# 深層学習時代における音韻論の素性

Sora Nagano

The University of Tokyo

2025-07-10

# 目次

# 目次

# 1. 先行研究レビュー

# 1.1 理論的基盤: 記号 vs 分散記号

💡 補足

**構造主義・生成音韻論**: 言語は離散的記号システム

- 弁別素性体系（Chomsky & Halle 1968）
- 最適性理論（Prince & Smolensky 1993）
- 制約ベース文法による音韻現象の説明

例: 日本語「さくら」[sakɯ̟ᵝa] → /sakura/

- 記号: /s/, /a/, /k/, /u/, /r/, /a/ という離散単位の列
- 素性: [+consonantal], [-voice], [+coronal]……など属性の組み合わせ

# 1.1 理論的基盤: 記号 vs 分散

## 分散

🔧 技術的詳細

コネクショニズム・分散表現: 知識は連続値ベクトルに分散 (Staples & Graves, 2020)

- word2vec, Skip-gram（Mikolov et al. 2013）
- **音韻類推の創発**:king - man + woman ≈ queen (Silfverberg et al., 2018)
- **統計的共起パターンからの知識獲得** (Kolachina & Magyar, 2019)

# 1.2 教師なし音韻学習

## 生成モデルによるアプローチ

### 📊 成果

**GAN による音韻獲得** (Begŭs, 2020):

- 生音声から音韻論的制約を教師なし学習
- VOT（声開始時間）分布の自発的学習
- しかし学習表象は必ずしも言語学理論と対応しない (Chen & Elsner, 2023)

# 1.2 教師なし音韻学習

## クラスタリング・離散化手法

### 🔧 技術的詳細

**Vector Quantization (VQ)**: 連続→離散変換の核心技術 (Higy et al., 2021)

- k-means クラスタリング→コードブック生成
- Gumbel-Softmax: 微分可能な離散化
- コードブックサイズ選択問題: 128, 256, 512...最適値は？

### 💡 補足

VQ の仕組み:

連続ベクトル [0.3, 0.8, −0.2] → 最近傍コード「ID:47」

全音声を有限個の「音韻的単位」で表現可能に。

# 1.3 プロービング研究の知見

## SSL モデルの音韻知識

### 成果

**階層的情報符号化**が判明 (Venkateswaran et al., 2025):

- 下位層: 音響音声的特徴（F0, formant）
- 中位層: 音素・異音レベル情報
- 上位層: 形態・統語レベル情報

# 1.3 プロービング研究の知見

## 具体的発見事例

### 🔧 技術的詳細

- **有気性検出**: 英語/p/-/$p^h$/の区別 (Medin et al., 2024)
- **声調符号化**: 中国語の語彙声調表現 (Pasad et al., 2024)
- **異音変異**: 環境条件による音素変化の学習 (Pouw et al., 2024)
- **韻律情報**: アクセント・境界の自動獲得 (Gosztolya et al., 2024)

# 1.4 ハイブリッドアーキテクチャの可能性

## ニューロシンボリック統合

🔧 技術的詳細

記号×ニューラル融合の試み (Panchendrarajan & Zubiaga, 2024):

- 論理規則エンジン + 深層学習モジュール
- 解釈可能性とパフォーマンスの両立

💡 補足

ハイブリッドの利点:

ニューラル部→データから柔軟学習

記号部→言語学理論との整合性

例: 音響特徴（NN）→制約重み（記号）→音韻出力

# 1.5 研究ギャップの特定

## 既存研究の限界

個別現象の存在証明に留まり、体系的比較が欠如:

- プロービング:「モデル X は特性 Y を持つか?」
- 単発評価:特定タスク・特定モデルの分析
- 評価軸の限定: 精度のみ、解釈可能性軽視

# 1.5 研究ギャップの特定

## 本研究の新規性

### 成果

多軸・多タスク・多単位の包括的比較:

- 表象単位: 連続値・VQ・音素・素性の系統的比較
- 評価軸: 精度・解釈性・認知妥当性・計算効率
- タスク群: 音素分類・配列論・形態音韻論・言語獲得シミュレーション

# 2. 背景と目的

# 2.1 中核的問い

## 言語をモデル化するための最適な表象単位は何か?

> 💡 補足
>
> 従来の音韻論概念（音素・弁別素性・音節 (Cho et al., 2025)）vs コンピュータの数値ベクトル表現、 どちらが優れているか? 組み合わせは可能か? を探る研究。

- **記号的素性**: 音素、音節、弁別素性など（人間解釈可能、理論的基盤）
- **連続値表現**: ニューラルモデルによる分散表現（強力だが不透明）(Staples & Graves, 2020)

# 2.2 NLP とは何か

💡 補足

NLP = Natural Language Processing（自然言語処理）: コンピュータによる人間言語（音声・テキスト）の理解・生成技術。
例: Google 翻訳、Siri、ChatGPT

## NLP の歴史的発展

- **1950 年代**: 規則ベース（言語学者が手作業で規則を記述）
- **1990 年代**: 統計的手法（大量データから確率的パターンを学習）
- **2010 年代**: 機械学習・深層学習（ニューラルネットワークによる自動学習）

# 2.2 NLP とは何か

🔧 技術的詳細

深層学習 = 人間の脳神経細胞（ニューロン）を模倣した「ニューラルネットワーク」の多層構造による機械学習。

# 2.3 音韻論と NLP の接点

## 従来の音韻論的分析

- 専門家による手作業での音韻規則記述
- 理論的知識に基づく素性体系
- 小規模データでの精密分析

## NLP 的アプローチ

- 大量データからの自動パターン発見
- 統計的・確率的なモデリング
- 高次元ベクトル空間での表現学習

> 💡 **補足**
>
> 例: 従来「/p/と/b/は[labial]で[± voice]が違う」と記述 → NLP では数百次元ベクトル（例: [0.2, −0.8, 1.3, …]）で表現し、コンピュータが自動的に類似性を学習。

# 3. SSL（自己教師あり学習）

# 3.1 SSL: 新しい音声理解パラダイム

💡 補足

**自己教師あり学習（Self-Supervised Learning, SSL）**＝「正解ラベル」なしで大量音声データから学習する画期的手法。

**従来:**「この音は/a/、この音は/k/」という**専門家ラベル**が**必要**
⇒ SSL: 音声の一部を隠して「次にくる音は?」を**予測学習** (Mohamed et al., 2022)。

# 3.1 SSL: 新しい音声理解パラダイム

## 従来手法の限界

- 大量の専門家による音韻転写が必要
- 言語・方言ごとに専用の音韻体系が必要
- 時間とコストが膨大

## SSL の革新性

- ラベルなし音声データのみで学習可能
- 言語普遍的な音韻構造を自動発見 (Choi et al., 2024)
- 大規模データ活用による高精度化

# 3.2 wav2vec 2.0: 代表的 SSL モデル

🔧 技術的詳細

**wav2vec 2.0**（Meta/Facebook 開発）= 現在最も成功している音声 SSL モデル (Baevski et al., 2022)。

- 訓練データ: LibriSpeech（960 時間の英語読み上げ音声）
- アーキテクチャ: Transformer（注意機構付きニューラルネット）
- 学習方式: 対照学習（正例と負例を区別）

# 3.2 wav2vec 2.0: 代表的 SSL モデル

## 学習プロセス

1. **特徴抽出**: 音声波形から初期特徴を抽出
2. **マスキング**: 特徴の一部をランダムに隠す
3. **予測**: 隠された部分を予測
4. **対照学習**: 正しい予測と間違った予測を区別

> 💡 **補足**
>
> 人間が「さく __ 」という音声から「ら」を予測するのと類似。大量音声での反復学習→言語の音韻構造理解。

# 3.3 プロービング: モデル内部の知識探査

💡 補足

プロービング = 訓練済みモデルが「本当に音韻論的知識を学習しているか」を調査する手法。 モデルの内部表現から音韻的特徴を予測できるかをテスト (Astrach & Pinter, 2025; Venkateswaran et al., 2025)。

# 3.3 プロービング: モデル内部の知識探査

## プロービング実験の設計

- モデルの内部表現（高次元ベクトル）を入力
- 音韻的特徴（有声性、調音部位など）を予測
- 高精度 = モデルが音韻的知識を保持

# 4. VQ:　連続 ⇔ 離散の架け橋

# 4.1 VQ の基本概念

💡 補足

ベクトル量子化（Vector Quantization, VQ）:
= 連続的な数値表現 → 離散的な「コード」に変換する技術。

例: 連続値 [0.73, −0.45, 1.23] → 離散コード「ID:15」
⇒ NLP の連続表現を音韻論の離散カテゴリーに近づける (Higy et al., 2021)。

## 音韻論的意義

- 音韻論: 音素は離散的カテゴリー（/p/, /t/, /k/ など）
- NLP: 連続値ベクトル表現
- VQ: 両者の橋渡し役

# 4.2 K-means による VQ 実装

## 実装手順

1. wav2vec 2.0 から**連続特徴抽出**
2. K-means で 128 クラスタに**分類**
3. 各フレームに ID を割り当て
4. **離散音韻**コード系列を生成

# 4.2 K-means による VQ 実装

## 🔧 技術的詳細

**具体的な実装詳細:**

- アルゴリズム: MiniBatchKMeans
- パラメータ: n_clusters=128, random_state=42, batch_size=2048, n_init=3
- 入力形状: (総フレーム数, 768) - 全音声を結合した巨大行列
- 出力: 128 個のクラスタ中心ベクトル + 予測関数
- 保存形式: joblib.dump による pickle 形式

**学習プロセス:**

1. 全フレーム結合: all_frames.shape = (15,234, 768)
2. KMeans 学習: 128 クラスタに分類
3. クラスタ中心生成: cluster_centers_.shape = (128, 768)
4. 予測機能: 新フレーム → 最近傍クラスタ ID (0-127)

# 4.2 K-means による VQ 実装

💡 補足

例: 「cat」という音声が [ID:52, ID:23, ID:78] という離散コード列で表現。
従来の音韻転写 [k æ t] に対応する可能性。

# 5. リサーチクエスチョン

# 5.1 RQ1: 表象単位比較

## RQ1

**問い:** 連続値表現、VQ 離散値表現、記号的表現のどれが 音韻論的現象を最もよくモデリングできるか?

💡 **補足**

「コンピュータが音韻を理解するのに、どの表現方法が最適か」を比較する実験。従来の音韻論理論との整合性も重要な評価軸。

# 5.1 RQ1: 表象単位比較

## 実験設計

- **共通基盤**: wav2vec2-base-960h 特徴抽出器
- **比較対象**: 連続値 vs VQ 離散値 (vs 記号値)
- **タスク**: 音素分類(、音韻的特徴予測)

## 評価指標

- **F1 スコア**: 精度と再現率の調和平均
- **正解率**: 分類精度
- **計算効率**: 処理速度とメモリ使用量

# 5.2 RQ2: ハイブリッドモデル

**RQ2**

**問い**: ニューラル表現とその他の音響特徴を組み合わせたハイブリッドモデルは従来手法を上回るか？ (Panchendrarajan & Zubiaga, 2024)

# 6. 実験環境とデータ

# 6.1 マイクロスケール実験概要

💡 補足

本研究では「概念実証（Proof of Concept）」として小規模データで手法の有効性を確認。実用化には大規模実験が必要だが、まず技術的実現可能性を検証。

## 実験環境

- **Docker + Poetry**: 再現可能な実験環境構築
- **計算資源**: CPU 環境（MacBook Pro）- GPU 不要で実行可能
- **データサイズ**: 各データセット 100 サンプル（マイクロスケール）
- **実行パイプライン**: 3 段階の自動化された処理フロー

# 6.1 マイクロスケール実験概要

## データセット

| データセット | サンプル数 | 特徴 | 用途 |
|---|---|---|---|
| **LibriSpeech** | 100 | 高品質読み上げ音声 | 音韻的特徴分析 |
| **Common Voice** | 100 | 多様な話者、年齢情報 | ハイブリッドモデル検証 |

🔧 技術的詳細

- LibriSpeech: オーディオブック由来の高品質英語音声
- Common Voice: Mozilla 提供の多言語音声データセット

⇒ 両データセットとも 16kHz サンプリングレートに統一

# 6.2 実験手順

## RQ1: プロービング実験詳細

# 6.2 実験手順

## 🔧 技術的詳細

**アライメント手法:**

- G2P-EN による音素変換（text → phoneme list）
- ヒューリスティック時間分割：np.linspace 使用
- フレーム-音素対応付け：均等分割方式

**特徴量準備:**

- 連続値：wav2vec2 隠れ状態（768 次元）
- 離散値：VQ クラスタ ID（0-127 の整数）
- プーリング：時間軸平均でフレームレベル特徴生成

**プローブ設計:**

- 分類器：ロジスティック回帰（線形プローブ）
- 分割：train-test split (70%-30%)
- 評価：63 種類の音素カテゴリ分類

# 6.2 実験手順

## RQ2: ハイブリッドモデル詳細

# 6.2 実験手順

## 🔧 技術的詳細

ベースライン（ニューラル特徴のみ）:

- 入力: wav2vec2 隠れ状態の時間軸平均（768 次元）
- 正規化: StandardScaler 適用
- タスク: 話者年齢層予測（8 クラス分類）

ハイブリッド（ニューラル + 音響特徴）:

- ニューラル特徴: 上記と同じ（768 次元）
- 音響特徴: F0 統計量（平均・標準偏差）
- 抽出手法: librosa.pyin によるピッチ推定
- 統合: 水平結合で 772 次元の特徴ベクトル生成
- 正規化: 統合後に StandardScaler 適用

# 6.3 実験実行パイプライン

## ステップ 1: データダウンロード

### 🔧 技術的詳細

**LibriSpeech test.clean**（RQ1 用）:

- Hugging Face ストリーミング経由で効率的取得
- 最初の 100 サンプルを抽出
- 高品質な読み上げ音声（オーディオブック由来）
- 各サンプル: 音声波形 + テキスト転写

**Common Voice 13.0**（RQ2 用）:

- Mozilla 提供の多言語音声コーパス
- 年齢情報付きサンプルをフィルタリング
- **対象年齢層**: teens, twenties, thirties, forties, fifties, sixties, seventies, eighties
- 各サンプル: 音声波形 + 発話文 + 話者年齢層

# 6.3 実験実行パイプライン

## 🔧 技術的詳細

実際のデータ例:

LibriSpeech:


```
{
  "file": "6930-75918-0000.flac",
  "audio": {"array": [-6.10e-05, 9.15e-05, ...], "sampling_rate": 16000},
  "text": "CONCORD RETURNED TO ITS PLACE AMIDST THE TENTS",
  "speaker_id": 6930
}
```


Common Voice:


```
{
  "audio": {"array": [0.001, -0.002, ...], "sampling_rate": 48000},
  "sentence": "The quick brown fox jumps over the lazy dog",
  "age": "twenties"
}
```

# 6.3 実験実行パイプライン

## ステップ 2: 特徴抽出

# 6.3 実験実行パイプライン

## 🔧 技術的詳細

**連続値特徴抽出:**

1. wav2vec2-base-960h モデルをロード
2. **音声前処理:** 16kHz にリサンプリング、正規化
3. **隠れ状態抽出:** Shape (フレーム数, 768 次元)
4. **保存:** librispeech_micro_continuous.npy

**VQ モデル学習:**

1. **全音声フレームを結合:** Shape (総フレーム数, 768)
2. **MiniBatchKMeans 実行:** 128 クラスタ生成
3. **クラスタ中心保存:** vq_kmeans_128_micro.pkl
4. **離散音韻コード体系の確立**

# 6.3 実験実行パイプライン

## 🔧 技術的詳細

**実際の形状例:**

- 1 音声: (149, 768) → 149 フレーム×768 次元隠れ状態
- 100 音声結合: (15,234, 768) → 総 15,234 フレーム
- VQ クラスタ中心: (128, 768) → 128 個の代表ベクトル

**VQ 変換例:**

- 連続ベクトル: [0.73, −0.45, 1.23, …] (768 次元)
- → 離散コード ID: 25 (0-127 の整数)
- VQ コードシーケンス例: [25, 25, 25, 52, 52, 78, 78, …]

# 6.3 実験実行パイプライン

## ステップ 3: 実験実行(各 notebook で検証)

| Notebook | 目的 | 検証内容 |
|---|---|---|
| rq1_probing_pipeline | 表象単位比較 | 連続値 vs VQ 離散値での音素分類性能 |
| rq2_hybrid_model_poc | ハイブリッド検証 | ニューラル特徴 + 音響特徴の統合効果 |

# 6.3 実験実行パイプライン

## 実装上の技術的詳細

# 6.3 実験実行パイプライン

## 🔧 技術的詳細

### 実行ログ例（データダウンロード）:

```
データは次の場所にキャッシュされます: /workspace/data/.cache
[デバッグ] LibriSpeechストリームから 100 個のサンプルを取得しました。
[デバッグ] 最初のLibriSpeechサンプルの構造:
{'file': '6930-75918-0000.flac', 'text': 'CONCORD RETURNED...'}
LibriSpeechのサンプル 100 個を /workspace/data/raw/librispeech_micro に正常に保存しました。
```

### 実行ログ例（特徴抽出）:

```
[デバッグ] K-Means用のall_framesのshape: (15234, 768), Dtype: float32
VQモデル（KMeans）を 128 クラスタで学習中...
[デバッグ] クラスタ中心のshape: (128, 768)
VQモデルを outputs/models/vq_kmeans_128_micro.pkl に保存しました。
```

### 依存関係管理:

- Poetry による Python 環境管理
- Docker コンテナによる OS レベル再現性
- requirements 固定による バージョン統一
- Hugging Face datasets/transformers ライブラリ活用

# 7. 実験結果と現状

# 7.1 RQ1: プロービング実験結果詳細

## 成果

**音素分類タスクでの実証結果:**

**実験設定:**

- **対象音素:** 63 種類の英語音素カテゴリ
- **データセット:** LibriSpeech micro (100 サンプル、総 33,464 フレーム)
- **アライメント手法:** G2P-EN + ヒューリスティック時間分割
- **評価方法:** train-test split (70%-30%) でロジスティック回帰

**連続値特徴（wav2vec 2.0 ）の性能:**

- **入力次元:** 768 次元隠れ状態の時間軸平均プーリング
- **線形分離可能性:** ロジスティック回帰で音素分類を実行
- **音韻混同行列:** 類似音素間の予測パターンを可視化
- **結果解釈:** ランダム分類（ 1/63≈1.6% ）を大幅に上回る性能

# 7.1 RQ1: プロービング実験結果詳細

## 🔧 技術的詳細

アライメント詳細:

- G2P-EN: テキスト “A MAN SAID...” → 音素列 [‘AH0’, ‘ ‘, ‘M’, ‘AE1’, ‘N’, ...]
- 時間分割: np.linspace で音声フレームを音素数で均等分割
- 例: 150 フレーム · 10 音素 → 各音素 15 フレーム割り当て
- データ生成: フレームレベル特徴量と音素ラベルのペア作成

実際のアライメント例 ( 最初のサンプル ):

- テキスト: “CONCORD RETURNED TO ITS PLACE AMIDST THE TENTS”
- 音素列 ( 42 個): [‘K’, ‘AA1’, ‘N’, ‘K’, ‘AO2’, ‘R’, ‘D’, ‘ ‘, ‘R’, ‘IH0’, ‘T’, ‘ER1’, ‘N’, ‘D’, ...]
- フレーム数: 175 フレーム
- 境界配列: [0, 4, 8, 12, 16, 20, 25, 29, ...] → 各音素に約 4 フレーム割り当て
- 最終データセット: 33,464 フレーム ( 全 100 音声 ) × 768 次元特徴量

# 7.1 RQ1: プロービング実験結果詳細

## VQ 離散値特徴の評価結果

### 成果

**離散化効果の検証:**

- VQ クラスタ数: 128 個(英語音素数の約 2 倍設定)
- 変換方式:連続特徴 → 最近傍クラスタ ID(0-127 の整数)

**VQ 離散値特徴の性能:**

- 線形分離可能性:離散化後も音素分類が可能
- 次元削減効果: 768 次元→1 次元への劇的な圧縮
- 情報保持度: 離散化による一定の音韻情報保持を確認(連続値での結果と変わらず)

# 7.1 RQ1: プロービング実験結果詳細

## 💡 補足

**VQ 離散化の効果と限界:**

**検証された効果:**

- ランダム分類を大幅に上回る分類性能
- 768 次元→1 次元への効率的な圧縮
- 記号的表現との親和性(クラスタ ID = 離散音韻カテゴリ)
- 音韻的類似性の構造的保持

**実際の VQ 変換例:**

- **連続特徴**: [0.73, −0.45, 1.23, ...] (768 次元)
- VQ 変換: クラスタ ID=52 (最近傍クラスタ)
- 音声「cat」: [ID:52, ID:23, ID:78] → 離散コード列
- 解釈: ID=52→/k/、ID=23→/æ/、ID=78→/t/ の対応可能性

# 7.2 RQ2: ハイブリッドモデル実験結果詳細

## 📊 成果

**年齢層予測タスクでの検証結果:**

**実験設計:**

- **データセット:** Common Voice micro (100 サンプル、**年齢情報付き**)
- **タスク:** 8 クラス**年齢層分類** (teens, twenties, thirties, forties, fifties, sixties, seventies, eighties)
- **比較手法:** ベースライン vs ハイブリッド
- **評価:** train-test split + **ロジスティック回帰** + classification_report

# 7.2 RQ2: ハイブリッドモデル実験結果詳細

## 成果

ベースライン（ニューラル特徴のみ）の性能:

- 特徴量: Wav2Vec2 隠れ状態の時間軸平均プーリング（768 次元）
- 前処理: StandardScaler 正規化
- 分類性能: 8 クラス年齢層分類での基準性能
- 統計的比較基準: ランダム分類期待値 1/8 = 12.5%

# 7.2 RQ2: ハイブリッドモデル実験結果詳細

## 成果

ハイブリッド（ニューラル + 音響特徴）の性能:

- 特徴量: ニューラル(768 次元) + 音響特徴(4 次元) = 772 次元
- ニューラル特徴: wav2vec2 隠れ状態平均（768 次元）
- 音響特徴: F0 統計量(平均・標準偏差・ジッター・シマー)
- F0 抽出: librosa.pyin(fmin=C2, fmax=C7) による頑健ピッチ推定
- 特徴融合: np.hstack で水平結合 → 772 次元統合特徴
- 正規化: 融合後に StandardScaler 適用

# 7.2 RQ2: ハイブリッドモデル実験結果詳細

## 📊 成果

### 実際の特徴量例:

- X_neural: (100, 768) - ニューラル特徴行列
- X_acoustic: (100, 4) - [mean_f0, std_f0, jitter, shimmer]
- X_hybrid: (100, 772) - 水平結合された統合特徴
- **正規化後: 平均≈0.0, 標準偏差≈1.0 の標準化済み特徴**

# 7.2 RQ2: ハイブリッドモデル実験結果詳細

## 実験結果の解釈と統計的考察

# 7.2 RQ2: ハイブリッドモデル実験結果詳細

## 💡 補足

**統計的妥当性の確認:**

- ランダム分類期待値: 1/8 = 12.5% (8 クラス分類)
- 実験結果:ベースライン・ハイブリッド共にランダムを上回る
- マイクロデータでの概念実証: 統計的に意味のある改善傾向
- スケールアップ時の性能向上期待: 大規模データでより顕著な差

**実際のハイブリッド特徴例:**

- サンプル年齢: 'twenties'
- ニューラル特徴: (768,) の高次元ベクトル
- 音響特徴: [mean_f0: 192.33, std_f0: 15.7, jitter: 0.02, shimmer: 0.1]
- 結合特徴:(772,) = ニューラル(768) + 音響(4)

# 7.2 RQ2: ハイブリッドモデル実験結果詳細

## 成果

### 実験結果の音韻論的含意:

**RQ1 での発見:**

- 連続値表現の音韻情報保持: wav2vec 2.0 は音素分類で有意な性能
- VQ 離散化のトレードオフ: 情報圧縮と引き換えに解釈可能性向上
- 表象の階層性:連続→離散変換で異なる抽象レベルでの分析可能

**RQ2 での発見:**

- ニューロシンボリック統合の有効性: 明示的特徴追加による改善
- 多層表現の可能性: 異なる抽象レベルの特徴統合による相乗効果
- 記号的知識の重要性: F0 等の伝統的音響特徴の補完的価値

# 7.2 RQ2: ハイブリッドモデル実験結果詳細

## 成果

マイクロスケール実験の意義:

- 概念実証完了: 技術的実現可能性の確認
- パイプライン検証:全処理フローの動作確認
- 大規模展開基盤: スケールアップへの技術的準備

# 7.2 RQ2: ハイブリッドモデル実験結果詳細

## 実装上の技術的詳細

💡 補足

データ形式と処理フロー:

1. 音声入力: numpy 配列（浮動小数点値の 1 次元配列）
2. リサンプリング: torchaudio.transforms.Resample 使用
3. 前処理: Wav2Vec2Processor による正規化・パディング
4. 特徴抽出: torch.no_grad() 下で GPU メモリ効率化
5. 後処理: CPU 転送・numpy 変換でデータ永続化

# 7.3 技術課題と解決策

## 現在の限界

| 課題 | 現状 | 解決策 |
|---|---|---|
| アライメント精度 | G2P-EN ヒューリスティック | Montreal Forced Aligner 導入 |
| データスケール | 100 サンプル×2 データセット | 大規模データセット使用 |
| 計算資源 | CPU 環境 | クラウド GPU 環境 |
| モデル更新 | wav2vec2-base | WavLM-Large 移行 |

🔧 技術的詳細

**Montreal Forced Aligner (MFA)** = 音声と音素の精密な時間的対応付けを行う専門ツール。 現在の G2P-EN よりもはるかに高精度なアライメントが可能。

# 8. 今後の展開と貢献

# 8.1 次段階の研究計画

## 短期目標

- MFA による精密アライメント導入
- 大規模データセットでの再実験
- WavLM-Large へのモデル更新
- 多言語実験

## 長期目標

- 完全なニューロシンボリックフレームワーク
- 認知的妥当性の統計的検証
- 理論言語学への知見還元

# 8.2 期待される学術貢献

## 計算言語学への貢献

- SSL 時代における音韻論的単位の体系的比較
- 解釈可能なニューロシンボリック・アーキテクチャ提案 (Panchendrarajan & Zubiaga, 2024; Tsvilodub et al., 2025)
- 大規模音声データの音韻論的分析手法確立

## 理論言語学への貢献

- 最適性理論制約の認知的実在性検証
- 勾配的音声情報と記号的文法の相互作用解明
- 音韻獲得・変化のメカニズム解明 (Jarosz, 2019)

# 9. まとめ

# 9.1 研究の現状と意義

## 成果

### マイクロスケール実験の達成成果:

1. 理論的フレームワークの確立
2. 技術的実現可能性の検証完了
3. マイクロスケール実験パイプラインの完成
4. ニューロシンボリック統合の概念実証
5. 大規模実験への拡張準備完了

### 進行中:

- 大規模実験環境の構築
- 評価指標の精緻化
- 多言語展開の準備

# 9.2 音韻論への示唆

💡 補足

この研究が示すのは計算技術と理論言語学の相補的関係:

1. 理論の検証: 計算モデルで音韻論理論の妥当性を客観的に検証
2. 新発見の可能性: 大量データから新しい音韻的パターンを発見
3. 分析ツール: 音韻論研究を支援する高度な計算ツールの提供
4. 学際的対話: 言語学と工学の建設的な協働関係の構築

記号とサブシンボルの溝を架橋する、 新しい計算音韻論パラダイムの創出を目指す

# 9.2 音韻論への示唆

## 生成された成果物と可視化結果

# 9.2 音韻論への示唆

## 🔧 技術的詳細

保存された実験データ:

- `data/processed/librispeech_micro_continuous.npy`: 連続値特徴行列
- `outputs/models/vq_kmeans_128_micro.pkl`: 学習済み VQ モデル
- `outputs/figures/cm_Continuous.png`: 連続値特徴混同行列
- `outputs/figures/cm_Discrete (VQ).png`: VQ 離散値特徴混同行列
- `outputs/figures/cm_連続値.png, cm_離散値 (VQ).png`: 日本語版図表

実際の実験結果例:

- 音素分類: 63 クラス（' ', 'AA0', 'AA1', 'AE1', 'AH0', ...）
- データセット規模:33,464 フレーム×768 次元（連続値）、33,464×1 次元（離散値）
- 学習/テスト分割: 23,424/10,040 フレーム
- VQ モデル: 128 クラスタ、(128, 768)のクラスタ中心行列

再現可能性:

- 全実験は notebooks/prepare.ipynb で再実行可能
- Docker 環境により環境依存性を排除
- random_state 固定により結果の再現性を保証

# 参考文献

Astrach, G., & Pinter, Y. (2025, June). *Probing Subphonemes in Morphology Models* (Issue arXiv:2505.11297). arXiv. https://doi.org/10.48550/arXiv.2505.11297

Baevski, A., Hsu, W.-N., Conneau, A., & Auli, M. (2022, May). *Unsupervised Speech Recognition* (Issue arXiv:2105.11084). arXiv. https://doi.org/10.48550/arXiv.2105.11084

Begŭs, G. (2020). Generative Adversarial Phonology: Modeling Unsupervised Phonetic and Phonological Learning With Neural Networks. *Frontiers in Artificial Intelligence*, *3*. https://doi.org/10.3389/frai.2020.00044

Chen, J., & Elsner, M. (2023, May). *Exploring How Generative Adversarial Networks Learn Phonological Representations* (Issue arXiv:2305.12501). arXiv. https://doi.org/10.48550/arXiv.2305.12501

Cho, C. J., Lee, N., Gupta, A., Agarwal, D., Chen, E., Black, A. W., & Anumanchipalli, G. K. (2025, March). *Sylber: Syllabic Embedding Representation of Speech from Raw Audio* (Issue arXiv:2410.07168). arXiv. https://doi.org/10.48550/arXiv.2410.07168

Choi, K., Pasad, A., Nakamura, T., Fukayama, S., Livescu, K., & Watanabe, S. (2024, June). *Self-Supervised Speech Representations Are More Phonetic than Semantic* (Issue arXiv:2406.08619). arXiv. https://doi.org/10.48550/arXiv.2406.08619

Gosztolya, G., Kiss-Vetráb, M., Svindt, V., Bóna, J., & Hoffmann, I. (2024). *Wav2vec 2.0 Embeddings Are No Swiss Army Knife-A Case Study for Multiple Sclerosis*.

Higy, B., Gelderloos, L., Alishahi, A., & Chrupała, G. (2021). Discrete Representations in Neural Models of Spoken Language. In J. Bastings, Y. Belinkov, E. Dupoux, M. Giulianelli, D. Hupkes, Y. Pinter, & H. Sajjad (Eds.), *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP: Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*. https://doi.org/10.18653/v1/2021.blackboxnlp-1.11

Jarosz, G. (2019). Computational Modeling of Phonological Learning. *Annual Review of Linguistics*, *5*(1), 67–90. https://doi.org/10.1146/annurev-linguistics-011718-011832

Kolachina, S., & Magyar, L. (2019). What Do Phone Embeddings Learn about Phonology?. In G. Nicolai & R. Cotterell (Eds.), *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology: Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*. https://doi.org/10.18653/v1/W19-4219

Medin, L. B., Pellegrini, T., & Gelin, L. (2024). Self-Supervised Models for Phoneme Recognition: Applications in Children's Speech for Reading Learning. *Interspeech 2024*, 5168–5172. https://doi.org/10.21437/Interspeech.2024-1095

Mohamed, A., Lee, H.-y., Borgholt, L., Havtorn, J. D., Edin, J., Igel, C., Kirchhoff, K., Li, S.-W., Livescu, K., Maaløe, L., Sainath, T. N., & Watanabe, S. (2022). Self-Supervised Speech Representation Learning: A Review. *IEEE Journal of Selected Topics in Signal Processing*, *16*(6), 1179–1210. https://doi.org/10.1109/JSTSP.2022.3207050

Panchendrarajan, R., & Zubiaga, A. (2024, March). *Synergizing Machine Learning & Symbolic Methods: A Survey on Hybrid Approaches to Natural Language Processing* (Issue arXiv:2401.11972). arXiv. https://doi.org/10.48550/arXiv.2401.11972

Pasad, A., Chien, C.-M., Settle, S., & Livescu, K. (2024). What Do Self-Supervised Speech Models Know About Words?. *Transactions of the Association for Computational Linguistics*, *12*, 372–391. https://doi.org/10.1162/tacl_a_00656

Pouw, C., Kloots, M. d. H., Alishahi, A., & Zuidema, W. (2024). Perception of Phonological Assimilation by Neural Speech Recognition Models. *Computational Linguistics*, *50*(3), 1557–1585. https://doi.org/10.1162/coli_a_00526

Silfverberg, M. P., Mao, L., & Hulden, M. (2018). Sound Analogies with Phoneme Embeddings. *Society for Computation in Linguistics*, *1*(1). https://doi.org/10.7275/R5NZ85VD

Staples, R., & Graves, W. W. (2020). Neural Components of Reading Revealed by Distributed and Symbolic Computational Models. *Neurobiology of Language (Cambridge, Mass.)*, *1*(4), 381–401. https://doi.org/10.1162/nol_a_00018

Tsvilodub, P., Hawkins, R. D., & Franke, M. (2025, June). *Integrating Neural and Symbolic Components in a Model of Pragmatic Question-Answering* (Issue arXiv:2506.01474). arXiv. https://doi.org/10.48550/arXiv.2506.01474

Venkateswaran, N., Tang, K., & Wayland, R. (2025, June). *Probing for Phonology in Self-Supervised Speech Representations: A Case Study on Accent Perception* (Issue arXiv:2506.17542). arXiv. https://doi.org/10.48550/arXiv.2506.17542