Mobile and Sensor Systems

Lecture 5: Modeling and Inference

Sourav Bhattacharya

UNIVERSITY OF CAMBRIDGE

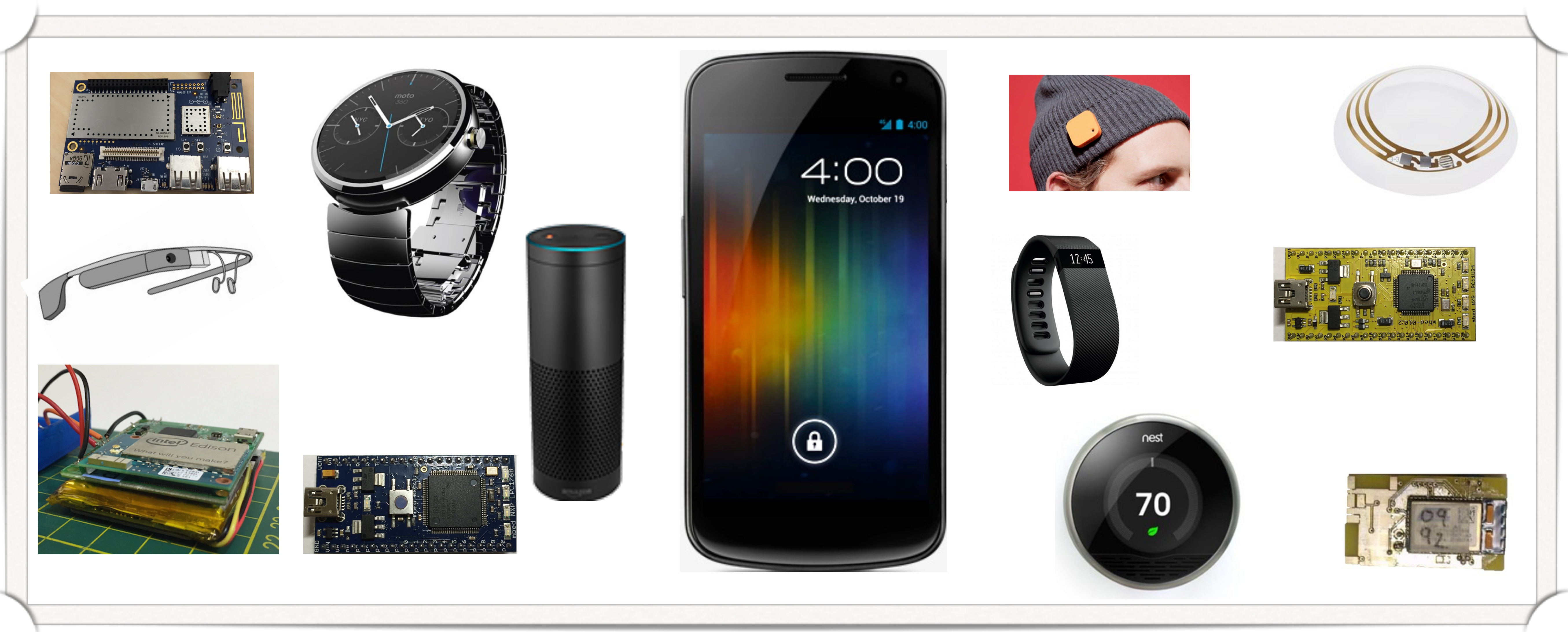
Lecture Overview

- **Part - 1**
  - Introduction to mobile and wearable sensing
  - Mobile sensing applications
  - Understanding the key tasks in mobile sensing
  - Challenges in mobile sensing
- **Part - 2**
  - Modeling audio using Deep Neural Networks
  - Multi-task learning through shared architecture
  - Open research questions

UNIVERSITY OF CAMBRIDGE

Part - 1

Mobile and Wearable Sensing



The mobile phone and wearable sensing domain is filled with **hacks**, and imaginative techniques that are used to circumvent the limitations of a platform that was **designed for a different purpose.**

Mobile / Wearable Sensing Vs. Sensor Networks

Mobile Sensing

- Well suited for human activities
- General purpose sensors, often not well suited for accurate sensing of the target phenomena
- Multi-tasking OS. Main purpose is to support various applications
- Low cost of deployment and maintenance (millions of users charge their devices)

Sensor Networks

- Well suited for sensing the environment
- Specialized sensors, designed to accurately monitor specific phenomena
- All resources dedicated to sensing
- High cost deployment and maintenance (regular charging thousands of sensor nodes)

Mobile Sensing Applications

Individual sensing:

- fitness applications
- behaviour intervention applications

Group/community sensing:

- sense common activities and help achieving group goals
- examples: assessment of neighbourhood safety, environmental sensing, collective recycling efforts

Urban-scale sensing:

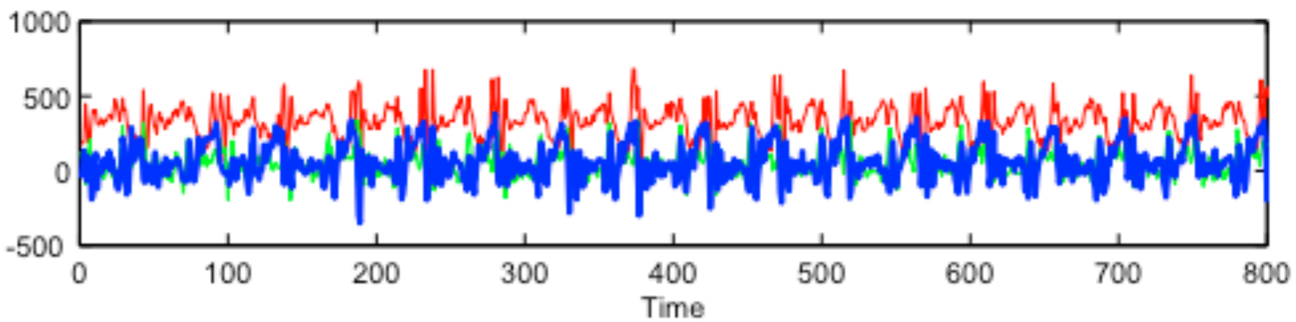
- large scale sensing, where large number of people have the same application installed
- examples: tracking speed of disease across a city, congestion and pollution in a city

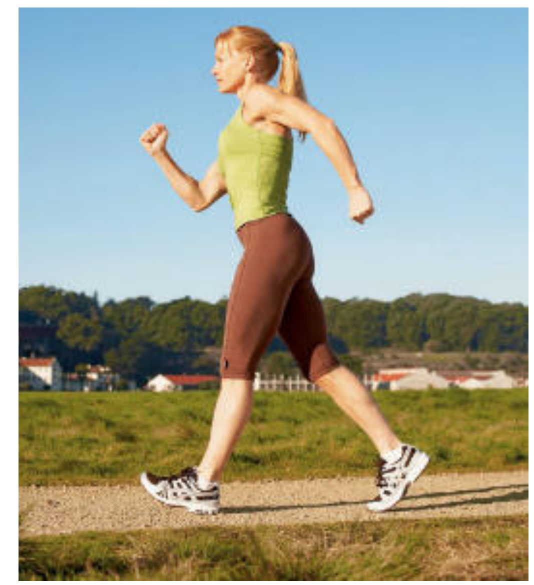
UNIVERSITY OF CAMBRIDGE

Human Activity Recognition

Sensor used:

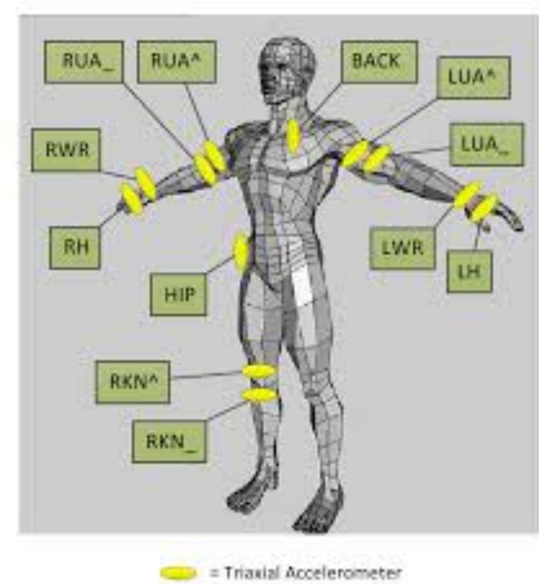
- Accelerometer or Gyroscope





Example inference:

- Walking, running, biking, up/down stairs etc.

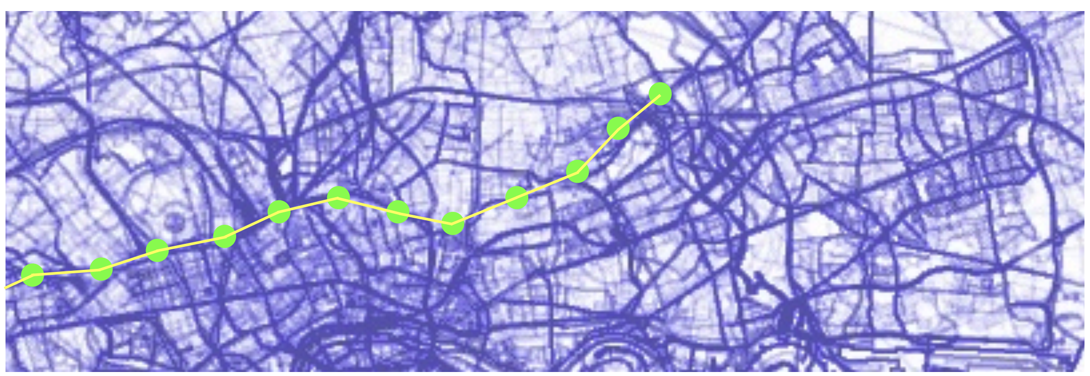


Applications:

- Health / behaviour intervention
- Fitness monitoring
- Sharing within a community

Transportation-mode Detection

Sensor used:
- Accelerometer or Gyroscope
- GPS, WiFi localization



Example inference:
- Bus, bike, tram, train, car etc.

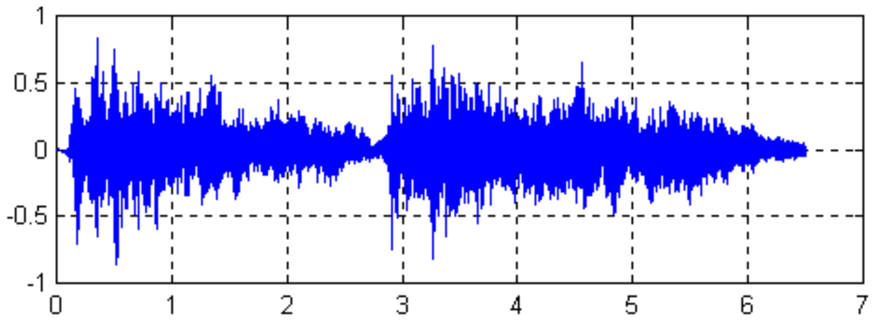
Applications:
- Intelligent transportation
- Smart commuting

UNIVERSITY OF CAMBRIDGE

Emotion Detection

Sensor used:

- Microphone, bluetooth
- GPS, WiFi localization
- Map speaking features to emotional state



Example inference:

- Emotional state, location and co-location with others

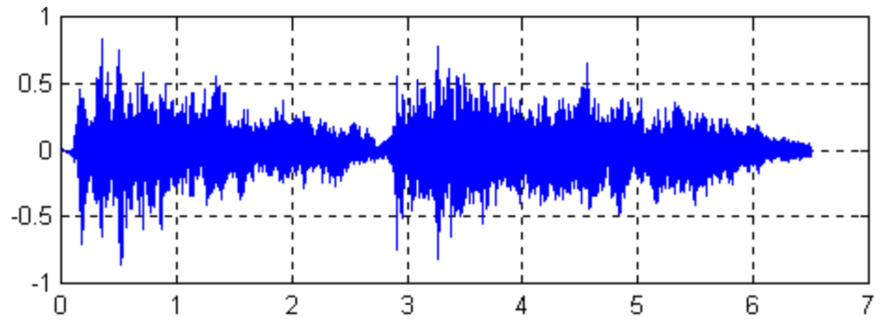
Applications:

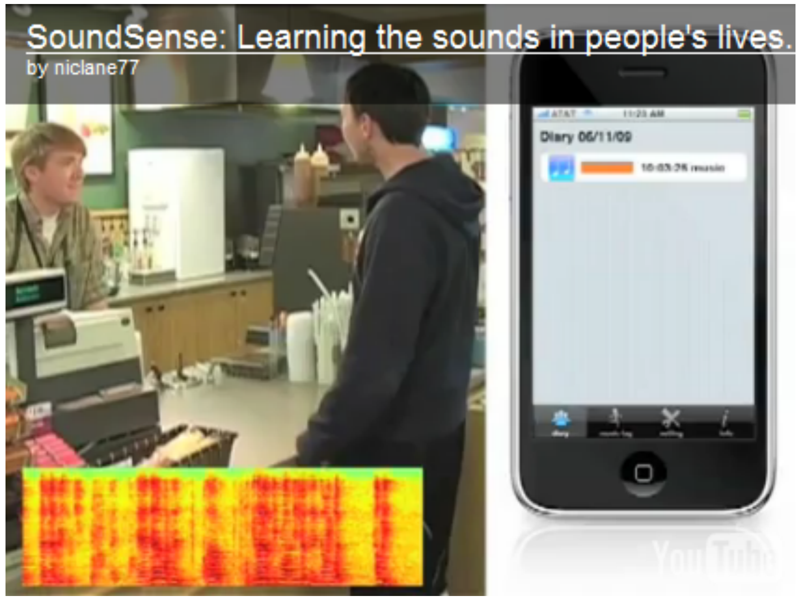
- Behaviour intervention
- Computational social science
  - Using mobile sensing for quantifying theories in social science

UNIVERSITY OF CAMBRIDGE

Context and Environment

Sensor used:
- Microphone
- Camera





Example inference:
- Conversation, music, party, activity-related sound etc.

Applications:
- Automated diary
- Health and wellness

UNIVERSITY OF CAMBRIDGE

Challenges in Mobile Sensing

- Complex natural environment
- Heterogeneity of sensors
  - Vary in sampling frequency, sensitivity
- Noisy measurements
- Different sensor position and orientation
- Diverse population
- Privacy
- Limited processing and battery power

Common sensing platforms

Noisy data

Diverse user population

UNIVERSITY OF CAMBRIDGE

Challenges in Mobile Sensing

- **Sensing is resource intensive**


Battery


Memory


CPU

GPU


Storage

- **The purpose of the embedded platform is to support multiple applications**

- **A sensing application needs to maintain a balance between**
  - The amount of resource needed to operate
  - The accuracy of the detection that is achieved

UNIVERSITY OF CAMBRIDGE

Context Recognition: Machine Learning

Supervised Learning:
- Labeled data (training data)
- Objective: Learn a function from training data

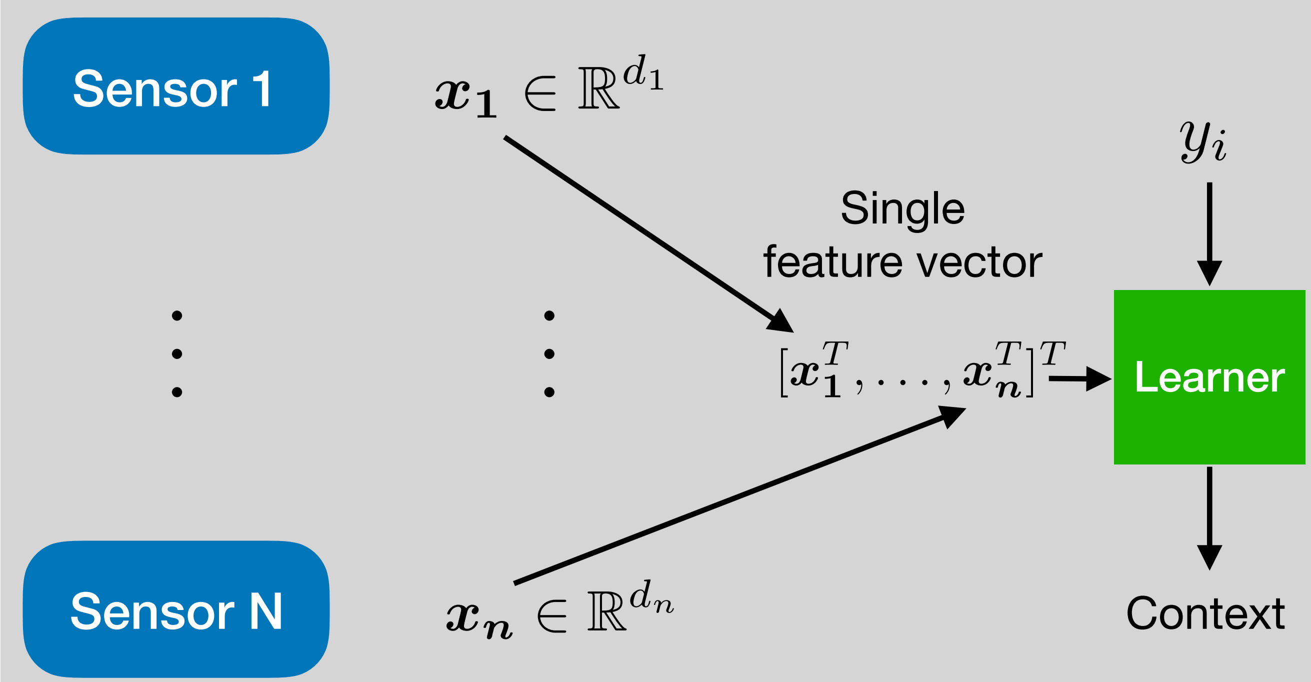
$$\mathcal{F} : \mathbf{X} \rightarrow \mathbf{Y} \qquad \boldsymbol{x_i} \in \mathbb{R}^d$$

Feature vector	Label
$\boldsymbol{x}_1$	y_1
$\boldsymbol{x}_2$	y_2
$\vdots$	$\vdots$
$\boldsymbol{x}_n$	y_n

Classification
- Label is discrete / categorical variable

Regression
- Label is real-valued / continuous variable

In mobile sensing we have a large number of sensors



UNIVERSITY OF CAMBRIDGE

Context Recognition: Machine Learning

Feature vector	Label
$\boldsymbol{x}_1$	y_1
$\boldsymbol{x}_2$	y_2
$\vdots$	$\vdots$
$\boldsymbol{x}_n$	y_n

Supervised Learning:

- Labeled data (training data)
- Objective: Learn a function from training data

$$\mathcal{F} : \mathbf{X} \rightarrow \mathbf{Y} \qquad \boldsymbol{x_i} \in \mathbb{R}^d$$

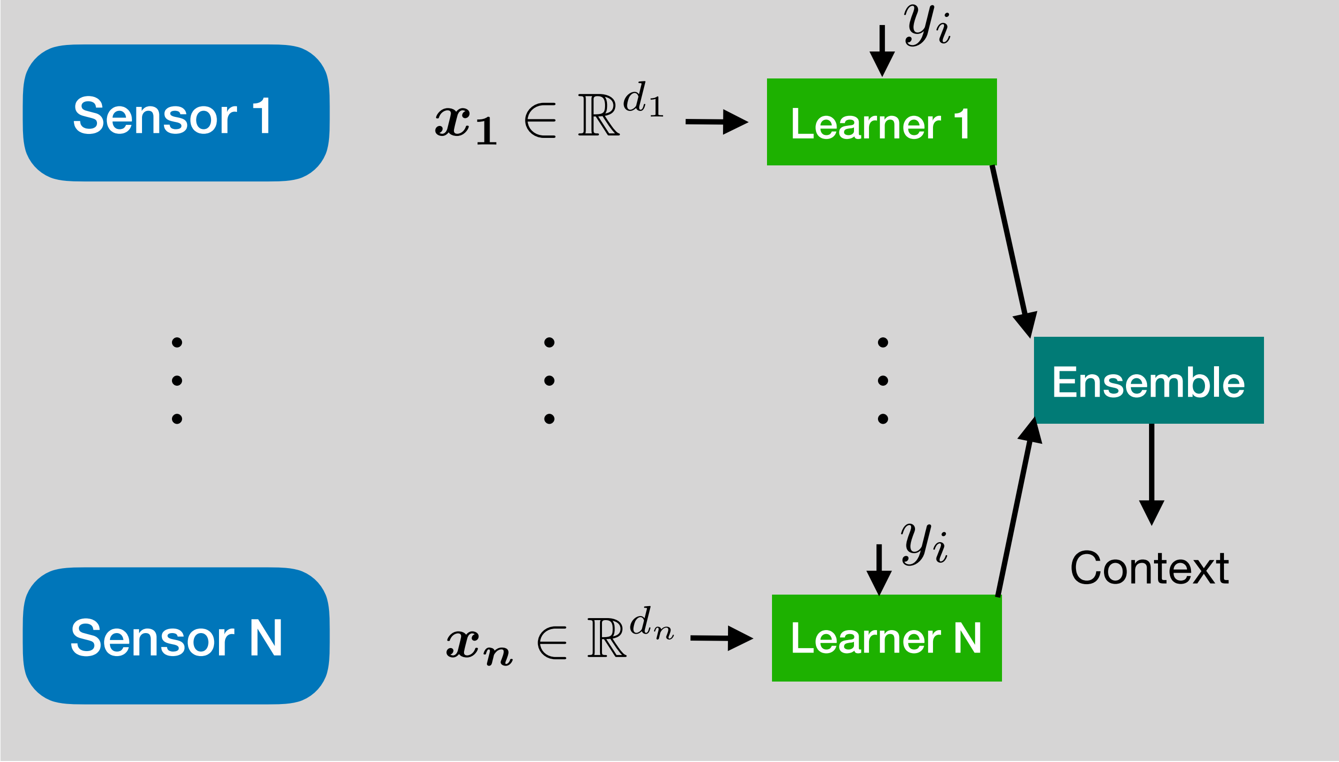
Classification

- Label is discrete / categorical variable

Regression

- Label is real-valued / continuous variable

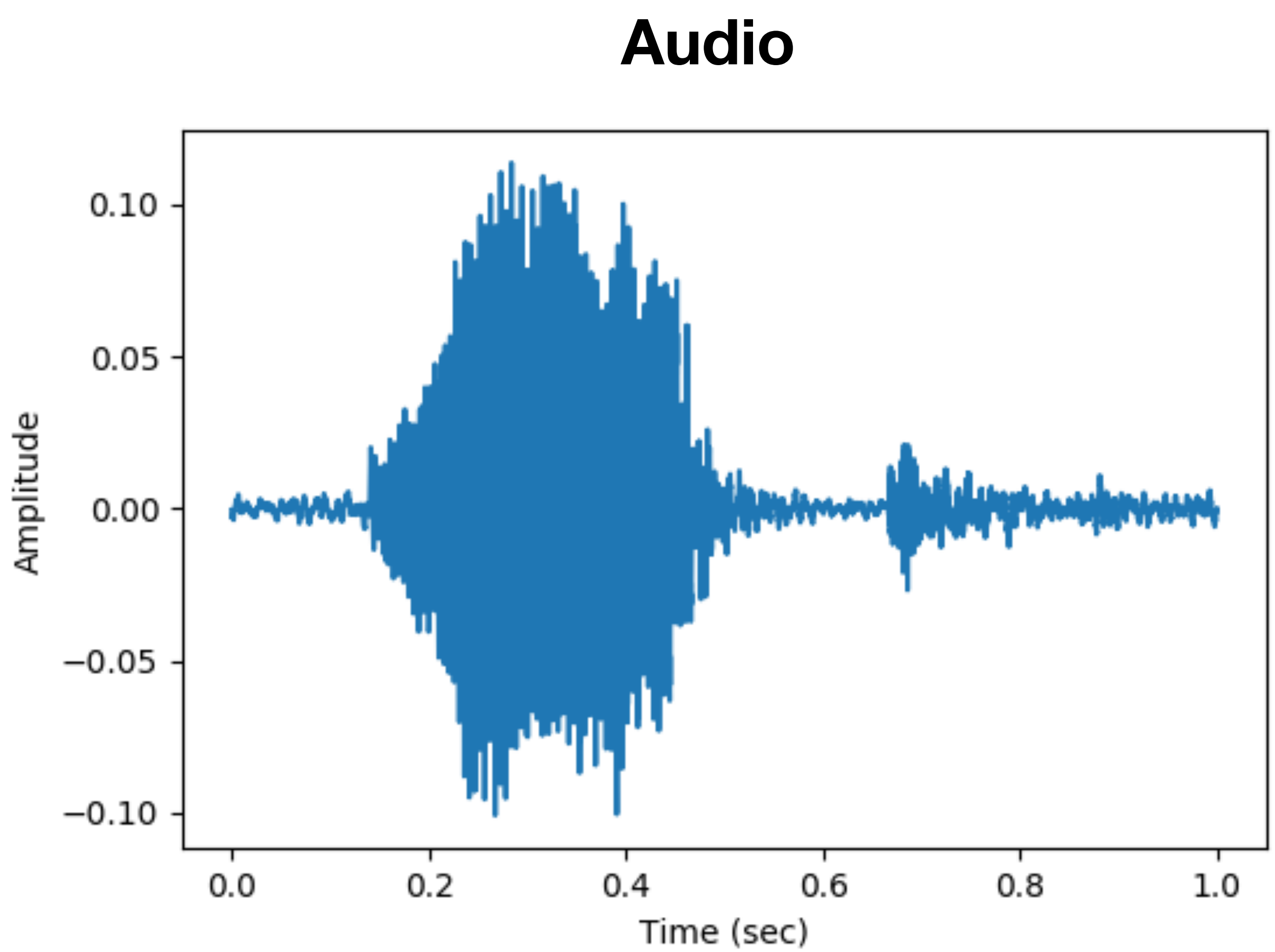
In mobile sensing we have a large number of sensors



UNIVERSITY OF CAMBRIDGE

Part - 2

Hot-keyword Detection: Problem Definition



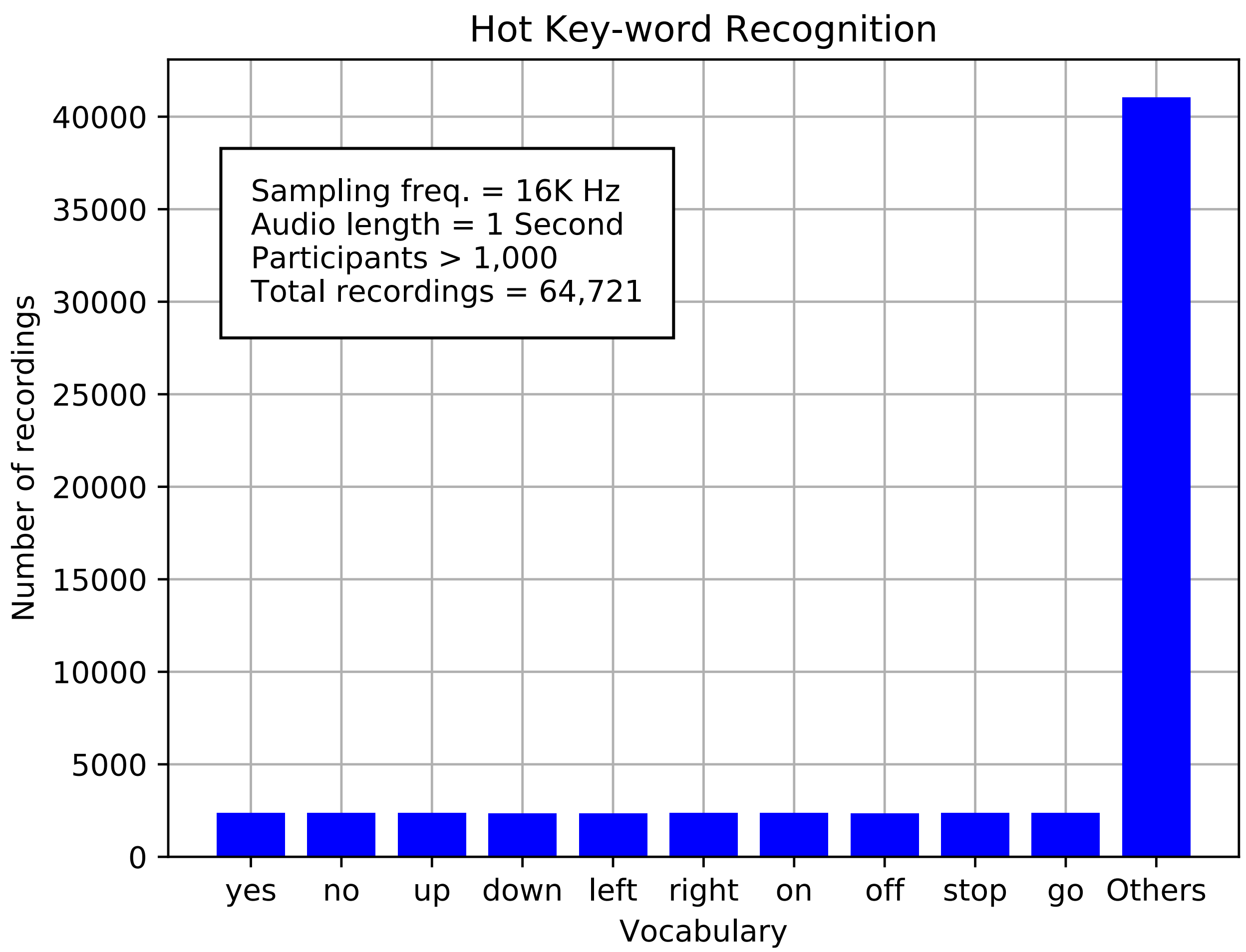
Target classes

- Yes
- No
- Up
- Down
- Stop
- Go
- Left
- Right
- On
- Off
- Silence
- Unknown

$$h_\theta : X \rightarrow \{C_1, \ldots, C_{12}\}, \text{where } X \in \mathcal{R}^d$$

UNIVERSITY OF CAMBRIDGE

HotKeyword Dataset



Dataset

- 16 KHz, 16-bit audio

$$h_\theta : X \to \{C_1, \ldots, C_{12}\}$$

$$X \in \mathcal{R}^{16,000}$$

Training a CNN: Supervised Learning

$$h_\theta, \text{where } \theta \in \mathcal{R}^p$$

Feature vector	Label
$\boldsymbol{x}_1$	C_1
$\boldsymbol{x}_2$	C_1
$\vdots$	$\vdots$
$\boldsymbol{x}_n$	C_{10}

Acknowledgement: Pete Warden, Google

https://www.tensorflow.org/versions/master/tutorials/audio_recognition

UNIVERSITY OF CAMBRIDGE

End-to-end CNN Architecture

```
HotKeywordNet(
  (layer1): Sequential(
    (0): Conv2d(1, 16, kernel_size=(1, 64), stride=(1, 2), padding=(0, 32))
    (1): BatchNorm2d(16, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace)
    (3): MaxPool2d(kernel_size=(1, 8), stride=(1, 8), padding=0, dilation=1, ceil_mode=False)
  )
  (layer2): Sequential(
    (0): Conv2d(16, 32, kernel_size=(1, 32), stride=(1, 2), padding=(0, 16))
    (1): BatchNorm2d(32, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace)
    (3): MaxPool2d(kernel_size=(1, 8), stride=(1, 8), padding=0, dilation=1, ceil_mode=False)
  )
  (layer3): Sequential(
    (0): Conv2d(32, 64, kernel_size=(1, 16), stride=(1, 2), padding=(0, 8))
    (1): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace)
  )
  (layer4): Sequential(
    (0): Conv2d(64, 128, kernel_size=(1, 8), stride=(1, 2), padding=(0, 4))
    (1): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace)
  )
  (layer5): Sequential(
    (0): Conv2d(128, 256, kernel_size=(1, 4), stride=(1, 2), padding=(0, 2))
    (1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace)
    (3): MaxPool2d(kernel_size=(1, 4), stride=(1, 4), padding=0, dilation=1, ceil_mode=False)
  )
  (layer6): Linear(in_features=512, out_features=256, bias=True)
  (layer7): Linear(in_features=256, out_features=12, bias=True)
)
```

- Input: Raw audio samples
- Output: Logits (dimension=12)
- Normalization:

$$\sigma(x)_j = \frac{e^{x_j}}{\sum_{i=1}^{K} e^{x_i}}$$

- Distance metric: Cross-entropy, KLD

UNIVERSITY OF CAMBRIDGE

Loss Function

- In case of supervised learning

$$\mathcal{L}(h_\theta(x_i), y_i)$$

$$\mathcal{L}(\sigma(h_\theta(x_i)), onehot(y_i))$$

- Cross-entropy loss

$$-\sum_{i=1}^{K} p_i \log(q_i)$$

$$-\sum_{I=1}^{K} y_i \log\left(\frac{\exp(h_\theta(x)_i)}{\sum_{j=1}^{K}\exp(h_\theta(x)_j)}\right) = -\log\left(\frac{\exp(h_\theta(x)_y)}{\sum_{j=1}^{K}\exp(h_\theta(x)_j)}\right)$$

$$= \log\left(\sum_{j=1}^{K}\exp(h_\theta(x)_j)\right) - h_\theta(x)_y$$

UNIVERSITY OF CAMBRIDGE

Training CNN: Loss Minimization

Average loss:
$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(h_\theta(x_i), y_i)$$

Gradient descent:
$$\theta \leftarrow \theta - \frac{\alpha}{|\mathcal{B}|} \sum_{i=1}^{N} \nabla_\theta \mathcal{L}(h_\theta(x_i), y_i)$$

Problems with gradient descent:

- No guarantee that it will find a global minimum
- Convergence to a local minimum can be slow

UNIVERSITY OF CAMBRIDGE

HotKeyword recognition: A Practical Guide

Step 1: Splitting dataset into training, validation and test sets

Step 2: Perform data normalization, e.g., 0 dbFS

Step 3: Model architecture selection and parameter initialization

Step 4: Fast mini-batch generation

Step 5: Data augmentation to make the trained model resilient to noise

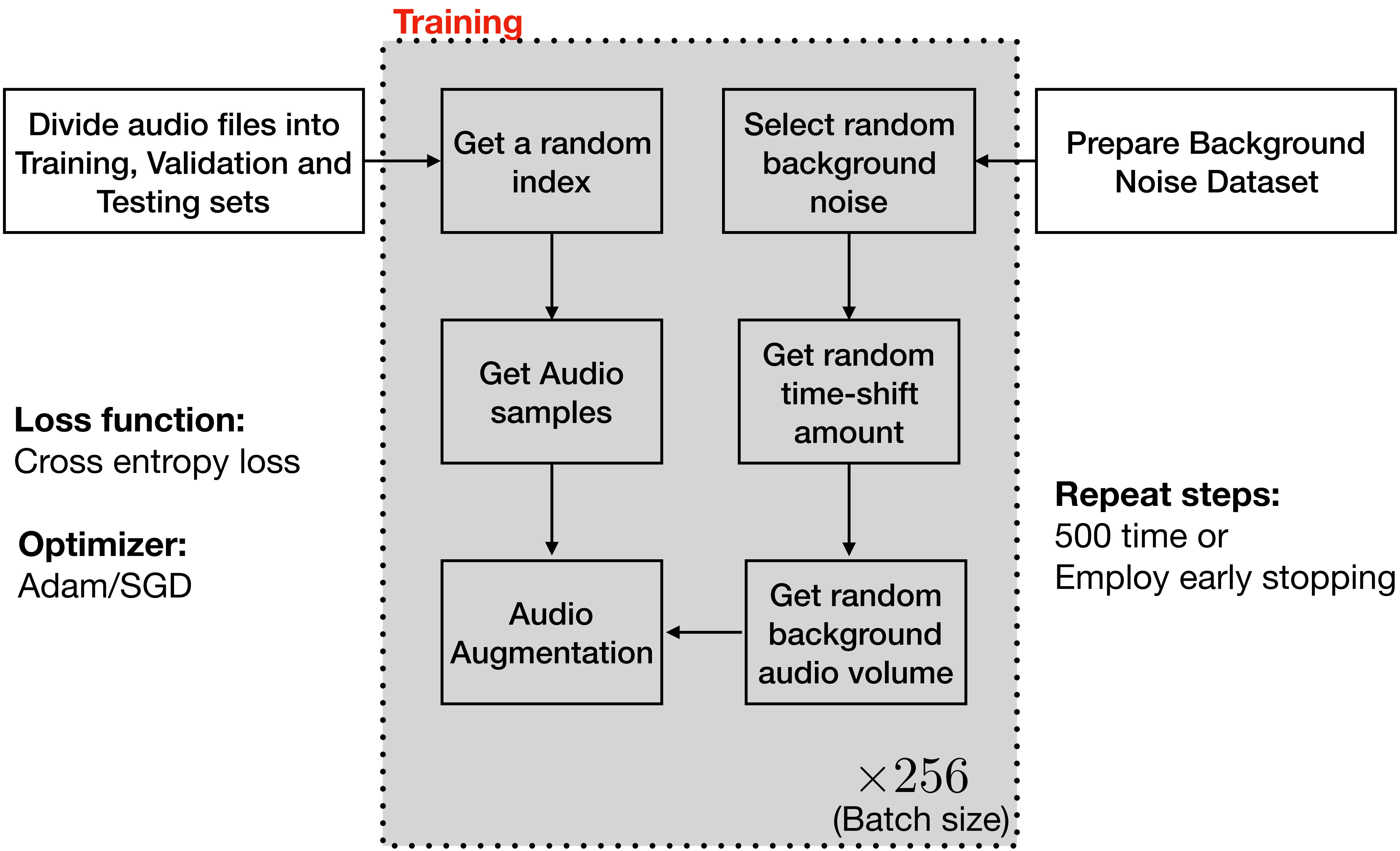
Step 6: Perform model prediction on the augmented mini-batch

Step 7: Compute loss and perform gradient descent

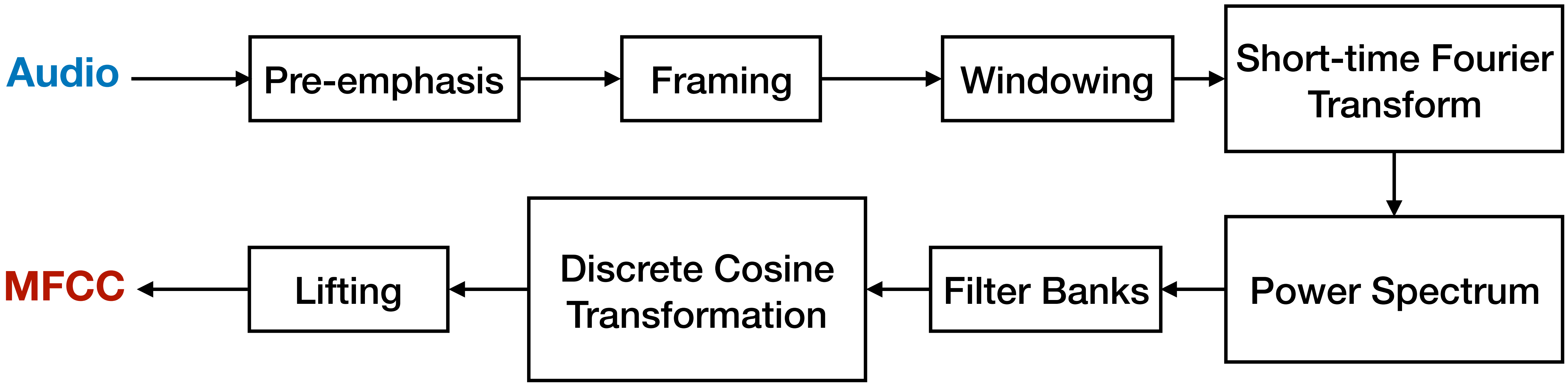
Step 8: Stop if the model has converged, otherwise go to **Step 4**

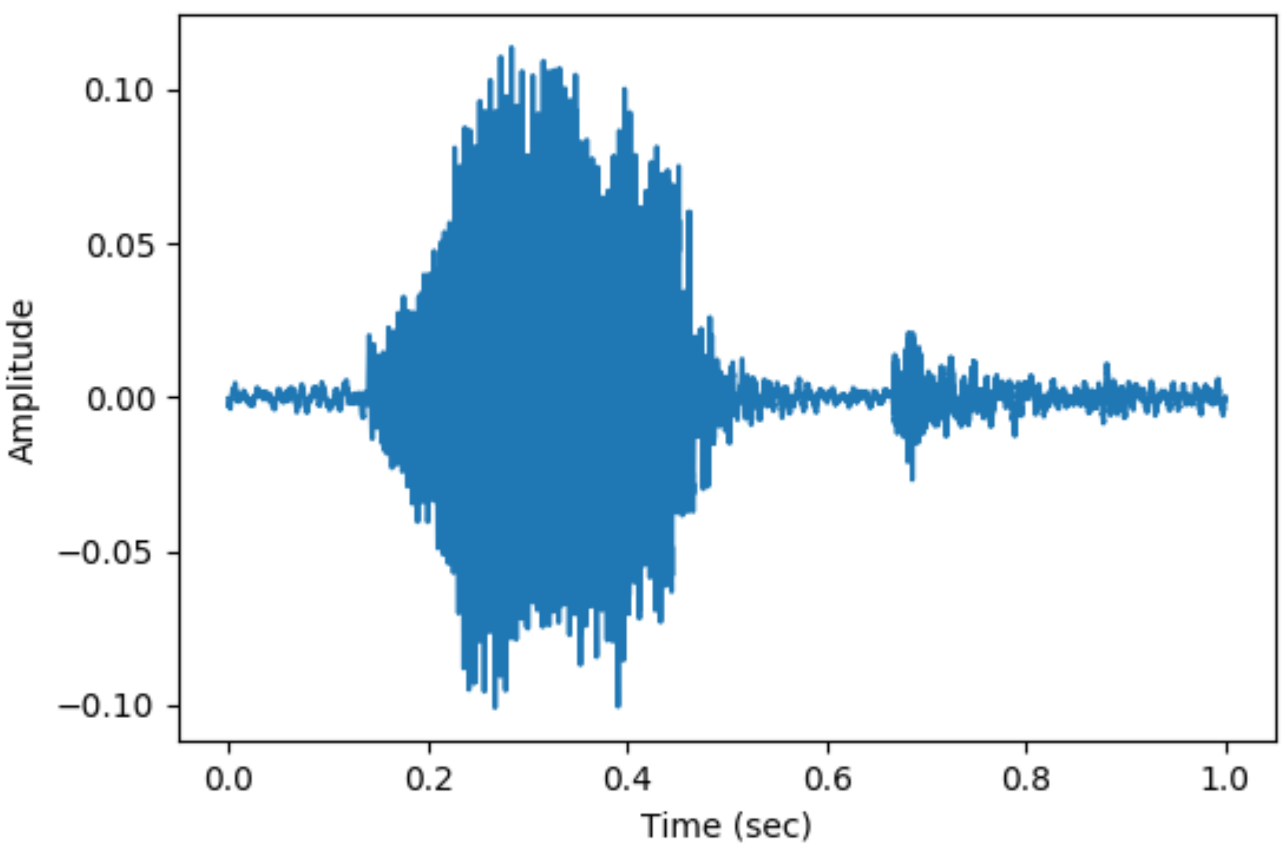
UNIVERSITY OF CAMBRIDGE

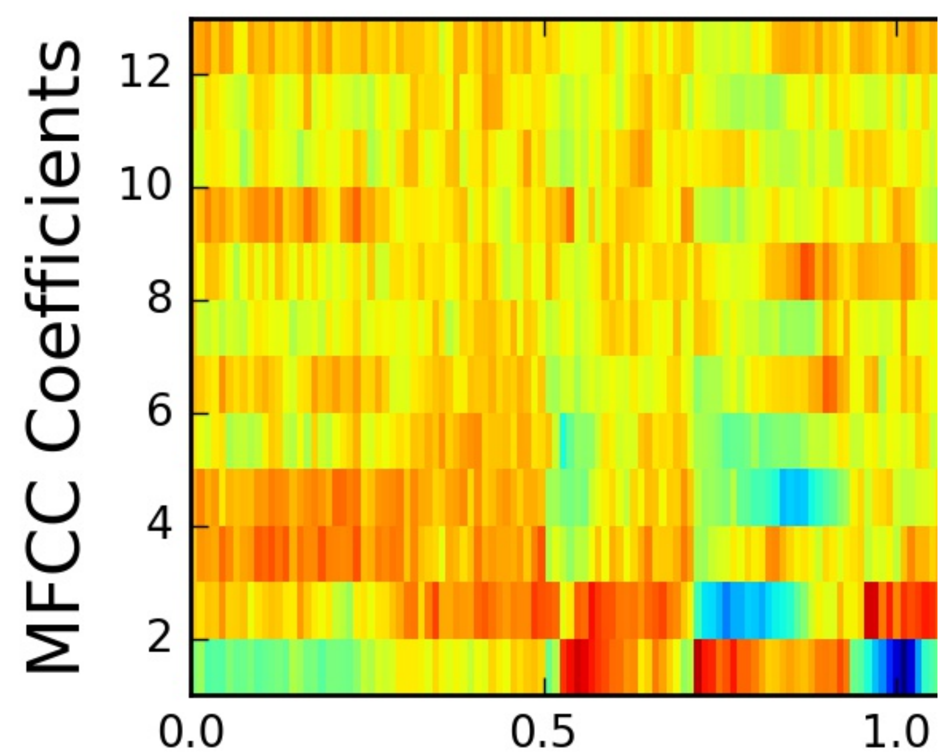
Convolutional Neural Network Training for Hot Key-word Recognition



Input : MFCC Features



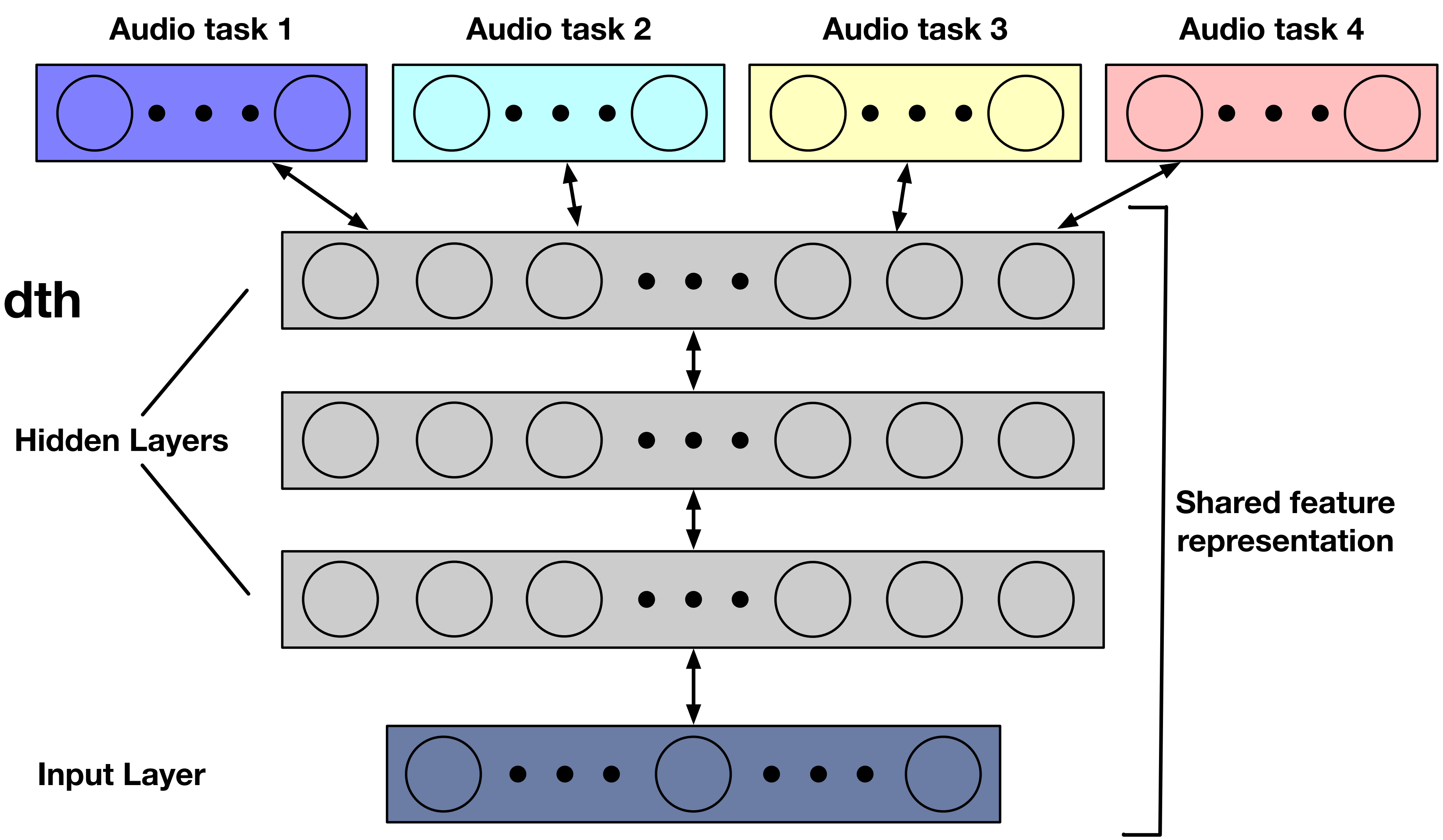




Multi-task Audio Inferencing

- **Objective:** Infer multiple contexts from the same input audio
  - Who is the speaker? Is the person stressed? Male or female speaker?

- **Save memory & bandwidth**
- **Improve latency**
- **Preserve privacy**

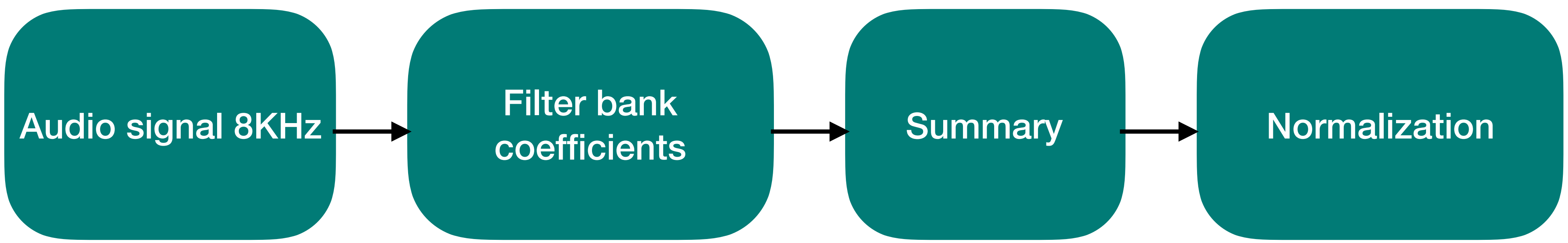


Shared hidden layer architecture

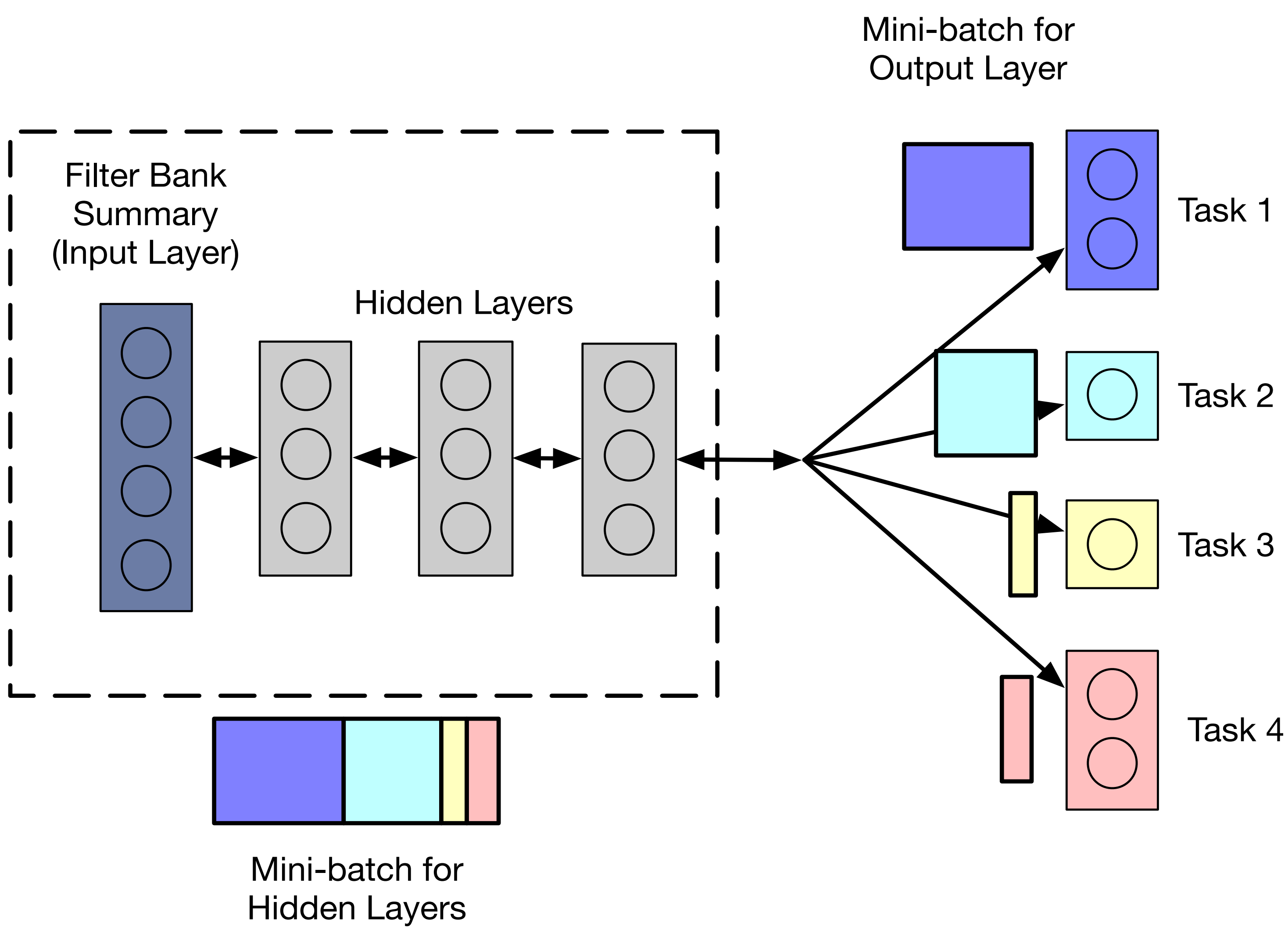
UNIVERSITY OF CAMBRIDGE

Multi-task Training

- **Audio pre-processing**



- **Training shared architecture**



UNIVERSITY OF CAMBRIDGE

Open Research Questions

- How can we use unsupervised data to bootstrap the training procedure and reduce the amount of labeled data?
- How can we squeeze the resource requirements of large-scale neural networks for resource-constrained devices?
- Protecting privacy of the users.
- Multi-modal rich modeling of sensor data for accurate high-level context-recognition.

UNIVERSITY OF CAMBRIDGE

References

- N.D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, A. Campbell. A survey of mobile phone sensing. IEEE Computer Magazine. Vol. 48. No 9. September 2010.

- K.K. Rachuri, M. Musolesi, C. Mascolo, P.J. Rentfrow, C. Longworth, A. Aucinas. EmotionSense: A Mobile Phones based Adaptive Platform for Experimental Social Psychology Research. Ubicomp'10. September 2010.

- S. Hemminki, P. Nurmi, S. Tarkoma, Accelerometer-based transportation mode detection on smartphones, SenSys 2011.

- S. Bhattacharya and N.D. Lane, Sparsification and Separation of Deep Learning Layers for Constrained Resource Inference on Wearables, SenSys 2016.

- P. Georgiev, S. Bhattacharya, N.D. Lane, C. Mascolo, Low-resource Multi-task Audio Sensing for Mobile and Embedded Devices via Shared Deep Neural Network Representations. IMWUT (UbiComp) September 2017.