2006－SLP－60（8）
2006／2／4

PLSA 言語モデルの学習最適化と語彙分割に関する検討

栗山　直人†　　鈴木　基之†　　伊藤　彰則†　　牧野　正三†

† 東北大学大学院工学研究科　〒 980-9579 宮城県仙台市青葉区荒巻字青葉 6-6-05
E-mail: †{kriya,moto,aito,makino}@makino.ecei.tohoku.ac.jp

あらまし　PLSA は，文章の特徴「話題」を反映した言語モデルを構築する手法である．この PLSA 言語モデルの拡張を提案する．前半では PLSA 言語モデルの学習について，既存の複数の方法を比較し，EM アルゴリズムのアニーリングスケジュール最適化についての検討を行う．後半では PLSA 言語モデルを内容語モデルと機能語モデルに分割し，話題（トピック）と話し方（スタイル）を，別々に学習・適応することで従来の PLSA 言語モデルよりもより柔軟な言語モデル適応を試みる．その結果，学習最適化については β を 1.0 から特定の値に向けて減少させるアニーリングスケジュールが最適という結果が得られた．内容語・機能語に分割したモデルについては trigram に対する Perplexity が従来の PLSA 言語モデルの 83.90% から 82.23% へ改善した．

キーワード　言語モデル，PLSA，文脈適応，EM アルゴリズム

Training optimization and vocabulary division of PLSA language model

Naoto KURIYAMA†, Motoyuki SUZUKI†, Akinori ITO†, and Shozo MAKINO†

† Graduate School of Engineering, Tohoku University　Aza-Aoba 6-6-05, Aramaki, Aoba-ku, Sendai-shi, Miyagi, 980–9579 Japan
E-mail: †{kriya,moto,aito,makino}@makino.ecei.tohoku.ac.jp

Abstract　PLSA is a method of composing language model which can reflect the global charactetistics of linguistic context as “topic”. We propose more extention of PLSA language model. First, we compare the conventional learning methods of PLSA language model, and examine the optimization of EM annealing schedule. As a result, we found that the best method is to reduce β from 1.0 to some special value. Next, we compose a PLSA language model whose vocabulary set is divided, into content words and function words. Then training and adaptation to topic or style are performed separately. In the experiment, we acheived 82.23% perplexity reduction against conventional way 83.90% .

Key words　Language model, PLSA, context adaptation, EM algorithm

1. はじめに

現在，大語彙連続音声認識に一般的に用いられる言語モデルは N-gram モデルである．N-gram は直前数単語の情報を基に確率を見積もるので，言語モデルの制約に反映される文脈的情報はごく一部にすぎない．文脈全体から得られる特徴には「話題」「話し方」といった情報があり，これらに言語モデルを最適化する事ができれば，より自然で高い認識精度を持つ音声認識が可能になる．

言語モデルを認識対象の話題や話者性に適応する方法には，認識結果から推定される話題について関連のあるテキストを集め，そのコーパスで言語モデルを再構築する方法や，話題別に学習された複数の unigram モデルを持ち，認識結果の単語履歴に対して最適な混合比を求めて混合する方法がある．

本研究では後者について，Thomas Hofmann によって提案された PLSA [1] を用いる．PLSA は「話題」を言語モデルに反映させるための手法であり，高い Perplexity 削減効

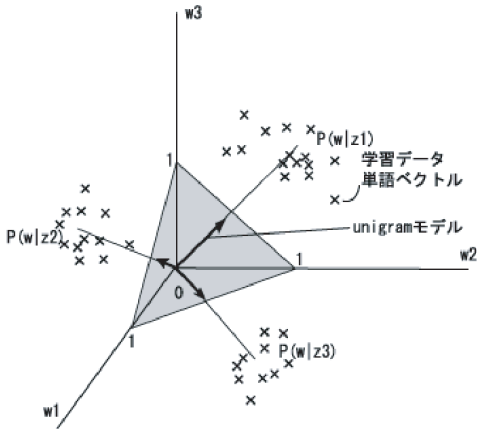

図 1　PLSA 言語モデル生成 概念

果が報告されている [2] [3] [4].

この PLSA を用いた言語モデルについて，２つの事項を検討した．一つには，これらの先行研究はどれも EM アルゴリズムによる最尤推定法で PLSA 言語モデルを構築しているが，それぞれ違ったアニーリングスケジュールが用いられている．本研究ではこれらの方法について比較実験を行い，PLSA 言語モデル構築に最適なアニーリングスケジュールを探る．二つ目には，PLSA 言語モデルを語彙基準で２つに分割することを試みた．トピック（話題）を表す語彙だけで構成されるモデルと，スタイル（話し方，文体）を表す語彙だけで構成されるモデルの２つを別々に学習し，文脈に対しても別々に適応する．こうすることでトピックとスタイルをより柔軟に学習・適応することがねらいである．

2.　PLSA 言語モデルの概要

2.1　PLSA 言語モデルについて

PLSA(Probabilistic Latent Semantic Analysis, 潜在意味解析) とは，単語の出現頻度を基に,「話題」を，モデル化する手法である [1]．そのモデルの実体は，特定の話題や話し方を反映した unigram 言語モデルの複数混合モデルである．PLSA 言語モデルの与える文脈 h を反映した単語 w の出現確率 $P(w|h)$ を式（1）に示す.

$$P(w|h) = \sum_{z \in Z} P(z|h)P(w|z) \qquad (1)$$

$P(w|z)$ は単語 w に対する内部 unigram モデル z が与える確率である．この内部モデルを潜在モデルと呼ぶ．潜在モデルはそれぞれが異なる話題や話し方を学習していて，目的の話題 h に対して最適な混合比 $P(z|h)$ で混合することで，目標の話題に言語モデルを適応することができる．この確率 $P(w|h)$ を trigram と併用して用いる.

2.2　モデル生成の原理

新聞記事のように決まった話題が書かれたテキストから，単語出現回数を求める．なお本研究ではこのような特定の話題を持つテキストをこれ以降「記事」と呼ぶ．記事の出現単語の中には話題を直接表すもの・文体あるいは話し方を表す単語が存在し，それらの出現傾向から元の記事が持っていた話題の情報や文章のスタイル情報を間接的に知ることができると言える.

この記事の単語出現回数は，全語彙数を次元とするベクトル空間上において，一点を指すベクトルとして考えることができる．そこでこのデータを単語ベクトルと呼ぶ．大量のテキストから単語ベクトルを得てベクトル空間上に配置すると，似たようなトピック・スタイルを持つ記事は，ベクトル空間の近い位置に存在すると考えることができる.

PLSA 言語モデルは，この単語ベクトル群を学習データとし，それらに対し尤度最大化するような任意数のベクトルを学習することで構築する．具体的には式 (2) の尤度を最大化する.

$$l(\theta; N) = \sum_{w \in W} \sum_{d \in D} n(d,w) log \sum_{z \in Z} P(z|d)P(w|z) \qquad (2)$$

$n(d,w)$ は学習記事 d 中の単語 w の出現回数であり，学習対象の単語ベクトルにあたる．$n(d,w)$ の集合 N について尤度最大化するようにパラメータ $P(z|d)$, $P(w|z)$ を学習する.

図 1 では x 点が学習データの単語ベクトルを表し，実線の方向のベクトルが学習されるベクトルである．このベクトルを正規化したものが潜在モデルであり，ある特定の話題・話し方の特徴を反映した unigram モデルとなる.

学習には，Tempered EM アルゴリズムという反復学習法を用いる [1]．式（3）～（6）にそれを示す.

E-Step:

$$P^{(k)}(z|d,w) = \frac{\{P^{(k)}(z)P^{(k)}(d|z)P^{(k)}(w|z)\}^{\beta}}{\sum_{z \in Z} \{P^{(k)}(z)P^{(k)}(d|z)P^{(k)}(w|z)\}^{\beta}} \qquad (3)$$

M-Step:

$$P^{(k+1)}(w|z) = \frac{\sum_{d \in D} n(d,w)P^{(k)}(z|d,w)}{\sum_{w \in W} \{\sum_{d \in D} n(d,w)P^{(k)}(z|d,w)\}} \qquad (4)$$

$$P^{(k+1)}(d|z) = \frac{\sum_{w \in W} n(d,w)P^{(k)}(z|d,w)}{\sum_{d \in D} \{\sum_{w \in W} n(d,w)P^{(k)}(z|d,w)\}} \qquad (5)$$

$$P^{(k+1)}(z) = \frac{\sum_{w \in W} \sum_{d \in D} n(d,w)P^{(k)}(z|d,w)}{\sum_{w \in W} \sum_{d \in D} n(d,w)} \qquad (6)$$

E-Step と M-step を交互に反復することで式（2）を最大化するモデルが生成される．通常の EM アルゴリズムと異なる点として，E-Step 右辺全体を β 乗（$0 < \beta \leqq 1.0$）する．$\beta = 1.0$ のときが通常の EM アルゴリズムになる．β が 1.0 に対し小さければ小さい程 E-Step の事後確率 $P(z|d,w)$ は

平滑化され，その結果潜在モデルの確率 $P(w|z)$ も平滑化を受ける．同時に尤度関数も同様に平滑化を受け，尤度の局所的なピークを打ち消す効果がある．

Tempered EM アルゴリズムはこの β を，E-Step と M-Step の反復が進む毎に一定割合で変化させる．β を変化させる手続きを，アニーリングスケジュールと呼ぶ．

2.3 目的文脈への適応

文脈のトピック・スタイルへの適応は，潜在モデルの混合比を目的の文脈に対し最尤推定することで行う．具体的には適応したいテキストの単語ベクトル $n(h,w)$ に対し PLSA 言語モデルが尤度最大化する混合比を，学習時と同じ Tempered EM アルゴリズムで推定する．式（7）～（8）に示す．

E-Step:

$$P^{(k)}(z|h,w) = \frac{\{P^{(k)}(z)P^{(k)}(h|z)P^{(k)}(w|z)\}^{\beta}}{\sum_{z\in Z}\{P^{(k)}(z)P^{(k)}(h|z)P^{(k)}(w|z)\}^{\beta}} \tag{7}$$

M-Step:

$$P^{(k+1)}(h|z) = \frac{\sum_{w\in W} n(h,w)P^{(k)}(z|h,w)}{\sum_{z\in Z}\{\sum_{w\in W} n(d,w)P^{(k)}(z|h,w)\}} \tag{8}$$

2.4 N-gram との混合

PLSA モデルは本質的には unigram モデルであるので，文法的制約の反映には弱い．そこで文法制約の反映に強い trigram と混合して使用する．式（9）の unigram rescaling という手法 [2] を用いる．

$$P(w_i|h,w_{i-2},w_{i-1}) \propto \frac{P(w_i|h)}{P(w_i)}P(w_i|w_{i-2},w_{i-1}) \tag{9}$$

音声認識システムに実装した場合では，認識結果を基に言語モデルを動的に適応することになる．

3. PLSA 言語モデルの学習最適化

3.1 はじめに

ここでは基本的な構成の PLSA 言語モデルを，先行研究で用いられているいくつかのアニーリングスケジュールによって構築し，それらの性能比較を行う．

3.2 EM アルゴリズムのアニーリングスケジュール

PLSA 言語モデル構築のためのアニーリングスケジュールは先行研究により異なっている．三品ら [3]，秋田ら [4] は β を学習が進むにつれ段階的に増やし，最終的に 1.0 に至るという方法をとっている．これは特に Determinisitic Annealing EM(DAEM) [5] という手法であり，学習初期では尤度関数を単峰化し，局所最適解に収束することを防ぐ効果がある．

一方 Thomas Hofmann ら [1] [2] は逆に β を 1.0 から段階的に減らす手法をとっている．この手法は“ Inverse annealing ”と呼ばれ, 学習を加速させると言われている．また事後確率 $P(z|d,w)$ を平滑化することから過学習を防ぐ効果がある．

そこでこの 2 つのアニーリングスケジュールで PLSA 言語モデルを学習し，その性能を比較する実験を行った．

3.3 実験条件

PLSA 言語モデルの条件を表 1 に示す．学習データとして新聞記事を記事毎に分割したテキストを用いている．ただし新聞記事の中には，単語数の極めて少ない記事も存在し，そのようなものは明確な話題を持たないので学習データに適さない．そこで 1 記事が１５０形態素に満たない大きさのものは学習データから除外している．この処理で全体の 3 割程度が除去されている．

評価条件，言語モデル適応のアニーリングスケジュールを表 2 に示す．適応時のアニーリングスケジュールには過適応を防ぐため Inverse annealing を用いている．言語モデルの評価尺度は Testset Perplexity である．テストセットには学習データ翌年の新聞記事を用いている．

アニーリングスケジュールには以下の三つの方法を用いた．

I　DAEM を用いた方法で，特に三品らの研究 [4] と同じアニーリングスケジュール．β を 0.50, 0.60, 0.70, 0.75, 0.80, 0.80, 0.90, 0.93, 0.97 の順に増加させ，各値で 3 回ずつ反復を行う．最後に β を 1.00 にし１５回反復して終了する．

II　DAEM による方法で，反復 k 回目と $k-1$ 回目の間でのパラメータの収束を式（10）で確認し，閾値 $C_{th.}$ 以下まで収束したら β を更新する方法．β は上と同じ $0.50, 0.60, \ldots, 0.97, 1.00$ の順で更新する．閾値は $C_{th.}=0.3$, 0.5, 1.0, 1.3 の 4 パターンについて実験した．

$$Conv(k) = \sum_{z\in Z}\sum_{w\in W}|P_k(w|z) - P_{k-1}(w|z)| \qquad Conv(t) < C_{th.}\text{で}\beta\text{更新} \tag{10}$$

III　Inverse annealing による方法. β は 1.0 でスタートし，その更新乗数 $\beta_{renew}(0<\beta<1.0)$ を決めておく．EM アルゴリズムの反復回数は 100 回に固定し，反復が 20 回進むごとに β を β_{renew} 倍する．β_{renew} が 0.70, 0.75, 0.80, 0.85, 0.90, 0.93, 0.95 の場合について実験した（式（11））.

$$\beta = \{\beta_{renew}\}^{\lfloor \frac{EMiteration}{20} \rfloor} \tag{11}$$

いずれの方法でも学習進度による性能変化を追跡するため，学習が終了した時点での言語モデルだけでなく，β が変わる毎にその時点での言語モデルも出力させている．

3.4 実験結果

実験結果を図 2 に示す．グラフは左から順に方法 I, II, III で，方法 II では $C_{th.}$ を，方法 III では β_{renew} を変化させた結果である．グラフの縦軸は trigram に対する Perplexity の比率 [%] である．

I. の三品らの手法では 4 %程度しか削減できていないこ

表 1　言語モデル学習条件

	PLSA 言語モデル
語彙数	1 万＋未知語
学習データ	72369 記事 （毎日新聞 '00 版 1 年分）
潜在モデル混合数	50
	N-gram 言語モデル
学習データ	毎日新聞 '00 版 全文

表 2　言語モデル評価条件

適応 EM 反復回数	60
β	1.00 → 0.95 → 0.90（20 反復で更新）
テストセット	100 記事（毎日新聞 '01 版 300 単語以上）

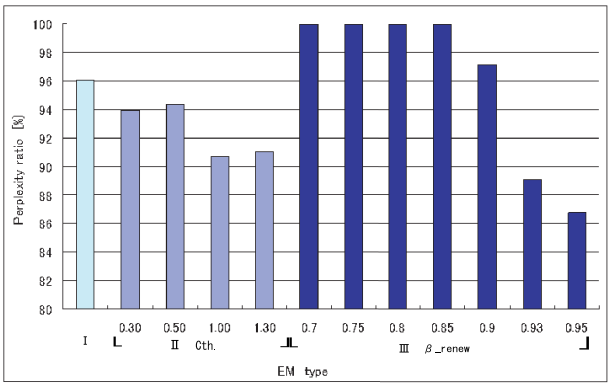

図 2　アニーリングスケジュールによる PLSA 言語モデルの性能差

とが分かる．同じ DAEM でも II. のパラメータの収束を確認してからβを更新する方法では，最高で 9.3 %の削減が得られている．収束を確認する閾値 $C_{th.}$ が小さいほうがより十分なパラメータ収束を行なったモデルだが，性能は $C_{th.}$ が大きいほうが勝っている．

最も優れているのは III. の β を減らしていく手法である．最高で β_{renew} が 0.95 のとき 13.2% の削減が得られた．

3.5　学習終了時のβと性能の関係

図 2 の Inverse annealing の結果に，β が変化する EM 反復 20, 40, 60, 80 回の時点でのモデルの結果も加え，「学習終了時のβの値」を横軸にしてプロットすると，図 3 が得られる．

ここにははっきりとした傾向が現れ，（学習終了時のβ）＝ 0.80 付近に性能のピークがある．終了時のβが 0.80 より小さい場合はモデルの平滑化が進み過ぎ，0.80 より大きい場合は逆に過学習が起きているという傾向が見られた．

3.6　DAEM と Inverse annealing についての考察

DAEM では $C_{th.}$ が大きい方が良い結果となった．これは学習データに対して最尤学習すると過学習が起きるということを示していると考えている．

図 4 に，最も性能が良かった Inverse annealing の $\beta_{renew} = 0.95$ モデルについて学習過程の対数尤度の変

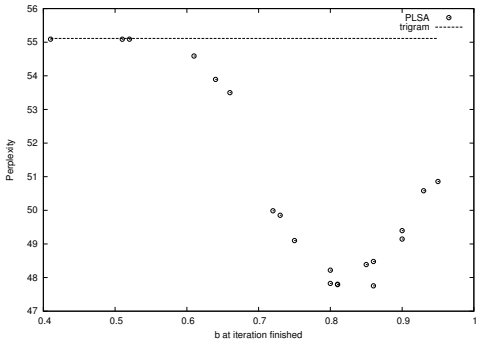

図 3　図 2 を学習終了時のβを基準にプロット

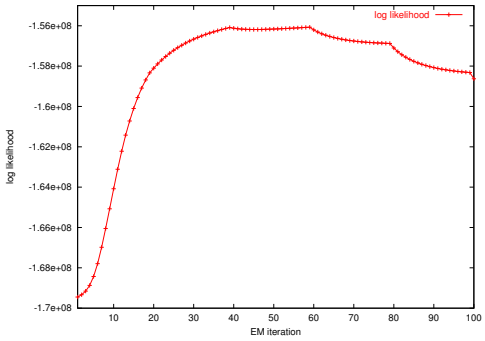

図 4　学習過程での尤度変化

化を示す．反復４０回までは上昇しているが，それ以後は下降に転じている．β は学習が進む程小さくなっていくので，この例では学習データに最尤な点を一度経由し，それよりやや平滑化されたモデルが得られていることが分る．

DAEM でβを増加させるのは，局所最適解に収束するのを防ぐためであり，目的は尤度最大化である．一方βを減少させる Inverse annealing では，過学習を防ぎ，生成されるモデルの確率を平滑化する効果が重要と考えられる．

この実験結果からは Inverse annealing による言語モデルの確率平滑化が，学習最適化に重要であると分かる．この効果と DAEM の局所最適性の解消効果は別のものと考えると，βを小さい値から 1.0 に近付け，いったん尤度最大のモデルを作ってから再びβを減少させて確率を平滑化する，という方法も考えられる．

4.　語彙分割を行った PLSA 言語モデル

4.1　潜在モデルの反映する特徴の推定

3.4 で最も性能の良かったモデル（$\beta_{renew} = 0.95$）について，潜在モデルが実際にトピック・スタイルを反映したモデルになっているのかどうか調べた．式（12），（13）に示す評価量を用いた．

$$RATIO(w|z) = \frac{P_{PLSA}(w|z)}{P_{unigram}(w)} \qquad (12)$$

$$DIF(w|z) = P_{PLSA}(w|z) - P_{unigram}(w|z) \qquad (13)$$

MODEL: 0	RATIO		DIF
1:（ 訴状 ）	70.6487	（ 被告 ）	0.0143
2:（ 証人 ）	70.2978	（ た ）	0.0127
3:（ 敗訴 ）	69.7208	（ 判決 ）	0.0102
4:（ サリン ）	69.6360	（ を ）	0.0088
5:（ 検察官 ）	69.2114	（ 被害 ）	0.0082
6:（ 陪審 ）	69.0577	（ 側 ）	0.0081
7:（ 傍聴 ）	69.0426	（ し ）	0.0078
8:（ 本件 ）	68.8950	（ 事件 ）	0.0078
9:（ 陳述 ）	68.8768	（ 裁判 ）	0.0070
10:（ 原告 ）	68.8313	（ 訴訟 ）	0.0069
11:（ 真須美 ）	68.5270	（ 地裁 ）	0.0067
12:（ 慰謝 ）	68.3019	（ 弁護士 ）	0.0064
13:（ オウム ）	67.9671	（ れ ）	0.0052
14:（ 尋問 ）	67.9255	（ 認め ）	0.0050
15:（ 弁護 ）	67.8458	（ 法 ）	0.0043
16:（ 判決 ）	67.7454	（ 元 ）	0.0043
17:（ 裁判 ）	67.6612	（ 罪 ）	0.0041
18:（ 裁判官 ）	67.4792	（ 求め ）	0.0040
19:（ 審 ）	67.3934	（ 賠償 ）	0.0039
20:（ 訴訟 ）	67.3401	（ 長 ）	0.0037

図 5　潜在モデル 0 番の確率上昇率，上昇量の上位二十単語

どちらもある単語 w について，PLSA の潜在モデル z が与える確率と普通の unigram モデルの確率の間の確率変動を測る評価量である．$RATIO$ は確率変化比率，DIF は確率差を表す．一例として潜在モデル 0 番を用い全単語について $RATIO$ と DIF を求め，各上位２０単語を図 5 に示す．直感的に「裁判に関するトピックを反映したモデル」であると推定することができる．特に $RATIO$ の上位に，学習されたトピックを示すような特徴的な単語が現れる．

そこでこの処理を 50 個の潜在モデル全てについて行ない，確率上昇量の大きい単語から反映されたトピック・スタイルを人手で推定した．結果を表 3 に示す．

4.2　トピック・スタイルの分離

表 3 から，従来の PLSA 言語モデルはトピックだけでなくスタイルを学習する効果もあるということが分かる．

単語の出現頻度に現れる特徴には話題・話し方などがあるとここまで述べてきたが，この二つはほぼ独立な特徴と考えることができる．例えば「話題は同じだが話し方が違う」ということが可能である．もしこのニつの特徴を分離することができれば，トピックとスタイルを別々に文脈適応することが可能になり，適応の自由度が向上すると考えられる．

さらにこの分離が可能であれば，認識対象のトピックとスタイルの特徴をそれぞれ別のコーパスから獲得することが期待できる．例えば従来の PLSA を話し言葉のスタイルに適応すると，その際に同時に適応できる話題は話し言葉の学習データに共起していたトピックに限定されてしまい，広い分野の話題には対応できない．トピックとスタイルが分離されることで，話題性は新聞記事のように広い話題をカバーするコーパスから学習した特徴を，発話のスタイルは話し言葉のコーパスから得られる特徴に適応することが可能になる．

4.3　内容語・機能語の分離

以上の事から，PLSA の反映する「話題（トピック）」と「話し方 (スタイル)」を分離し，より柔軟な学習・適応を行うことを検討する．その方法として PLSA 言語モデルの語彙を，話題の特徴を受ける内容語クラスと，話し方の特徴を受ける機能語クラスに分離し，別々に言語モデル学習・適応を行う．語彙の分割は表 4 の基準で行う．なお品詞の分類には形態素解析システム chasen [6] を用いている．

表 3　推定されたトピック・スタイル

0	裁判	17	交通	34	選挙
1	野球	18	情報通信	35	大相撲
2	音楽・映画	19	記者会見	36	数表スタイル
3	遺跡	20	申し込み	37	水泳・陸上
4	社説スタイル	21	教育	38	サッカー
5	国際情勢	22	囲碁・将棋	39	高校野球
6	スポーツ	23	アジア	40	家庭
7	政界	24	スタイル	41	自然災害
8	スタイル	25	料理	42	競馬
9	核問題	26	福祉	43	医療
10	国政	27	刑事事件	44	スタイル
11	書籍	28	記号スタイル	45	天気
12	税制	29	住居・車	46	経済
13	口語スタイル	30	ゴルフ	47	スタイル
14	曜日スタイル	31	スタイル	48	銀行
15	歴史問題	32	産業	49	宝くじ
16	絵画	33	スタイル	(モデル)	(推定ラベル)

表 4　語彙の分割基準

内容語モデル	名詞，形容詞， 動詞，アルファベット
機能語モデル	接頭詞，副詞，連体詞， 接続詞，助詞，助動詞， 感動詞，その他（フィラーなど）

4.4　語彙分割 PLSA 言語モデルの概要

語彙分割を行った PLSA 言語モデルの与える確率は式（14）～（16）で表される．確率を与える単語 w が内容語，機能語のクラスに属する確率 $P(C|h)$, $P(F|h)$ は，式 (15)，(16) のように直前二単語のコンテキストに対して次に出現する内容語（または機能語）の trigram 確率を語彙クラス全体で加算することで求めている．C，F はそれぞれ内容語 (content word)，機能語 (function word) のクラスを表す．この二つの語彙クラスについて別々の PLSA 言語モデルから確率 $P(w|h,C)$, $P(w|h,F)$ が与えられる．ただし片方のクラスに属する単語は，他方では語彙に含まれないので，実質的には内容語モデルか機能語モデルのどちらかの確率が選択的に使用されることになる．

$$P(w|h) \propto P(C|h) \cdot S(\frac{P(w_i|h,C)}{P(w_i)} P(w_i|w_{i-2},w_{i-1})) + P(F|h) \cdot S(\frac{P(w_i|h,F)}{P(w_i)} P(w_i|w_{i-2},w_{i-1})) \quad (14)$$

$$P(C|h) = \sum_{w \in C} P(w_i|w_{i-2}, w_{i-1}) \qquad (15)$$

$$P(F|h) = \sum_{w \in F} P(w_i|w_{i-2}, w_{i-1}) \qquad (16)$$

$$S(x) = \frac{2}{1 + exp(-kx)} - 1 \qquad (k > 0) \qquad (17)$$

この方法を用いると従来の PLSA 言語モデルより強い文脈適応が可能だが，そのため過適応を起こしやすくなる．実際に予備実験から，過適応が性能悪化の原因になることが分っている．そこで従来の unigram rescaling （式（9））に SIGMOID 関数 $S(x)$ を組み込む．式を（17）に示す．SIGMOID 関数は unigram rescaling で全語彙について正規化前の確率を求める際に，特に大きい確率を削り，コンテキストに対する確率の平滑化を行っている．パラメータ k が大きいほど平滑化効果は強く作用する．

4.5 実験条件

実験条件を表 5，表 6 に示す．学習データには新聞記事の他に，異なるスタイルを持つテキストとして CSJ 講演書き下し文を用いる．潜在モデル数は，内容語モデルは 3.3 と同じ 50 混合とした．機能語モデルは，スタイルの特徴がトピック程多様ではないと予想されるので，2〜50 混合のモデルを構築し，混合数の変化による性能比較を行った．またテストセットには新聞記事と CSJ を 19:1 で使い，学習データの記事数の比と合わせている．SIGMOID 関数については，予備実験で最適と分った $k = 1.4$ を用いている．

4.6 実験結果

実験結果を図 6 に示す．グラフ横軸は機能語モデルの混合数である．比較のため，同じデータで学習した 50 混合の単一 PLSA モデルの性能を右端に示した．

全体では機能語モデルの潜在モデル数が大きい程性能が向上する結果となり，機能語モデル 50 混合で trigram に対する Perplexity 比率が 82.23% であった．単一モデルの 83.90% に対し，1.67% 改善した．

新聞記事と CSJ で効果を比べると新聞記事のテストセットに効果が現れている．従来の PLSA では，主に話題性の違いによって潜在モデルの特徴が別れていたが，スタイルを表す機能語を別モデルに分離することで書き言葉と話し言葉の違いが明確にモデル分けされ，書き言葉のスタイルに適応がされたためであると考えられる．

CSJ のテストセットについては全ての場合で従来の PLSA が勝る結果となったが，機能語モデルの混合数を増やしていくと CSJ での性能向上が大きいことから，モデルが詳細になっていくことで話者ごとの発話の特徴がうまくモデル化されているのではないかと考えている．

5. まとめ

PLSA を用いた言語モデルについて，学習の最適化の検討，および語彙を内容語・機能語に分割する事による，より柔軟な言語モデル適応を試みた．

その結果既存の複数の方法の中でも Inverse annealing による方法が最も優れており，特に β を 1.0 から特定の値に向けて減少させるアニーリングスケジュールで最適な言語モデルが得られた．また内容語・機能語に語彙を分割したモデルでは trigram に対する Perplexity 比率が従来法で 83.90% であったところから 82.23% に改善した．

表 5　言語モデル学習条件

	内容語モデル	機能語モデル
学習データ記事数	毎日新聞 '００から 63497 記事, CSJ から 3267 講演	
語彙数	9476+未知語	524 語
形態素数	新聞 1920 万 CSJ 400 万	新聞 770 万 CSJ 320 万
潜在モデル混合数	50	2，5，10，25，50
	N-gram 言語モデル	
学習データ	毎日新聞 '００全文及び CSJ3267 講演	

表 6　言語モデル評価条件

適応 EM 反復回数	60
β	1.00 → 0.95 → 0.90（20 反復で更新）
テストセット	毎日新聞 '０１から 95 記事, CSJ から 5 講演
SIGMOID 関数	$k = 1.4$

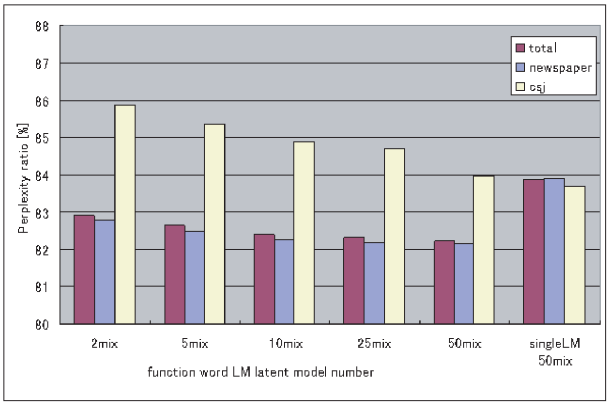

図 6　語彙を 2 分割した PLSA 言語モデルの性能

文　献

[1] Thomas Hofmann：”Probabilistic Latent Semantic Analysis ”Uncertainity in Artificial Intelligence (1999)

[2] D.Glidea and T.Hofmann：”Topic-based langage models using EM ”EuroSpeech'99, pp.2167-2170(1999)

[3] 秋田祐哉，河原達也： ”話題と話者に関する PLSA に基づく言語モデル適応 ”，信学技報 NLC2003-61，SP2003-124，pp67-72

[4] 三品拓也，山本幹雄： ”確率的 LSA に基づく ngram モデルの変分ベイズ学習を利用した文脈適応化 ”，電子情報通信学会論文誌 Vol.J87-D-II 2004-7， pp1409-1417

[5] 上田修功，中野良平： ”確定的アニーリング EM アルゴリズム ”，電子情報通信学会論文誌 Vol.J80-D-II 1997-1，pp267-276

[6] http://chasen.naist.hp/hiki/ChaSen/