英単語難易度算出のための各種コーパス間の単語頻度・単語頻度順位の比較の研究

英単語難易度算出のための各種コーパス間の単語頻度・単語頻度順位の比較の研究

Comparison-based Analyses of Word Frequencies and Word Frequency Rankings of Various Corpora for Accurate Identification of Difficult Words for English-as-a-Second-Language Learners

江原 遥 *
Yo EHARA

Abstract : In English-as-a-Second-Language (ESL) education, mainly word frequencies and word frequency rankings from balanced corpora are used as sources of reliable estimators of English word difficulty. However, numerous recent studies on educational applications of natural language processing show that corpora of movie subtitles and learner corpora also improve the accuracy of identifying words that are difficult or complex for ESL learners to read. In these studies, the frequencies and frequency rankings of the difficult words are used as features to build automatic complex word identifiers by using machine-learning techniques. In this study, word frequencies and word frequency rankings from various corpora and word difficulty indexes are compared and the differences are subjected to quantitative and qualitative analyses.

1. はじめに

コーパス中の単語頻度や単語頻度順位は，英語教育において単語難易度を算出するための代表的な指標になってきた．このためのコーパスとしては，従来，特定の分野になどに偏らない均衡コーパス (Balanced Corpus) の単語頻度が単語難易度算出の上で重視されてきた．例えば，日本の大学英語教育を念頭に作成された単語難易度指標である JACET 8000[21] は，代表的な英語の均衡コーパスである British National Corpus (BNC) [6] の頻度順位をもとに作成されている．

一方，近年，自然言語処理の言語教育応用分野においては，テキスト中で語学学習者のための平易化が必要となる語を発見する複雑単語検出 (Complex Word Identification) をはじめとするタスクにおいて，こうした均衡コーパスの他に，学習者が書いた英文を収集した学習者コーパス (Learner Corpus) 中の単語頻度・単語頻度順位 [11] や，口語に特化した映画の字幕コーパス中の単語頻度・単語頻度順位 [19] が，有力な特徴量となることが報告されている．

本研究では，従来用いられてきた均衡コーパスの単語頻度・単語頻度順位と，学習者コーパスや字幕コーパスのそれがどのように違うのかについて，量的・質的に分析する．具体的には，対象とするコーパスのうち，下記の基本的なリサーチクエスチョンに答えることを目標とする．

(1) どのコーパス間で単語頻度の相関や順位相関が高く，どのコーパス間で低いのか？

(2) 単語頻度の相関・順位相関の計算方法には複数の方法があるが，どの相関・順位相関でも同様の結果になるか？

(3) 単語頻度の相関と，頻度を頻度順位に直した単語頻度順位の順位相関では結果に違いが出るか？

(4) 既存の主要な単語難易度指標の中には，単語頻度順位を，例えば 1,000 語ごとなどに大雑把に区切って組を作り，組内の単語難易度指標は大体同じであると仮定しているものがある．この仮定は，同順を考慮した順位相関の観点からも妥当と言えるか？

(5) 近年，機械学習で単語の難しさを測る際の特徴量として，有効性が報告されている映画字幕コーパスや学習者コーパスは，結局，どのように利用するとよいか？

2. 関連研究

言語教育分野では，単語難易度の算出に均衡コーパス中の単語頻度順位を用いることが一般に行われてきた．

2019 年 6 月 26 日受理
* 情報学部　コンピュータシステム学科

例えば，日本の大学の英語教育のための単語集として標準的に使われている JACET8000 は，British National Corpus [6] の出現単語頻度の順位をもとにしているし，その他，Contemporary Corpus of American English[8] の出現単語頻度順位も幅広く用いられている．こうした単語頻度順位を基にした単語難易度表は，特に学習者の語彙力を測定する単語テストの作成において有用である．例えば，文献 [17] では，Vocabulary Size Test という英語学習者の語彙サイズを測定するテストを提案している．この手法では，単語を BNC コーパスの単語頻度の降順に 1,000 語で区切り，その 1,000 語の単語の難易度を大まかに同じとみなして，各 1,000 語の集合から 5 語を選ぶ形でテストを構成している．この方法に従い，BNC コーパスの 20,000 語から 1,000 語ごとに 5 語を選択して，合計 100 語の単語テストを構成している．各設問は，文中に埋め込まれた語の意味と最も近い単語を４つの選択肢の中から選択する多肢選択式問題である．実際の学習者の語学能力の関係については，特に語彙力と読解力の関係について詳細な報告がある [16, 12, 14]．

一方，自然言語処理分野においては，テキスト全体の難易度を測定するテキストリーダビリティ指標の問題は古くから研究があったものの [9, 7]，語の難しさを特定する問題は，子供用新聞記事などを自動作成するテキスト簡単化 (Text Simplification) のタスクとの関係で注目され始めた．より具体的には，テキスト簡単化の前処理として，語学学習者や子供にとって難しい単語を見つける複雑単語特定 (Complex Word Identification, CWI) が，自然言語処理において単語の難しさを直接的に推定する代表的なタスクの一つである．

自然言語処理においては，テキスト簡単化は，難しいテキストから簡単なテキストへの翻訳，という翻訳の一種として捉えることが多い．実際，大量の翻訳元の文と翻訳先の文の対である対訳コーパスから，自動的に翻訳システムを構築する機械翻訳（Machine Translation, 自動翻訳）技術さえあれば，単純に，翻訳元を難しい文，翻訳先を簡単な文とするだけで，テキスト簡単化を一応は実現することが可能となる．特に，ニューラルネットワークによる機械翻訳，ニューラル機械翻訳 (Neural Machine Translation) が，精度上のブレイクスルーを起こして有望視され始めたあと [13]，テキスト簡単化でも同様のブレイクスルーが起きるのではないかと，集中的に研究が進められている [18]．

テキスト簡単化と，通常の２言語間の機械翻訳が大きく異なる点として，テキスト簡単化では原文（難しい文）のうち，大部分が翻訳先言語に残るという点がある．そこで，テキスト簡単化の前段階の処理として，そもそも簡単化が必要なテキスト中の箇所を特定するタスクが提案され，これが複雑単語特定と呼ばれるタスクである．

複雑単語特定タスクは，近年，様々な特徴量を用いてこれを高精度に行う方法が提案されてきている．語が難しいか簡単かは，当然，語学学習者の言語能力によって感じ方が異なるが，複雑単語特定タスクはテキスト簡単化の前段階のタスクであるため，基本的には語学学習者間の言語能力を捨象し，単純にテキスト中で多くの語学学習者が難しいと報告した語を難しい，と，定義している．実際，ある程度の人数の語学学習者を雇い，テキスト中のどの語が難しいのかを横断的につけさせたデータセットが公開されており，このデータセット上で，難しいと報告された語をより高精度に特定できる手法を良い手法とみなしている．自然言語処理の大部分がそうであるように，まずは精度を比較してみるという立場で，単語のどのような特徴量を組み合わせて使用しても良いので，例えば，複数のコーパスからの単語頻度や単語頻度順位を組み合わせて機械学習による判別器の特徴量として用いても良い．この点は，単語頻度や単語頻度順位の算出に用いたコーパスの性質について深い議論を行う，応用言語学やコーパス言語学分野のアプローチとは大きく異なる．

さて，近年の複雑単語特定タスクで高精度を達成したと報告する研究 [19, 11] では，均衡コーパスの単語頻度や単語頻度順位はもちろん有効な特徴量ではあるが，こうした一般的な内容のコーパス以外に，精度向上に貢献する特徴量として，字幕コーパスの単語頻度・単語頻度順位や，学習者コーパス中の単語頻度・単語頻度順位が挙げられる．こうしたコーパスは，従来の均衡コーパスにはなかった，単語の難易度特定に有用な情報を含んでいると思われる．そこで，本稿では，こうしたコーパスと，従来の均衡コーパスの違いについて，量的・質的に分析する．

3. 比較対象のコーパス

本研究では，下記のコーパスを比較した．下記に，そのコーパスを列挙する．また，太字部分は，コーパスの略称として，論文の以下の部分で用いる．

BNC British National Corpus [6]. イギリス英語の均衡コーパス（特定専門分野に偏りがないように人手で収録文書を調整したコーパス）．

COCA Corpus of Contemporary American English [8]. アメリカ英語の均衡コーパス．

Wiki 英語版 Wikipedia．Wikipedia は多数のユーザによって作られた大規模な百科事典である．収録する文書を研究者が選定しているわけではないため，均衡コーパスではないものの，百科事典の性質から特定分野への偏りが小さいと思われるため，入手の容易性もあり，自然言語処理分野では均衡コーパスのか

表 1: 各コーパスの基本的な性質．Lang8 については 3 mil. 文という報告があるものの，単語数の報告が見つからなかったため，1 文に 10 単語程度含まれると考え概算を示した．

コーパス名	コーパスの性質	述べ語数
BNC	均衡	100 mil.
COCA	均衡	560 mil.
Wiki	百科事典	3,600 mil.[2)]
Subtitles	字幕	225 mil.
Lang8	学習者	30 mil.

わりに頻繁に用いられる．複雑単語特定で均衡コーパスのかわりに用い，高精度を達成したと報告されている [11)]．

Subtitles 映画字幕コーパス．聞き取りによって，オープンな映画の字幕コーパスを作成するプロジェクトがあり [4)]，このプロジェクトの字幕を収集したものである [1)]．複雑単語特定で有益な特徴量となったという報告がある [19)]．

Lang8 学習者コーパス．学習者間の相互採点による語学学習用 SNS から採集され [15)]，公開されているもの．文献 [11)] において，複雑単語特定で有益な特徴量であると報告されている．

表 1 に各コーパスの基本的な情報を列挙した．述べ語数の単位は 100 万語 (million) である．また，述べ語数は，今回比較に用いた各コーパスの単語頻度の和ではなく，低頻度なども全て含むコーパス中の全延べ語数である．

3.1 前処理

コーパスの前処理について説明する．コーパスには，テキストがそのままの状態で含まれているので，単純に英文を空白やカンマやピリオドで分割するだけでは，複数形が別の語としてカウントされてしまうなどの問題がある．そこで，辞書に書かれている原型（lemma）の形に直す作業（lemmatization）が前処理として必要になるが，これには様々な方法がある．

今回は，語学学習の目的における単語頻度順位の比較であるため，既に，語学学習者言語教育の目的で既にそのまま使われている原型化（lemmatization）か，それに近い方法が望ましい．今回は，語学学習者の語彙力測定の代表的な試験である Vocaublary Size Test の作成時に使われた原型化の方法を，そのまま利用した [17)]．文献 [17)] は学習者の英語の語彙サイズを算出する試験であるが，そもそも，語のどのような変形を 1 語とみなすのか，例えば “player” と “play” は同じ語としてカウントするのか，といった原型化の違いにより，語彙サイズの値は大きく変わってしまうため，具体的にどの形を 1 語とみなすのかについて詳述されている．基本的には，文献 [17)] では *word family* という単位を提案しており，端的に言えば，“player” と “play” は 1 語としてカウントするなど，できるだけ原型に近づけた形でカウントすることを提案している．彼らは，実際に文献 [17)] の論文の中で，どの語を 1 語とみなしてカウントしたかを，データの形で，著者らの Web サイト上で提供しているので，このデータを用いて lemmatization を行った．

今回用いた lemmatization の方法は下記の通りである．まず，入力を単純に空白やカンマなどで分割（トークン化）する．これには，Python 言語用の自然言語処理ライブラリである **nltk** [3)] の **word_tokenize** 関数を用いた．次に，各トークンのうち，文献 [17)] が公開している原型化のための語リスト中に含まれている語は，このリストにしたがって原型化した．さらに，文献 [17)] のリストに含まれていない語も，**nltk** ライブラリ中の WordNetLemmatizer 関数によって，文献 [17)] のリストの見出し語のいずれかに変形できれば，文献 [17)] のリストを使って原型化した．

3.2 コーパスの比較に用いる指標

コーパス同士の単語頻度分布の類似している度合いを数値として表すには，どのような指標が適切だろうか？

対応する数値列の似ている度合いを算出する方法として，広く知られていてすぐに思いつく手法としては，単純に 2 変量の相関係数（ピアソンの積率相関係数）を求める方法だが，単語の頻度分布は通常は非線形な分布であるので，相関係数の値を計算しても解釈が難しい．相関係数の値の大小が頻度分布の類似の度合いに起因するものか，頻度分布の非線形な形状に起因するものかがわからないからである．あえて相関係数を使う方法としては，次のような手法が考えられる．単語の頻度分布は，単語を頻度の降順に並べ，縦軸を頻度，横軸を順位とする両対数グラフ上で直線に近くなるという Zipf の法則という経験則に従う事が知られている．この性質を利用して，双方のコーパスともに Zipf の法則に従うことを仮定した上で，頻度値を線形になるように変換してから相関係数を適用すれば，非線形性の問題は緩和され，相関係数の数値が分布の類似を表しているとみなしやすくはなる．しかし，この方法は，そもそも仮定がどの程度妥当かという疑問が生じる上，Zipf の法則は極端な高頻度語や低頻度語では成立せずに，直線から離れることが知られているので，あまり妥当な比較方法とは言えない．

より理論的な疑問の生じにくい単語頻度分布の比較方法としては，大別して次の 2 つが挙げられる．1 つは，単語頻度を単語の総単語数で割ることによって単語の出現確率を求め，確率分布間の類似度や距離を用いる方法である．確率分布間の類似の度合いを測る尺度とし

ては，Kulback-Leibler 情報量（KL 情報量）が広く知られているが，KL 情報量は対称性がない．対称性のある Jensen-Shanon 情報量を用いるほうが，数値を解釈しやすい．

こうした確率分布同士の類似度や距離の尺度は文献 [5] にまとまっている．この中から，本研究では，尺度の著名性を考慮し，コサイン類似度 (文献 [5], (26))，Hellinger 距離 (文献 [5], (35))，Jensen-Shanon 距離 (文献 [5], (51))，Tanimoto 係数 (文献 [5], (23)) を選んだ．

もう 1 つは，語学学習の目的では，単語頻度分布の中でも，特に単語頻度順位に興味があるので，単語頻度順位のみの類似度を計算する方法である．例えばある 2 単語を取ってきた時に，単語頻度の順位が同じであっても，単語頻度分布は，２単語の頻度の比や差に影響されてしまう．語学教育の目的では，結局，どの単語の学習を優先させるかという観点から，単語の難易度の序列だけに興味があるので，頻度の比率や差分の情報は捨象してしまおう，という考え方である．

順位の相関係数としては，Spearman の ρ と Kendall の τ の 2 つが著名であり，実際に自然言語処理分野でも広く使われている．直感的な解釈としては，前者は順位をそのまま数値としてみた時の相関係数（ピアソンの積率相関係数）と説明でき，後者は全ての 2 単語の組み合わせのうち 2 コーパス間で順序が一致しているペアの比率と説明できる．

これらの順位相関係数の式としては，同順 (tie) がないことを仮定した式が広く使われている．一方，語学学習の文脈では，コーパス中の単語頻度順位の他に，あらかじめ何段階に難易度を分けるかを決め，人手で調整しながら単語に難易度の数値を割り振った指標も使われている．こうした指標では，同じ難易度に割り当てられている単語が多数存在する．例えば，後述の **SVL** という指標では 12,000 語に 12 段階の難易度を付与しているので，鳩ノ巣原理より，同じ段階に複数の単語が割り当てられている．同順を考慮した順位相関係数を用いれば，こうした単語難易度指標と，コーパスの単語頻度順位を比較することが可能となる．このため，本研究では同順を考慮した順位相関係数を用いた．

具体的な式を次に記述する．X, Y を n 個の実数値から成る数列として，rg を，所与の数列に対して同順を考慮した順位を返す関数とする．Cov は所与の 2 数列の共分散，SD は所与の数値列の標本分散を表す．同順を考慮した Spearman の ρ は，式 1 のように定義される．式 1 は Python 言語の scipy ライブラリにおいて実装されている．

$$\rho_{X,Y} = \frac{\mathrm{Cov}(\mathrm{rg}(X), \mathrm{rg}(Y))}{\mathrm{SD}(\mathrm{rg}(X))\mathrm{SD}(\mathrm{rg}(Y))} \tag{1}$$

また，同順を補正した Kendall の τ は次のように定義される．この同順の補正方法は tau-b と呼ばれ，Python 言語の scipy ライブラリにおいて実装されているのはこの方法である．

$$\tau_{X,Y} = \frac{\mathrm{N_C}(X,Y) - \mathrm{N_D}(X,Y)}{(\mathrm{N_P}(n) - \mathrm{N_{TP}}(X))(\mathrm{N_P}(n) - \mathrm{N_{TP}}(Y))} \tag{2}$$

ただし，式 2 において，$\mathrm{N_C}$ は同順を除き順序が一致するペアの数，$\mathrm{N_D}$ は同順を除き順序が一致しないペアの数，$\mathrm{N_P}(n)$ は単純な n 個のペアの数，すなわち，$\mathrm{N_P}(n) = n(n-1)/2$，$\mathrm{N_{TP}}$ は所与の数列が作るペアのうち同順のものの数である．

表 2: 単語頻度分布間のコサイン類似度

-	COCA	Wiki	Subtitles	Lang8
BNC	0.0086	0.0138	0.0078	0.0092
COCA	-	0.0113	0.0093	0.0105
Wiki	-	-	0.0061	0.0071
Subtitles	-	-	-	0.0133

3.3 順位相関で比較する単語難易度指標

前述のように，同順を考慮した順位相関を用いれば，コーパスの単語頻度順位と，単語に人手で何段階かの難易度を割り振った単語難易度指標の類似度を比較することが可能となる．本研究の実験では，具体的に次の難易度指標を比較した．

JACET 大学英語教育学会基本語リスト，2003 年版 [21]．BNC コーパスの単語頻度順位を基に，時事英語など様々なコーパスの頻度リストを用いて統計的に調整したもの [22]．

SVL12000 標準語彙水準 SVL(Standard Vocabulary List)12000[20]．やはり，BNC コーパスの単語頻度順位を基にして，様々なコーパスの頻度リストを組み合わせて統計的に作成されているが，加えて，英語だけでなく日本語コーパスをも用いて，「日本人がよく使うような表現をリストに」加えることによって作成されている [22]．

BNClevel 英語学習者の語彙量を計測するための Vocabulary Size Test[17] の作成のために使われた，BNC コーパスの頻度 20,000 語について，頻度順位の降順に 1,000 語ごとのグループに分け，各単語に ⌊順位/1000⌋ + 1 のレベルをつけたもの．ただし，実数 x に対して $\lfloor x \rfloor$ は x を超えない最大の整数を表す（床関数）．

表 3: 単語頻度分布間の Hellinger 距離

-	COCA	Wiki	Subtitles	Lang8
BNC	0.3714	0.5556	0.8309	0.7467
COCA	-	0.5780	0.6731	0.6141
Wiki	-	-	1.0305	0.8707
Subtitles	-	-	-	0.5456

表 4: 単語頻度分布間の Jensen-Shanon 距離

-	COCA	Wiki	Subtitles	Lang8
BNC	0.1709	0.3618	0.8666	0.7813
COCA	-	0.3814	0.4897	0.4100
Wiki	-	-	1.2672	0.8529
Subtitles	-	-	-	0.3237

4. 実験結果

まず，単語頻度分布間の相関を見てみよう．比較するコーパスとしては，表 1 に挙げたコーパスを比較した．コーパス同士の単語頻度分布間の類似度/距離として，表 2 にコサイン類似度を使った場合を示す．同様に，表 3 に Hellinger 距離を用いた場合，表 4 に Jensen-Shanon 距離を用いた場合，表 5 に Tanimoto 距離を用いた場合を示す．類似度の場合は値が大きいほど分布が類似しており，距離の場合は値が小さいほど分布が類似している．これらの類似度/距離関数は対称であるため，各表の下三角部分は，対応する上半分のコーパスペアの値と同一となる．このため，見やすさを考慮して下三角部分は省略してある．また，同じコーパス同士の単語頻度分布の類似度/距離には今回は興味がないため，これも各表中で省略した．この結果，各表の中には，相異なるコーパスのペアごとに，10 個の類似度/距離の値が算出された．

これらの表をさらに表 6 にまとめ，数値を省略し，各類似度/距離の算出手法ごとに，単語頻度分布が似ていると算出された順に各コーパスのペアを記載した．コーパスのペアはコーパス名同士を “-” でつないで表記した．順位の列は，各コーパスのペアが何位目に似ていると算出されたかを示す．

表 6 を見ると，コサイン類似度を除く Hellinger 距離，Jensen-Shanon 距離，Tanimoto 距離において **BNC-COCA** の単語頻度分布が最も似ていると算出されたことがわかる．イギリス英語とアメリカ英語の違いがあるものの，両者とも大規模な均衡コーパスであること，表 1 で同じ性質を持つコーパスはこの 2 つしかないことから，この結果は理解できる結果といえよう．逆に，表 6 では，コサイン類似度，Hellinger 距離，Jensen-Shanon 距離，Tanimoto 距離において，**Wiki-Subtitles** が最も似ていないと算出されていることは興味深い．百科事典と字幕という，直感的にも似ているとは思えないコーパスの性質が，そのまま反映されていると考えられる．

表 5: 単語頻度分布間の Tanimoto 距離

-	COCA	Wiki	Subtitles	Lang8
BNC	0.2650	0.4232	0.6255	0.5609
COCA	-	0.4546	0.5567	0.4992
Wiki	-	-	0.7325	0.6545
Subtitles	-	-	-	0.4293

ここで，表 2 を見てみると，どのコーパスペアの類似度も小さい値を取り，あまり違いがないことがわかる．コサイン類似度は，他の確率分布間の距離とは異なり，単語頻度分布をベクトルとみなして，ベクトル同士の角度の違いをみるものである．高次元空間では，ランダムなベクトル同士でも次元が増えることによって直交しやすくなるので，ほとんどのベクトルが直交している，すなわち，ほとんど類似していないとみなされてしまう，「次元の呪い」と呼ばれる現象が知られている [10]．コサイン類似度のみ，他の距離と比べて大きく違った結果になっている原因は，おそらくこの現象のためであると推察される．したがって，表 6 の最も左のコサイン類似度によるコーパスペアの比較は信頼性がないと思われる．前段落の分析については，信頼性のないコサイン類似度を除いた，他の 3 種類の距離を用いた分析である．

4.1 順位相関係数による比較

次に，順位相関による比較を見よう．表 7 に各コーパス・単語難易度指標間の Spearman の順位相関係数を，表 8 に Kendall の順位相関係数を表示した．前述のように，どちらも同順を考慮してある．

表 7 と表 8 の 2 つをまとめ，順位相関係数が高く出たコーパス・単語難易度指標のペアを表 9 にまとめた．

表 9 では，著者の知る限りこれまで報告されていなかった有益な知見が何点かある．まず，Spearman と Kendall の 2 つの順位相関係数において，**Lang8-SVL** が最も相関が高いペアであると算出された．SVL は BNC の頻度を基にはしているものの，前述のように，日本語が母語の英語学習者が特に書きそうな表現を考慮して，単語難易度を調整してある．一方，Lang8 は BNC とは全く関係ない学習者が書いた英文のコーパスである．Lang8 の利用者の母語は日本語に限定されておらず，今回は英語を学習対象としている利用者の英語の日記を，母語を限定せずに抽出しているが，Lang8 自体が日本のサービスであるため，利用者の大半が，日本語を母語として英語を学ぶ学習者である．

おそらく，**Lang8-SVL** は，SVL の単語難易度順位に，日本語を母語とする英語学習者特有の考慮がうまく働

表 6: 単語頻度分布間の類似度算出法ごとの似ているコーパスペア（似ている順）

順位	コサイン類似度	Hellinger 距離	Jensen-Shanon 距離	Tanimoto 距離
1	**BNC-Wiki**	**BNC-COCA**	**BNC-COCA**	**BNC-COCA**
2	**Subtitles-Lang8**	**Subtitles-Lang8**	**Subtitles-Lang8**	**BNC-Wiki**
3	**COCA-Wiki**	**BNC-Wiki**	**BNC-Wiki**	**Subtitles-Lang8**
4	**COCA-Lang8**	**COCA-Wiki**	**COCA-Wiki**	**COCA-Wiki**
5	**COCA-Subtitles**	**COCA-Lang8**	**COCA-Lang8**	**COCA-Lang8**
6	**BNC-Lang8**	**COCA-Subtitles**	**COCA-Subtitles**	**COCA-Subtitles**
7	**BNC-COCA**	**BNC-Lang8**	**BNC-Lang8**	**BNC-Lang8**
8	**BNC-Subtitles**	**BNC-Subtitles**	**Wiki-Lang8**	**BNC-Subtitles**
9	**Wiki-Lang8**	**Wiki-Lang8**	**BNC-Subtitles**	**Wiki-Lang8**
10	**Wiki-Subtitles**	**Wiki-Subtitles**	**Wiki-Subtitles**	**Wiki-Subtitles**

表 7: 単語頻度順位間の Spearman's ρ

-	COCA	Wiki	Subtitles	Lang8	JACET	SVL	BNClevel
BNC	0.7808	0.7396	0.6891	0.7179	0.8250	0.7224	0.8094
COCA	-	0.6427	0.8013	0.7911	0.8146	0.7492	0.7900
Wiki	-	-	0.6734	0.7071	0.6909	0.6308	0.7072
Subtitles	-	-	-	0.8121	0.7471	0.8072	0.7919
Lang8	-	-	-	-	0.8180	0.8509	0.7751
JACET	-	-	-	-	-	0.7803	0.8148
SVL	-	-	-	-	-	-	0.7833

いたため，日本語母語話者の学習者による作文を集めた **Lang8** コーパスと高い順位相関が見られている，と推察される．

また，単語難易度指標 **SVL** は BNC コーパスを基にはしているものの，**BNC** との順位相関は Spearman でも Kendall でも 20 位と低く算出されていることがわかる．**SVL** は BNC の単語頻度順位に比較的忠実に作られていると報告されている **JACET** とは異なり，相当に日本語母語話者の英語学習者向けの調整をして設計されたことが報告されているが [22]，この事が実際にデータからも判明した．

5. 考察

本節では，最初のリサーチクエスチョンとした問いに，前節の結果を用いて，現状で分かる範囲の回答を与える．

(1) どのコーパス間で単語頻度の相関や順位相関が高く，どのコーパス間で低いのか？
表 6, 表 9 に示した．

(2) 単語頻度の相関・順位相関の計算方法には複数の方法があるが，どの相関・順位相関でも同様の結果になるか？
表 6, 表 9 の上位・下位に来ているペアは，ほぼ共通していることから，上位・下位の数件は同様の結果になるものの，当然，中央付近では異なる結果になる．ただし，表 6 のコサイン類似度のように，そもそも単語頻度分布の類似度の計測に適さない尺度もある．

(3) 単語頻度の相関と，頻度を頻度順位に直した単語頻度順位の順位相関では結果に違いが出るか？
表 6, 表 9 を比べると分かるように，明確な違いが出る．例えば，表 6 では最も頻度分布が近いとされた **BNC- COCA** は，表 9 を見ると分かるように，単語頻度順位はそこまで近いとは言えない．語学学習上は，語を学習させる順番など，順序に主たる興味がある場合があるが，こうした場合には，頻度と頻度順位を明確に区別した分析を行った方が良いことが推察される．

(4) 既存の主要な単語難易度指標の中には，単語頻度順位を，例えば 1,000 語ごとなどに大雑把に区切って組を作り，組内の単語難易度指標は大体同じであると仮定しているものがある．この仮定は，同順を考慮した順位相関の観点からも妥当と言えるか？
BNClevel は前述のように，**BNC** の 1,000 語ごとのグループの難しさは全て同一とみなしてしまう，粗い単語難易度指標である．当然なら，同一とみなさ

表 8: 単語頻度順位間の Kendall's τ

-	COCA	Wiki	Subtitles	Lang8	JACET	SVL	BNClevel
BNC	0.6275	0.5809	0.5118	0.5447	0.6416	0.5538	0.6477
COCA	-	0.4772	0.6129	0.6114	0.6270	0.5767	0.6214
Wiki	-	-	0.4864	0.5222	0.5006	0.4657	0.5301
Subtitles	-	-	-	0.6296	0.5587	0.6366	0.6210
Lang8	-	-	-	-	0.6351	0.6853	0.6105
JACET	-	-	-	-	-	0.6154	0.6511
SVL	-	-	-	-	-	-	0.6328

表 9: 単語頻度順位間の順位相関係数ごとの似ているコーパスペア（似ている順）

順位	Spearman's ρ	Kentall's τ
1	**Lang8-SVL**	**Lang8-SVL**
2	**BNC-JACET**	**JACET-BNClevel**
3	**Lang8-JACET**	**BNC-BNClevel**
4	**JACET-BNClevel**	**BNC-JACET**
5	**COCA-JACET**	**Subtitles-SVL**
6	**Subtitles-Lang8**	**Lang8-JACET**
7	**BNC-BNClevel**	**SVL-BNClevel**
8	**Subtitles-SVL**	**Subtitles-Lang8**
9	**COCA-Subtitles**	**BNC-COCA**
10	**Subtitles-BNClevel**	**COCA-JACET**
11	**COCA-Lang8**	**COCA-BNClevel**
12	**COCA-BNClevel**	**Subtitles-BNClevel**
13	**SVL-BNClevel**	**JACET-SVL**
14	**BNC-COCA**	**COCA-Subtitles**
15	**JACET-SVL**	**COCA-Lang8**
16	**Lang8-BNClevel**	**Lang8-BNClevel**
17	**COCA-SVL**	**BNC-Wiki**
18	**Subtitles-JACET**	**COCA-SVL**
19	**BNC-Wiki**	**Subtitles-JACET**
20	**BNC-SVL**	**BNC-SVL**
21	**BNC-Lang8**	**BNC-Lang8**
22	**Wiki-BNClevel**	**Wiki-BNClevel**
23	**Wiki-Lang8**	**Wiki-Lang8**
24	**Wiki-JACET**	**BNC-Subtitles**
25	**BNC-Subtitles**	**Wiki-JACET**
26	**Wiki-Subtitles**	**Wiki-Subtitles**
27	**COCA-Wiki**	**COCA-Wiki**
28	**Wiki-SVL**	**Wiki-SVL**

ず単語頻度順位そのままであれば，どちらの順位相関係数でも，完全に **BNC** 単語頻度順位と相関して 1.0 の値を取るはずである．とすると， 表 7 や表 8 において，1.0 から **BNC-BNClevel** が下がった度合いが，この操作の粗さを数値的に示していると捉えることができる．例えば，表 7 では，**BNC-BNClevel** は 0.8094 の値を取っており，1,000 語ごとのグループ化は，順位相関係数を 0.1906 ほども押し下げる効果があることが分かる．しかし，表 9 を **BNC-BNClevel** が，両方の順位相関係数で，他のペアに比べて相対的には上位に来ていることから，もとの **BNC** の単語頻度順位が持つ情報は，この粗い近似でも相対的には保たれているといえる．人間が解釈する必要のある場面ではそれなりに妥当と言えよう．

(5) 近年，機械学習で単語の難しさを測る際の特徴量として，有効性が報告されている映画字幕コーパスや学習者コーパスは，結局，どのように利用するとよいか？

機械学習の特徴量として用いる場合，複数の特徴量の有効な組み合わせ方は機械学習器が決定するので，通常は，特徴量はなるべく相関しておらず，異なる種類の特徴量を組み合わせた方が効果的である．この観点からは，表 6 の下位のペアほど，頻度分布が相関しておらず，頻度分布が相異なる情報を持っていると考えられるので，組み合わせて利用することで，単語の難易度を判別する特徴量として有効である事が期待される．

表 6 で最も下位に来ているのが **Wiki-Subtitles** であり，Wikipedia の単語頻度分布と字幕コーパスの単語頻度分布の組み合わせが有効そうである事が見て取れる．実際に，文献 [19] では，文献 [19] 以前に頻繁に用いられていた Wikipedia 中の単語頻度に加えて，字幕コーパス中の単語頻度をあわせて特徴量として用いることで，学習者に取って難しい語を予測する複雑単語特定が高精度にできることを報告し，字幕コーパスの有用性を主張している．

文献 [19] では，作成に労力のかかる「実際に学習者に難しい単語を報告してもらった教師データ」に対

して回帰する事により，字幕コーパスの有用性を実証している．一方，本研究は，そうした作成に労力のかかる教師データなしに，単純なコーパス間の単語頻度の相関を見るだけで，有用な特徴量の組み合わせを予想することができた．

上記の (5) について補足する．複雑単語特定のタスクでは，字幕コーパスの他，[11] が学習者コーパス **Lang8** を Wikipedia の単語頻度と組み合わせて，高精度を達成している．Wikipedia の単語頻度に加え，字幕コーパスの単語頻度と学習者コーパスの単語頻度を両方用いる方法が，すぐに思いつき，この方法は，著者の知る限り，まだ試されていない．しかし，表 6 では，**Lang8-Subtitles** が単語頻度分布の類似度２位に来ていることから，両者が強く相関する特徴量であることが示唆される．したがって，字幕コーパスと学習者コーパス **Lang8** の単語頻度を両方特徴量に入れる方法は，複雑単語特定の精度をそこまで向上させないであろう，と予想できる．

この予想は，コーパスの性質からは推察することが難しい予想である．実際，表 1 を見ると，学習者の英作文を集めた学習者コーパスと，映画字幕コーパスというように，**Lang8** と **Subtitles** は性質が大きく違っているようにみえる．

表 6 からは，それよりも，Wikipedia とも，そこまで相関が高いとは言えない **COCA** を特徴量として加えた方が，有望そうに見える．こうした予想が的中するかを検証する事が，今後の課題である．

6. まとめ

本研究では，英語学習者にとって難しい単語を予測する複雑単語特定などのタスクで，近年有効性が報告されている，映画字幕コーパスと学習者コーパスなどを含め，様々なコーパスの単語頻度・単語頻度順位を比較した．単語頻度順位の比較においては，同順をも考慮する事により，単語難易度指標と比較することをも可能にした．

分析の結果，特に重要な知見として，映画字幕コーパスと **Lang8** 学習者コーパスの単語頻度分布が非常に似通っている事，Wikipedia の単語頻度分布と，これらのコーパスの単語頻度分布が大きく異なっていることがわかった．これは，Wikipedia と映画字幕コーパスまたは **Lang8** 学習者コーパスをあわせて特徴量として用いる事によって性能を向上したという近年の報告と一致する結果であり，なぜこれらを組み合わせる事が有用であるのかを一部説明することに成功した．

また，日本で広く使われている英単語難易度指標として **JACET** や **SVL** が挙げられるが，どちらも **BNC** コーパスの単語頻度を基に作成されているにもかかわらず，**SVL** は日本語母語話者に合わせた調整によって **BNC** コーパスとは大きく異なっていることがわかった．**SVL** は，日本語母語話者による英作文が大半を占める **Lang8** コーパスの単語頻度順位と強く相関しており，**SVL** の調整の妥当性を数値的に示すことができた．

今後の課題としては，有効性が予想された特徴量の組み合わせを，実際に複雑単語特定の特徴量として用いて，予想された結果が達成されるかを検証する事が挙げられる．

謝辞

本研究は JST（ACT-I，課題番号: JPMJPR18U8）の支援を受けたものである．また，産業技術総合研究所（産総研）の AI 橋渡しクラウド (ABCI) を利用した．

参考文献

(1) http://ghpaetzold.github.io/subimdb/.
(2) https://en.wikipedia.org/wiki/Wikipedia:Size_comparisons.
(3) https://www.nltk.org/.
(4) https://www.opensubtitles.org/.
(5) Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions. *City*, Vol. 1, No. 2, p. 1, 2007.
(6) The BNC Consortium. *The British National Corpus, version 3 (BNC XML Edition).* Bodleian Libraries, University of Oxford, 2007.
(7) Edgar Dale and Jeanne S. Chall. A formula for predicting readability. *Educational Research Bulletin*, Vol. 27, No. 1, pp. 11–20, 28, 1948.
(8) Mark Davies. The 385+ million word corpus of contemporary american english (1990–2008+): Design, architecture, and linguistic insights. *International journal of corpus linguistics*, Vol. 14, No. 2, pp. 159–190, 2009.
(9) Rudolf Flesch. A new readability yardstick. *Journal of applied psychology*, Vol. 32, No. 3, p. 221, 1948.
(10) Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 統計的学習の基礎: データマイニング・推論・予測. 共立出版, 2014.
(11) Tomoyuki Kajiwara and Mamoru Komachi. Complex word identification based on frequency in a learner corpus. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 195–199, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.
(12) Batia Laufer and Geke C Ravenhorst-Kalovski. Lexical threshold revisited: Lexical text coverage, learners' vocabulary size and reading comprehension. *Reading in a foreign language*, Vol. 22, No. 1, pp. 15–30, 2010.

(13) Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

(14) Paul M. Meara. *EFL Vocabulary Tests (Second Edition)*. Lognostics (Center for Applied Language Studies, University of Wales), Swansea, 2010.

(15) Tomoya Mizumoto, Mamoru Komachi, Masaaki Nagata, and Yuji Matsumoto. Mining revision log of language learning sns for automated japanese error correction of second language learners. pp. 147–155, Chiang Mai, Thailand, November 2011. Asian Federation of Natural Language Processing.

(16) Paul Nation. How large a vocabulary is needed for reading and listening? Vol. 63, No. 1, pp. 59–82, 2006.

(17) Paul Nation and David Beglar. A vocabulary size test. Vol. 31, No. 7, pp. 9–13, 2007.

(18) Sergiu Nisioi, Sanja Štajner, Simone Paolo Ponzetto, and Liviu P. Dinu. Exploring neural text simplification models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 85–91, Vancouver, Canada, July 2017. Association for Computational Linguistics.

(19) Gustavo Paetzold and Lucia Specia. Collecting and exploring everyday language for predicting psycholinguistic properties of words. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 1669–1679, Osaka, Japan, December 2016.

(20) SPACE ALC Inc. Standard vocabulary list 12,000, 1998.

(21) Toshihiko Uemura, Shin ’ichiroIshikawa. Jacet 8000 and asia tefl vocabulary initiative. *Journal of Asia TEFL*, Vol. 1, No. 1, pp. 333–347, 2004.

(22) 投野由紀夫. 教材とコーパス. 立命館言語文化研究, Vol. 16, No. 4, pp. 157–167, 2005.