3R-8

カタカナ抜け文のための自動解法アルゴリズムの提案

島 広幸† 建石 由佳†† 小西 克巳†††
†工学院大学大学院工学研究科情報学専攻
††ナラプロ・テクノロジーズ
†††工学院大学情報学部コンピュータ科学科

1. はじめに

カタカナ抜け文とは，カタカナ語が抜けている文章を同じ文字には同じ番号があてはめられているという制限と，周辺の情報を手がかりに復元する文字パズルである．図1に例としてカタカナ抜け文の問題の1文を示す．

カタカナ抜け文を解くためには，周辺の文章からカタカナ語を推測・連想しなければならない．周辺の文章から推測・連想を行うためには何らかの言語処理が必要となる．

本研究では，このような推測・連想処理を必要とするカタカナ抜け文を，単純な共起情報とパズルの制限を利用することで解くアルゴリズムを提案する．

2. 関連研究

言語処理を必要とするパズルの研究として，クロスワードパズル(以下，クロスワードと略記)の研究が行われている[1,2,3]．クロスワードは，カギと呼ばれるヒントから単語を推測し，グリッドと呼ばれる格子のマス目の空欄を埋めるパズルである．

[1]で提案されているアルゴリズムは American-style と呼ばれる英語のクロスワードを解くアルゴリズムで，New York Times のクロスワードを単語正解率[1]95.3%で解くことができる．[2,3]は日本語クロスワードを解くアルゴリズムを提案し，単語正解率はそれぞれ 44%，15%である．日本語クロスワードの単語正解率が低い原因は，クロスワードの制約が弱く正確に単語を推測しなければならない点や，実装に一部未完成な部分がある点などが挙げられる．

クロスワードを解くアルゴリズムでは，処理を「答え候補生成処理」と「解生成処理」という 2 つの段階に分けてクロスワードを解く．「答え候補生成処理」では，カギを解析し，言語資源から答えの候補となる単語を収集する．「解生成処理」では，答え候補から最適解を選択し，グリッドを埋め解生成を行う．本研究もクロスワードを解くアルゴリズムと同様に，処理を「答え候補生成処理」と「解生成処理」の 2 つに分けて行うが，それぞれの処理はカタカナ抜け文の特徴に合わせた処理を行う．

3. カタカナ抜け文の特徴

クロスワードと比較した場合，カタカナ抜け文には，第一に，問題が 1 つの文章という特徴がある．そのため，未知語が文中に埋め込まれてしまうため，クロスワードとは異なり未知語に対するヒントを明確に得ることは難しい．また，クロスワードのカギはいくつかのジャンルやタイプに分類が可能である．したがって，属するジャンルやタイプごとに探索する言語資源を変え，候補生成を行うことができるが，カタカナ抜け文はそのようなことはできない．

__ __ 停前の古い __ __ __ __ から，
1 2 3 4 5 6
新しい __ __ __ __ __ に引っ越しだ．
7 8 9 10 8

図1 カタカナ抜け文の例

一方で，カタカナ抜け文は単語同士の制約がクロスワードより強いという特徴もある．クロスワードでは，ある単語は，最大でも単語長個の単語としか共通部分を持たない．例えば 4 文字の単語であれば，最大で 4 つの単語としか交差しないので，共通部分を持つ単語は最大で 4 つしかない．しかし，カタカナ抜け文は，単語長以上の個数の単語と共通部分を持つことがある．例えば，図 1 の 1 つ目の未知語は番号 1 を含む 2 文字の単語であるが，他に番号 1 を持つ単語が 3 つある場合もあり，すると，この未知語は 3 つの単語と交差することになる．

また，クロスワードでは他の 1 つの単語と共通する文字は 1 文字しかないが，カタカナ抜け文は同じ番号を複数持つ単語が存在し得るので共通する文字は 1 文字とは限らない．

このようにカタカナ抜け文は，クロスワードより強い制約を持つ．

4. カタカナ抜け文を解く手法

カタカナ抜け文を解くために，処理を「答え候補生成処理」と「解生成処理」の 2 つの段階に分けて考える．

4.1 答え候補生成処理

問題文から各未知語の答え候補を生成する．第 2 節で述べたようにカタカナ抜け文では，未知語の周辺の情報などから答え候補を連想しなければならない．そこで，本研究では答え候補を生成するために共起情報を用いる．共起情報のデータは Google が提供する Web 日本語 N グラム[4]を利用した．Web 日本語 N グラムデータは，Web から抽出した 200 億文の日本語データから作成された N グラムデータである．ただし，N グラムデータからデータの先頭の単語がカタカナで構成されるデータと，末尾の単語がカタカナで構成されるデータのみを抽出しておき，その抽出したデータを本研究では利用する．

答え候補生成の流れは以下の通りである．

1．未知語と共起する単語を取得するために，問題文を形態素解析し，未知語と同じ文中に出現する自立語を取得する．図 1 の例文の場合，文中の未知語 3 つに共起する単語は，文中の自立語「停，前，古い，新しい，引っ越し」となる．本研究では，形態素解

[1] クロスワードパズルに含まれる単語のうち，正解した単語の割合．

析は形態素解析器 MeCab[2]を利用し，行った．

2．N グラムデータから，1 で取得した自立語を 1 つ以上含むデータを探索し，そのデータの先頭または末尾のカタカナ語が未知語の文字数と一致した場合，答え候補として追加する．また，同時に頻度も取得し，すでに候補内に同じ単語を存在する場合は加算し総頻度を計算する．頻度は単語の重みとし利用する．この処理を行う際，使用する N グラムデータの N の数を変更することで，共起範囲を変更することができる．

3．候補にフィルタをかけ候補の絞り込みおよび重みの変更を行う．フィルタは以下の 3 つを用いる．

・文字重複フィルタ

1 つの単語内で同じ番号が複数個所にある場合，同じ番号の場所には同じカナが入らなければいけないという制限を利用し，候補を絞るフィルタ．例えば，図 1 の 3 番目の未知語では，2 文字目と 5 文字目の番号が同じなので，同じカナが入らなければならない．そこで，答え候補から 2 文字目と 5 文字目が異なるカナの単語は答え候補から削除できる．

・頭文字フィルタ

他の未知語で頭文字になる場所には「ン」と「ー」は入らないという制限を利用し，候補を絞るフィルタ．図 1 では，各未知語の頭文字の番号は「1，2，7」である．この場合，番号「1，2，7」には「ン」と「ー」が入らないことがわかるので，それぞれの未知語の答え候補の中で番号「1，2，7」の場所に「ン」や「ー」が入っている候補単語は削除できる．

・最高頻度フィルタ

カタカナ抜け文 17 問のうち 16 問で出現頻度が一番高い番号には「ー」が入るという結果を利用し，単語の重みを変更する．本研究では，最高頻度の番号の場所に「ン」がある単語の重みを 0.95 倍し，そうでない単語の重みは 0.05 倍した．

4．各未知語に対する候補を正規化し，候補を生起確率付き候補にする．

5．候補の中から生起確率が閾値以下の単語を削除する．カタカナ抜け文の答えには，一般的に使用されているカタカナ語が使用されるため，N グラムデータの抽出元である Web 上でも頻度のある程度高い確率で出現する単語であると考えられるためこのような処理を行う．

4.2 解生成処理

最適解は A*探索アルゴリズムを使用し，生起確率が最大となる解を探索する．その際のヒューリスティック関数は，確定されていない各未知語の候補から，現在までに確定された単語から得られる文字制約(どの番号にどのカナが入るかという制約)を満たす単語の最大確率の総積とする．

5. 実験・考察

市販のパズル雑誌から集めたカタカナ抜け文 17 問に対し，実験を行った．問題の平均未知語数は 42 語，平均未知文字数は 36.2 文字である．

表 1 平均単語正解率 (%)

N	0.001	0.01	0.1
3	78.6	39.0	15.0
4	73.8	36.4	8.8
5	72.6	26.7	10.0
6	68.1	26.0	10.0
7	66.3	30.0	7.5
平均	71.9	31.6	10.3

実験は N=3～7 の N グラムデータを使用した場合それぞれで行った．また，それぞれの N グラムデータを使用した場合に対し，候補削減のための確率の閾値を 0.1，0.01，0.001 に変え，実験を行った．

表 1 に結果として平均単語正解率を示す．平均単語正解率は，正解単語のうちいくつ正解したかの割合である．

実験の結果，平均単語正解率は 71.9%であった．

まず，確率の閾値を変化させた場合の正解率の変化に注目すると，閾値の値が小さくなるにつれ正解率は大きく増加していることがわかる．これは，閾値の値が大きすぎると候補の中に正解単語が含まれなくなってしまうためと考えられる．

次に，使用する N グラムデータを変化させた場合に注目すると，多くの場合 N の値が大きくなるにつれて正解率は低下する傾向にあることがわかる．各 N グラムデータ使用時における平均候補数を調べる予備実験を行った結果，N の値が大きくなるにつれて候補数は増加することがわかった．候補数が増えることで候補内の各単語の平均生起確率は低下する．この結果より，候補内の正解単語の生起確率も低下してしまったことが，N の値が大きいほど単語正解率が低下した原因であると考えられる．

実験の結果，候補に正解単語が多く含まれているほど正解率が高くなることが明確になったので，候補生成の処理の性能を向上させることで，正解率も向上するものと考えられる．

6. まとめ

本研究では，カタカナ抜け文のための自動解法のアルゴリズムを提案した．処理を「答え候補生成処理」と「解生成処理」に分け，答え候補生成は共起情報を利用して行い，解生成は A*探索アルゴリズムを用いて行った．実験の結果，平均単語正解率は 72%であった．

今後の課題として，より候補を絞る手法や解生成処理の計算量の削減などが挙げられる．

参考文献

[1] Noam M. Shazeer, Michael L. Littman and Greg A. Keim. Solving Crossword Puzzles as Probabilistic Constraint Satisfaction, Proceedings of the Sixteenth National Conference on Artificial Intelligence,pp.156-162,1999.

[2] 佐藤理史. 日本語クロスワードパズルを解く,情報処理学会自然言語処理研究会,NL-147-11,pp.69-76,2002.

[3] 神保一樹, 高村大也, 奥村学. 拡張 Potts Model を用いたクロスワードパズルの解き方,人工知能学会全国大会,2J3-01,2008.

[4] 工藤拓, 賀沢秀人. Web 日本語 N グラム第 1 版,2007.

[2] http://mecab.sourceforge.net/