[招待講演]

自然言語処理におけるベイズ統計

持橋大地

ATR 音声言語コミュニケーション研究所 /

NICT

daichi.mochihashi@atr.jp

電気情報通信学会 ニューロコンピューティング研究会

2006-10-11, NAIST

はじめに

- 自然言語処理 · · · 自然言語 [英語, 日本語, 中国語, ラテン語, · · ·] の計算機での処理と理解
  - 現在の主流は, 統計的自然言語処理
    - 大量のコーパスに基づく言語の統計モデル
- 自然言語処理の特徴:
  - データが離散的 ($\leftrightarrow$ ベクトル空間)
  - 超高次元, かつ超スパース ($\leftrightarrow$ DNA 系列など)

  $\Downarrow$
- 対象は自然言語に限られない
  - 高次元かつ離散的なデータは, 実は Ubiquitous
    1. 映画, 音楽, 書籍, . . . などの購入データ
    2. 人々の間のメールのやり取りの時系列
    3. Web ページの間に張られたハイパーリンク構造
    4. ゲームなどの手

       · · ·
  - これらの処理の基礎理論ともなっている

Overview

- 自然言語処理における問題
  - 教師あり学習タスク / 教師なし学習タスク
- 教師あり学習 (識別モデル) のベイズ推定
  - 形態素解析, 構文解析, 係り受け解析, . . .
- 教師なし学習 (生成モデル) のベイズ推定
  - ナイーブベイズから DM, LDA へ
  - LDA と画像および音楽
- 自然言語のベイズモデルの未来
- 結論とまとめ

自然言語処理における問題

- 統計的自然言語処理の主なタスク
  - 統計的機械翻訳
  - 形態素解析 (eg. Chasen/MeCab)
  - 係り受け解析 (eg. Cabocha), 構文解析
  - 文書要約, 意見抽出, 質問応答
  - 対話や独話, テキストのモデル化
- 応用として, リンク解析, 協調フィルタリング
- 学習用言語データの種類
  - タグ付きコーパス
  - タグなし生言語データ
  - ユーザーが勝手にタグをつけてくれたデータ

タグ付きコーパスと識別モデル

- 言語に, 内観による正解タグが付いているデータ
  - 構文解析結果
  - 形態素解析結果
  - 言葉の間の照応関係
  - 評価表現 (よい評価/悪い評価)
- 詳細は, http://cl.naist.jp/ –> 自然言語データに関する情報 など
- 正しいタグ付きデータを作成する労力は膨大だが, タスクによっては不可欠
- 学習の目標は, 未知データに対してその持つべきタグを正しく推定すること
  - 構文解析タグ/形態素解析タグ
    - 探索空間は指数的に爆発する
  - この意見は + か − か?
  - 識別結果は離散的

タグ無しコーパスと生成モデル

- タグを持たない生言語データ
  - 実質的にタグのついているものもある (次ページ)
  - 深い言語的構造は, 推定が難しい
- Web や CD-ROM などから, 大量に集めてくることが可能
- 隠れクラスに基づく言語の生成モデル
  - 文書モデル, 文モデル (n-gram など)
  - 「タグ」に相当するものを隠れ変数とする
- 学習の目標は, 未知のタグなしデータの予測性能を上げること
  - 言語表現として適切なものに高い確率
  - 言語の高次元空間での, Density Modeling

ユーザによるタグ付き言語データ

- タグ付きコーパス以外にも, ユーザーが勝手にタグを付けてくれたデータがある
  - ブログ記事についたカテゴリ情報
  - Amazon の評価☆の数と, 評価テキスト
  - “文書” もある種の教師データ (その中で意味が一様)
  - その他, 探せば色々あるはず
- 欠点:
  - タグは必ず正解とは限らない
  - タグの言語的深さには限界もある
- 可能な範囲で, 自動的に付いたモダリティを活用する学習モデルがこれから有用
  - 現在の自然言語処理は, 前の二分野に分かれている

識別モデルのベイズ学習

1. 形態素解析/タギング
2. 構文解析
3. 係り受け (依存構造) 解析

形態素解析/タギング [1/3]

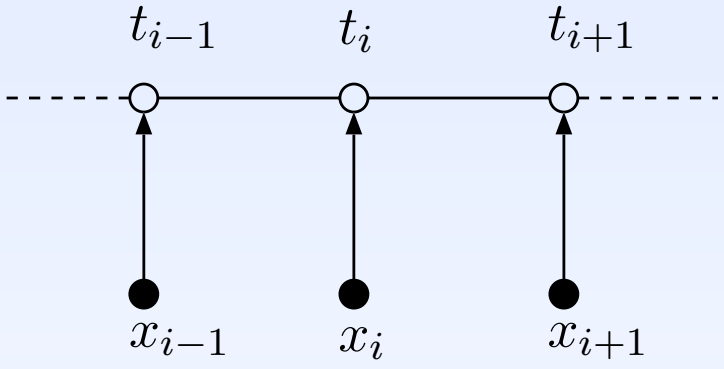
国境	名詞, 一般, *, *, *, *, 国境, コッキョウ, コッキョー
の	助詞, 格助詞, 一般, *, *, *, の, ノ, ノ
長い	形容詞, 自立, *, *, 形容詞・アウオ段, 基本形, 長い, ナガイ, ナガイ
トンネル	名詞, 一般, *, *, *, *, トンネル, トンネル, トンネル
を	助詞, 格助詞, 一般, *, *, *, を, ヲ, ヲ
抜ける	動詞, 自立, *, *, 一段, 基本形, 抜ける, ヌケル, ヌケル
と	助詞, 接続助詞, *, *, *, *, と, ト, ト
雪国	名詞, 一般, *, *, *, *, 雪国, ユキグニ, ユキグニ
で	助動詞, *, *, *, 特殊・ダ, 連用形, だ, デ, デ
あっ	助動詞, *, *, *, 五段・ラ行アル, 連用タ接続, ある, アッ, アッ
た	助動詞, *, *, *, 特殊・タ, 基本形, た, タ, タ
。	記号, 句点, *, *, *, *, 。, 。, 。
EOS	

テキスト $\mathbf{x}$ タグ系列 $\mathbf{t}$ [MeCab による解析例]

- 目標: 教師データを基に, 入力テキスト $\mathbf{x}$ から正しいタグ系列 $\mathbf{t}$ を求めること.
- タグは一つではなく, 階層的/スパース $\Longrightarrow$ HMM では不充分
  - Conditional Random Fields (Lafferty et al. 2001) とその拡張が, 現在標準的に使われている

形態素解析/タギング [2/3]

- 入力列を $\mathbf{x} = x_1 x_2 \cdots x_n$, タグ列を $\mathbf{t} = t_1 t_2 \cdots t_n$ とすると, 下の隠れグラフでのタグ系列の予測問題



  - t_{i-1} : 助詞, 格助詞, 一般, 「を」
  - t_{i+1} : 動詞, 自立, 基本型, 一段活用 「抜ける」
  - x_i : 「雪国」

- CRF のパラメータを $\mathbf{\Lambda}$ とすると, 次のタグ系列全体の確率を最大化

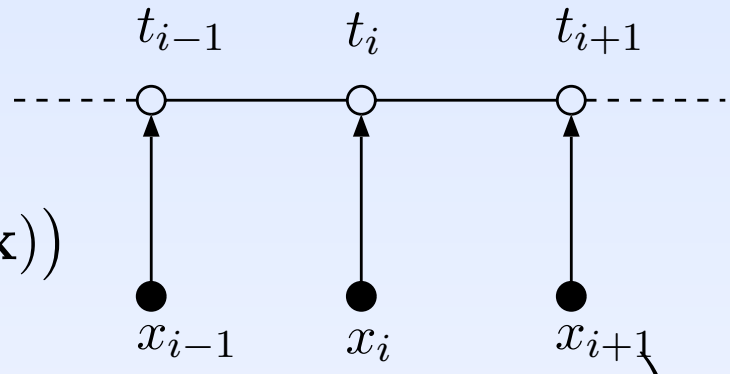
$$p(\mathbf{t}|\mathbf{x}, \mathbf{\Lambda}) = \frac{1}{Z(\mathbf{\Lambda})} \exp\big(\mathbf{\Lambda} \cdot F(\mathbf{t}, \mathbf{x})\big) \tag{1}$$

$$= \frac{1}{Z(\mathbf{\Lambda})} \exp\left(\sum_{i=1}^{n}\sum_{k=1}^{K} \lambda_k \underbrace{f(\langle t_i, x_i\rangle, \langle t_{i-1}, x_{i-1}\rangle)}_{\text{Clique potential}}\right) \tag{2}$$

  - $\mathbf{\Lambda} = (\lambda_1, \ldots, \lambda_K)$ は素性 k に対する重み
  - 系列全体のロジスティック回帰問題 (Forward-Backward で解く)

形態素解析/タギング [3/3]

- CRF の推定法: 最尤推定



$$p(\mathbf{t}|\mathbf{x},\mathbf{\Lambda}) = \frac{1}{Z(\mathbf{\Lambda})}\exp\big(\mathbf{\Lambda}\cdot F(\mathbf{t},\mathbf{x})\big) \tag{3}$$

$$= \frac{1}{Z(\mathbf{\Lambda})}\exp\left(\sum_{i=1}^{n}\sum_{k=1}^{K}\lambda_k f(\langle t_i, x_i\rangle, \langle t_{i-1}, x_{i-1}\rangle)\right) \tag{4}$$

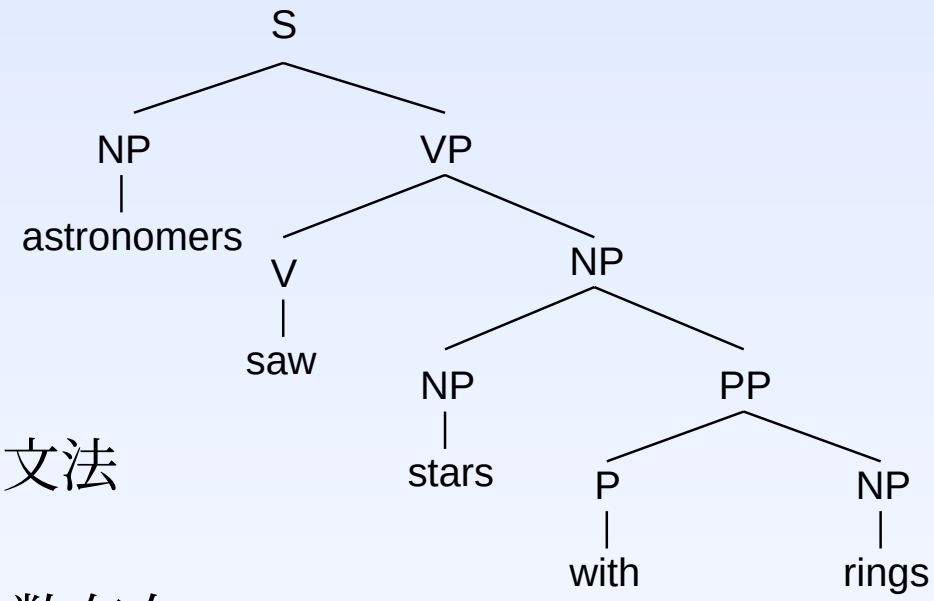
- 各 λ_k がオーバーフィットする危険性→ $\mathbf{\Lambda}$ に事前分布を与える

$$p_0(\mathbf{\Lambda}) \sim \mathrm{N}(0, \mathrm{diag}(\boldsymbol{\alpha})). \tag{5}$$

  - Forward-Backward の代わりに, Power EP (Minka 2004) で解く
    - α-ダイバージェンス最小化
  - 訓練データの各系列について, $\mathbf{\Lambda}$ の事後分布が得られる→経験ベイズ法で $\boldsymbol{\alpha}$ を推定
  - 通常の CRF よりも, 常に高性能 (Qi et al. 2005)

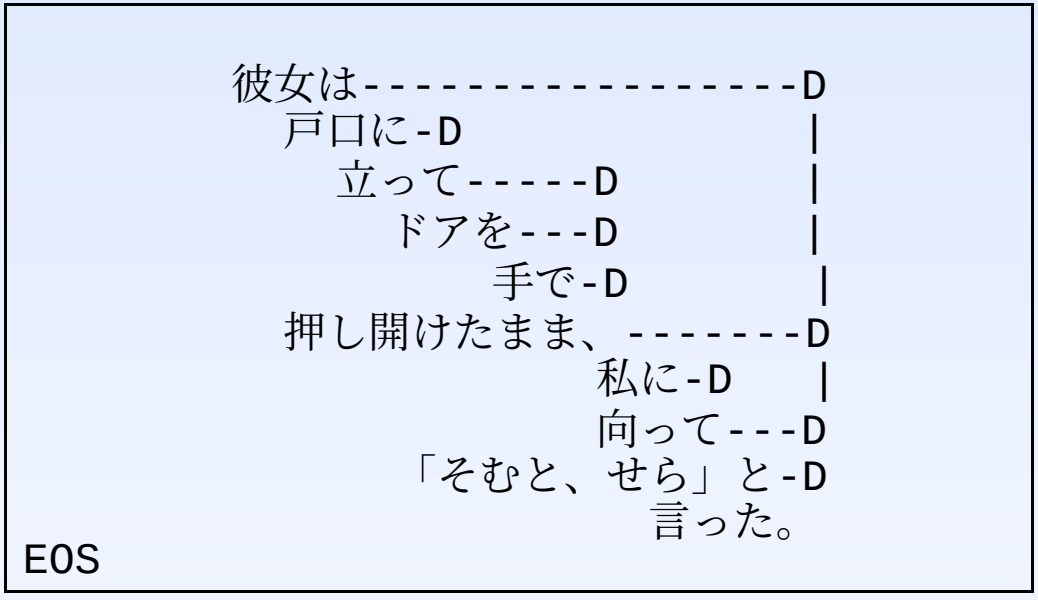
確率的文脈自由文法 (PCFG)

- 確率的文法規則:
  - S $\rightarrow$ NP VP [1.0]
  - NP $\rightarrow$ NP PP [0.4]
  - NP $\rightarrow$ astronomers [0.07]
  - NP $\rightarrow$ stars [0.1]
  - :



- 実際の膨大なテキストを解析する文法
  $\rightarrow$ 規則の数が巨大
  - ほとんど使われない規則が, 多数存在
  - オーバーフィット問題
- ベイズ解法: 規則 $A \rightarrow \alpha$ $(\sum_\alpha p(\alpha|A) = 1)$ [離散分布] に, ディリクレ事前分布を与える (栗原他 2004)
  - 変分ベイズ法による Forward-Backward
  - どの分岐規則についても, 同じ事前分布に従うのでよいか?
    (栗原他 2004 では, ハイパーパラメータ $u = 2$)

係り受け解析 [1/2]



[Cabocha による解析例]

- 文法を必要としない構文解析法として, 最近特に注目されている
  - 日本語以外の言語にも適用可能
- 学習の目標: 入力文 $\mathbf{x}$ に対し, 最適な係り受け木 $\mathbf{y}$ を求めること
  - 可能な係り受け木 $\mathbf{y}$ の探索空間は厖大
  - 総係り受けコスト $s(\mathbf{x}, \mathbf{y}) =$ 各係り受けコストの総和
    - 各係り受け関係は独立と仮定する

係り受け解析 [2/2]

- 入力文 $\mathbf{x}$ からの係り受け解析木 $\mathbf{y}$ の推定：

$$\hat{\mathbf{y}} = \arg\max_{y \in Y} s(\mathbf{x}, \mathbf{y}) \tag{6}$$

ここで, 解析コスト $s(\mathbf{x}, \mathbf{y})$ は

$$s(\mathbf{x}, \mathbf{y}) = \sum_{\langle i,j \rangle \in \mathbf{y}} d(i,j) \tag{8}$$
(7)

$$= \sum_{\langle i,j \rangle \in \mathbf{y}} \mathbf{w} \cdot \mathbf{F}(i,j) = \sum_{\langle i,j \rangle \in \mathbf{y}} \sum_{k} w_k f_k(i,j) \tag{9}$$

  - 正解の係り受け $\langle i, j \rangle$ を分類する perceptron
  - 重みベクトル $\mathbf{w}$ を学習 (次元数＝素性数 k は厖大)
- Bayes Point Machines による学習 (Corston-Oliver et al. 2006)
  - オンライン学習された perceptron の重み付き平均
  - 最近のマージン最大化法 (MIRA) とほぼ同精度
    - $\mathbf{w}$ の空間で Bayes point を求める $\leftrightarrow$ マージンを最大化する

自然言語処理の識別モデルのベイズ学習 (まとめ)

- ごく最近になって, 自然言語処理の識別モデルにもベイズ学習が導入されている
  - これまでは SVM, ME, Boosting などの分類器
- 目的 = パラメータの過学習緩和
  - モデルはベクトル空間
  - 事前分布はガウス分布や一様分布
  - 隠れ変数や階層モデルは無い
- 識別モデルの長所: 生成モデルが不要
  - 構文木や係り受けの生成モデル?
  - 複雑な言語タグの生成モデル?
    - 識別モデルは近似的な部分が多いが, versatile
    - **ただし, タグ付き教師データが必要**

自然言語の生成モデルのベイズ推定

1. Naïve Bayes 法
2. Dirichlet Mixtures (DM)
   - Unsupervised “Non-Naïve” Bayes 法
3. Latent Dirichlet Allocation (LDA)
4. DM/LDA の応用 (画像/音楽との統合モデル)

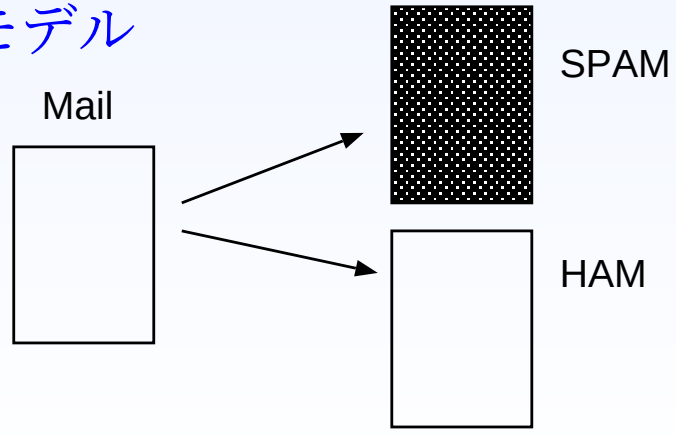
Naïve Bayes 法

- SPAM の判定法で有名 (Graham, 2002)
  - クラスは 2 個: $c = 1$(SPAM), $c = 0$(No SPAM = HAM)
- 学習データ: HAM/SPAM のタグつき電子メール
- 目標: メールのクラス判別
  - 文書 (メール)d について, クラス事後確率

$$p(c|d) \propto p(c)p(d|c) \tag{10}$$

を求めたい.

  - $p(d|c)$: 文書 d のクラス c からの生成モデル
- 具体的には..



Naïve Bayes 法 (2)

- “bag of words” の仮定
  - 単語がクラス (HAM/SPAM) だけに依存してバラバラに出現 (1-gram)
  - クラスごとの単語の生起確率 (1-gram 分布) $p(v|c)$
- 文書 d の生成確率

$$p(d|c) = \prod_{v \in V} p(v|c)^{n(d,v)} \tag{11}$$

  - $n(d, v)$ は単語 v が文書 d に出現した回数
- よって,

$$p(c|d) \propto p(c) \prod_{v \in V} p(v|c)^{n(d,v)} \tag{12}$$

  - Naive Bayes 法のパラメータ: $p(c), p(v|c)$ $(= p(v|0), p(v|1))$
  - パラメータはどうやって求める?

Naïve Bayes 法 (3)

- パラメータの求め方: 最尤推定

$$p(v|c) \propto \sum_{d \in c} n(d, v) \tag{13}$$

  - HAM/SPAM 別に計算した, 単語の生起分布
  - “money” が偶然, SPAM にしか出現しなかった

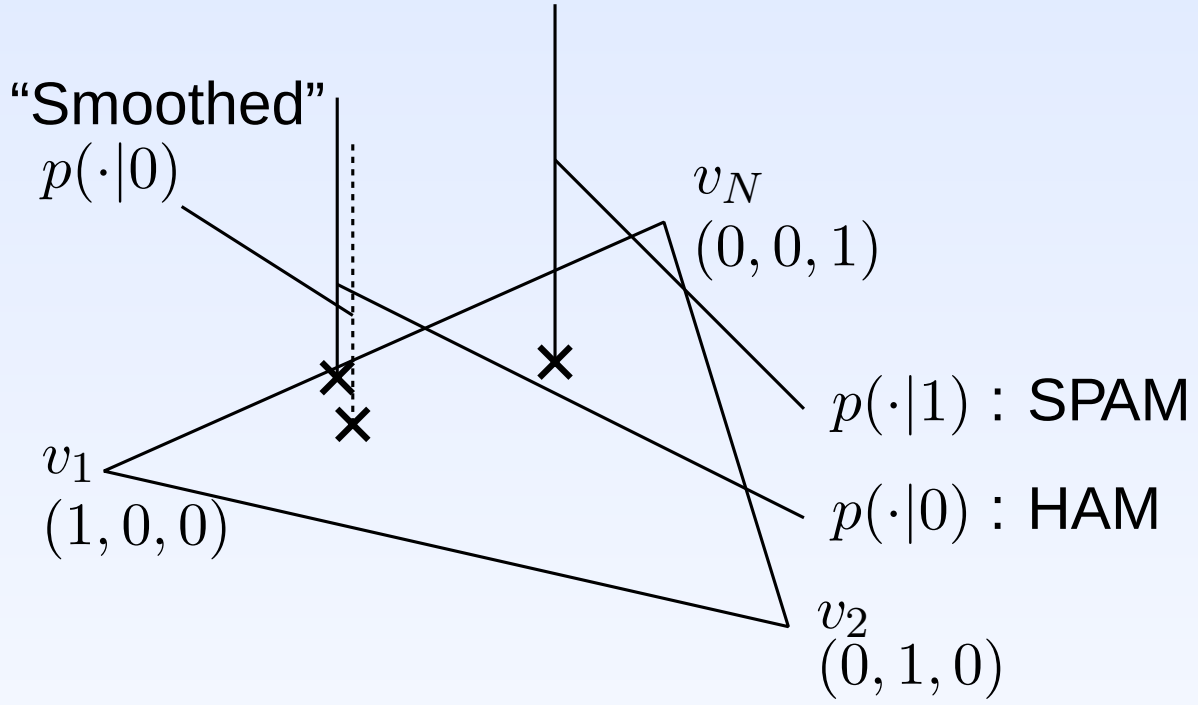
    ⇓

    友人からの “money” を含んだメールは絶対に SPAM に!

$$p(c = 0|d) \propto p(0)p(v_1|0) \cdots p(\text{money}|0) = 0 \tag{14}$$

- スムージングが必要
  - (Generalized) Laplace smoothing: $p(v|c) \propto \epsilon + \sum_{d \in c} n(d, v)$
  - 自然言語の単語数は厖大 $\cdots$ $\epsilon = 0.01$ でも, 100,000 語について加えると総和は 1,000
    - Uniform Dirichlet prior での, 事前の観測値
  - ϵ を決める方法は? $\rightarrow \times$.

Geometry of Naïve Bayes



- $p(v|c)$ は (例えば N=10000 次元上の) 離散分布

$$\mathbf{p} = (p(v_1|c), p(v_2|c), \ldots, p(v_N|c)) \; ; \quad \sum_v p(v|c) = 1 \qquad (15)$$

  なので, $(N-1)$-単体上の 1 点

- Dirac δ の高さが, $p(c)$ を反映
  - 点以外には, どこにも動けない

Dirichlet Mixtures (DM)

- $\mathbf{p} = p(v|c)$ 自体が曖昧 $\cdots$ 事前分布を与える

$$\mathbf{p} \sim \mathrm{Dir}(\boldsymbol{\alpha}_c) \quad ; \ \boldsymbol{\alpha}_c = (\alpha_{c1}, \ldots, \alpha_{cN}) \tag{16}$$

- $\mathbf{p}$ を積分消去 (Polya 分布)

$$p(c|d) \propto \ p(c)p(d|c) \tag{17}$$

$$= \ p(c) \int p(d|\mathbf{p})p(\mathbf{p}|c)d\mathbf{p} \tag{18}$$

$$= \ p(c) \frac{\Gamma(\sum_v \alpha_{cv})}{\prod_v \Gamma(\alpha_{cv})} \prod_{v \in V} \frac{\Gamma(\alpha_{cv} + n(d,v))}{\Gamma(\alpha_{cv})} \tag{19}$$

  - 単語毎の適応的なスムージング (α_{cv})
  - DM のパラメータ: $p(c)$, α_{cv} for $c = 1 \ldots C$, $v = 1..N$
  - パラメータ推定法は??

Dirichlet Mixtures (DM) [2]

- パラメータ推定: EM-Newton アルゴリズム

  E step:

$$p(c|d) \propto\ p(c)\ \frac{\Gamma(\sum_v \alpha_{cv})}{\prod_v \Gamma(\alpha_{cv})} \prod_v \frac{\Gamma(\alpha_{cv}+n(d,v))}{\Gamma(\alpha_{cv})} \tag{20}$$

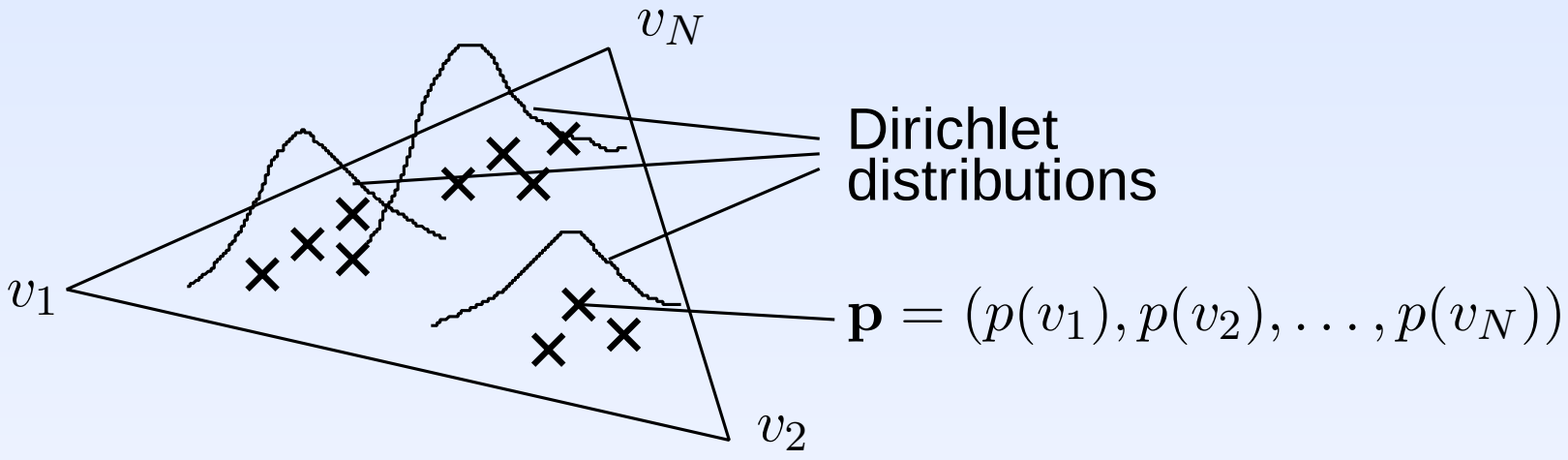
  M step:

$$p(c) \propto\ \sum_d p(c|d)\,, \tag{21}$$

$$\alpha'_{cv} =\ \alpha_{cv}\ \cdot\ \frac{\sum_d p(c|d)\ n(d,v)/(n(d,v)+\alpha_{cv}-1)}{\sum_d p(c|d)\ \sum_v n(d,v)/(\sum_v n(d,v)+\alpha_{cv}-1)} \tag{22}$$

  - HAM/SPAM の場合は, $p(c|d)$ は既知 (supervised)
  - 通常, DM の隠れクラス数 $C \leq 500$ 程度
    - DM のパラメータ数: $C-1+NC \sim$ 数 100 万個.

Geometry of Dirichlet Mixtures



- $\mathbf{p} = p(v|c)$ を点推定ではなく, ベイズ推定
- 単語単体上の密度推定 (混合モデル)
  - 単語単体上のすべての領域をモデル化
- 文書 d は, 一つの山の上の $\mathbf{p}$ から生成

  $\Downarrow$

  トピックの組み合わせ? $\rightarrow$ LDA
  - 途上国の経済政策の, 教育への影響
  - インドネシアの民族音楽

Latent Dirichlet Allocation (LDA) [1]

- 文書は 1 つのクラス (トピック) ではなく, トピック分布 $\boldsymbol{\theta}$ を持つ
- Bag of words だが, 単語ごとに異なったトピック
- 文書 d の生成:
  1. Draw $\boldsymbol{\theta}_d \sim \mathrm{Dir}(\boldsymbol{\alpha})$.
  2. For $i = 1 \cdots n$,
     - Draw $c_{di} \sim \mathrm{Discrete}(\boldsymbol{\theta}_d)$.
     - Draw $v_{di} \sim p(v|c_{di})$.
- $p(v|c) = \boldsymbol{\beta}$ とおくと, 全体の生成確率は

$$p(d|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \int p(d|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\boldsymbol{\alpha}) d\boldsymbol{\theta} \tag{23}$$

$$= \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \int \left(\prod_k \theta_k^{\alpha_k - 1} \right) \prod_{n=1}^{N} \sum_{k=1}^{K} \theta_k \beta_{k v_n} d\boldsymbol{\theta} \tag{24}$$

  - この積分は intractable なため, 推定には近似が必要になる

Latent Dirichlet Allocation (LDA) [2]

- ベイズ学習: 変分ベイズ法
  - E step: $p(c|v_{di}) \propto \beta_{cv_{di}} \exp\big(\Psi(\theta_{dc})\big)$
  - M step: $\theta_{dc} = \alpha_c + \sum_{i=1}^{n} p(c|v_{di})$
    - 単語毎にトピックの事後分布が求まる
    - 文書の事後分布 $\boldsymbol{\theta}_d$ は, ディリクレ事前分布＋単語の事後分布.
    - http://chasen.org/˜daiti-m/dist/lda/
- ベイズ学習: MCMC (Gibbs)
  - $p(c|v_{di}) \propto p(v_{di}|c)p(c|d)$

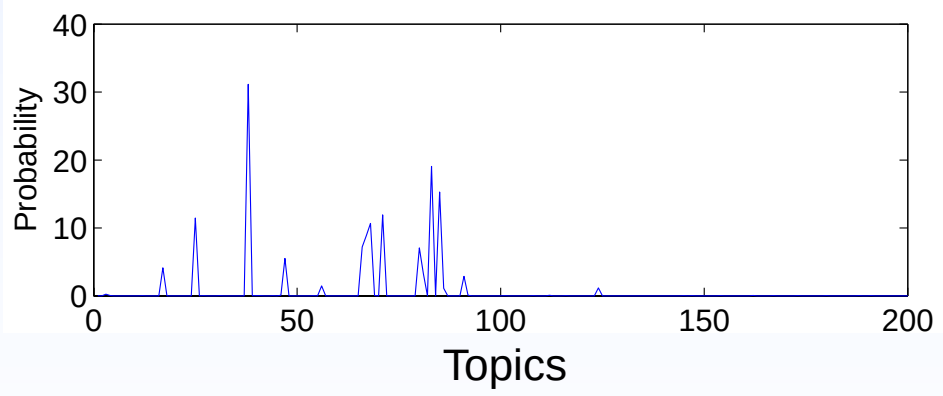
$$\sim \frac{n^{v_{dn}}_{-dn,c} + \beta}{n^{\cdot}_{-dn,c} + V\beta} \cdot \frac{n^{d}_{-dn,c} + \alpha}{n^{d}_{-dn,\cdot} + K\alpha}.$$

  - http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm (Topic modeling Toolbox)

LDA の学習例 (事後分布)

雪国
国境の長いトンネルを抜けると雪国であった。夜の底が白くなった。
信号所に汽車が止まった。
向側の座席から娘が立って来て、島村の前のガラス窓を落した。
雪の冷気が流れこんだ。
. . .

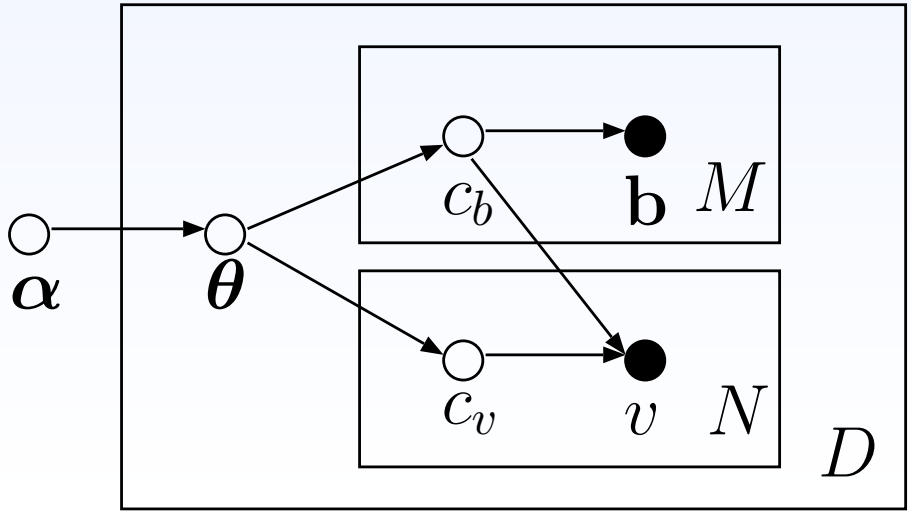
- 事後確率 $p(c|v_{di})$ 最大のトピックで色分け
- 文書冒頭のトピック分布:



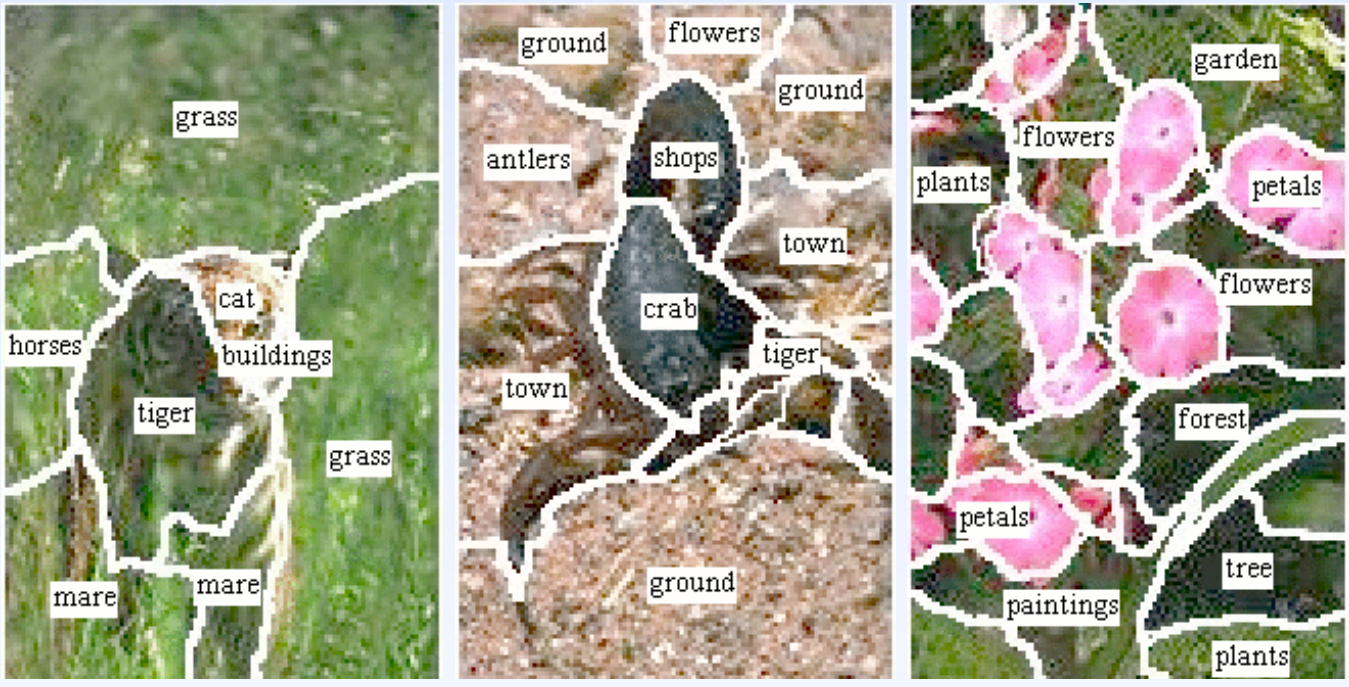
- 毎日新聞 2000 年度版のデータで学習 (10 万記事, 135MB)
  学習時間: 約 20 時間, 総語彙: 71000 語

Matching Words and Pictures [1]

- LDA は文書ごとの混合モデル
  - $\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha})$ が文書ごとの混合比
  - 単語 1-gram 分布の混合に限られない
- 画像–言葉の結合モデル (Barnard et al. 2003)
  - キャプション付き Corel 画像データベース
    - Web ページの画像と周囲のテキスト (Google 画像検索)
    - 映画や TV 番組の字幕 (& 音声)
  - 1 つの $\boldsymbol{\theta}$ から, 画素 b と単語 v を両方生成



Matching Words and Pictures [2]

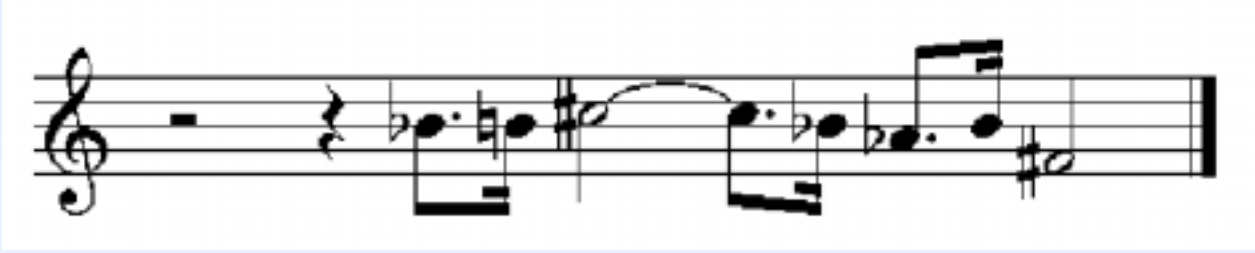


- 画像をあらかじめ領域分割しておき, 画像に隠れたトピック分布 $\boldsymbol{\theta}$ と言葉との対応付けを計算
  - 各画像領域に, 単語の確率分布 $p(v|b)$ (Max のみ表示)
- さらに進んだモデル: Image Parsing (Tu et al. ICCV 2003) http://civs.stat.ucla.edu/

音楽と歌のモデル [1]

- “Name That Song!” Brochu and Freitas, NIPS 2002

[_*3/4 b&1*3/16 b1/16 c#2*11/16 b&1/16 a&1*3/16 b&1/16 f#1/2



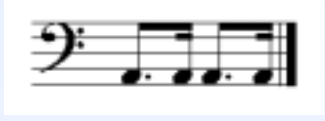
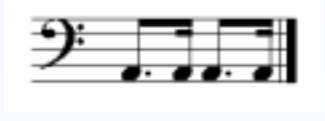
  - 音の遷移: 相対音階バイグラム $\mathbf{M}_k$ + 歌詞テキスト $\mathbf{T}_k$
  - 曲 $\mathbf{X}_k = \{\mathbf{M}_k, \mathbf{T}_k\}$

$$\mathbf{X}_k|\boldsymbol{\theta} \sim \sum_c p(c) \prod_j p(j|c)^{I_j(\mathbf{M}_{k,0})} \prod_j \prod_i p(j|i,c)^{M_{i,j,k}} \prod_v p(v|c)^{T_{v,k}} \tag{25}$$

$$\boldsymbol{\theta} = \{p(c), p(j|c), p(j|i,c), p(v|c)\} \tag{26}$$

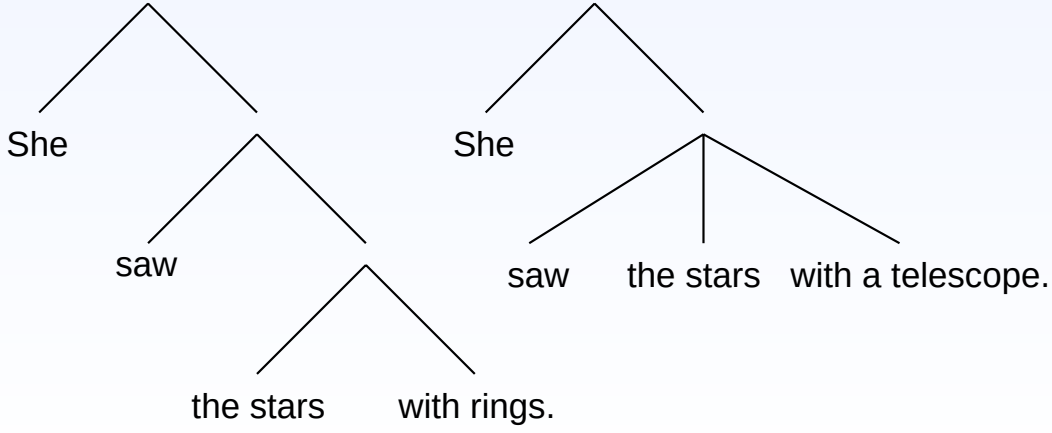
音楽と歌のモデル [2]

- $p(c|\mathbf{X}_k)$ を EM で計算
- 曲の検索 $\cdots$ 歌詞のみ $(\mathbf{T}_k)$,
  旋律のみ $(\mathbf{M}_k)$,
  歌詞+旋律 $(\mathbf{T}_k + \mathbf{M}_k)$
- かなりラフな近似
  - 歌詞は 1-gram で生成
  - 旋律は, 隣りあった音の遷移のみ
    - FFT と結合した生成モデル?
    - Polyphony の扱い?

QUERY	RETRIEVED SONGS
come on, come on, get down	*Erksine Hawkins – Tuxedo Junction* *Moby – Bodyrock* *Nine Inch Nails – Last* *Sherwood Schwartz – 'The Brady Bunch' theme song*
	*The Beatles – Got to Get You Into My Life* *The Beatles – I'm Only Sleeping* *The Beatles – Yellow Submarine* *Moby – Bodyrock* *Moby – Porcelain* *Gary Portnoy – 'Cheers' theme song* *Rodgers & Hart – Blue Moon*
 *come on, come on, get down*	*Moby – Bodyrock*

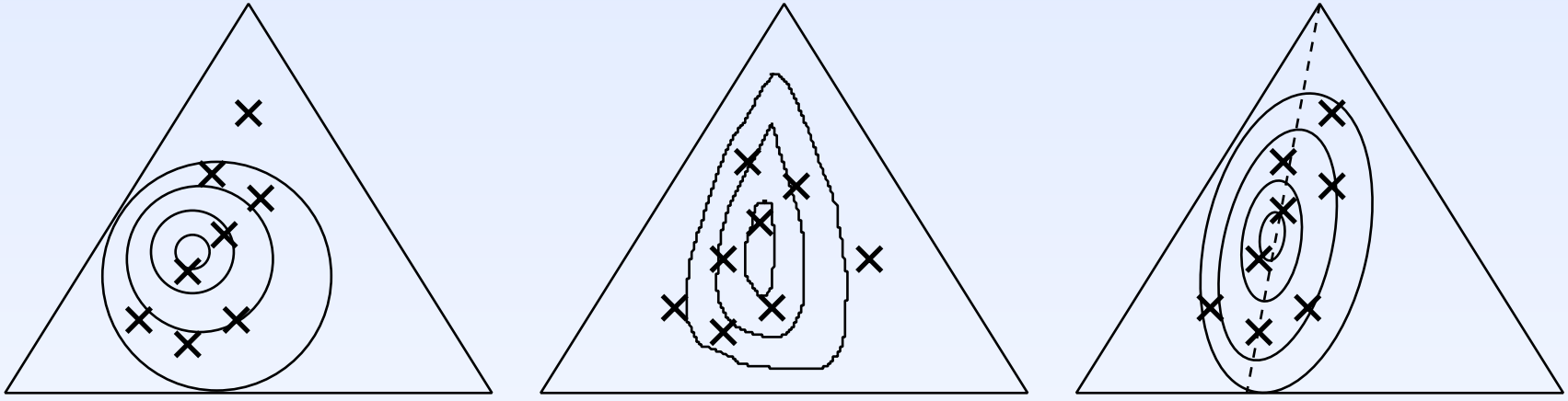
自然言語のベイズモデルの最近の発展 [1]

- “Bag of words” からの脱却
  - 2-gram, 3-gram, 4-gram, . . . のベイズモデル (Teh 2006)
  - n-gram 自体に, 限界がある ($n \leq 6$ 程度で飽和; Goodman 2001)
- 構文木 (係り受け構造) の生成モデル?
  - 今までは, 0/1 の素性ベクトル化して識別モデルで取り扱い
  - Random Trees, Tree distributions
    - Leaf の情報をどのように入れるか?



自然言語のベイズモデルの最近の発展 [2]

- Dirichlet 分布 ∼ 等分散 Gaussian (ベクトル空間での)



$$\mathrm{Dir}(\boldsymbol{\theta}|\boldsymbol{\alpha}) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_k \theta_k^{\alpha_k - 1} \propto \prod_k \theta_k^{\alpha_k - 1}. \qquad (27)$$

- Nonparametric Bayes 法
  - Dirichlet process およびその拡張
  - 上のような, 連続的な分布関数を用いない

自然言語のベイズモデルの未来

- 自然言語処理でのベイズ学習の利点
  - 識別モデル
    - 過学習抑制
    - モデルの学習基準が自然 (?)
  - 生成モデル
    - 隠れ変数による階層ベイズモデル
    - 連続的対象との結合モデル
  - 識別モデル + 生成モデル?
- 生成モデル ... 教師データを必要としない
  - 多くの実際的な生データを, そのまま扱える
  - 深い言語的構造を表現する確率モデルは, 現時点では困難
  - 結果が離散的でなくてよい
    - 文書の書かれた時間帯分布 / ブログの地域推定分布
- 画像, 音楽 (音声), ... (+ 連続的な時間データ) と組み合わせることにより, 自然言語処理の枠を超えたモデルが可能になる

End

- ありがとうございました.