Deep Learning for Climate

Nicolas Thome - Cnam Paris - CEDRIC / MSDMA
Joint work with V. Le Guen - EDF R&D
January 11, 2019

Climate and AI Workshop - Sorbonne Université (SU)
Outline

1. Context
2. Neural Networks Models & Architectures
3. Deep Learning for Solar Irradiance Estimation
4. Perspectives
Big data in Climate and Beyond

- **Superabundance of data**: times series (sensor measurements), images (fisheye, satellite), spatio-temporal data (weather forecasts), videos, text, etc.

- Obvious need for **Artificial Intelligence** with these data
  ⇒ Recognition, Decision Making
Decision Making in Climate

- **Huge number of applications**: classification, e.g. RADAR images, segmentation, e.g. eddies, forecasting, e.g. extreme weather event

[Chen et al., 2018b]

- Anticyclonic
- Cyclonic
- Non eddy

[Lguensat et al., 2018]  [Racah et al., 2017]
Decision Making in Climate

- Using Multiple inputs (wind, height, meta-data): hurricane track forecast

- Exploiting external knowledge: sea surface temperature prediction

[de Bezenac et al., 2018]

[Giffard-Roisin et al., 2018]

nicolas.thome@cnam.fr - Deep Learning and AI for Climate
Recognition of low-level signals: filling the semantic gap

What we perceive vs What a computer sees

- Illumination variations
- View-point variations
- Deformable objects
- intra-class variance
- etc

⇒ Need for "good" Intermediate Representations
Deep Learning (DL) & Recognition of low-level signals

- **DL**: learning intermediate representations
  - **Deep**: hierarchy, gradual learning
  - **Common** learning methodology, few expert knowledge

**Diagram:**
- Input: Image, Speech
- Process:
  1. $f(W^1x)$
  2. $h_1$ → $f(W^2h_1)$
  3. $h_2$ → $f(W^3h_2)$

- Output: cat, phonem

nicolas.thome@cnam.fr - Deep Learning and AI for Climate
Outline

1. Context

2. Neural Networks Models & Architectures

3. Deep Learning for Solar Irradiance Estimation

4. Perspectives
Neural Networks (NN)

- **The formal Neuron**

  \[ y = f(w^T x + b) \]

  - \( x_i \): inputs
  - \( w_i, b \): weights
  - \( f \): activation function
  - \( y \): output of the neuron

  \[ \hat{y}_1 = f(s_k) = \frac{e^{s_k}}{\sum_{k'=1}^{K} e^{s_{k'}}} \]

  \( \Rightarrow \) **Logistic Regression (LR) Model!**

- **Neural Networks**: Stacking several formal neurons \( \Rightarrow \) **Perceptron**

- **Soft-max Activation**: 

  \[ \hat{y}_k = f(s_k) = \frac{e^{s_k}}{\sum_{k'=1}^{K} e^{s_{k'}}} \]

  \( \Rightarrow \) **Logistic Regression (LR) Model!**
Deep Neural Networks (DNN)

- **Multi-Layer Perceptron (MLP):** Stacking layers of neural networks
  - More complex and rich functions / Logistic Regression (LR)
  - Neural network with one single hidden layer ⇒ universal approximator [Cybenko, 1989]

- Basis of the "deep learning" field
  - Hidden layers: intermediate representations from data
  - Can be learned with Backpropagation algorithm [Lecun, 1985, Rumelhart et al., 1986] (chain rule)
**Convolutional Neural Networks (ConvNets)**

- **ConvNets**: sparse connectivity + shared weights

- **Local feature extraction (≠ FCN)**
- Overcome parameter explosion for FCN on images
Convolutional Neural Networks (ConvNets)

- **Elementary block:** Convolution + Non linearity (e.g. ReLU) + pooling
  - **Stacking:** deep ConvNets [LeCun et al., 1989]

- **Parameters ↓, invariance ⇒ ↑ generalization & manifold disentangling!**

---

![Diagram of ConvNet](image-url)

---

2D t-SNE - NH=93.30833333333312

MLP_100_t-SNE - NH=95.86833333333321

CNN_100_t-SNE - NH=98.76833333333333

---

MLP_100 t-SNE fitting ellipses

CNN_100 t-SNE fitting ellipses
Recurrent Neural Networks (RNNs)

- **RNN Cell:** $h_t = \phi(x_t, h_{t-1}) = f(Ux_t + Wh_{t-1} + b_h)$ [Elman, 1990]
  - Loop, $h_t$ depends on current $x_t$ and previous state $h_{t-1}$
  - $h_t$: network memory up to time $t$ ⇒ Sequence processing

- Universal program [Siegelmann and Sontag, 1995] approximators

- Can be trained with Back-Propagation Through Time (BPTT)
Recurrent Neural Networks (RNNs)

- BUT Back-Propagation Through Time $\Rightarrow$ vanishing gradients
- Specific architectures:
  - LSTM [Hochreiter and Schmidhuber, 1997], GRU [Cho et al., 2014]
    - LSTM: Cell gate $\Rightarrow$ uninterrupted gradient flow

Uninterrupted Gradient flow

LSTM

GRU

RNN
Deep Learning Success since 2010

- 90’s / 2000’s: difficult to train large ConvNets / RNNs on big data

- Deep Learning renewal since 2010
  - 2011: Speech Recognition

<table>
<thead>
<tr>
<th>Acoustic model</th>
<th>Recog WER</th>
<th>RT03S FSH</th>
<th>Hub5 SWB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traditional features</td>
<td>1-pass -adapt</td>
<td>27.4</td>
<td>23.6</td>
</tr>
<tr>
<td>Deep Learning</td>
<td>1-pass -adapt</td>
<td><strong>18.5 (-33%)</strong></td>
<td><strong>16.1 (-32%)</strong></td>
</tr>
</tbody>
</table>
Deep Learning Success since 2010

- Deep Learning and ConvNet for Image Classification
  - ImageNet ILSVRC Challenge (Stanford):
    - 1,200,000 training images, 1,000 classes, mono-label
    - Based on WordNet hierarchy (ontology)
  - Up to 2012, leading approaches: handcrafted features + shallow ML (SVM)
  - ILSVRC’12: the deep revolution
    ⇒ outstanding success of ConvNets [Krizhevsky et al., 2012]

<table>
<thead>
<tr>
<th>Rank</th>
<th>Name</th>
<th>Error rate</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>U. Toronto</td>
<td>0.15315</td>
<td>Deep learning</td>
</tr>
<tr>
<td>2</td>
<td>U. Tokyo</td>
<td>0.26172</td>
<td>Hand-crafted features and learning models. Bottleneck.</td>
</tr>
<tr>
<td>3</td>
<td>U. Oxford</td>
<td>0.26979</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Xerox/INRIA</td>
<td>0.27058</td>
<td></td>
</tr>
</tbody>
</table>

- RNNs SOTA for many sequential decision making tasks: speech, translation, text/music generation, times series, etc
2012: the deep revolution

Deep ConvNet success at ILSVRC’12

Two main practical reasons:

1. Huge number of labeled images ($10^6$ images)
   - Possible to train very large models without over-fitting
   - Larger models enables to learn rich (semantic) features hierarchies

2. GPU implementation for training
   - Relatively cheap and fast GPU
   - Training time reduced to 1-2 weeks (up to 50x speed up)
Current Trends in Deep Learning

Feature design $\Rightarrow$ network architecture design

- **Improved training properties**, *e.g.* Res-Net or DenseNet $\sim$ LSTM

  - **ResNet** [He et al., 2016]
  - **DenseNet** [Huang et al., 2017]
  - **PolyNet** [Zhang et al., 2017]
  - **Xception** [Chollet, 2017]

- **Combining blocks for specific tasks**, *e.g.* detection or ConvLSTM

  - **RoI** [Dai et al., 2016]
  - **ConvLSTM** [Shi et al., 2015]
Current Trends in Deep Learning

Training Models

- **Combining DL & structured prediction**, e.g. Conditional Random Fields (CRF)
  - Speech recognition (RNN+CRF), Semantic segmentation (ConvNets+CRF/RNN)

- **Generative Adversarial Networks**: Game Theory (generator vs discriminator)
  - Adversarial cost used beyond generation for distribution matching
Outline

1. Context
2. Neural Networks Models & Architectures
3. Deep Learning for Solar Irradiance Estimation
4. Perspectives
Context: PhotoVoltaic (PV) energy forecasting

Different data sources for different horizons

- **Ground based images**: very short term spatial & temporal horizons (0-20 min)
- **Application**: dynamic control of a hybrid system with PV, storage, diesel,...
Data

Meteorological campaign EDF R&D

- EDF R&D experimental test site at La Reunion since 2012
- Many devices evaluated for solar resource assessment
Data
Meteorological campaign EDF R&D

- **Choice:** ground images + pyranometer
- **Goal:** Can we use low-cost cameras instead of pyranometers to estimate current and future solar irradiations?

⇒ More than 7 Millions images and corresponding irradiation measurements collected since 2010

nicolas.thome@cnam.fr - Deep Learning and AI for Climate
Data

Instrumentation

**Fisheye camera**: 180° hemispheric view of the sky, images every 10s

- **Pyranometer**: solar irradiance measurements every 10s
  - GHI: Global Horizontal Irradiance
  - DHI: Diffuse Horizontal Irradiance
  - DNI: Direct Normal Irradiance

- **Preprocessing**: irradiance values normalized by a clear sky model to remove seasonality ⇒ KGHI
Baseline
Irradiance estimation module

1. **Image segmentation**: with handcrafted thresholds on Luminance and R-B

   ![Image segmentation example]

   - Each image described with feature \( x_i \in \mathbb{R}^5 \) \( \leftrightarrow \) pixel class ratios
   - **Database**: \((x_i, y_i)_{i=1:N}\) with \(x_i\) images and \(y_i\) corresponding KGHI

2. **Estimation**: kernel regression (Nadaraya-Watson model [Nadaraya, 1964]) for an unknown image \( x_0 \):

   \[
   \hat{y}(x_0) = \frac{C}{N} \sum_{i=1}^{N} e^{-\frac{||x_0 - x_i||^2}{2h^2}} y_i
   \]
Baseline
Forecasting module

0. Camera calibration: \((x, y) \leftrightarrow\) spherical coordinates (azimuth, elevation)

1. Hemispherical image projection in a plane at a given altitude (where clouds have same direction)

2. Optical flow between two frames

3. Warp main wind direction and reproject

4. Estimate future image

Estimate +4min

True future +4min
Estimating solar irradiance with ConvNets

Proposed neural network models

- **Small Convnet**: 475,000 parameters

- **Densenet model**: Densenet conv layers + dense regression layers
  201 layers: 18 Millions parameters
Estimating solar irradiance with ConvNets

Experiments

- **Experimental setup:**
  - Training set: years 2012-2015 (4,190,064 images)
  - Test set: year 2016 (1,265,717 images)

- **Implementation:**
  - Python with Keras & Tensorflow backend
  - Adam optimizer

- **Training time:** - Nvidia Quadro P6000 (24 Go RAM)
  - 1 day for ConvNet, 6 days for DenseNet on a

- **Results for KGHI Estimation** (test set): large improvements of ConvNets wrt baseline

<table>
<thead>
<tr>
<th>Model</th>
<th>Baseline</th>
<th>ConvNet</th>
<th>DenseNet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Absolute Error (MAE)</td>
<td>0.1010</td>
<td>0.0448</td>
<td>0.0197</td>
</tr>
<tr>
<td>Normalized(^1) MAE</td>
<td>14.9 %</td>
<td>6.59 %</td>
<td>2.90 %</td>
</tr>
<tr>
<td>Root Mean Square Error (RMSE)</td>
<td>0.1467</td>
<td>0.06992</td>
<td>0.0328</td>
</tr>
<tr>
<td>Normalized RMSE</td>
<td>21.6 %</td>
<td>10.3 %</td>
<td>4.83 %</td>
</tr>
</tbody>
</table>

\(^1\)by the mean KGHI value over the training set
Estimating solar irradiance with ConvNets

Training Evolution

Learning from scratch possible (ImageNet pre-training not necessary)

200 époques
3 Millions images
Estimating solar irradiance with ConvNets

Results on a particular day
Estimating solar irradiance with ConvNets

t-SNE visualization

Figure: Clustering on Densenet features. Upper left: clear sky, upper right: cloudy, bottom: very cloudy, rainy
Ongoing work: irradiance forecasting

- **First proposed model:** light for computational purposes
- Input sequence: 10 grayscale 60x60 images every 30s
- Predict future image and irradiance at 5min
- Stacked ConvLSTM layers as spatiotemporal feature extractor
Ongoing work: irradiance forecasting

Preliminary results

RMSE BASELINE = 33.3%
RMSE CONVLSTM = 22.8%
Outline

1. Context
2. Neural Networks Models & Architectures
3. Deep Learning for Solar Irradiance Estimation
4. Perspectives
Conclusion & Perspectives

- Effective deep ConvNet solutions for irradiance predictions on static images
  - Favorable context: huge volume of annotated data

- Promising first results for future irradiance forecasting
  - Short-term: contribution of temporal information to irradiance & forecasts
  - Longer term: improve forecasting models and training methodologies
Deep learning for video prediction

- Direct RGB future image generation still challenging for large and complex natural images, predictions become blurry [Srivastava et al., 2015]
- To mitigate this: learn geometric transforms between images [Finn et al., 2016], use an adversarial loss instead of L2 [Mathieu et al., 2015]
Direct irradiance prediction with deep learning without future image prediction

- Predict latent features in encoder-decoder RNN architectures [Luo et al., 2017]

- Predict future image segmentations [Luc et al., 2018]
Forecasting future irradiances

Direct irradiance prediction

**Prediction with physical knowledge**

- Introduce a priori physical information (advection diffusion PDE) [de Bezenac et al., 2018]

\[
\frac{\partial I}{\partial t} + (w \cdot \nabla) I = D \nabla^2 I
\]

- Approximate differential equation solutions with neural nets [Long et al., 2017, Chen et al., 2018a]

Figure 5: Images of the true dynamics and the predicted dynamics. The first row shows the images of the true dynamics. The second row shows the images of the predicted dynamics using the PDE-Net
Choice of loss function
How to penalize delays on the predicted irradiance time series?

**Goal:** forecast irradiance ramps on time
Classical loss functions (MAE, RMSE) ill adapted to distinguish absolute value errors from temporal distortion errors. Possible solutions:
- signal gradient loss [Mathieu et al., 2015]
- loss based on Dynamic Time Warping [Cuturi and Blondel, 2017],...

**Figure:** 2 forecasts with similar RMSE but different anticipating skills. Forecast 2 model seems always late
Multi-modal fusion

Ground images, satellite images, numerical weather forecasts

- Improve forecasting temporal and spatial scales by fusing different data sources:
  - PV production measurements
  - ground based camera
  - satellite image
  - numerical weather forecasts

- Use a network of multiple cheap cameras on a territory to anticipate global phenomena
Thank you for your attention!

Questions?
Neural ordinary differential equations.
NIPS 2018.

Labeled sar imagery dataset of ten geophysical phenomena from sentinel-1 wave mode (tengeop-sarwv).
SEANOE.

Learning phrase representations using rnn encoder-decoder for statistical machine translation.

Xception: Deep learning with depthwise separable convolutions.


Approximation by superpositions of a sigmoidal function.


Deep learning for physical processes: Incorporating prior scientific knowledge.
ICLR.
Finding structure in time.
*COGNITIVE SCIENCE*, 14(2):179–211.

Unsupervised learning for physical interaction through video prediction.

Deep residual learning for image recognition.
In *CVPR*.

Long short-term memory.

Densely connected convolutional networks.
*CVPR 2017*.

Imagenet classification with deep convolutional neural networks.

*Une procedure d’apprentissage pour reseau a seuil asymmetrique (A learning scheme for asymmetric threshold networks)*, pages 599–604.

Backpropagation applied to handwritten zip code recognition.
IEEE.

Pde-net: Learning pdes from data.

Predicting future instance segmentations by forecasting convolutional features.

Unsupervised learning of long-term motion dynamics for videos.

Deep multi-scale video prediction beyond mean square error.

On estimating regression.

Extremeweather: A large-scale climate dataset for semi-supervised detection, localization, and understanding of extreme weather events.

Learning representations by back-propagating errors.

