

研究ノート

Hum パッケージを用いた Robert Henryson の コンコードانس作成*

西 納 春 雄

0 はじめに

近年パーソナルコンピュータ（以下、PC）の処理能力が飛躍的に向上し、かつては大型汎用機（mainframe computers）に頼らねばならなかった複雑で高度なデータ処理が、個人の所有するコンピュータ上で可能になっている。人文分野への応用も、数値、文字列、画像処理を中心に盛んに行われ、プログラムの開発が進んでいる。これらのプログラムのうちで、テキストと並行して参照するための、KWICコンコードانس（Key World In Context, 行中央にアルファベット順に出現単語を並べ、左右にテキストを展開する形式のコンコードانس）を作成するプログラムは、文学語学研究者にとって、作品における語義確定に、また、語句の用例検索に利用価値が高い¹。

研究環境において書籍形態のKWICコンコードانسが、個々の作品あるいは作家ごとに手近に入手できることは、理想的ではあるが、実際には、コンコードانسの出版は採算がとれにくいため、限られた作家と作品についてしか出版されていない。このような状況を克服するために、研究者が自力でコンコードانسを意のままに作成できることが望ましい。しかしながら、参照に耐える書籍形態のコンコードانسを完成するには、ある程度のコンピュータリテラシーと、蓄積された技術が必要である。小論ではこのような現状をふまえて、まず、コンピュータを用いたコンコードانس作成の歴史を概観した後に、コンコードانسの材料としての電子テキストの特徴を考察する。次に、書籍形態のコンコードانسのあり方を、近年出版されたいくつかの中世

英文学作品のコンコーダンスを例にとって比較考察する。その後、中世スコットランド詩人 Robert Henryson の作品を材料に、テキストの電子化から始めて、書籍形態のコンコーダンスの作成までを実践的に解説する。

小論でコンコーダンス作成のために用いるプログラムは、1981年にカリフォルニア大学でUNIXコンピュータのために開発されたプログラム群を、後年MS-DOS（以下、DOS）に移植した、Humパッケージとよばれるプログラム群である。(Janus 1990)²。これはC言語で書かれたプログラム群で、コンコーダンス作成のための基本的な文字列処理を満足している。DOSへの移植当時は、PCの演算能力が十分でなかったために、これらのプログラムは十分に能力を発揮できなかった。現在では、PCの性能向上にともない、このプログラム群を用いて、きわめて高速な文章処理ができる。しかしながら、これらのプログラムは、テキストファイルを与えるだけで自動的に理想的なコンコーダンスを作成するわけではない。完璧な出力を求めて、処理のすべてをプログラムに依存しようとするれば、さらに複雑な補助プログラムの作成が必要となる。そこで、コンピュータが得意とする処理過程を推し量り、もっとも大量で高速処理すべきところをコンピュータに任せて、補助的な微調整を利用者の手作業によって加えるという融通性が大切である。

1 コンコーダンスとコンピュータ

作品の読解において、コンコーダンスは、語義を厳密に定義するために欠かせない。特に、KWICコンコーダンスは、もっとも基本的な索引として用途が広い。コンコーダンス的な索引の発生は、12世紀後半に遡る(Illich, 1993, p.97-98)が、本格的なコンコーダンスの成立は19世紀を待たねばならなかった。19世紀中葉から、特に聖書のコンコーダンスを作成する試みが始まった。当時のコンコーダンスの作成は、丹念に語句を紙片に記述し、整理し、転写するという気の遠くなるような作業で行われた。一例をあげるならば、F.J. Furnivalによって1871年に開始された、Geoffrey Chaucerの作品のコンコー

ダンスは、57年の歳月を経て、Tatlock & Kennedy(1927)の形で完成を見た。20世紀の中葉から、コンピュータ技術が発展するに伴って、この作業がコンピュータを利用したものに急速に取って代わった、1970年代から80年代にかけては、様々なアルゴリズムを駆使してコンピュータコンコードダンスを作成することが試みられた。この時期には、米国を中心として博士号取得のために、相当数のコンコードダンスが作成されている³。

Wisbery(1971), Jones & Churchhouse(1976), Lancashire & McCarty(1988), Hockey(1980, 1996)を参考にして、文学語学研究へのコンピュータの応用をふり返ると、コンピュータをコンコードダンス作成に応用した最初の試みは、Father Roberto Busa による、Thomas Aquinas の著作のコンコードダンスであった。1949年のことである。初期のコンピュータとしては、IBM 社の VM や VMS のオペレーティングシステム（以下、OS）を搭載した大型汎用機が、加えて70年から80年代にかけて、VAX や UNIX OS を搭載した DEC 社のミニコンなどが用いられた。当時はまだ市販パッケージなどの、人文系学問分野に即座に利用できるアプリケーションソフトが皆無の時代で、それぞれの利用者は、独自にプログラミング技術を取得することからはじめる必要があった。しかも、データは当初パンチカードで入力し、データの保存は磁気テープ、出力も印刷出力のみという、現在と比較すれば、恐ろしく手間のかかるものであった。しかしながら、このような試みは、それまで手作業で行われていたコンコードダンス作成の能率を、飛躍的に高めたばかりでなく、本来数値計算に用いられることを目的として作られたコンピュータを、文字列処理を通じて、人文系の研究に応用するための貢献を果たした。博士論文としてのコンコードダンスは、プログラムの開発に加えて、コンコードダンスが出版されにくい文学ジャンルのコンコードダンス充実に貢献した。

その後、1980年代から90年代にかけて、コンピュータの劇的なダウンサイジングと性能の向上が行われた。それとともに、一部の研究所や個人によって開発された文字列処理のアプリケーションプログラムが、一般に公開

されたり、市販されるに至った。このため、現在では、一般の研究者にも、語彙や文体の高度な分析処理が可能な時代が到来している。しかしながら、環境は整いながらも、実際の分析には、分析に耐える電子テキストの作成から始まって、言語学的な知識と統計上の処理が必要なため、本格的な試みは多くはない。

2 電子テキストの利便性

電子テキストには独自の利便性がある。電子テキストとは、基本的に、印刷された文字を電子化したものであるが、文字データと電子データとでは、その性質が大きく異なる。紙に印字として固定された印刷文字データと異なり、電子データは特有の優れた検索性と、随意に再編集可能な可塑性を持つ。このため、容易に検索し、並べ替え、編集して、コンコーダンスへと作りかえることが可能である。すなわち、この意味では、電子テキストは、従来の書籍媒体に存在した文字データ（元になる書物）を超えて、それから生成される様々な種類のコンコーダンスや索引をそのうちに内包した、全く新しいテキストであると言えよう（Ross, 1996, p.225）。

電子テキストは検索の自由度が高い。前方一致、後方一致、近接語の検索など、語形と文字間のつながりを自由に検索できる。従来のコンコーダンスでは、語頭からのアルファベット順に語彙を整列することはできても、語尾から整列させることは難しく、ましてやその語と特定の語の近接関係を自由に検索することは不可能であった。ところが電子テキストでは、検索プログラムを用いることで、たちどころに可能になる。

また、結果の出力においても、検索語のある行ばかりでなく、その前後の任意の行数を出力でき、作品別に、出現順に、アルファベット順に、など、利用者の要求に応じて様々な出力形式が可能である。さらに品詞や格などの構文標識を示す符号を付した（tag 付けされた）テキストを分析できる場合には、語の属性まで含めて分析することも可能になった。また、出力結果は

電子データであるので、数値で結果が得られるならば、これを統計処理に付すこともできる。こうして成立した計量文体学により、著者推定、特定作家や作品の文体研究が可能になっている (Holmes, 1994; Tabata, 1994)。

3 電子テキストの制約と、書籍形態のコンコードダンス

電子テキストを扱うことは、一方で様々な制約をとまなう。まず、分析に足る電子テキストの作成が課題となる。一般に書籍媒体から誤りのない電子テキストを作成することは、コンコードダンス作成過程のおよそ90%以上の労力が必要とされる。近年は出版に先立って電子テキストが作成されるが、著作権の制約もあり、コンコードダンスを作成する場合にはこれとは別個に、新たに電子テキストを準備しなければならない場合が多い。ただし、厳密な正確さと、版を限定しなければ、英米文学関係では、主要な作品のほとんどがインターネットを通じて、また、CD-ROM の形で入手できるようになっている⁴。

電子テキストは、その利用環境に大きな制約を受ける。電子テキストは、コンピュータ、しかも特定のプログラムの利用可能な状態にあるコンピュータが、物理的に存在する場所で、しかも利用すべく設定が行われた状態でしか利用できない。また、電子テキストを扱う利用者は、分析するプログラムの特性に通暁していなければならない。これは当然のことであるが、実際には、プログラムを作成し、分析できる者(プログラマ)が、分析結果を必要としている者(研究者)と異なる場合がしばしばである。一方で、結果の出力もまた問題を含んでいる。コンコードダンスをわずか十数インチのディスプレイに表示する場合には、同時に複数のページを並行して参照するような一覧性に欠け、長時間の利用は、利用者に多大な疲労を与える。

これに比較すると、書籍形態に印刷されたコンコードダンスは、参照する際の閲覧環境を選ばず、一覧性にも優れている。また、書物の持つ物理的な手触りは親しみやすく、長時間の閲覧にも向いている。しかし一方で、書籍形

態のコンコーダンスは、しばしば大部になりがちである。しかも、商業出版物としては採算がとれない場合が多いため、実際に出版に付される作家、作品は多くない。そこで、電子テキストと書籍コンコーダンス双方の長所を生かすために、研究利用に耐えるコンコーダンスを作成し、電子ファイルとして保存して、ディスプレイで閲覧したり、あるいは必要に応じて書籍形態に印刷出力して利用するのが、PC とコンコーダンスプログラムの、現在におけるもっとも効果的な利用法であるといえる。

4 コンコーダンスプログラム

現在入手できるコンコーダンスプログラムは、コンピュータの画面上に検索語を入力すると、その検索語に関する結果を即座に画面上に得ることができるプログラム（インタラクティブ・コンコーダンス）と、テキスト中のすべての語を対象に文字列を処理し、結果を長大なコンピュータファイルの形で出力するプログラム（ノンインタラクティブ・コンコーダンス）に分類することができよう。

インタラクティブ・コンコーダンスにおいては、対応する OS とプログラムによっていくつかの種類がある。現在 PC のインターフェースには GUI（Graphical User Interface）が採用されているので、検索語や検索条件が設定しやすく、また、出力画面のスクロールが容易なことから、インタラクティブ・コンコーダンスプログラムに、すぐれたものが開発されてきている。代表的なプログラムである WinGREP は、Windows の GUI 環境に対応している⁵。同じ Windows OS で動作する WordSmith Tools もすぐれたプログラムである⁶。Mac OS で動作する Conc も WinGREP と同様の機能を備えたプログラムである⁷。DOS 上で利用するプログラム、MicroConcord は、与えられたキーワードを用いて KWIC 形式のコンコーダンスを表示する。近接語の検索も可能である⁸。

上記のようなインタラクティブ・コンコーダンスプログラムでは、コン

ピユータ上で KWIC 形式の出力を得るのは比較的たやすい。しかしながら、問題となるのは、これらのプログラムはそのインタラクティブ性のゆえに、特定の検索語とその至近のコンテキストのみを画面に表示するのが特徴であり、検索語が多い場合にはその都度次々に入力と出力を繰り返さねばならない。このような個別検索出力は、インタラクティブ・コンコーダンスプログラムの特徴であり、長所であるが、一方で限界でもある。すなわち、このようなプログラムは、語を網羅して、辞書のように使える、書籍形態のコンコーダンスを作成することが難しいからである。この目的のためには、ノンインタラクティブ・コンコーダンスプログラムを利用して、出力ファイルの内容を詳細に制御できる文字処理が必要となる。

ノンインタラクティブ・コンコーダンスプログラムとしては、UNIX OS から誕生した `grep` プログラム群がある。これは、文字列中の特定の単語、語句を検索して、それが含まれるファイル名、行数、当該行などを表示するのが基本的な機能である⁹。本来プログラミングの援助プログラムとして開発された `grep` は、DOS に移植された後に様々に改良され、文章中の文字列検索の道具として、もっとも汎用的なプログラムの一つとなった。基本的には、OS のシステムプロンプトに対してコマンド、検索語、検索ファイル名、それにオプションスイッチや出力先などを記述する。現在ではその発展形態として、上述した WinGREP のように、GUI 環境で利用できるプログラムも開発されている。

`grep` と同じく UNIX から誕生した `awk`, `sed`, `perl` も文字列処理を得意とするプログラムである¹⁰。これらはコンコーダンスプログラムというよりはむしろ、汎用的な文字列処理プログラムと位置づけた方がよいであろう。いずれも、スクリプトファイルと呼ばれる、文字列処理方法の詳細命令を書いたファイルを作成し、これにもとづいて文字ファイルを処理して、結果を画面ないしは、ファイルに出力する。この中でも `perl` は、複雑な処理ができるばかりでなく、WWW (World Wide Web) と CGI (Common Gateway Interface)

システムを経由して、インタラクティブな検索を可能にするという特徴を持っている。これを効果的に利用すれば、WWW システムを介して、オンラインで、コンコーダンスを提供することができる¹¹⁾。WWW と CGI を利用したシステムは、応用範囲が広く、今後の発展が期待できる¹²⁾。

研究利用に耐える、本格的なコンコーダンスを作成するためには、作品名、行数、見出し語、検索語、などを過不足なく表示して、参照しやすい丁寧な出力を得る必要がある。コンコーダンスプログラムの基本的なものは、上述したように、1960年代から80年代に、大型汎用機やミニコンで利用されるために開発された。その後1980年代後半から90年代にかけて、これらのプログラムの一部は、PCで利用されるべく、DOSに移植された。代表的なプログラムに、Oxford University Computing Serviceが開発して、OUPより発売されている、Micro OCPがある(Hockey, 1988)。これは、1980年に大型汎用機用に開発された、OCP (Oxford Concordance Program) を1988年にPC用OSに移植したものである¹³⁾。これとは別に1981年に、UNIX OSのために、文字列処理の小さなプログラム群が、University of Californiaで開発された。Hum パッケージと呼ばれるプログラム群である。

Hum パッケージプログラム群は、人文系の文字列処理を助けるため、比較的小型のコンピュータで利用できるように開発され、フリーウェアとして無料で公開された。この点、当初から大型機を前提に開発され、商業ソフトウェアとして販売されたOCPとは対照的である。また、後述するように、プログラム作成のコンセプトもHumとOCPでは対照的で、興味深い。Humは、1987年にDOSに移植された。移植された当時は、PCの演算能力も低く、利用できるメモリにも制限があったため、扱えるデータの大きさや処理速度も満足できるものではなく、実用的ではなかった。しかしながら、移植から10年を経た現在では、PCの処理能力は、当時のUNIX機をしのぎ、同じDOS用のプログラムでも、大量のデータを短時間に処理でき、コンコーダンス作成に十分に耐えるものとなっている。

5 コンコーダンスのあり方

インターネットのさらなる普及と、出版の前提としての電子編集組み版作業の浸透によって、今後は、広範なジャンルにおいて、信頼できる電子テキストが、ますます簡便に入手できるようになるであろう。テキスト分析のためのプログラムも、より洗練されたものが開発されてゆくことは間違いない。文学研究においては、このことは、作品の解釈にかかわるとともに、その補助としてのコンコーダンスのあり方にかかわることでもある。ここで、近年出版されたいくつかのコンコーダンスを例にとって、今後のコンコーダンスのあり方を展望したい。

コンピュータ処理はコンコーダンスの分量を増大させる。コンピュータ処理によって、コンコーダンスの作成は容易になったが、それはしばしば、文字列を、恣意を排して「機械的に」処理し、並べることを意味する。その結果として生成されるコンコーダンスは、すべての語彙をアルファベット順に網羅的に並べたものとなる。近年出版された中世英文学関係のコンコーダンスのうち、Oizumi (1991) (以下、O) と Benson (1993) (以下、BN) は、この意味で対照的である。双方ともに、Benson (1987) に基づいた Geoffrey Chaucer の作品集のコンコーダンスである。O においては、完璧に網羅的なコンコーダンスを作成した結果、“A”, “a”, “a” の3項目に総計2937行を費やしている。語を余すところなく網羅した結果、コンコーダンスの分量が増大し、Oは結局、Chaucer の作品の KWIC コンコーダンスに全10巻を要するものとなった。一方でBNでは、語彙のうち、重要性和頻度の高いものに注目し、冠詞や前置詞など比較的重要性の低い語に関しては、ごく少数の例を挙げるにとどまり、コンコーダンスを全1巻に凝縮することに成功している。この編集方針は、Chaucer の最初のコンコーダンス Tatlock & Kennedy (1927) の方法を踏襲している。

Benson は、そのコンコーダンスを Glossarial Concordance と呼ぶ。その序

文において、書籍形態で出版するコンコーダンスを、重要な high frequency words に限り、その後別途出版予定の電子コンコーダンスにおいて、完全な網羅的なコンコーダンスを出版すると約束している (Benson, 1993, Preface)。Benson は、さらにこのコンコーダンスを、語義辞書を兼ねたものにするために、見出し項目に、品詞と語義に加えて、*OED* と *MED* の当該箇所への言及を掲載している。品詞を特定したことにより、同形異品詞語は、品詞ごとに分類分けされる。すなわち、“a” の項目では、“indefinite article” の品詞のもとに例として 10 語、“interjection” として 41 語、独立文字 (letter A) として 22 語を分類している。この方法は、Matsushita (1998) においても踏襲される予定である。Benson が試みる、書籍形態と電子形態コンコーダンスの棲み分けは、今後のコンコーダンスのあり方の一つの方向を示している。ただし、残念ながら Benson の約束している電子コンコーダンスの出版は未だに実現していない。

Chaucer の *The Canterbury Tales* においては、写本の相違による内容の異同がしばしば問題になるが、Blake, Bumley, Matsuo & Nakao (1994) のコンコーダンス (以下、BL) は、あえて Hengwrt 写本に基づく刊本テキストをコンコーダンスとしたもので、注目される。BL と、O あるいは BN との比較検討によって、それぞれの写本に特有な語用法や語義などが明らかにされるであろう。また、Saito & Imai (1988a), Saito & Imai (1998b) においては、前者が中英語ロマンス群のうち Matter of England の、後者が Breton Lays のコンコーダンスとなっている。このように、ある特定ジャンルの作品をまとめたコンコーダンスは、それら一群の作品の言語的傾向を理解する上に重要である。これはまた、同一作者の作品群の相互比較にも応用可能な視点である。

以上、近年出版された中世英文学作品のコンコーダンスのうち、注目すべきものの特色を概観した。明らかになった点は、1) 網羅的なコンコーダンスの作成、2) 重要度の高い語のコンコーダンス作成、3) 異本のコンコー

ダンスの作成, 4) 特定の作品群のコンコーダンスの作成, が着実になされている点である。いずれもそれぞれに独自のテキストデータを作成することから始め, コンピュータ処理により生成された。

現在, テキストデータの作成と公開が進み, 上述のコンコーダンスで用いられたテキストデータは, すべてがインターネット上で入手可能である。University of Toronto では, *The Riverside Chaucer* と Hengwrt 写本のテキストデータを公開している¹⁴。また, University of Rochester では, 中英語ロマンスの主要作品電子テキストのほとんどが公開されている¹⁵。これらは, 純粋なテキストファイルであるために, BN で行われたような分析を試みようとする場合は, さらに個々の単語にその構文標識を付する必要があるが, これも徐々に自動化されてきている¹⁶。このような状況にあって, コンコーダンスについては, 個々の研究者がそれぞれの研究目的と研究の進捗状況に応じて独自に編集すべき時代が到来していると言えよう。そうすることで, 特定語彙のみのコンコーダンスの作成, 異本の比較, 類似した作品群の特性分析, さらに広範な作品群との比較も可能になる。近接語の検索など, 電子テキスト独自の検索を利用することも, 個々の研究者の必要においておこなうことで, はじめて意味を持つ。

しかしながら一方で, このような時代にあっても, 良質な書籍形態のコンコーダンスは, 基本的な作品について, テキストを厳選し作成し続けなければならない。また今後は, BN において見たように, 単に機械的に生成したコンコーダンスではなく, 編集過程において, 有益な付加情報を付与したコンコーダンスの創作が求められる。

6 Hum パッケージでのコンコーダンス作成

以下においては, Hum パッケージを用いて, Robert Henryson の全作品の基本的な KWIC コンコーダンスを生成する方法を紹介する。処理手続きを正確に守れば, アスキー文字 (ASCII [American Standard Code for Information

Interchange], アルファベット 26 文字の大文字小文字と, 数字および記号から構成される文字セット) で記述された作品ならば, 広く応用できる。

6.1 必要機材

コンコーダンス作成の手順は, まず電子テキストの作成から始まり, テキスト編集, コンコーダンス作成, コンコーダンス編集, コンコーダンス印刷の順に進む。Henryson のコンコーダンス編集に使用したシステムは以下のとおりである。なお, これらは, 必要な機能を満たす限りにおいて, 他の機器やプログラムで代替できる。なお, KWIC コンコーダンスは, 一つの単語につき一行の出力を行うために, 出力ファイルが非常に大きくなる。このため, ファイルを保存するハードディスクには, 出力ファイルの容量の大きさを計算して, 十分な空き容量が必要である。

コンピュータ: IBM 互換コンピュータ (Gateway2000, Pentium 133MHz, 40MB RAM, 1.2GB HDD, Windows 95 OS)

電子テキスト作成: スキャナ (Hewlett Packard ScanJet IV, Auto Document Feeder), OCR プログラム (OmniPage Professional Ver 7.0¹ [以下, OmniPage])

電子テキスト編集: フルスクリーンテキストエディタ (Vz Editor Ver. 1.60 [DOS 用], 秀丸 Editor Ver. 2.15 [Windows 95 用])

コンコーダンス作成: Hum パッケージプログラム群, ソートプログラム (ssort.exe 使用)

コンコーダンス編集: フルスクリーンテキストエディタ (上記参照)

コンコーダンス印刷: ワードプロセッサプログラム (Microsoft Word 7.0 使用)

6.2 テキストファイルの準備

コンコーダンスづくりの第一歩は, 作品の電子化である。信頼できる刊本テキストを採用し, 誤りのないテキストファイルを作成する必要がある。現

在英米文学作品は多くのテキストが電子化されていて、ネットワークを通じて無料で、あるいは市販の CD-ROM から入手できる。利用に際しては、電子テキストの正確さを十分に確認しなければならない¹⁷。研究の用に立てるには、原本テキストとの比較校訂が必要であるし、処理プログラムの特性に合わせて必要な編集加工を行う必要がある。自由に利用できるテキストが入手できない場合には、電子テキストを自前で作成しなければならないが、コンコーダンスの作成においては、これに最も多くの時間を割くことになる。この作業を手助けするプログラムに、OCR (Optical Character Recognition program, 光学的文字認識プログラム) がある。これを用いることで、入力には大幅に簡略化できる。ただしOCRを利用する場合でも、テキストの完成に至る最終的な調整は、入念な校正と手作業による修正に頼らねばならない。

Henryson の作品分析に用意したテキストファイルの、記号、略号、ファイル名、作品名、行数一覧は以下のとおり。

| 記号 | 略号 | ファイル名 | 作品名 | 行数 |
|----|----|--------|-------------------------------------|------|
| 01 | FB | fb.txt | The Fables | 2975 |
| 02 | TC | tc.txt | The Testament of Cresseid | 616 |
| 03 | OE | oe.txt | Orpheus and Eurydice | 633 |
| 04 | AN | an.txt | The Annunciation | 72 |
| 05 | AW | aw.txt | The Abbey Walk | 56 |
| 06 | BS | bs.txt | The Bludy Serk | 120 |
| 07 | GG | gg.txt | The Garmont of Gud Ladeis | 40 |
| 08 | HC | hc.txt | Against Hasty Credence | 56 |
| 09 | PA | pa.txt | The Praise of Age | 32 |
| 10 | PP | pp.txt | Ane Prayer for the Past | 88 |
| 11 | RA | ra.txt | The Ressoning betuix Aige and Yowth | 72 |
| 12 | RD | rd.txt | The Ressoning betuix Deth and Man | 48 |
| 13 | RM | rm.txt | Robene and Makyne | 128 |

| | | | | |
|----|----|--------|----------------------------|----|
| 14 | SP | sp.txt | Sum Practiysis of Medecyne | 91 |
| 15 | TD | td.txt | The Thre Deis Pollis | 64 |

6.3 OCR の利用

OCR を利用する時には、¹⁷できるだけ読み込みやすい原稿を準備する。現在の OCR プログラムは、原稿に汚れがなく、コントラストがはっきりしていれば、99%以上の認識が可能である。しかし実際には、認識率は、テキストが印刷されている用紙の色、印字の質に大きく左右される。用紙が黄色みがかったいたり、印字されている文字が不鮮明であると、認識率が著しく低下する。まず、数ページの認識を試みて、認識率が低い場合には、手入力に切り替えた方が効率的であろう¹⁸。読み込み原稿を、少し倍率を上げて、純白用紙にコピーした原稿の方が、もとの原稿よりも認識率が高いことがしばしばある。コピー原稿は、また、スキャナのオートドキュメントフィーダーを用いて、人手を介さずに、多くのページのデータ入力ができる利点がある。

いったん読み込んだ後は、テキストの修正が必要になる。OmniPage では、読み込んだ画面のイメージと、それを文字に解読したテキストの両方が表示できるので、2つの画面を比較しながら修正を進めることができる。また、OmniPage に付属しているスペルチェッカーも修正に利用できる。古英語や中英語のテキストでは、ほとんどすべての単語がミスペリングであると認識され、通常スペルチェッカーは利用されない。しかしながら、新規単語の登録機能を利用して、ユーザー辞書に古英語中英語の単語を登録しながら修正すると、しだいに、修正効率を上げることができる。また、OCR プログラムの「読み癖」を知ることも大切である。読み込んだばかりのファイルを閲覧すると、読み込み原稿の中の特定の文字が誤認識される「読み癖」のパターンが見いだせる。Henryson のテキスト（日本語モードで表示）では、-（エムダッシュ）や；（セミコロン）が半角カタカナのムにしばしば誤認識

された。また、OCR プログラムは大部分のヨーロッパの諸言語に対応しているが、英語設定のまま利用するので、非英語文字の、特殊なアルファベット (非アスキー文字とも言う、フランス語ドイツ語の特殊文字、古英語中英語特有の文字など) が正しく認識されないことにも注意しなければならない。また、小文字の l (エル) を大文字の I (アイ) に、小文字の i (アイ) を小文字の l (エル) に、小文字の b を小文字の h に、誤認識することが多いことは、OmniPage を Henryson のテキスト読み込みに用いた場合に特徴的であった。

完成した *The Fables* のテキストの冒頭 (ll. 1 -14) を示す。Hum パッケージで処理するには、テキストは空行なしに記述し、ファイル名、行数などの記述は必要ない。

Thocht fein@eyit fabils of ald poetre
 Be not al grunded vpon truth, @yit than,
 Thair polite termes of sweit rhetore
 Richt plesand ar vnto the eir of man;
 And als the caus quhy thay first began
 Wes to repreif the of thi misleuing,
 O man, be figure of ane vther thing.
 In lyke maner as throw a bustious eird,
 Swa it be laubourit with grit diligence,
 Springis the flouris and the corne abreird,
 Hailsum and gude to mannis sustenance;
 Sa springis thair ane morall sweit sentence
 Oute of the subtell dyte of poetry,
 To gude purpois, quha culd it weill apply.

6.4 電子テキスト編集とファイル印刷

認識したテキストを、スクリーンエディタ(あるいはワードプロセッサプログラム)に読み込んで、印刷テキストと比較し、誤りを訂正してゆく。後に用いるコンコーダンスプログラムは非アスキー文字の処理ができないので、中英語特有の文字、**3**を@Yに、**3**を @yに、**p**を`Dに、**p**を`dに、などアスキー文字の組み合わせに置き換える。これらの代替文字は、最終出力の時に、必要ならば本来の特殊文字に戻す。

Henryson の *The Fables* のように、一つの作品が複数の詩から構成され、かつ行番号が通し番号の場合には、できあがったコンコーダンスにおいて、詩の冒頭行を表示する場合に、その直前の詩の末尾部分が先頭に来ることを抑制しなければならない。この処理は、新たにプログラムを組むよりも、手作業で行うほうが簡単である。この作業を補助するために、各詩の最後にスペースを空け、便宜的に\$記号をつけ、後の編集作業の際に詩末行認識の助けにする¹⁹。

上記の作業中に、幾度かテキスト確認のためにファイルを印刷する必要がある。この確認のための印刷は、PCでも可能であるが、必要に応じて大型汎用機にファイルを転送して印刷したり、出力センターで行うと、大量の印刷が高速に行える。

6.5 コンコーダンスファイルの生成

Hum パッケージが開発された UNIX OS の基本プログラムのコンセプトは、あらゆる機能を盛り込んだ一つの大きなプログラムを開発して様々な処理をまかせるのではなく、一つ一つが独自の機能を果たす複数の小さなプログラムをモジュールとして作成し、それらのプログラムによる処理を組み合わせ、求める結果を得ることに特徴がある。個々のプログラムで処理を行う際に、それぞれに結果ファイルが作成されるので、これを点検することで、正

しい処理がなされているかどうかを、その段階ごとに知ることができる。処理結果が正しいことを確認した後は、個々の処理過程を次々と連結させることによって、一気に最終結果を得ることもできる²⁰。Hum パッケージはこのようなコンセプトに基づいて作成されている。

6.6 kwic2.exe プログラム

コンコーダンスの作成は kwic2.exe プログラムを中心に行う。まず、テキストを正確に処理するために、プログラムにアルファベット文字のみを認識させるようにし、不要な記号のみを排除するよう指令しなければならない。kwic2.exe はこの作業を、排除したい記号を記述したテキストファイルを作成することで行う。排除文字ファイルを mark.doc とした。その内容は以下のとおりである。詩の冒頭行を検索するための \$ マークも、排除ファイルに記述した。

```
'",.,;:--$?!()
```

このファイルを指定して KWIC コンコーダンスファイルを出力する。カレントディレクトリ（この場合は、D:\>）に kwic2.exe プログラム、mark.doc、処理すべきテキストファイルを置く。作品は *The Fables*, 2975 行の詩行を持つ。これを入力ファイルとした。ファイル名は fb.txt。出力ファイル名は fb.kwc。KWIC 部分45桁の出力。KWIC コンコーダンス中に表示される作品名は、最終的に作品集の出現順に並べ替える必要があるので、事後のソート処理が容易になるように、便宜的に作品名を 01 とする。kwic2.exe プログラムでの処理。処理コマンドの記述は以下のとおり²¹。

```
D:\>kwic2 -dmark.doc -w "01" -c45 fb.txt > fb.kwc
```

KWIC コンコーダンス fb.kwc が完成して、カレントディレクトリにファイルが作成される。fb.txt のサイズは128キロバイト、生成される KWIC ファイル fb.kwc のサイズは、1751キロバイトと、約14倍になる。出力する行数(-c オプション)を多くすれば、ファイルサイズはさらに増加する。*The Fables* 冒頭部分 3 行の KWIC コンコーダンスは以下のとおり。

| | | | | |
|-----------|----|---|-----------------------|-----------------------------|
| thocht | 01 | 1 | | Thocht fein@yeit fabils |
| fein@yeit | 01 | 1 | Thocht | fein@yeit fabils of ald |
| fablis | 01 | 1 | Thocht fein@yeit | fabils of ald poetre. |
| of | 01 | 1 | fein@yeit fabils | of ald poetre/Be not |
| ald | 01 | 1 | fein@yeit fabils of | ald poetre/Be not al |
| poetre | 01 | 1 | fabils of ald | poetre/Be not al grunded |
| be | 01 | 2 | fabils of ald poetre/ | Be not al grunded vpon |
| not | 01 | 2 | of ald poetre/Be | not al grunded vpon truth, |
| al | 01 | 2 | of ald poetre/Be not | al grunded vpon truth, |
| grunded | 01 | 2 | ald poetre/Be not al | grunded vpon truth, @yit |
| vpon | 01 | 2 | Be not al grunded | vpon truth, @yit than, |
| truth | 01 | 2 | not al grunded vpon | truth, @yit than,/Thair |
| @yit | 01 | 2 | grunded vpon truth, | @yit than,/Thair polite |
| than | 01 | 2 | vpon truth, @yit | than,/Thair polite termes |
| thair | 01 | 3 | truth, @yit than,/ | Thair polite termes of |
| polite | 01 | 3 | @yit than,/Thair | polite termes of sweit |
| termese | 01 | 3 | than,/Thair polite | termes of sweit rhetore |
| of | 01 | 3 | Thair polite termes | of sweit rhetore/Richt |
| sweit | 01 | 3 | polite termes of | sweit rhetore/Richt plesand |
| rhetore | 01 | 3 | termes of sweit | rhetore/Richt plesand |

以下、同様にして他の作品の処理も進める。-w オプションの部分は、02, 03 等に順に番号を振る²²。tc.kwc, oe.kwc 等のファイルが作成される。

6.7 生成ファイルの修正加工

ここで、出力結果に若干の手直しが必要となる。詩の冒頭に空白行を入れるために、\$を検索。\$がキーワード文字列の右側にあるときには、\$の直前のスペースを含めて、行末までを削除。\$がKWIC文字列の左側にあるときには、行頭から\$までを空白に置き換える。エディタの検索機能とキーボードマクロ機能を利用。

以下の例は、*The Fables* の、“The Prologue”と“The Cock and the Jasp”連結部の、修正前と修正後を示す。

修正前

```

of      | 01 63 | fand ane iolie stone,/ | Of quhome the fabill
quhome  | 01 63 |   ane iolie stone,/Of | quhome the fabill @ye
the     | 01 63 |       stone,/Of quhome | the fabill @ye sall heir
fabill  | 01 63 | stone,/Of quhome the | fabill @ye sall heir
@ye     | 01 63 | Of quhome the fabill | @ye sall heir anone.
sall    | 01 63 | quhome the fabill @ye | sall heir anone. $/Ane
heir    | 01 63 | the fabill @ye sall | heir anone. $/Ane cok
anone   | 01 63 | fabill @ye sall heir | anone. $/Ane cok sum
ane     | 01 64 |   sall heir anone. $/ | Ane cok sum tyme with
cok     | 01 64 |   heir anone. $/Ane | cok sum tyme with feddrum
sum     | 01 64 | heir anone. $/Ane cok | sum tyme with feddrum
tyme    | 01 64 | anone. $/Ane cok sum | tyme with feddrum fresch
with    | 01 64 | $/Ane cok sum tyme | with feddrum fresch and
feddrum | 01 64 | Ane cok sum tyme with | feddrum fresch and gay,
fresch  | 01 64 | sum tyme with feddrum | fresch and gay,/Richt
and     | 01 64 | with feddrum fresch | and gay,/Richt cant and
gay,    | 01 64 | feddrum fresch and | gay,/Richt cant and crous,
```

修正後

```

of      | 01 63 | fand ane iolie stone,/ | Of quhome the fabill
quhome  | 01 63 |   ane iolie stone,/Of | quhome the fabill @ye
the      | 01 63 |       stone,/Of quhome | the fabill @ye sall heir
fabill   | 01 63 |   stone,/Of quhome the | fabill @ye sall heir
@ye      | 01 63 |   Of quhome the fabill | @ye sall heir anone.
sall     | 01 63 | quhome the fabill @ye | sall heir anone.
heir     | 01 63 |   the fabill @ye sall | heir anone.
anone    | 01 63 |   fabill @ye sall heir | anone.
ane      | 01 64 |                       | Ane cok sum tyme with
cok      | 01 64 |                       | Ane | cok sum tyme with feddram
sum      | 01 64 |                       | Ane cok | sum tyme with feddram
tyme     | 01 64 |                       | Ane cok sum | tyme with feddram fresch
with     | 01 64 |       Ane cok sum tyme | with feddram fresch and
feddram  | 01 64 |   Ane cok sum tyme with | feddram fresch and gay,
fresch   | 01 64 |   sum tyme with feddram | fresch and gay,/Richt
and      | 01 64 |   with feddram fresch | and gay,/Richt cant and
gay,     | 01 64 |   feddram fresch and | gay,/Richt cant and crous,

```

最後にすべての KWIC のファイルを結合して一つの大きな KWIC のコン
 コーダンスファイル all.kwc を作成する。all.kwc のサイズは、2871 キロバ
 イトになった。

6.8 KWIC ファイルのソート

完成した KWIC ファイルは、キーワードの出現順に行が並んでいるので、
 これをキーワードのアルファベット順にソート（並べ替え）する必要がある。
 Hum パッケージには sort.exe というソートプログラムが付属しているが、
 このプログラムは、DOS 環境では、管理されるメモリの制限から、64 キロ

バイトを超えるファイルを処理できない。また、DOS システムに標準添付されているソートプログラム `sort.exe` も同様である。そこで、DOS 環境でも、メモリの制限を受けないソートプログラムを用いる必要がある。ここでは `ssort.exe` というフリーウェアを用いた²³。ソートすべきファイルがさらに大きくなる場合には、さらに処理速度の速い UNIX システムを利用して、そこで処理するのがよいであろう。現在多くの大学研究所では UNIX システムを導入しており、利用はさほど難しいことではない²⁴。`ssort.exe` を利用したソートのコマンドは以下のとおりである。入力ファイルをプログラムに渡し、出力ファイル `allsrt.kwc` を得る。`all.kwc` と `allsrt.kwc` のファイルサイズは同一。

```
D:¥>ssort< all.kwc >allsrt.kwc
```

作成された `allsort.kwc` のキーワード “gentil” から “gentilnes” までの部分を示す。

```
gentil | 03 32 | excellent of beautee,/ | Gentil of blude, callit
gentill | 01 79 | stane, quod he,/°0 | gentill iasp, 0 riche
gentill | 01 110 | in grit honour?/Rise, | gentill iasp, of all
gentill | 01 127 | not to dreid./This | gentill iasp, richt different
gentill | 01 234 | glowmand brow?/Ane | gentill hart is better
gentill | 01 410 | Betwix ane foxe and | gentill Chantecleir.
gentill | 01 434 | morne, my maister, | gentill Chantecleir!'
gentill | 01 487 | 'Allace, now lost is | gentill Chantecleir!'
gentill | 01 711 | I fane pretend to | gentill stait.'/Weill,'
gentill | 01 898 | iolie ionet, and the | gentill steid,/The asse,
gentill | 01 1370 | said he 'I am off | gentill blude;/My natal
gentill | 01 1398 | succour mak. '/'@Yit, | gentill schir,' said
gentill | 01 1495 | Vont till be fed with | gentill vennesoun.'/My
```

```

gentill | 02 326 | Quhilk was sa sweet, | gentill and amorous?
gentill | 02 536 | Schir Troylus it is, | gentill and fre.I/Quhen
gentill | 03 9 | the lawis of nature/A | gentill man tobe degenerate,
gentill | 03 27 | tell,/Bot first his | gentill generation/I
gentill | 03 65 | he was fair and wyse,/ | Gentill and full of liberalite,
gentill | 03 112 | till hir court this | gentill quene couth call.
gentillar | 15 44 | of ws of kin was | gentillar,/Or maist expert
gentilnes | 011548 | I quit sumpart thy | gentilnes/Thow did to
gentilnes | 02 547 | thy lawtie, and thy | gentilnes./I countit small
gentilnes | 03 7 | reherse his eldirs | gentilnes./It is contrair

```

6.9 項目付けと整形

Hum パッケージには、ソートされた KWIC ファイルに、出現する単語ごとに行を区切り、項目を立て、整形するプログラム、format.exe が付随する。このプログラムを用いて、より閲覧しやすいコンコードランスを作成する。format.exe を利用するには、処理するドライブのルートディレクトリに、tmp という名前で、プログラムが処理中に作業ファイルを一次保存する作業ディレクトリが存在しなければならない。以下の DOS コマンドで作業ディレクトリを作成する。いったん作成した後は、このディレクトリの存在は忘れてよい。

```
D:¥>md tmp
```

format.exe を実行するコマンドは以下のとおり。出力ファイル名は、allfmt.kwc とした。

```
D:¥>format allsrt.kwc > allfmt.kwc
```

6.10 最終調整と印刷

最終調整として、便宜的に数字で置換しておいた、ファイル内の01から始まる作品名を略号 (FB, TC 等) に戻す。次にファイルをワードプロセッサプログラムに読み込む。必要であれば、置換機能を用いて、中英語文字などの特殊文字を復元する。閲覧しやすいようにページのレイアウトを決める。

コンコーダンスファイルを印刷する場合には、KWIC 形式を乱さないように、文字が一定の幅で表示印刷される、fixed font (COURIER など) で行う必要がある。精読に耐える印字の大きさに用紙に印刷する。

以下に、allfmt.kwc のキーワード “gentil” から “gentilnes” までを示す。

GENTIL (1)

| | | | |
|----|----|------------------------|-------------------------|
| OE | 32 | excellent of beautee,/ | Gentil of blude, callit |
|----|----|------------------------|-------------------------|

GENTIL (18)

| | | | |
|----|------|-----------------------|-------------------------------|
| FB | 79 | stane, quod he,/’ 0 | gentill lasp, 0 riche |
| FB | 110 | in grit honour?/Rise, | gentill lasp, Of all |
| FB | 127 | not to dreid./This | gentill iasp, richt different |
| FB | 234 | glowmand brow?/Ane | gentill hart is better |
| FB | 410 | Betwix ane foxe and | gentill Chantecleir. |
| FB | 434 | more, my maister, | gentill Chantecleir!’ |
| FB | 487 | ’Allace, now lost is | gentill Chantecleir!’ |
| FB | 711 | I fane pretend to | gentill stait.’/’Weill,’ |
| FB | 898 | iolie ionet, and the | gentill steid,/The asse, |
| FB | 1370 | said he, ’I am off | gentill blude;/My natall |
| FB | 1398 | succour mak. 3 it, | gentill schir,’ said |
| FB | 1495 | Vont till be fed with | gentill vennesoun./’ My |
| TC | 326 | Quhilk was sa sweit | gentill and amorous? |
| TC | 536 | Schir Troylus it is, | gentill and fre.’/Quhen |

| | | | |
|----|-----|------------------------|---------------------------------|
| OE | 9 | the lawis of nature/A | gentill man tobe degenerate, |
| OE | 27 | tell,/Bot first his | gentill generataion/I |
| OE | 65 | he was fair and wyse,/ | Gentill and full of liberalite, |
| OE | 112 | till hir court this | gentill quene couth call. |

GENTILLAR (1)

| | | | |
|----|----|------------------|----------------------------|
| TD | 44 | of ws of kin was | gentillar,/Or maist expert |
|----|----|------------------|----------------------------|

GENTILNES (3)

| | | | |
|----|------|----------------------|---------------------------|
| FB | 1548 | I quit sumpart thy , | gentilnes/Thow did to |
| TC | 547 | thy lawtie, and thy | gentilnes/I countit small |
| OE | 7 | reherse his eldirs | gentilnes./It is contrair |

7 まとめ

以上、一般に利用されているパーソナルコンピュータと、フリーウェアのプログラムを用いて、Henryson の全作品の KWIC のコンコーダンスを作成する過程を示した。現在市販されている PC を用いれば、十分に高度な処理が可能である。高額な報酬と時間を費やして、専用プログラムを開発するまでもなく、すでに開発されているプログラムに少々の工夫を加えるだけで、十分に実用的なコンコーダンスを作成することが可能である。今後より広範なジャンルの、より正確な電子テキストがより簡便に入手できるようになる。個々の研究者が、研究の目的に応じて、必要とする文字処理を行い、基礎資料を得ることが可能であり、そうすることが要求される時代が到来している。

* 小論は、1994-95年度の同志社大学学術奨励金を受けた共同研究『中世スコットランド詩人 Robert Henryson の全作品のコンコーダンスと翻訳作成』（安藤光史氏との共同研究）の研究成果の一部である。

注

- 1 現在入手できるプログラムは、市販されているもの、シェアウェアやフリーウェアなど約 20 種類ある。主要なものの一覧は、以下の URL を参照。<http://www.amherst.edu/~hnishino/corpus/texan.html>
- 2 Hum パッケージ (A Concordance and Text Analysis Package) の入手先 URL は、<https://iw.nim.niftyserve.or.jp/hns/nifty/flm/lib/5/2.html>。ドキュメントファイルは、<https://iw.nim.niftyserve.or.jp/hns/nifty/flm/lib/5/1.html> にある。プログラムそのものは無料で、使用上の制限はないが、上記サイトからダウンロードするには、NIFTY-Serve の会員であることが必要。なお、UNIX OS のために開発された Hum プログラムは、以下の URL で入手可能である <ftp://crl.nmsu.edu/CLR/tools/concordances/>。
- 3 *MLA Bibliography* on CD-ROM (1963.8-1997) によれば、コンコーダンスが博士論文として受理されたものは、1963-69 に 8 件、1970-79 年に 35 件、1980-1989 年に 11 件、1990-1993 年に 5 件あり、1994 年以降は例を見ない。なお、コンコーダンスの商業的な出版も、1990 年代に入って若干の減少傾向が見られる。
- 4 英米文学作品の電子テキストの所在に関しては、<http://www.amherst.edu/~hnishino/corpus/etext.html> を参照のこと。
- 5 WinGREP はシェアウェア。詳細は、<http://www.hurricanesoft.com/prod01.htm> 参照。
- 6 WordSmith Tools は Oxford University Press より発売されている。詳細は、<http://www.liv.ac.uk/~ms2928/homepage.html> を参照。
- 7 Conc はフリーウェア。詳細は、<http://www.sil.org/computing/conc/conc.html> 参照。
- 8 MicroConcord は OUP より発売されたが、現在はフリーウェア。詳細は、<http://www1.oup.co.uk/site.index.html> 参照。また、ソフトウェアについて詳細にコメントした Ball (1995) を参照。
- 9 grep: Globally search for the Regular Expression and Print the lines containing matches to it. 以下、プログラムの解説は、FOLDOC (The Free On-line Dictionary of Computing: <http://nightflight.com/foldoc/>) によった。
- 10 awk: An interpreted language included with many versions of Unix for massaging text data developed by Alfred Aho, Peter Weinberger, and Brian Kernighan 1978.
sed: The Unix stream editor. It has a powerful but cryptic command language and is based on regular expressions.
perl: Practical Extraction and Report Language (or Pathologically Eclectic Rubbish Lister). An interpreted language developed by Larry Wall and distributed over Usenet.
- 11 筆者個人の WWW サイトにおいて、Henryson のテキストの KWIC コンコーダンスを提供する試みを行っている。CGI と perl を用いた検索を、Henryson の *The Fables*

のテキストを用いて試行稼働中である。以下の URL を参照, <http://www.amherst.edu/~hnishino/search/search.html>。

- 12 まだ試行段階であるが、現在中英語だけでも約 5 つのサーバが様々な検索エンジンを利用して、語義や語用法の情報を提供している。詳細は <http://www.amherst.edu/~hnishino/search/search.html> の “Other Middle English Search Engines” を参照。
- 13 MicroOCP および OCP の利用に関しては, Hockey (1988), Jackson (1990), Romeln (1990) を参照。
- 14 これらのテキストは Ian Lacashire によって、1995 年に電子化された。University of Toronto の所蔵サーバの URL は、以下の通り, <http://utcat.library.utoronto.ca/utel/chaucer/chaucertitles.html>。
- 15 University of Rochester, TEAMS Middle English Texts。URL は, <http://rodent.lib.rochester.edu/camelot/teams/tmsmenu.htm>。
- 16 University of Birmingham は email による tagging サービスを行っている。ただし本稿執筆段階ではサイトの大幅な改良を行っているのでサービスは停止中。トップページ URL は <http://clg1.bham.ac.uk/>。
- 17 著名なサイトが紹介しているテキストに多数の誤入力を発見することも多い。Encyclopaedia Britannica Online がリンクを掲載している, Martin Luther King, Jr. の著名な “I have a dream” のスピーチは, Michigan State University の ftp アーカイブのもの (<ftp://ftp.msstate.edu/pub/docs/history/USA/Afro-Amer/dream.king>) は誤入力と欠落がきわめて多い, 他方, Stanford University の The Papers of Martin Luther King, Jr. プロジェクトのもの (<http://www-leland.stanford.edu/group/King/Docs/march.htm>) は, 実際のスピーチを忠実に転写している。
- 18 Hockey は, OmniPage Professional は, 利用者による高度なカスタマイズが可能なので, 活字の特徴が一定している作品 (群) については, 認識率が高いが, 認識率が低い場合には, 手入力 (複数のタイピストによる同一テキストの並行入力と, 比較による校訂) の方が優れていると指摘している (Hockey, 1996, p.4)。
- 19 *The Fables* の内容は以下のとおり。

| | | |
|------------------------|-------|------|
| The Prologue | 1- | 63 |
| The Cook and the Jasp | 64- | 161 |
| The Two Mice | 162- | 396 |
| The Cock and the Fox | 397- | 613 |
| The Fox and the Wolf | 614- | 795 |
| The Trial of the Fox | 796- | 1145 |
| The Sheep and the Dog | 1146- | 1320 |
| The Lion and the Mouse | 1321- | 1621 |

| | |
|---------------------------------------|------------|
| The Preaching of the Swallow | 1622- 1950 |
| The Fox, the Wolf, and the Cadger | 1951- 2230 |
| The Fox, the Wolf, and the Husbandman | 2231- 2454 |
| The Wolf and the Wether | 2455- 2615 |
| The Wolf and the Lamb | 2516- 2776 |
| The Paddock and the Mouse | 2777- 2975 |

20 これをパイプ機能という。UNIX および DOS のパイプ機能は、複数コマンドを | (バーチカルバー) で結合することで行われる。例えば、あるディレクトリで、.1997 というファイル名を含むファイルを検索し、その中から Internet という文字列を含む行を抽出して、ネットワーク上のレーザプリンタで印刷に付すコマンドは、以下のように記述する。なお、% 記号は UNIX のシステムプロンプトを示す。

```
% ls *.1997 | grep Internet | lp
```

21 処理は Windows95 の MS-DOS 窓で行う。パラメータを入力するバッチファイルを書くコマンドの記述が簡単に済む。例えば、

```
kwic2 -dmark.doc -w"%1" -c45 %2.txt > %2.kwc
```

というコマンドを kw.bat として D:\% におけば、数字とファイル名のパラメータを入力するだけで、処理ができる。すなわち、第 2 番目のファイル、tc.txt の処理は、

```
kw 02 tc
```

とコマンドを入力するだけでよい。

22 このプログラムは、行付けを、原則的にファイルの 1 行目から順にカウントするが、-p オプションで開始行を指定可能である。ページ付けと行付けの処理に関しては、kwic.doc と humtut.doc 参照。

23 ssort.exe の入手先は以下のとおり、<https://iw.nim.niftyserve.or.jp/hns/nifty/feng/liv/5/187.html>。利用制限や入手方法は、上記注 2 に準ずる。

24 UNIX での処理 (転送, 結合, ソート, 転送)

ファイルを適当な大きさに分割 (あるいは結合) して、必要であればプログラムによって圧縮して転送する。ファイル転送機能によりホストに送り込む。圧縮した場合はホスト上で展開する。複数ファイルがある場合には、UNIX の cat コマンドを用いて結合する。例として、3 つのファイルを 1 つに統合するには以下のコマンドを遣う。

```
cat file1 file2 file3 > file.all
```

ソートのコマンドは以下のとおり。

```
sort 入力ファイル名>出力ファイル名
```

参考文献 **

- ** 電子データの記述は, Crane (1997) と Li and Crane (1996) に拠った。
- Ball, C.N. & Taylor, K.B. (1995). "Micro Concord and Corpus Collections", hypertext reprint of an article in *Computers and the Humanities*, [Online]. Available: <http://www.georgetown.edu/cball/preprints/microconcord.html> [1997, November30].
- Benson, L.D. (1987). *The Riverside Chaucer*. Boston: Houghton Mifflin Company.
- _____. (1993). *A Glossarial Concordance to the Riverside Chaucer*. New York: Garland Publishing.
- Blake, N., Burnley, D., Matsuo, M. & Nakao, Y. (1994). *A New Concordance to the Canterbury Tales: Based on Blake's Text Edited from the Hengwrt Manuscript*. Okayama: University Education Press.
- Crane, N. (1997) "Electronic Sources: APA Style of Citation", [Online]. Available: <http://www.uvm.edu/~ncrane/estyles/apa.html> [1997, November 30].
- Hockey, S. (1980). *A Guide to Computer Applications in the Humanities*. Baltimore: The Johns Hopkins Press.
- _____. (1988). "Preface", *MicroOCP: User Manual*. Oxford: Oxford University Press.
- _____. (1996). "Creating and Using Electronic Editions", *The Literary Text in the Digital Age*. Ann Arbor: University of Michigan Press.
- Holmes, D. I. (1994). "Authorsip Attribution", *Computers and the Humanities*, 28, 87-106.
- Illich, I. (1993). *In the Vinyard of the Text*, Chicago: The University of Chicago Press.
- Jackson, H. (1990). "OCP and the Computer Analysis of Texts: the Birmingham Polytechnic Experience" *Literary and Linguistic Computing*, 5, 86-88.
- Janus, L. (1990). "HUM-A Concordance and Text Analysis Package for UNIX", *Computers and the Humanities*, 24, 510-512.
- Jones, A. & Churchhouse, R. F. (1976). *The Computer in Literary and Linguistic Studies: Proceedings of the Third International Symposium*. Cardiff: The University of Wales Press.
- Lancashire, I. & McCarty, W. (1988). *Humanities Computing Yearbook*. Oxford: Oxford University Press.
- Li, X. & Crane, N. B. (1996). *Electronic Style: A Guide to Citing Electronic Information*. Medford, NJ: Information Today.
- Matsushita, M. (1998). *A Glossarial Concordance to William Langland's The Vision of Piers Plowman: The B-Text*. Tokyo: Yushodo.
- Oizumi, A. (1991). *A Complete Concordance to the Works of Chaucer*. Hildesheim: Olms-Weidmann.

- Roman, G. S. (1990). "Using OCP: a Study of Characterization in the Two Don Quixotes", *Literary & Linguistic Computing*, 5, 314-318.
- Ross, C. L. (1996). "Text and the Dearth of the Critical Edition". *The Literary Text in the Digital Age*. Ann Arbor: University of Michigan Press, 225-231.
- Saito, T. & Imai, M. (1988a). *A Concordance to Middle English Metrical Romances: Volume One, The Matter of England*. Frankfurt am Main: Peter Lang.
- _____. (1988b). *A Concordance to Middle English Metrical Romances: Volume Two, The Breton Lays*. Frankfurt am Main: Peter Lang.
- Scott, M. (1993). "Acknowledgements", *MicroConcord: Manual*. Oxford University Press.
- Tabata, T. (1995). "Narrative Style and the Frequencies of Very Common Words: A Corpus-Based Approach to Dickens's First Person and Third Person Narratives", *English Corpus Studies*, 2, 91-109.
- Tatlock, D. & Kennedy, A. G. (1927). *A Concordance to the Complete Works of Geoffrey Chaucer and to the Romaunt of the Rose*. Washington: The Carnegie Foundation of Washington.
- Wisbey, R. A. (1971). *The Computer in Literary and Linguistic Research: Papers from a Cambridge Symposium*. Cambridge: Cambridge University Press.

Synopsis

Creating a Complete Concordance to Robert Henryson's Works

Haruo Nishinoh

As Personal computers come to have what used to be a mainframe computing power, they are now able to handle many complicated tasks to help humanities research. Creation of computer generated concordances is one of them. Although concordances are often indispensable for defining the meanings of the words in literary and linguistic studies, their publication has been relatively limited. This is partly due to the technical difficulty, and partly due to the marketability of the printed concordances. With today's high-speed computers and available concordancing software programs, however, scholars are given an opportunity to create computer concordances at their will. Creating a truly publishable and reference worthy concordance, however, needs fair amount of computer literacy and skill.

This paper attempts to review briefly the history of concordance creation, discusses the nature and the use of the electronic text, introduces current concordancing software programs, gives insight into the future use of computers in text analysis, and finally provides a practical guidance for creating complete concordance to Robert Henryson's works.

Tatlock and Kennedy (1927) was a monumental work of pre-computer age concordance. The Project was first started by F. J. Furnival in 1871 and took 57 years to complete. The first attempt of computer concordance was that of Father Roberto Busa's Thomas Aquinas concordance in 1949. After this, many

attempts were made using mainframe and minicomputers until late 1970s. In 1980s and 1990s, drastic downsizing of computers made it possible for scholars to create their own personalized version of concordances.

In creating a concordance, a printed text has to be first made into electronic text. Quite different from the printed text, which is firmly fixed on papers, the electronic text is characterized with its “liquidity”, or flexibility. That is, it is ready to be searched, processed, sorted, edited, and printed. In a way, bundled with concordancing program, the electronic text is a text and many forms of indexes and concordances bound together.

An electronic concordance needs little space for storage but requires special computing environment for reference. On the other hand, a printed concordance, despite its bulk, is easy to handle. Considering the advantages and disadvantages of both electronic concordances and printed concordances, it will be best to first create a concordance in an electronic form for viewing on computer display and then print and compile it in a book form as necessary.

There are approximately twenty text analysis programs now available. Among those, interactive concordance programs come handy with GUI computers. WinGREP, conc and WordSmith Tools are among these. These are best for interactive word search. Non-interactive programs, such as OCR, MicroOCR, are good at regulating the details of output file and therefore best for creating exhaustive concordances. Scholars should be able to use both types of software, as their research requires.

As the creation of computer concordance becomes easy, the content of publishable concordance has to be reexamined. To take some recently published concordances for instance, exhaustive concordance, such as Oizumi's (1991) will cease to be published in a book form, while glossarial concordance like Benson's (1993) and Matsushita's (1998) will be more in demand and will be

published in a book form. Concordances based on disputable manuscript, like Blake, Bumley, Matsuo & Nakano (1994), and concordances for related important works, such as Saito & Imai's (1988a, 1988b), will still be quite meaningful in a printed form.

Creation of Henryson's concordance is done as follows: setting up of the computer and peripherals; creating of the electronic text; editing the electronic text, creating the electronic concordance; editing it; and finally making the output in printed form.

Any of the current personal computer is fast enough to handle scholarly text analysis including concordance creation. "IBM compatible" computers will be the choice, as more and more software programs are developed on Windows OS with Pentium CPU based computers. Henryson's concordance is created on Gateway 2000 (Pentium 133MHz, 40MB RAM, 1.GB HDD, Windows 95 OS).

Creation of the error free electronic text is the most time consuming task in the whole process. In order to facilitate it, OCR (Optical Character Recognition) software is used with ADF (Automatic Document Feeder) equipped scanner. In this case Hewlett Packard ScanJet IV with ADF, and OmniPage Professional Version 7.1 were used as an OCR system. Since any kind of input method is not error free, most careful proofreading should be repeatedly done in order to completely eliminate errors. Correction of mistakes can be done with a help of spell checking function on word processors. Even a correction of Old and Middle English texts can be done with spell checkers by registering words and phrases into user dictionary. Good word processor and full screen text editors help this process a great deal. In Henryson's case, Vz Editor Version 1.6 and Hidemaru Editor Version 2.15 were used.

Creation of concordance itself is done by the "Hum" ("Hum" for humanities)

package. This is a collection of text analysis software programs, which was first developed for UNIX operating system in 1981 at University of California and was distributed free of charge. The package was later transplanted onto MS-DOS platform in 1987. The package satisfies basic text processing needs to create publishable concordances. A KWIC concordance creator is a program “kwic2.exe”. Before processing, the electronic text must be prepared in such a way that it meets the requirement of the program. Generated concordance file is further edited, sorted and processed under alphabetically arranged lemmas, or headwords. The program that creates this lemmatized concordance is the “format.exe” program in the package. With the help of sort program, editor, and word processor, the final publishable concordance is produced.

As have been examined above, in this age of high-speed personal computing, scholars should be able to and required to create concordances at their own will as necessary to their research.