英日機械翻訳における自然な訳文への言い換えシステム

宮地 洋太　田添 丈博
鈴鹿工業高等専門学校　専攻科　電子機械工学専攻
椎野 努
愛知工業大学　情報科学部　情報科学科

１．背景・目的

機械翻訳には「自動翻訳」と「翻訳支援」の２つがある。この２つは方向性が異なり「自動翻訳」は翻訳元の言語を理解することのできない人のための技術であるが、技術的に困難であり実現の目処は立っていない。また「翻訳支援」は翻訳元の言語に理解があり翻訳作業を効率的・正確に行うために翻訳ソフトを利用することで、翻訳者に高度な知識が必要とされている[1]。

既存の英日機械翻訳システムにおいて、短い英文の機械翻訳はそれなりの精度があり、口頭で用いられる英文は短いことから音声解析の技術の向上によりほぼリアルタイムの音声翻訳が可能となっている[2]。しかし、音声翻訳でも長文・複雑な文の翻訳は行えていないのと同じく、既存の英日機械翻訳システムにおいて、長文・複雑な文では翻訳結果に直訳に近い硬い表現が用いられることが多く、日本語として意味の取れない翻訳結果が出力されることもある。

本研究では、不自然な日本語となった機械翻訳文に対して「言い換え」を行い、より自然な日本語訳を出力するシステムを提案する。言い換えデータは、翻訳結果を得たユーザが、より自然と考えられる改良訳文をシステムに与えることにより、係り受け構造のマッチングを行い学習する。そして学習した言い換えデータを別の機械翻訳文を言い換える際にも活用する。

本論文は、２章でシステムの概要、学習機能、言い換え機能について示し、３章で機械翻訳文と改良訳文とのマッチングを行う学習機能の評価について示し、最後に４章でまとめを示す。

２．システムの構成

2.1　システムの概要

システムの構成を図１に示す。本システムはユーザから入力された英文を既存の英日機械翻訳システムへの入力とし、出力される機械翻訳文に対し「言い換え」を行うことでより自然な訳文を出力する。言い換えを行うためのデータは機械翻訳文に対して、ユーザがより自然であると考える改良訳文を入力し、それにより言い換えデータを学習する機能を設計し、その言い換えデータを用いて言い換えを行う機能を設計した。

本システムでは「学習機能」と、それを利用する「言い換え機能」とで、より自然な訳文を得ることを目的としている。

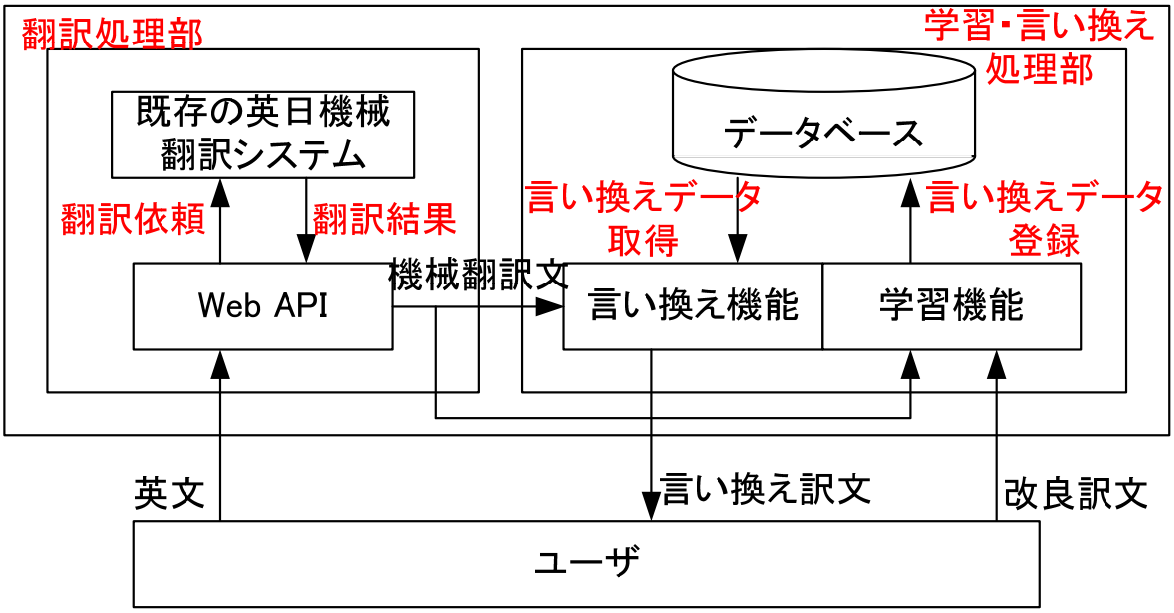


図１　システムの構成

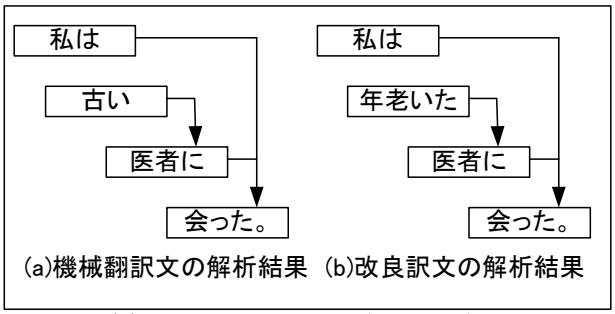
2.2　学習機能

言い換えデータの学習は機械翻訳文と改良訳文の２つに対し、構文解析を行った結果より文節単位で係り受け構造のマッチングを行う。

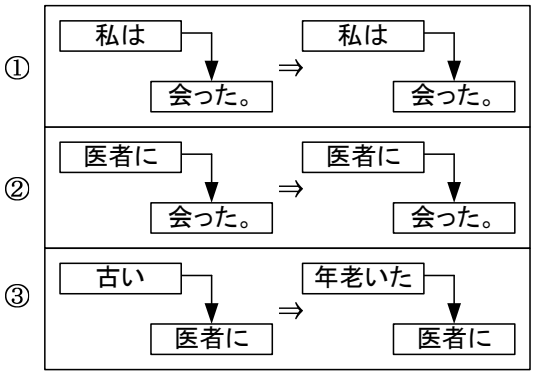
例を図２(A)に示し、構文解析を行った結果を(B)、登録されるデータを(C)に示す。

[英文]	I saw an old doctor.
[機械翻訳文]	私は古い医者に会った。
[改良訳文]	私は年老いた医者に会った。

(A)英文、機械翻訳文、改良訳文



(B)それぞれの訳文を構文解析した結果



(C)登録される言い換えデータ

図２　学習の手順

マッチング方法は、それぞれの訳文に完全に一致する文節を「一致文節」として、一致文節があるかを検索する。このとき、ある文節の一致が複数通り考えられる場合は、どの文節同士が対応しているのかを機械的に判断することが難しいため、マッチングを行わないこととした。一致文節が１対１であるならば一致文節の係り受け関係にある文節を検索する。

まず一致文節が係る関係にある文節を「係る文節」として検索する。係る文節は１つであり、それぞれの訳文の一致文節と係る文節を一組のデータとして学習する。係る文節がなければ学習しない。図２(C)では①「私は」と②「医者に」の一致文節と係る文節の組が学習される。

次に一致文節が受ける関係にある文節を「受ける文節」として検索する。受ける文節が２つ以上見つかった場合には、どちらの文節が正しい受ける文節であるかを機械的に判断することが難しいため学習しないこととした。１つの場合には係る文節と同様にデータを学習する。図２(C)では③「医者に」の一致文節と受ける文節が学習される。

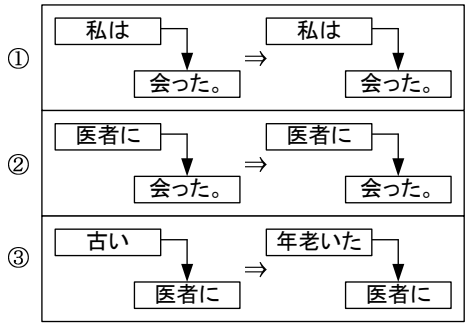
2.3　言い換え機能

言い換えの方法は機械翻訳文に構文解析を行い、その結果に言い換えデータの言い換え前と完全に一致した際に、言い換えデータの言い換え後に「言い換え」が行われ、より自然な訳文となる。

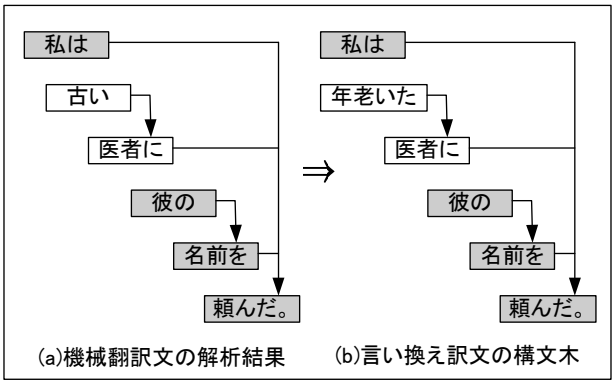
図３(A)に入力英文とその機械翻訳文、(B)に学習している言い換えデータ、(C)に機械翻訳文と言い換え訳文の構文木を示す。

[英文]	I asked an old doctor his name.
[機械翻訳文]	私は古い医者に彼の名前を頼んだ。

(A)英文、機械翻訳文



(B)登録されている言い換えデータ



(C)結果

図３　言い換えの手順

機械翻訳文が言い換え機能に渡されると文節とその係り受け関係の文節の組に一致する言い換えデータが既に学習されているかを調べる。図３の例では一致する文節の組は登録されている言い換えデータの③だけである。ここで一致する言い換えデータが見つかったので該当する機械翻訳文の文節が言い換えられる。

３．学習機能の検証

3.1　検証方法

実際にこのシステムに英文と改良訳文を入力し、どれだけの言い換えデータを学習することができるか、またその内で正しい言い換えデータ、誤った言い換えデータをどれだけ学習しているかを検証する。

検証には RFC862、863、864 の見出しを除く 51 文の英文と対訳を用いて学習を行わせた。51 文の機械翻訳文と対訳に対する我々の手による正解は 55 組として評価を行う。

既存の英日機械翻訳システムには Yahoo!BabelFish[3]を、構文解析には構文解析ソフトの Cabocha[4]を用いた。

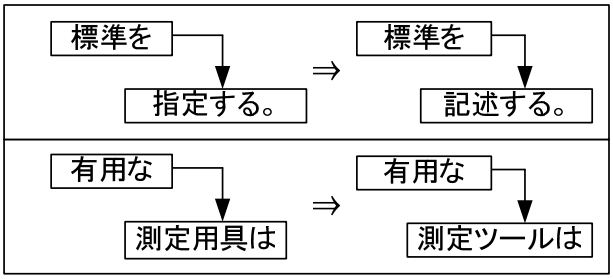
3.2　検証結果

学習した言い換えデータは全部で 61 組、その内 47 組が正しい言い換えデータ、14 組が誤って学習した言い換えデータ、正解の内学習することのできなかった言い換えデータは 8 組であった。

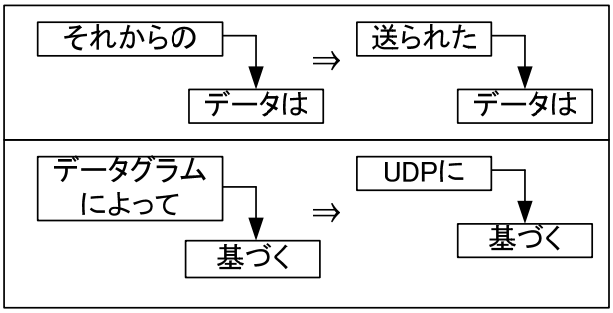
評価を行った結果を表 1 に示す。また学習した言い換えデータの一例を図 4 に示す。

表 1　RFC を学習した結果

再現率	適合率	F値
0.855(=47/55)	0.770(=47/61)	0.810



(a)　正しい言い換えデータ(47個)



(b)　誤った言い換えデータ(14個)

図 4　RFC を学習した言い換えデータの一例

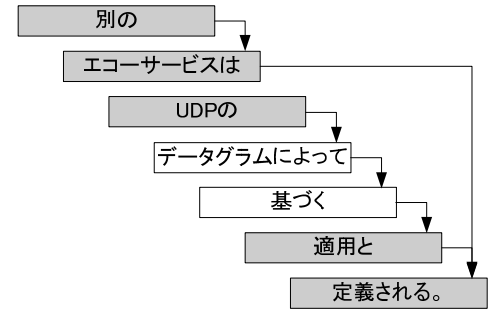
3.3 考察

検証により、誤った言い換えデータを学習した原因が 2 つ、学習できなかった言い換えデータの原因が 1 つ、計 3 つの問題があった。

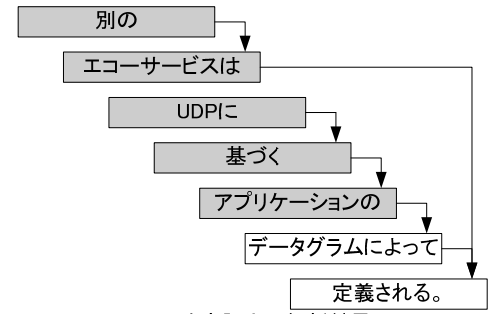
3.3.1　誤った言い換えデータ①

誤った言い換えデータの学習の原因として、機械翻訳文と改良訳文のそれぞれに対応する文節は存在するが構文木の構造・文節の順番の違いにより一致文節に対して誤った係り受け関係を学習してしまった言い換えデータが 8 組であった。

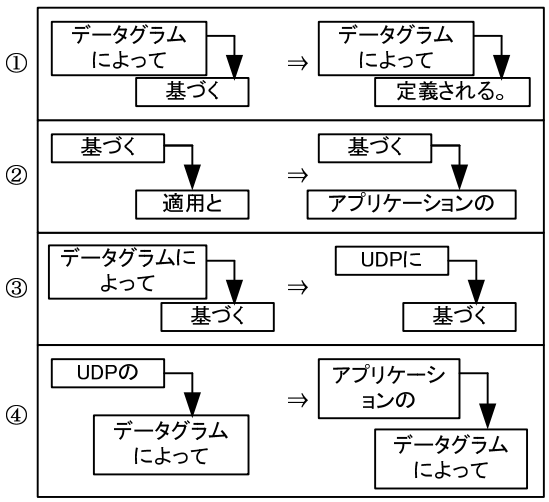
構文木の構造・文節の順番の違いについて、誤った言い換えデータを学習した機械翻訳文と改良訳文の構文木の一例を図 5 に示す。



(a) 機械翻訳文の解析結果



(b) 改良訳文の解析結果



(c)学習する誤った言い換えデータ

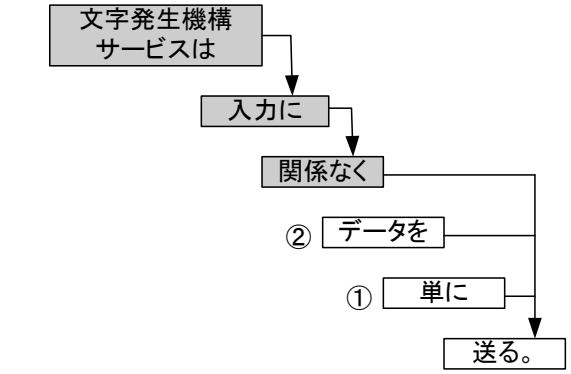
図 5　誤ったマッチングを行った構文木

図５の構文木では文節の順番の違いで係り受け関係が異なり、一致文節を中心に係り受け関係の文節の学習を無条件に行うと、一致文節と係り受け関係にある文節が「正しい言い換え」に対応していないことにある。

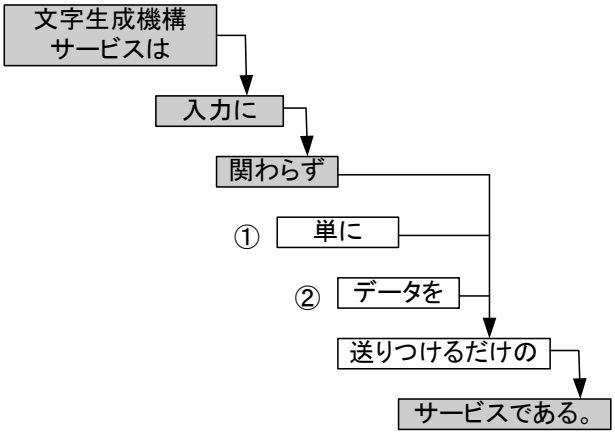
図５(a)、(b)で一致文節である「基づく」・「データグラムによって」の２つに注目すると、それぞれの構文木において文節の順番が入れ替わっていることがわかる。そして誤った言い換えデータのすべてがこれらの係り受け構造となっている。よってこの原因の対策として、一致文節の順番が入れ替わっている場合には学習しない、という方針で対策できるのではないかと考える。

3.3.2　誤った言い換えデータ②

機械翻訳文と改良訳文の表現の違いから対応する文節が改良訳文に存在しないために誤った言い換えデータが６組であった。その一例を図６に示す。



(a)機械翻訳文の解析結果



(b)改良訳文の解析結果

図６　それぞれの訳文の表現の違い

図６の①、②のどちらも「送る。」を「送りつけるだけの」と言い換えている。これは改良訳文だけに最後に「サービスである。」という文節が補われていることにある。これは日本語としての表現が関係してくるため様々なパターンがあったが、一方の訳文の文末にある文節が他方の途中にあることが多く、文末にあたる文節とそうでない文節とのマッチングを学習しないことで対策できると考える。

3.3.3　学習できなかった言い換えデータ

人手による正解では完全に一致する文節が複数存在する場合・受け関係が複数ある文節に対して理想のマッチングを行った。検証で学習できなかった８組の正しい言い換えデータはすべてこれに該当する。正解の内、学習することができなかった８組の言い換えデータについては学習機能で挙げた、一致文節が複数存在する、受ける文節が複数存在する、といった状況に対して人手によりマッチングを行った結果である。よって現在の手法における正しい言い換えデータはすべて学習できたと言える。

学習できなかった言い換えデータについて、一致文節が複数存在する場合にはそれぞれの一致文節の係る文節・受ける文節を解析し類似性の高い一致文節とマッチングを行わせ、受ける文節が複数存在する場合には受ける文節を解析し類似性の高い文節を対応した受ける文節としてマッチングを行う方法を考えている。

４．まとめ

本研究では言い換えデータの学習機能と、それを用いた言い換え機能による言い換えシステムを提案した。そのうち学習機能の評価を行い、現在の学習機能で学習できる正しい言い換えデータはすべて学習することができた。

今後の課題として、考察で述べた方法用いた検証を行い、同時に言い換え機能についての評価を行う必要があると考える。

参考文献

[1]　AP transways
http://www.aptransways.net/transaid01.htm
[2]　科学技術政策
http://www8.cao.go.jp/cstp/5minutes/004/index.html
[3]　Yahoo!BabelFish
http://babelfish.yahoo.com/
[4]　Cabocha
http://chasen.org/~taku/software/cabocha/