

韻と内容を制御した日本語ラップ歌詞の自動生成

織田 宥楽^{1,a)} 小川 隼斗^{1,b)} 河原 大輔^{1,c)}

概要: ラップはリズムに合わせて韻を踏むことを特徴とした歌唱法である。ラップ歌詞の作成は、韻を含めた形式的特徴と内容の一貫性の両立が求められるため難易度が高い。ラップや押韻に不慣れな人々を支援し、より良い作品や表現の創出を促す手段として、ラップ歌詞の自動生成が考えられる。本研究は、大規模言語モデル (LLM) を用いてテーマから日本語ラップ歌詞を生成することを目的とする。まず、ラップ歌詞コーパスを Web クロールによって構築する。次に、書籍コーパスを用いてラップ歌詞の形式と韻を学習し、続いてラップ歌詞コーパスを用いてラップらしい歌詞を学習する二段階の学習を行う。押韻の精度、ヒップホップ分類器によるスコア、LLM-as-a-Judge によるテーマ関連度を指標として評価し、ラップ歌詞の形式を保ちながらテーマに沿ったヒップホップらしい歌詞を生成できることを確認した。

Automatic Generation of Japanese Rap Lyrics with Control of Rhyme and Content

1. はじめに

ラップは、音楽に合わせてリズムカルに歌う歌唱法であり、ヒップホップ文化を構成する主要な要素の一つである。ヒップホップ文化は 1970 年代にアメリカのニューヨークにおいて形成され、貧困や人種差別といった社会的状況を背景に発展した。その後、1980 年代に日本へ持ち込まれ、日本でもラップやヒップホップが普及していった。その歌詞には、日常会話に近い口語表現やスラングが多用され、話し言葉ならではの勢いや臨場感が重視される。内容面では、貧困や差別といった社会問題への批判や風刺、自身の生き方や能力を誇示するセルフプロモーションなどが頻繁に扱われる。ラップにおいて重要な技法が「押韻」である。押韻とは、文末や文中の音において、とくに母音の響きを一致させることで、心地よさを生み出す表現手法である。一方、日本語の伝統的な詩歌である俳句や和歌は、音の数を一定に揃えることを重視しており、音の響きを合わせる技法は多くは見られない。このことから、押韻によってリズムを形成する技法は、俳句や和歌に見られる音数を合わせる技法と比べると、日本ではあまり浸透していないといえる。

ラップや押韻に不慣れな人々を支援し、より良い作品や表現の創出を促す手段として、ラップ歌詞の自動生成が考えられる。現時点では、日本語ラップ歌詞生成に関する研究は十分とはいえ、ChatGPT などの大規模言語モデル (LLM) においても、正確に韻を踏んだ日本語ラップ歌詞の生成は容易ではない。

本研究では、LLM に任意のテーマから日本語ラップ歌詞を自動生成させることを目的とし、Web からの収集によってラップ歌詞コーパスを構築するとともに、当該コーパスを用いてラップ歌詞生成モデルを学習する。学習の手法としては、モデルに段階的に知識を定着させるカリキュラム学習 [1] を採用する。まず、大規模な書籍コーパスを用いてリズムのまとまりである小節の概念や発音を学習する。次に、ラップ歌詞コーパスを用いて、ラップ特有の口語表現や押韻、社会風刺やセルフプロモーション的な歌詞の内容の学習を行う。提案手法の有効性を検証するため、生成された歌詞に対し、押韻の精度、表現の多様性、ヒップホップらしさ、およびテーマ関連度といった指標を用いて多角的な評価を行う。その結果、本手法はラップ歌詞としての形式を保ちながら、指定されたテーマに即したラップらしい歌詞を生成できることを確認した。

2. 関連研究

ラップ歌詞の生成に関する先行研究として、押韻やラッ

¹ 早稲田大学

^{a)} urakuodacchi@akane.waseda.jp

^{b)} cookie3120@ruri.waseda.jp

^{c)} dkw@waseda.jp

プバトルに焦点を当てた研究を紹介する。また、川柳や詩歌といった他の言語芸術形式の自動生成に関する動向についても述べる。

Nikolov らは、ニュース記事などの一般テキストを入力とし、それをラップ風の表現へ変換する生成モデルを提案した [2]。同研究では、BERT [3] を用いて単語を置換することで、生成結果における押韻の出現率が約 10% 向上することを示している。また Potash らは、LSTM を用いた歌詞生成モデルを構築し、特定のラッパーの文体的特徴を再現に焦点を当てながら新規の歌詞を生成する手法を提案している [4]。この手法は、従来手法と比較して高い評価を得たことが報告されている。これらの研究では、ラップらしい歌詞の再現や内容の客観的な評価手法の難しさを課題としている。

三林らは、日本語のラップバトルを取り上げ、BERT2BERT [5] を用いたラップ文の生成手法を提案している [6]。この手法では、まずラップの文を作り、その後 BERT を使って文の順番を整えるという流れをとっている。特に文末で韻を踏むことを重視し、通常のように文を先頭から作るのではなく、文末から逆向きに文を生成する方法を採用している点が特徴である。独自に集めたラップバトルのコーパスを使って学習を行い、生成した文を人が評価する実験を通して、この方法の有効性が確認されている。本研究は、対話的なラップバトルにおける返答文ではなく、テーマからラップ歌詞を生成することを目的とする。

我々は、PoeLM における structure-aware training [7] を参考に、ラップの形式を学習した日本語歌詞の生成モデルを構築している [8]。この研究では、既存の日本語コーパスに対し発音およびテーマ情報を付与し、教師なし学習によりラップの形式 (小節と韻) をモデルに学習させた。その結果、多様で口語的なコーパスを用いた場合にラップの形式をより良く学習できることを示した。しかし、生成結果された歌詞の内容について、テーマとの関連性が十分でないことやヒップホップらしい歌詞ではないことが課題として残った。

本研究では、日本語のラップ歌詞生成に焦点を当て、Web から構築したラップ歌詞コーパスを用いた学習により、既存手法の課題であった生成歌詞のテーマ関連度とラップらしい歌詞内容の獲得を目指す。

3. コーパスの構築

ラップの形式とその歌詞の内容について、モデルに段階的に知識を定着させるカリキュラム学習を行うため、2 種類のコーパスを構築する。第一段階であるラップの形式の学習には書籍コーパスを用い、第二段階であるラップの歌詞の内容の学習にはラップ歌詞コーパスを用いる。

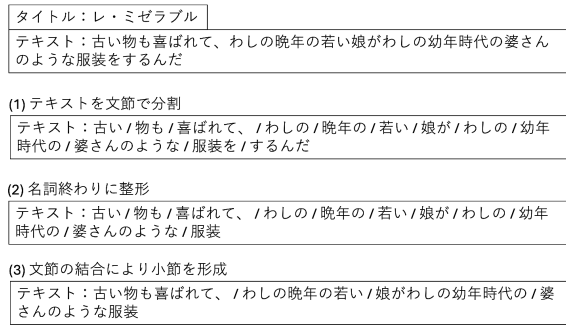


図 1 書籍コーパス構築の流れ

3.1 書籍コーパス

本研究では、先行研究 [8] において構築した書籍ベースの歌詞コーパス生成手法を踏襲する。青空文庫*1の各文をラップの 1 小節に相当する単位として分割し、小節末尾音を抽出するための発音付与処理を行う。コーパス構築の流れを図 1 に示す。事例数は約 113 万文であり、学習に 90%、テストに 10%を用いる。

3.2 ラップ歌詞コーパス

本研究では、日本語ラップ歌詞を対象としたコーパスを構築するため、IBG メディア株式会社が運営する総合音楽情報サイト UtaTen*2 から日本語ラップ歌詞を収集する。具体的には、同サイト上で「#HIPHOP」タグが付与されている楽曲を対象にスクレイピングを行い、5,901 曲を収集した。歌詞未掲載曲を除外した結果、合計 5,434 曲、323,421 行*3のラップ歌詞コーパスを構築した。

本研究では、指定したテーマに沿った歌詞を生成するモデルの学習を想定している。そのため、コーパス構築にあたっては、歌詞と対応するテーマ情報を付与したデータ構造を採用する必要がある。楽曲のタイトルと歌詞本文をそのまま対応づける方法も考えられるが、ラップ楽曲のタイトルは必ずしも歌詞の具体的なテーマや内容を直接的に表しているとは限らない。例えば、比喩的表現や楽曲全体の雰囲気象徴する抽象的なタイトルでは、歌詞の各節における内容との関連性が希薄なケースが存在する。

この問題に対処するために、歌詞本文の内容に基づいたテーマを付与したラップ歌詞コーパスを構築した。具体的には、5,068 曲、20,500 件 (1 行を 1 小節とし、4 行で 1 件とする) に対して LLM を用いてテーマを付与した。2 万件は学習、500 件はテストに用いる。このコーパスをテーマ付きラップ歌詞コーパスと呼ぶ。また、比較実験用として、書籍コーパスに対しても LLM によるテーマ付与を導入する。テーマ付与はコーパス中の 20,500 件の文に対して実施し、2 万件は学習、500 件はテストに用いる。このコーパ

*1 <https://huggingface.co/datasets/globis-university/aozorabunko-clean>

*2 <https://utaten.com/>

*3 歌詞において改行によって区切られた 1 単位を 1 行とする。

スをテーマ付き書籍コーパスと呼ぶ。テーマを付与の為の LLM は GPT-5-nano [9] を使用した。具体的なプロンプトの内容については、付録 A.1 に記載する。

4. モデルの学習

通常の言語モデルは、文字の並びだけを基に学習しており、言葉の読み方や音そのものは考慮していない。特に漢字が用いられる日本語では、アルファベットのような表音文字体系と異なり、表記情報のみから発音や押韻的な近さを直接学習することが難しく、文字列上の類似性を捉えるだけでは押韻を十分に表現できない。

ラップ歌詞の生成においては、押韻に加え、歌詞がテーマに沿っていることが求められる。このような背景から、我々は、先行研究で提案された structure-aware training (SAT) を活用し、発音情報とテーマ構造を同時に考慮したラップ形式の学習を提案している。

本研究では、SAT を基盤とした学習プロセスにカリキュラム学習の概念を取り入れ、モデルに段階的な知識定着を図る。具体的には、まず書籍コーパスを用いた学習により、ラップの形式や発音を習得させる。その後、ラップ歌詞コーパスを用いた二段階目の学習を行うことで、先行研究で実現したラップの形式に沿った歌詞生成を維持しつつ、ヒップホップらしい内容の生成能力を獲得させる。

我々のラップ生成実験 [8] では、複数の発音表記方式を比較した結果、ローマ字による発音付与が最も高い押韻再現精度を示したことを報告している。本研究では、発音表記としてローマ字を採用し、モデル学習を行う。また、学習させる発音の数を 3~5 音でランダムに指定する。作成したコーパスを図 2 のように整形し、事前学習済みの大規模言語モデルを用いた継続学習に利用する。具体的には、モデルのトークナイザに <PREF>, </PREF>, <BRK>, TOPIC, RHYME を特殊トークンとして新たに追加し、<PREF>と</PREF>で囲まれた区間に歌詞のテーマと韻(発音)の情報を記述する。<BRK>を区切りとして使用し、<TOPIC: >にテーマを、<RHYME: >には各小節に対応する発音をそれぞれ記述する。学習手法には Causal Language Model (CLM) による継続学習を採用し、推論の際には入力として与えられた<PREF>から</PREF>までの情報を手掛かりに、それに続くラップの歌詞を生成させる。SAT の全体像を図 2 に示す。

本研究におけるモデルの学習方法は、先行研究 [8] と同一であり、本研究では新たなコーパスを用いて、モデルにラップの形式とラップらしい歌詞を段階的に学習する点と評価方法が異なる。

5. 実験

5.1 実験設定

3 節で作成した 2 種類のコーパスを用いて 4 節の学習を

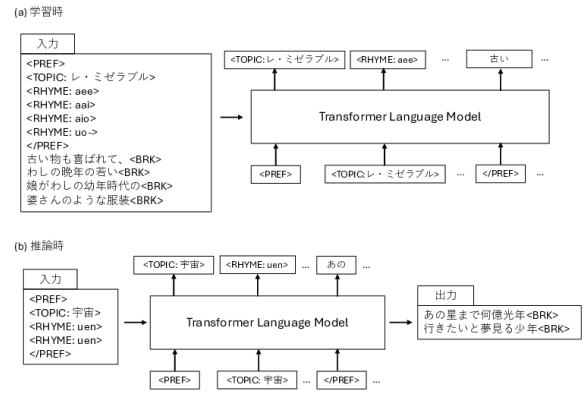


図 2 本研究における Structure-aware training

表 1 学習に使用したハイパーパラメータ

ハイパーパラメータ	値
learning rate	6e-5
batch size	16
epoch	5
weight decay	0.2

表 2 分類器の識別精度

クラス	適合率	再現率	F1 スコア
ラップ	0.84	0.90	0.87
非ラップ	0.89	0.83	0.86

行う。学習のベースモデルには、日本語大規模言語モデル llm-jp/llm-jp-3-1.8b*4 を使用する。学習時に使用したハイパーパラメータを表 1 に示す。本研究では、以下の 6 種類のモデルを比較対象として構築した。書籍コーパス約 100 万文を用いて SAT を適用したモデルを book、ラップ歌詞コーパス約 5 万件を用いて SAT を適用したモデルを rap とする。さらに、テーマ付きラップ歌詞コーパス 2 万件を用い book に対して継続学習を行ったモデルを book+rap_theme とする。またテーマ付き書籍コーパス 2 万件を用い book に対して継続学習を行ったモデルを book+book_theme とする。また、比較用にテーマ付きではないラップ歌詞コーパス 2 万件と書籍コーパス 2 万件をそれぞれ book に対して継続学習を行ったモデルを book+rap、book+book とする。

5.2 評価方法

モデルが、与えられたテーマに基づいて韻を踏んだ歌詞を生成できているかを評価するため、学習時には使用していないテーマを用いて評価を行う。具体的には、ラップ歌詞コーパスおよび書籍コーパスのそれぞれについて、LLM により付与されたテーマのうち、学習に用いなかった 500 件ずつを評価用テーマとして用いる。韻は、「あ」「い」「う」「え」「お」「ん」「っ」「ー」(長音記号) をランダムに 3 つ繋げたものを利用する。1,000 件のテスト用コーパス(テーマと韻のペア)を作成し、これをもとにモデルにラップの

*4 <https://huggingface.co/llm-jp/llm-jp-3-1.8b>

表 3 モデルの評価結果

モデル	押韻の精度 (↑)	反復率 (↓)	多様性 (↑)	ラップスコア (↑)	テーマ関連度 (↑)
book	0.63	0.00	0.13	0.27	0.58
rap	0.10	0.49	0.00	0.60	0.77
book+book	0.52	0.01	0.10	0.26	0.48
book+rap	0.51	0.07	0.05	0.66	0.59
book+book_theme	0.51	0.01	0.09	0.26	0.44
book+rap_theme	0.54	0.09	0.05	0.76	0.67

歌詞を生成させた。生成した歌詞の評価は、押韻の精度、表現の多様性、ロジスティック回帰を用いた二値分類器によるラップスコア、LLM-as-a-Judge によるテーマ関連度の4つの指標を用いて行う。

5.2.1 押韻の精度

生成した歌詞のそれぞれの小節末の母音が、指定した母音と一致しているかを確認し、押韻の精度を測る。

5.2.2 表現の多様性

押韻が成立していても、同一の単語やフレーズを繰り返している場合、表現としての価値は低い。本研究では、生成された4小節の歌詞において、異なる語彙を用いながら韻を踏めているか、また同一フレーズが繰り返し使用されていないかに着目し、表現の多様性を2種類の指標で評価する。具体的には、生成された4小節において、全ての小節で全く同じフレーズが用いられていた割合を反復率と定義する。また、各小節末の単語が4つとも全て異なる単語であった割合を表現の多様性と定義し、評価する。

5.2.3 ラップスコア

生成された歌詞が日本語ラップとしてどの程度妥当であるかを定量的に評価するため、分類器を用いて「日本語ラップらしさ」を数値化した指標（以下、ラップスコア）を導入する。具体的には、日本語ラップ歌詞を正例、非ラップ文を負例とする二値分類問題として分類器を学習し、生成歌詞が正例に分類される確率をラップスコアとして用いる。

正例であるラップデータには、ラップ歌詞コーパスから抽出した9,000行を用いた。一方、負例である非ラップデータとしては、歌詞検索サービスの歌ネット^{*5}から収集したJ-POP歌詞4,500行および書籍コーパスから抽出した4,500文を使用し、計9,000件を用いた。10%はテストデータとする。特徴量にはn-gramに基づくTF-IDF表現を採用し、分類器としてロジスティック回帰モデルを用いた。

本手法の有効性を検証するため、テストデータを用いた分類精度の評価を行った。その結果を表2に示す。評価の結果、正例であるラップデータにおいてF1スコア0.87という高い識別性能が得られた。特に再現率は0.90に達しており、日本語ラップの特徴を精度良く捉えられていることが確認できる。

この分類器によって得られる出力確率を、生成歌詞が日本語ラップ的表現を有している度合いを示す指標として用いる。

5.2.4 テーマ関連度

指定したテーマに対して、生成された歌詞内容がどの程度整合しているかを評価するため、LLM-as-a-Judge [10]による評価を行う。評価にはGPT-5-nano [9]を用い、生成された歌詞がテーマと無関係である場合を0、テーマに沿っている場合を1とし、評価を行う。評価のための具体的なプロンプトの内容については、付録A.1に記載する。

5.3 結果

5.3.1 定量評価

各モデルが生成した歌詞の押韻の精度、反復率、多様性、ラップスコア、テーマ関連度を表3に示す。

まず、第一段階の学習である、ラップの形式の学習を行ったbookとrapを比較する。押韻の精度においては、bookモデルが最も高い値を示しており、反復率も低い。一方で、rapは押韻の精度が0.10と低く、また、反復率は0.49と非常に高い。これは、rapが生成した歌詞の多くが、4小節で同じフレーズを繰り返してしまっていることを表している。bookの約100万件の訓練データに対して、rapの訓練データ数は約4万件と少なく、多様な発音を学習できなかったためと考えられる。また、ラップ歌詞コーパス自体に、語彙の偏りがあることも考えられる。rap, book+rap, book+rap_themeはラップ歌詞コーパスを用いて学習を行ったモデルであるが、書籍コーパスのみで学習したモデルと比較して多様性が低い。ラップ歌詞コーパスに含まれる言葉の発音や種類に偏りがあり、多様性が下がったと考えられる。

次に、第二段階まで学習をしたbook+モデルについて述べる。ラップスコアについては、book+rap_themeが他のモデルと比べて高い値を示しており、ラップらしい表現の再現という観点で最も優れていることが分かる。一方で書籍コーパスのみで学習を行った、book+bookおよびbook+book_themeは低い値にとどまっており、表現がラップ的にはならなかった。テーマ関連度に関して、book+rapとbook+rap_themeを比較すると、テーマ付きのコーパスで学習したbook+rap_themeがbook+rapのテーマ関連度を上回っており、よりテーマに沿った歌詞を生成できてい

*5 <https://www.uta-net.com/>

表 4 生成歌詞の例 (テーマ: 孤独感, 韻: aee)

モデル	生成歌詞
rap	「大丈夫」って言われて 「大丈夫」って言われて 「大丈夫」って言われて 「大丈夫」って言われて
book+book_theme	僕は窓に腰を掛けて 薄笑いを湛えて この男を眺めて、 やがて目
book+rap_theme	真っ赤な夕日に一人たたずむ俺は赤らめて 過ぎ去った時間を追いかけて 消えていく君を追いかけて また世界が終わって

ることがわかる。

表 3 において、rap (ラップ歌詞コーパスを用いてラップの形式と内容を同時に学習したモデル) の結果に注目すると、テーマ関連度およびラップスコアは高いものの、押韻の精度が極めて低く、反復率は非常に高い値を示している。表 4 に示す生成例のように、与えられたテーマに対して、ラップらしいフレーズを 4 小節繰り返すだけになっており、ラップ歌詞生成モデルとして機能していないことがわかる。一方で book+rap_theme は、book で獲得した高い押韻の精度は一定水準で維持しつつ、ラップスコアにおいても 0.76 と高い値となった。この結果は、ラップの形式 (小節と押韻) をまず書籍コーパスで学習させ、その後にラップ歌詞コーパスで内容や文体を学習させるというカリキュラム学習の有効性を示している。

5.3.2 定性評価

表 4, 5 に、同一のテーマおよび韻を指定した際の生成例を示す。book+book_theme の生成結果が文語的で情景描写のようであるのに対し、book+rap_theme では 1 人称に「俺」が使われていたり、自分の心情を表す歌詞の生成が見られ、歌詞らしい生成ができていくことがわかる。ただ、表 3 に示すように多様性は低く、押韻の精度も約 0.5 ほどであることから、一部で表 4, 5 のような優れた歌詞の生成が確認できる一方で、その安定性に課題があるといえる。

6. おわりに

本研究では、テーマから日本語ラップ歌詞を生成することを目的として、Web から収集したラップ歌詞コーパスを構築し、structure-aware training を用いた学習手法の有効性を検証した。書籍コーパスで事前学習したモデルにテーマ付きラップ歌詞コーパスで継続学習をした book+rap_theme モデルは、押韻の精度を一定水準に保ちながら、先行研究の課題であったラップスコア及び生成した歌詞とテーマの関連度のスコアが高く、テーマに沿ったヒップホップらしい歌詞が生成できることがわかった。一

表 5 生成歌詞の例 (テーマ: 倦怠感, 韻: iou)

モデル	生成歌詞
rap	何もないのにただ祈る 何もないはずが何一つ 無くさずにただ祈る 何もないはずが何一つ
book+book_theme	余は此時ほど面白く 自分の眼に映る限りの 事実をも棄て去る ことに六
book+rap_theme	Bye もう疲れた無気力 夏休みの終り夜 月並みで良い夜 僕の心を奪い取る

方で、このような良好な生成結果が得られる割合は依然として高いとは言えず、生成の安定性や品質のばらつきが課題として残っており、今後さらなる改善の余地がある。

謝辞

本研究は JSPS 科研費 JP23K17641 および JP24H00727 の助成を受けて実施した。実験には、産総研及び AIST Solutions が提供する ABCI 3.0 と、東京科学大学のスーパーコンピュータ TSUBAME4.0 を利用した。ABCI 3.0 は「ABCI 3.0 開発加速利用」の支援を受けて利用した。

参考文献

- [1] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09*, p. 41–48, New York, NY, USA, 2009. Association for Computing Machinery.
- [2] Nikola I. Nikolov, Eric Malmi, Curtis Northcutt, and Loreto Parisi. Rapformer: Conditional rap lyrics generation with denoising autoencoders. In Brian Davis, Yvette Graham, John Kelleher, and Yaji Sripada, editors, *Proceedings of the 13th International Conference on Natural Language Generation*, pp. 360–373, Dublin, Ireland, December 2020. Association for Computational Linguistics.
- [3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.
- [4] Peter Potash, Alexey Romanov, and Anna Rumshisky. GhostWriter: Using an LSTM for automatic rap lyric generation. In Lluís Màrquez, Chris Callison-Burch, and Jian Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1919–1924, Lisbon, Portugal, September 2015. Association for Computational Linguistics.
- [5] Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. Leveraging pre-trained checkpoints for sequence generation tasks. *Transactions of the Association for Computational Linguistics*, Vol. 8, pp. 264–280, 2020.

- [6] 三林亮太, 山本岳洋, 佃洗撰, 渡邊研斗, 中野倫靖, 後藤真孝, 大島裕明. ラップバトルにおける逆向き生成によるライムを含む返答バース生成. 情報処理学会論文誌データベース (TOD), Vol. 17, No. 2, pp. 28–39, apr 2024.
- [7] Aitor Ormazabal, Mikel Artetxe, Manex Agirrezabal, Aitor Soroa, and Eneko Agirre. PoELM: A meter- and rhyme-controllable language model for unsupervised poetry generation. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 3655–3670, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.
- [8] 織田宥楽, 小川隼斗, 河原大輔. 教師なし学習によるラップの形式の学習. 第 39 回人工知能学会全国大会論文集, No. 2Win567. 人工知能学会, 2025.
- [9] OpenAI. Gpt-5 nano model — openai api. <https://platform.openai.com/docs/models/gpt-5-nano>, 2025. [Accessed: 2025-12-01].
- [10] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23*, Red Hook, NY, USA, 2023. Curran Associates Inc.

付 録

A.1 プロンプト

ラップ歌詞コーパスをモデル学習用に整形する際、歌詞からテーマを生成する際のプロンプトを表 A.1 に示す。生成された 4 つの項目のうち、「1. 歌詞の内容を要約する言葉」をテーマとして採用した。

表 A.1 歌詞からテーマを生成する際のプロンプト

以下に歌詞があります。
この歌詞を読み取り、次の 3 点をそれぞれ 1 語または 1 フレーズで考えてください。
できるだけ歌詞に登場する単語は使わないでください。

1. 歌詞の内容を要約する言葉
2. 歌詞のタイトル
3. 歌詞全体の雰囲気やトーンを表す言葉
4. 聴く人の頭に浮かびそうな背景イメージ

[要約する言葉, タイトル, 雰囲気, 背景イメージ]
の順にカンマ区切りで出力してください。

単語: {text}

LLM を用いてテーマと歌詞の関連度 (テーマ関連度) を評価する際のプロンプトを表 A.2 に示す。

表 A.2 テーマ関連度を評価するプロンプト

以下の「タイトル」と「歌詞」の関連性を評価し、0 または 1 で回答してください。

評価基準:

- 1: 関連している (歌詞のテーマや内容が、タイトルの示すテーマ・対象・状況・概念と意味的に対応している)
- 0: 関連していない (歌詞のテーマや内容が、タイトルの示すものと無関係、または歌詞が意味を成していない)

返答は数値のみ出力してください。

【タイトル】
{section}
【歌詞】
{lyric}