Title	テキストデータからのコロケート抽出および共起度測定のためのGUIソフトウェアの開発
Author(s)	福田，薫
Citation	北海道教育大学情報処理センター紀要，11：47-53
Issue Date	2007-03
URL	http://s-ir.sap.hokkyodai.ac.jp/dspace/handle/123456789/8902
Rights	本文ファイルはNIIから提供されたものである。

テキストデータからのコロケート抽出および共起度測定のためのGUIソフトウェアの開発

Kwicker: A GUI Software for Automatic Extraction and Measurement of Significant Collocates from Corpora

福田　薫
Kaoru Fukuda
北海道教育大学函館校
Hokkaido University of Education, Hakodate

概　要

主に英語を対象とするコーパス探索とコロケーション分析および各種の頻度分析を行うGUIソフトウェアKwicker1.0を開発した。Java言語で開発されたこのソフトウェアは，柔軟で高速なテキスト探索と高性能なコロケーション分析機能を提供する。これに加えて，このソフトウェアは，語彙頻度リストの作成，ワード・スペクトルの作成，プラスワードの抽出，Nグラムの頻度調査，χ^2乗値や尤度比値，G^2得点の算出など，対象コーパスの特徴解明に役立つ多様な機能を提供する。これらのテキスト解析支援機能はユーザフレンドリーなGUI形式のインターフェイスにおいて実装されている。

1．はじめに

大規模なテキストデータの精密な探索とその結果の数値分析は，自然言語の使用に関する新たな知見の発見や蓄積を可能にするとともに，既存の理論や仮説の妥当性の検証を可能にする。とりわけ，特定の語と特徴的に共起するコロケート語（collocate）の抽出は，言語の使用実態や変異の動向の研究に有効な方法論として注目されてきている。

特定の語に対するコロケート語は，t得点，z得点，MI得点などのコロケーション統計量に基づいて抽出される。探索の対象となる言語テキストが大規模になるにつれ，これらの統計量の算出に必要な計算処理量は増大していく。そこで，高速なテキスト探索とコロケーション分析機能を兼ね備えたソフトウェアの開発が望まれている。

各種のコロケーション統計量をGUI形式で算出でき，高性能で使い易く，しかも無料で利用できる国内ソフトの価値は大きい。そこで本研究では，主に英語のコーパスを対象としてテキスト探索とコロケート語の自動抽出および多様な頻度分析機能を提供するKwicker1.0を開発した。この総合的なテキスト解析支援ツールはJava言語で開発された。開発環境は以下の通りである。

(1) a. OS：Windows XP Professional
　　b. CPU: AMD Athlon 3200+
　　c. JDK: Java 2 SDK5.0
　　d. IDE: Eclipse3.1.2

本稿では，Kwicker1.0が提供する主要な機能，その特徴と使用法および今後の課題を概述する。

2. Kwicker1.0の基本機能

Kwicker1.0は主に次に挙げる機能を提供する。

(2) a. テキスト探索とKWIC表示[1]
　　b. コロケート語の抽出
　　c. 各種頻度分析機能

Kwicker1.0はこれらの機能を実装する3つのタブ画面と，そこに表示されている情報を一時的に保存するタブ画面から構成される。この節では，これらの主要な機能を順に概説していく。

2.1 テキスト探索とKWIC表示

テキスト探索とKWIC形式のコンコーダンス作成は，後続する分析の出発点となる基本的な処理である。それゆえ，「Search & KWIC」タブ画面がKwicker1.0起動後の初期画面として表示される。この画面が提供する機能は，テキスト探索の実行，ヒットしたデ

ータの KWIC 表示，および KWIC データの編集の 3 つである。

画面上部にはこれらの作業をコントロールするボタン類がほぼ使用される順に配置されている。上部左端のファイル選択ボタンを押すとファイル選択ダイアログが表示される。[2] 特定のファイル群をコーパスとしてすでに登録してあると，ボタン表示された登録済みコーパスをクリックで選択できる。[3] 対象コーパスを複数選択することも可能で，選択状態になったコーパスのボタンは背景色が変化する。

探索対象のファイルまたはコーパスを指定した後，探索文字列を所定の欄に入力する。Kwicker1.0 は正規表現(regular expression)を用いた探索文字列の指定を許す。[4] これにより，柔軟で強力なキーワード指定が可能になった。対象ファイルのパス文字列や探索文字列を入力する欄は，「ヒストリー機能」を有する。以前の探索で使用された文字列をプルダウンメニュー内に保持し，次回以降の探索で利用できる。

「探索と KWIC」メニューから「KWIC 表示のオプション（K)」を選択すると，探索の様式や結果の表示様式をさらに細かくカスタマイズすることができる。そこでは，探索時における大文字・小文字の区別，KWIC データ表示の上限，KWIC 表示における文脈の表示幅，キーワードや前後文脈の文字色や背景色，表示フォントの種類やサイズをオプション設定できる。

テキスト探索ボタンをクリックすると，探索が開始される。[5] 探索実行中は，探索の進捗度合いを報告する画面を表示して，ユーザの不安を緩和する。その画面には探索中止ボタンがあるので，ユーザは実行中の探索を不要と判断したときはいつでもそれを中止することができる。

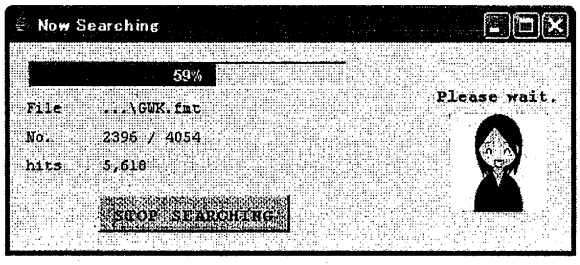


図 1　進捗表示画面

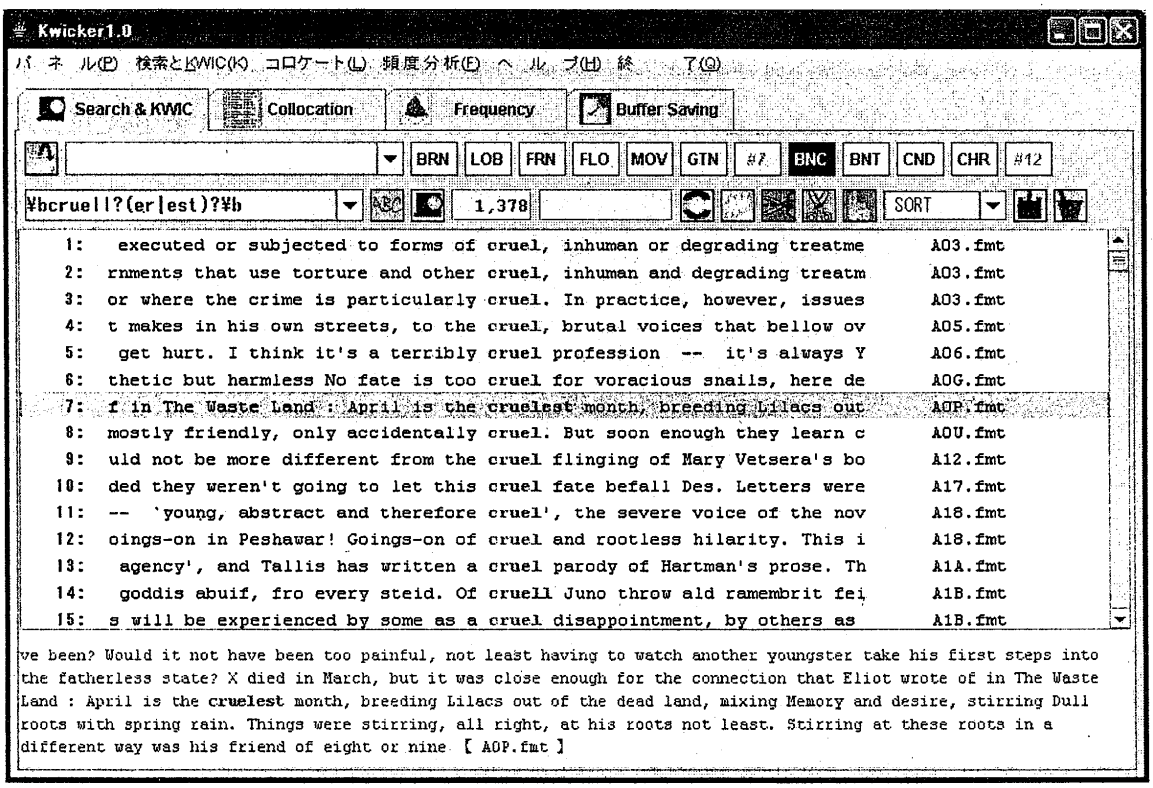


図 2　KWIC 表示の例（第 7 例が選択され，より広い文脈が画面下部に表示されている）

テキスト探索の結果はKWIC形式で画面に表示される。図2（前頁）は，BNCを対象とし，形容詞cruelの原級，比較級，最上級形をその変異形も含めて一括して探索したときのKWIC 表示画面である。

探索が終了するとヒットした件数，すなわち現在表示されているKWICデータの数が，実行ボタンの右欄に表示される。KWIC データが多量のときは画面右側にスクロールバーが自動的に現れる。

ユーザは画面上のKWICデータを見ながら，データの絞込み，削除，ソート，保存などの編集作業を行える。たとえば，任意のKWIC行を左クリック選択すると，より広い（計 1,000 文字）文脈データが画面下部に表示される。右クリックで選択すると，当該データの削除，KWIC データあるいは文脈データの一時保存などの処理を選択できる。もちろん，削除ボタンのクリックにより当該データを削除することもできる。

KWIC データが多量のときは，第2探索語を入力して，該当する部分を色づけ表示したり，それを含む例だけを絞込み表示することも可能である。また，キーワードの前後5語の位置をキーにしてKWICデータを並べ替えたり，キーワード前後の位置に出現する語を位置ごとに集計することもできる。編集されたKWICデータや集計結果のデータを選択してバッファ画面に一時保存したり，それらをファイルに保存したり，さらには，以前保存したKWICデータファイルを後で読み込んで編集することも可能である。

2.2 コロケート語の抽出

テキスト探索と KWIC 表示処理は，ある意味では，次のコロケーション分析へと移行するための準備ステップに過ぎない。特定のキーワードに対するコロケート語の自動抽出こそ，Kwicker1.0 が提供する中核的な機能であるとも言える。Kwicker1.0 の特長は，分析の準備，分析の実行，結果の編集という自然な流れに従って，t 得点(t-score)，z 得点(z-score)，MI 得点(mutual information score)という重要な統計量を見やすい表形式で提示する点にある。[6] これらの統計量の算出式は次の通りである(Barnbrook (1996, ch. 3))。

(3) $$z = \frac{O - E}{\sigma}$$

(4) $$t = \frac{O - E}{\sqrt{O}}$$

(5) $$I = \log_2 \frac{O}{E}$$

上の式において O はキーワード近隣におけるコロケート語の観測度数(observed frequency)，一方，Eは通常の環境下におけるその語の期待度数(expected frequency)を表わす。式(3)における σ はモデルコーパス全体におけるその語の出現の標準偏差の値を表し，具体的には式(6)により算出される。

(6) $$\sigma = \sqrt{N(p(1-p))}$$

ここで，p は全環境におけるコロケート語の出現率，N は集計範囲に生じている語の合計を表す。[7]

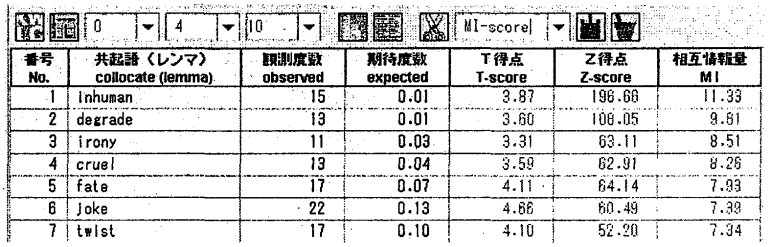
「コロケーション」タブ画面の上部には，ユーザの作業順にコントロール部品が配置されている。(次頁の図 3 参照) まず，左端のロードボタンを押して現在のKWICデータを表示する。次に，コロケートの集計モードを選択する。デフォルトではレンマ形(lemma form)で集計するが，トグルボタンを押すと表記語形(graphic form)での集計に切り替わる。その右に並ぶ 3 つのプルダウンメニューにおいて，コロケート語の集計範囲と最少頻度閾値を指定する。集計範囲はキーワード左右 5 語の範囲内で選択できる。頻度の下限を設定しておくのは，z 得点や MI 得点が低頻度のコロケート語にやや過度のウェイトを与える傾向があることが知られているからである。KWIC データ数や表示される分析結果に応じて，ユーザは下限値を指定できるので，異常値の混入を避けることができる。

分析実行ボタンを押すと，集計範囲内に生起する，指定頻度以上の全てのコロケート語について 3 種類のコロケーション統計量（t 得点， z 得点，MI 得点）が計算される。その結果は，コロケート語のアルファベット順に従って，一覧表の形式で表示される。[8]

図3（次頁）は，図2で示したKWICデータを対象と

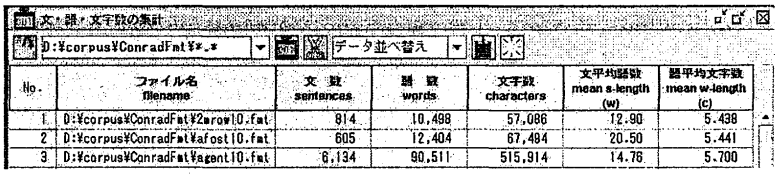
して，キーワードの右側４語を集計範囲，最小頻度を 10 に指定してコロケート語を抽出した後，MI 得点をキーにしてソートした結果の表示である。

3 種類の統計量はそれぞれ，統計的に有意と判定される境界値が定まっている。具体的には，t 得点は 2 以上，z 得点は約 3 以上，MI 得点は 3 以上が目安とされている（Barnbrook 1996）。Kwicker1.0 では，算出された統計量が境界値を超える時には，その値が有意であることを示すために赤い文字色で表示される。これにより，コロケート語抽出の結果が視覚的にも明瞭になっている。



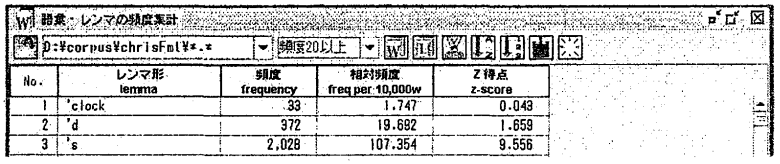
番号 No.	共起語（レンマ） collocate (lemma)	観測度数 observed	期待度数 expected	T得点 T-score	Z得点 Z-score	相互情報量 MI
1	inhuman	15	0.01	3.87	196.60	11.33
2	degrade	13	0.01	3.60	108.05	9.81
3	irony	11	0.03	3.31	63.11	8.51
4	cruel	13	0.04	3.59	62.91	8.26
5	fate	17	0.07	4.11	64.14	7.93
6	joke	22	0.13	4.66	60.49	7.38
7	twist	17	0.10	4.10	52.20	7.34

図 3　コロケーション分析結果の表示例



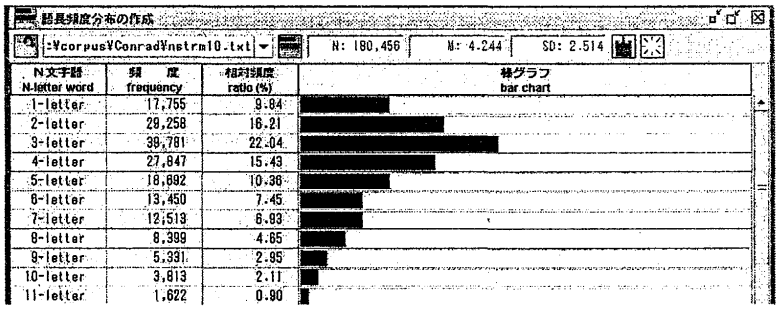
No.	ファイル名 filename	文数 sentences	語数 words	文字数 characters	文平均語数 mean s-length (w)	語平均文字数 mean w-length (c)
1	D:¥corpus¥ConradFmt¥2arow10.fmt	814	10,498	57,086	12.90	5.438
2	D:¥corpus¥ConradFmt¥afost10.fmt	605	12,404	67,484	20.50	5.441
3	D:¥corpus¥ConradFmt¥agent10.fmt	6,134	90,511	515,914	14.76	5.700

図 4　文数・語数・文字数の集計



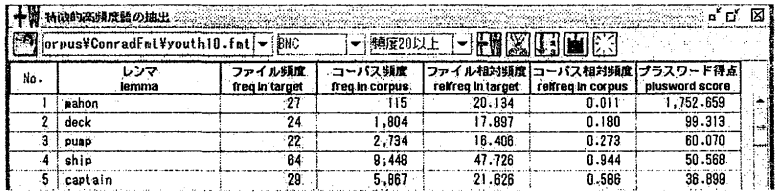
No.	レンマ形 lemma	頻度 frequency	相対頻度 freq per 10,000w	Z得点 z-score
1	'clock	33	1.747	0.043
2	'd	972	19.682	1.659
3	's	2,028	107.354	9.556

図 5　語彙頻度リストの作成



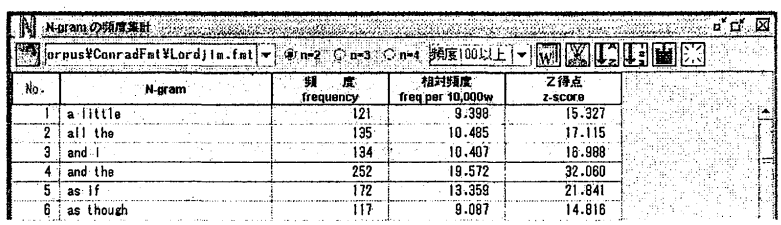
N文字語 N-letter word	頻度 frequency	相対頻度 ratio (%)	棒グラフ bar chart
1-letter	17,755	9.84	
2-letter	29,258	16.21	
3-letter	39,781	22.04	
4-letter	27,847	15.43	
5-letter	18,692	10.36	
6-letter	13,450	7.45	
7-letter	12,513	6.93	
8-letter	8,399	4.65	
9-letter	5,331	2.95	
10-letter	3,813	2.11	
11-letter	1,622	0.90	

図 6　ワード・スペクトルの作成



No.	レンマ lemma	ファイル頻度 freq in target	コーパス頻度 freq in corpus	ファイル相対頻度 relfreq in target	コーパス相対頻度 relfreq in corpus	プラスワード得点 plusword score
1	mahon	27	115	20.134	0.011	1,752.659
2	deck	24	1,804	17.897	0.180	99.313
3	pump	22	2,734	16.406	0.273	60.070
4	ship	64	9,448	47.726	0.944	50.568
5	captain	29	5,867	21.626	0.586	36.899

図 7　特徴的高頻度語のリスト



No.	N-gram	頻度 frequency	相対頻度 freq per 10,000w	Z得点 z-score
1	a little	121	9.398	15.327
2	all the	135	10.485	17.115
3	and I	134	10.407	16.988
4	and the	252	19.572	32.060
5	as if	172	13.359	21.841
6	as though	117	9.087	14.816

図 8　Nグラムの頻度リスト

2.3　頻度分析ツール群

第 3 の「Frequency」タブ画面は，対象テキストの頻度計算や分析を行う 9 種類のツール群を収納する。これらのユーティリティツールの目的は，分析対象となる（複数の）テキストのプロファイルや特徴を診断することである。以下では，各ツールを実行したときのスクリーンショットを提示して，その機能を簡単に紹介する。

「SWC カウント」は，対象ファイル中の文，語，文字の数をカウントして平均文長，平均語長を算出する。このツールは文体研究の基礎データを作成する。

「語彙頻度リスト」ツールは，対象テキスト中で使用される語を，表記語形あるいはレンマ形で頻度集計し，相対頻度と z 得点を添えて一覧表示する。

「語長スペクトル」は，対象ファイルに現れる単語の長さの頻度分布表を作成する。村上(1988)で紹介されている

Mendenhall の「ワード・スペクトル」を作成する。

村上(1988)は Ellegard の「プラスワード」も紹介している。「プラスワード」ツールは,対象ファイル中の語彙のうち,モデルコーパスでの出現率と比較して特徴的に高頻度で出現する語のリストを作成する。モデルコーパスは BNC, Brown, LOB の中から選択可能である。

「Nグラム」ツールは,テキスト中に生じる 2 語以上の連語（2 語, 3 語, 4 語）の頻度リストを作成する。

「基本統計」ツールは,ユーザが入力またはペーストしたm行n列のデータを読み込んで,データ数,平均,分散,標準偏差,最大最小,標準誤差,変動係数など 8 種類の基本統計量を表示する。

「χ^2検定」ツールは,たとえば Brown と LOB コーパスのような、等規模の 2 コーパス中における調査語彙の出現頻度を入力すると,(7)の差異係数 D (difference coefficinet)および(8)の χ^2値と適合度検定の結果を表示する。

$$(7)\quad D=\frac{freq(A)-freq(B)}{freq(A)+freq(B)}$$

$$(8)\quad \chi^2=\sum\frac{(O-E)^2}{E}$$

2 コーパスのサイズが異なるときは,「尤度比検定」ツールを用いる。 それぞれのコーパスのサイズと調査語彙の頻度を入力すると,尤度比(log-likelihood ratio)統計量(9)の値と,それに基づく検定結果が表示される。[9]

$$(9)\quad LL=G^2=2\sum O_i\ln\left(\frac{O_i}{E_i}\right)$$

最後に,「G^2 得点」ツールは 2 語連語(bigram)の共起の強さを,上記の尤度比を使って算出する。

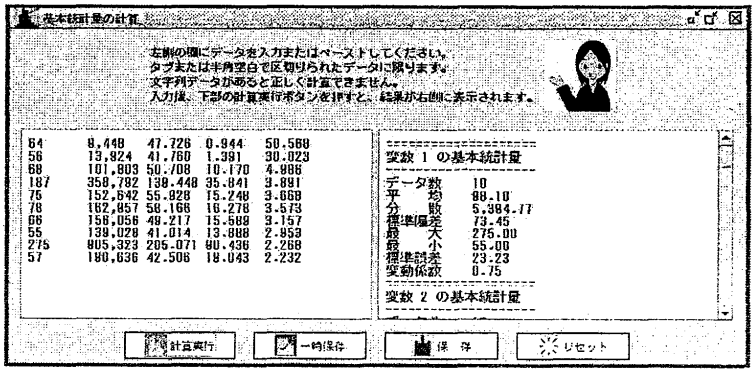


図 9　基本統計量の計算

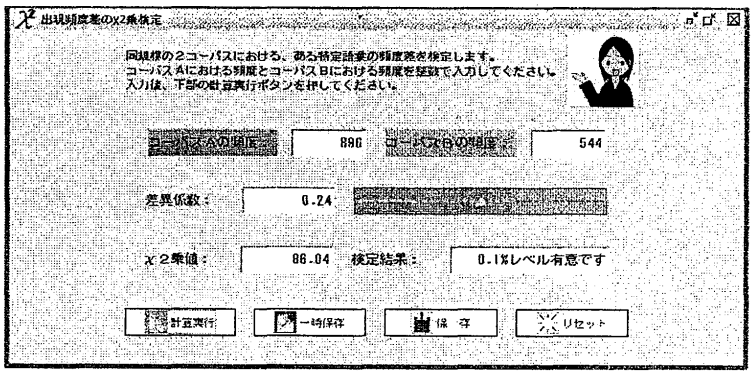


図 10　「χ^2検定」ツール

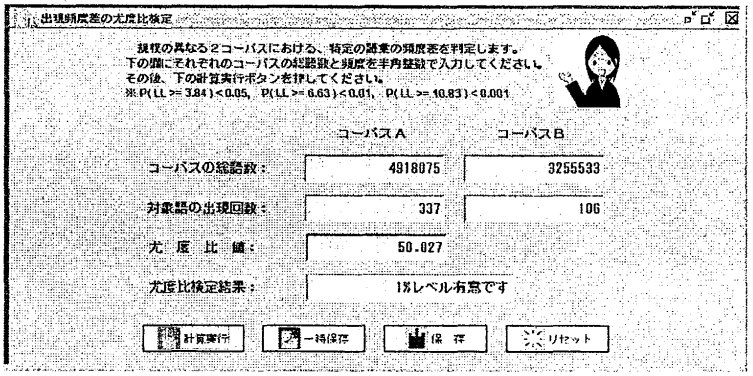


図 11　尤度比検定の画面

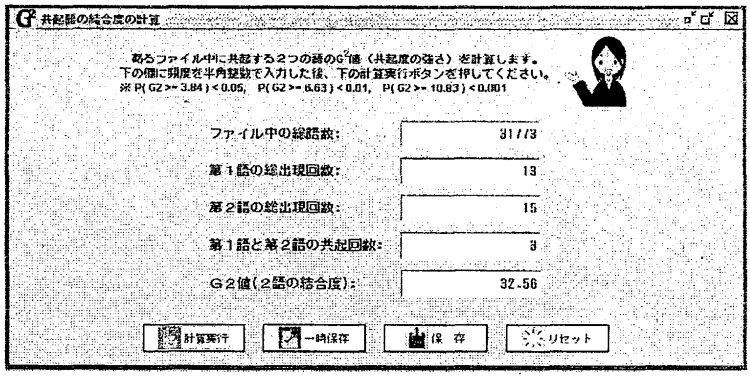


図 12　G^2 得点の算出

3. KWICKERの特徴と課題

この節ではKwicker1.0を特徴づける工夫を簡単に紹介するとともに，今後の課題について触れる。

3.1 Kwickerの特徴

このソフトウェアの特徴はユーザインターフェイスの改善と高速化の試みの２点が挙げられる。

ユーザフレンドリーなインターフェイスを構築するために以下のような工夫をした。

(10) a. go，knowなどの不規則変化動詞の活用形を一括で探索するとき，{go}，{know}のような簡略入力を許す。{go}はリスト参照して(goes|went|gone|going|go)に変換される。

b. 有意なコロケーション統計量を赤色で表示する。

c. 各タブ画面での作業結果をバッファ画面に順次追加出力し，後からの編集ができる。

d. 処理実行前に実行条件をチェックし，問題に応じて警告メッセージを出す。

e. KWIC表示のオプション，コーパスの登録情報，画面サイズなどの現在の設定が保持され，次回起動時にも有効である。

f. 多くの操作をショートカットキーで実行できるので，作業効率が高い。

一方，処理の高速化，パフォーマンスの向上に関して，以下のような改善を図った。

(11) a. 高速なテキスト探索を実現するため，コーパス登録時のテキスト整形，java.nioクラスによるファイル読み込み，正規表現クラスのコンパイルを採用する。これにより，コーパス探索の所要時間が半減された。

b. KWICデータの削除を迅速に処理するためjava.util.LinkedListクラスにKWICデータを格納する。

c. ヒットしたデータを探索終了時に一括表示することにより，約２万件の大量データを約１秒で描画可能である。

d. java.util.Comparatorクラスの採用により，高速なソートが実現されている。

3.2 今後の課題

今後 Kwicker の中に取り入れたい機能，充実が望まれる機能など，現時点で主要な課題と認識している点は以下の通りである。

(12) a. プログラムコードの信頼性を高めるため，JUnitを活用して十分なテストを行う。

b. 多様なユーザを想定し，ヘルプファイルの内容を一層充実させ，理解し易くする。

c. エディタジャンプ機能およびコロケーション統計量G^2 得点やMI3得点の算出機能を実装する。

終わりに，Kwicker1.0 の公開に当たり抱負を述べたい。Kwicker1.0 は，任意のテキストファイル群をコーパスとして登録する機能を備えている。現在，Project Gutenbergをはじめとする WEB サイト上には，著作権が消滅したり，著者がその権利を放棄した膨大な量のテキストが公開されていて，だれでもが自由にアクセスできる状況がある。Kwicker1.0 は，そのようなテキストを対象としたコーパス言語学的研究の可能性を提供する。Kwicker1.0 がそのような研究に資するならば開発者の幸せである。今後とも，ユーザのニーズを取り入れて，機能の拡張や性能の向上を図りたい。

注

[1] KWIC とは Key Word in Context の略で，探索テキスト中でキーワードが使われている箇所を，キーワードを行の中央に配置して，その前後文脈とともに表示する，コンコーダンスの一種である。Kwicker という名称は「KWIC を作り出すもの」に由来する。同時に，「より素早く」(quicker) KWIC を表示するという期待も込められている。

[2] ファイル選択が確定すると，選択されたファイルへのパス文字列が，ファイル欄に表示される。一度使用されたファイルパスの情報はプルダウンメニュー内の選択肢に追加されていき，次回以降のファイル指定の際に利用することができる。

[3] 「探索とKWIC」メニューから「コーパス登録(R)」を選択すると「コーパスファイルの登録」ダイアロ

グが表示される。その画面の指示に従って，最大 12 個までコーパスを登録することができる。登録コーパスの諸情報は，アプリケーション終了時に保存され，起動時にその情報が読み込まれる。コーパスの登録は一度だけでよく，起動のたびに行う必要はない。

コーパス登録には，対象ファイル指定の煩雑さから解放されるばかりでなく，コーパス探索が高速化されるという大きな利点がある（3.1 節参照）。

[4] Java 2 SDK1.4 以降，java.util.regex クラスが導入され，待望の正規表現がサポートされるようになった。Perl5.0 以降ほどの充実度には達していない(Fiedl(2003))としても，正規表現クラスが標準サポートされるようになった意義は，テキスト処理関連のソフトウェア開発にとっては，極めて大きい。

Kwicker1.0 では，正規表現の前後をスラッシュで囲む必要を省き，入力量の軽減を図っている。

[5] デフォルトでは探索テキストの言語は英語に設定されている。言語選択ボタンをクリックして，日本語テキストへの対象変更が可能である。

探索実行に必要な情報が指定されていないときは状況に応じて警告ダイアログが表示される。各種のメッセージをユーザへ表示する際に，本文中図 1 にあるような女性のイラストを配置した。イラストやアイコンの作成には，本校情報科学専攻情報デザイン分野所属の伊藤優華さんと森山聡子さんにご協力をいただいた。ここに記して感謝したい。

[6] それぞれの統計量を利用する際の注意点については Barnbrook(1996)や McEnery et al. (2006, pp. 208-225)に比較的詳しい記述がある。また，Oakes(1998, pp. 162-193)には，10 種類以上のコロケーション統計量が紹介されている。

[7] コロケーション統計量を計算する際，通常の環境下における当該コロケート語の出現確率データが必要となる。登録済みコーパスから得られた KWIC データには，登録時に作成したそのコーパスの語彙頻度データが使用される。それ以外の場合は，A. Kilgariff 氏が作成した BNC の語彙頻度データ all.al.o5 および lemma.al を編集したデータを利用している。

[8] Mason(2000, ch. 11)は，これら 3 種類の統計量を同時に表示し，その一つをキーにしてデータのソートも行うというプログラムデザインを提示しており，参考になる。ただし，Mason のプログラムが CUI であるのに対し，Kwicker1.0 は彼のデザインを GUI 方式で実装したものである。

[9] χ^2 乗検定と異なり，尤度比検定は頻度分布の正規性を仮定しないため，小規模コーパスの頻度差の判定にも使える(Dunning(1993))。なお，P. Rayson 氏が尤度比検定を行うツールを WEB 上に公開している。

参照文献

Barnbrook, G. 1996. *Language and Computers: A Practical Introduction to the Computer Analysis of Language*. Edinburgh: Edinburgh University Press.

Dunning, T. 1993. "Accurate Methods for the Statistics of Surprise and Coincidents," *Computational Linguistics*, 19, 61-74.

Fiedl, J. 2003.『詳説正規表現　第２版』, 東京：オライリー・ジャパン.

Kilgariff, A. "BNC database and word frequency lists" URL: http://www. kilgarriff. co.uk/bnc-readme.html

Mason, O. 2000. *Programming for Corpus Linguistics*. Edinburgh: Edinburgh University Press.

McEnery, T., R. Xiao and Y. Tono. 2006. *Corpus-Based Language Studies: An Advanced Resource Book*. London: Routgedge.

村上　征勝 1988.「著者はだれか？計量文献学への招待 1, 2」,『数学セミナー』1988 年 11 月号 pp. 55-59, 12 月号 pp. 74-79.

Oakes, M. P. 1998. *Statistics for Corpus Linguistics*. Edinburgh: Edinburgh University Press.

Rayson, P. "Log-likelihood calculator" URL: http://ucrel.lancs.ac.uk/llwizard.html