大規模言語モデルと自己修正に基づく歌唱可能な歌詞への phonemic translation

阪井 瞭介[1,a)]　深尾 貫太[1]　高道 慎之介[,2,b)]

概要：本研究では，原詞（元言語の歌詞）の意味の翻訳だけでなく発音の借用も行う歌詞自動翻訳手法を提案する．音楽の国際化が進む中で歌詞翻訳の需要は高まっているが，歌詞翻訳は一般的な文章翻訳とは異なり，訳詞をメロディに合わせた際に歌唱可能であることが求められる．また，歌詞翻訳技法の一つとして，原詞の発音を部分的に転写することで翻訳前後で感情的整合や没入感を維持する効果をもたらす phonemic translation が知られている．提案手法では，大規模言語モデルに基づいて訳詞候補を生成し，自動評価器により訳詞品質を定量化する処理を反復することで，原詞の発音を転写しつつ歌唱可能な訳詞を自動生成する．

1. はじめに

音楽は，社会的なつながりを促進する重要な共通言語としての役割を担っている [1]．近年では，音楽ストリーミングサービスの普及に伴い，アーティストが自らの楽曲の外国語版を制作する事例が増加している[*1]．このような事例では，原曲の歌詞（原詞）を外国語へ翻訳した歌詞（訳詞）を作成し，それを原曲のメロディに載せて歌唱する．歌詞翻訳は，訳詞としての自然さに加えて，訳詞をメロディに載せた際の歌唱可能性 [2] を考慮する必要がある点で，一般的な文章翻訳とは異なる専門的な領域とされる [3], [4]．このように歌唱可能性を満たした訳詞は，オペラやアニメソング（例：ディズニー作品），さらには童謡に至るまで，様々な音楽ジャンルにおいて作品の国際的な共感や魅力を高めるために広く必要とされている [5]．

歌詞翻訳の技法の一つとして，原詞の音声的特徴を優先的に模倣する phonemic translation がある [6][*2]．一般的な意味翻訳（translation）が原詞の意味内容を忠実に再現することを主目的とするのに対し，phonemic translation は，原詞の音声的特徴を訳詞において可能な限り再現することを目的とし，その過程において意味内容の一部を犠牲にすることも許容する点に特徴がある [6], [7]．ここで，原詞とその翻訳例を以下に示す．

- 原詞：しずむようにとけてゆくように [8]
- Translation（意味翻訳）：*As if sinking, as if melting away.*[*3]
- Phonemic translation：*Seize a move, you're on me, falling, and we were dissolving.* [9]
- Transliteration（翻字）：*shizumu you ni tokete yuku you ni*

意味翻訳は文法的に自然で意味内容も忠実であるのに対し，phonemic translation は，翻訳先言語話者でも理解可能な語を用いながら原詞と近い音節数や音声的響きを保とうとする点で異なる．また，

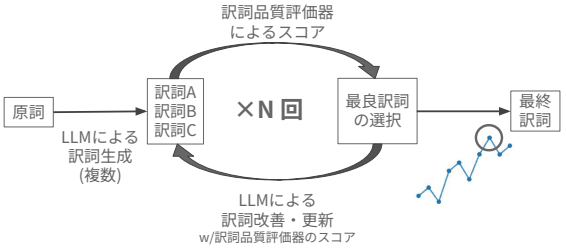


図 1　本研究の概要

[1] 慶應義塾大学
Keio University
[2] 東京大学
University of Tokyo
[a)] r73ryo.s@keio.jp
[b)] shinnosuke_takamichi@keio.jp

[*1] https://realsound.jp/2023/01/post-1224020.html
[*2] 我々の調べた限り，phonemic translation に対応する日本語専門用語は公知でないため，本論文では英語表記を用いる．
[*3] DeepL（https://www.deepl.com/ja/translator）にて翻訳した．

phonemic translation は，文字体系の変換を目的とする翻字とも本質的に異なり，意味をある程度保ったまま音声的に近い既存語を選び直す特徴がある．歌詞の phonemic translation は，メロディに訳詞を適合させ，原詞の音声的特徴を訳詞でも保持することで翻訳前後で感情的整合や没入感を維持する効果を持つと考えられる [2], [6].

次に，phonemic translation の自動化について考える．関連技術として，歌詞の意味自動翻訳に関する研究がある [4], [10], [11], [12], [13]．これらの研究では，原詞と訳詞からなる対訳データセットを用いて，機械学習モデルを教師あり学習する．一方で，著作権などの制約により，対訳データセットを大規模に構築することは難しい [4], [14]．特に，本研究が対象とする，原詞の音声的特徴を訳詞において再現した対訳データセットを体系的に構築することは，ほぼ不可能である．このため，既存の教師あり学習とは異なるアプローチが求められる．

そこで本研究では，大規模言語モデル（large language models；LLMs）と訳詞品質評価器に基づく，歌唱可能な訳詞への phonemic translation 手法を提案する（図 1 参照）．提案手法では，与えられたプロンプトおよび原詞に基づいて LLM が複数の訳詞候補を生成し，訳詞品質評価器がそれらの候補に対して phonemic translation としての訳詞品質を定量的に評価する．この生成と評価の過程を反復することで，原詞の発音を目標言語の音韻体系に基づいて転写しつつ，メロディに適合した歌唱可能な訳詞を自動生成する．訳詞品質評価器は，(1) 原詞と訳詞の**意味類似度**（*semantic similarity)*，(2) 文としての**訳詞自然度** *(sentence naturality)*，(3) 訳詞をメロディに載せた際の**歌唱可能度** *(singability)*，(4) 原詞から訳詞への**発音転写度** *(pronunciation transferability)* の 4 つの要素から構成され，各要素の重要度を用途に応じて個別に指定できる．本研究の主な貢献は，以下の 3 点である．

- 原詞の発音を転写しつつ歌唱可能な訳詞を自動生成する，新たな歌詞翻訳タスクを定義する．
- LLM に基づく訳詞生成と，意味類似度，訳詞自然度，歌唱可能度，発音転写度に基づく評価器を組み合わせた歌詞翻訳手法を提案する．
- 日英歌詞翻訳を対象とした実験的評価により，提案手法が発音や音節数といった音韻的制約をどの程度制御可能であるかを明らかにする．

2. 関連研究

2.1 歌詞翻訳と発音転写

歌詞翻訳．歌詞翻訳は翻訳家の音楽知識の欠如や歌詞翻訳の文化的背景の必要性などから翻訳の分野の中でも挑戦的なテーマとして位置付けられている [10]．歌詞翻訳において特に広く参照される原則に**ペンタスロン原則**（*pentathlon principle*）がある．この原則は，歌詞翻訳の際に音楽と歌詞間における歌唱可能性・韻・リズムと，翻訳品質における意味・自然さの 5 つの基準のバランスを取ることを目指す [2]．スコポス理論 [15] に基づき，訳詞の目的に応じて翻訳者がある基準を選択する一方で，どれか 1 つではなく，複数の基準をある程度満たすことも必要である．その他にも歌詞翻訳の方法にはさまざまな主張がある [16], [17] が，共通することとして歌詞翻訳において意味だけでなく歌唱性・リズム・韻律など非意味的側面への配慮が不可欠であることを強調している．

原詞から訳詞への発音転写．翻訳における発音転写の重要性は昔から議論されている [6], [7], [18], [19], [20]．詩翻訳では意味やスタイルの転写を優先する translation に対して，音声的特徴の模倣を優先する phonemic translation が定義されている [6]．つまり主目的は元言語における音声的響きを翻訳先言語で再現することであり，意味内容の一部を犠牲にすることも許容されるという考え方である．この考え方が歌詞翻訳にも表れており，例えばシューベルト作曲の歌曲において，歌詞の発音が作曲家の意図している音楽的効果と密接するとの主張がある [19]．そして，原詞と訳詞の音声的対応関係を IPA（international phonetic alphabet）に基づいてマッピングし，同一の音節数を保ちながら音素レベルで発音の類似性を評価している．さらに日本語のポップスに対しては言語学・音声学・認知科学の知見から歌の聞き手は歌詞の意味だけでなく，音声的質感も大切にしていることも主張されている [20].

2.2 歌詞の自動翻訳

歌詞自動翻訳． 2.1 節で言及した理論に基づき，近年では様々な歌詞自動翻訳手法が提案されている [4], [10], [11], [12], [13], [21]．対象言語は研究毎に異なるが，文章翻訳モデルを対訳歌詞データセットで追加学習したのち，歌唱可能性に関するプロンプトを翻訳時に追加する，あるいは歌唱可能性に関する評価器を用いる場合が多い [12]．このように多

くの自動歌詞翻訳では，意味的類似度と歌唱可能性を考慮するものが多い一方で，発音転写に着目した phonemic translation 的観点は考慮されていない．

発音転写を考慮する歌詞自動翻訳． 歌詞翻訳時に発音転写を考慮する研究が僅かながら存在する [13]．翻訳モデルの尤度に，原詞–訳詞候補間の調音パラメータ距離を加えることで，発音転写度合い基づいて訳詞を生成している．しかし，このようなリスコアリング手法は，歌詞を部分的に変更することしかできない．また，後述する反復的生成のメリットを享受できない．

2.3 LLM に基づく翻訳と反復

LLM に基づく特定ドメイン文章の翻訳． LLM の汎用さ [22], [23] から，LLM を文章翻訳モデルとして利用することも多い [14]．しかし LLM に基づく翻訳は，特定ドメインへの翻訳後文章に要請される制約（例えば語数）を，厳密に満たすことが難しい [24], [25]．そこで特定ドメインのデータセットで LLM を追加学習することが通例であり，歌詞翻訳においても同様の手続きが採られる [4], [12], [14]．十分な量のデータセットを用意できない場合は，コンテキストエンジニアリング [26] が用いられる．タスクや入出力例の提示をコンテキストとしてプロンプトに含めることで，汎用さを持って学習されたモデルで特殊なドメインタスクに対しても性能が向上する [26]．

生成と評価の反復． 生成と評価を反復することで, LLM の生成結果の質を向上できる [27], [28], [29], [30]．Self-refine [31] や Reflexion [28] は，LLM 自身が生成結果の欠点を言語的に記述し，その記述をコンテキストとして再度生成する自己修正フレームワークである．また OPRO（optimization by prompting）[29] や Tree-of-thoughts [30] は LLM に複数の候補解を生成させ，外部の評価関数や探索戦略に基づいて最適解を選びつつ，自己修正により再度生成させる．このような生成と評価の反復は科学的仮説探索など様々な領域で活用され [32]，歌詞自動翻訳においても，歌唱可能性に関する評価関数を用いる方法が提案されている [14]．

本研究では，LLM を用いた反復生成と外部評価関数の組み合わせにより，phonemic translation を実現する．ただし，Self-refine が LLM 自身にフィードバック文を生成させるのに対し，本研究ではフィードバック文はテンプレートとして用意しておき，再生成に係るコンテキストの部分だけが変数となり，各反復ステップごとに更新される．つまり，LLM はあくまで与えられたプロンプトに基づき，新しい翻訳歌詞候補を生成するためのものであり，評価やスコア計算は外部の評価モデルが担う．

Algorithm 1 Iterative phonemic translation for singable lyrics

Require: Input:
Vocal score of original song $S^{(\mathrm{org})}$
L-line lyrics of original song $W^{(\mathrm{org})}$
Require: Precondition, resources, parameters:
LLM as a translation model LLM$(\cdot)$
Score function Scorer$(\cdot)$
Prompt generator PromptGen$(\cdot)$
Lyric-score syllabification Syllabificator$(\cdot)$
Singing voice synthesizer VoiceSynthesizer$(\cdot)$
Text prompt for initial translation $P^{(\mathrm{init})}$
Text prompt for re-translation $P^{(\mathrm{re})}$
The number of translation candidates N_{cand}
The number of translation iterations N_{iter}
Ensure: Output:
Vocal score of translated song $S^{(\mathrm{trn})}$
L-line lyrics of translated song $W^{(\mathrm{trn})}$
Singing voice of translated song $V^{(\mathrm{trn})}$

$\mathcal{I} = \{1, \ldots, N_{\mathrm{iter}}\}$
$\mathcal{J} = \{1, \ldots, N_{\mathrm{cand}}\}$
▷ Initial translation
$\{W_j^{(\mathrm{trn},1)}\}_{j\in\mathcal{J}} \leftarrow \mathrm{LLM}(W^{(\mathrm{org})}, P^{(\mathrm{init})})$
for $i = 1$ to N_{iter} **do**
▷ Scoring and selecting best results
$j^* \leftarrow \arg\max_{j\in\mathcal{J}} \textsc{Scorer}(S^{(\mathrm{org})}, W^{(\mathrm{org})}, W_j^{(\mathrm{trn},i)})$
$W_{\mathrm{best}}^{(\mathrm{trn},i)} \leftarrow W_{j^*}^{(\mathrm{trn},i)}$
$\mathrm{score}_i \leftarrow \textsc{Scorer}(S^{(\mathrm{org})}, W^{(\mathrm{org})}, W_{\mathrm{best}}^{(\mathrm{trn},i)})$
▷ Re-translation
$P_{i+1}^{(\mathrm{re})} \leftarrow \textsc{PromptGen}(\mathrm{score}_i, W_{\mathrm{best}}^{(\mathrm{trn},i)}, P_i^{(\mathrm{re})})$
$\{W_j^{(\mathrm{trn},i+1)}\}_{j\in\mathcal{J}} \leftarrow \mathrm{LLM}(W^{(\mathrm{org})}, P_{i+1}^{(\mathrm{re})})$
end for
▷ Making score and synthesizing voice
$i^* \leftarrow \arg\max_{i\in\mathcal{I}} \mathrm{score}_i$
$W^{(\mathrm{trn})} \leftarrow W^{(\mathrm{trn},i^*)}$
$S^{(\mathrm{trn})} \leftarrow \textsc{Syllabificator}(W^{(\mathrm{trn})}, S^{(\mathrm{org})})$
$V^{(\mathrm{trn})} \leftarrow \textsc{VoiceSynthesizer}(S^{(\mathrm{trn})})$
return $W^{(\mathrm{trn})}, S^{(\mathrm{trn})}, V^{(\mathrm{trn})}$

3. 提案手法

提案手法の全体手順を Algorithm 1 に示す．提案法は，原曲のボーカル譜 $S^{(\mathrm{org})}$ と原詞 $W^{(\mathrm{org})}$ を入力とし，(1) LLM による訳詞候補の生成，(2) 外部評価器による訳詞スコアリング，(3) スコアに基づく再翻訳（自己改善）の反復，により，意味・自然さに加えて歌唱制約（音節数）および音声的類似性（発音転写度）を同時に満たす訳詞を探索する．以降の節では，アルゴリズム中の変数表記に従って詳細

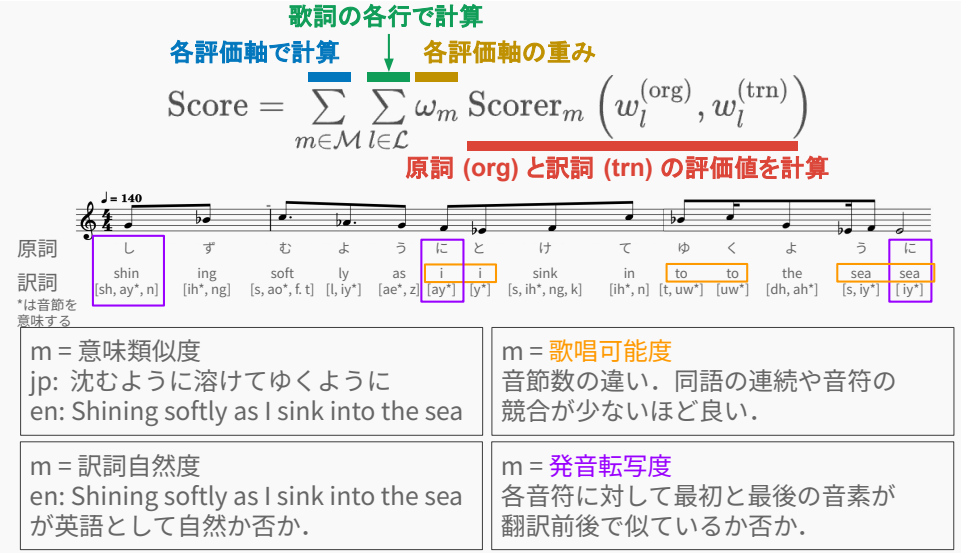


図 2　訳詞品質評価器の概要

を述べる．一連のコードはプロジェクトページ[*4] にて公開している．

3.1　初期訳詞生成

原詞 $W^{(\text{org})}$ と初期プロンプト $P^{(\text{init})}$ から，反復に用いる初期訳詞候補 $\{W_j^{(\text{trn},1)}\}$ を複数生成する翻訳には LLM を用いる．

$$\{W_j^{(\text{trn},1)}\}_{j\in\mathcal{J}} = \text{LLM}\left(W^{(\text{org})}, P^{(\text{init})}\right) \qquad (1)$$

初期プロンプトはプロの歌詞翻訳家という役割指定，翻訳タスク定義などの記述から構成される．プロンプトの骨子は付録 A.1 を参照されたい．

3.2　評価と翻訳の反復

訳詞候補の評価，最良候補の選択，再翻訳を N_{iter} 回だけ反復する．

3.2.1　訳詞品質評価器

原詞と訳詞の対から翻訳品質を評価する．評価基準は以下の 4 つであり，概要を図 2 に示す．

(1) **意味類似度** $\text{Scorer}_{\text{sem}}(\cdot)$：通常の翻訳のように，翻訳前後の文の意味の近さを評価する．
(2) **訳詞自然度** $\text{Scorer}_{\text{nat}}(\cdot)$：翻訳先言語の文としての訳詞の自然さを評価する．
(3) **歌唱可能度** $\text{Scorer}_{\text{sing}}(\cdot)$：原メロディに載せたときに訳詞が歌唱可能かを評価する．
(4) **発音転写度** $\text{Scorer}_{\text{pron}}(\cdot)$：原詞と訳詞の間で，発音の一致度を評価する．

各評価は原詞と訳詞の行ごとに行い，その総和をスコアとする．l 行目の原詞と訳詞をそれぞれ $w_l^{(\text{org})}, w_l^{(\text{trn})}$，行のインデックス集合を $\mathcal{L} = \{1, \ldots, L\}$，評価基準の添字集合を $\mathcal{M} = \{\text{sem}, \text{nat}, \text{sing}, \text{pron}\}$ とすると，原詞 $W^{(\text{org})} = \{w_l^{(\text{org})}\}_{l\in\mathcal{L}}$ と訳詞 $W^{(\text{trn})} = \{w_l^{(\text{trn})}\}_{l\in\mathcal{L}}$ の間のスコアは，

$$\text{Scorer}\left(W^{(\text{org})}, W^{(\text{trn})}\right) = \sum_{m\in\mathcal{M}} \sum_{l\in\mathcal{L}} \omega_m \text{Scorer}_m \left(w_l^{(\text{org})}, w_l^{(\text{trn})} \right) \qquad (2)$$

で計算される．ω_m は各評価基準の値の重みである．特定の行において発音転写を優先するなどのために，ω_m の値を行依存にすることも可能だが，本論文では一定値とする．i 番目の反復においては，訳詞候補 $\{W_j^{(\text{trn},i)}\}_{j\in\mathcal{J}}$ の各要素に対し，$W^{(\text{org})}$ の間で上記のスコアを計算する．

以降では各評価基準の詳細を述べる．ただし簡単化のため，行のインデックス l を省略した $w^{(\text{org})}, w^{(\text{trn})}$ を引数として表記する．

意味類似度． 学習済みの文埋め込みモデルを用いて，文埋め込みベクトル間のコサイン類似度 $\text{cossim}(\cdot)$ を計算する．多言語に対応する文埋め込みモデルを $\text{SentenceEmb}(\cdot)$ とすると，以下の式で定義される．

$$\boldsymbol{e}^{(\text{org})} = \text{SentenceEmb}\left(w^{(\text{org})}\right) \qquad (3)$$
$$\boldsymbol{e}^{(\text{trn})} = \text{SentenceEmb}\left(w^{(\text{trn})}\right) \qquad (4)$$

$$\text{Scorer}_{\text{sem}}(w^{(\text{org})}, w^{(\text{trn})}) = \text{cossim}\left(\boldsymbol{e}^{(\text{org})}, \boldsymbol{e}^{(\text{trn})}\right) \qquad (5)$$

訳詞自然度． LLM-as-a-judge [33] の枠組みを用

[*4] https://github.com/takamichi-lab/sakai26mus_phonemictranslation.git

いて，文の自然さを LLM に評価させる．この尺度は，LLM による訳詞が文として自然になることを目的としている．

$$\mathrm{Scorer}_{\mathrm{nat}}(w^{(\mathrm{org})}, w^{(\mathrm{trn})}) = \mathrm{LLM}\left(w^{(\mathrm{trn})}, P^{(\mathrm{nat})}\right) \tag{6}$$

評価用プロンプト $P^{(\mathrm{nat})}$ の骨子は付録 A.2 に，詳細はプロジェクトページを参照されたい．

歌唱可能度． 文の間の音節数一致度を計算する．文に含まれる音節数をカウントする関数を $\mathrm{SylCount}(\cdot)$ とすると，以下の式で定義される．

$$\mathrm{Scorer}_{\mathrm{sing}}(w^{(\mathrm{org})}, w^{(\mathrm{trn})}) = 1 - \frac{\left|\mathrm{SylCount}\left(w^{(\mathrm{org})}\right) - \mathrm{SylCount}\left(w^{(\mathrm{trn})}\right)\right|}{\mathrm{SylCount}\left(w^{(\mathrm{org})}\right)} \tag{7}$$

発音転写度． 原詞と訳詞の発音の近さを，音符頭と音符末における音素の一致数として評価する．原詞・訳詞をそれぞれ音素列へ変換したのち，音素列と音符列の対応（どの音素がどの音符に属するか）をとる．原詞側については，書記素（日本語歌詞ならばひらがな）列と音符列の対応が与えられているため，音素列と書記素列の対応をとることで音素列と音符列の対応をとることができる．訳詞側については，後述する譜割関数を用いて書記素列と音符列の対応をとり，同様に実現される．

評価対象の原詞の行に対応する音符の集合を $\mathcal{K}$ とすると，対応をとった音素集合は，

$$\left\{\left\{p_{k,m}^{(\mathrm{org})}\right\}_{m=1,\ldots,M_k^{(\mathrm{org})}}\right\}_{k\in\mathcal{K}} \tag{8}$$

$$\left\{\left\{p_{k,m}^{(\mathrm{trn})}\right\}_{m=1,\ldots,M_k^{(\mathrm{trn})}}\right\}_{k\in\mathcal{K}} \tag{9}$$

と表記される．原詞側 $p_{k,m}^{(\mathrm{org})}$ は，音符 k に含まれる m 番目の音素，$M_k^{(\mathrm{org})}$ は音符 k に含まれる音素数である．訳詞側 $p_{k,m}^{(\mathrm{trn})}$ についても同様に定義される．

ある原詞側音素と訳詞側音素の音素類似度関数 $\mathrm{match}(p^{(\mathrm{org})}, p^{(\mathrm{trn})})$ は，その音素対が知覚的に近ければ 1，それ以外は 0 を返すものとする．この関数を用いて，ある音符 k における類似度 $\mathrm{score}_{\mathrm{pron},k}$ を計算する．具体的には，音符頭と音符末の音素類似度により

$$s_k^{(\mathrm{beg})} = \mathrm{match}\left(p_{k,1}^{(\mathrm{org})}, p_{k,1}^{(\mathrm{trn})}\right), \tag{10}$$

$$s_k^{(\mathrm{end})} = \mathrm{match}\left(p_{k,M_k^{(\mathrm{org})}}^{(\mathrm{org})}, p_{k,M_k^{(\mathrm{trn})}}^{(\mathrm{trn})}\right) \tag{11}$$

$$\mathrm{score}_{\mathrm{pron},k} = \frac{1}{2}(s_k^{(\mathrm{beg})} + s_k^{(\mathrm{end})}) \tag{12}$$

と定義する．

以上より，発音転写度は以下のように定義される．

$$\mathrm{Scorer}_{\mathrm{pron}}(w^{(\mathrm{org})}, w^{(\mathrm{trn})}) = \frac{1}{|\mathcal{K}|}\sum_{k\in\mathcal{K}} \mathrm{score}_{\mathrm{pron},k} \tag{13}$$

3.2.2 最良の訳詞を選択

i 番目の反復において，訳詞候補 $\{W_j^{(\mathrm{trn},i)}\}_{j\in\mathcal{J}}$ から最良の訳詞 $W_{\mathrm{best}}^{(\mathrm{trn},i)}$ を選択する．具体的には

$$j^* = \arg\max_{j\in\mathcal{J}} \mathrm{Scorer}\left(W^{(\mathrm{org})}, W_j^{(\mathrm{trn},i)}\right), \tag{14}$$

$$W_{\mathrm{best}}^{(\mathrm{trn},i)} = W_{j^*}^{(\mathrm{trn},i)}, \tag{15}$$

で計算される．

3.2.3 再翻訳用のプロンプト生成

最良訳詞 $W_{\mathrm{best}}^{(\mathrm{trn},i)} = \{w_l^{(\mathrm{trn},i)}\}_{l\in\mathcal{L}}$ を用いて再翻訳用プロンプトを生成する．訳詞の各行に対する各評価基準の値を

$$s_{m,l} = \mathrm{Scorer}_m\left(w_l^{(\mathrm{org})}, w_l^{(\mathrm{trn},i)}\right) \tag{16}$$

と表記する．再翻訳用プロンプト $P_{i+1}^{(\mathrm{re})}$ は，初期プロンプトの骨子を継承しつつ，最良訳詞 $W_{\mathrm{best}}^{(\mathrm{trn},i)}$ と評価値 $\{s_{m,l}^{(i)}\}$ をプロンプト生成器 $\mathrm{PromptGen}(\cdot)$ に入力し，評価値の引く行に対して改善を促す．

$$P_{i+1}^{(\mathrm{re})} = \mathrm{PromptGen}\left(W^{(\mathrm{org})}, W_{\mathrm{best}}^{(\mathrm{trn},i)}, \{s_{m,l}\}_{m\in\mathcal{M},\, l\in\mathcal{L}}\right) \tag{17}$$

プロンプト生成器はテンプレート文の穴埋めである．具体的なテンプレートは付録 A.3 参照とする．

3.2.4 再翻訳

原詞 $W^{(\mathrm{org})}$ と再翻訳用プロンプト $P_{i+1}^{(\mathrm{re})}$ から，次の反復の候補を生成する．

$$\{W_j^{(\mathrm{trn},i+1)}\}_{j\in\mathcal{J}} = \mathrm{LLM}(W^{(\mathrm{org})}, P_{i+1}^{(\mathrm{re})}) \tag{18}$$

再翻訳用プロンプトには i 番目の反復における翻訳結果とそのスコアが含まれるため，$i+1$ 番目の反復においてその改善が期待される．

3.2.5 訳詞の最終決定

全ての反復を終えた後，全ての反復において最も高いスコアを獲得した訳詞を，最終的な訳詞 $W^{(\mathrm{trn})}$ とする．

$$W^{(\mathrm{trn})} = \arg\max_{i\in\mathcal{I}} \mathrm{Scorer}\left(W^{(\mathrm{org})}, W_{\mathrm{best}}^{(\mathrm{trn},i)}\right) \tag{19}$$

表 1　対象楽曲（日英対応）

日本語楽曲	英語楽曲
夜に駆ける [8]	Into the Night [9]
夢灯籠 [34]	Dream Lantern [35]
レット・イット・ゴー ～ありのままで～ [36]	Let It Go [37]
ラブ・ストーリーは突然に [38]	なし

3.3　楽譜生成と歌声合成

3.3.1　楽譜生成

原曲楽譜 $S^{(\mathrm{org})}$ の旋律を保持したまま，最終的な訳詞 $W^{(\mathrm{trn})}$ を音符列へ割り当てた歌詞付き楽譜 $S^{(\mathrm{trn})}$ を生成する．譜割関数（音節割当）を Syllabificator$(\cdot)$ とすると，

$$S^{(\mathrm{trn})} = \mathrm{Syllabificator}\left(S^{(\mathrm{org})}, W^{(\mathrm{trn})}\right) \qquad (20)$$

と定義される．具体的にこの関数は，$W^{(\mathrm{org})}$ と $W^{(\mathrm{trn})}$ の各行 $w^{(\mathrm{org})}$ と $w^{(\mathrm{trn})}$ に基づいて以下の処理を行う．

- $\mathrm{SylCount}\left(w^{(\mathrm{org})}\right) = \mathrm{SylCount}\left(w^{(\mathrm{trn})}\right)$: 原詞と訳詞の各音節が 1 対 1 に対応するように，原曲楽譜に訳詞を割り当てる．
- $\mathrm{SylCount}\left(w^{(\mathrm{org})}\right) > \mathrm{SylCount}\left(w^{(\mathrm{trn})}\right)$: 訳詞のうち母音で終わる音節を無作為に選択し，当該母音をメリスマ（母音伸長）として扱うことで音節の不足分を補う．
- $\mathrm{SylCount}\left(w^{(\mathrm{org})}\right) < \mathrm{SylCount}\left(w^{(\mathrm{trn})}\right)$: 複数音節を無作為に選択し，同一音符に割り当てる．ただし，1 つの音符に割り当てる最大音節数は 2 とする．

3.3.2　歌声合成

学習済みの歌声合成 (singing voice synthesis; SVS) モデルを用いて，訳詞付き楽譜 $S^{(\mathrm{trn})}$ から，歌声 $V^{(\mathrm{trn})}$ を合成する．

$$V^{(\mathrm{trn})} = \mathrm{VoiceSynthesizer}\left(S^{(\mathrm{trn})}\right) \qquad (21)$$

4.　実験的評価

4.1　実験条件

日英歌詞翻訳をタスクとして，提案手法を評価した．

4.1.1　翻訳対象の原曲

原曲として表 1 に示す日本語楽曲 4 曲を用い，各曲の 1 番のみを翻訳対象とした．原詞[*5]は

今、静かな夜の中で（いま、しずかなよるのなかで）
無計画に車を走らせた（むけいかくにくるまをはしらせた）
…

のようにひらがなを併記した．これは音節化と音素化を簡便にするためである．

[*5] 歌詞は YOASOBI『夜に駆ける』の一部

表 2　英語音素と日本語音素の対応関係

英語音素	日本語音素	英語音素	日本語音素	英語音素	日本語音素
AA	a, o	UH	u, o	N	n
AE	a, e	UW	u	NG	n
AH	a, o	B	b	P	p
AO	o	CH	ch, t	R	r
AW	au, a	D	d	S	s, sh
AY	ae, ai, a	DH	z	SH	s, sh
EH	a, e	F	h	T	t, ts
ER	a	G	g	TH	s
EY	e, i	HH	h	V	b
IH	i, e, i	JH	z	W	w
IY	i	K	k	Y	y
OW	ou, o	L	r	Z	z
OY	oe, oi, o	M	m	ZH	z

4.1.2　LLM による訳詞候補生成の設定

LLM として ChatGPT-4.1-nano[*6] を用いた．訳詞候補数を $N_{\mathrm{cand}} = 3$，反復回数を $N_{\mathrm{iter}} = 100$ とした．各訳詞候補は生成温度 $T \in \{0, 0.5, 1.0\}$ をそれぞれ割り当てて生成した．

4.1.3　訳詞品質評価器の実装

意味類似度 $\mathrm{Scorer}_{\mathrm{sem}}(\cdot)$**.** SentenceTransformer の `paraphrase-multilingual-MiniLM-L12-v2` [*7] を用いた．

訳詞自然度 $\mathrm{Scorer}_{\mathrm{nat}}(\cdot)$**.** 訳詞の自然さを LLM により $[0, 1]$ で評価した．

歌唱可能度 $\mathrm{Scorer}_{\mathrm{sing}}(\cdot)$**.** 日本語の音節数は，ひらがな 1 文字を 1 音節とみなし，ひらがなの文字数で定義した．この定義では，拗音（ゃ・ゅ・ょ），促音（っ），撥音（ん）もそれぞれ 1 文字として数えられ，音韻論的な音節やモーラの定義とは必ずしも一致しない．英語の音節数は，単語を CMUdict [*8] により音素列へ変換し，強勢付き母音（0, 1, 2 のいずれかの番号を含む母音音素）の個数として求めた．辞書外語に対しては，pyphen[*9] による音節分割を併用し補完した．

発音転写度 $\mathrm{Scorer}_{\mathrm{pron}}(\cdot)$**.** 日本語の音素は pyopenjtalk[*10] を用いて得る．英語の音素 CMU dict を用いて得る．知覚的に近い音素対は表 2 のように定義した．

[*6] https://platform.openai.com/docs/models/gpt-4.1-nano
[*7] https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2
[*8] https://www.nltk.org/_modules/nltk/corpus/reader/cmudict.html
[*9] https://pyphen.org/
[*10] https://github.com/r9y9/pyopenjtalk

4.1.4 各評価基準値の重み

本研究では，重み付けの違いが訳詞および歌唱結果へ与える影響を評価するため，以下の範囲で重みを変化させる．

- $\omega_{\text{sem}} \in \{1, 3, 5\}$
- $\omega_{\text{nat}} = 1$
- $\omega_{\text{sing}} \in \{0, 1, 3, 5\}$
- $\omega_{\text{pron}} \in \{0, 1, 3, 5\}$

重み付けの変化の仕方としては基準重み

$$(\omega_{\text{sem}}, \omega_{\text{nat}}, \omega_{\text{sing}}, \omega_{\text{pron}}) = (1, 1, 1, 1) \qquad (22)$$

から 1 つの評価基準のみを変更した条件に限定して比較した．

4.1.5 楽譜合成と歌声合成

楽譜のファイルフォーマットには MusicXML を用い，楽譜合成は原曲楽譜の MusicXML に含まれる原詞を訳詞に置換することで実装した．歌声合成には Sinsy [39] を用いた．

4.1.6 客観評価・主観評価の条件

主観評価に用いる刺激数と構成．主観評価では，各楽曲について，重み付け条件の異なる訳詞およびそれに対応する合成歌唱音声を提示する．提示形式は (i) 行単位（line-wise）に分割した提示，および (ii) 楽曲の複数行をまとめたセクション単位（section-wise）の提示の 2 種類とした．刺激提示順は参加者ごとにランダム化した．

客観評価．提案手法における重み付けが生成結果へ与える影響を定量的に分析した．重み付け条件ごとに訳詞を生成し，意味類似度・訳詞自然度・歌唱可能度・発音転写度の各指標を算出した．目的は，重みの変更により，所望した評価軸のスコアが変化するか，および評価軸間のトレードオフが生じるかを明示的に確認することである．なお，客観評価では歌声合成は行わず，翻訳テキストと評価器出力のみを分析対象とした．

主観評価の実施条件．主観評価は，日本語と英語の双方に十分な知見を有する大学生 11 名を対象に実施した．各刺激では，日本語歌唱音声・英語歌唱音声をそれぞれの各チャンネルに配したステレオ歌声を提示し，順番や回数などの聴取方法は参加者が選べる形とした．評価は 5 段階評定で行い，行単位提示では以下の 4 項目を回答させた：

(1) 原詞と訳詞の意味の一致度（Semantic similarity）
(2) 訳詞の英語としての自然さ（Sentence naturality）
(3) 英語歌声が歌声としてなめらかに聞こえるか（歌唱可能性：Singability）
(4) 日本語歌声と比較したときの英語歌声の発音の近さ（発音転写度：Pron. transferability）

一方，セクション単位提示では，上記の（1），(2)，(4）の 3 項目を評価させた．

4.2 結果

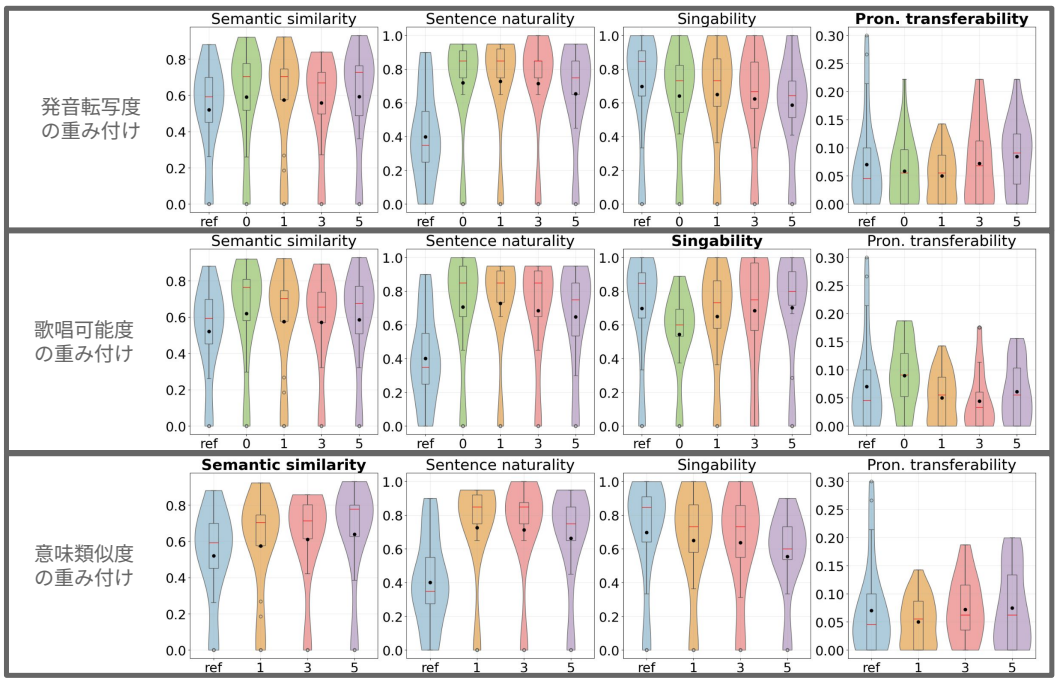
以降の節では（1）自己修正プロセスに伴うスコアの推移，(2）各評価軸の重み付けが訳詞品質に与える影響，(3）人手による主観評価の結果について報告する．また，自動評価スコアと主観評価の対応関係についても検討する．

4.2.1 自己修正プロセスによるスコア推移

提案手法における反復的な自己修正が訳詞品質に与える影響を検証するため，更新回数に伴うスコア（式 (2) により算出）の推移を分析した．図 4 に異なる試行におけるスコア変動の様子を示す．例示するように，多くの試行において更新を重ねるにつれてスコアは右肩上がりに推移し，初期生成に比べて品質が改善される様子が確認された．これは，LLM が自身の生成した訳詞に対する評価フィードバックを受け取り，より高いスコアとなる表現を探索できていることを示唆する．しかしながら，右肩上がりの改善が必ず保証されるとは限らない．同図では，ある反復回数（この例では 20 回目付近）でスコアはピークに達したのち乱高下している．そのため，スコアのピークの発生タイミングは予測困難である可能性があり，反復回数に基づく終了条件は，より良いスコアを逃すリスクがあることが明らかになった．

4.2.2 客観評価

重み付けと訳詞品質の関係を分析するため，発音転写度（ω_{pron}），歌唱可能度（ω_{sing}），意味類似度（ω_{sem}）の重みをそれぞれ変化させた条件で生成した訳詞を比較した．ここでは，4 曲のうち最も行数が多い『**夜に駆ける**』を代表例として述べる．図 3 は，各重み設定における行単位スコアの分布を示す．他の曲の結果は付録 A.4 を参照されたい．なお，これから述べる考察は『夜に駆ける』に対するものだが，同様の考察は概ね他の曲にも該当する．**発音転写度の重み** ω_{pron} の増加に伴い発音転写度も上昇し，評価器に基づく最適化が意図した評価軸を確かに強化できることが確認された．**歌唱可能度の重み** ω_{sing} と**意味類似度の重み** ω_{sem} でも同様の対応が確認された．これらの結果は，提案手法が重みという少数のパラメータを介して，訳詞生成の優先順位を制御可能であることを示す．また，$\omega_{\text{pron}} = 0$ あるいは $\omega_{\text{sing}} = 0$ の条件では，人間による既存の訳詞の ref と比較して当該スコアが低下する傾向が見られ，提案した評価器が「音（発音転写・音節適合）を意識



5

図 3 重み変更の際のスコアの違い．上から順に発音転写度，歌唱可能性，意味類似度の重みを変更したものと，アーティスト本人が作詞した英語歌詞 (ref) との比較．（夜に駆ける）

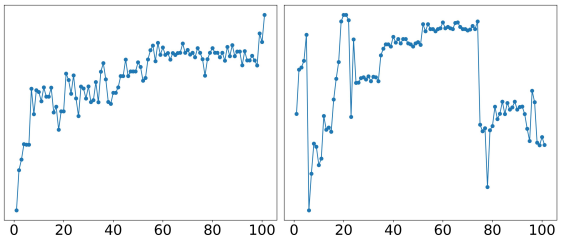


図 4 自己修正ループに伴うスコアの推移例．順調に上昇した場合（左）と変動が見られる場合（右）．横軸：更新数．縦軸：スコア．

表 3 発音転写度の重み付けの変更に対する主観評価結果．

mode	Weight	Sem sim.	Sent nat.	Sing.	Pron tran.
line	ref	3.194	3.744	3.483	2.950
	$\omega_{\text{pron}} = 0$	3.915	3.894	2.928	2.290
	$\omega_{\text{pron}} = 5$	3.851	4.080	3.186	2.678
section	ref	4.194	4.134	3.222	–
	$\omega_{\text{pron}} = 0$	4.366	4.391	2.633	–
	$\omega_{\text{pron}} = 5$	4.250	3.583	2.967	–

表 4 歌唱可能度の重み付けの変更に対する主観評価結果．

mode	Weight	Sem sim.	Sent nat.	Sing.	sing tran.
line	ref	3.194	3.744	3.483	2.950
	$\omega_{\text{sing}} = 0$	3.809	4.112	3.067	2.303
	$\omega_{\text{sing}} = 5$	3.580	3.811	3.216	2.363
section	ref	4.194	4.134	3.222	–
	$\omega_{\text{sing}} = 0$	3.975	4.042	2.800	–
	$\omega_{\text{sing}} = 5$	3.167	3.354	1.979	–

した歌詞翻訳」と「そうでない翻訳」をある程度弁別できることが示唆される．

次に評価軸間の関係に着目すると，ω_{pron} と ω_{sing} の間にはトレードオフが観察された．発音転写度を高めるためには，原詞に近い音素系列をもつ語を優先して選択する必要がある一方，歌唱可能度を高めるためには音節数や伸ばし方を旋律へ適合させる必要がある．実際には，両者を同時に満たす語彙・表現の探索空間が限られること，ならびに音節適合のための伸長や語の圧縮が，音素境界の一致を損ね得ることから，一方の改善が他方の低下を招く場合がある．

さらに，意味類似度や訳詞自然度は，発音転写度・歌唱可能度の重みを変化させても大きくは変動しない傾向が見られた．これは，発音転写度および音節一致度の評価が主として音韻・韻律情報に基づくため，意味・自然さとは比較的独立な制約として働くことを示唆する．一方で，意味類似度の重みを強めた場合には，意味類似度が向上するのと引き換えに，歌唱可能度が低下する傾向が見られ，意味忠実性と歌唱可能性の間に競合が生じ得ることが確認された．

4.2.3 主観評価

4.1.6 節での説明に従い，実施した主観評価結果を表 3 および表 4 に示す．これらの表は人間が作詞した英語歌詞（ref），該当する重みを 0 に設定して生成した翻訳歌詞（zero），該当する重みを非 0（ここ

では 5）に設定して生成した翻訳歌詞（not zero）に対しての主観評価結果である．行単位（line）では，発音転写度あるいは歌唱可能度の重み付けを導入することで，当該観点の評定が改善する傾向が見られた．ただし，全体として人間が訳した既存の ref に匹敵する水準には達しておらず，提案手法は音韻・韻律を一定程度改善し得るものの，人手翻訳に見られる高度な作詞的工夫を完全に代替するには課題が残る．一方で，意味類似度および英語としての自然さについては，ref よりも LLM 生成訳詞の評定が高い傾向が観察された．これは，人手の歌詞翻訳が旋律への適合や歌としての響きを優先することで，意味の直接性や通常文としての流暢さを意図的に犠牲にする場合があることに起因すると考えられる．

セクション単位（section）では，特に歌唱可能度を強く意識させた条件において，意味一致度・自然さが低下する傾向が見られた．行単位では局所的な言い換えで制約を満たしやすい一方，セクション全体では語彙選択や韻律調整の影響が累積したことで文脈的一貫性や自然な言い回しが損なわれたと考える．この結果は，局所最適（行単位の制約充足）と大域的品質（セクション全体の可読性・一貫性）が必ずしも一致しないことを示している．

4.2.4 主観評価による客観評価の妥当性

客観評価（自動評価器）と主観評価（知覚評定）の対応関係を検討する．発音転写度の重みを増加させた条件では，客観指標・主観評価の両方で「発音の近さ」が改善する傾向が確認された．歌唱可能度の重み付けでも同様の結果が得られた．これらは，本研究で設計した評価器が，少なくとも重み付けによる相対比較の観点では，人間の知覚と整合的な方向へ生成を誘導し得ることを支持する．

5. 課題と展望

本研究の実験結果を通じて，自動歌詞翻訳において以下の 3 つの技術的課題が明らかとなった．

- 生成の安定性と頑健性
- 更新停止条件（閾値）の設計
- 楽譜合成と歌声合成の品質

第一に，**生成の安定性と頑健性**である．結果で述べた通り，LLM を用いた自己修正プロセスは出力のたびに異なる軌跡を辿る場合がある．LLM の確率的な生成挙動により，同じ入力であっても更新結果が一貫しないため，実用化においてはより頑健なプロンプト設計や few-shot 事例の最適化などにより，意図した修正を確実に実行させる制御技術が求められる．また，LLM の内部パラメータ自体を強化学習等でファインチューニングし，評価関数に対する感度を高めるアプローチも検討に値する．そして，**更新停止条件（閾値）の設計**である．スコア推移の分析（図 4）から示唆されるように，固定回数の更新では必ずしも最高スコアの訳詞が得られるとは限らず，ピーク後の過剰な修正が品質劣化や振動を招く場合がある．したがって，スコアの改善幅が一定以下になった時点で打ち切る early stopping の導入などが必要である．最後に，**楽譜合成と歌声合成の品質**である．本研究ではテキストベースでの音素・音節評価を行ったが，実際の歌唱可能性を担保するには，楽譜から得られる音符の長短やピッチ情報との厳密なアライメントが不可欠である．今後は，本研究で扱った譜割での歌声合成と人間の手動による譜割での歌声合成の違いの検討も必要である．

6. まとめ

本研究では，大規模言語モデルと自動評価器を用いた自己修正ループにより，原詞の発音転写と歌唱可能性を考慮した歌詞翻訳手法を提案した．評価実験の結果，提案手法は重み付けパラメータによって訳詞の特性（発音重視，歌唱性重視など）を制御可能であり，自己修正を繰り返すことで総合的なスコアを向上させ得ることが確認された．一方で，更新プロセスの安定性や停止条件の決定，および人手翻訳レベルの韻律的工夫の再現には課題が残る．今後は，より高度な音楽的制約を扱えるマルチモーダルな評価系の構築や，生成の安定化に向けた最適化手法の改善に取り組む．

謝辞： 本研究を進めるにあたり，相談させていただいた青山学院大学内田洋子先生に感謝申し上げます．本研究は，JSPS 科研費 23K28108 の支援を受けて実施した．

参考文献

[1] I. Cross, “The evolutionary nature of musical meaning,” *Musicae Scientiae*, vol. 13, no. 2_suppl, pp. 179–200, 2009.
[2] P. Low, “Singable translations of songs,” *Perspectives: Studies in Translation Theory and Practice*, vol. 11, no. 2, pp. 87–103, 2003.
[3] H. Kim, K. Watanabe, M. Goto, and J. Nam, “A computational evaluation framework for singable lyric translation,” in *ISMIR 2023*, 2023, pp. 774–781.
[4] H. Kim, J. Jung, D. Jeong, and J. Nam, “K-pop lyric translation: Dataset, analysis, and neural-modelling,” in *LREC-COLING 2024*. ELRA and ICCL, May 2024, pp. 9974–9987.
[5] M. Mateo, “Music and translation,” in *Handbook of Translation Studies*, Y. Gambier and L. van

Doorslaer, Eds. John Benjamins, 2012, vol. 3, pp. 115–121.
[6] A. Lefevere, *Translating Poetry: Seven Strategies and a Blueprint*. Van Gorcum, 1975.
[7] S. Hervey and I. Higgins, *Thinking Translation: A Course in Translation Method: French to English*. Routledge, 1992.
[8] Ayase, "夜に駆ける," YOASOBI, digital single, 作詞・作曲：Ayase.
[9] Ayase and K. Aoki, "Into the night," YOASOBI, English version release, 作詞・作曲：Ayase／英詞：Konnie Aoki.
[10] L. Ou, X. Ma, M.-Y. Kan, and Y. Wang, "Songs across borders: Singable and controllable neural lyric translation," in *ACL*, A. Rogers, J. Boyd-Graber, N. Okazaki, and S. Reddy, Eds., Jul. 2023, pp. 447–467.
[11] C. Li, K. Fan, J. Bu, B. Chen, Z. Huang, and Z. Yu, "Translate the beauty in songs: Jointly learning to align melody and translate lyrics," in *EMNLP*, Dec. 2023, pp. 27–39.
[12] Z. Ye, J. Li, and R. Xu, "Sing it, narrate it: Quality musical lyrics translation," in *EMNLP*, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds. Association for Computational Linguistics, Nov. 2024, pp. 5498–5520.
[13] 池田昂太郎, 松平茅隼, 加藤大貴, 平山高嗣, 駒水孝裕, and 井手一郎, "歌詞の自動翻訳のための発音を考慮した訳語選択に関する研究," **電子情報通信学会（*IEICE*）**, 2024.
[14] S. Zhao, B. Li, Y. Tian, and N. Peng, "Reffly: Melody-constrained lyrics editing model," in *ACL*, L. Chiruzzo, A. Ritter, and L. Wang, Eds. Association for Computational Linguistics, Apr. 2025.
[15] H. J. Vermeer, "Ein rahmen für eine allgemeine translationstheorie," *Lebende Sprachen*, vol. 23, no. 3, pp. 99–102, 1978.
[16] J. Franzon, "Choices in song translation," *The Translator*, vol. 14, no. 2, pp. 373–399, 2008.
[17] Ş. Susam-Sarajeva, "Translation and music: Changing perspectives, frameworks and significance," *The Translator*, vol. 14, no. 2, pp. 187–200, 2008.
[18] I. Pilshchikov, "The semiotics of phonetic translation," *Studia Metrica et Poetica*, vol. 3, no. 1, pp. 53–104, 2016.
[19] K. Basu, "Phonetic journey: Sound in singable translations," Master's thesis, University of Victoria, Victoria, BC, Canada, Aug. 2020.
[20] 木石岳, **歌詞のサウンドテクスチャー：うたをめぐる音声詞学論考**. 東京: 白水社, 2023.
[21] F. Guo, C. Zhang, Z. Zhang, Q. He, K. Zhang, J. Xie, and J. Boyd-Graber, "Automatic song translation for tonal languages," in *ACL*, S. Muresan, P. Nakov, and A. Villavicencio, Eds., 2022, pp. 729–743.
[22] M. Jiao, T. Yu, X. Li, G. Qiu, X. Gu, and B. Shen, "On the evaluation of neural code translation: Taxonomy and benchmark," 2023.
[23] B. Gain, D. Bandyopadhyay, and A. Ekbal, "Bridging the linguistic divide: A survey on leveraging large language models for machine translation," 2025.
[24] J. Sun, Y. Tian, W. Zhou, N. Xu, Q. Hu, R. Gupta, J. F. Wieting, N. Peng, and X. Ma, "Evaluating large language models on controlled generation tasks," in *EMNLP*. Association for Computational Linguistics, 2023, pp. 3155–3168.
[25] D. Javorský, O. Bojar, and F. Yvon, "Prompting llms: Length control for isometric machine translation," in *IWSLT*, F. Bianchi, C. Federmann, B. Haddow, C. Herold, M. Negri, S. Peng, and M. Turchi, Eds. Association for Computational Linguistics, Jul. 2025, pp. 108–119.
[26] L. Mei, J. Yao, Y. Ge, Y. Wang, B. Bi, Y. Cai, J. Liu, M. Li, Z.-Z. Li, D. Zhang, C. Zhou, J. Mao, T. Xia, J. Guo, and S. Liu, "A survey of context engineering for large language models," 2025.
[27] A. Madaan, N. Tandon, P. Gupta, S. Hallinan, L. Gao, S. Wiegreffe, U. Alon, N. Dziri, S. Prabhumoye, Y. Yang, S. Gupta, B. P. Majumder, K. Hermann, S. Welleck, A. Yazdanbakhsh, and P. Clark, "Self-refine: Iterative refinement with self-feedback," 2023.
[28] N. Shinn, F. Cassano, E. Berman, A. Gopinath, K. Narasimhan, and S. Yao, "Reflexion: Language agents with verbal reinforcement learning," in *Advances in Neural Information Processing Systems*, 2023.
[29] C. Yang, X. Wang, Y. Lu, H. Liu, Q. V. Le, D. Zhou, and X. Chen, "Large language models as optimizers," in *ICLR*. OpenReview.net, 2024.
[30] S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan, "Tree of thoughts: Deliberate problem solving with large language models," 2023.
[31] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, "Optimization by simulated annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983.
[32] C. Lu, C. Lu, R. T. Lange, J. Foerster, J. Clune, and D. Ha, "The ai scientist: Towards fully automated open-ended scientific discovery," 2024.
[33] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica, "Judging llm-as-a-judge with mt-bench and chatbot arena," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023.
[34] Y. Noda, "夢灯籠," RADWIMPS, soundtrack album 『君の名は。』.
[35] R. B. W. Noda, Yojiro, "Dream lantern (english ver.)," RADWIMPS, 『Your Name. (Deluxe Edition / Original Motion Picture Soundtrack)』, 作詞・作曲：野田洋次郎.
[36] K. Anderson-Lopez and R. Lopez, "レット・イット・ゴー〜ありのままで〜,"『アナと雪の女王（日本版）』挿入歌, 作詞・作曲：Kristen Anderson-Lopez ／ Robert Lopez，日本語訳詞：高橋知伽江.
[37] ——, "Let it go," Song from Disney film Frozen, writers: Kristen Anderson-Lopez and Robert Lopez.
[38] K. Oda, "ラブ・ストーリーは突然に," Single: Oh! Yeah! (coupling / double A-side context varies by catalog).
[39] Y. Hono, K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "Sinsy: A deep neural

network-based singing voice synthesis system," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 2803–2815, 2021.

付　　録

A.1 初期プロンプト（翻訳）$P^{(\mathrm{init})}$ の骨子

本文の初期訳詞生成で用いる翻訳プロンプト $P^{(\mathrm{init})}$ は，以下の要素から構成される．

- **Role instruction:** モデルに「プロの歌詞翻訳者」である役割を与える．
- **Task definition:** 日本語歌詞を自然な英語へ翻訳することを指示する．
- **Structure-preserving constraints:** 入力と出力で行数を一致させ，各行を 1 対 1 で対応付ける．
- **Line tracking requirement:** 検証のため，出力に行番号を付与する．
- **Output-only constraint:** 説明等を付加せず，翻訳文のみを出力する．
- **Output format specification:** `n. <line translation>` 形式で逐次出力する．
- **Exception handling:** 空行は空行のまま保持し，入力に ‘‘`None`’’ を含む行は `None` を厳密に出力する．

完全なプロンプト文（逐語）は，再現性のためプロジェクトページに掲載する．

A.2 LLM-as-a-judge（訳詞自然度評価）プロンプト $P^{(\mathrm{nat})}$ の骨子

訳詞自然度を評価するため，本研究では LLM-as-a-judge に基づく評価用プロンプト $P^{(\mathrm{nat})}$ を用いる．本プロンプトは以下の要素から構成される．

- **Role instruction:** モデルに英語ネイティブ話者かつ厳密な言語評価者としての役割を与える．
- **Evaluation task definition:** 入力された英語文（訳詞）が，ネイティブ話者にとってどの程度自然で流暢であるかを評価する．
- **Evaluation criteria:** 文法的正確さ，語彙選択，コロケーション，および全体的な流暢さを考慮する．
- **Score semantics:** 評価結果を $[0, 1]$ の連続値で表し，0.00 は極めて不自然，1.00 は完全に自然で慣用的な表現を表す．
- **Output constraint:** 小数点以下 2 桁の数値のみを出力し，説明や付加的なテキストを出力しない．

完全なプロンプト文（逐語的記述）は，再現性のためプロジェクトページに掲載する．

A.3 再翻訳プロンプト $P_{i+1}^{(\mathrm{re})}$ の骨子

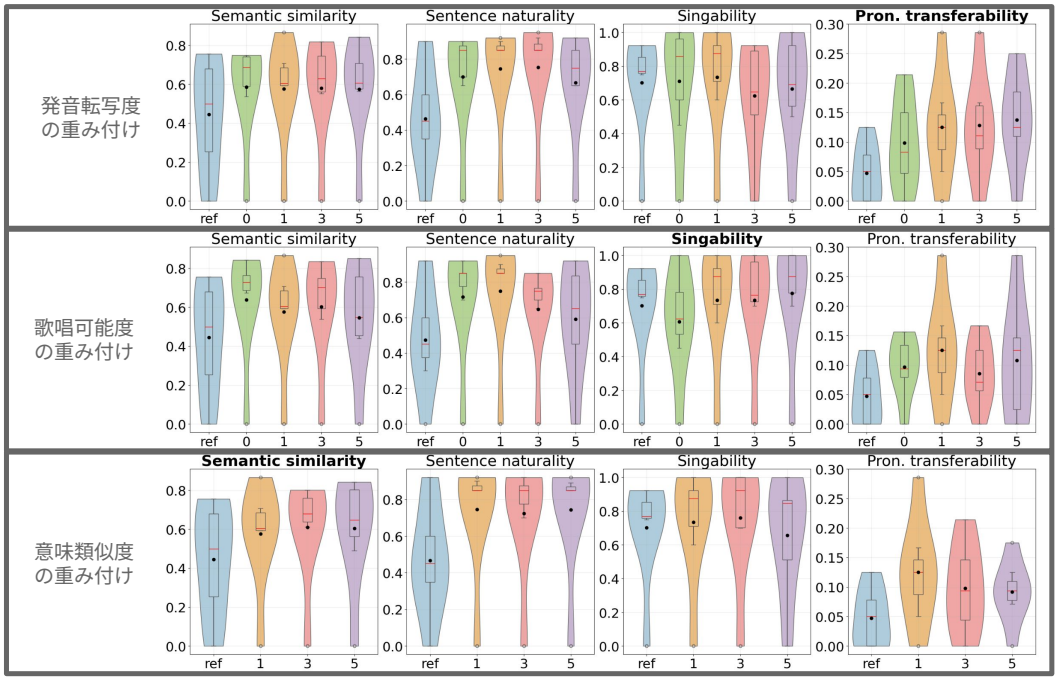
本研究における再翻訳用プロンプト $P_{i+1}^{(\mathrm{re})}$ は，反復的翻訳過程において，直前の最良訳詞とその評価結果を LLM に明示的に提示し，翻訳の改善を促すことを目的として設計されている．本プロンプトは，以下の要素から構成される．

- **Role inheritance:** 初期プロンプトと同様に，モデルに対して「プロの歌詞翻訳者」としての役割を与え，意味と音の調和を重視した翻訳を行うことを求める．
- **Task re-definition:** 原詞の逐語的翻訳ではなく，与えられた制約条件の下で，意味的・音韻的・韻律的により適切な訳詞へと**既存の訳詞を改善する** ことを明示する．
- **Line-wise contextual input:** 各行について，原詞，現在の訳詞，および行ごとの評価情報を入力として与える．評価情報には，意味類似度，歌唱可能度，発音転写度に関する指標が含まれる．
- **Quantitative feedback integration:** 行ごとに与えられた評価値を通じて，どの観点（意味・音節・発音など）が相対的に不足しているかを LLM が把握できるようにし，スコアの低い行に対して重点的な修正を促す．
- **Phonetic and rhythmic constraints:** 各行について，原詞に対応する先頭・末尾音素，許容される語数範囲，および原詞と訳詞の音節数差が提示され，訳詞が原曲の韻律および発音構造と整合するよう制約を課す．
- **Priority specification:** 訳詞自然度と歌唱可能度のバランスを取りつつ，特に音節数差の縮小を優先する方針を明示し，必要に応じて自由度の高い翻訳を許容する．
- **Output constraint inheritance:** 初期プロンプトと同様に，行数の一致，行番号付き出力，付加的説明を含まない翻訳文のみの出力，といった形式的制約を継承する．

完全なプロンプト文（逐語的記述）は，再現性のためプロジェクトページに掲載する．

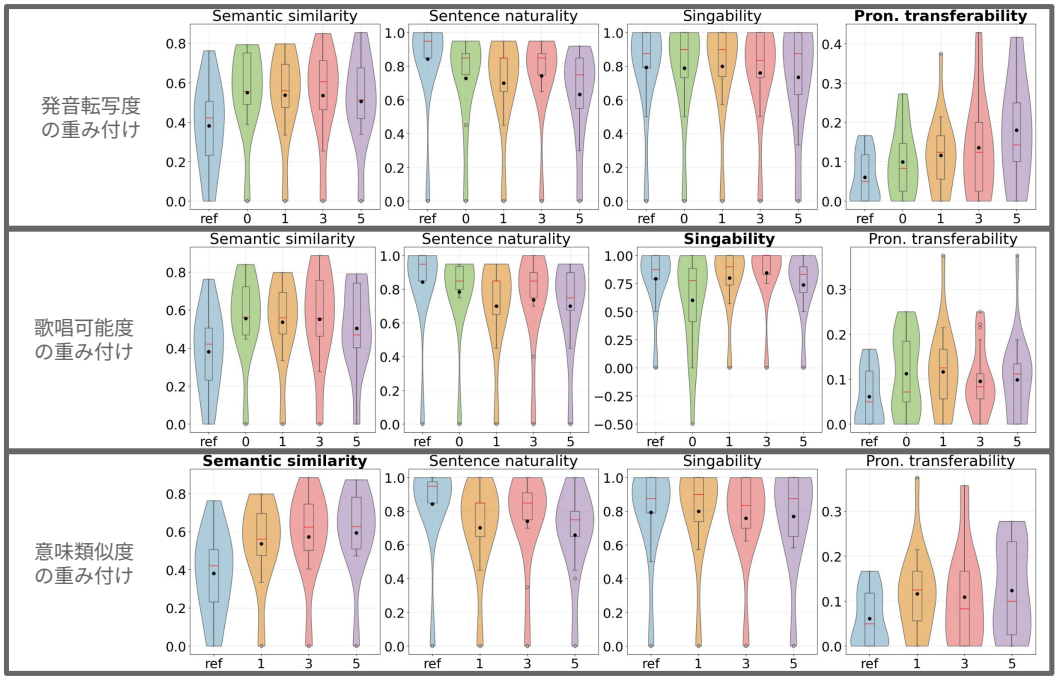
A.4 その他の曲における客観評価の結果

以下では，本実験で使用した『夢灯篭』，『Let it go』，『ラブストーリーは突然に』の歌詞翻訳結果のスコアの違いを示す．図 3 と同じ内容を示している．



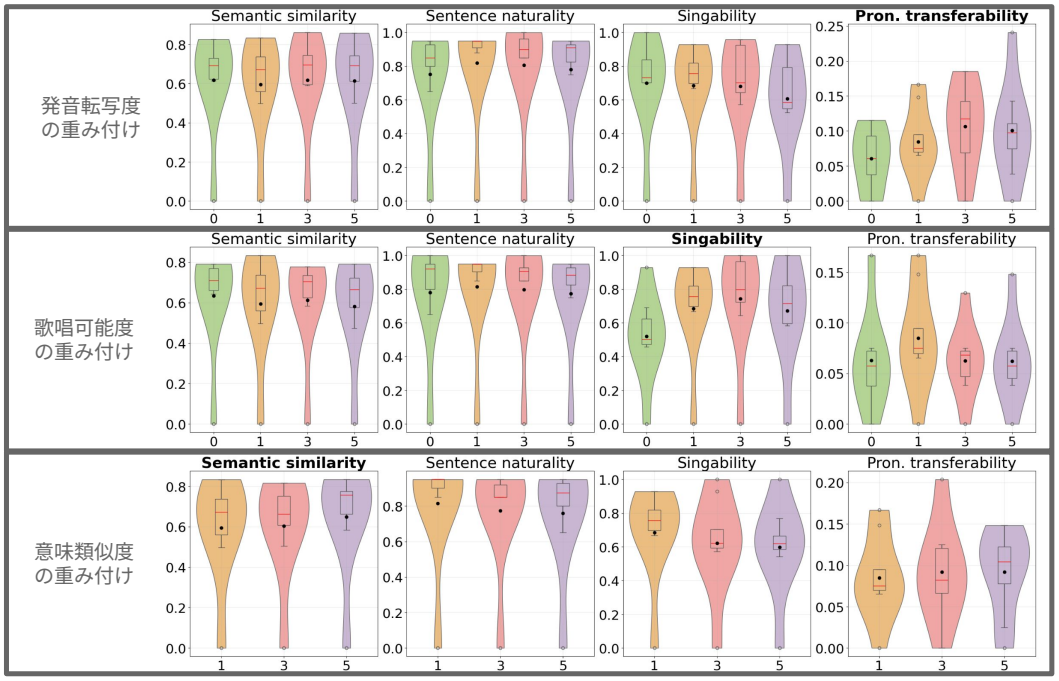
6

図 A·1 重み変更の際のスコアの違い．上から順に発音転写度，歌唱可能性，意味類似度の重みを変更したものと，アーティスト本人が作詞した英語歌詞 (ref) との比較．(夢灯篭)



7

図 A·2 重み変更の際のスコアの違い．上から順に発音転写度，歌唱可能性，意味類似度の重みを変更したものと，アーティスト本人が作詞した英語歌詞 (ref) との比較．(Let it go)



8

図 A·3 重み変更の際のスコアの違い．上から順に発音転写度，歌唱可能性，意味類似度の重みを変更したものの比較．（ラブストーリーは突然に）本曲はアーティスト本人による英語版の歌詞がなかったため，ref 無しの図になっている．