アナグラム生成における文節列の意味的適格性の判定法の検討

鈴木 啓輔　　佐藤 理史　　駒谷 和範
名古屋大学大学院 工学研究科 電子情報システム専攻
{kei_suzu, ssato, komatani}@nuee.nagoya-u.ac.jp

1　はじめに

ことば遊びの一つにアナグラムがある。ことば遊びの事典である『図説ことばあそび遊辞苑』[1]では、アナグラムを「ある語句のつづりを一字ずつに分解し、そのすべてを使って、まったく別の語句に仕立てるもの(p103)」と説明している。たとえば、「名古屋大学(なごやだいがく)」という語句からは、「長(なが)い訳語(やくご)だ」というアナグラムが構成できる。

日本語アナグラムを自動生成する試みは、これまでにも存在した[2][3]が、いずれも生成能力に大きな制限があった。これに対し、我々は、昨年、文字の並べ替えの制約と文法的適格性の制約を満たすアナグラム候補を、網羅的に生成するアルゴリズムを提案した[4]。しかしながら、最終的に生成すべきアナグラムは、意味が通る（意味的に適格である）必要があり、その判定法は、未解決の問題として残っていた。本論文では、この未解決の問題、すなわち、ある表現が意味的に適格かどうかを判定する一つの方法を示す。

本研究の目的は、アナグラムの自動生成法を明らかにすることであるが、それは、同時に、「文法的に適格で、かつ、意味が通じる日本語表現を生成する」方法を明らかにすることでもある。ある表現の意味的適格性を自動判定する方法については、チョムスキー以来、長い研究の歴史があるが、いまだに十分に解明されているとは言いがたい。

以下、まず、2節で、昨年提案したアナグラム候補の生成法の概略を示す。3節では、新たに導入する意味的適格性の判定法について、4節でその性能を検証する実験について述べる。

2　アナグラム候補の網羅的生成

2.1　アナグラムが満たすべき制約

まず、準備として、日本語表現を、表記(文字列)Jとその読み(かな文字列)K の対で表すものと約束する。ある日本語表現 $\langle J_s, K_s\rangle$ のアナグラム $\langle J_a, K_a\rangle$ とは、以下の制約をすべて満たす表現である。

制約 **1**　K_a は、K_s の並び替えである (並べ替え制約)
制約 **2**　$\langle J_a, K_a\rangle$ は、文法的に適格である
制約 **3**　$\langle J_a, K_a\rangle$ は、意味的に適格である

2.2　アナグラム候補生成法

アナグラム候補生成では、入力として、ある日本語表現 $\langle J_s, K_s\rangle$ の他に、文節集合 B を与える。ここで、文節集合 B に含まれる文節 b は、その表記と読みに加え、文節タイプを持つ。この文節タイプは、文法的適格性の判定に使用する。このような文節集合 B を与える理由は、アナグラム候補となりうる表現の全体集合を規定するためである。すなわち、生成するアナグラム候補は、文節集合 B に含まれる文節の列 $b_1 \cdots b_n$ に限定する。このように可能な表現の全体集合を規定することにより、アナグラム候補の網羅的生成が可能となる。

2.2.1　並べ替え制約を満たす文節列の生成

与えられた文字列 K_s から、文字を並び替えた文字列 K_a を生成することはたやすい。すなわち、文字列 K_s から文字を1つずつ適当な順番で抜き出し、それを並べて文字列 K_a を構成すればよい。この方法ですべての可能な場合を尽くせば、並び替え文字列の網羅的生成が実現できる。

得られた並び替え文字列 K_a に対し、文節の列が構成できるかどうかの判定も、単なる探索問題である。このような判定を行ない、文節列を構成できた K_a に対して、その文節列を出力すれば、並び替え制約を満たす文節列が網羅的に生成できる。

2.2.2　文法的適格性の判定

アナグラム候補生成において、我々は、文節列の文法的適格性の判定基準として、「文節係り受け木を構成できる文節列は、文法的に適格である」という条件を採用した。すなわち、アナグラム候補は、文節係り受け木として出力する。

文節係り受け木は、末尾の文節から先頭の文節に向

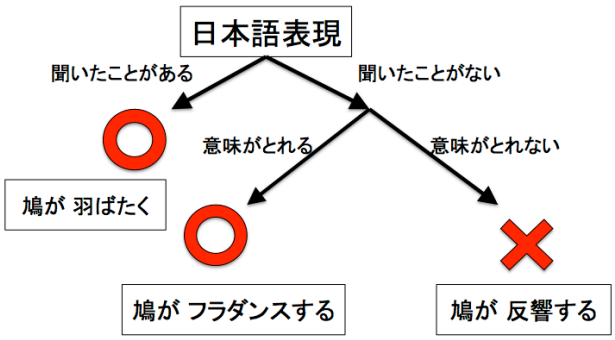

図 1: 表現の意味的整合性判定

かって、1 文節ずつ係り受け解析木に追加していく手法をとる。どの文節が係り先となり得るかは、各文節に付与されている文節タイプに基づいて決定する。現在は、可能な係り先のうち、もっとも近い係り先を選択する方法を採用している。

3 意味的適格性の判定

あるアナグラム候補 (文節係り受け木) が意味の通る日本語となっているためには、すくなくとも、個々の係り受けのすべてが、意味的に整合している必要がある。たとえば、「社会的損失」のアナグラム「起訴して勝つ社員」の意味が通るのは、⟨ 起訴して, 勝つ ⟩ と ⟨ 勝つ, 社員 ⟩ がともに、意味的に整合していることが、その前提条件となっている。このような考え方に基づき、我々は、意味的適格性の条件として、次の条件を採用した。

> アナグラム候補 (文節係り受け木) に含まれる
> 文節の係り受けの全てが、意味的に整合する。

このような条件の採用により、我々が解くべき問題は、ある文節係り受けが意味的に整合するかどうかをどのように判定するか、という問題となる。

3.1 アイディア

我々人間は、ある表現が意味的に整合するどうかを判断する際、過去に同じ表現を聞いたことがあるかどうかを判断基準とする。もし、その表現を聞いたことがあれば、「表現としてありうる (≈ 意味的に整合する)」と判断する。その一方で、聞いたことがない表現であっても、すぐに、「表現としてあり得ない」という判断を下すわけではない。聞いたことがなくても、「意味がとれる」表現もある (図 1)。

たとえば、⟨ 鳩が, フラダンスする ⟩ という表現を考えよう。おそらく、多くの人は、この表現に接するのは初めてであろう。それにも関わらず、ほとんどの人は、「この表現は意味がとれる」と判断するだろう。その一方で、⟨ 鳩が, 反響する ⟩ という表現に対しては、「おかしな (意味的に整合しない) 表現」と感じるだろう。

上記の現象を、我々は次のように説明する。我々人間は、未知の表現に遭遇した場合、それが意味的に整合するかどうかを、既知の表現との柔軟な照合によって判断する。たとえば、⟨ 鳩が, フラダンスする ⟩ を聞いたことがない場合、「鳩」は「鳥」で、「フラダンスする」は「踊る」だから、この表現は、⟨ 鳥が, 踊る ⟩ というレベルで、理解可能である (つまり、意味的に整合する)。一方、⟨ 鳩が, 反響する ⟩ は、⟨ 鳥が, 響く ⟩ であり、「響く」のは「音」だから、何かおかしい (つまり、意味的に整合しない) と判断する。

ここでの説明のポイントは、未知の表現を適切なレベルに抽象化して考えるというところである。このような抽象化が、未知の表現であっても、その意味的整合性を判定できる能力の源となっている。

3.2 実現法

上記のアイディアを近似的に実現する方法を考える。なお、ここでは、判定対象が文節係り受け $\langle b_c, b_p \rangle$ であることを前提する。

まず、図 1 の第 1 の分岐の「聞いたことがあるか」という判定を、「コーパス中に、$\langle b_c, b_p \rangle$ と一致する係り受けが存在するかどうか」で判定する。巨大かつ適切なコーパスを用意すれば、このような実現法は、「聞いたことがあるかどうか」という判定を、かなりよく近似すると考えられる。

第 2 の分岐の実現のためには、判定対象の文節係り受け $\langle b_c, b_p \rangle$ を、適切に抽象化する必要がある。ここでは、文節 b_c と b_p に含まれる内容語を、より一般的な上位語または類語に置き換える方法を採用する。これを**文節の一般化**と呼ぶ。すなわち、文節係り受け $\langle b_c, b_p \rangle$ に対して、それを一般化した $\langle \widehat{b_c}, \widehat{b_p} \rangle$ を作成する。こうして得た一般化された係り受けがコーパス中に存在するかどうかを調べ、存在した場合に「意味的に整合する」と判定する。

上記の説明からわかるように、コーパスを、ある特定の係り受けが存在するかどうかを調べるために用いる。そのため、あらかじめ、コーパスを係り受けデータベースに変換しておくとよい。

以上の方法をまとめた概略図を図 2 に示す。この実現法を実装するためには、係り受けデータベース、および、文節の一般化の具体的方法が必要となる。

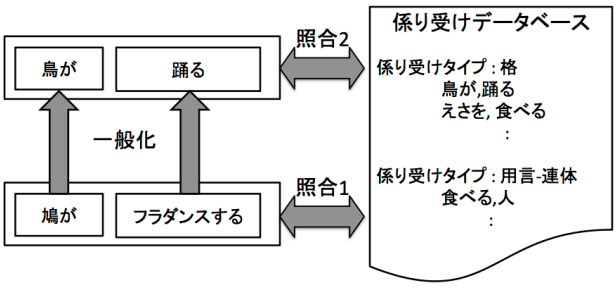

図 2: 係り受けの意味的整合性判定の概略図

表 1: 係り受けタイプ一覧

係り受けタイプ	係り文節	受け文節
格	体言 (+格助詞・係助詞)	用言
用言-連用	用言	用言
用言-連体	用言	体言
連用	副詞	用言
連体	連体詞	体言
の	体言＋の	体言

4 係り受けデータベースの作成

日本語の係り受けには、いくつかの種類が考えられるが、今回は、表 1 に示す 6 種類を採用し、これらのタイプ毎にデータを収集した。それぞれの係り受けデータは、係り受けタイプ、係り文節、受け文節の 3 つ組から構成する。係り文節および受け文節の内容語は、基本形に正規化する。このため、例えば「走った人」や「走る人」は、⟨ 用言-連体, 走る, 人 ⟩ という 1 つのデータに集約される。

係り受けタイプが「格」の係り受けデータは、KNP3.0 に付属する京都大学格フレーム辞書に定義された格関係のデータをそのまま利用した。その他の係り受けタイプのデータは、テキストコーパス (毎日新聞データ 1991-2005 年と日本語均衡コーパスの 2009 年度モニター公開版) から係り受けを抽出することで作成した[1]。最終的に、係り受けデータは 9,218,161 件となった。

5 文節の一般化

すでに説明したように、文節の一般化は、文節 b に含まれる内容語 w を、上位語 (あるいは類語) $\widehat{w}$ に置き換えることを意味する。これを実現するために、我々は、シソーラスと内容語の頻度を利用する。

シソーラスには、分類語彙表[5]を用いた。分類語彙表は、木構造シソーラスで、その葉ノードに、語が定義されており、より上位の階層である「小段落、段落、分類項目、中項目、···」は、類語グループを表す。

[1]連続する文節間の係り受けのみを抽出対象とした。

シソーラスを用いて、内容語 w からそれを一般化した $\widehat{w}$ を得るためには、(a) w から木構造の階層をどこまで遡り、(b) その階層で定義されている類語グループのなかからどのように $\widehat{w}$ を選ぶか、を定めればよい。これらを定めるために、我々は、語の頻度を利用する[2]。すなわち、w の類語のうち、頻度 $f(w)$ がしきい値 t を越える語を $\widehat{w}$ として採用する。

具体的には、次の手順で $\widehat{w}$ を定める。

1. w が、シソーラスの項目に存在していないなら、$\widehat{w} = null$ として終了 (一般化失敗)。
2. w の頻度 $f(w)$ が t 以上なら、$\widehat{w} = w$ として終了。
3. w が含まれる小段落の類語グループの中に、頻度が t 以上の語があれば、その語を $\widehat{w}$ として終了。なければ、次のステップへ進む。
4. 類語グループの階層を小段落から、段落、分類項目、中項目と順に上げて、ステップ 3 を実行。
5. 中項目まで階層を上げても一般化先を決定できなければ、$\widehat{w} = null$ とする (一般化失敗)。

なお、w が複数の意味を持つ場合、分類語彙表にその意味の数だけ項目が登録されているが、そのような場合は、その意味ごとに、$\widehat{w}$ を求める。

このように実現される文節の一般化を用いて、最終的に、係り受け $\langle b_c, b_p \rangle$ に対して、3 種類の一般化された係り受け $\langle \widehat{b_c}, b_p \rangle$, $\langle b_c, \widehat{b_p} \rangle$, $\langle \widehat{b_c}, \widehat{b_p} \rangle$ を作成する。文節 b_c や b_p に複数の一般化先がある場合は、その全てに対し、これら 3 種の係り受けを生成する。図 2 の照合 2 では、これらのすべてを用い、そのうちの 1 つでもデータベースに一致するものがあれば、意味的に適格であると判定する。

6 意味的適格性判定の性能実験

6.1 使用するテストセット

論文[4]で使用した 6-11 文字のかな文字列 39 個に対して、それぞれアナグラム候補生成を行い、39 のアナグラム候補リストを得た。ただし、その候補数が 100 を越えるものについては、上位 100 件のみを残した[3]。

この 39 のリストから、意味が通じるアナグラム候補を含む 19 のリストを人手で選び出した。その 19 のリスト (アナグラム候補数 1500) をこの実験のテストセットとする。このテストセットは、合計 49 件の意味が通じるアナグラム候補を含む。

[2]語 w の頻度 $f(w)$ は、毎日新聞データ 1991-2005 年と日本語均衡コーパスの 2009 年度モニター公開版における w の出現数を利用した。

[3]順位付けには、論文[4]に記載した手法を用いた。

表 2: テストセットに対する意味的適格性の判定結果

	判定前	直接法	提案手法 (照合 1+照合 2)			
閾値	-	-	1760	8907	25707	46289
出力数	1500	20	101	204	245	296
正解数	49	2	6	13	17	15
再現率	100%	4%	12%	27%	35%	31%
適合率	3%	10%	6%	6%	7%	5%
F 値	0.06	0.06	0.08	0.10	0.12	0.09

表 3: 提案手法 ($t = 25707$) の正解出力

広大な土地*	社員嫌でした	獣医のヒントだよ
審査もある*	気付く官僚	汚い各社査定
勝因麻痺か	官僚気付く	最適な貸借か
新婚急く	完了気付く	起訴して勝つ 社員
急く新婚	奥の飴かい	裁定した客かな
猿も思案	蟹狩る医者	

6.2 実験結果

前述のテストセットに含まれるアナグラム候補に対して、提案手法により意味的適格性を判定した結果を表 2 に示す。一般化のしきい値 t には、1760, 8907, 25707, 46289 を用い、それぞれのしきい値ごとに意味的適格性の判定を行った。これらのしきい値 t は、使用したコーパスにおいて、t を越える出現数を持つ形態素の総数が、コーパスに含まれる形態素の総数の 90%、75%、60%、50%となる値である。

なお、表 2 には、提案手法との比較のために、図 2 の照合 1 のみで判定 (これを直接法とよぶ) した結果も合わせて示した。

理想的には、再現率を保ったまま、適合率を向上させることが望ましい。表 2 より、直接法では、適合率は向上するものの、再現率が 4%と極めて小さい。一方、提案手法では、適合率は直接法よりは劣るものの判定前より向上し、再現率は最大で 35%と直接法より大きく向上した。

提案手法 ($t = 25707$) で得られた正解の一覧を表 3 に示す。ここで、 表 3 の*をつけたアナグラム候補は直接法における正解出力を示す。提案手法では、「獣医のヒントだよ」などの、係り受けデータベースに存在しない係り受けを含む正解を出力できていることが分かる。このことは、未知の表現が意味的に適格かどうかを、提案手法で判定できる可能性があることを示唆している。

6.3 再現率が低い原因の分析

提案手法は、直接法より再現率が向上した。しかしながら、再現率が最大である $t = 25707$ の提案手法でも、その値は 35%と低い。

その原因を探るため、意味が通じるアナグラム候補のうち、判定を通過しなかった 32 件から無作為に 5 件抽出し、どのような一般化がなされたかを調査した。

表 4: アナグラム候補の一般化結果

アナグラム候補	内包する内容語	一般化した内容語	一般化は適切か
新米 評価	新米	新人, 料理	○
	評価	評価	○
社宅 適さないかい	社宅	マンション	○
	適す	一致, 不良だ	×
女に 泣く 重役	女	女性	○
	泣く	求める	×
	重役	常務	○
釈迦に 怒る	釈迦	それぞれ, 患者	×
	怒る	求める	×
烏賊 煮る 釈迦	烏賊	蚤	×
	煮る	煎る	○
	釈迦	それぞれ, 患者	×

その結果を表 4 に示す。

表 4 から、現在の意味的適格性判定の実装には、以下のような問題点があることが分かる。

1. 一般化の際に、木構造の階層を遡りすぎる語がある。
   例えば、「釈迦」は、上位語でも類語でもない「それぞれ」や「患者」に一般化された。これは、不適切な一般化であるといえる。一般化の際に木構造の階層を遡りすぎると、類語と見なす範囲が広くなりすぎて、このような現象が起こる。
2. 適切に一般化された場合でも、係り受けデータベースと一致しないことがある。
   ⟨ 新米, 評価 ⟩ を一般化した ⟨ 新人, 評価 ⟩ は、比較的一般的な表現でありながら、係り受けデータベースと一致しない。これは、係り受けデータベースの大きさが不十分であることが原因と考えられる。

今後は、より適切に一般化する手法の検討、係り受けデータベースの拡大などに取り組む必要がある。

謝辞 本研究では、CD-毎日新聞データ集 (1991 版-2005 版)、および、国立国語研究所が開発した現代日本語書き言葉均衡コーパス 2009 年度モニター公開データ、分類語彙表-増補改訂版-を使用した。

参考文献

[1] 荻生待也 (編): 図説 ことばあそび遊辞苑, 遊子館 (2007).

[2] stabucky: アナグラム自動生成, http://tool.stabucky.com/anagram.php.

[3] 小野智司, 村山大介, 澤井陽輔, 中山茂: 進化計算法を用いたアナグラム文の生成, 人工知能学会第 90 回知識ベースシステム研究会, 人工知能学会研究会資料 SIG-KBS-B001, pp.17-22, (2010).

[4] 鈴木啓輔, 佐藤理史, 駒谷和範: 文節データベースを用いた日本語アナグラムの自動生成, 第 10 回情報科学技術フォーラム, (2011).

[5] 国立国語研究所：分類語彙表-増補改訂版-，大日本図書，2004.