確率的LSAとtrigramモデルを用いた日本語スペルチェッカ

三品 拓也
筑波大学 理工学研究科
tmishina@milab.is.tsukuba.ac.jp

山本 幹雄
筑波大学 電子・情報工学系
myama@is.tsukuba.ac.jp

1 はじめに

本稿では仮名漢字変換誤り，特に同音異義語を誤って選択した誤りを対象とした日本語スペルチェックの方法を報告する。同音異義語誤りの判定のための情報としては，局所的なものと大域的なものが利用される。例えば，次のような誤り例を考える（[] 内が正解）。

例1 新しいプロセッサを 内臓［内蔵］したマシンが発表された。

例2 ピアノを両手でうまく 引く［弾く］には練習が必要だ。

例1は，「内蔵」はサ変名詞であるが「内臓」はそうでないため，単語連鎖確率等の局所的な情報で比較的判定しやすい。例2は品詞が等しく，格構造も似ているため局所的には判定できない。この場合，距離は離れているが「ピアノ」等のキーワード情報から「弾く」の方が妥当であると判断されるべきである。実際の判定には，どちらの情報も重要であり2つの情報を融合する必要がある。これまで決定リストを用いる方法 [1, 2]，局所的なモデル（trigram モデル）と大域的なモデル（Naive Bayes）を使い分ける方法 [3] 等が提案されている。

しかし，これまでの方法は，大域的とは言ってもチェック対象単語の前後3〜10単語を見ている程度である。文献 [1] によれば，大域的モデルとして Naive Bayes 法を用いた場合，前後4単語以上の文脈を考慮すると性能が劣化することが報告されている。しかし，明らかに4単語より広い範囲の情報も有効である場合も多い。例えば，例2では記事全体が楽器や音楽について書かれていれば，記事全体に現れる多くのキーワードが「弾く」を支持する証拠として用いられるべきである。

本報告では，大域的な情報としてより広い範囲の情報（記事すべて）を用いるために確率的 LSA を利用し，trigram モデルと統合することにより同音異義語誤りを高い精度で検出・訂正できる方法を提案する。以下，2節で確率的 LSA を簡単に紹介した後に，3節で今回の提案手法で用いた trigram モデルと融合した誤り判定方法を述べ，4節で新聞記事に人為的に誤りを混入させたテキストを用いて評価を行う。評価結果として，764 の同音異義語の集合（1741 単語）を用いたとき，5%の誤りを含むテストデータに対して，再現率 95.5%・適合率 83.6%で誤りを検出できることを示す。

2 確率的 LSA

確率的 LSA（Probabilistic Latent Semantic Analysis，以下 PLSA）は，基本的に複数の unigram モデルの混合モデルと考えることができる。PLSA において，文脈 h を条件とする単語 w の確率 $p(w|h)$ は，次式で与えられる。

$$p(w|h) = \sum_{t=1}^{m} p(t|h)p(w|t) \qquad (1)$$

ここで t は unigram モデルの番号，m は混合数，$p(t|h)$ は文脈 h における t 番目のモデルの重み，$p(w|t)$ は t 番目の unigram モデルにおける w の確率である [4]。各 unigram モデルを求めるためには EM アルゴリズムが使われ，以下のような訓練データ D の尤度を最大化する（学習）。

$$\mathcal{L}(D;\theta) = \sum_{w}\sum_{d\in D} n(w,d)\log\sum_{t} p(w|t)p(t|d) \qquad (2)$$

$n(w,d)$ は記事 d 中の単語 w の出現頻度，$p(t|d)$ は記事 d における t 番目のモデルの重みである。実際の計算では局所最適解に落ちるのを避けるために deterministic annealing（または Tempered EM）法を用いる [4]。

学習を行った後である未知の文脈が与えられた場合の単語の出現確率を求めるには，文脈に応じて混合比 $p(t|h)$ を決定する必要がある（適応）。Hofmann らは EM アルゴリズムによって $p(t|h)$ を求めているが [4]，EM アルゴリズムは記事に出現した単語に過適応して性能が低下する場合がある。このため，過適応を避けるために今回は変分ベイズ学習による適応を行った [5]。

3 PLSAを利用したスペルチェック

3.1 概要

本稿で作成するスペルチェッカは，統計的言語モデルによって文脈中に出現した単語の確率とその同音異義語の確率を計算し，より確率の高い方をその文脈にふさわしい単語であると判断することでスペルチェックを行う。ここで文脈中に出現したあいまい語 w の尤度 $L(w)$ とその同音異義語 w' の尤度 $L(w')$ の比の対数 $d(w,w')$ を以下のように定義する。

$$d(w,w') = \log \frac{L(w)}{L(w')} \tag{3}$$

同音異義語が複数ある場合にはもっとも尤度の高い単語との間で d を計算し，d の値が一定の閾値以下になった場合には出現した単語は誤りであると判断する。ここで用いる閾値は訓練データからの学習で獲得する。具体的には訓練データの一部をテストデータとみなして誤りを混入させ，一度スペルチェックを行う。そこでF値（4.2節参照）最大となるような閾値を各単語（現れ）毎に設定する。

3.2 あいまい語リスト

本稿では以下のような同音異義語の集合を「あいまい語リスト」として用いた。

- 読みが同一である
- 文字数が2文字である
- 少なくとも1文字は漢字を含む
- 固有名詞ではない
- アラビア数字を含まない
- コーパス中に一定回数以上出現する

毎日新聞2000年版 [6] に100回以上出現した単語の中から上記の条件に合致する単語を抽出したところ，764の同音異義語の集合（1741単語）を得られた。

実際にプログラムで使っているのは，あいまい語リストを更に出現－置換リストに変換したものである。このリストは表2のようなリストで，あいまい語とその置換候補との組を記録したものである。このリストでは，読みが違っていても同じ表記の単語は1つにまとめている。

3.3 単語の尤度 $L(w)$ の定義

入力文中におけるある単語の出現確率を求めるには様々な方法があるが，本稿ではベイズ適応によるPLSA確率と $n$gram確率をunigram rescaling法 [7] で混合した。

表1: あいまい語リスト

読み	単語		
オリ	降り	下り	折り
フリ	降り	振り	
イライ	以来	依頼	
イガイ	意外	以外	
ヒク	弾く	引く	

表2: 出現－置換リスト

現れ	候補		
降り	下り	折り	振り
下り	降り	折り	
折り	降り	下り	
振り	降り		
以来	依頼		
依頼	以来		
意外	以外		
以外	意外		
弾く	引く		
引く	弾く		

$$L(w) = \frac{p_{PLSA}(w|h_G)}{p_{uni}(w)} p_{ngram}(w|h_L) \tag{4}$$

ここで $p_{PLSA}(w|h_G)$ はPLSAによってモデル化される大域的出現確率であり，$p_{ngram}(w|h_L)$ は $n$gram でモデル化される局所的出現確率である。PLSAにおいてはある記事中の全単語で適応を行い，$n$gram確率は前向き $n$gram確率 $p_f(w_i|w_{i-n+1}^{i-1})$ と後向き $n$gram確率 $p_b(w_i|w_{i+1}^{i+n-1})$ との幾何平均とする。

$$p_{ngram}(w_i|h_L) = \sqrt{p_f(w_i|w_{i-n+1}^{i-1}) p_b(w_i|w_{i+1}^{i+n-1})} \tag{5}$$

ただし，文の先頭や末尾で，前向き・後向き確率の計算をしようとしても履歴部分が足りない場合は，残り一方の確率のみを用いて，幾何平均は取らないことにする。

4 評価実験

4.1 実験条件

これまでに述べたような手法によるスペルチェッカの性能を確認するため，表3のような条件で実験を行った。なお，PLSAモデルは文献 [5, p. 16] における「中頻度語彙」モデルと同一である。コーパスは茶筌 [8] によって形態素解析を行い，1形態素を1単語として扱った。テ

表 3: 実験条件

パラメータ	使用コーパス	備考
PLSA モデル	毎日新聞 99 年版	語彙は頻出 103 語を除く出現頻度上位 19,000 語
trigram モデル	毎日新聞 94〜99 年版	語彙は出現頻度上位 20,000 語，Good-Turing discounting
あいまい語リスト	毎日新聞 2000 年版	100 回以上出現する同音異義語 764 組 1,741 語
テストデータ	毎日新聞 2000 年版	1%・5%・10%の割合で同音異義語を置換
閾値学習データ	毎日新聞 99 年版	5%誤りを混入させた場合の F 値を最大とする閾値

ストデータの作成に際して，ある単語に対して同音異義語が複数ある場合は，置換候補のうちで最も unigram 確率の高い単語に置換した。これは実際の誤りに近づけるためである。

4.2 評価指標

本実験では，再現率（R）・適合率（P）・F 値（F）を評価指標とする。順に以下のような値である。

$$R = \frac{\text{スペルチェッカの正解数}}{\text{テストデータ中の誤り単語数}} \tag{6}$$

$$P = \frac{\text{スペルチェッカの正解数}}{\text{スペルチェッカが検出した誤り単語数}} \tag{7}$$

$$F = \frac{2 \times R \times P}{R + P} \tag{8}$$

4.3 実験結果

実験結果を図 1 と表 4〜表 7 に示す。

図 1 は *n*gram と *n*gram+PLSA の性能差を比較したもので，閾値を全単語同一のものとして変化させたときの再現率・適合率を曲線で，各単語個別の学習済み閾値による F 値最大点を四角・丸・三角の各点で表している。前者については *n*gram+PLSA がやや有利という程度であるが，後者については *n*gram+PLSA の方が明らかに高い性能を示している。

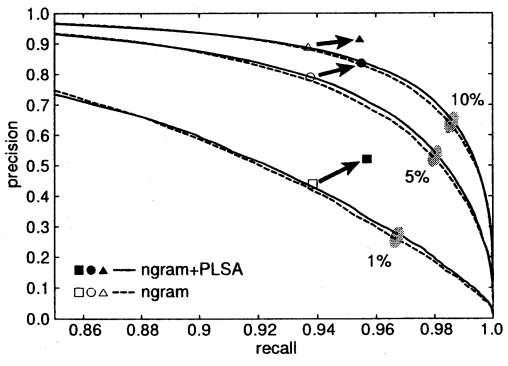

図 1: *n*gram と *n*gram+PLSA の性能比較

表 4 は，テストデータに混入させた単語の誤り比率と尤度計算に使う統計的言語モデルを変えた場合の再現率・適合率・F 値の結果についてまとめたものである。閾値は各単語個別の学習済みのものを用いた。全ての場合において *n*gram+PLSA は *n*gram を上回る性能を発揮しており，学習済みの閾値を用いた場合の性能差がはっきりと表れている。

表 5 は，一部の単語について個別に性能を比較したものである。「引く」「弾く」のような前後広い範囲に依存すると思われる単語については性能が改善されていることがわかる。また，名詞とサ変名詞の組の場合，「依頼」「以来」に対してはそれほどの性能改善がみられない半面，「内蔵」「内臓」に対しては PLSA との併用で性能が向上しており，PLSA の併用が有効に作用するものとそうでないものがあることがわかる。また，「以外」のような機能語に近い単語については PLSA を併用することで性能が悪化していることもわかる。PLSA によって F 値が改善・改悪された単語数を表 6 にまとめておく。

検出能力と訂正能力を比較したのが表 7 である。検出能力と訂正能力はほとんど同等であることがわかる。

5 おわりに

今回は同音異義語誤りを訂正・検出するため，PLSA と *n*gram を融合したモデルによるスペルチェッカについての検討を行った。その結果，*n*gram のみを用いて尤度を計算した場合に比べて常に性能が高く，特に同音異義語の組み毎に閾値を調整した場合には高い性能を発揮することがわかった。

今回は PLSA の適応を記事全体で行ったが，同一の記事中であっても距離が非常に遠い単語についてはノイズとなっている可能性がある。今後は一定の距離にある単語のみを用いて適応を行うなどして性能の向上を図りたい。

表 4: 誤り混入率と尤度計算方法を変えた場合の性能比較

混入させた誤り	尤度計算	検出	正解	R (%)	P (%)	F (%)
1% (17704)	*n*gram+PLSA	32477	16941	95.7	52.2	67.5
	*n*gram	37787	16615	93.8	44.0	59.9
	PLSA	65637	13214	74.6	20.1	31.7
5% (87874)	*n*gram+PLSA	100378	83926	95.5	83.6	89.2
	*n*gram	104269	82399	93.8	79.0	85.8
	PLSA	120246	65606	74.7	54.6	63.0
10% (177121)	*n*gram+PLSA	185197	169062	95.5	91.3	93.3
	*n*gram	186994	165968	93.7	88.8	91.2
	PLSA	185844	131934	74.5	71.0	72.7

表 5: 5%の誤りを含むテストデータに対する単語毎の性能比較

現れ	候補	混入させた誤り	*n*gram	*n*gram+PLSA
			F (%)	F (%)
内臓	内蔵	10	72.0	87.0
内蔵	内臓	9	70.6	94.1
引く	弾く	8	56.0	76.2
弾く	引く	20	51.4	71.4
景観	警官	43	64.4	92.0
警官	景観	14	33.9	77.8
以来	依頼	70	94.8	95.8
依頼	以来	292	95.6	95.6
議院	議員	451	98.2	98.5
議員	議院	13	92.9	89.7
自身	自信	114	84.3	85.8
自信	自身	363	89.9	92.1
以外	意外	36	80.0	52.2
意外	以外	202	94.8	95.3
作る	創る	18	33.3	19.0
創る	作る	126	92.5	93.4
写す	移す	16	81.3	55.3
移す	写す	262	98.5	97.6

表 6: *n*gram に PLSA を融合した場合に F 値が改良・改悪された単語数

F 値	単語種類数 (現れ)
1%以上改善	901 (61.0%)
ほぼ同等	186 (12.6%)
1%以上改悪	391 (26.4%)

表 7: 誤り検出能力と誤り訂正能力の比較

誤り	正解基準	R (%)	P (%)	F (%)
5% (87874)	検出	95.5	83.6	89.2
	訂正	94.9	83.1	88.6

参考文献

[1] Golding, A.: A Bayesian hybrid method for context-sensitive spelling correction, in *Proc. of 3rd WVLC*, pp. 39–53 (1995).

[2] 新納浩幸：表記情報をデフォルトの証拠として用いた決定リストによる同音異義語の誤り検出, 情報処理学会論文誌, Vol. 41, No. 4, pp. 1046–1053 (2000).

[3] Golding, A. and Schabes, Y.: Combining trigram-based and feature-based methods for context-sensitive spelling correction, in *Proc. of 34th ACL*, pp. 71–78 (1996).

[4] Hofmann, T.: Probabilistic Latent Semantic Indexing, in *Proc. of the 22nd Annual ACM Conference on Research and Development in Information Retrieval*, pp. 50–57, Berkeley, California (1999).

[5] 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, 信学技報 NLC2002-73, pp. 13–18 (2002).

[6] 毎日新聞社：CD-毎日新聞 1994 年版–2000 年版, 日外アソシエーツ.

[7] Gildea, D. and Hofmann, T.: Topic-based language models using em, in *Proc. of the 6th European Conference on Speech Communication and Technology (EUROSPEECH)* (1999).

[8] 松本裕治, 北内啓, 山下達雄, 平野善隆, 松田寛, 高岡一馬, 浅原正幸：形態素解析システム『茶筌』version 2.2.3 使用説明書 (2001).