1R-06

実時間で動作する音響イベント検出の大規模事前学習

大田 竹蔵[1,2]　坂東 宜昭[2]　井本桂右[3,2]　大西 正輝[2,1]
[1]筑波大学　[2]産業技術総合研究所　[3]同志社大学

1. はじめに

混合音に含まれる音響イベントの種類と発生時刻を推定する音響イベント検出は，周囲の音環境を把握するための重要な基盤技術である．このような技術は，周囲の環境に応じて適切に応答するロボットや，セキュリティ・見守りシステムなどに活用される．多くの枠組みは，混合音スペクトログラムを入力とし，各時刻の音響イベントの事後確率を予測するモデルを教師あり学習する．

既存の多くの音響イベント検出では，入力信号全体を一度に処理することを前提にしたオフライン型の枠組みが用いられる．特に，注意機構を用いて信号全体の大域的な特徴を捉えることが，性能改善に有効であると知られている [1, 2]．例えば，入力スペクトログラムをパッチに分割して，これらの間の関係性を学習する Audio Spectrogram Transformer (AST) [2] が高い性能を達成している．DCASE2023 Challenge Task4 [3] では，AST を大規模データから事前学習した BEATs [4] による埋め込みを用いることにより，少量の教師付きデータから高性能なモデルを学習できることが確認されている．

ロボットやセキュリティシステムへの応用では，実時間で動作する音響イベント検出が重要となる．最も素朴な実時間推論の方法は，オフライン型の推論モデルを時間方向にスライドさせながら逐次的に適用することである [5]．ある時刻の音響イベントを弁別するには，一定区間の未来の情報が必要になる場合も多いが，十分な区間がバッファに蓄積するまでの遅延が生じる．特に AST では，中間層の出力全てが入力系列全体に依存するため過去の計算結果を再利用できず，過去に一度推論した区間を繰り返し処理する必要があり非効率である．

本稿では，過去の区間の情報を蓄積する記憶トークンを導入し，短時間に分割した入力信号を逐次推論する AST を提案する．過去の区間の情報を記憶トークンへ繰り返し蓄積させることで，過去の区間に対する再計算を不要にする．さらに，大規模データから事前学習されたオフライン型 AST である BEATs を用いた知識蒸留に基づき，逐次推論型 AST の事前学習を行う．AST は少量データでは過学習しやすいが，本研究では知識蒸留したモデルを転移学習することにより比較的小さなデータセットからの学習を可能にした．

2. 実時間処理のための逐次推論型 AST

本稿では，入力信号を短時間のチャンクに分割して AST で逐次推論する．過去の信号を入力する代わりに，記憶トークンをチャンクと合わせて逐次的に入出力する．このモデルの学習には，事前学習済みのオフライン型モデルである BEATs の知識蒸留を用いる (図 1)．学習は一定長の信号からオフライン型モデルと同様に行う．

2.1 記憶トークンに基づく逐次推論型 AST

提案法では，入力スペクトログラムを 16×16 のパッチに区切り線形変換したトークン $\mathbf{x}_{tf} \in \mathbb{R}^D$ に対して逐次推論を行う．ただし，t と f はそれぞれ時間方向，周波数方向のインデックスとする．これらを時間方向に T パッチずつ区切ったものをチャンク $\mathbf{X}^{(i)} \triangleq \{\mathbf{x}_n^{(i)}\}_{n=1}^{N}$ として，現時点から K 個先までのチャンク列 $\mathbf{X}^{(i:i+K)} \triangleq \{\mathbf{X}^{(i)}, \cdots, \mathbf{X}^{(i+k)}\} \in \mathbb{R}^{N \times D}$ を入力として推論をする．ここで，i はチャンクの番号を表し，$n = 1, \cdots, N$ はチャンク内のパッチを直列に並べたときの番号である．また，過去の情報を伝播させるため，直前のチャンクの推論で出力された M 本の記憶トークン $\mathbf{M}^{(i-1)} \triangleq \{\mathbf{m}_m^{(i-1)}\}_{m=1}^{M} \in \mathbb{R}^D$ を入力として併用する．エンコーダ $\mathcal{F}$ は，これらの入力トークンから埋め込み $\mathbf{Z}^{(i:i+K)} \triangleq \{\mathbf{Z}^{(i)}, \cdots, \mathbf{Z}^{(i+k)}\} \in \mathbb{R}^{N \times D}$ と記憶トークン $\mathbf{M}^{(i)}$ を出力する．

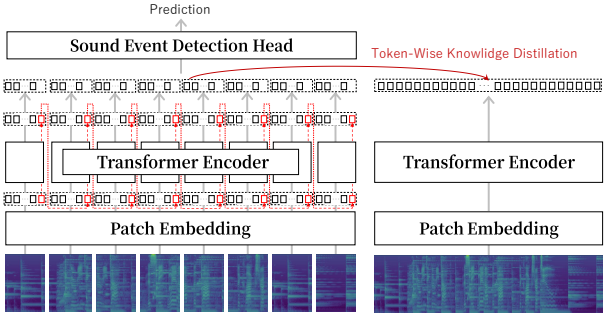

図 1: 逐次型 AST アーキテクチャ・蒸留

$$\{\mathbf{Z}^{(i:i+K)}, \mathbf{M}^{(i)}\} = \mathcal{F}(\mathcal{P}(\{\mathbf{X}^{(i:i+K)}, \mathbf{M}^{(i-1)}\})) \quad (1)$$

ただし，P は正弦波位置埋め込み層を表す．

2.2 BEATs の知識蒸留に基づく事前学習

提案する逐次推論型 AST と学習済みの AST である BEATs による埋め込みを近づけるように，オフライン型 AST を知識蒸留する．j 番目の $\mathbf{Z}^{(i)}$ の要素を $\mathbf{z}_j^{(s)} \in \mathbb{R}^D$ とすると，学習に用いる損失関数 $\mathcal{L}$ は，L 個のチャンクに対する逐次推論型 AST の出力埋め込み $\{\mathbf{z}_j^{(s)}\}_{i=j}^{LN}$ と BEATs の出力埋め込み $\{\mathbf{z}_j^{t}\}_{j=1}^{LN} \in \mathbb{R}^D$ に対するトークン対ごとのコサイン類似度の平均値で表される．

$$\mathcal{L} = -\frac{1}{LN} \sum_{j=1}^{LN} \frac{\mathbf{z}_j^{s\mathsf{T}}}{||\mathbf{z}_j^{s}||} \frac{\mathbf{z}_j^{t}}{||\mathbf{z}_j^{t}||} \quad (2)$$

大規模なデータから事前学習した BEATs の埋め込みを教師とした知識蒸留により，少ない量のデータからでも高性能な逐次推論型 AST の学習を可能にする．

2.3 音響イベント検出へのための逐次推論型 AST

チャンクの出力埋め込み $\mathbf{Z}^{(i)}$ で，周波数インデックス f が同じパッチに平均値プーリングを適用し，時間フレームごとの D' 次元の埋め込み $\mathbf{H}^{(i)} \in \mathbb{R}^{T \times D'}$ を得る．$\mathbf{H}^{(i)}$ は出力層 $\mathcal{H}$ により i 番目のチャンクのフレームごとの音響イベントの発生確率 $y_{ct}^{(i)} \in [0, 1]$ に変換される．

$$\{y_{ct}^{(i)}\}_{c,t=1}^{C,T} = \mathcal{H}(\mathbf{H}^{(i)}) \quad (3)$$

ただし，$c = 1, \cdots, C$ は音響イベントの種類を表し，出力層 $\mathcal{H}$ は 2 層の線形層で構成される．

A Large-Scale Pre-Training for Real-Time Sound Event Detection: *T. Ohta, Y. Bando, K. Imoto, M. Onishi*

表 1: URBAN-SED での検出結果

手法	T	PSDS1	PSDS2	F1
DCASE CRNN	N/A	0.344	0.483	44.9%
オフライン型 AST	N/A	0.314	0.495	41.6%
逐次推論型 AST	3	0.302	0.470	41.9%
逐次推論型 AST	6	0.350	0.541	43.7%
逐次推論型 AST	12	0.339	0.540	40.8%

表 2: DESED での検出結果

手法	T	PSDS1	PSDS2	F1
DCASE CRNN	N/A	0.321	0.559	47.5%
オフライン型 AST	N/A	0.243	0.641	53.1%
逐次推論型 AST	3	0.250	0.575	46.7%
逐次推論型 AST	6	0.251	0.650	47.9%
逐次推論型 AST	12	0.259	0.629	48.2%

2.4 音響イベント検出への転移学習

音響イベント検出の学習は一定の長さの音響信号単位の損失を計算して行う．チャンクごとに逐次的に推論した予測結果 $y_{ct}^{(i)}$ を，時系列順に並べたものをモデル全体の出力とする．得られた出力と正解ラベルとの交差エントロピー損失により学習する．

3. 評価実験

大規模データを用いた知識蒸留により事前学習した逐次推論型 AST の検出性能と計算時間を評価した．

3.1 データセット

BEATs の知識蒸留には AudioSet [6] を用いた．AudioSet は YouTube から収集されたデータセットであり，音声や音楽を含む様々な環境音で構成される．全体の 85%に当たる約 4,800 時間分の動画を収集し，そのうちの 10 秒前後の信号長のクリップを用いた．約 4,500 時間を学習データ，約 430 時間を検証データとした．

音響イベント検出の性能評価には URBAN-SED [7] と DESED [8] を用いた．URBAN-SED は 10 秒間の屋外環境のシミュレーション信号から成るデータセットであり，16.7 時間の学習データ，5.6 時間の検証データ，5,6 時間のテストデータで構成される．DESED は 10 秒間の屋内環境の実録音信号，シミュレーション信号で構成されるデータセットである．本稿では，発生時刻が付与されているもののみを用いて，36 時間を学習データ，3 時間を検証データ，2 時間をテストデータとした．

3.2 実験条件

入力信号は 16 kHz にリサンプリングし，メルスペクトログラムに変換した．短時間フーリエ変換は窓長を 400 サンプル，ホップ長を 160 サンプルとし，メルビン数を 128 とした．正解ラベルの時間解像度はパッチのシフト長に合わせて 160ms とした．バッチサイズは事前学習，転移学習それぞれで 256, 64 とした．転移学習では，学習データに Mixup [9]，SpecAugment [10] を適用した．性能の評価には PSDS1，PSDS2，イベント基準のマクロ F1 スコアを用いた．マクロ F1 スコアの時間許容幅は出力の時間解像度を考慮して 400ms とした．記憶トークンの数は $M = 20$ とした．比較手法として DCASE2023 Challenge Task4 Baseline の畳み込み再帰型ニューラルネットワーク (DCASE CRNN) と，入力信号をチャンクに区切らないオフライン型 AST の学習を行った．チャンクの時間方向のパッチ数 T は変化させて実験を行った．

3.3 実験結果

URBAN-SED, DESED に対する検出性能の評価結果を表 1，2 に示す．表 1 から，URBAN-SED では，提案法の AST が DCASE CRNN, オフライン型 AST と同等以上の性能であることが分かる．また表 2 から，実録音である DESED のテストデータでも，PSDS1 は DCASE CRNN より低いものの他の評価指標で同等以上の性能を達成した．また，どちらのデータセットでも時間方向のパッチ数 T は 6 の時に性能が高く，3 まで幅を狭めると性能が低下したことが分かる．

表 3: チャンク当たりの推論時間

手法	T	処理時間（秒）
オフライン型 AST	6	4.56×10^{-3} 秒
逐次推論型 AST	3	4.42×10^{-3} 秒
逐次推論型 AST	6	4.44×10^{-3} 秒
逐次推論型 AST	12	4.45×10^{-3} 秒

表 3 に，提案した逐次推論型 AST とオフライン型 AST の，NVIDIA V100 GPU を用いた場合の DESED のテストデータに対するチャンク当たりの推論時間の計測結果を表す．過去の信号を繰り返し推論するオフライン型 AST と，記憶トークンに基づく逐次推論型 AST では，推論時間に有意な差は見られなかった．また，時間方向のパッチ数を変化させても処理時間の差は確認されなかった．入力するチャンク分処理時間が増えると考えられるため，1 チャンク分の待ち時間が許容できる場合は，$T = 3$ (480 ms) よりも検出性能が高い $T = 6$ (960 ms) を選択すべきだと考えられる．

4. おわりに

実時間処理向けに拡張した逐次推論型 AST を提案し，BEATs の埋め込みに基づく知識蒸留により学習した．記憶トークンを活用することで，従来のオフライン型 AST と同等以上の性能を維持した．また，汎用 GPU 上での計算時間の計測を行った．今後は，限られた計算資源でのチャンク幅や入力チャンク数の評価を行う．

謝辞: 本研究の一部は，NEDO の支援を受けた．

参考文献

[1] K Miyazaki *et al.* Conformer-based sound event detection with semi-supervised learning and data augmentation. *Proc. of DCASE Workshop*, 1:100–104, 2020.
[2] K Li *et al.* AST-SED: An Effective Sound Event Detection Method Based on Audio Spectrogram Transformer. In *Proc. IEEE ICASSP*, 1–5. IEEE, 2023.
[3] M Chen *et al.* DCASE 2023 challenge task4 technical report. Technical report, DCASE2023 Challenge, Tech. Rep, 2023.
[4] S Chen *et al.* Beats: Audio pre-training with acoustic tokenizers. *arXiv preprint arXiv:2212.09058*, 2022.
[5] S Chen *et al.* Continuous speech separation with conformer. In *Proc. IEEE ICASSP*, 5749–5753. IEEE, 2021.
[6] J Gemmeke *et al.* Audio set: An ontology and human-labeled dataset for audio events. In *Proc. of IEEE ICASSP*, 776–780, 2017.
[7] J Salamon *et al.* Scaper: A library for soundscape synthesis and augmentation. In *Proc. of IEEE WASPAA*, 344–348, 2017.
[8] N Turpault *et al.* Sound event detection in domestic environments with weakly labeled data and soundscape synthesis. In *DCASE Workshop*, 1–5, 2019.
[9] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 1–13, 2017.
[10] D. Park *et al.* SpecAugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 1–6, 2019.