音声言語情報処理 28－2
(1999. 10. 29)

人手による認識用言語モデル構築についての考察

秋葉友良, 伊藤克亘

電子技術総合研究所

本稿では、新規の音声認識応用タスクについての言語モデルを、大量のコーパスを使うことなく獲得する手法について検討する。言語モデルのうち、タスクの依存度が低い言語一般の知識は既存の言語知識源(EDR辞書)を極力利用する。一方、文法作製者がタスク依存の知識を表現する手段として、文字列で表した発話例を思い付くまま記したラフスケッチを利用する。半自動獲得された知識は、見通しの良い表現を用いることにより、さらに人手を加えることでより良いモデルへと仕上げることができる。

An Attempt to Acquire a Language Model Semi-Automatically without Large Corpus

AKIBA Tomoyosi, ITOU Katunobu

Electrotechnical Laboratory

This paper presents a method to acquire a language model for speech recognition used for an intended task semi-automatically without large corpus. The language model is acquired from two type of knowledge; a domain independent knowledge and a domain dependent one. In order to acquire the domain independent knowledge, the existing large machine-readable dictionary (EDR dictionary) is utilized. On the other hand, in order to acquire the domain dependent knowledge, it is utilized the example of utterances, the number of which is much smaller than large corpus. As the acquired language model represented by an easily understandable notation, it can be easily modified by an expert in order to become a better one.

1 はじめに

音声認識の精度の向上には、認識器の利用する知識(音韻モデル, 言語モデル)の質が重要となる。その一方、たとえ優れた知識の表現形式があったとしても、それを獲得するために多くのコストが必要とされるのであれば、応用に際し大きな問題をかかえていることになる。音声認識技術利用の場を広げるためには、このような知識を獲得する方法論に関する議論も必要となるだろう。

近年、tri-gramなどの統計的言語モデルが非常に良い性能をもつことが確認されている。統計的言語モデルは、コーパスさえあれば容易に構築することができるため、新聞などの大規模なテキストデータなどを利用した音声タイプライタなどの分野で大きな成功を納めてきた。しかしその一方、既存のコーパスが存在しない分野では、まずコーパスの収集が必要となる。コーパスを得るためには、その領域に関する発話のシミュレーション環境の構築、さらに人手により大量の発話の書き起しが必要になるなど、たいへんな手間が必要となってしまう。

本稿では、新規の音声認識応用分野についての言語モデルを、大量のコーパスを使うことなく獲得する手法について考察する。コーパスを利用する代わりに、汎用の言語知識(EDR辞書[1])を極力利用することにより、人手による言語モデル設計のコストを大幅に削減することを目標とする。さらに、獲得した知識は、後で人手を加えることを考慮に入れて、文法作成者にとって見通しのよい表現になるように勤める。そのために、属性付き文法という表現形式を提案する。属性付き文法は、文脈自由文法にくら

べて大きく記述量を節約することができ、また認識エンジンの言語モデルとして利用するために文脈自由文法へと変換することができる。

2 言語モデルのための知識源

言語モデルに与える知識として役に立ち、何らかの方法で入手できると考えられる知識としては、次のようなものが考えられる。

1. 語彙の選択

   単語には、そのタスクに特徴的に現れるもの、どのタスクでも汎用に使われるものとが存在する。前者は、自立語である場合がほとんどで、後者は格助詞などの付属語である場合が多い。

2. 単語の連接

   単語と単語が連続する時の規則。統計的言語モデルで学習するのはこの単語の連接であることからも分かるように、言語モデルにとって非常に役に立つ情報である。EDR辞書では、各単語に「左連接続性」「右連接続性」が与えられている。単語辞書とは別に、各「右連接続性」と「左連接続性」が連接可能かどうかを示したテーブルが用意されている。

3. 修飾・被修飾関係

   句が別の句を修飾するための条件。統語的な修飾・被修飾関係はCFGなどの文法規則として表すことができ、タスク依存度は低い。一方、意味的な修飾・被修飾関係は、タスク依存度が高い。多くの場合は、句の構成要素の中の意味主軸の間の関係として捕らえることができる。

いずれにしろ、タスク依存度の高い知識は、汎用の知識(EDR辞書)から獲得することは困難であり、なんらかの方法で作製者が知識を与えてやる必要がある。本研究では、1の付属語と2に関してはEDR辞書から抽出した知識をそのまま用いる。一方、1の自立語と3の意味的関係に関しては、領域の特徴を示した知識として文字列で示した発話例(本稿ではラフスケッチと呼ぶ)を利用し、EDR辞書の知識で補強する。

3 属性付き範疇、属性付き文法

獲得した知識は、後で人手によって容易に保守や拡張ができるよう、理解しやすくかつ操作しやすい表現であることが望ましい。文脈自由文法のCFG規則や正規文法の有限状態オートマトンによる表現などは、認識エンジンがそのまま理解できるという利点があるが、必ずしも保守しやすい形式ではない。例えば、CFG規則に意味に関する情報を付加すると、(意味素毎にカテゴリを分離しなければならないので)規則の数が莫大に増加してしまう。そこで本研究では、知識の表現形式として、以下のような属性付き範疇、属性付き文法を導入する。

属性付き範疇とは、文脈自由文法の範疇に属性を付加したものである。属性は、属性名と属性値のペア(":"で区切る)の集合である。

属性付き文法とは、文脈自由文法規則中の範疇(左辺と右辺)それぞれに対して、その属性の満たすべき条件(^)と属性への値の代入(=)の記述で拡張したものである。値の代入による変更や削除が陽に記されていない属性は、暗黙的に規則の右辺から左辺の範疇へと継承される。例えば属性付き文法規則

```
NP(left=!) : NOUN(sh^*)
```

は、右辺が属性付き範疇

```
NOUN(sh:3ce7d1,left:JLN1 right:JRN1)
```

を受け入れ可能で、その時右辺の属性付き範疇は

```
NP(sh:3ce7d1, right:JRN1)
```

となる。ここで、"!"と"*"はそれぞれ、「値なし」「何か値がある」を表すsyntax sugarである。

また、規則中に現れる全ての範疇の持つ属性は無矛盾であるという制約を設ける。すなわち規則中では同じ属性名で異なる値を持つ属性をもった2つ以上の範疇は存在しない。例えば、

```
VP : PP VP
```

という規則に対して、属性付き範疇のペア

```
PP(caseNI:3d0603) VP(caseNI:3d0603)
```

は許されるが、

```
PP(caseNI:10a479) VP(caseNI:3d0603)
```

は許されない。

このような文法記述法を利用する利点は2つある。まず、属性を利用することで必要な情報の管理が容易になる。例えば、EDRの単語辞書から得られる情報を個別の属性に割り当てることで、属性名から必要な情報が簡単に取り出せる。文脈自由文法では、単

語の持つ全ての情報を一つのシンボルに集約する必要がある。

2つ目の利点は、文法記述量の節約が可能となることである。これは、文法規則を与える際、注目する属性についてのみ記述することができるためである。

例えば、親範疇(VP)に動詞の種類を表す属性を継承しつつ動詞の語幹(Verb)と活用語尾(Aux)を組み合わる規則を記述することを考える。属性付き文法では、この規則で注目すべき属性(連接属性)の記述だけを行えばよい。(ここで属性名"rc"と"lc"は、それぞれ右連接属性、左連接続性を表す) Verbの持つ他の属性(例えば、sh:1e855e)はVPに暗黙的に継承される。すなわち、(可能な連接続性(rc,lc)のペア数)だけ規則を記述すればよい

```
VP : Verb(rc==JRV5, rc=!)
        Aux(lc==JSV5, lc=!)
VP : Verb(rc==JRVK, rc=!)
        Aux(lc==JSVK, lc=!)
```

一方、文脈自由文法で同じ規則を記述する場合は、全ての属性の組み合わせを記述する必要があり、組み合わせ的に大量の記述を行わなければならない。例えば、動詞の種類のみを継承するとした場合でも、次のように＜動詞の種類数＞×＜可能な連接続性のペア数＞の記述が必要となる。

```
VP_sh:1e855e : Verb_sh:1e855e_rc:JRV5
        Aux_lc:JSV5
VP_sh:1e855e : Verb_sh:1e855e_rc:RVK
        Aux_lc:JSVK
VP_sh:1e855f : Verb_sh:1e855f_rc:JRV5
        Aux_lc:JSV5
VP_sh:1e855f : Verb_sh:1e855f_rc:JRVK
        Aux_lc:JSVK
...
```

さらに別の属性、例えば、Verbの左連接続性やAuxの右連接続性、を継承することを考えると、さらに＜必要な属性の数＞倍の記述が必要である。

このような文法記述量の節約は、自動抽出した後の文法を保守したり拡張を行う場合に都合がよい。また属性付き文法は、情報を落とすことなく文脈自由文法へ変換することが可能であり、文脈自由文法を利用する既存の音声認識器の言語モデルとして利用できることが保証される。

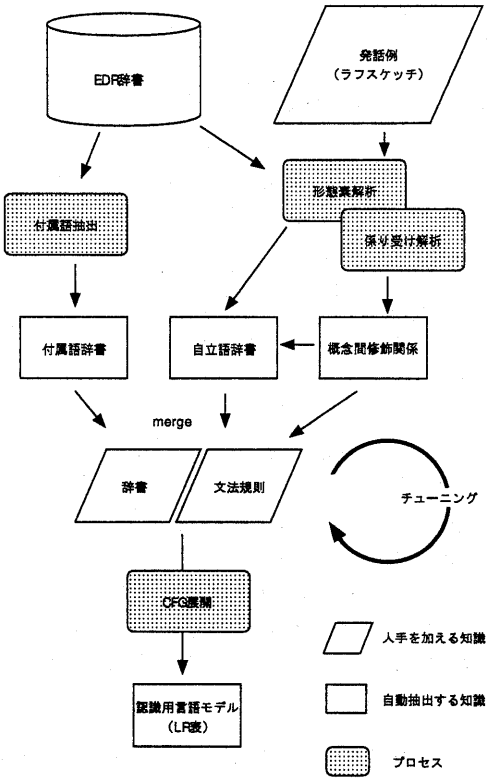

図 1: 処理の流れ

4 言語モデル獲得までの手順

図1に、本手法によって新たな文法を獲得するための手順を示す。まず、文法作製者は認識する発話をイメージし思い付く発話の具体例を(いくつでも)列挙し、ラフスケッチを作成する。ラフスケッチは、EDR辞書を利用して形態素解析と文節係り受け解析が行われ、自立語辞書と概念間の修飾・被修飾関係が抽出される。

以下では、例として図2のラフスケッチを利用することを考えて説明する。

4.1 付属語抽出

EDR単語辞書から単語の品詞が付属語に属するもの(助詞、助動詞など)で、なおかつ頻度が一定数以上のもの[1]を抽出し、辞書を構成する。辞書項目の登録には属性付き範疇を用い、左右連接属性、機能語

[1]頻度情報が利用できない助詞相当語、助動詞相当語は、全てを登録した

店に行きたいのですが

駅までは歩きます。

そこから店まで歩けますか。

図 2: 発話例(ラフスケッチ)

```
Aux(left:JSVM,right:JEM5)       m a    ま
Aux(left:JLW9,right:JED3)       m a s h i      まし
Aux(left:JLW9,right:JEE6)       m a s u ます
Aux(left:JLW9,right:JEM7)       m a s e ませ
Aux(case:MADE,left:JLJ1,right:JRJ2)     m a d e まで
Aux(case:MADE,left:JLJJ,right:JRJ2)     m a d e まで
Aux(left:JLJ1,right:JRJ5)       m a d e まで
Aux(left:JLJJ,right:JRJ5)       m a d e まで
Aux(left:JSVM,right:JED2)       m i    み
Aux(left:JSVM,right:JEE1)       m u    む
```

図 3: 付属語辞書(一部)

情報など言語モデルの制約に役に立つと考えられる情報を属性として登録した。

抽出した付属語辞書の一部を図3に示す。各行には、属性付き範疇、音素列、表記、が記されている。頻度500以上の制約を課すと、529語が抽出された。

4.2 形態素解析

文法作製者が記述したラフスケッチは、EDR単語辞書を利用して形態素解析される。辞書引で得られた単語ラティスに、左右連接の可能性、単語の頻度、文節数最小などを考慮にいれた優先度を付加し、最尤パスを求める。

4.3 係り受け解析

形態素解析結果として得られる単語列は、品詞情報や連接関係を手がかりに、まず文節に分割する。文節は、そこに含まれる自立語と付属語列を手がかりとして修飾する文節の種類(連体修飾か連用修飾か)と表層格(後置詞句の述語修飾の場合)を抽出し、それぞれどの文節を修飾するかを決定する。修飾する文節が一意に決定できない場合は、最も近い文節とする。

抽出した文節間係り受け関係からは、文節内の意味主軸間の修飾関係を抽出し、次の自立語辞書抽出に利用する。図2の発話例から得られる概念間修飾関係を図4に示す。

4.4 自立語辞書の抽出

形態素解析から得られた単語列から自立語を抜き出して自立語辞書を作成する。各単語はEDRの概念IDによって拡張し、同義語を同時に登録する。図2

店(3d0603) → NI → 行 (3d0603)

駅(10a479) → MADE → 歩 (3cefe6)

そこ (0faed1) → KARA → 歩 (3cefe6)

店(3d0603) → MADE → 歩 (3cefe6)

図 4: 概念間修飾関係

```
%       —— 店 10a479 店
NOUN(left:JLN1,right:JRN1,sh:10a479)  m i s e   店;見世
NOUN(left:JLN1,right:JRN1,sh:10a479)  m i s e y a   店屋
NOUN(left:JLN1,right:JRN1,sh:10a479)  t e N p o   店舗;店舗
NOUN(left:JLN1,right:JRN1,sh:10a479)  t e N y a   店屋
%       —— 行 1e84a2 行く
VERB(caseNI:10a479,left:JLV1,right:JRV5,sh:1e84a2)  i   行
VERB(caseNI:10a479,left:JLV1,right:JRV5,sh:1e84a2)  y u   行
VERB(caseNI:10a479,left:JLV1,right:JRVK,sh:1e84a2)  y u   往
%       —— 駅 3d0603 駅
NOUN(left:JLN1,right:JRN1,sh:3d0603)  e k i   駅
NOUN(left:JLN1,right:JRN1,sh:3d0603)  s u t e i s h o N   ステイション
NOUN(left:JLN1,right:JRN1,sh:3d0603)  s u t e- s h o N   ステーション
%       —— そこ 0faed1 そこ
NOUN&left=JLN4&right=JRN6&sh=0faed1     s o k o そこ;其所
%       —— 歩 3cefe6 歩く
VERB(caseKARA:0faed1,caseMADE:10a479+3d0603,left:JLV1,
        right:JRVK,sh:3cefe6)  a r u   歩
VERB(caseKARA:0faed1,caseMADE:10a479+3d0603,left:JLV1,
        right:JRVM,sh:3cefe6)  a y u   歩
```

図 5: 自立語辞書(第ニ候補以下は省略)

の例では、ラフスケッチに現れない単語「店舗」なども辞書登録される。さらに、同じ表記で異なる概念IDを持つ候補が存在する場合は、次候補として登録する。文法作製者は、辞書の修正を通して抽出された辞書から適切な概念IDを持つ候補を選択することができる

係り受け解析によって得られた概念間修飾関係も辞書に登録する。被修飾側の辞書項目の属性付き範疇に、修飾する概念に関する情報を記した属性を与える。図4の例からは、「歩(く)」の辞書項目に、属性caseMADE:10a479+3d0603(「まで」格には概念ID10a479または3d0603を取る)を加える。また当時に、概念のグルーピングに関する文法規則を作成する。

図2の発話例から自動抽出した自立語辞書を図5に示す。各行には、属性付き範疇、音素列、表記、が記されている。また、同時に作成される文法規則を図6に示す。

4.5 初期文法作成

図7に今回使用した文法を記す。[2]作成した文法にはタスクに依存した記述が無いので、多くのタスクでそのまま利用できると考えられる。

[2]この文法では、属性値による文字列の置換の記法($(属性名)で表す)を用いた。このような記法を無条件で許すと、文脈自由文法への変換が保証されなくなるが、ここでは記述量を節約するために利用した。

```
NOUN(sh=10a479+3d0603) : NOUN(sh^10a479)
NOUN(sh=10a479+3d0603) : NOUN(sh^3d0603)
```

図 6: 自動抽出した文法規則

```
% 付属語

Aux(final=1,left=!,right=!) : AUX(right^JEE1)
Aux(final=1,left=!,right=!) : AUX(right^JEE2)
Aux(final=1,left=!,right=!) : AUX(right^JEE3)
Aux(final=1,left=!,right=!) : AUX(right^JEE4)
Aux(final=1,left=!,right=!) : AUX(right^JEE5)
Aux(final=1,left=!,right=!) : AUX(right^JEE6)
Aux(final=1,left=!,right=!) : AUX(right^JRJ9)
Aux(final=1,left=!,right=!) : AUX(right^JEI1)
Aux(left=!,right=!) : AUX

AuxV(case=!) : Aux
AuxN : Aux
AuxN(case=!) : Aux(case^*)

% 動詞句

Verb(left=!,right=!) : VERB

VP0 : Verb AuxV
VP0 : VP0(final^!) AuxV
VP0(final=!) : VP0(final^*)

VP : VP0
VP(case$(case)=!,case=!) : PP(case$(case)^*) VP(case$(case)^*)

VPF(final=!,sh=!) : VP(final^1)
VPN(mod=$(sh),sh=!) : VP

% 名詞句

Noun(left=!,right=!) : NOUN

NP : Noun
NP(mod=!) : VPN Noun(mod^*)

% 後置詞句

PP0 : NP AuxN
PP0 : PP0(case^!) AuxN
PP0 : PP0(case^*) AuxN(case^!)

PP(case$(case)=$(sh),sh=!) : PP0(case^*,sh^*)

% 文

S : VPF
```

図 7: 初期文法

4.6 認識用言語モデルの抽出

獲得・チューニングした辞書と文法は、認識エンジンが利用できる表現へと変換する必要がある。現在の認識エンジンでは、正規文法や文脈自由文法などの形式文法を言語モデルとして利用するものがほとんどである。[3]

本研究では、LR表を言語モデルとする認識エンジンを用いた。先にのべたように、属性付き文法は文脈自由文法(すなわちLR表)へ変換することが可能である。また、記述された文法の多くの部分は連接性に関するものなので連接関係を反映したLR表[2]へと変換した。このようなLR表を用いることで、CFG規則数を大幅に削減できることが知られている[3]

図2から抽出したLR表を用いると、次のような発話を認識することができる。

店に行きませんか

そこから歩きたい

そこから店舗まで歩けるのでしょうか

[3]n-gramなどの統計モデルを用いるものも、正規文法を利用していると考えられる。

```
%       辞書の修正
VERB(caseNI:SPOT,left,left:JLV1,right:JRVK,sh:1e84a2) y u 征
VERB(caseNI:SPOT,left,left:JLV1,right:JRV5,sh:1e84a2) i 行
VERB(caseNI:SPOT,left,left:JLV1,right:JRV5,sh:1e84a2) y u 行
VERB(caseKARA=SPOT,caseMADE=SPOT,left=JLV1,
      right=JRVK,sh=3cefe6) a r u 歩
VERB(caseKARA=SPOT,caseMADE=SPOT,left=JLV1,
      right=JRVM,sh=3cefe6) a y u 歩
%       文法規則の追加
NOUN(sh=SPOT) : NOUN(sh^10a479)
NOUN(sh=SPOT) : NOUN(sh^3d0603)
NOUN(sh=SPOT) : NOUN(sh^0faed1)
```

図 8: 辞書・文法のチューニング

	修正前	修正後
辞書項目数	455	455
属性付き文法規則数	30	30
CFG規則数	435	421
LR表状態数	390	376

表 1: 獲得した文法の性質

4.7 文法のチューニング

自動抽出した辞書や文法は、先にのべた属性付き範疇、属性付き文法を用いて記述されているために、文法作成者によって容易に保守や拡張を行うことができる。

自動抽出した文法は、与えられた発話例に現れる修飾・被修飾パターンを持つ文は受け入れることができるが、発話例に現れる以外のパターンを持つものは受け入れない。例えば、図2から抽出した文法では、以下のような発話が認識できない。

そこに行きたい

店から駅まで歩けますか

しかし、図5の辞書、図7の文法を修正することによって、これらの発話を認識できる文法へと拡張することができる。図8は、「行く」の表層格「に」や「歩く」の表層格「から」「まで」には同じ種類(場所)の名詞句を取ることができることに注目して、辞書の修正とと文法の拡張を行ったものである。

図2の発話例から抽出した文法の持つ性質を1に示す。

5 関連研究

文法獲得に関する研究は、主に自然言語処理の分野で行われてきている[4][5][6]。ほとんどの研究では、事例(プレインテキストコーパスや構文木付きコーパス)だけから文法を獲得することを目標としている。一方本研究は、既存の言語知識を積極的に利用しつつ、少ない事例から文法を獲得することを目標とする。

また、自然言語処理で利用する文法は構造抽出のために用いるもので、不適切な入力を排除する目的で用いられる音声認識用言語モデルとは獲得する知識の性質が大きく異なる。本研究では、音声認識用言語モデルに利用することを前提に、効果があると考えられる制約について知識獲得を行った。

本稿では、獲得した知識の保守や拡張し易くするために、属性付き文法という表現形式を用いた。自然言語処理の分野で、文法の記述力を高める形式には、DCG(Definite Clause Grammar)[7]や一連の単一化文法[8]に関する研究がある。しかし、これらで記述した文法は文脈自由文法と等価ではなく、情報を落とすことなく文脈自由文法などの認識エンジンが利用できる形式に変換できるとは限らない。また、音声認識言語モデルとしての制約となり得る知識をうまく表現できるかどうかも明らかではない。比較的単純な構造をもつ本稿の属性付き文法は、2節で述べた音声認識用言語モデルとして役に立つ知識を問題なく記述でき、また文脈自由文法へ情報を落とすことなく変換することができる。

6 おわりに

汎用の知識と発話のラフスケッチから、新規タスクに適応した音声認識用言語モデルを獲得する手法について述べた。以下に、本方式の今後の改良点について述べる。

獲得した文法を用いて音声認識を行ったり、ランダムに文を生成してみると、一般にあまり見かけない付属語の列が現れることがある。一方、あって然るべき付属語が欠けていることもある。これは、EDR辞書の頻度情報を元に付属語の選別を行っているためであり、おそらく書き言葉での頻度と話し言葉での頻度の相違が原因と考えられる。より適切な付属語選択基準が必要となる。

本研究では、ラフスケッチから抽出した概念間修飾関係を用いて概念のグルーピングを行った。これに相当する知識は、EDR概念辞書からもある程度は自動的に抽出できそうである。しかし、汎用知識として作られた概念階層辞書に記載されたグループの粒度と、タスクに必要とされる粒度は必ずしも一致しないと考えられる。ここでも、汎用知識に加えて文法作製者の知見をうまく反映させる枠組みが必要となるだろう。

謝辞

LR表作成には、東工大田中穂積研究室の mslr [2] を使用させていただきました。

参考文献

[1] 日本電子化辞書研究所. EDR電子化辞書 version 1.5

[2] TANAKA Hozumi, TOKUNAGA Takenobu, AIZAWA Michio. Integration of Morphological and Syntactic Analysis Based on LR Parsing Algorithm. 自然言語処理 Vol. 2, No. 2, pp. 59-74, 1995.

[3] 田中穂積, 竹澤寿幸, 衛藤純司. MSLR法を考慮した音声認識用日本語文法-LR表工学(3)-. 情報処理学会音声言語情報処理研究会, No. 15-25, pp. 145-150, 1997.

[4] 白井清昭, 徳永健伸, 田中穂積. 括弧付きコーパスからの日本語確率文脈自由文法の自動抽出. 自然言語処理, Vol.4, No.1, pp.125-146, 1997.

[5] Masaki Kiyono and Jun'ichi Tsujii. Hypothesis selection i grammar acquisition. In Proceedings of the 14th International Conference on Computational Linguistics, Vol.2, pp.837-841.

[6] 森信介, 長尾真. 統計によるタグ付きコーパスからの統語規則の獲得. 情報処理学会自然言語処理研究会, Vol.110, pp.79-86.

[7] Pereira, F.C.N. and Warren, D.H.D. Definite Clause Grammar for Language Analysis, Artificial Intelligence, 13, 1980.

[8] Pollard, C. and Sag, I. Head-driven Phrase Structure Grammar, The University of Chicago Press, 1994.