E-022

字幕文字列自動対応付けのための連語 Ngram 音声認識に関する検討
Web Retrieval N-gram Language Model With Multi Word Expression for Closed-Captioning

高橋 伸弥† 森元 逞†
Shinya Takahashi Tsuyoshi Morimoto

1．はじめに

近年，聴覚障害者のための字幕付き番組放送の普及が重要課題とされ，官民を挙げた字幕放送の拡充が急ピッチで進んでいる．また，Web 上のストリーム映像に対しブラウザ上で付加情報を簡単に表示させることが出来るような仕組みも規格化されており(図１)，今後はテレビ放送だけでなく，Web 上の映像番組にも字幕サービスが普及していくものと予想される．

図 1. 字幕情報が付与された Web ページの例

映像番組へ字幕情報を付与する際に問題となるのは，第一に，「どのような内容・分量の文章を字幕として表示するか」という点である．画面上の字幕表示エリアに表示できる文章の長さに対する制限だけでなく，一度に読み取ることのできる文章量には限界があるため，一般に，字幕の文章においては不要な語句は省略されることが多い．また場合によっては，実際に喋った表現とは違う表現に置き換えられることもある．このため，字幕の文章は事前に人手で作成されていることが多いのが現状である．

第二の問題は，「どのタイミングで字幕を表示するか」という問題である．映像中のどの発話がどの字幕文字列と対応するのかを決定し，発話のタイミングに合わせて字幕を表示する必要がある．映像中の発話に対応する字幕文字列が予めわかっている場合であっても，映像中の音声信号には背景雑音や BGM 等が混在するため，音声信号から対象とする発話区間を切り出し，適切に対応付けることが必要となる．

映像内の発話音声と予め用意された字幕文字列とを自動的かつ高精度に対応付けする方法としては，音声認識結果の単語と字幕文字列中の単語を対応付ける方法[1]や音声単位，文単位で対応づけする方法[2]などが提案されている．しかし，前述のように字幕文字列は不要な語が取り除かれて簡潔な表現になっているため，冗長語や言い淀みなどを多く含む自然な発話に対しては対応付けが簡単ではない．また背景雑音を含んだり不明瞭な発音の発話であったりする場合には，発話区間の切り出しが難しいという問題や誤認識による影響が大きいという問題がある．

これらの問題に対し，我々の研究室ではこれまで，誤認識しやすい音素の組合せを予め求めておき，これを用いて認識結果の音素時系列と字幕文字列の発音時系列との対応づけを行う方法を検討してきた[7]．この方法では，ユニグラムのみを用いて単語音声認識を行い，単語の発音から音素時系列を取り出して字幕文字列との対応付けを行っている．これは，バイグラム／トライグラムを用いた場合，言語モデルの重みにより実際に発話された単語とかけ離れた単語が認識結果となる場合があるためである． この手法により，通常の単語音声認識を用いた場合に比べ，高精度に字幕対応付けが行えることが示されたが，一方で誤認識による誤った対応付けが生じるケースもあったことから，認識精度を高める必要があった．

そこで本稿では，音声認識の精度を改善するために，予め用意された字幕情報を基にして言語モデルを構築する手法について検討する．字幕情報は極めて少量の文章しか含まないため，そのまま学習用コーパスとして用いたのでは，信頼できる統計量を求めることができないことから，出現語句の Web 検索結果から近似的にトライグラム確率を推定することとし，更に出現頻度すなわち検索ヒット数の大きい Ngram については，それらを定型語句（連語）としてまとめて辞書に登録することにより，長単位での言語モデルを構築する．

2．自動字幕対応付けアルゴリズム

2.1 基本的な考え方

基本的な考え方は，正解となる字幕文字列のローマ字読みと音声認識結果の音素系列との対応付けを DP マッチングを用いて行い，音声波形の時間軸上のどの点と字幕文字列の開始点とが対応するかを求めるというものである．具体的には，(1)映像から音声波形を抽出する，(2)抽出した音声波形を適当な長さに分割する，(3)分割されたそれぞれの音声波形を音声認識する，(4)音声認識結果の音素と対応する時刻とを取得する，(5)全ての認識結果の音素系列を連結する，(6)字幕文字列をローマ字読みに変換する，(7)DP マッチングを用いて字幕音素時系列と認識結果音素時系列間で対応付けを行う，という手順で処理を行う．

ここで，上記の処理(2)における音声波形の分割は，純粋に音声認識プログラムの動作環境上の制約であり，発話区間を抽出しているわけではないことを断っておく．予め発話区間を切り出しておき，発話ごとに対応付けを行う手法もあるが，背景雑音等の影響で発話区間切り出しの精度が

†福岡大学工学部，Dept. EECS, Fukuoka Univ.

著しく低くなってしまうケースも多いため，本手法では一定長の無音区間によって音声を切り分け，字幕文字列との対応付けを行った．

2.2 DP マッチング

字幕音素時系列を$\boldsymbol{A}=\{a_i|\ i=0,\dots,N\}$，認識結果音素時系列を$\boldsymbol{B}=\{b_j|\ j=0,\dots,M\}$とする．このとき，$\boldsymbol{AB}$間の DP 距離は，以下の漸化式を計算することで求めることが出来る．

$$D(i,j)=\min\begin{bmatrix}D(i\quad,j-1)+p_{del}\\D(i-1,j-1)+d(i,j)*p_{sub}\\D(i-1,j\quad)+p_{ins}\end{bmatrix}$$

ここで$d(i,j)$は，音素a_iとb_j間の距離であり，p_{ins} p_{del}，p_{sub}はそれぞれ挿入，削除，置換ペナルティである．本稿では，各ペナルティは 1 とした．文献[7]では，$d(i,j)$に類似音素混同確率を用いているが，本稿では音素が一致すれば 0，一致しなければ 1 としている．

2.3 Web 検索結果を用いた Ngram 言語モデル

前節で述べたように，字幕文字列に現れる単語に対する Ngram 確率を求め，言語モデルを構築することを考える．このような少量の学習コーパスから言語モデルを構築する方法，いわゆる言語モデル適応手法は種々提案されているが，その多くは統計的に信頼できる程度に何らかの手段でコーパスを拡大するアプローチを採っている．またコーパスに現れない表現や単語をカバーするために既存の言語モデルとマージする手法が一般的である[8],[9].

さらに近年は，学習用の言語資源として World Wide Web 上のドキュメントを利用する手法も数多く提案されている[10],[11],[12]．基本的な戦略としては，認識結果の語句から Web ドキュメントを検索し，検索結果の文書群を用いて言語モデルを構築するというものであるが，大量の学習用 Web ドキュメントの収集はネットワークへの負荷が大きくなること，また不適切なドキュメントも多く含んでしまう可能性があることなどが問題となっている．

この問題に対し，単語の出現頻度を検索ヒット数で近似する方法が提案されている[13]．この方法は，かなり荒っぽい方法ではあるが十分に近似可能であること，そして何より膨大なコーパスを用意しなくてよいというメリットがある．そこで本研究でも同じく，単語出現頻度を検索ヒット数で近似することとした．すなわち，単語w_iの生起確率およびバイグラム・トライグラムの確率を、単語及び単語列の検索ヒット数$C_{web}(\cdot)$を用いて，以下のように算出する．ここで，w_{i-2}^{i}は単語列$w_{i-2}w_{i-1}w_i$を表す．

$$P^*(w_i)=\frac{C_{web}(w_i)}{\sum_i C_{web}(w_i)}$$
$$P^*(w_i|w_{i-1})=\frac{C_{web}(w_{i-2}^{i-1})}{C_{web}(w_{i-1})}$$
$$P^*(w_i|w_{i-2}^{i-1})=\frac{C_{web}(w_{i-2}^{i})}{C_{web}(w_{i-2}^{i-1})}$$

なお，文献[13]では実際の生起確率に近づけるために上記で求めた値に係数を乗じているが，本稿では，上式の値をそのまま用いることとした．

ここで，学習コーパスに現れなかった単語列の生起確率を求めるために，以下のようなスムージング処理を行う必要がある[14].

$$P(w_i|w_{i-2}^{i-1})=\begin{cases}\lambda(w_{i-2}^{i-1})\times P^*(w_i|w_{i-2}^{i-1}) & \cdots C(w_{i-2}^{i})>0\\(1-\beta(w_i))\times\alpha(w_i)\times P(w_i|w_{i-1}) & \\ & \cdots C(w_{i-2}^{i})=0 \text{ and } C(w_{i-2}^{i-1})>0\\P(w_i|w_{i-1}) & \cdots C(w_{i-2}^{i})=0 \text{ and } C(w_{i-2}^{i-1})=0\end{cases}$$

ここで，

$$\alpha(w_i)=\left(1-\sum_{w_i\ \text{s.t. } C(w_{i-2}^{i})>0}P(w_i|w_{i-1})\right)^{-1}$$

$$\beta(w_i)=\sum_{w_i\ \text{s.t. } C(w_{i-2}^{i})>0}\lambda(w_{i-2}^{i})P^*(w_i|w_{i-2}^{i-1})$$

である．また$C(\cdot)$は，コーパス内の単語出現数を表している．

一般に音声認識用言語モデルにおいては，上の式におけるディスカウント係数$\lambda(w_{i-2}^{i-1})$を

$$\lambda(w_{i-2}^{i-1})=\frac{C(w_{i-2}^{i-1})}{C(w_{i-2}^{i-1})+R(w_{i-2}^{i-1})}$$

とおくウィトンベル法が用いられることが多い．ウィトンベル法は後続する異なり語彙数$R(w_{i-2}^{i-1})$の大きさに応じて生起確率をディスカウントする手法である．しかし，今回，我々が対象とするような少量の学習コーパスでは統計的に信頼できる十分な量の異なり語彙数が得られないことから，本研究では，文字列長 $k(w_{i-2}^{i-1})$および定数μを用いて，

$$\frac{C(w_{i-2}^{i-1})}{R(w_{i-2}^{i-1})}=\mu k(w_{i-2}^{i-1})$$

と近似することとし，

$$\lambda(w_{i-2}^{i-1})=\frac{\mu k(w_{i-2}^{i-1})}{\mu k(w_{i-2}^{i-1})+1}$$

とした．これは，文字列が長くなるほど後続する異なり語彙数が少なくなる（すなわち C/R が大きくなる）という仮定に基づくものである．$\lambda(w_{i-2}^{i-1})$全体としては，短い文字列ほど小さくなり，長い文字列は大きくなるため，確率再分配時において長い文字列表現の語句を優遇することになる．この計算は，次節で述べる連語表現を単位とした Ngram の計算において，長単位の連語に重みをつける意味で有効であると考えられる．なお本稿では$\mu=1$として実験を行った．

2.4 連語 Ngram 言語モデルの作成

音声認識に用いられている言語モデルは，一般に形態素を単位とすることが多い．しかし助詞・助動詞のような単語長の短い付属語は誤認識を起こしやすいことが知られている．また熟語や慣用表現などは短い単位で認識するより

も長い単位で認識するほうがよい．これらの問題に対して，高頻度形態素連鎖語を辞書登録して言語モデルを改善する手法が提案されている[15],[16]．文献[15]ではコーパス内の高頻度な形態素連鎖語を扱っているのに対し，文献[16]では慣用表現などの定型表現を対象としている点で相違があるが，本稿ではまとめて連語と呼ぶこととする．

これまで，このような連語の頻度情報を取得するには膨大なコーパスが必要とされていたが，本研究ではこの頻度情報を Web 検索結果から取得し，連語 Ngram 確率を推定することを考える．

ここでは，頻度 K 以上の n 単語からなる連語を対象として，前節と同様の計算方法によりユニグラム確率，バイグラム確率，トライグラム確率を求め，スムージング処理を行ったのち，最終的な連語 Ngram 言語モデルを作成する．具体的には，学習コーパス（字幕文字列）内のそれぞれの文に含まれる語句を選定された連語で置き換えて語彙数を拡大した学習コーパスを新たに作成し，そこで現れた連語 Ngram について確率を算出する．例えば，連語$m_{i-1}^{(n-1)}$に単語w_iが後続する確率は，以下のように計算すればよい．

$$P^*\left(w_i|m_{i-1}^{(n-1)}\right)=\frac{C_{web}(m_i^{(n)})}{C_{web}(m_{i-1}^{(n-1)})}$$

このとき，検索語句によっては，$P^*\left(w_i|m_{i-1}^{(n-1)}\right)$が 1 以上となってしまうことがあるため，最大値を 1 とし，後の処理で正規化を行う．連語同士のバイグラム，トライグラム確率も上記と同様の計算方法により求めた．

表 1 に，高頻度な連語の例を示す．表に例示した連語は，検索エンジンに Google を使用して，実際に得られた結果より抽出したものである．表に示すように，文末表現が多く見られた．

表 1　高頻度な連語表現の例とその検索ヒット数

ヒット数	連語
2330000000	こと-が-あり-ます
1550000000	なっ-て-おり-ます
1040000000	よう-に-なり-まし-た
966000000	が-ござい-ます
828000000	こと_に_なり_ます
376000000	参加-し-て-い-ます
335000000	の-かも-しれ-ませ-ん
283000000	の-で-は-ない-でしょう-か
60000000	よろしく-お願い-いたし-ます
14500000	地球-温暖-化

3. 実験

3.1 実験条件

音声認識エンジンとして Julius Ver.4.1.5 を用いた．汎用言語モデル及び音響モデルには， Julius ディクテーションキット Ver. 4.1 付属のウェブテキストから学習した 6 万語のトライグラム言語モデルと性別非依存 PTM モデルを使用した．

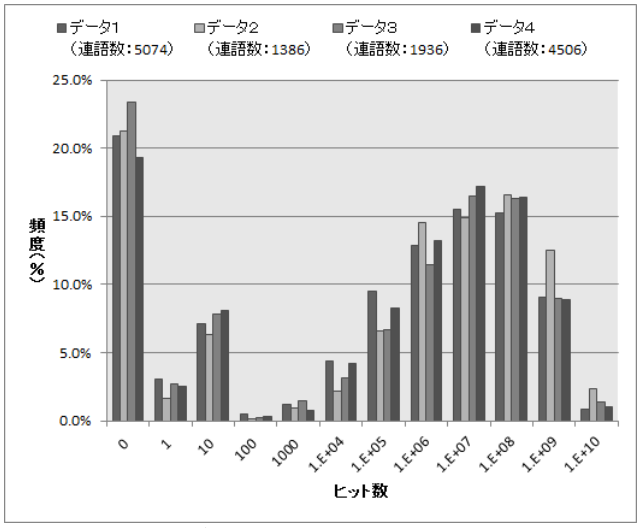


図 2. 連語ヒット数のヒストグラム

実験には，文献[7]との比較のため，福岡市広報のウェブサイトで 2007 年度に公開されていた市公報テレビ番組を使用した(詳細は表 2)．ここで，字幕単語数とは字幕文章に含まれる単語延べ数であり，音声単語数とは音声に含まれている単語の延べ数である．

表 2 のデータを用いて連語検索をした際のヒット数の分布を図 2 に示す．図からわかるようにそれぞれのデータでほぼ同じ傾向となり，部分的に重複する連語も含め，形態素 2～9 語から成る連語のうち，ヒット数 0 のものが約 2 割となり，ヒット数 1000 以下だと約 3 割となった．なお今回は，学習コーパスを 1 文ごとに分割し，句読点は検索対象から除外している．

形態素 2～9 語から成る連語を用いた言語モデルの語彙数は，連語を用いない場合と比較して 10 倍程度，拡大されるが，対象とする音声には学習コーパスとして使用した字幕テキストでは省略された語も多数含まれることから，既存の音声認識辞書の語彙を追加し，言語モデルに現れない語彙は未知語として扱うこととした．実験では，新聞記事から抽出した 2 万語の音声認識辞書を使用した．ここで未知語の生起確率は 0.01 とした．語彙数制限のためのカットオフは行っていない．

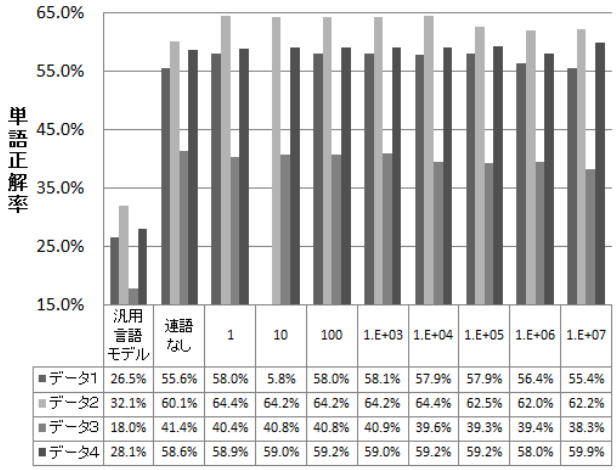
図 3 に，連語選択閾値に対する連語言語モデルの単語正解率の変化を示す．図には，汎用言語モデルおよび連語を使用せずに言語モデルを作成した場合の結果も併せて示している．この単語正解率は，書き起こしテキストを正解文とし，各連語は形態素単位に分割して計算した．また入力音声は雑音や非音声部分(音楽など)の除去等を行っていない．

実験の結果，汎用言語モデルを使用した場合と比較して，Web 検索結果より作成した言語モデルを使用することで約 3 割の正解率向上を得ることができた．またデータ 3 を除いて，連語言語モデルによる改善が見られた．

以上の音声認識実験結果を用いて，2.2 節で示した DP マ

表 2　実験に使用した映像

データ No.	番組名	放送日	放映時間	字幕数	文数	字幕単語数	音声単語数
1	コミュ! ふくおか	2007/6/8	8 分 19 秒	49	60	1284	1429
2	ギモン解決! ふくおか Q	2007/6/16	3 分	19	39	489	562
3	ギモン解決! ふくおか Q	2007/7/21	3 分	22	52	536	585
4	コミュ! ふくおか	2007/8/3	8 分 12 秒	49	62	1175	1281



	汎用言語モデル	連語なし	1	10	100	1.E+03	1.E+04	1.E+05	1.E+06	1.E+07
■データ1	26.5%	55.6%	58.0%	5.8%	58.0%	58.1%	57.9%	57.9%	56.4%	55.4%
■データ2	32.1%	60.1%	64.4%	64.2%	64.2%	64.2%	64.4%	62.5%	62.0%	62.2%
■データ3	18.0%	41.4%	40.4%	40.8%	40.8%	40.9%	39.6%	39.3%	39.4%	38.3%
■データ4	28.1%	58.6%	58.9%	59.0%	59.2%	59.0%	59.2%	59.2%	58.0%	59.9%

図 3. 連語選択閾値に対する単語正解率の変化

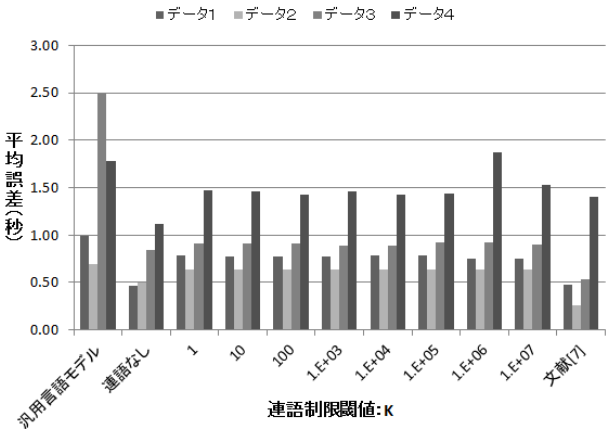


図 4. 字幕対応付け結果の平均誤差

ッチングを行い，字幕文字列との対応付けを行った．字幕文章の開始時刻と人手で与えた正解の開始時刻との平均誤差を図 4 に示す．参考のため文献[7]での結果も併せて示した．結果として，連語言語モデルを使用することで却って誤差が大きくなる傾向があることが示された．またデータ 4 では，字幕の存在しない不明瞭な音声区間や雑音区間で大きく誤差が生じてしまい，全体としての平均誤差を大きくしてしまう傾向にあったことから，今回のような音声認識をベースとした手法では事前の音声区間切り出しが不可欠であると思われる．

4. おわりに

本稿では，映像中の音声の認識結果と字幕文字列とを自動的に対応付けすることを目的として，極めて少量の学習コーパスである字幕文字列からウェブ検索結果のみを用いて音声認識用言語モデルを作成する手法を検討した．さらに高頻度形態素連鎖語（連語）をウェブ検索結果から選定し，連語言語モデルを作成する方法を示した．評価実験により，汎用言語モデルと比較して音声認識単語正解率が大幅に向上することを示した．しかし，単語認識率が向上したケースでも，字幕対応付けの精度が下がってしまう結果となった．このことから，今後は音声認識対象とする明瞭な音声区間の切り出しを検討する必要があることが明らかになった．

今回，言語モデルの評価としてテストセットパープレキシティを用いていないのは，言語モデルごとに語彙の単位が異なるためである．この点を考慮した言語モデル評価方法を検討する必要がある．

また今後は，Google Ngram を利用した連語言語モデルを検討したい．

謝辞

映像データおよび字幕データを提供下さった福岡市広報課と（株）JFIT に感謝します．

参考文献

[1] 渡邉 括行，杉山 雅英, “字幕自動生成における字幕と音声の時間軸整合の検討”, 信学技報 SP99-27, Vol. 99, No. 121 (1999).
[2] 渡邉 括行，杉山 雅英, “映像・音声検索のためのテキスト情報を利用した音声インデキシングの検討”, 音学講論, 1-P-19 (2004).
[3] 井倉 法久, 力宗 幸男, “日本語文章と音声データの同期の自動化に関する一手法”, 信学論, Vol.J89-D, No.2 (2006).
[4] C-W Huang, W Hsu and S-F Chang, “Automatic Closed Caption Alignment Based on Speech Recognition Transcripts”, Technical Report, Columbia ADVENT (2003).
[5] 西沢容子，杉山雅英, “音声特徴と言語情報を用いた音声とテキストの自動対応付け”, 音学講論，1-P-3 (2004)
[6] G.Boulianne, “Computer-assisted closed-captioning of live TV broadcasts in French”, Proc. of INTERSPEECH-2006 (2006)
[7] S. Takahashi, “Automatic Closed-Caption Alignment Using Pronunciation of Speech Recognition Transcripts for Public Relations TV Program”, Proc. of the Int. Multi-Conference on Engineer and Computer Science (2008).
[8] 小林 彰夫 他, “ニュース音声認識のための時期依存言語モデル”, 情報処理学会論文誌, Vol.40, No.4, pp. 1421-1429 (1999).
[9] 長友 健太郎 他,”相補的バックオフを用いた言語モデル融合ツールの構築”, 情報処理学会論文誌, Vol. 43, No.9, pp.2884-2893, (2002)
[10] A. Berger, R. Miller, "Just-in-time language modeling", Proc. of the IEEE International Conference on Acoustics. Speech and Signal Processing, Vol. II, pp. 705-708 (1998).
[11] I. Bulyko, M. Istendorf, *et.al*, "Gering more mileage from web text sources for conversational speech language modeling using class-dependent mixture", Proc. Human Language Technology 2003, Vol.2, pp. 7-9 (2003).
[12] S. Takahashi, T. Morimoto, and N. Tsuruta, “Document Filtering Based on Spectral Clustering for Speech Recognition Language Model”, Proc. of the Int. Multi-Conference on Engineer and Computer Science, Vol.1, pp.393-398 (2007)..
[13] X. Zhu, R. Rosenfeld, “Improving trigram language modeling with the World Wide Web”, Proc. of the ICASSP 2001, Vol. 1 (2001).
[14] 北 研二, “言語と計算 (4) 確率的言語モデル”, 東京大学出版会 (1999).
[15] 和田 陽介他,” 大語彙連続音声認識における連鎖語の追加による語彙拡大の効果”, 情報処理学会論文誌, Vol. 40, No. 4, pp. 1413-1420 (1999).
[16] 岩瀬 修, 森元 逞, 首藤 公昭, “連語を組み込んだ統計言語モデル”, 電子情報通信学会技術研究報告. NLC, 言語理解とコミュニケーション, Vol. 100, No.521, pp. 109-114 (2000)
[17] A. Lee, T. Kawahara, and K. Shikano, “Julius -- an Open Source Real-Time Large Vocabulary Recognition Engine”, Proc. of EUROSPEECH-2001 (2001).