音声言語情報処理 11－5
（1996．5．27）

大語彙言語データベースからのN-gram構築とタスク適応の検討

伊藤彰則，代島直人，丸山敦，加藤正治，好田正紀

山形大学　工学部

比較的規模の大きいコーパスであるEDRコーパスを用いて，N-gramの構築実験を行った．このとき，学習テキストの大きさを50万～500万単語のあいだで変化させ，語彙数・カットオフ条件などを変えて実験を行い，それぞれの場合の最適値を見出すことができた．また，EDRコーパスを学習テキストとしてタスク適応の実験を行った．適応タスクとして音響学会データベースの対話データを用い，さまざまなタスク・適応データ量・学習データ量について実験を行った．その結果，適応がない場合と比較してperplexityを1/3程度に減少させることが可能になった．

N-gram estimation from Japanese large corpus and task adaptation of N-gram

Akinori Ito, Naoto Daishima, Atsushi Maruyama, Masaharu Katoh and Masaki Kohda

Faculty of Engineering, Yamagata University

N-gram language models were constructed from EDR corpus, 5-million-word Japanese corpus. The models were investigated under various conditions about training text size, vocabulary and cut-off condition. The result of the experiments clarified the optimum condition under a certain training text size. We carried out another experiments about task adaptation. An N-gram model from a dialog was mixed with the N-gram from EDR corpus, which made about 60% reduction of perplexity.

1　はじめに

連続音声認識のための言語モデルとして，N-gramなどの統計的言語モデルが盛んに利用されている．このような統計的言語モデルを構築するためには，その確率推定のための言語データ（コーパス）が必要である．米国のARPAプロジェクトを中心とした研究プロジェクトでは，早くから言語モデル構築のためのコーパスの重要性が認識され，WSJコーパス[1]をはじめとする各種コーパスの整備と，それを用いた言語モデル構築の研究が行われてきた．一方，日本語については，コーパスの整備のために形態素解析などの前処理が必要なこともあって，大規模なコーパスによる言語モデルの研究が始まったのは比較的最近のことである[2]．そこで本稿では，比較的大規模なコーパスであるEDRコーパスを用いてN-gramモデルを構築し，その性質について調査を行った．

また，実際の音声認識システムに言語モデルを利用するには，その応用分野で話される言語表現を大量に収集する必要がある．しかし実際の応用ではこれは現実的とはいえないため，あらかじめ大量のテキストを用いて言語モデルを作成しておき，それを実際の応用分野に適応させる手法が最近注目されている．本稿では，このような手法の予備的な検討として，EDRコーパスから作った言語モデルを対話データに適応させる実験を行った．

2　大語彙データベースからのN-gram構築

2.1　語彙とカットオフ

N-gramとは，単語系列$w_1 w_2 \ldots w_n$の出現確率を

$$P(w_1 w_2 \ldots w_n) = \prod_{i=1}^{n} P(w_i | w_{i-N+1} \ldots w_{i-1}) \quad (1)$$

で近似する手法である．このとき，$P(w_i | w_{i-N+1} \ldots w_{i-1})$を計算するために，大量の言語データから統計を取り，その出現回数に応じて確率の推定を行うのが一般的である．すなわち，単

語列 $w_{i-N+1}\dots w_i$ の出現回数を $N(w_{i-N+1}\dots w_i)$ とするとき,

$$P(w_i|w_{i-N+1}\dots w_{i-1}) = \frac{N(w_{i-N+1}\dots w_i)}{N(w_{i-N+1}\dots w_{i-1})} \quad (2)$$

とする. しかし, これだけではサンプルデータに現れなかった N-gram の確率が推定できないという問題があり, またサンプル数が少ないデータについては確率推定の信頼性が低いという問題がある.

サンプルデータに現れなかった N-gram の問題については, back-off 平滑化[3]がよく用いられており, 本研究でもこれを用いている. 一方, 少数サンプルについては, 「語彙の設定」と「少数サンプルのカットオフ」という手法を用いている. 語彙の設定とは, そのモデルで取り扱う語彙を, 学習サンプルデータに出現した語彙数よりも少なく設定するということである. 例えば, 学習データ内の出現回数が 10 回以下の単語は語彙から除外するといった操作を行う. 語彙から除外された単語は, 未知語を表わす特別な単語に置き換えられる. 今回は N-gram 作成に CMU SLM toolkit[4]を用いたので, 語彙に入らなかった単語はすべて !!UNK!! という記号に置き換えられて学習される. 少数サンプルのカットオフとは, 式 (2) において, $N(w_{i-N+1}\dots w_i)$ が小さいときに, その値を捨てて back-off により確率を推定する手法である. 今回の実験では, この語彙とカットオフの条件を変えて言語モデルを作成し, その perplexity を比較してみた.

2.2 評価尺度

モデルの比較の尺度として, 補正 perplexity[11]を用いた. 通常のモデル比較には task perplexity

$$PP = P(w_1\dots w_n)^{-\frac{1}{n}} \quad (3)$$

が用いられる. しかし, 今回の実験のように語彙を変えて評価実験を行う場合, 評価テキスト中の未知語の数がモデルによって変わる可能性がある. ここで用いているモデルでは, すべての未知語に対して一定の確率を与えてしまうため, 未知語の数が違うとフェアな比較にならない. そこで, 評価テキスト中に出現した未知語の数 n_u と種類 m によって従来の task perplexity を補正したものが補正 perplexity(adjusted perplexity) である. 補正 perplexity は, 評価テキスト中に出現した未知語の種類 m と, 未知語の出現回数 n_u を用いて

$$APP = (P(w_1\dots w_n)m^{-n_u})^{-\frac{1}{n}} \quad (4)$$

で与えられる.

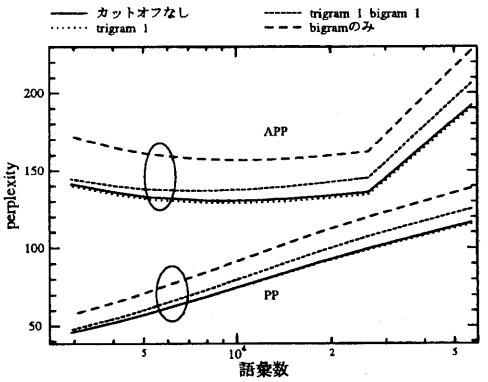

図 1: 語彙と perplexity (学習テキスト 120 万単語の場合)

2.3 実験条件

N-gram の学習・評価には, CMU SLM toolkit[4]を用いた. 学習・評価用のコーパスとして EDR コーパスを用いた. EDR コーパス 207802 文のうち, 9895 文 (240085 単語) を評価用として固定し, それ以外 (最大 197907 文, 4812911 単語) を学習用に用いた. カットオフとしては, 次の 4 種類を試している.

1. カットオフなし
2. 出現頻度 1 回の trigram をカットオフ
3. 出現頻度 1 回の bigram と trigram をカットオフ
4. すべての trigram をカットオフ (bigram モデル)

また, 語彙数を約 5000 ～ 65000 のあいだで変化させて実験を行った.

2.4 実験結果

カットオフ・語彙数を変化させて perplexity を求めた結果を図 1, 2に示す. 図 1は学習テキスト数約 120 万単語, 図 2は学習テキスト数約 480 万単語の場合である. PP は従来の task perplexity, APP は補正 perplexity である. カットオフ条件がいずれの場合でも, 傾向としては同じであった.

補正なしの perplexity は, いずれの場合も語彙数が大きいほど大きくなる傾向が見られる. これは, ここで使った N-gram の確率付与の方法が, 「既知語＋未知語 1 種類」となっていることに起因する[11]. すなわち, 未知語をすべて 1 つにまとめてモデル化しているため, 未知語が多い場合には正しい確率推定になっていないのである. 一方, 補正 perplexity の場合, 語彙数 10000 ～ 20000 単語付近

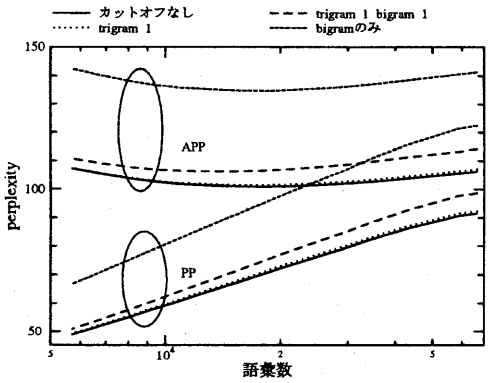

図 2: 語彙と perplexity (学習テキスト 480 万単語の場合)

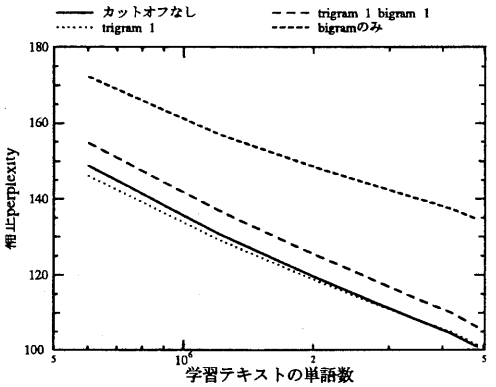

図 3: 学習テキスト量と最適な perplexity

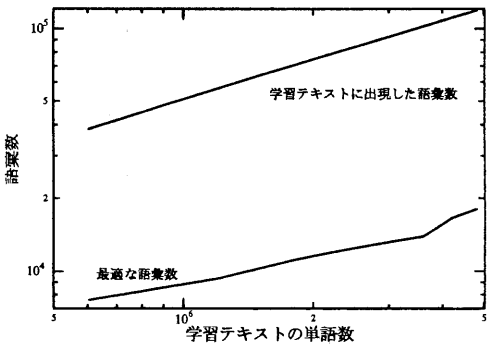

図 4: 学習テキスト量と語彙数

でperplexityが最小値をとる傾向が見られた．補正perplexityの場合，語彙数が少ない場合は未知語が増えるために，補正によってpreplexityが増加する．一方，語彙数が多い場合には，1単語に割り当てられる確率が小さくなるためにperplexityが増加する．今回の実験では語彙10000～20000付近が最適値となったが，この値は評価テキストの大きさ，未知語の数と種類の影響を受けて変動すると考えられる．カットオフの条件を比べると，「カットオフなし」と「出現1回のtrigramをカットオフ」の場合がほぼ同程度の性能で，最も良い結果となった．

学習テキストの大きさを変えた場合の，補正perplexityの最小値を図3に示した．このグラフから，学習テキストを増やすと補正perplexityが単調減少することがわかる．今回の実験では，perplexityが飽和するまでには至らなかった．カットオフ条件を比較してみると，学習テキストが少ない場合はtrigramをカットオフしたモデルが良く，300万単語付近を境にカットオフなしの方が性能が良くなっていた．それぞれの学習テキスト量での異なり語彙数と，最適な補正perplexityを与える語彙数を図4に示す．最適な語彙数は学習テキストの増加とともに増加するが，学習テキストの異なり語彙数の増加よりはゆるやかであることがわかる．また，それぞれの学習テキスト量での最適語彙数における未知語の数と種類を図5に示した．学習テキスト量が増えるにしたがって，未知語の少なくなっていることがわかる．

3 N-gram言語モデルのタスク適応

3.1 タスク適応－従来法

N-gramを構成する場合には，コーパスからの統計情報を元にして確率の推定をするのが一般的である．こうして作られたN-gramは，元のコーパスと似た文章の確率推定には良いモデルとなるが，元のコーパスと異なる分野の文章のモデルとしてはあまり良いものにならない．したがって，ある特定分野の音声認識のための言語モデルの構築のためには，その分野の文章を大量に集める必要がある．しかし，それぞれの応用分野について大量の文章をあらかじめ集めておくというのは非現実的である．そこで，最初は広い分野のモデルを作っておき，それを特定のタスクに適応させるという研究が行われている．

タスク適応の方式は，現在のところ大きく3つに分類することができる．1つは，認識結果を用いて言語モデルを逐次更新していく方法である[6, 8]．2

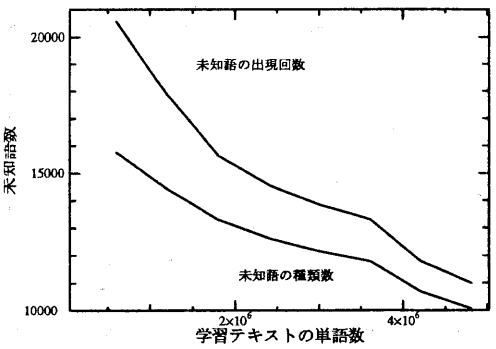

図 5: 学習テキスト量と未知語数

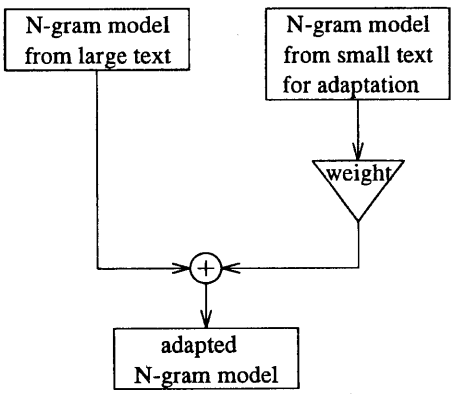

図 6: N-gram のタスク適応

つめは，あらかじめいくつかの分野を想定しておき，入力がどれにあたるかを推定する方法である[9]．3つめは，適応したい分野のテキストを少量用意しておき，それを使ってモデルを変更する手法である[10]．少量テキストによる適応と逐次適応を組み合わせる試みもなされている[7]．

3.2 本研究での方法

本研究では，最初に少量テキストを用意する場合について検討を行った．すなわち，大量の一般的なテキストから作成したN-gramと，少量・特定タスクのテキストから作成したN-gramを適当な割合で混合することによって，その特定タスク向きのN-gramを作ろうという試みである．混合の方法として，今回はもっとも簡単な「重みつき混合」を試してみた．これを図6に示す．まず，大規模なコーパス（ここではEDRコーパス）からN-gramを求めておく．次に，適応したい少量のテキストからN-gramを作り，それに重みをかけて混合する．例えば重みを10にした場合，適応テキスト中に1回出現したN-gramは，大規模コーパス中で10回出現したN-gramと同様に扱われる．

3.3 実験条件

実験には，学習テキスト・適応テキスト・評価テキストの3つのテキストを用いている．学習テキストとしては，EDRコーパスを用いた．テキストの量として，次の5種類を試している．これを，以下の実験では2500文，5000文，10000文，20000文，40000文セットと呼ぶ．

文数	単語数	単語種
2597	63437	9375
5195	125960	15226
10390	251767	23578
20780	504132	35431
41560	1008317	52445

評価用テキストには，日本音響学会データベースの模擬対話のうち，京都観光案内1 (KIT0001D) を用いた．テキスト量は117文，単語数1659，単語種330である．適応用には，同じく音響学会データベースの対話を用いており，次の4種類の組み合わせを試している．

対話テキスト	文数	単語数	単語種
京都観光案内2 KIT0002D	117	1488	290
筑波観光案内 TSU0005D	79	1789	388
スポーツ NTU1002D	136	2173	385
オートバイ＋スキー旅行 OSA0007D+KIT0005D	363	4130	643

なお，学習テキストと評価・適応用テキストでは，形態素解析基準が異なる．

N-gramの作成には，前節と同じCMU SLM Toolkitを用い，backoff trigramモデルを構築した．

3.4 各種テキストによる適応の効果

まず，1種類の学習テキストから学習されたN-gramについて，様々なテキストを適応させ，評価テキストへの適応の様子を調べてみた．学習にはEDRコーパス20000文セットを用い，語彙数は3693単語(10回以上出現した単語)とした．このテキストから作成したtrigramに，「京都観光案内2」「筑波観光案内」「スポーツ」「オートバイ＋スキー旅行」の4種類のテキストを適応させてみた．この結果得られた補正perplexityを図7に示す．その結果，評価テキストと同じカテゴリーの対

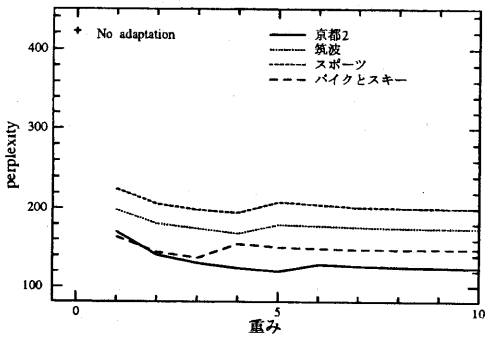

図 7: 様々なテキストによる適応の効果

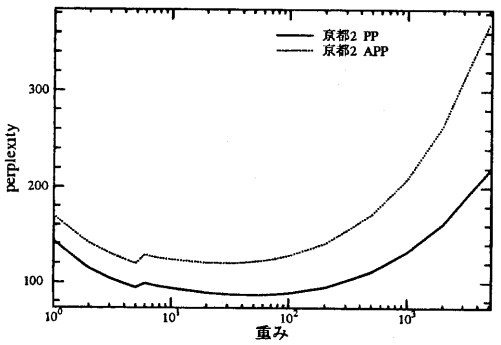

図 8: 重みを大きくした場合の適応の効果

話「京都 2」が最も良い性能を示した．カテゴリーの違う対話では，テキスト量の多い「オートバイ＋スキー旅行」が比較的良い性能であった．これは，それほどカテゴリーが近くないテキストであっても，量が多ければある程度の効果があることを示唆している．いずれの場合も，重み 3 ～ 5 で特異的な最小値を示す．この原因はまだ良くわかっていない．

次に，京都観光案内 2 について，もっと大きな重みをかけて性能の変化を調べた．これを図 8に示す．最初の特異的な最小値を除けば，重み 20 ～ 50 で perplexity が最小になることがわかる．perplexity は重みが 50 のとき，補正 perplexity は重みが 5 のときが最小であった．

3.5 学習テキストの語彙数を変えた場合

次に，EDR コーパス 20000 文セットについて，語彙を 2 回～ 20 回以上出現した単語に制限して，語彙による違いを調べた．適応テキストは京都観光案内 2 である．なお，適応テキストの方には語彙による

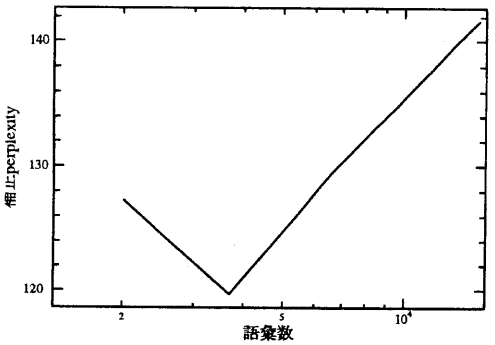

図 9: 語彙数を変えた場合の適応の効果

制限は加えていない．出現回数と語彙数の関係は次のようになっている．

2 回以上出現	15274
5	6619
10	3693
20	2012

このとき，語彙数と perplexity の最小値の関係を図 9に示す．この結果から，語彙が約 3500(10 回以上出現) の場合に最も性能が良いことがわかる．この最適値は，学習テキストの大きさにもよるのかもしれない．今回はそこまでの調査は行っていない．

3.6 学習テキストの大きさを変えた場合

最後に，学習テキストの大きさを変えた場合の効果を調べた．学習テキストに 2500, 5000, 10000, 20000, 40000 文セットを用い，京都観光案内 2 に対して適応を行った．語彙はすべて 10 回以上出現した単語としている．同時に，EDR コーパスをまったく使わずに，京都 2 だけから N-gram を作成した場合も試してみた．これを図 10に示す．この図では補正 perplexity の場合を示した．ただし，2500 文セットで重み 3 の場合に，Turing-Good の推定がうまくいかずエラーになったため，そのデータは除外している．この結果から，学習テキストが 5000 で重みが 3 の場合に最も良い性能を示すことがわかる．学習テキスト 5000,2500 の場合に，適応テキストのみから作った N-gram よりも良い結果が得られている．興味深い点として，学習テキストが大きくなるにつれて，最小のピークが重みの大きいほうへシフトする現象が見られた．

4 まとめ

EDR コーパスを用いて N-gram を構築し，各種の作成条件と perplexity の関係について調査した．その結果，補正 perplexity を評価値とした場合，学習テ

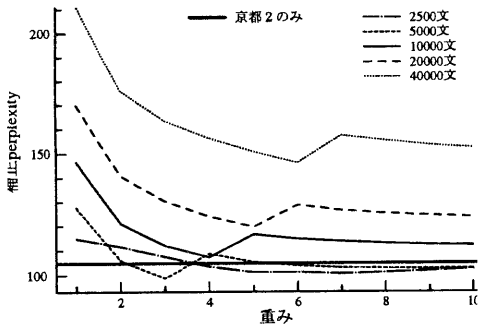

図 10: 学習テキスト量を変えた場合の適応の効果

キストの大きさに応じた最適な語彙数があるということが明らかになった．また，それぞれの学習テキストにおける補正 perplexity の最適値を調べた結果，今回用いたテキスト量 (約 19 万文，480 万単語) ではまだ十分な量とはいえず，さらに学習テキストを増やすことによって perplexity を減少させることができるという見通しが得られた．また，EDR コーパスと音響学会データベースを用いて，N-gram のタスク適応実験を行った．その結果，EDR コーパス単独・適応テキスト単独の場合を越える性能を得ることができた．また，目的となる分野を外れたテキストであっても，ある程度近い，あるいはある程度の量があるテキストで適応の効果が見られた．

今回は，各種の条件に対して言語モデルの性能を見てきたわけだが，まだ考慮していない要素がいくつかある．今後はこれらの条件についてさらに詳しい解析をするとともに，自動的に最適値を見つけるような手法[8]を検討していきたいと考えている．

参考文献

[1] D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR Corpus", Proc. ICSLP-92, pp. 899–902 (1992)

[2] 大附克年，森岳至，松岡達雄，古井貞熙，白井克彦：「新聞記事を用いた大語彙連続音声認識の検討」，信学技報 NLC95-55, SP95-90 (1995-12)

[3] S. M. Katz: "Estimation of probabilities from sparse data for language model component of a speech recognizer", IEEE Trans. ASSP, vol. 35, pp. 400–401 (1987)

[4] R. Rosenfeld: "The CMU statistical language modeling toolkit and its use in the 1994 ARPA CSR evaluation", Proc. ARPA Spoken Language Systems Technology Workshop, pp. 47–50 (1995)

[5] R. Schwaltz, L. Nguyen, F. Kubala, G. Chou, G. Zavaliagkos and J. Makhoul: "On using written language training data for spoken language modeling", Proc. ARPA Human Language Technology Workshop, pp. 94–98(1994-3)

[6] R. Kuhn and R. de Mori: "A Cache-Based Natural Language Model for Speech Recognition" IEEE Trans. PAMI, vol. 12, No. 6, pp. 570-583 (1990)

[7] S. Matsunaga, T. Yamada and K. Shikano: "Task adaptation in stochastic language models for continuous speech recognition", Proc. ICASSP-92, vol. I, pp. 165–168 (1992)

[8] R. Rosenfeld: "A hybrid approach to adaptive statistical language modeling", Proc. ARPA Human Language Technology Workshop, pp. 76–81 (1994-3)

[9] R. Iyer, M. Ostrndorf and J. R. Rohlicek: "Language modeling with sentence-level mixtures", Proc. ARPA Human Language Technology Workshop, pp. 82–87(1994-3)

[10] A. I. Rudnicky: "Language modeling with limited domain data", Proc. ARPA Spoken Language Systems Technology Workshop, pp. 66–69 (1995-1)

[11] J. Ueberla, "Analysing a simple language model – some general conclusion for language models for speech recognition", Computer Speech and Language, vol. 8, No. 2, pp. 153–176 (1994-4)