語義曖昧性解消としてのかな漢字換言システムの開発

山本 和英　　三上 侑城
長岡技術科学大学
{yamamoto, mikami}@jnlp.org

1 語義曖昧性解消をツール化したい

自然言語処理において語義曖昧性解消問題は非常に重要なタスクである。おそらくこれは多くの研究者で共有されている認識であろう。実際に多くの語義曖昧性解消手法がこれまでに提案されており、一定の性能も得られているようだ。同時に、多くのタスクにおいて語義曖昧性を解消していないことが性能劣化を引き起こしている、と定型文の如く報告されている。しかし、以上の状況にも関わらず日本語において語義を自動で決定してくれる公開ツールは見かけない[1]。これはなぜなのか？

この疑問に対し我々は、システム的な理由だと考えた。仮に語義曖昧性解消を行うツールが単独であっても、それを使いこなすのは難しい。例えば、この単語は国語辞典の語義３ですよと言われてもこれを後続の処理にどう活かしていいのかよく分からない。また、形態素解析器との単語体系の整合性も避けられない。

これに対し、我々はこのような問題が起こらない語義曖昧性解消の部分タスクを考えた。これが本稿で述べる (テキスト中の) かな漢字換言問題である。日本語テキスト中に書かれたひらがなの単語は、該当する漢字が複数ある場合にこのうちのどれかを同定することは、そのひらがな単語の語義曖昧性を解消することに相当する。また、語義３のような出力を行うのでなくひらがな単語を漢字単語に換言するだけなので後続の処理で語義に関する言語資源は不要である。さらに、この処理は結果としてひらがな語と漢字語の表記ゆれ解消にもなっている。まとめると、本タスクは語義曖昧性解消、換言処理、表記ゆれ解消のいずれにも該当する。

本研究の主眼は、語義曖昧性解消のツール化であり、新手法の提案ではない。我々は、高精度で曖昧性解消できない処理は行わない (複数の可能性を併記する) ほうが有益だと考えており、本研究においても高精度で換言可能な表現のみを処理対象とした。また、ツールの管理上の理由でシンプルな機構にした。これらはいずれも軽視されがちであるが、本当に使えるツールを作るためにはこのような判断は重要だと信じている。

実装は (Yamamoto et al. 2015)[4] による日本語解析システム「雪だるま」の単語解析部に組み込んだ。

[1] 日本語読解学習支援システム「あすなろ」[1] では語義を推測して語義の表示順を変更しているようである。また日本語以外ではいくつか知られている [2][3]。

2 かな漢字換言の手法

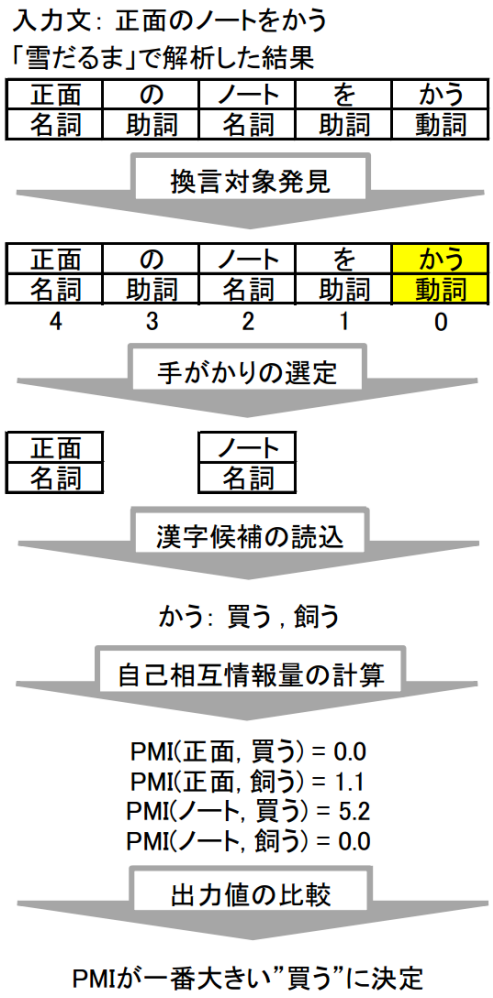

図 1: かな漢字換言器の換言過程

ここでは、作成したかな漢字換言器の換言手法について述べる。

漢字の換言候補が 2 つ以上ある場合には図 1 に示す手順でかな漢字換言をおこなう。まず、かな漢字テーブルに含まれるひらがなが入力文にあるかを判断する。対象のひらがなを発見した場合、ひらがなの周辺の語から品詞により手がかりを選定する。その後、漢字の換言候補を用意し、選定した語と漢字の全ての組み合わせで自己相互情報量を計算し、一番高い値のものの漢字を出力する。

また、かな漢字テーブルに含まれる漢字の候補が 1 つのみの場合は、入力文にそのひらがなが出現した時にそのまま候補の漢字を出力する。

次項から、この手法についての詳細を述べる。

2.1 換言候補の収集

本稿で作成するかな漢字換言器は、「雪だるま」(1) の単語解析結果を使用する。雪だるまは単語を ID で管理しており、また単語は表記統制辞書を用いて表記ゆれが吸収されている。すなわち表記揺れが吸収された ID から換言候補の収集を行い、ひらがなと漢字の対応表を作成する。

収集方法として、表記統制辞書中に存在するひらがなを全て列挙し、それらのひらがなと同じ品詞と読みの漢字をそれぞれに対応させた。品詞については、動詞, 名詞, 副詞, 形容詞のものとした。通常、候補から選択する方式では、候補が少ないほうが正解率は高くなる。そこで、同義の漢字を複数個、換言候補にしないために、(Yamamoto et al. 2015)[5] の同義語が収集された同義語辞書を使用した。一つのひらがなに対し、換言候補の漢字が 2 つ以上あり、同じ同義語集合に属していた場合、片方にまとめあげをおこなう。まとめあげをおこなう際に残す漢字の基準として、現代日本語書き言葉均衡コーパス (以後 BCCWJ)(2) を使用し、1-gram における各漢字の頻度を取得後、頻度の高い漢字にまとめあげるようにした。ひらがなの出現頻度が非常に多いもの (頻度 10,000 以上のひらがな) については、ひらがな表現が正しいとし、そのひらがなは対応表には入れず、かな漢字換言を行わない。逆に出現頻度が極端に低いひらがな (頻度 20 未満のひらがな) 及び漢字 (頻度 30 未満の漢字) については、ノイズだとし、対象の語は全て除外した。この過程を図 2 に示す。

これらの工程より作成された表を、かな漢字テーブルとここでは呼び、表 1 にその一部を示す。なお、かな漢字テーブル内の漢字の候補が 2 つ以上あるものについては、品詞が動詞または名詞のみとなっている。今後かな漢字換言を行う際にはこのかな漢字テーブルにあるものだけを用いる。

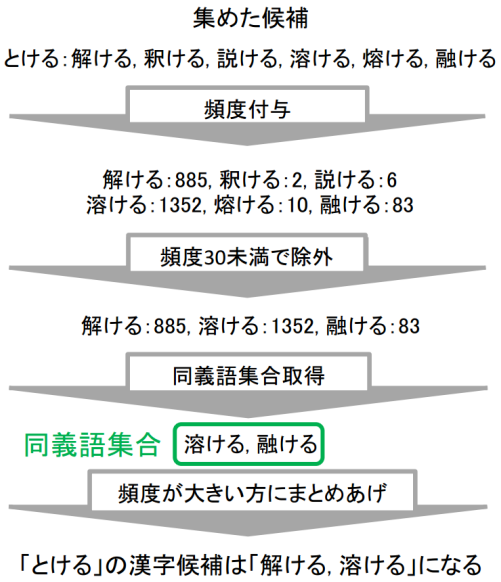

図 2: 換言候補の収集過程

表 1: かな漢字テーブルの一部

ひらがな	漢字 1	漢字 2	漢字 3
たつ	立つ	経つ	断つ
しめる	占める	閉める	湿る
とける	解ける	溶ける	-
かう	買う	飼う	-
きめる	決める	-	-
ことば	言葉	-	-

2.2 コーパスからの例文抽出

ある漢字に対して、周辺にはどのような語が頻出するかを取得するため、大量の例文が必要となる。そこで今回、Web 日本語 N グラム (第 1 版)(3) を使用する。一度、Web 日本語 N グラム中の 7-gram 全ての形態素列を結合し、雪だるまにより単語解析を行う。その解析結果から、かな漢字テーブル中の漢字が 1 つでも入っている文を抽出する。

2.3 手がかりとする語の対象

漢字とその周辺の語について、共起する組み合わせを知るために必要な周辺の語の対象に、助詞の「の」や「と」など、どの文にも頻出する語が存在すると、共起する組み合わせが正確に計算できない。そこで本稿では共起するとされている品詞のみを手がかりとして用いることにする。かな漢字換言する語の換言候補

が 2 つ以上あるもので動詞であった場合には名詞を、語が名詞であった場合には動詞及び名詞を手がかりとする語の対象にする。

また、手がかりとする語の範囲については、かな漢字換言する語の前後 4 語を対象とする。これらをまとめたものを図 3 に示す。

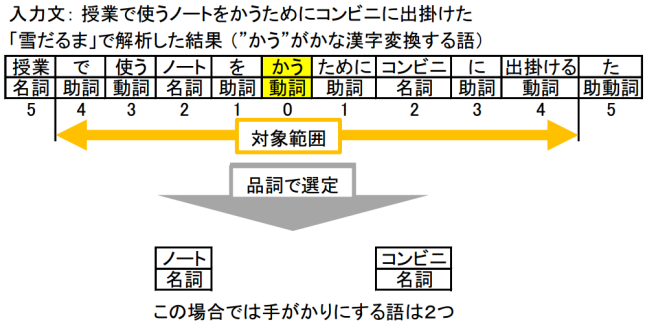

図 3: 手がかりとする語の選定方法

2.4 自己相互情報量の計算

例文抽出において述べた共起する組み合わせを数値として表すために、自己相互情報量 (PMI) を使用した。$p(x,y)$ は x と y が同時に出現する確率であり、$p(x)$ は x が出現する確率である。2 つの語 (x,y) における自己相互情報量の式は次のとおりである。

$$PMI(x,y) = log\frac{p(x,y)}{p(x)p(y)}$$

x と y にはそれぞれ対象の漢字と手がかりとなる語が入り、コーパスより抽出した例文から自己相互情報量を計算する。その結果相互情報量が一番高いものを、かな漢字換言の結果として採用する。

2.5 自己相互情報量での閾値

自己相互情報量の計算をおこなった時、強く手がかりとなる語がない場合、全体的に計算結果の値は小さくなり、正解率は低い。そこで、手がかりとなる語が存在する時のみ換言を行うとする。そのために自己相互情報量の計算結果が、ある閾値より高い場合のみ換言を行う。はじめにで書いたように、このかな漢字換言器は網羅性よりも正確さ (正解率) を優先しているため、正解率が低いものに関しては切り捨てる方針を取る。図 4 と図 5 は自己相互情報量の値による、かな漢字換言の正解数・間違数を記録したグラフである。BCCWJ の中からランダムにかな漢字換言したものを、ひらがなが動詞か名詞で各 100 づつ手作業で正解・不正解を判断し、更に単語解析誤りや人でも判断がつかないものは除外した時の自己相互情報量の値を記録した。2 つを見比べると、全体的に動詞のほうが正解率が高いことが分かる。ここで、正解が約 5 割残り、誤りを 1 割未満に抑えるために、自己相互情報量の閾値を 5 以上に設定した。これにより、5 未満のものについてはかな漢字換言をおこなわないようになる。

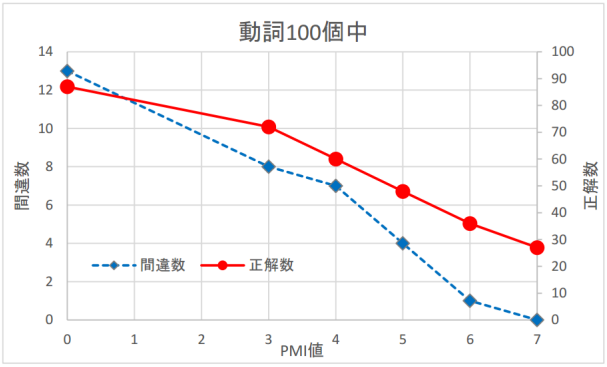

図 4: 自己相互情報量における正誤数 (動詞)

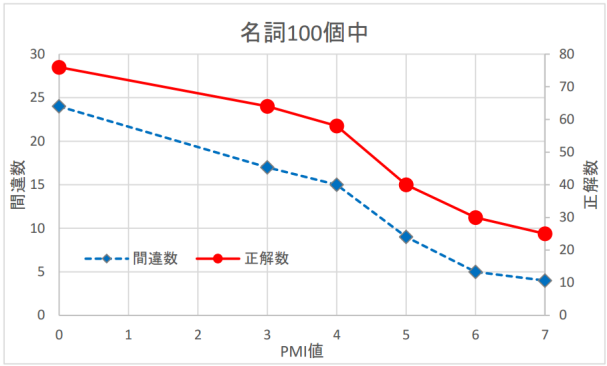

図 5: 自己相互情報量における正誤数 (名詞)

3 高精度に換言できないものの除外

高精度にかな漢字換言できないものは除外する。ここで判別をおこなうことで、さらなる精度向上が見込まれる。判別方法として、本手法を適用したかな漢字換言器を用意し、BCCWJ の中から対象のひらがなが入った文をランダムに取得して、かな漢字換言をおこない、その出力結果を見る。ただし前回同様、日本語解析の時点で間違っているもの、人でも判断がつかないものに関しては除外する。

まず漢字の候補が 1 つのみのものは、出力結果を 20 個中、9 割 (18 個以上) が正解していることを確認できたものを採用する。次に漢字の候補が複数あるものに関しては、出力結果を 20 個中、8 割 (16 個以上) が正解していることを確認できたものを採用することにした。またこの時、漢字候補に不備があるもの (主に

通常使われない漢字が残っているもの) は、手動でかな漢字テーブルを編集し、再度判別をおこなった。

4 実験と考察

前節の手作業での判別で出力した各 20 個に正解情報を付与したものをテストデータとし、作成したかな漢字換言器で出力をおこなった。この時の漢字候補が 1 つのみの場合と、2 つ以上ある場合に分けた出力結果を表 2 に示す。

表 2: かな漢字換言器の出力結果

漢字候補	対応数	テスト数	正解数	正解率
1 つのみ	52	1,040	1,031	99.1%
2 つ以上	71	1,420	1,336	94.1%
合計	123	24,60	2,367	96.2%

次に、本研究で対象とした単語の網羅性について議論する。表 3 に、ひらがなの頻度が 20 未満・20〜1 万 (かな漢字テーブル)・1 万以上の 3 種類 (内、20〜1 万は、漢字 2 つ以上のみでも記載) に分類した出現頻度を示す。

表 3: かな漢字テーブルの総頻度

測定箇所	20 未満	20〜1 万	1 万以上
総頻度	16,052	886,649	9,470,468
うち 2 つ以上	(不明)	316,190	(ほぼなし)

処理対象とした漢字候補一つのみのひらがな 52 単語は 67,809 回、漢字候補 2 つ以上のひらがな 71 単語は 80,962 回出現した。頻度 20 未満のひらがな語 16,052 語のうちどの程度漢字候補が 2 つ以上あるのかは調査していないため不明だが、仮にすべて多義性があり、逆に頻度 1 万以上は仮にすべて多義性がないと仮定すると、我々はひらがな多義語の 22.4%(=80k/(16k+316k)) を解いた計算になる。また、漢字候補一つのみの出力 (67k 回)、及び頻度 1 万以上の換言対象外 (9M 回) も処理対象に含めると、92.7%(=(67k+80k+9M)/(16k+886k+9M)) のひらがな語に対して対処済みであり、残りは全出現ひらがな語の 7.3%となる。

5 まとめ

日本語テキスト中に出現するひらがな語のうち、漢字に換言する形で語義曖昧性解消処理を日本語解析システム雪だるまに実装した。漢字が 2 語以上となる曖昧なひらがな語のうち出現頻度で 25%を占める 71 語に対して曖昧性を解消し、これらは 94%で適切に換言できることを確認した。機構は単純であり、(高精度に実装できると判断した) ひらがな語を追加していくだけで、実装済みのひらがな語の処理精度に全く影響を与えずに網羅性を上げていくことが可能である。この作業は、今後も継続的に進めていく。

使用したツールと言語資源

(1) 日本語解析システム「雪だるま」, (Yamamoto et al. 2015), 長岡技術科学大学 自然言語処理研究室, http://snowman.jnlp.org/

(2) 現代日本語書き言葉均衡コーパス (BCCWJ), Ver.1.1, 国立国語研究所

(3) Web 日本語 N グラム, 第 1 版, グーグル株式会社

謝辞

本研究は、平成 27〜31 年科学研究費補助金基盤 (B) 課題番号 15H03216 の助成を受けています。

参考文献

[1] 多言語対応日本語読解学習支援システム「あすなろ」https://hinoki-project.org/asunaro/index-j.php

[2] Word sense disambiguation resources. http://aclweb.org/aclwiki/index.php?title=Word_sense_disambiguation_resources

[3] UKB: Graph Based Word Sense Disambiguation and Similarity. http://ixa2.si.ehu.es/ukb/

[4] Kazuhide Yamamoto, Yuki Miyanishi, Kanji Takahashi, Yoshiki Inomata, Yuki Mikami and Yuta Sudo. What We Need is Word, Not Morpheme; Constructing Word Analyzer for Japanese. Proceedings of the International Conference on Asian Language Processing (IALP 2015), pp.49-52 (2015.10)

[5] Kazuhide Yamamoto and Kanji Takahashi. Construction of Japanese Semantically Compatible Words Resource. Proceedings of the International Conference on Asian Language Processing (IALP 2015), pp.61-64 (2015.10)