webコンテンツ群を対象とした難易度による対話式検索インタフェイスの実現方式

外崎 未空　石井 雄大　中西 崇文

武蔵野大学データサイエンス学部 〒135-8181 東京都江東区有明 3-3-3

E-mail: {s2222055, s2022052}@stu.musashino-u.ac.jp, takafumi.nakanishi@ds.musashino-u.ac.jp

あらまし　本稿では，web コンテンツ群を対象とした難易度による対話式検索インタフェイスの実現方式について示す．現在，インタネット上には様々な話題で様々な難易度の web コンテンツが散在している．これらのコンテンツから，ユーザのニーズに合致したコンテンツを選択するために，既存の検索エンジンが使用される．だが，ユーザのニーズと合致しない web コンテンツが上位に提案されることがある．本方式では，web コンテンツの難易度として，その web コンテンツに含まれる画像数，その web コンテンツのひらがなの割合，および，日本語の語彙レベルから定量化する．本方式では，その難易度の指定をユーザに対話式で実現することにより，ユーザのニーズに合致した web コンテンツを手軽に探索することが可能となる．

キーワード　対話型検索，難易度，web コンテンツ，web 検索システム

1. はじめに

現在，インタネット上には膨大な量で，様々な話題，難易度をもつコンテンツが散在している．それらのコンテンツを対象として，ユーザのニーズに合致したコンテンツを選ぶために，既存の検索エンジンが用いられる．本研究において，ユーザのニーズとは，ユーザが検索する際に求めている話題，難易度を指すこととする．

これまでの既存の検索エンジンでは，キーワードを指定することが多く，その内容に関して順位をつけて表示することが可能であった．

だが，これまでの既存の検索エンジンでは，下記の問題があった．

(1) ユーザが与えた検索キーワードに合致するコンテンツ群を対象として，どのような話題が存在するかを俯瞰的に理解すること

(2) ユーザの知識に合致した難易度をもつコンテンツ群を探索すること

そこで本研究では，話題と難易度の 2 つの観点から web コンテンツを整理し，ユーザのニーズと合致したコンテンツの探索を手軽に行えるようにする．本研究において，話題とは，web コンテンツが取り扱うメインの事柄と定義し，難易度とは，ひらがなの割合と画像スコア，および日本語のレベルによって決定づけられるものと定義する．

(1)については，ユーザが与えるキーワードに対して，既存の検索エンジンで取得されたトップ n 個のコンテンツ群を対象として，その出現する単語から話題を推定し，似た話題でクラスタリングを行うことにより，キーワードから推定される話題ごとにコンテンツを整理し，ユーザに提示することで，ユーザのニーズに合致したコンテンツを手軽に探索する機能を実現する．

(2)については，各コンテンツを構成する特徴から，難易度スコアを付与し，これによって整理することで，ユーザの知識に合致したコンテンツを手軽に探索する機能を実現する．

本稿では，日本語の web コンテンツを対象として，web コンテンツの話題・難易度を推定し，対話式でユーザに提案する検索インタフェイスの実現方式について示す．

本方式は，以下の手順で構成される．はじめに，ユーザを任意の検索ワードを入力とし，web コンテンツを取得する．次に，取得した web コンテンツを話題クラスタリング機能と難易度ソート機能により，話題ごとにクラスタリングし，難易度順にソートする．最後に，ソートした web コンテンツを対話機能により，ユーザに提案する．この提案した web コンテンツに対してユーザからフィードバックを受け取り，再提案を行う．このシステムは，システムが提案した web コンテンツに対して，ユーザが適当であると答えるか，すべての web コンテンツを提案すると終了する．

本稿は，次の構成からなる．2 節では，本方式の関連研究について紹介する．3 節では，日本語の web コンテンツを対象として，web コンテンツの難易度を定量化し，対話式で提示する検索インタフェイスの実現方式の詳細を述べる．4 節では，本方式を実現する実験システムを構築した上で行った実験について述べる．5 節では，本稿をまとめる．

2. 関連研究

本節では，提案方式に関連する研究について述べ

る．

2.1 節では，web 推薦システムに関する研究について示す．2.2 節では，難易度の定量化に関する研究について示す．2.3 節では，対話型検索に関する研究について示す．2.4 節では，本研究の位置付けについて示す．

2.1. web 推薦システムに関する研究

従来の web 推薦システムに関する研究として、高須賀ら[1]や永井[2]の研究が挙げられる．

高須賀ら[1]は，自動収集した閲覧履歴における URL を用いた協調フィルタリングによって，web ページの推薦を行った．これは閲覧履歴を用いた web 推薦システムの事例である．しかし，閲覧履歴を用いた web 推薦システムでは，今までとは全く違う知識レベルの分野について知りたい場合に十分でないことがある．

永井[2]は，一連の検索利用内において閲覧した web コンテンツ内のキーワードを利用して、ユーザに適した web サイトを提案する．これは，一連の検索ごとに独立して web サイトを推薦する事例である．

本研究では，web サイトを推薦する際にその web コンテンツの難しさにも考慮するため，web サイトの難易度に関する情報を抽出する必要がある．

本研究では，web サイトから抽出した情報をもとに，ユーザに web サイトを提案し，ユーザとの対話を繰り返すことで最適な web サイトを提案するシステムを作成した．

2.2. 難易度の定量化に関する研究

難易度の定量化に関する研究として、建石ら[3]や近藤ら[4]，砂川ら[5]の研究が挙げられる．

建石ら[3]は，日本語の文の表面的な難易度を，文の平均の長さ，各文字種類(ひらがな，カタカナ漢字等)の相対頻度，文字種ごとの平均の長さ，読点に対する句点の比の 4 つの観点から，テキストの難易度を求めた．この研究では，この 4 つの観点それぞれが関係しあっており，特定の観点のみを考慮しても読みやすくなるとは限らないと述べている．

近藤ら[4]は，テキストの文字からテキストの難易度を学年区分に分類する手法を構築した．学年区分とは，小学 1 年から高校 3 年に大学を加えた 13 区分のことを指す．これは，あらかじめ学習した言語モデルを利用してテキストの難易度を分類する事例である．

砂川ら[5]は，日本語学習辞書支援グループ (2015)「日本語教育語彙表 Ver 1.0」を作成した．日本語教育語彙表とは，約 18000 語の日本語を 6 段階の難易度をはじめとし，読み・品詞・語種などの情報が搭載されているデータベースである．本研究では，この日本語教育語彙表に記載されている情報のうち，見出し語と難易度のみを利用した．

提案方式では，web コンテンツの内容だけでなく，web サイトの外見的観点がユーザに与える印象にも考慮するため，外見的観点からの項目も加え，難易度の定量化を行う必要がある．よって，web コンテンツ内のひらがなの割合と画像の枚数から難易度を定量化する．

2.3. 対話型検索に関する研究

対話型検索に関する研究として、波多野ら[6]や吉田ら[7]の研究が挙げられる．

波多野ら[6]は，収集した web サイトを自己組織化マップ[8]を用いて自動分類し、この結果に対して、ユーザと対話的な操作を施すことで、分類結果を段階的に修正することが可能となる機構の開発を行った．自己分類マップ[8]とは，T. Kohonen によって提案された教師なしニューラルネットの一種であり，次元削減や可視化に用いられる．特徴の似たデータ同士は近い位置に配置されるという特徴を持つ．自己組織化マップには，web ページから抽出した単語をもとにベクトル化した特徴ベクトルを入力とする．

吉田ら[7]は，AND 検索，OR 検索，NOT 検索を意識せずに行うことが可能となるシステムの構築．検索結果中の重要語に注目し、その後の出現傾向の可視化を利用して、ユーザとの対話を行い動的にクエリ修正と再ランキングを行う．

これらの研究では，ユーザの情報を考慮することで分類結果やクエリを修正するが，これでは最終的にユーザに適した web サイトを一つに絞ることが出来ない．本研究では，ユーザの目的に近い web サイトではなく，クエリから取得した web サイトを全体で取得し，ユーザとの対話で絞っていき，最終的に 1 つの web サイトをユーザに提案する．

2.4. 本研究の位置づけ

本研究では，ユーザからの検索ワードから取得した web コンテンツを話題と難易度によって整理し，ユーザのニーズに合致する web コンテンツを提案する web コンテンツ推薦方式を実現する．

2.1 節では，web サイト推薦システムに関する研究として，閲覧履歴を用いた事例を挙げた．その上で，本稿では，今後の展望として，閲覧履歴を用いることも視野に入れたいと考えている．

2.2 節では，難易度を定量化する研究について取り上げた．本研究では，日本語教育語彙表を利用して，使用語彙の難易度に加え，ひらがなの割合と画像数を

利用する．

2.3 節では，対話式検索に関係する研究について，ユーザとの対話により，web サイトの分類や，クエリを更新する例を挙げた．本方式では，検索者の学習効率向上のため，最終的にユーザのニーズに最も合致する web コンテンツを提案する．

3. 提案方式

本節では，提案方式である Web コンテンツ群を対象とした難易度による対話式検索インタフェイスの実現方式について提示する．

本方式は，ユーザからの検索ワードを入力として，取得した web コンテンツを内容別に分け難易度順にソートすることで，ユーザに適切な web サイトを出力する．

3 節の構成について述べる．3.1 節では，本方式の全体像について述べる．3.2 節では，web コンテンツ取得機能について述べる． 3.3 節では，話題クラスタリング機能について述べる．3.4 節では，難易度ソート機能について述べる．3.5 節では，対話機能について述べる．

3.1. 全体像

本節では，本研究における提案手法の概要を述べる．提案方式の全体像を図 1 に示す．

本システムの入力は検索ワードで、出力は web コンテンツである．まず，検索ワードを入力し，検索ワードに関連する web コンテンツ群を取得する．次に取得した各 web コンテンツをベクトル化し，これを元にクラスタリングを行う．さらに，各 web コンテンツからひらがなの割合，画像スコア，日本語のレベルを求め，それらを利用して難易度スコアを算出，付与し，難易度スコアを元にソートする．その後，任意のクラスタにおいて，難易度スコアが中央値である web コンテンツをユーザに出力する．この時出力した web コンテンツに対し，ユーザは話題・難易度の観点からフィードバックを入力し，システムはその入力に応じた別の web コンテンツを出力する．これを繰り返すことで最終的にユーザのニーズと合致した web コンテンツを出力する．

本システムは，話題クラスタリング機能，難易度ソート機能，対話機能で構成される．

話題クラスタリング機能は，web コンテンツ内のテキストを TF-IDF[10]によりベクトル化し，これを用いて web コンテンツをクラスタリングする機能である．この機能は，単語のベクトル化機能，クラスタリング機能によって構成される．

難易度ソート機能は，web コンテンツの難易度をスコア化し，そのスコアによって web コンテンツを

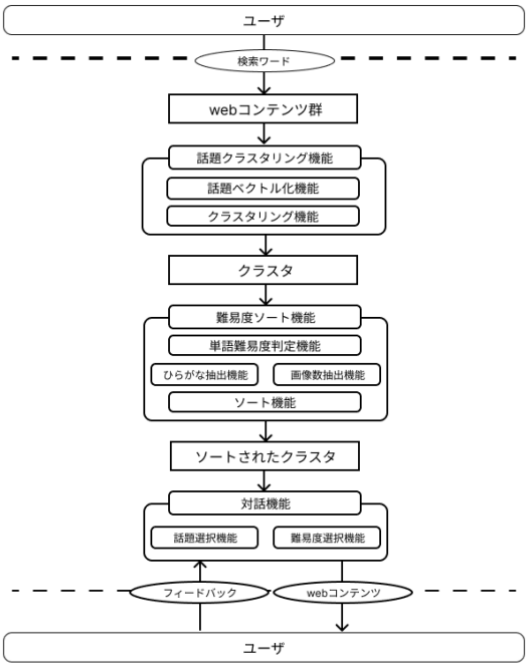

図 1　提案方式の全体像

ソートする機能である．難易度ソート機能は，単語難易度判定機能，ひらがなの割合算出機能，画像スコア算出機能，ソート機能によって構成される．

対話機能は，提案した web コンテンツに対してユーザからのフィードバックを受け取り，web コンテンツを選択，出力する機能である．対話機能は，話題選択機能，難易度選択機能によって構成される．

3.2. 話題クラスタリング機能

話題クラスタリング機能とは，web コンテンツをベクトル化し，これをもとにクラスタリングする機能である．

この機能は，取得した web コンテンツ内のテキストに出現する単語のうち，名詞に対して TF-IDF を適用する．各 web コンテンツをベクトル化し，その結果を用いてクラスタリングを行う．この機能は，単語のベクトル化機能とクラスタリング機能によって構成される．

3.3. 話題ベクトル化機能

話題ベクトル化機能とは，web コンテンツ内のテキストの名詞に着目し，TF-IDF を用いてベクトル化する機能である．この機能では，web スクレイピングにより取得した各 web コンテンツのテキストを形態素解析器である Mecab[10]を用いて形態素解析し，単語の品詞が名詞である単語のみを抽出する．形態素解析の辞書には Neolog-D[11]を使用する．このとき

抽出した名詞を TF-IDF によりベクトル化し，このベクトルを web コンテンツの話題を表す特徴量として扱う．

3.3.1. クラスタリング機能

クラスタリング機能とは，web コンテンツ群をクラスタリングする機能である．話題ベクトル化機能により出力された各 web コンテンツの話題を含有するベクトルを入力として，K-means 法[13]を用いてクラスタ数を 5 とするクラスタリングを行う．

この機能によって，話題の観点で，ユーザのニーズに合致した web コンテンツを手軽に探索することが可能になる．

このとき利用したクラスタ数 k = 5 は，以下の予備実験により決定した．予備実験では，検索ワードに関連する 30 個の web コンテンツを取得し，話題ベクトル化機能と同じ手法でベクトル化を行った．このベクトルに対して，k-means 法を用いてクラスタリングを行った．クラスタ数の評価には，GAP 統計量を用いた．GAP 統計量とは，異なる数のクラスタ数に対して，クラスタ内のデータのばらつきがどの程度小さく，かつクラスタ間のデータのばらつきがどの程度大きいかを比較し，最適なクラスタ数を推定するものである．この実験を，2≦k≦5 の範囲で，検索ワードを変化させ 4 回行った．その結果，クラスタ数を 5 に決定した．

3.4. 難易度ソート機能

難易度ソート機能とは，日本語のレベル，ひらがなの割合，および画像スコアから web コンテンツの難易度をスコア化し，その値によってクラスタ内の web コンテンツをソートする機能である．この機能では，web コンテンツがもつ難易度を 3 つの要素から定量化することが可能である．

日本語のレベルについては，日本語教育語彙表[5]の 6 段階の難易度のうち初級前半・初級後半に含まれる単語を利用する．ひらがなの割合については，web コンテンツに含まれる全文字数とひらがなの文字数を利用する．画像スコアについては，web コンテンツに含まれる img タグの数を利用する．

日本語のレベルを c，ひらがなの割合を y，画像スコアを s とすると，難易度スコア S は次のように表せる．

$$S = y + s + c$$

難易度スコアを昇順にソートし，小さい方から難易度が高いものとして扱う．

この機能は，単語難易度判定機能，ひらがなの割合算出機能，および画像スコア算出機能によって構成される．

この機能によって，web コンテンツの難易度を 3 つの要素から解釈することが可能になる．

3.4.1. 単語難易度判定機能

単語難易度判定機能とは， web コンテンツに含まれる単語のうち初級レベルの単語をカウントする機能である．

この機能では，砂川ら[5]が作成した，日本語教育語彙表(以後，語彙表と呼ぶ)を利用する．この語彙表に含まれる見出し語に対して，品詞と難易度による絞り込みを行う． 品詞は，名詞と動詞(動詞 1 類・2 類・3 類)を利用する．難易度については，語彙表が見出し語に対して付与している難易度である，初級前半・初級後半・中級前半・中級後半・上級前半・上級後半の 6 つのレベルのうち，初級前半・初級後半が付与されている見出し後を利用する．

ここで抽出した見出し後の単語と web コンテンツ内テキストを形態素解析することで抽出する名詞・動詞の一致する数をカウントし，この数を日本語のレベルとして扱う．これは，値が大きいほうがより簡単な日本語であるということを表す．

3.4.2. ひらがなの割合算出機能

ひらがなの割合算出機能とは，web コンテンツ内テキストのひらがながテキスト全体に対して占める割合を求める機能である．

この機能では，web コンテンツのテキスト全体の文字数とひらがなの文字数を抽出する．ひらがなの文字数を全体の文字数で割ることで web コンテンツのひらがなの割合を算出する．web コンテンツ内テキストの全文字数を a，web コンテンツ内テキストのひらがなの文字数を b とすると，ひらがなの割合 y は次のように表せる．

$$y = \frac{b}{a}$$

web コンテンツによって文字数に大きな差が見られることがあるが，割合を利用することでその影響の減少が可能となる．

3.4.3. 画像スコア算出機能

画像スコア算出機能とは，web コンテンツ内の画像の数を利用し，画像スコアを算出する機能である．web スクレイピングにより web コンテンツ内の img タグを取得し，この数をカウントする．

このカウントした画像数を，底が 2 の対数関数により変換する．web コンテンツ内の img タグの数を x(ここでは，x>0 を前提としている)とすると，画像スコア s は次のように表せる．

$$s = log_2 x$$

画像スコアを算出する際，web コンテンツ内に含まれる画像数は，増加しすぎると意味をなさない画像が増えると仮定する．そこで，$x(x>0)$が大きくなるにつれ，s の増加率が小さくなるという特徴を持つ対数関数を利用する．また，底に関しては，底が大きいと x の違いにおける s の値同士の差が生じにくくなるため，今回は 2 を底に用いる．

3.4.4. ソート機能

ソート機能とは，難易度スコアを元に web コンテンツをソートする機能である．難易度スコアとは，単語難易度判定機能，ひらがなの割合抽出機能，および画像数抽出機能の出力の各スコアの合計のことである．ひらがなの割合を y，画像スコアを s，日本語のレベルを c とすると，難易度スコア S は次のように表せる．

$$S = y + s + c$$

この難易度スコアは web コンテンツ 1 つずつに付与する．各クラスタ内で， web コンテンツを難易度スコア順に昇順にソートする．

この機能によって，難易度の観点で，ユーザのニーズに合致した web コンテンツを手軽に探索することが可能になる．

3.5. 対話機能

対話機能とは，出力に対するユーザのフィードバックによって，よりユーザのニーズに合致した web コンテンツを再度出力するという手順を繰り返すことで最終的な出力を決定する機能である．

対話機能のイメージ図を図 2 に示す．この機能は，内容・難易度の 2 つの観点からのフィードバックによって構成される．これによりユーザのニーズと合致した web コンテンツの提案が可能となる．

話題についてのフィードバックはクラスタ間の変更に使用し，難易度についてのフィードバックはクラスタ内の web コンテンツの変更に使用する．

3.5.1. 話題選択機能

話題選択機能とは，ユーザから話題の観点からのフィードバックを受け取り，クラスタを再選択する機能である．web コンテンツの話題に関するフィードバックは，話題がクラスタごとにクラスタリングされ

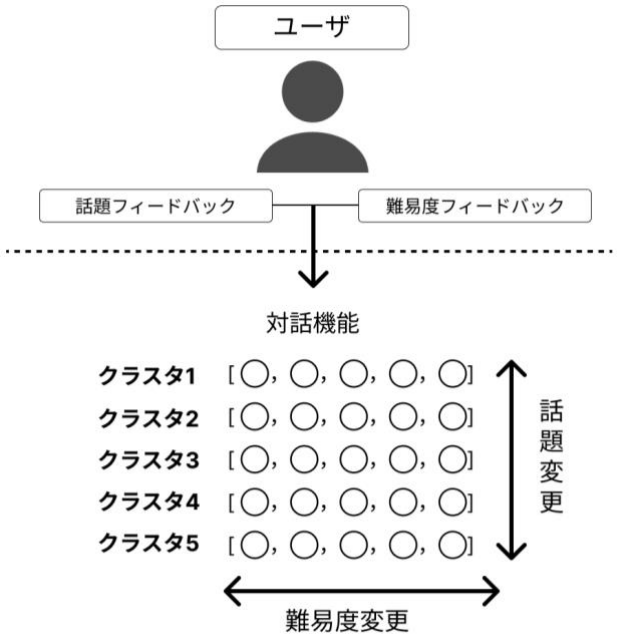

図 2　対話機能のイメージ

ている点を利用して，クラスタを変更することで対応する．

最初の提案では，ランダムにクラスタが選択される．この選択と，後述する難易度選択機能によって選択された web コンテンツを出力する．この出力された web コンテンツに対して，ユーザは「求めている話題であるか否か」をフィードバックとして入力する．「求めている話題でない」というフィードバックの場合には，その時点に選択されていないクラスタをランダムに選択する．全てのクラスタを出力した場合には，それまでの選択をリセットし，すべてのクラスタからランダムに選ばれる状態になる．「求めている話題である」というフィードバックの場合には，クラスタの変更は行わず，今回の提案で選択したクラスタを再び選択する．

3.5.2. 難易度選択機能

難易度選択機能とは，ユーザによる難易度の観点からフィードバックを受け取り，web コンテンツを選択・提案する機能である．この機能では，各クラスタ内で難易度が高い順にソートされていることを利用して，難易度の選択を行う．ユーザは現在閲覧した web コンテンツの難易度に関して「難しい」,「ちょうどよい」,「易しい」の 3 つの選択肢からフィードバックを選択することができる．難しいとフィードバックを返す場合，次に提案される web コンテンツは今提案された web コンテンツよりも難易度スコアが低いものとなる．

表 1　実行環境

ライブラリ	バージョン
Beautifulsoup4	4.11.2
Google-api-python-client	2.70.0
janome	0.4.2
numpy	1.22.4
pandas	1.4.4
Requests	2.27.1
Scikit-learn	1.2.3
Seaborn	0.12.2

この難易度選択機能には，ユーザの求める難易度を持つ web コンテンツが各クラスタに 1 つ以上存在すると仮定し，二分探索アルゴリズムを用いる．

最初の提案は，難易度スコア順にソートされたクラスタ内で難易度スコアが中央値である web コンテンツを提案する．提案した web コンテンツに対して，ユーザからのフィードバックが「難しい」,「易しい」を選択された場合は，中央のインデックスよりそれぞれ右，左を削除し，残った web コンテンツの中で難易度スコアが中央値を持つものを提案する．これを繰り返し，ユーザが「ちょうどよい」を選択する，または，リストの要素が 2 つになるまで続く．

表 2　実験 1 の結果 1

(1) Python 自然言語処理	
クラスタ	**タグ付け**
0	日本語の NPL ライブラリ
	NPL の本
1	ネガポジ判定
	NLP の本
2	テキスト要約について
	情報検索の仕組み
3	感情分析
	ツイート分析について
4	NLP に関するリソースを集めた
	KNP について

表 3　実験 1 の結果 2

(2) 英語 勉強法	
クラスタ	**タグ付け**
0	インド式
1	話せるようになるための勉強法
	英会話を独学で行うための勉強法
2	英会話アプリのサイト
3	単語の意味
	本の紹介
4	英語の 4 分野についての勉強法

4. 実験

本節では，3 節で提案したシステムを実装し，その有効性を検証するため，3 つの実験を行った．実装したシステムは，ユーザからの検索ワードを入力とし，取得した web コンテンツのテキストをもとに話題クラスタリング機能と難易度ソート機能を実行，対話機能によって web コンテンツの提案とユーザとの対話を繰り返し，web コンテンツの提案を行うものとなっている．

実験 1 では，話題クラスタリング機能のクラスタ数の有効性を検証する．実験 2 では，難易度ソート機能の難易度スコアの有効性を検証する．実験 3 では，任意の入力による出力結果を用いた被験者実験を行うことで，有効性を検証する．

4.1 節では，実験環境について述べる．4.2 節では，実験 1 の話題クラスタリング機能の有効性について述べる．4.3 節では，実験 2 の難易度ソート機能の有効性について述べる．4.4 節では，実験 3 のシステム全体の有効性の検証について述べる．4.5 節では，本研究による実験結果についての考察を行う．

4.1. 実験環境

システムの構築にはバージョン 3.9.16 の Python を用いた．また使用したライブラリについては，表 1 に示した通りである．

4.2. 実験 1(話題クラスタリング機能の有効性の検証)

実験 1 では，話題クラスタリング機能のクラスタ数の有効性を検証する．本実験では，特定の検索ワードから取得した web コンテンツを入力として，話題クラスタリング機能にかけ，出力される各クラスタから 2 つの web コンテンツを取得し，主著が主観でタグ付けを行う．

本実験では，「コロナ」，「Python 自然言語処理」を検索ワードとして，実験をおこなった．

実験結果を表 1，表 2 に示す．表 1 には，検索ワードが「Python 自然言語処理」の際の結果を示し，表 2 には検索ワードが「英語 勉強法」の際の結果を示した．

「Python 自然言語処理」では，自然言語処理の本を紹介している web コンテンツがクラスタ 0 とクラスタ 1 に存在している．このとき，それぞれの本が取り扱っている話題は，クラスタ 0，クラスタ 1 のどちらも，もう一方の web コンテンツとは関係の無い内容であった．一方，クラスタ 2 には，テキスト要約について取り上げた項目のみが存在した．他のクラスタには，テキスト要約について触れた記事が存在していない．

「英語 勉強法」では，すべてのクラスタにおいて，異なる内容の web コンテンツがクラスタリングされていた．

同じ話題が複数のクラスタにまたがっていることもあったが，クラスタごとに話題が異なっている場合も多く見られた．

4.3. 実験 2(難易度ソート機能の有効性の検証)

実験 2 では，難易度ソート機能の難易度スコアの有効性を検証する．本実験では，特定の検索ワードから取得した web コンテンツを入力として，難易度ソート機能を実行し，出力されたソート結果を検証する．本実験では，「コロナ」という検索ワードを入力として，クラスタリングした 5 つのクラスタのうち，ランダムに選んだ 2 つのクラスタに注目して実験を行う．それぞれのクラスタ内で難易度スコアが最も高い web コンテンツと最も低い web コンテンツの難易度スコアの要素を比較する．

実験結果を表 3 に示す．表 3 より，難易度が高い web コンテンツと低い web コンテンツを難易度スコアの要素で比較してみると，ひらがなの割合にほとんど差が無いことがわかる．一方、画像スコアと日本語のレベルでは、各サイトに違いが出ていることが読み取れる．また，難易度スコアが高いほうがひらがなの割合が，高くなっている．これは， web コンテンツの難易度が高い方が web コンテンツのひらがなの割合も高くなっていることを示す．

表 4　実験 2 の結果

難易度	高	低
日本語のレベル	40	54
ひらがなの割合	0.6616	0.6505
画像スコア	2	4.64

4.4. 実験 3 (システム全体の有効性の検証)

実験 3 では，システム全体の有効性を検証するため，被験者実験を行う．被験者実験では，それぞれの被験者に「Python スクレイピング」と「イヤホン」の 2 つの検索ワード検索を行ってもらう．最後に被験者に対して，計 2 問の質問をする．1 問目は，「話題に関して，求めるサイトが出力されたか」という質問に関して，「そう思う」または「そう思わない」で回答する形式である．2 問目は，「難易度に関して，求めるサイトが出力されたか」という質問に関して，「そう思う」または「そう思わない」と回答する形式である．実験には，10 代の男女 4 名が参加した．

実験結果を表 3 に示す．1 問目では，被験者に対して，「話題に関して，求めるサイトが出力されたか」という質問という質問をし，全員が「そう思う」と回答した．2 問目では，被験者に対して，「難易度に関して，求めるサイトが出力されたか」という質問をし，「Python スクレイピング」に関しては，全員が「そう思う」と回答したが，「イヤホン」に関しては，1 人が「そう思わない」と回答した．被験者からは「説明が目的ではないサイトの難易度の判定が難しい」という声や，「一覧で web サイトが表示されて，上から見ていくよりも楽だった」という声や「実行が重い」という声があった．

4.5. 考察

本節では，3 つの実験に関する考察を行う．

実験 1 では，話題クラスタリング機能の有効性を検証するための実験を行なった．

実験 2 では，難易度ソート機能の有効性を検証するための実験を行なった．

実験 3 では，本方式の有効性を検証するために自被験者実験を行なった．

実験 1 の結果より， 2 つの検索ワードのうち 1 つでは，web コンテンツを話題別にクラスタリングできたが，もう一方では話題別にクラスタリングできていない場合がみられた．この結果から，検索ワードによっては，話題別にクラスタリングができたと言える．

実験 2 の結果より，ひらがなの割合の差が小数点以下にしか現れず，画像スコアと日本語のレベルに比べて難易度スコアへの影響が小さいと言える．また，日本語のレベルの値が大きく，影響が大きくなりすぎてしまうことがわかった．今後の課題として，最適な正規化の方法を模索していきたいと考えている．

難易度スコアが高いほうがひらがなの割合が大きいという結果に対しての考察として，難易度スコアが低いサイトでは，画像の中で説明がされている場合が見られ，この場合，画像の見出しが簡潔な日本語で書かれている事が多かったことが要因ではないかと考えている．

実験 3 の結果より，ユーザのニーズと合致する web コンテンツの探索を可能にするといえるが，実験 3 の自由記述の中で実行時間が長いという問題点はあるが、

実験 3 の結果より，ユーザのニーズと合致する web コンテンツの探索を可能にするといえるが，実験 3 の自由記述の中で実行時間が長いという記述があった．この原因として，web コンテンツの取得に時間がかかることが挙げられる．キャッシュを利用するなど，アルゴリズムの最適化を行うことで改善を行いたいと考えている．

5. おわりに

本稿では，日本語の web コンテンツを対象として，web コンテンツの話題・難易度を推定し，対話式でユーザに提案する検索インタフェイスの実現方式について示した．

本方式は，ユーザを任意の検索ワードを入力とし，web コンテンツを取得する．取得した web コンテンツを，話題クラスタリング機能と難易度ソート機能により，話題と難易度の観点から整理し，対話機能により，提案した web コンテンツに対してユーザからフィードバックを受け取り，再提案を行う．また，提案方式を実装し，有効性を検証する実験を行なった．

今後の課題としては，インタフェイスの実現、本方式を実現する新たなプラグインの実装、規模を拡大した被験者調査によるシステムの有効性の検証が挙げられる．

本システムによって，ユーザのニーズと合致する web コンテンツを手軽に探索することが可能となる．

参 考 文 献

[1] 高須賀 清隆，丸山 一貴，寺田 実 “閲覧履歴を利用した協調フィルタリングによる Web ページ推薦とその評価”， 情報処理学会研究報告 Vol.65 pp.115-120，2007.

[2] 永井 洋一 “ユーザの閲覧履歴に基づくオンライン検索支援システム”，日本ソフトウェア科学会第 23 回大会論文集，2006.

[3] 建石 由佳，小野 芳彦，山田 尚勇 “日本文の読みやすさの評価式”，情報処理学会研究報告ヒューマンコンピュータインタラクション Vol.25 pp.1-8，1988.

[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] 砂川陽一・李 治鎬・高原正明 “日本語学習者用辞書作成を支援するデータベースの構築”，Acta Linguistica Asiatica，2 (2)，97‐115，2012.

[6] 波多野 賢治，佐野 綾一，段 一為， 田中克己 “自己組織化マップと検索エンジンを用いた Web 文書の分類ビュー機構”，情報処理学会論文誌 Vol.40 pp.47-59，1999.

[7] 吉田 大我，小山 聡，中村 聡史，田中克己 ” Web 検索結果におけるキーワード出現相関の可視化と対話的な質問変換”，DEWS2007，c7-2，2007.

[8] 自己組織化マップ，https://bsd.neuroinf.jp/wiki/自己組織化マップ

[9] Custom Search API，https://developers.google.com/custom-search/v1/introduction .

[10] Mecab，http://taku910.github.io/mecab/ .

[11] Neolog-D，https://github.com/neologd/mecab-ipadic-neologd/blob/master/README.ja.md .