ETH zürich

Project 2: Time Series and Representation Learning

Thomas Sutter, Alizée Pace

23.04.2024

Big challenge in applying AI to clinical data

Limited and highly heterogeneous data sources (1,2)

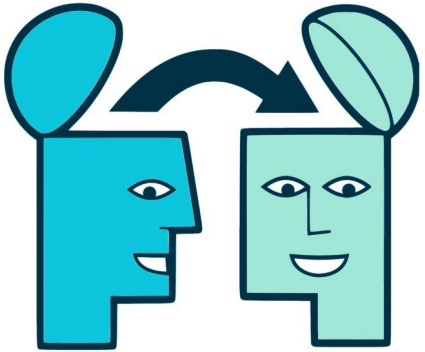
Transfer and Representation Learning can pool information across datasets

(1) Kuchinke W. et al. Heterogeneity prevails: the state of clinical trial data management in Europe - results of a survey of ECRIN centres. *Trials*. 2010 Jul 21;11:79.
(2) Houston L. et al. Heterogeneity in clinical research data quality monitoring: A national survey. *Journal of Biomedical Informatics*, 108, 2020.

Project 2: Time Series and Representation Learning

Part 1: Supervised Learning on Time Series

Part 2: Transfer and Representation Learning



Organisational

- Submit the report and code on Moodle until 21.05.2024 23:59.
- Report
  - Must be a PDF.
  - Word limit of 5000 words excluding references.
  - Should be self-contained, i.e. without references to code.
  - Code must be handed-in too and should follow the guidelines on the handout.
- Do not train on the test sets and only provide results that are evaluated on the test set.
- Both datasets are publicly available on Kaggle. It is allowed to use publicly available code but make sure to properly reference external sources. Of course, you are not allowed to use the code of other teams from the current and previous courses.
- For computation-heavy tasks, we have arranged access to the student cluster. The datasets can be accessed on [your home directory]/ml4h_data/project2/project2_TS_input.
  Please refer to the introductory tutorial slides for more information about the student cluster.

Motivation

- As covered in the previous project,
  - cardiovascular diseases (CVDs) are the leading cause of death globally. Coronary heart disease (CHD) is the most common. (1)
  - Heart attacks, or **Myocardial Infarction (MI)**, are commonly caused by CHD.
  - MI occurs when blood flow stops in one of the coronary arteries of the heart, causing tissue death to the heart muscle. (2)

In this project, you will also train ML models for the detection of heart diseases, but on a different data modality: electrocardiograms.

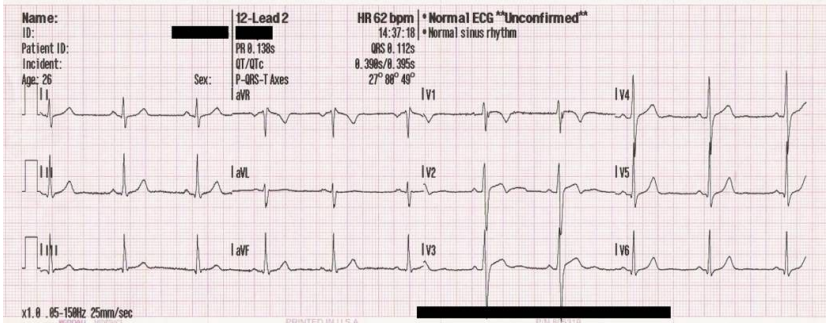
- We will explore **models for learning on time-series**, ubiquitous in the medical domain (e.g. clinical observations, videos).
- We will also study how different datasets can be leveraged to improve performance on distinct tasks through **transfer and representation learning**.

(1) IHME, Global Burden of Disease (2019)

(2) https://www.nhs.uk/conditions/heart-attack/

Electrocardiograms

- An **electrocardiogram** (ECG) is a record of the electrical activity of the heart.
  - It tracks voltage over time.
  - **Electrodes**, placed on the skin of the patient, detect small electrical changes caused by depolarization of the cardiac muscle during each cardiac cycle (heartbeat).



- Widely used as an inexpensive and noninvasive means of diagnosing heart physiology.
- Changes in ECG pattern occur in numerous **cardiac abnormalities**, including cardiac rhythm disturbances, inadequate coronary artery blood flow, etc.
- **Personal devices** that do ECG will become more and more frequent -> more data available! Machine Learning systems are needed to automatically evaluate all this data.

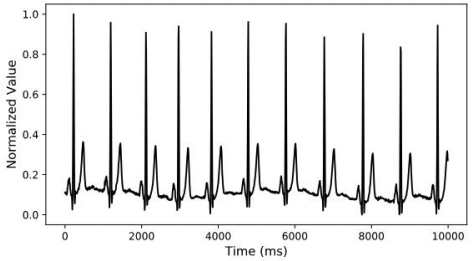
Example: ECG classification within the Apple watch

- **1 channel** ECG, using the electrical heart sensor to record the heartbeat.
- **Atrial Fibrillation** (AFib) occurs when the heart beats in an irregular pattern (the upper and lower chambers of the heart beat out of sync).
  - 9% of over 65yo. have AFib.
  - If left untreated, AFib can lead to heart failure or blood clots that may lead to stroke.

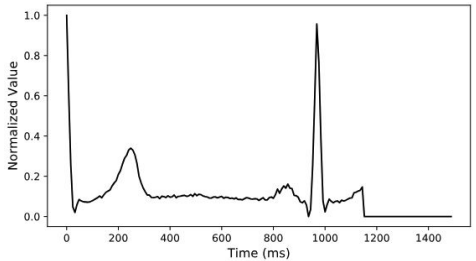
- In a clinical trial of 600 subjects, **ECG app** accurately classifies an ECG recording into AFib (98.3% sensitivity).

MIT-BIH Arrhythmia Database

- First generally available data for development of arrhythmia detectors.
- Raw data:
  - **47 subjects**.
  - **Two-channel** ECG recordings. Half include uncommon but clinically important arrhythmias.
  - Recordings digitized at **360 Hz** with 11-bit resolution over a 10 mV range.
- Multiple cardiologists independently annotated each record. Each beat (**110k** time-series in total!) is extracted and classified into one of **five categories** (see next slide).



ECG Signal



Extracted Heart Beat

G. B. Moody and R. G. Mark, "The impact of the MIT-BIH Arrhythmia Database," in *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45-50, May-June 2001.
https://physionet.org/physiobank/database/mitdb/

MIT-BIH Arrhythmia Database

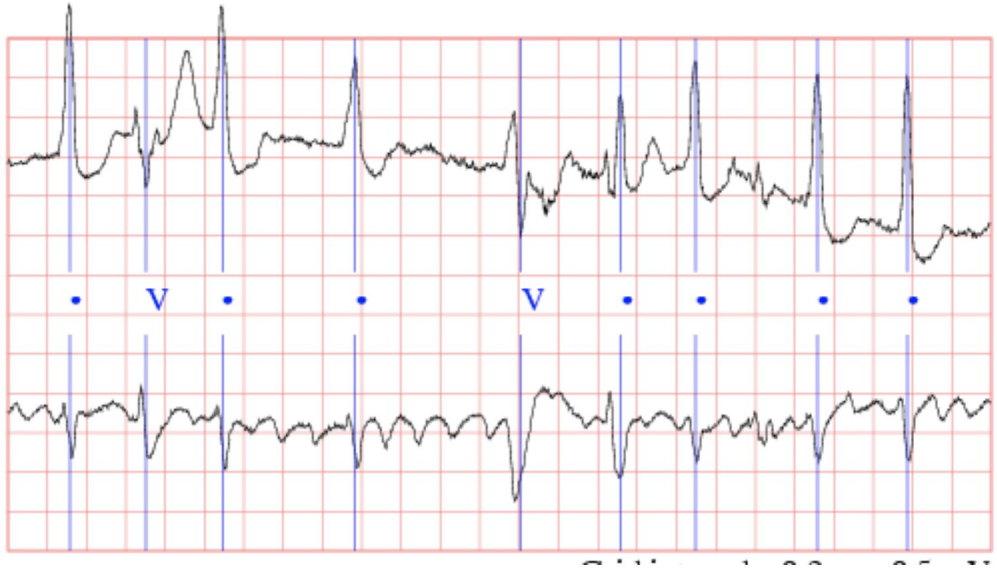


TABLE I: Summary of mappings between beat annotations and AAMI EC57 [18] categories.

Category	Annotations
N	• Normal • Left/Right bundle branch block • Atrial escape • Nodal escape
S	• Atrial premature • Aberrant atrial premature • Nodal premature • Supra-ventricular premature
V	• Premature ventricular contraction • Ventricular escape
F	• Fusion of ventricular and normal
Q	• Paced • Fusion of paced and normal • Unclassifiable

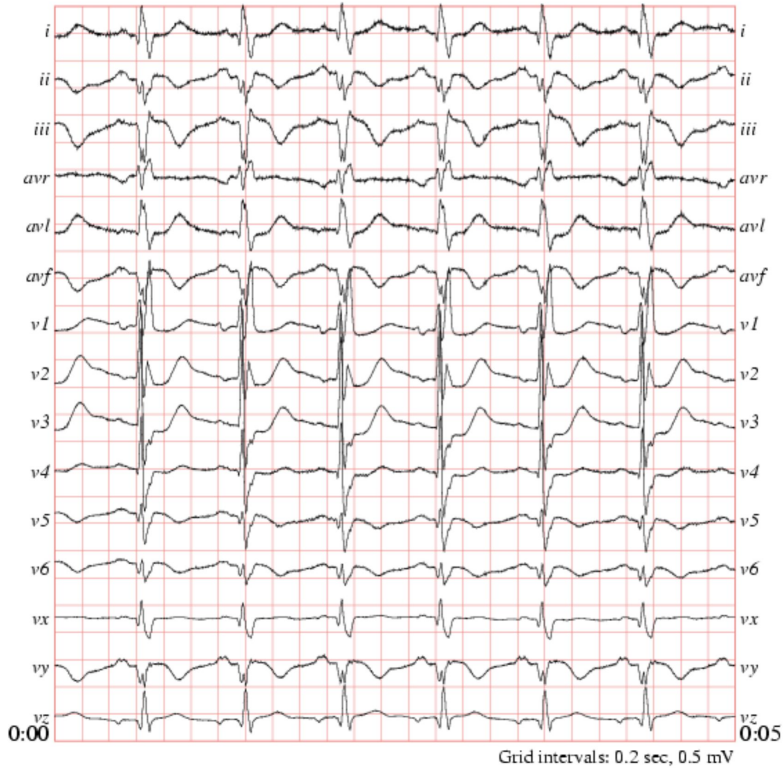
The PTB Diagnostic ECG Database

- Raw data:
  - 549 records from **290 subjects** (aged 17 to 87).
  - Each record includes **15 simultaneously measured signals**: the conventional 12 leads (i, ii, iii, avr, avl, avf, v1, v2, v3, v4, v5, v6) together with the 3 Frank lead ECGs (vx, vy, vz).
  - Recordings digitized at **1000 Hz** with 16 bit resolution over a ± 16 mV range.
- Each beat (**15k time-series** in total) is annotated with **diagnostic classes** extracted from patients' clinical summary (see next slide).

Bousseljot R, Kreiseler D, Schnabel, A. Nutzung der EKG-Signaldatenbank CARDIODAT der PTB über das Internet. *Biomedizinische Technik*, Band 40, Ergänzungsband 1 (1995) S 317.
https://physionet.org/physiobank/database/ptbdb/

The PTB Diagnostic ECG Database

Diagnostic class	Number of subjects
Myocardial infarction	148
Cardiomyopathy/Heart failure	18
Bundle branch block	15
Dysrhythmia	14
Myocardial hypertrophy	7
Valvular heart disease	6
Myocarditis	4
Miscellaneous	4
Healthy controls	52



Two classes only considered in this project: MI/Healthy.

Preprocessing of both datasets for this project

Our goal is to get the best possible performance on PTB. In the first part of the project, we study how to achieve this with supervised learning **on this dataset only**. In the second part, we explore how transfer or representation learning can help us **leverage the MIT-BIH dataset** to potentially improve performance on PTB.

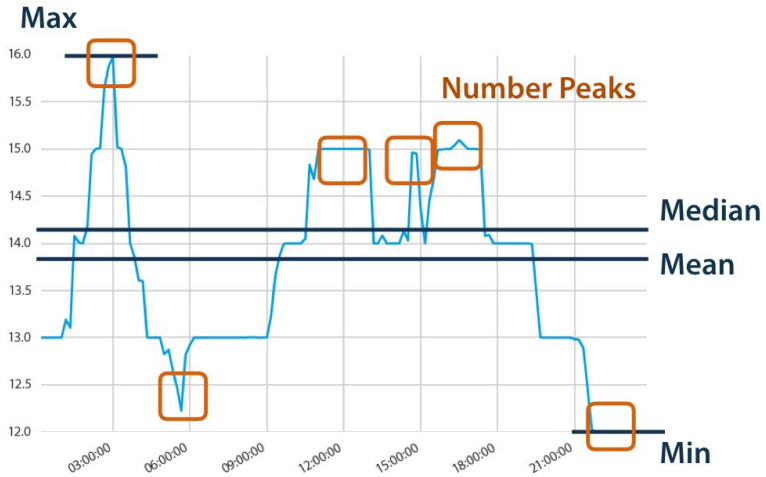
- We only consider ECG lead II in both datasets.
- All samples are cropped, downsampled to 125Hz and padded with zeros if necessary to the fixed dimension of 188.
- Each dataset consists of two CSV files, split into training and test data.
- Each of these CSV files contains a matrix, with each row representing an example in that portion of the dataset. The final element of each row denotes the class to which that example belongs.
  - MIT-BIH: 109446 samples and 5 classes ['N': 0, 'S': 1, 'V': 2, 'F': 3, 'Q': 4]
  - PTB: 14552 samples and 2 classes (MI or normal)

Tasks for Part 1: Supervised Learning on Time Series
(detailed description of deliverables in handout)

- Task 1: Exploratory Data Analysis
  - Before training any classifier, analyse the dataset at hand in order to get an understanding of the data.
  - Based on this, preprocess the data as you see it.

Tasks for Part 1: Supervised Learning on Time Series
(detailed description of deliverables in handout)

- Task 1: Exploratory Data Analysis
- Task 2: Classic ML Methods
  - examples: Logistic Regression, Random Forests, Gradient-Boosted models, etc.
  - These take a **fixed set of features** as input and learn to combine features to make a prediction.
  - Based on a priori knowledge about the nature of the data/task (e.g. from signal processing), can you **design some features** which help improve the model?



Example features from tsfresh (1)

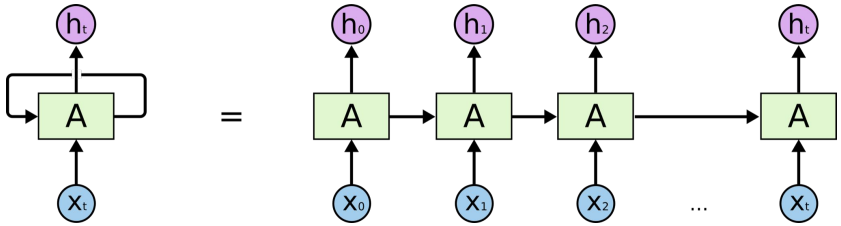
(1) Christ, M., Braun, N., Neuffer, J. and Kempa-Liehr A.W. (2018). *Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests (tsfresh – A Python package)*. Neurocomputing 307 (2018) 72-77.

Tasks for Part 1: Supervised Learning on Time Series
(detailed description of deliverables in handout)

- Task 1: Exploratory Data Analysis
- Task 2: Classic ML Methods
- **Task 3**: Recurrent Neural Networks (1)
  - RNNs handle sequential data of variable length -- see recap on next slide.
  - Does the unidirectional nature of the models pose an issue? Investigate bidirectional approaches.

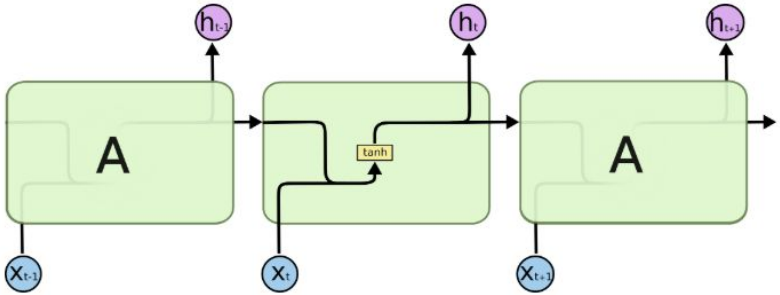
(1) Hochreiter and Schmidhuber, “Long short-term memory”, 1997.

Recap on RNN/LSTM



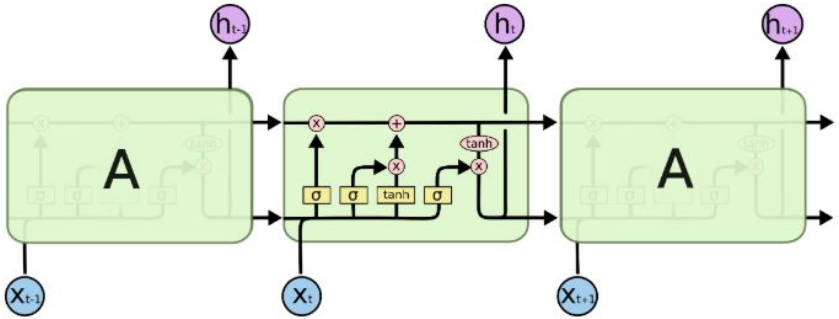
Recurrent network: the same network is applied to every time point. Memory is preserved by passing the hidden state to the successor.

RNN: vanishing gradient problem



The repeating module in a standard RNN contains a single layer.

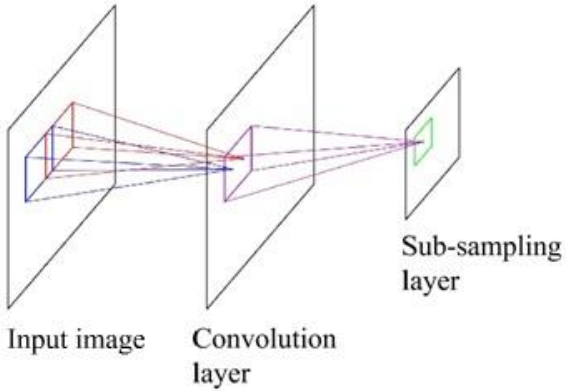
LSTM:

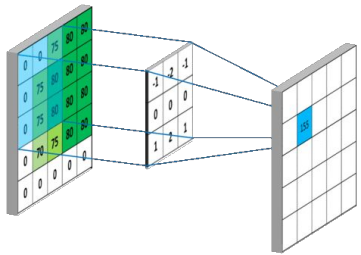


The repeating module in an LSTM contains four interacting layers.

Tasks for Part 1: Supervised Learning on Time Series
(detailed description of deliverables in handout)

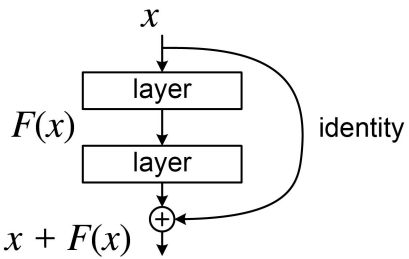
- Task 1: Exploratory Data Analysis
- Task 2: Classic ML Methods
- Task 3: Recurrent Neural Networks
- **Task 4**: Convolutional Neural Networks
  - CNNs also capture dependencies within the data, this time by encouraging robustness against spatial translation.





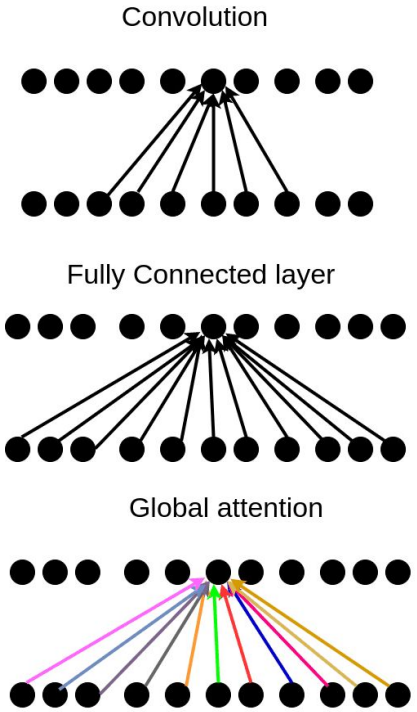
Tasks for Part 1: Supervised Learning on Time Series
(detailed description of deliverables in handout)

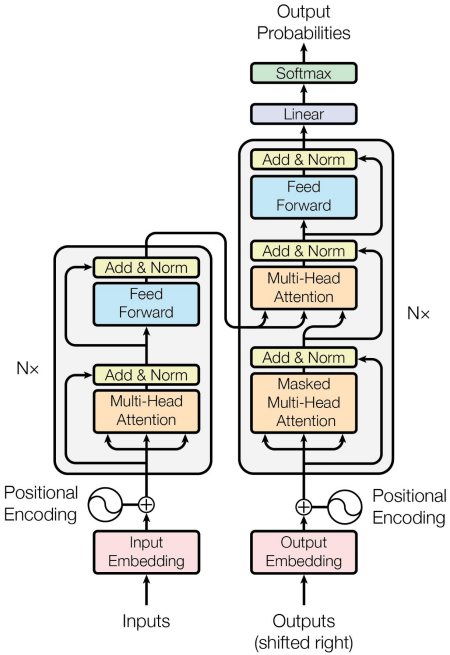
- Task 1: Exploratory Data Analysis
- Task 2: Classic ML Methods
- Task 3: Recurrent Neural Networks
- **Task 4**: Convolutional Neural Networks
  - CNNs also capture dependencies within the data, this time by encouraging robustness against spatial translation.
  - Residual connections have been shown to help scale CNNs on complex tasks. Do you find this to be the case here?



Tasks for Part 1: Supervised Learning on Time Series
(detailed description of deliverables in handout)

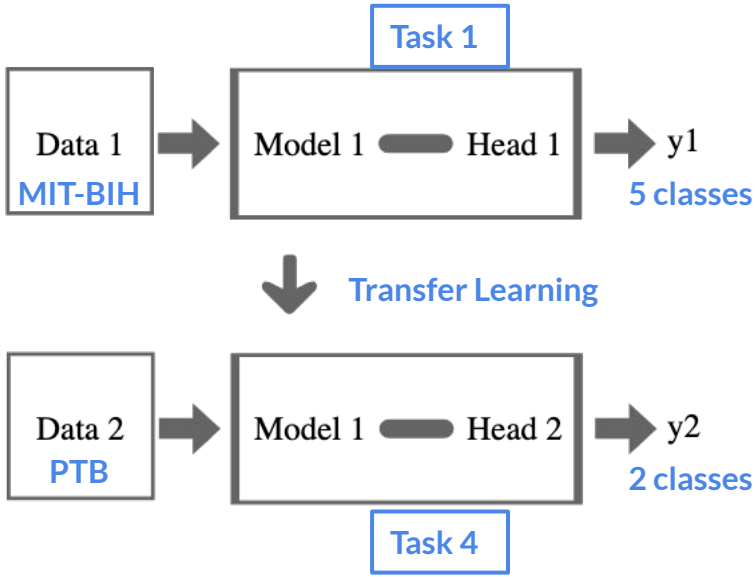
- Task 1: Exploratory Data Analysis
- Task 2: Classic ML Methods
- Task 3: Recurrent Neural Networks
- Task 4: Convolutional Neural Networks
- **Task 5**: Attention & Transformers
  - The attention mechanism learns a "soft" weight for each element within a sequence.
  - These weights can be computed either in parallel (e.g. in transformers) or sequentially (e.g. in RNNs).
  - What do these weights tell us about what the model is 'looking' at?





Tasks for Part 2: Transfer & Representation Learning
(detailed description of deliverables in handout)

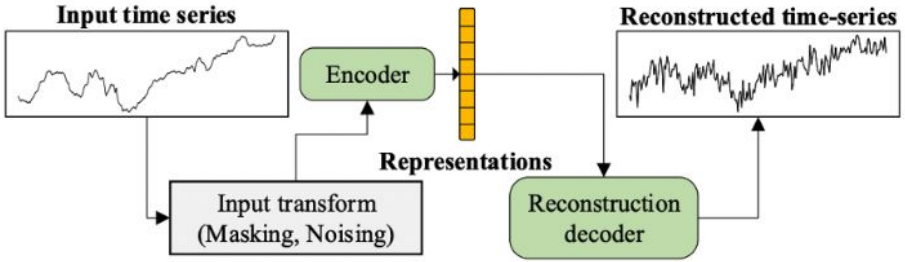
- Task 1: Supervised Model on MIT-BIH dataset
  - Part 1 was on the PTB dataset. We now want to leverage information from the larger MIT-BIH dataset.
  - One approach is simply to train a model supervised on MIT-BIH.



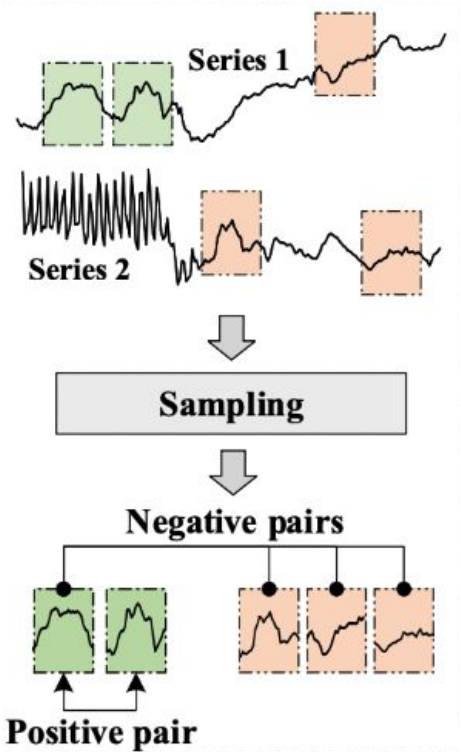
Tasks for Part 2: Transfer & Representation Learning
(detailed description of deliverables in handout)

- Task 1: Supervised Model on MIT-BIH dataset
- Task 2: Representation Learning on MIT-BIH dataset
  - Not all datasets are labeled as collecting labels, especially for medical data, is expensive.
  - Representation Learning overcomes this by first solving surrogate tasks.
  - Training results in low-dimensional representations that are transferable to similar datasets and useful for different downstream tasks.

Representation Learning on Time-Series



Autoencoder-based approaches



Contrastive approaches

Zhang et al., Self-Supervised Learning for Time Series Analysis: Taxonomy, Progress, and Prospects. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI).* 2024.

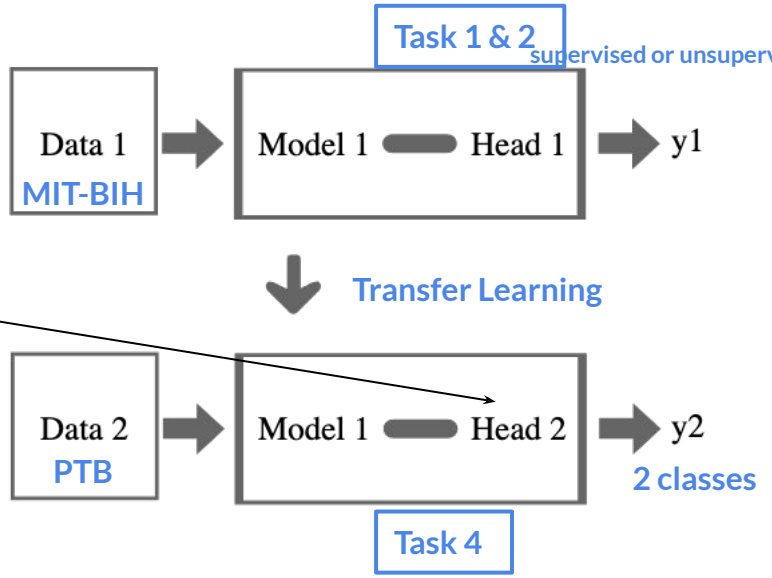
Tasks for Part 2: Transfer & Representation Learning
(detailed description of deliverables in handout)

- Task 1: Supervised Model on MIT-BIH dataset
- Task 2: Representation Learning on MIT-BIH dataset
- **Task 3**: Visualize Learned Representations
  - How do representations of both datasets compare with these two pre-training approaches?
  - Think about how best to visualise a high-dimensional embedding space.

Tasks for Part 2: Transfer & Representation Learning
(detailed description of deliverables in handout)

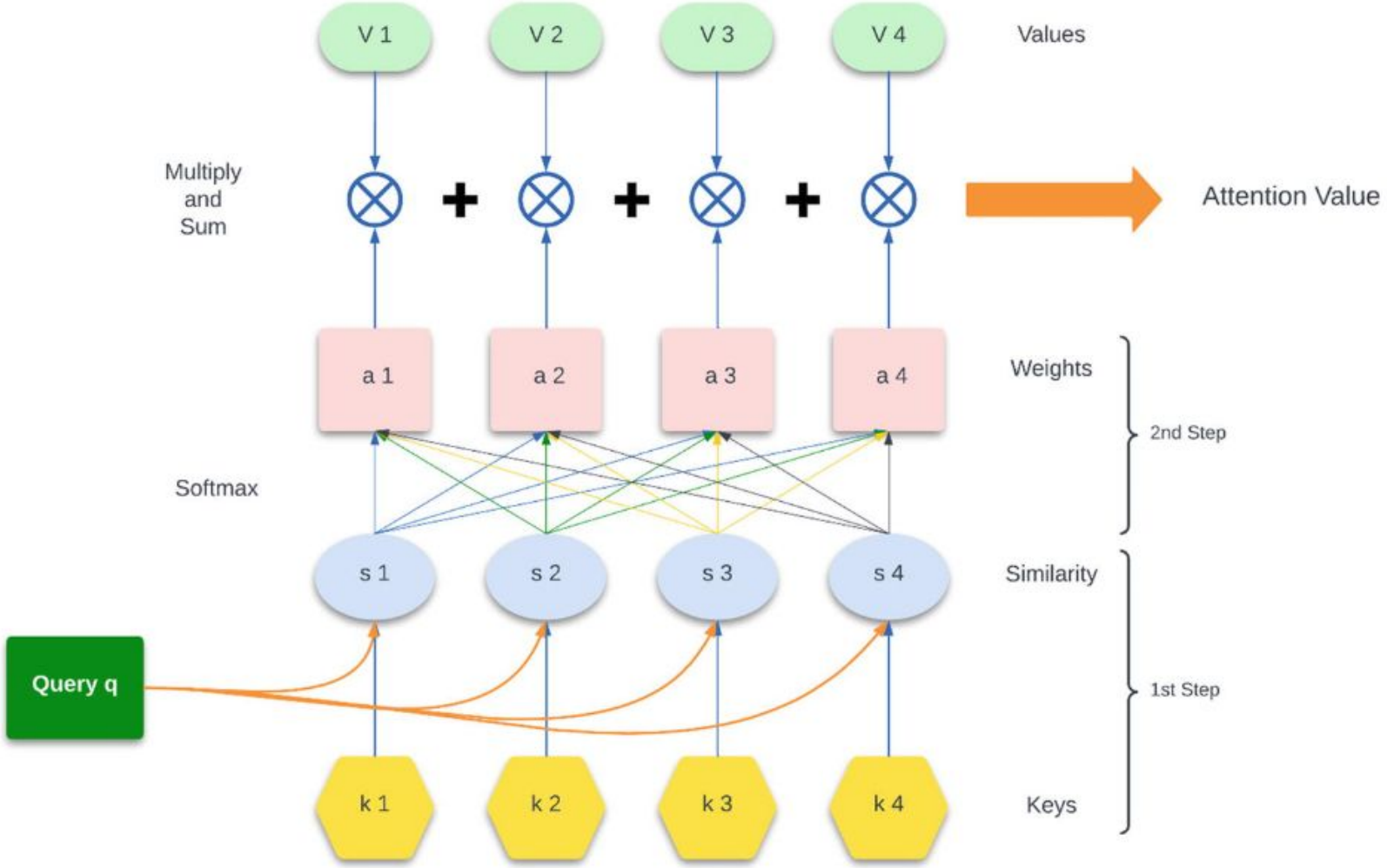
- Task 1: Supervised Model on MIT-BIH dataset
- Task 2: Representation Learning on MIT-BIH dataset
- Task 3: Visualize Learned Representations
- Task 4: Finetuning strategies, aka. 'Head 2'
  - Classic ML model or feed-forward NN layers
  - Train only output layers or also pre-trained model?

Finally: analyse your results and compare performance to Part 1.



Part 3

- General questions allowing you to reflect on your learnings during the project


V 1
V 2
V 3
V 4
Values
Multiply
and
Sum
Attention Value
a 1
a 2
a 3
a 4
Weights
2nd Step
Softmax
s 1
s 2
s 3
s 4
Similarity
Query q
1st Step
k 1
k 2
k 3
k 4
Keys