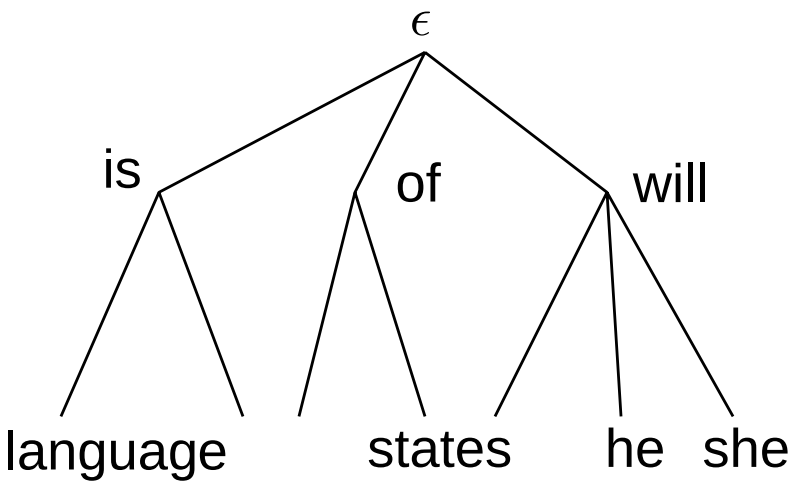
Pitman-Yor 過程に基づく可変長 *n-gram* 言語モデル

持橋大地 [□]　隅田英一郎
ATR 音声言語コミュニケーション研究所 /
情報通信研究機構
daichi.mochihashi@atr.jp

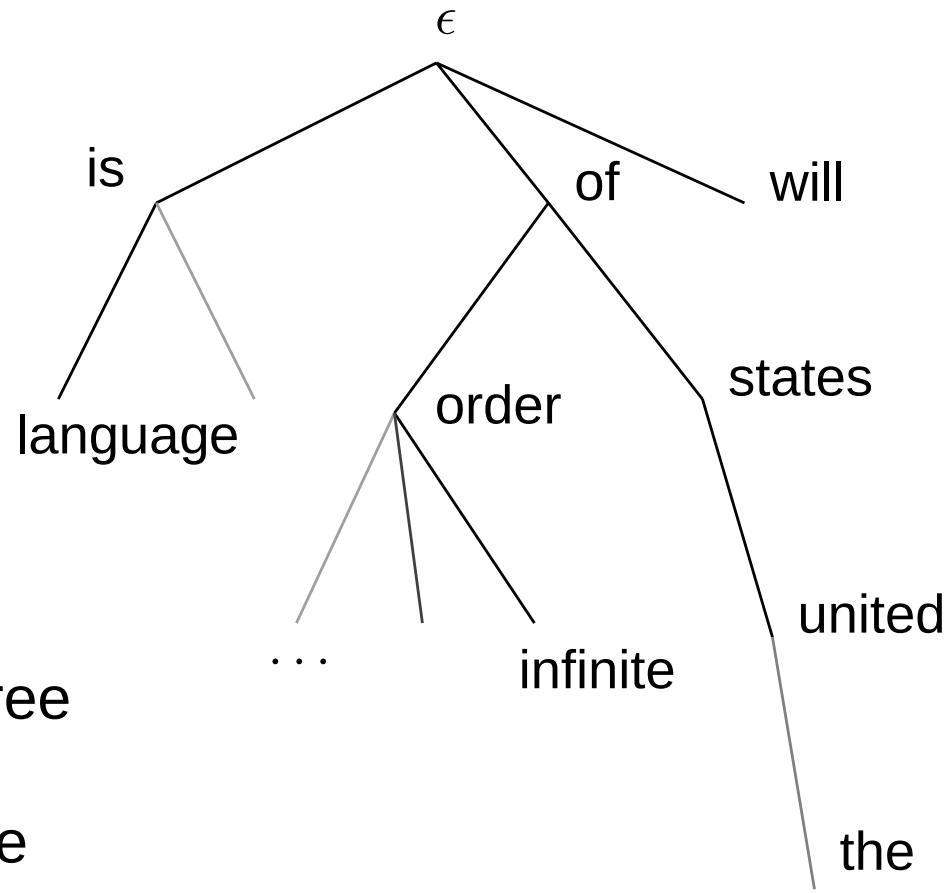
IPSJ SIGNL 2007-NL-178
2007/3/29 (Thu)

[□] 2007 年 4 月より, NTT コミュニケーション科学基礎研究所 PD

Overview



これまでの n グラム言語モデル



ベイズ可変長 ∞-グラム言語モデル

- 深さ $(n-1)$ の決まった Suffix Tree
  $\Downarrow$
  深さ無限の, 確率的な Suffix Tree
- Suffix Tree の事前 $\longrightarrow$ 事後確率分布
  - 木をどうやって推定するか?
- 無限可変長 Markov モデルの一般理論

n グラムモデルとその問題 (1)

- n グラムモデル $\cdots$ 言語の予測モデル
  - $p(\text{話す} \mid \text{彼女 が}) = 0.2,\ p(\text{処理} \mid \text{自然 言語}) = 0.7,$
    $p(\text{見る} \mid \text{彼女 が}) = 0.1 \ldots$
  - 文の確率を, 予測確率の積に分解 [マルコフ過程]
    $p(\text{彼女 が 見る 夢}) =$
    $p(\text{彼女}) \times p(\text{が} \mid \text{彼女}) \times p(\text{見る} \mid \text{彼女 が}) \times p(\text{夢} \mid \text{が 見る})$

n グラムモデルとその問題 (1)

- n グラムモデル $\cdots$ 言語の予測モデル
  - $p(\text{話す} \mid \text{彼女 が}) = 0.2,\ p(\text{処理} \mid \text{自然 言語}) = 0.7,$
    $p(\text{見る} \mid \text{彼女 が}) = 0.1 \ldots$
  - 文の確率を, 予測確率の積に分解 [マルコフ過程]
    $p(\text{彼女 が 見る 夢}) =$
    $p(\text{彼女}) \times p(\text{が} \mid \text{彼女}) \times p(\text{見る} \mid \text{彼女 が}) \times p(\text{夢} \mid \text{が 見る})$
- 各単語は, 前の $(n-1)$ 語の単語のみに依存する

$$p(w_1 \ldots w_T) = \prod_{t=1}^{T} p(w_t | w_{t-1} w_{t-2} \cdots w_1) \tag{2}$$

$$\simeq \prod_{t=1}^{T} p(w_t | \underbrace{w_{t-1} \cdots w_{t-(n-1)}}_{n-1 \text{ 語}}) \tag{2}$$

  - n グラムモデル = 前の $(n-1)$ 語を状態としたマルコフモデル

n グラムモデルとその問題 (2)

$$p(w_1 \dots w_T) \simeq \prod_{t=1}^{T} p(w_t | \underbrace{w_{t-1} \cdots w_{t-(n-1)}}_{n-1 \text{ 語}}) \tag{2}$$

- n グラムモデル = 単語の総数 V に対して, V^{n-1} 個の状態数
  - 指数的に爆発する
  - $V = 10000$ のとき, $V^2 = 100000000$ (3 グラム), $V^3 = 1000000000000$ (4 グラム), $\cdots$
- 通常, $n = 3 \sim 5$ 程度が限界
  - Google 5 グラムは gzipped 24GB, $V = 13653070$
    - $V^2 = 186406320424900$ (3 グラム)
    - $V^3 =$ (天文学的な数) (4 グラム)
  - しかし, そもそも…

n グラムモデルとその問題 (3)

- n グラムモデルは, 単純な $(n-1)$ 次のマルコフ過程
  ＝直前 $(n-1)$ 語を丸覚え
  - 3 グラム, 4 グラム, 5 グラム, ... のデータはノイズだらけ
    - に 英語 が
    - の が # #
    - は 修了 宮本 益
    - が あり 独自 に 法医学
    - は ゼネラル・モーターズ GM や
  - 空間計算量・時間計算量の点でも, 非常に無駄が大きい
- 言語的に意味がある n グラムは何か?

n グラムモデルとその問題 (4)

- 現実の言語データには, 3 グラム, 5 グラムを超えるような長い系列が頻繁に現れる
  - the united states of america
  - 京都 大学 大学院 情報 学 研究科
  - 東京 地検 特捜 部 の 調べ に よる と
  - そんな 事 より 1 よ 、 ちょいと 聞い て くれ よ 。…
- チャンク (句) とみなして一単語にする方法もあるが…
  - 人間の主観的な “正解” に依存
  - どこまでを句とすればよいか [境界は 1/0 か?]
  - 上のような, 慣用句などのフレーズを全て列挙するのは不可能
- バイオインフォマティクス等とも共通する問題
  - DNA, アミノ酸, タンパク質などの系列
    - “正解” が自明ではない

可変長 n グラム言語モデル

- n グラムの n を文脈に応じて可変長にできないか?
  - “可変長 n グラム言語モデル” … 音声言語分野を中心に提案
  - 踊堂, 中村, 鹿野 (1999), Stolcke (1998), Siu and Ostendorf (2000), Pereira et al. (1995) など
  - しかし,

    ⇓
- これまでの “可変長 n グラム言語モデル”= 巨大な n グラムモデルの枝刈りによる方法
  - 指数的に爆発する最大モデルを, 事前に作っておく必要
    - 可変長モデルの意図と矛盾
    - n グラムを減らすことはできても, 増やすことはできない
  - MDL, KL ダイバージェンスなどによる枝刈り
    - 性能があまり悪化しないように, 余分な n グラムを減らす
    - 基準はモデルとは別で, 後付け

可変長 n グラム生成モデル

- なぜ, 正しい可変長生成モデルが存在しなかったか?

  ⇓

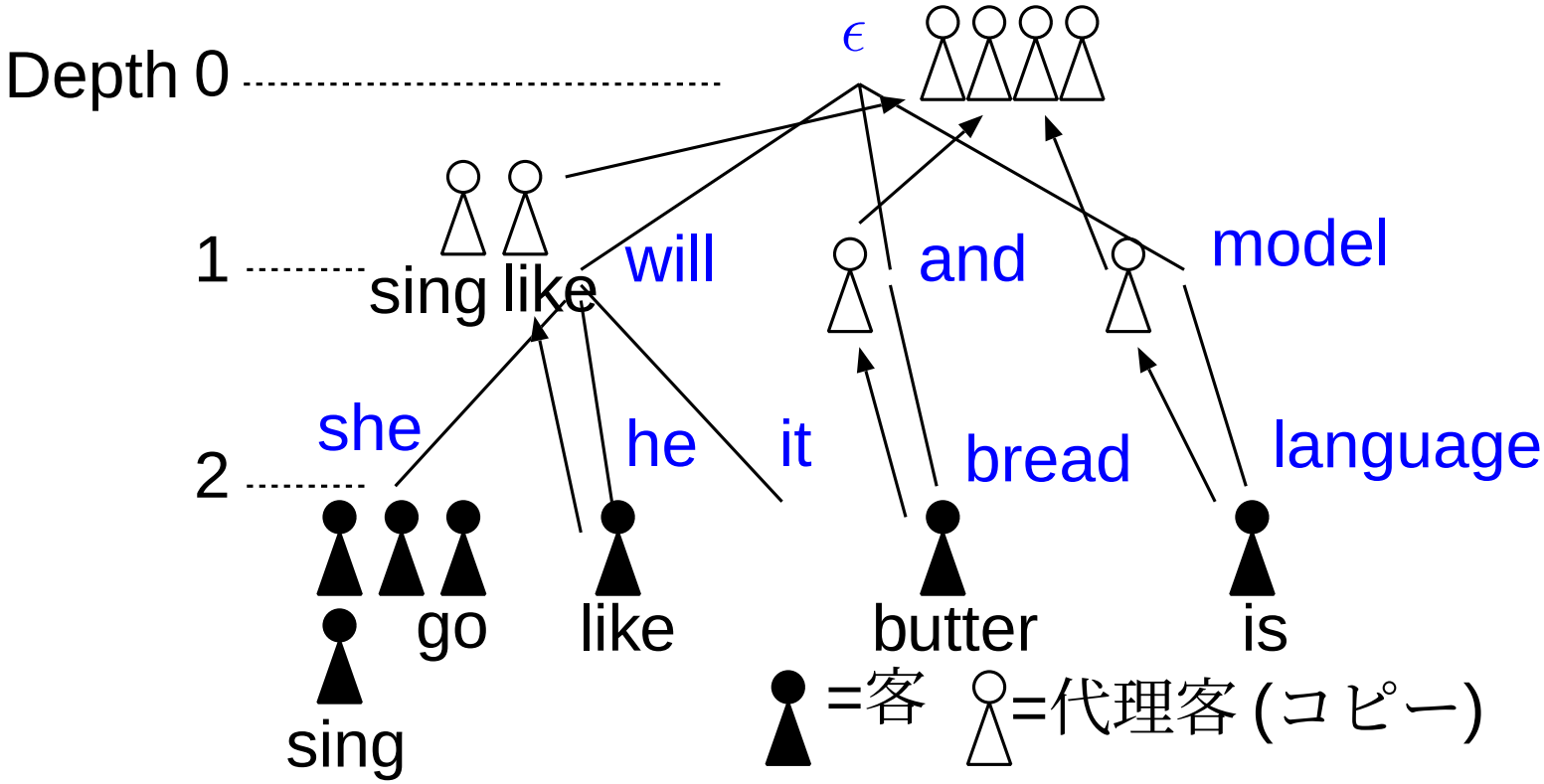
  - n グラム分布は, n が大きくなるほどスパース
  - n グラム分布は, $(n-1)$ グラム分布に依存
  - これを階層的に生成する理論的なモデルは存在しなかった
- しかし..

ベイズ n グラム言語モデル

- Hierarchical Pitman-Yor Language Model (HPYLM) (Yee Whye Teh, 2006)
  - 階層ベイズの考えに基づく, n グラム言語モデルの完全なベイズ生成モデル
  - Kneser-Ney スムージングと同等以上の性能 (K-N はその近似)
  - 階層ディリクレ過程 (HDP) の拡張
- Pitman-Yor 過程 (=2-パラメータポアソン=ディリクレ過程 $\mathrm{PD}(\alpha, \theta)$ (Pitman and Yor 1997)) を階層化
  -  Marc Yor (Université Paris VI, France)
  -  Jim Pitman (Dept. of Statistics, Berkeley)
  - 測度論に基づく数学的な詳細については, http://chasen.org/~daiti-m/paper/svm2006-hpylm.pdf を参照のこと.

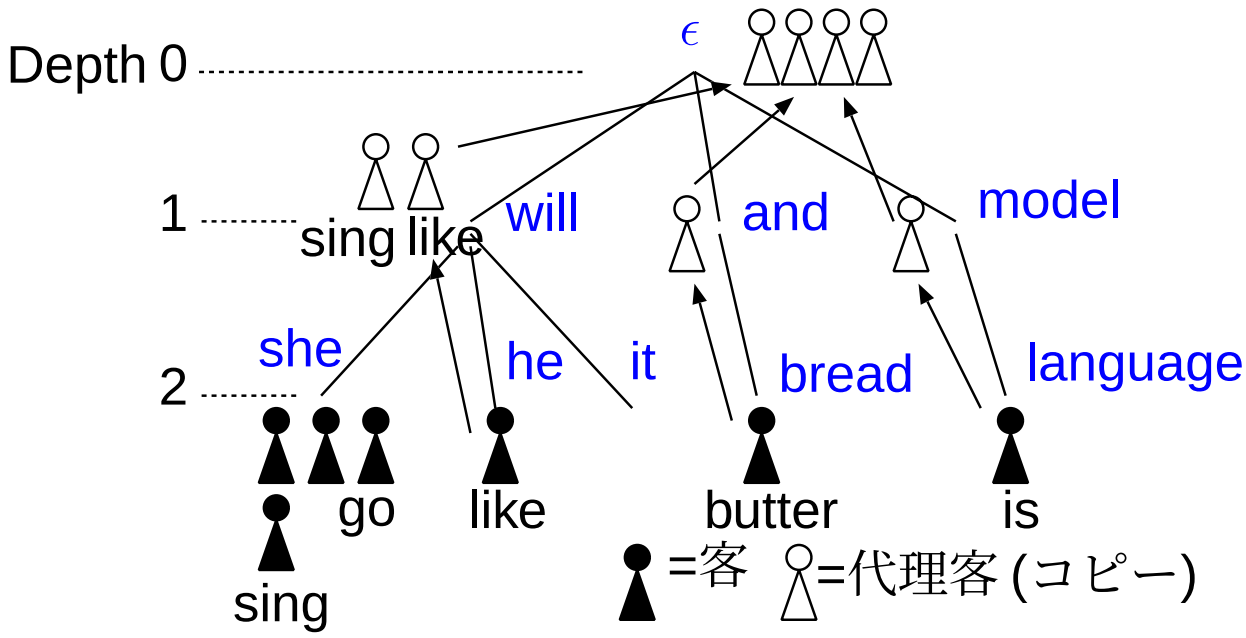
HPYLM (1)

- n グラム分布は, 深さ $(n-1)$ の Suffix Tree で表せる
- 例として, トライグラムを考える



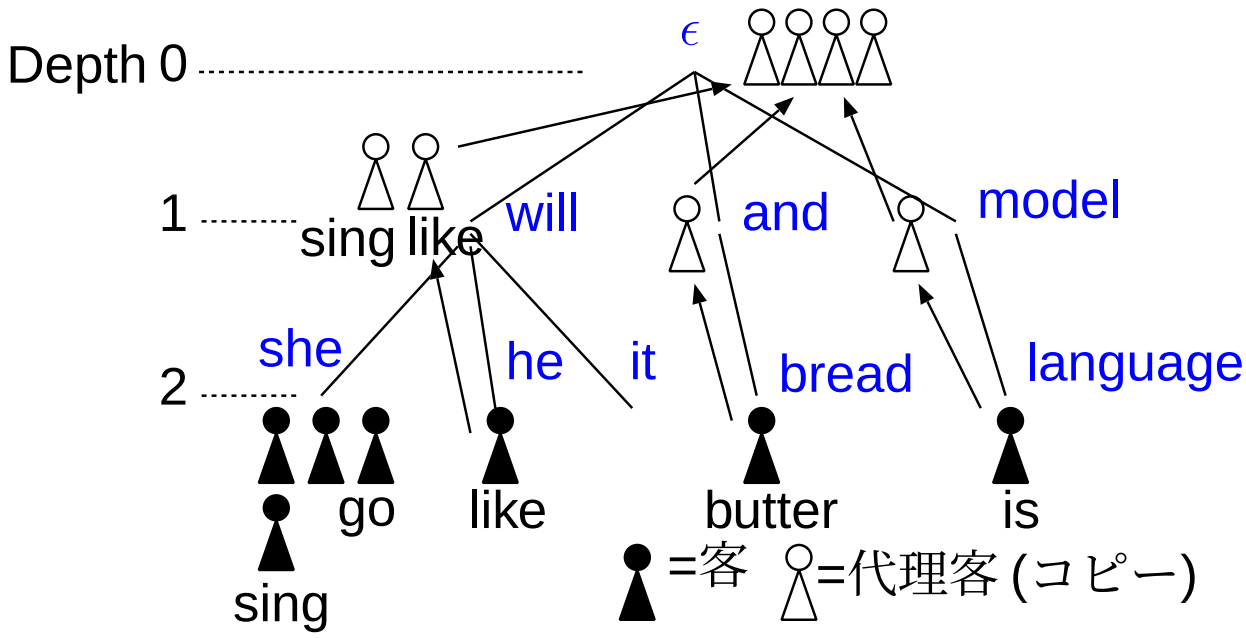
- ‘she will’→‘sing’ を予測…木を $\epsilon \rightarrow$ will $\rightarrow$ she の順にたどる
- 止まった, 深さ 2 のノード (トライグラム) から, sing の確率を計算

HPYLM (2)



- ノードの持つ客 (単語カウント) の分布から, $p(\text{sing}|\text{she will})$ を計算 $\rightarrow$ $p(\text{like}|\text{she will})$ はどうする?
  - 'like' のカウントがない
- 客のコピー (代理客) を上のノードに確率的に送る
  - 'he will like' から送られた上のノードの客 'like' を使って, バイグラムと補間して確率を計算

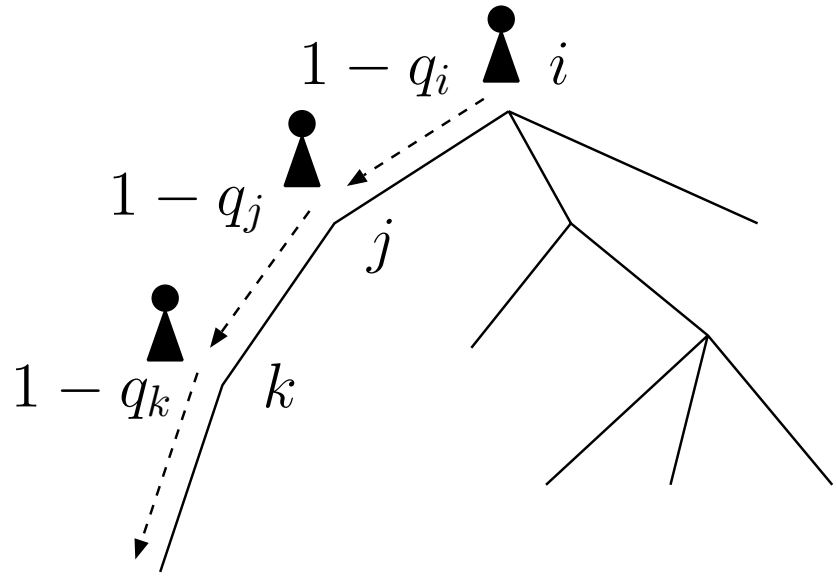
HPYLM から可変長モデルへ



- HPYLM の問題…客 = データのカウントがみな, 深さ 2 のノードに集まるのでいいか?
  - ‘will like’ は本当は深さ 1 (バイグラム) で十分
  - ‘the united states of america‘はもっと深いノードが必要

  ⇓
- 客を違った深さに追加する確率過程.

Variable-order Pitman-Yor Language Model (VPYLM)



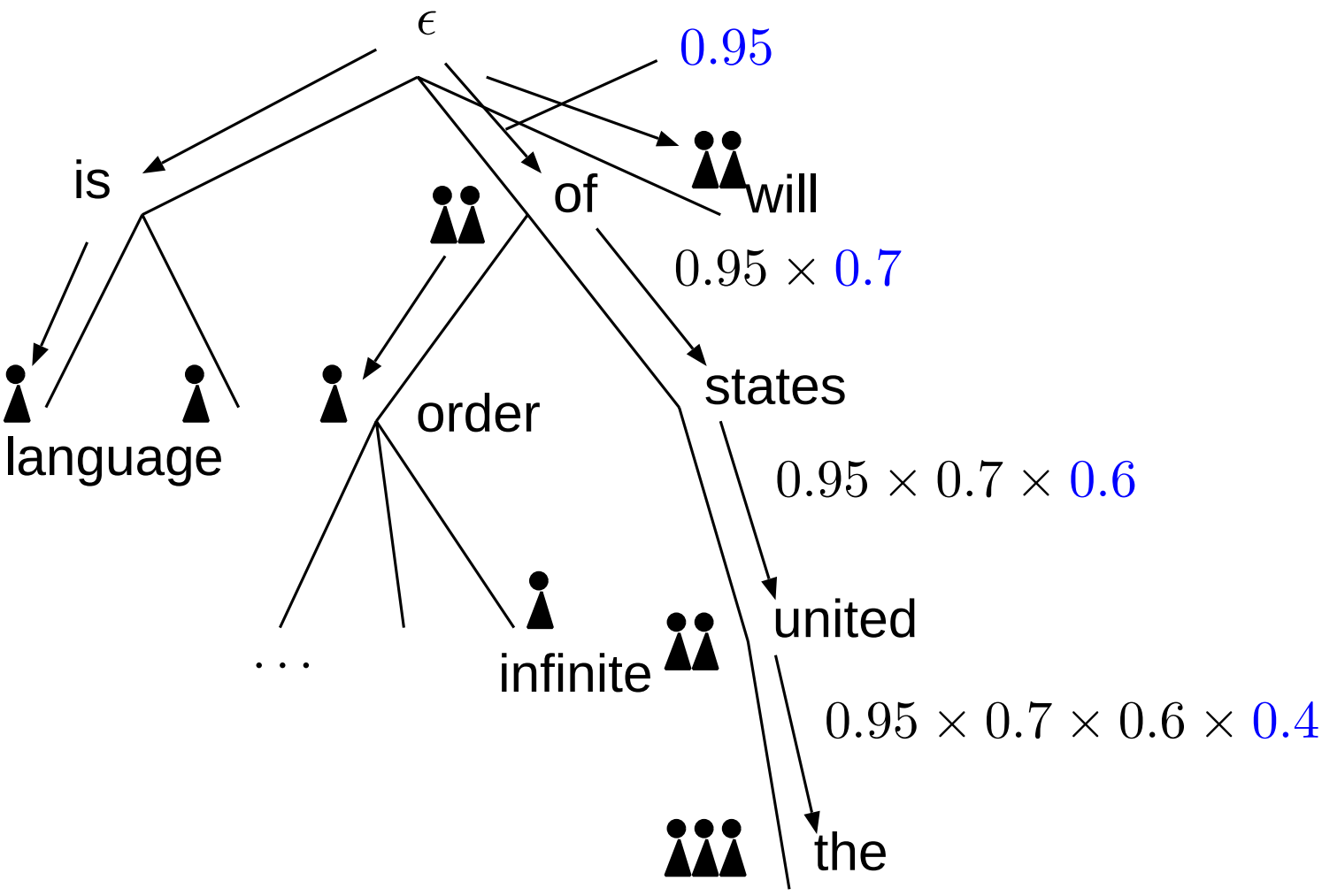
- 客 (カウント) を, 木のルートから確率的にたどって追加
- ノード i に, そこで止まる確率 q_i ($1-q_i$:「通過確率」) がある
  - q_i は, ランダムにベータ事前分布から生成される

$$q_i \sim \mathrm{Be}(\alpha, \beta) \tag{2}$$

  - ゆえに, 深さ n のノードで止まる確率は

$$p(n|h) = q_n \prod_{i=0}^{n-1} (1 - q_i)\,. \tag{2}$$

Variable-order Pitman-Yor Language Model (2)



- 「通過確率」 $1 - q_i$ が大きい … 客が深いノードに到達できる
  - 長い n グラムに対応する
- 「通過確率」が小さい … 'will' など, 浅いノードで充分な文法的関係

Inference of VPYLM

- もちろん, われわれは自然言語の Suffix Tree がもつ真の q_i の値は知らない
  - どうやって推定したらいい?
- VPYLM の生成モデル: 訓練データ $\mathbf{w} = w_1 w_2 w_3 \cdots w_T$ に, 隠れた n-gram オーダー $\mathbf{n} = n_1 n_2 n_3 \cdots n_T$ が存在

$$p(\mathbf{w}) = \sum_{\mathbf{n}} \sum_{\mathbf{s}} p(\mathbf{w}, \mathbf{n}, \mathbf{s}) \tag{2}$$

  $\mathbf{s}$: 代理客を含む客全体の配置
- Gibbs サンプリングにより, この $\mathbf{n}$ は推定できる

Inference of VPYLM (2)

- Gibbs サンプリング: マルコフ連鎖モンテカルロ法 (MCMC) の一種
  - ある隠れ変数を, それ以外を条件にして順番にサンプリング
  - 充分サンプリングを繰り返すと, 真の分布に収束する
- 単語 w_t の生成された n-gram オーダー n_t を,

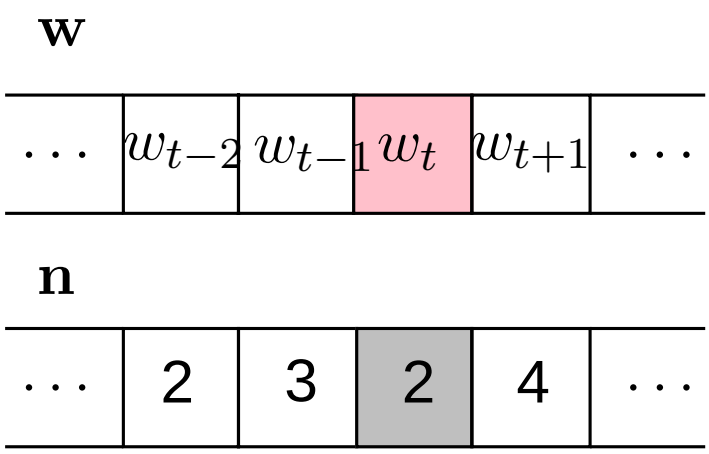
$$n_t \sim p(n_t|\mathbf{w}, \mathbf{n}_{-t}, \mathbf{s}_{-t}) \tag{2}$$

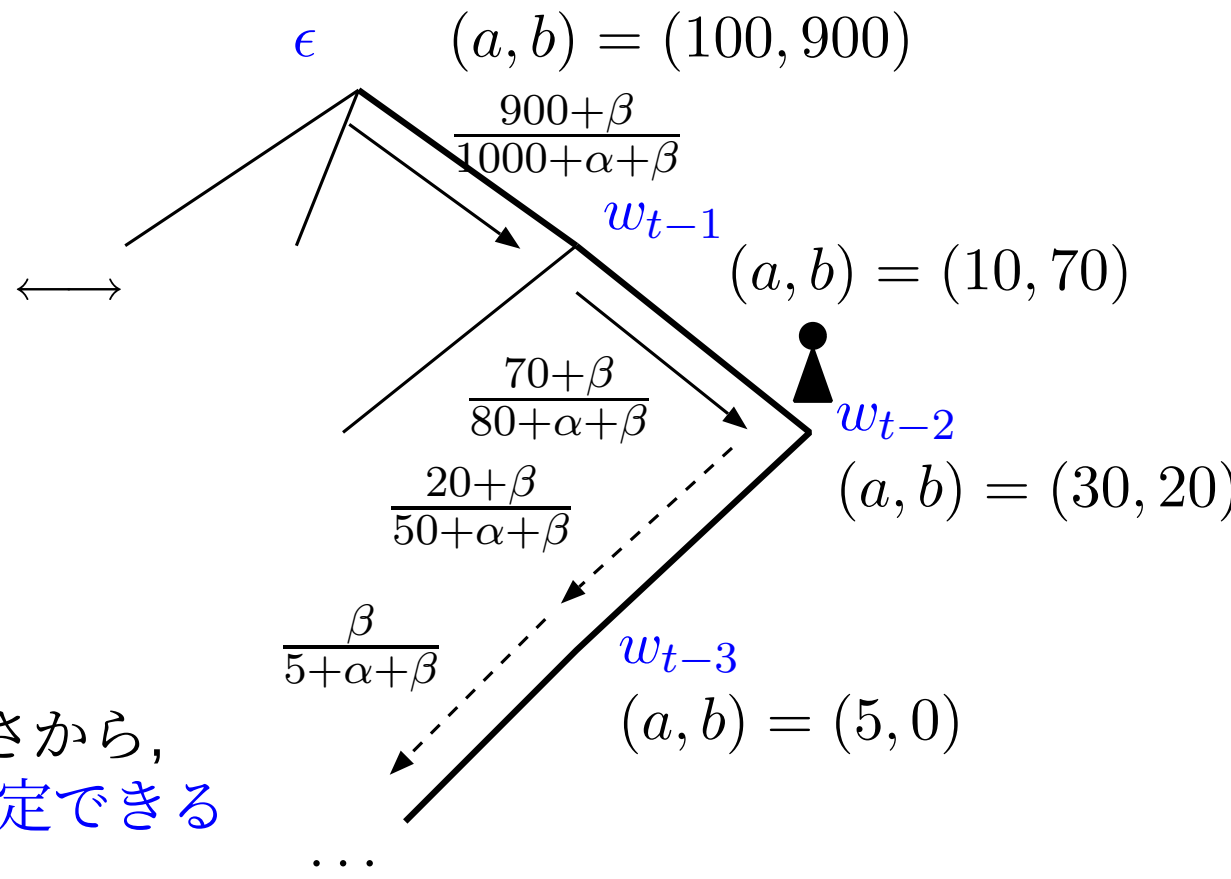
  のようにサンプリング
- ベイズの定理より, $n_t = 0, 1, 2, \cdots, \infty$ について

$$p(n_t|\mathbf{w}, \mathbf{n}_{-t}, \mathbf{s}_{-t}) \propto \underbrace{p(w_t|n_t, \mathbf{w}, \mathbf{n}_{-t}, \mathbf{s}_{-t})}_{n_t\text{-グラムの予測確率}} \cdot \underbrace{p(n_t|\mathbf{w}_{-t}, \mathbf{n}_{-t}, \mathbf{s}_{-t})}_{\text{深さ}n_t\text{に到達する確率}} \tag{2}$$

  - 2 つの確率のトレードオフ (n_t が深すぎるとペナルティ)
  - 第 1 項: HPYLM の n_t-グラム予測確率; 第 2 項は?

Inference of VPYLM (3)





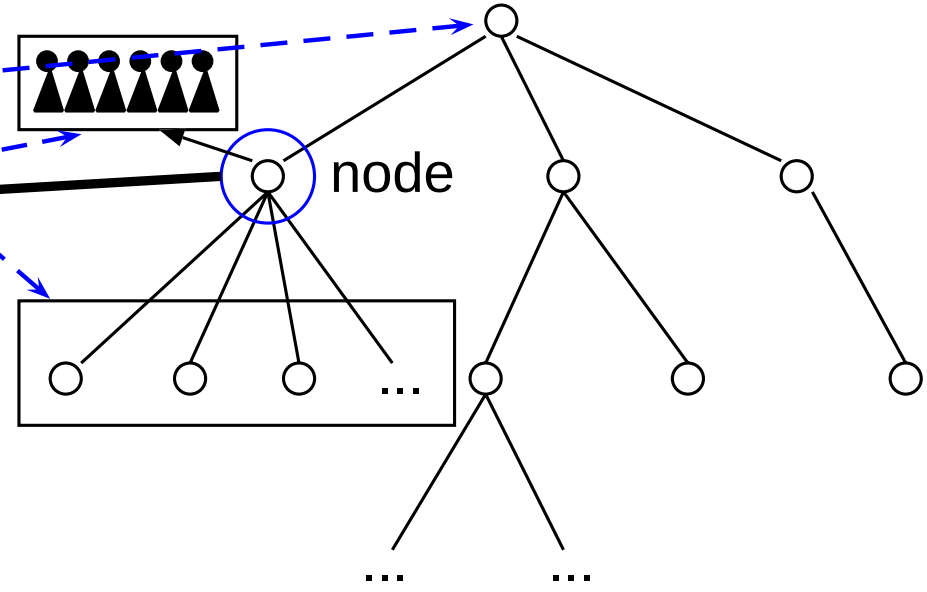
- 他の客の到達した深さから，
  ノードの持つ q_i が推定できる
- ノード i で他の客が止まった回数を a_i, 通過した回数を b_i とすると，

$$p(n_t = n|\mathbf{w}_{-t}, \mathbf{n}_{-t}, \mathbf{s}_{-t}) = q_n \prod_{i=0}^{n-1}(1 - q_i) \tag{2}$$

$$= \frac{a_n + \alpha}{a_n + b_n + \alpha + \beta} \prod_{i=0}^{n-1} \frac{b_i + \beta}{a_i + b_i + \alpha + \beta}. \tag{2}$$

Implementation

```
struct ngram {
    ngram *parent;
    splay *children;
    splay *words;
    int ncounts;
    int ntables;
    int stop;
    int through;
    int id;
};
```



- 子ノード/予測語の検索にスプレー木 (Sleator and Tarjan 1985) を使用
  - ならしオーダー $O(\log n)$ の二分順序木
  - アクセスの際に, 木を自己組織的に最適化して高速化
- `splay *children = (ngram **)`, `splay *words = (restaurant **)`
- Gibbs サンプリングの大幅な高速化 (100 倍以上)

実験

- 英語: 標準的な, NAB(North American Business News) コーパスの Wall Street Journal セットから 10M 語を訓練データ, 1 万文をテストデータ
  - Chen and Goodman (1996), Goodman (2001) などと同じデータ
  - 総語彙数 =26,497 語
- 日本語: 毎日新聞データ 2000 年度から, 10M 語 (52 万文) を訓練データ, 1 万文をテストデータ
  - 総語彙数 =32,783 語
- $n_{\text{max}} = 3, 5, 7, 8, \infty$ で実験
  - パープレキシティ自体は, $n=7$ 程度で飽和 (Goodman 2001)

テストセットパープテキシティとノード数

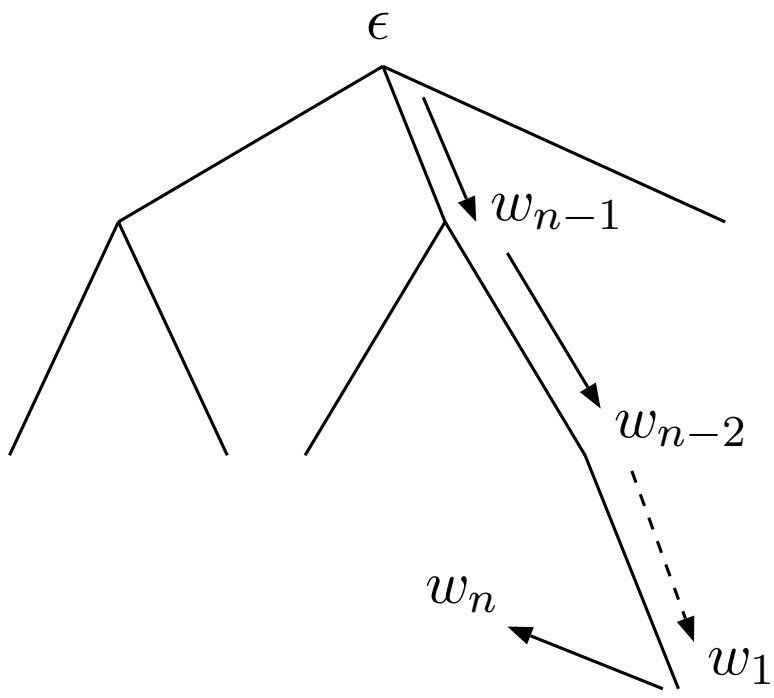
n	SRILM	HPYLM	VPYLM	Nodes(H)	Nodes(V)
3	118.91	113.60	113.74	1,417K	1,344K
5	107.99	101.08	101.69	12,699K	7,466K
7	107.24	N/A	100.68	N/A	10,182K
8	107.21	N/A	100.58	N/A	10,434K
∞	—	—	161.68	—	6,837K

() NAB コーパス (英語)

n	SRILM	HPYLM	VPYLM	Nodes(H)	Nodes(V)
3	84.74	78.06	78.22	1,341K	1,243K
5	77.88	68.36	69.35	12,140K	6,705K
7	77.51	N/A	68.63	N/A	9,134K
8	77.50	N/A	68.60	N/A	9,490K
∞	—	—	141.81	—	5,396K

() 毎日新聞コーパス (日本語)

Suffix Tree と確率的フレーズ



- 木の根 (ユニグラム) から単語を生成するかわりに,
  子ノードを確率的にたどってからフレーズを生成
  - VPYLM の確率オートマトン表現 (初期状態が ϵ)
- フレーズ $\mathbf{w} = w_1 \cdots w_{n-2} w_{n-1} w_n$ が Suffix Tree から生成される確率は,

$$p(\mathbf{w}) = \left(q_{n-1} \prod_{i=0}^{n-2} (1 - q_i) \right) \cdot p(w_n | w_1 \cdots w_{n-1}) \,. \qquad (2)$$

8-gram VPYLM から得られた確率的フレーズ [WSJ]

p	Stochastic phrases in the suffix tree
0.9784	primary new issues
0.9726	ˆ at the same time
0.9556	american telephone &
0.9512	is a unit of
0.9394	to # % from # %
0.8896	in a number of
0.8831	in new york stock exchange composite trading
0.8696	a merrill lynch & co.
0.7566	mechanism of the european monetary
0.7134	increase as a result of
0.6617	tiffany & co.
:	

8-gram VPYLM から得られた確率的フレーズ [毎日新聞]

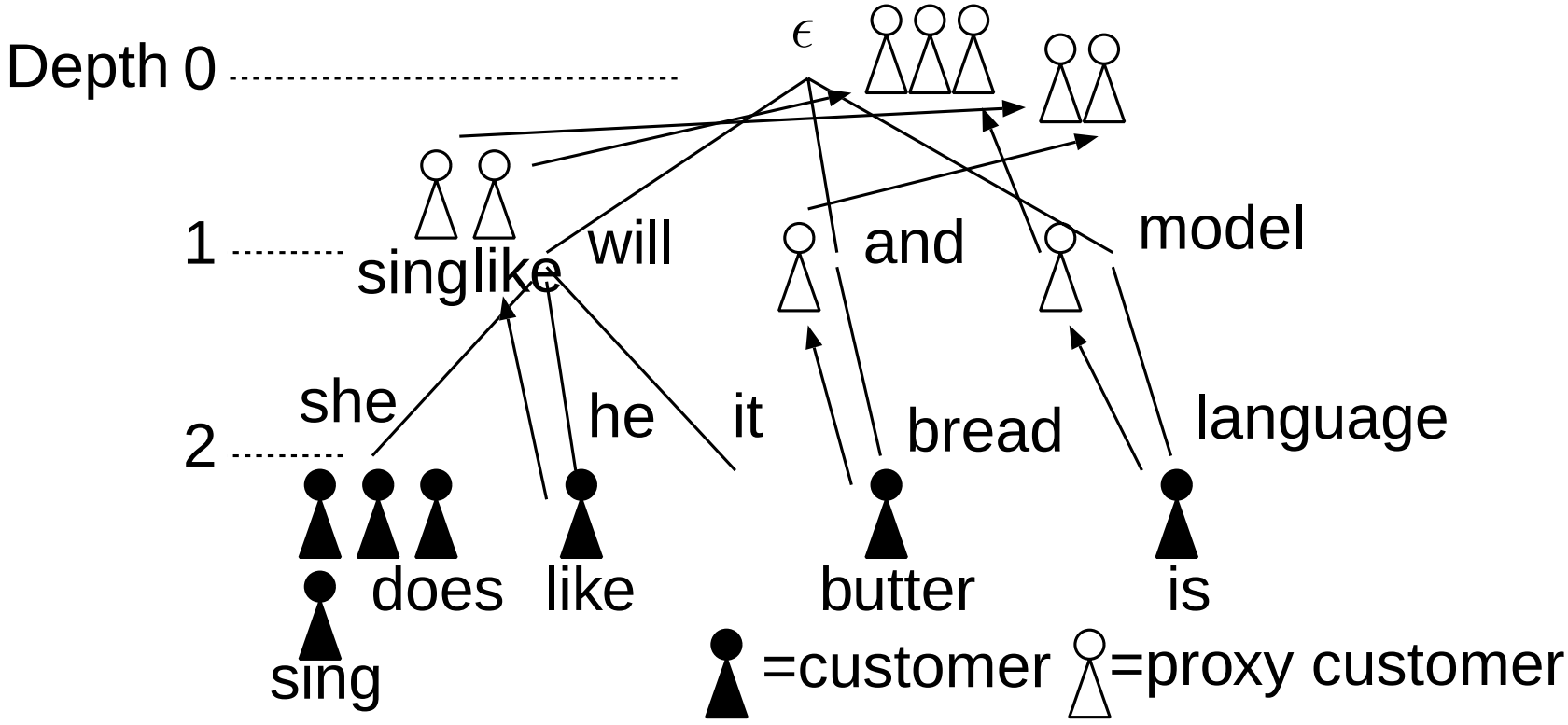
p	Stochastic phrases in the suffix tree
0.9725	早けれ　ば
0.9678	日　発表し　た
0.9592	来　年　＃　月
0.9584	官房　審議　官
0.9185	いる　に　も　かかわら
0.8877	震度　は　次　の　通り
0.8676	フェルメール　と　その　時代
0.8068	ある　と　し　て　いる　。
0.7870	こと　が　＃　日　明らか　に　なっ
0.7661	記録　＃　分　＃　秒　＃
0.7242	全豪オープン　第　＃　日　は　＃　日
0.6472	^　国際　オリンピック　委員　会　IOC
0.5019	ロッテ　＃　勝　＃　敗　＃　分　。
0.5000	行う　こと　を　明らか　に　し　た
：	

LDA によるトピック適応化 [1/2]

- LDA (Blei et al. 2003)$\cdots$ 文書のトピック混合モデル
  - 各文書に, 隠れたトピック混合比 $\boldsymbol{\theta}$ がある
  - 文書の単語は, $\boldsymbol{\theta}$ から確率的にトピックを選んで生成
  - 例: $\boldsymbol{\theta} = (p(t_1), p(t_2), p(t_3), p(t_4)) = (0.2, 0.4, 0.1, 0.3)$
    文書 = $w_1 w_2 w_3 w_4 w_5 w_6 w_7 \cdots$
    (単語ごとにトピックが異なる)
- トピック別の VPYLM を考えることができる
  - 単語ごとに違った可変長 n-gram から生成される
  - 複数の VPYLM の混合による予測
    - 性能悪化!
- データをトピック毎に分割すると, n グラムのスパースネスが深刻
  - 分割しない方が, カウントがヒットする
  - どうしたらいい?

LDA によるトピック適応化 [2/2]

- 解決…ユニグラム分布のみを混合分布にする (混合 Pitman-Yor 測度)
  - Suffix Tree のルートで, 客がトピック別のレストランに入る



実験結果 (VPYLDA)

Model	PPL
VPYLDA ($M=5$)	104.69
($M=10$)	103.57
($M=20$)	103.28
VPYLM	105.30

- 20news-18828 コーパスを使用 (4,000 training documents, 400 test documents), lexicon=10,477
- 性能は改善しているが, その差はわずか
- なぜか?
  - 木の枝に客を追加/削除しても, 木のルートにあるトピック別客分布に反映されないことが多い
  - カウントの追加/削除が, 枝のレベルで吸収されてしまう
  - 階層的混合モデルの, 新しい推定法が必要.

VPYLM からの生成

「レンタ・カーは空のグラスを手にとり、蛇腹はすっかり暗くなっていた。それはまるで獲物を咀嚼しているようだった。彼は僕と同じようなものですね」と私は言った。「でもあなたはよく女の子に爪切りを買った。そしてその何かを振り払おうとしたが、今では誰にもできやしないのよ。私は長靴を棚の上を乗り越えるようにした。...

- 村上春樹「世界の終りとハードボイルド・ワンダーランド」からのランダムウォーク生成 (VPYLM, $n = 5$)
- 普通の 5-gram LM では, オーバーフィットのため学習データがそのまま生成されてしまう
- 確率的に適切な長さの文脈を用いることで, 特徴をとらえた言語モデル
  - 確率的フレーズ: 『なるほど 」 と 私 は 言っ』(0.6560), 『やれやれ 」 と』(0.7953), 『、 と 私 は』(0.8424), . . .

本研究のまとめ

- 階層 Pitman-Yor 過程を拡張することで, 可変長 n グラムの完全な生成モデルを示した
  - 深さの様々に異なる確率的な Suffix Tree を考え, ベイズ推定
  - Markov モデル全てに適用できる, 一般的な理論
  - 無駄な n グラムを覚えず, 効率的
- 各単語の生まれた, 隠れた n グラム文脈長を推定できる
  - 原理的に, n は ∞ まで可能
- 言語モデルの副産物として, 「確率的な句」を取り出すことができる
  - 「Named Entity (NE)」の完全な教師なし学習による獲得
  - NE に限らず, 慣用句などのフレーズも取り出せる
  - 単語分割の粒度に影響されない

展望と課題

- 提案法は, 確率的な Suffix Tree を用いたベイズ正則化
  - Suffix Tree の事前分布には, 共通の一つのベータ分布を用いた
  - どの深さのノードでも, q_i の事前分布が同じでよいか?
  - q_i を生成する, もっと精密な確率モデル (eg. Pitman (2002))
- トピック適応化にはさらに理論が必要
  - トピック毎の n グラムを作ると, スパースネス増大
  - 1-gram,2-gram,.. の階層ごとに混合数が異なるはず
  - 階層的な混合モデルと, その推定方法
- 空間的・時間的計算量のさらなる削減
  - ギブスサンプリングは, 確率最大の点推定ではない
    $\rightarrow$ 確率を最大化するモデル探索 (cf. Hal Daume III (2007))
  - Suffix Tree および Splay Tree の, メモリ使用量の効率化

Thank you

ご清聴ありがとうございました。