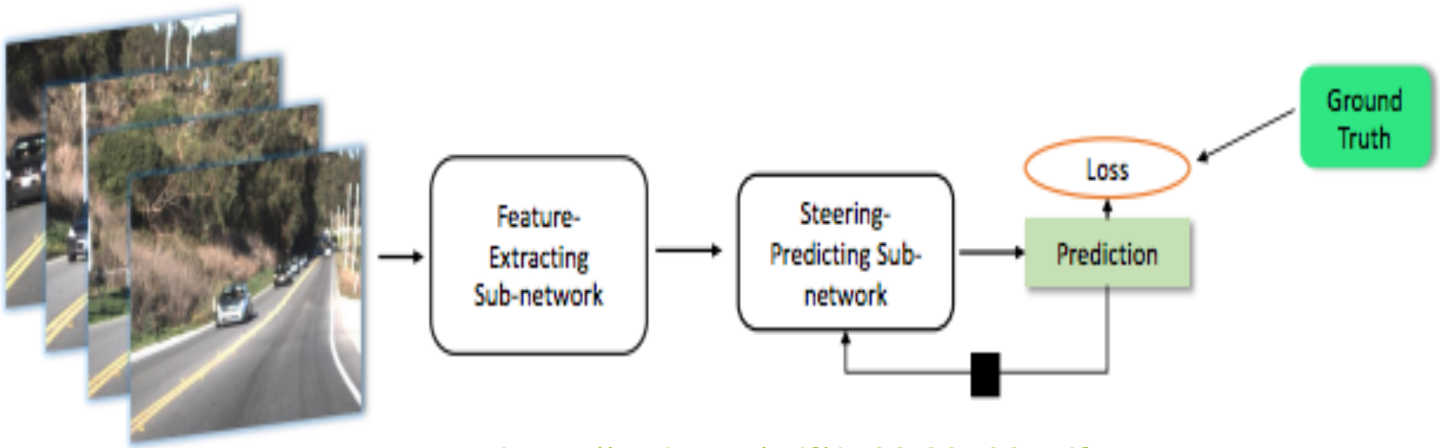


https://arxiv.org/pdf/1708.03798.pdf

Deep Sequence Learning

Jay Urbain, PhD

urbain@msoe.edu

LinkedIn https://www.linkedin.com/in/jayurbain/
Github https://github.com/jayurbain, https://github.com/jayurbain/DeepSequenceLearningIntro
Teaching http://jayurbain.com/msoe/index.html

Prologue

- Traditional machine learning methods require significant feature engineering to help capture the underlying concepts and relations in data.
- Feature engineering is especially common in multi-variate sequence learning tasks to capture spatial or temporal relations in data,
- Deep learning methods can learn hierarchical representations of concepts as part of the learning process.
- Can we learn effective end-to-end models using deep learning for sequence learning applications?

Difference in Workflow

Classic Machine Learning
[1990 : now]

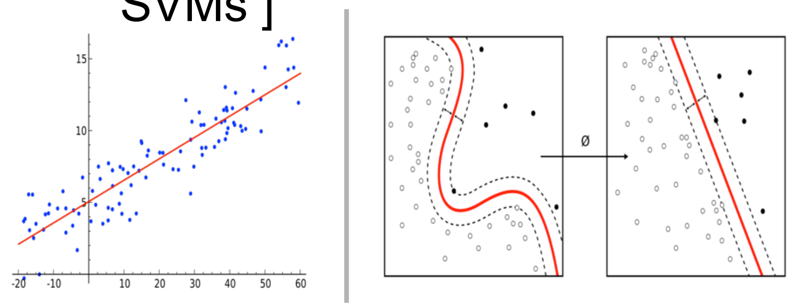
Inp
ut

Hand
Designed
Features

Model /
Mapping

Out
put

Examples [Regression and
SVMs]



Deep/End-to-End Learning
[2012 : now]

Inp
ut

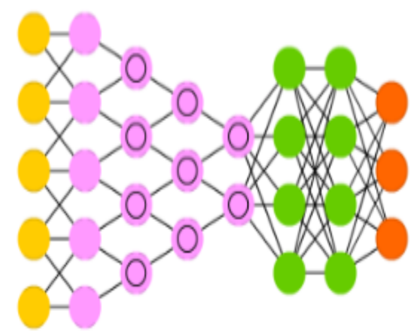
Simple
Feature
s

Comple
x
Feature
s

Mode
l/
Mapp
ing

Out
put

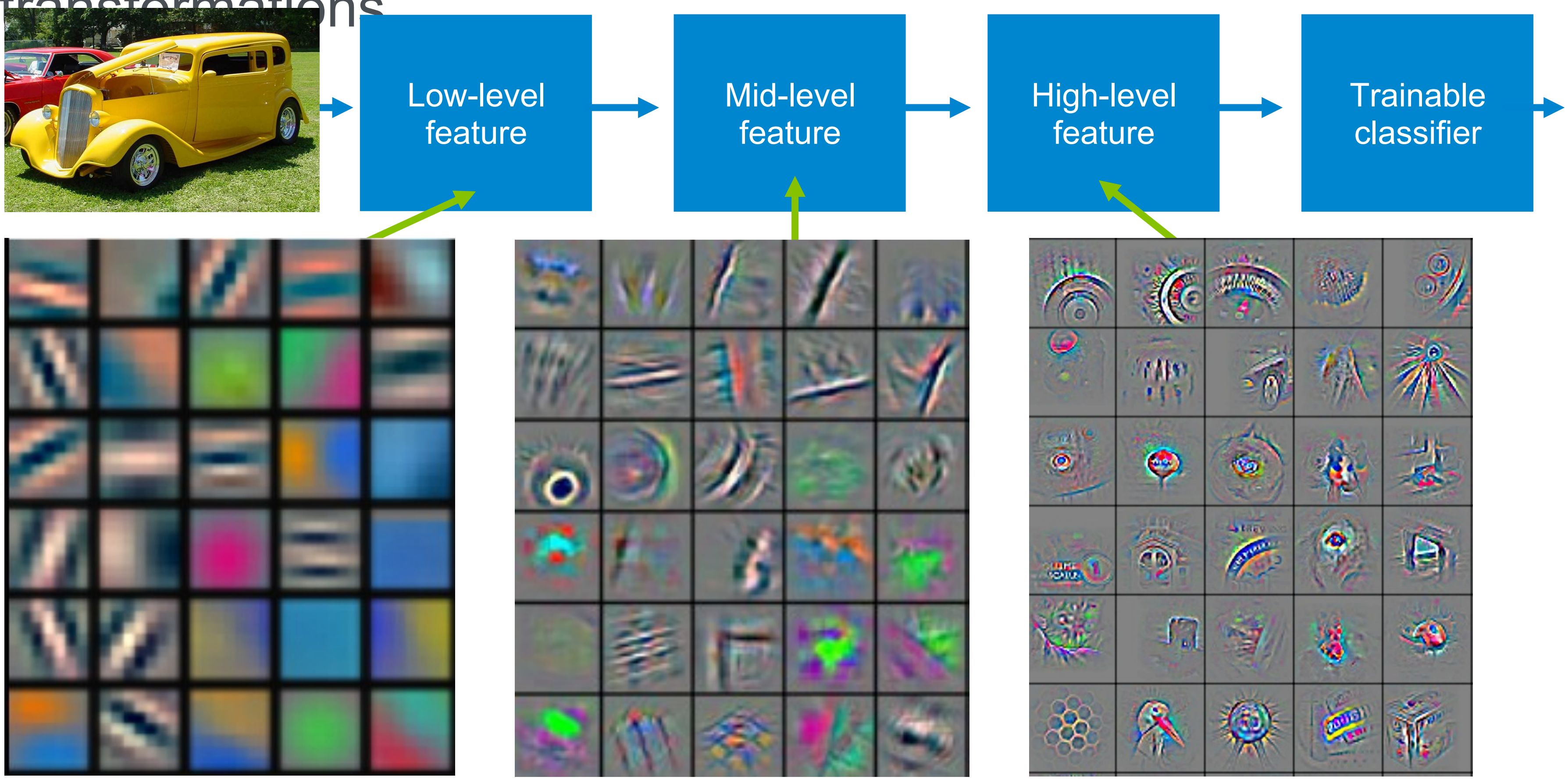
Example [Conv Net]



Machine learning workflow shifts from engineering features for “shallow” mode
to architecting deep learning models with the ability to learn hierarchical repres

Deep learning = learning hierarchical representations

It's deep if it has more than one stage of non-linear feature transformations



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Outline

- What is a sequence?
- Applications
- Traditional ML methods for sequences
- Deep learning for sequences
- RNN Models
- ConvNet Models
- Hybrid Models
- Coding example
- Future directions

Sequence Learning

Sequence data

- Data with a sequential dependency across space or time.
- Not IID.

Sequence prediction

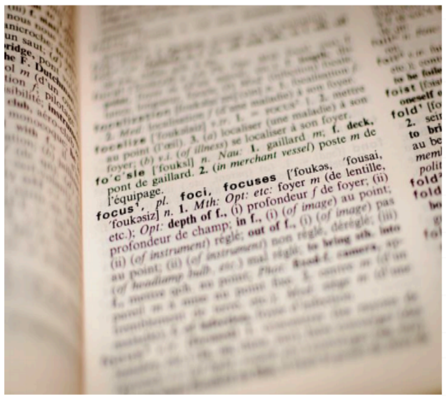
- Use historical sequence data, predict the next value or values in the sequence.
- Example: predict next word in a sentence, future price in a time series of stock prices.

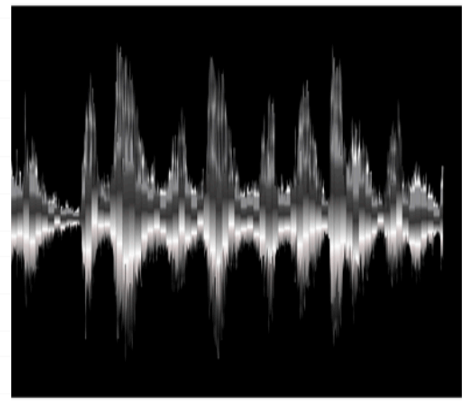
Sequence classification

- Using some sequence of data over space or time, predict a category for the sequence.
- Example: activity recognition, arrhythmia classification, named entity recognition, text classification, text

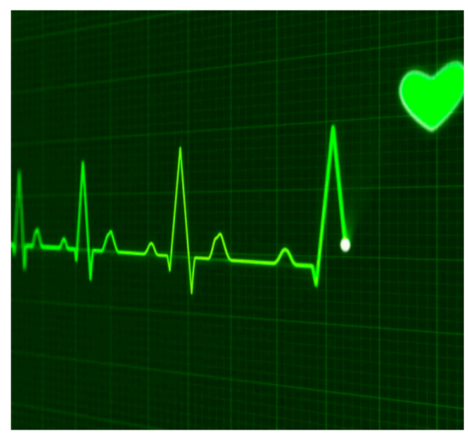
Sequence Data

Text, video, audio:





Time series: finance, industry, medicine



Sequence classification: patient de-identification
https://cis.ctsi.mcw.edu/deid/

Input text:

Jay Urbain, jay.urbain@gmail.com, born December 6, 2156 is an elderly caucasian male suffering from illusions of grandeur and LBP. He is married to Kimberly Urbain, who is much better looking. Patient father, Francis Urbain has a history of CAD and DM. Jay has been prescribed meloxicam, and venti americano. He lives at 9050 N. Tennyson Dr., Disturbia, WI with his wife and golden retriever Mel. You can reach him at 414-745-5102.

Data Format Pretty Print

Submit

Parsed results:

[PERSON] [PERSON], [xxx@xxx.xxx] , born [12_16_2156] is an elderly caucasian male suffering from illusions of grandeur and LBP. He is married to [PERSON] [PERSON], who is much better looking. Patient father, [PERSON] [PERSON] has a history of CAD and DM. [PERSON] has been prescribed meloxicam, and venti americano. He lives at [xxxxx x. xxxxx] Dr., Disturbia, WI with his wife and golden retriever [PERSON]. You can reach him at [xxx_xxx_xxxx] .

Sequence classification: medical named entity recognition

https://cis.ctsi.mcw.edu/nlp/

Parsed results:

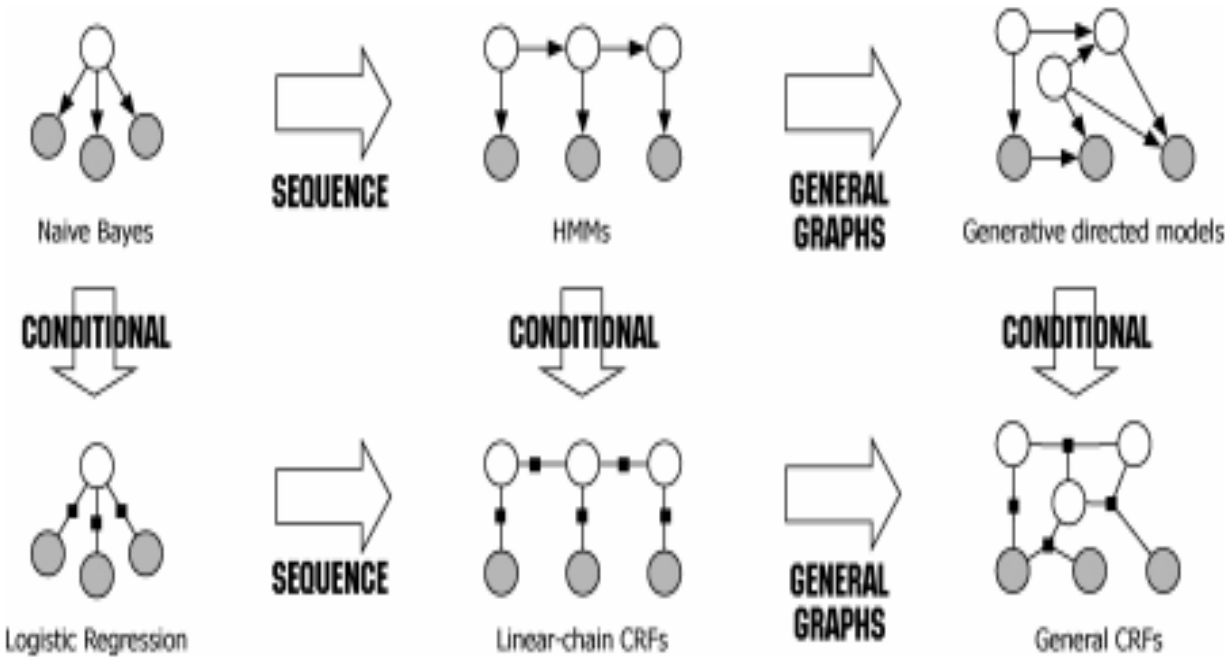
```
SENTENCE:  Jay Urbain is  an elderly caucasian male suffering from illusions of grandeur and low back   pain  .
           NNP  NNP   VBZ DT   JJ       JJ      NN     VBG     IN     NNS     IN    NN     CC  JJ   NN     NN
                                                    |=======|     |=======|    |======|           |======|
                                                      Event       Disorder      Event             Finding
                                                                  C0020903                        C0030193
                                                                                             |===========|
                                                                                               Finding
                                                                                               C0004604
                                                                                         |===============|
                                                                                             Finding
                                                                                             C0024031

SENTENCE:  Patient has a  family history  of   CAD    and DM.
             NN    VBZ DT   NN      NN    IN    NN     CC  NN
                                 |======|    |======|
                                 Finding     Disorder
                                 C0262926    C1956346
                          |=============|
                              Finding
                              C0241889
TLINKS:    history CONTAINS CAD

SENTENCE:  Prescribed meloxicam, and venti americano.
              VBN        NN      CC    JJ      NN
                     |=======|            |=======|
                        Drug                Event
                      C0083381
```

Traditional NLP Sequence Models: HMM, MEMM, CRF

http://homepages.inf.ed.ac.uk/csutton/publications/crftut-fnt.pdf



Traditional Models: HMM, MEMM, CRF

http://homepages.inf.ed.ac.uk/csutton/publications/crftut-fnt.pdf

Linear chain CRF for sequence class. Named

Definition 2.2. Let Y, X be random vectors, $\theta = \{\theta_k\} \in \Re^K$ be a parameter vector, and $\mathcal{F} = \{f_k(y, y', \mathbf{x}_t)\}_{k=1}^K$ be a set of real-valued feature functions. Then a *linear-chain conditional random field* is a distribution $p(\mathbf{y}|\mathbf{x})$ that takes the form:

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{t=1}^{T} \exp\left\{ \sum_{k=1}^{K} \theta_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\}, \qquad (2.18)$$

where $Z(\mathbf{x})$ is an input-dependent normalization function

$$Z(\mathbf{x}) = \sum_{\mathbf{y}} \prod_{t=1}^{T} \exp\left\{ \sum_{k=1}^{K} \theta_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\}. \qquad (2.19)$$

W=v	$w_t = v$	$\forall v \in \mathcal{V}$
T=j	part-of-speech tag for w_t is j (as determined by an automatic tagger)	$\forall$POS tags j
P=I$-j$	w_t is part of a phrase with syntactic type j (as determined by an automatic chunker)	
Capitalized	w_t matches `[A-Z][a-z]+`	
Allcaps	w_t matches `[A-Z][A-Z]+`	
EndsInDot	w_t matches `[^\.]+.*\.`	
	w_t contains a dash	
	w_t matches `[A-Z]+[a-z]+[A-Z]+[a-z]`	
Acro	w_t matches `[A-Z][A-Z\\.]*\\.[A-Z\\.]*`	
Stopword	w_t appears in a hand-built list of stop words	
CountryCapital	w_t appears in list of capitals of countries	
$\vdots$	many other lexicons and regular expressions	
$q_k(\mathbf{x}, t+\delta)$ for all k and $\delta \in [-1, 1]$		

CRF libraries

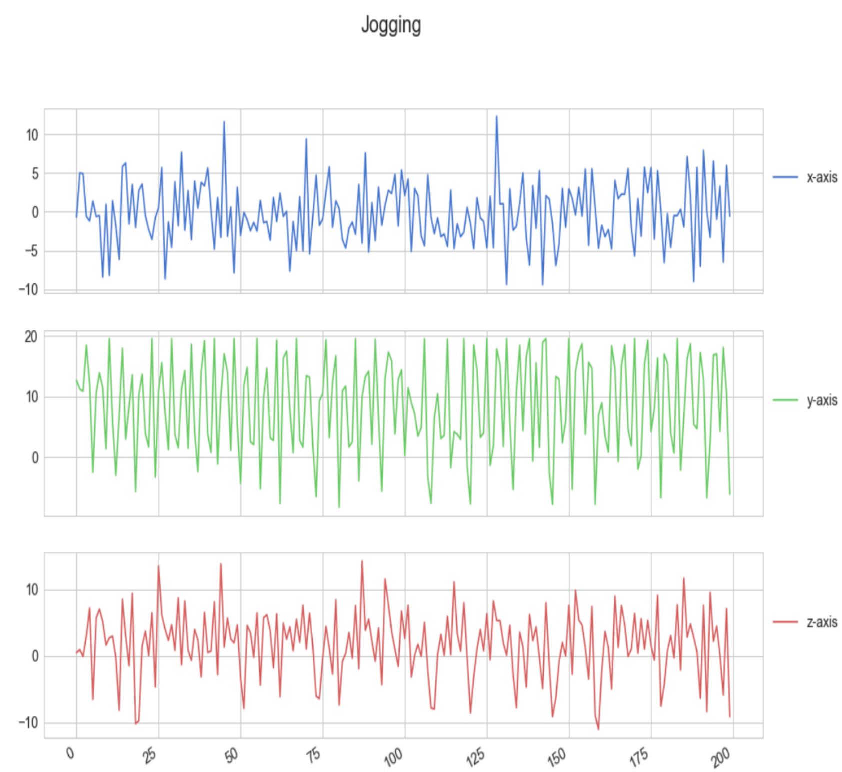
Stanford has a very nice Java implementation and pre-trained models for traditional entities, i.e., person, location, organization etc.

CRF++	http://crfpp.sourceforge.net/
MALLET	http://mallet.cs.umass.edu/
GRMM	http://mallet.cs.umass.edu/grmm/
CRFSuite	http://www.chokkan.org/software/crfsuite/
FACTORIE	http://www.factorie.cc

Sequence classification: activity recognition

Temporal sensor analysis and activity tracking

	user	activity	timestamp	x-axis	y-axis	z-axis
0	33	Jogging	49105962326000	-0.694638	12.680544	0.503953
1	33	Jogging	49106062271000	5.012288	11.264028	0.953424
2	33	Jogging	49106112167000	4.903325	10.882658	-0.081722
3	33	Jogging	49106222305000	-0.612916	18.496431	3.023717
4	33	Jogging	49106332290000	-1.184970	12.108489	7.205164



Traditional models: activity recognition

http://www.cis.fordham.edu/wisdm/public_files/sensorKDD-2010.pdf

Average[3]: Average acceleration (for each axis)

Standard Deviation[3]: Standard deviation (for each axis)

Average Absolute Difference[3]: Average absolute difference between the value of each of the 200 readings within the ED and the mean value over those 200 values (for each axis)

Average Resultant Acceleration[1]: Average of the square roots of the sum of the values of each axis squared $\sqrt{(x_i^2 + y_i^2 + z_i^2)}$ over the ED

Time Between Peaks[3]: Time in milliseconds between peaks in the sinusoidal waves associated with most activities (for each axis)

Binned Distribution[30]: We determine the range of values for each axis (maximum – minimum), divide this range into 10 equal sized bins, and then record what fraction of the 200 values fell within each of the bins.

Table 2: Accuracies of Activity Recognition

	% of Records Correctly Predicted			
	J48	**Logistic Regression**	**Multilayer Perceptron**	**Straw Man**
Walking	89.9	**93.6**	91.7	37.2
Jogging	96.5	98.0	**98.3**	29.2
Upstairs	59.3	27.5	**61.5**	12.2
Downstairs	**55.5**	12.3	44.3	10.0
Sitting	**95.7**	92.2	95.0	6.4
Standing	**93.3**	87.0	91.9	5.0
Overall	85.1	78.1	**91.7**	37.2

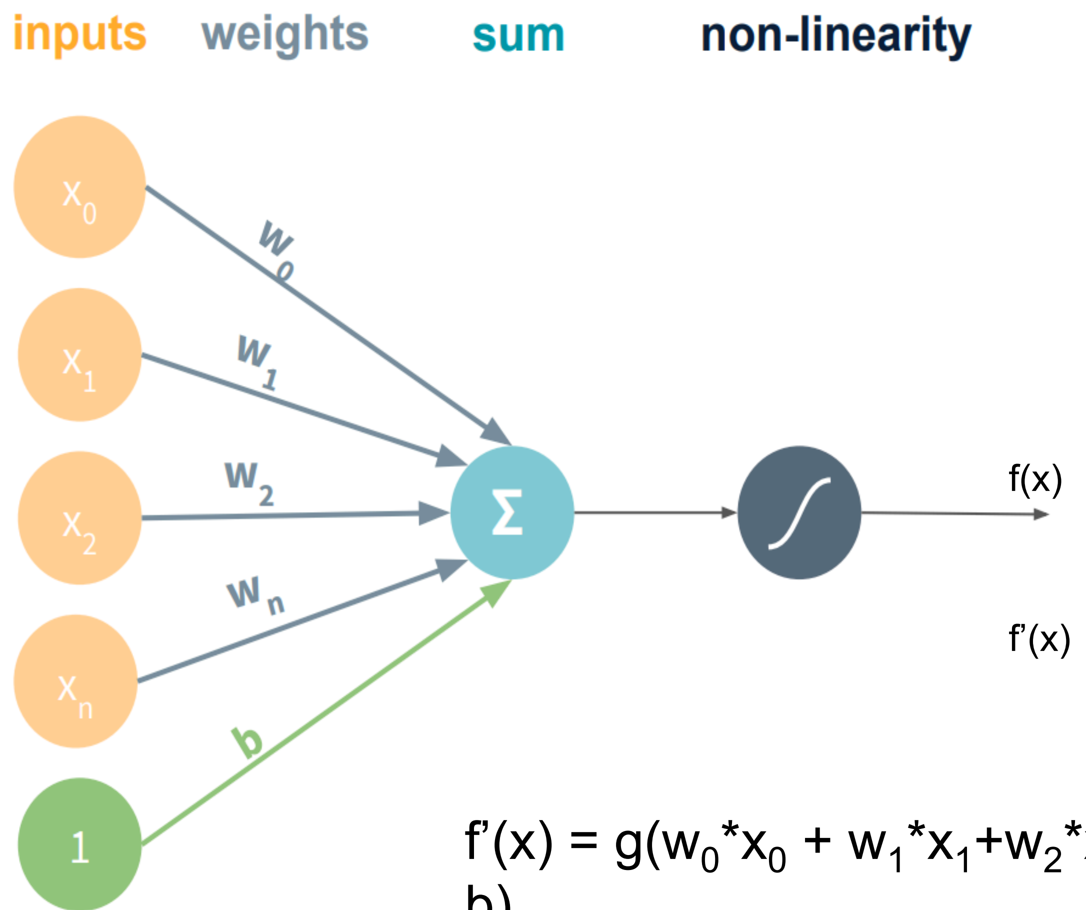
	user	activity	timestamp	x-axis	y-axis	z-axis
0	33	Jogging	49105962326000	-0.694638	12.680544	0.503953
1	33	Jogging	49106062271000	5.012288	11.264028	0.953424
2	33	Jogging	49106112167000	4.903325	10.882658	-0.081722
3	33	Jogging	49106222305000	-0.612916	18.496431	3.023717
4	33	Jogging	49106332290000	-1.184970	12.108489	7.205164

Traditional models: activity recognition

Note: Improved features using traditional machine learning models (SVM, RF, LR) can yield an accuracy in the ~92% range (Quihao Jin, Jay Urbain):

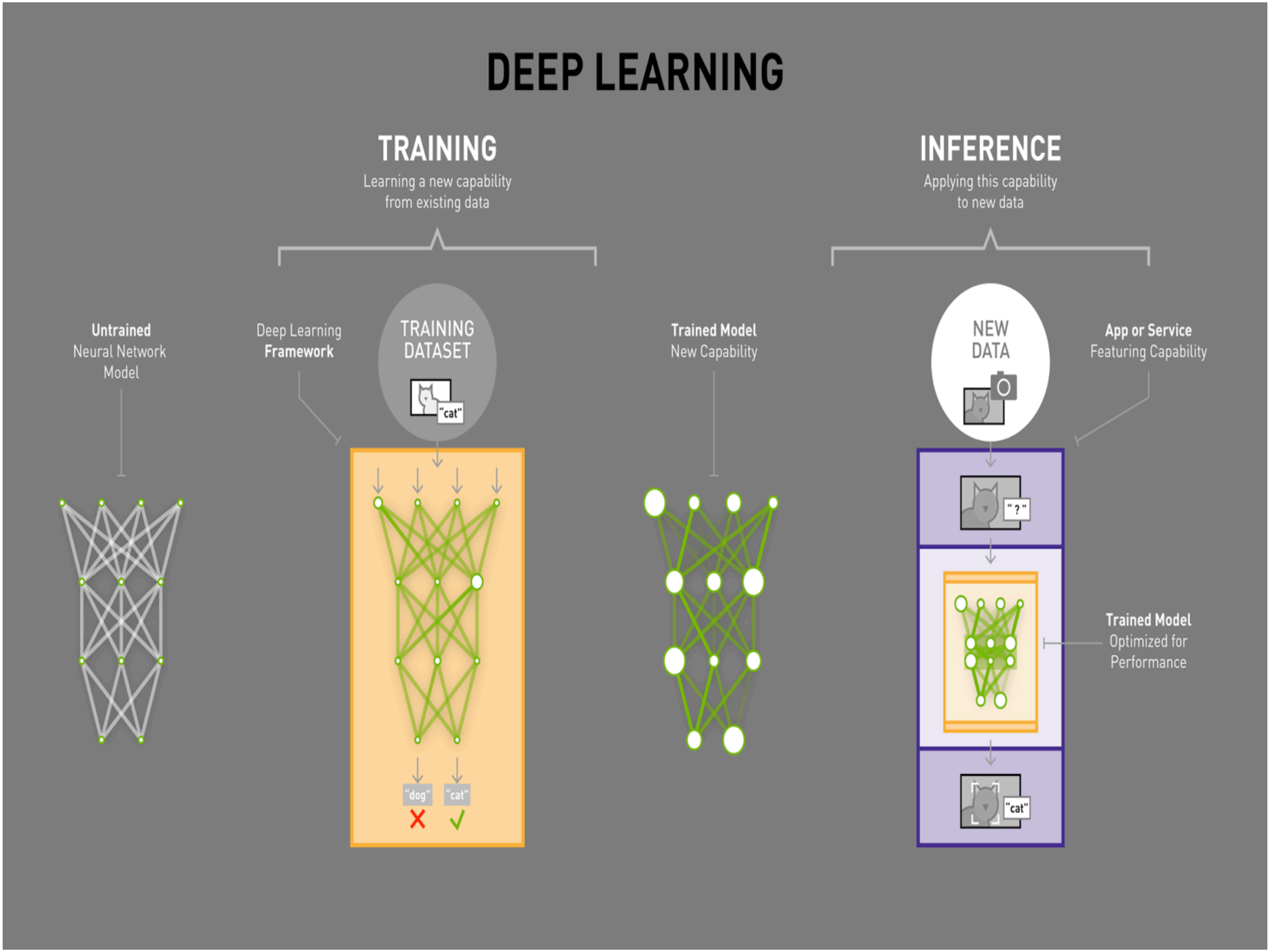
- 3-sec mean
- 5-sec mean
- Median filtering
- Acceleration axis deltas
- FFT – Fast Fourier Transform

Deep Learning - Perceptron



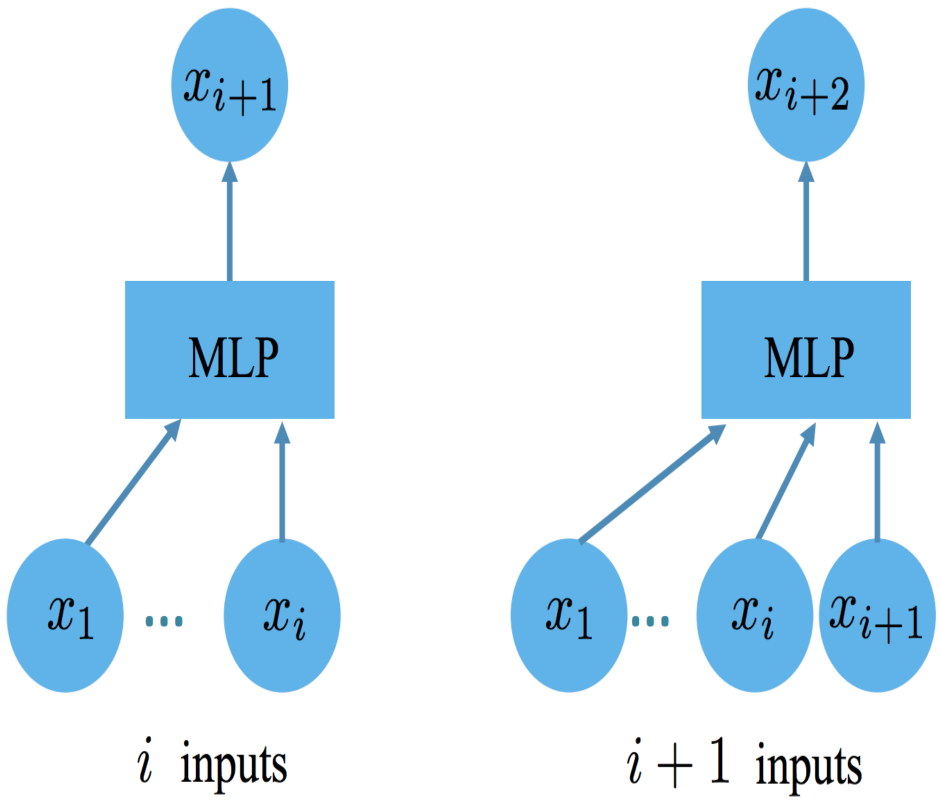
$f'(x) = g(w_0 * x_0 + w_1 * x_1 + w_2 * x_2 + \ldots + w_n * x_n + b)$

g(x) – non-linear activation function


DEEP LEARNING
TRAINING
Learning a new capability from existing data
INFERENCE
Applying this capability to new data
Untrained Neural Network Model
Deep Learning Framework
TRAINING DATASET
"cat"
"dog"
"cat"
Trained Model New Capability
NEW DATA
App or Service Featuring Capability
" ? "
Trained Model Optimized for Performance
"cat"

Why not MLP for sequences?

Problem 1: arbitrary length of sequences:

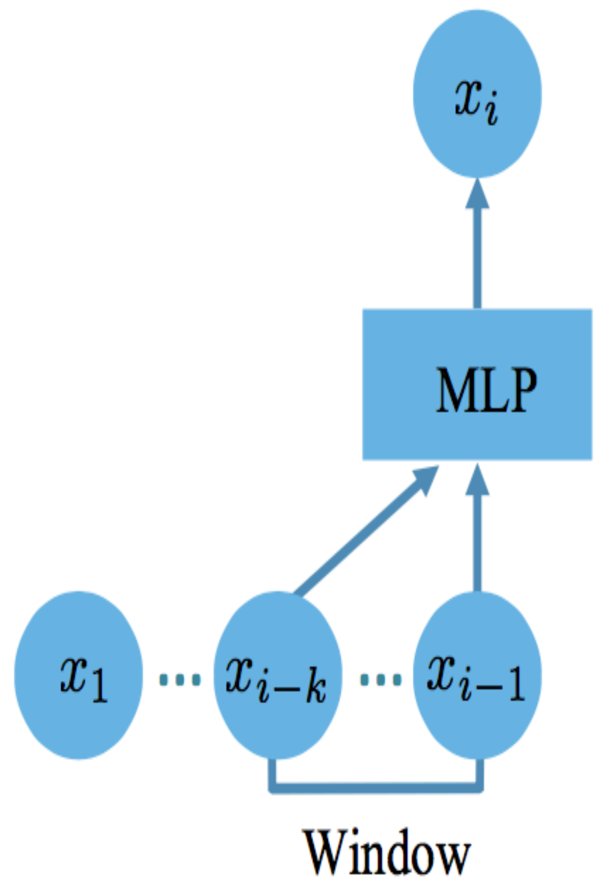


Why not MLP for sequences?

Problem 1: arbitrary length of sequences:

Can use a **window of fixed size**:
- Just a heuristic – need to choose window size.
- Some tasks require wide window and therefore there is a problem with the large number of parameters (weights)

Question
- How many weights are there in the first layer of the MLP?
- Hidden layer neurons: 100
- Input window width: 100
- Word embeddings size: 100

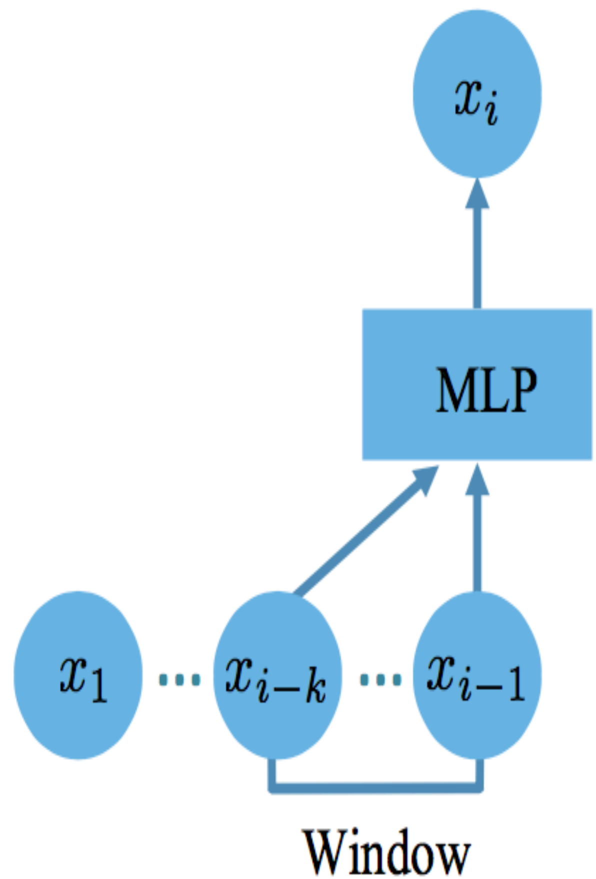


Why not MLP for sequences?

Problem 1: arbitrary length of sequences:

Can use a window of fixed size:

- Just a heuristic – need to choose window size.
- Some tasks require wide window and therefore there is a problem with the large number of parameters (weights)



Question

- How many weights are there in the first layer of the MLP?
- Hidden layer neurons: 100
- Input window width: 100
- Word embeddings size: 100

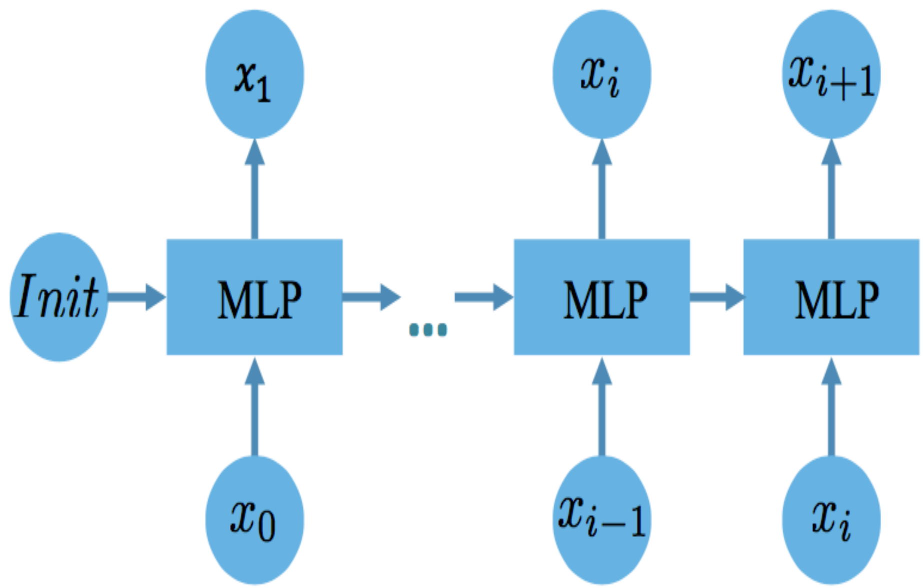
Consider recurrent architecture

Fixes problem 1: arbitrary length sequences.

- Fixed number of inputs at each time step.
- At the first step we use some initial vector as an input from previous time step.

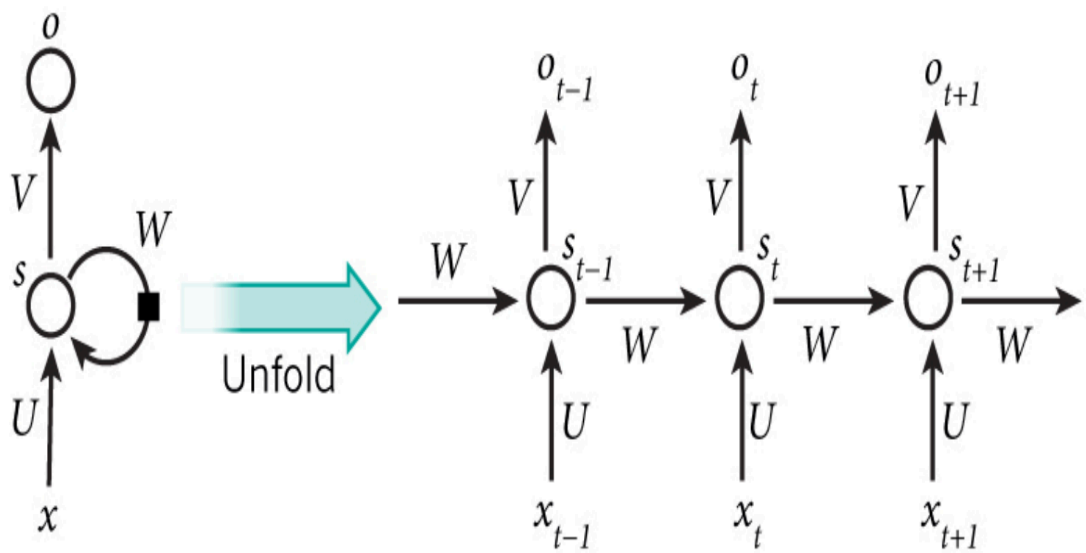
Problem #2: number of parameters to train.

Solution: All the parameters of an MLP are shared across the different time steps so we



Folded/unfolded: Recurrent Neural Network

- RNNs have connections between units along a sequence.
- RNN's use the hidden state from the last time step and the input at the current step to make predictions.
- Allow RNN's can capture dynamic temporal behavior for a time sequence.



SKIP - Long Short Term Memory (LSTM)

- RNN’s have trouble with long-term dependencies.
- Long chain of differentiation when back propagating can cause vanishing gradients
- Compose RNN with LSTM cells for "remembering" cell state over arbitrary time inte
- Forget gate – what to fo

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

- Input gate – what to sa

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

- Update cell state:

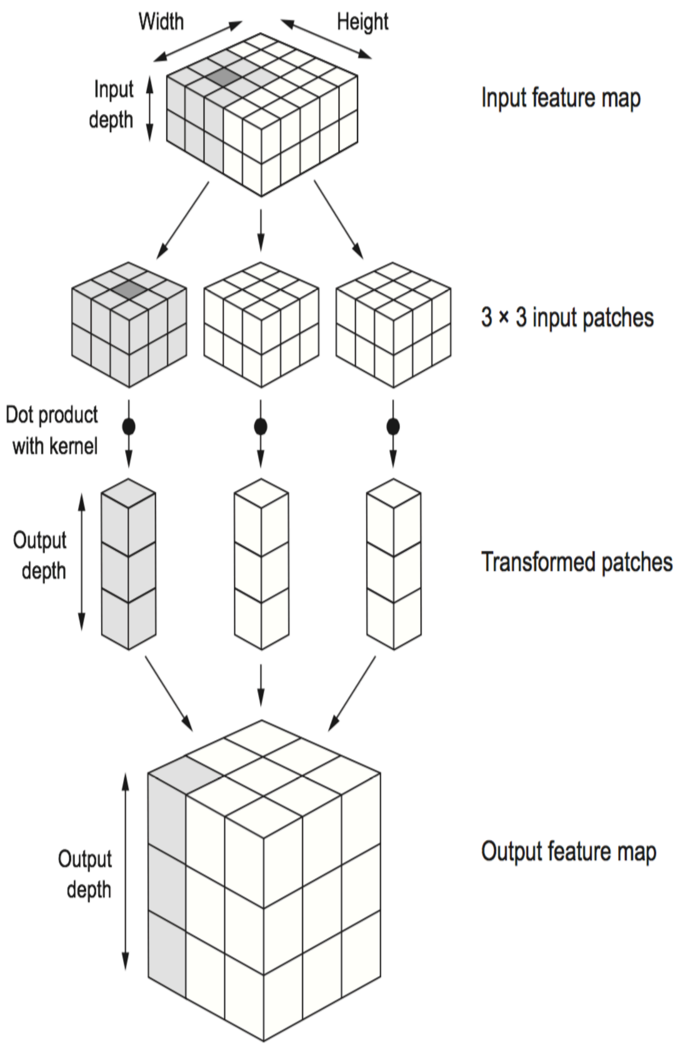
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$o_t = \sigma\left(W_o \ [h_{t-1}, x_t] \ + \ b_o\right)$$

- Output:

$$h_t = o_t * \tanh\left(C_t\right)$$

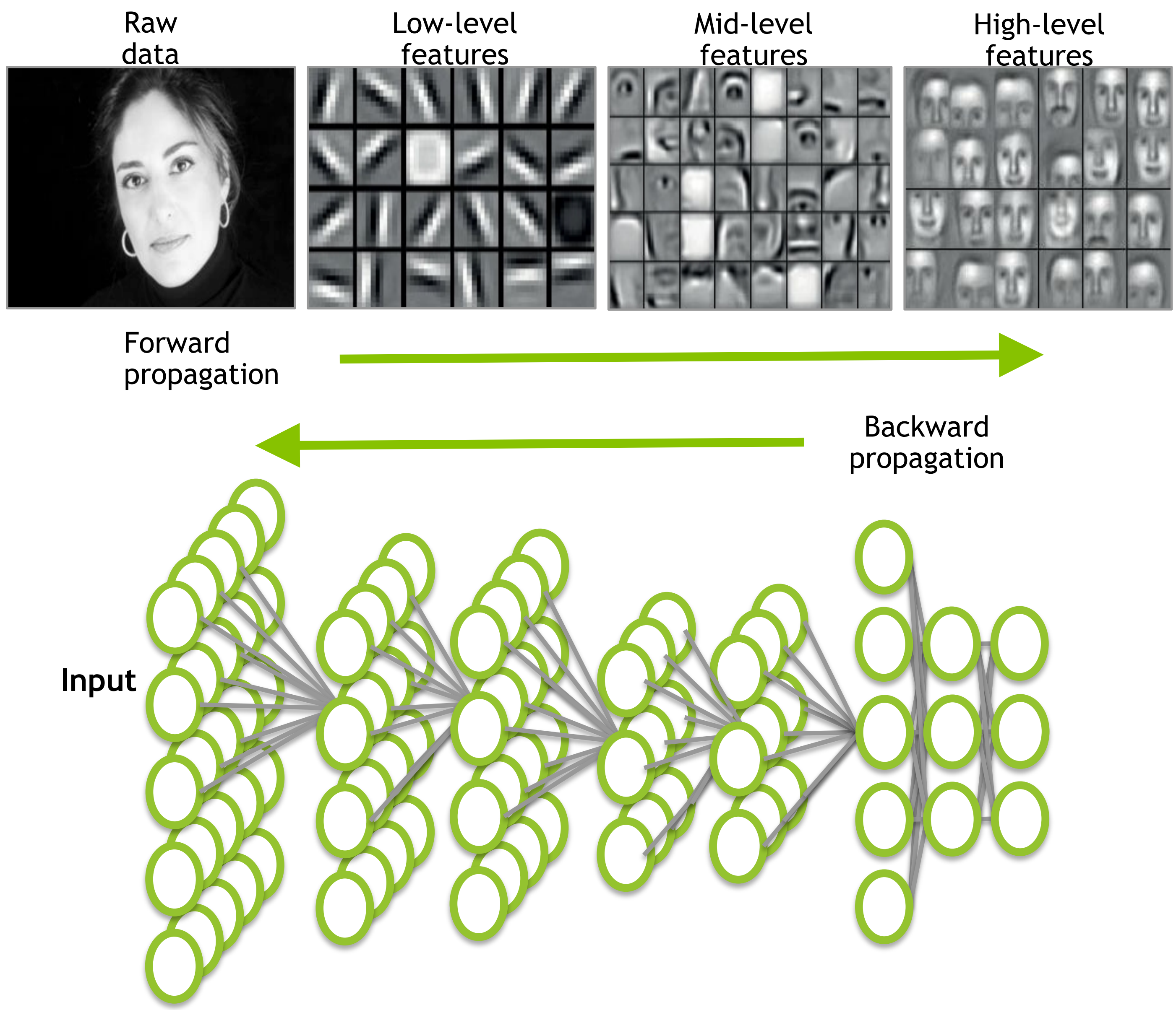
Convolution Neural Network (ConvNet)



FRANÇOIS CHOLLET, 2018

- A ConvNet is made up of Layers: convolution, pooling, fully connected.
- Each layer transforms an input 3D volume to an output 3D volume.
- Learns local parameters in input space using a stack of (e.g., 3x3) convolution filters
- The patterns the filters learn are translation invariant.
- Learn spatial hierarchies of patterns by adding conv and pooling layers. E.g., edges, shapes, motifs.

DEEP LEARNING APPROACH - TRAINING



Process

- Forward propagation yields an inferred label for each training image
- Loss function used to calculate difference between known label and predicted label for each image
- Weights are adjusted during backward propagation
- Repeat the process

Experiment #1 – WISDM Activity Dataset

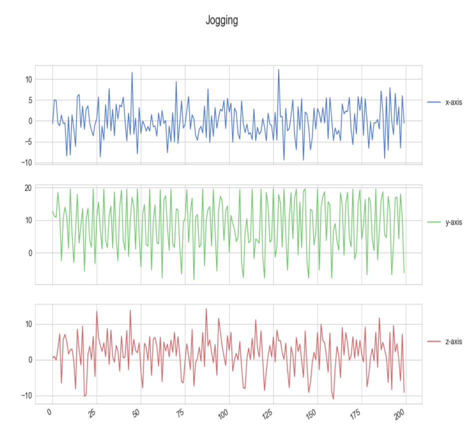
Temporal sensor analysis and activit

	user	activity	timestamp	x-axis	y-axis	z-axis
0	33	Jogging	49105962326000	-0.694638	12.680544	0.503953
1	33	Jogging	49106062271000	5.012288	11.264028	0.953424
2	33	Jogging	49106112167000	4.903325	10.882658	-0.081722
3	33	Jogging	49106222305000	-0.612916	18.496431	3.023717
4	33	Jogging	49106332290000	-1.184970	12.108489	7.205164

```
Layer (type)                 Output Shape              Param #
=================================================================
conv1d_9 (Conv1D)            (None, 100, 64)           1024
_________________________________________________________________
max_pooling1d_9 (MaxPooling1 (None, 50, 64)            0
_________________________________________________________________
dropout_9 (Dropout)          (None, 50, 64)            0
_________________________________________________________________
conv1d_10 (Conv1D)           (None, 50, 128)           41088
_________________________________________________________________
max_pooling1d_10 (MaxPooling (None, 25, 128)           0
_________________________________________________________________
dropout_10 (Dropout)         (None, 25, 128)           0
_________________________________________________________________
gru_5 (GRU)                  (None, 25, 50)            26850
_________________________________________________________________
flatten_5 (Flatten)          (None, 1250)              0
_________________________________________________________________
dense_8 (Dense)              (None, 6)                 7506
=================================================================
Total params: 76,468
```

Test accuracy: 0.985

Model	Accuracy
1) 1-layer LSTM	94.2
2) 2-layer LSTM	95.6
3) 1-layer Convnet	94.5
5) 2-layer Convnet	98.7
5) 2-layer Convnet + 1-layer LSTM	98.5

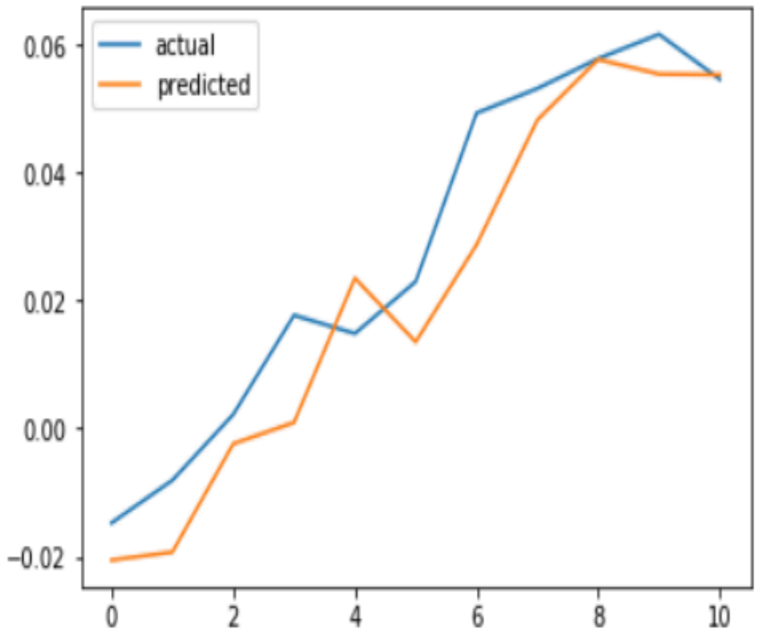


Experiment #2 - Sequence Prediction - Stock Prices

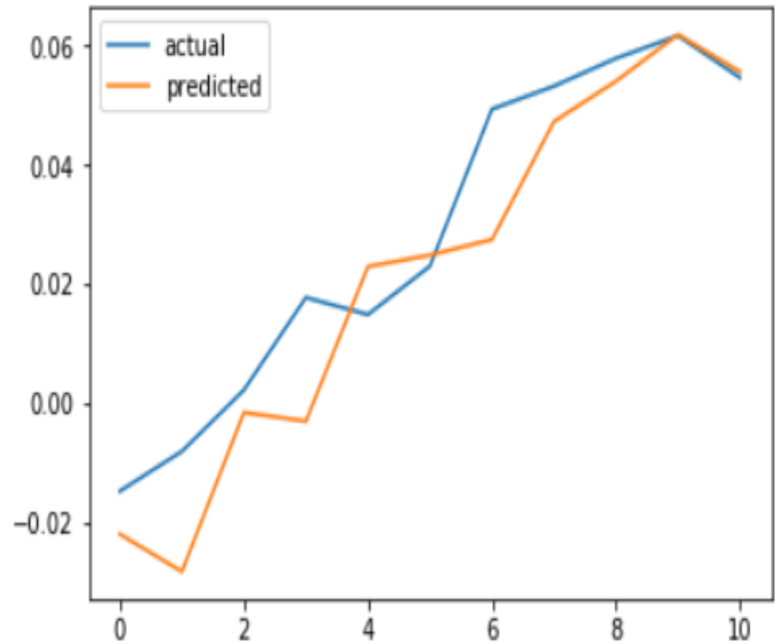
2-layer LSTM

+ 2 layer LSTM

Convnet



The Mean Absolute Error is: 0.00810866830379987



The Mean Absolute Error is: 0.008614427050262235

Experiment #3 - Multi-label classification

Predict stack overflow posting labels – 2 – layer biLSTM, embeddings, softmax

------------------- Train set quality: -------------------
processed 105778 tokens with 4489 phrases; found: 4532 phrases; correct: 4389.
precision: 96.84%; recall: 97.77%; F1: 97.31

------------------- Test set quality: -------------------
processed 13258 tokens with 604 phrases; found: 479 phrases; correct: 229
precision: 47.81%; recall: 37.91%; F1: 42.29

	title	tags
0	How to draw a stacked dotplot in R?	[r]
1	mysql select all records where a datetime fiel...	[php, mysql]
2	How to terminate windows phone 8.1 app	[c#]
3	get current time in a specific country via jquery	[javascript, jquery]
4	Configuring Tomcat to Use SSL	[java]
5	Awesome nested set plugin - how to add new chi...	[ruby-on-rails]
6	How to create map from JSON response in Ruby o...	[ruby, ruby-on-rails-3, json]
7	rspec test if method is called	[ruby]
8	SpringBoot Catalina LifeCycle Exception	[java, spring, spring-mvc]
9	How to import data from excel to mysql databas...	[php, codeigniter]

Takeaways

Futures

- Episodic temporal data
- Relational models
- Machine reading/summarization
- One-shot learning
- Online learning
- Model interpretability

Basics - Know your data

- LSTMs are sloooow
- Never start with the most complex model, establish a baseline with a basic model
- Deep learning works well when you have very large labeled datasets
- DL lacks interpretability

References

Adams Wei Yu, David Dohan, Minh-Thang Luong et al., QANET: COMBINING LOCAL CONVOLUTION WITH GLOBAL SELF-ATTENTION FOR READING COMPREHENSION
https://arxiv.org/pdf/1804.09541.pdf

Pranav Rajpurkar*, Awni Hannun*, Masoumeh Haghpanahi, Codie Bourn, and Andrew Ng, Cardiologist-Level Arrhythmia Detection With Convolutional Neural Networks
https://stanfordmlgroup.github.io/projects/ecg/

Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning
http://www.deeplearningbook.org/

Chris Olah, Understanding LSTM Networks
http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Chirs Olah, Conv Nets a Modular Perspective
http://colah.github.io/posts/2014-07-Conv-Nets-Modular/

Andrej karpathy, The Unreasonable Effectiveness of Recurrent Neural Network
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Andrej Karpatyh, Convolutional Neural Networks for Visual Recognition
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

François Chollet, Deep Learning with Python
https://www.manning.com/books/deep-learning-with-python

Aurélien Géron, Hands-On Machine Learning with Scikit-Learn and TensorFlow Concepts, Tools, and Techniques to Build Intelligent Systems
http://shop.oreilly.com/product/0636920052289.do