*形態素解析プログラムANIMAの設計と評価

1C－5

†櫻井博文, ‡久光徹
（株）日立製作所　基礎研究所

1．はじめに

近年CD－ROMやWWWをはじめとする大規模な電子化文書をオンデマンドで解析したいという要求から、形態素解析にも高速性が求められている。我々は、高速・汎用な形態素解析プログラムの作成・公開を目指してANIMAを開発した。本発表ではANIMAのいくつかの特徴を簡単に紹介し、特に単語辞書実装方法と解析処理各部の速度について報告する。

2．ANIMAについて

ANIMAは単語辞書と接続コストテーブルを使った横形探索に基づく作表型アルゴリズム[1]により解析を行う。既存の様々な電子化辞書・文法の利用を可能にするため、ANIMAでは以下のような単語辞書と接続コストテーブルの中間記述形式を規定している。

単語辞書::=辞書項目|辞書<CR>辞書項目
辞書項目::=見出し文字列<空白>記載内容
記載内容::=単語コスト<空白>接続関係情報</>拡張部
単語コスト::=実数値
接続関係情報::=文字列のk項組(kは固定)
拡張部::=改行以外の文字からなる任意文字列
接続ｺｽﾄﾃｰﾌﾞﾙ::=接続ｺｽﾄﾃｰﾌﾞﾙ項目|接続ｺｽﾄﾃｰﾌﾞﾙ<CR>接続ｺｽﾄﾃｰﾌﾞﾙ項目
接続ｺｽﾄﾃｰﾌﾞﾙ項目::=一般化接続関係情報</>一般化接続関係情報</>接続コスト
一般化接続関係情報::=接続関係情報の0個以上の項目を*で置換し、他の0個以上の項目の先頭に<!>を付加してできるk項組
接続コスト::=実数

ただし、<CR>は改行文字、<空白>は空白文字、</>はスラッシュ、<!>はエクスクラメーションマークを表す終端記号。

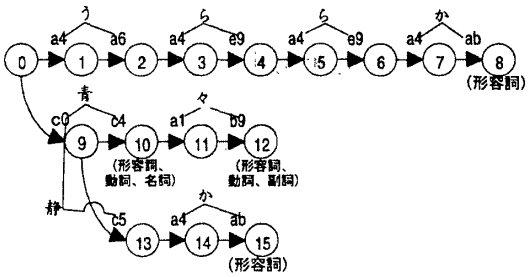
既存の辞書・文法をこの中間形式に変形することで、ANIMAで利用可能になる。ANIMAでは活用語尾のための特別なテーブルを用いない。活用語は語幹と語尾に分けて辞書に登録する。例えば、“走る”という動詞の基本型は以下のように記述される。

単語辞書
走 0701383 100＊ 動詞 ＊ 子音動詞ラ行 ＊ 不変部
る 0643171 0＊ 動詞 ＊ 子音動詞ラ行 基本形 変化部
接続コストテーブル
動詞 ＊ 子音動詞ラ行 ＊ 不変部/＊ 動詞 ＊ 子音動詞ラ行 ＊ 変化部/0

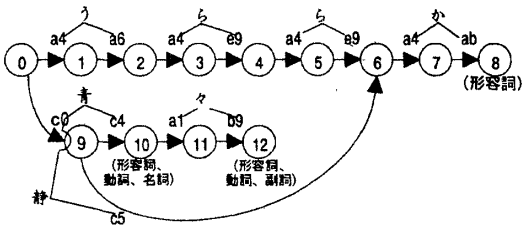
3．ANIMAの単語辞書実装方法

ANIMAの単語辞書はTRIE表現[2]の木の重複部分を併合し、それをダブル配列[3]を用いて実装した。

TRIEの各リンクは1バイトのデータでラベル付けされ、各ノードはルートノードからの経路上のリンクのラベルを並べた文字列データに対応する。各表記の品詞列は対応するノードに付記した。下図はその一例である。（日本語文字列にはEUCコードを用いている。丸がノードを矢印がリンクを表す。丸の中の数字はノード番号、矢印の上の数字がリンクのラベル、丸の下が品詞列である。リンクのラベル2つごとに対応する日本語文字を付記した。）



このようなTRIE表現では同一の部分木を併合することによりノード・リンク数を減らすことができる。上図の例の場合、ノード6の子の木とノード13の子の木を併合できる。下図は併合後のグラフである。



Juman3.0beta[4]の単語辞書を上記に述べた方法で実

---

* Design and Evaluation of Japanese Morphological analyzer “ANIMA”.

† Hirofumi Sakurai(Email:hirofumi@harl.hitachi.co.jp), Hitachi Ltd., Advanced Research Laboratory
‡Toru Hisamitsu(Email:hisamitu@harl.hitachi.co.jp), Hitachi Ltd., Advanced Research Laboratory

装した場合、ノードの数は併合により TRIE 表現の半分以下になった(併合前 1,457,507 個、併合後 699,246 個)。また、このグラフデータをダブル配列を用いて実装したところ、およそ10Mバイトになった。この大きさは最近のほとんどの計算機で利用できる主記憶より小さい。そこで、ANIMA ではこの辞書データを実行時にメモリに読み込んで動作することが可能となった。

4．ANIMA の速度評価

ANIMA は以下のようなマシンで動作が確認されている。各マシンでの処理速度を以下に示す(UNIX は CPU タイムを、MS-Windows は動作に要した時間を計測して1秒あたりに処理したバイト数を計算した。)

機種名、OS 名、CPU、クロック周波数	処理速度(バイト/秒)
DEC Alpha Server 8200 5/300, Digital UNIX 3.2C, DECchip21164, 300MHz	18,677
SUN Ultra-1, Solaris 2.5.1, Ultra SPARC 167MHz	7,919
Hitachi 3050RX-440, HI-UX, PA-7200, 100MHz	4,827
SUN SS-10/512, Solaric 2.5.1, Super SPARC, 50MHz	2,912
PC 互換機, MS-Windows95, Pentium 100MHz	3,698

これは Juman3.0beta のおよそ 10 倍、茶筅*や Juman3.1†とほぼ同じ処理速度である。一例として、Alpha server8200/300 上では新聞記事一年分約204 M バイトが3時間 3 分で解析できた。

実行時の各処理部分の所要時間は以下の表の通りである。(Alpha Server8200 5/300 で新聞記事 84,110バイトを使って計測した。表中で"割合"とは全体に対する所要時間の割合のことである。)

	所要時間(秒)	割合(%)	呼び出し回数
連接コスト計算	40.47	37.7	39,970,658
解析表の作成・参照	31.96	29.8	55,453,696
単語辞書の検索	13.19	12.3	9,272
その他	21.77	20.3	-

単語辞書の処理時間は処理全体に対して 12.3%と小さい。対して連接コスト計算・解析表作成・参照はその呼び出し回数が多いため時間がかかっている。これは解析表中の各文字境界で、その前後の形態素数の積の回数コスト計算をおこなうためである。

* http://cactus.aist-nara.ac.jp/lab/nlt/chasen.html

† ftp://ftp@pine.kuee.kyoto-u.ac.jp/pub/juman/juman3.1.tar.gz

5. ANIMA の利用について

ANIMA は個人または研究用の利用目的で自由にダウンロードし利用可能である。ANIMA の以下のような使いやすさを意識した特徴をもつプログラムである。

- C言語と標準的なライブラリ関数を用いて記述されており、ほとんどの UNIX マシンや MS-Windows95/NT でごく僅かの設定によってコンパイル・実行が可能である。
- 一行一文でない形式のテキスト(句読点が "。、")の入力に対しても文を切り出して処理できる。
- ディレクトリ内ファイルの一括処理が可能。
- EUC, JIS, SJIS のテキストを処理できる。

使用方法、使用条件など詳細は ANIMA のホームページ
http://www.harl.hitach.co.jp/~hirofumi/anima/index.html
を参照し、ぜひ一度利用していただきたい。

6. おわりに

単語辞書は、TRIE 表現の木の重複部分を併合し、それをダブル配列を用いて実装した結果、高速かつコンパクト(60万エントリが10Mバイト)となった。また、処理速度は約 18,677 バイト/秒(DEC Alpha server 8200/300)で、うち辞書引きに要した時間は12%である。

ANIMA では解析に横形探索による解析方法を用いた。この方法は速度の点では深さ優先の探索を行うものに比べて劣っているかも知れない。今後は処理速度を改善するためには処理途中の部分解の数や連接コストの計算回数を減らすなどにより解析木の大きさを小さくする工夫が重要であると考えられる。また、精度・使いやすさの点での改良も継続しておこなっていく必要がある。

参考文献

[1]久光徹 他, ゆう度付き形態素解析用の汎用アルゴリズムとそれを利用したゆう度基準の比較, 電子情報通信学会論文誌 D-II Vol.J77-D-II No.5(1994), pp.959-969

[2]G.H.Gonnet, et. al.:Handbook of Algorithms and Data Structures In pascal and C Second Edition. Addison-wesley, pp.133-137

[3]Aoe, J. et. al.:An efficient digital search algorithm by using a double-array structure, Proc. Of the 12th Int. Comput. Softw. & Appli. Conf., Chicago(1988).pp.472-479

[4]松本裕治 他, 日本語形態素解析システム JUMAN 仕様説明書 version 3.0beta