A (VERY) QUICK TOUR OF DEEP LEARNING FOR TIME-SERIES ANALYSIS

June 17^{th}, 2022

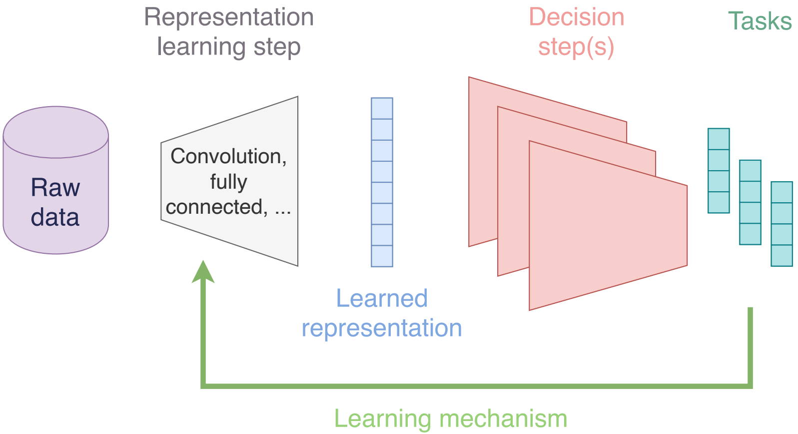
Vincent Guigue

Introduction

General idea of deep learning

Issues :

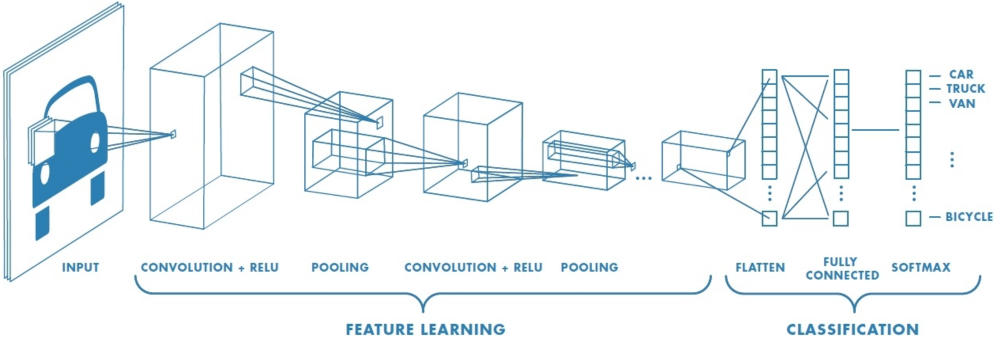
- Extracting relevant features
- ... by multi-task learning
- on multivariate time series



General idea of deep learning for time series

- Learn non linear combination of features (but no more than xgb...)
- Extract complex features = discriminant patterns ... In a noisy environment

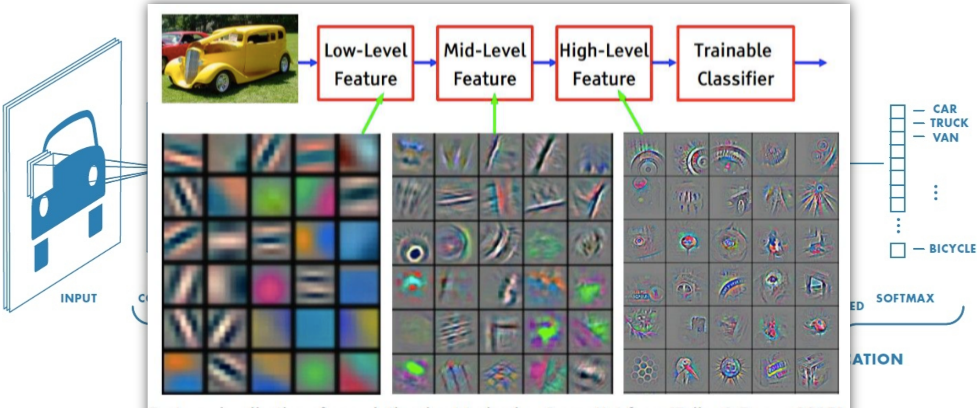
⇒ Once again, we have to think about our hypothesis



General idea of deep learning for time series

- Learn non linear combination of features (but no more than xgb...)
- Extract complex features = discriminant patterns ... In a noisy environment

⇒ Once again, we have to think about our hypothesis

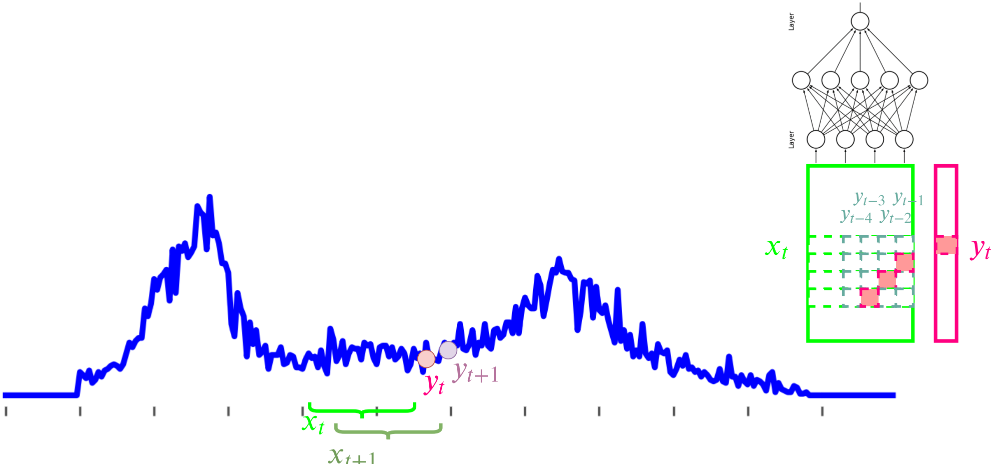


Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

From Multi-Layer Perceptron to Time Delay Neural Networks

Historical neural architectures

- TDNN : Time Delay Neural Networks



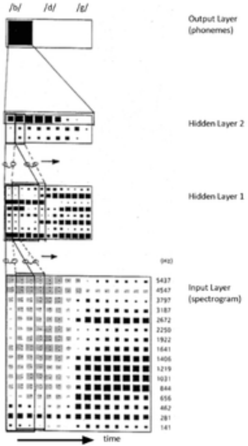
Applications

TDNN : Time Delay Neural Networks

- Originally : Multi Layer Perceptron on lag variables
- By extension : Any neural architecture on a temporal sliding window

Applications :

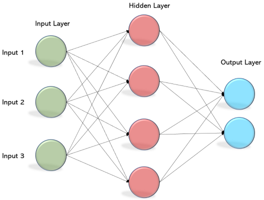
- Pattern classification :
  - Phoneme classification (speech recognition)
  - Handwriting recognition
- Signal processing
  - Echo and reverberation elimination



A. Waibel et al., IEEE Trans. ASSP, 1989
Phoneme Recognition Using Time-Delay Neural Networks

A non linear ML approach on rolling window

- Very close to XGBoost...
- But more subject to overfitting than ensembling approaches

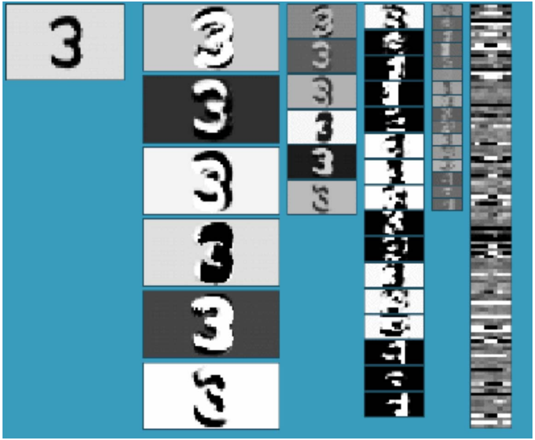


⇒ Compare them on classical ML procedure (while avoiding the pitfalls !)

Convolutional Neural Networks

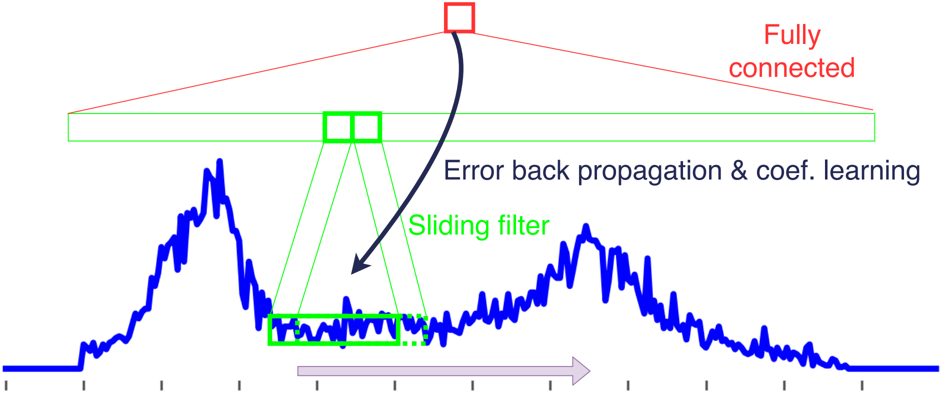
CNN History

One of the first breakthrough in machine learning : zip code recognition in 1989



- In 1989, 50 people in the world were able to set a CNN...
  the others were waiting for SVM !
- In 2010, 5000 people were able to set CNNs... That leads to AlexNet !

Definition of a Convolutional Neural Networks

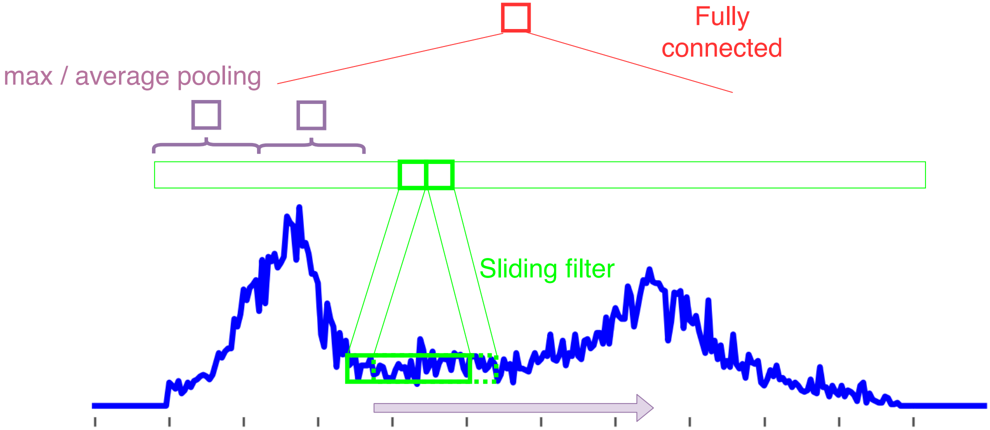


- Convolutional filter (few parameters)
- Signal representation + decision layer (more expensive)

$\Rightarrow$ Signal classification / pattern detection

Idea : learning to extract relevant features wrt a given task

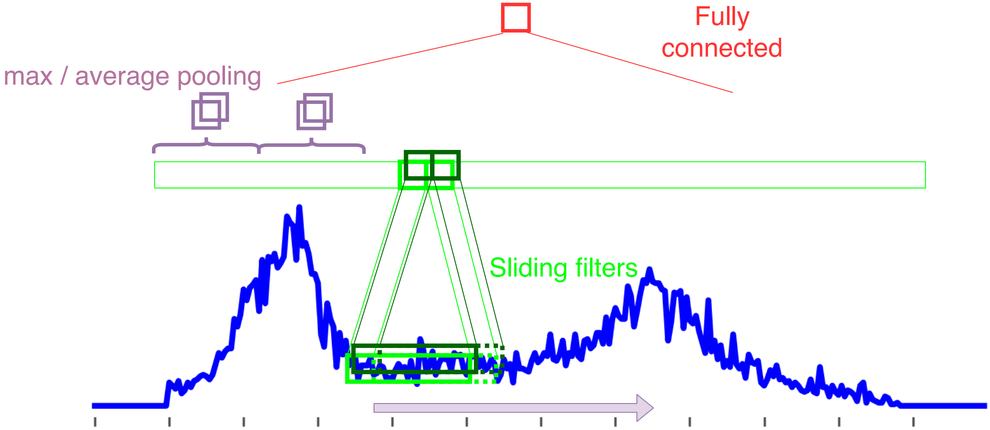
CNN, pooling & multiplication of the filters



Adding a pooling layer :

1. Reduce the cost of the fully connected layer
2. Add (slight but efficient) translation invariance

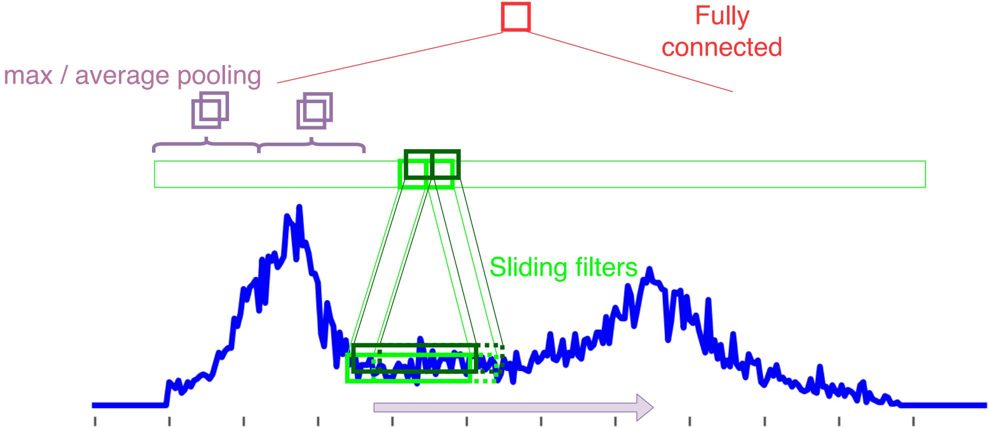
CNN, pooling & multiplication of the filters



Adding a pooling layer :

1. Reduce the cost of the fully connected layer
2. Add (slight but efficient) translation invariance

CNN, pooling & multiplication of the filters

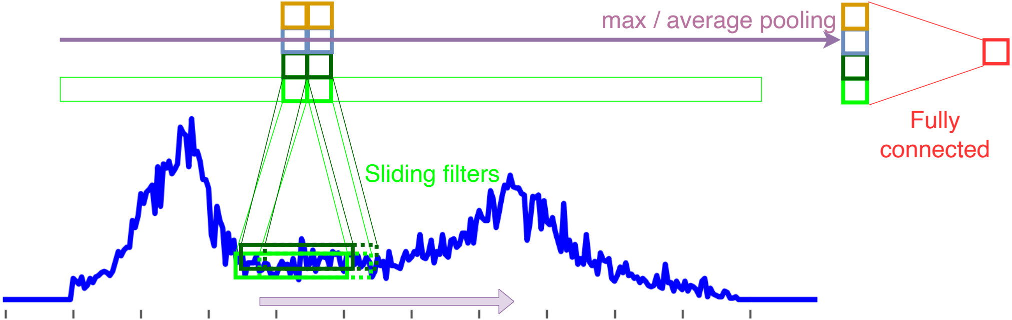


Even in 2009 = SVM golden age, CNN > SVM in handwriting reco.

⇒ Investigate variations of the signal in input

⇒ + Translation invariance

CNN & variable length time series

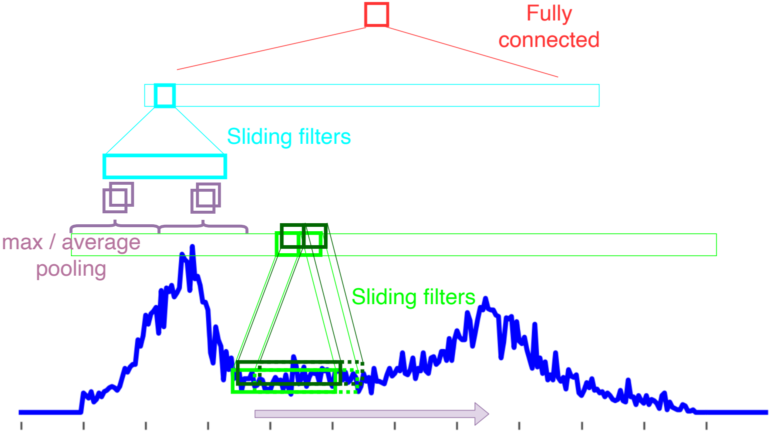


Enlarging pooling operation :

1. Reduce the cost of the fully connected layer
2. Each filter acts as a pattern detector
3. Fixed size signal representation
4. No more temporal descriptors in the representation

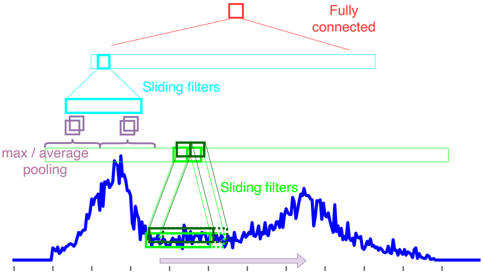
Multi-layer CNN

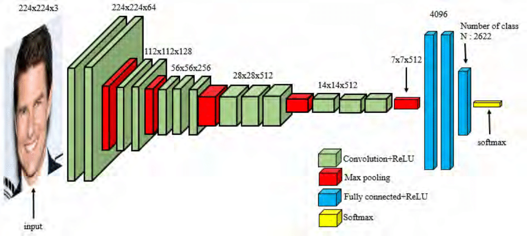
- State of the art in vision architecture
- An in depth explanation of classical internet illustrations

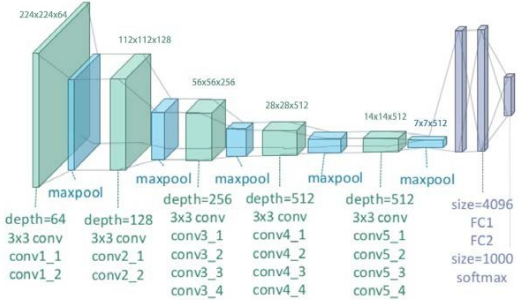


Multi-layer CNN

- State of the art in vision architecture
- An in depth explanation of classical internet illustrations



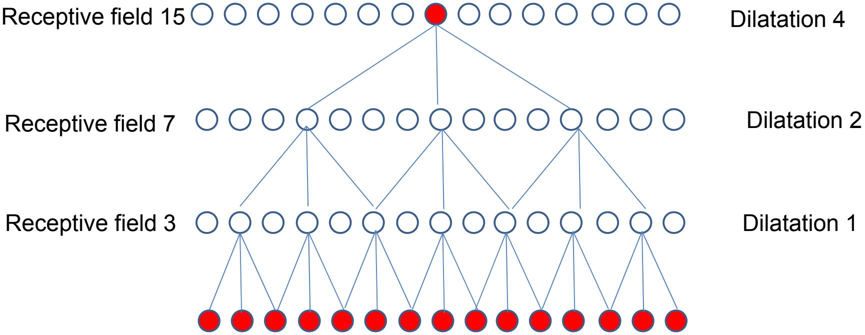




Dilated CNN

[Yu 2016]

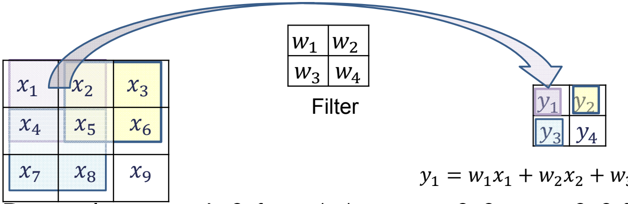
To analyse efficiently multi-scale aspects of a signal :



- Less computation
- More efficiency

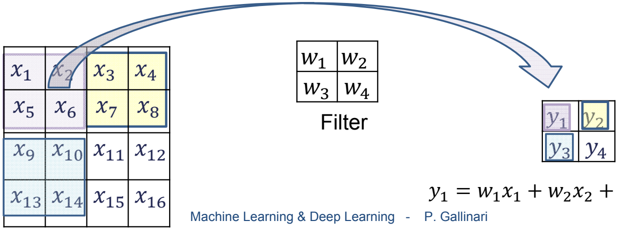
2D/3D CNN

- 2D convolution, stride 1, from 3x3 image to 2x2 image, 2x2 filter



$$y_1 = w_1 x_1 + w_2 x_2 + w_3 x_4 + w_4 x_5$$

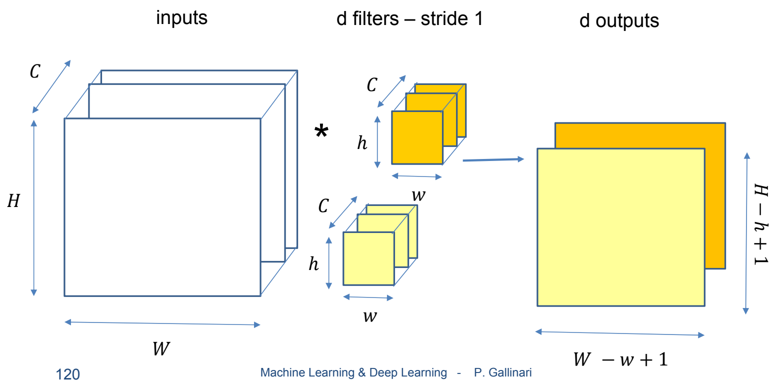
- 2 D convolution, stride 2, from 4x4 image to 2x2 image, 2x2 filter



$$y_1 = w_1 x_1 + w_2 x_2 + w_3 x_5 + w_4 x_6$$

+ pooling on spatial 2D windows

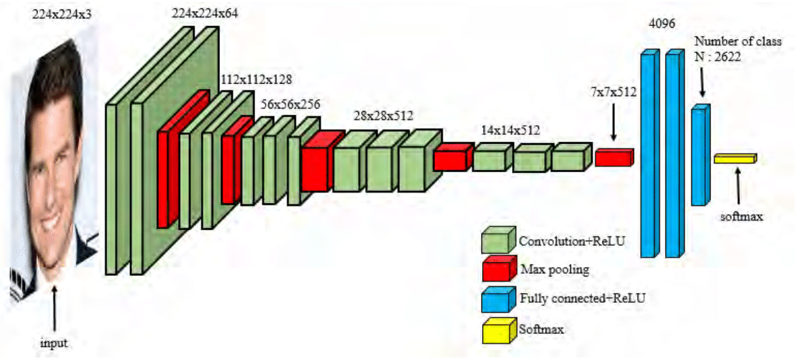
2D/3D CNN



Most of the time, we perform $N\times$ 2 dimensional convolutions instead of 3D conv.
$\Rightarrow$ It is linked to the nature of the data

Deep CNN

Image analysis / object recognition : AlexNet, VGG, ...



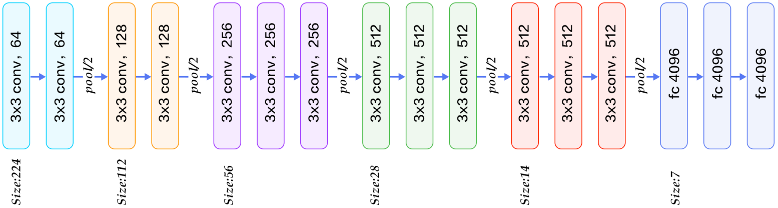


Image Reconstruction

Other use cases where image reconstruction is required :

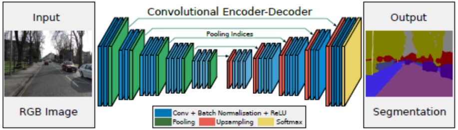
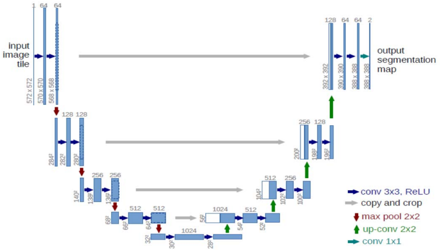


Fig. 2. An illustration of the SegNet architecture. There are no fully connected layers and hence it is only convolutional. A decoder upsamples its input using the transferred pool indices from its encoder to produce a sparse feature map(s). It then performs convolution with a trainable filter bank to densify the feature map. The final decoder output feature maps are fed to a soft-max classifier for pixel-wise classification.

SegNet – (Badrinarayanan 2017)

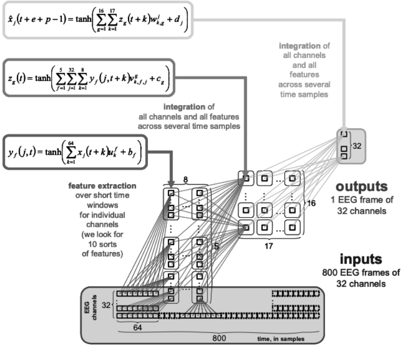


U-Net, (Ronneberger 2015)

⇒ We will come back to this point in the perspectives

Seizure detection

An application in signal classification : detecting seizure in EEG



CNN : a very elegant (& efficient) way to deal with multivariate time-series

P.W. Mirowski et al., IEEE, 2008
Comparing SVM and Convolutional Networks for Epileptic Seizure Prediction from Intracranial EEG

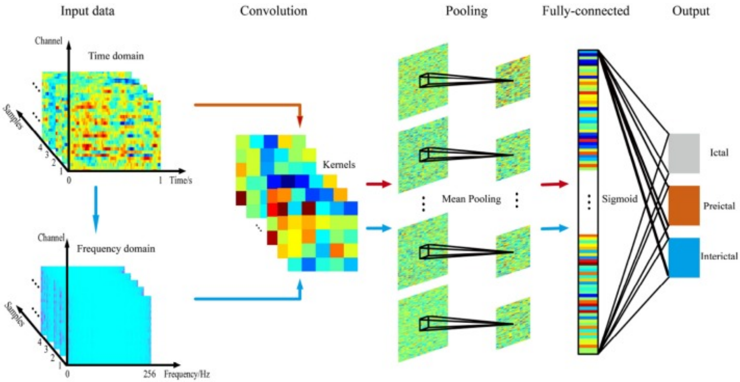
M Zhou et al., F. in Neuroinformatics, 2018
Epileptic Seizure Detection Based on EEG Signals and CNN

Seizure detection

An application in signal classification : detecting seizure in EEG

Recent architectures are mainly based on

- Time frequency decomposition
- Image analysis



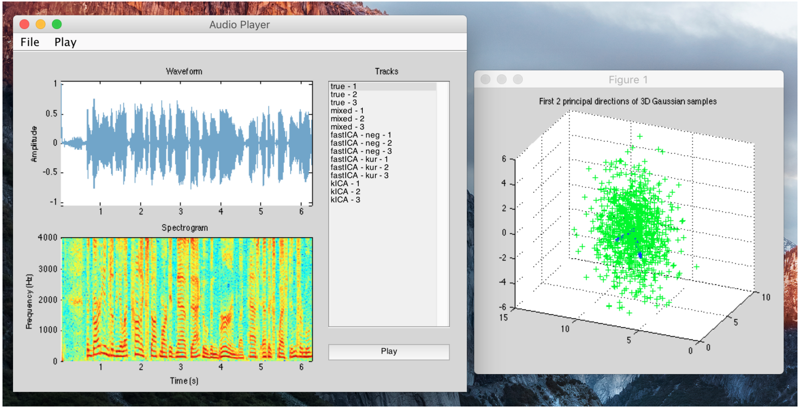
P.W. Mirowski et al., IEEE, 2008
Comparing SVM and Convolutional Networks for Epileptic Seizure Prediction from Intracranial EEG

M Zhou et al., F. in Neuroinformatics, 2018
Epileptic Seizure Detection Based on EEG Signals and CNN

CNN & Time-Frequency representation

The example of source separation (that makes great progress over the last 5 years)

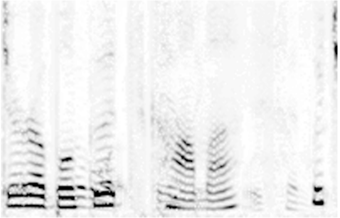
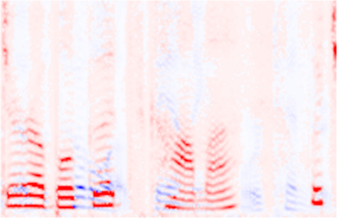
Original problem : ICA (independant componant analysis)
SVD algorithm (unsupervised) in time or time frequency domain :



CNN & Time-Frequency representation

The example of source separation (that makes great progress over the last 5 years)

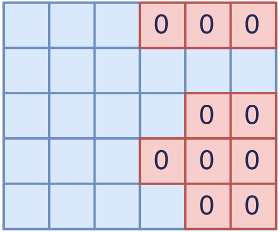
New Problem :
A supervised classification problem in the time frequency domain

Input/output sizes of CNN

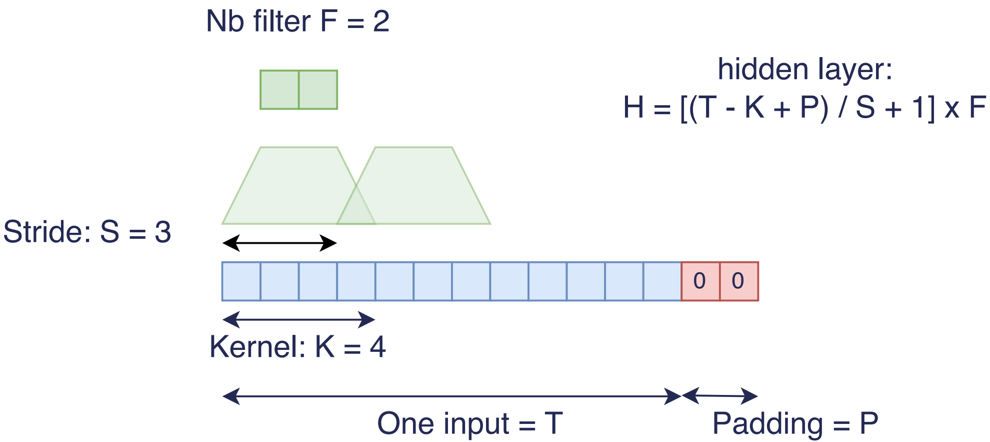
- Lightwise architecture
- Easy to catch hierachical dependancies
  ... And easy to set different kernel size (hour, day, week, ...)
- Made for fix sized entries...
  but padding may help
- CNN = often use for identification / pattern classification... But not only
- Features can be temporal (default) or not (pooling)

Padding :



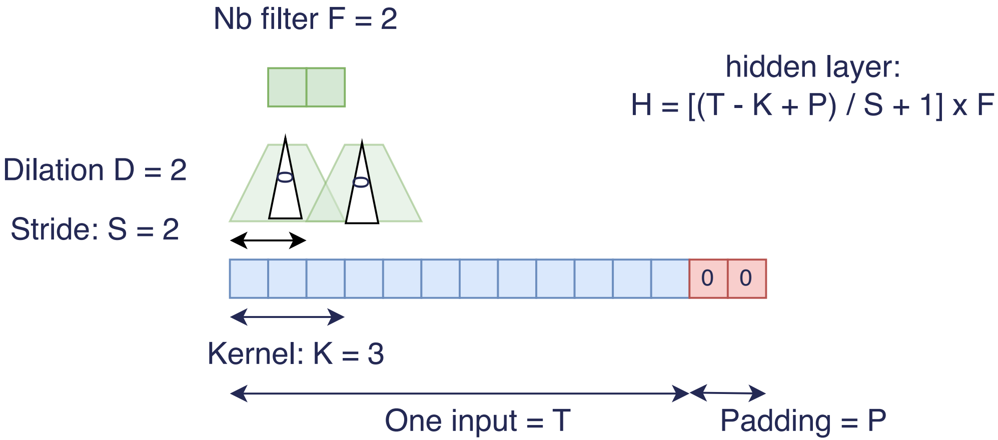
Signals

CNN Parameters



- Implemntation is easy...
- once you are able to compute properly the dimensions of all layers

CNN Parameters



- Implemntation is easy...
- once you are able to compute properly the dimensions of all layers

Learning Neural Network

- Gradient Backpropagation = gradient chain rule over the layer
- Gradient = easy to compute on the last layer...
- ... and gradient of the previous layers = easy to compute knowing the nex gradient
- Good news : nothing to do when using existing modules
  - Just encode the chain dependancy

```python
class EasyNet(nn.Module):
    def __init__(self, num_classes):
        super(EasyNet, self).__init__()
        self.conv1 = torch.nn.Conv1d(1,1,1) # => yields 24 values
        self.conv2 = torch.nn.Conv1d(1,1,2) # => yields 23 values
        self.conv4 = torch.nn.Conv1d(1,1,4) # => yields 21 values
        size_all_convs = 21+23+24 # To complete
        self.t1 = nn.Linear(size_all_convs, 24)
        self.t2 = nn.Linear(24, num_classes)

    def forward(self, x):
        all_convs = torch.cat([self.conv1(x),self.conv2(x),self.conv4(x)],dim=-1)
        first_transform = torch.tanh(self.t1(all_convs))
        second_transform = self.t2(first_transform)
        output = second_transform
        return output
```

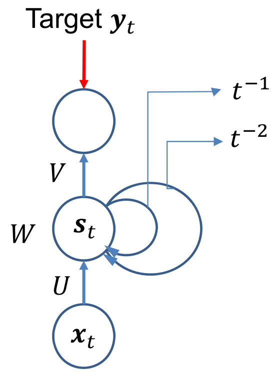
Recurrent Neural Networks

RNN History

- Appears in the 1990'
- Beautiful architecture... But hard to train

$\Rightarrow$ no real world application until 2006 (A. Graves)

- Today state of the art in :
  - Speech / handwriting transcription
  - Machine translation
  - NLP : language understanding / generation

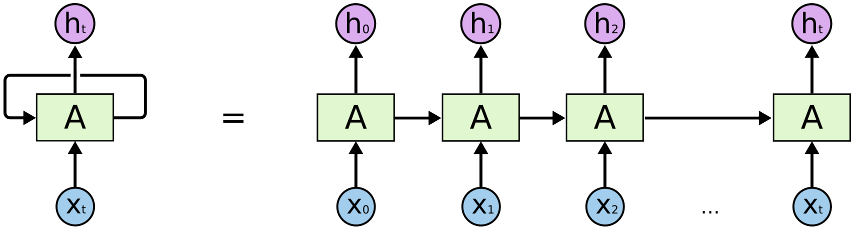


All weights learned

$$\boldsymbol{s}_{\mathrm{t}} = f(W\boldsymbol{s}_{t-1}) + U\boldsymbol{x}_t$$

Modern Recurrent Neural Network & LSTM : Rebirth of RNN

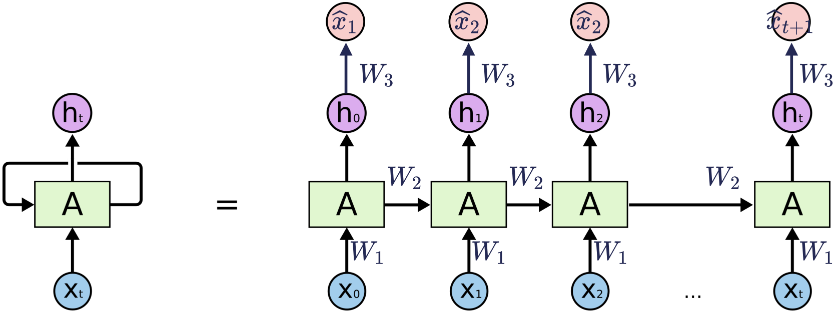
Unrolled a RNN for a better understanding :



- Lightwise (in theory)... $h_t = W_1 x_t + W_2 h_{t-1}$
- ... But impossible(/hard) to parallelise $\Leftrightarrow$ sequencial dependancies
- Quite costly in practice

Chris Olah's blog `http://colah.github.io/posts/2015-08-Understanding-LSTMs/`

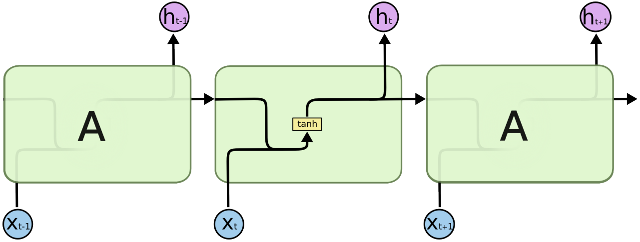
Modern Recurrent Neural Network & LSTM : Rebirth of RNN



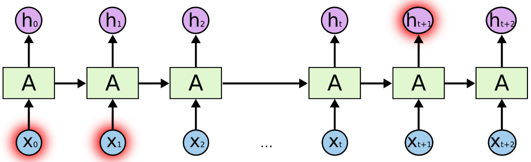
- $h_t = W_1 x_t + W_2 h_{t-1}$
- $\hat{x}_{t+1} = W_3 h_t$
- Play with W_1 : multivariate timeseries ; contexte modelling ; ...
- Play with W_3 : multiple outputs

Chris Olah's blog `http://colah.github.io/posts/2015-08-Understanding-LSTMs/`

Modern Recurrent Neural Network & LSTM : Rebirth of RNN



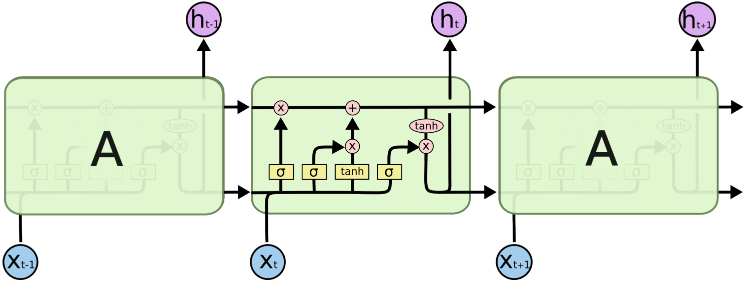
Gradient vanishes & long term dependancies are not modeled....



Chris Olah's blog http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Modern Recurrent Neural Network & LSTM : Rebirth of RNN

The phenomenon has been understood & (partially) overcome :
Neurons **learn** what should be **kept in memory** and what should be **forgotten**

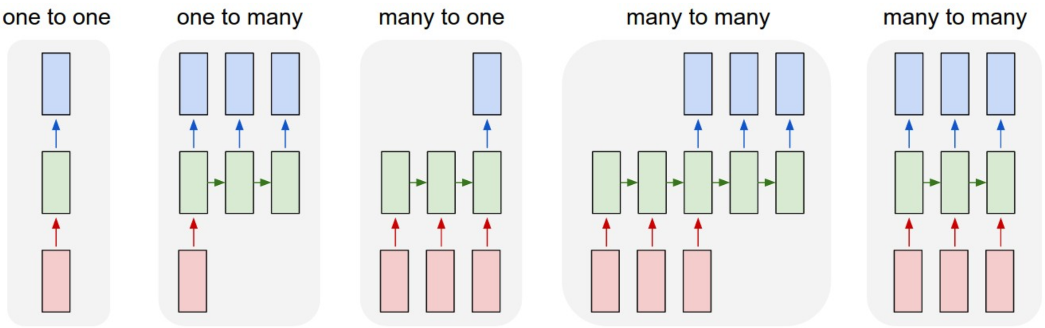


Gated architecture

S. Hochreiter, J. Schmidhuber, Neural computation 1997
Long short-term memory

Chris Olah's blog `http://colah.github.io/posts/2015-08-Understanding-LSTMs/`

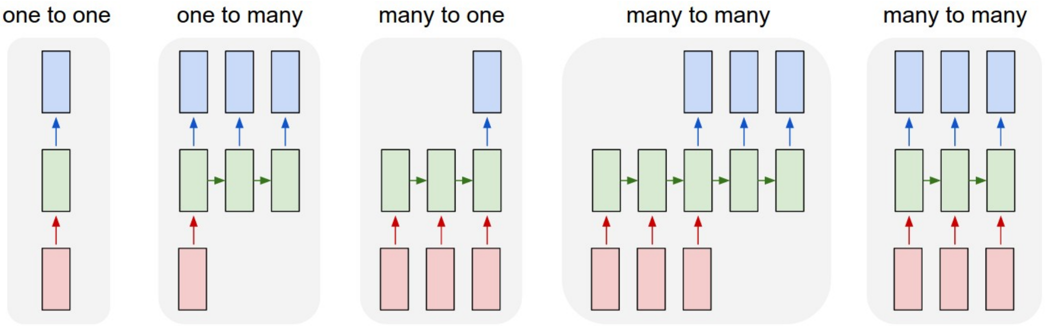
RNN architecture : different settings



- One to many : image annotation
- many to one : signal classification
- many to many : POS/NER tagging, sequence annotation
- seq to seq : machine translation

Karpathy's blog `http://karpathy.github.io/2015/05/21/rnn-effectiveness/`

RNN architecture : different settings

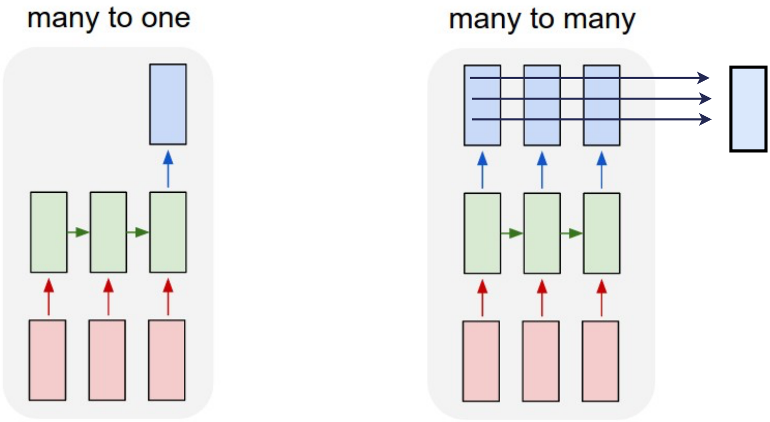


Seq-2-seq architecture are also known as encoder-decoder architecture :
red & blue part can be split into 2 distinct models

Karpathy's blog `http://karpathy.github.io/2015/05/21/rnn-effectiveness/`

Signal classification / many to one architecture

Architecture variation

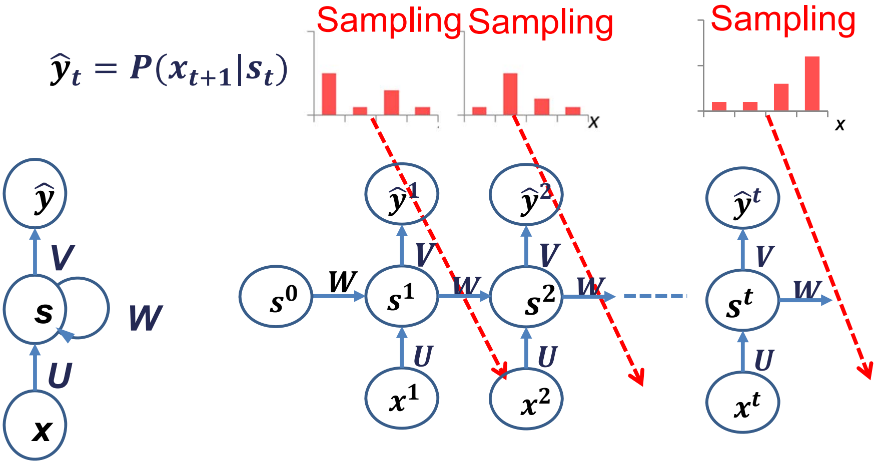


Learning a sequence model & generate new data :

Training a model to predict the next character given a sequence :

$\Rightarrow$ Sampling & beam search

$$\hat{y}_t = P(x_{t+1}|s_t)$$



Learning a sequence model & generate new data :

Training a model to predict the next character given a sequence :

$\Rightarrow$ Sampling & beam search

```
tyntd-iafhatawiaoihrdemot  lytdws  e ,tfti, astai f ogoh eoase rrranbyne 'nhthnee e
plia tklrgd t o idoe ns,smtt   h ne etie h,hregtrs nigtike,aoaenns lng
```

train more

```
"Tmont thithey" fomesscerliund
Keushey. Thom here
sheulke, anmerenith ol sivh I lalterthend Bleipile shuwy fil on aseterlome
coaniogennc Phe lism thond hon at. MeiDimorotion in ther thize."
```

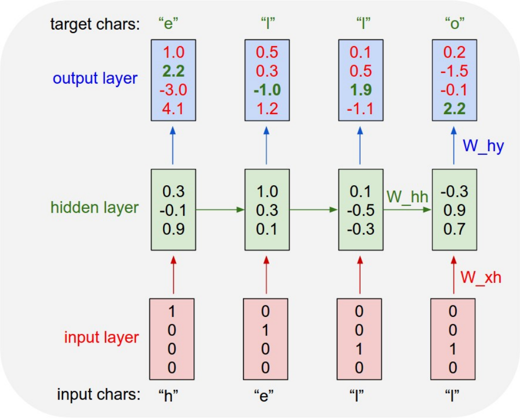
train more

```
Aftair fall unsuch that the hall for Prince Velzonski's that me of
her hearly, and behs to so arwage fiving were to it beloge, pavu say falling misfort
how, and Gogition is so overelical and ofter.
```

train more

```
"Why do what that day," replied Natasha, and wishing to himself the fact the
princess, Princess Mary was easier, fed in had oftened him.
Pierre aking his soul came to the packs and drove up his father-in-law women.
```

Karpathy's demonstration on char2char




```python
class RNN:
  # ...
  def step(self, x):
    # update the hidden state
    self.h = np.tanh(np.dot(self.W_hh, se
    # compute the output vector
    y = np.dot(self.W_hy, self.h)
    return y
```

+ multilayer architecture :

```python
y1 = rnn1.step(x)
y = rnn2.step(y1)
```

A. Karpathy's blog : The Unreasonable Effectiveness of Recurrent Neural Networks
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Karpathy's demonstration on char2char

Sample of *Shakespeare* generation

```
PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.
```

A. Karpathy's blog : The Unreasonable Effectiveness of Recurrent Neural Networks
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Karpathy's demonstration on char2char

Wikipedia sample

```
Naturalism and decision for the majority of Arab countries' capitalide was
grounded by the Irish language by [[John Clair]], [[An Imperial Japanese
Revolt]], associated with Guangzham's sovereignty. His generals were the
powerful ruler of the Portugal in the [[Protestant Immineners]], which
could be said to be directly in Cantonese Communication, which followed a
ceremony and set inspired prison, training. The emperor travelled back to
[[Antioch, Perth, October 25|21]] to note, the Kingdom of Costa Rica,
unsuccessful fashioned the [[Thrales]]
```

A. Karpathy's blog : The Unreasonable Effectiveness of Recurrent Neural Networks
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Karpathy's demonstration on char2char

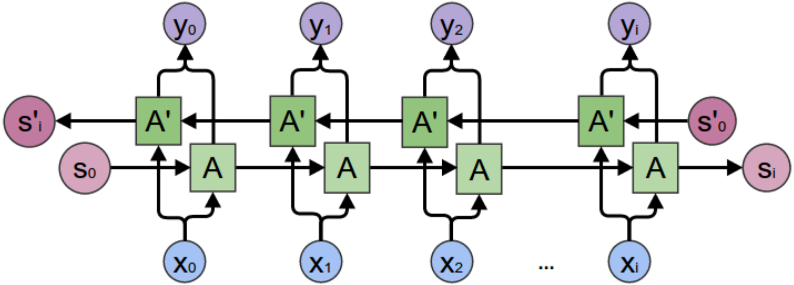
Linux code generation

```
 *
  * Increment the size file of the new incorrect UI_FILTER group information
  * of the size generatively.
  */
 static int indicate_policy(void)
 {
   int error;
   if (fd == MARN_EPT) {
     /*
      * The kernel blank will coeld it to userspace.
      */
     if (ss->segment < mem_total)
       unblock_graph_and_set_blocked();
     else
       ret = 1;
     goto bail;
   }
```

State Of The Art to represent a sequence : Bi-LSTM

LSTM

+ Sequential modeling
− Sequential dependencies ! = partial modeling



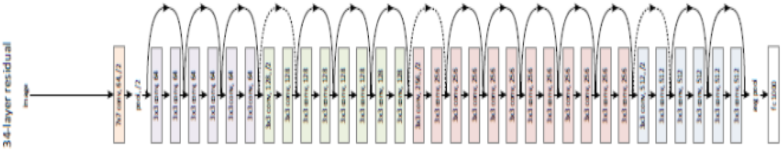
Bi-dimensional representation $[S_1, S'_1]$ is more powerful representation of the sentence S than each single representation.

Classical notation : $\mathbf{s} = [\overrightarrow{\mathbf{s}}, \overleftarrow{\mathbf{s}}]$

Recent proposals & trends

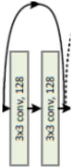
Introducing physical models in RNN

- Several NN use skip connections
  - e.g. ResNet



  - Resnet Module

x_t



$$x_{t+1} = x_t + f(x_t, \theta_t)$$

  - Changes the function composition perspective
    - Input x is progressively modified by a residual $f(x, \theta)$
    - x information is somewhat preserved in the forward propagation

Introducing physical models in RNN

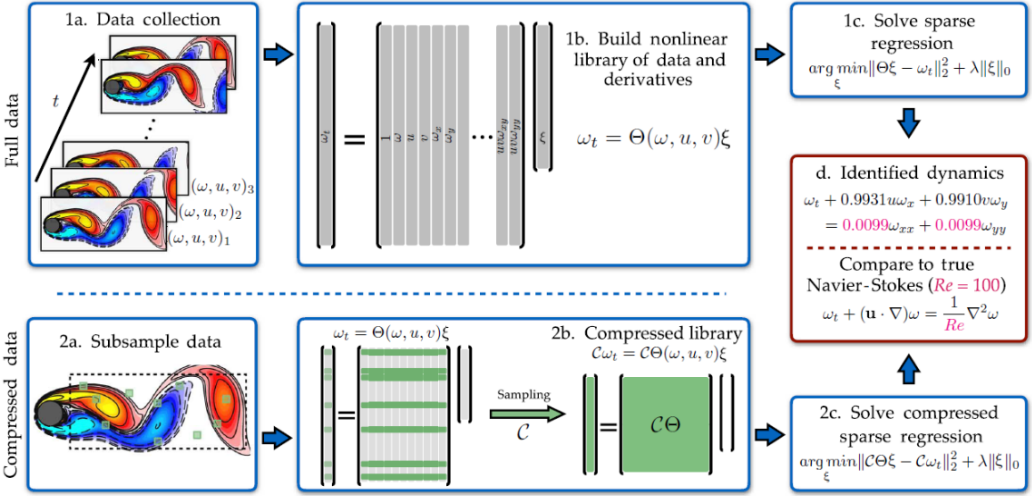


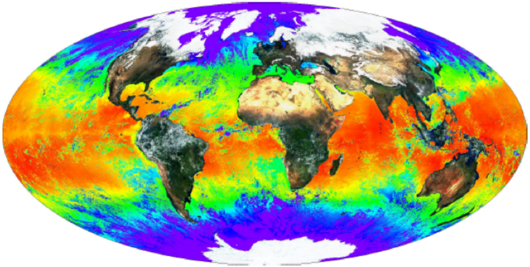
Fig. 1. Steps in the PDE functional identification of nonlinear dynamics (PDE-FIND) algorithm, applied to infer the Navier-Stokes equations from data. (1a) Data are collected as snapshots of a solution to a PDE. (1b) Numerical derivatives are taken, and data are compiled into a large matrix $\mathbf{\Theta}$, incorporating candidate terms for the PDE. (1c) Sparse regressions are used to identify active terms in the PDE. (2a) For large data sets, sparse sampling may be used to reduce the size of the problem. (2b) Subsampling the data set is equivalent to taking a subset of rows from the linear system in Eq. 2. (2c) An identical sparse regression problem is formed but with fewer rows. (d) Active terms in ξ are synthesized into a PDE.

Introducing physical models in RNN

- Describes transport of I through **advection** and **diffusion**

$$\frac{\partial I}{\partial t} + (w.\nabla)I = D\nabla^2 I$$

- I: quantity of interest (Temperature Image)
- $w = \frac{\Delta x}{\Delta t}$ motion vector, D diffusion coefficient



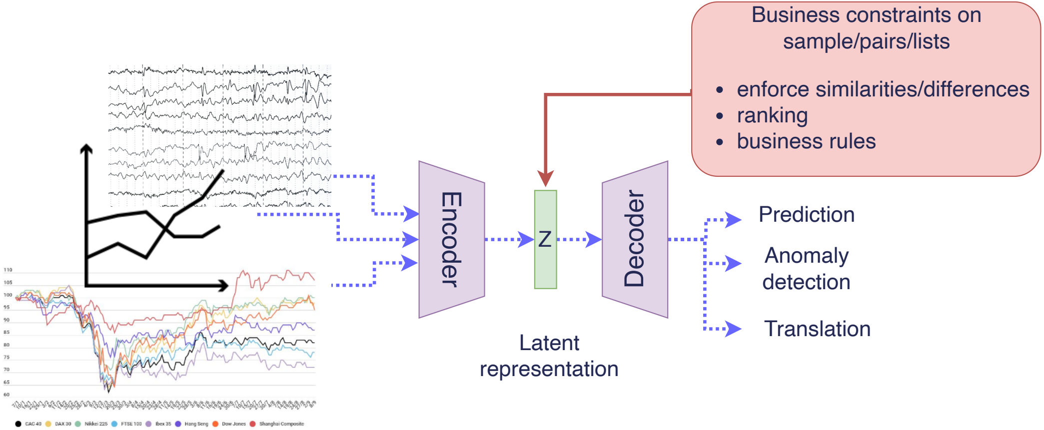
Great perspective :

- to combine simulation & data analysis
- to introduce diffusion process into ML model
  - ... And to enforce consistent behaviour of ML model !

End to end architectures

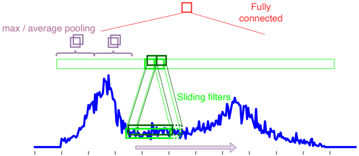
- From next value prediction to sequence encoding
  - = translation : from word for word translation to sentence encoding
- Multi-task : enhancing feature extraction ?
- Archi. plasticity + non convex formulation

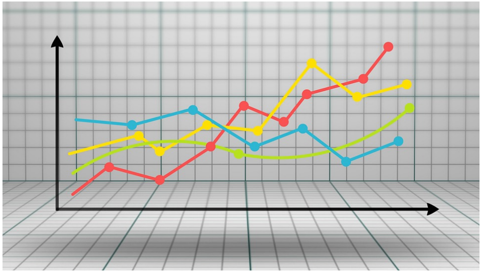
$\Rightarrow$ opportunities for business constraint encoding



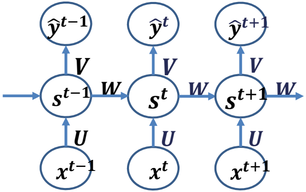
Various neural architectures to deal with multivariate time series

- CNN
  - Different filters for each channel / same filters





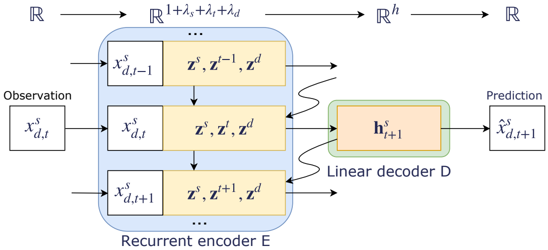
- RNN :
  - no problem to give a vector as input at each time step

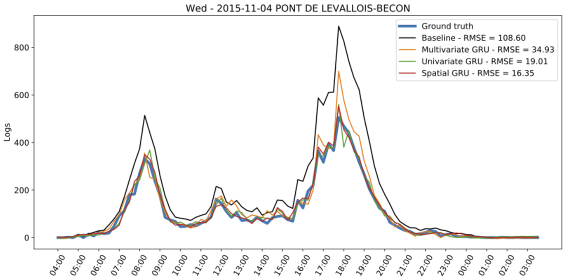


Modeling contextual information with RNN

Modeling a public transportation system :

- Station
- Day of the week
- Hour



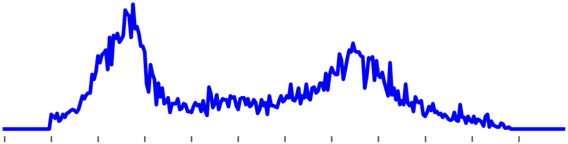


V. Guiguet et al., GRETSI 2019
Context aware forecasting for multivariate time series

RNN & latent factor disentanglement

Enforcing disentanglement :

Station 12
Wednesday



Cribier-Delande, ESANN, 2020
Time Series Prediction from Multiple Factors

Target :
encoding independtly
the station and the day

RNN & latent factor disentanglement

Proposed architecture :

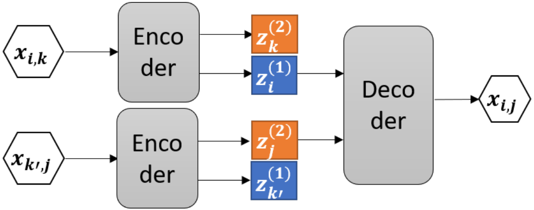
Encoder :

- 2 independent RNN
- or 2 independent CNN / MLP ...

Decoder :

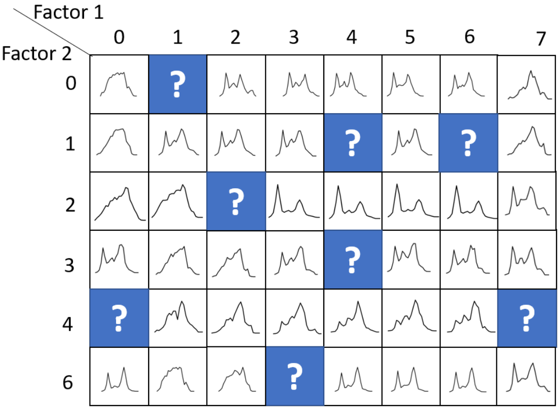
- Contextual CNN / RNN / MLP

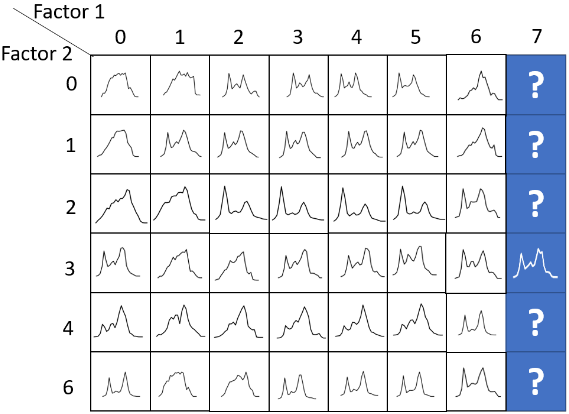
Cribier-Delande, ESANN, 2020
Time Series Prediction from Multiple Factors



RNN & latent factor disentanglement

Results :

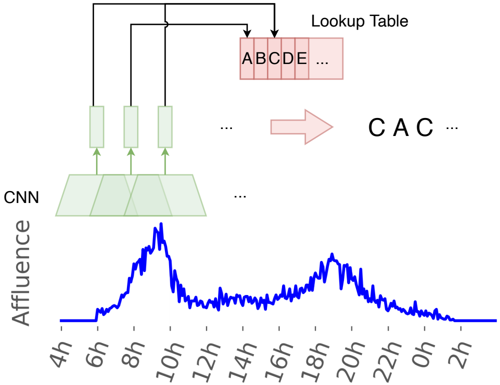




Cribier-Delande, ESANN, 2020
Time Series Prediction from Multiple Factors

Toward transfer & explainations

Idea : discretizing pieces of signals



- Pattern discretization = noise reduction
  as in matrix factorization / source decomposition
- Discrete sequence interpretation
- (re)Discovering Markov Models !

The Word2Vec paradigm (in NLP)

The distributional hypothesis [Harris et al. 1954]

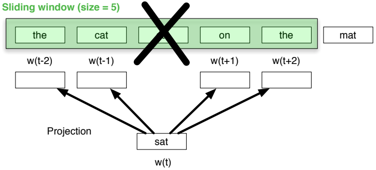
Word that appear in similar contexts in text tend to have similar meanings.

```
he curtains open and the moon shining in on the barely
ars and the cold , close moon " . And neither of the w
rough the night with the moon shining so brightly , it
made in the light of the moon . It all boils down , wr
 surely under a crescent moon , thrilled by ice-white
sun , the seasons of the moon ? Home , alone , Jay pla
m is dazzling snow , the moon has risen full and cold
un and the temple of the moon , driving out of the hug
 in the dark and now the moon rises , full and amber a
bird on the shape of the moon over the trees in front
 But I could n't see the moon or the stars , only the
rning , with a sliver of moon hanging among the stars
 they love the sun , the moon and the stars . None of
the light of an enormous moon . The plash of flowing w
man 's first step on the moon ; various exhibits , aer
 the inevitable piece of moon rock . Housing The Airsh
oud obscured part of the moon . The Allied guns behind
```

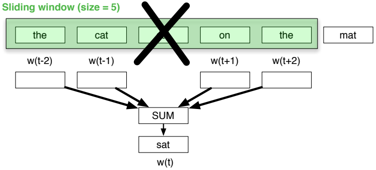
The Word2Vec paradigm (in NLP)

The distributional hypothesis [Harris et al. 1954]

Word that appear in similar contexts in text tend to have similar meanings.



Skip-gram



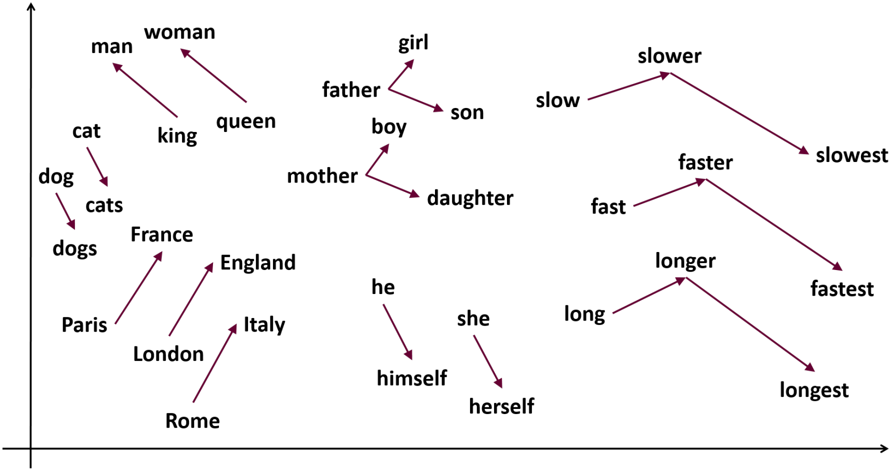
CBOW

```
he curtains open and the moon shining in on the barely
ars and the cold , close moon " . And neither of the w
rough the night with the moon shining so brightly , it
made in the light of the moon . It all boils down , wr
 surely under a crescent moon , thrilled by ice-white
sun , the seasons of the moon ? Home , alone , Jay pla
m is dazzling snow , the moon has risen full and cold
un and the temple of the moon , driving out of the hug
 in the dark and now the moon rises , full and amber a
bird on the shape of the moon over the trees in front
 But I could n't see the moon or the stars , only the
rning , with a sliver of moon hanging among the stars
 they love the sun , the moon and the stars . None of
the light of an enormous moon . The plash of flowing w
man 's first step on the moon ; various exhibits , aer
 the inevitable piece of moon rock . Housing The Airsh
oud obscured part of the moon . The Allied guns behind
```

$p(D=1|w_i, w_j; \theta) \Rightarrow$ proba. that w_i and w_j occur in the same context

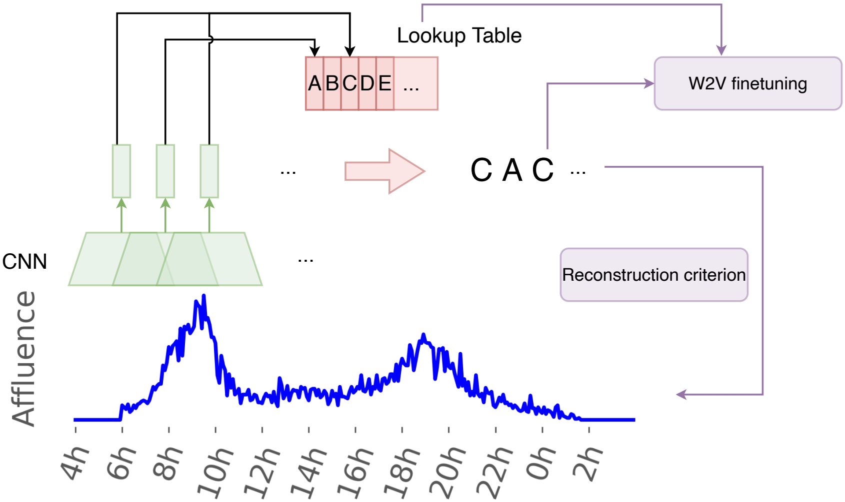
Modeling with a logistic function ; optimizing with Negative Sampling

The Word2Vec paradigm (in NLP)



- Synonyms are close...
- Semantic & grammatical geometric regularities arise
- (One of the) first really transferable semantic basis

W2V... On Signals



The opening of a new era in signal processing (as in NLP & vision earlier) ?

Franceschi et al., NeurIPS, 2019
Unsupervised scalable representation learning for multivariate time series

Conclusion

Benefits of deep learning architecture for time series modeling

- Efficient against noise
- Extract very relevant features
  - & relevant pattern with translation invariance
- Great software framework with GPU abilities
- Plasticity of the architectures
  - Naturally adapted to complex inputs
    - Variable length signals
    - Multivariate signals
  - New opportunities in signal generation / classification / understanding

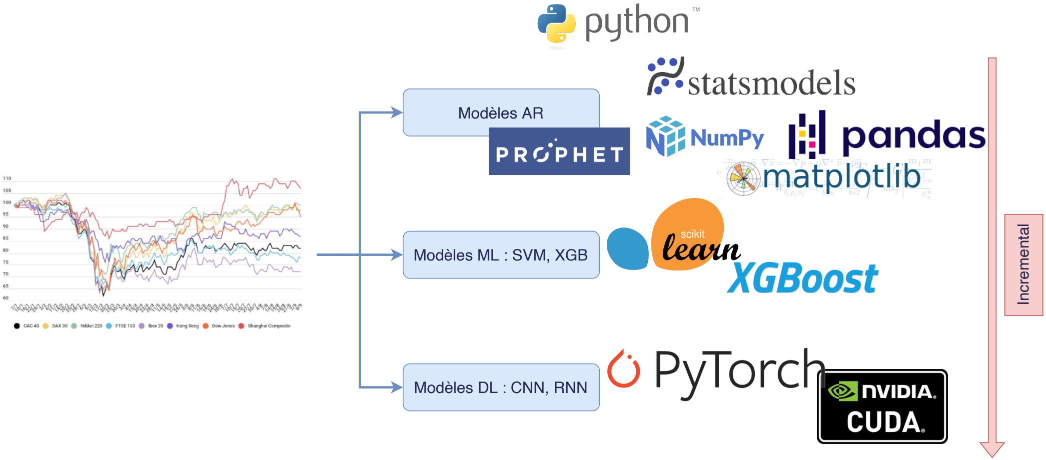
The question of pre-training

- Pre-training language model is a great advance for many application
  - Application with small corpus
  - Fine tuning
- Pre-training vision model is a great advance for many application
  - Recognizing cats on images improve the performance in detecting default on breaking pads...

⇒ It gives us a common knowledge of the world.

⇒ **Is is possible to learn a language model for signals ?**

Global picture



- Different approaches, different paradigms/syntaxes
- Different costs, different expectations
- Different hardware supports