MobileHCI 2018 tutorial: Machine Learning for Intelligent Mobile User Interfaces using Keras

Human Activity Recognition

Huy Viet Le, Sven Mayer, **Abdallah El Ali**, Niels Henze

CWI Centrum Wiskunde & Informatica

interactionlab.io

Human Activity Recognition (HAR)

- Automatic recognition of physical activities
- Important for context-awareness, health monitoring, social network apps, etc.

End User Applications	
Fitness tracking	Actitracker provides online activity history
Health monitoring	Evaluate patients over time rather than single session
Fall detection	Detect falls and take action
Context-aware behavior	Disable calls while jogging
Home/work automation	Smart homes that anticipate user's needs
Self-managing systems	Save battery by turning off WiFi while jogging
Third Party Applications	
Targeted advertising	Provide users with relevant ads
Research platform	Provide platform for collecting activity data
Corporate management & accounting	Track employee time and ensure spent appropriately
Crowd and Social Network (SN) Applications	
Traditional SNs	Share activity information with friends and followers
Activity-based SNs	Connect people based on their activity profiles
Place & event detection	Identify popular areas for exercise and recreation

Table 1: Summary of activity recognition applications (with example applications)

Jeffrey W. Lockhart, Tony Pulickal, and Gary M. Weiss. 2012. Applications of mobile activity recognition. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp '12). ACM, New York, NY, USA, 1054-1058. DOI=http://dx.doi.org/10.1145/2370216.2370441

USC-HAD dataset

- Mi Zhang and Alexander A. Sawchuk. 2012. USC-HAD: a daily activity dataset for ubiquitous activity recognition using wearable sensors. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp '12). ACM, New York, NY, USA, 1036-1043. DOI=http://dx.doi.org/10.1145/2370216.2370438

- Dealing with time series data!

- Our task: classify 12 daily activities based on IMU sensor readings (accelerometer, gyroscope)

USC-HAD dataset: data collection

- Device Type: MotionNode
- Sampling rate: 100Hz
- Accelerometer range: +-6g
- Gyroscope range: +-500dps

Figure 2. MotionNode sensing platform

- For sensor_readings field, it consists of 6 readings (left to right):

1. acc_x, w/ unit g (gravity)
2. acc_y, w/ unit g
3. acc_z, w/ unit g
4. gyro_x, w/ unit dps (degrees per second)
5. gyro_y, w/ unit dps
6. gyro_z, w/ unit dps

Figure 4. During data collection, a single MotionNode is packed firmly into a mobile phone pouch and attached to the subject's front right hip

USC-HAD dataset: data format

Each activity trial is stored in an .mat file.

The naming convention of each .mat file is defined as:

a"m"t"n".mat, where

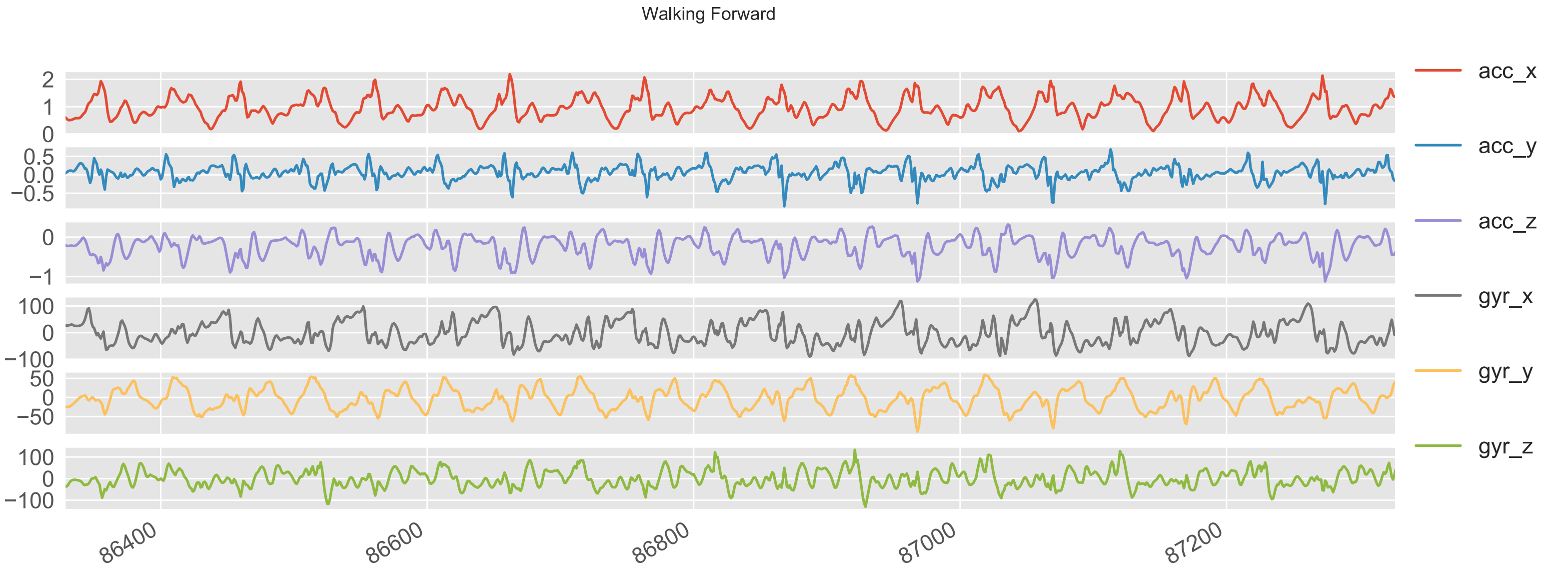
"a" stands for activity
"m" stands for activity number
"t" stands for trial
"n" stands for trial number

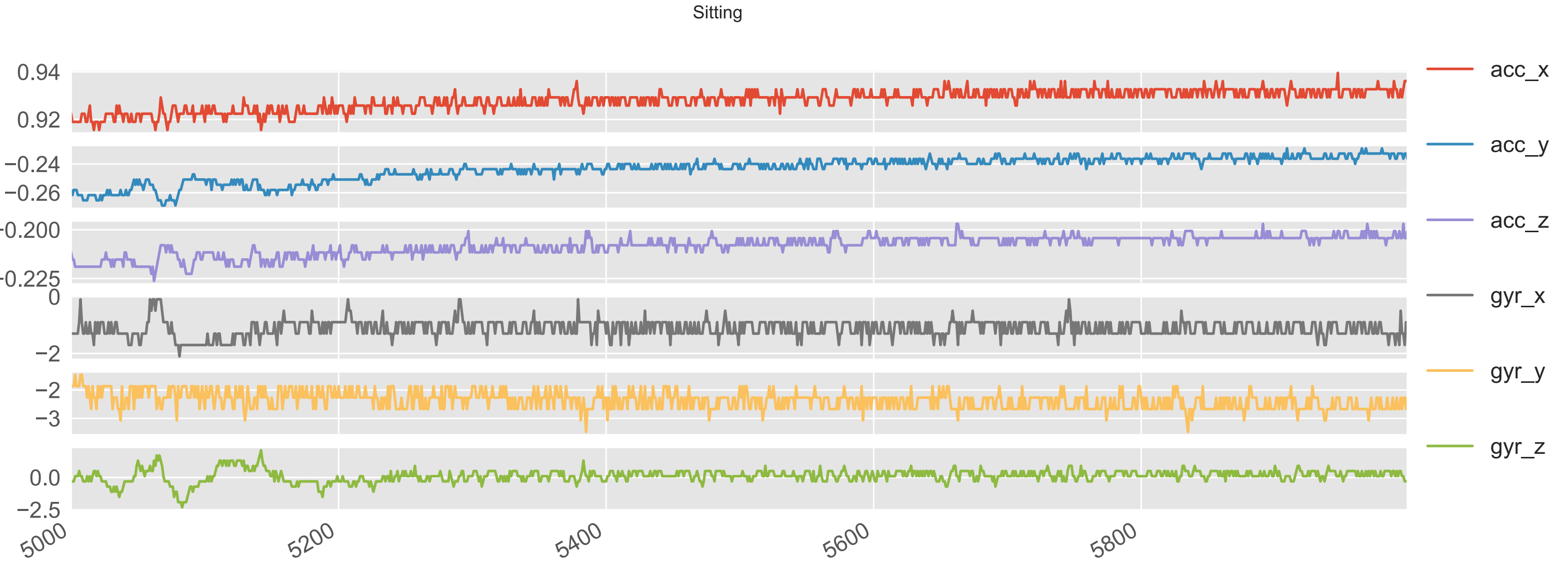
Each .mat file contains 13 fields:

1. title: USC Human Motion Database
2. version: it is version 1.0 for this first round data collection
3. date
4. subject number
5. age
6. height
7. weight
8. activity name
9. activity number
10. trial number
11. sensor_location
12. sensor_orientation
13. sensor_readings

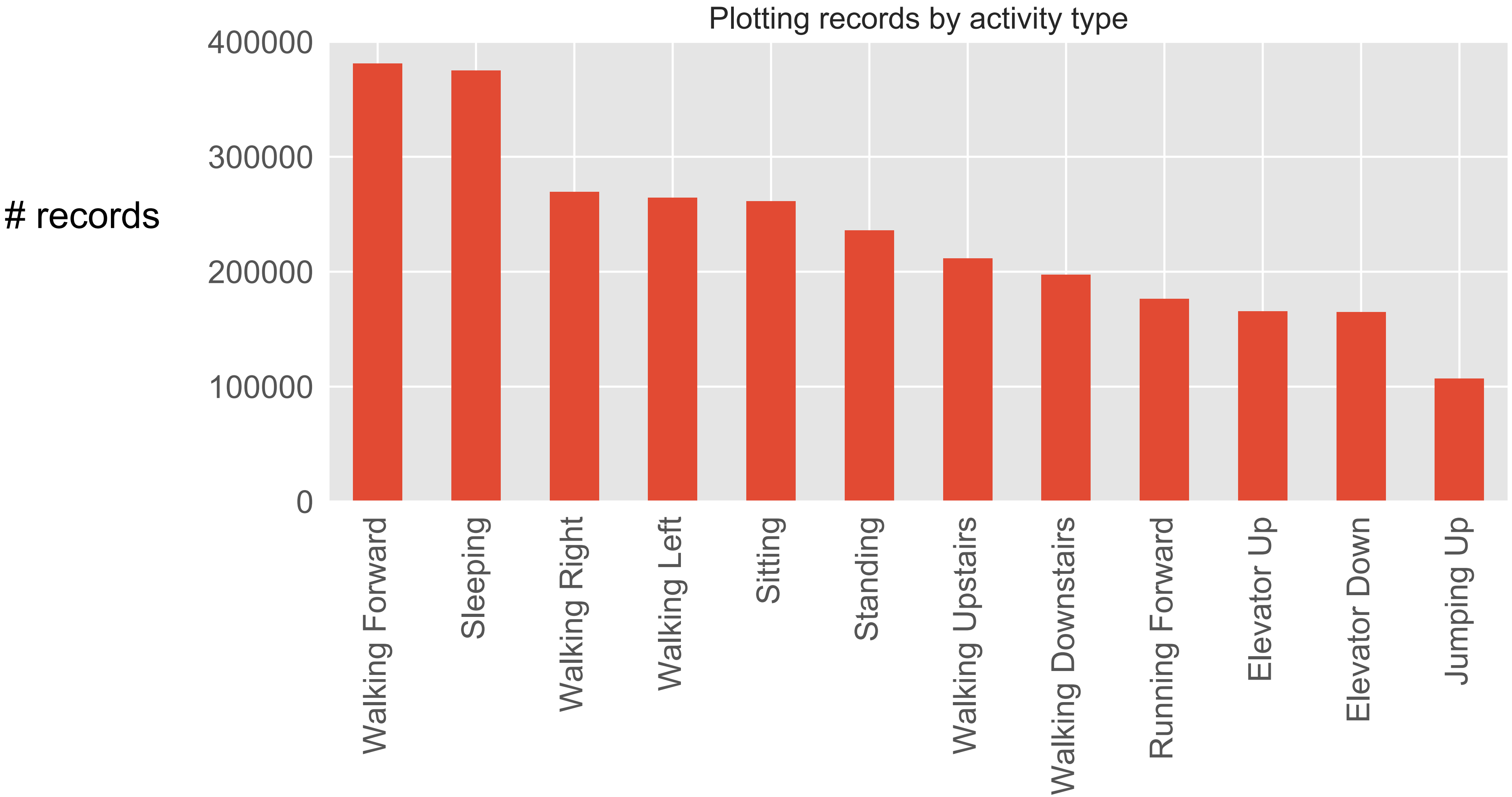
USC-HAD dataset: activities

1. Walking Forward
2. Walking Left
3. Walking Right
4. Walking Upstairs
5. Walking Downstairs
6. Running Forward
7. Jumping Up
8. Sitting
9. Standing
10. Sleeping
11. Elevator Up
12. Elevator Down

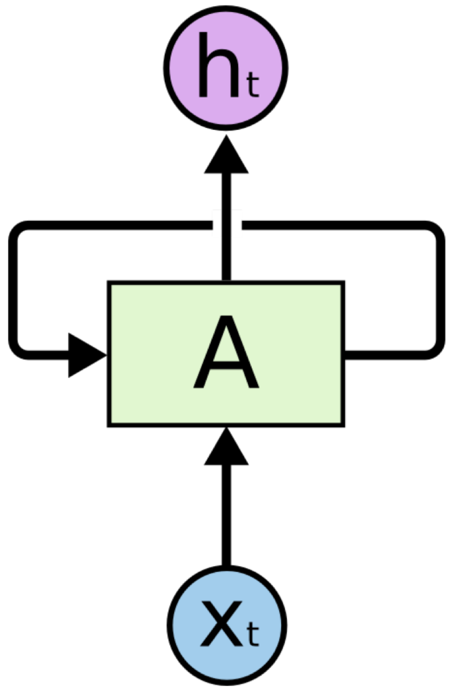




USC-HAD dataset: class distribution

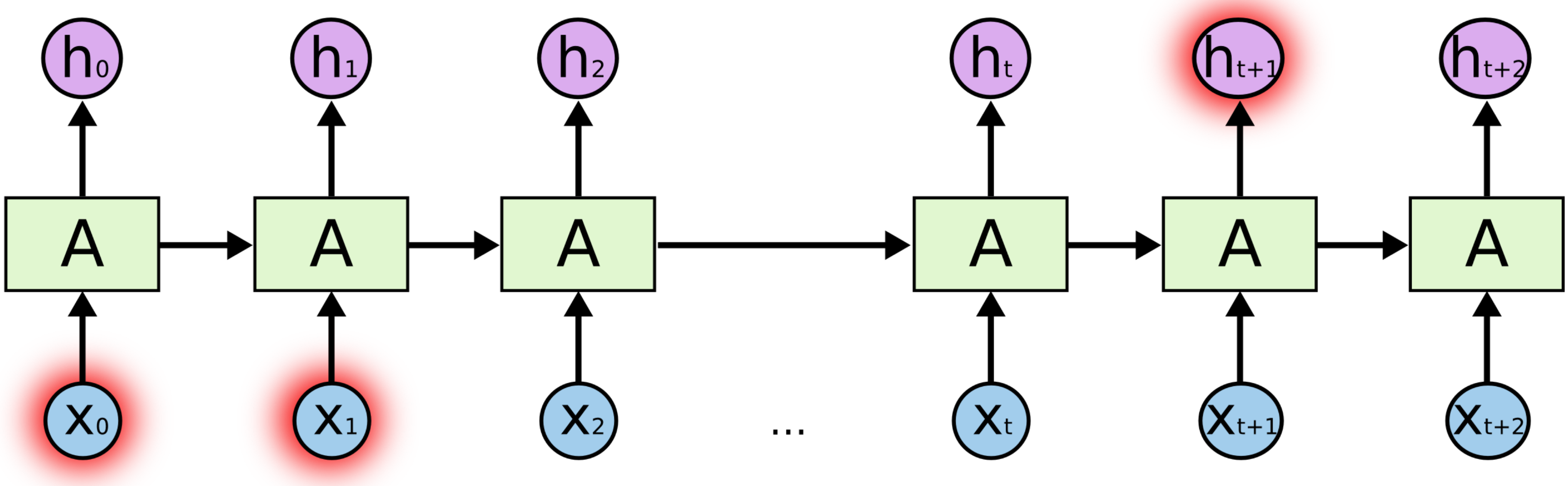


Recurrent Neural Networks (RNNs)



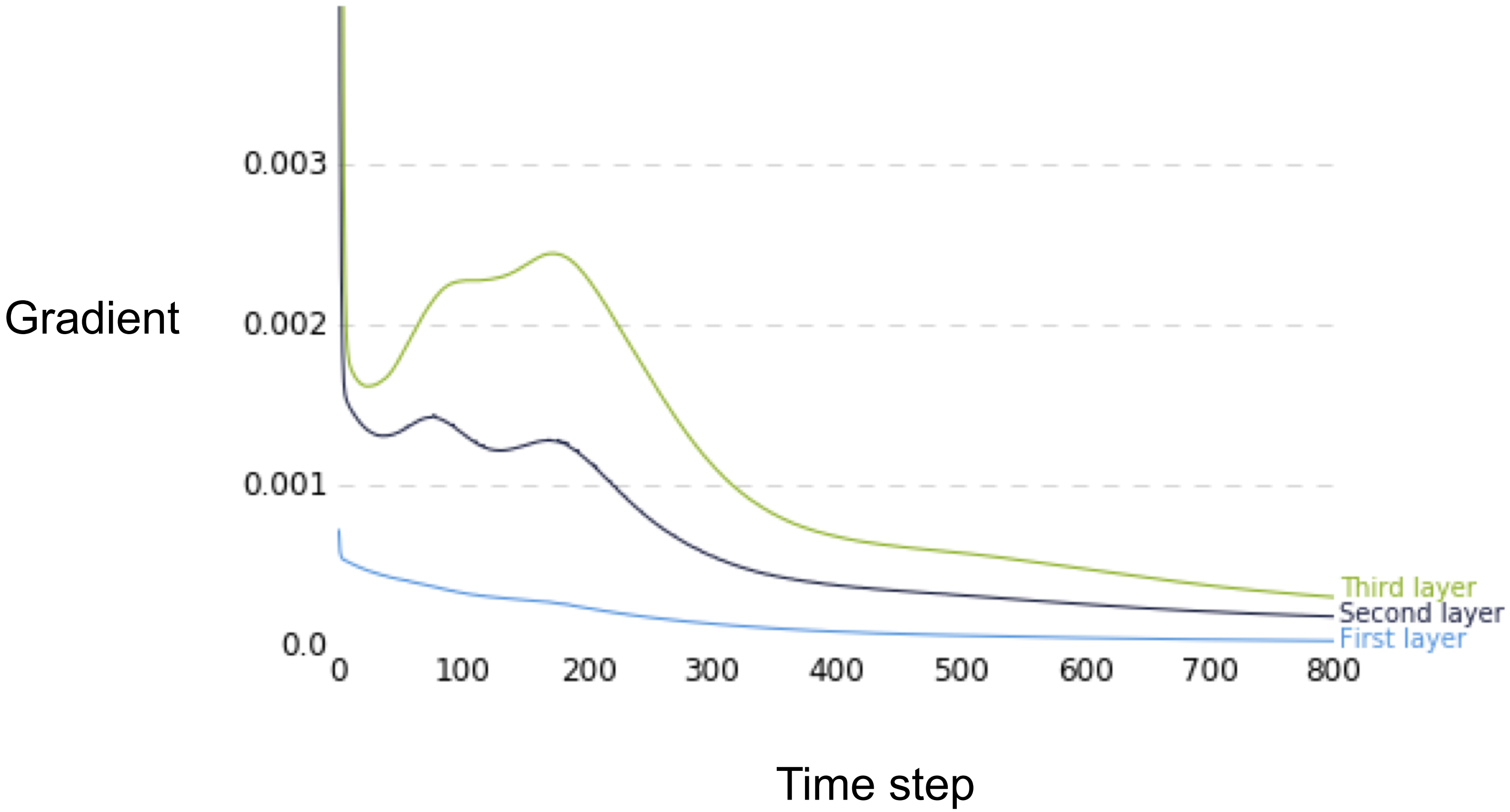
http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Problem with RNNs



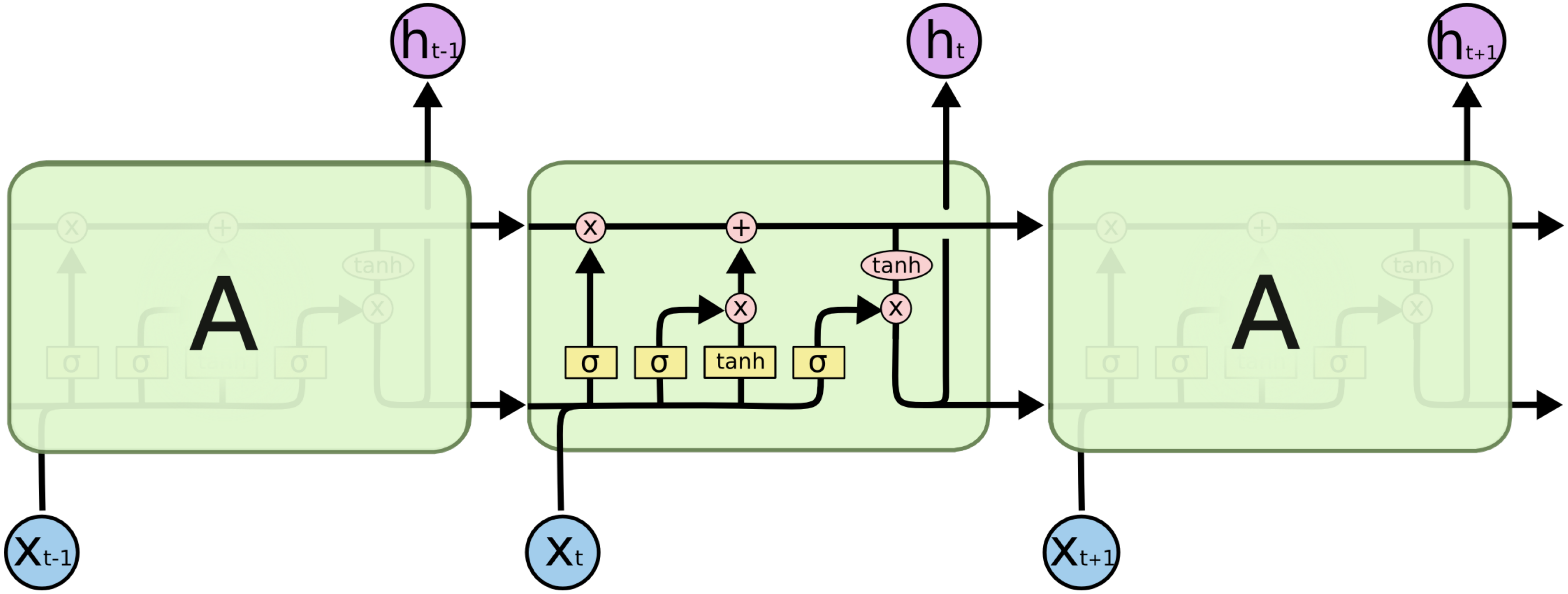
http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Problem with RNNs: vanishing gradient on MNIST



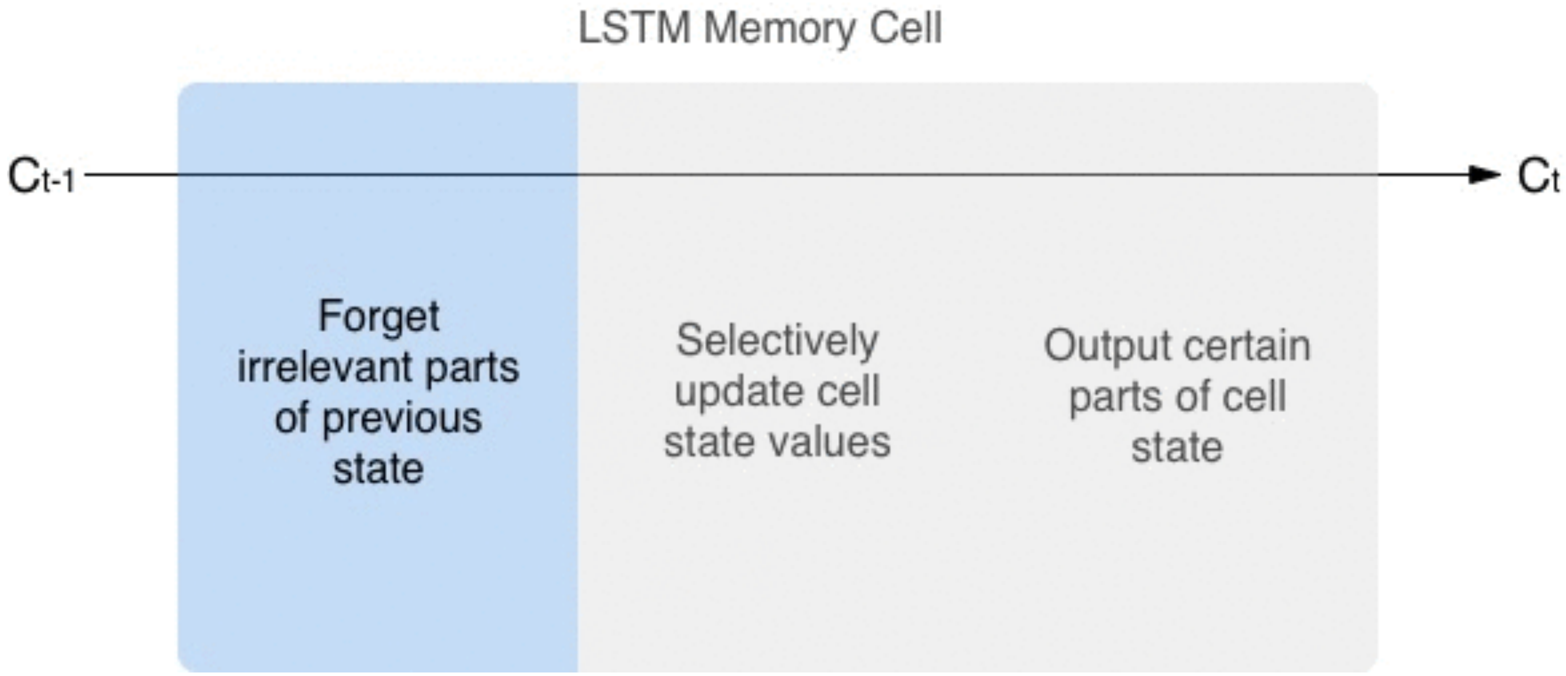
http://harinisuresh.com/2016/10/09/lstms/

Long-short Term Memory (LSTM) networks



http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Long-short Term Memory (LSTM) networks



http://harinisuresh.com/2016/10/09/lstms/

Mixed CNN + LSTM

- Conv net: learn local features automatically
- LSTM: account for time-series dependencies

- ConvLSTM2D: similar to an LSTM layer, but the input transformations and recurrent transformations are both convolutional

➔ better capture spatiotemporal correlations

- Morales, F. J. O. & Roggen, D. (2016). Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition.. *Sensors*, 16, 115.
- https://www.tensorflow.org/api_docs/python/tf/keras/layers/ConvLSTM2D