Application of Deep Learning Techniques in Satellite Telemetry Analysis.

Greg Adamski, Member of Technical Staff
L3 Technologies
Telemetry and RF Products

Julian Spencer-Jones, Spacecraft Engineer
Telenor Satellite AS

Agenda

- Artificial Neural Networks (ANNs) and Deep Learning
- Classic machine learning vs. ANNs
- Selected Neural Network architectures and their applications
    - Fully Connected (Dense)
    - Convolutional Neural Networks (CNNs)
    - Recurrent Neural Networks (RNNS and LSTMs)
- Predicting thermal parameter values based on historical record using LSTMs network

Artificial Neural Networks and Deep Learning

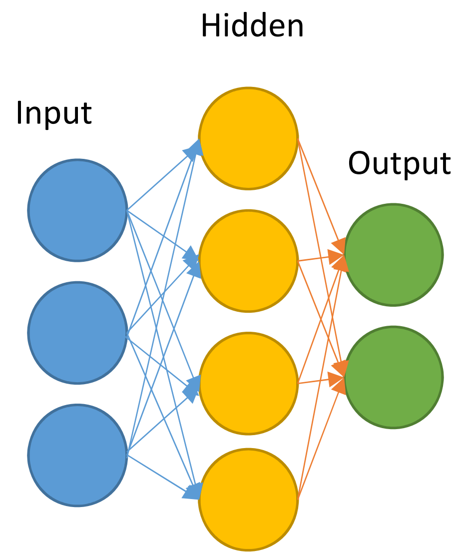
What they are not:

- Brand new toy for Data Scientists
  - Invented in 1940s, so it's been around for a while. Recent years have brought tremendous advancements in architectures, training approaches as well as computing power allowing to train large networks with a lot of data
- Magic bullet to solve all the world problems (yet)
  - Some areas are much better researched than others: computer vision, natural language processing, playing computer games (old Atari arcade, Chess, Go)
- A foolproof way for your boss to get you a new video card (well, maybe)
  - Training ANNs tends to be very computationally expensive. Using modern video cards can speed up the process by at least an order of magnitude (state of the art. CPU vs decent GPU)

Artificial Neural Networks and Deep Learning

What they are

- a network of connected functional units sometimes called neurons where outputs from one layer are fed into another layer
- A data modeling tool that can approximate any function [see https://en.wikipedia.org/wiki/Universal_approximation_theorem]
- A powerful machine learning tool
- A tool that automatically detects numerous input features without the need for complex feature engineering
    - Carefully engineered features as inputs can in some situations improve performance



Simple ANN diagram

Fully Connected Network

- Each neuron in layer n receives input from all neurons in layer n-1
- Typically the activation of the neuron is calculated by calculating a weighed sum of the inputs and applying a non-linearity to the result
- Very useful for classification and regression problems
- Does not deal well with context data:
  - E.g. for face recognition, looks at individual pixels, but has no ability to look at a broader context (like body parts etc.)
- For best performance, the inputs should be well engineered features
- Frequently used in tandem with CNNs and RNNs for final classification/regression

Convolutional Neural Networks (CNNs)

Each convolution layer applies a convolution filter to the input vector:

6	3	1	4
4	2	4	2
1	2	3	5
4	1	2	9

1	1
0	0

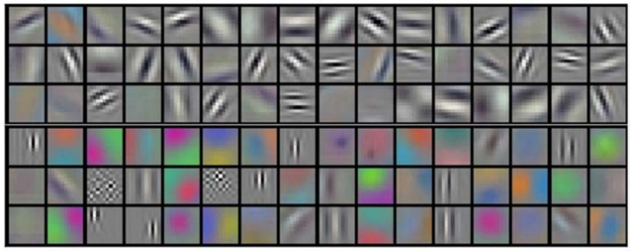
9	4	5
6	6	6
3	5	8

Convolution filters are learned during the training process

This allows the network to learn to recognize features within the input without explicitly being told to.

Convolutional Neural Networks (CNNs)

- For example for image data it can detect edges, gradients (early layers) or complex elements such as noses or books in deeper layers (images) [1],[2]
- For telemetry data, 1d convolutions may return 1d time series features relevant to the problem domain [3]
- These features can then be used to train a fully connected network to perform the task



CNNs – applications

- Image recognition

  Given a list of labeled images, define which class the image belongs to. For example for satellite imagery data the network can say if the picture contains trees, crops, cars, buildings etc.

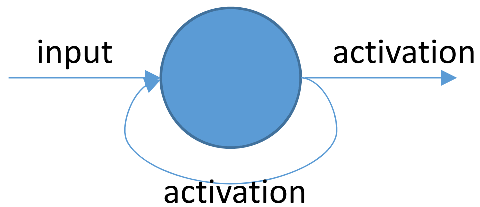
- Object detection

  Parse a satellite image to mark where each car, building or ship are (both labels and bounding boxes)

- Image processing
  - Advanced scaling image augmentation and processing techniques (super resolution, style transfer) which could enhance the quality of satellite imagery

Recurrent Neural Netwoks (RNNs)

- Output of a layer is re-fed back as input into that layer
- Neuron retains some state between executions
- Allows the neural network to learn the context
- Popular in natural language processing
- Gaining popularity in time series data analysis
- LSTM – Long Short Term Memory – a variant of RNN



Problem Statement

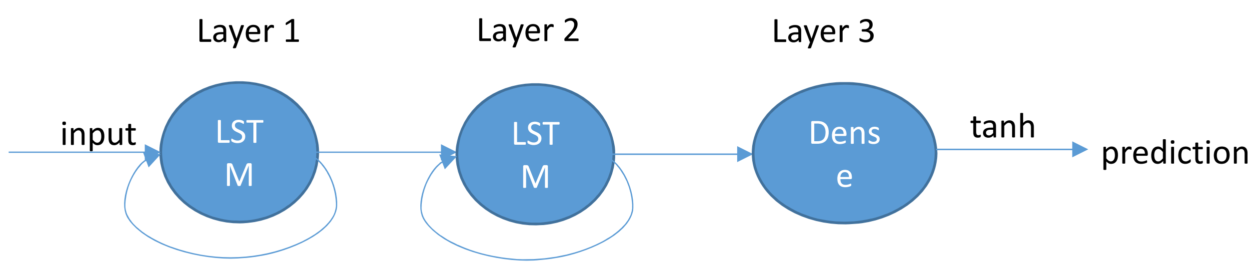
- Given a historical record of thermal performance on a Telenor satellite, predict the future behavior
- Allow the algorithm to discover data trends on its own without manual feature engineering
  - Allows the algorithm to be re-used on various measurands
- More specifically:

Based on a number of <n> length sequences, build a model that predicts sample n+1

- Important: values are aggregated over 20 minute periods to remove noisce and scaled to zero mean, unit variance

Architecture

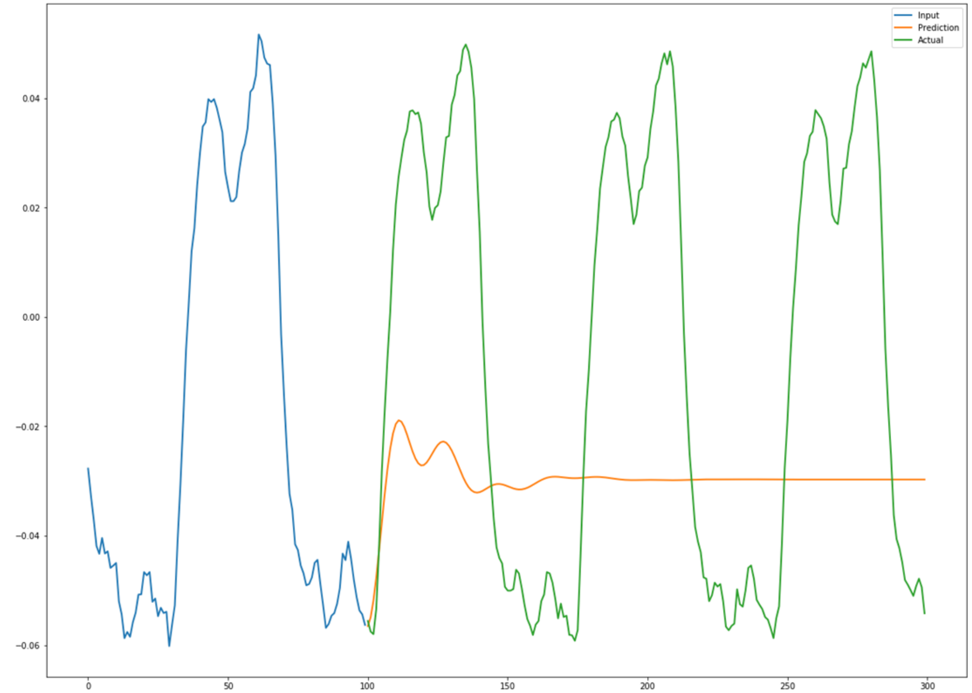
- 3 layer network with LSTM layers responsible for keeping the context and single “normal” neuron providing value calculation
- 100 samples of historical values used for input
- 101^{st} sample used for output
- Mean square error loss function



Training process and results

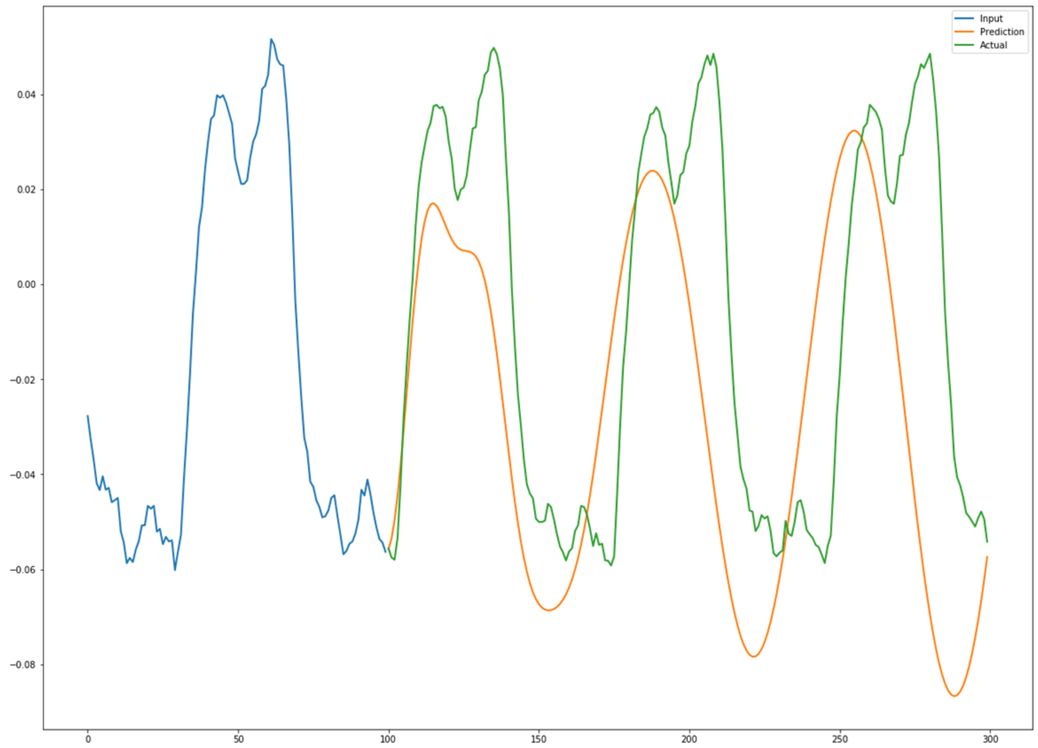
- After 4 training epochs some patterns emerge
- Blue – context data
- Green – actual values
- Orange – predicted values

Note: predictions are performed one sample at a time with a fixed window. With each step, oldest sample is purged while the newest prediction is used as “ground truth”



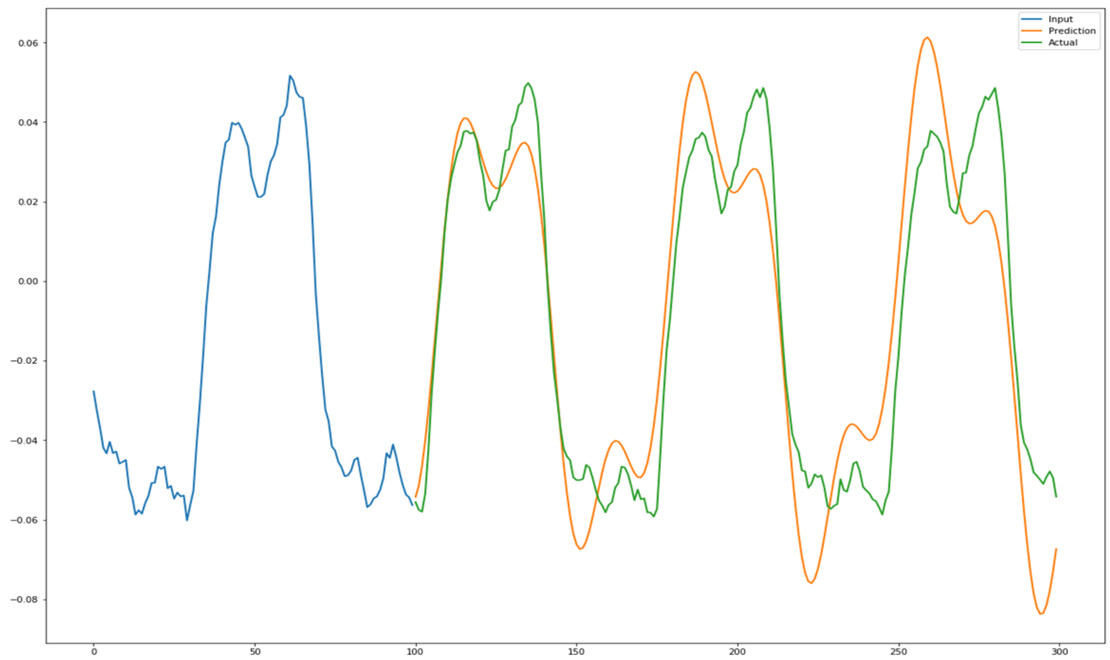
Training process and results - 2

- After 8 epochs, the recurring pattern is captured



Training process and results - 3

- After 64 epochs a more precise trend is captured reasonably well
- Spikes within each period are modeled in the first two prediction cycles



Results - summary

- Input data had a very easy shape to capture, more research and analysis needed for other data types. Our research shows that for a GEO satellite, parameters show a very strong 24 hour periodicity [as found by dr. Alan Higgings]
- Event longer term predictions seem to capture the longer term trend reasonably well. This is despite the fact that after 100 predicted samples, all predictions are based on past predictions
- Training is expensive. When CPU is used the process may take minutes per epoch. With a GPU with appropriate tuning each epoch takes only a few seconds.
- Each prediction takes << 1 second, however longer term predictions are not immediate

Results – applicability and future work

- The main area of application is to create a dynamic limit/anomaly detection framework. Further work is needed to tune the input algorithms for other data types. Initial results are promising though as the model can predict over 100 samples into the future with high accuracy.
- Training models on larger sets of parameters can be expensive
- Need to better quantify prediction performance when running on a CPU vs. a GPU
- Need to better determine the applicability of deep learning models vs. more traditional approaches

Helpful libraries

- Keras
  - Provides a high level API for building neural networks. Uses other libraries such as Tensorflow or Theano for heavy lifting
- Tensorflow
  - Neural network/numerical processing library from google
- Numpy
  - Numerical processing for python
- Matplotlib
  - Invaluable tool for result visualization

References

[1] Alex Krizhevsky et al, ImageNet Classification with Deep ConvolutionalNeural Networks, https://www.cs.toronto.edu/~kriz/imagenet_classification_with_deep_convolutional.pdf

[2] Jason Yosinski et al. Understanding Neural Networks Through Deep Visualization, https://arxiv.org/pdf/1506.06579.pdf

[3] Cardiologist-Level Arrhythmia Detection with Convolutional Neural Networks, Pranav Rajpurkar∗ et al., https://arxiv.org/pdf/1707.01836.pdf

Learning resources

- Deep learning, parts 1 and 2, Rachel Thomas, Jeremy Howard, http://www.fast.ai/
- Coursera.org deep learning specialization, Andrew Ng, https://www.deeplearning.ai/