WWWテキストのみを用いたオープンドメイン質問応答用音声認識言語モデル

Varga István[1]　大竹清敬[1]　鳥澤健太郎[1]　De Saeger Stijn[1]　松田繁樹[2]　林輝昭[2]
独立行政法人　情報通信研究機構　MASTARプロジェクト
[1]言語基盤グループ　[2]音声コミュニケーショングループ
{istvan, kiyonori.ohtake, torisawa, stijn, shigeki.matsuda, teruaki.hayashi}@nict.go.jp

概 要

本論文では、オープンドメインな音声質問応答システムで用いる音声認識言語モデルをWWWテキストのみから作成する方法を提案する。ヒューリスティックスによる文選択や、WWW上の情報を元に自動生成した質問文やフレーズ等を加える事を試みた。文選択と自動生成した疑問文を追加することによって単語誤り率が11%まで下がった。

1　はじめに

本論文ではオープンドメインの音声質問応答システムで用いる音声認識言語モデルをWWWから作成する手法を紹介する。従来研究のほとんどでは、ターゲットアプリケーションに合致したドメイン及びスタイルを持つよく手入れされたコーパスの存在を前提とし、そこにWWWから類似データを追加することで高性能な言語モデルを作成している。初期の研究ではWWWから主にn-gramの頻度を抽出することによってパープレキシティとWER (word error rate) を改善した[1, 6]。その後、n-gramではなく、文そのものをWWWテキストから抽出することによる言語モデル適合が行われるようになった[2, 3, 4]。

オープンドメインの音声認識を実現する場合に問題となるのは網羅性である。従来手法は、既存の言語モデル用コーパスにドメイン・スタイル共に近い文をWWWから収集するため、初期に与えるモデルの網羅性がその性能を決定してしまう。そこで、本論文では、そのようなシードコーパスの存在をまったく仮定せずにWWWテキストのみを用いて音声認識のための言語モデルを作成する。

2　オープンドメイン質問応答用音声認識システム

本研究の目的は、いつでも、どこでも有用な情報に容易にアクセスする手段を構築することである。日常のふとした思いつきから思考のオプションを広げることを可能にする。そのために、音声による質問応答システムを開発した。その質問応答システムのエンジンは、特定のドメインを前提とせずに、また、教師ありデータを必要とせずに、大量のWWWテキストを用いることで、オープンドメインな質問応答を実現している[7]。

その質問応答システムの音声入力を実現するためには、幅広いドメインを網羅する膨大なコーパスが必要となる。膨大な量そして高品質という点では、新聞記事は魅力的であるが、我々が日常で使用する話し言葉からはかけ離れたスタイルである。そこで、我々は、質問応答システムのエンジンが大量のWWWテキストを使用していることもあり、音声入力との語彙をそろえるという点を考慮し、WWWテキストのみから言語モデルを作成するアプローチをとる。

質問応答システムの音声入力を考えるとその形式は自ずと制限される。現在まで我々が開発した質問応答エンジンが回答できるのは、2つの名詞の間にある関係を述べた文に含まれるその名詞のうちいずれかを「何、どこ、誰、いつ」の疑問代名詞にした疑問文である。例えば、「河津川で鮎が釣れる」という文のうち「河津川」を尋ねるならば、「どこで鮎が釣れますか」という疑問文が、「鮎」を尋ねるならば、「河津川では何が釣れますか」という疑問文が考えられる。一方で、疑問代名詞をともなわない質問形式、例えば「河津川で釣れるものを教えて」があるが、このような形式にも回答できる。また、Yes/No疑問文には、現在の質問応答システムは対応していない。したがって、我々が目指す音声認識システムは、「何、どこ、誰、いつ」を疑問代名詞として持つ疑問文か、「教えて」などの要求を文末に持つ文を認識できなければならない。これを我々は「クエリー」と呼ぶ。

3　言語モデル構築

KNP[1]で解析したWWW6億ページのTsubakiコーパス[5]を利用してWWWテキストから言語モデルを作成する。さらに、クエリーを認識しやすくするために文選択を行って言語モデルを作成する。また、KNPの解析結果からパターンと呼ばれるデータを抽出し、パターンデータから、直接的にクエリーを自動生成し、言語モデルを構築するための資源として利用する。

3.1　ベースライン言語モデル

WWWテキストには、日本語以外の言語表現や、読み上げることができない記号類が多く含まれるため、Tsubakiコーパスに対して次のフィルタリングを行っ

[1]http://www-lab25.kuee.kyoto-u.ac.jp/nl-resource/knp.html

た: （1）アルファベットのみからなる文を削除する（2）日本語として許される文字・記号類以外の文字を含む文を削除する（3）一部の表記ゆれの標準化（例：ゼロ戦→零戦）（4）各文を形態素解析した際に次を含むものを削除する：(a) 未知語、ただし、片仮名のものは対象外とする (b) 指示詞と一部の連体詞（ここ、そこ、あそこ、など）、代名詞（私、あなた、など）(c) 数詞。以上のフィルタリングによって 1.79×10^{10} 形態素（1.35×10^9 文）のコーパス（www）を得た。

上記の形態素解析済みコーパスを作成するために chasen を使用した。chasen で使用した辞書は、ipadic-2.6.3 に基づきつつ、話し言葉に対してより頑健となるよう接続表の拡充、形態素辞書の追加を行ったものである。現在、形態素辞書の語彙サイズ（活用するものをすべて展開した大きさ）は 120 万ほどである。形態素解析に加えて、数詞と助数詞の読みをより正しくするために chawan を適用し、その結果を用いて言語モデルを作成した。

3.2 文選択によるクエリー言語モデル

言語モデルの元になるテキストをよりクエリーらしいものとするために www コーパスから次の条件を満たす文を選択した。（1）疑問文として「か」、「かい」「かしら」、「かな」および疑問符「?」で終了する文。（2）要求として「下さい」あるいは動詞の連用形+「て」で終了する文。選択された文によるコーパス（wwwq）は 1.28×10^9 形態素（1.04×10^8 文）から成っている。

3.3 クエリー認識のためのテキストの自動生成

大規模な WWW テキストから言語モデルを作成することで、言語モデルの語彙を大きくすることができるが、問題が 2 つある。一つは、言語表現として認められないような単語列（ノイズ）の存在である。もう一つは、WWW テキストには話し言葉に近いスタイルの文も含まれれば、ほとんど英単語をつなげたような文なども含まれ、クエリーとしてふさわしくない文も多く含まれることである。これらの問題に対処するために、我々は、WWW テキストの依存構造解析結果を利用して、ノイズを含まず、クエリーのスタイルに近いテキストを自動生成する。

ノイズに対しては、頻度の高い文字列を含むと考えられる「係り受けテキスト」の利用を提案する。また、WWW テキストが不足しているスタイルであるクエリー (「パターン疑問文」) を人工的に生成する。それぞれについて以下に述べる。

3.3.1 係り受けテキスト（DP）

Tsubaki コーパスから〈名詞，助詞，名詞の係り先〉を抽出し、係り先が活用している場合は、基本形に修正した。この 3 つ組をつなげた文字列の頻度上位 N を係り受けテキストとする。本論文の実験では N を 5×10^8 に設定した。その結果「トップページに戻る」、「健康と医学」、「同意を得ること」などの係り受けテキストが抽出された。

係り受けテキストには、WWW テキスト上の高頻度な形態素列が含まれるため、これを WWW テキストに加えることで、ノイズ単語列の確率を抑制できると考える。しかしながら、係り受けテキストの要素は文ではないため、文頭や、文末周辺の単語列の確率をゆがめてしまう。

3.3.2 パターン疑問文 (PQ)

我々は、Tsubaki コーパスの依存構造解析結果からパターンと呼ばれる形式を抽出し、様々な研究開発に利用している [8]。本質問応答システムも、パターンに基づいて回答を検索している。パターンは次の形式を持つ。「A (or B) [infix] B (or A) [postfix]」、ここで、変数 A と B は任意の名詞である。[infix] は任意の文字列を含むが、通常は助詞または助詞と 2 名詞間の関係を示す表現をとる。[postfix] は [infix] が助詞のみの場合は B (or A) の係先としての 2 名詞間の関係を示す表現をとる。また、[postfix] は [infix] が関係を示す表現を含むときは省略される。

例えば、「カビはアトピーの原因となる」という文からは「A は B の原因」というパターンを抽出できる。パターンの変数に抽出元の文にある名詞を代入したものをパターンインスタンスと呼ぶ。

パターンインスタンスから次の順番で疑問文を作成する。まず、変数 A または B のいずれかに対応する名詞を疑問代名詞に置き換える。疑問代名詞の候補は「何、どこ、誰、いつ」であるため、次の方法で決定する。パターンの抽出元の文の依存構造を参照し、置換しようとする名詞の係り先、あるいは係り元の要素を抽出する。抽出した依存構造において、名詞を 4 つの疑問代名詞に置き換えた場合のそれぞれの頻度を Tsubaki コーパスから求め、最も頻度が大きい疑問代名詞に決定する。

次に、文末表現を整える。パターンインスタンスの末尾が疑問代名詞、名詞、形容詞の場合は、次の 3 種類の方法で候補を生成し、元の文と候補を合わせた文からランダムに選択した。(i) パターンインスタンスの末尾に「ですか」または「でしょうか」を追加。(ii) パターンインスタンスの末尾が疑問代名詞の場合は、それを削除。(iii) パターンインスタンスの末尾が疑問代名詞の場合は、それを「教えて」または「教えて下さい」と置換。また、パターンの末尾が動詞の場合は、元の文または末尾の動詞を連用形にして「ますか」を加えた 2 文からランダムに選択した。その結果「静電気の原因は何ですか」、「ネットで買い物をするのはいつですか」、「北海道の名物を教えて下さい」などのような疑問文が生成された。

4 評価実験

4.1 評価設定

本 QA システムに音声入力インターフェースを導入するにあたって、当機構で開発している ATRASR を用いた。二種類の評価データを準備した。作業者 5 名に QA システムの概要やターゲットクエリーの種類を説明した上で合計 793 文の様々なドメインをカバーするクエリーを記述してもらった。次に女性 25 名、男性 25 名、

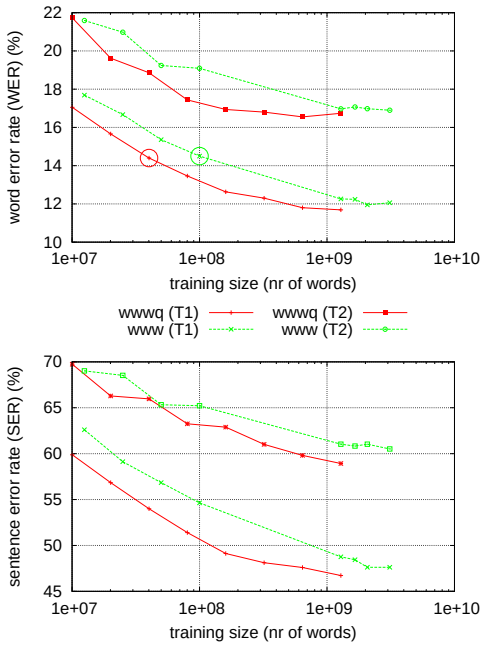

図 1: 学習データ量の増加による性能変化（logscale）

合計 50 名により、一人当たり 100 文をランダムに選択したテストクエリーを読み上げてそれを収録した。このうち、女性 13 名、男性 12 名による 2500 発話を「T1」と呼ぶ（表 1）。次に、読み上げをしてもらった 50 名に、一人当たり約 50 文のクエリーを自由に発話してもらい、それを書き起こしてテストセットを作成した。このうち、T1 で選択された 25 名による 1249 発話を「T2」と呼ぶ（表 1）。ただし、T2 には本 QA システムが回答できないクエリーも含まれている。

テストセット	読み上げ (T1)	自由発話 (T2)
形態素数	22,735	11,420
発話数	2,500	1,249
平均　形態素/発話	9.09	8.14
1-gram 数	1,550	2,159
2-gram 数	3,496	5,211
3-gram 数	4,127	6,473

表 1:　テストセットの詳細

4.2　ベースラインモデル (www) とクエリーモデル (wwwq) の比較

ベースラインモデル (www) とクエリーモデル (wwwq) における学習コーパスの量と性能の関係を図 1 に示す。

図中の y 軸の WER は単語誤り率、SER は文誤り率を表す。また x 軸は学習コーパスの量を形態素数で示している。WWW 上のクエリーから作成した言語モデル（wwwq）は WWW からランダムに抽出した文章から作成した言語モデルより WER と SER が低いことが確認できた。

また、学習量を 100 倍にすると wwwq, www いずれも両方のテストセットに対し WER が 5%程度改善されることがわかった。しかし、学習量が 10^9 形態素からは性能の改善が鈍化する傾向を示すため、そこから性能を上げるのは困難である。

2 つのモデル www と wwwq がほぼ同一性能となる場合の学習コーパスの量は、wwwq は www の半分以下であることがわかった（表 2）。当然ながら、同程度の性能を実現するための学習コーパス量が小さいので wwwq の RTF (real time factor) も小さくなっている。RTF は、音声認識における性能指標の一つで認識する音声の長さを x 秒とし、その音声認識にかかる時間を y 秒とする時 y/x で計算される。なお表中の RTF は平均値である。

言語モデル	LM-wwwq-40m	LM-www-100m
形態素数	40,139,345	100,000,000
1-gram 数	279,596	403,932
2-gram 数	5,450,517	10,259,713
3-gram 数	15,450,211	33,003,782
perplexity (3-gram)	73,4153	65,8195
WER-T1 (%)	14.40	14.50
SER-T1 (%)	54.00	55.64
WER-T2 (%)	18.86	19.09
SER-T2 (%)	65.97	65.22
RTF	1.094	1.480
OOV (%)	0.95	0.72

表 2:　性能の近い wwwq と www の比較

4.3　自動生成されたデータの追加

クエリーコーパス wwwq の全てと、それと同じ量 (1.28×10^9 形態素) の www コーパスそれぞれに 3.3 節で説明した 2 種類のコーパスを 1 億形態素ずつ追加して行き、最大で 10 億形態素を追加した多種の言語モデルを作成し、評価した。実験結果を図 2 に示す。図 2 にある 4 つのグラフの x 軸はすべて追加した学習コーパスの量を示している。y 軸はそれぞれ単語誤り率と文誤り率を示している。

読み上げテストセットに対して最も性能改善が顕著かつ、最も小さい単語誤り率を示したのはクエリーコーパスに係り受けテキストを追加したコーパスから作成した言語モデルであった（wwwq+DP）。クエリーコーパスのみから作成した言語モデルの単語誤り率は 11.69%であり、それに係り受けテキストを 2 億形態素追加することで単語誤り率は 11.16%になり、0.53%改善された (図 2 左上)。これは、読み上げテストセットにスタイルが近い wwwq に係り受けテキストを追加することで頻出 n-gram がノイズの影響を押さえたためだと考えられる。

自由発話テストセットに対して最も性能改善が顕著かつ、最も小さい単語誤り率を示したのはベースラインコーパスにパターン疑問文を追加したコーパスから作成した言語モデルであった（www+PQ）。ベースラインコーパスから作成した言語モデルの単語誤り率は 16.97%であり、それにパターン疑問文を 3 億形態素分を追加することによって単語誤り率は 0.54%改善され、16.43%に下がった (図 2 左下)。この理由は、自由発話テストセットは発話のバリエーションが多く (1)、同じく n-gram

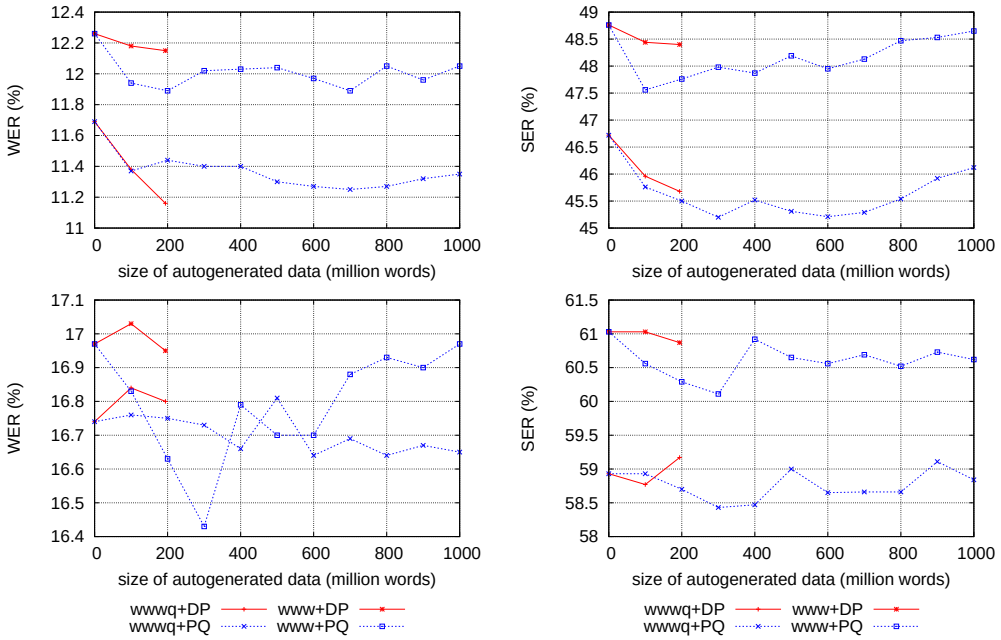

図 2: WWW から作成された言語モデルに追加した自動生成データの性能（上：読み上げテストセット T1；下：自由発話テストセット T2）

のバリエーションが豊富な www [2]に対してクエリーコーパスによりスタイルが近いパターン疑問文を追加したことで高い自由度を持つ疑問文を認識しやすくなったと考える。一方、クエリーに近いスタイルの wwwq に同じくクエリーに近いパターン疑問文を加えた場合は、性能の向上が見られなかった（wwwq+PQ)。

文誤り率の場合は、両テストセットともにクエリーコーパスにパターン疑問文を追加したコーパスから作成した言語モデルが最も小さい単語誤り率を示した（wwwq+PQ）。読み上げ発話 (図 2 右上) に対しては、パターン疑問文を全く追加しない条件では文誤り率は 46.42%であり、それにパターン疑問文を 3 億形態素分を追加することによって 1.21%改善して 45.20%になった。同じく、自由発話テストセット (図 2 右下) に対しては、58.93%の文誤り率から同じくパターン疑問文を 3 億形態素追加すると 0.5%改善し 58.43%になった。また、本音声質問応答システムのアプリケーションには、エラー回復機構として n-best 結果から擬似的なラティス構造を作成し、そこから期待する結果を効率的に選択するインタフェースを備えている。そのため、実際の使用感覚に近い評価指標として 20-best の中に正解があったかどうかを上記の文誤り率の最も低かった条件で計算すると次のようになった。読み上げテストセットでは 31.52%で、自由発話テストセットの場合は 41.07%である。従って、実用上は 6 割から 7 割の割合で完全な認識結果を容易に入力できる。

5 むすび

本論文では、オープンドメインな音声質問応答システムで用いる音声認識言語モデルについて述べた。言語モデルを作成するには WWW テキストのみを用い、ヒューリスティックスによる文選択が有効であることを証明できた。また、学習量の増大にともなう性能向上が鈍化したところで WWW テキストから自動生成したコーパスを加えることによって性能が 0.5%向上した。

[2]学習データの形態素数 1.28×10^9 の時、語彙サイズはそれぞれ www は 995,236、wwwq は 722,768 である。

参考文献

[1] A. Berger, R. Miller. 1998. Just-in-time language modeling. In *Proceedings of ICASSP-98*, pages 705–708.

[2] I. Bulyko, M. Ostendorf, M. Siu, T. Ng, A. Stolcke, Ö. Çetin. 2007. Web resources for language modeling in conversational speech recognition. In *ACM Trans. Speech Lang. Process.*, 5(1):1-25.

[3] M. Creutz, S. Virpioja, A. Kovaleva. 2009. Web augmentation of language models for continuous speech recognition of SMS messages. In *Proceedings of EACL*, pages 157–165.

[4] T. Misu, T. Kawahara. 2006. A bootstrapping approach for developing language model of new spoken dialog system by selecting web texts. In *Proceedings of INTERSPEECH '06*, pages 9–13.

[5] K. Shinzato, T. Shibata, D. Kawahara, C. Hashimoto, S. Kurohashi. 2008. TSUBAKI: An open search engine infrastructure for developing new information access. In *Proceedings of IJC-NLP*, pages 189–196.

[6] X. Zhu, R. Rosenfeld. 2001. Improving trigram language modeling with the world wide web. In *Proceedings of ICASSP*, pages 533–536.

[7] 松田繁樹, 林輝明, 大竹清敬, S. De Saeger, I. Varga, Y. Yan, 風間淳一, 磯谷亮輔, 河井恒, 鳥澤健太郎, 中村哲. 2010. QA システムのための音声入力インターフェース **情報処理学会研究報告** *SLP-84 No.21*

[8] S. De Saeger, 鳥澤健太郎, 風間淳一, 黒田航, 村田真. 2010. 単語の意味クラスを用いたパターン学習による大規模味的関係獲得 **言語処理学会第** *16* **回年次大会**, pages 932–935.

[9] 風間淳一, S. De Saeger, 鳥澤健太郎, 村田真樹. 2009. 係り受けの確率的クラスタリングを用いた大規模類似語リストの作成. **言語処理学会第** *15* **回年次大会**, pages 84–87.