文頭固定法による効率的な回文生成

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

回文は、平安時代から存在する伝統的ことばあそびである。「図説 ことばあそび遊辞苑」[1] では回文を「頭 (かしら) から読んでも尻から読んでも同じ音で、どちらも無理なく意味が通じる語句や文章 (p301)」と説明している。この説明を以下の 2 つの条件に分ける。

1. 頭 (かしら) から読んでも尻から読んでも同じ音であること
2. 無理なく意味が通じること

本論文では、1 つ目の条件を回文条件、2 つ目の条件を通意条件と呼ぶことにする。回文はこの 2 つの条件を満たした語句や文章である。

日本語の回文を作成することは、日本語を母国語とする我々でも比較的難易度が高く、「サンダルが上がる段差」といった読みが 10 文字程度の回文となると、即座に思いつくのは困難である。

我々は、このような作成難度の高い回文を自動生成することに挑戦している。回文の自動生成を以下の 2 つのステップで実現する。

step1　回文条件を満たす文節列 (以降、回文候補と呼ぶ) を大量に生成
step2　通意条件を満たす回文候補を選別

昨年、我々は step1 を実現するアルゴリズムを提案し、実際に、350 万の文節集合 (以降 D と記す) から、3 文節、4 文節の回文を網羅的に生成した [2]。生成に要した時間は、3 文節の場合で 1 日弱、4 文節では 4ヶ月半であった。

我々の最終目標は、種となる 1 文節 (以降シード文節) からリアルタイムで回文を生成することである。本稿では、シード文節からリアルタイムで 3 文節の回文候補を生成できるように、回文候補の生成の効率化に取り組んだ。

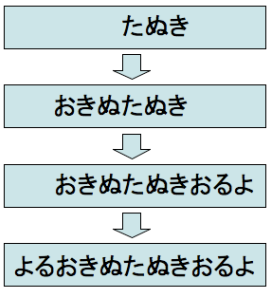

図 1: 折返し固定法

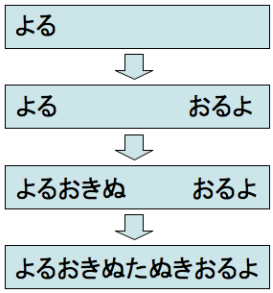

図 2: 文頭固定法

2　回文候補生成法

本節では、昨年提案した折返し固定法[2] と、本稿で提案する文頭固定法について述べる。いずれの方法も、シード文節に文節集合 D 中の文節を結合して回文候補を生成する。

2.1　折返し固定法

折返し固定法は、シード文節を「折返し点を含む文節」とする手法である。折返し固定法による「夜起きぬタヌキおるよ」の生成手順の概略を図 1 に示す。

図 1 のシード文節は「たぬき」である。まず、「た」を折返し点として「たぬき」を折り返す。すると、回文条件を満たすためには「たぬき」の左側に、「きぬ」という読みがこなければならない。この、「きぬ」のように回文条件を満たすために足りない文字列を以降不足文字列と呼び、□(き)□(ぬ)のように書くこととする。不足文字列「□(き)□(ぬ)」を補うために、タヌキの左側に「おきぬ」を結合し「おきぬたぬき」とする。同様に、「たぬき」の右側に「おるよ」、「おきぬ」の左側に「よる」を結合する。この時点の状態「よるおきぬたぬきおるよ」は回文条件を満たす文節列であり、回文候補として出力する。

実際は、上記の手順を探索として定義する。$<$ おきぬたぬき, □(お), R $>$ のように、文節列とその不足文字列、不足文字列が付くべき場所の 3 つ組を状態、文節列に文節を結合する行為をオペレータとして定義す

る。深さ制限探索によって、シード文節に D 中の文節を結合して生成可能な n 文節の回文候補を網羅的に求める。

2.2 文頭固定法

本論文で提案する文頭固定法では、シード文節を「文頭の文節」とする手法である。

文頭固定法による「夜起きぬタヌキおるよ」の生成手順の概略を図 2 に示す。

図 1 の折返し固定法では、シード文節は「たぬき」であったのに対し、図 2 の文頭固定法では文頭の文節の「よる」である。「よる」が文頭の文節であるので、回文条件を満たすためには文末が「るよ」でなければならない。そこで、文末の文節を「おるよ」とする[1]。文末に「おるよ」がくると、「よる」の右側が「お」である必要がある。そこで「よる」の右側に「おきぬ」を結合する。同様に、「おるよ」の左側に「たぬき」を結合する。この時点の状態「よるおきぬたぬきおるよ」は回文条件を満たす文節列であり、回文候補として出力する。

文頭固定法も、折返し固定法と同様に、上記の手順を探索として定義し、深さ制限探索によって、シード文節に D 中の文節を結合して生成可能な n 文節の回文を網羅的に求める。

折返し固定法も文頭固定法も、D の全項目をシード文節として n 文節の回文候補を生成すれば、D から生成可能な n 文節の回文候補を網羅的に生成できる。

2.3 速度比較実験

折返し固定法と文頭固定法の回文候補生成速度の違いを比べるために、同一の D を用いて、3 文節、4 文節の回文候補を網羅的に生成し、その生成時間を比較した。

結果を表 1 に示す[2]。3 文節の場合、折返し固定法では生成に 1 日弱かかっていたが、文頭固定法では、1 時間未満で生成が可能であった。4 文節の場合でも、折返し固定法では 6ヶ月半以上かかるのに対し、文頭固定法では、2 週間半程度で生成できた[3]。

[1]「よるおるよ」の時点で回文条件を満たす文節列となっているので、「夜おるよ」も回文候補として出力可能である。

[2]折返し固定法での 4 文節の網羅的生成は、1/10 のサイズのシード文節で実験した。表 1 の該当項目は、1/10 サイズの実験にかかった時間を単純に 10 倍して記載した。

[3]文献 [2] の実験では文法的に適切かどうかのフィルタをかけていたが、今回は 2 手法の探索能力の比較をするためにそのフィルタをかけていない。そのため 1 節で記載した生成時間と表 1 の折返し法の生成時間は異なる。

表 1: 2 手法の速度比較

文節数	折返し固定法	文頭固定法
3	21 時間 41 分	42 分
4	198 日 20 時間 34 分	17 日 14 時間 10 分

回文候補の網羅的生成は並列処理が可能である[4]。実際には、4 文節の回文候補は 10 並列で実行したため、折返し固定法の場合は、2 日程度で完了した。

2.4 検討

前節の実験で、文頭固定法の方が折返し固定法より高速であることが示された。これは、文頭固定法の方が折返し固定法に比べ探索空間が小さいからである。表 2 に、3 文節の回文候補の網羅的生成に必要な探索空間の状態数を示す。この表に示すように、文頭固定法の状態数は、折返し固定法の状態数に比べて、初期状態 ($n = 0$) で約 1/4、深さ $n = 1$ の状態で約 1/90 となっている。

このように、探索空間の状態数に大きな差が生じるのは、次の 2 つの理由による。

理由 1 文頭固定法は、シード文節から出現する初期状態数が少ない

文頭固定法では、1 つのシード文節から 1 つの初期状態を生成する。これに対して、折返し固定法では、1 つのシード文節から、可能な折返し点の数だけ初期状態を生成する。読みが 2 文字以上の文節では、少なくとも、「両端の文字」と「両端の文字境界」の 4ヶ所で折り返すことができる[5]。このことが、表 2 の初期状態 ($n = 0$) の差を生み出している。

理由 2 文頭固定法は、不足文字列の短い初期状態が出現しにくい

不足文字列が短い状態は、不足文字列が長い状態と比べて分岐度が大きい (接続可能な文節の数が多い)。このため、不足文字列の短い初期状態が出現しやすいかどうかは、探索空間のサイズに大きな影響を与える。

ここでは、両手法において、不足文字列の長さが 1 の初期状態がどのくらい生成されるかに注目する。文頭固定法では、シード文節の読みの長さが 1 のときのみ、このような初期状態が 1 つ生成される。これに対して、折返し固定法では、シード文節の読みの長さが 1 つの場合は 2 つ、読みの長さが 2 の場合は 2 つ生成

[4]D を n 分割し、分割した各々をシード文節として回文候補を生成すれば、n 並列処理となる。

[5]読みが 1 文字の文節の場合、折返し点は両端の文字境界の 2 つのみである

表 2: 各文節長における状態の数

深さ n	折返し固定法	文頭固定法
0	1.4×10^7	3.5×10^6
1	1.4×10^9	1.6×10^7
2	4.1×10^7	4.1×10^7

される。実際に、3 文節の回文候補の網羅的生成において、このような初期状態が生成された数は、文頭固定法では 69 個、折返し固定法では 13,456 個であった。このことが、深さ $n = 1$ の状態数の差の拡大を招いている。

3 文節集合のスリム化

効率化の一環として、文節集合 D をスリム化した。

3.1 文節集合の問題点

2.3 節の実験で使用した文節集合 D は、347 万の文節から構成されている。この文節集合 D は、次のような方法で作成した。

1. JUMAN5.2 の内容語辞書 (ContentW.dic) の各項目を、JUMAN の活用規則に基づき活用展開する。
2. 活用展開した内容語に、接続可能な機能語 (助動詞、格助詞、終助詞など) を付与する。

このような手順で機械的に作成した文節集合 D には、次のような文節が含まれている。

a. 「雨 (う)」や「委 (い)」などの語構成要素を含む文節。これは、D の生成のベースとして採用した JUMAN5.2 の辞書に、このような形態素が含まれていることによる。

b. 「愛育されろ」や「愛顧したがる」といった普段使われない文節。これは、機能語の付与において、文法的に許容されるものをすべて付与したことによる。

これらの文節は、文節単体で通意条件を満たさない。つまり、これらの文節を含んだ回文候補は、通意条件を満たさない。そのため、このような文節は、あらかじめ、文節集合 D から削除しておくことが望ましい。

3.2 文節集合のスリム化手法

まず、前者のタイプの文節を D から削除するために、JUMAN の辞書から語構成要素を削除した。具体的には、JUMAN の辞書に含まれるひらがな表記が 2 文字以下の形態素 (2096 項目) をすべて人手で調べ、語構成要素であると判断したもの (286 項目) を削除した。短い語構成要素を優先的に調べたのは、短い文節の方が回文候補に出現する頻度が高く、出力に与える影響が大きいためである。

次に、後者のタイプの文節を D から削除するために、D の要素のうち、コーパス中に出現しないものを削除した。コーパスには、1991 年から 2005 年までの毎日新聞と、現代日本語書き言葉均衡コーパス 2009 年度モニター公開版を用いた。JUMAN と KNP を用いてコーパスを文節に切り分け、文節集合 D_c を作成した。その後、D_c 中に含まれない D の要素を D から削除した。ここで、同一文節の判定には、「表記と読みがともに一致する」という条件を採用した。

回文では、「夜起きぬタヌキおるよ」のように終助詞を比較的多用する。これに対し、今回使用した書き言葉コーパスには、終助詞を使用した文が少ない。そのため、D_c との照合は終助詞を付与する前の D に対して行い、出現しない文節を削除した後に、終助詞を付与した。

これら 2 つのスリム化手法を行うことで、D の項目数は 347 万から 90 万に減少した。スリム化後の D を D' と表記する。

3.3 評価実験

文節集合 D のスリム化が適切に行われたのかを評価するための実験を行った。D と D' それぞれを用いて、3 文節の回文候補を文頭固定法で網羅的に生成し、その生成速度と、回文候補生成数を比較した。

D' の作成過程において、回文生成に必要な文節も削除されてしまう可能性が考えられる。そこで、D を用いて生成した回文候補に含まれていた 127 文の回文が、D' を用いて生成した回文候補に含まれるか調べた。この調査に使用した回文は、Web 検索によるフィルタリングをかけた回文候補群から人手で発見した回文である [2]。それらの一部を表 3 に示す。

実験結果を表 4 に示す。D' を用いた場合、D を用いた場合と比較して、生成数と生成時間は、おおよそ 1/3 となった。その一方で、D を用いた場合に生成された 127 個の回文のうちの 114 個 (約 90%) の回文を、D' を用いた場合にも生成できた。この結果から有効に D をスリム化することに成功したと言える。

D' を用いた回文候補の網羅的生成では、1 つのシード文節から 3 文節回文候補を文頭固定法で網羅的に生

表 3: 実験に用いた回文例

しきたり 懲りた 棋士
サンダルが 上がる 段差
コンパクトに 溶く パン粉
話題が いちいち 意外だわ
悔しくて 串 焼く
格安な 茄子 焼くか
累進に 誤認 強いる

表 4: 実験結果

	D を使用	D' を使用	D'/D
生成時間	42 分	15 分	35.7%
回文候補数	4093 万	1549 万	37.8%
回文含有数	127	114	89.8%

成した場合、最大でも 2.3 秒で生成可能で、リアルタイムで生成が可能な速度となったといえる。

3.4 検討

スリム化後の文節集合 D' を用いて生成できなかった回文 13 文を表 5 に示す。これらの回文には、D' に存在しない文節が含まれる。調査の結果、それらの文節は、すべてコーパスから作成した文節集合 D_c に含まれていないことが判明した。

これらの文節が D_c に含まれない原因は、以下の 4 つのタイプに分類できる。

1. コーパス中にそもそも出現しない。

2. コーパス中では、異なる表記が用いられている。
例えば、コーパス中では、「鰐」ではなく、「ワニ」が用いられている。

3. コーパス中に存在するが、JUMAN+KNP は正しい読みが付与できなかった。
例えば、「夜」は、「よ」という読みが付与されたため、D_c に含まれない。

4. コーパス中に存在するが、JUMAN+KNP は異なる文節区切りを採用した。
例えば、「関係無く」は、「関係」と「無く」の 2 文節に分割されたため、D_c に含まれない。

タイプ 2、3 の文節に関しては、ある程度対策が可能ではないかと考えられる。例えば、タイプ 2 の対策として、D と D_c のマッチングの際に、JUMAN の辞書に記載されている異表記も確認する、タイプ 3 の対策として、JUMAN の結果を複数取得するなどが考えられる。

表 5: D' に含まれない文節をもつ回文一覧

漢字仮名交じり表記	D' にない文節	タイプ
澄ました 神が 味方します	味方します	1
潰瘍も もう 良いか	潰瘍も	1
鶏 と 小鳥と 鰐	鰐	2
村 数々 絡む	数々	2
以下の 一意の 解	一意の	2
気怠く ついつい 作るだけ	気怠く	2
稲作 なかなか 無くさない	無くさない	2
相当 悠々と 嘘	悠々と	2
夜 家 居るよ	夜	3
夜 オーナー 居るよ	夜、居るよ	3
密談 組んだ 罪	罪	3
虫の 大会 楽しむ	虫の	3
関係無く 無い 喧嘩	関係無く	4

4 おわりに

今回我々は、文頭結合法という新たな回文候補自動生成手法の導入、文節集合 D のスリム化を行うことで、回文候補生成の効率化を実現した。

文頭結合法の導入により、3 文節の回文候補の網羅的生成に要した時間がおよそ 1/30 となった。さらに、D のスリム化により、再現率を約 90%に保ったまま、3 文節の回文候補の生成数をおよそ 1/3 に削減することに成功した。回文候補出力数の減少により、回文候補の網羅的生成に要した時間もおよそ 1/3 となった。

最終的に、3 文節の回文候補の網羅的生成に要した時間が、昨年提案した手法 [2] と比べおよそ 1/90 となった。それにより、1 つのシード文節から 3 文節の回文候補を生成した場合、もっとも時間がかかるシード文節でも 2.3 秒となり、リアルタイムで生成が可能となった。

謝辞 本研究では、CD-毎日新聞データ集 (1991 版-2005 版)、現代日本語書き言葉均衡コーパス 2009 年度モニター版を使用した。

参考文献

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

[2] 鈴木啓輔, 佐藤理史: 文節結合による回文の自動生成, 2010 年度人工知能学会全国大会論文集 (第 24 回), 3D4-3(2010).