



知的情報処理システム特論 第15回

二宮 崇

今日の講義の予定

- 系列ラベリングのための確率的識別モデル(Linear-chain CRF)
 - 解析 (ビタビアルゴリズム)
 - 学習 (前向き後ろ向きアルゴリズム)
- 教科書
 - 高村 大也 (著), 奥村 学 (監修) 言語処理のための機械学習入門 (自然言語処理シリーズ), コロナ社, 2010
 - Yusuke Miyao (2006) From Linguistic Theory to Syntactic Analysis: Corpus-Oriented Grammar Development and Feature Forest Model, Ph.D Thesis, University of Tokyo
 - Jun'ichi Kazama (2004) Improving Maximum Entropy Natural Language Processing by Uncertainty-aware Extensions and Unsupervised Learning, Ph.D. Thesis, University of Tokyo
 - Cristopher M. Bishop "PATTERN RECOGNITION AND MACHINE LEARNING" Springer, 2006



系列ラベリングのための確率的識別モデル

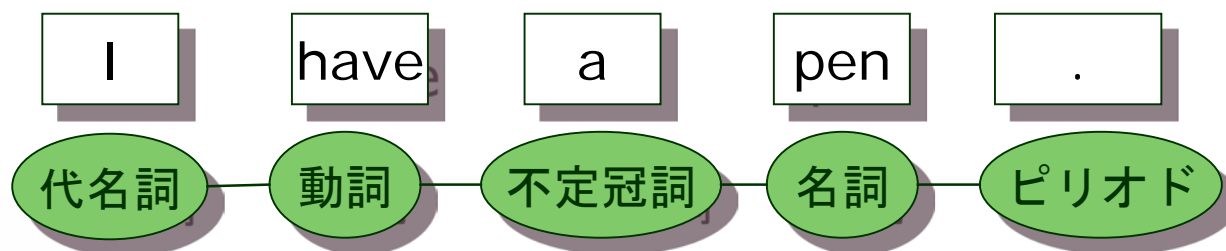
LINEAR-CHAIN CRF



系列ラベリングのための確率的識別モデル

Linear-chain CRF

- CRF (Conditional Random Fields, 条件付確率場)
 - 最大エントロピーモデルの一種
 - 出力 y は入力 x に対する条件付確率
 - 出力 y が無向グラフ(y をノードに分解できる)
 - 素性を各ノードの近傍で定義
- Linear-chain CRF
 - ノードが線形につながったCRF (系列ラベリングのためのCRF)
 - 例えば、品詞解析の場合、入力 x が文で、出力 y が単語-品詞列となる。単語と品詞のペアがノードになる。素性を各ノードの近傍で定義する。



x = "I have a pen."

y = "I-代名詞 have-動詞 a-不定冠詞 pen-名詞 .-ピリオド"

に対し $p(y|x)$ を求める



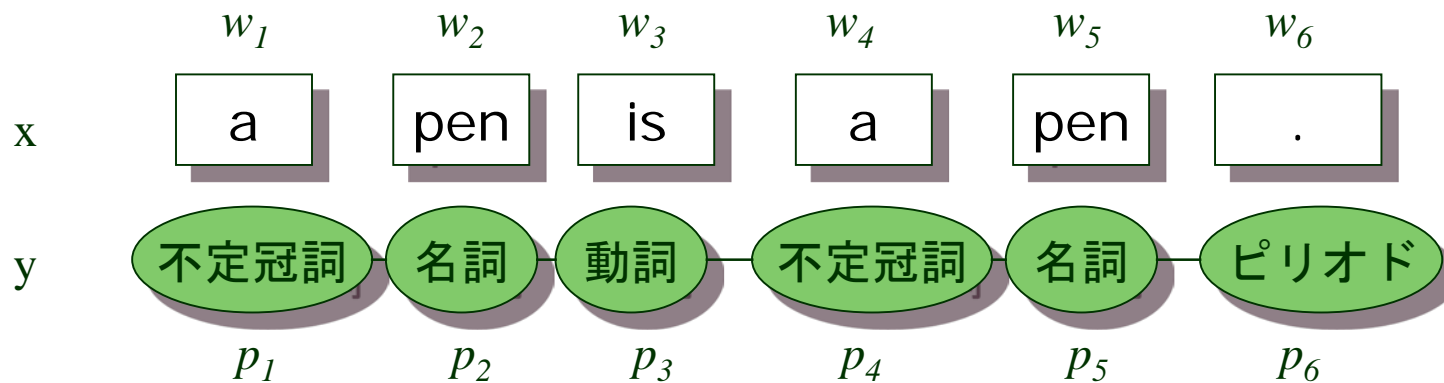
Linear-chain CRF: 素性テンプレート

- 素性テンプレート
 - どのような素性を採用するか表記したもの
 - 訓練データに対し素性テンプレートを適用し、実際に出現した値だけを素性として採用
- 素性テンプレートの例
 - i 番目のノードを c_i とする
 - i 番目の品詞を p_i とする
 - i 番目の単語を w_i とする
 - HMMと同様の素性の場合
 - 各ノード c_i に対し、 $p_{i-1}p_i$ [最大で品詞数×品詞数の次元ができる]
 - 各ノード c_i に対し、 $w_i p_i$ [最大で単語数×品詞数の次元ができる]
 - 他にもいろいろな組合せの素性を追加できる
 - 各ノード c_i に対し、 $w_{i-1}w_i p_i$
 - 各ノード c_i に対し、 $p_{i-2}p_{i-1}p_i$



Linear-chain CRF: 素性テンプレートと素性

- 素性テンプレートの例
 - HMMと同様の素性の場合
 - 各ノード c_i に対し、 $p_{i-1}p_i$
 - 各ノード c_i に対し、 $w_i p_i$



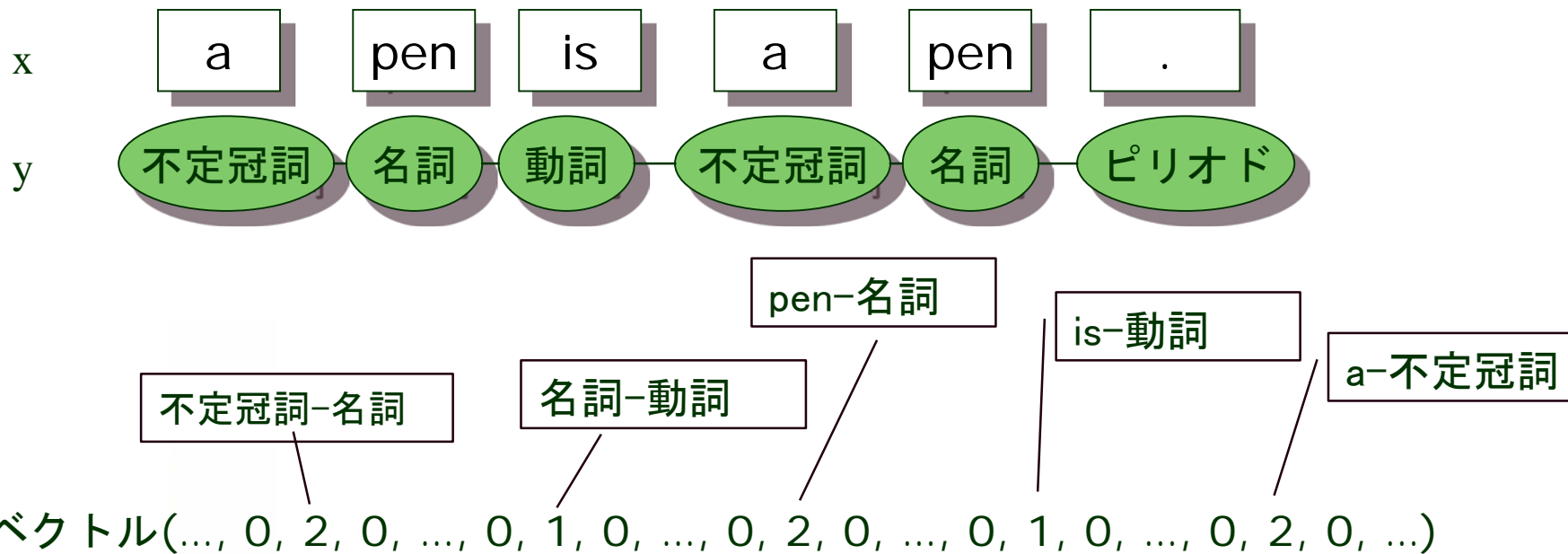
$p_{i-1}p_i$ から生成される素性	$w_i p_i$ から生成される素性
不定冠詞-名詞	a-不定冠詞
名詞-動詞	pen-名詞
動詞-不定冠詞	is-動詞
名詞-ピリオド	.-ピリオド



Linear-chain CRF: 全体の素性ベクトルと局所的素性ベクトルの関係

- 各素性関数の値は1文中に各素性が発火(出現)した回数

$$f_j(x, y) = \sum_c f_j(c)$$



Linear-chain CRF: 全体のスコアと局所的なスコアの関係

- 全体のスコア=各ノード c に対するスコアの積

$$p(y|x;\lambda) = \frac{1}{Z} \exp\left(\sum_j \lambda_j f_j(x,y)\right) = \frac{1}{Z} \exp\left(\sum_j \lambda_j \sum_c f_j(c)\right) = \frac{1}{Z} \exp\left(\sum_c \sum_j \lambda_j f_j(c)\right) = \frac{1}{Z} \prod_c \exp\left(\sum_j \lambda_j f_j(c)\right)$$

文全体の素性ベクトル: $(2,0,3,1,1)$

$$e^{\lambda_1 \cdot 2 + \lambda_2 \cdot 0 + \lambda_3 \cdot 3 + \lambda_4 \cdot 1 + \lambda_5 \cdot 1}$$

掛算

$$e^{\lambda_1 \cdot 1 + \lambda_2 \cdot 0 + \lambda_3 \cdot 1 + \lambda_4 \cdot 1 + \lambda_5 \cdot 0}$$

$$e^{\lambda_1 \cdot 0 + \lambda_2 \cdot 0 + \lambda_3 \cdot 1 + \lambda_4 \cdot 0 + \lambda_5 \cdot 0}$$

$$e^{\lambda_1 \cdot 1 + \lambda_2 \cdot 0 + \lambda_3 \cdot 1 + \lambda_4 \cdot 0 + \lambda_5 \cdot 1}$$

$(1,0,1,1,0)$

W_1

p_1

$(0,0,1,0,0)$

W_2

p_2

$(1,0,1,0,1)$

W_3

p_3



Linear-chain CRF: スコアの分解と HMMとの対応

- 単語-品詞列の確率

$$p(y|x; \lambda) = \frac{1}{Z(x, \lambda)} \exp\left(\sum_j \lambda_j f_j(x, y)\right) = \frac{1}{Z(x, \lambda)} \prod_c \exp\left(\sum_j \lambda_j f_j(c)\right)$$

品詞列集合全体のスコアの和 (前向きアルゴリズムで計算)

HMMの遷移確率と出力確率に対応

- 解析: ビタビアルゴリズムの適用

- 遷移確率と出力確率 \Leftrightarrow ノードのスコア $\exp \sum_j \lambda_j f_j(c)$

- 学習: 前向き後向きアルゴリズムの適用

- 遷移回数と出力回数 \Leftrightarrow 素性値(素性の発火回数)



1次のLinear-chain CRF

- ラベル集合を Q 、入力を x 、出力を $q_1q_2\dots q_T$ とする。
- Linear-chain CRF

$$p(q_1q_2\dots q_T | x) = \frac{1}{Z_x} \prod_{t=1}^T \exp \left\{ \sum_j \lambda_j f_j(q_{t-1}, q_t, x, t) \right\}$$

$$Z_x = \sum_{q'_1 \in Q, \dots, q'_T \in Q} \prod_{t=1}^T \exp \left\{ \sum_j \lambda_j f_j(q'_{t-1}, q'_t, x, t) \right\}$$

ただし、 Z_x は分配関数、 λ_j は f_j に対する重み、 $q_0 = \text{DUMMY}$ (ダミーのラベル) とし、素性関数 f の引数は次のように定義される。

f (時刻 $t-1$ のラベル, 時刻 t のラベル, 入力, 時刻)

- 入力(第三引数)と時刻(第四引数)を与えることによって、入力の情報を素性として用いることができる
 - 例えば、品詞解析の場合、ラベルが品詞であり、入力が単語列となり、時刻 t やその前後における単語を素性として用いることができる。

LINEAR-CHAIN CRFの解析



Linear-chain CRFの解析

- 入力 x
- ラベル集合 Q
- Linear-chain CRFの解析

$$\hat{q}_1 \hat{q}_2 \cdots \hat{q}_T = \arg \max_{q_1 q_2 \cdots q_T} p(q_1 q_2 \cdots q_T | x) = \arg \max_{q_1 q_2 \cdots q_T} \prod_{t=1}^T \exp \left\{ \sum_j \lambda_j f_j(q_{t-1}, q_t, x, t) \right\}$$

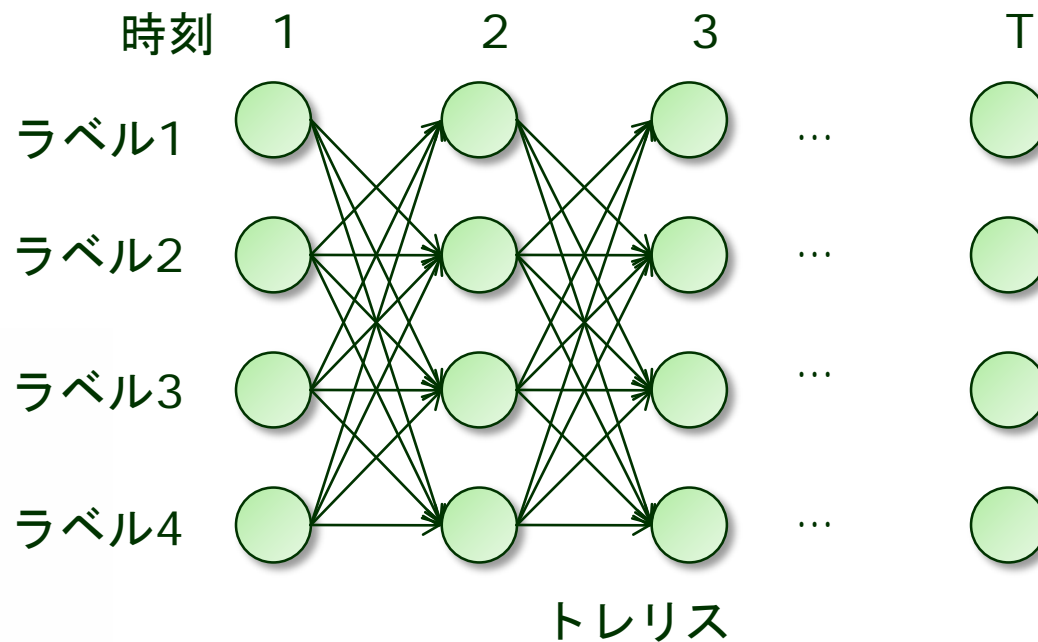
(Z_x は定数であるため、必要ない)

Linear-chain CRFの解析: ビタビアルゴリズム

● 動的計画法

$\delta(t, q)$: 時刻 t にラベル q となるラベル列の中で最大のスコア

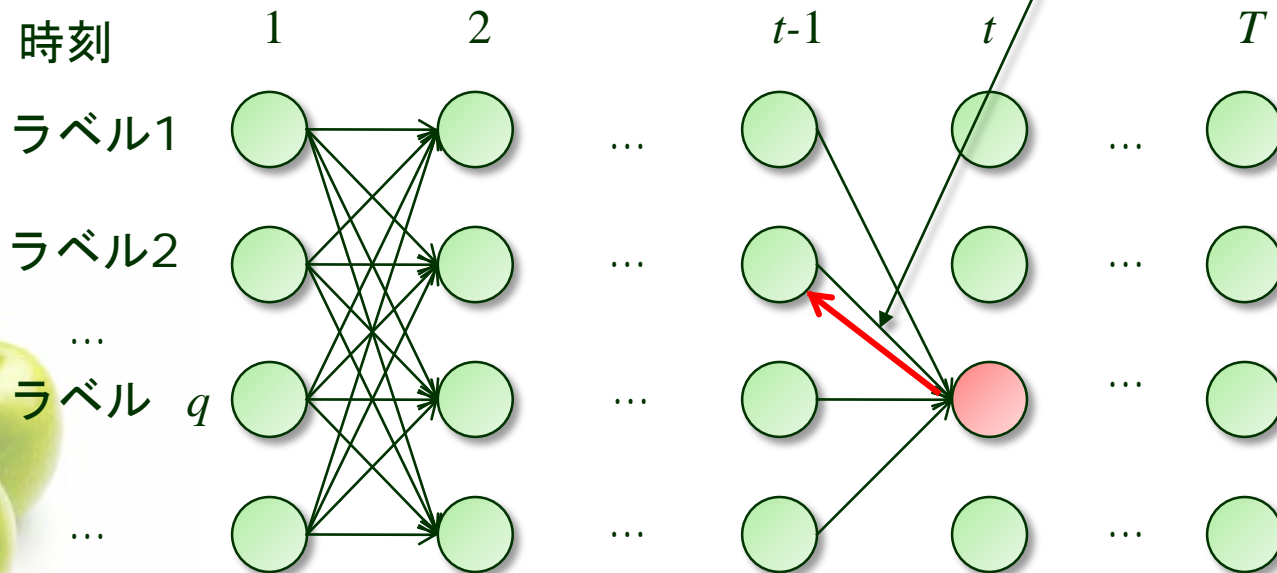
$\max_{q \in Q} \delta(T, q)$ を求めれば良い



Linear-chain CRFの解析: ビタビアルゴリズム

$$\begin{aligned}
 \delta(t, q) &= \max_{q_1 \in Q, \dots, q_{t-1} \in Q} \left(\prod_{u=1}^{t-1} e^{\sum_j \lambda_j f_j(q_{u-1}, q_u, x, u)} \right) e^{\sum_j \lambda_j f_j(q_{t-1}, q, x, t)} \\
 &= \max_{q_{t-1} \in Q} \max_{q_1 \in Q, \dots, q_{t-2} \in Q} \left(\prod_{u=1}^{t-2} e^{\sum_j \lambda_j f_j(q_{u-1}, q_u, x, u)} \right) e^{\sum_j \lambda_j f_j(q_{t-2}, q_{t-1}, x, t-1)} e^{\sum_j \lambda_j f_j(q_{t-1}, q, x, t)} \\
 &= \max_{q_{t-1} \in Q} \delta(t-1, q_{t-1}) e^{\sum_j \lambda_j f_j(q_{t-1}, q, x, t)}
 \end{aligned}$$

最大スコアで遷移したパスをバックポインタで保存



Linear-chain CRF:解析(ビタビアルゴリズム)

- 入力 x
- Linear-chain CRFの解析

$$\max_{q_1 q_2 \cdots q_T} p(q_1 q_2 \cdots q_T | x) = \max_{q_1 q_2 \cdots q_T} \prod_{t=1}^T \exp \left\{ \sum_j \lambda_j f_j(q_{t-1}, q_t, x, t) \right\} = \max_q \delta(T, q)$$

- Linear-chain CRFのビタビアルゴリズム

HMMとの対応

for $q \in Q$

$$\delta[1, q] := \exp\{\sum_j \lambda_j f_j(\text{DUMMY}, q, x, 1)\} \longleftrightarrow \pi[q] b[q, o_1]$$

for $t = 2$ to T

for $q \in Q$

$$\delta[t, q] := \max_{q' \in Q} \delta[t-1, q'] \exp\{\sum_j \lambda_j f_j(q', q, x, t)\}$$

$$\text{bp}[t, q] := \text{argmax}_{q' \in Q} \delta[t-1, q'] \exp\{\sum_j \lambda_j f_j(q', q, x, t)\}$$

$$a[q', q] b[q, o_t]$$



LINEAR-CHAIN CRFの学習



Linear-chain CRFの学習

- 学習 (パラメータ推定)

- 訓練データの対数尤度(=目的関数)に対する勾配を0にする

$$\frac{\partial L(\lambda)}{\partial \lambda_j} = \sum_{i=1}^N f_j(x_i, y_i) - \sum_{i=1}^N \sum_{y \in Y(x_i)} p(y | x_i; \lambda) f_j(x_i, y)$$

- 目的関数とその勾配が計算できれば勾配法でパラメータが求まる。

- 問題点

- 素性関数の期待値の計算: 「ある文 x に対する全ての可能なラベル列集合 $Y(x)$ に対する確率」を計算しないといけない
- 文長 n 、ラベル数 l に対し、可能なラベル列は l^n ある。全て展開するのはほぼ不可能

- 解決策

- 前向き後ろ向きアルゴリズム
- 畳み込まれたデータ構造を展開することなく素性関数の期待値を計算



Linear-chain CRFの学習: 目的関数

- 訓練データ $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$
- 各 y_i を $y_i = q_{i1}q_{i2}\dots q_{iT_i}$ とする。
- 目的関数: 訓練データに対するLinear-chain CRFの対数尤度
- パラメータ λ に対する目的関数 $L(\lambda)$

$$\begin{aligned} L(\lambda) &= \log \left(\prod_{i=1}^N p(y_i | x_i; \lambda) \right) \\ &= \sum_{i=1}^N \log \frac{1}{Z(x_i, \lambda)} \prod_{t=1}^{T_i} \exp \left(\sum_j \lambda_j f_j(q_{i(t-1)}, q_{it}, x_i, t) \right) \\ &= - \sum_{i=1}^N \log Z(x_i, \lambda) + \sum_{i=1}^N \sum_{t=1}^{T_i} \sum_j \lambda_j f_j(q_{i(t-1)}, q_{it}, x_i, t) \end{aligned}$$

Linear-chain CRFの学習: 目的関数の勾配 (1/2)

- 目的関数 $L(\lambda)$ に対するパラメータ λ の勾配

$$\begin{aligned}\frac{\partial L(\lambda)}{\partial \lambda_j} &= \sum_{i=1}^N \sum_{t=1}^{T_i} f_j(q_{i(t-1)}, q_{it}, x_i, t) - \sum_{i=1}^N \frac{1}{Z(x_i, \lambda)} \frac{\partial Z(x_i, \lambda)}{\partial \lambda_j} \\ &= \sum_{i=1}^N \sum_{t=1}^{T_i} f_j(q_{i(t-1)}, q_{it}, x_i, t) \\ &\quad - \sum_{i=1}^N \frac{1}{Z(x_i, \lambda)} \sum_{q'_1 \in Q, \dots, q'_{T_i} \in Q} \sum_{t=1}^{T_i} f_j(q'_{t-1}, q'_t, x_i, t) \exp \left\{ \sum_{u=1}^{T_i} \sum_j \lambda_j f_j(q'_{u-1}, q'_u, x_i, u) \right\} \\ &= \sum_{i=1}^N \sum_{t=1}^{T_i} f_j(q_{i(t-1)}, q_{it}, x_i, t) - \sum_{i=1}^N \sum_{q'_1 \in Q, \dots, q'_{T_i} \in Q} p(q'_1 \cdots q'_{T_i} | x_i) \sum_{t=1}^{T_i} f_j(q'_{t-1}, q'_t, x_i, t)\end{aligned}$$

↑
素性関数の合計の期待値となっている



Linear-chain CRFの学習: 目的関数の勾配 (2/2)

● 目的関数 $L(\lambda)$ に対するパラメータ λ の勾配

(前のスライドの続き)

$$\begin{aligned}\frac{\partial L(\lambda)}{\partial \lambda_j} &= \sum_{i=1}^N \sum_{t=1}^{T_i} f_j(q_{i(t-1)}, q_{it}, x_i, t) - \sum_{i=1}^N \sum_{q'_1 \in Q, \dots, q'_{T_i} \in Q} p(q'_1 \cdots q'_{T_i} | x_i) \sum_{t=1}^{T_i} f_j(q'_{t-1}, q'_t, x_i, t) \\ &= \sum_{i=1}^N \sum_{t=1}^{T_i} f_j(q_{i(t-1)}, q_{it}, x_i, t) - \sum_{i=1}^N \sum_{t=1}^{T_i} \sum_{q'_{t-1} \in Q, q'_t \in Q} f_j(q'_{t-1}, q'_t, x_i, t) \sum_{q'_1 \in Q, \dots, q'_{t-2} \in Q, q'_{t+1} \in Q, \dots, q'_{T_i} \in Q} p(q'_1 \cdots q'_{T_i} | x_i) \\ &= \sum_{i=1}^N \sum_{t=1}^{T_i} f_j(q_{i(t-1)}, q_{it}, x_i, t) - \sum_{i=1}^N \sum_{t=1}^{T_i} \sum_{q'_{t-1} \in Q, q'_t \in Q} f_j(q'_{t-1}, q'_t, x_i, t) p(q'_{t-1} q'_t | x_i)\end{aligned}$$

$q_{t-1} q_t$ の周辺確率を効率的に
求められればよい



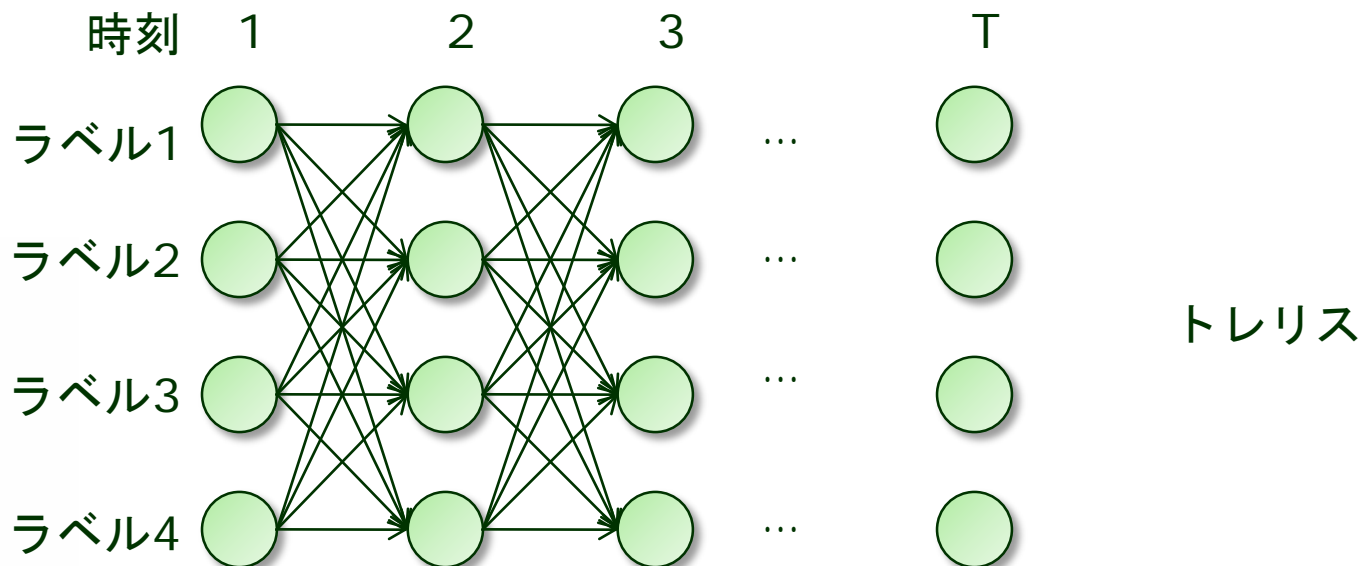
Linear-chain CRFの学習: 前向きアルゴリズム (1/3)

動的計画法

$\alpha(t, q)$: 時刻 t にラベル q であるスコアの合計

$$\alpha(t, q) = \sum_{q_1 \in Q, \dots, q_{t-1} \in Q} \left(\prod_{u=1}^{t-1} e^{\sum_j \lambda_j f_j(q_{u-1}, q_u, x, u)} \right) e^{\sum_j \lambda_j f_j(q_{t-1}, q, x, t)}$$

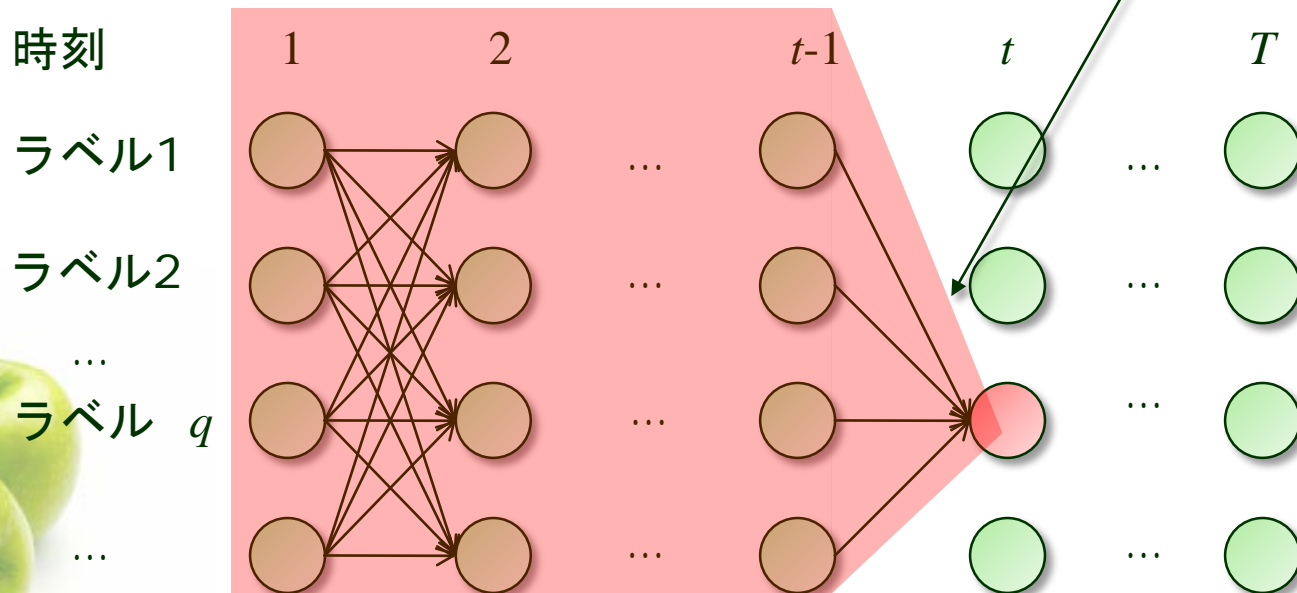
全ての t, q に対し、 $\alpha(t, q)$ を求めれば良い



Linear-chain CRFの学習: 前向きアルゴリズム (2/3)

$$\begin{aligned}
 \alpha(t, q) &= \sum_{q_1 \in Q, \dots, q_{t-1} \in Q} \left(\prod_{u=1}^{t-1} e^{\sum_j \lambda_j f_j(q_{u-1}, q_u, x, u)} \right) e^{\sum_j \lambda_j f_j(q_{t-1}, q, x, t)} \\
 &= \sum_{q_{t-1} \in Q} \sum_{q_1 \in Q, \dots, q_{t-2} \in Q} \left(\prod_{u=1}^{t-2} e^{\sum_j \lambda_j f_j(q_{u-1}, q_u, x, u)} \right) e^{\sum_j \lambda_j f_j(q_{t-2}, q_{t-1}, x, t-1)} e^{\sum_j \lambda_j f_j(q_{t-1}, q, x, t)} \\
 &= \sum_{q_{t-1} \in Q} \alpha(t-1, q_{t-1}) e^{\sum_j \lambda_j f_j(q_{t-1}, q, x, t)}
 \end{aligned}$$

全てのスコアの和



Linear-chain CRFの学習: 前向きアルゴリズム (3/3)

- Linear-chain CRFの前向きアルゴリズム

for $q \in Q$

$$\alpha[1, q] := \exp\{\sum_j \lambda_j f_j(\text{DUMMY}, q, x, 1)\}$$

for $t = 2$ to T

for $q \in Q$

$$\alpha[t, q] := \sum_{q' \in Q} \alpha[t-1, q'] \exp\{\sum_j \lambda_j f_j(q', q, x, t)\}$$

- 入力 x とパラメータ λ が与えられているとする
- 分配関数 Z_x を効率的に求めることもできる
 \Rightarrow 目的関数を効率的に求めることができる!



Linear-chain CRFの学習: 後ろ向きアルゴリズム (1/3)

- 前向きアルゴリズムを逆方向に適用
 - 文末から文頭に向かって実行
 - 向きが逆になることを除けば前向きアルゴリズムとまったく同じ
- $\beta(t, q)$: 時刻 t にラベル q となっているとき、時刻 $t+1$ から文末までのスコアの合計

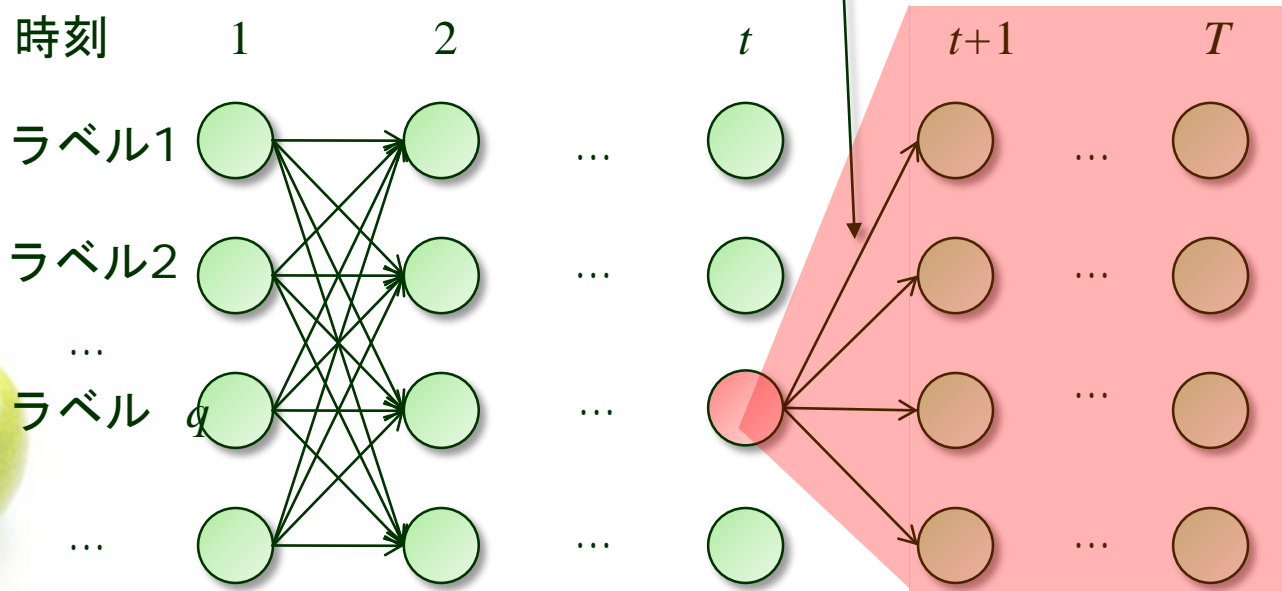
$$\beta(t, q) = \sum_{q_{t+1} \in Q, \dots, q_T \in Q} e^{\sum_j \lambda_j f_j(q, q_{t+1}, x, t+1)} \prod_{u=t+2}^T e^{\sum_j \lambda_j f_j(q_{u-1}, q_u, x, u)}$$



Linear-chain CRFの学習: 後ろ向きアルゴリズム (2/3)

$$\begin{aligned}
 \beta(t, q) &= \sum_{q_{t+1} \in Q, \dots, q_T \in Q} e^{\sum_j \lambda_j f_j(q, q_{t+1}, x, t+1)} \prod_{u=t+2}^T e^{\sum_j \lambda_j f_j(q_{u-1}, q_u, x, u)} \\
 &= \sum_{q_{t+1} \in Q} \sum_{q_{t+2} \in Q, \dots, q_T \in Q} e^{\sum_j \lambda_j f_j(q, q_{t+1}, x, t+1)} e^{\sum_j \lambda_j f_j(q_{t+1}, q_{t+2}, x, t+2)} \prod_{u=t+3}^T e^{\sum_j \lambda_j f_j(q_{u-1}, q_u, x, u)} \\
 &= \sum_{q_{t+1} \in Q} e^{\sum_j \lambda_j f_j(q, q_{t+1}, x, t+1)} \beta(t+1, q_{t+1})
 \end{aligned}$$

全てのスコアの和



Linear-chain CRFの学習: 後ろ向きアルゴリズム (3/3)

- 後ろ向きアルゴリズム

for $q \in Q$

$\beta[T, q] := 1$

for $t = T - 1$ to 1

for $q \in Q$

$\beta[t, q] := \sum_{q' \in Q} \beta[t + 1, q'] \exp\{\sum_j \lambda_j f_j(q, q', x, t + 1)\}$



Linear-chain CRF

学習(前向き後ろ向きアルゴリズム)

- $q_{t-1} q_t$ の周辺確率

$$\begin{aligned}
 p(q'_{t-1} q'_t | x_i) &= \sum_{q'_1 \in Q, \dots, q'_{t-2} \in Q} \sum_{q'_{t+1} \in Q, \dots, q'_{T_i} \in Q} p(q'_1 \cdots q'_{T_i} | x_i) \\
 &= \frac{1}{Z(x_i, \lambda)} \sum_{q'_1 \in Q, \dots, q'_{t-2} \in Q} \sum_{q'_{t+1} \in Q, \dots, q'_{T_i} \in Q} \prod_{u=1}^{T_i} e^{\sum_j \lambda_j f_j(q'_{u-1}, q'_u, x_i, u)} \\
 &= \frac{1}{Z(x_i, \lambda)} \sum_{q'_1 \in Q, \dots, q'_{t-2} \in Q} \sum_{q'_{t+1} \in Q, \dots, q'_{T_i} \in Q} \left(\prod_{u=1}^{t-1} e^{\sum_j \lambda_j f_j(q'_{u-1}, q'_u, x_i, u)} \right) e^{\sum_j \lambda_j f_j(q'_{t-1}, q'_t, x_i, t)} \left(\prod_{u=t+1}^{T_i} e^{\sum_j \lambda_j f_j(q'_{u-1}, q'_u, x_i, u)} \right) \\
 &= \frac{1}{Z(x_i, \lambda)} e^{\sum_j \lambda_j f_j(q'_{t-1}, q'_t, x_i, t)} \sum_{q'_1 \in Q, \dots, q'_{t-2} \in Q} \left(\prod_{u=1}^{t-1} e^{\sum_j \lambda_j f_j(q'_{u-1}, q'_u, x_i, u)} \right) \sum_{q'_{t+1} \in Q, \dots, q'_{T_i} \in Q} \left(\prod_{u=t+1}^{T_i} e^{\sum_j \lambda_j f_j(q'_{u-1}, q'_u, x_i, u)} \right) \\
 &= \frac{1}{Z(x_i, \lambda)} e^{\sum_j \lambda_j f_j(q'_{t-1}, q'_t, x_i, t)} \sum_{q'_1 \in Q, \dots, q'_{t-2} \in Q} \left(\prod_{u=1}^{t-1} e^{\sum_j \lambda_j f_j(q'_{u-1}, q'_u, x_i, u)} \right) \beta(t, q'_t) \\
 &= \frac{1}{Z(x_i, \lambda)} e^{\sum_j \lambda_j f_j(q'_{t-1}, q'_t, x_i, t)} \beta(t, q'_t) \sum_{q'_1 \in Q, \dots, q'_{t-2} \in Q} \left(\prod_{u=1}^{t-1} e^{\sum_j \lambda_j f_j(q'_{u-1}, q'_u, x_i, u)} \right) \\
 &= \frac{1}{Z(x_i, \lambda)} e^{\sum_j \lambda_j f_j(q'_{t-1}, q'_t, x_i, t)} \beta(t, q'_t) \alpha(t-1, q'_{t-1}) \\
 &= \frac{1}{Z(x_i, \lambda)} \alpha(t-1, q'_{t-1}) e^{\sum_j \lambda_j f_j(q'_{t-1}, q'_t, x_i, t)} \beta(t, q'_t)
 \end{aligned}$$

Linear-chain CRF

学習(前向き後ろ向きアルゴリズム)

- 前向き後ろ向きアルゴリズムによる目的関数 $L(\lambda)$ に対するパラメータ λ の勾配

$$\begin{aligned}\frac{\partial L(\lambda)}{\partial \lambda_j} &= \sum_{i=1}^N \sum_{t=1}^{T_i} f_j(q_{i(t-1)}, q_{it}, x_i, t) - \sum_{i=1}^N \sum_{t=1}^{T_i} \sum_{q'_{t-1} \in Q, q'_t \in Q} f_j(q'_{t-1}, q'_t, x_i, t) p(q'_{t-1} q'_t | x_i) \\ &= \sum_{i=1}^N \sum_{t=1}^{T_i} f_j(q_{i(t-1)}, q_{it}, x_i, t) \\ &\quad - \sum_{i=1}^N \sum_{t=1}^{T_i} \sum_{q'_{t-1} \in Q} \sum_{q'_t \in Q} f_j(q'_{t-1}, q'_t, x_i, t) \frac{1}{Z(x_i, \lambda)} \alpha(t-1, q'_{t-1}) e^{\sum_j \lambda_j f_j(q'_{t-1}, q'_t, x_i, t)} \beta(t, q'_t)\end{aligned}$$

まとめ

- 系列ラベリングのための確率的識別モデル
 - Linear-chain CRF
 - 解析 (ビタビアルゴリズム)
 - 学習 (前向き後ろ向きアルゴリズム)



今までの講義資料

<http://aiweb.cs.ehime-u.ac.jp/~ninomiya/iips/>