

TR-IT-0240

アナロジーに基づく構文解析手法とその評価
Linguistic Structure Analysis by Analogy and Its Efficiency

安藤 真一 Yves Lepage
Shinichi ANDO Yves Lepage

1997年9月

要旨

本稿では、ツリーバンクを用いて入力文と類似した文の構文木から入力文に対する構文木を類推する手法を提案する。この手法は用例に基づく解析手法の1つであるが、入力文とツリーバンク内データとの類似関係という新たな原理に基づいているため、構文解析における曖昧性解消のための新たな指標が提供できると考えられる。またこの手法は辞書などを用いず、データ間の類似性のみに基づいて解析を行うため、未知語などを含む入力に対しても頑健に働く。ここでは特に基本原理として働く類似関係の有効性を評価するために Penn Treebank を用いて評価実験を行った。その結果、単語の表層情報と品詞情報を用いることで解析可能な文の約70%が一意に正しく解析でき、また誤ったものについても比較的
正解に似た構文木を出力することができた。

エイ・ティ・アール音声翻訳通信研究所
ATR Interpreting Telecommunications Research Laboratories

©(株) エイ・ティ・アール音声翻訳通信研究所
1996 by ATR Interpreting Telecommunications Research Laboratories

もくじ

1	はじめに	1
2	4項アナロジー関係	2
2.1	4項アナロジー関係の定式化	2
2.2	編集距離	3
3	4項アナロジー関係を用いた構文解析	4
3.1	基本原理	4
3.2	類推妥当性	4
4	評価	7
4.1	実験対象	7
4.2	実験手順	7
4.3	基本原理と類推妥当性の評価実験	10
4.4	品詞情報を用いた評価実験	11
4.5	単語情報と品詞情報を用いた評価実験	13
5	考察	15
6	おわりに	16
	参考文献	16
A	単語情報を用いた評価実験の詳細結果	18
A.1	4項全体のバランスを考慮した類似性を用いた実験結果の詳細	18
A.2	各項を独立に評価した類似性を用いた実験結果の詳細	19
B	品詞情報を用いた評価実験の詳細結果	20
C	単語情報と品詞情報を用いた評価実験の詳細結果	22

1 はじめに

構文解析における重要な問題として、複数の解析候補から正しい構造を選び出すという曖昧性解消の問題や、未知語や未知の言語現象に対する頑健さの問題がある (Briscoe 1995)。これらは主にシステムの知識不足に起因するものであるが、2つの問題はトレードオフの関係にあり、両者を十分に解決できるだけの知識を人手で収集、整備することは難しい。そこで近年、コーパスやツリーバンク (構文解析木付きコーパス) から知識を抽出、利用する手法が盛んに研究されている。

例えば曖昧性を解消するために、ツリーバンクから得られた統計情報を用いる手法 (Black, Jelinek, Lafferty, Magerman, Mercer, and Roukos 1993; Resnik 1992) や、解析済みの用例と入力との間の意味的類似性を用いる手法 (Sumita and Iida 1991) が提案されている。またツリーバンクから構文解析知識を自動抽出する手法も言語現象を網羅し、頑健性を高めるためには重要であり、これには統計情報を用いる手法 (Sekine and Grishman 1995) や構造的な類似性を用いる手法 (Brill 1993) がある。

これらの手法は従来に比べると精度向上が見られるが、実用性の面では十分とは言いがたく¹、さらに多くの知識が必要であると考えられる。このため、より大きなコーパスを知識源として利用することが考えられているが、そのようなコーパスの作成にはコストやデータの信頼性が問題となる。さらに、各種の手法で得られる多様な知識を組み合わせる利用するアプローチも考えられる。しかし上記の各手法は、知識源となるコーパスや利用している知識の表現形式が異なるために、組み合わせることが難しい。様々な手法の融合という観点から見れば、例えば Bod が提案しているようにツリーバンク内のデータを直接、知識として用いる手法 (Bod 1993) などが有用であると考えられる。

本稿では入力文と類似した文の構文木から入力文に対する構文木を類推する手法について述べる (Lepage and Ando 1996; 安藤, Lepage 1997)。この手法はツリーバンク内の知識表現形式をそのまま使って構文解析を行うため、比較的容易に他の解析手法との融合を考慮することができる。また本手法における構文解析は、複数のツリーバンク内データの間で定義される特定の類似関係に従って行われる。このように統計情報や意味的類似性とは異なる原理でツリーバンクを用いるため、実データに含まれる情報の利用について新たな視点を提供できるだろう。さらにこの手法は辞書などを用いず、データ間の類似性のみに基づいて解析を行うため、未知語などを含む入力に対しても頑健に働く。

ここではまず、動作原理となる類似関係 (4 項アナロジー関係) を定義し、これを構文解析に利用する手法について述べる。次に Penn Treebank を用いた種々の実験から、この類似関係が構文解析に有効であることを示す。

¹Briscoe は、実用的であるためには 90% 以上の精度が必要としている (Briscoe 1995)

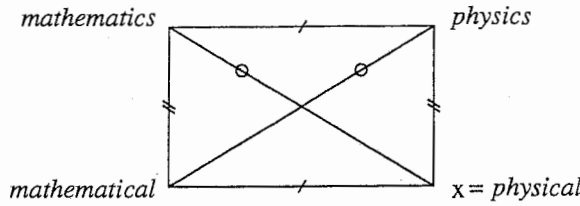


図 1: 4 項アナロジー関係のモデル

2 4 項アナロジー関係

Saussure は、ある語の語形変化パターンを他の語に適用することで新たな語が創造される現象をアナロジーと呼び、言語の創造性という観点からその重要性を指摘した (Saussure 1916)。例えば、以下の 3 つの単語のアナロジーからは「physical」が導出できる。

$$\begin{aligned} \text{mathematics} : \text{mathematical} &= \text{physics} : x & (1) \\ x &= \text{physical} \end{aligned}$$

この現象は多くの言語の様々な言語現象 (活用、派生など) で見ることができる。また語形変化だけでなく、文法的なパターンの上でも働くと考えられる。本稿ではこの種のアナロジーを取り扱い、特にこのアナロジーが成立する 4 項の間の関係を 4 項アナロジー関係と呼ぶことにする。

2.1 4 項アナロジー関係の定式化

我々は、4 項アナロジー関係にある各項が互いに交換可能であることに着目し、この関係の定式化を行った (Lepage and Ando 1996)。例えば上式 (1) のアナロジーが成り立つ場合には下記の 2 式によっても同じ x が導出できる。

$$\text{mathematics} : \text{physics} = \text{mathematical} : x \quad (2)$$

$$\text{mathematical} : \text{physics} = \text{mathematics} : x \quad (3)$$

ここで式 (1) では、意味を保存しながら語彙的な機能を変化させる左辺の語形変化パターンが右辺に写像されており、式 (2) では逆に、語彙的な機能を保持しながら意味を変化させる語形変化パターンが写像されていると考えることができる。また式 (3) では各々の変化が同時に起こっていると見ることができる。左辺右辺の変化パターンはそれぞれ等しいため、アナロジーの見られる 4 単語の関係は図 1 のような長方形で表すことができる。ここで図中の各語を結ぶ線は各語の間の距離を表し、同じ記号の線はその大きさが同じであることを表す。すなわち 4 項アナロジー関係は下記のように定式化できる。

定義 1 (4 項アナロジー関係)

$$u : v = w : x \stackrel{\Delta}{\iff} \begin{cases} \text{dist}(u, v) = \text{dist}(w, x) \\ \text{dist}(u, w) = \text{dist}(v, x) \\ \text{dist}(v, w) = \text{dist}(u, x) \end{cases}$$

ここで、 $\text{dist}(a, b)$ は a と b の間の距離を表す。

2.2 編集距離

定義1の距離として、ここでは編集距離を採用した。編集距離とは2つの文字列を同じくするために必要な最小編集操作コストである (Wagner and Fischer 1974)。特に今回は、削除、挿入、置換の3つの編集操作を考え、その操作数によって定義した。以下に編集距離の定義を示す。

定義2 (編集距離)

文字集合を V 、空集合を ε とすると、関数 $dist$ は交換法則を満たし、かつ次のように定義される。

$\forall(a, b) \in V^2, \forall(u, v) \in (V^*)^2$ について

$$\begin{aligned} dist(\varepsilon, \varepsilon) &= dist(a, a) = 0 \\ dist(\varepsilon, a) &= dist(a, \varepsilon) = 1 \quad (if a \neq \varepsilon) \\ dist(a.u, \varepsilon) &= dist(a, \varepsilon) + dist(u, \varepsilon) \\ dist(a.u, b.v) &= \min \begin{pmatrix} dist(a, b) + dist(u, v), \\ dist(a, \varepsilon) + dist(u, b.v), \\ dist(\varepsilon, b) + dist(a.u, v) \end{pmatrix} \end{aligned}$$

例えば「mathematical」と「mathematics」を考えると、「mathematical」の末尾部分で「a」を「s」に置換し、「l」を削除することで2つは同じ文字列となる。このため、この2単語の間の編集距離は2と計算できる。

同様に単語単位の編集操作を考えると、単語列、すなわち文における編集距離も定義できる。例えば「the green lamp turns off」と「the lamp turns on」という2つの文を考えると、前者の文は「green」という単語を削除し、「off」を「on」に置換することで後者の文と等しくなる。このため、この2文間の距離は2と計算できる。さらにノード単位の編集操作を考えると、木構造間でも同様の距離が定義できる (Selkow 1977)。

これらの編集距離を用いると、文や構文木における4項アナロジー関係も同様に定義1によって定式化できる。

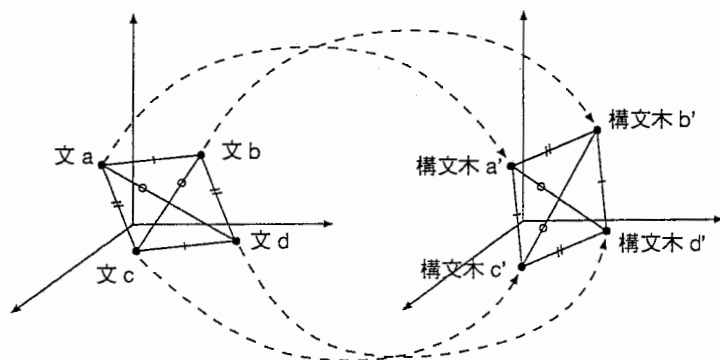


図 2: 基本原理における仮定

3 4項アナロジー関係を用いた構文解析

3.1 基本原理

文中の単語と構文木中のノードの間には密接な対応関係があり、片方の変化は他方に影響を及ぼす。そこで我々は「文においてアナロジー関係が成り立つならば、各々の文に対応する構文木の間でもアナロジー関係が成り立つ」と仮定した(図2参照)。この仮定に基づくと、ツリーバンク内のデータを直接用いて、入力文に対する構文木が計算できる(Lepage and Ando 1996)。

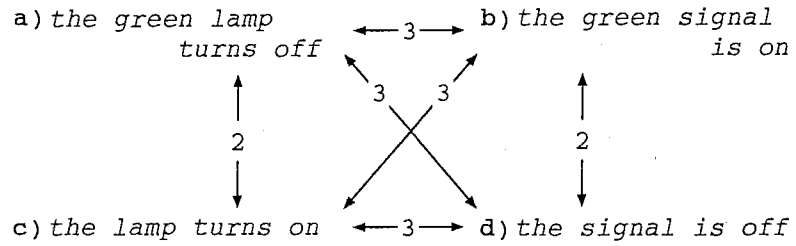
具体的には、まず入力文(図3のa)と4項アナロジー関係にある3文(図3のb、c、d)をツリーバンクから検索する。ここで得られた文はツリーバンク内でそれぞれ構文木を持っている。そこで、検索された文に対応する3つの構文木(図3のb'、c'、d')に再度4項アナロジー関係を適用する。上記の仮定に基づくと、この操作によって入力文に対する構文木(図3のa')が計算できる。

ここで4つ目の構文木(図3のa')は、例えばBodの手法のようにツリーバンクに納められた構文木の部分木を組み合わせることによって新たに生成することもできる(Bod 1993)。しかし、ここではまず4項アナロジー関係による解析の正当性を評価することを考え、出力すべき構文木もツリーバンク内から検索する機構として実験システムをインプリメントした。例えば図3の場合、本システムはツリーバンク内の文「the big cat wakes up」に付加された構文木a'を、入力文aに対応する構文木として検索、出力する。

3.2 類推妥当性

本手法ではデータ間の類似関係を手がかりとして入力文に対応する構文木を検索するが、ここで得られる構文木は一意には決まらない。これは本来のアナロジー関係が変化パターンの保存関係を意味するにもかかわらず、編集距離の保存関係のみで記述しているためであると考えられる。すなわち編集操作列の関係を編集操作数の関係で定義したために、類推が成り立ちにくい文や構文木の組合せが4項アナロジー関係に含まれてしまい、これが曖昧性を生み出す。そこで4項アナロジー関係にある文や構文木が互いに類推しやすいものであるかどうかを調べることで、検索結果のもっともらしさを評価することを考えた。

文における4項アナロジー関係



構文木における4項アナロジー関係

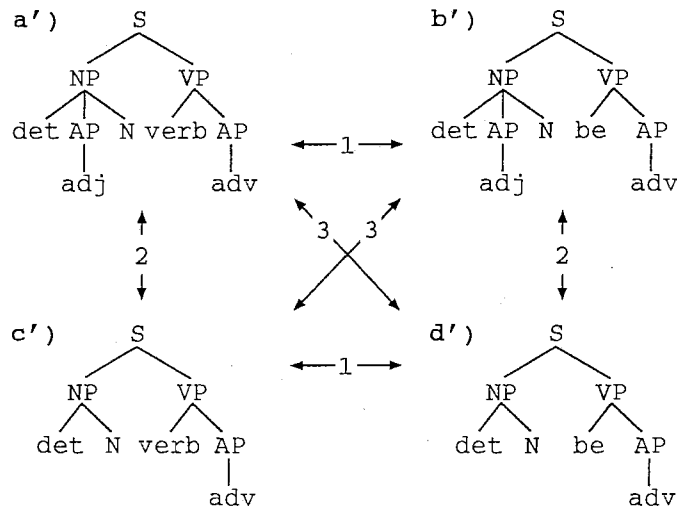


図 3: 4 項アナロジー関係を用いた動作原理

以下ではこの尺度を類推妥当性と呼ぶ。類推妥当性の要素としては、以下の2つが考えられる。

類似性

直観的にアナロジー関係にある4項が互いに似ていれば、お互いの間の類推が成立しやすいと考えられる。これは4項間の距離が小さいほど、類推が成り立ちやすいことを意味する。

これは数学的には次のように説明できる。編集距離によって定義される多次元空間の上で、1つの文(あるいは構文木)は1つの点として表現される。また定義1を満たす空間は、3つの等式が表す超球(多次元の球)の交わりとして表現される。ここで各超球の大きさを小さくすると検索結果の空間(交わり部分)が狭くなり、不正な検索結果が除くことができる。

ここでは具体的な類似性の計算式としては以下の2つを考える。

1つは類似性を4項全体で評価する計算式であり、入力文 x 、出力構文木 y に対する類似性 (Sim) は以下の式によって計算できる。

$$Sim(x, y) = \max_{\forall(u, v, w, u', v', w')} \left(\frac{1}{dist(u, x) + dist(v, x) + dist(w, x)} + \frac{1}{dist(u', y) + dist(v', y) + dist(w', y)} \right) \quad (4)$$

ここで u, v, w は構文木 y を導出するために用いたツリーバンク内の 3 つの文を表し、 u', v', w' はその各々に対する構文木を表す。また $dist$ の値が 0 となり逆数が計算できない場合には、その値として 2 を用いた²。

もう 1 つは各項間の類似性を独立に評価する計算式であり、4 項全体の類似性はその和によって定義した。すなわち入力文 x 、出力構文木 y に対する類似性 (Sim) は以下の式によって計算できる。

$$Sim(x, y) = \max_{\forall(u, v, w, u', v', w')} \left(\frac{1}{dist(u, x)} + \frac{1}{dist(v, x)} + \frac{1}{dist(w, x)} + \frac{1}{dist(u', y)} + \frac{1}{dist(v', y)} + \frac{1}{dist(w', y)} \right) \quad (5)$$

類推パス数

本手法では構文木を検索するためにツリーバンク内の 3 つのデータを用いるが、複数の 3 つ組を介して同じ構文木が出力される場合がある。このように同じ結果を出力する 3 つ組の数を以下では類推パス数 ($Freq$) と呼ぶ。類推パス数が多い場合、すなわち多くの 3 つ組から同じ構文木が類推できる場合には、その類推結果の信頼性も高いと考えられる。

先と同様に数学的な観点から見ると、類推パス数が大きくなるほど制約となる超球の数が増える。このため、全ての超球が交わる部分空間は狭くなり、不正な結果を生み出す検索結果の空間を小さくすることができる。

ここでは下に示すように、上記 2 つの要素を正規化したものの重み付きの和で類推妥当性 (\mathcal{W}) を表現し、実験システムに組み込んだ。

$$\mathcal{W}(x, y) = \alpha \frac{Sim(x, y)}{\max_{\forall(y_i)}(Sim(x, y_i))} + \beta \frac{Freq(x, y)}{\max_{\forall(y_i)}(Freq(x, y_i))} \quad (6)$$

² $dist$ の値は整数であるため、順序関係に問題はない

表 1: 実験に用いたツリーバンクの性質

	文	構文木
大きさの範囲	1 ~ 34	1 ~ 44
大きさの平均	7.51	8.35
距離の平均	8.84	8.21

4 評価

本手法の有効性を評価するために、実際のツリーバンクを用いて実験を行った。ここでは特に 4 項アナロジー関係の評価に焦点をあてた実験について報告する。

4.1 実験対象

知識源となるツリーバンクとして、比較的似た文が集められている Penn Treebank (rel.2.0) 内の ATIS (Air Travel Information System) データを利用した。ただし Penn Treebank に含まれる構文木は最下位ノードのほとんどが文中の単語やその品詞であったため、本実験では最下位ノードを除いた木を構文木として利用した。この操作により ATIS の全データ 577 文中の 376 文が他の文と同じ構文木を持つようになった。ツリーバンク内のデータの例を以下に示す³。

構文木 : S(NP-SBJ, VP(NP, NP(NP, PP-DIR(NP), PP-DIR(NP))))

文 : Show me the fare from Dallas to San Francisco

上記では省略したが、実際に用いたツリーバンク内では構文木内のノードと文中の単語の間の対応関係も記述されている。さらに表 1 に実験に用いたツリーバンクの性質を示す。ここで表中の大きさとはそのデータの要素数を指す。すなわち文であれば単語数、構文木であればノード数を表す。

4.2 実験手順

ここでは評価実験として下記の手順で実験を行った。

[実験手順]

1. ツリーバンクから 1 文を選び出し、これに対応する構文木とともに抽出する。
2. (1) で抽出した文を入力文、残ったデータをツリーバンクとして本システムにより解析を行う。
3. システムの出力した構文木を (1) で抽出した構文木と比較、評価する。
4. (1) から (3) を繰り返す。

³ 構文木中のノードについては Marcus 等の文献を参照 (Marcus, Kim, Marcinkiewicz *et al.* 1994)

表 2: 実験に用いた入力文の性質

	文	構文木
大きさの範囲	1 ~ 16	1 ~ 17
大きさの平均	6.45	7.18

今回用いたシステムは検索システムとして実現したため、入力文に対応する構文木がツリーバンク内に存在しない場合、正しく解析できない。そこで上記の入力文は、対応する構文木が他の文の構文木としてツリーバンク内に存在する文、すなわち前述の 376 文とした。このテスト文の性質を表 2 に示す。

また入出力の全体を評価するために再現率 (recall) と適合率 (precision) を用いた。ここで再現率とは解析可能な文 (上記の 376 文) 中でシステムが正しく解析できた文の割合であり、適合率はシステムが出力した構文木中で正しく解析できた構文木の割合を表す。ここでは特に類推妥当性に従って順位付けされた出力のうち上位 N 位以上の結果を用いて、再現率と適合率を調べた。

ただし実際問題として、編集距離の計算コストは非常に高く、また文や構文木の組み合わせ全てに対して類似関係を計算する必要があるため、実験には相当な時間がかかる。そこで予めインデックスを作り、これを用いる形で実験環境を作成した。この実験環境を構成するプログラムについての説明を以下に示す。

1. mkMatrix

ツリーバンク内の全ての文、構文木について、その相互間の距離を計算し、距離インデックスを作成する。距離インデックスは以下の情報を含む。

- リスト構造の数 (sizeList)
文の数、すなわちツリーバンク内のデータ数を表す。ここで、ツリーバンク内のデータにはファイル先頭から順に $0 \sim (\text{sizeList}-1)$ のリスト構造 ID がふられる。
- 木構造の数 (sizeTree)
異なる木構造の数を表す。ここで同じ木構造はマージし、同じ木構造には同じ木構造 ID をふることを仮定している。このためツリーバンク内のデータにはファイル先頭から出現順に $0 \sim (\text{sizeTree}-1)$ の木構造 ID がふられる。
- リスト構造 ID-木構造 ID の対応テーブル (conv[sizeList])
sizeList の大きさを持つ配列。conv[i] にはリスト構造 ID が i であるデータに対応する木構造 ID が格納される。
- 木構造 ID-リスト構造 ID の対応テーブル (inconv[sizeTree])
sizeList の大きさを持つ構造体の配列。構造体の要素としては、id と num がある。ここで inconv[i].id には、木構造 ID が i であるデータに対応する最も小さ

なりリスト構造 ID が格納される。また `inconv[i].num` には、木構造 ID が i であるデータに対応するリスト構造の数が格納される。

- リスト構造の大きさ (`lenList[sizeList]`)

`sizeList` の大きさを持つ配列。 `lenList[i]` にはリスト構造 ID が i であるデータの大きさ (リストの要素数) が格納される。

- 木構造の大きさ (`lenTree[sizeTree]`)

`sizeTree` の大きさを持つ配列。 `lenTree[i]` には木構造 ID が i であるデータの大きさ (木構造のノード数) が格納される。

- リスト構造間の距離 (`distList[sizeList][sizeList]`)

`sizeList` × `sizeList` の大きさを持つ 2 次元配列。 `distList[i][j]` にはリスト構造 ID が i であるデータとリスト構造 ID が j であるデータとの間の距離が格納される。ただし、 `distList[i][j] = distList[j][i]` であるため、インデックスには $i < j$ のデータのみを格納している。

- 木構造間の距離 (`distTree[sizeTree][sizeTree]`)

`sizeTree` × `sizeTree` の大きさを持つ 2 次元配列。 `distTree[i][j]` には木構造 ID が i であるデータと木構造 ID が j であるデータとの間の距離が格納される。ただし、 `distTree[i][j] = distTree[j][i]` であるため、インデックスには $i < j$ のデータのみを格納している。

さらにこの距離インデックスを利用するために、インデックスのロード関数や各値の参照関数も用意した。

2. mkIndex

距離インデックスを用いて、4 項アナロジー関係を満たすリスト構造の組合せを計算し、類似関係インデックスを作成する。ただしこの組合せは数が非常に多いため、インデックス化する際に圧縮形式を使っている。圧縮形式では、1 つの組合せを 6byte で表現し、12bit 毎にリスト構造 ID をバイナリ形式で格納する。

さらにこの類似関係インデックスを利用するために、インデックスのロード関数も用意した。

3. rerank

距離インデックスと類似関係インデックスを利用して、構文解析を行う。具体的にはまず、類似関係インデックスから得られたリスト ID の組合せに対してアナロジーによる構文解析の基本原則を適用し、出力すべき構文木を計算する。さらに距離インデックスを用いて、各々の出力構文木に対する類似性、類推パス数を計算し、出力する。出力形式は次の通りである。

Sim Freq (In Out) Dist

ただし、Sim : 類似性

Freq : 類推パス数

In : 入力 (リスト構造 ID)
Out : 出力 (木構造 ID)
Dist : 入出力に対する木構造間距離

4. mkResult

rerank の出力結果を処理して結果集計を行う。具体的にはまず、1 行ごとに類推妥当性を計算した後、入力 ID ごとに類推妥当性の順でソートする。さらにオプションで与えられた順位までの結果を用いて、適合率、再現率、正解構文木と出力構文木の間の平均距離、正解構文木と出力構文木の間の最大距離を出力する。出力形式は以下の通りである。

N	Prec	Rec1	Rec2	AveDist	MaxDist
ただし、N					
	Prec				
	Rec1				
	Rec2				
	AveDist				
	MaxDist				

ここで類推妥当性の計算方法はオプションによって変更できるようにした。

4.3 基本原理と類推妥当性の評価実験

実験結果を図 4 と図 5 に示す (詳細は付録 A を参照)。ここで図 4 は類似性として式 (4) を用いた結果であり、図 5 は式 (5) を用いた結果である。2 つのグラフの各点に付加した数字は上位何位までの結果を用いたかを表す。特にここでは、類推妥当性として a) 類似性のみ (式 (6) 中で $\alpha = 1$ 、 $\beta = 0$)、b) 類推パス数のみ (式 (6) 中で $\alpha = 0$ 、 $\beta = 1$)、c) 両者の重み付き和 (式 (6) 中で $\alpha = 10$ 、 $\beta = 1$) を用いた結果を示した。

図 4 と図 5 を比較すると、類似性として各項の類似性を独立に計算した評価値の方が若干高い精度が出ている。これは、4 項間で定義される 3 つの類似性に 1 つ大きなものがあれば、より類推妥当性が高くなることを意味している。

また全出力に対する結果 (図 4 と図 5 のグラフで最も左側にある点) を見ると、376 文の入力に対して 375 文で正しい構文木が得られたことが分かる (再現率 99.7%)。このことから類似関係のみを用いた本手法が十分に構文解析器として機能し得ると考えられる。ここで失敗した 1 文は入力文としては長い文であり (16 単語)、4 項アナロジー関係にある適切な文がツリーバンクに存在しなかったことが原因であった。

次に各々のグラフの全体を見ると、各々の評価値を用いた結果が全てに右上がりの曲線を描いており、高い順位に正しい構文木が偏在していることが分かる。これは各々の評価値が解析結果の尤度として有効に働いていることを意味する。

さらに各々のグラフで a) と b) を比べると、a) の類似性のみを類推妥当性として用いた方がより高い精度が出ることが分かる (図 5 では最適解だけで再現率 48.1%、適合率 57.2%)。これはツリーバンクのサイズが小さいために、データの偏りによる誤差が類推パス数に悪

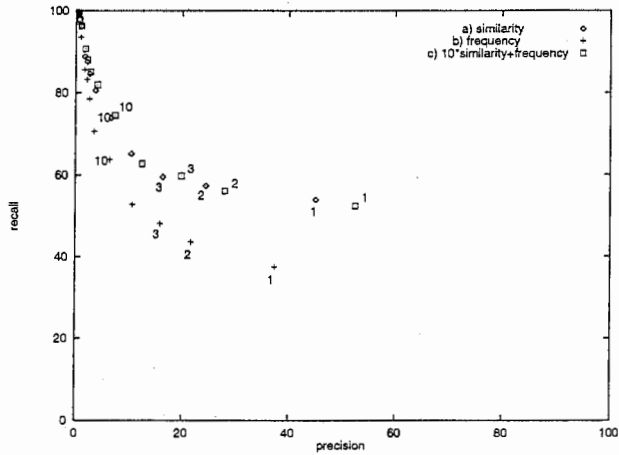


図 4: 評価実験の結果 (類似性の計算では 4 項全体を考慮)

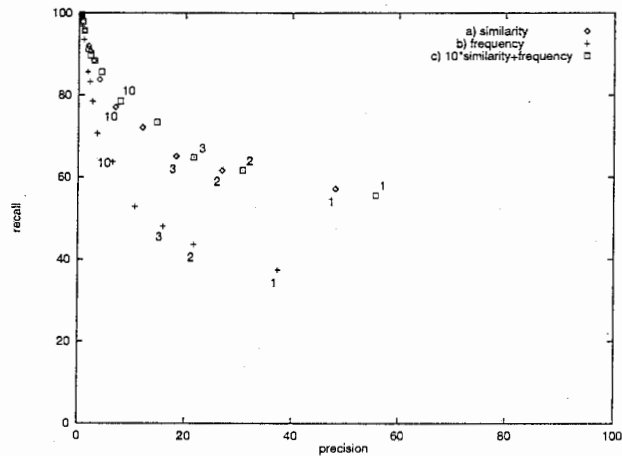


図 5: 評価実験の結果 (類似性の計算では各項を独立に評価)

影響を及ぼした結果であると考えられる。この結果を考慮して、c) では類推妥当性の式 6 中で類似性の重み α を大きくした。これにより、類似性だけでは一意に決められなかった入力についても曖昧性が解消できるようになり、適合率をほとんど落さずに再現率をあげることができた (図 5 では最適解だけで再現率、適合率ともに 55.6%)。

4.4 品詞情報を用いた評価実験

本手法で正しく解析を行うためには、入力に類似したデータがツリーバンク内に存在している必要がある。しかし文、すなわち単語列を入力とすると、単語の種類が多いためにデータのスパース性が問題となりやすい。実際、先の実験ではこの問題のために正しく解析できない例が見られた。一方、一般的な構文解析手法では、比較的高い精度が実現されている形態素解析の出力を前提として品詞情報を用いるものが多い。

そこで以下では、文の代りに品詞列を用いることを考える。品詞という抽象度の高いデータを用いることで、上記のスパース性の問題が緩和できると考えられる。また本手法では、データに対する要件はそれがリスト構造もしくはツリー構造で表現できることだけであり、

表 3: 品詞-構文木を納めたツリーバンクの性質

	品詞列	構文木
大きさの範囲	1 ~ 34	1 ~ 44
大きさの平均	7.54	8.35
距離の平均	7.44	8.21

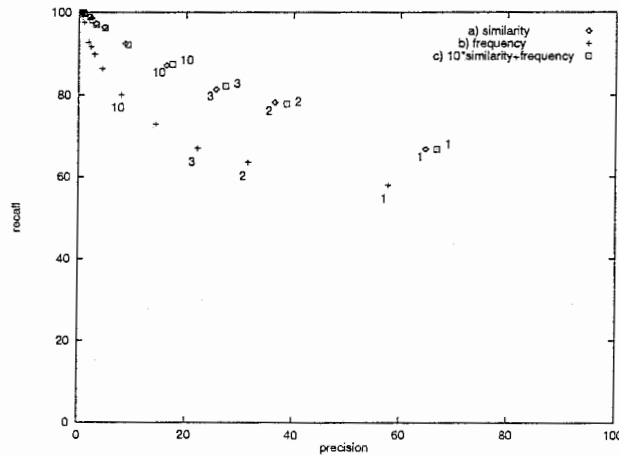


図 6: 品詞情報を用いた評価実験の結果

品詞列についても文と全く同じように扱うことができる。

これを評価するために、先のツリーバンク内の文データを Penn Treebank で提供されているタギングデータに置き換えて、実験を行った。この置き換えによって先のツリーバンク内データの例は以下のようになる。

構文木 : S(NP-SBJ, VP(NP, NP(NP, PP-DIR(NP), PP-DIR(NP))))

文 : VB PRP DT NN IN NNP TO NNP NNP

表 3 にこのツリーバンクの性質を示す。表 1 と比べると、品詞列はその大きさ(長さ)が文とほとんど変わらないにもかかわらず、距離の平均が小さくなっていることが分かる。これは品詞列の距離空間が文の距離空間に比べて狭く、データが密に存在していることを意味する。

この品詞-構文木の組を納めたツリーバンクを用いて、先と全く同じ実験を行った。ただし類似性としては先の実験で精度のよかった式(5)を用いた。その結果を図 6 に示す(詳細は付録 B を参照)。

図 6 を見るとまず、文を入力とした実験と同様の議論が成り立ち、関係性や類推妥当性が有効に働いていることが確認できる。また、図 5 に比べて再現率、適合率がともに向上していることが分かる(c の最適解だけで再現率、適合率ともに 66.8%)。特に全出力に対する結果(図 6 のグラフで最も左側にある点)を見ると、376 文の入力全てに対して正しい構文木が得られており(再現率 100%)、文を入力とした実験では解析できなかった文が解析できるようになった。これらの結果から、スパース性の緩和が精度の向上に有効である

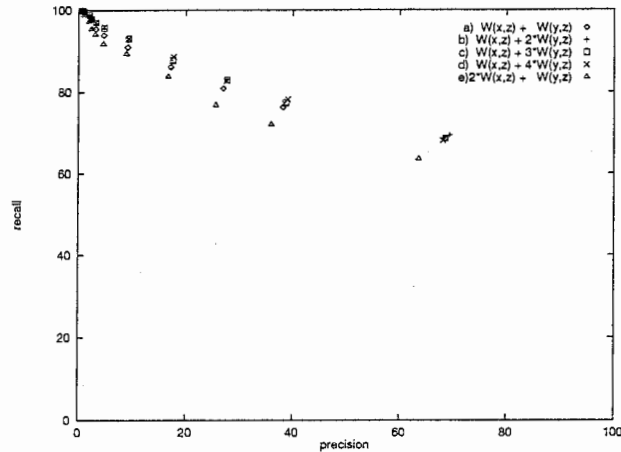


図 7: 単語情報と品詞情報を用いた評価実験の結果

ことが分かる。

また入力とした 376 の品詞列全てについて、各々の解析で用いたツリーバンクとの関係を調べたところ、入力と全く同じ品詞列がツリーバンク内に格納されており、かつこのツリーバンク内データの構文木が入力に対する正しい構文木と等しいものである例が 215 例あった。ただしこのうちの 26 例については、品詞列としては同じでも構文木の異なるデータがツリーバンク内に存在し、曖昧性があった。これは、入力と同じ品詞列に付加された構文木を出力するという単純な手法では 50%(189/376) ~ 57% (215/376) の再現率、適合率しか得られないことを意味する。これに対して本手法は最適解だけでもこれ以上の精度が得られており、より頑健な解析が実現されていることが分かる。

4.5 単語情報と品詞情報を用いた評価実験

上記で示した品詞列を入力とした実験について誤りを調べると、品詞の抽象度が高すぎることに起因したものが多数見られた。例えば、Penn Treebank 内では「in」「from」「on」などの前置詞には「IN」という共通の品詞が与えられているにもかかわらず、これらの前置詞句に対する構文木中のノードでは意味情報まで考慮して「PP-LOC」(場所を表す前置詞句)、「PP-DIR」(方向を表す前置詞句)、「PP-TMP」(時間を表す前置詞句)などの使い分けが必要であった。このような場合には単語の情報まで必要となるだろう。

そこで、単語と品詞の情報を組み合わせて利用することを考えた。ここでは単純に上記の 2 つの実験で得られた結果を組み合わせることで評価実験を行った。すなわち入力文 x 、その品詞列 y に対して構文木 z が出力されたとき、その類推妥当性を下式によって計算し、曖昧性解消に用いた。

$$W'(x, y, z) = \alpha \frac{W(x, z)}{\max_{y(z_i)}(W(x, z_i))} + \beta \frac{W(y, z)}{\max_{y(z_i)}(W(y, z_i))} \quad (7)$$

ここで $W(x, z)$ 、 $W(y, z)$ としては、上記の 2 つの実験で精度のよかった類似性と類推パス数の重み付き和を用いた。上式 7 の α 、 β の値を変えた 5 つの実験の結果を図 7 に示す(詳細は付録 C を参照)。

図 7 によると b) の $\alpha = 1$ 、 $\beta = 2$ のときに最も精度がよいことが分かる(最適解だけで

再現率、適合率ともに69.4%)。また図7中の5つの結果を見ると、品詞列に対する結果に重きをおいた類推妥当性を用いることで、上記2つの実験よりも高い精度が実現できることが分かる。これは4.3や4.4で示した実験結果や検討とも一致しており、他のツリーバンクを用いるときにも同様のことがいえると考えられる。

さらに最適解として出力された構文木と正しい構文木との距離を調べたところ、全体の平均では0.95、誤った出力だけを対象としても平均で3.18であった。この値は、表1で示したツリーバンク内構文木の距離の平均と比べて十分小さな値であり、本手法では完全に解析できなくても正解に比較的似た構文木を出力していることを意味している。

5 考察

一般にコーパスを用いる手法の精度は、利用するコーパスの質や量に依存する。本手法も、入力に類似したデータがツリーバンク内に存在しない場合は正しい構文木を得ることができない。しかし上記の実験結果とここで用いたツリーバンクのサイズを考えると、本手法は非常に頑健に働いていると考えられる。実際、本手法では辞書を全く用いておらず、周囲の環境がある程度似ている文がツリーバンク内に存在すれば未知語(ツリーバンク内に出現しない語)を含む文についても解析が可能である。また実験システムの出力した最適解が正しい構文木に比較的似ており、誤りは部分的なものであることも、本手法の頑健性を表している。

また本手法は、リスト構造と木構造からなるツリーバンクであればどのような文法が用いられていても、これを利用することができる。さらにツリーバンクに文を加えるという単純な操作だけで、インクリメンタルに学習を進められるという利点もある。

このため本手法の応用として、ツリーバンク構築の支援が考えられる。人手でツリーバンクを構築する際の問題点として、膨大な作業コストとデータの無矛盾性確保の難しさがある。そこで例えば、新たな構文木の作成作業に際し、対象ツリーバンクの完成部分に本手法を適用することで構文木候補を提示する支援ツールが考えられる。評価実験でシステムの最適解と正しい構文木との距離が平均で 0.95 だったことを考えると、提示された構文木の 1 つのノードを編集することで正しい構文木が得られることになり、ツリーバンク構築作業のコストを低減することができるだろう。また作成したデータを参照しながら構築作業が進められるため、データの信頼性も向上すると考えられる。さらに新たに作成した構文木をツリーバンクに登録するだけで、次の構文木作成作業にこの結果を利用することができる(安藤, Lepage 1996)。

さらに本手法は他の構文解析手法と融合することによって、その精度を高めることができると思われる。本手法ではデータ全体の構造的な類似性を利用して、文と構文木との対応関係が正当であるかどうかを評価する。このため、従来の局所的な解析の組合せとして構文解析を行う手法や、統計情報、意味的類似性を用いる手法とは、出力される曖昧性の質が異なると考えられる。これらの手法を融合することによって構文解析の曖昧性を小さくできるだろう。特に本手法は文法体系に依存しないため、他の手法との融合が比較的容易に考え得る。例えば、ある構文解析手法によって出力された複数の構文木を類推という観点から評価し、これを曖昧性解消に利用することができる。ここでツリーバンクとしては、過去に正しく解析できた例を集め、これを用いればよい(安藤, Lepage 1997)。

6 おわりに

本稿では、ツリーバンクを用いて入力文に対する構文木を類推する機構を提案した。ここでは特に基本原理として働く類似関係の有効性について評価した。Penn Treebank を用いた評価実験の結果、単語の表層情報と品詞情報を用いることで解析可能な文の約70%が正しく解析でき、また誤りについても比較的正確に似た構文木を出力することができた。

この手法はツリーバンクを知識として直接用いるため、比較的容易に他の構文解析手法との融合が考えられ、さらに精度を向上することができると考えられる。特に統計情報や意味的類似性とは異なる原理でツリーバンクを用いるため、ツリーバンク内の情報利用について新たな視点を提供できるだろう。さらにこの手法はデータ間の類似性に基づいており、未知語などを含む入力に対しても頑健に働く。

今回は文単位の類似性を用いたが、句や節などのより小さな構造の上での類似性を利用することでツリーバンクのスパース性の問題をより緩和できると考えられる。今後これについても検討する。また他の構文解析手法との融合についても検討、評価を進める予定である。

参考文献

- 安藤真一, Lepage, Y. (1996). “類似検索機能を備えたツリーバンク構築エディタ.” 情報処理学会第52回全国大会 予稿集, 3巻, pp. 53-54.
- 安藤真一, Lepage, Y. (1997). “4項アナロジー関係の構文解析への応用.” 自然言語処理学会第3回年次大会 発表論文集, pp. 213-216.
- Black, E., Jelinek, F., Lafferty, J., Magerman, D., Mercer, R., and Roukos, S. (1993). “Towards History-based Grammars: Using Richer Models for Probabilistic Parsing.” In *Proceedings of the ACL-93*, pp. 31-37.
- Bod, R. (1993). “Using an Annotated Corpus as a Stochastic Grammar.” In *Proceedings of the EACL-93*, pp. 37-44.
- Brill, E. (1993). “Automatic Grammar Induction and Parsing Free Text: A Transformation-Based Approach.” In *Proceedings of the ACL-93*, pp. 259-265.
- Briscoe, T. (1995). *Survey of the State of the Art in Human Language Technology*, chap. Robust Parsing. <http://www.cse.ogi.edu/CSLU/HLTsurvey/ch3node9.html#SECTION37>.
- Lepage, Y. and Ando, S. (1996). “Saussurian Analogy: a theoretical account and its application.” In *Proceedings of the Coling-96*, Vol. 2, pp. 717-722.
- Marcus, M., Kim, G., Marcinkiewicz, M. et al. (1994). “The Penn Treebank: Annotating Predicate Argument Structure.” In *Proceedings of the ARPA Human Language Technology Workshop*.

- Resnik, P. (1992). "Probabilistic Tree-Adjoining Grammar as a Framework for Statistical Natural Language Processing." In *Proceedings of the Coling-92*, Vol. 2, pp. 418-424.
- Saussure, F. (1916). *Cours de linguistique générale*. publié par Charles Bally et Albert Sechehaye.
- Sekine, S. and Grishman, R. (1995). "A Corpus-based Probabilistic Grammar with Only Two Non-terminals." In *Proceedings of the 4th International Workshop on Parsing Technology*, pp. 216-223.
- Selkow, S. (1977). "The Tree-to-Tree Editing Problem." *Information Processing Letters*, 6 (6), 184-186.
- Sumita, E. and Iida, H. (1991). "Experiments and Prospects of Example-Based Machine Translation." In *Proceedings of the ACL-91*, pp. 185-192.
- Wagner, R. and Fischer, M. (1974). "The String-to-String Correction Problem." *Journal for the ACM*, 21 (1), 168-173.

A 単語情報を用いた評価実験の詳細結果

A.1 4項全体のバランスを考慮した類似性を用いた実験結果の詳細

○ 類似性を用いた実験結果

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	45.1 (203/450)	54.0 (203/376)	2.26	15
2	24.5 (216/882)	57.4 (216/376)	3.19	15
3	16.4 (224/1363)	59.6 (224/376)	3.67	15
5	10.4 (245/2351)	65.2 (245/376)	4.06	15
10	6.7 (278/4162)	73.9 (278/376)	4.46	15
20	3.7 (303/8242)	80.6 (303/376)	4.76	16
30	2.6 (318/12059)	84.6 (318/376)	5.08	16
40	2.1 (329/16009)	87.5 (329/376)	5.29	17
50	1.7 (334/19917)	88.8 (334/376)	5.44	17
100	1.0 (362/38100)	96.3 (362/376)	6.08	17
150	0.6 (368/56768)	97.9 (368/376)	6.61	19
200	0.5 (373/75199)	99.2 (373/376)	7.15	20
240	0.4 (375/89258)	99.7 (375/376)	7.68	20
all	0.4 (375/92018)	99.7 (375/376)	7.83	23

○ 類推パス数を用いた実験結果

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	37.4 (141/377)	37.5 (141/376)	3.04	12
2	21.7 (164/755)	43.6 (164/376)	3.96	14
3	15.9 (181/1137)	48.1 (181/376)	4.35	14
5	10.6 (199/1886)	52.9 (199/376)	4.52	14
10	6.4 (240/3779)	63.8 (240/376)	4.81	19
20	3.5 (266/7577)	70.7 (266/376)	5.27	19
30	2.6 (295/11364)	78.5 (295/376)	5.41	19
40	2.1 (313/15136)	83.2 (313/376)	5.56	19
50	1.7 (322/18898)	85.6 (322/376)	5.69	19
100	0.9 (352/37816)	93.6 (352/376)	6.25	20
150	0.6 (364/56758)	96.8 (364/376)	6.78	20
200	0.5 (373/75444)	99.2 (373/376)	7.21	20
218	0.5 (375/82051)	99.7 (375/376)	7.42	20
all	0.4 (375/92018)	99.7 (375/376)	7.83	23

○ 類似性と類推パス数を用いた実験結果

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	52.4 (197/376)	52.4 (197/376)	1.99	15
2	28.1 (211/752)	56.1 (211/376)	3.04	15
3	19.9 (225/1128)	59.8 (225/376)	3.50	15
5	12.5 (236/1883)	62.8 (236/376)	4.00	15
10	7.4 (280/3763)	74.5 (280/376)	4.42	15
20	4.1 (308/7520)	81.9 (308/376)	4.72	16
30	2.8 (320/11284)	85.1 (320/376)	4.93	17
40	2.2 (331/15047)	88.0 (331/376)	5.12	17
50	1.8 (341/18805)	90.7 (341/376)	5.30	17
100	1.0 (362/37559)	96.3 (362/376)	5.99	17
150	0.7 (368/56305)	97.9 (368/376)	6.57	18
200	0.5 (373/74989)	99.2 (373/376)	7.15	20
220	0.5 (375/82371)	99.7 (375/376)	7.39	20
all	0.4 (375/92018)	99.7 (375/376)	7.83	23

A.2 各項を独立に評価した類似性を用いた実験結果の詳細

○ 類似性を用いた実験結果

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	48.1 (215/447)	57.2 (215/376)	1.99	13
2	26.9 (232/862)	61.7 (232/376)	3.01	15
3	18.3 (245/1338)	65.2 (245/376)	3.47	15
5	11.9 (271/2269)	72.1 (271/376)	3.93	15
10	6.9 (290/4191)	77.1 (290/376)	4.37	15
20	3.9 (315/8093)	83.8 (315/376)	4.74	17
30	2.8 (332/11853)	88.3 (332/376)	5.06	17
40	2.2 (341/15677)	90.7 (341/376)	5.29	17
50	1.8 (346/19418)	92.0 (346/376)	5.52	20
100	1.0 (360/37885)	95.7 (360/376)	6.28	20
150	0.7 (368/56547)	97.9 (368/376)	6.88	20
200	0.5 (373/75039)	99.2 (373/376)	7.23	20
240	0.4 (375/89185)	99.7 (375/376)	7.68	20
all	0.4 (375/92018)	99.7 (375/376)	7.83	23

○ 類推パス数を用いた実験結果

A.1の実験結果と同じであるため省略する。

○ 類似性と類推パス数を用いた実験結果

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	55.6 (209/376)	55.6 (209/376)	1.71	13
2	30.8 (232/754)	61.7 (232/376)	2.87	14
3	21.6 (244/1128)	64.9 (244/376)	3.33	15
5	14.7 (276/1881)	73.4 (276/376)	3.83	15
10	7.8 (295/3762)	78.5 (295/376)	4.34	15
20	4.3 (322/7522)	85.6 (322/376)	4.71	17
30	2.9 (332/11284)	88.3 (332/376)	4.97	17
40	2.2 (337/15042)	89.6 (337/376)	5.21	20
50	1.8 (343/18804)	91.2 (343/376)	5.44	20
100	1.0 (360/37555)	95.7 (360/376)	6.21	20
150	0.7 (368/56303)	97.9 (368/376)	6.86	20
200	0.5 (374/74992)	99.5 (374/376)	7.22	20
233	0.4 (375/86941)	99.7 (375/376)	7.58	20
all	0.4 (375/92018)	99.7 (375/376)	7.83	23

B 品詞情報を用いた評価実験の詳細結果

○ 類似性を用いた実験結果

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	64.7 (251/388)	66.8 (251/376)	1.03	11
2	36.6 (294/804)	78.2 (294/376)	2.05	15
3	25.6 (306/1196)	81.4 (306/376)	2.59	15
5	16.4 (328/2003)	87.2 (328/376)	3.08	15
10	8.7 (348/3986)	92.6 (348/376)	3.60	15
20	4.7 (363/7799)	96.5 (363/376)	4.18	18
30	3.2 (366/11468)	97.3 (366/376)	4.60	18
40	2.4 (371/15292)	98.7 (371/376)	4.92	18
50	2.0 (372/19007)	98.9 (372/376)	5.21	18
100	1.0 (375/37752)	99.7 (375/376)	6.16	18
123	0.8 (376/46392)	100.0 (376/376)	6.36	18
all	0.4 (376/90521)	100.0 (376/376)	7.76	21

○ 類推パス数を用いた実験結果

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	57.8 (218/377)	58.0 (218/376)	1.78	11
2	31.6 (239/756)	63.6 (239/376)	2.91	13
3	22.2 (252/1134)	67.0 (252/376)	3.42	13
5	14.5 (274/1890)	72.9 (274/376)	3.78	16
10	8.0 (301/3773)	80.1 (301/376)	4.28	16
20	4.3 (325/7579)	86.4 (325/376)	4.81	19
30	3.0 (338/11408)	89.9 (338/376)	5.13	19
40	2.3 (345/15209)	91.8 (345/376)	5.31	19
50	1.8 (349/19015)	92.8 (349/376)	5.48	19
100	1.0 (367/38105)	97.6 (367/376)	6.11	19
150	0.7 (372/57099)	98.9 (372/376)	6.68	19
174	0.6 (376/66211)	100.0 (376/376)	6.93	20
all	0.4 (376/90521)	100.0 (376/376)	7.76	21

○ 類似性と類推パス数を用いた実験結果

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	66.8 (251/376)	66.8 (251/376)	1.00	10
2	38.8 (293/755)	77.9 (293/376)	2.03	15
3	27.4 (309/1129)	82.2 (309/376)	2.53	15
5	17.5 (329/1880)	87.5 (329/376)	2.98	15
10	9.2 (347/3760)	92.3 (347/376)	3.53	15
20	4.8 (362/7521)	96.3 (362/376)	4.13	18
30	3.2 (365/11284)	97.1 (365/376)	4.54	18
40	2.5 (369/15042)	98.1 (369/376)	4.86	18
50	2.0 (372/18801)	98.9 (372/376)	5.14	18
100	1.0 (375/37604)	99.7 (375/376)	6.15	18
117	0.9 (376/44001)	100.0 (376/376)	6.29	18
all	0.4 (376/90521)	100.0 (376/376)	7.76	21

C 単語情報と品詞情報を用いた評価実験の詳細結果

○ 単語情報と品詞情報の重みが 1:1

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	68.6 (258/376)	68.6 (258/376)	1.06	10
2	38.2 (287/752)	76.3 (287/376)	2.18	15
3	27.0 (304/1128)	80.9 (304/376)	2.66	15
5	17.2 (324/1880)	86.2 (324/376)	3.24	15
10	9.1 (342/3760)	91.0 (342/376)	3.81	16
20	4.7 (353/7520)	93.9 (353/376)	4.32	17
30	3.2 (359/11281)	95.5 (359/376)	4.68	17
40	2.4 (368/15043)	97.9 (368/376)	4.95	17
50	2.0 (370/18803)	98.4 (370/376)	5.17	17
100	1.0 (374/37602)	99.5 (374/376)	6.17	20
140	0.7 (376/52640)	100.0 (376/376)	6.61	20
all	0.4 (376/93527)	100.0 (376/376)	7.87	23

○ 単語情報と品詞情報の重みが 1:2

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	69.4 (261/376)	69.4 (261/376)	0.95	10
2	38.4 (289/752)	76.9 (289/376)	2.03	15
3	27.7 (313/1128)	83.2 (313/376)	2.53	15
5	17.4 (328/1880)	87.2 (328/376)	3.09	15
10	9.2 (347/3762)	92.3 (347/376)	3.67	16
20	4.8 (359/7522)	95.5 (359/376)	4.24	17
30	3.2 (364/11280)	96.8 (364/376)	4.61	17
40	2.4 (368/15043)	97.9 (368/376)	4.91	18
50	2.0 (372/18802)	98.9 (372/376)	5.15	18
100	1.0 (375/37600)	99.7 (375/376)	6.14	20
140	0.7 (376/52640)	100.0 (376/376)	6.56	20
all	0.4 (376/93527)	100.0 (376/376)	7.87	23

○ 単語情報と品詞情報の重みが 1:3

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	68.6 (258/376)	68.6 (258/376)	0.95	10
2	38.7 (291/752)	77.4 (291/376)	2.01	15
3	27.7 (312/1128)	83.0 (312/376)	2.53	15
5	17.6 (330/1880)	87.8 (330/376)	3.01	15
10	9.3 (350/3760)	93.1 (350/376)	3.61	16
20	4.8 (360/7520)	95.7 (360/376)	4.20	17
30	3.2 (365/11281)	97.1 (365/376)	4.57	17
40	2.4 (368/15040)	97.9 (368/376)	4.90	18
50	2.0 (372/18800)	98.9 (372/376)	5.14	18
100	1.0 (375/37600)	99.7 (375/376)	6.13	20
140	0.7 (376/52641)	100.0 (376/376)	6.54	20
all	0.4 (376/93527)	100.0 (376/376)	7.87	23

○ 単語情報と品詞情報の重みが 1:4

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	68.1 (256/376)	68.1 (256/376)	0.96	10
2	39.1 (294/752)	78.2 (294/376)	1.98	15
3	27.6 (311/1128)	82.7 (311/376)	2.52	15
5	17.7 (333/1880)	88.6 (333/376)	2.96	15
10	9.3 (350/3760)	93.1 (350/376)	3.58	16
20	4.8 (361/7520)	96.0 (361/376)	4.18	17
30	3.2 (365/11280)	97.1 (365/376)	4.56	18
40	2.4 (368/15040)	97.9 (368/376)	4.88	18
50	2.0 (371/18800)	98.7 (371/376)	5.14	18
100	1.0 (375/37602)	99.7 (375/376)	6.13	20
123	0.8 (376/46250)	100.0 (376/376)	6.36	20
all	0.4 (376/93527)	100.0 (376/376)	7.87	23

○ 単語情報と品詞情報の重みが 2:1

有効順位	適合率 (正解出力数 / 全出力数)	再現率 (正解出力数 / 全入力数)	平均距離	最大距離
1	63.6 (239/376)	63.6 (239/376)	1.30	11
2	36.0 (271/752)	72.1 (271/376)	2.38	15
3	25.6 (289/1128)	76.9 (289/376)	2.86	15
5	16.7 (315/1882)	83.8 (315/376)	3.37	15
10	8.9 (336/3762)	89.4 (336/376)	4.00	16
20	4.6 (345/7521)	91.8 (345/376)	4.44	17
30	3.1 (354/11280)	94.1 (354/376)	4.75	17
40	2.4 (359/15040)	95.5 (359/376)	5.03	17
50	1.9 (366/18800)	97.3 (366/376)	5.23	20
100	1.0 (372/37600)	98.9 (372/376)	6.19	20
143	0.7 (376/53771)	100.0 (376/376)	6.70	20
all	0.4 (376/93527)	100.0 (376/376)	7.87	23