Cross-lingual POS Tagging

Daniel Zeman, Rudolf Rosa

March 27, 2020

EUROPEAN UNION
European Structural and Investment Fund
Operational Programme Research,
Development and Education

Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics

POS Tags Projection across Parallel Corpora

- David Yarowsky, Grace Ngai (2001). Inducing Multilingual POS Taggers and NP Bracketers via Robust Projection across Aligned Corpora
  - In *Proceedings of the Second Meeting of the North American Association for Computational Linguistics (NAACL-2001)*, pp. 200–207, Pittsburgh, PA, USA
  - Source language: English
  - Target languages: French, Chinese

POS Tags Projection across Parallel Corpora

- David Yarowsky, Grace Ngai (2001). Inducing Multilingual POS Taggers and NP Bracketers via Robust Projection across Aligned Corpora
  - In *Proceedings of the Second Meeting of the North American Association for Computational Linguistics (NAACL-2001)*, pp. 200–207, Pittsburgh, PA, USA
  - Source language: English
  - Target languages: French, Chinese
  - Align words using EGYPT/IBM Model 3 (Al-Onaizan et al., 1999)
    - 1:N English-target word alignment
    - or 0:1 or 1:0 for unaligned words

POS Tags Projection across Parallel Corpora

- David Yarowsky, Grace Ngai (2001). Inducing Multilingual POS Taggers and NP Bracketers via Robust Projection across Aligned Corpora
  - In *Proceedings of the Second Meeting of the North American Association for Computational Linguistics (NAACL-2001)*, pp. 200–207, Pittsburgh, PA, USA
  - Source language: English
  - Target languages: French, Chinese
  - Align words using EGYPT/IBM Model 3 (Al-Onaizan et al., 1999)
    - 1:N English-target word alignment
    - or 0:1 or 1:0 for unaligned words
  - Tag the English side with an existing tagger (e.g., Brill, 1995)

POS Tags Projection across Parallel Corpora

- David Yarowsky, Grace Ngai (2001). Inducing Multilingual POS Taggers and NP Bracketers via Robust Projection across Aligned Corpora
  - In *Proceedings of the Second Meeting of the North American Association for Computational Linguistics (NAACL-2001)*, pp. 200–207, Pittsburgh, PA, USA
  - Source language: English
  - Target languages: French, Chinese
  - Align words using EGYPT/IBM Model 3 (Al-Onaizan et al., 1999)
    - 1:N English-target word alignment
    - or 0:1 or 1:0 for unaligned words
  - Tag the English side with an existing tagger (e.g., Brill, 1995)
  - Direct projection across alignment
    - *Laws* → *Les lois*
    - $\texttt{NNS} \rightarrow \texttt{NNS}_a\ \texttt{NNS}_b$

Training on Noisy Data

- Train a tagger on the target side
- Problem: lot of noise!
- Core tags only: first letter, i.e.:
  - **N** ... noun
  - **J** ... adjective
  - **V** ... verb
  - **R** ... adverb
  - **I** ... preposition or subordinating conjunction (?)

Training on Noisy Data

- Train a tagger on the target side
- Problem: lot of noise!
- Core tags only: first letter, i.e.:
  - **N** ... noun
  - **J** ... adjective
  - **V** ... verb
  - **R** ... adverb
  - **I** ... preposition or subordinating conjunction (?)
- Aggressive smoothing towards two most frequent core tags of each word
  - $\hat{P}(t_{(2)}|w) = \lambda_1 P(t_{(2)}|w)$ where $\lambda_1 < 1.0$
  - $\hat{P}(t_{(1)}|w) = 1 - \hat{P}(t_{(2)}|w)$
  - $\hat{P}(t_{(c)}|w) = 0$ for all $c > 2$

Training on Noisy Data

- Recursively apply the smoothing to subtags
  - E.g. distribute the prob. mass of **N** to the two most probable subtags, **NN** and **NNS**

Training on Noisy Data

- Recursively apply the smoothing to subtags
  - E.g. distribute the prob. mass of **N** to the two most probable subtags, **NN** and **NNS**
- Linear interpolation of model obtained from 1:1 alignments, and of model obtained from 1:N alignments: $P(t|w) = \lambda_2 P_{1:1}(t|w) + (1 - \lambda_2) P_{1:N}(t|w)$
- λ_2 is some weight from $(0; 1)$

Training on Noisy Data

- Recursively apply the smoothing to subtags
  - E.g. distribute the prob. mass of **N** to the two most probable subtags, **NN** and **NNS**
- Linear interpolation of model obtained from 1:1 alignments, and of model obtained from 1:N alignments: $P(t|w) = \lambda_2 P_{1:1}(t|w) + (1 - \lambda_2) P_{1:N}(t|w)$
- λ_2 is some weight from $(0;1)$

- Estimate tag sequence model on filtered, high-confidence alignment data. There are fewer parameters, therefore we can afford it.
  - Alignment confidence score provided by Model 3
  - Sentences where directly projected tags are compatible with the estimated lexical prior probability for each word – penalize less compatible sentences by pseudo-divergence weighting:
    - sentence length $k \Rightarrow weight = \frac{1}{k} \sum_{i=1}^{k} \log \hat{P}(projected\_tag_i|w_i)$

POS Tags Projection across Parallel Corpora

- Dipanjan Das, Slav Petrov (2011). Unsupervised Part-of-Speech Tagging with Bilingual Graph-Based Projections. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pp. 600–609, Portland, Oregon, USA.
  - Differences from Yarowsky and Ngai (2001):
  - Graph-based projection
  - Projected labels are features in an unsupervised model
- Željko Agić, Dirk Hovy, Anders Søgaard (2015). If all you have is a bit of the Bible: Learning POS taggers for truly low-resource languages. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers)*, pp. 268–272, Beijing, China.

Projection Graph

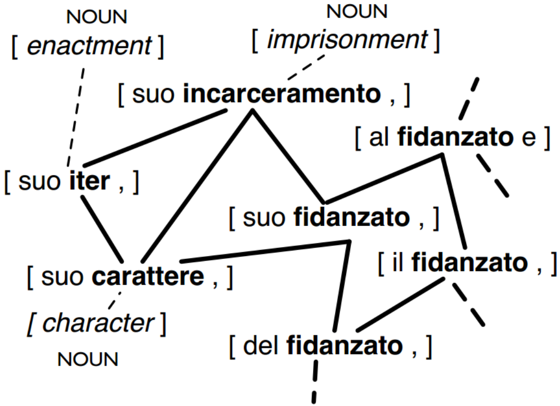
- English vertices = word types
- Foreign vertices = word trigram types

Projection Graph

- English vertices = word types
- Foreign vertices = word trigram types
- English vertices are connected to foreign vertices

Projection Graph

- English vertices = word types
- Foreign vertices = word trigram types
- English vertices are connected to foreign vertices
- Foreign vertices are connected to other foreign vertices



Training

- Parallel English-foreign corpus, word-aligned
    - English side labeled by a supervised English tagger
- Monolingual foreign corpus, unlabeled
    - Used to compute target edge weights (similarity)
    - $\Rightarrow$ We will propagate tags across edges

Monolingual Similarity of Foreign Trigrams

- Trigram type $x_2x_3x_4$ in a sequence $x_1x_2x_3x_4x_5$
- Features:
  - Trigram + Context: $x_1x_2x_3x_4x_5$
  - Trigram: $x_2x_3x_4$
  - Left Context: x_1x_2
  - Right Context: x_4x_5
  - Center Word: x_3
  - Trigram – Center Word: x_2x_4
  - Left Word + Right Context: $x_2x_4x_5$
  - Left Context + Right Word: $x_1x_2x_4$
  - Suffix: $\texttt{HasSuffix}(x_3)$

POS Tags Projection across Parallel Corpora (continued)

- Pruthwik Mishra, Vandan Mujadia, Dipti Misra Sharma (2017). POS Tagging for Resource Poor Indian Languages through Feature Projection
  - In *Proceedings of ICON 2017,* Jadavpur, India
  - Source language: Hindi
  - Target languages:
    - Urdu, Punjabi, Gujarati, Marathi, Konkani, Bengali (Indo-Aryan, i.e., related to Hindi)
    - Telugu, Tamil, Malayalam (Dravidian, i.e., unrelated)

POS Tags Projection across Parallel Corpora (continued)

- Pruthwik Mishra, Vandan Mujadia, Dipti Misra Sharma (2017). POS Tagging for Resource Poor Indian Languages through Feature Projection
  - In *Proceedings of ICON 2017*, Jadavpur, India
  - Source language: Hindi
  - Target languages:
    - Urdu, Punjabi, Gujarati, Marathi, Konkani, Bengali (Indo-Aryan, i.e., related to Hindi)
    - Telugu, Tamil, Malayalam (Dravidian, i.e., unrelated)
  - Parallel corpora: “Health” and “Tourism” (250 to 500K tokens each; not publicly available)
  - Align words using GIZA++

Source Feature Extraction

- Hindi Treebank (450K tokens)
- Prefix features
  - 1 to 7 prefix characters
- Suffix features
  - 1 to 4 suffix characters
- Length of the word
- Previous word
- Current word
- Next word

Features in Hindi – Example

- पत्रकारों *patrakāroṁ* "journalists"

Prefix(1)	प *pa*
Prefix(2)	पत *pata*
Prefix(3)	पत् *pat*
Prefix(4)	पत्र *patra*
Prefix(5)	पत्रक *patraka*
Prefix(6)	पत्रका *patrakā*
Prefix(7)	पत्रकार *patrakāra*
Suffix(1)	◌ं *ṁ*
Suffix(2)	◌ों *oṁ*
Suffix(3)	रों *roṁ*
Suffix(4)	◌ारों *āroṁ*
Length	9
Current	पत्रकारों *patrakāroṁ*
Previous, Next	*context dependent*

Parallel Features in Hindi and Punjabi

- विवाहित *vivāhita* "married"
- ਵਿਆਹੁਤਾ *viāhutā* "married"

Prefix(1)	व *va*	→	ਵ *va*
Prefix(2)	वि *vi*	→	ਵਿ *vi*
Prefix(3)	विव *viva*	→	ਵਿਆ *viā*
Prefix(4)	विवा *vivā*	→	ਵਿਆਹ *viāha*
Prefix(5)	विवाह *vivāha*	→	ਵਿਆਹੁ *viāhu*
Prefix(6)	विवाहि *vivāhi*	→	ਵਿਆਹੁਤ *viāhuta*
Prefix(7)	विवाहित *vivāhita*	→	ਵਿਆਹੁਤਾ *viāhutā*
Suffix(1)	त *ta*	→	ਾ *ā*
Suffix(2)	ित *ita*	→	ਤਾ *tā*
Suffix(3)	हित *hita*	→	ੁਤਾ *utā*
Suffix(4)	ाहित *āhita*	→	ਹੁਤਾ *hutā*
Length	7	→	7
Current	विवाहित *vivāhita*	→	ਵਿਆਹੁਤਾ *viāhutā*

Feature Mapping

- Source features obtained from the Hindi Treebank.
- Projected through word alignment.
- Only the eleven affix features are projected.
- Unclear: what is the rest good for?

Feature Mapping

- Source features obtained from the Hindi Treebank.
- Projected through word alignment.
- Only the eleven affix features are projected.
- Unclear: what is the rest good for?

- "If the same source feature maps to multiple target features, the most probable target feature is selected."
  - 11 mapping files, 1 for each feature type
  - Previous slide: just one aligned pair of words
  - Hindi word occurred multiple times, different targets?

Feature Mapping

- Source features obtained from the Hindi Treebank.
- Projected through word alignment.
- Only the eleven affix features are projected.
- Unclear: what is the rest good for?

- "If the same source feature maps to multiple target features, the most probable target feature is selected."
  - 11 mapping files, 1 for each feature type
  - Previous slide: just one aligned pair of words
  - Hindi word occurred multiple times, different targets?

  - Unclear:
    - Probabilities of the alignment?
    - Or just the count of this correspondence?

Feature Mapping

- Known source feature, but no projection available?

Feature Mapping

- Known source feature, but no projection available?
- Back-off model $\Rightarrow$ shorter feature.
  - Unclear:
    - Map the long source feature to the short target feature?
    - Or simply omit the long feature from the tagging model?

Tagging Model

- POS tags from the Hindi Treebank
- Each Hindi word gets target features
  - $\Rightarrow$ its Hindi features projected to target language
- Similar to word-by-word translation of the training corpus

- Train a model that looks at the target features and predicts a POS tag
- Such model can be applied to the target language
- Features can be obtained directly there

- Method in the paper: CRF++ (Conditional Random Fields)