Comparative Study on Generating and Predicting Swarm Satellite Data by Deep Neural Networks

Yaxin Bi[1], Maja Pavlovic[1], Christopher O'Neill[1], Mingju Huang[2], Xuemin Zhang[3], Jianbao Sun[4]
y.bi@ulster.ac.uk

[1]School of Computing, Faculty of Computing, Engineering and the Built Environment, Ulster University, Belfast, UK
[2]Belfast School of Architecture and the Built Environment, Faculty of Computing, Engineering and the Built Environment, Ulster University, Belfast, UK
[3]Institute of Earthquake Forecasting, China Earthquake Administration, Beijing 100060, China
[4]Institute of Geology, China Earthquake Administration, Beijing 100029, China

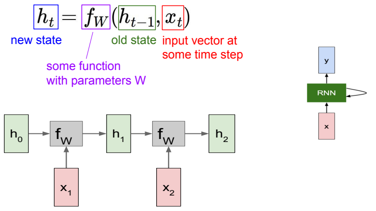
Abstract

The effective detection of seismic precursors from the electromagnetic Swarm satellite data will provide opportunities for generating early warnings of earthquakes and minimizing their social and economic impact. Over the past few decades, substantial efforts have been made to recognize potential characteristics of seismic precursors before seismic events using various data sources. Although there has not been a breakthrough in addressing this challenge, seismologists have continued to hold the theoretical assumption that the Earth's electromagnetic field could contain precursor signs before earthquakes. In this report, we will present the latest developments in anomaly detection algorithms supported by Deep Neural Networks (DNN). These algorithms focus on predicting and generating electromagnetic data from historical Swarm data. We will describe our investigation into the architecture of Recurrent Neural Networks (RNN), particularly emphasizing the development of Long-Short Term Memory (LSTM)-based architectures and a flow-based generative model. We will detail the design of these architectures, their implementation, and compare the predicted and generated results obtained by applying these approaches to Swarm's historical data. Finally, we will outline our methods for detecting anomalies in both synthesized and authentic Swarm data, along with potential applications in identifying seismic precursors using the same synthesized and authentic Swarm data.

Methodology

Recurrent Neural Networks (RNN): Given vectors x, they can be processed by applying a recurrence formula of RNN at every time step:

$$h_t = f_W(h_{t-1}, x_t)$$



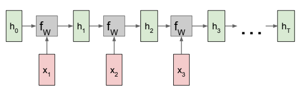
The same function and the same set of weights and bias are used at every time step.

In a Simple Vanilla RNN, the state consists of a single "hidden" vector h_t:

$$h_t = f_W(h_{t-1}, x_t)$$

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$



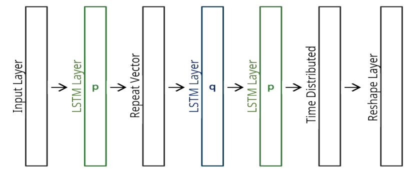
where the activation function is **tanh**, the weight at the recurrent neuron is W_{hh} and the weight at the input neuron is W_{xh}, the output state is y_t.

Issues with RNN:

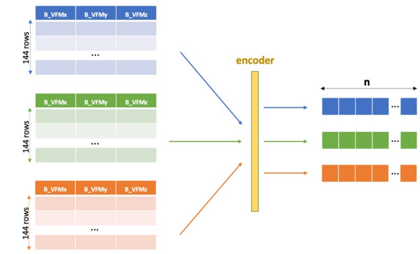
- In theory, RNNs are capable of handling "long-term dependencies", i.e. through carefully picking parameters for them. Unfortunately, it is not practical to manually picking parameters, RNNs don't seem to be able to learn them.
- Long-Short Term Memory networks (LSTMs) – are a special kind of RNNs, capable of learning long-term dependencies.
- LSTMs are designed with the following features to handle the long-term dependency problem:
  - ✓ Repeating modules (neurons) with very simple structure such as a single tanh layer
  - ✓ Remembering information for long periods of time as their default behavior, which is not something they struggle to learn
- They work tremendously well on handling time series data, which are now widely used.

Designed Solution

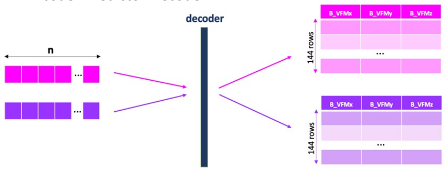
- Three LSTM layers with *p*, *q* and p number of neurons respectively.
- Repeat Vector, Time Distributed and Reshape layers to adjust the shape of data.
- Activation function: tanh with glorot uniform kernel initialiser.
- Loss functions (error metrics): Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE).



- Incorporating LSTMs into Developing an Encoder-Predictor-Decoder



- Encoder-Predictor-Decoder



Swarm Data

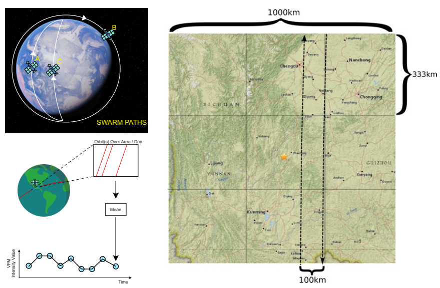


Table 1 – Predict Frames

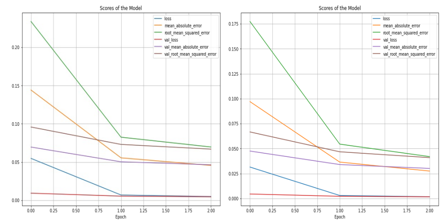
Target	Input A	Input B	Input C	Input D	Output
1 week	7	21	30	-	7
1 month	7	30	60	-	30
6 months	30	90	120	180	180
8 months	180	320	-	-	240

days

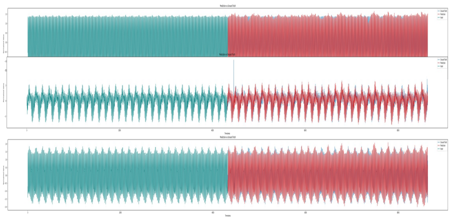
Predict Frames

- Our experiments cover four prediction targets: 1 week, 1 month, 6 months and 8 months.
- We examine multiple input/output data windows.
- For our analysis, we gather all available *MAGx LR* packages between January 2014 and July 2021. As we acquire measurements recorded by Swarm A. All the mentioned packages belong to the *Level 1B* type

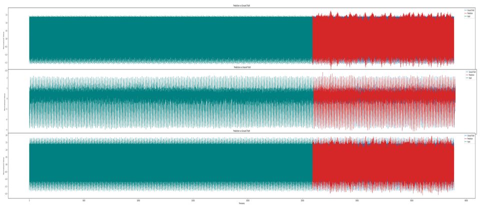
Predict Frames



Prediction: 30 days -> 30 days

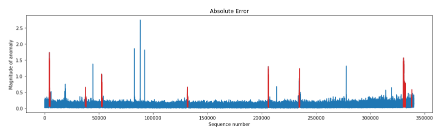


Prediction: 6 months -> 3 months



Anomaly Detection

- To calculate the discrepancies between reconstructed or predicted data and the ground-truth data, we use three metrics: Absolute Error, Area Error and Dynamic Data Warping (DTW).
- Absolute Error allows for examining point discrepancies.
- Area Error and DTW are more useful for detecting collective discrepancies in data.
- Plots of these metrics over time provide an understanding about a similarity between predictions and ground-truth data.




```
Sequence; Timestamp; Length: 584
  4152    29/07/2014 20:00
  4153    29/07/2014 20:10
  4154    29/07/2014 20:20
  4155    29/07/2014 20:30
  4156    29/07/2014 20:40
                ...

Sequence; Timestamp; Length: 695
 52419    30/06/2015 01:30
 52420    30/06/2015 01:40
 52421    30/06/2015 01:50
 52422    30/06/2015 02:00
 52423    30/06/2015 02:10
```

Summary

- We proposed an approach – incorporating LSTM layers to build an Encoder-Predictor-Decoder architecture.
- The experiments demonstrate that Encoder-Predictor-Decoder allows for predicting and reconstructing Swarm data for longer periods of time and also outperforms a Stacked LSTM layers approach which started to perform poorly after 500 time steps and the training process became extremely slow.
- Implementing three similarity metrices for computing discrepancies between predicted results and ground truth Swarm data.