

外国語の歌曲を訳詞するためのインターフェースの検討

西村 綾乃* 伊藤 貴之*

概要. 音楽は世界中で作曲されており、インターネットが普及した現代では、外国語の歌曲を気軽に聞くことができるようになった。しかしながら、母国語ではない言語の歌詞を完全に理解することは難しい。そこで、外国語歌詞の日本語訳を出力するインタラクティブな訳詞システムを提案する。本稿では、訳詞の作業効率を上げるために適したインターフェースについて調査するため、各機能の実装の切り替えが可能な Web アプリケーションの実装を行い、19 人を対象に実験を行った。

1 はじめに

インターネットの普及により、現在では世界中の音楽をいつでも気軽に聞くことが可能になった。しかしながら、外国語の歌曲は音楽そのものを楽しむことは出来ても、馴染みのない言語による歌詞の内容を完全に理解することは難しい。歌詞の意訳や直訳が掲載されていることもあるが、単なる翻訳によって生成された文章はメロディの音数や流れと合致しないため、音楽的な要素が無視されている。

一方で、海外の歌曲の多くが日本語に訳詞されている。しかし、訳詞を行うには歌曲に対して、原語の意味、音楽的なフレーズやリズム構成、歌唱における日本語の発音への理解などが必要であり、非熟達者には難しい作業である。多数の歌曲で 1 つの作品となるミュージカルやオペラでは、全編を通した話の流れや、メロディへの理解が必要となり、アマチュアにとって訳詞の労力は多大なものとなる。

そこで著者らは、外国語歌詞を日本語に訳詞するためのインタラクティブなシステムの開発に取り組んでいる。本システムは、ユーザが入力した譜面情報から仮の訳詞を自動生成し、好みに合わせて編集するためのユーザインタフェース上に仮訳詞付きの譜面を表示する。この開発の一環として著者らは、ユーザの負担が少なく、訳詞の作業効率の向上に適したインターフェースについて検証するため、歌詞とメロディの編集に関する各機能の実装の切り替えが可能な Web アプリケーションの実装を行った。本稿ではシステムの構想と Web アプリケーションの実装を説明し、さらにこれを用いて 19 人を対象に実験を行った結果を報告する。

2 訳詞の課題

日本語への訳詞の歴史は近藤朔風らが切り拓いた [1]。近藤は原言語の意味と韻律を可能な限り日本語に再現しようとする訳詞を重視した。本提案におい

ても、原言語の意味をなるべく日本語に再現するための自動訳詞を目指している。

日本語への訳詞システムの開発に着手するにあたり、まず訳詞特有の課題を列挙する。

原言語の歌詞を日本語に訳すと多くの場合、音数 (モーラ数) が増えるという問題点にぶつかる。メロディに使われている音符の数は決まっているため、訳詞においてはモーラ数を極端に増やすことは避けなければならない。結果として日本語への訳詞において、原言語の意味を省略する必要がある [2]。

一方で、歌唱に適した日本語の発音についても課題が指摘されている [3]。音節の観点から、撥音、促音、長音、二重母音と呼ばれる発音を、モーラ数を 1 とする場合と、直前の発音と合わせて歌うことで、モーラ数に含めない場合がある。これは、歌唱におけるリズム感に影響を与える。

さらに、「あ」「お」は歌唱において響きやすい母音であるが、「え」「い」「う」は響きにくい母音であり、訳詞を行う際に、選択した単語の母音に対して、言葉の意味を重視するか、歌唱的な面を重視するかといった問題が生じることがある。

ここで、訳詞をする際に課題となる点をまとめる。

課題 1 日本語への翻訳によってモーラ数が増える

課題 2 原言語の意味を取捨選択する必要がある

課題 3 撥音などの発音を 1 音とするかどうか、リズムと音節を考慮して判断する必要がある

課題 4 音楽的な理由から、特定の音に対する母音を限定する必要がある

課題 1 については、これまでの実験より、解決には及ばずとも童謡等であればモーラ数を抑えて譜面を生成できることが分かっている [4]。

これらの課題について、人手で訳詞を行った場合でも、結果にばらつきが出ることは自明であり、システムにおいても訳詞の解を 1 つに絞るのは難しい。そのため、ユーザが好みに合わせてパラメータの設定や、譜面の編集ができるシステムであるべきだと考えた。しかしながら、システムの制約が少なく自

由度が高い場合は、ユーザは手作業で訳詞を行うのと変わらない状況になってしまう。そこで、今回の実験では、どこまで機械で生成し、どこまでユーザが操作するべきかも調査対象に含める。

初期段階の実装において、ユーザが編集する項目として、メロディと歌詞が挙げられる。メロディの編集について、音楽的知識がないユーザにとって自由編集は困難であり、システム側で候補を提示または自動で編集するべきである。一方、歌詞の編集については、ユーザが訳詞の対象言語を日常的に使っている場合、システムで提示した候補のみからの選択では制約が厳しすぎる。そのため、歌詞の編集については自由入力を許容すべきと考えた。

3 関連研究

歌詞に関連する研究は数多く存在する。阿部らによる「pâtissier」[5]は、モーラ数と母音を入力することで歌詞候補文を生成する。伊藤らによる替え歌自動生成システム[6]は、入力文のモーラ数を算出した結果から、入力文を歌詞として割当可能な楽曲リストを作成し、リストから選択された曲に対して入力文を歌詞とした歌曲を生成する。

作曲支援に関する研究の例として、哇原らのシステム[7]は、音楽理論を対話型遺伝的アルゴリズムに適用し、視覚的に楽曲を把握できるインターフェースを構築することで、ユーザの負担を軽減している。

一方で、翻訳した文章のモーラ数を減らす課題は、自然言語処理における文圧縮の問題と関連がある。井手上らの部分的機械翻訳[8]は、原言語文と原言語文の一部の表現(部分的表現)を入力とし、部分的表現を中心とした簡潔な翻訳を出力する。また山本らの文末整形手法[9]では、新幹線車内の電光掲示板で流れるような簡潔な表現のために、体言止めや助詞止めを適用している。

これ以外の関連研究もその大半は「歌詞」か「メロディ」のどちらか一方を新しく作り出すものである。具体的には、歌詞の自動生成、あるいは入力歌詞に対応するメロディの自動生成のいずれかの手法が多数提案されている。それに対して本研究では、「歌詞」と「メロディ」がすでに両方存在することを前提として、形式や内容の点で整合性を保ったまま歌詞を翻訳することを目的としている。

4 自動訳詞システムの構想

図1は著者らが開発中の自動訳詞システムのシーケンス図である。図中の番号はそれぞれ(1)訳詞処理と譜面の生成、(2)生成された譜面の編集、(3)編集した譜面の保存、の流れを示している。なお現段階では、対象となる原語は英語に限定している。

(1)では訳詞のパラメータ(モーラ数の制限、母音の制限など)をユーザが設定し、英語歌詞付きの

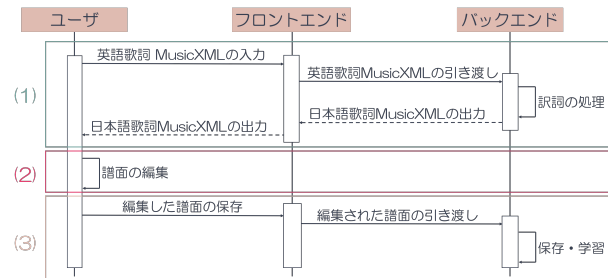


図 1. シーケンス図

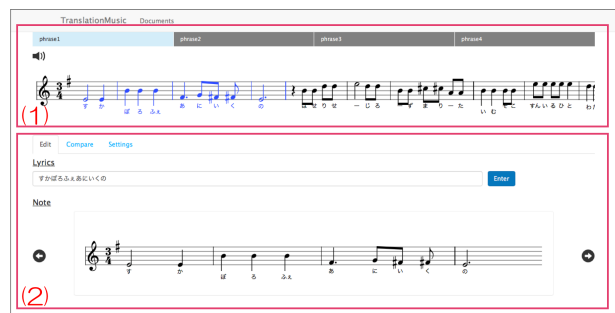


図 2. Web アプリケーション画面

MusicXMLを入力することで、訳詞の付随した譜面を生成する。(2)では、生成された譜面中のフレーズに対する部分修正のためのインターフェースを提供する。ここでは訳詞候補やメロディの譜割り候補をユーザに提示して選択させる。また(3)では、ユーザが何度も繰り返しシステムを利用し、多数の楽曲の編集を繰り返すことで、パラメータの調整や単語の選び方の好み、メロディの修正方法を学習する。

今回は(2)について、作業効率を向上するために、どのようなインターフェースが適切かを見極めるため、また自動処理すべき工程と人間が編集すべき工程を特定するために実験を行った。この実験のために、メロディと歌詞の編集機能に関する各モジュールの実装を切り替え可能なアプリケーションを実装した。

5 インターフェースの提案

実装したアプリケーションの画面の機能は(1)譜面表示部分、(2)タブ機能部分に大別される(図2)。

譜面表示部分では、譜面を左右にスクロールできる。上部には楽曲中のどのフレーズを編集しているかを示すバーがあり、対象フレーズを青色で表示する。譜面左上のスピーカーアイコンをクリックすると、編集中の曲を試聴できる。本実装ではSinsy[10]を適用して、クリック時に動的に音源を生成する。

タブ機能部分では、Edit, Compare, Settingsの3種類のタブが用意してある。Editタブでは歌詞および音符の編集ができる。Compareタブでは英語

表 1. 各機能のモジュール内容

	訳詞候補 (A)	歌詞編集 (B)	音符編集 (C)
0	複数の訳詞候補から選択	初めに原語歌詞のみを見て複数の訳詞候補から選択	自動で編集
1	訳詞候補は1つのみ表示	訳詞が出力された譜面から任意で編集	候補から選択

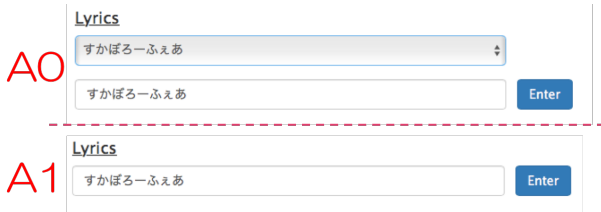


図 3. 訳詞候補 A0, A1 の表示形式

歌詞と日本語歌詞を比較できる。Settings タブではアプリケーションのヘルプ画面表示や、譜面の編集を終了するためのボタンが設置してある。

本実装では各機能をモジュール化し、2 種類の実装を任意に切り替えられる設計にした。モジュールの内容を表 1 に示す。以下、訳詞候補を A、歌詞編集を B、音符編集を C と表現し、各モジュールに用いる実装の組み合わせを (A B C) の順に (0 0 0) のように表記する。ここで 0, 1 は各モジュールの 2 種類の実装を示す。またモジュール A の 2 種類の実装をそれぞれ A0, A1 と表記する。8 通りの実装の組み合わせのうち (1 0 0) と (1 0 1) は、訳詞候補が 1 つしかない場合 (A1)、フレーズごとの候補は一意になるため、複数の訳詞候補から選択する (B0) という操作に矛盾する。そこでこの 2 つの組み合わせを除外し、他 6 通りのいずれかがランダムに適用される仕様とした。

続いて各モジュールの機能の詳細を述べる。

5.1 訳詞候補

訳詞候補は用意された訳詞のうち 1 個をユーザに選択させる。用意された候補はタブ機能部分の Edit 上部に表示される (図 3)。

A0 では複数 (現在の実装では 4 個) の訳詞候補を提示し、プルダウン形式の表示によって選択させる。実験では現時点での自動訳詞生成の実装から出力した候補を第 1 候補とし、第 2 候補から第 4 候補には人手で作成した訳詞を用いている。人手の訳詞では、英語歌詞の機械翻訳文中で最も tf-idf 値が高かった単語を訳詞の先頭に使用している。tf-idf 値を算出するための文書として Project Gutenberg や青空文庫などから作成されたコーパス [11] を利用した。テキストフォームにはプルダウンから選択した訳詞が表示され、ユーザが自由に編集できる。

A1 ではテキストフォームのみが表示される。このテキストフォームには、いかなる場合も初期値として訳詞候補群のいずれかの訳詞が表示されており、

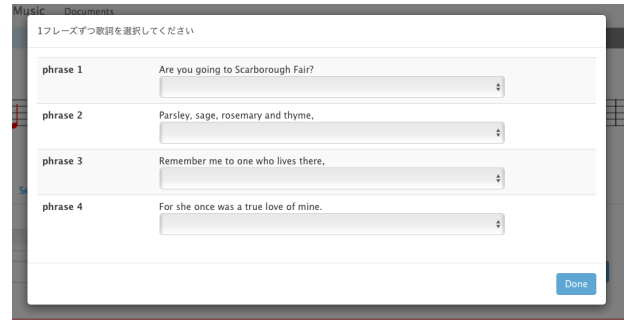


図 4. 歌詞編集 B0 の表示形式



表 2. 被験者の分類内容と被験者数

	A0	A1	B0	B1	C0	C1	全体
被験者の分類	(0 1 0) (0 1 1)	(1 1 0) (1 1 1)	(0 0 0) (0 0 1)	(0 1 0) (0 1 1)	(0 0 0) (0 1 0) (1 1 0)	(0 0 1) (0 1 1) (1 1 1)	(0 0 0) (0 0 1) (0 1 0) (0 1 1) (1 1 0) (1 1 1)
被験者数	6	6	7	6	9	10	19

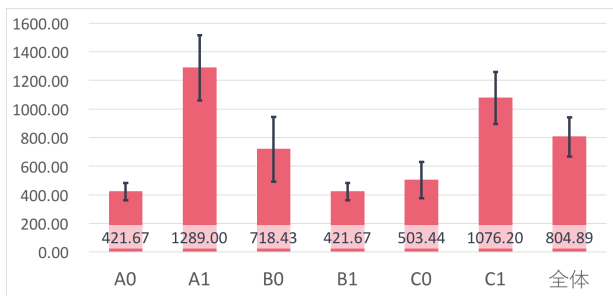


図 6. 各モジュールの平均作業時間と標準誤差

には歌詞とメロディの譜割り候補が表示される。「歌詞の音数 > メロディの音数」「歌詞の音数 < メロディの音数」のどちらの場合でも、最大で2種類の候補が表示される。ここで左右の矢印のボタンをクリックすることで、候補の表示が切り替わる。中央の譜面を押すと、その譜割りが選択され、画面上部の譜面表示部分に表示される。譜割りが未選択であれば譜面表示部分から歌詞が消え、フレーズの表示が赤色に切り替わる。

6 実験と結果

本システムをサーバ上で稼働し、限定公開することで実験環境を用意した。ユーザがシステムにアクセスすると、システムの説明が表示されて訳詞の作業に入り、作業を完了するとアンケート入力画面に切り替わる、という流れでデータを収集した。訳詞作業の工程において、システム上でユーザがクリックした場所と回数および作業時間、表示されたモジュールの組み合わせを記録した。実験終了後、モジュールの組み合わせごとに被験者を分類し、それを母集団としてモジュール別の記録値の平均値および標準誤差を算出した。

今回は、インターフェースの決定のために、主に定量的な評価を基に結果について考察を行った。実験では、訳詞を行う曲として、フレーズ数が4つである「Scarborough Fair」を選択した。被験者は男性9人、女性10人、合計19人、23歳~37歳の平均年齢は27歳であった。被験者の分類内容と被験者数を表2に示す。また、全ての数値を四捨五入により小数第2位まで示してある。

図6に各モジュールの平均作業時間と標準誤差を示す。A0とB1は421.67秒となり、他のモジュールより明らかに作業時間が短い。これは訳詞候補を

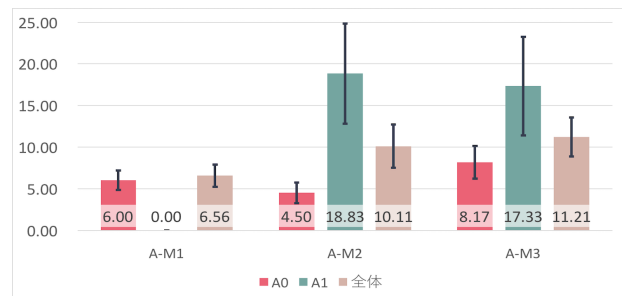


図 7. A0 と A1 の比較結果

提示することで作業時間を短縮できることを示唆している。続いてモジュールごとに結果を比較する。

6.1 訳詞候補の結果

A0とA1について、クリック平均回数および標準誤差の比較結果を図7に示す。グラフはそれぞれ以下の内容を表している。

A-M1 プルダウンから訳詞候補を選択

A-M2 テキストフォームで歌詞を編集し決定した

A-M3 Compare タブを押した

A-M1において、A1が0.00となっているのは、モジュールとしてプルダウンが表示されていなかったことを表している。A0において、プルダウンがクリックされた平均回数は6.00回だった。

A-M2では、A0のテキストフォームでの平均編集回数は4.50回となり、A1の18.83回と比較すると4倍以上の差が見られた。全体と比較しても、訳詞候補が選択できるA0は、ユーザがテキストフォームから歌詞を編集する回数は少なかった。

A-M3についても、A0では英語と日本語の歌詞を比較する回数が少ないことがわかった。

6.2 歌詞編集の結果

B0とB1について、クリック平均回数および標準誤差の比較を図8に示す。グラフはそれぞれ、以下の内容を表している。

B-M1 モーダル上のプルダウンから訳詞候補を選択

B-M2 プルダウンから訳詞候補を選択

B-M3 テキストフォームで歌詞を編集し決定した

B-M4 Compare タブを押した

B-M1において、B1が0.00となっているのは、システムへのアクセス初期画面でモーダルが表示さ

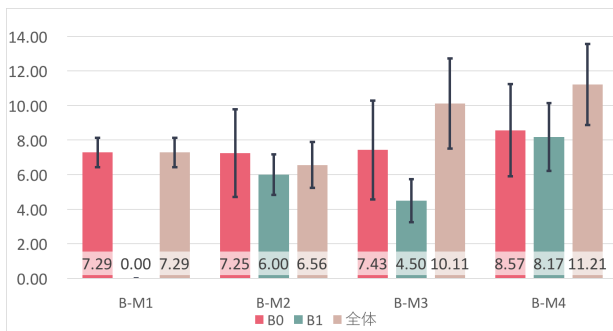


図 8. B0 と B1 の比較結果

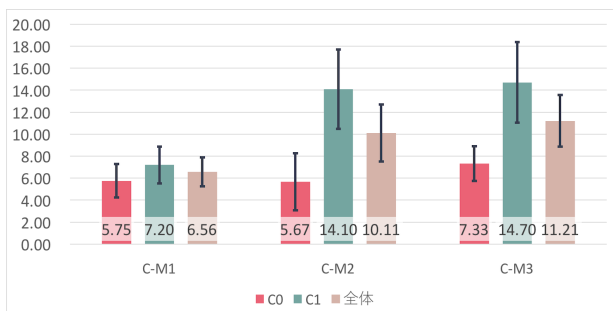


図 9. C0 と C1 の比較結果

れないことを示している。B0 でモデルを表示した際に、訳詞候補のプルダウンがクリックされた平均回数は 7.29 回だった。

B-M2 では、B0 について、最初にモデル上のプルダウンから選択したにも関わらず、その後の編集画面にてプルダウンをクリックした回数は 7.25 回となっており、B1 の 6.00 回を上回った。

B-M3 にて、テキストフォームから歌詞が編集された回数についても、B0 が 7.43 回、B1 が 4.50 回と B1 の方が作業回数は少なかった。A の比較結果も踏まえると、A0 と B1 は他のモジュールと比較して作業回数が少ない傾向にあることがわかった。

B-M4 に注目すると、B0 と B1 はどちらも訳詞候補が表示されるが、全体と比較しても訳詞候補がある場合は、Compare タブを押した回数について、クリックされる回数が低いことがわかった。

6.3 音符編集の結果

C0 と C1 について、クリック平均回数および標準誤差の比較を図 9 に示す。グラフはそれぞれ、以下の内容を表している。

C-M1 プルダウンから訳詞候補を選択

C-M2 テキストフォームで歌詞を編集し決定した

C-M3 Compare タブを押した

C-M1, C-M2, C-M3 のすべての項目において、C0 の方が作業回数が少ない結果となった。全体と比較しても、C1 は作業回数が増える傾向が伺えた。

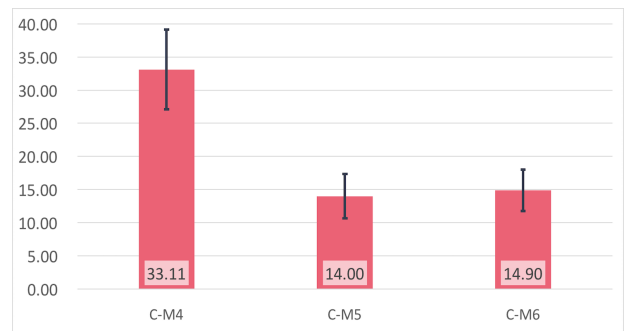


図 10. 音符編集部分のクリック平均回数

また、音符編集部分のクリック平均回数と誤差範囲について図 10 に示す。グラフはそれぞれ、以下の内容を表している。

C-M4 音符編集の右ボタンを押す

C-M5 音符編集の左ボタンを押す

C-M6 音符編集の決定を押す

グラフに示している通り、右矢印、左矢印がクリックされた回数に対して、音符が決定された回数が少なかった。これは、候補が最大 2 つまでしか表示されないにもかかわらず、インターフェースに候補の最大数を示すデザインを組み込まなかったため、ユーザが何種類の候補があるか把握できず何度も矢印ボタンを押してしまったためである。

6.4 アンケート結果

被験者から得た意見を以下に述べる。

良かった点

- 訳詞を候補から選択できる (3 人)
- 歌詞の編集結果を歌唱として試聴できる (3 人)
- 全体的に見た目がわかりやすく操作しやすい (2 人)
- 横スクロールでフレーズが切り替わる (3 人)
- 音符編集を決定すると譜面の色が変わるので、作業が完了したのがわかりやすい

悪かった点

- 譜面上部のフレーズを示すバーをクリックして移動したい (3 人)
- フレーズごとに試聴したい (3 人)
- 最初に何をすればいいかわからなかった (2 人)
- 編集する部分に英語を表示してほしい
- 音符編集の選択肢がいくつあるかわからない
- メロディに歌詞をいい感じに割り振る方法がわからなかった
- 各タスクを区切って示して欲しい

5 段階評価のアンケート結果を図 11 に示す。A0 と A1 を比較すると、A0 の方が操作回数は少なかったにもかかわらず、システムの使いやすさ、完成した曲に対する満足度は A1 が全て上回っていた。プル

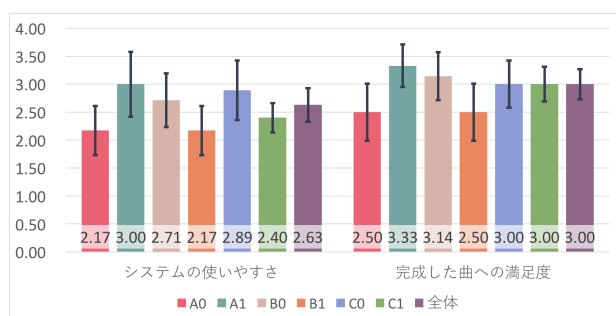


図 11. システムに関するアンケート結果

ダウンから歌詞を選択できる点が良かったという意見が多かったことから、これらの値が低い理由として、悪かった点にも列挙されているとおり、他のインターフェースの開発不十分な点が影響していると考えられる。

B0とB1では、作業回数の多かったB0が、どちらにおいてもB1を上回る結果となった。Aにおいても作業回数の多いA1の満足度が高かったことを考えると、大きな作業回数と作業時間を経ることが完成した曲への満足度に寄与している可能性がある。

C0とC1を比較すると、システムの使いやすさはC0がC1を上回ったが、完成した曲に対する満足度は変わらなかった。

7 考察

実験の結果から以下の点がわかった。

1. 訳詞候補があると作業回数が少なく、作業時間も短縮される
2. 訳詞候補があると英語と日本語の比較作業が減る
3. 最初にモデルを表示した方が使いやすいと感じるユーザが多い
4. 歌詞の編集での作業回数が多いほうが満足度が高くなる傾向がある
5. 音符編集は自動編集の方が作業回数と作業時間が少なく、システムも使いやすく、満足度も変わらない

3.について、説明の直後に編集画面を出した場合、最初に何をすればいいかわからなかったという意見があった。そのため、最初にモデルを表示し、強制的にプルダウンからユーザに訳詞を選択させることで、作業内容が明確化したため、システムの使いやすさにポイントが入ったと推測する。これは、ユーザのシステムに対する習熟度で評価が変わる可能性がある。そこで、クリック平均回数と作業時間を考慮すると、モデルは表示せずに譜面から気に入らない部分を編集してもらう方が作業効率が良い。

以上の考察結果から著者らは、訳詞システムの開発は以下のような方向性が妥当であると判断した。

- 訳詞はプルダウンによる複数候補を提示する

- 選択された訳詞が出力された譜面から、必要であればユーザが訳詞を編集する
- 音符は訳詞に合わせて自動編集する

本システムの開発において「訳詞候補生成」「音符編集」の自動化と満足度向上が著者らの急務である。

8 まとめと今後の課題

本稿では、訳詞の作業効率の向上に適したインターフェースについての調査のため、訳詞の各工程をモジュール化したWebアプリケーションの実装と、19人のユーザを対象とした実験結果を報告した。実験結果より、訳詞の作業効率を向上するためのインターフェースの構成を見極めるとともに、自動処理すべき工程とユーザが編集すべき工程について考察を深めることができた。

今後の課題として、訳詞候補の自動生成、音符の自動編集に関するアルゴリズムを改良したい。また、訳詞の課題4に挙げた母音の限定について、インターフェース上で候補を提示できるよう実装したい。

参考文献

- [1] 松田 直行. 訳詞の誕生: 近代日本音楽史における訳詞家近藤朔風の位相. 駒沢短大国文, 37:15-43, 2007.
- [2] 松田 直行. 「訳詞」とは何を訳すのか: 近藤朔風と森外によるオペラ『オルフェウス』訳詞の比較研究のための序章. 駒澤日本文化, (7):165-192, 2013.
- [3] 山本 金雄. 歌唱における日本語の発音に関する一考察. 千葉大学教育学部研究紀要. 第2部, 22:233-249, 1973.
- [4] 西村 綾乃, 伊藤 貴之. 外国語の歌曲の自動訳詞システムに関する初期実装. 研究報告音楽情報科学 (MUS), 2017(16):1-5, 2017.
- [5] 阿部 ちひろ, 伊藤 彰則他. patissier-アマチュア作詞家のための作詞補助システム. 研究報告音楽情報科学 (MUS), 2012(17):1-6, 2012.
- [6] 伊藤 悠真, 寺田 努, 塚本 昌彦他. Mnemonic DJ: 暗記学習のための替え歌自動生成システム. 情報処理学会論文誌, 56(11):2165-2176, 2015.
- [7] 畦原 宗之, 鬼沢 武久. インタラクティブ作曲支援システム-ユーザの負担の軽減. 人工知能学会全国大会論文集, 2003.
- [8] 山本 和英. 全文を翻訳しようとしないう機械翻訳: ワードグラフによる部分的機械翻訳の試み (機械翻訳技術の向上). *Japio year book*, pp. 276-279, 2013.
- [9] 山本 和英, 池田 諭史, 大橋 一輝. 「新幹線要約」のための文末の整形. 自然言語処理, 12(6):85-111, 2005.
- [10] K. Oura, A. Mase, T. Yamada, S. Muto, Y. Nankaku, and K. Tokuda. Recent development of the HMM-based singing voice synthesis system—Sinsy. In *Seventh ISCA Workshop on Speech Synthesis*, 2010.
- [11] 日英対訳文対応付けデータ. <http://www2.nict.go.jp/astrec-att/member/mutiyama/align/index.html>, 2017/10/16.