DEIM2021 F25-2

ウィキペディアを利用した単語の難易度推定

山河絵利奈† 田島 敬史††

† 京都大学工学部情報学科 〒606–8501 京都府京都市左京区吉田本町
†† 京都大学大学院情報学研究科 〒606–8501 京都府京都市左京区吉田本町
E-mail: †yamakawa@dl.soc.i.kyoto-u.ac.jp, ††tajima@i.kyoto-u.ac.jp

あらまし Web 上にある技術記事は，様々な知識レベルの人に向けたものが混ざっており，自分が読むのに適したレベルの記事を探す必要がある．単語の難易度コーパスをもとに文章の難易度を推定する方法はあるが，技術分野において適切な難易度コーパスを見つけることは容易でない．本研究では，難易度区別のないデータから単語の難易度を推定する方法を提案した．難易度の推定には，ウィキペディアのデータを使用する．まず，1 つのページを単語の説明とみなし，単語の説明にどの語を使用するかという情報から二単語間の難易度順を推定する．それを元に作成した重み付き有向グラフに対し，重みを考慮するよう拡張された Relational Power の計算を行い，全体の難易度順を決定する．実験では，提案手法で推定した難易度と，実際の認知度を比較して，推定精度の検証を行なった．実験の結果として，提案手法の一部は既存手法やベースラインよりも高い精度で単語の難易度を予測できていることがわかった．

キーワード 難易度，Web 検索，文書データ，テキストマイニング

1 序 論

Web 上には，初学者から専門家まで，様々なレベルの知識を持つ人に向けた技術記事がある．多くの場合，ユーザーは記事を読む際に，簡単すぎず，かつ自分の知識で理解できる難易度の記事を求める．記事に対して難易度を推定することができれば，ユーザーが求める難易度の記事を選ぶことが容易となる．

記事，すなわち文章に対して難易度を推定する方法は複数研究されている．その方法の一つに，単語に難易度を設定し，それを用いて文章の難易度を計算する方法がある．例えば，西原 [1] らは，文章に対して展望台システム [2] により特徴語を抽出し，特徴語の難易度を評価した上でその和から Web ページの難易度を測定した．川村ら [3] は，単語を難易度別に 6 段階に分け，段階ごとの単語の数と一文の長さを指標に重回帰分析を行うことで，文章の難易度判定式を得た．

このような方法を用いて文章の難易度を推定するためには，単語に対して難易度を付与する必要がある．川村ら [3] は，日本語学習者を対象とした難易度判定システムの構築を目的として，日本語検定の出題基準の級ごとに難易度を分けている．近藤ら [4] は，小学校から高校の全学年の教科書を基準とし，漢字一字を擬似的な単語とみなして文字 unigram モデルで難易度の推定を行なっている．これらの手法はいずれも難易度の段階が与えられているテキストをもとにコーパスを作成している．そのため，この方法を利用するためには分野ごとに適切に難易度が付与されたテキストを発見する必要がある．しかし，技術分野において多段階の難易度が設定されたテキストを収集することは難しい．また技術分野においては高頻度で新たな単語が出現するため，単語が増えるごとにその単語を含む難易度が与えられているテキストを収集することは非現実的となる．

このような背景から，本論文では，難易度が与えられていない文章である日本語版ウィキペディア [1]を利用した単語難易度の推定方法を提案する．本研究では，ウィキペディアの文章データを，各ページのタイトルを一つの単語とみなし，そのページを単語を説明する文章であると考える．

難易度を推定するにあたり，難易度の定義が必要である．本研究では，単語の難易度は，その単語の持つ概念の難しさであると考える．したがって，原語での表記とカタカナでの表記など，同一のものを表す場合は同じ単語として扱うよう留意する．

提案手法ではまず，各ページにおける単語の出現状況から，二つの単語間の推定難易度順を表す有向グラフを作成する．さらに，重みを与え，より正確に状態を表現することを考える．

そして，作成した有向グラフから，全単語の難易度の順序を推定する方法を提案する．これは，Relational Power を用いる．Relational Power を用いた有向グラフからのランキング生成はエッジの重みを考慮しないものであるが，重みの情報を使用した計算方法に拡張する．

実験は，提案手法を用いてウィキペディアのデータから難易度を推定した．また，その順序と単語の実際の認知度の順序の Spearman の順位相関係数 [5] と Kendall の順位相関係数 [6] を計算した．難易度の推定は，提案手法による推定のほか，西原 [1] らによる手法，提案手法の一部分のみを適応した手法での推定も行い，それぞれ認知度の順位との順位相関係数を算出し，比較した．これにより，難しいと推定された単語へ向けてエッジを張った重み付き有向グラフに対して Relational Power を用いて難易度順を推定する方法の内 1 手法がベースラインよりも高い精度を示した．また，重みを与えることが難易度推定の精度の向上に寄与していることがわかった．

本論文は以下の構成を取る．まず，第 2 章で関連研究について述べる．続いて第 3 章では提案する単語の難易度を推定手法

1：https://ja.wikipedia.org/

について述べる．第 4 章では提案手法に対する実験と実験結果について説明し，第 5 章ではその結果を踏まえた考察を述べる．最後に第 6 章で，本研究の結論と今後の展望を述べる．

2 関 連 研 究

本章では，本研究に関連する研究について述べる．

2.1 コーパスを用いた難易度推定

難易度の推定方法の一つに，難易度の段階が与えられているテキストをもとに作られたコーパスを利用したものがある．Collins-Thompson ら [7] は，難易度が既知であるテキストで構成されたコーパスを用いた英語文章の難易度推定を行なった．具体的には，難易度 Gi に対して単語 Unigram モデルである言語モデル Mi を考え，Mi において文字 x が生起する確率 $P(x|M_i)$ を

$$P(x|M_i) = \frac{C(x, D_i)}{\sum_{z \in D_i} C(z, D_i)} \tag{1}$$

として，テキスト T の難易度を

$$L(M_i|T) = \sum_{z \in T} C(z, T) \log P(z|M_i) \tag{2}$$

で得られた尤度が最大となる言語モデルに対応する難易度としている．ここで，D_i は難易度 G_i のテキスト集合を指し，$C(z, D_i)$ は D_i 中の文字 z の出現回数を表す．

水谷ら [8] は愛知県名古屋市で使用されている小学校・中学校・高校の英語を除く全教科の教科書と大学の教養課程のテキストを学習データとし，単語のウェブ出現頻度，単語が日本漢字能力検定に出現した最も簡単な級，難易度が与えられた文章集合の中で単語が出現した最も難易度が低い文章の難易度，単語の難易度ごとの出現頻度で重み付けした単語が出現する文書の難易度の平均値を素性として Support Vector Regression (SVR) による単語の難易度推定を行なった．

藏ら [9] は，日本語能力試験の試験区別がついている単語をもとに，SVM を利用したブートストラップ法で単語の難易度推定を行なっている．

これらはいずれも難易度が与えられた単語データセットが存在することが前提となっている．本論文では，難易度がない単語・文章データから単語の難易度を推定することを試みる．

2.2 出現数を用いた単語難易度推定

難易度段階のあるデータを用いない推定も研究されている．

西原ら [1] は，単語は出現頻度が低いほど難しい単語であると考え，Web ページ集合における単語の出現頻度を用いて単語に対して難易度の評価を与えている．単語 t に対して，評価値は式 (3) で与えられている．

$$Sp(t) = \frac{ALLP - Pnum(t)}{ALLP} \tag{3}$$

ただし，$ALLP$ は Web ページの総数，$Pnum(t)$ は t を含む Web ページ数を表している．

本研究では，このアイデアを拡張し，単語を説明する文章における他の単語の出現状況を利用する手法を提案する．

2.3 ランキング生成

難易度の順序を生成する方法は，一種のランキング生成であるとみなせる．ランキングを生成する方法は，古くから研究がされ，数多くの手法が提案されている．

櫻庭ら [10] は，二者間の重み付き順序集合をもとに全体の順位付けを行う線形順序付け問題に対する，様々な解法をまとめている．その中には，厳密的な解を得る方法，発見的解法，局所探索法などがあげられている．厳密的な解を得る方法は，代表的な手法として分枝限定法があげられている．しかしこれは非常に時間のかかるものが多く，データ数が多い場合には有効でない．発見的解法は，頂点に何らかのスコアを与え順序を決める方法や，二者間の重み付き順序集合を有向グラフとして，閉路をなくすことで順序関係を作成する方法があげられている．局所探索法は，適切な初期解から 1 つずつ適切な順位に挿入し直すことで解を得る方法で，短時間で解を得る方法を挙げている．しかし，これは最適解が得られる保証はない．

頂点に何らかのスコアを与え順序を決定する方法には様々なものがあり，そのうちの一つに Relational Power を計算するものがある．Ryuo ら [11] も，その計算方法を提案している．詳細については，3.4 で述べる．

3 提 案 手 法

この章では，難易度が付与されていない文章データであるウィキペディアから単語の難易度を推定する方法を提案する．

3.1 使用するデータ

本研究では，日本語版ウィキペディアを用いて単語の難易度を推定することを試みる．ウィキペディアは難易度の情報持たず，また，誰でも編集ができるという特性上，新たな技術用語や専門用語のページの作成が迅速に行われる．よって，本研究のデータとして使用するのにふさわしいと考えられる．

3.2 難易度の定義

本研究で評価したい単語の難易度とは，単語の持つ概念に対する難易度である．言い換えると，その単語の理解に必要な知識が多いほど，単語が難しいと考える．そのため，表記の違いによる難易度差は生まれないとする．例えば,「危険」という単語の難易度は，その英訳である「risk」や英訳の発音をカタカナで表した「リスク」などと同じであると考える．

3.3 単語間関係の有向グラフでの表現

本節では，二つの単語の比較してその難易度順を推定し，その関係を有向グラフとして表すことを試みる．本研究では，ウィキペディアの各ページを，単語に関する説明であると解釈する．具体的には，各ページのタイトルを一つの単語とみなし，ページ内容をその単語について説明している文章であると考える．

まず，ある語を説明するのにそれよりも難しい語を使うことは少ないと考える．つまり，ある単語 W のページに出現する単語は，単語 W より簡単な単語であると推定する．

この推定を元に，単語をノードとした，二単語間の難易度順

推定を表す有向グラフを作成することを考える．このグラフは，二種類考えることができる．一つは，単語 i から単語 j へのエッジを，i より j が簡単であるとするグラフ，もう一つは，単語 i から単語 j へのエッジを，i より j が難しいとするグラフである．これらのグラフは，互いにエッジを逆に向けたものとなる．なお，ノード i からノード i への自己ループは作成しない．

ここで，いくつかの事象を考慮する必要がある．

一つ目は，表記揺れの存在である．同じ概念を表す場合でも，様々な理由によって表現が異なる可能性がある．例として,「漢字での表記」と「平仮名での表記」,「正式な表記」と「略語での表記」などがある．これらは内容として表しているものは同じである．本研究は，単語の持つ概念に対し難易度を推定するため，これらを同じ単語として扱うべきである．

二つ目に，各ページのタイトルに，区別上の表記が加えられている場合である．ウィキペディアはページのタイトルが全て異なるように作成されているため，同じ単語で複数の意味を持つ語がある場合に，括弧書きで区別できるような表記を加えていることがある．この場合，タイトルをそのまま単語とみなすと，括弧書きを含めた表記で他のページの文章中に記載されているとは考えにくく，適切な有向グラフが得られない．

三つ目は，双方向にエッジが存在する可能性である．作成した有向グラフは，二単語間の難易度順推定を表すことが目的であるが，実際は単語の出現を表している．そのため，単語 A のページに単語 B が現れ，単語 B のページにも単語 A が現れる場合には，単語 A と単語 B を表すノード間に互いに逆を向いた二本のエッジが存在することになる．厳密に難易度順推定を表すならば，少なくとも一方のエッジを削除する必要がある．

四つ目は，文章中に高い難易度の語が現れる可能性である，例えば，その単語の持つ概念を応用するような概念があり，そのことについて説明内で言及するケースが挙げられる．

五つ目は，説明における単語の重要度の差である．作成した有向グラフのエッジは，出現したか否かのみを表す．しかし説明には，その説明に必要で何十回も出現する単語もあれば，あまり本質的に関係するわけではなく数回出現するだけの単語もあると考えられ，これらを一律で扱うことが適切とは言い難い．

これらの事象の反映を試み，以下のようにグラフを作成する．

まず，単語が出現しているか否かを，単純に表記の一致で数えるのではなくその語のページへのリンクが存在するか否かで考えて，エッジを作成する．つまり，単語 A のページに，単語 B のページへのリンクがある場合，その単語が出現したとみなし，エッジを作成する．単語 B そのものが出現しても，リンクがない場合には出現とはみなさない．これは，ウィキペディアの文章中のリンクは，リンクを表す文字列と同じタイトルのページへ繋げることが基本であるが，異なるタイトルを持つページに繋げることもできることに基づくものである．これにより，単語とみなしたタイトルと文章中の出現で表記が異なっても，出現したと反映できる．また，ページ区別上の表記がつけられた単語の出現についても認識することができる．このとき，単語の出現回数は，リンクを表す文字列の出現回数とする．例えば,「オペレーティングシステム」のページへのリンクが「OS」という文字列で表されていたとき，文章中の「OS」の出現回数を数え,「オペレーティングシステム」の出現回数とする．

続いて，エッジの片方向化を行う．単語 A と単語 B を比較したとき，単語 A が難しいとし，どちらのページにも他方の単語が現れているとする．この時，単語の説明にはその単語よりも簡単な単語が使用されることが基本であると考えると，単語 A のページに単語 B が出現する数の方が，単語 B のページに単語 A が出現する数よりも多いと考えられる．よって，出現回数を比較し，一方のエッジを削除する．単語の出現回数は文書長にも強く依存する．しかし，ウィキペディアの文章の長さはページによって全く異なる．そのため，文章の長さが影響を及ぼさないように，そのページに出現する単語の総数で割った値を比較する．これは以下のように表される．

$$\frac{c(i,j)}{\sum_k c(i,k)} > \frac{c(j,i)}{\sum_k c(j,k)} \Leftrightarrow \text{エッジ } (i,j) \text{ が存在する} \quad (4)$$

ここで，$c(i,j)$ は j のページへのリンクを表す言葉の i のページでの出現回数である．これにより，双方向にエッジが張られることがなく，エッジが二つの単語間の難易度順推定を表すことになる．また，これにより，総出現単語数に対する出現数が同じである単語間には，どちらが難しいかの推定はされない．

最後に，エッジに重みを与える．重みはそのページでの出現度合いを表す．単純には，そのページに出現する単語の総数で割ることでそのページの中での出現の割合を表せる．しかし，これでは (出次数が 1 以上の) 全てのノードにおいて，ノードから出るエッジの重みの和が 1 となる．出ていくエッジ数の比較を行うために，エッジの重みを，単語の出現回数をそのページで最も出現する単語の出現回数で割ることで与える．ノード i から j へのエッジが，i より j が簡単であることを表すとき，ノード i から j へのエッジの重み w は以下の式で表される．

$$w_1(i,j) := \frac{c(i,j)}{\max_k c(i,k)} \quad (5)$$

3.4 有向グラフからの難易度推定

この節では，3.3 節で作成した二単語間の難易度順推定を表す重み付き有向グラフから，全ノードの難易度順を推定する方法を提案する．

提案する手法は，*Relational Power* を用いたランキング手法を元にした方法である．有向グラフ D におけるノード i の Relational Power は，ノード $\{1,\cdots,n\}$ を持つ N 点の有向グラフの集合 $(D \in)\mathcal{D}$ に対して，関数 $f : \mathcal{D} \to \mathbb{R}^N$ を尺度としたときの $f(D)$ の i 番目の要素である．適切な関数 f を用いることで，ノード i から j へのエッジ (i,j) が，i が j に勝つことを表しているときに全体の中での勝者ほど大きな値が割り振られる．このような関数は様々なものが考案されている．今回は，二つの方法を元に提案を行う．一つは，Ryuo ら [11] の提案する方法である．有向グラフ D を，ノード集合 $N = \{1,2,\cdots,n\}$ とエッジ集合 E で表す．この時，E には自己ループすなわち $i(\in N)$ から i へのエッジを含まず，またノード $i,j(\in N)$ に対して i から j へのエッジ (i,j) と j から i へのエッジ (j,i) を同時に含まないものとする．ここで，エッジ集合 E での i の後継

ノードを

$$S_D(i) := \{j \in N | (i,j) \in D\} \tag{6}$$

で表し，その濃度を

$$s_D(i) := \kappa(S_D(i)) \tag{7}$$

で表す．同様に，先行ノードを

$$P_D(i) := \{j \in N | (j,i) \in D\} \tag{8}$$

で表し，その濃度を

$$p_D(i) := \kappa(P_D(i)) \tag{9}$$

で表す．Ryuo らは，

$$\alpha_i(D) := \sum_{j \in S_D(i)} \frac{s_D(j)+1}{p_D(j)} \ (\forall i \in N, \forall D \in \mathcal{D}) \tag{10}$$

という関数 $\alpha : \mathcal{D} \to \mathbb{R}^N$ を提案している．もう一つは，Borm ら [12] の提案する方法である．彼らは，

$$\beta_i(D) := \sum_{j \in S_D(i) \cup \{i\}} \frac{1}{p_D(j)+1} \ (\forall i \in N, \forall D \in \mathcal{D}) \tag{11}$$

という関数 $\beta : \mathcal{D} \to \mathbb{R}^N$ を提案している．

これらの関数を作成した有効グラフに適応することで得られる値を考える．単語 i から単語 j へのエッジを，i より j が簡単であるとする有向グラフをもとに考える．関数 α_i は，ノード i の出次数が大きいほど，多くの値の和をとる．また，$s_D(j) \geqq 0$, $p_D(j) \geqq 0$ より

$$v_{\alpha D}(i) := \frac{s_D(j)+1}{p_D(j)} \tag{12}$$

とすると，$\forall j \in N$ において $v_{\alpha D}(i) < 0$ であるので，関数 α_i は，ノード i の出次数が大きいほど大きな値をとる．つまり，多くの語を説明に使う場合に大きな値が算出される．また，$v_{\alpha D}(i)$ は，出次数が大きく，入次数が小さいほど，値が大きくなる．$v_{\alpha D}(i)$ は多くの語を説明に使い，他の語の説明に使われにくい語に対し，大きな値となる．つまり，関数 α_i は，難しい語は説明に多くの語を使い，かつ他の語の説明には使われにくく，また難しい語を説明に使う語はより難しい語である，という理論のもとで難しい語に大きな値を与える．関数 β_i も同様に，ノード i の出次数が大きいほど多くの値の和をとり，かつ，大きな値をとる．また，

$$v_{\beta D}(i) := \frac{1}{p_D(j)+1} \tag{13}$$

としたとき，$v_{\beta D}(i)$ は，入次数が小さいほど，値が大きくなるので，他の語の説明に使われにくい語に対し大きな値となる．つまり，関数 β_i も同様に，難しい語は説明に多くの語を使い，かつ他の語の説明には使われにくく，また難しい語を説明に使う語はより難しい語である，という理論のもとで難しい語に大きな値を与える．単語 i から単語 j へのエッジを，i より j が難しいとする有向グラフに関しても同様に言える．$\forall j \in N$ において $v_{\alpha D}(i) < 0$ より，関数 α_i は多くの語の説明に使われるほど大きな値となり，$v_{\alpha D}(i)$ は多くの語の説明に使われ，かつその単語の説明に使う語が少ない場合大きな値をとる．つまり，関数 α_i は，簡単な語は多くの語の説明に使われ，かつその語の説明には他の語使われにくく，また簡単な語の説明に使割れる語はより簡単な語である，という理論のもとで簡単な語に大きな値を与える．関数 β_i についても，多くの語の説明に使われるほど大きな値となり，かつ $v_{\beta D}(i)$ はその単語の説明に使う語が少ない場合大きな値をとるので，簡単な語は多くの語の説明に使われ，かつその語の説明には他の語使われにくく，また簡単な語の説明に使割れる語はより簡単な語である，という理論のもとで簡単な語に大きな値を与える．

これらの関数は，エッジの重みが全て等しいことを前提としている．本研究では重み付き有向グラフを扱えるように拡張した手法を提案する．まず，Σ によって値を加算する際に重みを与える．α, β いずれの関数も，全ての後続ノード，つまり全てのエッジを同等に扱っている．しかし，重み付きグラフを扱う際には，その重みに合わせてエッジを扱うべきである．よって，後続ノードごとに計算した値に，その後続ノードへのエッジの重みをかけた値の和を取る．また，$p_D(i), s_D(i)$ の計算においても重みを考慮する．Ryuo ら及び Borm らの方法では，$p_D(i)$ は入次数，$s_D(i)$ は出次数を表していた．これを，$p_D(i)$ は i へのエッジの重みの和，$s_D(i)$ は i から出るエッジの重みの和とし，重み付きの入次数，出次数を計算する．

これらをまとめ，以下のような手法を提案する．$S_D(i)$ を有向グラフ D における i の後継ノードの集合，$P_D(i)$ を i の先行ノードの集合，$w(i,j)$ を i から j へのエッジの重みとする．ただし $w(i,i)$ は 1 とする．ここで，以下の二つの値を定義する．

$$s'_D(i) := \sum_{j \in S_D(i)} w(i,j) \tag{14}$$

$$p'_D(i) := \sum_{j \in P_D(i)} w(j,i) \tag{15}$$

a） 手　法　α

関数 $\alpha' : \mathcal{D} \to \mathbb{R}^N$ を

$$\alpha'_i(D) := \sum_{j \in S_D(i)} w(i,j) \cdot \frac{s'_D(j)+1}{p'_D(j)} \ (\forall i \in N, \forall D \in \mathcal{D}) \tag{16}$$

とする．α'_i を，i 番目のノードの Relational Power として，その値で難易度を設定する．

b） 手　法　β

関数 $\beta' : \mathcal{D} \to \mathbb{R}^N$ を

$$\beta'_i(D) := \sum_{j \in S_D(i) \cup \{i\}} w(i,j) \cdot \frac{1}{p'_D(j)+1} \ (\forall i \in N, \forall D \in \mathcal{D}) \tag{17}$$

とする．β'_i を，i 番目のノードの Relational Power として，その値で難易度を設定する．

この拡張は，全てのエッジの重みが 1 のとき，元の定義と一致するものである．また，重みが 0 であることとそのエッジが存在しないことは同値である．

なお本研究ではこの他に PageRank を用いた手法も検討した

が，実験で良い性能が得られなかったため，PageRank を用いた手法についての説明は本稿では省略する．

4 実　　験

本章では，提案手法を用いて行なった実験の内容とその結果について述べる．本実験の目的は，提案手法の有効性を検証することである．実験のウィキペディアのページのページのデータを元に，単語の難易度を推定した．

4.1 使用したデータ

ウィキペディアは，コンテンツなどのデータをデータベース・ダンプとして提供している [2]．各ページのデータは，2021 年 1 月 7 日時点で最新の版として配布されているデータを使用した．

また，ウィキペディアのページにはカテゴリという概念が存在し，分野別に記事を参照することができる．本実験では，トピックによるページ数の偏りによる推定への影響を排除するために，カテゴリによるページのトピックの選択を試みた．本実験では，後述する比較データと対応づけたページが所属するカテゴリを対象として実験を行った．

4.2 比較したデータ

単語の難易度に対して，正解は存在しないと考えられる．ただし，単語の難易度があがり，理解するために要する知識が増えるほど，その単語を理解している人は減ると言える．よって本実験では，認知度を単語の難易度と考えることとし，認知度が低い単語ほど難易度が高く，認知度が高ければその単語の難易度は低いと考える．

ただし，実際には必ずしも認知度と単語の難易度の順序が一致するとは言えない．人気の分野・話題の分野などにおいては，それ以外の分野と比べてより難しい語まで知っている人が増えると考えられるからである．故に，様々な分野の単語の難易度を推定し認知度と比較するのであれば，分野の認知度も考慮して行う必要がある．本実験ではカテゴリを絞った実験を行うため，特定の分野の単語に限った推定となる．そのため，認知度と難易度は一致しているとみなす．

認知度のデータには，文部科学省と科学技術・学術政策研究所が実施した科学技術に関する国民意識調査 [3]で報告されている 2019 年 3 月のデータを使用した．この調査では認知度が男性と女性に分けて報告されている．本研究では男性女性を区別した推定は行わないため，男女比を 1:1 であるとして，男性の認知度と女性の認知度の平均値を男女合わせた際の認知度の推定値とした．また，調査では 14 の単語について報告されているが，カテゴリに情報技術及びその下位カテゴリに属するものを持つ 9 単語のみを利用した．なお，ウィキペディアのページのタイトルと一致しない単語については該当する単語を対応させている．詳細は表 1 の通りである．なお，ID は後述の表 3 における表示と一致している．

2：https://ja.wikipedia.org/wiki/Wikipedia:データベースダウンロード
3：https://www.nistep.go.jp/archives/40989

4.3 提 案 手 法

3.3 節で提案した方法で構成した重み付き有向グラフに対して，3.4 節で提案した方法により難易度の計算を行う．提案手法は，以下の組み合わせにより構成する．

a） 有向グラフのエッジ方向

グラフ I： 単語 i より単語 j が簡単であるという推定を i から j へのエッジで表した有向グラフ

グラフ II： 単語 i より単語 j が難しいという推定を i から j へのエッジで表した有向グラフ

b） グラフからの推定方法

方法 α： 関数 $\alpha' : \mathcal{D} \to \mathbb{R}^N$ を

$$\alpha'_i(D) := \sum_{j \in S_D(i)} w_1(i,j) \cdot \frac{s'_D(i)+1}{p'_D(i)} \ (\forall i \in N, \forall D \in \mathcal{D}) \tag{18}$$

とし，α'_i を，i 番目のノードの Relational Power として難易度順を推定する．

方法 β： 関数 $\beta' : \mathcal{D} \to \mathbb{R}^N$ を

$$\beta'_i(D) := \sum_{j \in S_D(i) \cup \{i\}} w(i,j) \cdot \frac{1}{p'_D(i)+1} \ (\forall i \in N, \forall D \in \mathcal{D}) \tag{19}$$

とし，β'_i を，i 番目のノードの Relational Power として難易度順を推定する．

これらを組み合わせ，表 2 の 4 通りにより難易度を推定する．

4.4 比 較 手 法

提案手法の有効性を示すため，以下の六つの比較手法に対しても実験を行う．

a） ベースライン手法 1

収集した文章データをもとに，単語の出現頻度を難易度とした．難易度を推定する単語を w とすると，推定難易度 $d_{B_1}(w)$ を式 (20) で計算した．この手法を Base1 とする．これは，値が大きいほど簡単な単語であると推定する．

$$d_{B_1}(w) = \frac{c_w}{max_{t \in T} c_t} \tag{20}$$

なお，c_w は収集した文章データ中に単語 w そのものが現れた回数，T は推定する全ての単語の集合を表す．

b） ベースライン手法 2

西原ら [1] の提案する手法による難易度推定を行う．単語 w に対して，難易度 $d_{B_2}(w)$ を式 (21) で与える．この手法を Base2 とする．これは，値が小さいほど簡単な単語であると推定する．

$$d_{B_2}(w) = \frac{ALLP - Pnum(w)}{ALLP} \tag{21}$$

ただし，$ALLP$ は Web ページの総数，$Pnum(w)$ は w を含む Web ページ数を表している．ウィキペディアでは，一つの単語が一つのページで説明されていることから，$ALLP$ は推定する単語の数，$Pnum(w)$ は説明に単語 w を含む単語の数を表すことになる．

c） 重みを利用しない手法

単語の重みを考慮することの有効性を図るため，提案手法か

表 1: 使用した単語とその認知度

単語 ID	調査における単語 (対応するページのタイトル)	男性認知度	女性認知度	推定認知度
A	クラウドサービス (クラウドコンピューティング)	54%	34%	44%
B	サイバーセキュリティ技術 (サイバーセキュリティ)	46%	28%	37%
C	サイバー空間 (サイバースペース)	42%	23%	32.5%
D	情報通信技術	40%	19%	29.5%
E	Internet of Thing(モノのインターネット)	40%	15%	27.5%
F	機械学習	24%	9%	16.5%
G	ヒューマンインターフェース技術 (マンマシンインタフェース)	24%	8%	16%
H	強化学習	16%	8%	12%
I	エッジコンピューティング	9%	3%	6%

表 2: 提案手法

手法	グラフ	推定方法	備考
I-α	I	α	小さな値を持つものが簡単と推定
II-α	II	α	大きな値を持つものが簡単と推定
I-β	I	β	小さな値を持つものが簡単と推定
II-β	II	β	大きな値を持つものが簡単と推定

ら重みを考慮した部分を除いた難易度推定を二通り行なった．一つは Ryuo ら [11] による Relational Power の計算方法を用いる手法である．

$$\alpha_i(D) := \sum_{j \in S_D(i)} \frac{s_D(i)+1}{p_D(i)} \ (\forall i \in N, \forall D \in \mathcal{D}) \qquad (22)$$

という関数 $\alpha : \mathcal{D} \to \mathbb{R}^N$ を用いて，α_i を i 番目のノードの Relational Power として難易度順を推定する．この手法を，4.3 節で述べたグラフ I に適応した手法を W-I-α，グラフ II に適応した手法を W-II-α とする．もう一つは Borm ら [12] による Relational Power の計算方法を用いる手法である．

$$\beta_i(D) := \sum_{j \in S_D(i) \cup \{i\}} \frac{1}{p_D(i)+1} \ (\forall i \in N, \forall D \in \mathcal{D}) \quad (23)$$

という関数 $\beta : \mathcal{D} \to \mathbb{R}^N$ を用いて，β_i を，i 番目のノードの Relational Power として難易度順を推定する．この手法を，4.3 節で述べたグラフ I に適応した手法を W-I-β，グラフ II に適応した手法を W-II-β とする．

いずれも，グラフ I に適応した手法では値が小さいほど簡単な単語であると推定し，グラフ II に適応した手法では値が大きいほど簡単な単語であると推定する．

4.5 評価指標

各手法で推定された難易度の順序について，4.2 節で述べた認知度のデータでの順序との Spearman の順位相関係数 [5] と Kendall の順位相関係数 [6] を求め，比較した．Spearman の順位相関係数 ρ は，以下の式 (24) で求められる．

$$\rho = 1 - \frac{6\Sigma D^2}{N^3 - N} \qquad (24)$$

ここで，D は各値の順位の差，つまりそれぞれの単語の認知度の順位と推定難易度の順位の差であり，N は値のペアの数である．

Kendall の順位相関係数 τ は，ランキング内に同順位のものが無いとき，以下の式 (25) で求められる．

$$\tau = \frac{P-Q}{n(n-1)/2} \qquad (25)$$

ここで，n はデータの数であり，P はそのうち二つを選んだときに二種類のランキングでの順序関係が一致している数，Q は二種類のランキングでの順序関係が一致していない数を表す．ランキング内に同順位があるとき，以下の式 (26) で求められる．

$$\tau = \frac{P-Q}{\sqrt{n-T_x}\sqrt{n-T_y}} \qquad (26)$$

ここで，T_x はを一つ目のランキングに同順位が $t_1, \cdots, t_k$ 個あるとして

$$T_x = \sum_{i=1}^{k} \frac{t_i(t_i-1)}{2} \qquad (27)$$

であり，T_y はを二つ目のランキングに同順位が $u_1, \cdots, u_l$ 個あるとして

$$T_y = \sum_{j=1}^{j} \frac{u_j(u_j-1)}{2} \qquad (28)$$

である．

本研究及び本実験では，順位が上位であるか下位であるかによって順位の重みが変わることはない．よって，順位相関係数による評価を行うことができる．

4.6 実験結果

各手法による難易度推定の結果は次の表 3 のようになった．推定順位のアルファベットは，表 1 の ID である．認知度の行は，科学技術に関する国民意識調査における認知度による順位を表している．ρ は Spearman の順位相関係数の値を，τ は Kendall の順位相関係数の値を示す．

5 考　　察

本研究で提案した手法の有効性について考察する．

5.1 Relational Power を用いた手法に関する考察

Relational Power を用いる難易度推定の有効性について検討するため，提案手法の I-α から II-β の四つの手法と，ベースライン手法 Base1，Base2 を比較し考察する．

表 3: 実験結果

手法	推定順位									ρ	τ
	A	B	C	D	E	F	G	H	I		
認知度	1	2	3	4	5	6	7	8	9	-	-
Base1	1	5	6	4	3	2	8	7	8	0.658	0.423
Base2	1	6	5	7	3	2	8	4	9	0.450	0.333
W-I-α	7	8	4	6	9	5	1	3	2	-0.600	-0.444
W-I-β	7	9	3	5	6	8	2	4	1	-0.567	-0.389
W-II-α	1	6	5	7	3	2	8	4	9	0.450	0.333
W-II-β	1	6	4	7	3	2	8	5	9	0.533	0.389
I-α	4	6	3	9	8	5	2	7	1	-0.250	-0.222
II-α	1	6	4	7	3	2	8	5	9	0.533	0.389
I-β	1	5	4	6	2	3	8	7	9	0.717	0.556
II-β	1	5	4	6	3	2	8	7	9	0.700	0.500

5.1.1 Relational Power を用いた手法の有効性と関数の違いについて

まず，両ベースライン手法よりも認知度の順位との相関が，いずれの順位相関係数においても高かった手法は，I-β，II-β の二つである．これは式 (17) で示した関数 β' を使った手法全てであり，これにより，関数 β' は有効であると考えられる．II-α は，Base2 より認知度の順位との相関が，いずれの順位相関係数においても高かったものの，Base1 より低かった．I-α は，認知度の順位との相関がいずれの順位相関係数においてもあまり見られなかった．このため，式 (16) で示した関数 α' は，関数 β' を使った手法と比べ有効であるとは言えないだろう．

関数 α' と関数 β' における大きな違いに，Σ 関数内での値の計算の分母に (重み付き) 出次数を含むか否かがある．本実験結果では，出次数を含まない方が高い性能を示している．関数 α'_i は以下の式で与えられる．

$$\alpha'_i(D) := \sum_{j \in S_D(i)} w(i,j) \cdot \frac{s'_D(i)+1}{p'_D(i)} \ (\forall i \in N, \forall D \in \mathcal{D}) \tag{16}$$

Σ 関数内の値は，難しい単語ほど難しい単語を説明に使用し，簡単な単語の説明に使用される単語は簡単であると考えられるため，グラフ I においては難しいほど高い値に，グラフ II においては簡単なほど高い値になることが期待される．

図 1 は，科学技術に関する国民意識調査で認知度が報告されているもののうち，今回の実験で使用した 9 単語の，認知度の順位とその単語が使用している単語の数・その単語を使用するページの数の関係を表したものである．ここで，

$$v_{\alpha D}(i) := \frac{s_D(j)+1}{p_D(j)} \tag{12}$$

として，科学技術に関する国民意識調査で認知度が報告されているもののうち，今回の実験で使用した 9 単語の順位とグラフ I における v_α の値の関係を図 1a で表した．

これより，v_α の値と認知度による順位との間の相関は弱いとわかる．故に，関数 α' による難易度順推定は精度が低いものとなったと考えられる．

5.1.2 使用したグラフの違いについて

本研究では，難易度順推定を表す有向グラフについて，エッジの向きが異なる二種類を提案した．Relational Power を用いた難易度推定にその差が及ぼす影響について考察する．

エッジの向き以外の条件が同じものに対して比較を行う．つまり，I-β と II-β，I-α と II-α の 2 組の比較を行う．どちらの組も，グラフ II を用いた推定の方が二種類の順位相関係数で精度が良かった．これより，グラフ II，つまりより難しいと推定される単語に向けてエッジを張ったグラフを用いた方が良いと言える．

これは，二種類の関数がどちらも出次数が多いほど多くの値の和をとることになるためであると考えられる．グラフ I における出次数は，そのノードの表す単語のページに出現する単語の種類を表している．単語のページに出現する単語の種類は，図 1b から，認知度が高いほど多くなる傾向があると言える．つまり，認知度が高いほど，多くの値の和をとることになってしまうと考えられる．グラフ I では，難しい単語ほど大きな値が与えられることを期待しているが，Σ 関数内の値の大小よりも多くの値の和をとることの影響が大きくなった結果，難易度推定の精度が低くなってしまったと考えられる．一方グラフ II における出次数は，そのノードの表す単語が出現するページの数を表しており，これは図 1c より認知度との相関が低いと言える．相関があると見ても，認知度が高いほど多くなる傾向であり，認知度が高いほど多くの値の和をとることは，簡単な単語ほど大きな値が与えようとしていることと一致する．

5.2 重みの利用の有効性の考察

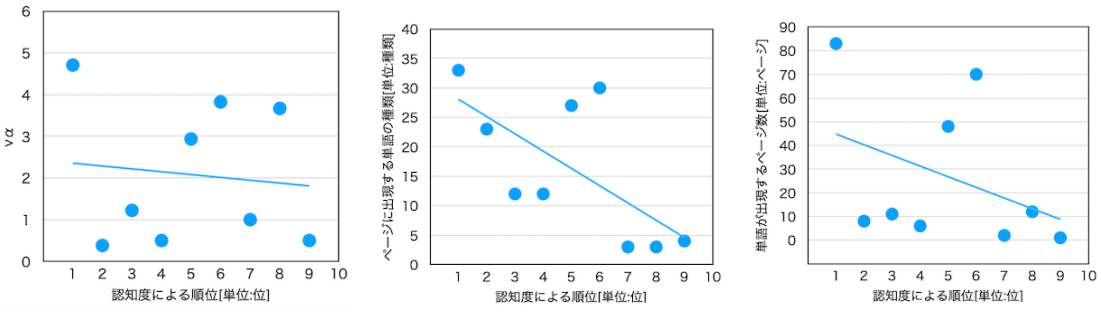
エッジに重みを与えたことの有効性について検討するため，提案手法の I-α から II-β の四つの手法と，エッジに重みを与えずに Relational Powe の計算を行なった比較手法 W-I-α，W-I-β，W-II-α，W-II-β の四つを比較し，考察する．

まず，重みを利用したものと，重みを与えていないものを比較し，重みを与えることについての有効性を検討する．Relational Power の計算に使用した関数と使用したグラフが同じであるもの同士を比較する．つまり，I-β と W-I-β，II-β と W-II-β，I-α と W-I-α，II-α と W-II-α の計 4 組の比較を行う．いずれの組においても，重みを与えた手法が，重みを与えていない手法よりも双方の順位相関係数において高い精度で難易度を予測した．よって，重みを与えることは有効であったと言える．

6 結　　論

本研究では，難易度の与えられていないデータであるウィキペディアを用いて単語の難易度を推定することを目的に取り組んだ．ウィキペディアのタイトルを単語と考え各ページをその単語の説明とみなして，説明にどの単語が出るかという情報をもとに，単語の難易度を推定する手法を提案した．

まず，説明にどの単語が出るかという情報から，自己ループや相反する向きに張られた並行なエッジの無い重み付き有向グラフの作成を行なった．ここでは，二つの単語に対し双方のペー



(a) グラフ I における t の値　(b) 説明に使用している単語の数　(c) 単語を使用するページの数

図 1: 認知度の順位との関係

ジへの出現回数を比較し，出現回数が多かった方が簡単であると考えて，より簡単だと推定された方へ向けてエッジを張ったグラフとより難しいと推定された方へ向けてエッジを張ったグラフの二種類を作成した．さらに，各ページ内での出現割合や，並行なエッジの出現回数との差を，エッジの重みとして与えた．

そして，作成した重み付き有向グラフをもとに，Relational Power を計算して全体の難易度順を推定する手法を提案した．Relational Power の計算に関しては，二つの既存手法それぞれに対して重みを考慮する方法を提案した．これらの手法の有効性を評価する実験を行った結果，より難しいと推定された方へ向けてエッジを張ったグラフをもとに Relational Power を計算して難易度順をつける方法の一方が高い精度を示した．また，重みを与えることの有効性も示された．

今後の課題としては，以下の三つが挙げられる．

- トピックの認知度や人気の影響
- 難易度の数値化
- 技術記事の難易度推定

まず，単語そのものの難易度推定について，本研究での提案手法から更なる検討の余地がある．

一つは，トピックの認知度や人気の影響についてである．本研究では同じトピックの範囲での実験においては推定難易度の順序と認知度の順序が一致すると考えたが，トピックが異なる単語が混ざることによりこれは成り立たなくなると考えられる．なせなら，メジャーなトピックでは難易度が高い語でも知っている人が多く，マイナーなトピックにおいては難易度が低い語でも知っている人が少なくなってしまう可能性があるからである．これらは，トピックの認知度を算出し，難易度の推定に用いることで対応できると考えられる．

また，本研究では難易度を数値化し比較するのではなく，計算で得られた数値から必要な単語間での難易度順を決定し，その順序を比較した．四つ目の課題ともつながるが，実際に技術記事の難易度推定を行うためには，単語を比較した際に難しいかどうかだけでなくどのくらい難しいかがわかっていることが望ましい．そのため，難易度の数値化も大きな課題である．

最後に，今回提案した単語の難易度推定手法をもとに，技術記事の難易度推定を行うことが挙げられる．文章の難易度推定は西原ら [1] やの手法や川村ら [3] の手法など，様々な方法が考案されている．しかし，技術記事においては，記事の中で専門用語の出現とともにその語についての説明がなされている場合も多いため，その点も考慮し，技術記事の特性に合わせた難易度推定を行う必要があるだろう．

謝　　辞

本研究は，JST CREST (JPMJCR16E3)，JSPS 科研費 18H03245 の支援を受けたものである．

文　　献

[1] 西原陽子, 砂山渡, 谷内田正彦: Web ページの難易度と学習順序に基づく情報理解支援システム, 電子情報通信学会論文誌, Vol. J89–D, No. 9, pp. 1963–1975 (2006).

[2] 砂山渡, 谷内田正彦: 観点に基づいて重要文を抽出する展望台システムとそのサーチエンジンへの実装, J 人工知能学会論文誌, Vol. 17, No. 1, pp. 14–22 (2002).

[3] 川村よし子, 北村達也: 日本語学習者のための文章の難易度判定システムの構築と運用実験, *Journal CAJLE*, Vol. 14, pp. 18–30 (2013).

[4] 近藤陽介, 松吉俊, 佐藤理史: 教科書コーパスを用いた日本語テキストの難易度推定, 言語処理学会第 14 回年次大会発表論文集, pp. 1113–1116 (2008).

[5] Spearman, C.: The Proof and Measurement of Association between Two Things, *The American Journal of Psychology*, Vol. 15, No. 1, pp. 72–101 (1904).

[6] Kendall, M. G.: A New Measure of Rank Correlation, *Biometrika*, Vol. 30, No. 1/2, pp. 81–93 (1938).

[7] Collins-Thompson, K. and Callan, J.: Predicting Reading Difficulty with Statistical Language Models, *Journal of the American Society for Information Science and Technology*, Vol. 55, No. 13, pp. 1448–1462 (2005).

[8] 水谷勇介, 河原大輔, 黒橋禎夫: 日本語単語の難易度推定の試み, 言語処理学会第 25 回年次大会発表論文集, pp. 670–673 (2018).

[9] 藏培慶, 小林伸行, 椎名広光: 単語難易度推定による中日単語学習システム, 言語処理学会第 20 回年次大会発表論文集, pp. 2–11 (2014).

[10] 櫻庭 セルソ智, 睦憲柳浦: 線形順序付け問題の解法, オペレーションズ・リサーチ = Communications of the Operations Research Society of Japan : 経営の科学, Vol. 57, No. 6, pp. 327–334 (2012).

[11] Ryuo, S. and Yamamoto, Y.: RANKING BY RELATIONAL POWER BASED ON DIGRAPHS, *Journal of the Operations Research Society of Japan*, Vol. 52, No. 3, pp. 245–262 (2009).

[12] Borm, P., van den Brink, J. and Slikker, M.: *An Iterative Procedure for Evaluating Digraph Competitions*, FEW Research Memorandum, Vol. 788, Microeconomics (2000).