

事例の一般化による機械翻訳

野 美 山 浩†

規則に基づく機械翻訳システム (Rule-Based Machine Translation) では、規則数が膨大になった場合の全体の制御の記述の困難性や特定の分野への適合性が低いなどの問題点が指摘されている。事例に基づく機械翻訳 (Example-Based Machine Translation) は、これらの問題点を解決する有望な方法として提案されているが、事例をそのまま翻訳に用いる方法は、実用的なシステムを構築する上で、まだ様々な問題を抱えている。本稿では、EBMT と同様の知識獲得容易性を持ち、かつ、シソーラスを用いて事例ベースから一般化された概念を学習することによって探索空間の小さい効率的な実行を可能とする新しい翻訳機構を提案する。本手法は、事例を一般化することによって、人が記述する規則と事例を融合し、処理の上では同等に扱うことによって、効率的処理を実現する。

Machine Translation by Case Generalization

HIROSHI NOMIYAMA†

Example-based machine translation is a promising approach to resolving problems in rule-based machine translation systems, such as difficulties in control of rules and low adaptability to specific domains. We propose a new mechanism for machine translation, in which effective concepts for translation are learned from a huge set of cases. As the result of the learning process, a huge set of cases is generalized into a smaller set of cases. And by dealing with rules in the same way as generalized cases, it makes possible to acquire knowledge effectively.

1. はじめに

機械翻訳システムにおける知識獲得の困難性を解消するために、既存の翻訳事例を模倣することによって翻訳する手法、Example-Based Machine Translation (EBMT)、が提案されている^{1),3)-5),7),8)}。この手法は、事例ベース中で入力文と類似したものを用いて翻訳を行うという単純な原理に基づく。

EBMT は以下に示す性質を持つ。

- 実際のユーザが扱っている情報を知識として採り入れることができる
 - 何が良い結果を出すために有効であるかをあらかじめ考える必要がない
 - 個々の知識 (規則) を宣言的に取り扱う
- これらの性質により EBMT は、翻訳の知識の獲得を容易にするが、その実現においては、まだいくつかの欠点を持つ。
- 膨大な計算量
- 入力と類似した事例を検索するために、非常に多

くの事例を検索し、かつ、類似度を計算しなければならない*。

● 特異点およびノイズに対する弱さ²⁾

特異点とは、「特定の単語が用いられた場合にのみ、他の非常に類似した単語が用いられた場合と異なった訳となる」場合を指す。自然言語においてよく見られる現象である慣用表現はこの特異点に相当する。例えば、「頭が切れる」は、次の2つの訳し分けがあるとす。1つは、一般的な意味で、“hurt one’s head”であり、もう1つは、慣用的な意味で、“be smart”という訳である。しかし、“頭部が切れる”を考えた場合、“頭部”と“頭”は、非常に類似しているにもかかわらず、この文は慣用的意味を持たず、“hurt one’s head”としか訳されない。EBMT においては、“頭部が切れる”の正しい訳を得るためには、正しい例を事例ベースに加えればよい。しかし、単語Xが意味的に“頭”に近い場合、“Xが切れる”に対して慣用的解釈が排除されることは、依然として保証

† 日本アイ・ビー・エム株式会社東京基礎研究所
IBM Research, Tokyo Research Laboratory

* 実用的なシステムでは、最低でも、数十万の事例が必要であろう。

されない。

また、同様の問題として、目的言語側で非常に特殊な言い回しが使われている事例（例えば文脈や文化的相違を考慮して非常に練られた訳文、あるいは、誤った訳文など）が、他の類似した事例と同じ基準で選択されるという問題がある。

これらの問題は、連続した意味空間から類似性を計算することに起因している。連続な空間を仮定すると、特異点のすべての近傍に対して事例を持たない限り特異点の影響が他の類似した入力に及ぶことを妨げることができない。

●単一の意味空間

1つの意味空間であらゆるタイプの翻訳に十分かどうかという問題がある。例として、“食べる” (eat, feed, …) を翻訳する場合を考えてみよう。“犬が食べる”は、“a dog eats.”、“牛が食べる”は、“a cow feeds.”、“馬が食べる”は、“a horse feeds.”と翻訳される。翻訳結果から考えると、“牛”と“馬”の場合、同じ訳語が選択されているので、“牛”は“犬”より“馬”に類似してはいない。次に、“走る” (run, gallop, …) を翻訳する場合を考えてみる。この場合は、“犬が走る”は、“a dog runs.”、“牛が走る”は、“a cow runs.”、“馬が走る”は、“a horse gallops.”に翻訳される。この場合は、“犬”と“牛”に同じ訳語が選択されているので、“牛”は“犬”より“馬”に類似してはいない。これは、最初の例と矛盾する。このように、厳密にはすべての翻訳に対し平等に成り立つような意味空間は存在しない。

●事例ベースの知識源としての有用性・効率性

翻訳の規則性を否定し、すべての知識を事例から得ることは、必ずしも効率的ではない。規則性が成り立っているような翻訳に対しては、規則は非常に効率的な知識である。また、あらかじめ実用的な翻訳に十分な量の事例を準備できるかという問題もある。

本稿では、佐藤⁶⁾および隅田⁷⁾で議論されている固定された文型における翻訳の問題について論じる。このため、翻訳全体として適用できる範囲は限られているが、文型を固定したとしても、従来提案されているEBMTは効率的の実現手法を明らかにしていない。

2. 事例の一般化による機械翻訳

事例の集合（事例ベース）は、規則の集合と異なり本質的に冗長である。複数の同じ原言語の入力が全く同じ訳を生成するのであれば、その事制は1つあれば十分である。さらに、1つの同じ概念に属する語が翻訳の際に同じ結果を与えるならば、それらの語を、より抽象度の高い概念に置き換えることができる。事例の収集において、事前の選別を行わなければ、事例ベースは冗長な情報を含む。しかし、事例を一般化することによって、膨大な事例の集合を小さな一般化された事例の集合にすることが可能となる。

語あるいは概念の一般化のパスは、シソーラスによって規定することができる。一般化は、事例中の語をより一般的な概念に置き換えることにより行う。語が概念で置き換えられるかどうかは、事例ベース中の頻度を考慮しなくてはならない。また、事例の頻度は、一般化された事例の重み付けに用い、頻度の高い事例は、頻度の低い語より高く重み付けする必要がある。

一般化の過程では、例外を一般化されないものとして認識することが可能である。いったん例外と認識してしまえば、それが一般的に解釈されることを抑止することができる。

一般化事例に加え、規則も事例と同様の形式で記述される。規則は一般化の性質を考慮して2つに分ける必要がある。

例外規則 ある概念内でのみ成立するが、その上位概念では成立しない。この規則に対応する事例は、成立する概念を越えて一般化を行わない。

制限規則 ある概念内で成立し、その上位概念でも成立する。この規則に対応する事例は一般化を行う。

一般化された事例は、人が記述した規則と融合することによって知識源としての有効性を失うことなく、効率的に実行可能となる。事例だけでなく、規則も組み込めるということは、事例ベースが十分な事例を含んでいない場合、特に有効である。もし、規則が組み込めないならば、システムを使用する当初から十分な事例を必要とする。一般的な規則が用意できれば、システムを使用しながら漸進的に事例を蓄積していくことが可能となる。

以上の考えに基づき、事例を一般化することによって、効率的に機械翻訳を実現する手法を提案する。

["3月"(March), "こわす"(destroy)]]→["in"]
["4月"(April), "議論する"(discuss)]]→["in"]
["5月"(May), "採決する"(vote)]]→["in"]
["6月"(June), "冷える"(cool)]]→["in"]
["月曜"(Monday), "洗う"(wash)]]→["on"]
["火曜"(Tuesday), "決まる"(decide)]]→["on"]
["週末"(weekend), "上がる"(raise)]]→["on"]
["東"(east), "移る"(move)]]→["to"]
["東京"(Tokyo), "移動する"(move)]]→["to"]
["7月"(July), "移動する"(move)]]→["in"]

図1 「名詞+“に”+動詞」の翻訳
Fig. 1 Translation of "Noun+NI+Verb".

3. 事例の一般化

3.1 線形翻訳事例

本手法では、事例を以下のように定義する。これを線形翻訳事例 $LTPC$ と呼ぶ。

$LTPC: L_s \rightarrow L_t$

L_s : 原言語の語彙変数項の値の集合

L_t : 対象言語の変数項の値の集合

$LTPC$ の右辺を結果値と呼ぶ。また、 L_s 中の変数項の数を項数 (M) と呼ぶ。

図1に日本語の“に”格の訳し分けにおける $LTPC$ の例を示す。“に”格がとる名詞と、それが掛かる動詞によって英語の前置詞が決まるという翻訳を仮定している。

次に、翻訳規則 (TR) を定義する。 TR は、 $LTPC$ と同じ形式で定義される。 TR とは、 $LTPC$ の L_s の変数の値として概念あるいは語を記述したものである。翻訳規則は、その概念のみで生じる例外を記述した例外規則と、その概念における解釈の制限を記述した制限規則に分けられる。

“に”格の訳し分けにおける規則の例を示す。〈Time〉の下位概念に〈Week〉があるとすると、〈Time〉では、一般的に["in"]となり、〈Week〉では、〈Time〉とは異なって["on"]となるとすると、制限規則として [〈Time〉, 〈〉]→["in"], 例外規則として [〈Week〉, 〈〉]→["on"] のように定義できる。

このように定義される $LTPC$ において、 L_s 中の語をより抽象度の高い概念に置き換え、全体に重要度を付与することで一般化線形翻訳事例 ($GLTPC$) を求める。

3.2 シソーラスによる事例の一般化

3.2.1 j 項部分シソーラスの生成

$LTPC$ を一般化するために、まず、与えられるシソーラス (図2参照) から、 $LTPC$ の各項 j に対し、 L_s 中に出現する語を含む部分シソーラス、 $PTH(j)$

($1 \leq j \leq M$)、を生成する。そして、各々の $PTH(j)$ において L_s 中に出現する語のノードを生成し、結果値とその頻度の対を設定する。

ここで、 $LTPC$ の重要度を計算するためにいくつかの補助的な重要度を定義する。なお、 $LTPC$ の総数を T とする。

まず、一般化事例の重み付けをするために用いられる重要度の定義を行う。

リンクの重要度 (IL) $PTH(j)$ 中のリンクの重要度は、そのリンクに直接繋がっている子ノードをルートとする部分木を含む事例の、全翻訳パターン事例に対する生起確率と定義する。つまり、そのような部分木を含む事例の総数を S とすると、 $IL = \frac{S}{T}$ で表される。

ノードの重要度 (IN) ノードが持つ結果値の揺れが少ないほど、そのノードが表現する概念は安定しており、そのノードは重要であるとする。 $PTH(j)$ 中のノードの重要度は、そのノードをルートとする部分木中の結果値の生起確率を p_i とすると、 $IN = \sqrt{\sum p_i^2}$ と定義する*。 IN は、 $0 \leq IN \leq 1$ となる。例えば、あるノード〈Time〉において、["in"] となる確率が $\frac{5}{8}$,

["on"] となる確率が $\frac{3}{8}$ 、であるとすると、 IN の値

は、 $\sqrt{\left(\frac{5}{8}\right)^2 + \left(\frac{3}{8}\right)^2} = 0.729$ となる (図3参照)。

結果値の重要度 (IV) $PTH(j)$ 中のノード k における結果値 L の重要度 (IV_{kL}) は、ノード k の子ノードを m 、それらのノード中で結果値 L の重要度を IV_{mL} とすると、 $IV_{kL} = IN_{kL} \sum_m (IL_m \times IV_{mL})$ と定義する。

一般化事例の重要度 (IC) $GLTPC$ の重要度は、 L_s 中の語または概念の PTH 中の同じ結果値 L の重要度を IV_{jL} とすると $IC = \sum_{j=1}^M IV_{jL}$ と定義する。

これらの定義に基づき、最初に、 $PTH(j)$ 中のすべての IL 、すべての IN 、および、概念リーフノード (シソーラス中の末端の概念ノード) におけるすべての IV を計算する。

3.2.2 概念リーフノードの分割

概念リーフノードにおいて、 IV が最大でない結果値に着目し、その IV が所定の閾値よりも大きく、かつ、細分化によって事例の数が減少するならば、そのノードの細分化を行う。細分化は、着目した結果値を持つ語ノードを子ノードに持つ新たなノードを生成することによって行われる。概念リーフノードが細分化

* この式は、Stanfill¹⁾とSumita²⁾のものと同様である。

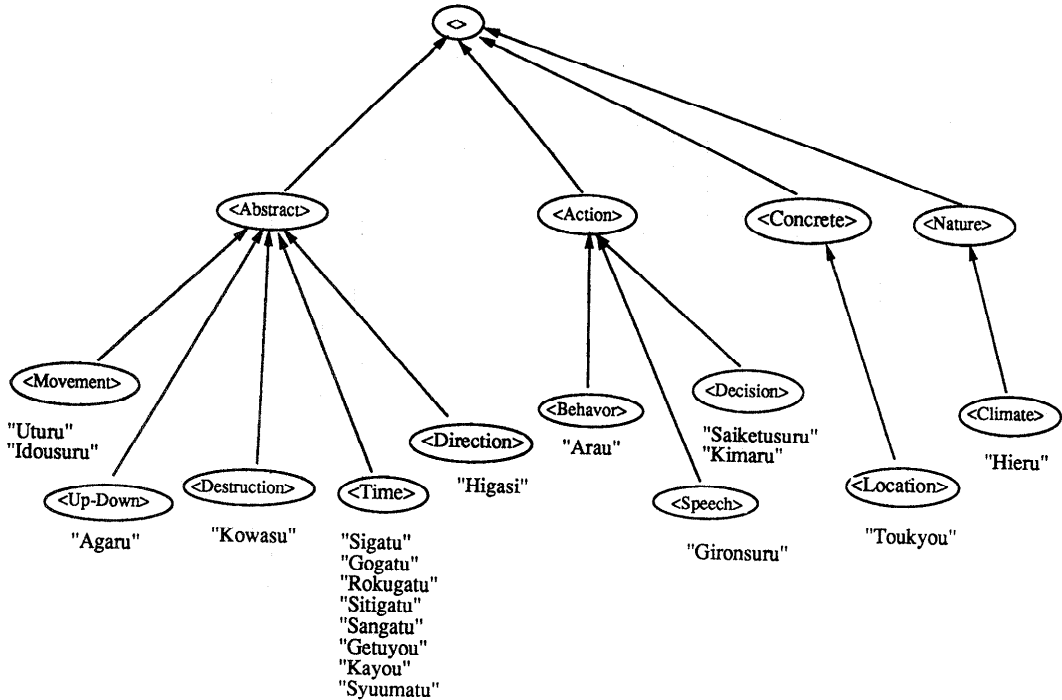


図 2 シソーラス
Fig. 2 Thesaurus.

されるのは、多くの場合、特定の翻訳パターンに個別的に有効な分類が存在することによる。これらの部分的分類のみを翻訳パターン・シソーラス (TPTH) としてシソーラスとは別個に保持する*。

3.2.3 結果値の重要度の伝搬

次に、結果値の重要度を伝搬させることによって、概念リーフノード以外のノードにおける IV を計算する。IV の伝搬は、まず概念リーフノードから始めて上向きに行く。上向きの伝搬の後、下向きの伝搬を行う。下向きの伝搬は、下位ノードが概念ノードであり、かつ、伝搬された値が下位ノードの IV の最大値以上である場合にのみ行う。

下向きの伝搬は、例外的な事例が上位ノードにまで影響することを防ぐ。例えば、あるシソーラス中の部分木において1つしか事例がない場合、それが全体としてみると非常に例外的なものであったとしても、上向きの伝搬のみでは、その部分木のルートノードまで一般化されてしまう。

* 対応する事例の数を r 個とすると、この分割によって、 r 個の語に対する新たに生成された概念との対応、および、その概念と上位概念との対応を加える必要があるが、事例の数は r 個から 1 個になる。

重要度の計算例を図3と図4に示す。図3は、第1項に対するもので、図4は第2項に対するものである。図3においては、<Time> において概念ノードが分割され、新しいノード <*X* > が生成されている。また、ノード <Concrete> において下向きの伝搬が生じている。これによって、ノード <Concrete> において “in” は、“to” よりも重要度が高くなっている。

3.2.4 LTPC の項別一般化

以上のようにして計算された重要度に基づき、LTPC をまず項別に一般化する。下位ノードで最大の重要度を持つ結果値がその上位ノードで最大の重要度を持つ結果値と同じならば、その項は上位ノードに一般化される。つまり、その項が事例中でとった値(語)をより抽象度の高い概念に置き換える。項別一般化は、その項の PTH(j) 中の末端のノード(語)から上向きに一般化を可能な限り繰り返すことにより行い、最も一般化された項のみを残す。結果として得られた LTPC のうち、重複する LTPC は、1 つだけを採用する。

事例 [“火曜”, “決まる”] → [“on”] を用いて、項別一般化を説明する。最初に、第1項の “火曜” が一般化される。この事例の値 [“on”] は、親ノード <*X* >

で最も高い IV を持つ値と同一であるので (図3参照)，“火曜”は、<X*> に一般化される。さらに上位のノードにおいては、値 [“on”] は最も高い IV を持つ値と同一でないので、これ以上一般化されない。次に、第2項“決まる”を一般化する。“決まる”の上位概念 <Decision> においては、値 [“on”] は最も高い IV ではないので一般化されない。最終的に [<X*>]、

“決まる”]→[“on”] が、項別一般化の結果として得られる。

3.2.5 LTPC の項間一般化

次に項間の一般化を行う。項間一般化は、IV でなく、IC を考慮して一般化を行う。M=1 の場合は、項別一般化事例に IC を付与したものが GLTPC となる。M>1 の場合は、得られた項別一般化事例の各々に対し、第1項から順に第k項最大一般化 (1 ≤ k ≤ M) を行うことにより、GLTPC を得る。第k項最大一般化は、まず第k項以外の項は固定した状態で、第k項をそのノードからシソーラス中の上位に向かって最大限に一般化する。次に残りの M-1 個の項を順に最大一般化を行う。一般に、M>1 ならば、項間一般化によって得られる GLTPC は、延べ M × (M-1) 個存在する。ただし、得られた GLTPC のうち、重複する GLTPC は1つだけを採用する。

[<Direction>]、[<Abstract>] → [“to”] の例を用いて、項間一般化を説明する。最初に、第1項の一般化を行う。ノード <Abstract> (<Direction> の親ノード) における IV の値を以下に示す (図3参照)。

- [“in”] : 0.192
- [“on”] : 0.035
- [“to”] : 0.007

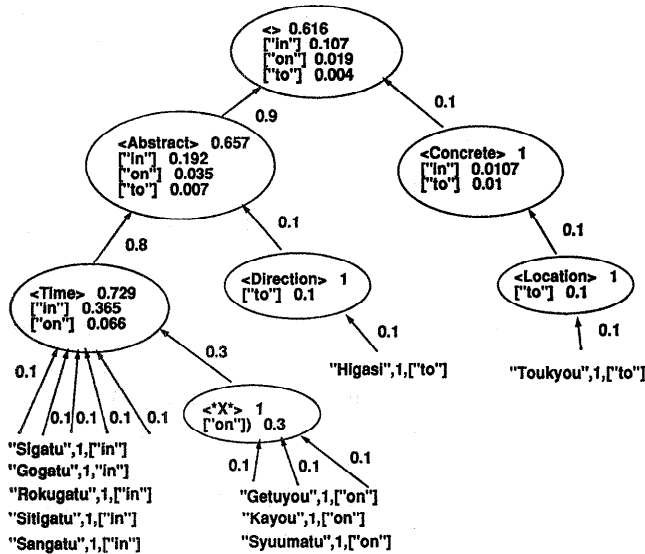


図3 第1項部分シソーラス
Fig. 3 First-Term partial thesaurus.

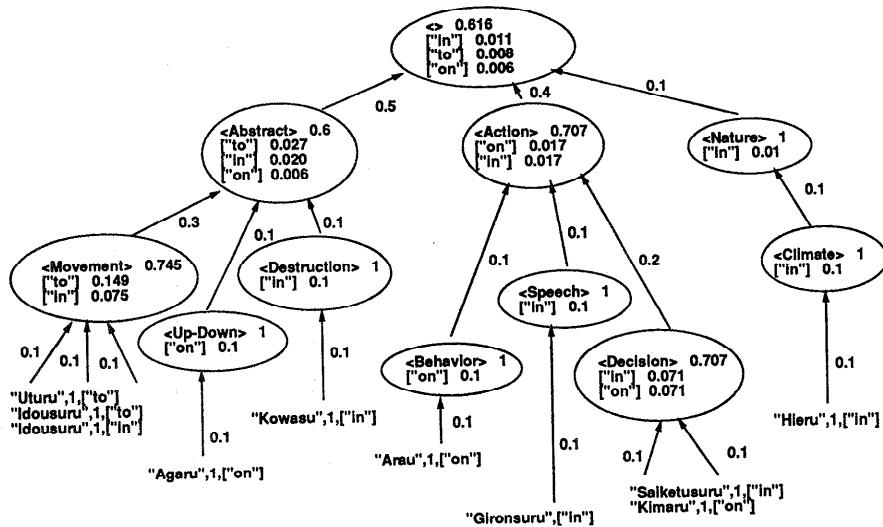


図4 第2項部分シソーラス
Fig. 4 Second-Term partial thesaurus.

ノード〈Abstract〉における IV の値を以下に示す (図4参照).

["to"]: 0.027

["in"]: 0.020

["on"]: 0.006

IC は、それぞれの対応する値の合計となる.

["to"]: $0.007+0.027=0.034$

["in"]: $0.192+0.020=0.212$

["on"]: $0.035+0.006=0.041$

ここで, ["to"] は, 最も高い重要度を持たないので, 第1項はこれ以上一般化されない.

次に, 第2項を一般化する. ノード〈Direction〉における IV の値を以下に示す (図3参照).

["to"]: 0.1

〈Abstract〉の親ノード〈〉の IV を以下に示す (図4参照).

["in"]: 0.011

["to"]: 0.008

["on"]: 0.006

IC は、それぞれの対応する値の合計となる.

["to"]: $0.1+0.008=0.108$

["in"]: $0+0.011=0.011$

["on"]: $0+0.006=0.006$

値 ["to"] は、重要度が最大であるので、第2項はルートノードにまで一般化される。これ以上、一般化はできないので、ここで終了する。よって、最終的に [〈Direction〉, 〈〉] → ["to"] 0.108 が第1項最大一般化事例の結果となる。

図1中のすべての事例に対する一般化の結果を図5に示す。

3.2.6 翻訳規則と一般化事例の統合

最後に, TR を $GLTPC$ と統合する.

制限規則については、以下の手順に従う。 $GLTPC$ 中にすでに TR と全く同一のものがあれば、それは追加しない。 TR と結果値のみ違う $GLTPC$ が存在すれば、それを TR で置き換える。それ以外の場合は、 TR の重要度を計算し、 $GLTPC$ に加える。 TR の重要度は、一般化事例の場合と同じ部分ソーラス

[〈>,〈>] → ["in"] 0.118
[〈*X*〉,〈>] → ["on"] 0.306
[〈Concrete〉,〈Abstract〉] → ["to"] 0.037
[〈Location〉,〈>] → ["to"] 0.108
[〈Direction〉,〈>] → ["to"] 0.108

図5 一般化の結果

Fig. 5 Result of generalization.

を用いて同様に計算される。

例外規則は、全く別の部分空間として前述した一般化を個々の例外規則ごとに行う。最後に個々の例外規則に対して得られた $GLTPC$ を全体の $GLTPC$ に加える。

このように、本手法では翻訳事例と人間が記述した抽象的な規則を融合し、後述する最適照合では同様のものとして取り扱う。一般性の乏しい規則を追加しても、その規則には低い重要度しか与えられず、翻訳処理において適用される尤度が環境に応じて適切に与えられる。

4. 最適照合処理

最適照合処理は、入力 L と最も近い事例を $GLTPC$ 中から求める。 $M=1$ の場合は、 L 中の変数にあてはまる入力中の語自身またはその上位概念であり、かつ、 $GLTPC$ に変数の値として含まれているものを、 $TPTH$ およびソーラス中を上向きに探索し、最初に見つかった語または概念を含む $GLTPC$ が、最短距離一般化翻訳事例 ($SDGLTPC$) である。

$M>1$ の場合は、 j 項 ($1 \leq j \leq M$) について第 j 項最短距離翻訳事例 ($SDGLTPC_j$) を求め、その中で最も重要度の大きなものを $SDGLTPC$ として選択する。 $SDGLTPC$ はすべての項について入力の一般化パス上になくなくてはならない。そのため、 $k-1$ 項 ($k>1$) において、その一般化パス上にある事例を探索しておけば、 k 項では、その事例の中から探索すればよい。

$M=2$ の場合、 $SDGLTPC_1$ は、第1項に関し最短距離の語または概念を含む事例である。このようにして $SDGLTPC_1$ と $SDGLTPC_2$ を求め、重要度の大きい方を $SDGLTPC$ として選択する。一般に、 M ($M>1$) に対し、第1項から第 M 項までの最短距離事例として、最悪の場合、延べ $M \times (M-1)$ 個の $SDGLTPC_j$ が求まる。その中で、最も重要度の高いものを $SDGLTPC$ として選択する。

例外規則に対応する部分空間は、全く別に一般化され全体の空間に対する $GLTPC$ に加えられるが、最適照合の際、例外規則に一致すべき事例は、例外規則に対応する部分空間を越えて一致することはない。

最適照合の例として、“月曜に降る”に最も類似した事例を検索する例を示す。ここで、“降る”は、〈Climate〉の上位概念とする。まず、第1項最短距離翻訳事例が一般化翻訳事例 (図5参照) 中から探索される。“月曜”は、一般化翻訳事例のどの第1項にも

存在しないので、その親ノードである $\langle *X* \rangle$ を第1項として持つ事例が探索される。その結果、 $[\langle *X* \rangle, \langle \rangle] \rightarrow [\text{"on"}]$ 0.306 が見つかる。この事例の第2項は、“降る”の上位概念であるので、これが、入力に対する第1項最短距離事例となる。次に、第2項最短距離事例が探索されるが、これも同じ事例が得られる。結果として、最も類似した事例は、 $[\langle *X* \rangle, \langle \rangle] \rightarrow [\text{"on"}]$ 0.306 となる。

5. 議 論

5.1 効 率 性

本手法は、以下に示す点で探索空間を狭くすることができる。

- 事例ベースの冗長性を排除する
同じ結果を与えるという観点から一般化することにより処理の冗長性を減らす。
- 探索空間を一般化パス上に制限する
- 項間依存性を考慮する
本手法では、1つの項に対する類似性を計算することによって、一致する可能性のある事例を制限できる。残りの項は、その制限された事例に対して探索を行えばよいので、すべての事例に対して探索する必要がない。

5.2 特異点およびノイズに対する対処

本手法では、一般化の際に一般化できないもの(例外)を認識することで、例外的な事例を一般的に解釈することを防ぐことができる。しかし、本手法で検知できるのは基本的に統計的な例外であって、翻訳の観点から見た例外を検知することはできない。例えば、あるカテゴリにおいてそれに属する語を含む慣用句が頻繁に用いられているならば、そのカテゴリの翻訳として慣用的解釈が選択されてしまう。しかし、本手法は、最もよく用いられる結果が採用されるので、結果の誤り率は、最も低くなる。

しかし、本手法では、例外規則を導入することによって、事前に例外を規則化できれば、この問題を解決できる機構を備えている。

5.3 一般化の計算量

5.3.1 大量の入力に対する計算量

本手法の効率に関しては、自然言語における sparseness と冗長性¹⁰⁾から、定性的には効率的に実現可能であると予測できる。sparseness とは、理論的に組み合わせ可能な候補は非常に多いが、実際意味のある文として出現する組み合わせは少ないという現象であり、

冗長性は、個々の要素が確率的に独立でなく、依存しているということの意味する。項数 M の空間が sparse であり、単語間の依存性がシソーラスによってうまく獲得できれば、一般化の最初のステップである、計算量の小さい項別一般化によって事例の数は劇的に減少する。

5.3.2 再計算のコスト

厳密に言えば、1つの事例を加えることですべての再計算をすることが必要であるが、厳密な再計算を行うのではなく、以下に示す方法で効率的に実現可能である。

- 結果が同じ場合には何も行わず、結果が異なる場合のみ事例をそのまま保持する
最もよく生じる例は一般化されているので、たまに出てくる例しか保持する必要はない
- 選択された事例の重要度が低い場合はシソーラス上の部分木での一般化を行う
しかし、一般化されない事例がある程度たまった時点で、すべての再計算を厳密に行う必要がある。

6. お わ り に

本稿では、事例を一般化することによって、小さな探索空間での効率的な最適照合の手法を提案した。この手法は、一般化によって重要度があらかじめ計算されているために、最適照合の際に、重要度の計算をする必要がない。最適照合は、一般化事例に含まれる概念または語とシソーラス中の概念または語との完全一致のみで実現される。また、規則との融合も可能で、使用開始時点でほとんど事例を持たない場合でも、一般的な規則を前もって準備し、システムをカスタマイズしていくことが可能となる。また、例外的解釈を規則として記述することによって、例外事例が一般的な事例に対する結果に影響することを防ぐことができる。

謝辞 本稿の予稿に対してコメントしてくださった査読者の方々、および、様々な有益な議論をしていただいた日本アイ・ビー・エム東京基礎研究所の自然言語処理グループの皆様へ感謝します。

参 考 文 献

- 1) 佐藤：実例に基づく翻訳，情報処理，Vol. 33, No. 6, pp. 673-681 (1992).
- 2) 渡辺，浦本：Example-Based Machine Translation の問題点に関する考察，第45回情報処理学会全国大会論文集，Vol. 3, pp. 105-106 (1992).

- 3) Nagao, M.: A Framework of a Mechanical Translation between Japanese and English by Analogy Principle, Elithorn, A. and Banerji, R. (eds.): *Artificial and Human Intelligence*, NATO (1984).
- 4) Sadler, V.: *Working with Analogical Semantics: Disambiguation Techniques in DLT*, FORIS Publications (1989).
- 5) Sato, S. and Nagao, M.: Toward Memory-based Translation, *Proc. of Coling '90*, Vol. 3, pp. 247-252 (1990).
- 6) 佐藤: MBT 1: 事例に基づく訳語選択, 人工知能学会誌, Vol. 6, No. 4, pp. 592-600 (1990).
- 7) Sumita, E., Iida, H. and Kohyama, H.: Translating with Examples: A New Approach to Machine Translation, *Proc. of the 3rd Int'l. Conf. on Theoretical and Methodological Issues in Machine Translation*, pp. 203-212 (1990).
- 8) Watanabe, H.: A Similarity-Driven Transfer System, *Proc. of Coling '92*, Vol. 2, pp. 770-776 (1992).
- 9) Stanfill, C. and Waltz, D.: Toward Memory-Based Reasoning, *Comm. of ACM*, Vol. 29, No. 12, pp. 1213-1228 (1986).
- 10) Dagan, I., Itai, A. and Schwall, U.: Two Languages Are More Informative Than One, *Proc. of the 29th Annual Meeting of the ACL*, pp. 130-137 (1991).

(平成4年9月3日受付)

(平成5年3月11日採録)



野美山 浩 (正会員)

1961年生. 1983年九州大学工学部情報工学科卒業. 1985年同大学院工学研究科情報工学専修士課程修了. 同年日本アイ・ビー・エム(株)入社. 現在, 同社東京基礎研究所に勤務. 機械翻訳システムの研究開発に従事.