自 然 言 語 処 理 103－5
（1994．9．15）

類似度テンプレートを利用した高速類似用例検索

宇津呂 武仁

奈良先端科学技術大学院大学 情報科学研究科

従来，用例に基づく自然言語処理においては，用例データベース中の全用例と入力例の間で類似度計算を行なう(全用例検索)ため，用例検索時間が用例数に比例して大きくなるという問題があった．これに対して，本論文では，用例間の類似度を用いて構造化された用例空間中を効率よく探索することにより，全用例検索を行なわずに類似用例を高速に検索するという手法について述べる．具体的には，類似度計算のあらゆるパターンを抽象化して表現した類似度テンプレートというデータ構造を考え，類似度テンプレートから適当な検索質問を生成して類似用例を検索する．実験の結果，用例数の増加に対して，用例検索時間がほぼ一定となった．

Efficient Retrieval of Similar Examples based-on Similarity Templates

Takehito UTSURO

Graduate School of Information Science,
Nara Institute of Science and Technology

In example-based NLP, the problem of computational cost of example retrieval is severe, since the retrieval time increases in proportion to the number of examples in the database. This paper proposes a novel example retrieval method for avoiding full retrieval of examples. The basic idea is to efficiently search the most similar examples through the example space which is structurized by the similarity measure of examples. The idea is realized by the notion of *similarity template* and retrieval query. Similarity templates are used for enumerating all the possible patterns of calculating the similarity. The method achieved almost constant time retrieval, independent of the number of examples.

1 はじめに

文献 [3] において「アナロジーによる翻訳」の考え方が提唱されて以来，用例に基づく自然言語処理に関する研究がこれまで数多くなされてきた．用例に基づく自然言語処理の基本的な考え方は，自然言語処理におけるあるタスクを実行する際に，人間によって書かれた規則を用いるのではなく，過去に類似用例に対してそのタスクを実行した結果を模倣することによってそのタスクを実行するというものである．その際の処理手順の概要は以下のようになる，1) まず，あらかじめ，用例およびそのタスクを実行した結果を用例データベースに集めておく，2) 入力が与えられると，用例データベースから類似用例およびタスクの実行結果を検索する，3) 類似用例に対するタスクの実行結果を現在の入力に適合するように修正して，現在の入力に対する出力を得る．

規則に基づくアプローチに比べ，用例に基づくアプローチは以下のような利点を持つ，a) 一度システムを構築してしまうと，新たな用例を追加していくだけでシステムのパフォーマンスの向上が望めるため，システムの維持が容易である，b) 入力例と用例との間でよりきめ細かく定義された類似度を採用するだけで，統語レベルあるいは意味レベルでのより細かな類別が可能である．

用例に基づく自然言語処理の研究としてこれまでなされたものを挙げると，例えば，機械翻訳における動詞の訳語選択において用例を用いるもの [5]，格解析における格フレーム選択において用例を用いるもの [2] などがある．また，用例に基づく自然言語処理においてこれまで採用されてきた類似度の尺度としては，単語間の意味的近さを既存のシソーラスによって測り，単語間の類似度を用いてあるまとまった構造の間の類似度を定義するというものが一般的である．

ところが，用例に基づくアプローチにおいては，用例データベースから類似用例を検索する際に，用例データベース中の全用例と入力例の間で類似度の計算を行って最も類似した用例を求める必要があった．用例データベース中の全用例と入力例の間で類似度の計算を行うことをここでは**全用例検索** (Full Retrieval) と呼ぶが，全用例検索を行なう場合，用例検索の計算コストが用例データベース中の用例数に比例して大きくなるため，用例に基づくアプローチにおいて大きな問題となっていた．

これに対して，最近，自然言語処理における計算コストの問題を克服するものとして，超並列計算機を用いた自然言語処理のアプローチが提唱され盛んに研究されている [1, 8]．特に，用例に基づく自然言語処理では，「記憶に基づく推論 (Memory-Based Reasoning: MBR)」 [6] というモデルに基づいたものが多く，そこでは超並列計算機上の用例がデータ並列に探索され入力と最も類似した用例が高速に検索される．例えば，文献 [7] においては，大量の用例を用いて日本語の「A の B」というフレーズを英語に翻訳するシステムを超並列計算機上に実現した結果，計算コストの大幅な削減が達成されたという報告がなされている．

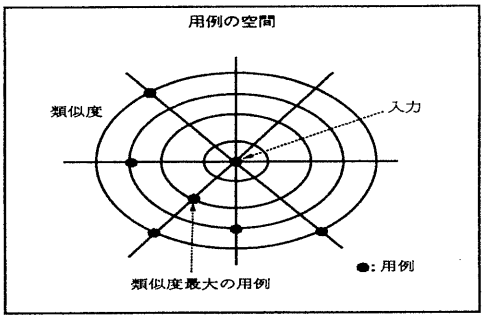

図 1: 用例の空間の構造化

しかし，超並列計算機を用いたアプローチも基本的には全用例検索に基づくため，入力例と用例データベースの全用例との間で類似度の計算を行なう必要があり，検索の際の総計算コストは削減されていない．また，超並列計算機を用いたアプローチは，当然超並列計算機上でのみ実現可能であるが，価格面および普及度からいって，現段階では，超並列計算機が一般利用者に対して広く利用可能になっているとは言い難い．したがって，比較的低価格で手に入るワークステーション級の逐次計算機上で，用例に基づく自然言語処理のシステムを適当な規模で実現するという目的のもとでは，全用例検索をおこなわない高速検索法を実現して，用例検索の際の計算コストの問題を回避することは十分意味があると言える．

そこで本論文では，全用例検索を行なわずに用例データベースから類似用例を効率よく検索する方法として，用例間の類似度を用いて用例の空間を構造化し，その構造を利用して類似用例を高速に検索する手法について述べる．この方法を模式的にとらえると図 1のようになる．図に示すように，入力が与えられると，用例間の類似度を用いることにより入力を中心とした同心円状に用例の空間を構造化してとらえる．このように構造化された用例空間を効率よく探索することにより，全用例を検索することなく類似度最大の用例を検索する．

具体的に用例の空間を構造化する際には，入力から適当な距離にある用例を検索するための**検索質問** (Retrieval Query) を生成し，これらの検索質問を組み合わせることにより用例の空間を入力を中心とする同心円状に構造化してとらえるという方法をとる．検索質問を生成するには，類似度計算のあらゆるパターンを抽象化し

て表現した類似度テンプレート (Similarity Template) というデータ構造を考え，類似度テンプレートと入力から適当な類似度の用例を検索するための検索質問を生成するという方法をとる．これらについては，第 2 章で述べる．次に，第 3 章で，類似用例検索の例として，日本語の表層格構造の用例を検索する問題をとりあげる．本論文で述べる日本語の表層格構造の類似度では，名詞間の類似度をシソーラスから計算するため，第 4 章で述べるように，シソーラスの構造を利用することにより与えられた検索質問を満たす用例を高速に検索することが可能となる．第 5 章で，日本語表層格構造の類似用例検索の例を示し，さらに，第 6 章で，システムのサイズおよび検索時間について本論文の手法を評価する．

2 類似度テンプレートを利用した類似用例検索

ここでは，まず，用例間の類似度および類似度最大の用例の検索について述べる．次に，類似度計算のあらゆるパターンを抽象化して表現したデータ構造として類似度テンプレートを導入し，類似度テンプレートを介して検索質問を生成する方法について述べる．さらに，検索質問の包含関係を利用して類似度最大の用例を二分探索によって効率よく探索する方法を説明する．

2.1 用例間の類似度および類似度最大の用例の検索

あらゆる用例の集合を E，類似度の値の集合を実数の集合 Rとすると，一般に，二つの用例間の類似度を計算する関数 sim は，以下のように，二つの用例を引数にとり類似度を返す関数とみなすことができる．

$$sim:\ E \times E \mapsto R$$

また，現在の用例データベースを用例集合 Egとすると，類似度最大の用例の検索とは，用例集合 Egから入力用例 e_{in}との間で最大の類似度を与える用例の集合 $Eg_{max}(e_{in})$ を検索することに相当する．

$$Eg_{max}(e_{in}) = \left\{ e \mid e \in Eg, sim(e, e_{in}) = \max_{e} sim(e, e_{in}) \right\}$$

2.2 類似度テンプレートを介した類似度計算および検索質問生成

まず，類似度に関して，次の

条件 1

用例間の類似度計算の結果を有限個のパターンに帰着することができる．

という条件が成り立っているとする．このとき，用例間の類似度計算において，類似度計算のあらゆるパターンを抽象化して表現したデータ構造として**類似度テンプレート** (Similarity Template) というものを考え，この類似度テンプレートを介して類似度計算をすることとする．全ての類似度テンプレートの集合を Tとし，用例間の類似度を計算する関数 sim を，二つの用例から類似度テンプレートを得る関数 $tmpl$と類似度テンプレートから類似度を得る関数 sim_tの合成関数と考えると，以下の関係が成り立つ．

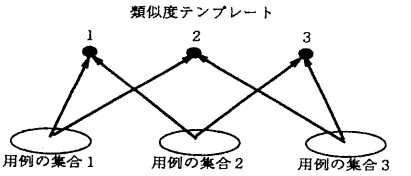

図 2: 類似度テンプレートを介した類似度計算

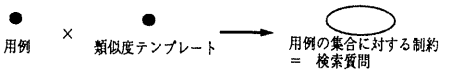

図 3: 類似度テンプレートを介した検索質問生成

$$sim = sim_t \circ tmpl$$

$$tmpl:\ E \times E \mapsto T$$

$$sim_t:\ T \mapsto R$$

類似度テンプレートを介した類似度計算の考え方を図 2 に示す．例えば，この図では，用例の集合 1 の要素と用例の集合 2 の要素の間の類似度計算の結果は，全て類似度テンプレート 1 となる．このように，類似度テンプレートは，類似度計算のあらゆるパターンをできる限り抽象化して表現したデータである．

一方，このような類似度テンプレートを用いれば，類似度計算の逆の演算として，入力用例 e_{in}と類似度テンプレート tを引数として，e_{in}との間で類似度テンプレート t の内容を満たす用例の集合に対する制約 qを返す関数 $qgen$ を考えることができる．用例の集合に対する制約 qは用例を検索する際の**検索質問** (Retrieval Query) とみなすことができるので，あらゆる検索質問の集合を Q とすると，以上の内容は以下のように表現される．

$$qgen:\ E \times T \mapsto Q$$

$$q = qgen(e_{in}, t)$$

また，制約 qを満たす用例の集合を $Eg(q)$ とすると，$Eg(q)$ 中の用例と e_{in}の間の類似度計算の結果が類似度テンプレート tとなるという関係が成り立つ．

$$\forall e \in Eg(q),\ t = tmpl(e_{in}, e)$$

類似度テンプレートを介した検索質問の生成の考え方を図 3に示す．

2.3 検索質問の包含関係

ここでは，用例集合に対して入力を中心とする包含関係を導入することにより，用例の空間を構造化してとらえる．まず，検索質問に関して，次の

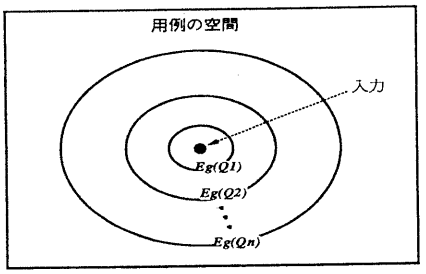

図 4: 用例集合間の包含関係

条件 2

用例の集合に対する制約 (検索質問) の間に，制約の強さに関する包含関係を定義することができ，しかもこの制約の強さに関する包含関係が検索される用例集合の包含関係に対応する．また，制約が強いほど検索される用例の類似度が大きくなる．

という条件が成り立っているとする．このとき，全類似度テンプレートの集合 Tを順序列 $T_1,\ldots,T_n$に分割し，さらに，入力 e_{in}と各部分集合 T_i中の類似度テンプレートから得られる検索質問の集合の順序列を同様に $Q_1,\ldots,Q_n$とする．

$$Q_i \;=\; \left\{q \mid \exists t \in T_i, q = qgen(e_{in}, t)\right\} \quad (i=1,\ldots,n)$$

また，現在の全用例集合 Eg中で $Q_1,\ldots,Q_n$を満たす用例の集合を同様に $Eg(Q_1),\ldots,Eg(Q_n)$ とする．ここで，上の**条件 2** を厳密に表現した次の二つの条件が成り立つように最初の類似度テンプレートの集合の分割 $T_1,\ldots,T_n$を考えておくこととする．

1. 用例集合の包含関係 — 用例集合間に，図 4に示すような以下の包含関係が成り立つ．

$$Eg(Q_1) \subseteq \cdots \subseteq Eg(Q_n)$$

2. 用例の類似度 — 内側の用例集合中の用例ほど大きい類似度を与える，すなわち，

$$Eg(Q_i) \subseteq Eg(Q_j) \;\Rightarrow\; \forall e \in Eg(Q_i), \forall e' \in Eg(Q_j),$$
$$e' \notin Eg(Q_i) \text{ ならば, } sim(e_{in}, e) \geq sim(e_{in}, e')$$

そうすれば，次節で示すように，包含関係上での二分探索を行なうことにより，類似度最大の用例の効率よい検索が可能となる．

2.4 二分探索による類似度最大の用例の検索

用例集合 $Eg(Q_1),\ldots,Eg(Q_n)$ の間に前節で述べたような包含関係および類似度の関係が成立していれば，入力との間で類似度最大となる用例は，最も内側の空集合でない用例集合中に存在することになる．そこで，ここでは，最も内側の空集合でない用例集合を，以下のような二分探索によって検索する方法を示す．

用例集合 *Eg(Q1),...,Eg(Qn)* の二分探索木

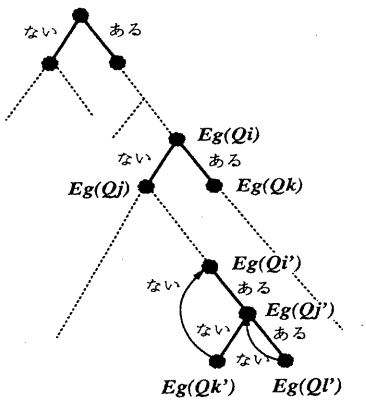

図 5: 類似度最大の用例の二分探索

まず，用例集合 $Eg(Q_1),\ldots,Eg(Q_n)$ を，図 5のような二分探索木の形で表現し，この木を以下の手順で根節点から探索していき類似度最大の用例を求める．

1. 葉節点以外の各節点 $Eg(Q_i)$では，用例集合 $Eg(Q_i)$ 中に用例が存在していなければ左分岐 $Eg(Q_j)$ をたどり，存在していれば右分岐 $Eg(Q_k)$ をたどる．

2. 葉節点 $Eg(Q_{k'})$ あるいは $Eg(Q_{l'})$ までくれば，

   (a) その葉節点に用例が存在していれば，その葉節点中に類似度最大の用例が存在する．

   (b) その葉節点に用例が存在していなければ，葉節点が図の $Eg(Q_{k'})$ の場合は $Eg(Q_{i'})$ 中に，$Eg(Q_{l'})$ の場合は $Eg(Q_{j'})$ 中に類似度最大の用例が存在する．

3. 類似度最大の用例が存在すると判定された用例集合から，類似度最大の用例の集合 $Eg_{max}(e_{in})$ を求める．

ただし，このような二分探索の際には，検索質問を満たす用例を検索するという処理がかなりの回数行なわれることになる．したがって，このような二分探索によって類似度最大の用例が高速に検索できるためには，

条件 3

一つの検索質問を満たす用例の集合が高速定数時間で検索可能である．

という条件が成り立つ必要がある．

2.5 システム構成

以上の考え方をもとに，実際の類似用例検索システムの構成を図 6に示す．システムとしては，類似度テン

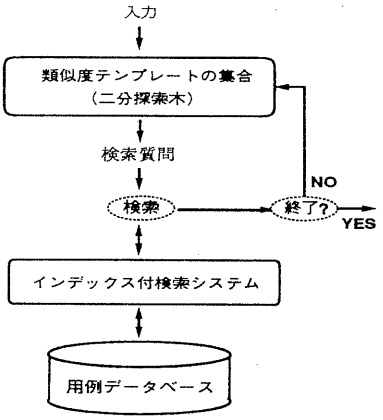

図 6: システム構成

プレートの集合の順序列 $T_1, \ldots, T_n$ の二分探索木を用意しておき，これを前節で述べた手順で二分探索していき，その都度検索質問を生成してインデックス付検索システムを介して，用例データベースから用例を検索する．インデックス付検索システムは，検索質問を満たす用例を高速定数時間で検索するためのもので，日本語表層格構造の用例検索の場合は，第 4 章で述べるようにシソーラスの構造を利用して高速検索を実現している．

3 日本語表層格構造の類似用例検索

本論文では，用例検索の例として，日本語文の表層格構造の用例の検索を扱う．表層格構造間の類似度としては，文献 [2] で用いられた類似度を修正したものを用いる．

3.1 日本語表層格構造のデータ構造

我々の類似用例検索の枠組では，類似用例は入力と同じ動詞を持つ．また，類似度計算の際には，各格要素の名詞の意味カテゴリだけが問題である．そこで，ここでは，表層格構造 e を表層格ラベル p および格要素の名詞のシソーラス中での意味カテゴリ Sem のペアの集合として定義する[1]．

$$e = \left\{ \langle p_1, Sem_1 \rangle, \ldots, \langle p_n, Sem_n \rangle \right\}$$

また，ここでは，名詞シソーラスを各節点が意味カテゴリを表わすような木構造とみなし，有向根付き木 $\langle SC, E_t \rangle$ として定義する．ここで SC は意味カテゴリの集合であり，$E_t \subseteq SC \times SC$ は有向枝の集合である．一つの名詞にはシソーラス中の一つ (あるいは複数) の葉の位置の意味カテゴリが割り当てられる．現在のところ，我々は，計算機上で利用可能な日本語シソーラスとして，「分類語彙表」[4] を用いている．

[1]本論文では，簡単化のため，名詞には多義性がなく名詞シソーラス中でただ一つの意味カテゴリを持つとしているが，実際のシステムは，名詞が多義性をもつという形で実装されている．

3.2 日本語表層格構造の類似度

日本語表層格構造間の類似度は，表層格の対応および格要素の名詞の意味カテゴリの類似度をもとに計算される．

「分類語彙表」は語を意味的に分類して 6 レベルの木構造体系にまとめたもので，各語にはそれぞれ木構造の葉の位置の意味カテゴリが割り当てられている．まず，二つの意味カテゴリ s_1，s_2 の類似度 $sim_s(s_1, s_2)$ は，共有する親節点が下位であるほど大きくなるように定める．

レベル	1	2	3	4	5	6	一致
sim_s	未定義	5	7	8	9	10	11

次に，二つの表層格構造 e_1 と e_2 の間で対応する格のペアの集合を $M(e_1, e_2)$ とする．ただし，格が対応するためには，格助詞が一致し，さらに格要素の名詞の間で意味カテゴリの類似度が定義されなければならない[2]．さらに，対応する格要素のペア m の意味カテゴリの類似度を $sim_{ps}(m)$ とすると，二つの表層格構造 e_1 と e_2 の類似度 $sim(e_1, e_2)$ が次式で定義される．

$$sim(e_1, e_2) = \sqrt{|M|} \times \frac{\sum_{m \in M} sim_{ps}(m)}{|M|} \times \sqrt{\frac{|M|}{|e_1|}} \times \sqrt{\frac{|M|}{|e_2|}}$$

$|M|$ は対応した格の数で，$|e_1|$ と $|e_2|$ はそれぞれ e_1 と e_2 の格の数である．この類似度の定義は，1) 類似度は対応する格の数が大きいほど大きくなる (第一項)，2) 類似度は対応する格の格要素の名詞の意味カテゴリの類似度が大きいほど大きくなる (第二項)，3) 類似度は各々の表層格構造において，全格要素数に対する対応した格の割合が大きいほど大きくなる (第三，四項)，という条件を満たす定義となっている[3]．

例 1:「彼が本を買う」および例 2:「彼が息子にノートを買う」の表層格構造 e_1,e_2 の類似度の計算の例を示す．まず，対応する格のペアの集合 $M(e_1, e_2)$ が求められる．さらに,「分類語彙表」では，対応する名詞の意味カテゴリの類似度は，$sim_s(Sem_{彼}, Sem_{彼}) = 11$ および $sim_s(Sem_{本}, Sem_{ノート}) = 9$ であり，$|M|, |e_1|, |e_2|$ はそれぞれ 2, 2, 3 であるから，$sim(e_1, e_2)$ は次のように求められる．

$$M(e_1, e_2) = \left\{ \langle\langle が, Sem_{彼} \rangle, \langle が, Sem_{彼} \rangle\rangle, \langle\langle を, Sem_{本} \rangle, \langle を, Sem_{ノート} \rangle\rangle \right\}$$

$$sim(e_1, e_2) = \sqrt{2} \times \frac{11+9}{2} \times \sqrt{\frac{2}{2}} \times \sqrt{\frac{2}{3}} = 11.55$$

3.3 類似度テンプレートを介した検索質問生成

3.3.1 類似度テンプレート

前節で定義した日本語表層格構造の類似度においては，2.2 節で述べた**条件 1** が満たされ，類似度計算の因子を

[2]本論文の定式化では，簡単化のため格助詞のみを扱い,「は」「も」などの係助詞は扱っていないが，実際のシステムでは，係助詞を含む例についても，格助詞との対応を考慮し適切な類似度計算を行なっている．

[3]文献 [2] では，格解析による統語的曖昧性解消の精度の観点から，条件 1) および 3) でのオーダを 1/2 乗としており，ここでもこれに従う．

抜きだして以下の3つ組からなる類似度テンプレートとして表すことができる．

$$t = \langle l_{in}, l_{db}, CS \rangle \quad (|CS| \leq l_{in}, l_{db} \leq l_{max})$$

l_{in} および l_{db} は，それぞれ，入力表層格構造および用例データベース中の用例の格の数であり，あらかじめ決められた格の数の上限 l_{max} 以下である．また，CSは，対応する格の格要素の名詞の間の類似度を要素とする多重集合である[4]．例えば，3.2節の例1および例2の表層格構造の類似度計算の場合は，類似度テンプレートは $\langle 2, 3, \{11, 9\} \rangle$ となる (ただし，例1が入力で，例2が用例データベースにあるとする．)．

3.3.2 検索質問

本論文の日本語表層格構造の類似用例検索においては，検索質問 qは，検索されるべき用例の格の数を l_{db}とし，用例が持つべき格 p とその格要素の名詞に対する意味的制約 (意味カテゴリ)Sem の組 $\langle p, Sem \rangle$ の集合を csp として，2つ組 $\langle l_{db}, csp \rangle$ で定義される．

例えば，検索対象の動詞が「買う」の場合，

$$q_1 = \left\langle 3, \left\{ \langle \text{が}, Sem_{\text{彼}} \rangle, \langle \text{を}, Sem_{\text{文房具}} \rangle \right\} \right\rangle$$

という検索質問 q_1は，「格の数が3で，「が」格の名詞が“彼”で，「を」格の名詞が“文房具”の意味カテゴリに属する」という要求に相当するので，これにより検索を行なえば「彼が息子にノートを買う」，「彼が娘に本を買う」といった用例が検索される．

3.3.3 検索質問の生成

入力表層格構造を e_{in}，類似度テンプレートを $t = \langle |e_{in}|, l_{db}, CS \rangle$ とし，これらから，検索質問 $q = \langle l_{db}, csp \rangle$ を生成する手順は以下のようになる．まず，検索される用例の格の数 l_{db}は，類似度テンプレート中の値をそのまま用いる．また，CSは，対応する格要素の名詞の類似度の多重集合であるが，実際に検索要求を生成する際には，入力のどの格がどの類似度に対応するのかを決めなければならない．そこで，CSから e_{in}の中への1対1写像 (単射) を考え，これによって CS中の類似度から e_{in} 中の格への対応を写像として規定する．

例えば，入力表層格構造 e_{in}を例1の表層格構造とし，類似度テンプレート t を $\langle 2, 3, \{11, 9\} \rangle$ とした場合は，CSから e_{in}の中への1対1写像は二つ考えられ，以下のように，二つの検索質問 q_1および q_2が生成される．(ただし，$sim_s(Sem_N, Sem_{N,x}) \geq x$)

$$\begin{aligned} e_{in} &= \left\{ \langle \text{が}, Sem_{\text{彼}} \rangle, \langle \text{を}, Sem_{\text{本}} \rangle \right\} \\ q_1 &= \langle 3, \{ \langle \text{が}, Sem_{\text{彼},11} \rangle, \langle \text{を}, Sem_{\text{本},9} \rangle \} \rangle \\ q_2 &= \langle 3, \{ \langle \text{が}, Sem_{\text{彼},9} \rangle, \langle \text{を}, Sem_{\text{本},11} \rangle \} \rangle \end{aligned}$$

[4]可能な l_{in}, l_{db}，および CSの組合せは，格の数の上限 l_{max}が与えられさえすれば，実際の入力例や用例データベースに関係なくあらかじめ数えあげておくことができる．例えば，l_{max}が3の場合は，可能な l_{in}, l_{db}，および CSの組合せの数は203である．

3.4 検索質問および類似度テンプレートの包含関係

ここでは，2.3節で述べたような用例集合の包含関係および用例の類似度の関係を可能とするために，まず検索質問の間に制約の強さに関する包含関係 $\prec_q$を定義する．検索質問に関する包含関係 $\prec_q$は，格の数に対する条件が同じである二つの検索質問 $q = \langle l_{db}, csp \rangle, q' = \langle l_{db}, csp' \rangle$ に対して，csp 中のどの格に対しても csp'中に対応する同じ格が1対1に存在し，格要素の名詞に対する意味的制約が同じかより制限が強い場合に，qが q'を包含する $(q \prec_q q')$ と定義する．厳密には，csp から csp'への1対1写像 gを考えて，以下のように定義する．

$$\begin{aligned} q \prec_q q' \iff & \exists g : csp \mapsto csp', \text{ ただし } g \text{は1対1写像}, \\ & \forall \langle p, Sem \rangle \in csp,\ g(\langle p, Sem \rangle) = \langle p, Sem' \rangle, \\ & Sem' \text{はシソーラス中で } Sem \text{と同じかより下位である}. \end{aligned}$$

例えば，次の二つの検索質問 q_1, q_2の場合，格の数 l_{db}が同じであり，q_2には「に」格に対する条件があり，また「が」格に対する条件も q_1より強いので，検索される用例に対して q_2の方が q_1よりもより強い条件を課していることになる．

$$\begin{aligned} q_1 &= \left\langle 3, \left\{ \langle \text{が}, Sem_{\text{人間}} \rangle, \langle \text{を}, Sem_{\text{文房具}} \rangle \right\} \right\rangle \\ q_2 &= \left\langle 3, \left\{ \langle \text{が}, Sem_{\text{彼}} \rangle, \langle \text{を}, Sem_{\text{文房具}} \rangle, \langle \text{に}, Sem_{\text{人間}} \rangle \right\} \right\rangle \end{aligned}$$

このとき，q_1が q_2を包含している．ここで，2.3節の**条件2**で述べたように，この包含関係 $\prec_q$は検索される用例集合の包含関係に対応する．

また，入力用例 e_{in}が与えられると，各検索質問は e_{in}と類似度テンプレートから生成されるので，入力用例 e_{in}が同じという条件のもとで類似度テンプレート間にも同様の包含関係 $\prec_t$を定義することができる．類似度テンプレートに関する包含関係 $\prec_t$は，入力およびデータベース中の用例の格の数に対する条件が同じである二つの類似度テンプレート $t = \langle l_{in}, l_{db}, CS \rangle$ および $t' = \langle l_{in}, l_{db}, CS' \rangle$ の間で，CSおよび CS'中の類似度の対応を考えることによって，以下のように定義される．

$$\begin{aligned} t \prec_t t' \iff & \exists g : CS \mapsto CS', \text{ ただし } g \text{は1対1写像}, \\ & \forall sim_s \in CS,\ g(sim_s) \geq sim_s \end{aligned}$$

例えば，二つの類似度テンプレート $\langle 2, 3, \{9, 9\} \rangle$ と $\langle 2, 3, \{8, 9, 11\} \rangle$ との間の検索の条件の強弱の関係は，

$$\langle 2, 3, \{9, 9\} \rangle \quad \prec_t \quad \langle 2, 3, \{8, 9, 11\} \rangle$$

となる．この場合，前者における格要素の名詞の類似度を表わす二つの9は，後者では9および11に対応する．

また，入力用例を e_{in}とし，$q = qgen(e_{in}, t)$，$q' = qgen(e_{in}, t')$ とし，さらに各検索質問を満たす用例の集合をそれぞれ $Eg(q), Eg(q')$ とすると，$\prec_t$，$\prec_q$，用例集合の包含関係の3つの間には以下の関係が成り立つ．

$$t \prec_t t' \quad \Rightarrow \quad q \prec_q q' \quad \Rightarrow \quad Eg(q) \supseteq Eg(q') \qquad (1)$$

3.5 類似度テンプレートの集合の分割

前節で述べた類似度テンプレートの包含関係を用いて，次の二つの条件を満たすように全類似度テンプレートの集合を順序列 $T_1, \ldots, T_n$に分割すれば，2.3節で述べた

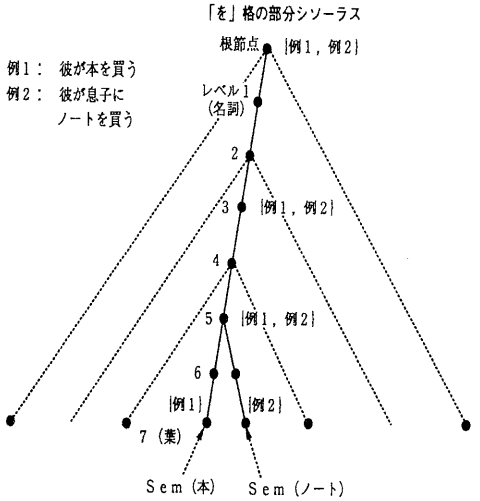

図 7: 「を」格の部分シソーラスの例

用例集合の包含関係および用例の類似度に関する条件を満たすことができる．

1. 類似度テンプレートの包含関係 — $i<j$として，T_i中のどのt_iに対しても，$t_j \prec_t t_i$となるt_jがT_j中に存在する．また，逆に，T_j中のどのt_jに対しても，$t_j \prec_t t_i$となるt_iがT_j中に存在する．

2. 類似度テンプレートの類似度 — 2.2 節で述べた類似度テンプレートの類似度sim_tを考えると，$i<j$として，T_i中の任意の類似度テンプレートt_iとT_j中の任意の類似度テンプレートt_jの間に以下の関係が成り立つ．

$$sim_t(t_i) > sim_t(t_j)$$

この類似度テンプレートの包含関係に関する条件から，前節の式 1の関係を用いることにより，2.3 節の用例の包含関係に関する条件が導かれる．また，類似度テンプレートの類似度に関する条件から，2.3 節の用例の類似度に関する条件が導かれる．

4 部分シソーラスを用いた高速検索

2.4 節で述べた**条件 3** より，実際のシステム構成では検索質問を満たす用例を高速定数時間で検索するためのインデックス付検索システムが必要である．本論文の日本語表層格構造の検索においては，検索質問$q=\langle l_{db}, csp\rangle$を満たす用例の検索は，$csp$ 中のそれぞれの格の制約を満たす用例を高速に検索し，検索結果の交わり集合をとることにより行なわれる．それぞれの格の制約は，「用例が表層格ラベル p でマークされる格を持ち，格要素の名詞がシソーラス中の意味的制約 Sem を満たす」という要求に相当するが，この要求を満たす用例の集合は，全用例の集合からあらかじめ作っておくことができる．これらの用例集合を表層格ラベル毎にまとめ用例集

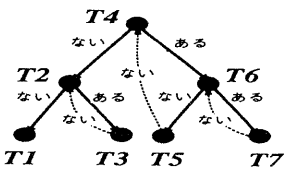

図 8: 類似度テンプレートの集合の二分探索木の例

合が空集合となるものを除くと，名詞シソーラスの部分構造とみなすことができるので，これを**部分シソーラス** (Sub-Thesaurus) と呼ぶ．図 7に，3.2節の例 1 および例 2 が用例データベースにある場合の「を」格の部分シソーラスを示す．この場合，“本” の意味カテゴリと “ノート” の意味カテゴリの共通の親節点が 5 レベル目であるので，6 レベル目および 7 レベル目 (葉) の用例の集合は { 例 1} あるいは { 例 2} となっており，5 レベル目から上の節点では用例の集合が { 例 1, 例 2} となる．

部分シソーラスを用いて制約 $\langle p, Sem\rangle$ を満たす用例を検索するには，表層格ラベル p についての部分シソーラスを根節点からたどっていき，意味カテゴリ Sem を持つ節点を探索すればよい．この探索は，シソーラスの木構造をそのまま利用すれば定数時間で高速に行なえる．

5 日本語表層格構造の用例検索の例

ここでは，「買う」という動詞の表層格構造の用例検索の例を簡略化して示す．類似度テンプレートの集合の二分探索木として図 8のようなものがあるとし，入力が$e_{in}=\{\langle$ が$, Sem_{彼}\rangle, \langle$ を$, Sem_{本}\rangle\}$ であるとする．このとき，T_4, T_5, T_6, T_7に注目して類似度テンプレートとして$l_{db}=2$ であるものだけを考え，検索質問を生成した結果についてみると以下のようになる．

$$
\begin{aligned}
T_4 &= \{\langle 2,2,\{8,11\}\rangle\} \longrightarrow \\
Q_4 &= \{\langle 2, \{\langle \text{が}, Sem_{彼,8}\rangle, \langle \text{を}, Sem_{本,11}\rangle\}\rangle, \\
&\quad \langle 2, \{\langle \text{が}, Sem_{彼,11}\rangle, \langle \text{を}, Sem_{本,8}\rangle\}\rangle\} \\
T_5 &= \{\langle 2,2,\{9,11\}\rangle\} \longrightarrow \\
Q_5 &= \{\langle 2, \{\langle \text{が}, Sem_{彼,9}\rangle, \langle \text{を}, Sem_{本,11}\rangle\}\rangle, \\
&\quad \langle 2, \{\langle \text{が}, Sem_{彼,11}\rangle, \langle \text{を}, Sem_{本,9}\rangle\}\rangle\} \\
T_6 &= \{\langle 2,2,\{10,11\}\rangle\} \longrightarrow \\
Q_6 &= \{\langle 2, \{\langle \text{が}, Sem_{彼,10}\rangle, \langle \text{を}, Sem_{本,11}\rangle\}\rangle, \\
&\quad \langle 2, \{\langle \text{が}, Sem_{彼,11}\rangle, \langle \text{を}, Sem_{本,10}\rangle\}\rangle\} \\
T_7 &= \{\langle 2,2,\{11,11\}\rangle\} \longrightarrow \\
Q_7 &= \{\langle 2, \{\langle \text{が}, Sem_{彼,11}\rangle, \langle \text{を}, Sem_{本,11}\rangle\}\rangle\}
\end{aligned}
$$

これから．検索される用例集合の間に

$$Eg(Q_4) \supseteq Eg(Q_5) \supseteq Eg(Q_6) \supseteq Eg(Q_7)$$

という包含関係が成り立つことがわかる．また，$Eg(Q_6)$ に類似度最大の用例$e_{max}=\{\langle$ が$, Sem_{彼女}\rangle, \langle$ を$, Sem_{本}\rangle\}$ が存在しているとすると，$T_4 \rightarrow T_6 \rightarrow T_7 \rightarrow T_6$という順で探索が行なわれ，$Eg(Q_6)$ から e_{max}が検索される[5]．

[5]類似度テンプレートの包含関係と 3.4 節の式 1の関係から，用

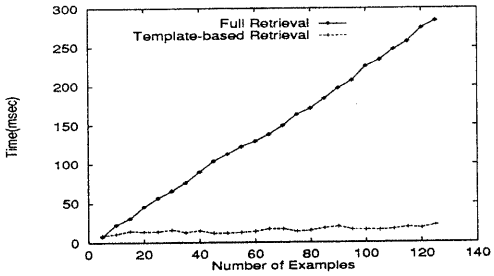

図 9: 検索速度/用例数の比較

6 評価

システムの主要な構成要素は用例データベース・類似度テンプレートの集合の二分探索木・部分シソーラスの三つであるが，このうち類似度テンプレートの集合の二分探索木は用例数に依存せず一定のサイズであり[6]，全部分シソーラスのサイズは O(用例数) である[7]．また，類似度テンプレートによる検索 (Template-based Retrieval) プログラムについて用例数に対する検索時間 (CPU 時間) の変化を測定し，これを全用例検索 (Full Retrieval) と比較したものを図 9に示す．実験は，用例データベース中に動詞「買う」の用例を入れておき，動詞「買う」を持つ入力表層格構造との間で類似度が最大となる用例を検索するという形で行なった．類似度テンプレートによる検索プログラムにおいては，格の数の上限 l_{max} を 3 とした．全用例検索の方が，用例数に比例して検索時間が増加するのに対して，類似度テンプレートを用いたプログラムの方は検索時間がほぼ一定となった．

7 おわりに

本論文では，用例に基づく自然言語処理において，全用例検索を行なわずに用例データベースから類似用例を効率よく検索する方法として，用例間の類似度を用いて用例の空間を構造化し，その構造を利用して類似用例を高速に検索する手法について述べた．具体的には，類似度計算のあらゆるパターンを抽象化して表現した類似度テンプレートというデータ構造を考え，類似度テンプレートから検索質問を生成して類似用例を検索するという方法をとった．実験を行なった結果，用例数の増加に対して，検索時間がほぼ一定となった．

用例に基づく自然言語処理における他の類似度の例としては，機械翻訳における動詞の訳語選択を扱ったもの [5] や日本語の「A の B」というフレーズの英語への訳し分けを扱ったもの [7] における類似度がある．これらの例でも，本論文で述べた表層格構造の場合と同様に，用例がいくつかのスロットからなり各スロットの要素の類似度がシソーラスに基づいて計算されるか，あるいは用例集合から計算される．本論文で述べた類似度と異なる点としては，各スロットに重みを許している点が挙げられる．この重みは用例集合が与えられた時点で計算され，用例検索中は一定の値をとるので，本論文の類似度テンプレートの考え方がそのまま適用できる[8] [9]．

類似度の定義に関連して，類似度や類似度を決める要因となる各因子がとり得る値が離散値でなく連続値の場合には，連続値を適当な数の離散値に変換することによって，類似度計算の結果を有限個のパターンに帰着することが可能となると考えられる．また，用例集合が与えられた段階で，用例の分布を調べた結果，大きい類似度の部分の分布が密であるとか，逆に小さい類似度の部分の分布が密であるというような分布に関する傾向が得られれば，その傾向にあわせて二分探索の開始点を変えることも可能であると考えられる．

例が検索されなかった類似度テンプレートに包含される類似度テンプレートからは用例が検索されないことがわかるので，実際の探索においては枝刈をして無駄を省くことができる．

[6] l_{max} が 3 の場合，順序列 $T_1,\ldots,T_n$ の長さは，l_{in} が 1 のものに対して 7，2 のとき 9，3 のとき 11 である．

[7] ある用例中の一つの名詞は，その名詞は部分シソーラス中のある一つの葉の全祖先節点に含まれるので，シソーラスの深さ d 回だけある部分シソーラスに含まれる．また，一つの用例に含まれる格要素の名詞の数は，高々，格の数の上限 l_{max} 個であるので，全用例に含まれる名詞の数は，のべ数で高々，(用例数)$\times l_{max}$ である．これより，全部分シソーラスに含まれる用例ののべ数は (用例数)$\times l_{max} \times d$ である．

[8] 動詞の訳語選択の場合は，動詞毎に各スロットの重みが一定であるので，類似度テンプレートの集合の順序列を二分探索木にしたものは，動詞毎に異なってくると考えられる．

[9] 各スロットの要素の類似度が用例集合から計算される場合も，スロットの要素の集合を階層化してとらえることにより，シソーラスと同等の扱い方が可能になると考えられる．

参考文献

[1] 北野宏明. 超並列人工知能. 人工知能学会誌, Vol. 7, No. 2, pp. 244–262, 1992.

[2] 黒橋禎夫, 長尾眞. 格構造解析への評価関数の導入による統語的曖昧性の解消. 情報処理学会研究報告, Vol. 92, No. 93 (92–NL–92), pp. 65–72, 1992.

[3] M. Nagao. A framework of a mechanical translation between Japanese and English by analogy principle. In A. Elithorn and R. Banerji, editors, *Artificial and Human Intelligence*. Elsevier Science Publishers, B.V, 1984.

[4] 国立国語研究所. 分類語彙表. 秀英出版, 1964.

[5] 佐藤理史. MBT1: 実例に基づく訳語選択. 人工知能学会誌, Vol. 6, No. 4, pp. 592–600, 1991.

[6] C. Stanfill and D. Waltz. Toward memory-based reasoning. *Communications of the ACM*, Vol. 29, No. 12, pp. 1213–1228, 1986.

[7] E. Sumita, K. Oi, O. Furuse, H. Iida, T. Higuchi, N. Takahashi, and H. Kitano. Example-based machine translation on massively parallel processors. In *Proceedings of the 13th IJCAI*, pp. 1283–1288, 1993.

[8] 苫米地英人. 超並列自然言語処理. 情報処理, Vol. 33, No. 7, pp. 768–779, 1992.