研究ノート

英文単語統計と品詞解析への試み

福井正康，光平直嗣*

福山平成大学経営学部経営情報学科
*福山平成大学大学院経営学研究科経営情報学専攻

概要

この研究ノートでは我々が試作した英文単語統計と品詞解析のプログラムについて、その構造と機能を解説する。単語統計では辞書を用いて原形で単語出現回数の統計が取れること、品詞解析では各単語に品詞のタグが付けられることを目標にしている。一応の骨組みはできあがったが、今後の問題点も数多く明らかになった。

キーワード

英文，単語統計，品詞解析，プログラム

１．はじめに

英語教科書の時系列的な変化や国別の比較を行う際に、総単語数や新語の密度などを比較することが考えられている [1]。そのためコンピュータを利用して特定の単語の出現回数が計測されるようになったが、同じ綴りで異なる品詞のものや同じ意味で異なる綴りのものの扱いはこれまで人手に頼ることが多かった。最近この処理にもコンピュータを導入することが試みられているが、本来は自動翻訳 [2] などで十分な検討を加えられているはずの問題である。これは単に研究者間の交流が少なかったことが原因か、我々はこの格差に不思議さを感じ、試作的なプログラムを通してこの問題がどのように扱われ、行く行くはどのようにして自動翻訳に結びついて行くのか考えてみたいと思うようになった。手始めに、我々は各単語を原形に直して単語数を計測することと各単語に品詞のタグを付けることを目標にプログラム作りを開始した。ここではその製作過程の中で、一応の方向性や問題点が見えてきたので研究ノートとしてまとめておくことにする。

プログラムは実行中にいくつかのファイルを参照する。特に重要なものの１つは辞書ファイルで、単語ごとに原形やその変化形のデータが含まれている。もう１つは採用ルールファイルで、慣用句や頻度の高い品詞の並びなどが記述されており、品詞を決定する際には積極的に採用する。３番目は棄却ルールファイルで実用上ありえない品詞の並びが記述されており、品詞決定の候補から外すときに利用される。我々はこれらをうまく組み合わせることで品詞の解析をかなり進めることができるのではないかと考える。プログラムは Visual Basic Ver.6 で書かれ、後に Visual Basic.NET に書き換えられた。

２．プログラムの構造

ここではプログラムの構造を機能、処理の流れ、利用するファイルの構造などから見てみよう。プログラムは、図１のようにいくつかのファイルを読み込んで動作する。

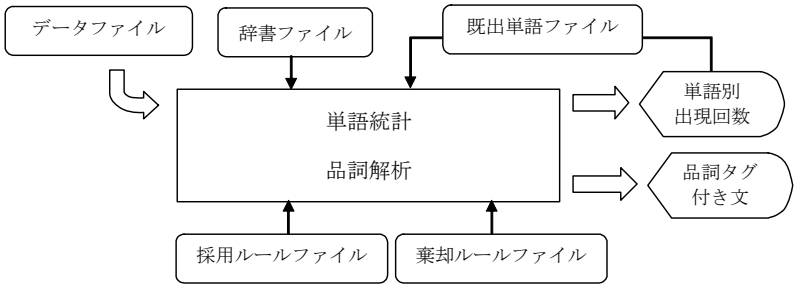


図１ システムのファイル構成図

解析を加えて行く英文のファイルをデータファイル、単語辞書に相当するものを辞書ファイル、辞書を用いて得た品詞を選別するためのデータが採用ルールファイルと棄却ルールファイルで、前者は慣用句などを元にした品詞の並びを集めたもの、後者は品詞の並びの可能性のないものを集めたものである。それぞれの書式については後に説明する。

単語の出現回数を求める単語統計では、原形が同じなら同じものとみなすか、綴りが違えば違うものとするか選択可能とする。また統計を白紙の状態で行うか、既出単語ファイルを用いてすでに他のデータファイルで出た単語は統計に含めないか選べるようにする。この既出単語ファイルは、他のデータファイルを元にしたシステムの出力結果をそのままファイルに保存して利用してもよい。最終的に、単語統計では単語別の出現回数の表、品詞解析では解析の過程を表すデータと品詞タグ付の英文が出力される。

これらの単語統計と品詞解析の処理の過程を見ると、図 2 と図 3 のようになる。

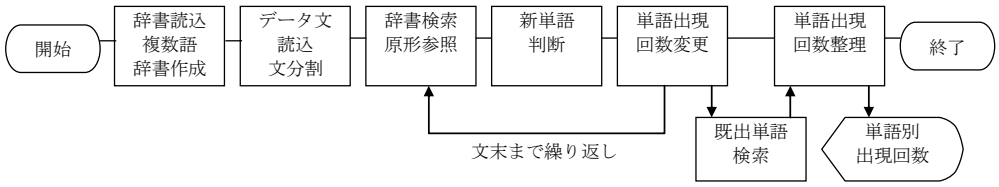


図 2 単語統計の過程と出力

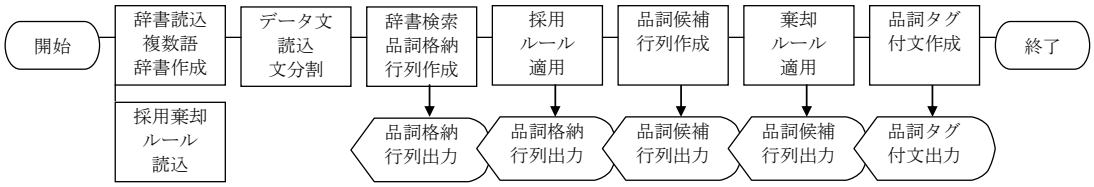


図 3 品詞解析の過程と出力

システムが立ち上がると既定の辞書ファイルと採用ルールファイル、棄却ルールファイルを読み込み、辞書配列と 2 つのルール配列が作られる。データ文を読み込むと余分な空白やタブなどを取り除き、1 文ずつ区切って配列に格納する。単語統計や品詞解析が始まると、最初に選ばれた 1 つの文が単語に分割される。

単語統計には単語の原形を用いて集計する場合と綴りが違えば別の単語として集計する場合の 2 通りがあるが、原形を用いる場合は辞書を使って原形を参照する必要がある。その後新しく現れた単語かどうか判断しながら同一の単語ごとに出現回数を求め、単語をアルファベット順に並べる。既出ファイルを参照する場合は、このデータから既出ファイルに含まれる単語を取り除いて出力する。

品詞解析では最初に辞書を使って各単語の品詞を参照する。その際同一の単語に複数の品詞が対応する場合は、すべてを候補として取り上げる。その後、採用ルールを適用して、単語並びの中で品詞の確定できるものは確定する。この段階で候補をかなり減らすことができるが、十分とはいえない。次にそれらの候補を文に割り当て、１単語１品詞が割り当てられた文の候補を示す。さらに棄却ルールを参照し、棄却ルールが適用できる品詞の割り当ては候補から外す。これらの過程はそれぞれの結果を逐次表示しておく。最後に品詞のタグを付けて文を表示する。

ここまではプログラムの流れを中心に述べてきたが、ここからは具体的な辞書ファイルやルールファイルの形式を示しておこう。以後サンプルに挙げるこれらのファイルは現時点の検討段階のものの一部で、まだ間違いが含まれるかも知れない。

最初に辞書ファイルについて見ておこう。この書式は、表１のようになっている。

表１ 辞書ファイル（一部）

'特殊記号 'アポストロフィｓ as,¥s 'カンマ cm,¥c 'be 動詞省略形 be,¥b '記号 pd,. qs,? ex,! '特殊単語 be,be,is,being,was,been be,be,are,**,were,** do,do,does,doing,did,done hv,have,has,having,had,had le,let,lets,letting,letted,letted	'一般 ar,a ar,an ar,the pr,this,these pr,that,those co,that pr,it,its pr,I,my,me pr,we,our,us pr,you,your,** pr,he,his,him pr,she,her,her pr,they,their,them pp,in pp,on pp,at pp,of pp,to	pp,for pp,by co,but co,and ad,much,more,most no,share,shares vt,share,shares,sharing,shared,shared no,experience,experiences vt,visit,visits,visiting,visited,visited vi,visit,visits,visiting,visited,visited no,visit,visits pn,Canada pn,Canadian aj,Canadian aj,late,latter,last ad,last no,last pn,United^States pn,U.S.

基本的に１単語１品詞を１行にし、「品詞,見出し語（原形）,変化形１,変化形２,・・・」のように半角のカンマを区切り記号として続ける。原形と変化形の順番は、名詞なら「単数（原形）,複数（複数がない場合は原形のみ）」、人称代名詞では単数形と複数形別に「主格,所有格,目的格」、形容詞や副詞なら「原級,比較級,最上級」、動詞なら「原形,三人称単数現在,現在分詞,過去,過去分詞」などである。基本的な品詞は表 2 のように

表２ 品詞コード

固有名詞(proper noun)	pn
名詞(noun)	no
代名詞(pronoun)	pr
冠詞(article)	ar
形容詞(adjective)	aj
副詞(adverb)	ad
自動詞(intransitive verb)	vi
他動詞(transitive verb)	vt
助動詞(auxiliary verb)	au
前置詞(preposition)	pp
接続詞(conjunction)	co
間投詞(interjection)	in

品詞コード２文字で表すが、疑問符や感嘆符などにも特殊な名前を付けている。また、be とか do など特別視すべき単語には特殊な名前を付けている。

変化形のところでそこの記述を避けたい場合は、「**」で代用する。United States などの複数の単語からなる語は、空白の代わりに「^」を使用する。また辞書の中では先頭にアポストロフィを付けることにより、コメント文にできる。カンマやアポストロフィなど編集記号に使われる文字について、データ文中のそれらの文字はすべて辞書ファイルで指定する特殊な記号に書き換えられる。現在プログラムの中で辞書ファイルのデータは２次元配列に取り込まれるようになっているが、検索スピードを考えて木構造などのデータ構造を持たせることも重要であろう。

次に品詞の候補を絞るための採用ルールファイルについてその一部を表 3 に示す。

表 3 採用ルールファイル（一部）

'ルール文法 '<a> 語　<b>語先頭 '<p>品詞　<q>品詞先頭 'デフォルト　<a> '基本的に原形で記述すればよい '慣用句ルール either * or -> co1 * co1 either * * or -> co1 * * co1 either * * * -> co1 * * * co1 one of -> nm1 pp1 one out of -> nm1 ad1 pp1 last <p>no <p>pd -> aj3 no1 pd1 back in -> ad1 pp1 each other -> pr1 pr1 in English -> pp1 no1 in Japanese -> pp1 no1 one out of * -> no1 ad1 pp1 no1 as long as -> pp1 aj1 pp1 as many as -> pp1 aj1 pp1 as soon as -> pp1 ad1 pp1	and so on -> co1 ad1 pp1 how many -> ad1 aj1 ¥c so -> cm1 ad1 <b>so -> ad1 last <p>no -> aj3 no1 '品詞採用ルール many other -> aj1 aj1 other <p>no -> aj1 * <p>be <p>vt5 -> * vt5 <p>be <p>ad <p>vt5 -> * * vt5 <p>hv <p>vi5 -> * vi5 <p>hv <p>ad <p>vi5 -> * * vi5 <p>hv <p>vt5 -> * vt5 <p>hv <p>ad <p>vt5 -> * * vt5 <p>le * <p>vi -> le1 * vi1 <p>le * <p>vt -> le1 * vt1 <p>vt5 by -> vt5 pp1 <p>ad1 <p>aj1 -> ad1 aj1 <p>pr1 <p>vt1 <p>no -> pr1 vt1 no3 <p>pr1 <p>vt4 <p>no -> pr1 vt4 no3 when <p>pr1 -> co1 pr1

この採用ルールファイルも１行１ルールで構成され、大きく分けて、慣用句を採用する部分と品詞の並びから採用する部分からなる。それぞれ文中で使われるものと文の先頭で使われるものとに分けられ、先頭に<a>, <b>, <p>, <q> の記号を付けることによって分類される。先頭にこれらの付かないルールについては、デフォルトとして文中で使われる慣用句として扱われる。ルール中、矢印記号「->」の左辺の語句および品詞並びが出てきたら、右辺の品詞ならびにする。また、左辺の「*」は任意の 1 語を表し、右辺の「*」はその部分について品詞の限定は行わないことを示す。

最後に、品詞候補をさらに絞り込むために使われない品詞並びを集めた棄却ルールファイルを表 4 に示す。

表 4 棄却ルールファイル（一部）

'以下の並びがあったら削除 pp pr1 aj no be vi be vt1 be vt2 be vt4 hv vt1 hv vt2 hv vi4 hv vt4 vi1 no1 vi4 no1 vi5 no1 vi pn vi pr	aj pp ar vi ar vt ad pn pr1 no pp pp vt aj pd vt aj qs vt pn ad vt pn vi pr1 vt5 pp pr1 vi5 pp pr1 vi5 no1 vi5 pn1 vi5 pr1 vt5	no1 vt5 pn1 vt5 '<top>に続くものが先頭から並んでいたら削除 <top> vi4 <top> vt4 <top> le2 <top> le3 <top> le4 <top> le5 <top> aj be <top> aj hv <top> aj vi <top> aj vt

棄却ルールファイルでも文中に出てきたら削除するルールと先頭に現れたら削除するルールに分けてある。ここで述べた辞書ファイルやルールファイルについてはまだまだ検討段階で、今後は大いに工夫して行かなければならない。

3．実行画面

ここでは実際の画面を見ながらプログラムの動きを説明する。プログラムを実行し、データファイルを入力すると図 4 のようなメニュー画面になる。

図 4 プログラム実行画面

メニューウィンドウの上部に表示されている英文は、データファイルの中の 1 文を取り出したものである。その英文の右下の 1/16 は 16 文中の 1 番目の文という意味である。詳細な品詞解

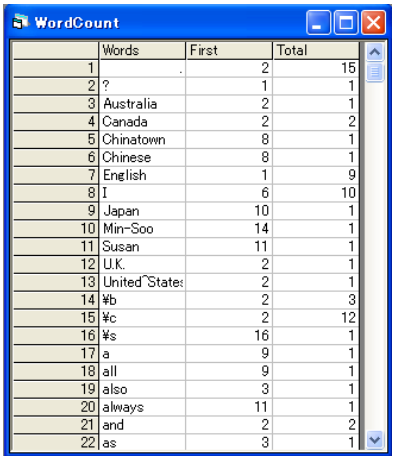
析と結果表示はこの表示されている文を対象に行われる。コマンドボタン「Top」、「Prev」、「Next」、「Last」はそれぞれデータ文の先頭、1 つ前、1 つ後ろ、最後を呼び出すボタンである。「Dicload」と「Ruleload」は、辞書ファイルやルールファイルに変更を加えたとき、再起動しなくても済むように再読み込みを行うボタンである。

コマンドボタン「SentRef」はデータ文の参照で、読み込まれた文が、段落で分けられ、文ごとに図 5 のように表示される。

図 5 データ文の参照実行画面

カンマなど辞書で特殊な記号を指定した単語はその記号に置き換えられている。この段階で、文が正しく分けられているかどうか確認しておくとよい。

次に単語統計については、コマンドボタン「WordCount」をクリックして、図 6 のような集計結果を得る。図の First は、最初に単語が登場した文の番号で、Total は単語が合計何度出現したかを表す。これらの集計を単語のそのままの形で行うか、一度原形に変換して行うかについては、オプションボタン「OrgTrans」によって選択する。また、図 6 のデータを保存しておくことによって、ここに現れている単語を除く統計を取ることができるが、それはオプションボタン「PrevWords」で選択する。この統計の中でも辞書で特殊形を指定した単語はそのまま表示するようにしている。



	Words	First	Total
1	.	2	15
2	?	1	1
3	Australia	2	1
4	Canada	2	2
5	Chinatown	8	1
6	Chinese	8	1
7	English	1	9
8	I	6	10
9	Japan	10	1
10	Min-Soo	14	1
11	Susan	11	1
12	U.K.	2	1
13	United States	2	1
14	¥b	2	3
15	¥c	2	12
16	¥s	16	1
17	a	9	1
18	all	9	1
19	also	3	1
20	always	11	1
21	and	2	2
22	as	3	1

図 6 単語統計の実行画面

品詞解析はコマンドボタン「Analyze」で表示されている 1 文について実行される。

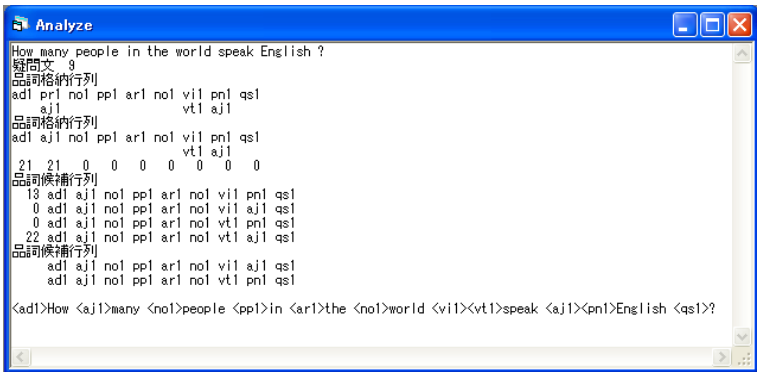


図 7 品詞解析実行結果

まず、文の種類を解釈し、次に品詞格納行列をそのまま表示し、その後採用ルールを適用し、品詞が限定された結果をもう一度品詞格納行列を表示して示す。その際、どのルールが適用されたか、ルールの先頭からの番号で表示する。但し、0 は適用がないことを表す。次にこの品詞格納行列を用いて文の品詞並びをすべて作り、品詞候補行列として表示する。その際、各行の先頭に棄却ルールの番号を付けておき、それらを適用して残った候補を再度表示する。最後に、この結果を各単語にタグとして付けて表示する。複数の候補がある場合は、タグを複数にすることにしている。この品詞解析をすべての文に連続的に実行した結果はコマンドボタン「SetTag」によって図 8 のように得られる。

図 8 連続的な品詞解析

4．おわりに

ここまで我々の試作プログラムを見てきたが、これらのプログラムには今後検討すべき課題がいくつもある。大きな問題のひとつは辞書のサイズで、現在の段階ではほとんど考慮されていない。これを一般に通用するような辞書にした際、プログラムのスピードがどうなるか予想

が難しい。高速化する工夫は必ず必要になろう。これは採用ルールファイルの慣用句についても同様である。次に、品詞解析の際の、採用ルールと棄却ルールであるが、ルールの数や規則の形を増やすにしても、このままの形で良いのか、多くのルールが入ってきた場合のルール間のバッティングは起こらないのかなど、少し考えただけでも問題はありそうである。また、単語統計にしても、現在は原形を使えるだけであるが、品詞が別の同じ綴りの単語も当然区別しなければならない。これには単語統計と品詞解析の 2 つの機能の統合も考えなければならない。

現在、単語統計や品詞解析に関するソフトウェアがいくつか欧米で開発されている。これらのソフトウェアに近づき、さらにここで考えた品詞解析の問題と自動翻訳との関係を考えるにはいくつもの壁を乗り越えて行かなければならない。これには相当の覚悟が必要であり、我々が取り組むべき問題か否か真剣に検討しなければならない。ここで紹介したプログラムはこれまで関わってきた社会システム分析用のプログラムとかなり違っていて、開発中は知的好奇心が大いに刺激された。このような興味深い問題を教えていただいた小篠敏明先生に心より感謝します。

参考文献

1) 小篠敏明, 日本の英語教育の課題と可能性 －歴史研究, 国際比較からの提言－, 拓殖大学論集（260）人文・自然・人間科学研究, 第 14 号, 93-113, 2005.
2) 田中穂積, 自然言語解析の基礎, 産業図書株式会社, 1988.