COS 584

Spring 2021

# P5: Conditional Random Fields

# Shallow Parsing with Conditional Random Fields

**Fei Sha** and **Fernando Pereira**
Department of Computer and Information Science
University of Pennsylvania
200 South 33rd Street, Philadelphia, PA 19104
(feisha|pereira)@cis.upenn.edu

# Conditional Random Fields

- Generative models (HMMs) great for modeling and predicting entire sequences
  - But require lots of (strong) assumptions
- Discriminative models (MEMMs):
  - Great for adding arbitrary features (both local and global)
  - Cannot trade off decisions at different positions

CRFs provide a middle ground - combine the best of generative and discriminative

# History of CRFs

**Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data**

**John Lafferty**[†*] LAFFERTY@CS.CMU.EDU
**Andrew McCallum**[*†] MCCALLUM@WHIZBANG.COM
**Fernando Pereira**[*‡] FPEREIRA@WHIZBANG.COM
[*]WhizBang! Labs–Research, 4616 Henry Street, Pittsburgh, PA 15213 USA
[†]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213 USA
[‡]Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104 USA

- Lafferty, McCallum, Pereria (2001): introduced CRFs for sequence modeling
  - Mitigates the label bias problem (in HMMs/MEMMs)
  - Better empirical performance compared to HMMs/MEMMs
  - Parameter estimation not straightforward

# History of CRFs

- Very popular in the 2000s
- Wide variety of applications:
  - Information extraction
  - Summarization
  - Image labeling/segmentation

Information extraction from research papers using conditional random fields ☆

Fuchun Peng [a], Andrew McCallum [b]

**Multiscale conditional random fields for image labeling**

Publisher: IEEE Cite This PDF

Xuming He ; R.S. Zemel ; M.A. Carreira-Perpinan All Authors

**Document Summarization using Conditional Random Fields**

**Dou Shen[1], Jian-Tao Sun[2], Hua Li[2], Qiang Yang[1], Zheng Chen[2]**
[1]Department of Computer Science and Engineering
Hong Kong University of Science and Technology, Hong Kong
{dshen, qyang}@cse.ust.hk
[2]Microsoft Research Asia, 49 Zhichun Road, China
{jtsun, huli, zhengc}@microsoft.com

# History of CRFs

- Very popular in the 2000s
- Wide variety of applications:
  - Information extraction
  - Summarization
  - Image labeling/segmentation

## Software [ edit ]

This is a partial list of software that implement generic CRF tools.

- RNNSharp CRFs based on recurrent neural networks (C#, .NET)
- CRF-ADF Linear-chain CRFs with fast online ADF training (C#, .NET)
- CRFSharp Linear-chain CRFs (C#, .NET)
- GCO CRFs with submodular energy functions (C++, Matlab)
- DGM General CRFs (C++)
- GRMM General CRFs (Java)
- factorie General CRFs (Scala)
- CRFall General CRFs (Matlab)
- Sarawagi's CRF Linear-chain CRFs (Java)
- HCRF library Hidden-state CRFs (C++, Matlab)
- Accord.NET Linear-chain CRF, HCRF and HMMs (C#, .NET)
- Wapiti Fast linear-chain CRFs (C)[15]
- CRFSuite Fast restricted linear-chain CRFs (C)
- CRF++ Linear-chain CRFs (C++)
- FlexCRFs First-order and second-order Markov CRFs (C++)
- crf-chain1 First-order, linear-chain CRFs (Haskell)
- imageCRF CRF for segmenting images and image volumes (C++)
- MALLET Linear-chain for sequence tagging (Java)

# CRFs for shallow parsing (Sha and Pereira)

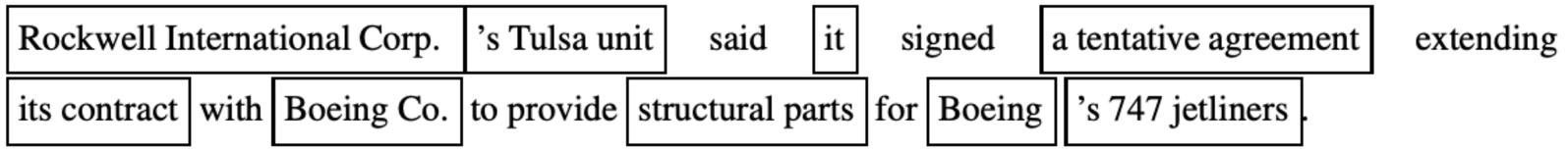


Figure 1: NP chunks

- Predict non-recursive noun phrases
- Framed as a tagging task in BIO format
- Local features defined on **X** (word sequence) and **Y** (tag sequence)
- Maximize log likelihood:

$$\begin{aligned} \mathcal{L}_{\boldsymbol{\lambda}} &= \sum_k \log p_{\boldsymbol{\lambda}}(\boldsymbol{y}_k|\boldsymbol{x}_k) \\ &= \sum_k \left[\boldsymbol{\lambda} \cdot \boldsymbol{F}(\boldsymbol{y}_k, \boldsymbol{x}_k) - \log Z_{\boldsymbol{\lambda}}(\boldsymbol{x}_k)\right] \end{aligned}$$

# CRFs for shallow parsing (Sha and Pereira)

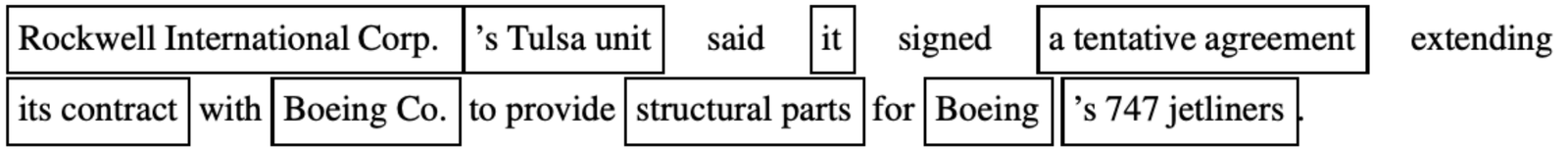


Figure 1: NP chunks

- Maximize log likelihood:

$$\begin{aligned} \mathcal{L}_{\boldsymbol{\lambda}} &= \sum_k \log p_{\boldsymbol{\lambda}}(\boldsymbol{y}_k|\boldsymbol{x}_k) \\ &= \sum_k \left[\boldsymbol{\lambda} \cdot \boldsymbol{F}(\boldsymbol{y}_k, \boldsymbol{x}_k) - \log Z_{\boldsymbol{\lambda}}(\boldsymbol{x}_k)\right] \end{aligned}$$

- Use forward-backward to compute this efficiently!

# Training and features

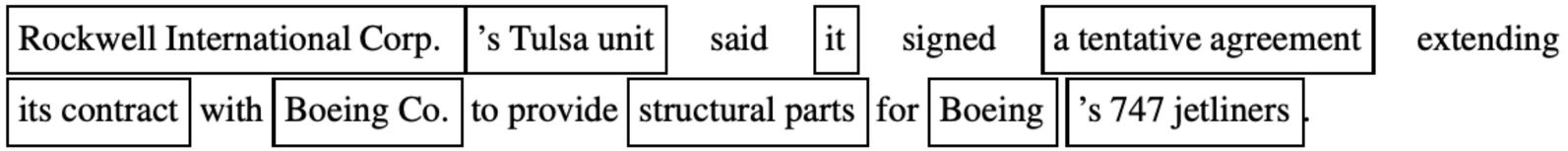


Figure 1: NP chunks

- Various optimization techniques: conjugate GD, quasi-newton, voted perceptron
  - Nowadays - can use SGD with backpropagation
- Second-order markov assumption
  - Constraints on certain feature bigrams (e.g. OI) by setting their weights to $-\infty$

# Results

| Model | F score |
|---|---|
| SVM combination (Kudo and Matsumoto, 2001) | 94.39% |
| CRF | 94.38% |
| Generalized winnow (Zhang et al., 2002) | 93.89% |
| Voted perceptron | 94.09% |
| MEMM | 93.70% |

Table 2: NP chunking F scores

| null hypothesis | p-value |
|---|---|
| CRF vs. SVM | 0.469 |
| CRF vs. MEMM | 0.00109 |
| CRF vs. voted perceptron | 0.116 |
| MEMM vs. voted perceptron | 0.0734 |

Table 4: McNemar's tests on labeling disagreements

# CRFs in deep learning era

Conditional Random Fields as Recurrent Neural Networks

***Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, Philip H. S. Torr***, Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1529-1537

**Neural Architectures for Named Entity Recognition**

**Guillaume Lample**[♠] **Miguel Ballesteros**[♣♠]
**Sandeep Subramanian**[♠] **Kazuya Kawakami**[♠] **Chris Dyer**[♠]
[♠]Carnegie Mellon University [♣]NLP Group, Pompeu Fabra University
{glample,sandeeps,kkawakam,cdyer}@cs.cmu.edu,
miguel.ballesteros@upf.edu

**Bidirectional LSTM-CRF Models for Sequence Tagging**

**Zhiheng Huang** Baidu research huangzhiheng@baidu.com
**Wei Xu** Baidu research xuwei06@baidu.com
**Kai Yu** Baidu research yukai@baidu.com

- Use CRFs on top of neural representations (instead of features and weights)
- Joint sequence prediction without the need for defining features!
- Recent architectures such as seq2seq w/ attention or Transformer may implicitly do the job

# Discussion

- Q1: Sha and Pereira (2003) use a BIO labeling scheme where B indicates start of a chunk, I indicates continuation of the chunk and O indicates a word is outside any chunk. Could we add one more tag E for indicating the end of a chunk? What would be some advantages and disadvantages of doing so?

- Q2: The authors make use of words and POS tags to create features for shallow parsing with CRFs. Can you think of other inputs that might result in better features and help do this task better? Think especially about what a noun phrase fundamentally entails (and doesn't) and what information might help identify one.

- Can you think of any applications related to your research/area of study where you can use CRFs?