

日本語クロスワードパズルのカギの解法

内木 賢吾[†]

佐藤 理史[‡]

駒谷 和範[‡]

[†]名古屋大学 工学部 電気電子・情報工学科 [‡]名古屋大学大学院 工学研究科 電子情報システム専攻

1. はじめに

クロスワードパズル(以下、クロスワードと略記)は世界中で親しまれている言語パズルのひとつである。クロスワードを解くためには、「カギ」と呼ばれるヒントから答となる語を推定し、グリッドと呼ばれるマス目を埋める(グリッドフィリング)必要がある。

クロスワードを解くシステムに、アメリカ式クロスワードを対象とした Proverb [1] がある。このシステムは、実例データベース(約 35 万件)を利用してカギから答の候補を求めた後、確率的制約充足問題としてグリッドフィリングを解く。グリッドフィリングのアルゴリズム [2] は、言語非依存のため、日本語クロスワードを解く場合にも、そのまま用いることができる。

本論文では、日本語クロスワードを対象として、カギから答の候補を求める方法について述べる。今回使用したクロスワードは、『クロスワード掲示板 [3]』に掲載されている 33 問(1282 個のカギ)である。このうち、8 問(264 個のカギ)をシステム開発用に使い、12 問(499 個のカギ)をシステム評価用に用いた。

2. カギの解法

システムの構成を図 1 に示す。本システムは、(1) カギの解析、(2) 連想タイプ別候補生成、(3) 候補リストのマージ、の 3 ステップで、カギから答の候補を求める。

2.1 カギの解析

「海の反対(2)[§]」というカギを考えよう。このカギに対し、我々は「海」の反対語、たとえば、「りく」や「おか」を思い浮かべる。この過程を連想過程と捉えた場合、カギに含まれる「X の反対」という表現が、X の反対語の連想を促す役割を果たしていると思なすことができる。このように、カギには特定の種類の連想を促す情報が含まれる場合が多い [4]。

本ステップでは、カギを解析し、その特徴に基づいて、カギを解くための連想タイプと引数を決定する。例えば、「海の反対(2)」というカギに対しては、「X の反対」という特徴に基づいて、連想タイプとその引数、すなわち、「反対語(海)」を割り当てる。

本研究で定義した連想タイプの分類を表 1 に示す。これらの連想タイプは、システム開発用のカギ(264 個)の調査に基づいて定めた。なお、この分類は、排他的分類ではない。現在のシステムは、これらのタイプのうち、「説明」「同義語」「反対語」「穴埋」の 4 タイプを使用する。「説明」を除く 3 つのタイプは、カギに特定の表現や空欄が存在した場合に、割り当てる。一方、「説明」は、すべてのカギに対して割り当てる。すなわち、ひとつのカギに対して複数の連想タイプが割り当てられることがある。たとえば、前述の「海の反対(2)」に対する出力は、「反対語(海)、説明(海の反対)」となる。

2.2 連想タイプ別候補生成

本ステップでは、各連想タイプに対して定義される連想モジュールを呼び出し、スコア付きの答の候補リ

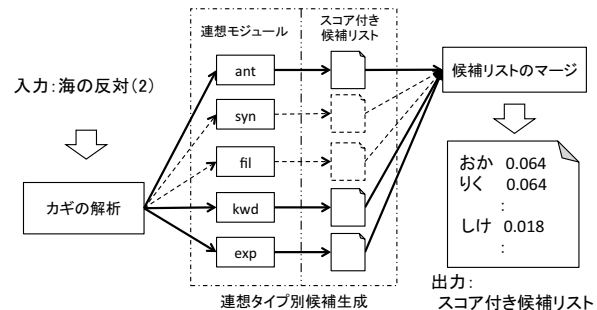


図 1: システム構成

表 1: 連想タイプ分類

連想タイプ	例
説明 : 対象を説明している	書物の厚み(2)
同義語 : 同義語が答えになるもの	いわゆる月賦(4)
反対語 : 反対語が答えになるもの	海の反対(2)
連想 : 実生活などから連想されるもの	ことこと煮込む時の火加減は(3)
推論 : なんらかの推論が必要なもの	「遊ぶ」の「ぶ」などを?(5)
穴埋 : カギに含まれる穴を埋めるもの	犬が□□向きや尾は東(2)
その他 : 上記以外のもの	ウィリアムですう~!(5)

表 2: 連想タイプと連想モジュール

連想タイプ	syn	ant	fil	kwd	exp
説明	△				
同義語	○				
反対語		○			
穴埋			○		

ストを作成する。表 2 に連想タイプと連想モジュールの対応表を示す。連想タイプ「説明」では、カギの最後が内容語で終わる場合に限り、syn も呼び出す。以下に、各連想モジュールの概要を示す。

- (1) syn(X) : 分類語彙表(101,070 語)を使用して同義語を探す。引数 X と分類番号上位 7 桁が一致する語を、答の候補とする。候補の数が 20 より少ない場合は、分類番号上位 5 桁が一致する語まで候補を広げる。得られた候補のスコアは、一致する分類番号の桁数に基づいて定める(桁数が多いものほど高い)。
- (2) ant(X) : 岩波国語辞典等から抽出・作成した反対語辞書(6,084 語)を用い、引数 X に対する反対語を返す。得られた候補は、すべて同一スコアとする。
- (3) fil(X) : はてなキーワード(318,886 語)、慣用句・ことわざ辞書(11,059 語)、岩波国語辞典 第 3 版(57,618 語)の見出し語を利用する。引数 X に読みを付与し(例: 笑う□□には福来る ⇒ わらう□□にはふくきたる)、各辞書の読みと照合する。一致した□の部分を含め答の候補とする。得られた候補のスコアは、一致した前後の文字列の長さに基づいて定める。
- (4) kwd(X) : 岩波国語辞典 第 3 版(57,618 語)、EDR 日本語単語辞書(194,019 語)、日本語 Wikipedia(801,404 語)を用いる。引数 X で渡されるカギを Mecab+Unidic で形態素解析し、キー

[§]カギの後ろの数字は、答えの文字数を示す

表 3: 実験結果

モジュール	呼ばれる回数	候補を出力した回数	上位 N 位までに含まれる正解数						候補リストの平均長
			1	3	5	10	50	∞	
<i>all</i>	499	499	85 18%	136 27%	162 33%	195 40%	263 53%	351 70%	2273
<i>kwd</i>	499	499	53 10%	97 16%	120 24%	149 30%	216 43%	313 63%	2190
<i>syn/tail</i>	237	221	20 9%	35 16%	41 19%	51 23%	53 24%	55 25%	67
<i>exp</i>	499	38	19 50%	22 58%	22 58%	22 58%	22 58%	22 58%	1
<i>fil</i>	156	135	59 44%	67 50%	70 52%	71 53%	74 55%	74 55%	265
<i>syn</i>	23	22	9 41%	11 50%	12 54%	14 59%	14 64%	14 64%	44
<i>ant</i>	7	6	6 100%	6 100%	6 100%	6 100%	6 100%	6 100%	1
<i>all w/o weight</i>	499	499	83 16%	132 26%	154 30%	192 38%	259 52%	351 70%	2273

ワード (内容語) を抽出する。それらのキーワードで辞書の定義文を検索し、その見出し語を答の候補とする。候補のスコアは、定義文中に含まれるキーワードの TF-IDF 値の和とする。

- (5) $\text{exp}(X)$: 岩波国語辞典 第3版 (57,618 語) を用いる。引数 X が文字列としてそのまま含まれる定義文を見つけ、その見出し語を答の候補とする。得られた候補は、すべて同一スコアとする。

2.3 候補リストのマージ

本ステップでは、ステップ 2 で得られた複数の候補リストを 1 つの候補リストに集約する。答の候補 (語) を w_i 、候補リストを C^j 、候補リスト C^j におけるある候補 w_i のスコアを s_i^j とするとき、候補 w_i の最終スコア s_i を次の式で計算する。

$$s_i = c \sum_j k^j s_i^j \quad (1)$$

ここで、 c は、集約された候補リストのスコアの総計を 1 とするための正規化係数、 k^j は、候補リスト C^j のスコアの最大値を \max_j とするための係数である。この \max_j は、連想モジュールの信頼度に相当し、現在は、*ant*、*syn*、*exp* に対して 20、*fil*、*kwd* に対して 15、連想タイプ「説明」で呼び出される *syn* に対して 10 を用いている。

3. 評価実験および結果

作成したシステムに「カギ」と「答の文字数」を与え、出力されたスコア付き候補リストの中に、正解が含まれるかどうかを調べた。本実験では、テストセットとして、前述の評価用パズル 12 問 (499 個のカギ) を用いた。

実験結果を表 3 に示す。この表の先頭行 (*all*) が、システムの性能を表している。本システムは、カギ 499 個中 351 個 (70%) に対し、正解を含む候補リストを出力した。この性能は、アメリカ式クロスワードを対象とした Proverb [1] よりも 25% 低い、日本語クロスワードを対象とした過去の研究 [4] より高い。なお、文献 [4] と同じテストセットを用いた場合は 74% となり、文献 [4] の 63% よりも 11% 高い結果となった。

候補リストの 1 位が正解である割合は 18% である。グリッドフィリングで正しい組合せを得るためには、候

補リスト中に正解ができるだけ上位に位置することが望ましい。正解の順位を上げるために、各連想モジュールでのスコア計算の再検討や、連想モジュールの信頼度に相当する \max_j の調整が、今後、必要である。

表 3 の *kwd* から *ant* までの行は、それぞれの連想モジュールの性能を示している。ここで、*syn/tail* は、連想タイプ「説明」で呼び出される *syn* を表す。本システムが出力した正解 351 個のうち、313 個は *kwd* が出力した候補リストに含まれる。*kwd* は、すべてのカギに対して呼び出されるため、その性能はシステム全体の性能を大きく左右する。今後、*kwd* モジュールが正解を出力する割合、および、正解を 1 位として出力する割合の両方を向上させる必要がある。

fil は、穴埋タイプのカギを解くための専用モジュールである。穴埋タイプのカギは 499 個中 156 個 (31%) と比較的数が多く、かつ、*kwd* では解けない場合が多いため、このモジュールは、2 番目に重要なモジュールである。現在の性能は、156 回の呼び出しに対し正解を出力した回数は 74 回に留まっており、十分とは言えない。穴埋のカギと照合する表現を拡充することにより、性能の向上を図る必要がある。

表 3 の最後の行 (*all w/o weight*) は、候補リスト C^j のスコアの最大値 \max_j をすべて 1 とした場合の結果である。上位 N 位までに含まれる正解数の割合は、*all* の方がすべての N で 1-3% 高くなっており、 \max_j による重み付けに効果があることがわかる。現在は、 \max_j に暫定的な値を用いているが、今後、実験的に最適値を求めることにより、*all* の性能を改善できる可能性がある。

参考文献

- [1] Greg A. Keim, Noam Shazeer, Michael L. Littman, Sushant Agarwal, Catherine M. Cheves, Joseph Fitzgerald, Jason Grosland, Fan Jiang, Shannon Pollard, and Karl Weinmeister: "Proverb: The Probabilistic Cruciverbalist", In Proceedings of the Sixteenth National Conference on Artificial Intelligence, pp.710-717, 1999.
- [2] Noam M. Shazeer, Michael L. Littman, and Greg A. Keim: "Solving crossword puzzles as probabilistic constraint satisfaction", In Proceedings of the Sixteenth National Conference on Artificial Intelligence, pp.156-162, 1999.
- [3] 『クロスワード掲示版』, <http://www.calib.jp/pa/crossword>
- [4] 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.