2C−10

# DTP形式情報を保存する機械翻訳支援システム

伊藤悦雄　武田公人　平川秀樹　天野真家
(株)東芝　総合研究所

## 1. はじめに

DTP(Desk Top Publishing)システムの発達により計算機を利用した文書作成が行われるようになってきた。そして、DTPシステムによって作成した文書を翻訳し、さらに翻訳結果をDTPシステムで扱う場合が増加している。こういった場合、DTPシステムと機械翻訳システムを独立したシステムとするより、両者を結合し総合的な文書作成システムとして扱うことによって、文書作成/編集環境の向上を図ることができる。

本稿では、そういった状況を背景として、DTPシステムで作成した文書を書式情報を保存したまま翻訳することができる機械翻訳支援システムについて述べる。

## 2. 利用形態

DTPシステムで作成した文書を翻訳する形態には以下の三方法が考えられる。

1：DTPシステムの画面上で指定した部分を翻訳する。
2：DTPシステムで作成した文書から文字列を抽出して、機械翻訳システムで利用する。
3：DTPシステムで作成した文書から文字列を抽出し、翻訳を行った結果を再びDTPシステムで使用する。この場合、翻訳結果がもとの文書のフォーマットを保つ必要がある。

これらの内、第一の方法は文書全体を翻訳するのではなく、文書の一部を翻訳する為の支援システムである。DTPシステムで作成した文書を他のシステムに移行する場合には第二の方法を採ることになる。しかし、今後の文書作成/編集環境の発展を考えると、最も必要とされる形態は第三の方法である。

以下、第三の利用形態について述べる。

## 3. DTP形式情報の保存方法

本方法でDTP文書を翻訳する場合、①書式情報(フォント・文書レイアウトなど)と文字列の混在したDTP文書から文字列を抽出する、②抽出した文書を翻訳する、③訳文を原文の形式に復元する、という手順をとる。

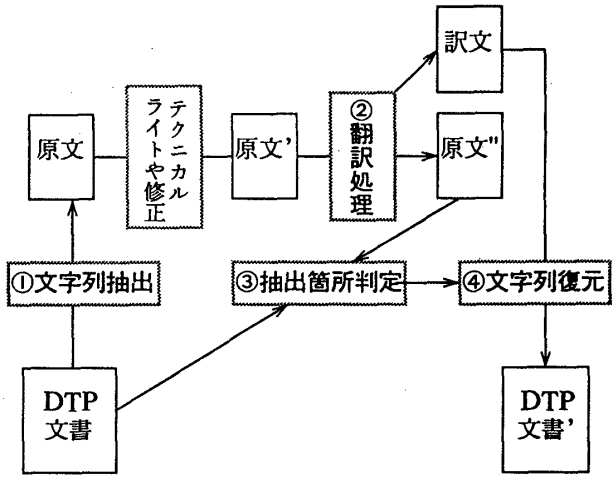
このため、DTP文書から文字列を抽出する際、形式情報と文字列の対応をいかに保存するか、またどの程度の形式を復元すべき情報として保存するかがポイントとなる。

こういった保存すべき情報を、抽出した文字列と共に記録する方法が考えられる。しかし、この方法は、抽出した文字列が特殊な構造を持つ必要があるため、その文書を扱うことができるシステムが限定される。DTP文書中から文字列を抽出した後、文書のテクニカルライトや修正が行われる場合がある。こういった修正に使用できるシステムが限定されることはシステム拡張性およびユーザインターフェースの立場から好ましくない。

従って、本方式では以下の手順を採用する(第一図)。

①DTP文書から文字列のみを抽出する。
②文書の翻訳を行なう(翻訳前にテクニカルライトや修正が行われる場合がある)。
③翻訳後、原文とDTP文書中の文字列を比較し、文書の抽出箇所を決定する。
④翻訳システムが記録している原文・訳文の対応をもとに、原文の抽出箇所の情報を翻訳結果に反映する。

本方式においては、復元できる形式情報は翻訳システムの原文・訳文の対応付けの単位に依存する。例えば、原文・訳文の対応が文単位行われていれば文以上の単位(文や段落単位)で付加されている情報の復元が可能である。




第一図 処理の流れ

Machine Translation System which Saves the Format of DTP-Documents
Etsuo ITOH, Kimihito TAKEDA, Hideki HIRAKAWA and Shin-ya AMANO
TOSHIBA Corp.

一方、形式情報を文字列に対応させて記憶しない方式においては以下の問題点がある。

第一に、DTP文書の文字列と翻訳後の文書の比較をいかに行って原文の抽出箇所を決定するかという問題である。つまり、修正された部分はDTP文書中の文字列とは一致しないため、単純な比較では原文の抽出箇所の決定ができない。

この問題は、本方式では一致する文同士をキーにして、その間にある一致しない文の対応付けを行う方法により解決した。

対応付けに用いるキーは一致する文の他に、改行などの文書中の特徴や特別な属性を付加された語句を用いることができる。なぜならば、こういった情報は、文書の全体構成やキーワードに係わるため、修正によって削除されることは希であると考えられるためである。

第二の問題は、原文と訳文では語順が異なることに起因し、文の一部に対して施された情報には対応できないという点である。例えば、連続する語句に下線が付加された原文を翻訳する場合を考える。この場合、原文で下線が付加された語句の訳語が翻訳結果では連続するとは限らない。このため下線情報の復元ができなくなる。

この問題は、周囲と異なる情報を持つ語句はテクニカルタームであり、訳語は連語となり訳文中に連続して出現する場合が多いという文書上の特徴によってかなりの部分が解決できる。さらに、上記の特徴により解決できない部分も、訳文が日本語など語順の制約が少ない言語の場合は「原文で特徴付けられた語句を連続して生成する」といった手法などにより解決が可能である。

## 4.評価

上記の方法でDTPシステム(AS-Documents)の文書を、形式情報を保存して翻訳する実験システムを作成した。翻訳に用いたASTRANSACは原文と訳文を文および単語単位で対応付けて記録することができるため、文単位・単語単位で文書情報を回復することできた。

このシステムで実験を行った結果、原文の抽出後大幅に修正された文に対しても有効であることが実証された。本システムで翻訳を行った文書の例を第二図に示す。

## 5. まとめ

本方式の実験はAS-Documentsについて行ったが、この他、TEXの文書など文字列と形式情報が混在している文書に適用可能である。

今後、原文と訳文の長さが異なるために発生する、①ページ溢れや図表と参照ページのずれ、②図表中の文字が長くなった場合の図表が乱れ、などを自動的に調整する方式を検討する。

**2.2 意味係り受けグラフの生成**

ここで対象とする処理は、構文/意味解析処理である。このため、入力は、文節の列である。意味係り受けグラフは、a.構文解析、b.係り受け候補生成の2つの過程より生成される。図2に、意味係り受けグラフの生成を含む解析処理の過程を示す。

**2.2.1 構文解析**

構文解析は、文節の列を入力し、曖昧性内蔵型の中間的な木構造(ここでは、G木と呼ぶ)を1つ出力する。解析には、文脈自由文法ベースのパーサを用いている。この木構造の例を図2.1aに示す。この木構造は、次の性質を持っている。

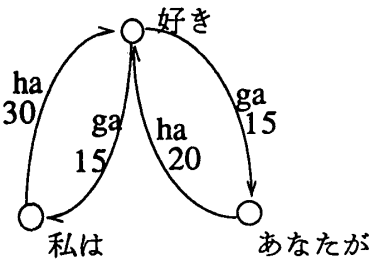



図1 「私はあなたが好きです。」の意味係り受けグラフ

**2.2 Generation of a semantic-relation graph**

Here the processing in question is syntactic / semantic analysis processing. Therefore, the input is a sequence of clauses. The semantic-relation graph is generated by the following two processes : a. syntactic analysis b. generation of other possible relations. The process of the analysis processing, including the generation of a semantic-relation graph, is shown in Fig. 2.

**2.2.1 Syntactic analysis**

In syntactic analysis a sequence of clauses is inputted and one intermediate tree structure of the ambiguity

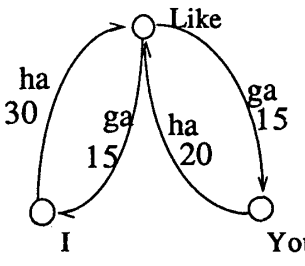



Fig. 1 Semantic-relation graph of "I like you"

翻訳前のDTP文書 → 翻訳後のDTP文書

第二図 文書情報を保存した翻訳の例