Vol.2010-SLP-82 No.19
2010/7/24

単語重要度を用いた N-gram 補完手法が与える音声認識性能の調査

島田敏明[†1] 西村竜一[†1]
河原英紀[†1] 入野俊夫[†1]

単語 3-gram モデルは，テキストコーパスから統計的手法に基づいて構築される．しかし，テキスト量が少ないと統計量を正しく算出できない．そこで本研究では，Google N-gram データに含まれる 3-gram エントリを用いて，3-gram 情報の補完を行った．3-gram エントリを選別せず補完すると，3-gram エントリ数が爆発的に増加する問題が発生する．そこで，提案手法では TF・IDF 指標と Yahoo!関連キーワードから算出した単語重要度に基づき，追加する 3-gram エントリを選別した．これにより，重要性の低い 3-gram エントリの追加と，エントリ数の爆発的増加を防ぐ事が出来た．評価では，CSJ コーパスを用いて認識実験を行った．その結果，補完前より単語正解精度において 1.64%の向上が得られた．

Comlementing 3-gram information using the Google Japanese N-gram database and term weighting

Toshiaki Shimada,[†1] Ryuichi Nisimura,[†1]
Hideki Kawahara[†1] and Toshio Irino[†1]

We have developed a method that utilizes the Google N-gram database to complement 3-gram entries in a language model. Our aim was to improve the accuracies of LVSR systems even when a 3-gram model trained on short texts is being used. This method is based on 3-gram occurrence information in external web documents and consists of three main steps. First, 3-gram entries are searched in the Google database. Secondly, 3-gram appearance counts are normalized on the basis of the ratio of total number of 3-gram entries. Lastly, 3-gram entries are selected on the basis of keywords. To prevent the addition of redundant or not relevant entries, 3-gram entries without a keyword are excluded to calculate 3-gram probabilities. The keywords were composed by measuring the TF-IDF weights and employing the web API of Yahoo! Japan. Experimental results confirmed 1.64% improvement in a recognition accuracy using the CSJ Corpus.

1. はじめに

本稿では，単語 3-gram 言語モデルを対象に Google N-gram[1)] を用いた 3-gram エントリの補完手法について検討する．単語 N-gram は，大語彙連続音声認識システムにおける言語モデルとして広く用いられている．この N-gram モデルは，学習用コーパスの特徴（語彙，言い回し等）を引き継ぐ傾向がある．その為，トピックに特化したモデルを構築する事で音声認識精度の向上が可能である．しかし，あらゆる発話内容を想定して，コーパスを準備しておくことは難しい．学習用コーパスが少ない場合，学習用コーパスに存在しない単語の組に対しては出現確率を推定する事はできない．その場合，音声認識システムはその N-gram エントリを受理できない．これを防ぐために，学習用コーパスに存在しない N-gram に対して確率の割り当てが行なわれる．その代表的な手法が，バックオフ平滑化法[2)] である．しかし，バックオフ平滑化法により推定された N-gram 確率の信頼性は高いとは言い難い．そのため，信頼性の高い確率推定を行うには，不足する N-gram の情報を補う必要がある．

また，昨今，言語モデルの研究において，Web 上の各種リソース，及び集合知の情報の活用の重要性が増している．例えば，固有名詞や難読語の収集及び，読み獲得の方法として，はてなキーワードや Wikipedia 等の Web 集合知の利用が進んでいる[3)4)]．翠らにより Web 等からテキストの自動収集による N-gram 言語モデルの構築が提案されている[5)]．また，鈴田らによってトピックに依存した小規模の N-gram 言語モデルを Web 上のリソースを利用して構築する方法の提案がなされている[6)]．本研究では，これらの研究とは異なる方法で Web 上の資源を利用し，3-gram 情報を補うことで音声認識精度の向上を目指す．

2. Google データを用いた N-gram 補完手法

学習用コーパスに不足する 3-gram 情報を補うには，他の用意したデータから必要な情報を流用すれば良い．その対策として，前述のように Web 集合知を用いた言語モデルの構築手法が提案されている．しかし，大規模な Web テキスト収集などはネットワーク資源に負

†1 和歌山大学
Wakayama University

Vol.2010-SLP-82 No.19
2010/7/24

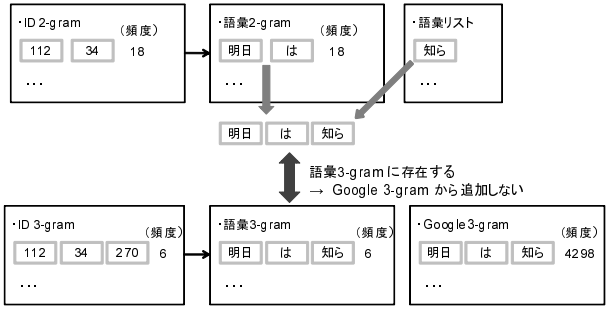

図 1 3-gram 補完の手順（3-gram を追加しない場合）

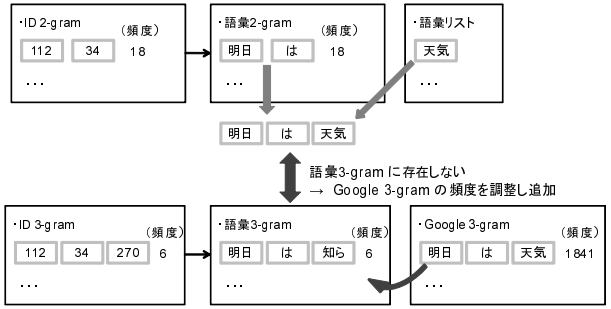

図 2 3-gram 補完の手順（3-gram を追加する場合）

担の大きい処理が必要で, 利便性に欠ける．そこで, 学習テキストに含まれない 3-gram を既存のデータベースである Google N-gram データ から抽出し，補完する方法を検討した．Google N-gram データは Web 上の 200 億文から抽出したデータである．単語分割には形態素分析ツール MeCab[7] が用いられている．配付ファイルの中には 1-gram から 7-gram までのエントリとその頻度情報が記載されている．

提案手法である，3-gram エントリの補完手法を図 1，図 2 に示す. まず，用意した 2-gram エントリの後に，単語辞書の単語 1 つを接続し，3-gram エントリを構成する．次に，図 1 のように，3-gram エントリが学習用コーパスにすでに存在する場合は，学習元に存在する頻度情報をそのまま 3-gram 確率の学習に用いる．図 2 のように，学習用コーパスには存在しないが，Google 3-gram に存在する場合，Google 3-gram 内に記述された頻度を調整して 3-gram 確率の算出に新たに加える．この過程において，単語辞書は 3-gram エントリの追加前後で変化しない．

この際，追加する 3-gram エントリ w_i^{i-2} の頻度 $C(w_i^{i-2})$ には，式 (1) から算出した値を用いる．

$$C(w_i^{i-2}) = C_{Google}(w_i^{i-2}) \times \frac{N_{Original}}{N_{Google}} \times \alpha \begin{cases} entry & if \quad C(w_i^{i-2}) \geq 1 \\ non \quad entry & otherwise \end{cases} \tag{1}$$

ここで，C_{Google} は，Google N-gram 内に記述された 3-gram 頻度を示す．また，$N_{Original}$ は，学習用コーパスから求められた 3-gram 頻度の総数，N_{Google} は Google N-gram の 3-gram 頻度の総数である．つまり，学習用コーパスと Google N-gram がそれぞれ含む 3-gram の総数を用いて，頻度のスケールを調整したことになる．

ここで，α は調整用の重み係数である．学習用コーパスに対して，Google N-gram のボリュームが圧倒的に大きく，$\alpha = 1$ のとき，膨大な数の 3-gram エントリが，3-gram 確率の算出に追加される．このため，追加後の 3-gram モデルから学習元テキストの特徴が失われる可能性がある．本研究では $\alpha < 1$ を掛けることで，それを制限することにした．調整用の重み係数の値の決定については，4.1 節の音声認識実験の中で述べる．

なお，計算で頻度 $C(w_i^{i-2})$ が 1 未満になった 3-gram エントリの追加は行なわず，学習から除外した．そして，3-gram エントリとその頻度情報を追加した後，単語 3-gram モデルを Palmkit[8] を用いて構築した．図 3 に 3-gram 言語モデルと単語辞書の構築手順を示す．このように，ID 3-gram ファイルにエントリとその頻度情報の追加する事で提案手法を実装した．

3. 単語重要度を考慮した 3-gram 補完手法

ここまでの説明では，単語ごとの重要度を考慮していないため，重要性の低い 3-gram エントリも追加の対象としている．これを防ぐために，単語重要度に基づき追加 3-gram エントリの選別を行う事とした．単語重要度を用いた N-gram 補完手法は，認識対象のトピックに関連のある重要単語を含む 3-gram エントリのみを追加の候補とする事で実現した．

3.1 重要単語リストの作成

本研究では, 単語重要度に基づいて重要単語リストを作成した．ここで，音声認識システムの語彙リストに含まれる認識対象トピックと関係性が高い名詞を重要単語とする．その指標には，以下に述べる Yahoo!関連キーワードと TF・IDF を利用した．

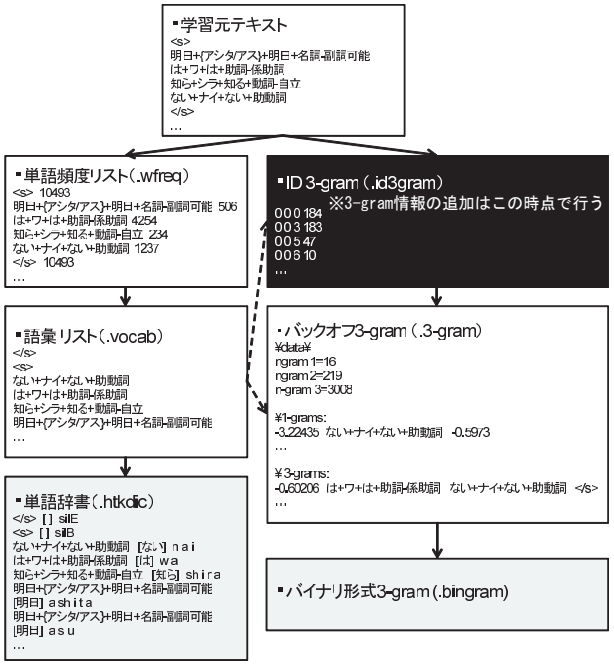

図 3 ツールキット (Palmkit) を用いた 3-gram 言語モデルと単語辞書の構築手順
(図中で黒く塗りつぶした部分が変更部分である)

3.1.1 Yahoo!関連キーワード

Yahoo!関連キーワード[9] は Yahoo!JAPAN が提供するサービスであり，ユーザの入力する検索クエリである単語に対して, 関連のある単語を出力する Web API サービスである．Yahoo!検索の利用者が入力したキーワードや，その組み合わせを機械的に収集・処理した結果をもとに，検索キーワードの組み合わせなどを出力する．出力結果は最大 100 件まで表示することができる．なお，関連度の高い単語が上位から順に出力される．例として「政治」と入力した時の Yahoo!関連キーワードの出力結果（抜粋）を図 4 に示す．

3.1.2 TF・IDF 指標

TF・IDF 指標[10] は文章中の特徴的な単語（重要とみなされる単語）を抽出するための指標である．主に情報検索や文章要約などの分野で利用される．TF・IDF は, 式 (2) のように TF（Term Frequency: 単語の出現頻度）と IDF（Inverse Document Frequency:逆出

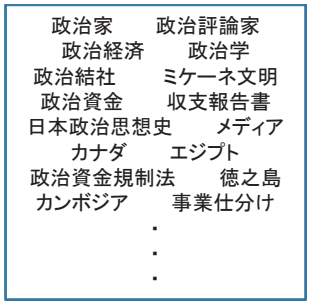

図 4 Yahoo!関連キーワードの出力結果（「政治」の例・抜粋）

現頻度）の 2 つの指標で計算される．

$$TF \cdot IDF = TF(t) \cdot \log \frac{N}{DF} \tag{2}$$

式 (2) において TF は，ある 1 つの学習コーパス中に含まれる指定した単語の出現頻度である．また，DF には，指定する単語を検索エンジン (Yahoo! JAPAN) にて検索した時のヒットページ数を指定した．例えば，検索ワードが「政治」の時，DF は約 1,130,000,000 となった．N は，検索エンジン (Yahoo! JAPAN) に存在するウェブページの総数である．N は DF の値より十分大きい値とした．DF の値が N の値より大きくなり，式 (2) より TF・IDF の値が負となるのを防ぐためである．本研究では，公開されている情報を参考に，N を 19,200,000,000（192 億）に設定した．

3.2 単語重要度を用いた Google 3-gram の補完

図 5 に，選別の過程を追加した Google 3-gram の追加手順を示す．追加する 3-gram エントリ w_i^{i-2} の頻度 $C(w_i^{i-2})$ は，式 (1) を用いて導出する．追加の候補となる Google N-gram 中の 3-gram エントリに，重要単語が含まれている場合は，式 (1) のこれまでと同様，頻度 $C(w_i^{i-2})$ に調整用の重み係数 α を与える．重要単語が 1 つも含まれていない場合は，調整用の重み係数 α の値を 0 とし，追加から除外した．これにより，追加する 3-gram エントリの数を制限できる．以上より，重要単語を含む 3-gram エントリのみを追加することが可能になった．

4. 評 価 実 験

提案手法の有効性を示す為に，日本語話し言葉コーパス（CSJ コーパス）[12] を用いた評

Vol.2010-SLP-82 No.19
2010/7/24

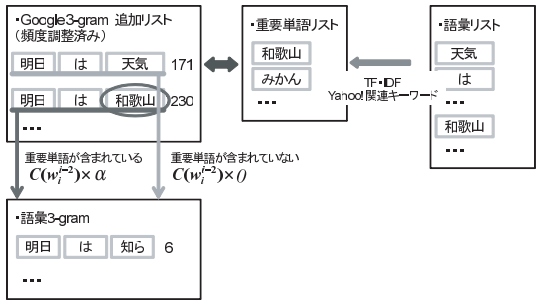

図 5 重要単語による選別の過程を加えた Google 3-gram の追加手順

表 1 言語モデルの仕様 (認識トピック「政治」)

言語モデル	学習データ	「政治という単語を 1 つでも含む CSJ 学会講演・模擬講演　合計 140 講演
	総文章数	9,331
	総単語数	414,960
	3-gram エントリ数	250,702
	Back-off 手法	Witten-Bell 法[11]
単語辞書	異なり語彙数	17,070

価実験を行った.

4.1 実験条件

本実験では音声認識エンジンに Julius（version 4.1.5）[13][14] を使用した．音響モデルは, CSJ 付属のトライフォンモデルを使用した. 本実験で用いた学習用コーパスは, CSJ コーパスに含まれる講演の中で,「政治」という単語が 1 語でも含む講演のテキストである．表 1 は, Google 3-gram のエントリを追加していない従来手法で作成した言語モデルの仕様である.

重要単語リストは, TF・IDF 指標と Yahoo!関連キーワードの単語重要度をもとに名詞のみを抽出して作成した. TF・IDF 指標で得た上位 1000 語の単語と Yahoo!関連キーワードで得た 100 語の単語を合わせて重要単語リストとした.

テストセットについては，学習用コーパスの中から節情報があり，機械的に文章に分割可能な「政治」に関する 8 講演を用意した．使用した音声データは，男性 6 名，女性 2 名の講演発話の音声である. テストセットの仕様を表 2 に示す.

3.2 節で示した Google N-gram の頻度を調整するための重み係数 α には 0.001〜0.01

表 2 テストセットの仕様

講演数	話者数	総文章数	総単語数
8（学会講演 2，模擬講演 6）	男性 6 名・女性 2 名	636	16,072

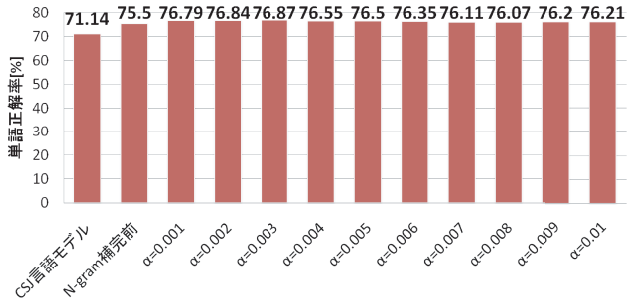

図 6 評価実験結果（単語正解率）

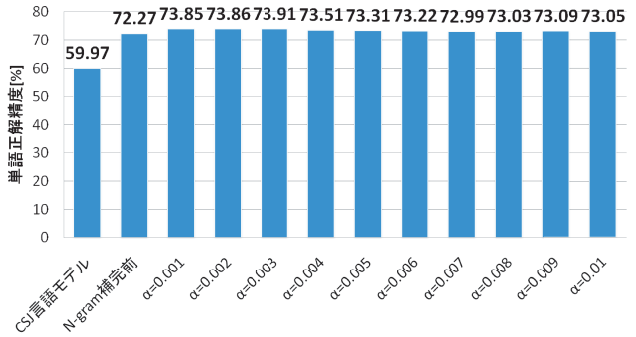

図 7 評価実験結果（単語正解精度）

を用いて比較した．また，比較対象として CSJ に付属する言語モデルと，従来手法で作成した補完する前の言語モデルでも評価実験を行った.

4.2 実験結果

実験結果を図 6-9 に示す．図 6 は，CSJ に付属する言語モデル，従来手法で作成した補完する前の言語モデルと，提案手法で作成した 3-gram 補完後の言語モデル（α=0.001〜

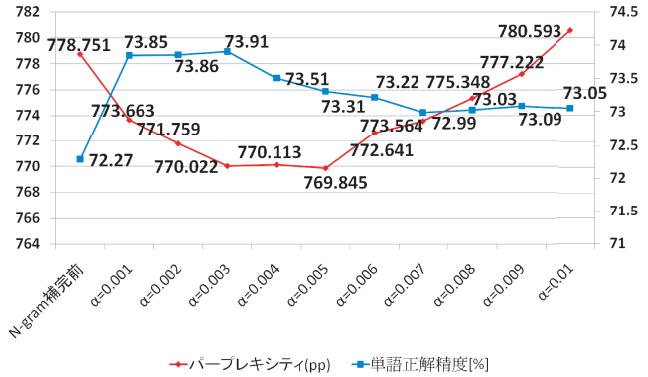

図 8 評価実験結果（テストセットパープレキシティの変化と単語正解精度との比較）

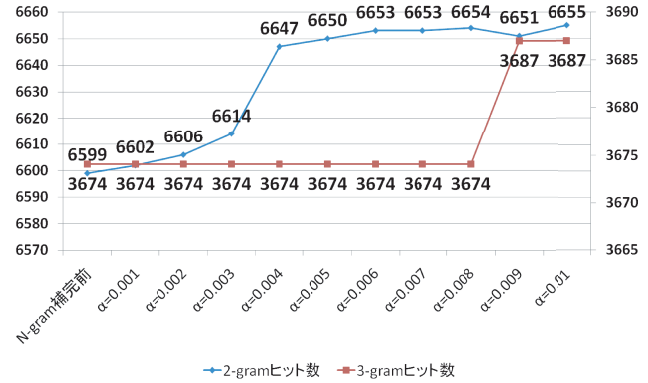

図 9 評価実験結果（2-gram ヒット数と 3-gram ヒット数）

α=0.01）の単語正解率を示す．また，図 7 は，同じく単語正解精度である．図 8 は，3-gram テストセットパープレキシティと単語正解精度である．図 8 中の左軸の数値はテストセットパープレキシティ，右軸は単語正解精度を示す．図 9 では，同様に N-gram 補完前の言語モデルと，N-gram 補完後の言語モデル（α=0.001〜0.01）のテストセットに対する 3-gram ヒット数・2-gram ヒット数を比較した．左軸の数値が 2-gram ヒット数，右軸が 3-gram ヒット数である．図 10 は言語モデルに含まれる 3-gram エントリ数の推移を示したものである．

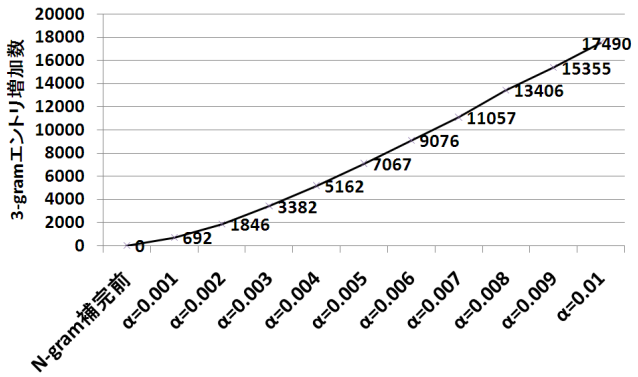

図 10 言語モデル中の 3-gram エントリ数の変化

図 7 より単語正解精度は，重み係数 $\alpha = 0.003$ のときに最も精度を向上させることができた．この時，従来手法と比較して， 1.64%の向上を得られた．有意差検定[15] を行った結果，有意水準 1%で有意差が認められた．また，CSJ 言語モデルと補完後の言語モデルを比較した場合は，同じく α=0.003 の時に 13.94%の精度向上を得る事ができた．

次に単語正解率について述べる（図 6）．単語正解率は，単語正解精度と同様に α=0.003 の時に最も精度を向上させることができた．従来手法から，1.37%の向上を得た．こちらも有意差検定[15] を行った結果，有意水準 1%で有意差が認められた．また，CSJ 言語モデルと補完後の言語モデルを比較した場合は，同じく α=0.003 の時に 5.73%の精度向上がみられた．

補完前・補完後のどちらの言語モデルより，CSJ 言語モデルの認識率は低くなっている．これは，CSJ 言語モデルが幅広いトピックの講演に対応するため，テストセットの特徴的なトピック (政治) と合致しなかったためと考えられる．

続いて 3-gram テストセットパープレキシティと 2-gram ヒット数・3-gram ヒット数についてまとめる（図 8・9）．パープレキシティは，α=0.005 のとき最も低くなり，それ以降，悪化の傾向にある．また図 8 より，パープレキシティの減少と単語正解精度の向上は相関性がある事が分かる．パープレキシティの増加は，追加 3-gram エントリ数の増加が原因であると考えられる．無用な 3-gram エントリが登録された事で，言語モデルが音声認識システ

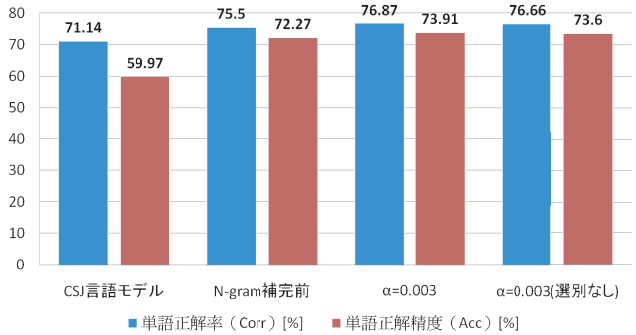

図 11 重要単語選別の有無による性能比較結果

ムの最尤解探索部へ与える情報量が減少した事が原因と考えられる．しかし，追加 3-gram エントリがテストセットに合致した場合，2-gram ヒット数・3-gram ヒット数は増加しており，α=0.001〜0.005 でのテストセットパープレキシティを削減する事が出来た．

以上の結果をまとめる．単語正解精度，単語正解率は，α=0.003 の時，最も精度を向上させることができた．また，テストセットパープレキシティは α=0.005 の時，最も良い結果であったが，α=0.006 以降は 3-gram エントリ数が増えるごとに増加（悪化）する傾向があった．

最後に重要単語による選別をせずに 3-gram エントリを追加した場合の結果も参考にここに述べる．図 11 に示すように，選別をしない場合は選別した時よりも単語正解率は 0.21%，単語正解精度は 0.31%精度が低下した．このことから重要単語で選別する事は重要であるといえる．

5. ま と め

本稿では単語 3-gram モデルの学習テキストで不足した 3-gram 頻度情報を，Google N-gram データから補う手法を提案した．その際，追加対象の 3-gram エントリを選別するため，Yahoo!関連キーワードと TF・IDF 指標から求めた単語重要度に基づいて重要単語リストを作成した．そして，重要単語の有無によって 3-gram エントリを選別した後，頻度情報を調整した．重要単語を含む 3-gram エントリのみを追加することで，重要性の低い 3-gram エントリの追加と，追加 3-gram エントリの爆発的な増加を抑えることができた．その結果，従来手法（3-gram エントリの追加なし）よりも音声認識の精度を向上させることができた．今後は，追加する 3-gram エントリの選別を改善する新たな指標を検討する予定である．

謝辞 本研究の一部は，科学研究費補助金および和歌山大学 H22 年度学長裁量経費の支援を受けた．

参 考 文 献

1) 工藤拓, 賀沢秀人：Web 日本語 N グラム第一版，言語資源協会, 2007.
2) Slava M. Katz : Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer, Acoustics, IEEE Transactions on Speech, and Signal Processing, vol.35, No.3, pp.400–401, 1987.
3) 松原 他：ポッドキャスト音声認識の性能向上手法: 集合知によって更新される Web キーワードを活用した言語モデリング， 情報処理学会研究報告， 2008-SLP-71-6, 2008.
4) 中野 他：集合知を利用した語彙情報の収集・共有・管理システム， 情報処理学会研究報告，2008-SLP-71-12, 2008.
5) 翠 他：ドメインとスタイルを考慮した Web テキストの選択による音声対話システム用言語モデルの構築， 電子情報通信学会論文誌， vol.J90-D， No.11， pp.3024–3032 , 2007.
6) 鈴田 他：Web 知識を二段階利用した単語辞書の更新手法， 日本音響学会 2008 年春季研究発表会講演論文集，pp.123–124 , 2008.
7) 工藤 拓：MeCab: Yet Another Part-of-Speech and Morphological Analyzer, ”http://mecab.sourceforge.net/”, 2006.
8) 伊藤彰則, 好田正紀：単語およびクラス N-gram 作成のためのツールキット, 電子情報通信学会技術研究報告, SP2000-106, pp.67–72, ”http://palmkit.sourceforge.net/”, 2000.
9) Yahoo!デベロッパーネットワーク, “ http://developer.yahoo.co.jp/ ”.
10) 天野 他：IT TEXT 自然言語処理， オーム社, 2007.
11) Ian H. Witten, Timothy C. Bell: The zero-frequency problem: estimating the probabilities of novelevents in adaptive text compression, IEEE Transactions on Information Theory, vol.37, No.4, pp.1085-1094, 1991.
12) 国立国語研究所, 情報通信研究機構： 日本語話し言葉コーパス, 2004.
13) A. Lee, T. Kawahara, K. Shikano：Julius An Open Source Real-Time Large Vocabulary Recognition Engine，Proc. 7th European Conference on Speech Communication and Technology （EUROSPEECH2001）， pp.1691-1694, 2001.
14) 河原達也, 李晃伸：連続音声認識ソフトウエア Julius，人工知能学会誌, Vol.20, No.1, pp.41–49, “ http://julius.sourceforge.jp/ ” ,2005.
15) 中川聖一，高木英行：パターン認識における有意差検定と音声認識システムの評価法, 日本音響学会誌, vol.50, No.10, pp.849–854, 1994.