following question* (features: $s_0$=ADJ, $b_0$=NOUN)
  - fr: *la question suivante* (features: $s_0$=NOUN, $b_0$=ADJ)

- Preprocess training data
  - Reorder words
  - Remove words

- How do we know?
  - Heuristics based on WALS
  - UPOS language model
    - Generate all permutations in window of 3 words
    - Discard non-projective subtrees; if nothing left, retain source sequence
    - Score them by target-language model
    - Take the best permutation