2001 年 ベイジアンネットチュートリアル
Tutorial on Bayesian Networks (BN2001)
Tokyo, Japan, July 29, 2001.

音声、言語、対話における知識と確率の統合
— 音声対話における事例研究 —

秋葉友良*
産業技術総合研究所 情報処理研究部門

Abstract: 音声言語処理分野において、背景知識と、学習コーパスから獲得する統計情報を統合する手法を概観する。音声対話システムは、その構成要素である、音響モデル、言語モデル、対話モデル、それぞれの段階で、対象に見合った構造を持つ確率モデルが採択されている。音響モデルについては、現在の音声認識システムで広く利用されている HMM を、言語モデルについては、n-gram などの単純な構造を持つモデルから、文法構造に確率を与える構造的確率言語モデルまでを概観する。また対話モデルに、確率的手法を利用した研究を紹介する。

1 はじめに

近年の計算機の性能向上と、インターネットの発展による大量の計算機可読資源の増大により、確率モデルと学習データを様々な用途のために利用する、汎用の枠組みが求められつつある。

確率モデルに与える知識は、確率変数の間の独立性、および条件付き独立性として一般化できる。近年、確率変数間の条件付き独立性を表現するグラフィカルなモデルとして、ベイジアンネット (bayesian network) が注目を集めている。ベイジアンネットは確率変数をノードとした有向非循環グラフで、確率変数間のトポロジカルな関係がその依存関係を表現する。すなわち、人が持つ対象についての知識を、有向非循環グラフとして表現することが出来れば、学習データの持つ統計情報と、容易に統合する汎用の枠組みが得られることになる。しかし、知識をベイジアンネットで表現するための一般的な方法論はまだ存在せず、今後の研究の成果が期待される。

一方、音声認識の分野では、すでに 1980 年代頃から確率モデルの利用が開始されてきており、問題独自の確率モデルを見いだす努力が行われ、対象に見合った確率モデルが検討されてきた。その結果、現在では、音韻モデルには HMM、言語モデルに N-gram モデルを利用するのが最も一般的な方法とされており、その性能は様々な応用システムにおいて実証されてきている。

本稿では、これまで音声言語処理の分野で検討されてきた確率モデルを、より一般的なモデルであるベイジアンネットの視点から、概観することを試みる。特に、言語モデルにおいて、近年その展開が注目される構造的言語モデルを取り上げて紹介する。構造的言語モデルは、言語学の研究成果 (知識) を確率モデルに取り込む試みであると見ることができる。人が持つ記号的知識を、学習データから得られる統計知識とどのように統合できるかを示す、良い研究事例となるであろう。

また、言語モデルの学習方法についても触れる。構造的言語モデルのような複雑な構造を持つ確率モデルにおいて、学習データから条件付き確率を学習する一般的な枠組みとして、音声言語処理の分野で近年注目を集めている手法である最大エントロピー法を紹介する。

*〒 305-8568 茨城県つくば市梅園 1-1-1 つくば中央第二, e-mail: t-akiba@aist.go.jp

2 音声言語処理の問題設定

音声認識は、音声信号を入力として、それに対応するシンボル列を出力として取り出す問題として定式化される。入力音声信号と出力シンボル列を表す確率変数を、それぞれ A,W とする。音声認識は、音声信号 A を観察して、事後確率 $P(W|A)$ を最大とするようなシンボル列 W を見つける問題として、次のように定式化できる。

$$\mathrm{argmax}_W P(W|A)$$

これは、ベイズ則を用いて、次のように変形できる。

$$= \mathrm{argmax}_W \frac{P(W)P(A|W)}{P(A)}$$

$$= \mathrm{argmax}_W P(W)P(A|W)$$

ここで、$P(W)$ はシンボル列の事前確率を表し、言語モデルと呼ばれる。$P(A|W)$ は、シンボル列 W が与えら

れたとき、音声信号 A が観察される条件付き確率で、音響モデルと呼ばれる。上式によると、A が与えられたとき、言語モデルと音響モデルを別々に計算し、両者のバランスをとる W を見つける問題として定式化されたことになる。

W (や A) の標本空間として何を想定するかによって、問題の複雑さが左右される。W を単語とする場合 (単語認識) は $P(W)$ をそのまま考慮することも可能であるが、より大きな単位 (例えば文) を扱う場合、W を分割して多変数のモデルとして扱うのが現実的である。分割の単位としては、言語モデルでは単語、音響モデルでは音素を用いることが多い。この時、分割した確率変数は、音声言語処理の処理対象、すなわち「音声」、の特徴を十分に考慮に入れた関係付け (モデル化) が行われる。

音声は、時間間隔を持つイベントで、その時系列順序に意味がある。音声が伝える内容は、時系列順に漸進的である。すなわち、任意の先頭部分列は、何かしらの意味的単位に対応することができる。

このような特徴を持つ音声を処理する場合、その時系列順に処理を行うことは、自然な考えであろう。また、入力開始と同時に処理が開始できるという利点も大きい。発話速度に比べ処理が十分に高速であれば、リアルタイム処理も可能である。

したがって、これまでに検討されてきた音響モデルや言語モデルは、時系列順にモデル化したものが圧倒的に多い。本稿では、このようなモデルを left-to-right モデルと呼ぶ。音響モデルと言語モデルを共に left-to-right モデルとすることで、両者の同時並行的処理が可能となる。両者を同時に考慮した探索手法 (たとえばビームサーチ) の導入も容易である。

3 言語モデル

文に対応する確率変数 W を単語列を表す変数 $w_1 w_2 \cdots w_n$ に分解すると、言語モデル $P(W)$ は、次のように展開できる。

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

ここで、2 節で述べた音声言語処理の問題の独自性を考慮して、left-to-right なモデルへと展開した。すなわち、w_i の予測に $w_1 \cdots w_{i-1}$ の値が利用できるとする。この時、$w_1 \cdots w_{i-1}$ を (w_i の) 履歴 (history) と呼ぶ。また、現存のほとんどのモデルでは、i に関する対象性を仮定する。すなわち $P(w_i | w_1 \cdots w_{i-1})$ は、i に依らずに同じモデルを仮定する。

標本空間の巨大さから履歴をそのまま考慮することは出来ない。単語 w の標本空間は、辞書大きさ (現実的なタスクでは、万の単位) と一致することから、その組み合わせがどれほど大きな数となるか想像できるであろう。そこで、履歴と予測する単語の間に何らかの知識 (すなわち、確率変数間の独立性や条件付き独立性) を与えることで対処することになる。知識の与え方によって、様々な言語モデルが得られる。

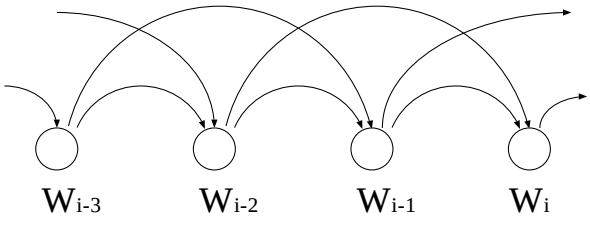

図 1: 3-gram model

3.1 N-gram モデル

直前の (N-1) 単語以外は、直接の依存関係がないと仮定する。$N = 3$ のとき、確率変数 w_i の間の関係を、ベイジアンネットで表現すると、図 1 のようになる。このネットワークは、$w_1, w_2, \cdots$ の順に、すなわち left-to-right に、証拠が得られたとき、次単語の事後確率が容易に計算できるように構成されている点に注意されたい。履歴が与えられたとき、w_i の事後確率を計算するには、$P(w_i | w_{i-2} w_{i-1})$ の分布さえ分かっていればよい。

ところで、言語の語彙数が多くなると、$N = 3$ の場合でも、条件付き確率 $P(w_i | w_{i-2} w_{i-1})$ の分布を学習するのは困難になる。そのために、様々なスムージング手法が提案されている (3.3 節) が、モデル化の工夫で対処することも可能である。クラス N-gram モデルは、履歴の直前 $(N-1)$ 単語のクラスだけが直接関係を持つとする。

$$P(w_i | w_1 \cdots w_{i-1}) \approx P(c_i | c_{i-1} c_{i-2}) P(w_i | c_i)$$

図 2 に、ベイジアンネット表現を示す。各単語を、単語に比べて標本空間の小さなクラスに対応する確率変数に抽象化することによって、モデルのパラメータを減少させることができる。

3.2 構造的言語モデル

より複雑な言語に関する知識をモデルに導入することを考えると、言語に関する知識として、言語学からの知見を導入することを考えるのは自然な流れであろう。言語の持つ構造を考慮に入れた言語モデルを構造的言語モデルと呼ぶ。

left-to-right モデルにおいて、言語の構造を値とする

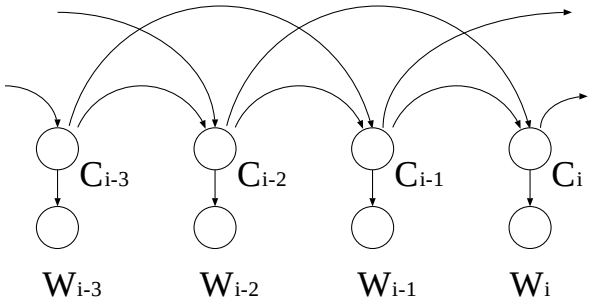

図 2: class 3-gram model

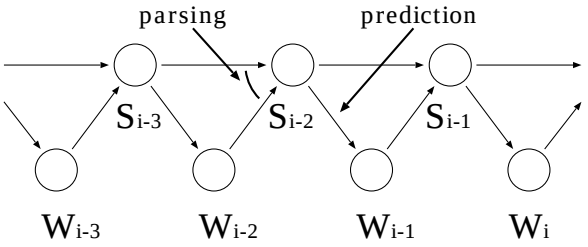

図 3: structured model

確率変数を導入し、w_i の予測にはこの構造だけが直接関係すると仮定する。すなわち、履歴 $w_1 \cdots w_{i-1}$ から構文構造 s_{i-1} が得られるとすると、

$$P(w_i|w_1 \cdots w_{i-1}) \approx P(s_{i-1}|w_1 \cdots w_{i-1})P(w_i|s_{i-1})$$
$$\approx P(s_{i-1}|s_{i-2}w_{i-1})P(w_i|si-1)$$

図 3 に、ベイジアンネット表現を示す。ここで、$P(s_{i-1}|s_{i-2}w_{i-1})$ は、$w_1 \cdots w_{i-2}$ を説明する直前の構文構造 s_{i-2} と次の入力単語 w_{i-1} が与えられたとき、$w_1 \cdots w_{i-1}$ を説明する次の構文構造 s_{i-1} を予測するモデルであり、自然言語の構文解析に関係する。一方、$P(w_i|s_{i-1})$ は、履歴を説明する構文構造から次単語を予測するモデルである。

入力単語列 $w_1 \cdots w_n$ が一意である (あるいは、比較的少数の組合わせで表現できる) 自然言語処理では、前者 $P(s_{i-1}|s_{i-2}w_{i-1})$ だけが意味を持ち、構文解析の構文的多義性解消 (尤もらしい構文木を求める) のために利用される。

一方、音声信号に対応する単語列が多義的で、その中から単語列の尤もらしさを決めることが期待される音声認識では、後者 $P(w_i|s_{i-1})$ が言語モデルとして本質的に重要であると考えられる。

3.2.1 一般化 LR 法と確率言語モデル

単語列を入力として、対応する構文構造を取り出す処理を構文解析と呼び、自然言語処理の分野で多くの研究が行われてきた。構文構造は構文解析の過程で漸進的に得られるので、構文解析過程を構文構造と同一視することができる。ここでは、一般化 LR 構文解析法の構文解析過程を元にした、確率言語モデルを紹介する。

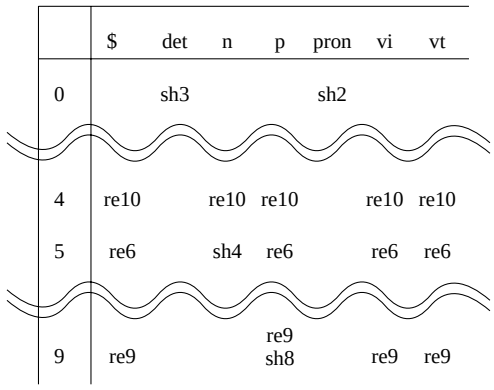

図 4: LR parsing table

一般化 LR 構文解析法 (GLR 法)[24] は、自然言語の構文解析アルゴリズムとして最も効率の良い手法の一つとして知られる。GLR 法による構文解析単独でも、音声認識用の (確率のない) 言語モデルとして利用されている [19, 17]。これは、LR 構文解析法が left-to-right に解析を進める上で、各時点で文法的に妥当な次単語を予測するのに利用できるからである。GLR 法では、文法が与えられた時点であらかじめ計算できる解析過程を先に求めた解析表 (LR 表) を利用する。LR 表は、文脈自由文法の別表現となっており、構文解析過程を表した有限状態オートマトン (より正確にはプッシュダウンオートマトン) と考えることができる。

LR 表の例を図 4 に示す[1] この表はパーザの状態 (各行) と次の単語 (各列) から、パーザの動作を求めるための表になっている。動作は shift(‘sh’),reduce(‘re’), accept(‘ac’) のいずれかである。reduce 動作では単語を消費せず (すなわち次の状態でも同じ語を参照し) 構文木を導出、shift 動作では単語を消費し次の先読み語を用いる状態へと遷移する。accept は構文解析の成功を表す。動作の記述がない項目は、解析の失敗を表す。図 4 の状態 9 単語 p の場合のように、動作が一意に決まらない場合もある。これを conflict と呼ぶ。

ここで、各状態においてある動作および次単語が選択される確率を与えることで確率モデルを定義することができる。各動作の頻度は、学習コーパスを構文解析し、選択された動作の頻度を記録することによって容易に得られる。この頻度から、動作および単語選択の確率を配分する。例えば、Briscoe と Caroll[7] は、頻度を各状態で配分したモデルを提案している。

[1] LR 表には別に GOTO 表が存在する。その扱い方は決定的なので、確率の割り当てとは無関係であり、ここでは省略する。

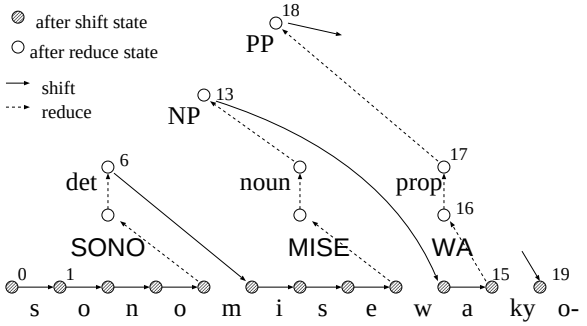

図 5: GLR parsing process

しかし、状態間の遷移関係には依存関係があり、その依存性を考慮した確率の配分を考える必要がある。LR 構文解析の解析過程を考えると、(1)shift 動作の後、(2)0 回以上の reduce 動作が続き構文木を生成、(3)shift 動作により次の単語へ、といった順序で解析が進む (図 5)。このとき、(1) で次単語を考慮した後、(2) の間は単語が変化しないことを考慮に入れる必要がある。

3.2.2 PGLR モデル

LR 構文解析過程は、パーザの状態 (σ)、先読み語 (l)、動作 (a) の列として表現できる。結果として得られる構文解析木 T の生成確率は、次のように表される。

$$P(T) = P(\sigma_0 l_1 a_1 \sigma_1 l_2 \cdots \sigma_{n-1} l_n a_n \sigma_n) \tag{2}$$

$$\approx P(\sigma_0) \prod_{i=1}^{n} P(l_i a_i \sigma_i | \sigma_{i-1}) \tag{3}$$

Inui ら [16] は、LR 表の各状態での確率を次のように配分したモデルを提案した。

$$P(l_i a_i \sigma_i | \sigma_{i-1}) \approx \begin{cases} P(l_i | s_{i-1}) P(a_i | s_{i-1} l_i) & s_{i-1} \in S_s \\ P(a_i | s_{i-1} l_i) & s_{i-1} \in S_r \end{cases} \tag{4}$$

ここで、s_{i-1} は、σ_{i-1} におけるスタックトップの LR 表状態、S_s, S_r はそれぞれ、shift 動作直後の状態集合、reduce 動作直後の LR 表状態集合を表す。この確率モデルを Probabilistic GLR(PGLR) モデルと呼ぶ。

式 (4) は、shift 直後の状態では、ある状態中の次単語と動作の組み合わせ全てに対して、足して 1 となるように確率を配分することを示している。一方、reduce 直後の状態では、単語がすでに決まっているとして、conflict となっている動作に対して、足して 1 になるように確率を配分することを示す (図 6)。(conflict がなければ、確率は常に 1 である。) このことにより、reduce 動作の間では次単語が変化しないという状態間の依存関係を考慮に入れた、正しい確率の配分を達成している。

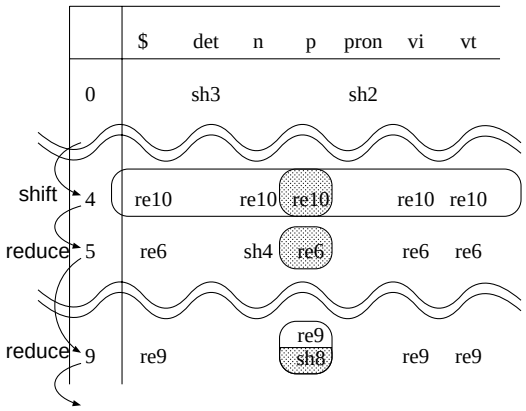

図 6: PGLR distribution

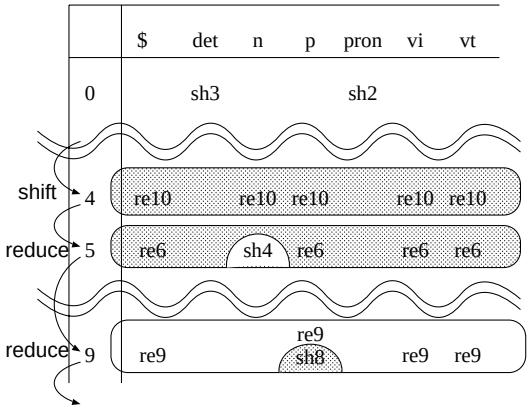

図 7: APGLR distribution

3.2.3 APGLR モデル

式 (4) を、3.2 節で述べた構造的言語モデルの視点から問い直してみよう。構造的言語モデルには (a) 入力単語を説明する構文構造を予測するモデル $P(s_{i-1}|s_{i-2}w_{i-1})$ と、(b) 構文構造から次単語を予測するモデル $P(w_i|s_{i-1})$ があり、音声認識で重要なのは (b) であった。PGLR モデルでは、(b) に相当するのが、式 (4) の shift 動作直後の状態 S_s の場合の第 1 項 $P(l_i|s_{i-1})$ であり、その他が (a) に相当する。すなわち、LR 表の shift 動作直後の状態 $s_{i-1} \in S_s$ を利用して、次単語の予測を行うことになる。

実は、この shift 動作直後の状態は、構文解析過程で単語を認識した直後に到達する状態 (例えば、図 5 における、音素 'a' を認識した直後の状態 15) であり、ほとんど構文構造を反映しない。構文構造を反映した次単語予測には、shift 直後の状態ではなく reduce 後の状態 (例えば図 5 の、後置詞句 PP を認識した直後の状態 18) を使わない手はない。

式 (2) において、動作列 $a_1 a_2 \cdots a_n$ から shift 動作だけを取り出し、順序関係を保持した部分列を $a_{x(1)} a_{x(2)} \cdots$

$a_{x(m)}$ とすると、次のように書ける。

$$P(T) \approx P(\sigma_0) \prod_{k=1}^{m} P(l_{x(k)} a_{x(k-1)+1} \sigma_{x(k-1)+1} \cdots \cdots a_{x(k)} \sigma_{x(k)} |\ \sigma_{x(k-1)}) \quad (5)$$

shift 直後から次の shift(即ち状態 $\sigma_i (x(k) < i \leq x(k+1))$ までを解析の 1 ステップと見て式 (5) を変形し、次のように近似する。

$$\approx P(\sigma_0) \prod_{k=1}^{m} [\{ \prod_{i=x(k-1)+1}^{y(k)} P(a_i|s_{i-1})\} \cdot P(l_{x(k)}|s_{y(k)}) \cdot \{ \prod_{j=y(k)+1}^{x(k)} P(a_j|l_{x(k)} s_{j-1})\}] \quad (6)$$

ここで $y(k)$ は、各 k について $x(k-1) \leq y(k) < x(k) (k = 1 \cdots m)$ を満すように定めた数である。この言語モデルを Abstracted PGLR(APGLR) モデルと呼ぶ [4]。

式 (6) の $y(k)$ は、次単語予測にどの状態を利用するかを指定する。実際の $y(k)$ の決め方には、様々な戦略を用いることができる。例えば、各 k 共通に $y(k) = x(k) - 1$ とした場合は、各時点で最も抽象度の高い構文カテゴリから次単語を予測することに相当する。また左文脈を計算する構文カテゴリをあらかじめ決めておき、そのカテゴリに対応する $y(k)$ を用いることもできる。さらに、十分なサンプル数が得られている状態に対応する $y(k)$ など、解析中に動的に $y(k)$ を決めることも考えられる。特に $y(k) = x(k-1)$ とすると、shift 直後の状態を利用することになり、PGLR モデルと等価となる。すなわち、APGLR は PGLR の一般化となっている。

APGLR モデルのもう一つの特徴は、スムージングの効果が得られることである。抽象度の高い構文カテゴリほど異なる文で共有されることが多い。したがって、shift 直後より reduce を繰り返して到達する状態の方が、学習データからより多くのサンプルを獲得できる。結果として、より信頼性の高い次単語予測が可能となる。

式 (6) は、shift 直後の状態から $y(k)$ で指定された状態までは、動作毎 (単語ではない点に注意) に確率を割り当て、各状態で足して 1 になるように配分する。$y(k)$ で指定された状態で、各状態で単語と動作の組み合わせ全てに足して 1 となるように確率を配分。以降、次の shift 動作直前の状態までは、PGLR モデルの場合と同様に、単語がすでに決まっているとして、conflict となっている動作に対して、足して 1 になるように確率を配分することを示す (図 7)。

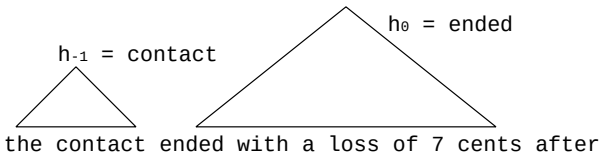

図 8: two exposed headwords

3.2.4 その他の言語モデル

音声認識に適用した left-to-right な構造的言語モデルとしては、Chelba と Jelinek の研究 [12] がある。彼らは独自の left-to-right 構文解析器を定義し、その上で確率モデルを定義している。彼らのモデルは、次の 3 つのモジュールから構成され、それぞれに確率モデルを与える。

WORD-PREDICTOR すでに得られた構文木から次単語を予測する。

$P(w_i|W_{i-1}T_{i-1})$

TAGGER 構文木と単語から品詞を予測する。

$P(t_i|w_i W_{i-1} T_{i-1})$

PARSER 構文木を生成する。

$P(p_j^i|w_i t_i W_{i-1} T_{i-1}, p_1^i \cdots p_{j-1}^i)$

ここで、W_i は履歴 $w_1 \cdots w_i$、T_i は履歴から構文解析される部分的な構文木の列、t_i は単語 w_i の品詞、p_j^i は PARSER の位置 i での j 番目の動作を、それぞれ表す。各モジュールの確率は、それぞれ近似が行われる。特に、WORD-PREDICTOR については、次のような近似が行われる。

$$P(w_i|W_{i-1}T_{i-1}) \approx P(w_i|h_0 h_{-1}) \quad (7)$$

ここで、$h_0 h_{-1}$ は、直前に解析された部文解析木 2 つの主辞 (その部分木を代表する単語) を表す。(図 8)

その他の left-to-right モデルとして、依存文法に確率を導入する試み [11][20] も行われている

構文構造に確率を割り当てるモデルは、主に自然言語処理の分野で研究が行われている。これらの多くは、書換規則に確率を割り当てることで確率モデルを定義する。最も単純なモデルは、確率文脈自由文法 (PCFG) であり、同じ左辺記号を持つ規則の確率の和が 1 となるように、確率を配分する。PCFG は、規則を越えた文脈を確率構造に持たないため、あまり良いモデルとならないことが指摘されている。そのため、様々な PCFG の拡張モデルが提案されている [9][15]。

本節で述べた left-to-right な言語モデルに対し、規則に確率を割り当てるモデルは、文法の開始記号から単語

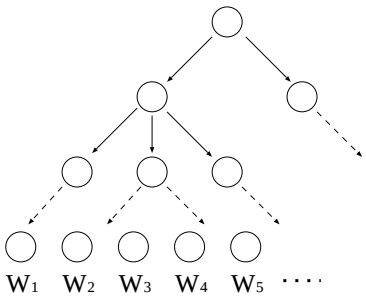

図 9: Probabilistic CFG

へと、top-down に確率変数間の依存関係を表現したモデルとなっている (図 9)。PCFG の場合、GLR 法や (構文解析法の一つである)Early 法と組み合わせて、left-to-right に先頭部分列に対応する確率を計算するアルゴリズムが示されている [25][23]。しかし、PCFG の拡張モデルでは、必ずしも left-to-right に計算が可能であるとは限らない。特に、構文木決定問題で最も有望とされている語彙化された PCFG[15] は、left-to-right なモデルにはならない [10] という問題点がある。

3.3 言語モデルの学習

言語モデルの構造を選択した後、その構造に応じた事前確率を、学習データから獲得する必要がある。モデルの構造が複雑になるほど、パラメータ数が増加する。例えば N-gram モデルの場合、言語の語彙数が多くなると、$N = 3$ の場合でも、事前確率 $P(w_i|w_{i-2}w_{i-1})$ の分布を学習するのは困難になる。そのため、何らかのスムージング手法が不可欠となる。

音声言語処理でこれまで最も多くの研究が行われてきた N-gram モデルについては、N-gram モデルの構造に特化した様々なスムージング手法が考案されてきた [13]。しかし近年、確率モデルの構造に依存しないスムージング手法として、最大エントロピー法が注目を集めている。本節では、最大エントロピーを紹介し、言語モデルの学習への応用事例について紹介する。

3.3.1 最大エントロピー法

前節で述べた言語モデルは、履歴を全て使う代わりに、履歴を代表するある特徴に着目し、その特徴だけから次単語の予測を行う確率モデルであると考えることができる。例えば、N-gram モデルでは、直前の (N-1) 語を特徴とした。構造的言語モデルでは、履歴を説明する構文構造を特徴とした。このような任意の特徴に着目して、学習データからモデルを学習する汎用の枠組みが最大エントロピー法である。

特徴の指定には、素性関数 (feature function) と呼ばれる $\{0, 1\}$ を値域とする関数が使われる。いま、履歴を h、予測する事象 (言語モデルの場合、次単語) を w とした場合、ある素性関数 f を次のように定義する。

$$f(h, w) = \begin{cases} 1 \cdots (h, w) \text{ がある特徴を満たす} \\ 0 \cdots \text{それ以外} \end{cases} \tag{8}$$

例えば、言語モデルにおいて、履歴の最後の単語との関係のみを特徴と考える (すなわち 2-gram モデル) 場合、次のような素性関数 (の集合) を導入する。

$$f_{w_a, w_b}(w_1, \cdots, w_{i-1}, w_i) = \begin{cases} 1 \cdots w_{i-1} = w_a \wedge w_i = w_b \\ 0 \cdots \text{それ以外} \end{cases} \tag{9}$$

各素性関数の期待値について、次のような制約を与える。

$$\sum_{h,w} P(h, w) f_i(h, w) = \sum_{h,w} \hat{P}(h, w) f_i(h, w) \tag{10}$$

$\hat{P}(h, w)$ は学習データから相対頻度によって求める。この制約式を満たしつつ、エントロピーが最大となるような、確率 $P(w|h)$ は、次の式によって求まる。

$$P(w|h) = \frac{\prod_i \alpha_i^{f_i(h,w)}}{\sum_w \prod_i \alpha_i^{f_i(h,w)}} \tag{11}$$

ここで、α_i は素性パラメータと呼ばれ、学習データから反復スケーリング法と呼ばれるアルゴリズムにより推定される。上式の理論的背景については、[1] を参照されたい。

最大エントロピー法の最大の特徴は、素性関数の選び方に制限がないことである。実際、事象が重なり合う素性関数を任意に導入できる。例えば、言語モデルに適用する場合、(h, w) の任意の (部分的な) 事象を表す素性関数を導入できる。このことにより、ある言語現象に着目した種々の言語モデルを統合することができる。Khudanpur と Wu[18] は、N-gram モデル、構造的言語モデル、トピックを反映した言語モデル (例えば、cache モデル [14]、word trigger を用いたモデル [22] など) の全てを、最大エントロピー法で統合している。

ベイジアンネットとの関係を図 10 に示す。履歴の同値類を表す確率変数の集合 H_{i-1} と w_i との間の条件付き確率 $P(w_i|H_{i-1})$ の学習に、確率変数の集合 $H_{i-1} \cup \{w_i\}$ のある部分集合への、ある値の代入を素性関数として与え、学習データから各素性関数毎の頻度を観察する。

最大エントロピー法で推定されるモデルの質は、素性関数の選び方に依存する。よって、いかに適切な素性関数を与えられるかが大きな鍵となる。また、素性関数集合から、関連する素性関数を自動的に選択するアルゴリズムも知られている [6]。

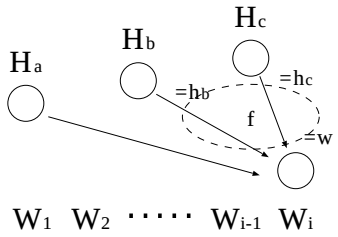

図 10: a feature of Maximum Entropy Model

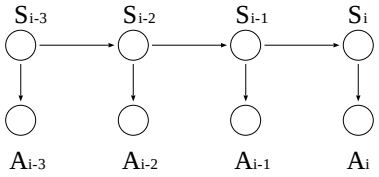

図 11: Hidden Markov Model

4 その他の確率モデル

4.1 音響モデル

抽象度が高く論理的構造を想定できる言語モデルに比べて、音響モデルにおいて各音素がどのような構造を持っているかという知識を人が見いだすことは難しい。音声波形の遷移が、時系列順に何かしらの特徴的な段階を経るといった、非常に漠然とした構造しか想定できないであろう。このように内部状態の分からない漠然とした構造を、確率モデルに反映するのに適したモデルが、隠れマルコフモデル (Hidden Markov Model) である。

HMM は、隠れ状態 S を持ったベイジアンネットとして、図 11 のように表現される。隠れ状態 S の値の遷移関係に制約を加えるのが普通である。音響モデルに利用される典型的な状態遷移関係を図 12 に示す。

音響モデルとして HMM が有効に機能することは、もはや音声認識コミュニティにとって一致した見解となっている。HMM の詳細については、文献 [2] などを参照されたい。

4.2 対話モデル

発話を単位としてとらえることで、対話モデルも left-to-right なモデルとして定式化することができる。Nagata と Morimoto[21] は、N-gram モデルによる対話のモデル化を行った。発話を発話タイプと呼ばれるシンボリックなクラスに分類して、その系列をモデル化する。

一方、対話の計算モデルとしては、発話を行為ととらえた言語行為理論と人工知能の分野で研究が行われてきたプランニングを組み合わせたモデル [5] が最も一般的である。論理的推論と確率推論を組み合わせる試みとして、一般化できるであろう [8]。

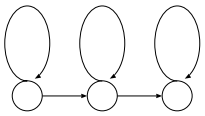

図 12: Hidden Markov Model

また、対話モデルに必要な知識の推定に確率モデルを利用することも試みられている。Akiba と Tanaka[3] は、ベイジアンネットを利用して対話相手のユーザモデルを対話の流れから漸進的に推定し、対話管理に利用している。

5 まとめ

言語モデルを中心に、音声言語処理の分野でどのように確率モデルが利用されてきたかを、より一般的なモデルであるベイジアンネットワークの視点から概観した。特に、人の持つ知識を確率モデルに取り込む試みとして注目される、構造的言語モデルの最近の動向について、詳しく紹介した。

最後に、構造的言語モデルの性能について言及しておきたい。残念ながら、現状では、被服率 (coverage) と予測力 (perplexity) の両方の観点から見て、単独で N-gram モデルを凌駕する性能を持つ構造的言語モデルは開発されていない。しかし、局所的性質を扱う N-gram とより広い範囲の性質を扱う構造的モデルは、互いに補い合う関係であるとも考えられる。実際、N-gram モデルと共に利用することによる性能向上は (もっとも、当然の結論ではあるが) 数多く報告されている。また、より優れた構造を持つモデルは、N-gram モデルを不必要にするかもしれない。今後の研究成果が期待される。

参考文献

[1] 北研二. 確率的言語モデル. 東京大学出版会, 1999.

[2] 中川聖一. 確率モデルによる音声認識. 電子情報通信学会, 1988.

[3] T. Akiba and H. Tanaka. A bayesian approach for user modeling in dialogue systems. In Proceedings of International Conference on Computational Linguistics, pp.1212-1218, 1994.

[4] T. Akiba and K. Itou. A Structured Statistical Language Model conditioned by Arbitrarily Abstracted Grammatical Categories based on GLR parsing. In proceedings of Eurospeech 2001, (to appear).

[5] J. Allen. Analyzing Intension in Utterrances. Artificial Intelligence, Vol.15, pp.143-178, 1980.

[6] A. Berger, S. Della Pietra and V. Della Pietra. A Maximum Entropy Approach to Natural Language Processing. Computational Linguistics, Vol.22, No.1, pp.39-71, 1996.

[7] T. Briscoe and J. Carroll. Generalized Probabilistic LR Parsing of Natural Language (Corpora) with Unification-Based Grammars. Computational Linguistics, Vol. 19, No. 1, 1993.

[8] E. Charniak and R. Goldman. A Bayesian model of plan recognition. Artificial Intelligence, Vol.64, No. 1, pp.53-79, 1993.

[9] E. Charniak and G. Carroll. Context-sensitive statistics for improved grammatical language models. Proceedings of AAAI-94, pp.728-733, 1994.

[10] E. Charniak. Immediate-Head Parsing for Language Models. In Proceedings of ACL-EACL 2001 (to appear).

[11] C. Chelba, et al. Structure and Performance of a Dependency Language Model. In proceedings of Eurospeech '97, pp. 2775-2778, 1997.

[12] C. Chelba and F. Jelinek. Exploiting syntactic structure for language modeling. In Proceedings for COLING-ACL 98, pp.225-231, 1998.

[13] S. F. Chen and J. Goodman. An Emprical Study of Smoothing Techniques for Language Modeling. In Proceedings of the 34th Annual Meeting of the ACL, pp.310-318, 1996.

[14] P. Clarkson and A. Robinson. Language Model Adaptation Using Mixtures and an Exponentially Decaying Cache. In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.799-802, 1997.

[15] M. J. Collins. Three generative lexicalised models for statistical parsing. In Proceedings of the 35th Annual Meeting of the ACL, pp.16-23, 1997.

[16] K. Inui, V. Sornlertlamvanich, H. Tanaka, and T. Tokunaga. A new formalization of probabilistic GLR parsing. In Proceedings of the 5th International Workshop on Parsing Technologies, pp. 123-134, 1997.

[17] K. Itou, S. Hayamizu, H. Tanaka. Continuous speech recognition by context-dependent phonetic HMM and an efficient algorithm for finding N-best sentence hypotheses. 1992 International Conference on Acoustics, Speech and Signal Processing, pp. I-21-24, 1992.

[18] S. Khudanpur and J. Wu. Maximum Entropy Techniques for Exploiting Syntactic, Semantic and Collocational Dependencies in Language Modeling. Computer Speech and Language, Vol.14, No.4, pp.355-372, 2000.

[19] K. Kita, T. Kawabata, and H. Saito. HMM continuous speech recognition using predictive LR parsing. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, pp. 703-706, 1989.

[20] S. Mori, et al. A Stochastic Parser Based on a Structual Word Prediction Model. In Proceedings of International Conference on Computational Linguistics, pp.558-564, 2000.

[21] M. Nagata and Y. Morimoto. First steps towards statistical modeling of dialogue to predict the speech act type of the next utterance, Speech Communication, V.15, 193-203, 1994.

[22] R. Rosenfeld. A maximum entropy approach to adaptive statistical language modeling. Computer, Speech and Language, Vol.10, pp.187-228, 1996.

[23] A. Stolcke. An Efficient Probabilistic Context-free Parsing Algorithm that Computes Prefix Probabilities. Computational Linguistics 21, pp.165-202, 1995.

[24] M. Tomita. An Efficient Parsing for Natural Language: A Fast Algorithm for Practical Systems. Kluwer Academic Publishers, Boston, Mass.

[25] J. Wright and E. Wrigley. Probabilistic LR parsing for speech recognition. In Proceedings of 1st International Workshop on Parsing Technologies, pp.193-202, 1989.