「森を見ながら木を見る」コーパス研究の意義 : 複数テキストから統合語彙頻度表を作成するEJWFTGの開発

石川, 慎一郎

(Citation)
統計数理研究所共同研究リポート,469:95-122

(Issue Date)
2024-03-25

(Resource Type)
departmental bulletin paper

(Version)
Version of Record

(JaLCDOI)
https://doi.org/10.24546/0100487709

(URL)
https://hdl.handle.net/20.500.14094/0100487709

「森を見ながら木を見る」コーパス研究の意義

―複数テキストから統合語彙頻度表を作成する EJWFTG の開発―

石川　慎一郎（神戸大学）

Seeing Both the Wood and the Trees in Corpus Studies

―Development of An Automatic Word Frequency Table Generator, EJWFTG―

ISHIKAWA, Shin'ichirio (Kobe University)

Abstract

This paper first discusses the need to see both the wood and the trees―namely, both group and individual data―in corpus studies. Then, it introduces the English/Japanese Word Frequency Table Generator (EJWFTG), a newly developed online application that automatically produces integrated word frequency tables from plural text files. Finally, it touches upon several applications of the frequency tables obtained from the EJWFTG.

キーワード

マージデータと個体データ、頻度表、形態素解析、検定、多変量解析

1. はじめに

応用言語学では、言語変種の特性や、変種間の差異の解明が重要な課題となる。たとえば、第2 言語習得研究では母語話者と学習者の、ジェンダー言語研究では男性話者と女性話者の、LSP（専門目的言語）研究では異なるジャンルの言語を取り上げ、その位相を探る。こうした場合、幅広いテキストを観点ごとに集約したマージファイルを作って比較することが一般に行われる。

この場合、分析の対象になるのは変種というまとまりであり、たとえて言えば、1 つの森に関心が向けられている。しかし、森の実態は雑多な樹木の集合であり、その中には、高い木もあれば低い木もあり、常緑樹もあれば落葉樹もある。このことをふまえれば、言語変種研究においても、変種をまとまりでとらえるだけでなく、変種に属する個々のテキストのありようを細かく観察する姿勢、言い換えれば、「森を見ながら木を見る」アプローチが重要になるだろう。

こうしたアプローチを取る場合、まずもって必要になるのは、数十種、時には、数百種におよぶテキストごとに、そこに含まれるすべての語の頻度を調べ、それらを全テキスト間で相互比較できるようにした統合語彙表の作成である。しかし、個々のテキストから作成した頻度表を、エクセルなどで加工して 1 つの頻度表にまとめていくのは膨大な時間がかかる。また、語彙頻度を扱う場合は、出現形（表層形、書字形、表記形）と集約形（語彙素、レマ）といった単位の違いや、個々の語の品詞

にも着目する必要があるが、これらの処理もきわめて煩瑣なものとなりうる。

そこで、筆者は、一連の作業を自動化する English/Japanese Word Frequency Table Generator（EJWFTG）を開発し、2024 年の 3 月末に正式公開した。EJWFTG は、(1)日本語・英語の 2 言語に対応し、(2)OS 環境を問わず稼働し、(3)出現形から集約形への変換と品詞情報の付与を自動で処理する統合語彙頻度表作成ツールである。EJWFTG は、Google Colaboratory 上で作成されており、Python で処理が行われる。

本稿は、EJWFTG が貢献しうる多様な研究領域のうち、学習者コーパス研究を取り上げ、習得研究における「森を見ながら木を見る」アプローチの必要性、EJWFTG の具体的な操作手順、さらには、EJWFTG から得られた統合語彙頻度表の研究応用について概説する。

2. 学習者コーパス研究におけるデータ

学習者コーパス研究では、通例、個別学習者の言語使用よりも、母語や習熟度を単位としてまとめられた学習者群の言語使用の傾向の解明が優先される。実際、「A 君が作文で I think を 3 回使っていた」や、「B さんはテイル形を使えていなかった」といったことがわかっても、それらは再現性の保証されない個人特性に過ぎず、得られた知見を一般化しにくいからである。これに対し、群を議論の単位として、「日本人英語学習者は母語話者よりも I think を多用する」や、「初級日本語学習者は上級学習者よりもテイル形の使用率が低い」といったことがわかれば、知見の有用性は高まる。

学習者コーパス研究が群の議論を志向する背景には、コーパス研究全体の志向性がある。自分の主張したい内容に合致する 1 例ないし数例を見つけて、それらを主観的に分析する伝統的な用例研究へのアンチテーゼとして、コーパス研究は、匿名化されたデータを大量に収集し、それらを網羅的に調査することで、研究手法の客観化と、得られる知見の一般化の向上を目指してきた。このため、学習者コーパス研究においても、マージデータが広く使用されてきたわけである。

マージデータを用いた分析にはメリットとデメリットの両面がある。まず、メリットとしては、処理するファイル数が減るため、形態素解析や頻度計量の手間が大きく省力化される。また、多数の学習者の産出をまとめることで、幅広い語について一定の頻度値が得られやすい。さらに、あらかじめ議論したい観点で群化しているため、結果の解釈も行いやすい。一方、デメリットは、個人を見ないことで誤った結論を導くリスクが存在することである。たとえば、上級日本語学習者 100 名の作文マージデータを調べて、テイル形が 300 回使用されていた場合、「上級学習者はテイル形を多用する」という主張がなされるわけだが、もし、10 人が 30 回ずつテイルを使っていて、残り 90 名が一度も使っていないのだとすれば、あるいは個人により使用頻度のばらつきが極度に大きいのだとすれば、上級学習者が群として真にこうした傾向を持っていたかどうかは疑わしい。

このように考えてくると、他の変種研究と同じく、学習者コーパス研究においても、所属する群の情報を生かしながら、個々人の産出を丁寧に調べる、つまりは、「森を見ながら木を見る」アプローチが重要になってくると言えるだろう。こうしたアプローチで研究を行う場合、前述のように、個々の学習者がどの語を何回使っているかを一覧できる統合語彙頻度表が必要となる。また、語彙の多

層構造を考慮すると、テキストに出ている出現形の頻度表だけでなく、それらを基本形にまとめた集約形の頻度表、さらには、品詞の情報も重要になるだろう。出現形は、英語学では表記形（wordform）、日本語学では表層形や書字形と呼ばれる。また、複数の出現形をまとめた集約形は、英語学ではレマ（lemma）、日本語学では語彙素と呼ばれることが多い。

表 1
語彙の三層構造

	第 1 層	第 2 層	第 3 層
分析単位	出現形	集約形	品詞タイプ
英語	表記形 （例：play, plays, played, playing）	レマ （例；play）	品詞 （例：一般動詞）
日本語	表層形・書字形 （例：し、する、すれ、しろ）	語彙素 （例：為る）	品詞 （例：サ変動詞）

従来、こうしたマルチレベルの頻度表の作成は作業負荷が大きかった。一般的な作業手順としては、はじめに、タガーや形態素解析器でテキストファイルを加工し、出現形を記したテキスト、集約形を記したテキスト、品詞を記したテキストを用意する。その後、それぞれをコンコーダンサで処理し、頻度表を作成する。この作業を分析したいファイルの数だけ繰り返し、最後に、得られた頻度表を 1 つのエクセルファイルに転写し、エクセル関数などを使って同じ語の頻度をまとめて同じ行に表示する。数個のファイルならなんとかなるとしても、大量のファイルからマルチレベルの統合語彙頻度表を作成するのは煩瑣な作業となる。

こうした作業を自動化するツールとしては、今尾康裕氏の制作による CasualConc などがある。CasualConc は統計解析を含む多機能コンコーダンサで、複数ファイルを入力すると、行方向にテキスト、列方向に単語が配置された統合語彙頻度表を自動作成する機能も備えるが、Mac 専用のため、Windows 環境では使えない。

そこで、筆者は、CasualConc の語彙頻度表作成機能を手本として、OS 環境を問わず、マルチレベルの統合語彙頻度表を自動作成する English Japanese Word Freuqncy Table Generator（EJWFTG）の開発に着手した。以下、EJWFTG（2024/3 版）の概要と、EJWFTG で出力される統合語彙頻度表を使った研究例を示す。

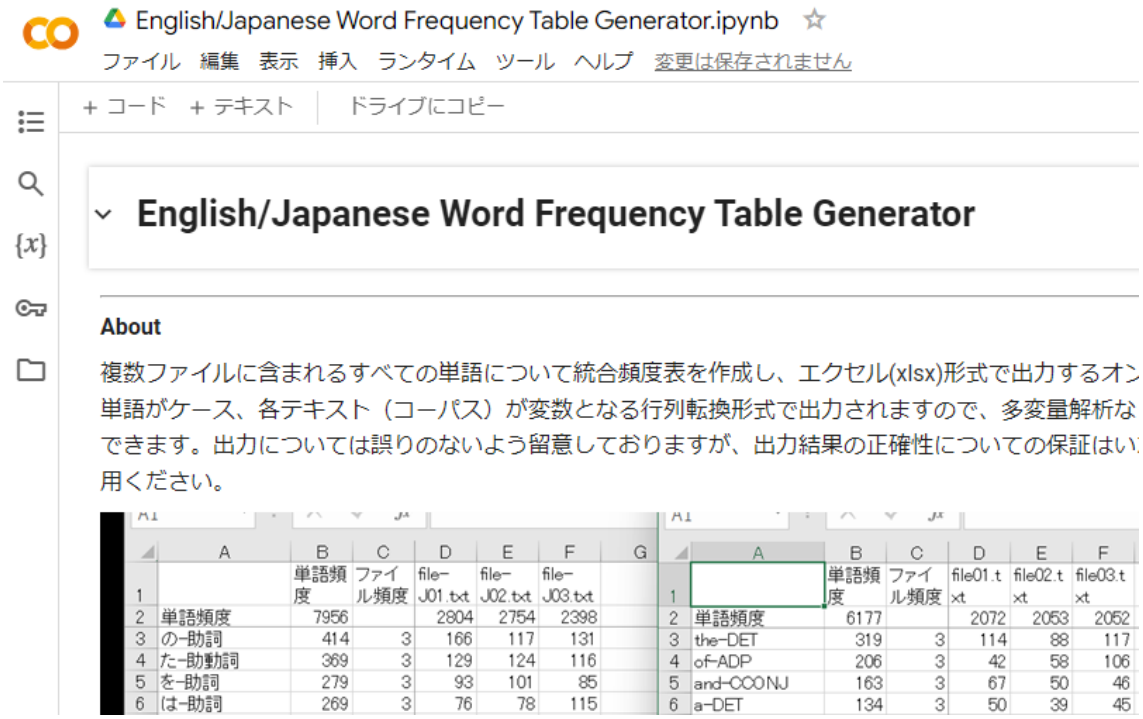
3. EJWFTG を用いた統合語彙頻度表作成

以下、EJWFTG を使うための事前準備と、具体的な操作過程について概観する。なお、本節の記述は 2024 年 3 月時点のバージョンに基づくもので、今後のアップデートにより、出力される頻度値や操作系の一部が若干変わる可能性があるので留意されたい。

EJWFTG の URL にアクセスすると、以下のような画面が表示される。

図 1

EJWFTG のスタート画面



	A	B	C	D	E	F	G
1		単語頻度	ファイル頻度	file-J01.txt	file-J02.txt	file-J03.txt	
2	単語頻度	7956		2804	2754	2398	
3	の-助詞	414	3	166	117	131	
4	た-助動詞	369	3	129	124	116	
5	を-助詞	279	3	93	101	85	
6	は-助詞	269	3	76	78	115	

	A	B	C	D	E	F
1		単語頻度	ファイル頻度	file01.txt	file02.txt	file03.txt
2	単語頻度	6177		2072	2053	2052
3	the-DET	319	3	114	88	117
4	of-ADP	206	3	42	58	106
5	and-CCONJ	163	3	67	50	46
6	a-DET	134	3	50	39	45

テキストアップロードから、作成された頻度表のダウンロードまで、すべての操作は、上記の単一の画面上で行うことができる。

3.1 用意するデータ

EJWFTG を使用するにあたり、利用者側で用意するのは、UTF-8 形式で保存されたテキストファイルである。サイズの小さいファイルなら 1,000 個でも 10 分程度で処理可能である。ファイル内のすべての文字を処理するため、たとえば、ファイル内に発話者コードなどが記載されている場合は、あらかじめ取り除いておくことが望ましい。

3.2 留意事項

EJWFTG の一般的な留意事項として、以下の点があげられる。下記は、EJWFTG 画面からの引用である（以下、囲み内はすべてツール画面からの引用）。

> (1) 本ツールの実行には Google アカウントへのログインが必要です。
> (2) 本ツールは、各自の Google ドライブ内にプログラム（仮想マシン）を作成して各種の処理を行います。利用者のパソコンには何ら変更を加えません。
> (3) アップロードしたデータは、各自の Google ドライブ上に格納されますので、外部から見られることはありません。

(4) 仮想マシンは使用開始 90 分後に自動で接続が遮断されます。90 分より後に、別のファイルをアップロードして処理を行おうとする場合は、手順 2 から改めて実行ください。
(5) 本ツールで使用できるのは、UTF-8 のエンコード形式で保存されたテキストファイルのみです。Word ファイルや PDF からの頻度表作成はできません。なお、テキストファイルが UTF-8 以外（shift-jis など）で保存されている場合は、あらかじめ、エンコードを UTF-8 に修正しておいてください。
(6) 動作不良の際は、ツールのリセットを行います。メニューの「ランタイム」→「ランタイムを接続解除して削除」を押して、手順 2 からやり直してください。

このうち、とくに重要なのは(5)で示したテキストの UTF-8 化である。テキストファイルにはさまざまなエンコード形式があり、英語では ANSI、日本語では shift-jis などでエンコードされたものも多いが、EJWFTG は UTF-8 のみの対応となっている。

てもとのファイルが UTF-8 でない場合は、たとえば、MS Word で「名前をつけて保存」を押し、ファイルの種類を「書式なし（*.txt)」にした後、エンコード方法をその他→UTF8 にする必要がある。なお、この手順は、EJWFTG にリンクされたビデオでも解説している。

図 2
MS Word 上での「名前をつけて保存」画面

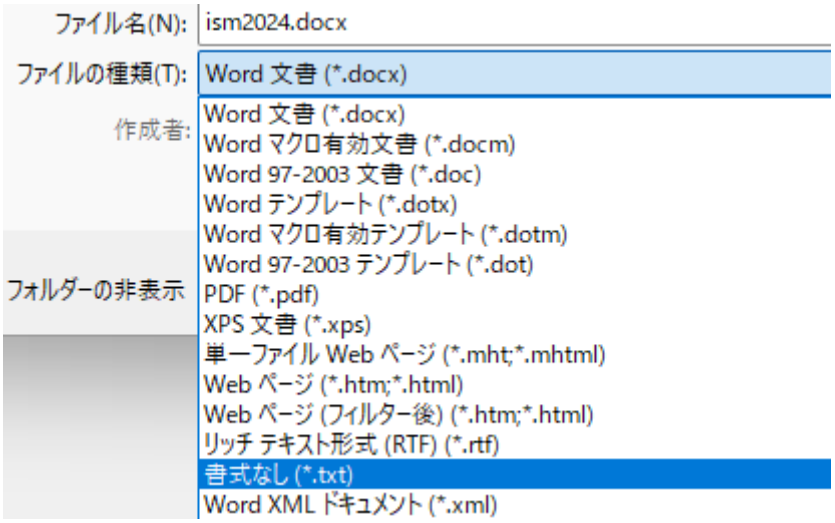
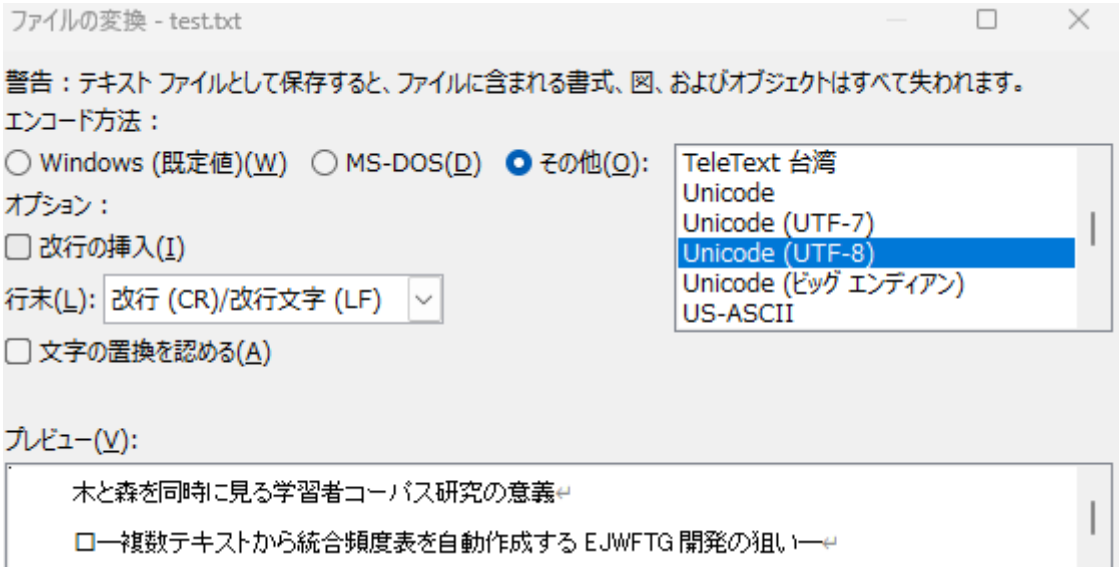


図 3

テキストファイル形式の選択画面



以上の手順を経ることで、ファイルを UTF-8 形式で保存することができる。また、UTF-8 以外のファイルが大量にあり、それらを一括で UTF-8 に変換したい場合は、Laurence Anthony 氏が開発した Encode Ant などの専用ツールを使用することもできる。

3.3 処理のステップ

3.3.1 Step 1 ツールコピー

最初に行う作業は、ツールを利用者自身の Google ドライブにコピーすることである。これにより、各自の Google アカウント内で処理が実行されるようになる。

Step 1. 各自の Google ドライブにツールをコピーする (1) Google にログインする。 (2) 画面上部にある「ドライブにコピー」を押す。 (3) 「コピーを作成しています」というメッセージが出て、画面が一度白くなり、新しい画面が表示されれば、コピーは完了しています。以後は、新しくできた画面の上で作業を続けていきます。 (☛Step 1 の実行例動画(40 秒))

3.3.2 Step 2 日本語形態素解析環境の準備

EJWFTG では、日本語テキストの処理に先立ち、MeCab および UniDic を外部から読み込んで、単語の分割や品詞の決定を行う。よって、日本語テキストを扱う場合のみ、このための環境設定が事前に必要となる。

Step 2. 日本語処理環境を用意する(※日本語データを扱う場合のみ実行)下記の▸ボタンを押す。
(1)「警告: このノートブックは Google が作成したものではありません」と表示されますが、「このまま実行」を押してください。
(3) MeCab(処理エンジン)→UniDic(形態素解析辞書)の順序で設定を行います。
(3) Downloaded UniDic v3.1.0...to... という行が表示され、▸ボタンの回転が止まり、▸ボタンの横に緑のチェックマークと所要時間が表示されれば Step 2 は完了です。(☛Step 2 の実行例動画(70 秒))

「コードの表示」を押すと、進行中の作業を確認することができる。

図 4
「コード表示」で UniDic 辞書のダウンロードを確認

```
(UniDic辞書のダウンロード)
download url: https://cotonoha-dic.s3-ap-northeast-1.amazonaws.com/unidic-3.1.0.zip
Dictionary version: 3.1.0+2021-08-31
Downloading UniDic v3.1.0+2021-08-31...
unidic-3.1.0.zip: 100% 526M/526M [00:28<00:00, 18.2MB/s]
Finished download.
Downloaded UniDic v3.1.0+2021-08-31 to /usr/local/lib/python3.10/dist-packages/unidic/dicdir
```

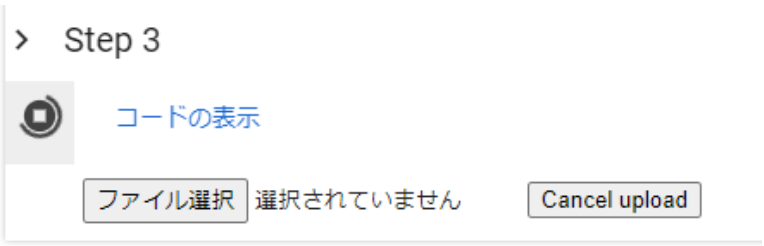
3.3.3 Step 3 データのアップロード

続いて、分析したいデータをアップロードする。以下では、英語学習者コーパス ICNALE (Ishikawa, 2023)の Written Essays モジュール(v 6.2)に含まれる、日本人学習者 400 名によるアルバイト(PTJ)作文 400 本を指定する。これらは、"It is important for college students to have a part-time job"というテーマに対する意見を 200-300 語の長さで書いたものである。

Step 3. テキストファイルをアップロードする
(1) 下記の▸ボタンを押す。
(2)「ファイル選択」ボタンが表示されるのでこれを押し、各自のパソコン内にあるテキストファイルを指定する(複数同時指定可)。
(3) User upload file "…" with length …bytes という行が表示され、▸ボタンの回転が止まり、▸ボタンの横に緑のチェックマークと所要時間が表示されれば Step 3 は完了です。(☛Step 3 の実行例動画(40 秒))
(4) アップロード済みのファイルを削除して別のファイルをアップロードしたい場合は、画面左端の上から 5 つ目(鍵マークの下)にあるファイルアイコンを押し、不要なファイルを指定して右クリック→削除。その後、上記の(2)から操作を行う。

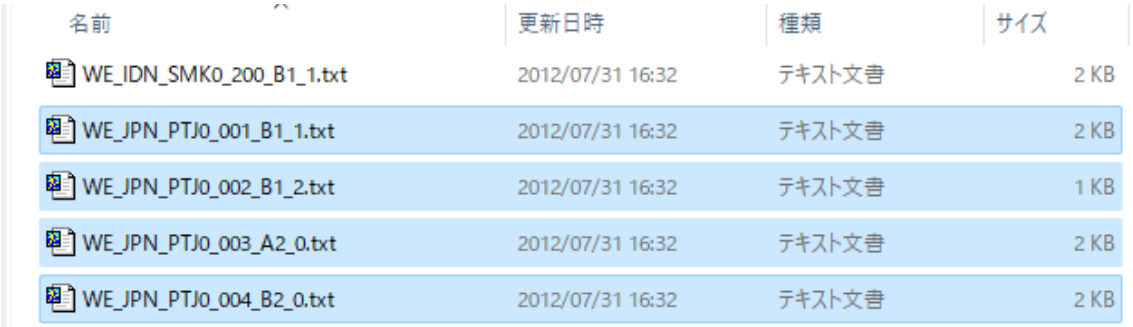
上記(1)のステップを行うことで、下記の画面が表示される。

図 5
アップロードファイル指定のための「ファイル選択」



「ファイル選択」を押し、自身のパソコン内にあるファイルを指定する。Ctrl+A などでフォルダ内にある全ファイルを一括して指定を行うこともできる。

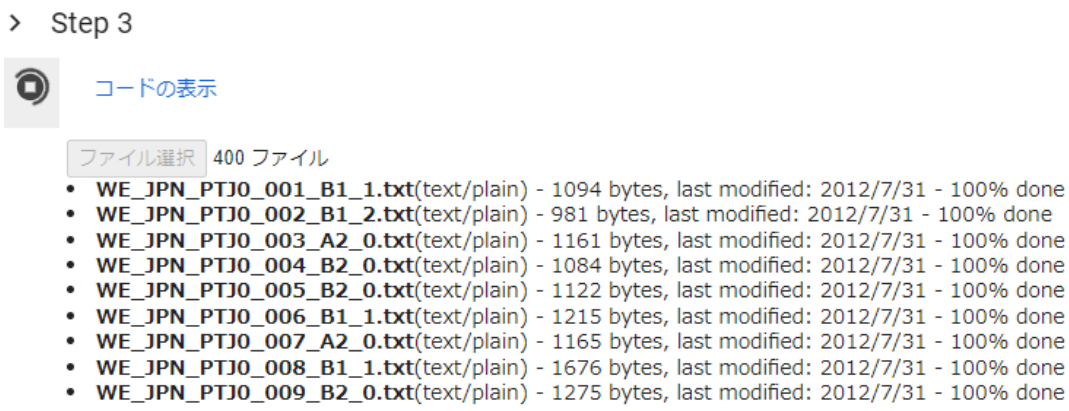
図 6
パソコン内にあるファイルの指定



指定が終わると、自動的にファイルのアップロードが始まる。

図 7

ファイルのアップロードと読み込み



400 本のファイル（各々は 200-300 語の英文。ファイルサイズは 1000-1300 バイト程度）をアップロードするのに必要な時間は、筆者の環境では 6 分 30 秒であった。この時間は、コンピュータの性能や、ネット接続の速度によって変わる。

3.3.4 Setp 4 テキスト解析

このプロセスは、Step 4.1 日本語解析と、Step 4.2 英語解析に分かれる。日本語テキストを扱う場合は、先に Step 2 で導入した MeCab/UniDic で処理が行われる。

英語テキストを扱う場合は、この段階で Python 用のオープンソースの自然言語処理ライブラリである spaCy が読み込まれる。spaCy は MIT ライセンス（著作権および許諾表示を記載すれば、非営利、商用を問わず、使用、改変、複製、再頒布が可能）で公開されている。本稿執筆時点において、EJWFTG が用いる spaCy のバージョンは 3.7.4、英語解析用のパイプライン（※パイプラインとは、直列に連結された処理プログラムの意）は en_core_web_sm（3.7.1）である。英語のパイプラインは 4 種存在するが、ここで使用しているものは、英語（English）のタグ付け・構文解析・レマ化といった汎用目的に即したコアデータ（core）で、ブログ・ニュース・各種コメントといったウェブテキスト（web）からトレーニングされた小規模（small）の解析辞書である。

図 8

spaCy、パイプラインのバージョン表示

```
============================== Info about spaCy ==============================

spaCy version    3.7.4
Location         /usr/local/lib/python3.10/dist-packages/spacy
Platform         Linux-6.1.58+-x86_64-with-glibc2.35
Python version   3.10.12
Pipelines        en_core_web_sm (3.7.1)
```

ここでは、英語テキストを扱うため、Step 4.2 を選ぶ。

Step 4.2 英語テキスト解析（UTF-8 のみ） (1) 標準で大文字と小文字は区別しません（＝大文字をすべて小文字化します）が、区別したい場合は下記▸ボタン右の「コードの表示」を押し、プログラムの 4 行目を case_sensitive = 1 に変更してください。 (2) 下記の▸ボタンを押す。 (3) ▸ボタンの回転が止まり、▸ボタンの横に緑のチェックマークと所要時間が表示されれば Step 4 は完了です。 (4) 途中でエラーになる場合は、アップロードしたファイルが UTF-8 でエンコードされていたかどうか再度確認ください。

英語の場合、デフォルトでは、he、He、hE などを区別せず、すべて he として処理するようになっている。このため、I も i と処理される。大文字と小文字を区別したい場合は、「コードの表示」を押すと、以下のような画面が表示されるので、case _sensitive = 0 の「0」を「1」に置き換える。

図 9
大文字・小文字区分のためのコードの修正

Step 4.2

```
#@title Step 4.2

####
case_sensitive = 0  # 大文字と小文字を区別したい場合は 1 に変更する
####

print ("(英語ライブラリのインストール)")
import spacy
nlp = spacy.load("en_core_web_sm")
```

このステップで、英語テキストをレマ化し、個々の語に品詞情報の付与を行う。筆者の環境では、400 本のファイルの処理に要した時間は約 2 分であった。

3.3.5 Step 5 頻度表の出力

最後のステップは頻度表の出力である。

Step 5. 頻度表をダウンロードする (1) 下記の▸ボタンを押す。

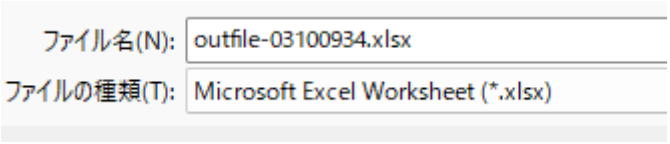
(2)「名前をつけて保存」ウィンドウが表示される場合は、デフォルトの名前（outfile-03110915.xlsx など。数字は月日時分)を任意の名前に変更して「保存」を押す。(☛Step 5 の実行例動画(50 秒))

▸ボタンを押すと、下記のような画面が表示される。

図 10
頻度表のダウンロード準備

プログレスバーが右端に達すると、下記の画面がポップアップで表示される(この画面が出るかどうかはコンピュータの設定による)。

図 11
「名前をつけて保存」ウィンドウ



上記は、(2024 年の)3 月 10 日の午前 9:34 に操作が完了したため、03100934 というコードとなっているが、これは自由に修正することができる。たとえば、ICNALE_JPN_PTJ_all.xlsx などと変更しておけばよいだろう。以上で統合語彙頻度表の作成作業は終了である。

4. 統合語彙頻度表の概要

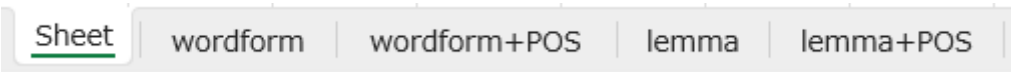
EJWFTG の特徴の 1 つは英語と日本語の 2 言語に対応している点である。以下、4.1 節では英語データから得られた頻度表を、4.2 節では日本語データから得られた頻度表をそれぞれ示し、頻度表の構成や読み方を示す。

4.1 ICNALE 作文データに基づく統合語彙頻度表

ここでは、3 節で作成した ICNALE の日本人学習者によるアルバイト作文 400 本に基づく統合語彙頻度表の内容を概観する。まず、出力されるエクセルファイルには、以下の 5 枚のシートが含

まれる。

図 12
英語統合語彙頻度表のファイル構造



冒頭の Sheet には処理設定の詳細が記録されている。

図 13
冒頭 Sheet の記載内容

解析モード：英語	
case_sensitive：0	
解析ファイル数：400	
	WE_JPN_PTJ0_001_B1_1.txt
	WE_JPN_PTJ0_002_B1_2.txt
	WE_JPN_PTJ0_003_A2_0.txt
	WE_JPN_PTJ0_004_B2_0.txt
	WE_JPN_PTJ0_005_B2_0.txt

冒頭行には解析モード（解析された言語種別）が示され、2 行目には case sensitive（大文字・小文字を区別するか否か）の状態が示される。0 は区別しない、1 は区別する、の意味である。
3 行目には、解析されたファイルの総数が示され、4 行目以降に、解析されたファイル名が一覧表示される。
次に、テキストでの出現形である wordform（表記形）に基づく頻度表の一部を見てみよう。

図 14

表記形頻度表

	単語頻度	ファイル頻度	WE_JPN_PTJ0_001_B1_1.txt	WE_JPN_PTJ0_002_B1_2.txt	WE_JPN_PTJ0_003_A2_0.txt	WE_JPN_PTJ0_004_B2_0.txt
単語頻度	91076		214	190	213	212
to	3361	398	7	11	7	8
i	2866	399	18	3	6	11
a	2734	391	8	11	10	4
time	2505	400	6	8	2	3
is	2125	396	4	7	2	3
part	2114	400	5	8	2	3

はじめに、表記形頻度表に注目する。400 ファイルの総語数は 91,076 語である。また、3 行目以降に個々の語が並ぶが、それらの行数を数えれば語種総数もわかり、今回の場合で言うと 2,978 種となる。なお、処理に当たって、各種の記号やアラビア数字はスペースに読み替えて処理している(例:I'll→i/ll)。

まず、単語の側に注目すると、最も高頻度の語は to(3,361 回)で、以下、i(2,866 回)(※デフォルト設定ではすべてを小文字として扱うため、I も i として出力される)、a(2,734 回)などが上位に並ぶ。単語頻度の右側にはファイル頻度の列がある。これは、全 400 種のファイルの中で、当該語が 1 回以上出現しているファイル数を示し、コーパス研究では、ふつう、これをレンジ(range)と呼ぶ。レンジは百分率で示すこともあり、たとえば、to のレンジ比率は 398/400=99.5%と計算される。重要語の抽出などでは、頻度とともにレンジに注目することが重要である。上記の a と is を比べると、頻度は a のほうが高いが、レンジは is のほうが高く、つまりは is のほうがより幅広い学習者に使用されていることがわかる。頻度だけでなくレンジの情報を取り出せるのが、個々の木を見る―つまりは、個体データを処理する―大きな利点と言える。高頻度語の大半は、to、I、a、is といった基本語で、part と time は、作文トピックとして示された part-time job の一部である。

続いて、個別ファイルの頻度に着目してみよう。上記に示された JPN_001～004 のファイルの総語数はそれぞれ 214 語、190 語、213 語、212 語となっている。ICNALE は 1 つの作文の長さが 200～300 語(±10%)と決められており、すべてのファイルがこの範囲に入っているわけだが、それでも、ファイルにより(書き手により)作文の長さには一定の違いがあることが確認できる。

ここで注目すべきは、作文長だけでなく、個別語の頻度についても、学習者によってかなり大きな違いが存在することである。4 人の中での最小・最大頻度に注目すると、to の場合は差が小さいが(7 vs 11)、I は 6 倍(3 vs 18)、a は 2 倍以上(4 vs 11)、time は 4 倍(2 vs 8)、is は 3 倍以上(2 vs 7)、part は 4 倍(2 vs 8)の差が開いている。I に関して、400 人全体をまとめて森として見れば「日本人英語学習者は I を多用する」と言えそうであるが、個々人を 1 本の木として見れば、そうした傾向に合致しない学習者も一定数存在することが見えてくる。

続いて、wordform+POS（表記形＋品詞）のシートを見てみよう。

図 15

表記形＋品詞頻度表

	単語頻度	ファイル頻度	WE_JPN_PTJ0_001_B1_1.txt	WE_JPN_PTJ0_002_B1_2.txt	WE_JPN_PTJ0_003_A2_0.txt	WE_JPN_PTJ0_004_B2_0.txt
単語頻度	91076		214	190	213	212
to-PART	2908	398	7	11	4	6
i-PRON	2866	399	18	3	6	11
a-DET	2734	391	8	11	10	4
time-NOUN	2501	400	6	8	2	3
is-AUX	2088	395	4	7	2	3

表記形＋品詞頻度表においては、個々の語に対して、spaCy の解析結果に基づく頻度コードが付与されている。to は PART（particle：不変化詞）、i（I）は PRON（pronoun：代名詞）、a は DET（determiner：決定詞）、time は NOUN（名詞）、is はコピュラとしての AUX（auxiliary verb：助動詞）である。

同じ表記形でも、別品詞として解釈されるものもあり、表記形頻度表と比べた場合、一部の語の頻度は変化している。たとえば、表記形 to の頻度は 3,361 回であったが、to-PART の頻度は 2,908 回である。これは、全体で 3 種類の to が存在していたためである。

図 16

to の品詞別頻度

	単語頻度	ファイル頻度	WE_JPN_PTJ0_001_B1_1.txt	WE_JPN_PTJ0_002_B1_2.txt	WE_JPN_PTJ0_003_A2_0.txt	WE_JPN_PTJ0_004_B2_0.txt
単語頻度	910		2	1	2	2
to-PART	2908	398	7	11	4	6
to-ADP	448	244	0	0	3	2
to-AUX	5	5	0	0	0	0

なお、spaCy の解析精度は総じて高いが、100％ではない。品詞の判断に関して、誤解析の可能性がありうることにはあらかじめ留意されたい。

続いて、動詞活用形などを基本形に繰り込んだ lemma（レマ）頻度表の一部を見てみよう。

図 17

レマ頻度表

	単語頻度	ファイル頻度	WE_JPN_PTJ0_001_B1_1.txt	WE_JPN_PTJ0_002_B1_2.txt	WE_JPN_PTJ0_003_A2_0.txt	WE_JPN_PTJ0_004_B2_0.txt
単語頻度	91076		214	190	213	212
be	3567	400	11	9	3	6
to	3361	398	7	11	7	8
i	3043	399	19	3	6	12
a	2734	391	8	11	10	4
time	2540	400	6	8	2	3

レマ化しても総語数（91,076 語）は変わらないが、活用形がまとめられることで語種数は 2,978 種から 2,313 種にまで圧縮される。表記形の頻度 1 位は to であったが、レマ化を行うことで、is、am、are などの頻度が 1 つに統合され、レマの頻度 1 位は be に変わっている。

なお、レマ化には、何をどこまで丸めるか（基本形に繰り込むか）について様々な立場・考え方があり、絶対的な正解というものは存在しない。EJWFTG は spaCy の基準に基づいており、動詞の活用形などは基本形にまとめられるが、an が a にならない、my が I にならない、its が it にならない、といった注意点もある。工学的なテキスト処理では、判断の分かれるレマ化については慎重な立場が多く、スペル揺れ以外の集約化は避けるべきだという意見も見られる（https://github.com/UniversalDependencies/docs/issues/517）。

下記は、lemma（レマ）+POS（品詞）頻度表の一部である。

図 18

レマ＋品詞頻度表

	単語頻度	ファイル頻度	WE_JPN_PTJ0_001_B1_1.txt	WE_JPN_PTJ0_002_B1_2.txt	WE_JPN_PTJ0_003_A2_0.txt	WE_JPN_PTJ0_004_B2_0.txt
単語頻度	91076		214	190	213	212
be-AUX	3320	399	10	9	3	5
i-PRON	3043	399	19	3	6	12
to-PART	2908	398	7	11	4	6
a-DET	2734	391	8	11	10	4
time-NOUN	2536	400	6	8	2	3

前回と同様、頻度に変化が生じているように見えるものもある。たとえば、表記形の be の頻度は 3,567 回だが、コピュラ（助動詞）の be-AUX は 3,320 回である。これは、ほかに、be-VERB（存在を意味する動詞としての be）が 247 回存在するためである。

4.2 I-JAS 作文データに基づく統合語彙頻度表

次に、日本語データからの統合語彙頻度表作成の一例として、日本語学習者コーパス I-JAS（迫田他、2020）に含まれる、中国語（大陸）母語学習者、英語母語話者、韓国語母語話者、合計300 名のストーリーライティング第 1 課題（SW1）作文のデータから作成した頻度表を概観する。

SW1 というのは、「夫婦がバスケットにサンドイッチを入れてピクニックに行ったが、知らぬ間に犬がバスケットの中に忍び込んでおり、目的地でバスケットを開けると、サンドイッチが食べられていた」といった経緯を描く 4 コマ漫画を見て、そのストーリーをパソコン上で作文するというものである。なお、I-JAS のデータは対面インタビューで取得されており、学習者は、インタビューの冒頭で、同じ漫画を見て内容を口頭で話すストーリーテリング課題を先に済ませている。

1 ファイルのサイズは 500～1100 バイト程度である。なお、I-JAS のテキストファイルには、1 行ごとに学習者コードが記載されているが、すべて英数字であり、学習者の書いた語とは区別可能であるため、今回は事前にコードを除去せずにそのまま解析にかけた。

今回のデータの処理時間は、日本語形態素解析環境の準備に 1 分程度、300 本のファイルのアップデートに 4 分 10 秒程度、日本語テキスト解析に 40 秒程度で、全体で 6 分程度であった。

さて、以上の処理を経て出力されるエクセルファイルには以下のシートが含まれる。

図 19

日本語統合語彙頻度表のファイル構造

まず、表層形の出力を概観する。

図 20

表層形頻度表

	単語頻度	ファイル頻度	CCH02－SW1.txt	CCH03－SW1.txt	CCH06－SW1.txt	CCH07－SW1.txt
単語頻度	34323		104	143	96	105
た	2156	298	9	8	6	6
K	1841	300	5	7	6	5
SW	1841	300	5	7	6	5
に	1443	296	3	8	4	11
を	1442	299	5	6	3	5
まし	1418	290	5	6	4	4
は	1311	289	4	4	6	5
て	1306	279	5	4	3	5

300 ファイルの総語数は 34,323 語である。英語の場合と同様、3 行目以降に個々の語が並ぶ

が、それらの行数を数えれば語種総数がわかる。今回の場合は 1,231 種である。

単語の側に注目すると、最も頻度が高いのは「た」で、頻度は 2,156 回、ファイル頻度は 300 種中 298 種、レンジ比率は 298/300≒99.3%となる。上位語の大部分は助詞であるが、その中で、「て」のレンジ数が相対的に低い。全体頻度がほぼ同等の「は」に比べ、「て」の習得はより難しく、一度も使えない学習者が 21 人(全体の 7%)存在することが確認できる。こうしたことが明らかになるのも個別ファイル処理の利点である。なお、今回は行コードを事前に除去していないため、I-JAS の元データで各行に付与されていた行コードの一部である K や SW が上位語含まれている。

続いて、個別ファイルの頻度に着目したい。CCH(中国語母語学習者)_02、03、06、07 の 4 つのファイルの総語数はそれぞれ 104 語、143 語、96 語、105 語となっている。ICNALE の作文とは異なり、I-JAS のストーリーライティングでは語数は自由だが、同じイラストを描写するためか、語数はかなり揃っている。英語の場合と同じく、日本語の場合も、4 ファイル間で頻度差が大きい語もある。最小・最大頻度に注目すると、「に」は 3 倍以上(3 vs 11)、「を」は 2 倍(3 vs 6)の開きがある。

続いて、品詞付きの出力結果を見ておこう。

図 21

表層形＋品詞頻度表

	単語頻度	ファイル頻度	CCH02－SW1.txt	CCH03－SW1.txt	CCH06－SW1.txt	CCH07－SW1.txt
単語頻度	34323		104	143	96	105
た－助動詞	2156	298	9	8	6	6
SW－名詞	1841	300	5	7	6	5
K－名詞	1838	300	5	7	6	5
を－助詞	1442	299	5	6	3	5
まし－助動詞	1418	290	5	6	4	4
に－助詞	1367	296	2	7	4	10
は－助詞	1311	289	4	4	6	5
て－助詞	1290	278	5	4	3	5
が－助詞	1004	275	1	6	2	1
バスケット－名詞	855	282	2	3	2	4

同じ語が複数の品詞に解析されることがあるため、語種の総数は、品詞なしの場合の 1,231 種から 1,301 種に増えている。同様に、一部の語の頻度も変わる。たとえば、表層形の「に」は 1443 回であったが、助詞の「に」の頻度は 1367 回である。これは、助動詞と判断された「に」が 76 回存在するためである。

次に、活用形を集約した後の語彙素頻度表と、語彙素＋品詞頻度表を併せて見ておこう。なお、UniDic に基づく語彙素出力では、2 つの点に注意が必要である。1 点目は、表層形との違いを示すため、一部の語彙素に特殊な漢字表記が充てられるということである(例:する→為る、いる→居

る、その→其の、しまう→仕舞う)。また、2 点目は、外来語には原語が(例:ピクニック-picnic)、多義語には用法識別コードが(例:そう-様態、そう-伝聞)、それぞれ、ハイフンの後に付記されるということである。

図 22

語彙素頻度表

	単語頻度	ファイル頻度	CCH02－SW1.txt	CCH03－SW1.txt	CCH06－SW1.txt	CCH07－SW1.txt
単語頻度	34323		104	143	96	105
た	2244	298	10	8	6	7
K	1841	300	5	7	6	5
SW	1841	300	5	7	6	5
ます	1687	298	5	6	6	4
を	1443	299	5	6	3	5
に	1367	296	2	7	4	10
て	1366	283	6	4	3	5
は	1311	289	4	4	6	5
が	1004	275	1	6	2	1
犬	891	294	2	3	3	3
の	862	272	2	4	3	4
バスケット-basket	855	282	2	3	2	4

図 23

語彙素＋品詞頻度表

	単語頻度	ファイル頻度	CCH02－SW1.txt	CCH03－SW1.txt	CCH06－SW1.txt	CCH07－SW1.txt
単語頻度	34323		104	143	96	105
た-助動詞	2244	298	10	8	6	7
SW-名詞	1841	300	5	7	6	5
K-名詞	1838	300	5	7	6	5
ます-助動詞	1687	298	5	6	6	4
を-助詞	1443	299	5	6	3	5
に-助詞	1367	296	2	7	4	10
て-助詞	1366	283	6	4	3	5
は-助詞	1311	289	4	4	6	5
が-助詞	1004	275	1	6	2	1
の-助詞	862	272	2	4	3	4
犬-名詞	858	293	2	3	3	2
バスケット-basket-名詞	855	282	2	3	2	4

表層形の語種数は 1,231 種(品詞付きだと 1,301 種)であったが、活用形の集約により、語彙素

の種別数は 874 種(品詞付きだと 893 種)となる。語彙素化により、一部の語の頻度は上昇する。たとえば、表層形の「まし」頻度は 1,418 回であったが、語彙素の「ます」頻度は 1,687 回になっている。一方、「バスケット」のように、活用形を持たない名詞は語彙素にしても頻度は変わらない。

5. 統合語彙頻度表の研究活用

以下では、統合語彙頻度表を加工する際に必要となる行列転換の方法について概説する。その後、EJWFTG から出力される統合語彙頻度表の研究上の活用例を示したい。

5.1 頻度表の加工

語彙頻度表を作る際には、ケース(行方向)にテキストファイルを、変数(列方向)に単語を置くことが多い。しかし、語彙研究では、通例、テキスト数が数個から数百個程度までであるのに対し、語種数は数千を超えることも少なくない。この場合、極端に横長の表となって扱いにくくなる。

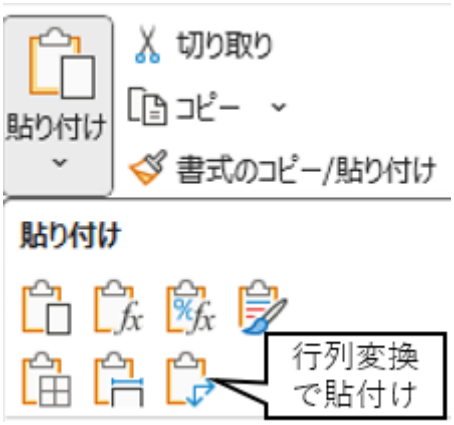
そこで、EJWFTG では、意図的に行列を転換した形で出力している。これにより、単語が行方向に並ぶこととなり、Excel のフィルタ機能を使って、特定の文字列を含む語や、特定品詞の語のみを抜き出すことが可能になる。また、多くの統計分析では、ケースの数が変数の数より多いことが前提になっているため、ケース側に単語が入ることで、適用できる統計手法の幅も広がる。

もっとも、単語ではなく、学習者コードでフィルタリングを行うような際には、行方向にテキスト(学習者)が来る形式のほうが便利であろう。こうした場合も、簡単な手続きで、行列を転換し、行方向にテキスト、列方向に単語がくる形式に変形することができる。以下では、ICNALE から作成された統合語彙頻度表を行列転換し、学習者コードを手掛かりとして、初級学習者(CEFR の A2 レベルの学習者)のみをフィルタリングして取り出す方法について概説する。

まず、頻度表の全体を Ctrl+A などでコピーし、新しいシートに「行列転換」して貼り付ける。

図 24

エクセル上での行列転換後の表の貼り付け



注:吹き出しは筆者が追記

「貼り付け」というボタンの下にある「∨」を押すと、貼り付けオプションが表示されるので、右下のアイコンを選ぶ。すると、行列転換された状態で貼り付けが完了する。その後、「ファイル頻度」とある3行目を全体を範囲指定した状態で「データ」→「フィルタ」を押し、フィルタボタン（下向き▼のボタン）を設定する。

図 25

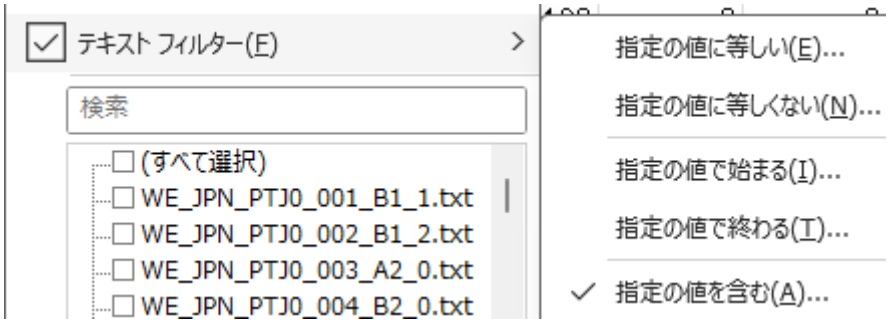
行列転換後の頻度表にフィルタを設定した画面

	単語頻度	to	i	a	time	is
単語頻度	91076	3361	2866	2734	2505	2125
ファイル頻度		3	3	3	4	3
WE_JPN_PTJ0_001_B1_1.txt	214	7	18	8	6	4
WE_JPN_PTJ0_002_B1_2.txt	190	11	3	11	8	7
WE_JPN_PTJ0_003_A2_0.txt	213	7	6	10	2	2
WE_JPN_PTJ0_004_B2_0.txt	212	8	11	4	3	3
WE_JPN_PTJ0_005_B2_0.txt	216	14	1	10	5	10
WE_JPN_PTJ0_006_B1_1.txt	234	5	2	8	12	6
WE_JPN_PTJ0_007_A2_0.txt	229	11	11	7	9	5
WE_JPN_PTJ0_008_B1_1.txt	307	16	3	4	5	4
WE_JPN_PTJ0_009_B2_0.txt	229	9	6	11	7	9
WE_JPN_PTJ0_010_B1_2.txt	240	11	6	5	6	9

ファイル頻度のセルにあるフィルタボタンを押し、テキストフィルタを選ぶと、右側に様々な検索条件が出てくるので、「指定の値を含む」を選ぶ。

図 26

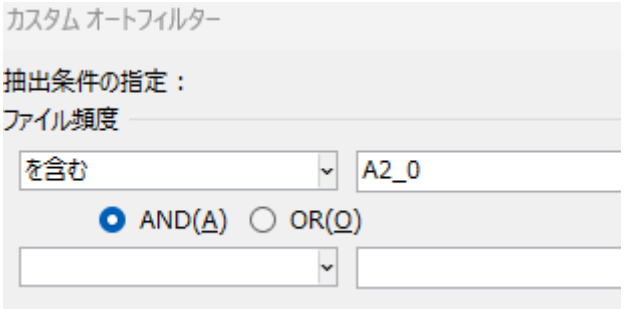
フィルタ機能で「指定の値を含む」を指定



ICNALE の話者コードでは、.txt の前の 4 文字コードが CEFR のレベルを示している（A2_0、B1_1、B1_2、B2_0）。よって、「指定の値を含む」条件に A2_0 と記入する。

図 27

ファイルコードの一部を検索キーとして指定



これにより、全 400 名の学習者中、A2_0 の学習者のデータだけがまとめて出てくる。

図 28

初級（A2）学習者のみを抽出した例

1		単語頻度	to	i	a	time	is
2	単語頻度	91076	3361	2866	2734	2505	2125
3	ファイル頻度		3	3	3	4	3
6	WE_JPN_PTJ0_003_A2_0.txt	213	7	6	10	2	2
10	WE_JPN_PTJ0_007_A2_0.txt	229	11	11	7	9	5
16	WE_JPN_PTJ0_013_A2_0.txt	223	12	2	10	10	6
30	WE_JPN_PTJ0_027_A2_0.txt	213	5	3	5	8	4
31	WE_JPN_PTJ0_028_A2_0.txt	214	14	3	7	4	8
36	WE_JPN_PTJ0_033_A2_0.txt	237	7	10	11	5	5
39	WE_JPN_PTJ0_036_A2_0.txt	210	5	4	10	7	6

同様のやり方で、A2_0 のみ、B1_1 のみ、B2_0 のみ、などの条件でデータを取り出せば、習熟度別の比較などが行いやすくなる。

I-JAS についても、学習者コードを使って、母語別にデータを取り出すことができる。ただし、I-JAS のコードには習熟度情報が入っていないため、それらはフェイスシートで調べて別途手作業で区分する必要がある。

5.2 有意性検証の精緻化

さて、森だけを見るマージデータに代えて、個々の木、つまり、個体データの調査を行うことは、統計的検証の質の向上にもつながる。一例として、ICNALE を用いて、初級学習者（CEFR の A2 レベル、全 154 名）と上級学習者（B2+レベル、全 18 名）の間で、助動詞 can の頻度に差があるかを調べる場合を考えてみよう。

習熟度別のマージデータで計量すると、A2 は総語数が 34,980 語で can 頻度は 432 回、B2+は総語数が 4,285 回で can 頻度は 61 回である。1,000 語あたりで標準化すれば、A2 頻度は

12.35、B2+頻度は 14.24 となり、B2+のほうが can を多く使っているように見える。

この差が誤差を超えるレベルに達しているかどうかを調べるには、いわゆる有意性検定を行うわけだが、マージデータの場合、使える数値は頻度と総語数のみで、適用可能な検定手法はカイ二乗統計量(またはその代替統計量)を用いた比率検定ということになる。具体的な手順としては、当該語頻度と、「それ以外の語の総頻度」を求め、以下のような 2×2 の分割表を用意する。なお、「それ以外の語の総頻度」とは、総語数から can 頻度を引いた残りである。

表 2

マージファイルから得られる頻度表の例

比較群	can	それ以外
A2 学習者群	432	34,548
B2+学習者群	61	4,285

その後、2 群間で差がない場合に予想される頻度を記した予測値表を作る。そして、セルごとに、予測値と実測値の差分を求め、それらを加工して合算することでカイ二乗統計量が求まる。こうして得られた統計量が基準値を超えているかどうかを見ることで差の有意性が判定できる。上記のデータであれば、$X^2(1)=.76$; $p=.384$; $\phi=.0048$ で、差は有意ではなかったことになる。

一方、統合語彙頻度表があれば、マージ頻度ではなく、個々人ごとの can 使用頻度の一覧表が簡単に入手できる。

表 3

個体データから得られる頻度表の例

学習者	A2(154 名)	B2+(18 名)
Sub_001	2	3
Sun_002	7	3
Sun_003	2	5
Sun_004	6	5
…	….	…
平均	3	2
SD	2.32	1.73

注:上表の Sub_001、002 などは、各群におけるコードの 1 番目、2 番目などを表すもので、対応ありデータではない。

こうしたデータがあれば、ANOVA や平均値の差の検定を適用し、分散(つまりは個体差による誤差)を加味した検討が可能となる。上記に対応なし t 検定(Welch)をかけると、$t(24.92)=1.28$; $p=.212$ となる。

今回は、マージデータに基づくカイ二乗検定でも、個体データに基づく t 検定でも、差は有意でないという結論になった。ただ、それぞれから得られた p 値は異なっており、使用するデータによっては、差がある／ない、という結論そのものが変わってくることもありうるだろう。EJWFTG を用いた統合語彙頻度表があれば、使える検定手法のオプションが増え、差の有意性をより慎重に検討することが可能になる。

5.3 多変量解析への応用

コーパス研究では、頻度表に多変量解析を適用し、単語やテキストの分類を試みることが多い。このとき、単語については大量の情報を得やすいが、マージデータを使う場合、テキストの種別数は少なくなる。これに対し、個々人のデータが使えれば、テキストについても十分なサンプル数が得られ、結果として、解析の精度を高めることができる。本節では、多変量解析のケーススタディとして、I-JAS から得られた統合語彙頻度表を用いた対応分析を試みる。

対応分析では、頻度表の列方向に置かれた第 1 アイテムと、行方向に置かれた第 2 アイテムの相関を最大化する次元が取り出される。そして、通例、寄与率の最も大きい上位 2 つの次元を横軸・縦軸として散布図を作成する。これにより、アイテムカテゴリデータ（コーパス研究で言えば、個々のテキストと個々の単語）が同一平面上に布置され、相互の関係を質的に検討することが可能になる。

EJWFTG から得られる統合語彙頻度表を使った対応分析には、(1)群化をボトムアップで行える、(2)品詞情報を加味した分析が行える、といったメリットがある。(1)に関して、たとえば、上級者と初級者の間に差があるかどうか調べるには、ふつう、上級者マージデータと初級者マージデータを作って比較する。だが、この場合、両者に差があるか否かを検証しようとしているのに（つまり、差の存在はまだ証明されていないのに）、その差の存在を前提としてあらかじめトップダウン的にデータを群化していることになり、データの扱いとして違和感も感じられる。これに対し、群情報をラベル化した個体データ（たとえば、上級者群に属する Upper_1, 2, 3…、初級者群に属する Novice_1, 2, 3…など）を用意し、多変量解析にとって、それらが本当に上級群と初級群に分かれるかをボトムアップで観察するようにすれば、群化の妥当性はより高まることになるだろう。こうした「ラベル付き個体データ分類」は、所属する森の情報を保持したまま、個々の木を探索的に分類するもので、学習者コーパス研究はもちろん、性差研究を含め（石川, in press）、幅広い変種研究に応用可能なものである。

次に、(2)に関して、一般には、上位 10 語、上位 100 語など、頻度上位語を機械的に抽出して分類に使うわけだが、こうした手法では、雑多な性質の語を混ぜて分類していることになり、得られた結果の解釈は困難になる。これに対し、名詞上位 50 語、動詞上位 100 語など、文中での性質や機能が似通った同一品詞に限って上位語を分析することで、手法上の透明性を高めることができる。

以上の観点をふまえ、本節では、表層形＋品詞頻度表から、助詞頻度上位 20 語を第 1 アイテム（下表参照）、中国語（個体コードは C**＋連番）・英語（E**＋連番）・韓国語（K**＋連番）を母

語とする日本語学習者各 100 名、合計 300 名を第 2 アイテムとする頻度表を用意して対応分析を試みる。

表 4
分析対象とする助詞頻度上位 20 語

分析対象語
を／に／は／て／が／と／の／で／から／も／ながら／か／へ／や／ん／ね／だけ／など／な／まで

対応分析の結果、第 1 次元と第 2 次元の寄与率は 12.33%と 10.17%となった。第 1 アイテムのカテゴリ数が 20 と多いため、見かけ上の寄与率は低くなっているが、2 つの次元で全体の分散の 2 割以上が説明されている。下記は全体散布図と、中央部（原点を中心とする±2.0 範囲）を拡大した散布図である。

図 29
I-JAS の SW1 課題における高頻度助詞 20 語と学習者 300 名の関係性（全体）

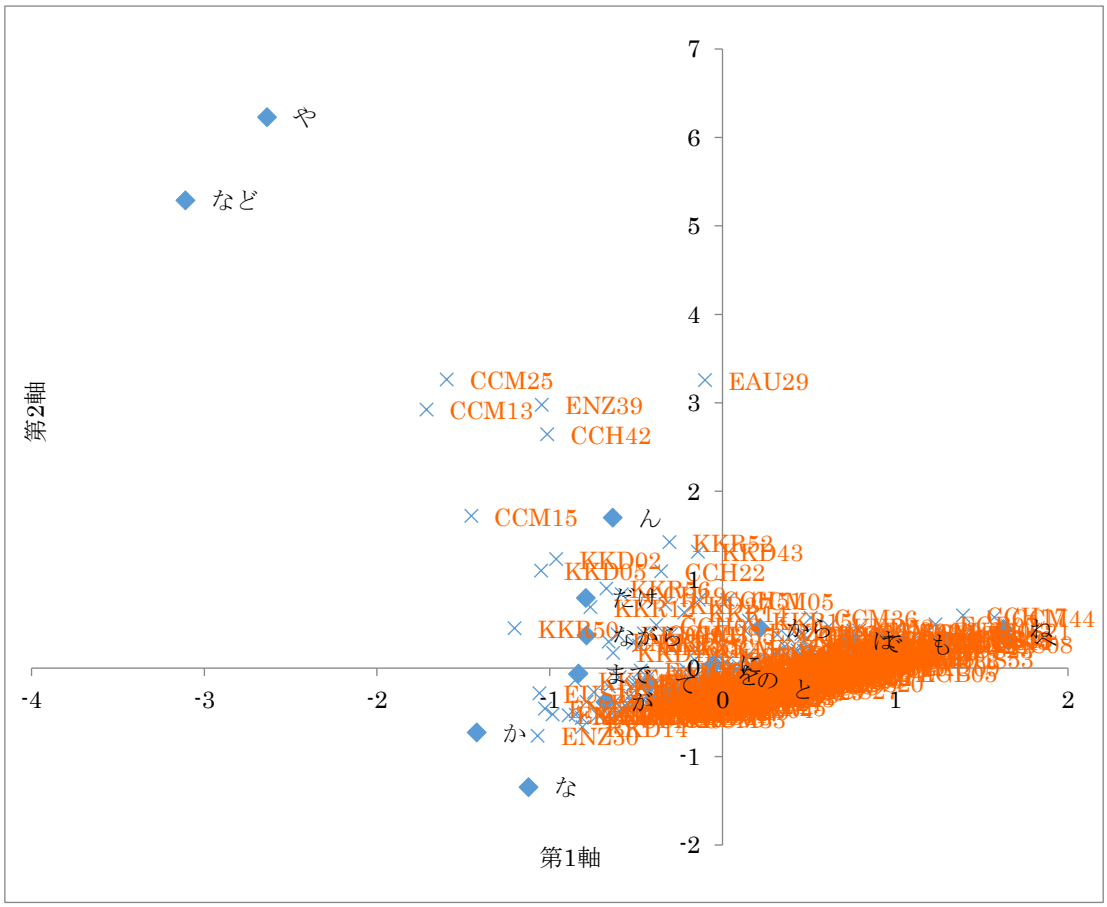
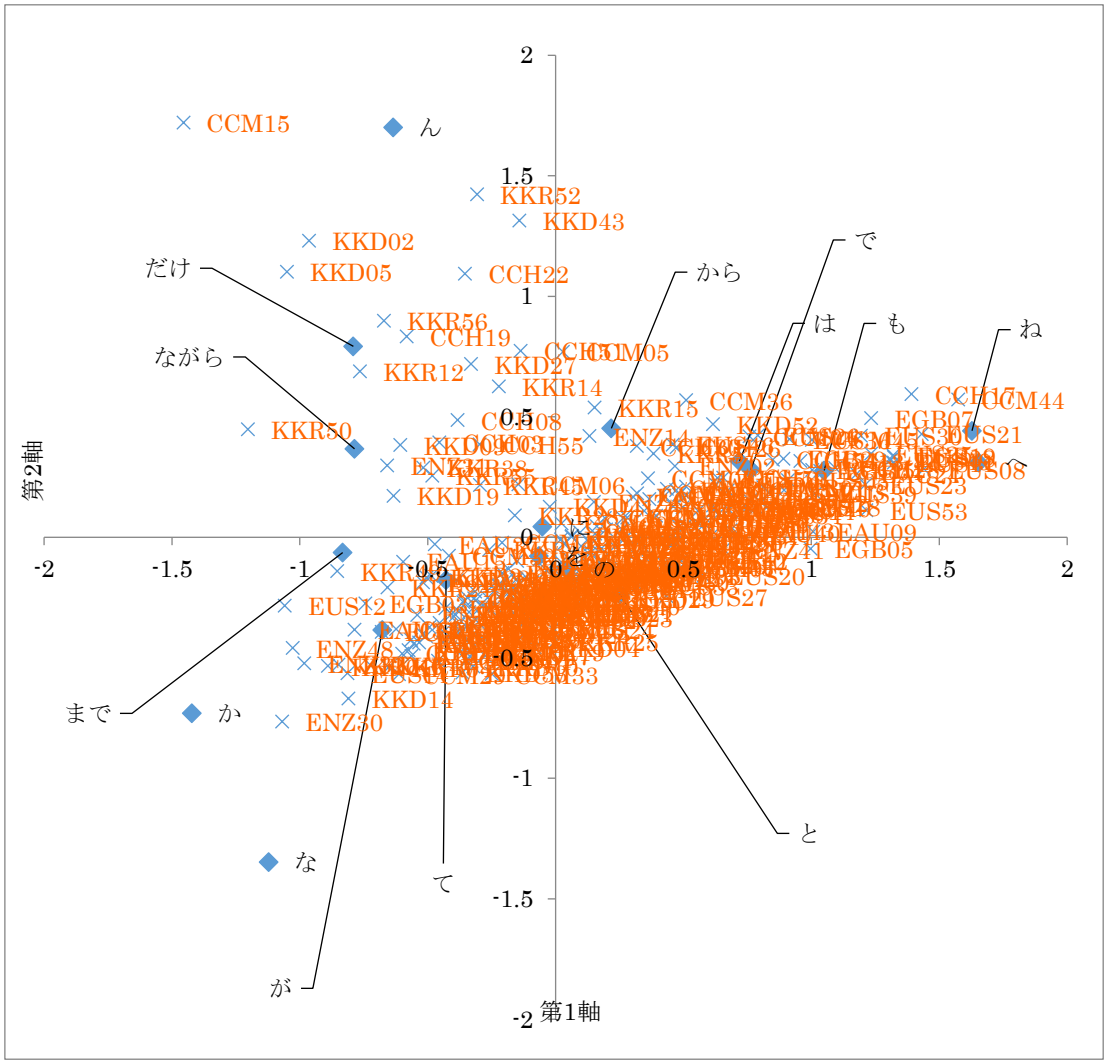


図 30

I-JAS の SW1 課題における高頻度助詞 20 語と学習者 300 名の関係性（中心部拡大）



まず、 図 29 より、20 品詞の中で、事項を並列する「や」と、事項を例示する「など」の 2 つが原点から顕著に離れた位置に布置されており、複数の情報項目の導入に関わるこれら 2 語が他の助詞とは使われ方が大きく異なっていることがわかる。

次に、図 30 に注目すると、4 つの象限ごとに異なる助詞と学習者が配置されていることがわかる。全 300 人と 20 種の助詞を 4 つの象限に整理すると、以下のようになる。なお、散布図は縦横の 2 軸で 4 つの領域に区分されるが、右上を第 1 象限と呼び、以下、反時計回りに、第 2、第 3、第 4 象限と呼ぶ。なお、4 象限に区分される助詞のうち、原点近傍部（横軸・縦軸の座標がともに±0.2 より小さい）に入るものは、象限特徴を強く持っていないと思われるため、下表ではカッコ書きで示している。

表 5

象限別の助詞・学習者の分布

象限	助詞タイプ・関連助詞	学習者母語背景			合計
		中国語	韓国語	英語	
1	A:から／で／ね／は／へ／も	**47**	6	**40**	93
2	B:だけ／ながら／など／や／ん／(に)	11	16	3	30
3	C:か／が／て／な／まで／(を)	23	**65**	29	117
4	D:と／(の)	19	13	28	60
合計	---	100	100	100	300

注:太字の数字は、話者群ごとに、象限別で最も人数が多いものを示す。

詳細な分析は本稿の狙いを超えるが、全体の概観として、SW1 課題内で学習者が使用する助詞は、A:内容を取り立て(「は」)、始点(「から」)・終点(「へ」)・手段(「で」)など、行為の背景を説明するもの、B:複数の内容を導入して(「ながら」「など」「や」))一部を強調する(「だけ」)もの、C:主格を導入し(「が」)、行為の連鎖(「て」)や終点(「まで」)を示すもの、D:項目を並列する(「と」)もの、の 4 種に大別されること、また、中国語と英語を母語とする学習者は A 型を、韓国語を母語とする学習者は C 型を多用する者が多いこと、複数の情報を導入して内容を複雑化させる B 型を多用する学習者は母語を問わず少ないことなどが示された。また、あわせて特筆すべきは、3 群の学習者はいずれかの助詞タイプに属するわけではなく、A～D 型のすべてに広くばらけていることである。これらは、群としての傾向があるとしても、同時に、個体差の要因が非常に大きく、「森を見ながら木を見る」アプローチが必要であることを示す結果と言える。

6. まとめ

以上、本稿では、コーパス研究における「森を見ながら木を見る」視点の重要性を指摘し、こうした研究を支える目的で開発された EJWFTG の概要を示した。また、EJWFTG で作成される統合語彙頻度表の研究上の意義として、有意性検証の信頼性の向上と、多変量解析への応用の可能性について触れた。

もちろん、EJWFTG で実装している処理は、ある程度のプログラミングの知識があれば、自力でできるものである。しかし、コーパス研究者のすべてがプログラミングの知識を持っているわけではない。この意味で、プログラミングを使わなければ膨大な時間がかかる頻度調査の地道な基礎作業を省力化する EJWFTG の公開は、学習者コーパス研究を含め、幅広いコーパス研究にとって、一定の意義を持つものと思われる。こうしたツールの活用によって、「森」だけで済ませず、「森を見ながら木を見る」丁寧なコーパス研究が広がっていくことを期待したい。

謝辞

EJWFTG は、科学研究費プロジェクト「言語から見た日米マインドスケープ比較：データサイエンス志向型小説研究の試行」（20K20699）による研究成果の一部である。EJWFTG の開発にあたっては、基本構想・基本設計・インタフェース設計・出力結果検証などを本稿筆者が担当し、それに基づくプログラミング作業は山本和英氏が担当した。EJWFTG が、当初想定していたスタンドアロン型のソフトウェアではなく、Google Colaboratory 上のシステムとして実装されることになったのは山本氏の発案による。また、筆者がこのようなプログラムの開発を着想したきっかけは、今尾康裕氏の CasualConc に触れた経験であった。すぐれた言語学者であると同時にすぐれたプログラマーでもある山本氏と今尾氏に感謝申し上げる。

参考文献

Ishikawa, S. (2023). *The ICNALE guide: An introduction to a learner corpus study on Asian learners' L2 English*. Routledge.

石川慎一郎 (in press).「コーパス基盤型性差研究の方法論―男女大学生の会話に見る性差をめぐって―」 森山由紀子・加藤大鶴(編)『「女ことば」「男ことば」を越えて―日本語のジェンダー研究の新たな地平―』ひつじ書房.

迫田久美子・石川慎一郎・李在鎬(編). (2020).『日本語学習者コーパス I-JAS 入門：研究・教育にどう使うか』くろしお出版.