クライアント／サーバ型機械翻訳システムの学習方式

5R－10

伊藤悦雄　野村浩一　武田公人
（株）東芝　東京システムセンター

１．はじめに

２１世紀に向けて、ネットワークの整備が急速に進むと同時に高機能低価格の個人情報機器（ＰＤＡ）の出現で、誰でも、いつでも、どこでも、様々な局面で種々の情報にアクセスできる社会インフラが整備されると予想される。

このような状況において、次世代の機械翻訳システムの形態として考えられるのが、無線ＬＡＮなどを利用したクライアント／サーバ型の機械翻訳システムである。ここで問題になるのが、そのようなシステムにおける学習方式、及び翻訳結果が曖昧であるときの、非決定性データの送信方法である。本稿では、上記システムを実現するにあたり必要になる技術について検討する。

２．方式と問題点

本節では、クライアント／サーバ型の機械翻訳システムにおける学習方式と通信方式について、従来の問題点とその対策について述べる。

２．１．学習方式

従来の機械翻訳システムにおいては、原言語の語句に対して複数の訳語候補をオペレータへ提示し、オペレータが最も適切であると判断した語を学習していた。学習方式としては、(1)原語と選択された訳語をペアにして記憶する方法、(2)選択された訳語を辞書上で第一訳語とする方法、(3)全ての訳語に対して、その訳語を設定している語彙規則の番号と、その語彙規則で設定している訳語中の何番目に位置するかを示す番号をつけて、それらを記憶する方法等が知られている。

しかし、機械翻訳の辞書には、原語に対応する訳語だけではなく、訳し分け規則のような語彙規則も蓄積されている。見出し語「take」の目的語が「bath」である場合は訳語を「入る」に、目的語の意味マーカが「乗り物」である場合は訳語を「乗る」に設定するような規則である。そのため、前述の(1)や(2)の学習方式では原語に対して１つの訳語しか学習できないため、「take」に対して、例えば「握る」を学習すると、「take」が原文中に出現する度に「握る」と訳してしまっていた。すなわち、学習前は訳し分け規則によって、原文「I take a bath.」は「私は風呂に入る。」、原文「We took a bus.」は「私達はバスに乗った。」と訳出されていたのに、学習によって訳文がそれぞれ「私は風呂を握る。」、「私達はバスを握った。」となってしまう訳である。また、語彙規則の種類によっては、ある制約条件（目的語の種類・属性、動詞の種類・属性など）が満たされた時のみに構造変換を伴う訳出を指示するものあるが、そのような規則によって作られる構造を持てない語が訳語学習された場合、文法的に誤った翻訳結果が生成されてしまっていた。

（３）の学習方式では、原文に依存しない方式であり、語彙規則毎に学習を行えるので、訳し分け規則が反映されないという問題は解決できる。しかし、語彙規則と訳語の番号をユニークにする必要があるので、辞書がバージョンアップする度に全ての学習辞書内の番号を変更する必要性があり、実用的ではなかった。

以上の問題点を踏まえ、訳語の学習方式として、オペレータによって選択された訳語とその訳語に対応する見出し語及び適用された語彙規則の最初に記憶されている訳語とを対応付けて記憶する方式を提案した。この学習方式では、以降の翻訳において見出し語に対応する訳語が語彙規則に対応する最初の訳語と一致する場合に、対応付けて記憶している選択された訳語と入れ替える。

２．２．通信方式

自然言語処理では、必ず曖昧性のあるデータが存在する。前述の学習結果もその一つである。状況情報の利用や語彙知識を豊富に持ったとしても、言語処理に特有の曖昧性を解消することは困難である。このため、自然言語処理のインタフェースには曖昧性のあるデータを扱う枠組みが必須である。

Leaning for Client/Server-Model Machine Translation System
Etsuo ITOH, Koichi NOMURA and Kimihito TAKEDA
TOSHIBA corp.

クライアント／サーバ型では、通常、決定性データの転送のみが実行可能である。すなわち、処理要求を行ったクライアント端末に対して、サーバ側は翻訳結果を一意に決定してから転送する。そのため、クライアント端末からは、翻訳文の別解釈などが得られなかった。

サーバがクライアントの状況を常に把握し、次候補要求があったときにそれに応じて他のデータを転送する、という方式（ステートフル方式）の場合は別解釈などを得ることはできるが、サーバ／クライアント間で頻繁に通信が行われるため、例えば通信障害などによってクライアントの実際の状況とサーバが把握しているクライアントの状況との間に食い違いが生じると、動作不良を起こしていた。また、この方法では、１回の通信量は少なくても、問い合わせ回数が増加するので、ネットワークの負荷の増加やサーバにおける処理の複雑化という問題があった。

以上の問題点を考慮し、非決定性データを含むデータの転送を効率良く行うため、以下の通信方法を検討した。

- サーバはクライアントの状況を監視せず、クライアントへデータを送信した後は、サーバはそのデータに関して関知しない。
- サーバは非決定性データなども含む翻訳結果をクライアントへ送信する。
- クライアントは送信されたデータが非決定性データを含んでいる場合は、それを展開し、見やすい形でオペレータへ提示する。

本方式（ステートレス方式）を用いると、ステートフル方式と比べ１回当たりの通信量は増加する。しかし、ステートレス方式では、毎回サーバに対して問い合わせが必要な曖昧データ候補を同時に送信してしまうため、サーバとの通信回数を減少させることができるという特徴を持つ。通信内容がテキストデータであり動画像などと比較してかなり少量の通信量で対処できる上、サーバが多数のクライアントとセッションを持つ場合には、割り込みによるＣＰＵパワーの分断が無視できなくなるため通信回数の減少はシステム全体の能力向上に有効な手段である。また、前節で述べた学習方式を用いるためには、語彙規則の第一訳語を翻訳結果と同時にクライアントへ転送することで対応できる。本方式を用いた通信データの例を以下に示す。

翻訳要求（クライアント->サーバ）

He has an apple.

原文解釈結果（サーバ->クライアント）

He(1/he/Prop)has(2/have/vt)an(3/a/det)apple(4/apple/n).(5/./punc)

翻訳結果（サーバ->クライアント）

彼(1/{彼|やつ|あいつ}/彼)は(6/{は|が}/は)リンゴ(4/{リンゴ|りんご}/りんご)を(7/{を}/を)持っている(2/{持っている|持つ}/持つ)。(5/{。|．}/。)

この例において、原文解釈結果では、原文の表層文字列の他に、訳語との対応を取るための番号、原形、品詞を、翻訳結果では、第一解の他、原文との対応を取るための番号、他の訳語、語彙規則における最初の訳語を転送している。アプリケーションでは、このデータより第一解を集め、第一翻訳結果として提示したり 他の訳語の提示を行ったりすることができる。学習に際しても、原文と翻訳結果に於て同じ番号の語句を参照することにより、原形語彙規則の最初の訳語、学習する訳語などを得ることができる。

3．結論

学習方式としては、通常形態の翻訳システムへも適用できる訳語学習の方法を提案した。新方式では、学習によって語彙規則が反映されなくなるという従来からの問題点を解決した。

通信方式としては、曖昧性を含んだデータも含めた送信を行うことにより、訳語選択などの度に通信を行う必要がなくなるので、通信によるサーバへのＣＰＵ割り込みを減少させることができる。クライアント側の処理は、割り込みは増えるがテキストデータであるため、負荷の増大は問題ないレベルに留まると思われる。

4．おわりに

本稿では、クライアント／サーバ型機械翻訳システムに用いる学習方式とそれに適した通信方式について述べた。今後、機械翻訳以外の自然言語処理への適用の検討を行う予定である。

参考文献

[1] 江原他、「自然言語処理技術の応用」、情報処理学会誌vol.34, No.10, pp1240-1296, 1993

[2] 長尾他、「自然言語処理における曖昧さとその解消」、情報処理学会誌vol.33, No.7, pp746-756, 1992