Practical Processing of Mobile Sensor Data for Continual Deep Learning Predictions

Kleomenis Katevas, Ilias Leontiadis, Martin Pielot, Joan Serrà

1st International Workshop on
Embedded and Mobile Deep Learning
June 23, ACM MobiSys 2017

NFC

GPS

Ambient Light Sensor

Magnetometer

WiFi

Camera(s)

Bluetooth

Barometer

Accelerometer

Microphone(s)

Gyroscope

Proximity Sensor

Motion Coprocessor

Calls

Network data

Screen orientation

App usage

Screen status

Notifications

Ringer mode

Battery

Charging state

Notification center

SMS

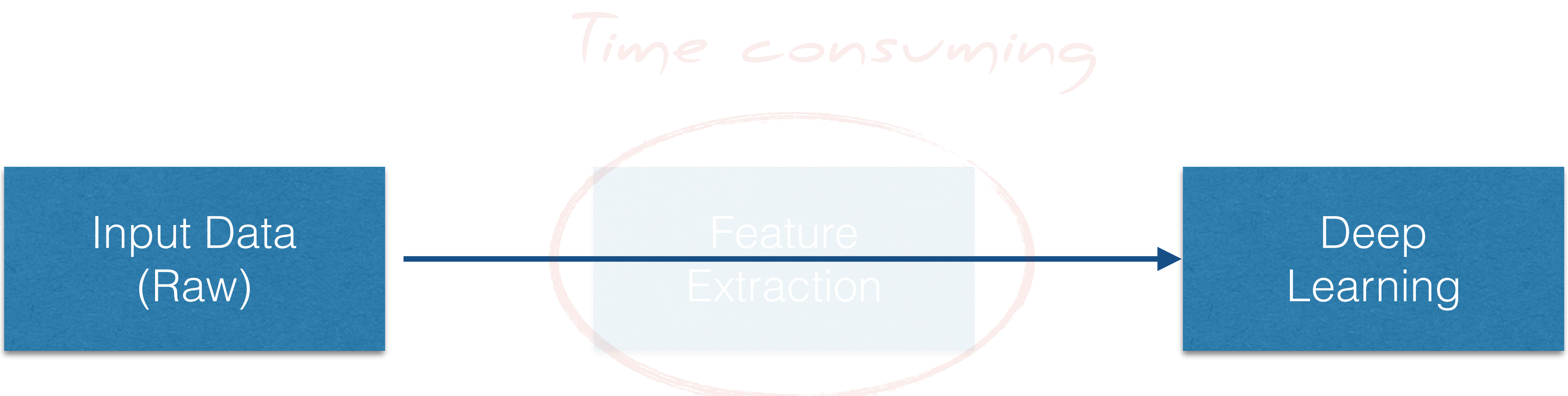
Traditional Machine Learning Pipeline

- Accelerometer
- Battery level
- Network data activity
- Noise level
- Call logs
- Location
- Screen Status
- …

- Mean and Max Acceleration
- Battery drain levels in the last hour
- Data received during the day
- Average ambient noise level during the last five minutes
- Time since the last outgoing call
- Current distance from home
- Number of device unlocks
- …

- Random Forests
- XGBoost
- …

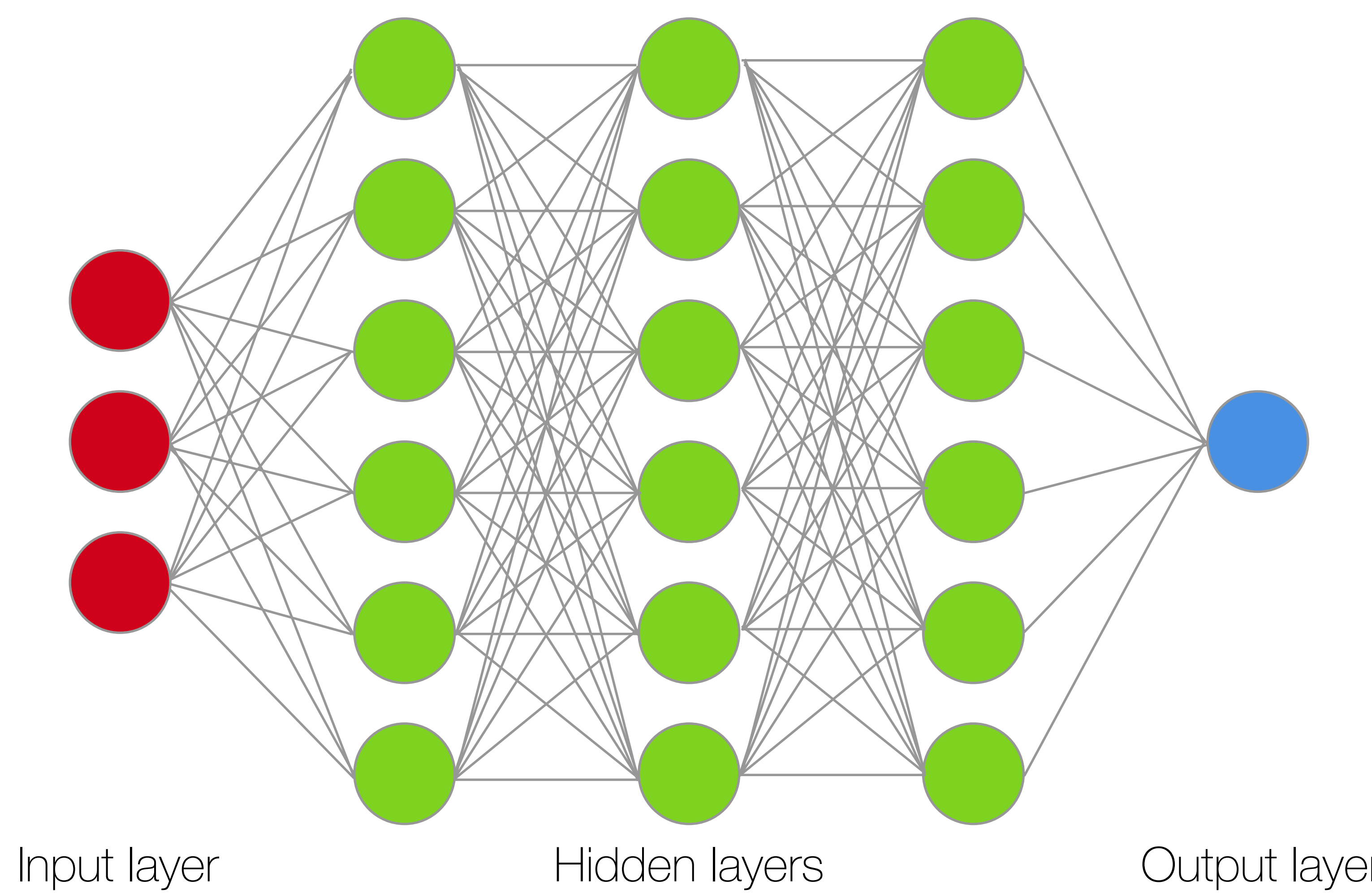
Who needs Feature Engineering?



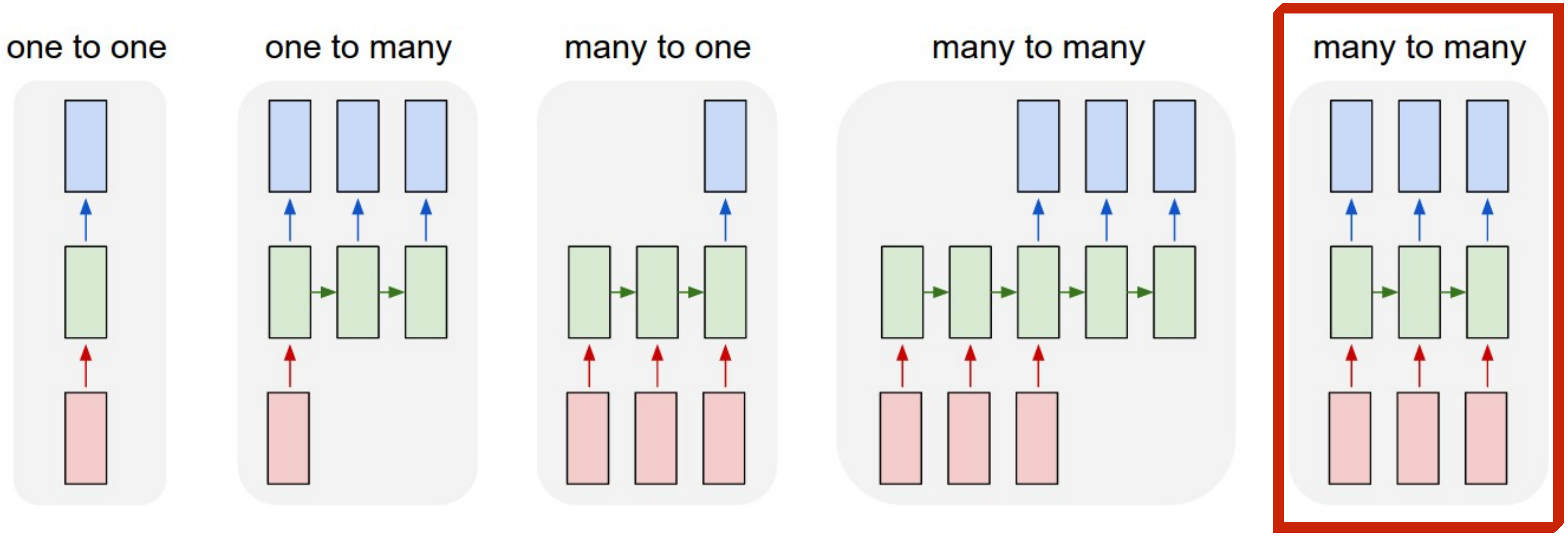
- Accelerometer
- Battery level
- Network data activity
- Noise level
- Call logs
- Location
- Screen Status
- …

- Mean and Max Acceleration
- Battery drain levels in the last hour
- Data received during the day
- Average ambient noise level during the last five minutes
- Time since the last outgoing call
- Current distance from home
- Number of device unlocks
- …

Deep Neural Networks



Recurrent Neural Networks (RNNs)



Source: Andrej Karpathy, 2015

Deep Learning Pipeline

		x				y	w
S_1	t_1	0	0	0.4	0	0	0
S_2	t_2	0.1	0.2	0	0	0	0
S_3	t_3	0	0	0	0.2	0	0
S_4	t_4	0	0	0.5	0	0	0
S_5	t_5	0.3	0.1	0	0	1	0.7
S_6	t_6	0	0	0	0.5	0	0

- S_i: Sensor event (one-hot encoded)
- t_i: Time delta
- *x*: Sensor values
- *y*: Ground truth
- *w*: Weight

Structure Sensor Data and GT → Normalisation → Capping → Time-based Compression → Structure Data for Training

Deep Learning Pipeline

		x				y	w
S_1	t_1	0	0	0.4	0	0	0
S_2	t_2	0.1	0.2	0	0	0	0
S_3	t_3	0	0	0	0.2	0	0
S_4	t_4	0	0	0.5	0	0	0
S_5	t_5	0.3	0.1	0	0	1	0.7
S_6	t_6	0	0	0	0.5	0	0

- *Rescale all interval data to 0.05-1.*

Structure Sensor Data and GT | **Normalisation** | Capping | Time-based Compression | Structure Data for Training

Deep Learning Pipeline

		x				y	w
S_1	t_1	0	0	0.4	0	0	0
S_2	t_2	0.1	0.2	0	0	0	0
S_3	t_3	0	0	0	0.2	0	0
S_4	t_4	0	0	0.5	0	0	0
S_5	t_5	0.3	0.1	0	0	1	0.7
S_6	t_6	0	0	0	0.5	0	0

- *Cap to the 95th percentile.*
- *Cap time deltas to 60 minutes.*

Structure Sensor Data and GT | Normalisation | **Capping** | Time-based Compression | Structure Data for Training

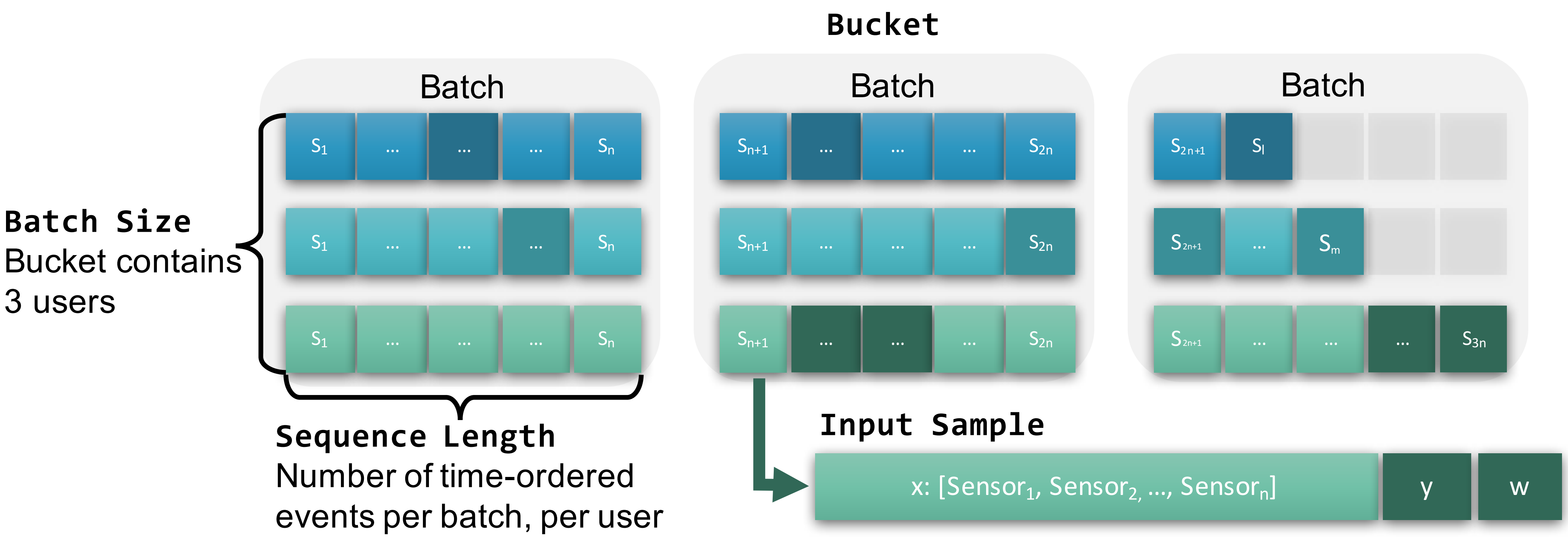
Deep Learning Pipeline

		x				y	w
S_1	t_1	0	0	0.4	0	0	0
S_2	t_2	0.1	0.2	0	0	0	0
S_3	t_3	0	0	0	0.2	0	0
S_4	t_4	0	0	0.5	0	0	0
S_5	t_5	0.3	0.1	0	0	1	0.7
S_6	t_6	0	0	0	0.5	0	0

	x				y	w
t_{1+2+3}	0.1	0.2	0.4	0.2	0	0
t_{4+5}	0.3	0.1	0.5	0	1	0.7
t_6	0	0	0	0.5	0	0

Structure Sensor Data and GT | Normalisation | Capping | **Time-based Compression** | Structure Data for Training

Deep Learning Pipeline



Structure Sensor Data and GT | Normalisation | Capping | Time-based Compression | **Structure Data for Training**

Case Study: Predicting Reactiveness to Notifications

- 279 users: collected detailed mobile phone usage logs using an Android app.
- Age: 18 to 66 years (M = 37.7, SD = 11.1).
- Gender: 52.7% female and 47.3% male.
- For a period of 5 weeks we collected 446,268 notifications from a variety of apps.

Data Collection

Periodical (10 minutes)	Event-driven
Accelerometer	App Usage
Battery	Audio (Source, Music)
Data (Rx, Tx, MobRx, MobTx)	Charging State
Light	Notification Received
Noise	Notification Center
Semantic Location	Ringer Mode
	Screen Status
	Screen Orientation

Data Collection

Periodical (10 minutes)	Event-driven
Accelerometer	**App Usage**
Battery	Audio (Source, Music)
Data (Rx, Tx, MobRx, MobTx)	Charging State
Light	**Notification Received**
Noise	Notification Center
Semantic Location	Ringer Mode
	Screen Status
	Screen Orientation

Model Architecture

Data Input

↓

Fully Connected Layer (50 units)

Time Distributed PreLU

LSTM layer 1 (500 units)

LSTM layer 2 (500 units)

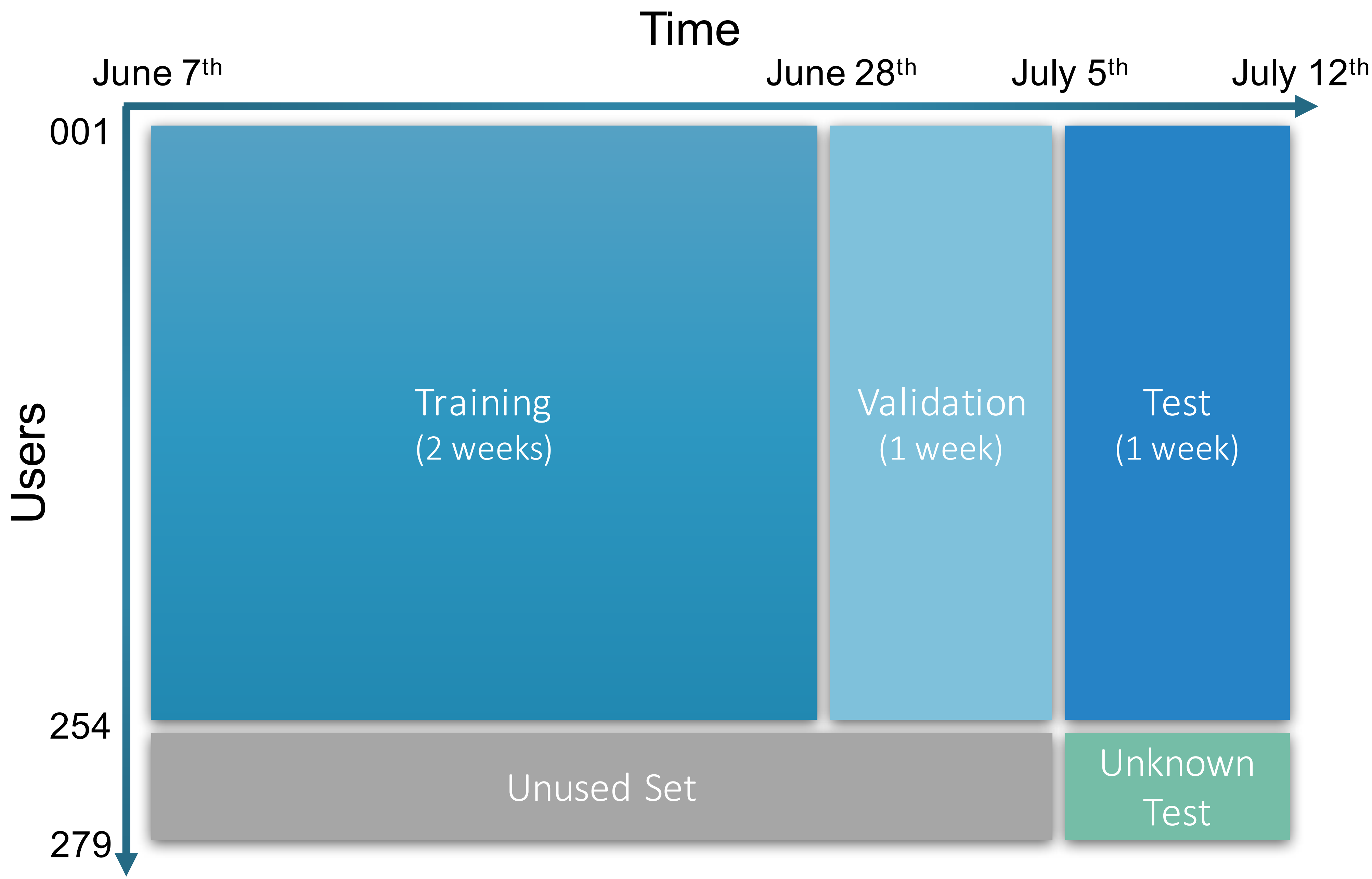
Fully Connected Layer with Sigmoid Activation Function

↓

Prediction

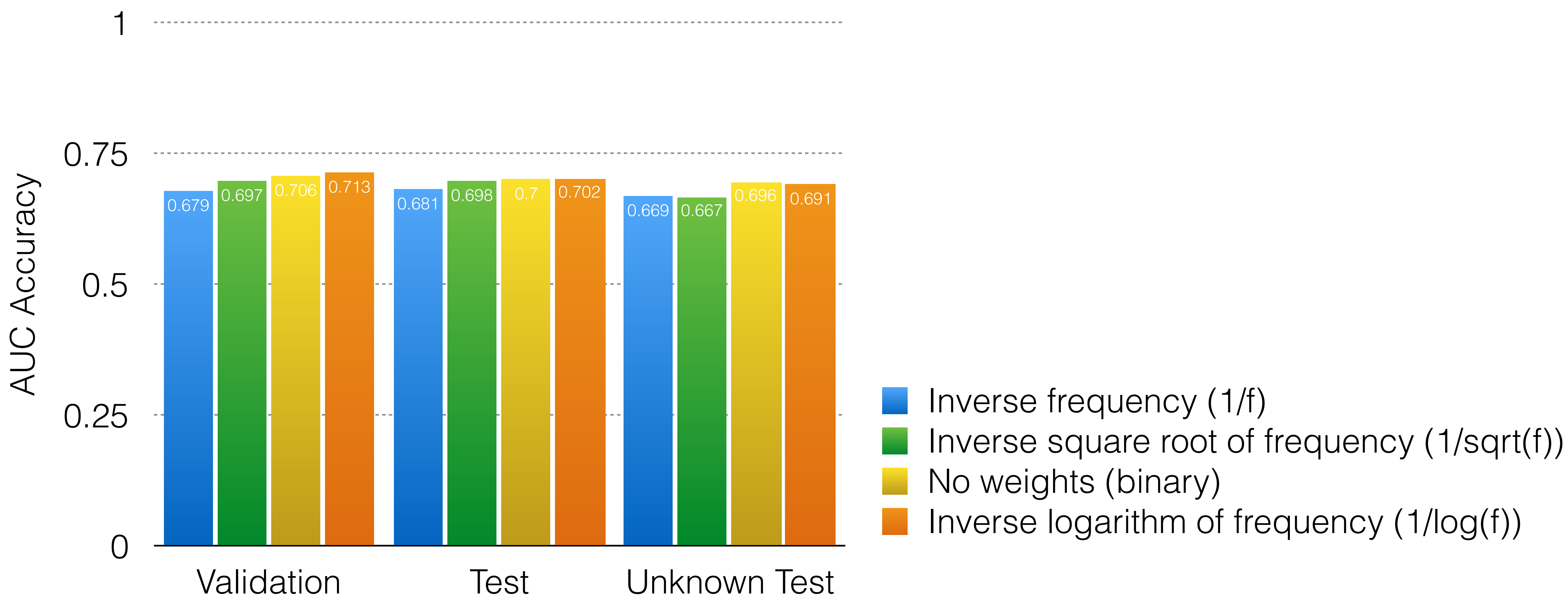
Keras + theano

Evaluation

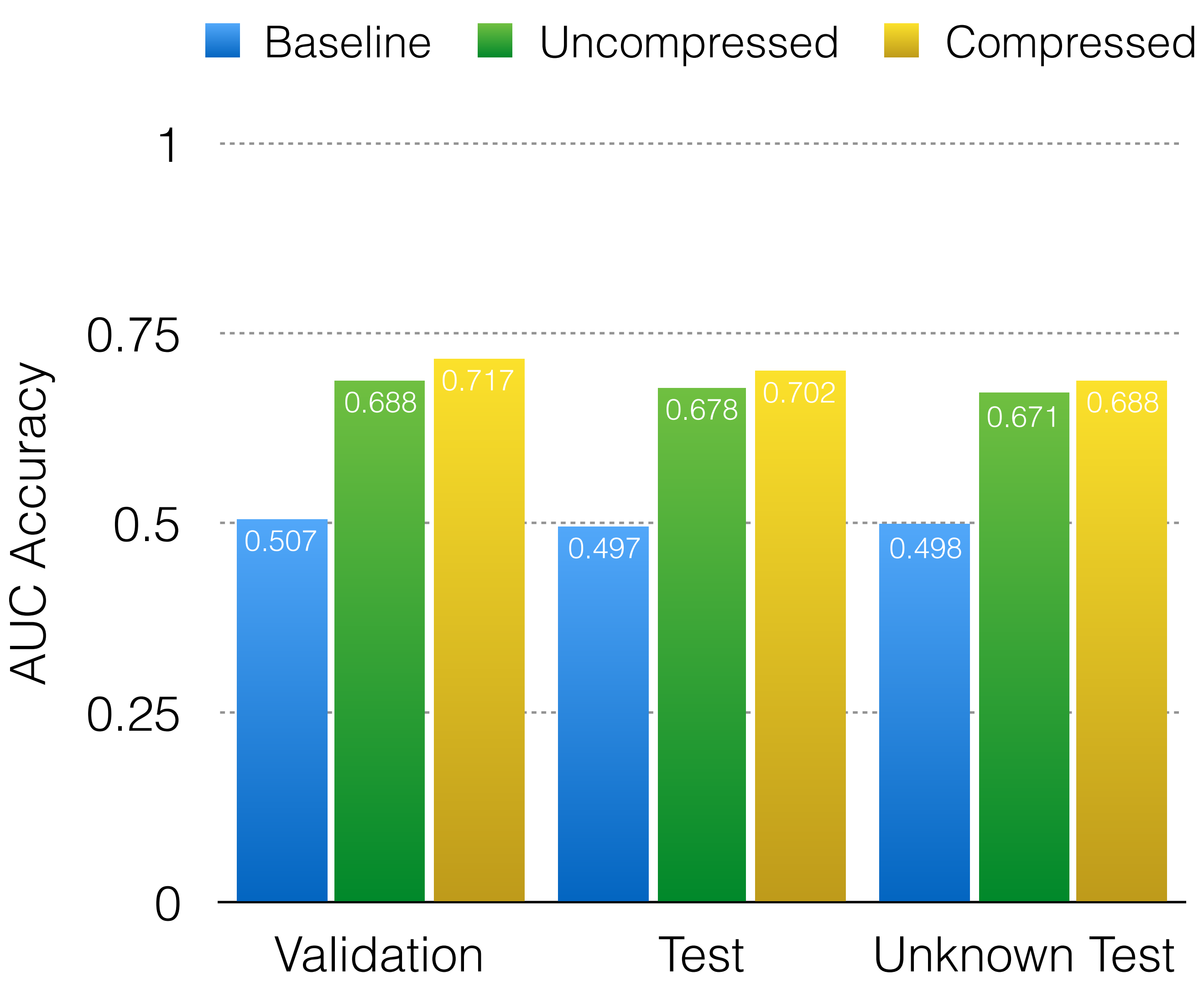


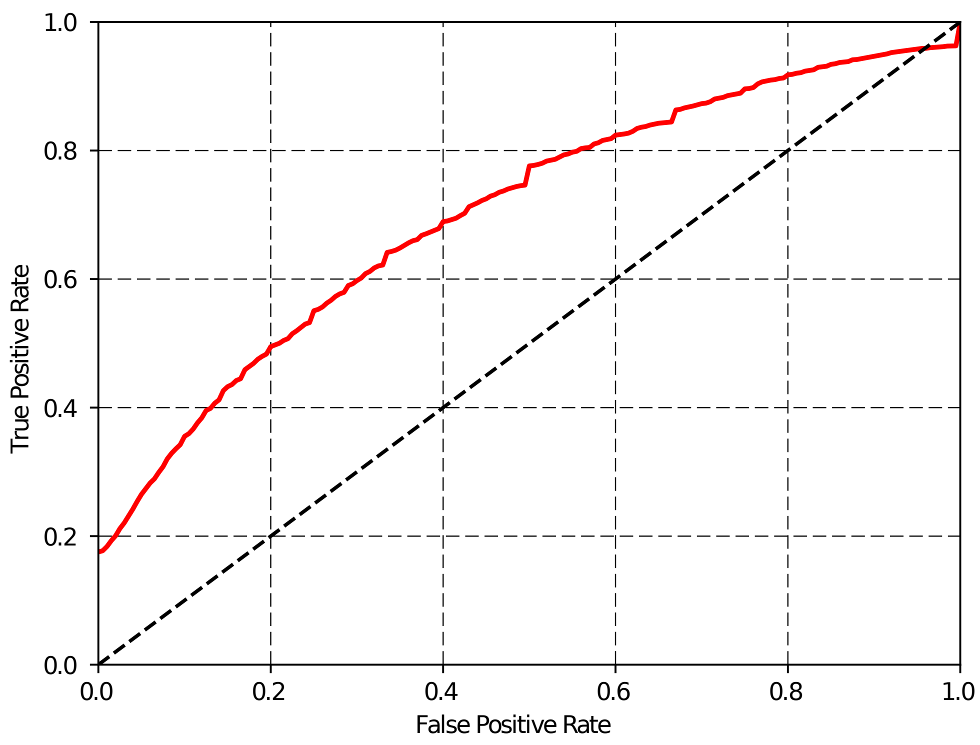
Time
June 7th
June 28th
July 5th
July 12th
001
254
279
Users
Training
(2 weeks)
Validation
(1 week)
Test
(1 week)
Unused Set
Unknown
Test

Results


1
0.75
0.5
0.25
0
AUC Accuracy
0.679
0.697
0.706
0.713
0.681
0.698
0.7
0.702
0.669
0.667
0.696
0.691
Validation
Test
Unknown Test
Inverse frequency (1/f)
Inverse square root of frequency (1/sqrt(f))
No weights (binary)
Inverse logarithm of frequency (1/log(f))

Results





ROC of the test set using the time-based compression and inverse log-frequency weights

Conclusions

- Introduced an approach for preparing time series sensor data for deep learning applications.
- Demonstrated the effectiveness in a case study.
- Achieved a 40% performance increase compared to a probabilistic random baseline.
- The model generalises to unknown users.

Future Work

- Evaluate each notification category separately.
- Compare the performance to canonical approaches (*i.e.* XGBoost - feature engineering).
- Improve the compression strategy.
- Explore more sophisticated deep learning techniques (e.g. transfer learning, generative adversarial networks).

Ilias Leontiadis

Martin Pielot

Joan Serrà

Telefónica