

Web コンテンツを用いたなぞかけ自動生成の提案と評価

内村 圭佑[†] 灘本 明代^{††}

[†] 甲南大学大学院自然科学研究科 〒658-8501 兵庫県神戸市東灘区岡本 8-9-1

^{††} 甲南大学知能情報学部 〒658-8501 兵庫県神戸市東灘区岡本 8-9-1

E-mail: [†]mn924002@center.konan-u.ac.jp, ^{††}nadamoto@konan-u.ac.jp

あらまし 「なぞかけ」とは「A とかけて B ととく . そのところは C+D . 」という定型句に , ある程度の関連性を持った単語を当てはめて楽しむ日本に古くからある言葉遊びの一種である . 本研究では Web を利用してこのなぞかけ文を自動生成するシステムを提案する . 我々の提案する , なぞかけ自動生成システムは入力単語を A と D とし , 各々に関連する単語 B と単語 C を自動抽出しなぞかけを生成する . 具体的には , 入力された単語 A (名詞) と単語 D (形容詞) から関連する動詞 C を Web より取得する . そして , その動詞と同音異字にある単語 C' を用いて , それに関連する名詞 B を Web から取得する . 単語 A と単語 B との意味距離を Wikipedia を用いて計り , この意味距離が離れているとき , なぞかけを生成する .

キーワード なぞかけ , 意味的距離 , Wikipedia , 上位概念抽出

Automatic Generation of NAZOKAKE by Using Web content

Keisuke UCHIMURA[†] and Akiyo NADAMOTO^{††}

[†] Konan University Okamoto 8-9-1, Higashinada-ku, Kobe city, Hyogo, 658-8501 Japan

^{††} Konan University Okamoto 8-9-1, Higashinada-ku, Kobe city, Hyogo, 658-8501 Japan

E-mail: [†]mn924002@center.konan-u.ac.jp, ^{††}nadamoto@konan-u.ac.jp

Abstract “ NAZOKAKE ” is a Japanese traditional word plays. The grammar is “ A to kakete B to toku. Sonokokoro ha C+D ”. There is no semantic relation between A(noun) and B(noun), but there is relation between A and C(verb), B and C '(verb) based on D(adjective). In this paper, we propose the system which generates NAZOKAKE automatically. In our system, user inputs word A (noun) and D(adjective) and the system extracts word C(verb) and B(noun) from the web pages. In this time we calculate semantic distance between A and B by using Wikipedia.

Key words NAZOKAKE , Semantic distance , Wikipedia , Extract concept hierarchy

1. はじめに

近年のインターネットの普及に伴い , 人々が取得できる情報は数年前と比較して格段に増加し , インターネットは情報の宝庫となっている . しかしながら , 我々はこれらの情報のほんの一部を利用するにすぎなく , これらを十分に使いこなしていないのが現状である . 特に , 現在の若者は語彙力・発想力に乏しいと言われている [1] が , 我々はこの語彙力・発想力の強化にインターネットの情報を使えないかと考え , インターネット上にある様々な情報から発想を支援するシステムとして , ユーザーの入力に基づいたなぞかけ文を自動で生成する手法を提案する .

「なぞかけ」とは「A とかけて B ととく . そのところは C+D . 」という定型句に , ある程度の関連性をもった単語を当てはめて楽しむ日本に古くからある言葉遊びの一種である . そ

の起源は享保 13 年 (1728) に刊行された京都の俳人・重田梧山の著書「謎車水室桜」(なぞぐるま・ひむろのさくら) [2] と言われていたり , 686 年に天武天皇が問うたとされる無端事(あとなしこと) [3] , また平安時代に始まった芸能で新年を祝った万歳(まんざい)から派生した尾張万歳(おわりまんざい)や三河万歳(みかわまんざい)である [4] とも言われており , その真相は定かではない .

本研究ではこの「なぞかけ」の構造に注目し , 上記の単語 A と D を入力として各々に関連する単語 B と単語 C を抽出することを試みる . 例えば , 「アイロンとかけてギャンブルととく . そのところはどちらもカケル(かける/賭ける)とアツい」というなぞかけ文では , ユーザーの入力を (A: アイロン , D: アツい) とし , システムは (B: ギャンブル) と (C: カケル) を抽出することを行う . ここで単語 A, B, C, D に関して , これらの単語はそれぞれある程度の関連性を持ってはいるがここでは「単語 A

と C+D, 単語 B と C+D にはそれぞれ関連性があるが、単語 A と B の間には関連性が無い」という関係性が好ましい。なぜなら単語 A と B とが関連性を強く持ち、意味的に近い単語の場合、その二つがどちらも C+D の関係にあるのはさも当然であり、なぞかけとして成り立たないからである。その為、本研究においては単語 A 及び単語 B に関連する C+D に対し「読みは同じだが意味は違う(同音異字)」という性質を与え、つまり C+D に 2 つの意味合いを持たせている。そこで本論文では、単語 A を名詞、単語 D を形容詞とし、単語 B (名詞) と同音異字を持つ単語 C (動詞) の 2 つのセットを Web から取得することを行う。具体的には単語 B は抽出の際に、日本語 Wikipedia を用いた簡単な上位概念抽出手法を提案し、その抽出した概念構造を用いて単語 A との意味的距離を測り、単語 C に関しては Web 検索の結果を用いる。

なぞかけには先の例のように「アイロンとかけてギャンブルととく。そのところはどちらもカケル(かける/賭ける)とアツい」といった同音異字を用いるタイプと「ナイフとかけて水泳ととく。そのところはバタフライもある。」といった単語 A, B それぞれに共通の事象を提示するタイプの 2 パターンが考えられる。本研究では、誰にでも分かりやすいなぞかけを生成することを目的とし、前者の同音異字を用いるタイプのなぞかけを生成することを行う。

なぞかけ自動生成システムの特徴は以下の通りである。

- ユーザーの入力より、なぞかけを生成する。
- Wikipedia から出力単語(名詞)と入力単語(名詞)の上位概念を抽出する。

- 抽出した上位概念を用いて単語間の意味的距離を求める。

以下、2 章では関連研究、3 章では Wikipedia を用いた概念構造の抽出について述べ、4 章ではなぞかけの自動生成について、5 章で実験・考察とそれに基づいた改善と改善後の実験・考察、そして 6 章でまとめと今後の課題について述べる。

2. 関連研究

なぞかけ生成

前田ら [5] は、データベースを用いた面白いなぞかけの生成の研究を行っている。データベース内には単語が任意に収納されており、それらにはさらに付加情報としてある単語 A からある単語 B に対する関連度なども含まれている。前田らはアンケートによって得られた関連度も提案している。なぞかけとは言葉を用いた遊びの一種で、つまり言葉の意味や関連度というのはとても大事な要素の一つである。そのためなぞかけの生成においてこのような辞書の機能を果たすデータベースを作っておくことは有効であるといえる。しかしながら、それでは時事の単語に対応することが出来ないという短所もある。仮に対応しようとしてもその都度アンケートを取って関連度を測らなければならないため、現実的であるとは言いがたい。そこで我々はなぞかけを自動生成する手法として、人手により作成したデータベースを用いず、Web を用いて単語間の関連度をリアルタイムに自動で求める事を行う。これにより、前田らの研究のアンケートで得られるほどの正確な数値化の出来る関連度は期

待できないが、時事の単語に対応することが可能となる。また、同じ入力であったとしてもその時期によって異なる結果を得ることのできる時代に即したなぞかけの生成が可能である。

Wikipedia

中山ら [6] は、Wikipedia から独自の手法を用いたオントロジー構築を提案している。彼らは Wikipedia 内の記事における文法に着目し、独自の手法として LSP 法(記事のリード部分を重要文と見做して解析する手法)と ISP 法(記事の中から重要な文章を抽出して解析する手法)を提案し、それらを用いたオントロジー構築を行っている。しかしここで扱われている記事というのはあくまで英語 Wikipedia のものであり、日本語 Wikipedia には対応していない。つまり LSP 法においては記事の中からまず冒頭文を抽出して来る必要があるが、ここでは英語の文法のしくみを考慮して、文章の初めの単語の頭文字が大文字であることやピリオド(もしくは改行コード)の直前が小文字であること等を利用し、単純なルールで効率よく抽出を実現している。我々の研究では日本語 Wikipedia を用いた上位概念抽出を行うので、これを日本語 Wikipedia に対応させようと考えたとき、上記の単純な冒頭文抽出のロジックは言語構造が違うので通用せず、英語 Wikipedia で構築されるオントロジー程の精度は望めないが、冒頭文は同じく is-a 関係にあることから LSP 法は実用的な手法であると言える。他にも同様に、Wikipedia の冒頭文から上位概念を取得している研究として隅田らの研究 [7] が挙げられる。

また、中山らによると Wikipedia のリンク構造を用いた解析手法 [8], [9] は情報量の少なさが原因で、満足する結果が得られないことなども挙げられている。よって本研究においても Wikipedia 解析に際してはリンク構造を用いずに行うこととする。

3. Wikipedia を用いた概念構造の抽出

本章では Wikipedia からの簡単な上位概念抽出手法と、意味距離の計測方法を提案する。提案手法は、記事のリード部分(冒頭文)を重要文と見做して解析する手法、LSP(Lead Sentence Parsing) 法 [6] に基づいている。

3.1 Wikipedia からの上位概念の抽出

Wikipedia 内の文章は多くの場合冒頭文が is-a 関係にあることに注目し、この関係を利用して単語の上位概念を抽出することを行う。具体的には、最初の一文(冒頭文)の最後に出てきた名詞をその単語の上位概念とする。例えば Wikipedia で「なぞかけ」という記事を探すと、冒頭文の一文目は「なぞかけとは、なぞなぞの一種と言える言葉遊びである。」となっており、最後に出てくる名詞は「言葉遊び」である。これは「なぞかけ」の上位概念としてふさわしい。しかし実際に抽出される上位概念は精度の低い上位概念抽出である場合がある。つまりはすぐ上にある概念ではなく、場合によってはある単語の上位概念の上位概念、もしくはそれ以上上の上位概念である場合があるが、本研究では必ずしもすぐ上の上位概念を必要としないため、本手法を用いる。実際には以下に示す精度の結果を得ることが出来る。

- [0] => マティーニ, [1] => カクテル, [2] => 飲料, [3] => もの
- [0] => 塩化ナトリウム, [1] => 物質, [2] => 存在, [3] => 呼称
- [0] => 温暖化, [1] => 現象, [2] => 事実, [3] => 表現, [4] => 事

ここで [0] は初めに入力された単語を, [1] は [0] の, [2] は [1] の, [3] は [2] の上位概念を示す.

より具体的に, Wikipedia を用いた上位概念抽出では, Wikipedia の特徴から, クエリに対して冒頭のパラグラフを抽出している. 特徴とは, Wikipedia ではクエリに対してパラグラフの頭が「<p>クエリ」の形で出現するケースが多く, またパラグラフの終わりは「</p>」で表されるというものである. 次に形態素解析を行い, 抽出されたパラグラフの最初に出てくる「。」の手前に出現する名詞を上位概念として抽出し, 次はそれをクエリとして同じ作業を繰り返す (図 1 参照).

3.2 意味距離

単語 a_0, b_0 において得られた上位概念を用いて, 2 つの単語の間にある意味的距離を以下の手順で求める.

1. a_0, b_0 の上位概念を取得し, これを各々 a_1, b_1 とする.
2. 同様に順次 15 階層上までの上位概念を取得し $a_n (n = 0, \dots, i, \dots, 15), b_m (m = 0, \dots, j, \dots, 15)$ とする.
3. a_n と b_m を比較する.
4. $a_i = b_j$ となった場合, 2 つの単語 a_0 と b_0 は意味的に近いとする.
5. $a_i = b_j$ とならない場合, 2 つの単語 a_0 と b_0 は意味的に離れているとする (図 2 参照)

上記の手順で意味的な距離を求めるが, 上記 3. において $a_n = b_m$ となった時, a_0 と b_0 は共に $a_n (= b_m)$ を上位概念に持つことから, a_n を根ノードとするツリー構造で表わすことが出来るため, 数値としての距離は単純に $n + m$ で求められる. 例えば, a_0 が「塩化ナトリウム」の時, a_1 は「物質」, a_2 は「存在」, a_3 は「呼称」である. また b_0 が「ダイヤモンド」の時, b_1 は「物質」, b_2 は「存在」, b_3 は「呼称」である. この時 $a_1 = b_1$ となるので, 二つの単語「塩化ナトリウム」と「ダイヤモンド」は距離が 2 であり, 意味的に近いと判断できる.

4. なぞかけの自動生成

この章ではなぞかけを自動生成するための仕組みを説明する. 本論文では, 同音異字を用いたなぞかけの自動生成を目標とする. 同音異字を用いたなぞかけの例として「アイロンとかけてギャンブルととく. そのころはカケル (かける/賭ける) とアツい。」を挙げ, これ (以下「甲」と略) を使ってこれよりシステムの説明をしていく.

初めに, 人がなぞかけを生成する手順を説明する. 甲の場合まず「アイロン」を足掛かりとして, それがどんなものかを考える. そこからアイロンとは「かけるとアツい」ものであるということ連想し, 次に, 同じく「かけるとアツい」ものについて考える. ここで「かける」の部分に注目して, これを同音異字である「賭ける」と置き換え, 「賭けるとアツい」ものを考

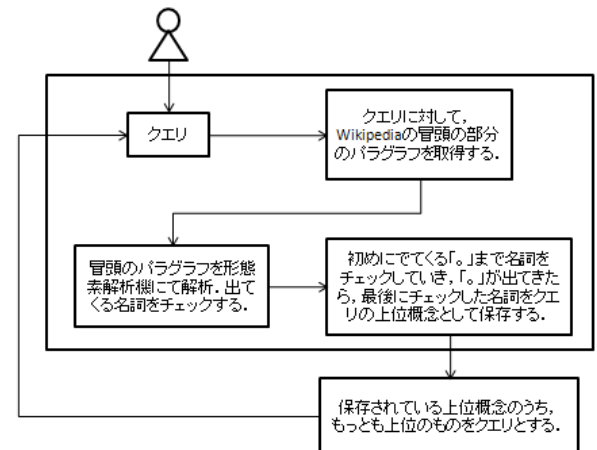


図 1 Wikipedia からの上位概念抽出

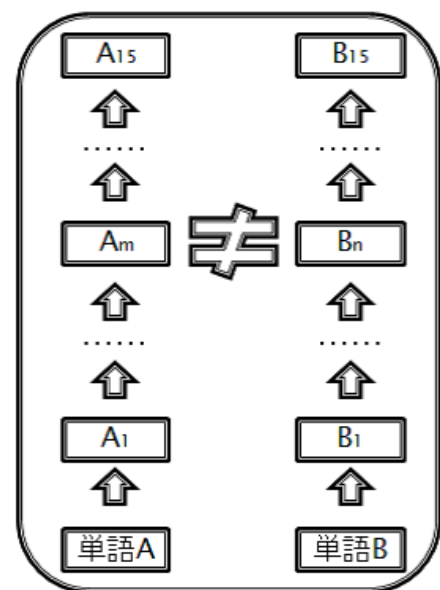


図 2 意味距離

え出すことで一種の駄洒落要素を付け加える. これによりなぞかけに面白味を持たせることが出来る. そして最後に「賭けるとアツい」ものとして「ギャンブル」を連想すれば, それらを組み合わせると甲が出来る.

次に上記の作業をコンピュータにさせる手法を説明する.

1. ユーザーの入力した名詞 A と形容詞を基に, それらを結びつける動詞を Web 上から取得する.
2. 取得した動詞の同音異字にあたるすべての単語をユーザーに提示する.
3. ユーザーの選択した同音異字と上記 1. で入力された形容詞を基に, それらに関連する名詞 B を Web 上から取得する.
4. 上記 1. と 3. で取得した名詞 A, B 間の意味距離を測る.
5. 測った意味距離が離れている時, なぞかけを出力する. ユーザーから入力された「アイロン (名詞)」と「アツい (形容詞)」から, それら二つの単語を結びつける単語として「かけ

表 1 システムの結果例

ユーザーの入力	システムが抽出した動詞	生成したなぞかけ
「結婚式」「綺麗」	「着る」	「結婚式」とかけて「剣」ととく．そのころはどちらも「(着る/斬る) 綺麗」
「Wii」「面白い」	「買う」	「Wii」とかけて「犬」ととく．そのころはどちらも「(買う/飼う) 面白い」
「ブランコ」「上手い」	「乗る」	「ブランコ」とかけて「絵」ととく．そのころはどちらも「(乗る/載る) 上手い」
「パトカー」「早い」	「乗る」	「パトカー」とかけて「記事」ととく．そのころはどちらも「(乗る/載る) 早い」
「猫」「難しい」	「撮る」	「猫」とかけて「レギュラー」ととく．そのころはどちらも「(撮る/獲る) 難しい」

Table 1 Example of Results

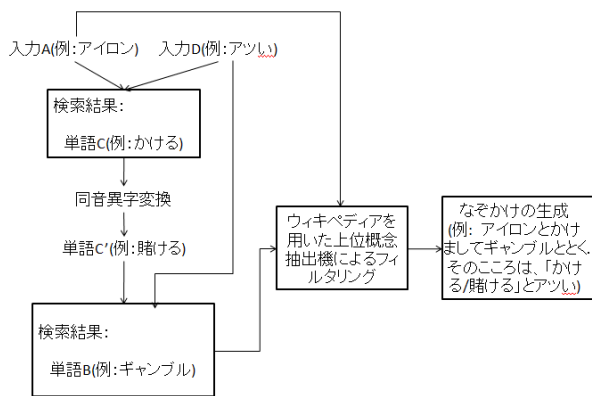


図 3 なぞかけ生成システムの流れ

る(動詞)」を得るために、「アイロン アツい」をクエリとした時の Web の検索結果を用いる。この「かける」の抽出基準は、検索結果の中の単語(動詞)の頻出度により決定することとする。次に「かける」を同音異字変換し「賭ける」にする。これは人がなぞかけを生成する過程における、駄洒落要素を絡ませる部分に相当する。得られた「賭ける(動詞)」と、入力された単語「アツい(形容詞)」を用いて再度「賭ける アツい」をクエリとした Web 検索を行い、検索結果から頻出度を基準として「ギャンブル(名詞)」を抽出する。これをもって「アイロン」に対して、同じく「かけるとアツい」ものとして「ギャンブル」が選出され、それらの単語を組み合わせることで甲を生成する。

また上記の駄洒落要素を創る部分はユーザーのセンスによって十人十色の組み合わせが予想される部分である為、一概にシステム側で選択を行うのは好ましくないと考える。よってあくまでシステムとしては同音異字を選択肢としてユーザーに示し、それらをユーザー側に選択させる方式をとる。つまり上記の例では「かける」の同音異字として「賭ける」を選択したが、この部分はユーザーの好みによって「掛ける」や「駆ける」等の単語を選択することも可能であるし、その場合最後に得られる単語も選択された単語に合うものになってしまう。これによりユーザーは自分の嗜好に沿ったなぞかけの生成を目標むことが可能となる。

最後に、入力された単語「アイロン」及び、得られた単語「ギャンブル」に対して、Wikipedia に基づく上位概念抽出器を用いて単語間の意味的な距離を測る。それによって二つの単語が意味的に離れていることをチェックするためである。これは最終的に得られた単語が入力単語である「アイロン」と意味的に近かった場合、「アイロン」と「アイロンに意味的に近いも

の」が、同じく「かけるとアツい」のは当たり前であり、つまりそれらを用いたなぞかけは面白味に欠けると考えられるからである(図 3 参照)。

表 1 に本システムにて作成したなぞかけの例を示す。

5. 実験・考察

ここではシステムの有用性を測る為に、プロトタイプシステムを用いた実験を行った。実験は被験者 12 名を対象とし、それぞれを 4 人 1 組の 3 グループ(A,B,C)に分け、グループ A の実験結果を基に改善を行い、その後グループ B と C に実験を行う順序を踏む。また実験は被験者にあらかじめシステムの使い方を説明して行った。

5.1 実験(1)

ここではグループ A に対して実験を行った。プロトタイプシステムにはユーザーが無作為に単語を入力できる場所が二ヶ所(名詞(A)と形容詞(D))ある。被験者は二つの入力をし、途中システムがユーザーの入力に基づいて出力した同音異字群の中から意に沿った同音異字(C')を選ぶ。実験結果の表記例は以下の通りである。

・「A」とかけて「B」ととく．そのころはどちらも「(C/C') D」

ここで単語 C はユーザーの入力に基づいてシステムが Web から抽出したものであり、単語 C' は選択された単語 C の同音異字である。以下に実験結果の一部を表記例に従って示す。

・「役者」とかけて「人」ととく．そのころはどちらも「(思う/おもう) 渋い」

・「元気」とかけて「人」ととく．そのころはどちらも「(言う/いう) 若い」

・「夜」とかけて「最近」ととく．そのころはどちらも「(思う/おもう) 怖い」

上記のなぞかけは面白いとは言えない。それは被験者が C' の部分に C と同じ意味の単語を選択しており、その結果なぞかけ自体に駄洒落の要素が含まれていない為である。つまりユーザーは同音異字を選択する際、なるべく C' に対して意味的に離れた C' を選ばないと面白いなぞかけを得ることが出来ないことがわかる。

5.2 考察(1)・改善

実験の後「使い勝手、満足度、実用性」についてアンケートを取った結果、全体的に控え目な結果が得られた。また、コメント欄には「形容詞がなかなか思い浮かばなかった。」や「同音異字の選択肢が少ない。」といった意見が目立った。

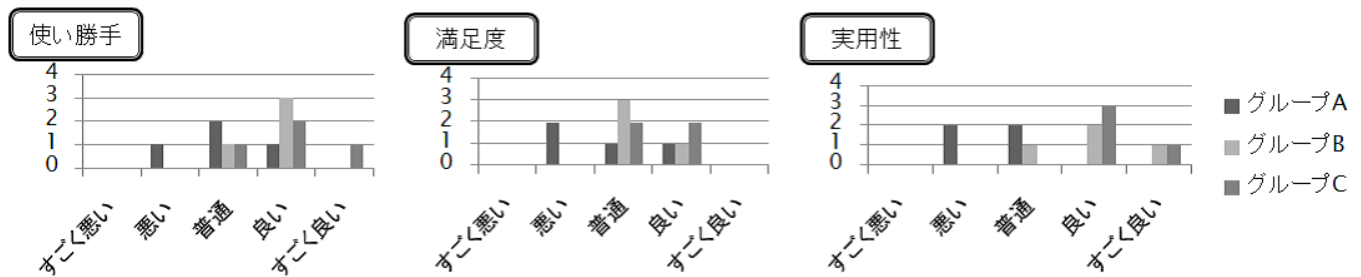


図 4 実験アンケート

「形容詞がなかなか思い浮かばない」という問題に対して、これは原因として、もともと日本語で品詞それぞれの単語の種類数を考えたとき、名詞や動詞に比べて形容詞は数が圧倒的に少ない[10]ことが挙げられる。また「同音異字の選択肢が少ない」という問題に対して、これはユーザーの思考に問題があると考えられる。つまり、ユーザーは単語を入力する際「車は速い」や「夜は暗い」といった発想で単語 A と D を決める。しかしそのように形容詞を決めてしまうと、システムが出力する単語 C は「言う」や「思う」に偏りやすい。そして「言う」や「思う」には同音異字があまり存在しないことから、上記のような問題が発生したものと考えられる。

以上の問題を解決するために、ユーザーが入力した単語 A(名詞)に対して、システムが単語 D(形容詞)の候補を抽出することで、ユーザーの思い浮かばない単語 A に関連した形容詞を推奨できる他、それによりシステムが抽出する単語 C についても同音異字の少ない「言う」や「思う」に偏ることが解消されるのではないかと考えられる。

そこで、プロトタイプシステムにさらに形容詞候補抽出機能を実装する。なお形容詞候補抽出機能は非常に単純な作りとなっており、ユーザーの入力単語 A(名詞)に対して Web 上で検索をかけ、検索結果スニペットの中で出現する形容詞を頻出度順に並び替えて提示する仕組みである。

5.3 実験・考察 (2)

プロトタイプ改善後、続いてグループ B, C に対して実験を行った。以下が得られたなぞかけである。

- 「メール」とかけて「ページ」ととく。そのころはどちらも「(来る/繰る) 鬱陶しい」
- 「荷物」とかけて「けが」ととく。そのころはどちらも「(おる/折る) 軽い」
- 「夏」とかけて「料理」ととく。そのころはどちらも「(乗り切る/乗り切る) 暑い」
- 「捨て猫」とかけて「母」ととく。そのころはどちらも「(見る/看る) 哀しい」
- 「コンピュータ」とかけて「気」ととく。そのころはどちらも「(使う/遣う) はやい」

またグループ A, B, C のアンケート結果をまとめて図 3 に示す。アンケートの結果から、形容詞抽出システムの付加により形容詞に悩むことがなくなったため、使い勝手に対する評価が上がり、また実用性の評価も向上した。しかし、満足度に関してはそれほど評価が良くなっていない。これに関して被験者が

らの意見として「同音異字の選択肢が少ない」というものが大半を占めた。この問題に関して例えば「見る」という単語なら同音異字として「看る」や「観る」、さらには「診る」等が挙げられるが、どれも意味的に近い単語であることが一目瞭然である。このように同音異字はあるが意味的に近い単語ばかりである場合、ユーザーが満足するなぞかけを抽出することは容易ではないということが解った。この問題の解決策として、例えば「はじめに」で挙げているような単語 A, B に対して共通の事象を提示するタイプのなぞかけ等、同音異字にとらわれないなぞかけ生成ロジックが必要であると考えられる。

6. まとめと今後の課題

本研究では Wikipedia からの上位概念抽出と、それに基づくなぞかけの生成手法を提案し、また実験の結果から、当初の目的であった Web からの同音異字を用いたなぞかけ文自動生成を達成した。これによりインターネット上に存在する膨大な量の情報の中から、少しでも有用であり、ユーザーの発想を支援できる情報を提示できるなら重量である。また今後の課題を以下に記す。

- 日本語 Wikipedia からのより精度の高い上位概念抽出手法を提案し、単語間の意味距離を厳密に測る。
- 実験で得た結果を活かして、同音異字の選択肢がユーザーを満足させない時、違うロジックでなぞかけを生成するようにパターンを作っておく。
- 現段階でユーザーからの入力名詞と形容詞に限られるが、それ以外の品詞にも対応したルールを考え、生成されるなぞかけの幅を広げる。

謝 辞

本研究の一部は、平成 21 年度科研費特定領域「コミュニティ型コンテンツのコンテンツホール検索に関する研究」(課題番号: 21013044, 代表: 灘本明代) 及び甲南大学平生太郎基金科学研究奨励助成金による。ここに記して謝意を表します。

文 献

- [1] 藤原正彦「教室を超えて生きる国語力」VIEW21[小学版]2005 年 4 月号 pp. 13
- [2] お笑いメルマガ なぞかけめ～る第 3 号 はぐまぐ:
<http://archive.mag2.com/0000159712/20050621200000000.html?page=7&js>
- [3] なぞかけユーモア (吉村外喜雄のなんだかんだ):

http://www.noevir-hk.co.jp/magazine/2008/06/post_471.html

- [4] 元祖！俺は熱いうちに打て！：なぞかけ講座：
<http://shariki.hamazo.tv/c9613.html>
- [5] 前田実香，鬼沢武久，“言葉の関連性とおもしろさを取り入れたなぞかけ生成”，2004「エンタテインメント感性」ワークショップ講演論文集，pp. 37-42, December, 2004
- [6] 中山浩太郎，原隆浩，西尾章治郎，“自然言語処理とリンク構造解析を利用した Wikipedia からの Web オントロジー自動構築”，日本データベース学会論文誌，Vol. 7, No. 1, pp. 67-72, June, 2008
- [7] 隅田飛鳥，吉永直樹，鳥澤健太郎，萬成賢太郎，“Wikipedia からの大規模な上位下位関係の獲得”，言語処理学会年次大会発表論文集，JST 資料番号 L3842A，Vol. 14th, pp. 769-772
- [8] 桜井慎弥，手島拓也，石川雅之，森田武史，和泉憲明，山口高平，“日本語 Wikipedia からの汎用オントロジーの構築と評価”，第 22 回人工知能学会全国大会論文集 3f2-3，(2008.6)
- [9] 宮城良征，當間愛晃，遠藤聡志，“日本語オントロジー辞書システム Ontolopedia を用いた検索手法に関する一考察”，第 22 回人工知能学会全国大会 (jsai2008)，2D2-01 (2008.06.12)
- [10] 玉村文朗，“語彙論から見た形容詞”，同志社国文学 10 号<TV10274584>pp. 87, 1975-02