

文法推論に基づいた翻訳文法の学習方式

佐藤理史、長尾 真

京都大学工学部

二言語間の翻訳を行なう規則集合を、実/反例集合(対/誤訳集合)から獲得する試みについて述べた。まず、翻訳を行なう規則集合として、翻訳文法と呼ぶ対訳を生成(受理)する文法を提案した。この文法によって二言語間の双方向翻訳が可能である。次に、翻訳文法を文法推論に基づいて学習する二つの方式について述べた。一括学習方式は、あらかじめ与えられた対/誤訳集合を満足する翻訳文法を生成する。順次学習方式は、あらたな対/誤訳に対して、それを満足するように翻訳文法を変更する。この二つの方式のアルゴリズムと、それらを用いた英日翻訳文法の学習実験について報告した。

A Method for Learning Language Translation Grammar based on Grammatical Inference

Satoshi SATO, Makoto NAGAO

Department of Electrical Engineering, Kyoto University

This paper describes a method for learning a rule set for language translation between two languages from positive/negative examples(pairs of sentences). We propose Translation Grammar(TG), which can generate/accept pairs of sentences and perform bidirectional translation. We developed two processes for learning TG based on grammatical inference. The batch learning process can generate the TG which satisfies a given set of positive/negative examples. The incremental learning process can modify the TG for satisfying a newly given positive/negative example. Experiments to generate English-Japanese TGs from different sets of examples are given.

1. はじめに

知識ベースを作成したり、保守(改良)したりすることは、エキスパート・システムの構築において大きな問題となっている。その一つの解決策として、学習機能、すなわち、実例集合から知識ベースを獲得する機能の実現が期待されている。

我々は、二つの言語間の翻訳を行なう規則集合を実例集合から獲得する機能の実現を段階的に試みている。本稿の内容は、その第一段階に当たり、以下のような目標を設定し、実現を試みた。

- 実/反例集合からの学習に適した、翻訳をおこなう知識表現を開発すること。
- 与えられた実/反例集合に雑音がない(実例集合と反例集合の積集合が空である)という条件下において、与えられた実/反例集合を満足する(実例集合を全て覆い反例集合を全て覆わない)規則集合を生成することができる学習機構を開発すること。

本稿で述べる学習方式の基本的アイデアは、以下の二点に要約される。

- 規則集合を表す表現形式として、形式文法に基づいた表現(翻訳文法)を採用した。
- 規則集合の学習に文法推論の枠組を用いた。

これらのアイデアを具体化し、機械学習/翻訳システム「琢磨II」を作成した。「琢磨II」の主要部は、学習機構と翻訳機構から成り、前者が実/反例集合から翻訳に必要な知識である翻訳文法を獲得し、後者はそれを用いて翻訳を行なう。本システムにおいて翻訳文法は、以下の手順に従って獲得される。(図1-1参照)

- (1) 実/反例集合、すなわち、対訳例と誤訳例の集合を用意する。
- (2) (1)を「琢磨II」の学習機構に与え、(1)を満足する翻訳文法を生成させる。(一括学習方式)
- (3) (2)で生成された翻訳文法を用いて、翻訳(実験)を行なう。
- (4) (3)の結果を人間が判断し、得られた訳が誤っていた場合はそれを指摘したり、訳せなかった場合は正しい訳を与えたりして、翻訳文法を改良させる。(順次学習方式)

以下、2節では本システムの知識表現である翻訳文法について述べる。3節において翻訳文法の学習について述べる。4節では学習実験とその結果について述べる。

2. 翻訳文法

本節では、「琢磨II」の知識表現である翻訳文法について述べる。翻訳文法は、対訳を生成(受理)する文法、翻訳を行なう規則集合の二面性をもつ。

2.1 翻訳文法の定義と意味

翻訳文法(非終端記号共有文法)は以下によって定義される。

- (1) 次の記号集合がある。
 - (a) 非終端記号の集合： V_N
 - (b) 終端記号の集合： V_T
 - (b-1) 言語Aの終端記号の集合： V_{TA}
 - (b-2) 言語Bの終端記号の集合： V_{TB}
- (2) 次のような規則の集合Pがある。

$$y \leftarrow X \rightarrow z$$

$$\text{但し、} X \in V_N, y \in (V_N \cup V_{TA})^+,$$

$$z \in (V_N \cup V_{TB})^+$$

なお、Xを左辺、yを言語A側の右辺、zを言語B側の右辺と呼ぶ。

(1) [実(反)例集合の用意]

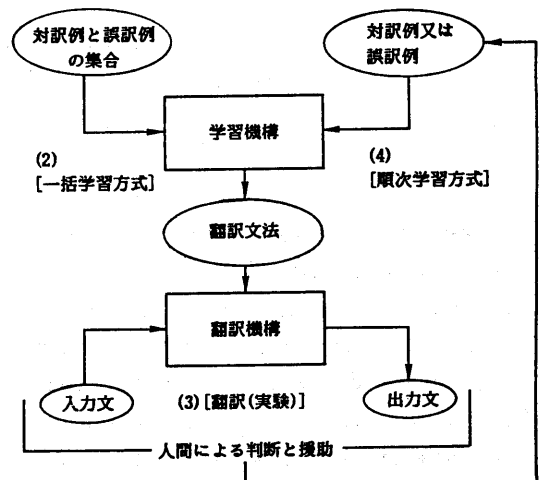


図1-1 「琢磨II」の概要

(3) 初期記号 $\sigma \in V_N$ があって、翻訳文法 G はこれらの要素から

$$G = \langle V_N, V_{TA}, V_{TB}, P, \sigma \rangle$$

と表現される。

簡単な翻訳文法の例を図2-1に示す。

翻訳文法は、直観的には、文脈自由文法(CFG)の各規則に右辺をもう一つ追加したものと考えればよい。あるいは、言語Aと言語Bの二つのCFGを、対応する非終端記号に共通な記号を用いることによって、一つに統合したものと考えることもできる。

翻訳文法は、対訳を生成(受理)することができる。

いま、

$$[s, t] \quad s \in (V_N \cup V_{TA})^+, t \in (V_N \cup V_{TB})^+$$

を状態(state)とする。この特殊な場合、すなわち、

$$[a, b] \quad a \in V_{TA}^+, b \in V_{TB}^+$$

を対訳と呼ぶことにする。

規則

$$y \leftarrow X \rightarrow z \quad X \in V_N, y \in (V_N \cup V_{TA})^+, z \in (V_N \cup V_{TB})^+$$

は、

$$[c \ X \ d, e \ X \ f]$$

↓

$$[c \ y \ d, e \ z \ f]$$

$$c, d \in (V_N \cup V_{TA})^*, e, f \in (V_N \cup V_{TB})^*$$

なる状態間の書き替えを行なうものとするとき、翻訳文法は、初期状態 $[\sigma, \sigma]$ に規則を適用することによって、対訳を生成することができる。すなわち、翻訳文法においては、対訳がいわゆる『文(sentence)』に相当し、対訳の集合が『言語(language)』に相当することになる。

2.2 翻訳方式

翻訳文法を用いると、二言語間の双方向翻訳が可能である。以下、その原理について述べる。例として言語Aから言語Bへの翻訳を考える。いま、言語Aの文 $a \in V_{TA}^+$ が与えられたとする。この文を翻訳するということは、翻訳文法によって生成される対訳のうち、 $[a, b]$ の形をした対訳の言語B側の文 $b \in V_{TB}^+$ を求めることにほかならない。これは次のようにして求めることができる。

(1) 翻訳文法に含まれる各規則の言語B側の右辺を無視して言語AのCFGとみなし、言語Aの文に対して規則を解析方向に適用し、初期記号にまで書き替える規則適用系列を得る。

(2) 翻訳文法の言語A側の右辺を無視して言語BのCFGとみなし、初期記号に対して、(1)で得た規則適用系列の逆順に規則を生成方向に適用する。これによって、言語Bの文を得る事ができる。

なお、一つの言語Aの文に対して解析結果が複数得られるときは、言語Bの文が複数求まる。

```

%0001 %0002 %0003 . <- %s -> %0001 は %0003 を %0002 .
      I <- %0001 -> 私
      you <- %0001 -> あなた
      speak <- %0002 -> 話す
      learn <- %0002 -> 学ぶ
      English <- %0003 -> 英語
      Japanese <- %0003 -> 日本語
    
```

図2-1 簡単な翻訳文法の例

%で始まる記号が非終端記号で、他の記号が終端記号である。各規則の二つの右辺が二言語間の対応関係を表わし、同じ左辺を持つ規則群が翻訳において(統語的あるいは意味的に)同じ振舞をする記号列のグループを表わす。

3. 翻訳文法の学習

翻訳文法を対訳を生成(受理)する文法とみなすことにより、この文法の学習を、従来のCFGの推論(学習)の枠組^[1]で考えることができる。最大の違いは、翻訳文法では規則の右辺が二つあるということである。本節では、翻訳文法の変更操作を設定したのち、一括学習方式と順次学習方式の二つの学習方式について述べる。

3.1 文法の変更操作

文法の学習では、まず、文法に対してどのような変更を許すのかを決める必要がある。今回設定した変更操作は、以下の通りである。ただし、英大文字は非終端記号を、英小文字は任意の記号列を表すものとする。

- (1) 規則の追加：文法に新しい規則を追加する。
- (2) 規則の削除：文法に含まれる規則を削除する。
- (3) 非終端記号化：n個の記号列の組

$[y_1, z_1], [y_2, z_2], \dots, [y_n, z_n]$
を非終端記号化する。すなわち、新しい非終端記号Xをつくり、n個の規則

$y_1 \leftarrow X \rightarrow z_1, y_2 \leftarrow X \rightarrow z_2, \dots, y_n \leftarrow X \rightarrow z_n$
を文法に追加する。

- (4) 規則の一般化：規則

$y_1 \leftarrow X \rightarrow z_1$
があるとき、規則
 $a y_1 b \leftarrow A \rightarrow c z_1 d$
を規則

$a X b \leftarrow A \rightarrow c X d$
で置き換える。

- (5) 規則の統合：n個の規則

$a y_1 b \leftarrow A \rightarrow c z_1 d, a y_2 b \leftarrow A \rightarrow c z_2 d, \dots$
 $a y_n b \leftarrow A \rightarrow c z_n d$
の右辺の差

$[y_1, z_1], [y_2, z_2], \dots, [y_n, z_n]$
を非終端記号化(新しい非終端記号Xを生成)し、1つの規則
 $a X b \leftarrow A \rightarrow c X d$

で置き換える。

- (6) マージング：新しい非終端記号Yをつくり、文法の規則に含まれるn個の非終端記号 X_1, X_2, \dots, X_n をすべてYで置き換える。

- (7) 非終端記号の展開：ある規則の右辺に含まれる非終端記号を展開する。規則

$a X b \leftarrow A \rightarrow c X d$
の右辺の非終端記号Xを展開する場合を例にとると、Xを左辺に持つ規則が
 $y_1 \leftarrow X \rightarrow z_1, y_2 \leftarrow X \rightarrow z_2, \dots, y_n \leftarrow X \rightarrow z_n$
のn個ある場合、規則

$a X b \leftarrow A \rightarrow c X d$
を削除し、n個の規則
 $a y_1 b \leftarrow A \rightarrow c z_1 d, a y_2 b \leftarrow A \rightarrow c z_2 d, \dots$
 $a y_n b \leftarrow A \rightarrow c z_n d$
を追加する。

変更前の文法をG、変更後の文法をG'とし、文法G、G'が生成する言語をそれぞれL(G)、L(G')とするとき、上記の変更操作は以下の(a)~(c)の3つに分類される。

- (a) $L(G)=L(G')$ ：文法の等価変換。(3),(5),(7)。

(b) $L(G) \subseteq L(G')$: 文法の一般化。(1), (4), (6)。

(c) $L(G) \supseteq L(G')$: 文法の特特殊化。(2)。

与えられた実/反例集合を全て満足する文法を生成するためには、(b)(c)に属する変更操作を適用する際に、注意が必要である。そのため、以下のような適用制限条件を課す。

□ 文法の一般化は、変更後の文法が与えられた反例を一つも生成(受理)しない限りにおいて適用可能とする。

□ 文法の特特殊化、すなわち、規則の削除は、その規則が与えられた実例集合を生成(受理)するために使用されない限りにおいて削除可能とする。

このような条件を課せば、これらの変更操作を適用しても、それによって与えられた実(反)例が満足されなくなることはない。

3.2 一括学習方式

一括学習方式は、実/反例集合が(一度に)与えられたとき、実例を全て受理し、反例を一つも受理しない文法を求める方式である。一括学習方式では、まず、左辺が初期記号で右辺が与えられた実例という実例規則からなる文法を作り、次にこの文法に対して、反例を一つも受理しないという条件下で一般化と単純化を施すことに他ならない(文法の特特殊化は使わない)。(1)規則間の差の計算、(2)最小の差に対する操作、(3)規則の一般化の3ステップを繰返すアルゴリズムを用いている。

(1) 規則間の差の計算

左辺が同じ二つの規則の各右辺を相互に比較し、前方および後方から同じ記号列を切り取り、差を求める。すなわち、二つの規則

$$a \ y_1 \ b \ \leftarrow \ X \ \rightarrow \ c \ z_1 \ d$$
$$a \ y_2 \ b \ \leftarrow \ X \ \rightarrow \ c \ z_2 \ d$$

を比較し、差 $\langle y_1, y_2, z_1, z_2 \rangle$ を得る。この差を全ての組み合わせに対して求める。

(2) 最小の差に対する操作

(1)で得られた差を(a)~(f)の6つのタイプに分類する。差のうち最も小さいものを選び、それに応じた操作を施す。差の大小は、(a) < (b) .. < (f)とし、同一タイプにおいては右辺全体に対する差の割合が小さいものを小さいとする。もし最小の差が(f)なら終了する。

差の分類とそのときの操作は以下の通りである。(|x|は記号列xの長さとする)

(a) $|y_1|=|y_2|=|z_1|=|z_2|=0$: 二つの規則は同一なので、片方の規則を削除する。

(b) $y_1=z_1=1$ つの非終端記号, $|y_2| \geq 1, |z_2| \geq 1$, ただし(d)以外 : 規則

$$y_2 \leftarrow y_1 \rightarrow z_2$$

の追加を試み、成功したなら比較した規則のうち y_2, z_2 を含む規則を削除する。

(c) $|y_1|=|z_1|=0, |y_2| \geq 1, |z_2| \geq 1$, ただし $y_2=z_2=1$ つの非終端記号でない場合 : $[y_2, z_2]$ を非終端記号化する。

(d) $y_1=z_1=1$ つの非終端記号, $y_2=z_2=1$ つの非終端記号 : y_1 と y_2 のマーキングを試みる。失敗した場合は、二つの規則を統合する。

(e) $1 \leq |y_1| \leq 4, 1 \leq |z_1| \leq 4, 1 \leq |y_2| \leq 4, 1 \leq |z_2| \leq 4$, かつ、差の部分が右辺全体の50%以下。ただし

(b)(d)以外 : 二つの規則を統合する。

(f) (a)~(e)以外 : 繰返しを終了する。

但し、(b), (c)には、 y_1 と y_2, z_1 と z_2 を入れ替えた場合も含まれるものとする。

(3) 規則の一般化

(2)で新しい規則が追加された場合は、その規則を文法に含まれる規則の右辺に適用し、規則の一般化を試みる。規則の一般化が成功し、新しい規則が生成された場合は、再帰的に繰返す。

3.3 順次学習方式

順次学習方式は、実(反)例が一つ与えられるたびに必要に応じて文法を変更する方式であ

る。実例が与えられた場合と反例が与えられた場合の二つの場合がある。

実例が与えられた場合は、まず、現在の文法がその実例を受理するかどうか調べる。受理する場合は、文法を変更する必要はない。受理しない場合は与えられた実例を受理するように文法を一般化する。文法を一般化する変更操作は3つあるが、新たな実例を受理(生成)するための一般化には、一つの規則の追加による一般化を用い、それ以外の可能性は探索しないものとする。文献[1]の精密化のアルゴリズムを翻訳文法に合うように変更し用いている。すなわち、実例に対して文法を解析方向に適用して部分的解析(partial parse)を得、得られた部分的解析を右辺とし初期記号を左辺とする規則のうち、追加可能で最も一般的な規則を追加する。その後、新しい規則が一つ追加された文法に対して、一括学習方式を適用し、文法の一般化と簡単化を試みる。

反例が与えられた場合は、まず、現在の文法がその反例を受理するかどうかを調べる。受理しない場合は、文法を変更する必要はない。受理する場合は、与えられた反例を受理しないように文法を特殊化する。ある反例が受理されるということは、文法が過度に一般化されているということである。そこで、その部分を見つけて特殊化してやればよい。この特殊化は、(1) 展開すべき非終端記号の特定、(2) 展開すべき規則の特定、(3) 非終端記号の展開、(4) 規則の削除、の4ステップから成る。(図3-1参照)

(1) 展開すべき非終端記号の特定

反例に文法を(解析方向に)適用し、反例から初期状態に至るラティスを作る。このラティスにおいて、各節点(状態)が、今までに与えられた実例を覆う(その実例を受理する過程でその状態を通る)かどうかを、横方向優先探索によって調べる。最初に見つけた、与えられた実例を覆う節点を分岐節点とよび、最後にたどった枝(規則の適用)によって書き替えられた非終端記号を展開すべき非終端記号とする。

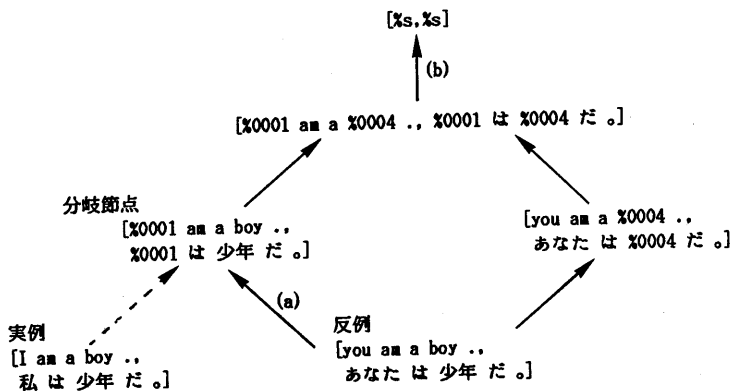


図3-1 順次学習方式(反例が与えられた場合)

反例に対して規則を解析方向に適用し、反例から初期状態[xs, xs]に至るラティスを作る。翻訳文法は図2-1に示した文法に3つの規則

```
x0001 am a x0004 . <- xs -> x0001 は x0004 だ。
boy <- x0004 -> 少年
girl <- x0004 -> 少女
```

を追加した文法を仮定する。このラティスにおいて分岐節点を探す。

分岐節点への枝(a)で書き替えられる非終端記号(x0001)を展開すべき非終端記号とする。分岐節点から初期状態に至る枝で適用される規則のうち、x0001を含む記号列を書き替える規則(枝(b)で適用)

```
x0001 am a x0004 . <- xs -> x0001 は x0004 だ。
```

を展開すべき規則とする。

(2) 展開すべき規則の特定

次に、分岐節点から初期状態に向けて縦方向優先探索を行ない、たどる枝において適用される規則のうち、(1)で求めた非終端記号を含む記号列を書き替える規則を見つける。最初に見つかった規則を展開すべき規則とする。

(3) 非終端記号の展開

(2)で求めた規則の右辺に含まれる(1)で求めた非終端記号を展開する。

(4) 規則の削除

(3)で追加された規則、及び、(1)で求めた非終端記号を左辺に持つ規則の削除を試み、削除可能な規則を削除する。

この特殊化の適用後、特殊化された文法が反例を受理するかどうか調べる。受理する場合は、この特殊化を繰り返す。受理しない場合は、特殊化を終了し、一括学習方式を用いて、文法の一般化と単純化を試みる。

上記の方法で、与えられた反例を受理しないように文法を特殊化することができる。何故なら、この特殊化を繰り返し適用していけば、最後には、与えられた実例を右辺とする規則(実例規則)だけから成る文法まで特殊化される。この文法は、雑音がないという条件下では、反例を受理しない。

4. 実験及び考察

簡単な英日翻訳文法を生成させる実験を行なった。英語文は中学一年の範囲^[2]から簡単な平叙文を選び、日本語文はそれに対する訳文を用意した。入力した対訳は両言語とも単語列であり、英語文の文頭の大文字は小文字に変換したものをを用いた。活用形、語尾変化などを原形に戻す処理は施さなかった。実験結果を付録に示す。

これらの実験を通じて以下のことがわかった。

□ 翻訳文法と文法推論の枠組を用いることによって、対/誤訳例集合から二言語間の翻訳を行なう規則集合を比較的単純なアルゴリズムで生成(学習)することができる。生成された翻訳文法は、与えられた対訳と誤訳を全て満足し、対訳として与えなかった未知の翻訳をある程度予測することができる。実例と反例をうまく与えれば、簡単な訳し分けもできるようになる。

□ 自然言語間の翻訳を行なう知識表現として、翻訳文法の表現力は十分ではない。複雑な文の翻訳やきめ細かい翻訳を行なうためには、膨大な数の非終端記号と規則が必要になる。非終端記号にfeatureを導入して非終端記号や規則の数をへらすなどの拡張を考える必要がある。

□ 用いているアルゴリズムは、与えられた実(反)例集合を満足する翻訳文法のうち、最も一般的な文法に近い文法を生成する。すなわち、明示的な反例が与えられない限り、一般化が抑制されることはない。かつ、非終端記号を動的に変更できるため、反例は特異な点として排除される傾向が強く、一般に多くの反例を必要とする。これを防ぐために、反例の一般化を考える必要がある。

5. おわりに

本稿では、対/誤訳集合から翻訳を実行する規則集合(翻訳文法)を獲得する試みについて述べた。対訳を生成する文法と翻訳を行なう規則集合の二面性をもつ翻訳文法という知識表現を考察し、それが文法推論の枠組で学習できることを示した。今後の課題としては、(a)非終端記号にfeatureを導入して知識表現を豊かにする、(b)featureを発生させる機能を実現する、(c)反例の一般化を行なうなどがあげられる。

[参考文献]

[1] Cohen and Feigenbaum編, 人工知能ハンドブック第III巻, 第XIV章 学習と帰納的推論, 共立出版。

[2] 鈴木進, ピクリー, V. C. 共著, チャート式中学英語1年, 数研出版。

| ID | 132 | | 1122 | | 1530 | |
|-----------|------|-----|--------|-----|---------|-----|
| | 英語 | 日本語 | 英語 | 日本語 | 英語 | 日本語 |
| 実例集合 | 30 | | 74 | | 208 | |
| 実例数 | 30 | | 74 | | 208 | |
| 単語の種類 | 37 | 29 | 66 | 48 | 159 | 145 |
| 平均文長(単語数) | 5.0 | 6.1 | 4.9 | 6.3 | 5.3 | 6.5 |
| 反例集合 | 2(2) | | 50(15) | | 345(61) | |
| 反例数 | 2(2) | | 50(15) | | 345(61) | |
| 翻訳文法 | 14 | | 25 | | 73 | |
| 非終端記号数 | 14 | | 25 | | 73 | |
| 規則数 | 42 | | 92 | | 270 | |
| 平均右辺長 | 1.7 | 1.9 | 1.6 | 1.8 | 2.0 | 2.2 |

(a) 学習実験結果一覧
 反例数の括弧で囲まれた数字は、一般化の抑制および
 特殊化で使われた反例の数を表す。

Positive instances : 30
 1 : [I am Takuma.] * [私はタクマだ。]
 2 : [I am Taro.] * [私は太郎だ。]
 3 : [you are Takuma.] * [あなたはタクマだ。]
 4 : [you are Hanako.] * [あなたは花子だ。]
 5 : [I am a boy.] * [私は少年だ。]
 6 : [you are a girl.] * [あなたは少女だ。]
 7 : [you are a boy.] * [私は少年だ。]
 8 : [you are a tall boy.] * [私は背の高い少年だ。]
 9 : [I am a small girl.] * [あなたは小さな少女だ。]
 10 : [this is a book.] * [これは本だ。]
 11 : [that is a book.] * [それは本だ。]
 12 : [this is an apple.] * [これはりんごだ。]
 13 : [this is my apple.] * [これは私のりんごだ。]
 14 : [this is my book.] * [これは私の本だ。]
 15 : [this is your book.] * [これはあなたの本だ。]
 16 : [that is a dog.] * [それは犬だ。]
 17 : [that is Taro's dog.] * [それは太郎の犬だ。]
 18 : [he is a boy.] * [それは少年だ。]
 19 : [she is a girl.] * [それは少女だ。]
 20 : [he is my friend.] * [私は私の友達だ。]
 21 : [she is a teacher.] * [彼女は先生だ。]
 22 : [she is Hanako's teacher.] * [彼女は花子の先生だ。]
 23 : [that is a tennis ball.] * [それはテニスボールだ。]
 24 : [it is your tennis ball.] * [それはあなたのテニスボールだ。]
 25 : [I'm not Taro.] * [私は太郎ではない。]
 26 : [she is a nurse.] * [彼女は看護婦ではない。]
 27 : [you aren't a nurse.] * [あなたは看護婦ではない。]
 28 : [he isn't my teacher.] * [私は私の先生ではない。]
 29 : [she isn't Hanako's mother.] * [彼女は花子の母親ではない。]
 30 : [it isn't a tennis ball.] * [それはテニスボールではない。]
 Negative instances : 2
 31 : [this is an apple.] * [これは本だ。] : (used)
 32 : [this is an apple.] * [これはりんごだ。] : (used)

(b) 与えた実例集合と反例集合(132)

Rules : 42
 apple * X0016 * Uんこ]: 13
 Taro * X0016 * X0018]: 11 14 15 17 23 24 30
 it * X0017 * それ]: 24 30
 that * X0017 * 犬]: 11 15 17 23
 dog * X0018 * 犬]: 16 17
 book * X0018 * 本]: 10 11 14 15
 tennis ball * X0018 * テニスボール]: 23 24 30
 Taro's * X0019 * 太郎]: 17
 your * X0019 * あなたの]: 15 24
 teacher * X0020 * 先生]: 21 22 28
 girl * X0020 * 少女]: 6 9 19
 boy * X0020 * 少年]: 5 7 8 18
 nurse * X0020 * 看護婦]: 26 27
 mother * X0020 * 母親]: 29
 friend * X0020 * 友達]: 20
 he * X0021 * 彼]: 18 20 28
 she * X0021 * 彼女]: 19 21 22 26 29
 Taro * X0022 * 太郎]: 2 25
 Takuma * X0022 * タクマ]: 4
 Hanako * X0022 * 花子]: 1 3
 a * X0020 * X0022 * X0020]: 5 6 7 18 19 21 26 27
 a * X0023 * X0020 * X0022 * X0023 * X0020]: 8 9
 a * X0028 * X0020 * X0022 * X0028 * X0020]: 20 22 28 29
 small * X0023 * 小さな]: 9
 tall * X0023 * 背の高い]: 8
 is * X0024 * だ]: 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 26
 isn't * X0024 * ではない]: 28 29 30
 you aren't * X0025 * あなた]: 27
 I'm not * X0025 * 私は]: 25
 an apple * X0026 * りんご]: 12
 a * X0018 * X0026 * X0018]: 10 16
 X0028 * X0016 * X0026 * X0028 * X0016]: 13 14
 you are * X0027 * あなた]: 3 4 6 7 9
 I am * X0027 * 私は]: 1 2 5 8
 my * X0028 * 私の]: 13 14 20 28
 Hanako's * X0028 * 花子]: 22 29
 X0017 * X0024 * X0016 * X0017 * X0016 * X0024]: 11 23 30
 X0021 * X0024 * X0022 * X0021 * X0022 * X0024]: 18 19 20 21 22 26 28 29
 X0017 * X0024 * X0019 * X0016 * X0017 * X0019 * X0016 * X0024]: 15 17 24
 X0025 * X0022 * X0025 * X0022 * X0025 * X0022]: 25 27
 this * X0024 * X0026 * X0024 * X0026 * X0024]: 10 12 13 14 16
 X0027 * X0022 * X0027 * X0022 * X0027 * X0022]: 1 2 3 4 5 6 7 8 9

(c) 獲得された翻訳文法(132)

規則の後ろに付けられた数字は、
 翻訳時にその規則が使われる実例番号を示す。

【付録】学習実験結果

「琢磨II」による英日翻訳文法の学習実験の結果を示す。
 (a)は、実験結果の一覧表であり、そのうち、132に関し
 ては、与えた実例集合を(b)に、獲得された翻訳文法
 を(c)に示す。