Dynamic Conditional Random Fields

Factorized Probabilistic Models for Labeling and Segmenting Sequence Data

Martin Zimmer Kristensen

November 1, 2016

Outline

Sequential Data

Part-of-speech Tagging

The [DT] little [JJ] dog [NN] was [VBD] furious [JJ] and [CC] barked [VBD] at [IN] the [DT] large [JJ] human [NN]

Noun-phrase Chunking

The [B-NP] little [I-NP] dog [I-NP] was [O] furious [O] and [O] barked [O] at [O] the [B-NP] large [I-NP] human [I-NP]

Sequential Data

Other

- Named Entity Recognition
- Speech Recognition

Motivation

Generative Models (HMMs, DBNs):

- The joint probability $p(x, y)$
  - Generate features
  - Unnecessary when only segmenting and labeling data
  - Assumptions among features to achieve tractability
    - Hurts performance

Motivation

Discriminative Models (CRFs, DCRFs):

- The conditional probability: $p(y|x)$
  - Dependencies among x not explicitly modeled
  - No assumptions among (interdependent) features
    - Capitalization, prefixes, suffixes, neighboring words...
  - Unseen words can be labeled by using interdependent features

Conditional Random Fields

Definition (Conditional Random Fields)

- Let G be an undirected model over sets of random variables y and x
- Let $C = \{\{y_c, x_c\}\}$ be the set of cliques in G
- Conditional probability defined as:

$$p_{\Lambda}(y|x) = \frac{1}{Z(x)} \prod_{c \in C} \Phi(y_c, x_c)$$

- Φ is a potential function
- $Z(x)$ is the partition function

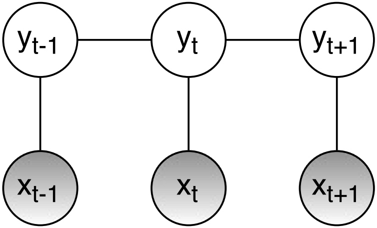
Conditional Random Fields

Feature Functions

- Potentials factorize according to a set of weights $\{\lambda_k\}$ and feature functions $\{f_k\}$:

$$\Phi(y_c, x_c) = exp\left(\sum_k \lambda_k f_k(y_c, x_c) \right)$$

Linear-chain CRF



- Previous applications use linear-chain CRF
    - Tractable exact inference algorithms
- Conditional version of hidden Markov models
- Feature functions can be described as:

$$f_k(y_{t-1}, y_t, x, t)$$

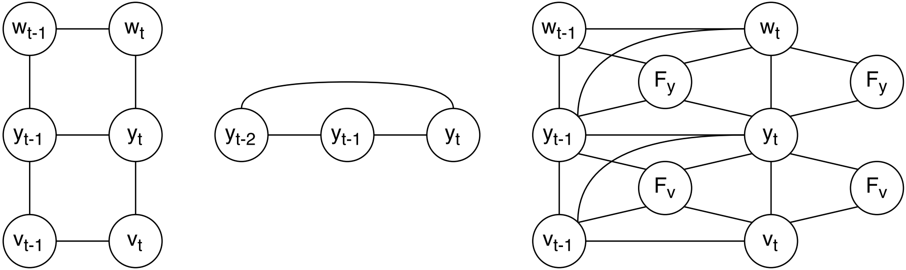
Linear-chain CRF

Feature Functions:

- $f(y_{t-1}, y_t, x, t) = 1$:
  - *iff* $y_{t-1} = \textit{adjective}$, $y_t = \textit{proper noun}$, and x_t begins with a capital letter.
- $f(y_{t-1}, y_t, x, t) = 1$:
  - *iff* $y_t = \textit{organization}$, $x_t = \text{``New''}$, $x_{t+1} = \text{``York''}$, and $x_{t+2} = \text{``Times''}$

Key Contributions

- Dynamic Conditional Random Fields (DCRF)
  - Generalization of linear-chain CRFs
  - Similar to DBNs
  - Factorial CRF
  - Inference approximation algorithm for complex models



Dynamic Conditional Random Fields

- y_{ij} is the variable j at time i

Definition (Clique Index)

- Given a time t, denote any variable y_{ij} in y by:
  - Its index j in y_i
  - Its time offset $\Delta t = i - t$
- $c = \{(\Delta t, j)\}$ is a clique index
- $y_{t,c}$ is the set of variables in clique index c at time t

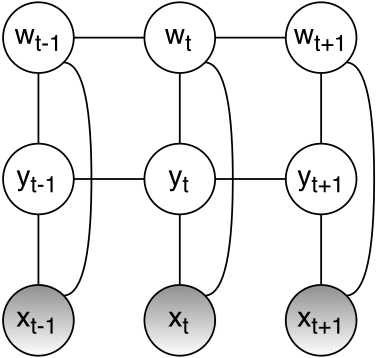
Dynamic Conditional Random Fields

Definition (Dynamic Conditional Random Field)

- $p(y|x) = \frac{1}{Z(x)} \prod_t \prod_{c \in C} \exp\left(\sum_k \lambda_k f_k(y_{t,c}, x, t) \right)$
- where $Z(x)$ is the partition function

Factorial Conditional Random Field

- A DCRF which has linear chains of labels with edges between cotemporal labels.



Factorial Conditional Random Field

Cliques

- The cliques are of the form:
  - Within-chain edges: $\{(0, \ell), (1, \ell)\}$
  - Between-chain edges: $\{(0, \ell), (0, \ell + 1)\}$

Factorial Conditional Random Field

Definition (Factorial CRF)

$$p(x|y) = \frac{1}{Z(x)}\left(\prod_{t=1}^{T-1}\prod_{\ell=1}^{L}\Phi_\ell(y_{\ell,t}, y_{\ell,t+1}, x, t)\right)\left(\prod_{t=1}^{T}\prod_{\ell=1}^{L-1}\Psi_\ell(y_{\ell,t}, y_{\ell+1,t}, x, t)\right)$$

- $\{\Phi_\ell\}$ are the factors over within-chain edges
- $\{\Psi_\ell\}$ are the factors over between-chain edges
- $Z(x)$ is the partition function.

Factorial Conditional Random Field

Factors

- The factors are modeled using features $\{f_k\}$ and weights $\{\lambda_k\}$ of G as:

$$\Phi_\ell(y_{\ell,t}, y_{\ell,t+1}, x, t) = \exp\left\{ \sum_k \lambda_k f_k(y_{\ell,t}, y_{\ell,t+1}, x, t) \right\},$$

$$\Psi_\ell(y_{\ell,t}, y_{\ell+1,t}, x, t) = \exp\left\{ \sum_k \lambda_k f_k(y_{\ell,t}, y_{\ell+1,t}, x, t) \right\}.$$

Inference

- Exact inference intractable for some models
- Approximate inference using loopy belief propagation

Inference

Loopy Belief Propagation

- Message from node x_u to node x_v:

$$m_{x_u}(x_v)$$

- Value of $m_{x_u}(x_v)$:
  - The belief of x_u about the probability $p(x_v)$
- Iteratively send messages until convergence or early cutoff
- Different schedules can be applied
  - Random
  - Tree-based (send messages from leaves to root and back)

Parameter Estimation

- Given training data $D = \{x^{(i)}, y^{(i)}\}_{i=1}^{N}$
  - Find s set of parameters $\Lambda = \{\lambda_k\}$
- Use L-BFGS

Experiments

Noun-phrase Chunking

The [B-NP] little [I-NP] dog [I-NP] was [O] furious [O] and [O] barked [O] at [O] the [B-NP] large [I-NP] human [I-NP]

Usual approach:

1. POS tagging
2. Noun-phrase Chunking

Challenge:

- Mistakes in POS tagging will cascade onto noun-phrase chunking

Experiments

Data:

- CoNLL 2000

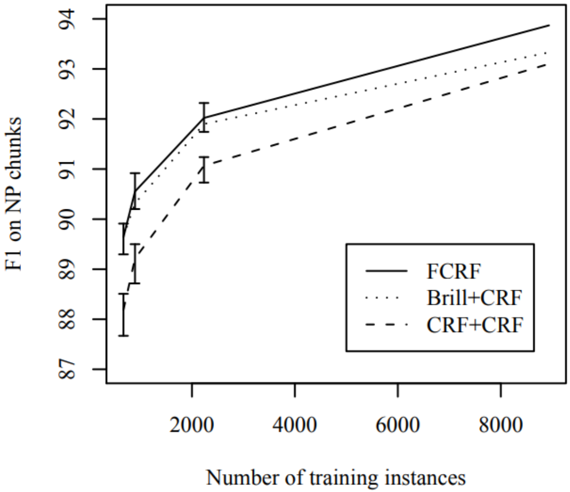
Approach:

- Use a factorial CRF to jointly do POS and chunking

Compare to:

- CRF+CRF
- Brill+CRF
  - Brill tagger trained on over four times more data including the CoNLL 2000
  - More than 40.000 sentences

Results



F1 on NP chunks
87
88
89
90
91
92
93
94
2000
4000
6000
8000
Number of training instances
FCRF
Brill+CRF
CRF+CRF

Results

	Size	CRF+CRF	Brill+CRF	FCRF
POS accuracy	223	86.23	N/A	**93.12**
	447	90.44		**95.43**
	670	92.33		**96.34**
	894	93.56		**96.85**
	2234	96.18		**97.87**
	8936	98.28		**98.92**
NP accuracy	223	92.67	93.75	**93.87**
	447	94.09	94.91	**95.03**
	670	94.72	95.46	95.46
	894	95.17	95.75	**95.86**
	2234	96.08	96.38	**96.51**
	8936	96.98	97.09	**97.36**
Joint accuracy	223	81.92	N/A	**89.19**
	447	86.58		**91.85**
	670	88.68		**92.86**
	894	90.06		**93.60**
	2234	93.00		**94.90**
	8936	95.56		**96.48**
NP F1	223	83.84	86.02	**86.03**
	447	86.87	88.56	**88.59**
	670	88.19	**89.65**	89.64
	894	89.21	90.31	**90.55**
	2234	91.07	91.90	**92.02**
	8936	93.10	93.33	**93.87**

Inference Algorithms

Method	Time (hr)		NP F1		LBFGS iter
	μ	s	μ	s	μ
Random (3)	15.67	2.90	88.57	0.54	63.6
Tree (3)	13.85	11.6	88.02	0.55	32.6
Tree (∞)	13.57	3.03	88.67	0.57	65.8
Random (∞)	13.25	1.51	88.60	0.53	76.0
Exact	20.49	1.97	88.63	0.53	73.6

Conclusions

- Jointly perform several labeling tasks at once perform better than the sequential approach
    - Useful for many NLP tasks
- Loopy belief propagation:
    - Reduces training time
    - Performs equally to exact inference