Vol.2014-NL-215 No.4
2014/2/6

認知文法に基づく言語モデル

呉 暁一[a)]　松本 裕治[b)]

概要：本論文では、認知文法の基本理論に基づき、非連続表現にも適用できる言語モデルを提案する。更に、提案モデルのトレーニングに関して、認知文法の「漸次的抽象」理念を借り、あらゆる言語にも適用できる教師無し学習法を考案する。提案モデルと手法は認知文法の理論に相容れるため、認知文法の部分実装ともみることができる。日本語のデータセットでの実験により、提案法の有効性を確認した。

キーワード：言語モデル, 認知文法, 非連続表現

A Language Model based on Cognitive Grammar

Abstract: This paper proposes a language model based on the idea of Cognitive Grammar in dealing with discontinuous expressions. By following the 'gradual abstraction' idea of Cognitive Grammar, we also propose an approach to train the model in an entirely unsupervised fashion. For the reason that our model is in accord with the theory of Cognitive Grammar, this research can also be considered as a partial implementation of Cognitive Grammar. Experiments verified the effectiveness of our model.

Keywords: Language Model,Cognitive Grammar,Discontinuous Expression

1. はじめに

現在、自然言語処理技術に用いられる言語モデルは殆どN-gram など連続文字列を言語単位とし、要素合成法を基本とするモデルである。この類のモデルに基づき、多くの実用システムが開発されてきたが、すべての文字列を一概に連続表現として扱う点に問題がある。

例えば、入力文の一部に未知語が存在した場合、あるいは一部を誤解析した場合、周りの解析ないし全文の解析結果にも影響を与えかねない。もし入力文「重電機器からPCB が検出された」から「…から…が…された」の様な文の主機能を表す非連続表現を真っ先に識別し、処理すれば、「…」に未知語が入っても、解析結果に与える悪影響を大幅に軽減できる。

近年、言語学の分野ではラネカーの認知文法 (Langacker 1986,1999,2008) が注目されつつある。この理論は言語表現のゲシュタルト性（全体性）を着目点とし、前述した非連続表現にも対応できる構文構造を主張している。

本稿では、認知文法の立場を踏まえ、非連続表現も扱える言語モデルを提案する。更に、認知文法の「漸次的抽象」という考え方に基づき、非連続表現を自動的獲得する教師無し学習手法を提案する。

以下ではまず、2 章で認知文法の立場及び基本的な見解について紹介する。3 章で認知文法に基づく言語モデルを示す。4 章では認知文法の見解のもとに考案した「漸次抽象法」を説明する。5 章で日本語を対象として実験を行い、この手法の有効性を示す。6 章で本研究をまとめる。

a) xiaoyi-w@is.naist.jp
b) matsu@is.naist.jp

2. 認知文法

狭義的認知文法は、主に認知言語学理論の中で、構文論に関わる部分を指す。代表的な研究としてはラネカーの認知構文論が挙げられる。

2.1 文法単位

ラネカーの認知構文論では、文法に必要とされる要素は三つしかない。

- (i)：意味と記号の統一体。
- (ii)：(i) を抽象化した構造。
- (iii)：(i) と (ii) のカテゴリ関係。

この三つの要素をラネカーは「content requirement」と

呼ぶ (Langacker 1986,2008)。

この考えに基づけば、おおよそすべての文法単位が「一定の意味を有する抽象化表現」に帰することができる。この様な単位をラネカーは「スキーマ (schema)」と呼ぶ。

一口にスキーマと言っても、 複雑度 (complexity)・抽象度 (schematicity) [*1]・慣用度 (conventionality) と三つの属性 [*2] により差異が見られる (Langacker 1986,2008) 。

例えば、品詞と単語の複雑度はほぼ同じであるが、品詞の方は抽象度が高い；形態素、単語、文はいずれも抽象度ゼロの単位とみなせるが、文の複雑度が相対的に高いと考えられる。

2.2 文法構造

認知文法では、文の生成過程を抽象度が相対的に高いスキーマの漸次的具体化過程として捉えられる。

Langacker(2008) ではこの過程を説明するため、以下の例を挙げた。

$$V_s\ X \text{ in the } N_b$$
$$\downarrow$$
$$\text{kick } X \text{ in the shin}$$
$$\downarrow$$
$$\text{kick my pet giraffe in the shin}$$

この例では、「V_s X in the N_b」は最も抽象的なスキーマで、中に３つのカテゴリ変数 V_s、X、N_b があり、それぞれ「打撃動作」「打撃されるもの」と「身体部位」を表す。「打撃動作 → 蹴る」「身体部位 → 脛」のようなカテゴリ関係さえ分かれば、このスキーマをカテゴリを表す変数がなくなるまで少しずつ具体化させ、最終的に文生成が完成する。

3. 認知文法に基づく言語モデル

3.1 関連研究

Daelemans(1998) は、認知文法に基づく用例ベースのモデルを提案している。しかし、この手法は実質的に最近傍法とみなせるので、認知文法の基本理念を矮小化している。認知文法による実装とは言いがたい。

池原 (2009,2010) では、構造的に認知文法に酷似する言語モデルを考案しているが、定義はそれほど厳密的とは言えない。それに、このモデルに用いられるルールも手作業で作らなければならない。

Chiang(2007) は階層フレーズベースの機械翻訳モデルを提案している。サブフレーズを再帰的に非終端記号に置き換えることにより、非連続的な翻訳パターンを自動生成することが可能となる。このモデルは同期式 CFG（Synchronous CFG）に基づくので、表現力は文脈自由文法に相当する。本研究では、Chiang(2007) と同じく非連続的な表現を一つのまとまりとして扱うが、CFG より高い表現力及び認知文法との融和性を目指す。

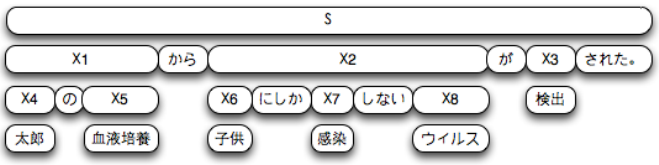

図 1 提案モデル

3.2 提案モデル

まず、 2.1「content requirement」の記号部分を実装するため、

- アルファベット集合 Σ: a,b,c...
- カテゴリ集合 N: $A, B, C...$

を定義する。

連続表現及び非連続表現を含める言語構造を捉えるため、Σ と N からなる系列を認知文法の「スキーマ」として扱い、$(\Sigma \bigcup N)^+$ で表す。

2.2 の例では、認知文法の文法構造に基づき、文は「V_s X in the N_b → kick X in the shin」の様な全局的書き換えにより漸次的に生成される。もし $S \in N$ が抽象度が最も高いスキーマで、R がスキーマからその具体例に書き換えるルールの集合 $\{(\Sigma \bigcup N)^+ \rightarrow (\Sigma \bigcup N)^+\}$ であれば、S を起点として任意の文を生成できる。

認知文法では、非終端記号が N_b と V_s の様に細分化されているので、非終端記号自体に構文情報とある程度の意味情報が入っているとみなせる。ゆえに、「V_s X in the N_b→kick X in the shin」の様な全局的書き換えを局所的書き換え「V_s→kick」と「N_b→shin」に転換できる。つまり、R は $\{\alpha N \beta \rightarrow \alpha(\Sigma \bigcup N)^+ \beta \mid \alpha, \beta \in (\Sigma \bigcup N)^*\}$ の形で一般化できる。提案モデルは四つ組 (Σ,N,R,S) により定義される。

文脈情報を吸収させるほどカテゴリ変数 N を細分化すれば、多くの書き換えルールは $N \rightarrow (\Sigma \bigcup N)^+$ の形で記述できるので、本文では $N \rightarrow (\Sigma \bigcup N)^+$ の形で記述する。

提案モデルの表現力は文脈依存文法 (Context Sensitive Grammar) に相当する。モデルの文法構造は図 1 で示す。

更に、提案モデルを統計的言語モデルとして使うには、各書き換えルールの確率を計る確率関数 P を追加する必要がある。すると、提案モデルは (Σ,N,R,S,P) の五つ組となる。S から $r \in R$ を n 個適用し、文 s に到達する場合、適用された書き換え規則の列 $D = <r_1, r_2, \ldots r_n>$ を s の導出として定義する。すると、導出 D の確率は

$$P(D) = \prod_i^n P(r_i) \quad (1)$$

で計算できる。文 s の確率は S から s を生成するすべての導出の確率の総和である。

[*1] 逆の視点から見れば具体度 (specificity) になる。
[*2] Langacker(1986) では複雑度と抽象度しか言及しなかったが、Langacker(2008) では慣用度を追加した。

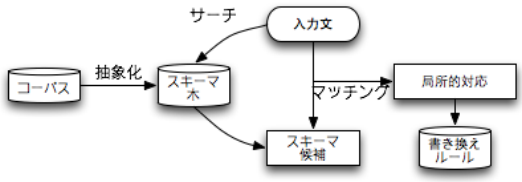

図 2 提案手法のアーキテクチャ

$$P(s) = \sum_i P(D_i) \quad (D_i \in R^{+}_{(S,s)})^{*3}. \qquad (2)$$

4. 提案モデルの教師無し学習

4.1 関連研究

Ptaszynski et al.(2011) では、「今日はなんて気持ちいい日なんだ！」から「なんて... なんだ！」のような文型パターンを抽出するため、「SPEC」という手法を考案している。まず各単位が文型パターンに入るか否かによりすべての組み合わせを列挙する。さらに列挙した各パターンのコーパスでの出現頻度を集計する。この手法のやり方はやや単純で、アルゴリズムの効率性と有効性はそれほど高くない。

文型抽出に関しては MSA(multiple sequence alignment) アルゴリズムも参考となる。MSA は元々生物情報学 (bioinformatics) に用いられ、同じ親族に属する DNA 序列の共通性を検証するアルゴリズムである。MSA を自然言語に応用する研究は Barzilay&Lee(2002,2003) が挙げられる。MSA を応用するため、まずコーパスをクラスタリングし、幾つかのクラスターに分け、各クラスターに対し MSA を応用し、ラティスを獲得する。更に各ラティスをスロット化すれば、ある種の文型パターンとなる。しかし、この手法は内容が単一的な小規模コーパスでない限り、有効なパターンを獲得するのが難しい。しかも、手法自体は言語モデルとして利用するわけでないため、言語構造の再帰性を考慮していない。本研究では、認知文法の「漸次的抽象」という理念を借り、これらの欠点を克服する。

4.2 基本的考え

前述した様に、提案モデルは五つ組 (Σ,N,R,S,P) により定義される。この中に、R は「content requirement」の三要素を全部含むので、認知文法における言語知識の中核にあたる。

教師無し手法で近似的に R を獲得するため、図 1 の「$X_1 \rightarrow X_4$ の X_5」と「$X_3 \rightarrow$ 検出」の様なカテゴリ変数とその具体例の対応関係を構築しなければならない。

この様な局所的対応関係獲得するため、まず図 1 の文とそれに対応するスキーマ「X_1 から X_2 が X_3 された。」の様に、文字列と対応スキーマの全局的対応関係を構築する必要がある。

ゆえに、文字列の対応スキーマをサーチできるように、スキーマを大量蓄積しなければならない。サーチの効率を考慮して木構造を採用する。この木は、コーパスに「漸次的抽象」を行うことにより構築される。アーキテクチャは図 2 で示す。

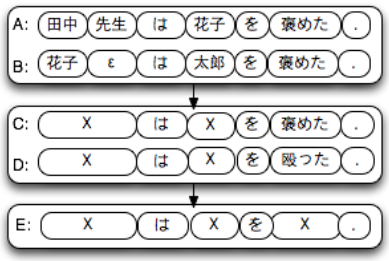

図 3 漸次的抽象化

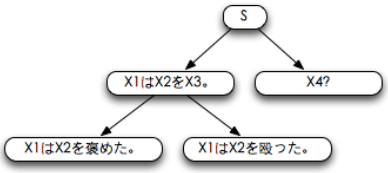

図 4 スキーマ木

4.3 スキーマ木の構築

まず、コーパスの各文に対し、同コーパス内から最小編集距離 *4 が一番近い文を探し、ペアを作る。

さらに、各ペアに対し、DP-マッチングアルゴリズムを適用する。図 3 で示している様に、文 A と B の異なる部分を「X」に置き換えることによって、スキーマ C を獲得できる。更に、C と類似スキーマ D を抽象化すれば、抽象度がより高いスキーマ E を生成できる。

最上層のスキーマがただ一つ、つまり、「X」になるまで、この抽象化過程を繰り返す。収束すれば、獲得したすべてのスキーマから重複スキーマを削除する。最後に、スキーマ内のすべてのカテゴリ変数「X」に番号を振り分ける *5 と、図 4 の様なスキーマ木を構築できる。

4.4 スキーマ候補の探索

構築されたスキーマ木をトラバースすることで、任意の文字列に対応する上位スキーマを探すことが可能となる。

スキーマ木の各ノードが入力文の上位スキーマであるかどうかも DP-マッチングにより判断する。ただし、この処理における DP-マッチングの過程では、終端記号（Σ 要素）は完全に一致しなければならない。スキーマ木の根ノードから葉ノードまでこの様なマッチングを行う。

同じ経路の中のマッチング結果に対し、深い方を出力する。例えば、「X_1 は X_2 を X_3。」と「X_1 は X_2 を褒めた。」は何れも「太郎は次郎を褒めた。」の上位スキーマである

*4 コスト設定：挿入=1, 削除=1, 置き換え=2。
*5 根ノードは抽象度が最も高いスキーマであるので、「S」を振り分ける。 番号数を抑えるため、同じ文字列に囲まれた非終端記号に同じ番号を振り分ける。

が、後者を出力する。複数の経路で獲得した結果はすべてを候補として出力する。

根ノード S はすべての文字列のスキーマであるので、候補として扱うべきではない。もし S が入力文字列の唯一の上位スキーマであれば、この文字列に上位スキーマが存在しないことにする。

4.5 解析：再帰的フィルタリング

「太郎の血液培養から子供にしか感染しないウイルスが検出された。」という入力文に対し、もしスキーマ木での探索により「X_1 から X_2 が X_3 された」の様な上位スキーマを獲得できれば、「S→X_1 から X_2 が X_3 された」をこの文の導出の中に最初適用した書き換えルールとみなすべきである。更に、このスキーマをフィルタとして入力文に適用すると、「太郎の血液培養」「検出」の様な文字列がフィルタされる。フィルタされた各文字列を再帰的にスキーマ木で探索する。「検出」の上位スキーマが存在しないので、適用したルールを「X_3→ 検出」にする。もし「太郎の血液培養」に対し「X_4 の X_5」というスキーマを出力できれば、「X_1→ 太郎の血液培養」の代わりに、「X_1→X_4 の X_5」「X_4→ 太郎」「X_5→ 血液培養」をこの導出の適用ルールとしてみなす。結果として、入力文は図 1 の様に解析できる。

4.4 で述べたように、複数のスキーマ候補が出力される可能性があるので、複数の導出過程もあり得る。動的計画法を適用することで、すべての適用ルールによって構成された構文木を構築できる。構文木の中の任意の経路は入力文の導出となる。

トレーニングデータのすべての文をこの様に解析した後、適用ルールの出現頻度を統計すれば、新しい入力文に対し、式 1 と式 2 を使って確率を計算できる。

5. 評価

5.1 実験設定

提案手法が日本語を対象とする有効性を検証するため、BCCWJ（現代日本語書き言葉均衡コーパス）のデータを使って実験を行った。

BCCWJ の新聞データ（PN − LUW）のうち、単語数が 3 以上かつ 25 以下の文から重複無しランダムで選んだ 20000 文を学習データ、同様に選んだ 2000 文を実験データとして用いた。

5.2 統計的言語モデルとしての性能

前述したように、提案モデルで学習した言語知識は書き換えルールで表すため、N グラム言語モデルに用いられるスムージング手法も利用することができる。式 3 [*6] をスムージング式とし、バイグラム言語モデルと提案手法を比較してみた。実験結果は表 1 で示している。

表 1 提案モデルの性能

モデル	単語あたりパープレキシティ
bigram	801.219
CG	215.071

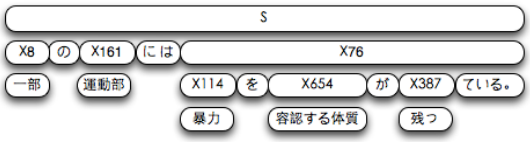

図 5 解析例

$$P(unit|context) = \frac{c(context, unit) + 1}{c(context) + V} \quad (3)$$

モデルのロバスト性を考察するため、未登録語への対応力も検証した。学習データの中にある「双六小屋などは、１０月中旬ごろまで営業している 。」という文に対し、バイグラムモデルでの計算結果は 8.561e-15 で、提案モデルでの計算結果 1.680e-15 である。更に、この文の三箇所を未登録語に置き換え、「双七小屋などは、９月中旬ごろまで閉店している。」という文を作成し、提案モデルで計算し直した結果は 4.201e-16 で大差ないが、バイグラムモデルの結果は 1.522e-28 で、確率が大幅に下がった。

その原因としては、三つの未登録語が入ったものの、「... は、... まで ... ている。」という文の骨格は全然変わっていない。提案モデルはこの点を最大限に利用できるのに対し、N グラムのような連続言語モデルはこのような変動にうまく対応できない。

5.3 提案モデルによる文解析

4.5 のように文を解析すると、導出構造は複数存在する。統計的言語モデルとして使うには差し支えないが、文の真の構造を知るには更に工夫する必要がある。ベイズ理論により、入力文が与えられた場合、各導出の確率は下式で求められる：

$$P(D_i|s) = \frac{P(D_i)}{\Sigma_j P(D_j)} \quad (4)$$

すると、式 4 の確率を最大化させる導出は式 5 となる：

$$argmax_D(D_i|s) = argmax_D P(D_i) \quad (5)$$

つまり、入力文が与えられた場合、最も可能な導出は確率を最大化させる導出である。

例えば、「一部 の 運動部 に は 暴力 を 容認する 体質 が 残っ ている 。」の解析結果は 146 通りあるが、確率が一番高い導出は 4.02e-26 の＜(S→X_8 の X_{161} に は X_{76}),(X_8→ 一部),(X_{161}→ 運動部),(X_{76}→X_{114} を X_{654} が X_{387} ている 。),(X_{114}→ 暴力),(X_{654}→ 容認する 体質),(X_{387}→ 残っ)＞である。この構造を図 5 に示す。

[*6] V は context を文脈とする unit の種類数である。

5.4 提案モデルによる文生成

S を始点とし、書き換えルールの確率に準じ、ランダムで少しずつ具体化すれば、下例のような、文全体の構造から見ると、N グラムより相対的に自然な文を生成できる。

- 食堂・喫茶 は 相続 の 中神正博氏 を 見下ろす 六本木 ヒルズ だ 。
- 砂漠 に つり落とさ れ た 出版関係 、 夢 に 推移する こと を 占め ている と 考え られ た 。
- 残し ておく だけ で 「 同時行動原則 」 の 首相 と の 入学者 を 振りかざし た こと が できる はず 。

6. まとめ

本研究では、認知文法の理論に基づき、非連続表現にも対応できる言語モデル及び該当モデルを教師無しでトレーニングする手法を提案した。

提案法は統計的言語モデルとして良い性能を持っている。その上、今まで N グラム言語モデルに用いられるスムージング手法を加えれば、性能の更なる向上も期待できる。

本論文では、カテゴリ変数が単にスキーマ内の表層要素により番号を振り分けられている。提案モデルをより一層認知文法に近づけるため、各カテゴリ変数に対し、より多くの構文情報と意味情報を与える必要がある。一部の情報は既存のリソース、例えば、シソーラスから獲得できる、一部は最近の意味学習に関する研究から獲得できる。これらの研究を如何にして提案法に取り組むのかは今後の課題とする。

参考文献

[1] D. Chiang.: *Hierarchical Phrase-based Translation*, Computational Linguistics, Vol33, No.2, 201–228 (2007).
[2] M. Ptaszynski, Rafal Rzepka, Kenji Araki, and Yoshio Momouchi.: *SPEC - Sentence Pattern Extraction and Analysis Architecture*, Proceedings of the Seventeenth Annual Meeting of the Association for Natural Language Processing, 667–670 (2011).
[3] M. Ptaszynski, Rafal Rzepka, Kenji Araki, and Yoshio Momouchi.: *Language Combinatorics:A Sentence Pattern Extraction Architecture Based on Combinatorial Explosion*, International Journal of Computational Linguistics(IJCL), Volume(2):Issue(1):24–36 (2011).
[4] R. Barzilay, and Lilian Lee.: *Bootstrapping Lexical Choice via Multiple-Sequence Alignment*, Proceedings of the Conference on Empirical Methods in Natural Language Processing(EMNLP), 164–171 (2002).
[5] R. Barzilay, and Lilian Lee.: *Learning to Paraphrase An Unsupervised Approach Using Multiple-Sequence Alignment*, Proceedings of HLT-NAACL, 16–23 (2003).
[6] R.W. Langacker.: *An Introduction to Cognitive Grammar*, Cognitive Science, 10:1–40 (1986).
[7] R.W. Langacker.: *Grammar and Conceptualization*, Mouton de Gruyter, DE (1999).
[8] R.W. Langacker.: *Cognitive Grammar - A Basic Introduction*, Oxford (2008).
[9] W. Daelemans.: *Toward an exemplar-based computational model for cognitive grammar*, English as a Human Language, Munchen: LINCOM, 73–82 (1998).
[10] 池原悟.: 非線形言語モデルによる自然言語処理, 岩波書店 (2009).
[11] 池原悟.: 非線形言語モデルと重文複文の意味類型パターン化, Association for Machine Translation, 47:7–14 (2010).