

モーラ数と音節数を考慮した自然文から歌詞への変換

山本 貴史[‡] 松原 正樹[‡] 斎藤 博昭[‡]

[‡]慶應義塾大学 理工学部 情報工学科

[‡]慶應義塾大学大学院 理工学研究科

E-mail:yamatoka@nak.ics.keio.ac.jp

1 はじめに

近年、相手に自分の思いなどを伝える手段として、手紙の代わりに E メール、ブログなどがあげられるが、これらは自然文で表現されており、インパクトに欠け、思いが伝わりにくい。例えば、主張したいことを伝え、相手に強い印象を与えるものとしては、五音や七音の言葉を用いたキャッチフレーズや標語などがあるが、これらにリズムを与えることで、フレーズが耳に残りやすくなる。リズムのついたフレーズは歌として多くの人に親しまれている。以上のことから分かるように、歌には相手に自分の思いを伝える力があると言える。しかし、自然文をそのまま歌詞としたのでは、音楽にはなりにくい。なぜならば、音楽には人間が音楽とを感じるための構造があり、歌詞にもまた構造があるためである。ゆえに音楽的知識の無い人間にとって構造を考慮しつつ歌詞を書いていくことは、言葉の音数と伝えたい思いの双方を同時に考慮しなければならないため、容易ではない。

そこで本稿では、伝えたい思いが書かれた自然文を、モーラ数と音節数を考慮することで、五音や七音のフレーズで形成される構造へと変換する手法を提案する。

2 関連研究および前提知識

コンピュータを用いた歌詞の作成に関する研究には、自動的に歌詞を生成する自動作詞と、人間の作詞を支援する作詞支援がある。自動作詞は、マルコフモデルを用いて歌詞コーパスから単語の前後関係を学習し、歌詞を生成していく方法が主流であり、作詞支援は、ユーザの求める音数の単語や類語を提案する方法が主流である。また、楽曲作成方法には、先に作詞を行い、詞を基に作曲をする「詞先」と、先に作曲を行い、曲を基に作詞をする「曲先」がある。

吉川らは曲のメロディーに合わせて日本語の歌詞生成を半自動的に行う曲先作詞システム LYRICA[1]を開発している。LYRICA は作詞支援機能も備えている。また、Feisas らの自動作詞システム迷句りり[2]は、マルコフ連鎖を利用して、ユーザの入力した音節数の言葉で歌詞を作成する自動作詞システムで

ある。ユーザの求める音節数の言葉やイメージする言葉を提示する作詞支援システムとしては、inzeiO[3]、ミュージックビルダー [4] がある。秋月らは歌詞の言語学的な研究を行っている [5][6] が、歌詞を作るためには、音楽的な要素も考慮しなければならない。

本稿では、自然文を入力とし、モーラ数と音節数による形態素の音数の変化および歌詞では省略されることのある表現の追加と削除により、五音や七音のフレーズで形成される構造へと変換することを目的とする。音節とは母音を中心とする音のまとまりの単位であり、モーラとは音節を部分的に分解した単位である [7]。本稿では、実際に音符一つに割り当てる文字を 1 として数えた数を形態素の音数と定義する。形態素の最大音数はモーラ数であり、最小音数は音節数となる。

五音や七音は日本語の伝統的韻律であり、俳句に用いられているだけでなく、演歌や歌謡曲においてもそのほとんどの歌詞が五音や七音のフレーズを多く含んでいる。RWC 音楽データベース [8] の歌謡曲 10 曲の 692 フレーズ中、五音や七音のフレーズの割合は 51%にも及ぶ。音楽と 4 という数字は密接に関係しており、基本的にフレーズは 4 拍 4 小節 4 回の中に作られる。さらに、フレーズにおいて同じパターンを続けて、一つだけ違うパターンを持つ音楽構造は、音楽において音楽らしいと感じることのできる構造である。例えば、図 1 のように 4 拍目に別のパターンを持つことで、五音や七音となり、実際にも五音と七音を使ったフレーズの方が音楽的な構造を持ちやすいと言える。この構造のことを本稿では 3-1 構造と呼ぶ。3-1 構造は音数だけでなく、句や文レベルでも成り立つことを補足しておく。



図 1: 五音と七音における 3-1 構造の例

3 提案手法

システムを構築するにあたっての手法を提案し、提案システムの内部について述べる。

本システムは、前処理部、構造解析部、音数決定部、構造決定部からなる。それぞれについて順に述べる。

3.1 前処理部

入力された自然文を，“，”，“。”，“、”，“。”，“，”，“。”で区切る。これは CaboCha¹ を用いて構文解析する際に、句読点で区切られない場合があるためである。入力が図 2 のようであった場合、出力は図 3 のように二つの文に分けられる。

今回はダメだったけど、次までには頑張ってあそこにとどり着きたい。

図 2: 前処理部の入力例

1. 今回はダメだったけど
2. 次までには頑張ってあそこにとどり着きたい

図 3: 前処理部の出力例

3.2 構造解析部

前処理部の結果を CaboCha を用いて構文解析を行い、句、文節、形態素のまとまり、そして読みを得る。ここで句とは、直後の文節に係る文節をひとまとまりにしたものとする。以下に例を示す。

次までには頑張ってあそこにとどり着きたい

図 4: 入力の例

入力が図 4 のようであった時、構文解析結果は図 5 のようになる。

図 5 において、一行一行が一つの文節である。また、この例において、「あそこに」が直後の「たどり着きたい」に係っているためにひとつの句となり、句は図 6 の 3 つとなる。

¹<http://chasen.org/~taku/software/cabocha/>

次までには-----D
頑張って---D
あそこに-D
たどり着きたい

図 5: 構文解析の例

1. 次までには
2. 頑張って
3. あそこにとどり着きたい

図 6: 句の例

3.3 音数決定部

音数決定部では句ごとに、以下に示すフレーズ決定アルゴリズムと音数調整法を用い、フレーズごとの音数を決定する。

3.3.1 フレーズ決定アルゴリズム

五音や七音のフレーズを作成するフレーズ決定アルゴリズムを以下のように定義する。

- (1) 句に含まれる文節 W_0, \dots, W_E とそれぞれのモーラ数 M_0, \dots, M_E を入力とする。 $N \leftarrow 0$ とする。リストを空にする。
- (2) リストに W_N を加える。
- (3) リスト内の文節群のモーラ数の和 S を計算する。
- (4) $N = E$ の場合、リスト内の文節群を音数 S のフレーズとする。ステップ (11) へ飛ぶ。
- (5) $S \geq 8$ の場合、音数調整により音数を 7 にできるならば、リスト内の文節群を音数 7 のフレーズとし、リストを空にして、ステップ (10) へ飛ぶ。音数を 7 にできないならば、リスト内の最初に加えられた文節 W_i がリストから外され、 W_i は、音数 M_i のフレーズとなり、ステップ (3) に戻る。
- (6) $S = 7$ の場合、リスト内の文節群を音数 S のフレーズとし、リストを空にしてステップ (10) へ飛ぶ。

- (7) $S = 5$ の場合, $S + M_{N+1} = 7$ であれば, ステップ (10) へ飛ぶ. そうでなければ, リスト内の文節群を音数 5 のフレーズとし, リストを空にしてステップ (10) へ飛ぶ.
- (8) $S \leq 6$ の場合, $S + M_{N+1} = 5, 7$ であれば, ステップ (10) へ飛ぶ.
- (9) リスト内の文節群が音数調整により音数を 5 か 7 にできるならば, ひとつのフレーズとし, リストを空にしてステップ (10) へ飛ぶ.
- (10) $N \leftarrow N + 1$ とする. ステップ (2) に戻る.
- (11) 音数の決まったフレーズ群が得られる.

3.3.2 音数の調整方法

音数の調整方法について述べる. いずれのルールも 33 曲の歌詞の分析に基づく経験的なものである.

- 助詞の省略と追加

格助詞 “が”, “を”, “に”, “で”, 係助詞 “は”, “も” は省略可能であるとする (図 7).

変換前: 今回はダメだったけど
変換後: 今回ダメだったけど

図 7: 助詞省略の例

間投助詞 “よ”, “ね”, “さ” は追加可能である. 今回は動詞の後には “よ” を, 名詞, 終助詞の後以外には “ね” または “さ” を追加するようにする (図 8).

変換前: 今回はダメだったけど
変換後: 今回はダメだったけどね

図 8: 間投助詞追加の例

- い抜き言葉による “い” の省略と追加 (図 9)

い有り: 走っている
い抜き: 走ってる

図 9: い抜き言葉の変換例

- 接続詞 “けれど” の変換

接続詞 “けれど” は, “けど” や “けれども” としても意味は変わらない. 逆に “けど”, “けれども” もそれぞれ変換が可能である (図 10).

今回はダメだったけど
今回はダメだったけれど
今回はダメだったけれども

図 10: 接続詞 “けれど” の変換例

- 音節数を考慮し, 言葉の音数を減らす

3.4 構造決定部

同一文内であれば句を並び替えても文の意味は伝わるということを仮定した上で, 句の並べ替えを行い, 可能な限りの並べ替え候補を作る. すべての候補に対し評価を行い, 最も高い評価値の候補を出力する. 評価は表 1 の項目を考慮して行われる.

表 1: 評価項目

| |
|-----------------------|
| 音数レベルでの 3-1 構造数 |
| 句レベルでの 3-1 の構造数 |
| 文レベルでの 3-1 の構造数 |
| 句レベルで同じ音数構造があった数 |
| 文レベルで同じ音数構造があった数 |
| 文末に 5 音 7 音以外の音数があった数 |
| 五七調と七五調の数の差 |

4 実験結果

本章では, 提案手法の有用性を評価するための歌詞への構造変換実験について報告する.

4.1 実験条件

五行程度の文を入力する. 数字は CaboCha では読みが出てこないで, 漢数字での入力とする.

夜中の三時に目が覚めた。
携帯にもメールが来ないけれど、
私は朝までずっと待っている。
結局は睡眠不足。
あなたに逢うと文句が言えない。
まるで飛べない鳥だね。

図 11: 入力する自然文

4.2 実験結果

図 11 を入力とした場合の変換結果は図 12, 図 13 のようになる。最も高い評価値の構造が複数あったために、複数の結果が出力されている。また、図中の括弧内の数字はフレーズの音数である。

夜中の三時 (7) 目が覚めた (5)
携帯に (5) メール来ないけど (7)
朝までさ (5) ずっと待ってる (7)
結局は (5) 睡眠不足 (7)
あなたに逢うと (7) 文句言えない (7)
まるで飛べない (7) 鳥だね (4)

図 12: 歌詞への変換結果 (a)

夜中の三時 (7) 目が覚めた (5)
メール来ないけど (7) 携帯に (5)
ずっと待ってる (7) 朝までさ (5)
結局は (5) 睡眠不足 (7)
文句言えない (7) あなたに逢うと (7)
まるで飛べない (7) 鳥だね (4)

図 13: 歌詞への変換結果 (b)

以上の実験結果を見ると、提案手法は五音や七音のフレーズを作成することができ、さらに文レベルで 3-1 構造も作られており、日本語の韻律と音楽構造に適した歌詞へと変換できる能力を持っていると言える。

5 考察

実験結果より、提案手法によって自然文から自然な区切りで、五音や七音のフレーズへと変換できることが示された。また、入力した自然文と比較してみると、文全体の意味が変わっていないことも分かる。しかし、助詞の追加と削除により、微妙なニュアンスが変わってしまうことも考えられ、助詞の追加と省略条件についても考慮したいと考えている。

また、語句の言い換えを用いて、より多様な表現を提案することは、ユーザの伝えたい思いが変わってしまうことがあるかもしれないが、これにより言い換えを用いない場合よりも、より音楽的な構造を生成することができれば、より良い歌詞を提示できると考えられる。

6 おわりに

本稿では自分の想いを伝えられる歌詞を生成できる、自然文からの歌詞への構造変換手法について述べた。モーラ数と音節数による音数の変化と、歌詞では省略されることのある表現の追加と削除を行うことで、元の自然文の意味を変えずに歌詞構造への変換を実現した。また実験結果から提案手法の有用性を確認することができた。これにより、歌詞からの自動作曲システム Orpheus[9] と組み合わせることで、自然文からの自動作曲などの応用が期待できる。

参考文献

- [1] 吉川 美奈子, 原 陽一: 自動作詞システムの開発「リリカ LYRICA」, <http://www.ipa.go.jp/about/jigyoseika/04fy-pro/mito/2004-324d.pdf>, 2004.
- [2] Feisas: LYRICALOID 迷句リリ, <http://lyricaloid.seesaa.net/>, 2008.
- [3] SIMPSON Inc. inzeiO, <http://www.simpson.co.jp/inz/inzeio.html>.
- [4] ミュージカル・プラン・ミュージックビルダー, http://www.musicalplan.com/pro_musicb.html, 2001.
- [5] 歌詞の言語学, 尚絅学院大学女子短期大学部英文科, 秋月高太郎, 2006.
- [6] 歌詞の言語学 2, 尚絅学院大学女子短期大学部英文科, 秋月高太郎, 2007.
- [7] 窪園 晴夫, 本間 猛: 音節とモーラ, 研究社, 2002.
- [8] 後藤 真孝, 橋口 博樹, 西村 拓一, 岡 隆一: "RWC 研究用音楽データベース: 研究目的で利用可能な著作権処理済み楽曲・楽器音データベース", 情報処理学会論文誌, Vol.45, No.3, pp.728-738, March 2004.
- [9] 米林 裕一郎, 中妻 啓, 西本 卓也, 嵯峨山 茂樹, "Orpheus: 歌詞の韻律を利用した Web ベース自動作曲システム," 情報処理学会インタラクショナル 2008 論文集, IPSJ Symposium Series Vol.2008, No.4, pp.27-28, Mar., 2008.