環境音の特徴を活用した音響イベント検出・シーン分類

同志社大学 理工学部
井本 桂右

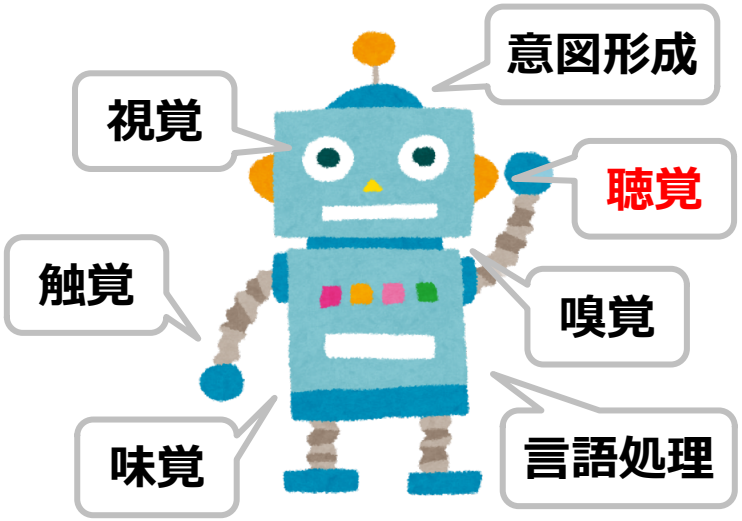
2021.02.26

目次

- 環境音分析の背景と基礎（20分程度）
  - 研究背景，環境音分析の目的
  - 環境音分析の主な問題設定
    - 音響シーン分類，音響イベント検出，異常音検知
  - 環境音分析の課題
- ドメイン知識を活用した環境音分析（30分程度）
  - イベントの共起に着目した音響イベント検出
  - データ不均衡が環境音分析にもたらす影響の調査
- おわりに

機械は音をどこまで理解できる？

- 人の声（音声）の認識
  - 音声認識技術は実用レベルに達している
    - Googleアシスタント
    - Siri
    - Amazon Alexa
    - LINE CLOVA
- その他の音の認識は？



音声のさらなる理解

- パラ言語情報 [藤崎 1994] の理解
  - 発話の**意図**，**発話態度**
    - 疑い，感心，叱責，揶揄，丁寧
- 非言語情報 [藤崎 1994] の理解
  - **感情**
    - 怒り，悲しみ，喜び，嫌悪
  - 年齢
  - 性別
  - 健康状態

音声以外の音の理解

- 環境音の理解
  - 音声や楽音に限らないあらゆる音
    - 足音，車のクラクション，鳥の鳴き声
    - 人が転倒する音，窓ガラスが割れる音

- 環境音の情報ってそんなに重要？
  - 目に見えない場所の状況を把握できる
    - 背後から車が迫ってきている
    - 誰かがドアをノックしている

→ **普段意識しないが人は多くの情報を環境音から得ている**

環境音分析の具体的な活用例

音の種類の分析

- **高齢者/乳幼児の見守り**：転倒する音，泣き声
- **セキュリティ，機械の異常検知**：ガラスが割れる音
- **聴覚障がい者支援**：生活音，車の通過音，警報機の音
- **自動運転**：緊急車両の音，車の通過音

シーンの自動分析

- **動画への自動タグ付与**：動画の内容，撮影された場所
- **ライフログ**：人の行動(料理，睡眠中)，音が収録された場所

見守りシステム

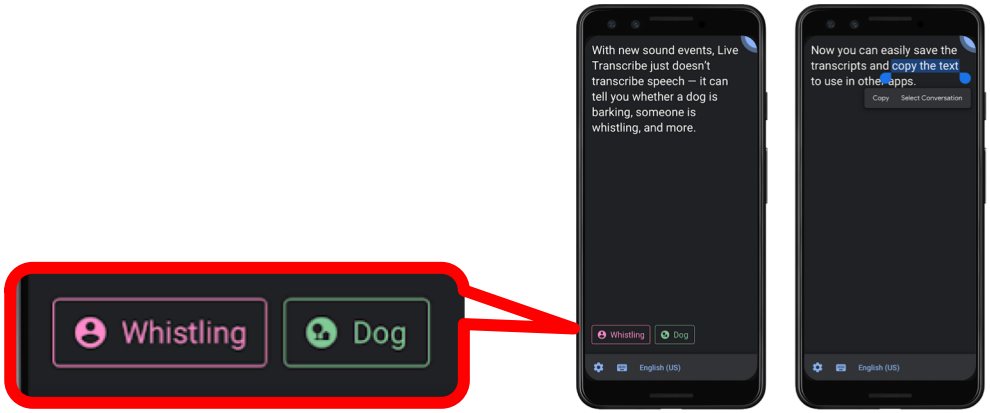
自動運転

防犯システム

環境音分析の実現例

- 手洗い検出 (Apple Watch)
  - マイクロホンとモーションセンサを用いて手洗い検出
- 環境音の書き起こし (Google Live Transcribe)
  - 60種類程度の環境音を書き起こし可能
- ホームセキュリティ (Amazon Alexa Guard)
  - 火災報知器やガラスがわれる音を検出

※ https://www.apple.com/jp/newsroom/2020/06/watchos-7-adds-significant-personalization-health-and-fitness-features-to-apple-watch/



※ https://blog.google/products/android/new-features-make-audio-more-accessible-your-phone/

なぜ今環境音分析なの？

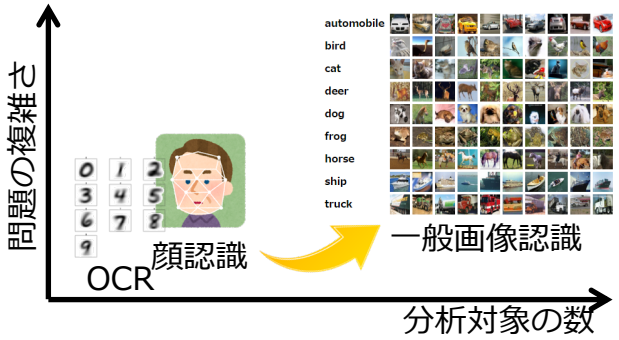
メディア処理を取り巻く環境

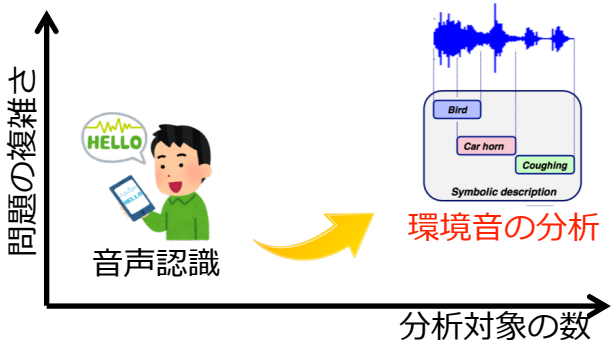
- 気軽に利用可能なセンサが急増
  - スマートホン，スマートスピーカ
  - 情報家電，IoT機器
  - ロボット型デバイス

→ **いつでもどこでも情報の取得が可能に**

メディア処理技術の現状

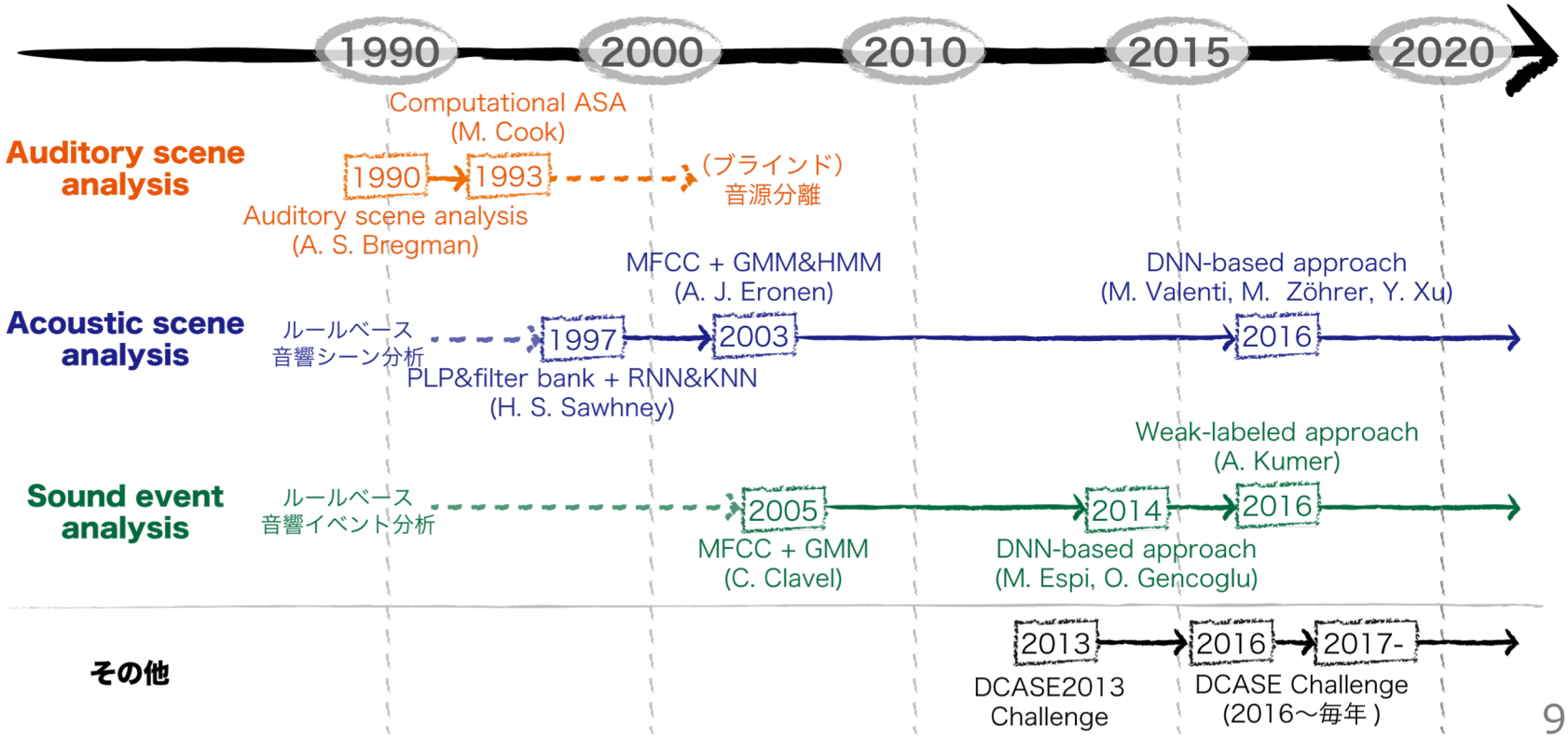
- 画像認識，音声認識，言語処理技術の実用化
- 機械学習理論の発展，計算機環境の充実





環境音分析の歴史

- **比較的歴史の浅い研究分野**
  - 古くから検討は行われていたが論文になりにくかった？
    → 共通的な問題設定，データセット，幅広く利用可能な（統計的）手法がなかった？
- **2013年頃を境に研究が活発に**
  - 深層学習の発展，共通的なデータセット，コンペの登場がきっかけ？

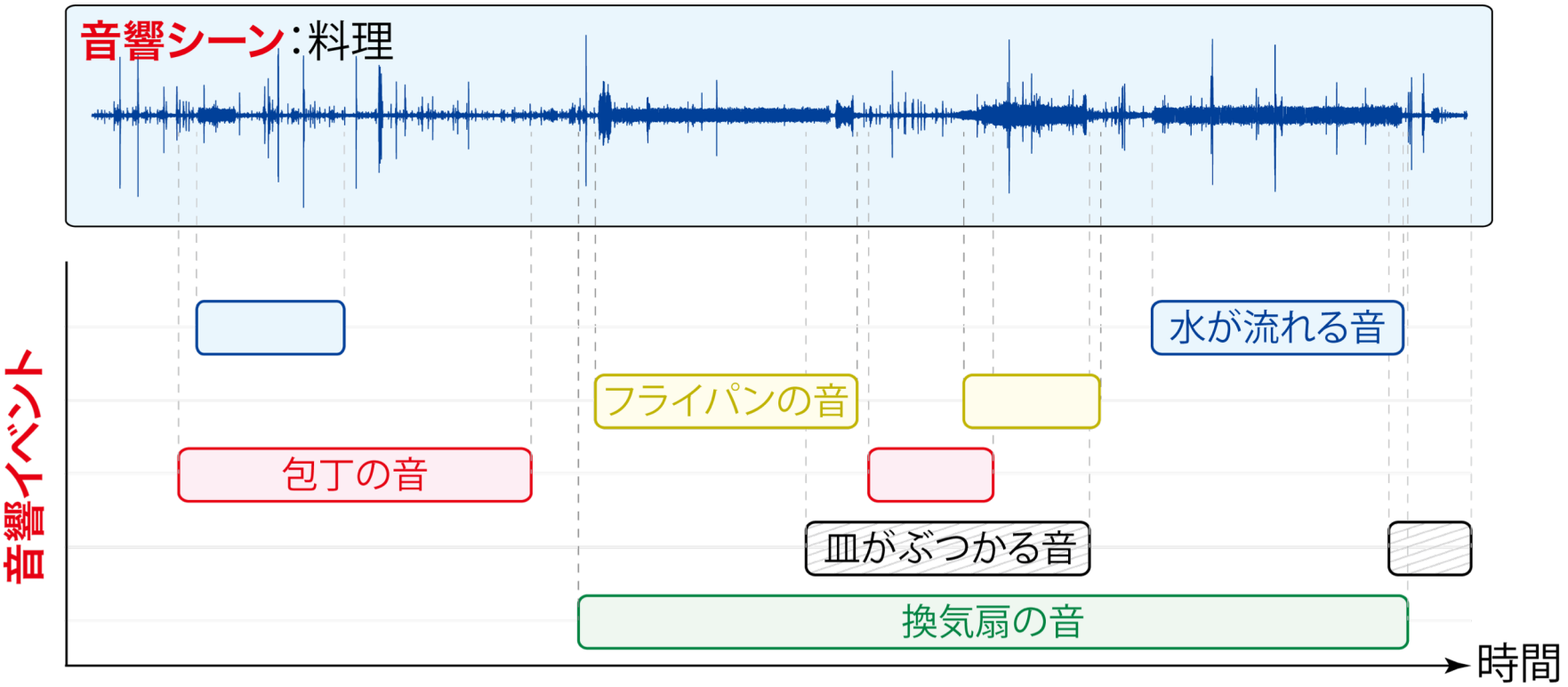


目次

- 環境音分析の背景と基礎（20分程度）
  - 研究背景，環境音分析の目的
  - 環境音分析の主な問題設定
    - 音響シーン分類，音響イベント検出，異常音検知
  - 環境音分析の課題
- ドメイン知識を活用した環境音分析（30分程度）
  - イベントの共起に着目した音響イベント検出
  - データ不均衡が環境音分析にもたらす影響の調査
- おわりに

環境音分析の用語

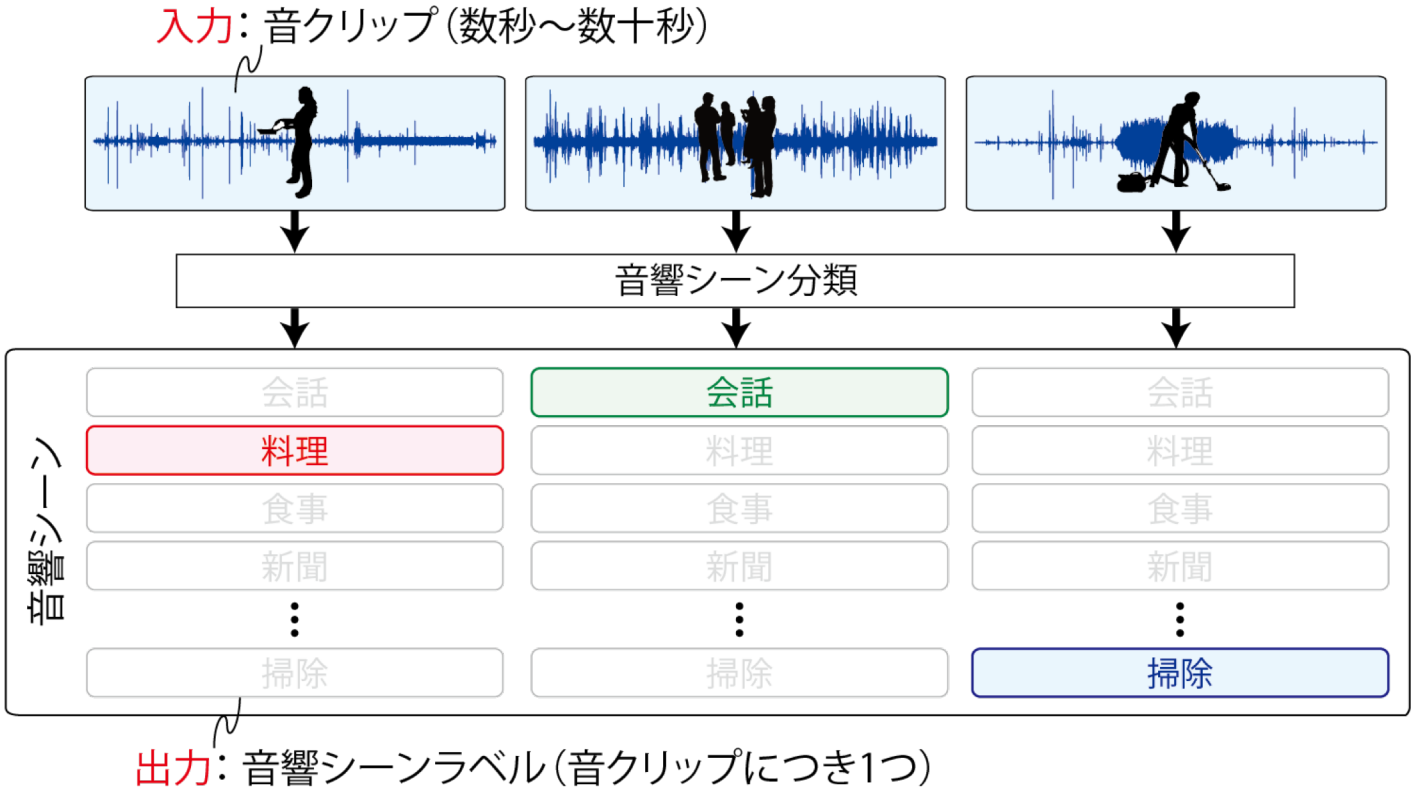
- **音響イベント**：発生音の種類を表す
- **音響シーン**：音が収録された状況や人の行動などを表す



問題設定① 音響シーン分類

環境音から(事前に決められた)シーンを推定

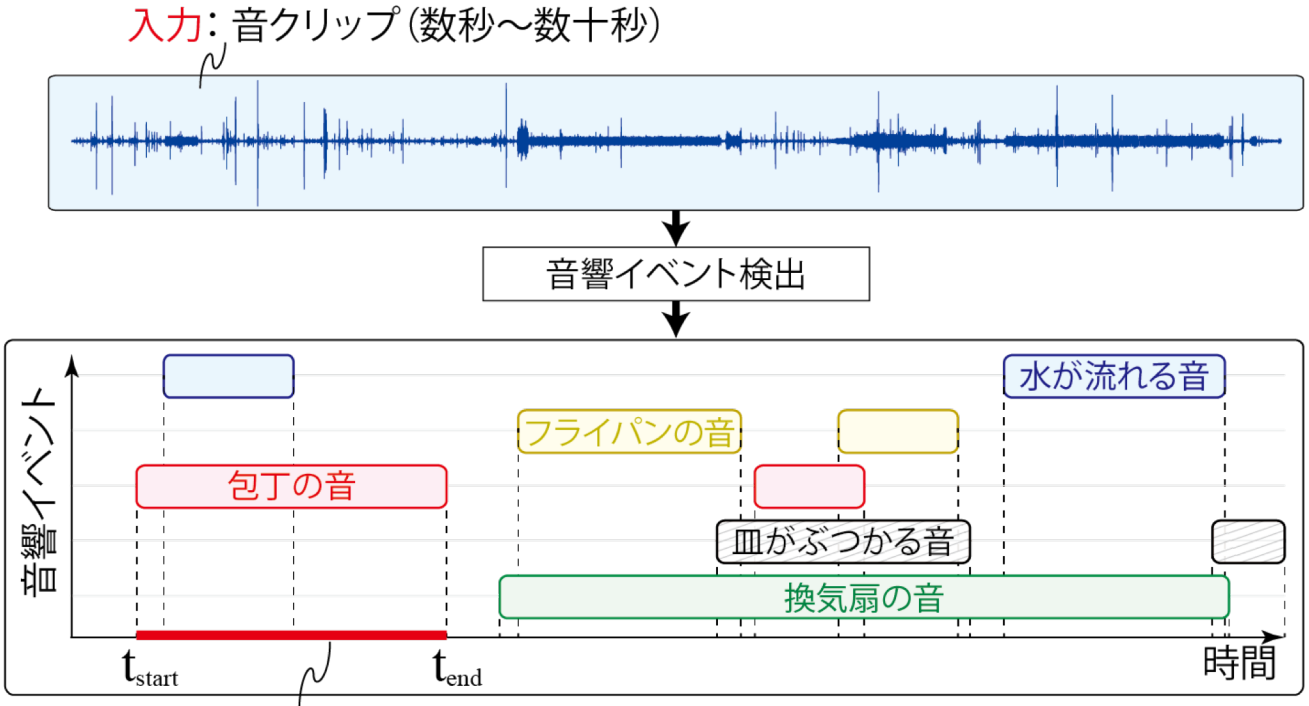
- シーンの例：人の行動，状況，場所など
  - 例）料理，会議中，電車の中，家の中…



問題設定② 音響イベント検出

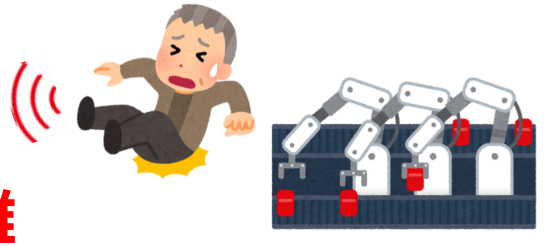
発生した環境音の種類と発生時刻を推定

- 音響イベント検出の特徴
  - **音の重複を考慮する必要**
    - 音声認識では音の重複を考えないことが多い

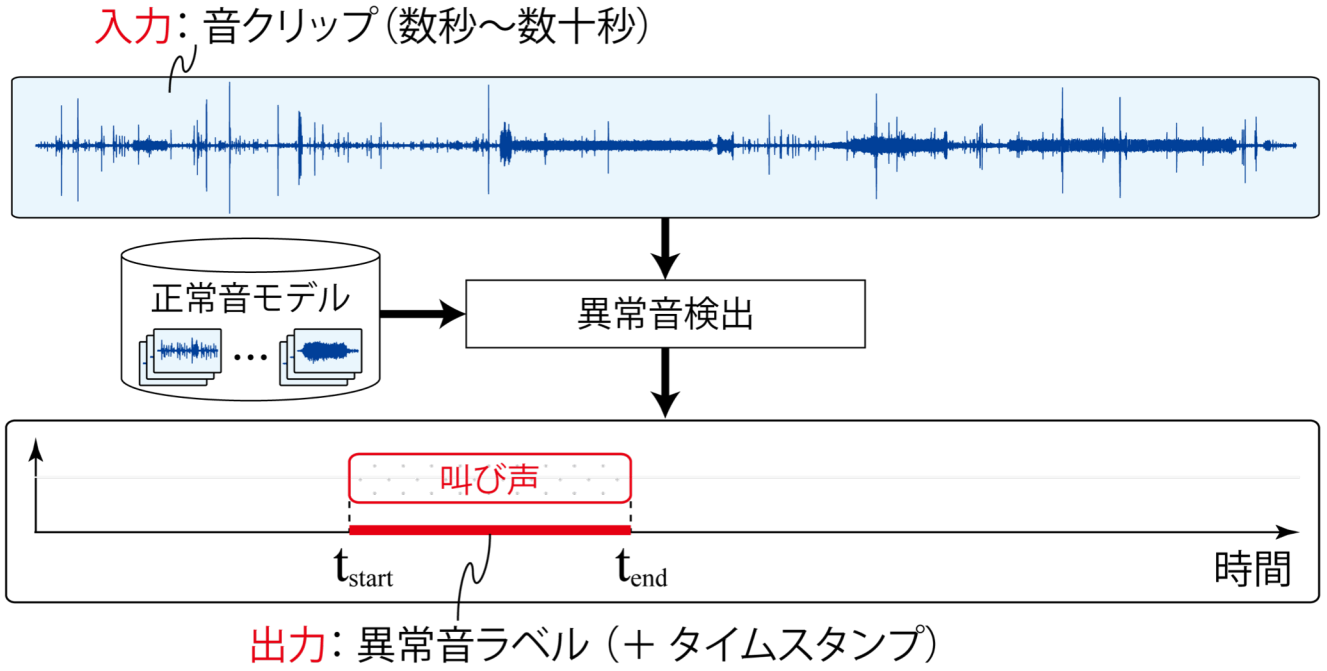


問題設定③ 異常音検知

異常音の発生の有無（と発生時刻）を推定

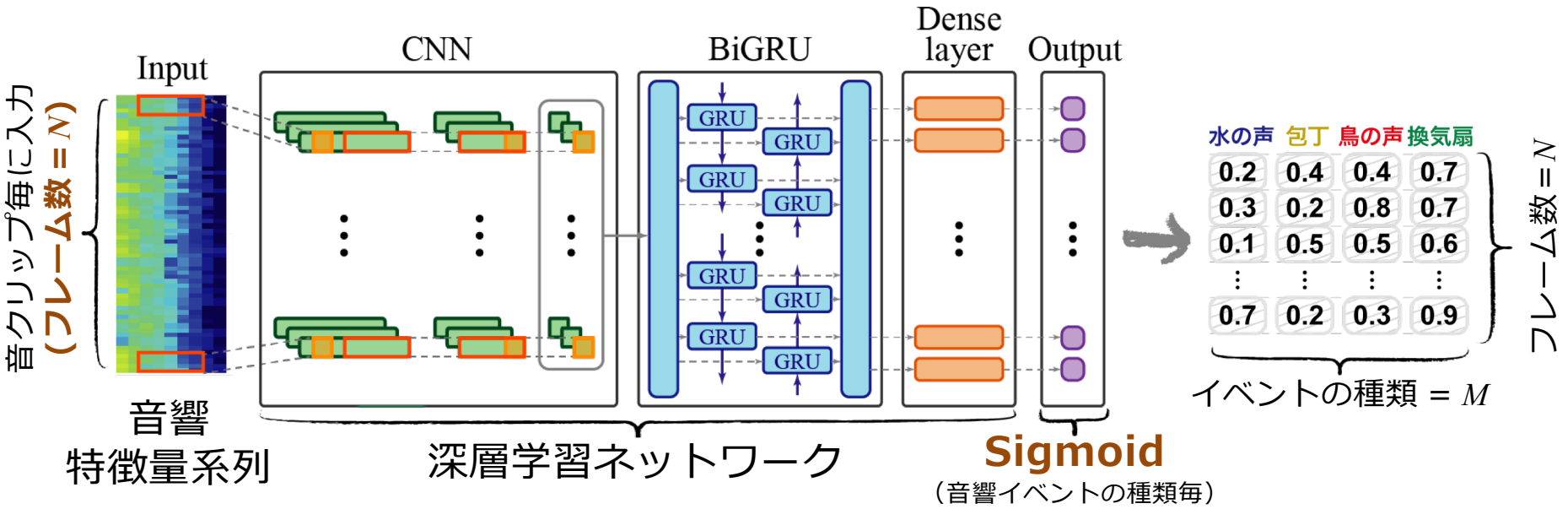


- 異常音検知の特徴
  - 音響イベント検出とよく似ている
  - **検出対象となる異常音が事前に収集困難**
    - 音響イベント検出のように事前に「パターン」を学習できない



音響イベント検出の実現方法

- 深層学習に基づく音響イベント検出
  - CNN + Bidirectional GRUがよく用いられる [Cakır+ 2017]
- 音の発生区間や重複をどう扱う？
  - 時間フレームごとに結果を出力
  - 音響イベントの種類ごとに結果を出力

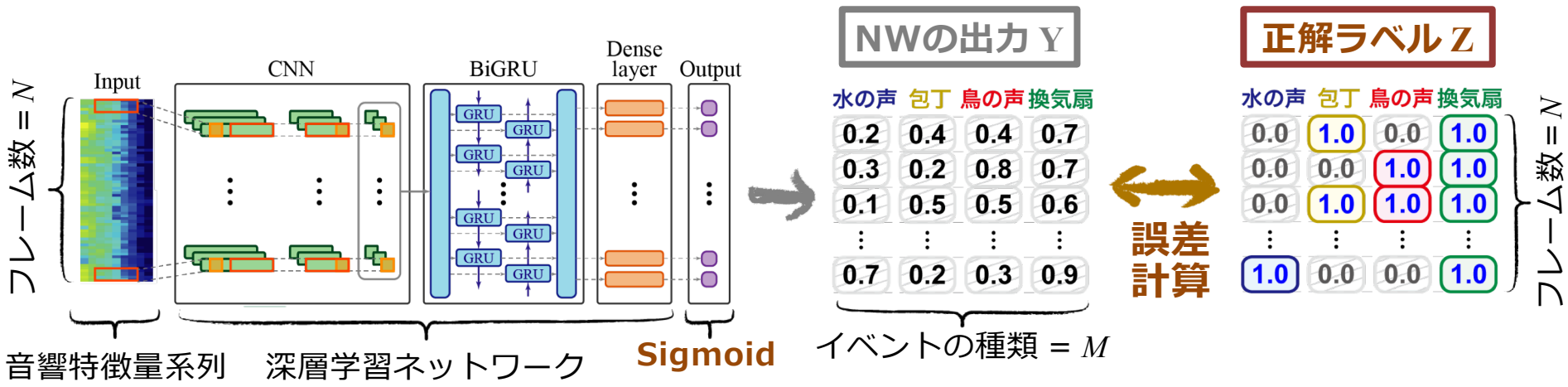


音響イベント検出のモデル学習

モデルパラメータの学習方法

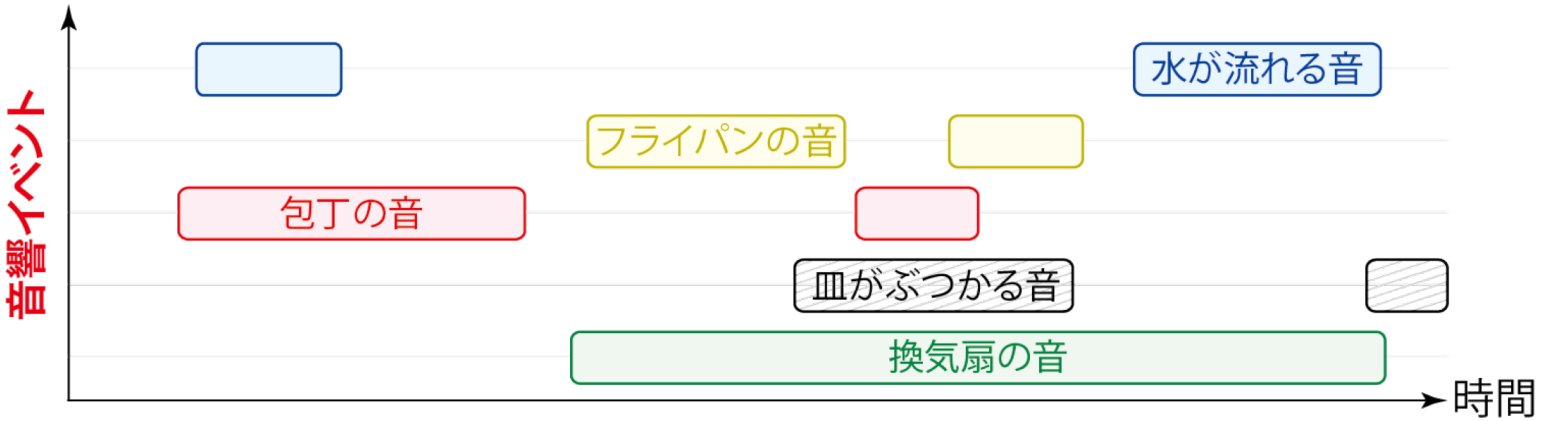
- 各フレーム/イベントの誤差 (Binary cross entropy) の総和が小さくなるようにモデル学習

$$\mathrm{E}_{\mathrm{BCE}}(\theta) = -\sum_{n=1}^{N}\sum_{m=1}^{M}\left\{z_{n,m}\log(y_{n,m}) + (1-z_{n,m})\log(1-y_{n,m})\right\}$$



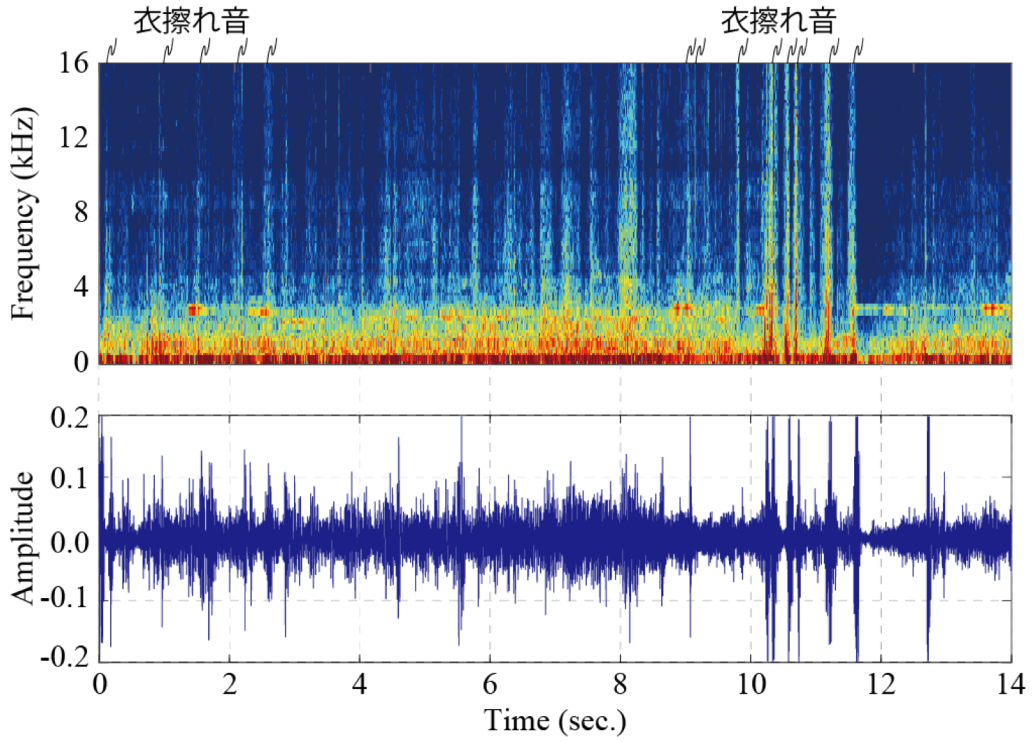
環境音分析の課題①

- データセットの作成が容易でない
  - 音の種類数が膨大，音の発生場所/時間がさまざま
  - 音の発生を制御できないものが多数存在
    - 例）雨音，鳥や動物の鳴き声
  - アノテーションに多大な時間を要する
    - 音が時間的に重複する
    - SNRが低い音も多く含まれる
    - 音オントロジー (音の階層構造)



環境音分析の課題②

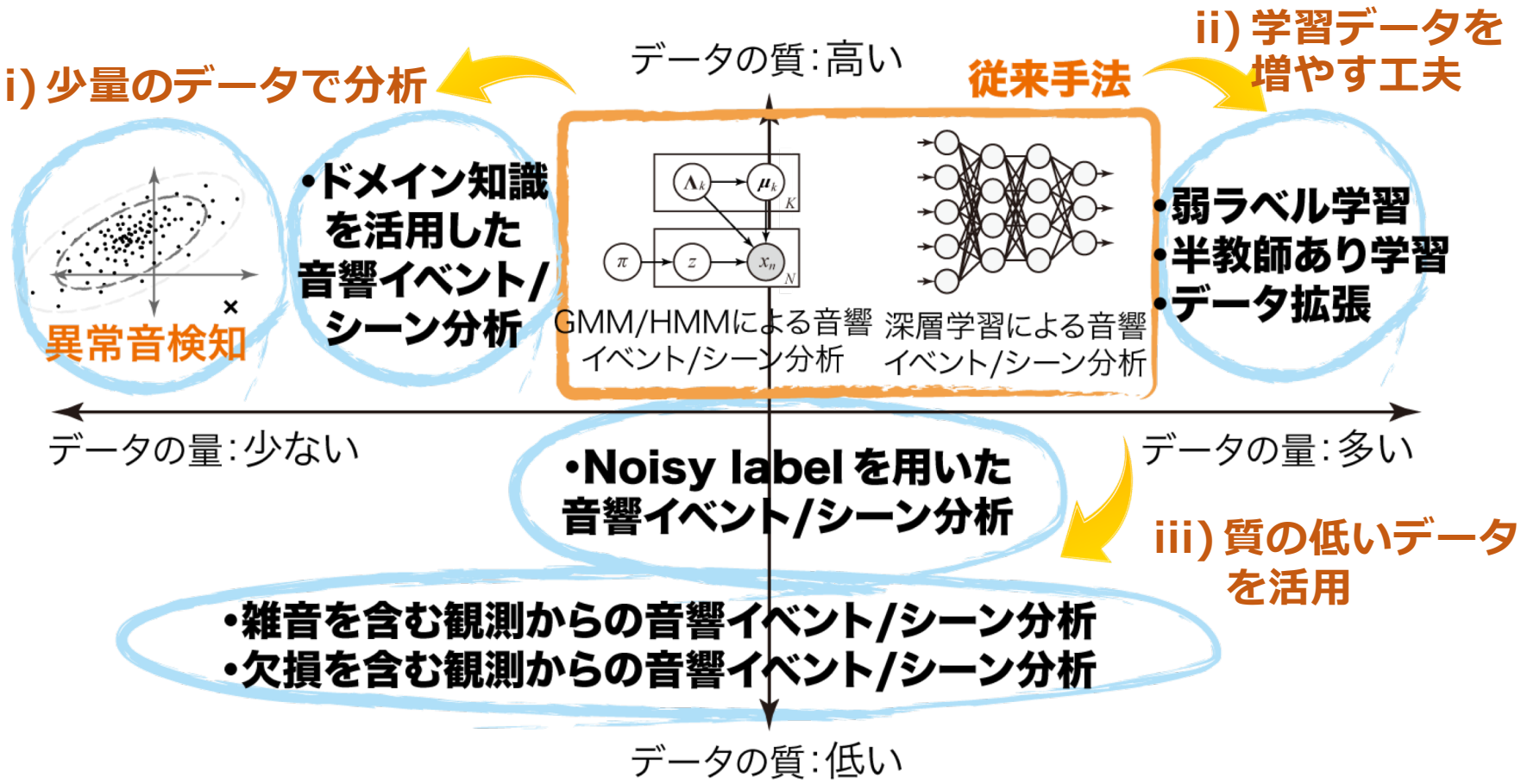
- 観測音に雑音や欠損が混入しやすい
  - 屋外での収音，移動しながらの収音
  - マイクが音源の近傍に配置されることはまれ
  - 通信時のパケットロス



環境音分析の課題の整理

環境音分析の課題と研究動向

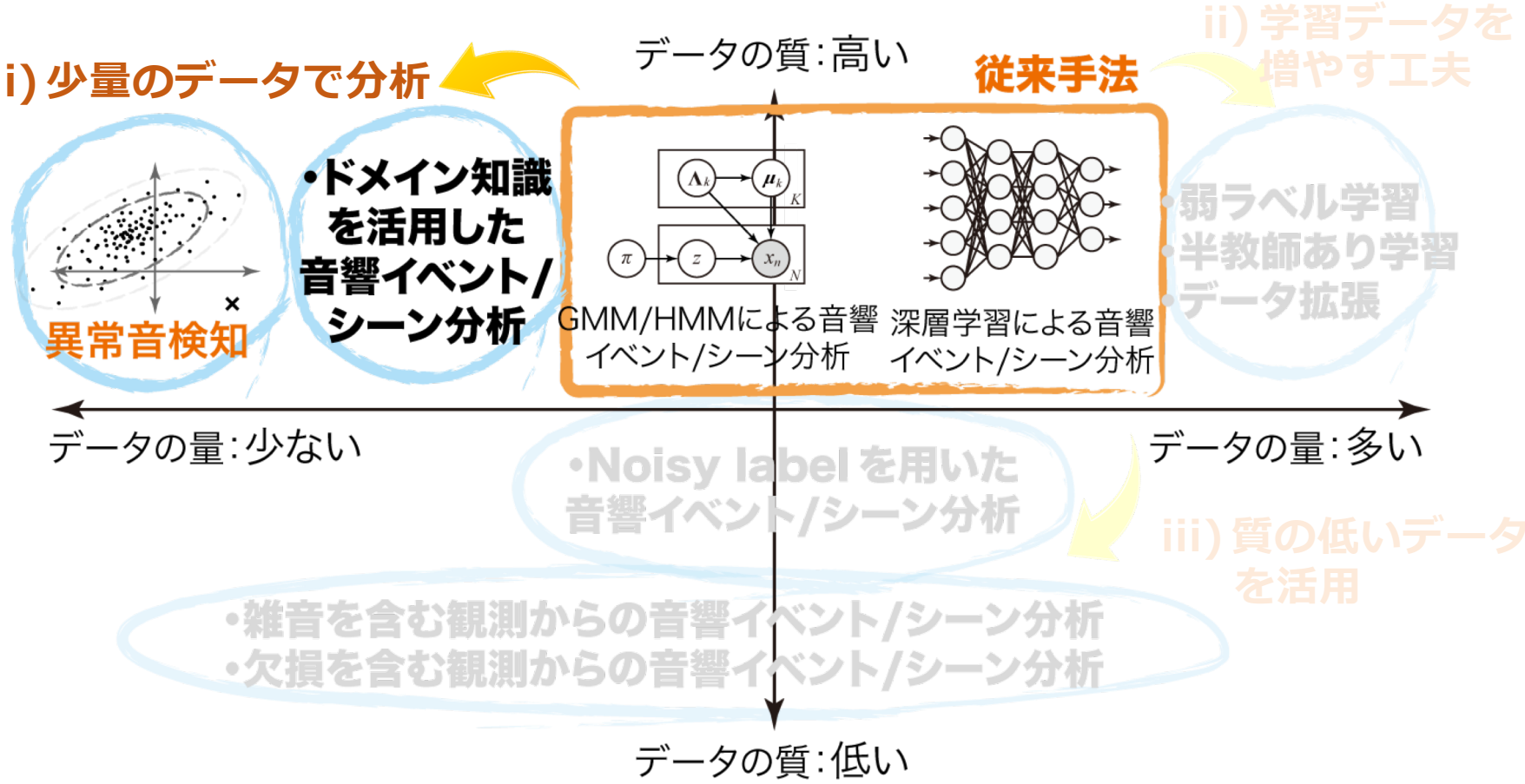
- データの量と質で技術の方向性をプロット



環境音分析の課題の整理

環境音分析の課題と研究動向

- データの量と質で技術の方向性をプロット



目次

- 環境音分析の背景と基礎（20分程度）
  - 研究背景，環境音分析の目的
  - 環境音分析の主な問題設定
    - 音響シーン分類，音響イベント検出，異常音検知
  - 環境音分析の課題
- ドメイン知識を活用した環境音分析（30分程度）
  - イベントの共起に着目した音響イベント検出
  - データ不均衡が環境音分析にもたらす影響の調査
- おわりに

従来法の課題

- CRNNに基づく音響イベント検出の課題
  - 各音響イベントの誤差を独立に計算

$$\mathrm{E}_{\mathrm{BCE}}(\theta) = -\sum_{n=1}^{N}\sum_{m=1}^{M}\left\{z_{n,m}\log(y_{n,m}) + (1-z_{n,m})\log(1-y_{n,m})\right\}$$

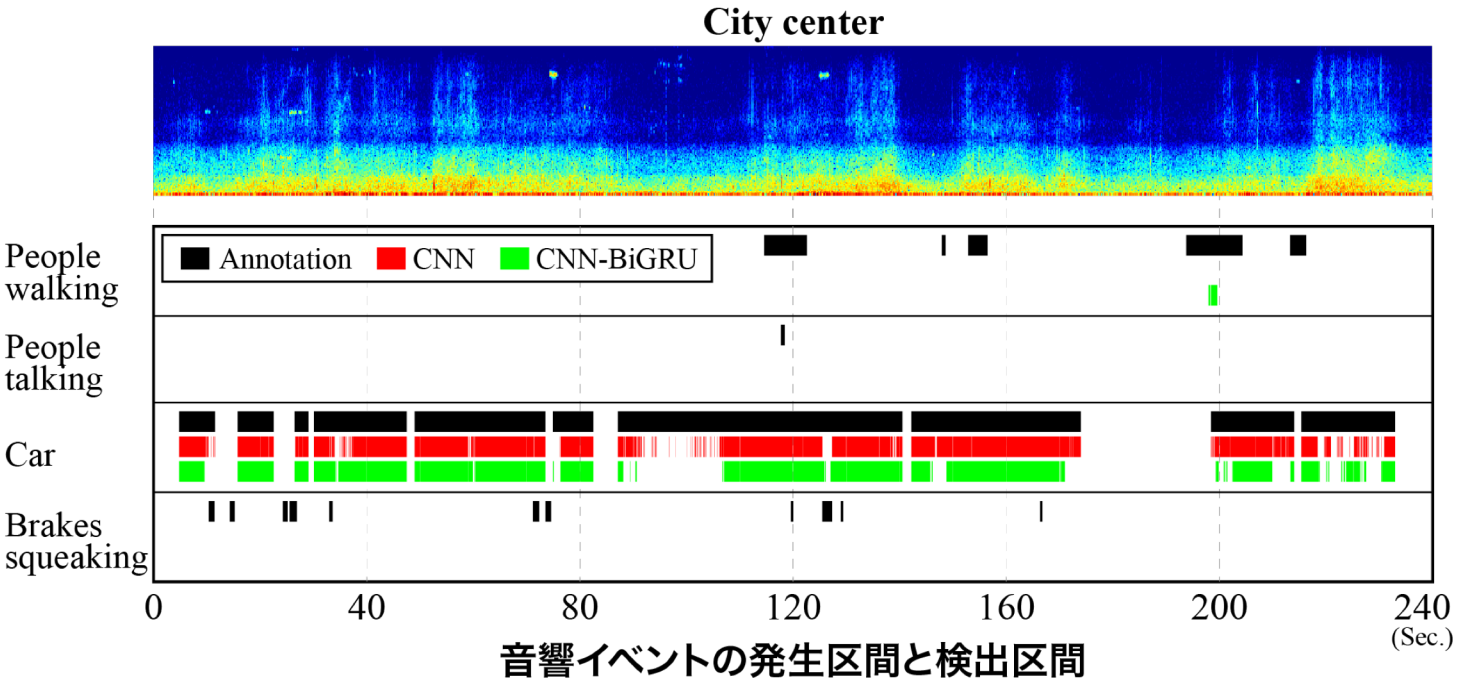
  - 分析する音響イベントの種類に比例した量の学習データが必要
    - AudioSetには527の音響イベントクラス
    - まだまだ不十分...

着眼点

音響イベントには共起性がある

- 例）carとbrakes squeakingは共起しやすい

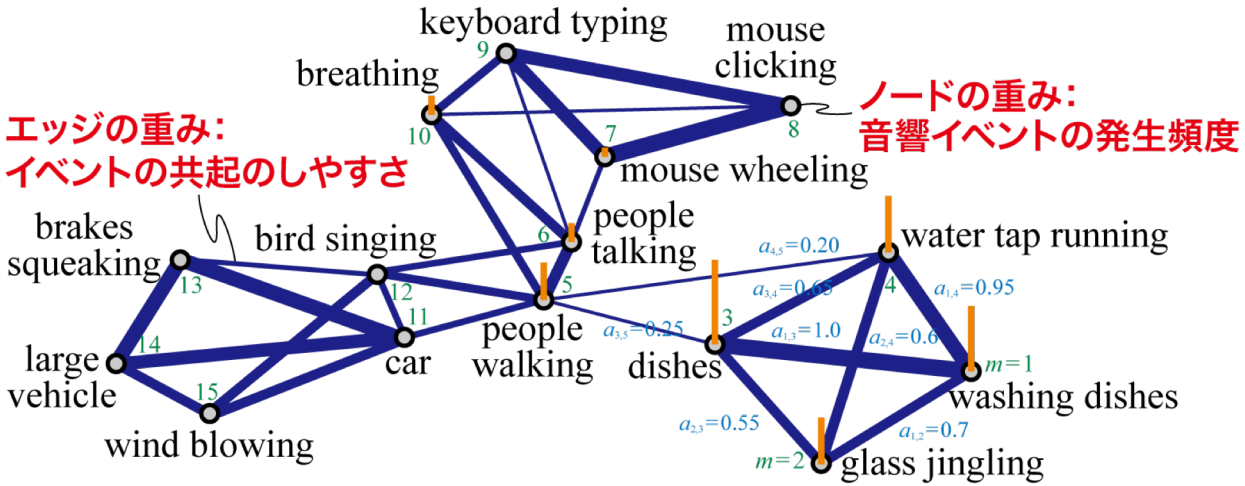
→ **イベントの共起をモデル学習に組み込めないか？**



音響イベントの発生区間と検出区間

イベント共起のグラフ表現

- 共起のしやすさと発生頻度をグラフで表現
  - 共起のしやすさ：**エッジの重み** $A_{i,j}$
  - 音響イベントの発生頻度：**ノードの重み** v_i
- グラフ構造を考慮してイベントをモデル化
  - グラフ構造に基づくラプラス正則化を利用



グラフラプラシアン正則化 (GLR)

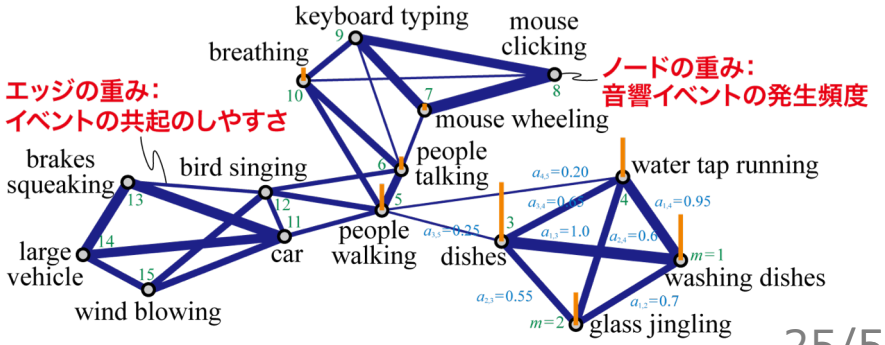
- グラフ構造を考慮した正則化を可能とする
  - **共起のしやすさ：エッジの重み** $A_{i,j}$
  - **音響イベントの発生頻度：ノードの重み** v_i
    - 次数行列 $\Delta_{i,i} = \sum_j A_{i,j}$ とする

$$\frac{1}{2}\sum_{i,j=0}^{M} A_{i,j}\|v_i - v_j\|^2 = \sum_{i=0}^{M} v_i v_i \Delta_{i,i} - \sum_{i,j=0}^{M} v_i v_j A_{i,j}$$

$$= \mathrm{Tr}(\mathbf{v}^\mathsf{T}\mathbf{\Delta}\mathbf{v}) - \mathrm{Tr}(\mathbf{v}^\mathsf{T}\mathbf{A}\mathbf{v})$$

$$= \mathrm{Tr}(\mathbf{v}^\mathsf{T}\mathbf{L}\mathbf{v})$$

共起しやすいイベントの
発生頻度が大きく異なると
大きなペナルティ



GLRを用いた音響イベント検出

- 深層学習の目的関数に正則化項を追加
  - 正則化重みをαとする

$$E(\Theta) = -\sum_{m=1}^{M}\sum_{n=1}^{N}\{z_{m,n}\log(y_{m,n}) + (1-z_{m,n})\log(1-y_{m,n})\} + \alpha \mathrm{Tr}(\mathbf{v}^{\mathsf{T}}\mathbf{L}\mathbf{v})$$

- 共起のしやすさ$\mathbf{L}(A_{i,j})$とイベント発生頻度$\mathbf{v}$
  - $\mathbf{L}(A_{i,j})$：音クリップ内で共起したイベントを数え上げ

$$A_{i,j} = \sum_{s=1}^{S} c_{i,j} / \max\left(\sum_{s=1}^{S} c_{i,j}\right) \text{ ただし } c_{i,j} = \begin{cases} 0 & \text{if 共起しない場合} \\ 1 & \text{else if 共起する場合} \end{cases}$$

  - $\mathbf{v}$：$\mathbf{v} = \sum_{n=1}^{N} \mathbf{y}_n$ で近似

$$E(\Theta) = -\sum_{m=1}^{M}\sum_{n=1}^{N}\{\mathbf{z}_n\log(\mathbf{y}_n) + (1-\mathbf{z}_n)\log(1-\mathbf{y}_n)\} + \alpha \mathrm{Tr}\{(\sum_{n=1}^{N}\mathbf{y}_n)^{\mathsf{T}}\mathbf{L}(\sum_{n=1}^{N}\mathbf{y}_n)\}$$

評価実験：実験条件

実験に用いたデータセット

- TUT sound events 2016/2017 dev. [Mesaros+ 2016, 2017]
  - 合計**192分**のデータ（学習データ，評価データ含む）
  - 4分割交差検証
- City center, Home, Office, Residential areaの4シーン
- Car, Brakes squeakingなど**25の音響イベント**
  - Officeのイベントラベルは自身らで付与

イベント検出手法

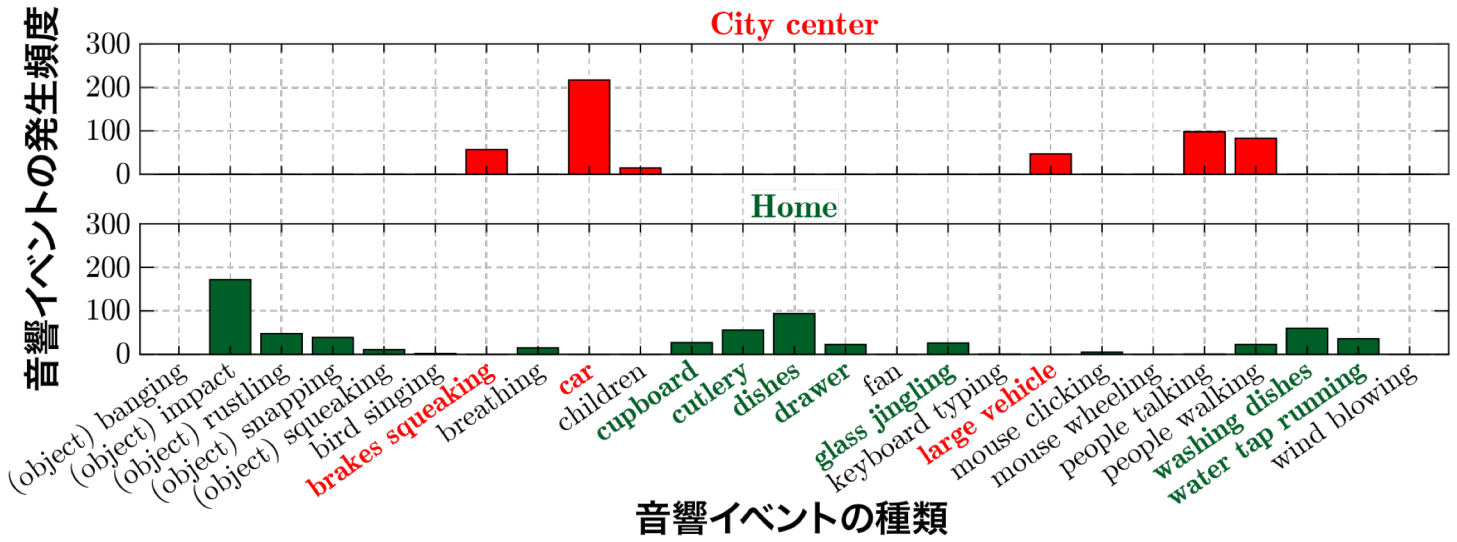
- CNN, CNN+BiGRU
- CNN+BiGRU (GRL)

により性能を評価

音響特徴量	Log mel-band energy
フレーム長/シフト長	40 ms / 20 ms
系列長	500 (10 s)
正規化重み α	1.0×10^{-5}
ネットワーク	3 conv. & 1 BiGRU 層
CNN 層のフィルタサイズ	3×3
プーリング	3×1 max pooling
活性化関数	ReLU
CNN 層のチャネル数	128, 128, 128
GRU 層のユニット数	32
最適化手法	Adam

評価実験：実験条件

- 実験に用いたデータセット
  - TUT Sound Events 2016/2017 [Mesaros+ 2016, 2017]
  - TUT Acoustic Scenes 2016 [Mesaros+ 2016, 2017]
    - 合計**192分**のデータ（学習データ，評価データ含む）
  - City center, Home, Office, Residential areaの4シーン
  - Car, Brakes squeakingなど**25の音響イベント**



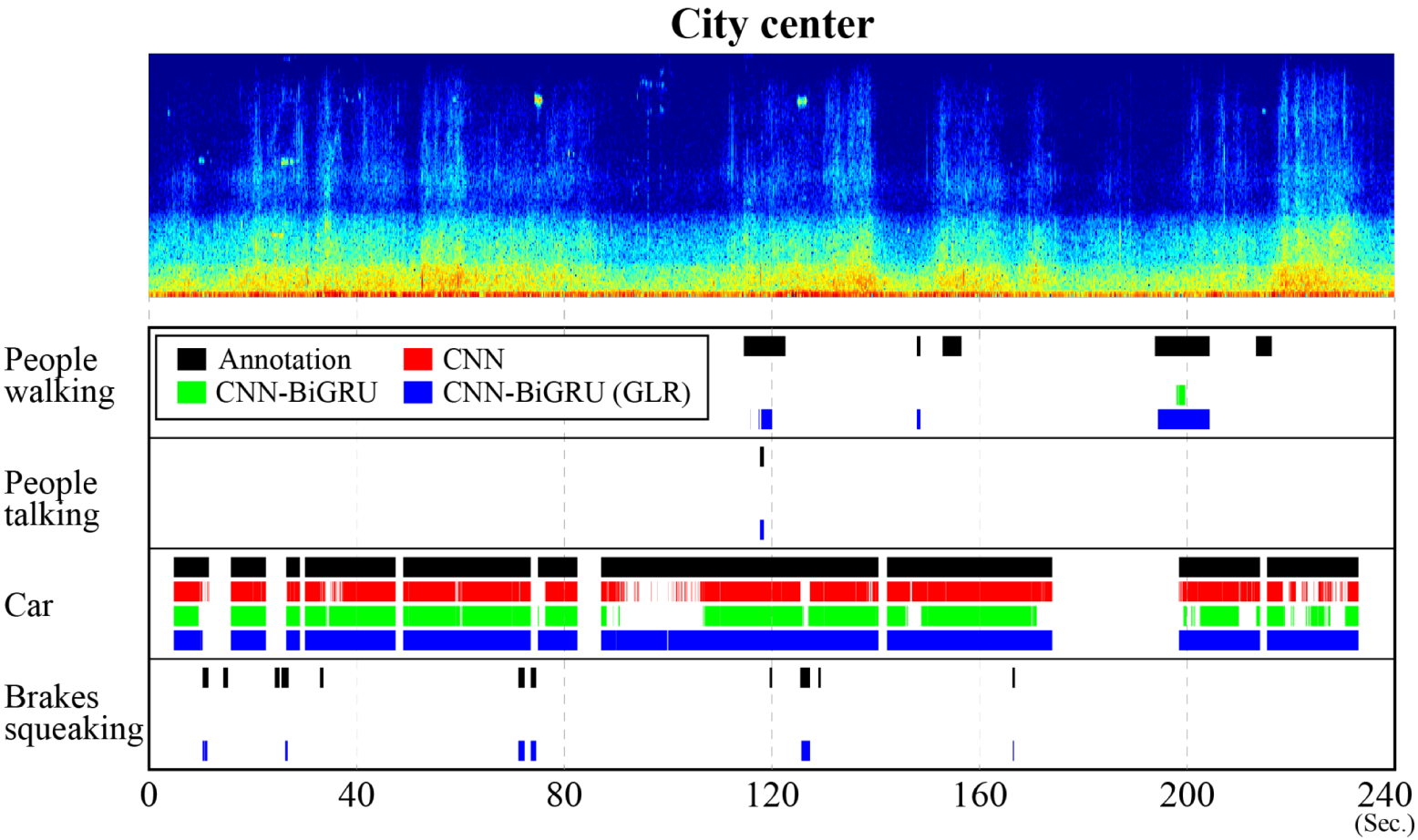
評価実験：イベント検出性能

- 評価指標
  - Frame-based micro F-score [Mesaros+ 2017]
  - Error rate
- イベント検出結果
  - F値：提案法はCNN+BiGRUよりも**7.9%ポイント**向上
  - 誤り率：CNN+BiGRUよりも**0.043**向上

Method	Average	
	F1 score	Error rate
CNN	34.17%	0.812
CNN-GRU	39.57%	0.782
CNN-BiGRU	41.24%	0.761
CNN-BiGRU w/ GLR	**49.16%**	**0.718**

評価実験：イベント検出結果例

- City centerの検出結果例
  - Brakes squeaking, people walkingの検出性能が向上



評価実験：イベント毎の検出性能

- 共起関係にある多くのイベントで性能向上
  - dishes, glass jingling, water tap running
- 残された課題
  - 依然として多くのイベントを検出できていない

Event	(object) banging	(object) impact	(object) rustling	(object) snapping	(object) squeaking	bird singing	brakes squeaking	person breathing	car	children	cupboard	cutlery
CNN	0.00%	0.03%	0.02%	0.00%	0.00%	46.78%	3.58%	0.00%	59.32%	0.00%	0.00%	0.00%
CNN-BiGRU	0.00%	0.00%	6.45%	0.00%	0.00%	**55.13%**	0.00%	0.00%	54.29%	0.00%	0.00%	0.00%
CNN-BiGRU w/ GLR	0.00%	**0.83%**	**16.81%**	0.00%	0.00%	39.54%	**6.20%**	0.00%	**60.63%**	0.00%	0.00%	0.00%

Event	dishes	drawer	fan	glass jinging	keyboard typing	large vehicle	mouse clicking	mouse wheeling	people talking	people walking	washing dishes	water tap running	wind blowing
CNN	0.21%	0.00%	36.39%	0.61%	**2.77%**	43.93%	**20.41%**	0.00%	0.00%	0.07%	20.01%	**54.95%**	0.06%
CNN-BiGRU	0.28%	0.00%	61.29%	0.00%	0.42%	43.22%	0.00%	0.00%	0.00%	11.39%	5.29%	33.91%	0.00%
CNN-BiGRU w/ GLR	**14.16%**	0.00%	**68.96%**	**2.53%**	1.09%	**49.66%**	0.00%	0.00%	**0.01%**	**48.88%**	**33.82%**	41.62%	**6.14%**

目次

- 環境音分析の背景と基礎（20分程度）
  - 研究背景，環境音分析の目的
  - 環境音分析の主な問題設定
    - 音響シーン分類，音響イベント検出，異常音検知
  - 環境音分析の課題
- ドメイン知識を活用した環境音分析（30分程度）
  - イベントの共起に着目した音響イベント検出
  - データ不均衡が環境音分析にもたらす影響の調査
- おわりに

音響イベント検出の課題

- イベントの種類が増えると性能が急激に低下
  - 感覚的に15～20を超えると分析が上手くいかなくなる
- DCASE2020 Challenge task4の例
  - 10種類の音響イベントを分析

DESED dataset

DESED dataset is the dataset that was used in DCASE 2019 task 4. The dataset for this task is composed of 10 sec audio clips recorded in domestic environment or synthesized using Scaper to simulate a domestic environment. The task focuses on 10 class of sound events that represent a subset of Audioset (not all the classes are present in Audioset, some classes of sound events are including several classes from Audioset):

- Speech `Speech`
- Dog `Dog`
- Cat `Cat`
- Alarm/bell/ringing `Alarm_bell_ringing`
- Dishes `Dishes`
- Frying `Frying`
- Blender `Blender`
- Running water `Running_water`
- Vacuum cleaner `Vacuum_cleaner`
- Electric shaver/toothbrush `Electric_shaver_toothbrush`

イベント検出性能と音の継続長

イベント検出性能と継続長は関連している？

音響イベント検出性能

Event	(object) banging	(object) impact	(object) rustling	(object) snapping	(object) squeaking	bird singing	brakes squeaking	person breathing	car	children	cupboard	cutlery
CNN	0.00%	0.03%	0.02%	0.00%	0.00%	46.78%	3.58%	0.00%	59.32%	0.00%	0.00%	0.00%
CNN-BiGRU	0.00%	0.00%	6.45%	0.00%	0.00%	**55.13%**	0.00%	0.00%	54.29%	0.00%	0.00%	0.00%
CNN-BiGRU w/ GLR	0.00%	**0.83%**	**16.81%**	0.00%	0.00%	39.54%	**6.20%**	0.00%	**60.63%**	0.00%	0.00%	0.00%

Event	dishes	drawer	fan	glass jinging	keyboard typing	large vehicle	mouse clicking	mouse wheeling	people talking	people walking	washing dishes	water tap running	wind blowing
CNN	0.21%	0.00%	36.39%	0.61%	**2.77%**	43.93%	**20.41%**	0.00%	0.00%	0.07%	20.01%	**54.95%**	0.06%
CNN-BiGRU	0.28%	0.00%	61.29%	0.00%	0.42%	43.22%	0.00%	0.00%	0.00%	11.39%	5.29%	33.91%	0.00%
CNN-BiGRU w/ GLR	**14.16%**	0.00%	**68.96%**	**2.53%**	1.09%	**49.66%**	0.00%	0.00%	**0.01%**	**48.88%**	**33.82%**	41.62%	**6.14%**

音響イベントの平均継続長

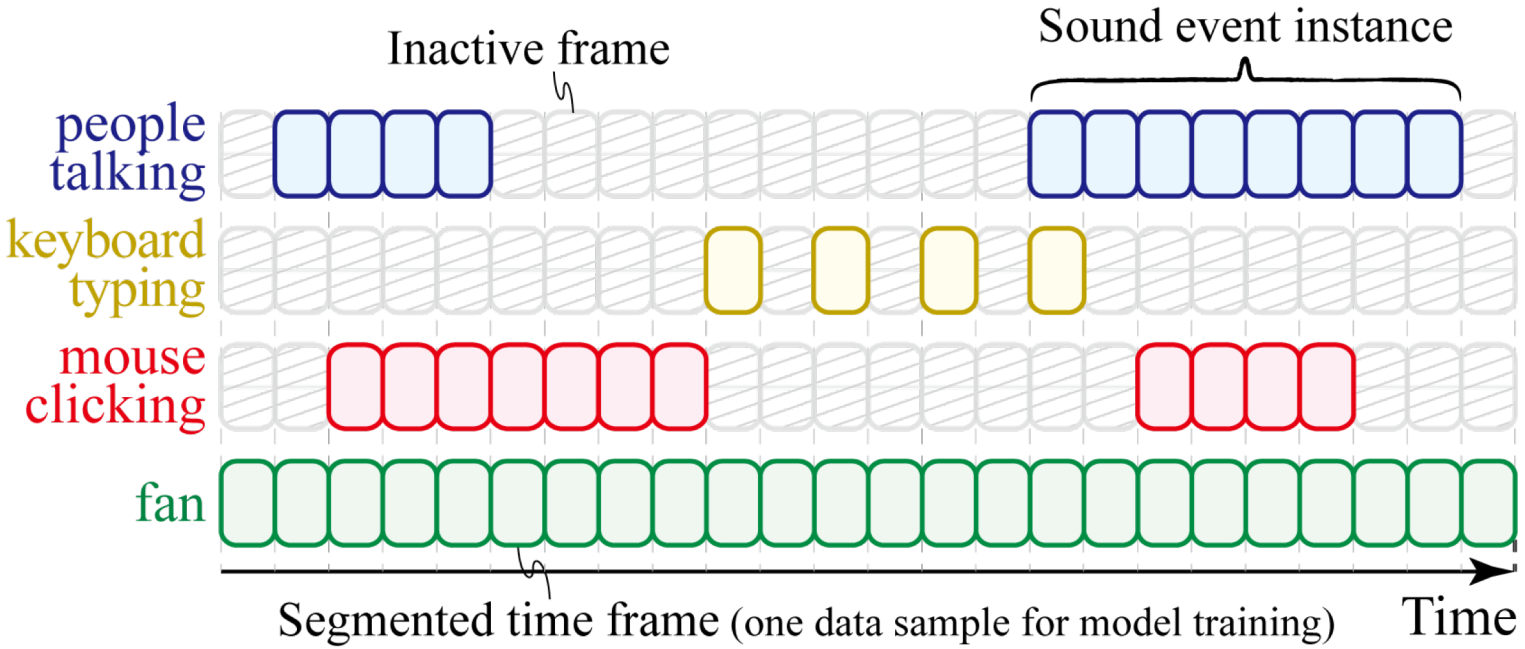
Sound event	Duration (s)
(object) banging	0.78
(object) impact	0.35
(object) rustling	2.24
(object) snapping	0.46
(object) squeaking	0.74
bird singing	7.63
brakes squeaking	1.65
breathing	0.43
car	6.88
children	6.87
cupboard	0.65
cutlery	0.74
dishes	1.24

Sound event	Duration (s)
drawer	0.80
fan	29.99
glass jingling	0.80
keyboard typing	0.21
large vehicle	14.68
mouse clicking	0.14
mouse wheeling	0.16
people talking	4.09
people walking	6.63
washing dishes	4.15
water tap running	5.92
wind blowing	6.09

音響イベントと時間フレーム

イベント継続長は時間フレーム数に影響

- 音響イベント検出では**１フレーム＝１つのデータ**
- ☹ **発生回数が同じ場合でもデータ不均衡が生じる**

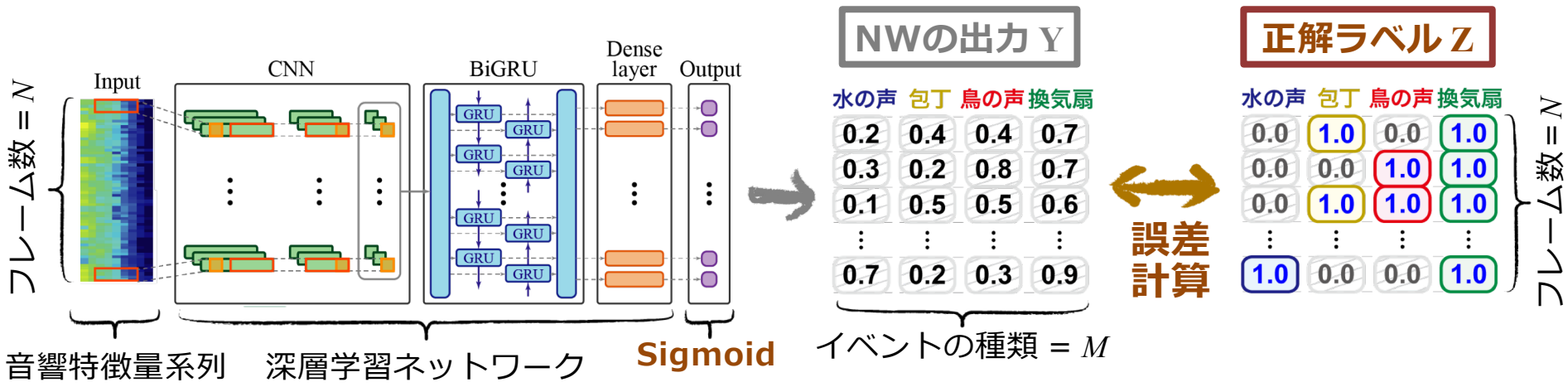


【再掲】イベント検出のモデル学習

モデルパラメータの学習方法

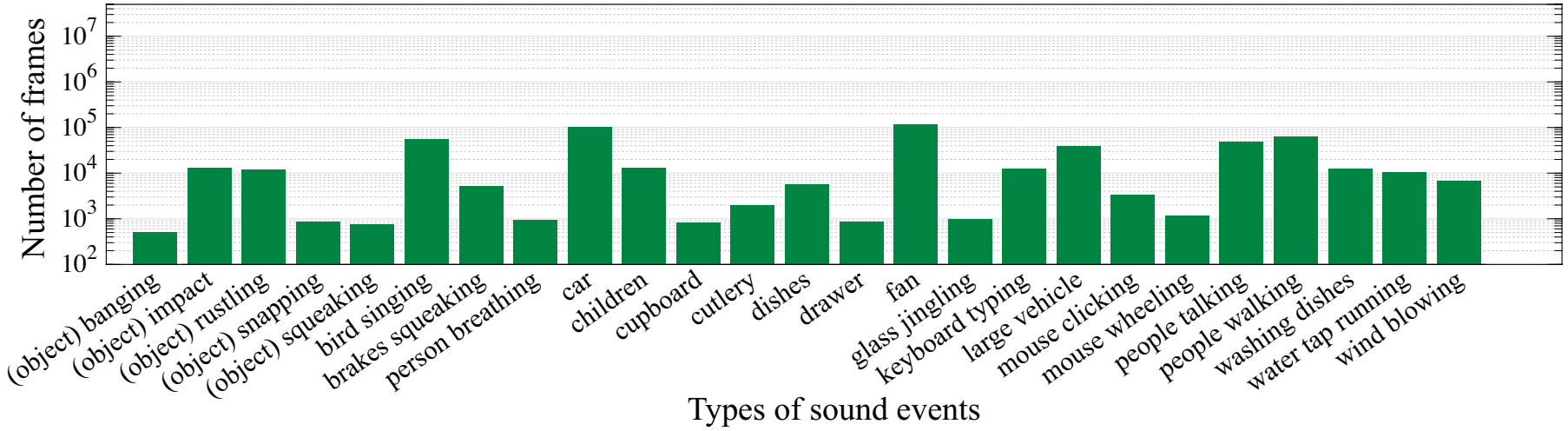
- 各フレーム/イベントの誤差 (Binary cross entropy) の総和が小さくなるようにモデル学習

$$\mathrm{E}_{\mathrm{BCE}}(\theta) = -\sum_{n=1}^{N}\sum_{m=1}^{M}\left\{z_{n,m}\log(y_{n,m}) + (1-z_{n,m})\log(1-y_{n,m})\right\}$$



音響イベント間のデータ不均衡

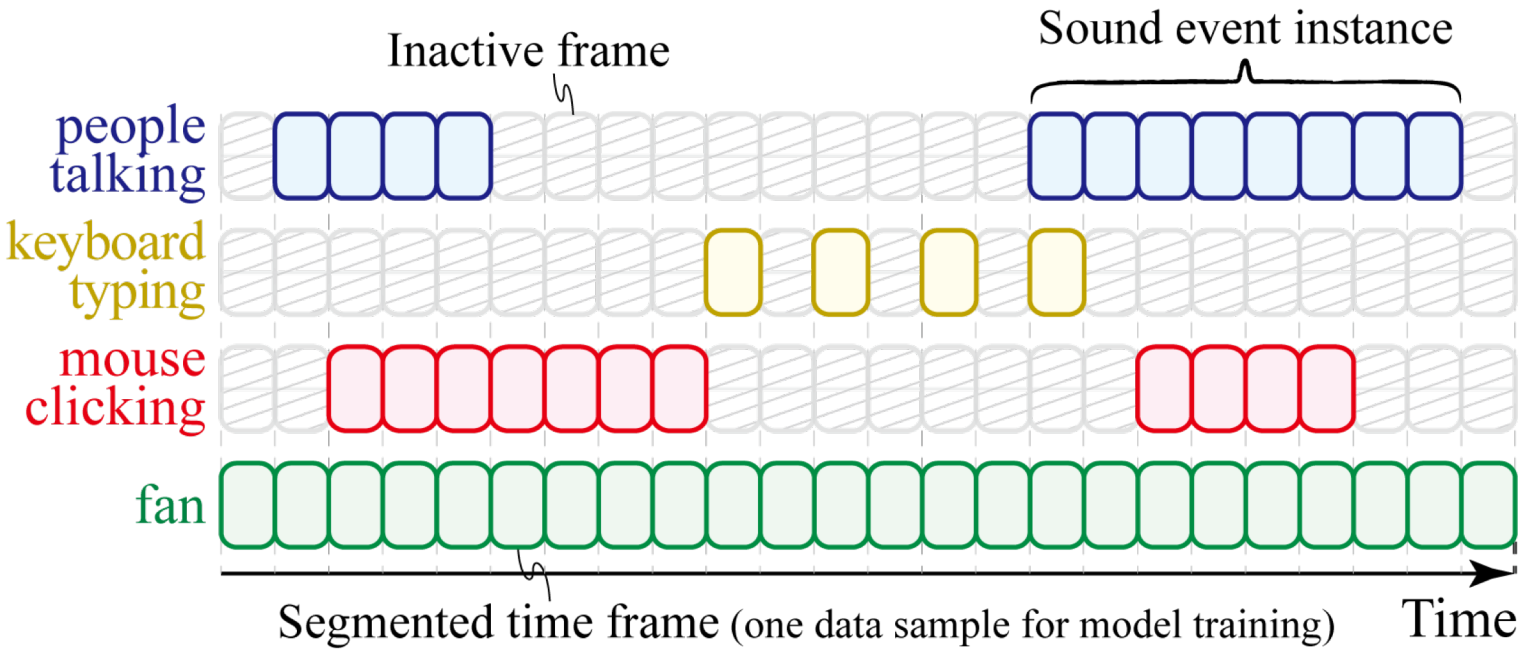
- 評価実験で用いたデータのイベント発生回数
  - TUT Sound Events 2016/2017 [Mesaros+ 2016, 2017]
  - TUT Acoustic Scenes 2016/2017 [Mesaros+ 2016, 2017]
  - ☹ **音響イベント間で最大10^2倍の差**



イベント検出における非活性区間

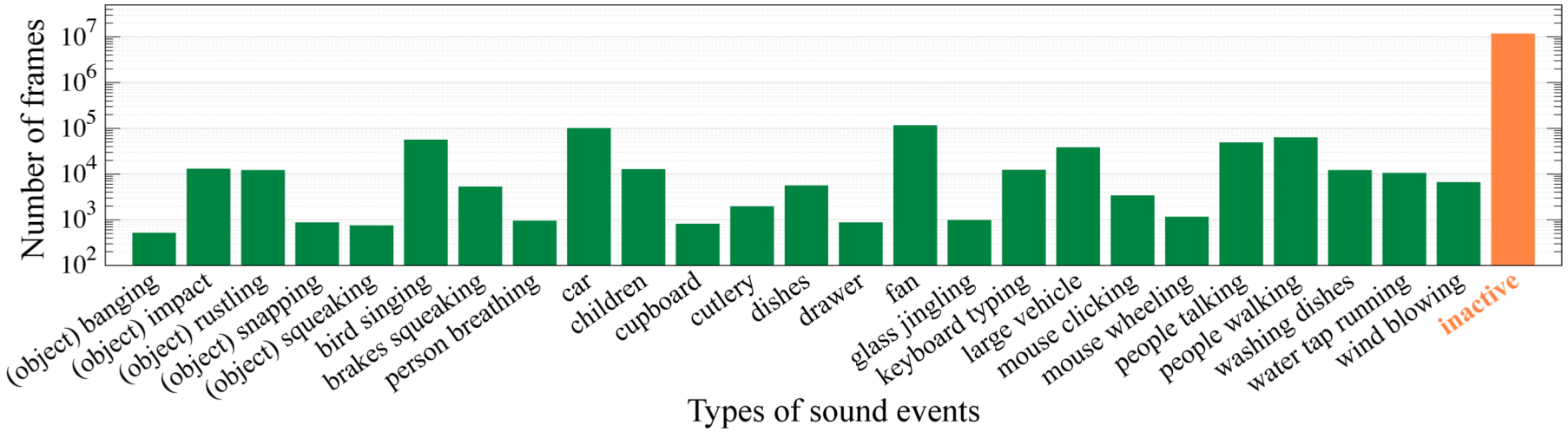
音響イベントの非活性区間

- 音響イベント検出では**非活性区間も１つのデータ**



非活性区間におけるデータ不均衡

- 音響イベント間の不均衡よりもさらに深刻
  - ☹ **音響イベント間で最大 10^4 倍の差**
  - ☹ **音響イベントの種類が増える毎に差は大きくなる**



評価関数による不均衡の調整

- Simple reweighting loss (SRL)
  - 活性/非活性フレームそれぞれをα, βで重み付け
    - 今回は$\alpha = 1.0$

$$\mathrm{E}_{\mathrm{SRL}}(\theta) = -\sum_{n=1}^{N}\sum_{m=1}^{M}\left\{\alpha z_{n,m}\log(y_{n,m}) + \beta(1-z_{n,m})\log(1-y_{n,m})\right\}$$

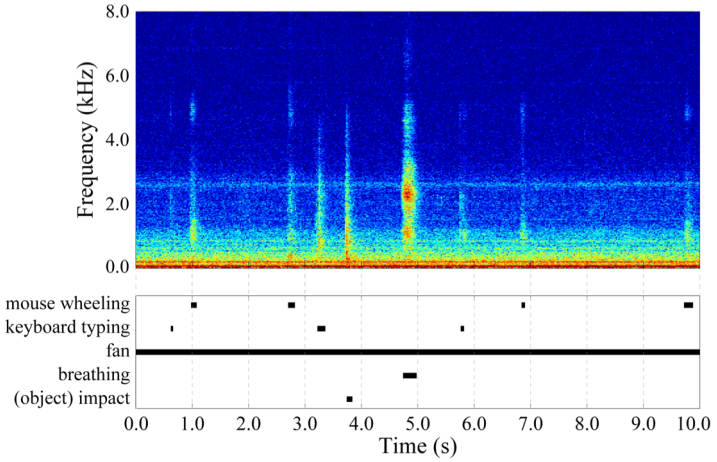
- Inverse frequency loss (IFL)
  - 音響イベントの発生フレーム数の逆数で重み付け

$$\mathrm{E}_{\mathrm{IFL}}(\theta) = -\sum_{n=1}^{N}\sum_{m=1}^{M}\left\{\left(\frac{C}{N_m + C}\right)^{\gamma} z_{n,m}\log(y_{n,m}) + (1-z_{n,m})\log(1-y_{n,m})\right\}$$

Asymmetric focal loss (AFL)

着眼点

- 継続時間が長い音や非活性区間はパターンが少なく学習が容易
- 学習の進み具合に応じて誤差を動的にscaling
  - focal lossの導入



Asymmetric focal loss (AFL)

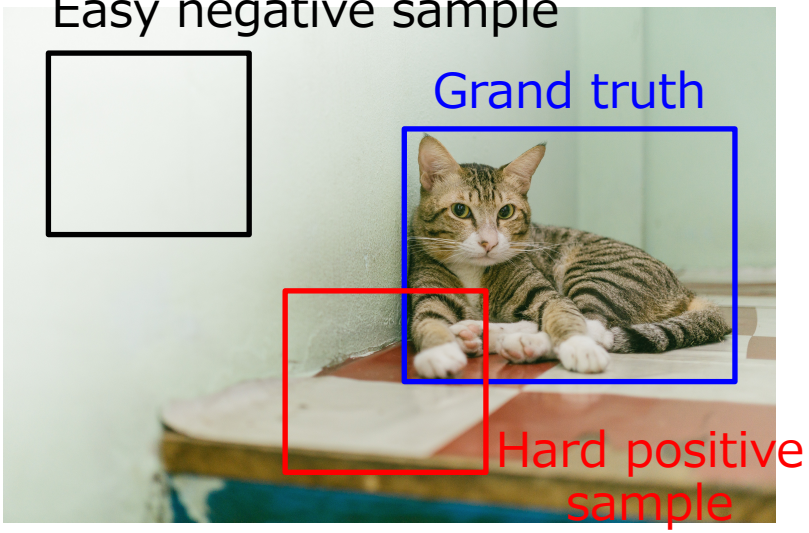
- 活性/非活性区間を別々にscaling

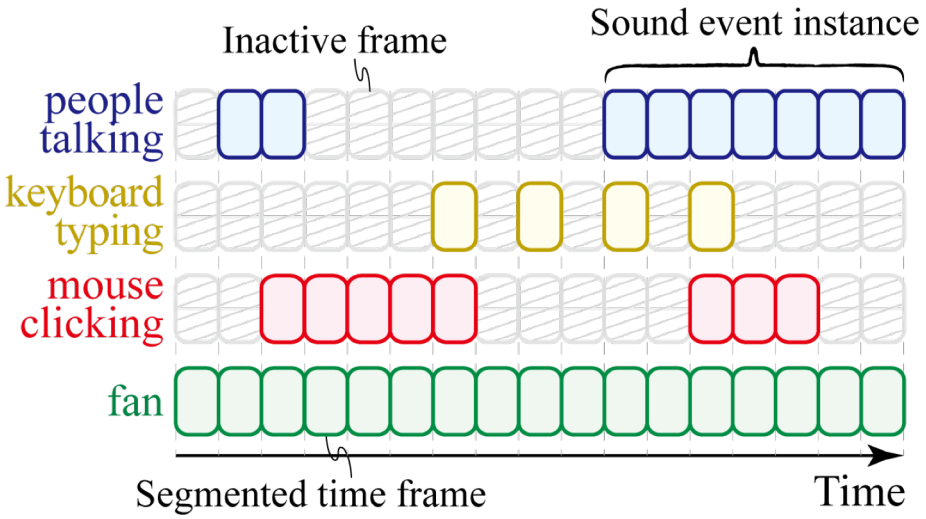
$$\mathrm{E}_{\mathrm{AFL}}(\theta) = -\sum_{n=1}^{N}\sum_{m=1}^{M}\Big\{(1-y_{n,m})^{\gamma} z_{n,m}\log(y_{n,m}) + (y_{n,m})^{\zeta}(1-z_{n,m})\log(1-y_{n,m})\Big\}$$

余談

物体検出と音響イベント検出の類似点

- 各タスクの課題
  - 物体検出：（検出対象でない）**背景領域のサンプルが多い**
  - イベント検出：**非活性区間のサンプルが多い**
- 背景領域/非活性区間の特徴
  - **パターンが少なく学習が容易**（Easy negative sample）





Dice coefficient

着眼点

- True negativeを追い求めなければ良いのでは？
- Dice coefficient (DSC, F-scoreとも呼ばれる) に着目

$$
\begin{aligned}
\mathrm{DSC} &= \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FN} + \mathrm{FP}} \\
&= \frac{\sum 2 y_{n,m} z_{n,m}}{\sum 2 y_{n,m} z_{n,m} + (1 - y_{n,m}) z_{n,m} + y_{n,m}(1 - z_{n,m})} \\
&= \frac{\sum 2 y_{n,m} z_{n,m}}{\sum y_{n,m} + z_{n,m}}
\end{aligned}
$$

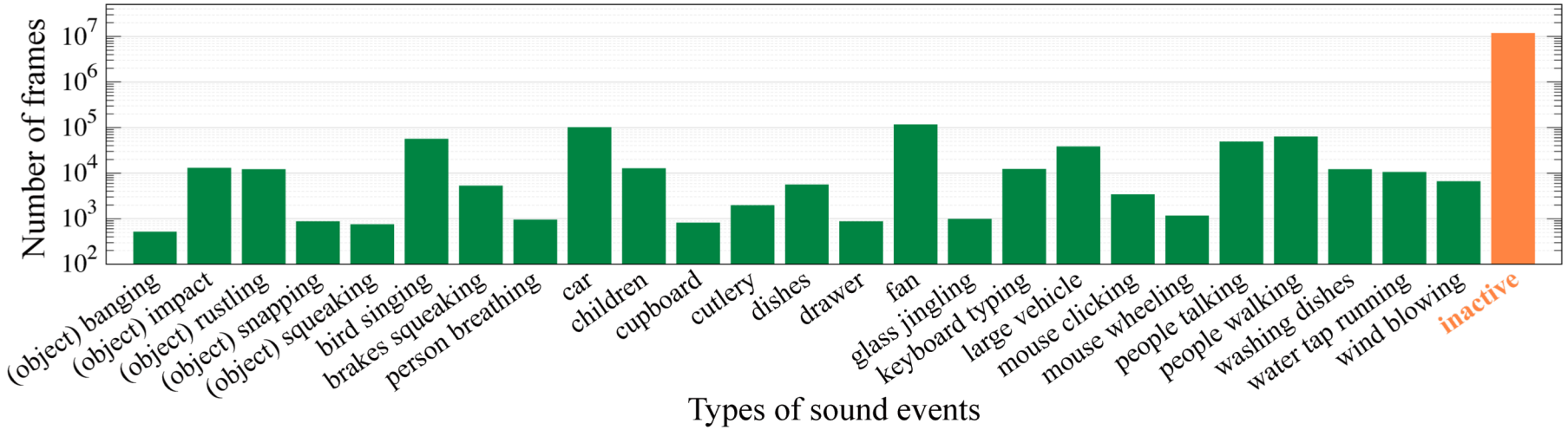
Focal batch Tversky loss (FBTL)

- DSCに以下の工夫を加える
  - Focal lossの考え方を導入
  - FPとFNのバランスを調整する係数を導入 (Tversky index)
- Focal batch Tversky loss (FBTL)

$$\mathrm{E}_{\mathrm{FBTL}}(\theta) = 1 - \frac{\sum_{n,m=1}^{N,M} (1 - y_{n,m})^{\gamma} y_{n,m} z_{n,m} + \eta}{\sum_{n,m=1}^{N,M} \alpha (1 - y_{n,m})^{\gamma} y_{n,m} + \sum_{n,m=1}^{N,M} \beta z_{n,m} + \eta}$$

評価実験：実験条件

- 実験に用いたデータセット
  - TUT sound events 2016/2017 [Mesaros+ 2016, 2017]
  - TUT acoustic scenes 2016/2017 [Mesaros+ 2016, 2017]
    - 合計266分（学習192分，評価74分）のデータ
  - Car, Brakes squeakingなど**25の音響イベント**



評価実験：実験条件

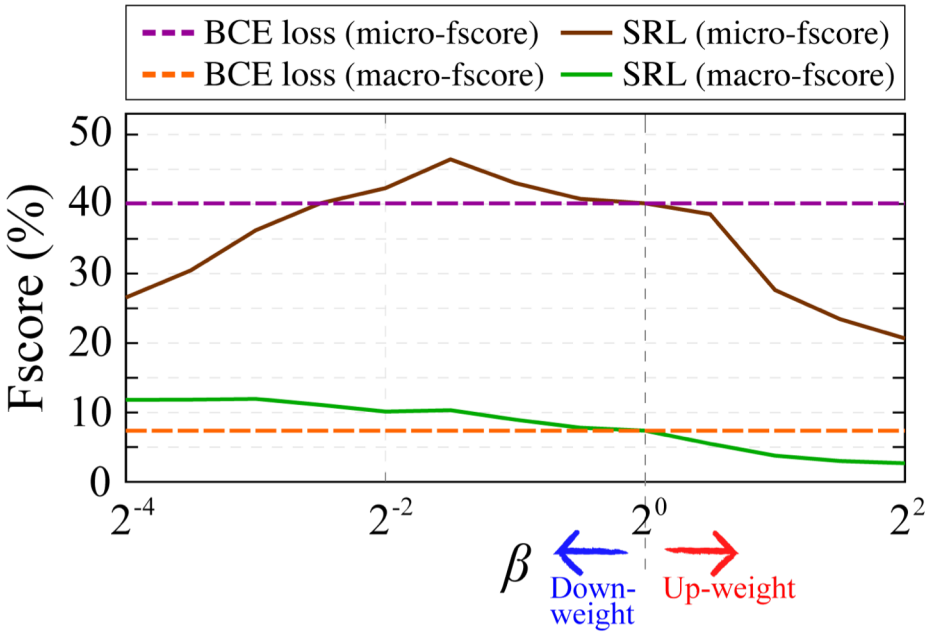
- 実験に用いたデータセット
  - TUT sound events 2016/2017 [Mesaros+ 2016, 2017]
  - TUT acoustic scenes 2016/2017 [Mesaros+ 2016, 2017]
    - 合計266分（学習192分，評価74分）のデータ
  - Car, Brakes squeakingなど**25の音響イベント**
- 学習モデル
  - CNN + bidirectional GRU
  - Transformer
  - など

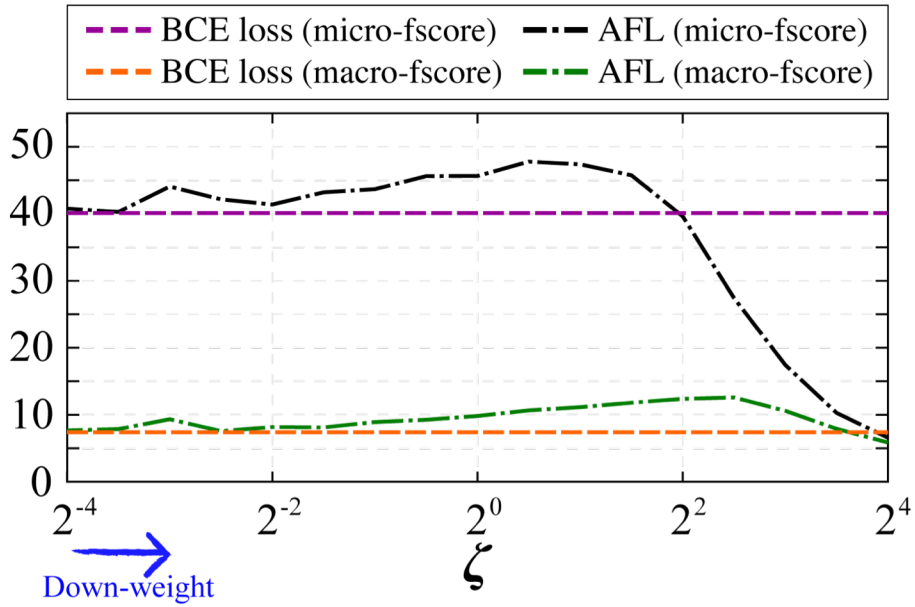
Length of sound clip	10 s
Network for CNN-BiGRU	3 CNN + 1 BiGRU + 1 dense layer
# channels of CNN layers	128, 128, 128
Filter size	3×3, 3×3, 3×3
Pooling size	1×8, 1×4, 1×2 (max pooling)
# units in GRU layer	32
# units in fully conn. layer	32
Network for Transformer	3 CNN + 2 Transformer encoder layers + 2 dense layers
# attention heads	32
Activation function	Leaky ReLU
Optimizer	RAdam [24]
Detection threshold	0.5
Constant number C	500
Smoothing parameter η	1.0

評価実験：非活性区間の影響

- 非活性区間の誤差の重みを調整した場合
  - Simple reweighting loss $\mathrm{E}_{\mathrm{SRL}}(\theta) = -\sum_{n,m=1}^{N,M} \left\{ z_{n,m} \log(y_{n,m}) + \beta(1 - z_{n,m}) \log(1 - y_{n,m}) \right\}$
  - Asymmetric focal loss $\mathrm{E}_{\mathrm{AFL}}(\theta) = -\sum_{n,m=1}^{N,M} \left\{ z_{n,m} \log(y_{n,m}) + (y_{n,m})^{\zeta}(1 - z_{n,m}) \log(1 - y_{n,m}) \right\}$

  ともにベースラインから**6%ポイント以上**の性能向上



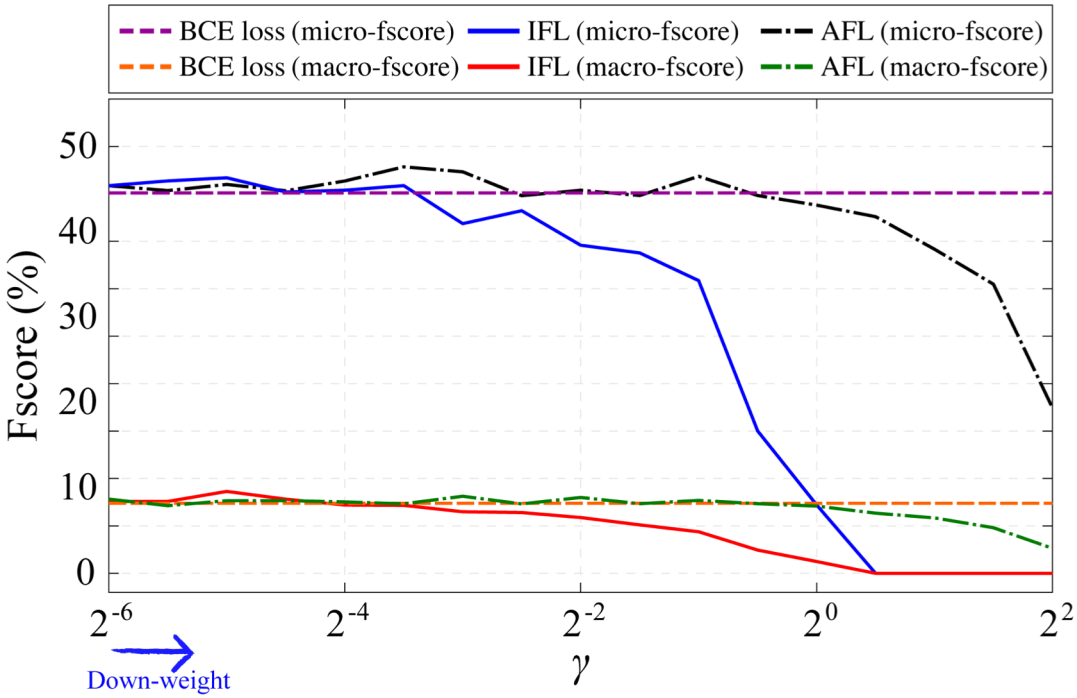


評価実験：イベント間不均衡の影響

音響イベント間の誤差の重みを調整した場合

- Inverse frequency loss $\mathrm{E}_{\mathrm{IFL}}(\theta) = -\sum_{n,m=1}^{N,M} \left\{ \left(\frac{C}{N_m + C} \right)^{\gamma} z_{n,m} \log(y_{n,m}) + (1 - z_{n,m}) \log(1 - y_{n,m}) \right\}$
- Asymmetric focal loss $\mathrm{E}_{\mathrm{AFL}}(\theta) = -\sum_{n,m=1}^{N,M} \left\{ (1 - y_{n,m})^{\gamma} z_{n,m} \log(y_{n,m}) + (1 - z_{n,m}) \log(1 - y_{n,m}) \right\}$

ともにベースラインから**大きな性能向上は見られず**



評価実験：両者の不均衡を調整

- 両者の重みを調整するとより性能が向上
  - CNN-BiGRU以外のNWでも効果を確認
  - ROC-AUCでも効果を確認

Method	Micro-Fscore	Macro-Fscore	Micro-ROC AUC	Macro-ROC AUC
[Conventional methods]				
CNN-BiGRU w/ BCE loss (**Baseline**)	40.10%	7.39%	89.15%	65.85%
CNN-BiGRU w/ α min-max subsampling & BCE loss	44.12%	9.35%	90.27%	67.55%
CNN-BiGRU w/ batch dice loss	45.06%	9.79%	86.99%	63.89%
MTL of SED & SAD w/ BCE loss	43.35%	8.64%	91.40%	70.97%
Transformer w/ BCE loss	45.15%	9.27%	90.32%	66.64%
[Loss reweighting between active and inactive frames]				
CNN-BiGRU w/ simple reweighting loss ($\beta = 0.3535$)	46.44%	10.34%	91.07%	69.31%
CNN-BiGRU w/ asymmetric focal loss ($\gamma = 0.0$, $\zeta = 1.414$)	47.78%	10.65%	92.35%	76.18%
CNN-BiGRU w/ focal batch Tversky loss ($\alpha = 0.6$, $\beta = 0.4$, $\gamma = 0.001$)	46.97%	10.28%	87.95%	65.08%
[Loss reweighting between sound event classes]				
CNN-BiGRU w/ inverse frequency loss ($C = 500$)	41.89%	7.57%	89.89%	66.46%
CNN-BiGRU w/ asymmetric focal loss ($\gamma = 0.125$, $\zeta = 0.0$)	42.33%	8.13%	91.08%	70.46%
[Loss reweighting both between event classes and between active/inactive frames]				
CNN-BiGRU w/ asymmetric focal loss ($\gamma = 0.0625$, $\zeta = 1.0$)	48.29%	10.46%	92.62%	77.03%
Transformer w/ asymmetric focal loss ($\gamma = 0.0625$, $\zeta = 1.0$)	**49.14%**	**11.11%**	**92.74%**	**77.49%**

おわりに

- 環境音分析の魅力
  - 応用先が非常に豊富
    - 見守り，異常検知，自動運転，障がい者支援…
  - 未解決の課題が多い
  - にも関わらず研究者は少ない
- 環境音ならではの工夫は十分可能
  - ５年後も生き残る技術を目指せる