「ソフトウェア紹介」

音声・映像コーパス構築ツール「Speech Indexer」の紹介

後藤　一章

1. はじめに

一般に，通常の文書コーパスと比較し，音声・映像コーパスの構築には多大なコストがかかり，未だ広く実践されているとは言い難い。ICNALE-Spoken（Ishikawa, 2023）や TCSE（Hasebe, 2015）等によってその有用性や可能性は認識されつつあるものの，音声の文字起こし作業やその後の検索方法が課題となり，個々の研究者が音声・映像コーパスを自ら構築することは容易ではなかった。

本研究は，こうした問題の解消を目指し，音声・映像コーパスの構築を支援するツール「Speech Indexer」を開発した。Speech Indexer は OpenAI の「Whisper」と呼ばれる音声認識システムを利用し，音声の自動文字起こしと検索文字列の該当箇所再生機能を有する Windows プログラムである。

本稿では，まず Whisper の概要と，その派生プログラムである「Whisper.cpp」について紹介し，続けて Speech Indexer の機能と操作方法について簡潔に述べる。さらに，Speech Indexer を使用し，英語学習者が話す英語の音声認識精度について小規模な調査を行ったため，合わせて報告する。

2. 音声認識システム

2.1 Whisper

Whisper は，OpenAI が 2022 年 9 月，MIT ライセンスのオープンソース・ソフトウェア（ソースコードを自由に利用，改変，再配布等が可能なライセンス形式）として公開した音声認識システムである。公開されて間もなくその認識精度の高さが注目を集め，オープンソースということもあり，国際的に広く使用されることとなった。また，2023 年 3 月には，有料ではあるが WebAPI と

しても公開され，同社の文章生成 AI である ChatGPT との組み合わせが容易になるなど，より柔軟にアプリケーションや Web サービスに Whisper を統合することが可能となった。

OpenAI の公式サイトによると，Whisper は Web から収集した 68 万時間分の多言語音声データを，Transformer と呼ばれる深層学習モデルによって学習している。詳細は Radford et al.（2022）に譲るが，入力された音声データはログメルスペクトログラム（log-Mel spectrogram）という特徴量に変換され，対応する文字列データ等と共に学習が行われる。認識精度は，英語で約 95.5%，日本語で約 93.6% とされている（Radford et al., 2022）。

2.2 Whisper.cpp

Whisper.cpp は，オープンソース版の Whisper を Georgi Gerganov 氏が C 及び，C++ というプログラミング言語によって新たに書き起こしたオープンソース・ソフトウェアである。オリジナルの Whisper は Python で作成されており，GPU（Graphics Processing Unit）の使用を前提としているが，Whisper.cpp では必ずしも GPU は必須ではない。プログラム全体が C/C++ で書かれていることもあり，CPU（Central Processing Unit）のみでも高速な文字起こしを実現している。ソースコードが公開されているため，コンパイル環境さえあれば OS を問わず実行が可能であり，また，FFmpeg 等の外部プログラムに依存しない点も利点として挙げられる。

2.3 Speech Indexer 開発の背景

Whisper や Whisper.cpp は極めて有用なソフトウェアであるが，その利用にはコマンドラインでの操作が必須となる。コマンドライン操作は柔軟で自由度が高いが，場合によっては煩雑となる。また，ファイルの一括処理や，文字起こしテキストからの音声・映像検索等にはある程度のプログラミング知識が求められ，必ずしもコーパス言語学や外国語教育研究で手軽に活用できるとは言い難い。そこで，本研究では，基本的な認識処理はもちろん，複数ファイルの処理や，検索処理を直観的に行える GUI（Graphical User Interface）を備えたユーザフレンドリーなツールを開発することとした。

3. Speech Indexer

3.1 文字起こし

図 1 は，Speech Indexer の文字起こし画面である。操作方法は，任意の音声または映像ファイルを選択し，モデルファイル等を設定したうえで，文字起こしを実行するのみである。

音声ファイルは WAV，MP3，M4A フォーマットに対応し，映像ファイルは MP4, MOV フォーマットに対応している。Whisper.cpp は 16 ビットの WAV ファイルにのみ対応しているが，本ツールではその他の形式でも内部処理によって適切なフォーマットに変換している。

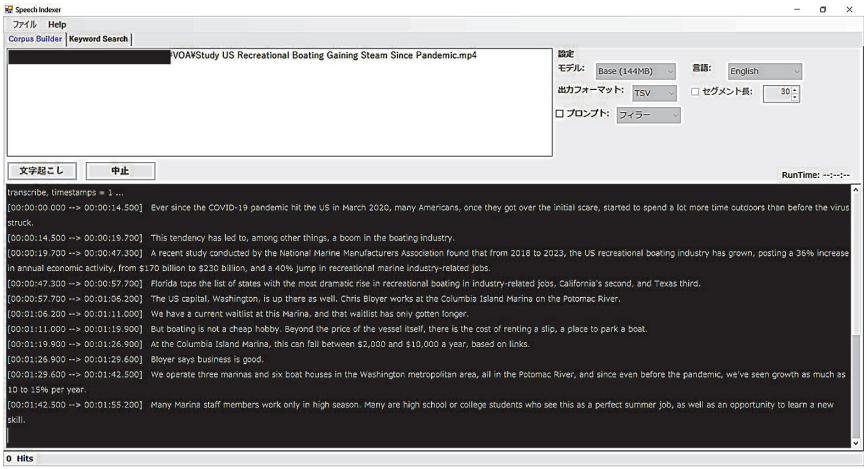

図 1. Speech Indexer の文字起こし画面

3.1.1 モデル

使用するモデルは，文字起こしの処理速度と認識精度に直接的に影響する。モデルファイルのサイズが大きければそれだけ精度は向上するが，必然的に処理時間も増加する。モデルファイルは，TINY，BASE，SMALL，MEDIUM，LARGE の 5 種類に大別される。

モデルファイルのファイルサイズと，処理に要する時間の例を表 1 に示す。データには，約 11 秒と約 1 分 41 秒の英語音声ファイルを使用した。計測は，Intel Core i5-10210U，コア数 / スレッド数：4/8，動作周波数 : 1.60GHz，RAM:

16.0 GB，の環境で行った。なお，GPU は不使用である。最小の TINY モデルでは，オリジナル音声の十分の一程度の処理時間となったが，最大の Large モデルでは，オリジナル音声の時間の 3〜6 倍程度必要となった。

表 1. モデル別による文字起こしに要する処理時間

	File Name	File size & Length	Processing Time
TINY (77.7MB)	Test1.wav	350KB (11 秒)	2 秒
	Test2.mp4	2.3MB (1分 41秒)	10 秒
BASE (148MB)	Test1.wav	350KB (11 秒)	3 秒
	Test2.mp4	2.3MB (1分 41秒)	15 秒
SMALL (488MB)	Test1.wav	350KB (11 秒)	10 秒
	Test2.mp4	2.3MB (1分 41秒)	54 秒
MEDIUM (1.53GB)	Test1.wav	350KB (11 秒)	42 秒
	Test2.mp4	2.3MB (1分 41秒)	3 分 17 秒
LARGE (3.09GB)	Test1.wav	350KB (11 秒)	1 分 12 秒
	Test2.mp4	2.3MB (1分 41秒)	6 分 10 秒

Speech Indexer には予め BASE モデルを同梱しており，ダウンロード後即座に使用できる状態となっている。ただし，高精度での認識には MEDIUM や LARGE モデルの導入が推奨され，特に日本語認識には，最低でも SMALL 以上のモデルが望ましい。

3.1.2 言語

認識言語を「English」,「Japanese」,「Auto」から選択する。「Auto」は認識速度がやや低下するが, 言語ごとに設定を切り替える作業が不要となる。また, Whisper は実際には 98 言語の音声認識に対応しており，英語と日本語以外の文字起こしも可能である。ただし，Speech Indexer の音声・映像検索機能は，英語と日本語以外には原則として対応していない。

3.1.3 出力フォーマット

文字起こしされたファイルには，セグメント（行）の開始時間，終了時間，文字起こしテキストが含まれる。出力フォーマットは，以下のようなタブ区切りの TSV 形式か，カンマ区切りの CSV 形式から選択する。タイムスタンプが不要な場合は, テキストのみの TXT 形式での出力も可能である。Whisper.cpp は, VTT 形式，SRT 形式，JSON 形式にも対応しているが，現時点では本ツールでは対応していない。なお, 出力ファイルは, 原則として入力ファイルと同じフォ

ルダ内に生成される。

start	end	text
0	7600	“And so my fellow Americans ask not what your country can do for you,”
7600	10600	“ ask what you can do for your country.”

3.1.4 セグメント長

「セグメント長」は，1 セグメントに含まれる語数を意味する。設定は任意だが，指定しない場合は 1 セグメントが大幅に長くなる場合もあるため（図 1 は未指定），必要に応じて設定することが望ましい。特に，本ツールにおける検索文字列の計測は，1 セグメントに当該文字列が複数回生起している場合でも，1 度しかカウントされていない。セグメントを短くすることで，実態に近い値が得られることになる。

3.1.5 プロンプト

Whisper の基本的な文字起こしでは，“uh” や “um” などのフィラーや，“I have ... I have to” などの言い淀みは省略され，明らかな文法ミスもある程度修正されて文字起こしされることがある。本仕様は, 発話の内容把握には合理的だが, 言い淀みや言い誤りの調査には不都合である。こうした際，プロンプト機能が有効となる場合がある。

プロンプトに特定の語句を指定すると，それらの語句は原則として省略されずに文字起こしされる。プロンプト機能を使用した場合と，使用していない場合の結果を以下に示す。

［プロンプト機能を使用］　Um, they, they have to, they have to work after graduation.
［プロンプト機能を不使用］　They have to work after graduation.
（ICNALE Spoken: SM_JPN_PTJ1_024_A2_0）

なお，デフォルトでは以下の語句をプロンプトとして設定しているが，「Prompt」フォルダ内にある “Filler_en.txt” を修正することで，指定する語句の設定が可能である。

［Umm, umm, Hmph, hmph, Um, um, Ah, ah, Uh, uh, Er, er, Oh, oh］

また，OpenAI のウェブサイトによると，プロンプト機能は人名や専門用語の認識精度の向上にも利用できるとされている。Speech Indexer ではその場合，プロンプト機能を有効にする際に，設定を「フィラー」ではなく「その他」を選択し，Others.txt ファイルに任意のプロンプトを記載する。ただし，4 節で示すように，プロンプト機能を有効にすると予期しない結果が得られることも多く，現時点ではやや実験的な機能と言える。

3.2 文字列検索

図 2 は，文字起こしされたファイル内の語句検索を行う画面である。通常のキーワード検索と同じ要領で，検索文字列をテキストボックスに入力して検索する。検索文字列を含むセグメントがあれば，画面下の領域に「開始時間」「終了時間」「テキスト」が表示される。任意のセグメントをダブルクリックすることで，該当部分の音声または映像が別ウィンドウで再生される。

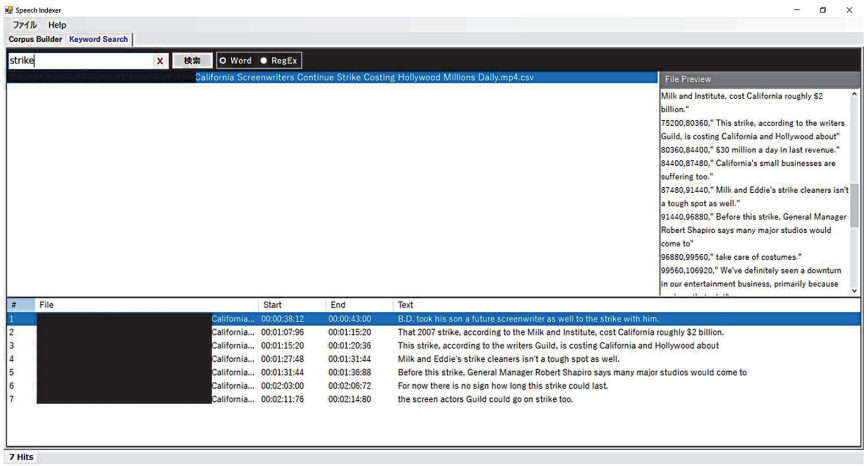

図 2. Speech Indexer の文字列検索画面

3.3 検索文字列該当箇所の再生

図 3 は，検索文字列がどのように発話されているかを確認する画面である。

画面上部に，音声ファイルの場合は波形が表示され，映像ファイルの場合は映像が音声と共に再生される。画面下部にはセグメント単位で文字起こしされたテキストが表示され，現在発話されているセグメントはハイライトされる。音声または映像の再生に合わせて，下部のテキスト表示画面も自動でスクロールする。特定のセグメントのリピート再生も可能である。

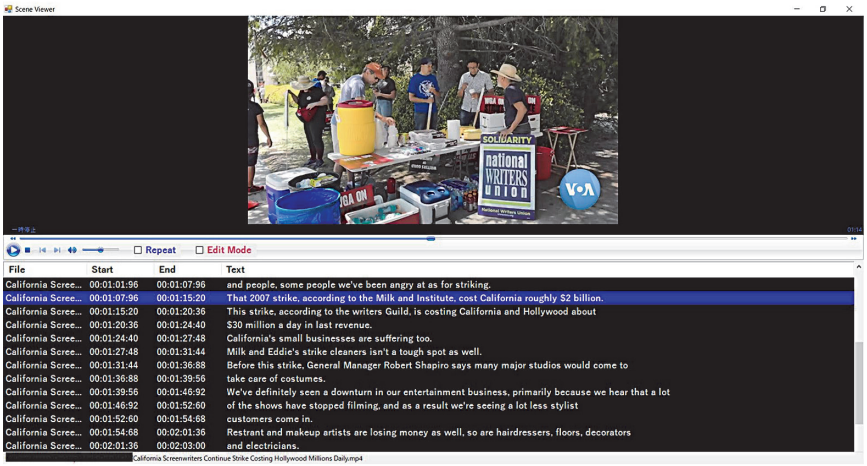

図 3. Speech Indexer の音声・映像確認画面

4. 学習者の英語音声認識精度

2.1 節で述べた通り，Whisper の英語の認識精度は約 95.5% である（Radford et al., 2022）。しかし，Radford らが評価に使用した「FLEURS Dataset」は基本的に英語母語話者の音声であり，英語学習者による英語の認識精度については十分に明らかにされていない。そこで本節では，ICNALE Spoken を使用し，日本人英語学習者の英語認識精度を調査した結果について報告する。

ICNALE（International Corpus Network of Asian Learners of English）は神戸大学の石川慎一郎氏が作成した英語学習者コーパスであり，日本をはじめ，アジア圏の国々における学習者の英文が収録されている。ICNALE Spoken はその中でも特に，学習者の発話を録音した音声ファイル，発話の様子を録画した映像ファイル，そしてそれらを手作業で文字起こししたテキストファイルを含む

音声・映像コーパスとなっている。

ICNLE Spoken には話者の CEFR レベルが付与されており，今回は A2，B1，B2 レベルから任意に 2 ファイルずつを選択し，調査に使用した。認識精度の評価については，音声認識評価において一般的に用いられる「単語誤り率：WER（Word Error Rate）」を利用した。Urban & Mehrotra（2023）によると，単語誤り率は以下のように計算できる。この時，I（Insertion）は「誤って追加された単語」，D（Deletion）は「検出されなかった単語」，S（Substitution）は「置き換えられた単語」，N は手作業で文字起こしされた正解単語数を意味する。

$$WER = \frac{I + D + S}{N} \times 100$$

I，D，S の具体例を図 4 に示す。上側の Human-labeled Transcript が手作業による文字起こし結果，下側の Speech Recognition Result が音声認識システムによって文字起こしされた結果である。

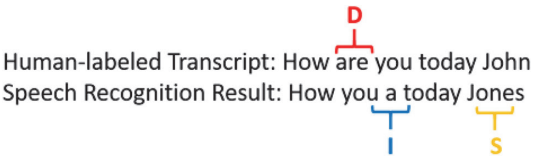

図 4. 音声認識の誤り例（Urban & Mehrotra（2023）より）

表 2 は，使用したファイル名，話者の CEFR レベル，正解ファイルにおける単語数，及び WER を示したものである。WER は，プロンプト機能を使用した場合と，使用していない場合の両方を示している。モデルはいずれも LARGE を使用した。また，WER の計算には，「JiWER」と呼ばれる Python プ

表 2. 英語学習者による音声の認識結果

#	ファイル名	CEFRレベル	語数	WER（プロンプト無効）	WER（プロンプト有効）
1	SM_JPN_PTJ1_024_A2_0	A2	65	31.74%	80.95%
2	SM_JPN_PTJ1_050_A2_0	A2	37	123.53%	70.59%
3	SM_JPN_PTJ1_023_B1_1	B1	54	25.00%	28.84%
4	SM_JPN_PTJ1_046_B1_1	B1	49	40.00%	40.00%
5	SM_JPN_PTJ1_003_B2_0	B2	111	21.57%	18.62%
6	SM_JPN_PTJ1_005_B2_0	B2	75	5.48%	6.85%

ログラムを使用した。

一般に，WER は 10〜20% 以下であることが望ましいとされている。B1 レベルと B2 レベルの認識精度については，「〜_046_B1_1」がやや精度が低いものの，学習者の音声であることを考慮すると，概ね良好な結果であると思われる。特に「〜_005_B2_0」において誤りとなった個所はカンマを含めた下線部のみであり，ほぼ正確に文字起こしされている（実際の正解データは“. Therefore, this experience”）。ただし，予想にやや反し，いずれもプロンプトの影響は特には見られなかった。

I agree with this opinion. There are three reasons. First, young people can learn something important about the relationship between elderly people and them. Second, young people can learn the importance of money through part-time job. Third, young people can use polite words, especially to elderly people. It is said that these days young people cannot know how to use polite words, therefore these experiences make them feel the importance of part-time job.

（ICNALE Spoken: SM_JPN_PTJ1_005_B2_0 ※プロンプト無効）

一方，A2 レベルでは WER が高いことに加え，プロンプトの有無による差も顕著となった。「〜_024_A2_0」に関しては，プロンプトを有効にすることで，特定の音声パタンに過度に反応する挙動が見られた。具体的には，以下のように “um” が何度も繰り返されており，この音声ファイルの話者は確かにフィラーが散見されたが，ここまで極端なものではなかった。

Um, um, um, um, um, um, um, um, um, um, um, um, um, um, um, um, um, um, um, um, um, um, um, Um, they, um, and, and, they, they want to, they need, they need to money.

（ICNALE Spoken: SM_JPN_PTJ1_024_A2_0 ※プロンプト有効）

また，「〜_050_A2_0」については，話者のレベルよりも音声の録音環境の影響が大きかったと考えられる。当該ファイルは今回使用した音声の中で最も録音環境が悪く，メイン話者の発話が終わった後も背景で以下のような雑音が収録されてしまっていた。そのため，本来の発話の後に不要なテキストが追加されてしまっており，プロンプトの有無にかかわらず，WER が著しく高くなっ

ている。

Okay, this slide is your name, student number and part-time job number 1, E.T.J.1. Alright. Stop the button and then

（ICNALE Spoken: SM_JPN_PTJ1_050_A2_0 ※プロンプト無効）

以上，極めて小規模ではあるが，英語学習者の文字起こしにおける課題がいくつか浮き彫りとなった。一方，課題はあるものの，録音環境が整えられ，一定水準の流暢さで話す学習者であれば，決して低くない精度で文字起こしが行える可能性も示された。今後，英語母語話者の音声ではなく，ICNALE Spoken のような英語学習者の音声で訓練された音声認識システムが開発されれば，さらに精度の向上が見込まれる。より正確な音声認識が可能になれば，スピーキング学習や習熟度評価等への様々な利用も可能となり，今後のさらなる研究の発展が期待される。

5. まとめ

本稿では，音声認識システム「Whisper」・「Whisper.cpp」を利用した音声・映像コーパス構築ツール「Speech Indexer」について紹介した。Whisper は有用なプログラムであるが，プログラミングやコマンドライン操作に馴染みの薄い研究者にとっては使用するうえで少なからず技術的なハードルが存在すると考えられるため，本ツールの開発に至った。また，小規模ではあるが，英語学習者による英語の認識精度についても調査を行い，英語学習者による発話の文字起こしの可能性についても言及した。

音声認識システムは, 従来はコストの問題で困難となっていた音声･映像コーパスの開発を手軽に実践できる点において，コーパス言語学と極めて相性のいい技術であると考える。Speech Indexer が今後の音声・映像コーパス研究のさらなる進展に資することを願う。

注

Speech Indexer は筆者のウェブサイトにて公開中である。
https://www.setsunan.ac.jp/~corpus/SpeechIndexer.htm

引用文献

Hasebe, Y. (2015). Design and implementation of an online corpus of presentation transcripts of TED Talks. *Procedia: Social and Behavioral Sciences, 198*(24), 174–182.

Ishikawa, S. (2023). *The ICNALE Guide: An Introduction to a Learner Corpus Study on Asian Learners' L2 English*. Routledge.

Radford, A., Kim, J.W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2022). Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*. https://arxiv.org/abs/2212.04356

Urban, E. & Mehrotra, N. (2023). Test accuracy of a custom speech model. *Microsoft Learn*. Retrieved December 11, 2023, from https://learn.microsoft.com/ja-jp/azure/ai-services/speech-service/how-to-custom-speech-evaluate-data?pivots=speech-studio

参考ウェブサイト

FLEURS Dataset（https://paperswithcode.com/dataset/fleurs）（2023 年 8 月）

OpenAI『Introducing Whisper』（https://openai.com/research/whisper）（2023 年 8 月）

Whisper（https://github.com/openai/whisper）（2023 年 8 月）

Whisper.cpp（https://github.com/ggerganov/whisper.cpp）（2023 年 8 月）

JiWER（https://github.com/zszyellow/WER-in-python）（2023 年 11 月）

（後藤　一章　摂南大学　Email: goto@ilc.setsunan.ac.jp）