論文 / 著書情報
Article / Book Information

論題(和文)	事後確率最大化手法を用いた言語モデルの学習
Title(English)	
著者(和文)	花沢 健, 篠田浩一
Authors(English)	Koichi Shinoda
出典(和文)	日本音響学会平成10年度秋季研究発表会講演論文集, Vol. 2-1-21, No. , pp.
Citation(English)	, Vol. 2-1-21, No. , pp.
発行日 / Pub. date	1998,

2—1—21　事後確率最大化手法を用いた言語モデルの学習 *

⊙ 花沢 健　篠田 浩一
(NEC C&C メディア研究所)

1. はじめに

現在、音声認識に用いる言語モデルとして、単語の連鎖、すなわち、bigram, trigram などの n-gram の出現確率を用いる確率モデルが主流である。各 n-gram 確率は大規模なテキストデータから推定されるが、その際、未観測の n-gram や、少ない回数しか観測されない n-gram (singleton, doubleton, etc) が多数存在するため、n-gram 出現確率を十分良い精度で推定できないという、いわゆるスパースネスの問題が起きる。その解決のためには、何らかのヒューリスティックスを制約として用いて出現確率のスムージングを行う必要がある。従来、この問題に対しては Katz のバックオフスムージング [1] に代表される、Good-Turing 推定 [2] あるいは Leaving-one-out 法 [3] に基づいたスムージング手法がもっぱら用いられてきた。

本稿では、事後確率最大化手法 (e.g.[5]) に基づく n-gram の推定を検討する。この手法は、従来のバックオフスムージングと異なり、未観測の n-gram 以外の n-gram の出現確率も、対応する低次の (n-1, n-2, ...)-gram の確率を考慮して計算される。

2. 事後確率最大化手法

今、単語の総種類数を N とし、k 個の単語から構成される単語列を観測する確率を考える。まず、単語を $\{w_n; n = 1,\ldots,N\}$ とし、$\boldsymbol{p} = (p_1,\ldots,p_N)$、$\boldsymbol{c} = (c_1,\ldots,c_N)$ と定義する。ここで p_n は単語 w_n の出現確率、c_n は単語 w_n の単語列における出現回数である。このとき、単語列の確率分布は多項 (multi-nomial) 分布に従い、

$$f(\boldsymbol{c}|k,\boldsymbol{p}) = \frac{k!}{\prod_{n=1}^{N} c_n!}\prod_{n=1}^{N} p_n^{c_n} \tag{1}$$

と表せる。今、この多項分布 f のパラメータ $\boldsymbol{p}$ の事前分布は、以下のディレクレ分布であると仮定する。

$$g(\boldsymbol{p}|\boldsymbol{q}) \propto \prod_{n=1}^{N} p_n^{q_n} \tag{2}$$

ここで $\boldsymbol{q} = (q_1,\ldots,q_N)(q_n > -1; n = 1,\ldots,N)$ は事前分布 g のパラメータ (hyper parameter) である。このとき、パラメータ $\boldsymbol{p}$ を事後確率を最大にする値をとるように定めると約束すると、簡単な計算の結果、以下のようにその推定値 $\hat{p}_n$ を求めることができる。

$$\hat{p}_n = \frac{c_n + q_n}{\sum_{n=1}^{N} c_n + \sum_{n=1}^{N} q_n} \tag{3}$$

* Language model learning using maximum a posteriori estimation by Ken HANAZAWA and Koichi SHINODA (NEC Corporation)

さて、この方法を trigram 確率の推定に応用する。ここでは、(n-1)-gram 確率分布を n-gram 確率のパラメータの事前分布として用いる。

まず 0-gram 確率 p_0 を仮定する。これは unigram の出現確率に対する先験確率である。ここでは、データベースに出現する総 unigram 種類数を N_{all} としたとき、総単語種類数 N を $N = \alpha N_{all}$, $(\alpha > 1)$ と近似し、p_0 を以下のように仮定する。

$$p_0 = \frac{1}{N}, \quad \text{for } w = 1,\ldots,N. \tag{4}$$

次に $p(w)$ および条件付き確率 $p(w|v)$, $p(w|u,v)$ を順に求める。今、$c(w)$ を単語 w の出現回数、$c(v,w)$ を単語列 (v,w) の出現回数、$c(u,v,w)$ を単語列 (u,v,w) の出現回数とする。まず unigram 確率分布 $\{p(w)\}$ のパラメータ $p(w)$ に対する事前分布の hyper parameter を $q(w) = \beta_1 p_0$ と仮定すると、$p(w)$ は以下の式で求められる。

$$p(w) = \frac{c(w) + \beta_1 p_0}{\sum_w c(w) + \beta_1} \tag{5}$$

ここで β_1 は制御変数であり、すべての w について一定と仮定する。さらに、bigram の条件付き確率 $p(w|v)$、trigram の条件付き確率 $p(w|u,v)$ は順に以下のように求められる。

$$p(w|v) = \frac{c(v,w) + \beta_2 p(w)}{c(v) + \beta_2} \tag{6}$$

$$p(w|u,v) = \frac{c(u,v,w) + \beta_3 p(w|v)}{c(u,v) + \beta_3} \tag{7}$$

ここで β_2、β_3 は制御変数である。

3. 言語モデル適応への応用

本手法を言語モデル適応へ応用する。今、基準となる大規模コーパスを A、言語モデル適応の対象となる新しいコーパスを B とする。ここでは、例として bigram の条件付き出現確率を推定する場合について説明する。今、単語 w の B における出現確率を $p^B(w)$、A における bigram 条件付き確率を $p^A(w|v)$、B における単語組 (v,w) の出現回数を $c(v,w)$ とする。求めるべき単語 w の B における bigram 条件付き確率 $p^B(w|v)$ は、その事前分布がディレクレ分布であり、$p^B(w)$ と $p^A(w|v)$ のパラメータがその hyper parameter であると仮定したとき、

$$p^B(w|v) = \frac{c(v,w) + \beta_2 p^B(w) + \gamma_2 p^A(w|v)}{c(w) + \beta_2 + \gamma_2} \tag{8}$$

と求められる。ここで γ_2 は制御変数である。

4. 評価実験

本手法によるスムージングの有効性を、bigramにより英語と日本語の二通りについて評価した。

4.1. discounting 手法

本手法の比較対象とする従来の手法として、観測されなかった事象のカウントを観測された事象のカウントをdiscount することによって補間するバックオフ法を用いる。

ある事象の起こったカウント r を、discount係数 d_r を用いて discounting し、discount 後のカウント $\tilde{r} = d_r \times r$ を用いて確率計算を行う。以下、いくつかの手法での d_r の求め方を示す。*Good-Turing discounting* [2] では、n_r を r 回起こった事象の種類数、K を定数として、

$$d_r = \frac{\frac{(r+1)\times n_{r+1}}{r\times n_r} - \frac{(K+1)\times n_{K+1}}{n_1}}{1 - \frac{(K+1)\times n_{K+1}}{n_1}} \qquad (9)$$

と求める。*Absolute discounting* [6]、*Linear discounting* [6]、*Witten-Bell discounting* [7] では、それぞれ式 (10)、式 (11)、式 (12) のように求める。

$$d_r = \frac{r-b}{r} \qquad (10)$$

$$d_r = 1 - \frac{n_1}{N_{all}} \qquad (11)$$

$$d_r(t) = \frac{N_{all}}{N_{all}+t} \qquad (12)$$

ここで、b は定数、n_1 は 1 回起こった事象の種類数 (singleton)、t はある条件のときに起こり得る事象の種類数である。

4.2. テストセットパープレキシティによる評価

英語における評価実験では、言語モデルはテキストコーパス Wall Street Journal 0 (WSJ0) [8] による 160 k 文学習のものを用い、語彙は 66.8k である。評価データは、同コーパスより学習文に用いなかったもの 160 文 (未知語なし) である。

日本語での評価実験では、言語モデルは会話テキストコーパスによる 18 k 文学習のものを用い、語彙は約 8k である。評価データは、会話評価文 375 文 (未知語率 3 %) である。スムージング法としては、前節で述べた各手法と本手法とを用いる。

制御変数 β_1, β_2 は個数の次元を持つ。ここでは、最適値の推定を容易にするために以下のような正規化を行った。unigram, bigram において同一の β' を使用した。

$$\text{unigram} \quad : \quad \beta_1 = \beta' \times \sum_w c(w)$$

$$\text{bigram} \quad : \quad \beta_2 = \beta' \times \frac{\sum_w c(w)}{N_{all}}$$

この β' は、β_1, β_2 をそのまま用いる場合に比べてコーパスの規模の違いに対し頑健であると予想される。

β'	perp.
1	247
0.01	202
0.0001	239

表 1: 英語評価 : 本手法での test-set perplexity

手法 (パラメータ)	英語	日本語
Good-Turing (K=4)	258	103
Absolute (b=0.5)	270	114
Linear	691	299
Witten-Bell	602	260
本手法 (β'=0.01)	202	79

表 2: 各手法での test-set perplexity

また従来の各手法では変更可能なパラメータの設定についてそれぞれ予備実験を行い、最良のものを採用している。

英語の評価データに対し、本手法において β の値を変更したときの perplexity の変化を表 1 に、英語および日本語の評価データにおける各手法の perplexity の比較を表 2 にそれぞれ示す。

実験から、英語日本語両方の結果において、本手法は従来手法に比べて低い perplexity が得られることがわかった。また、表 1 に示す英語の実験で最適化された β の値が、日本語の実験でも有効であることがわかった。

5. おわりに

事後確率最大化手法を用いた言語モデルのスムージング手法を検討した。本手法では従来手法に比べて低い perplexity が得られた。今後は、認識実験による評価を行うとともに、言語モデル適応へも本手法を応用していきたい。

参考文献

[1]S.Katz, *IEEE Trans. ASSP*, vol.35, no.3, pp.400-401, 1987.

[2]I.J.Good, *Biometrika*, vol.40, parts 3 and 4, pp.237-264, 1953.

[3]A.Nadas, *IEEE Trans. ASSP*, vol.33, pp.1414-1416, 1985.

[4]F.Jelinek and R.L.Mercer, *Proc. of the Workshop on Pattern Recognition in Practice*, pp.381-397, 1980.

[5]M.H.DeGroot: "Optimal Statistical Decisions," McGraw-Hill, 1970.

[6]H.Ney, U.Essen, and R.Kneser, *Computer Speech and Language*, vol.8, pp.1-38, 1994.

[7]I.T.Witten and T.C.Bell, *IEEE Trans. TIT*, pp.1085-1094, 1991.

[8]D.Paul and J.Baker, *Proc. of DARPA SNL Workshop*, pp.357-362, 1992.