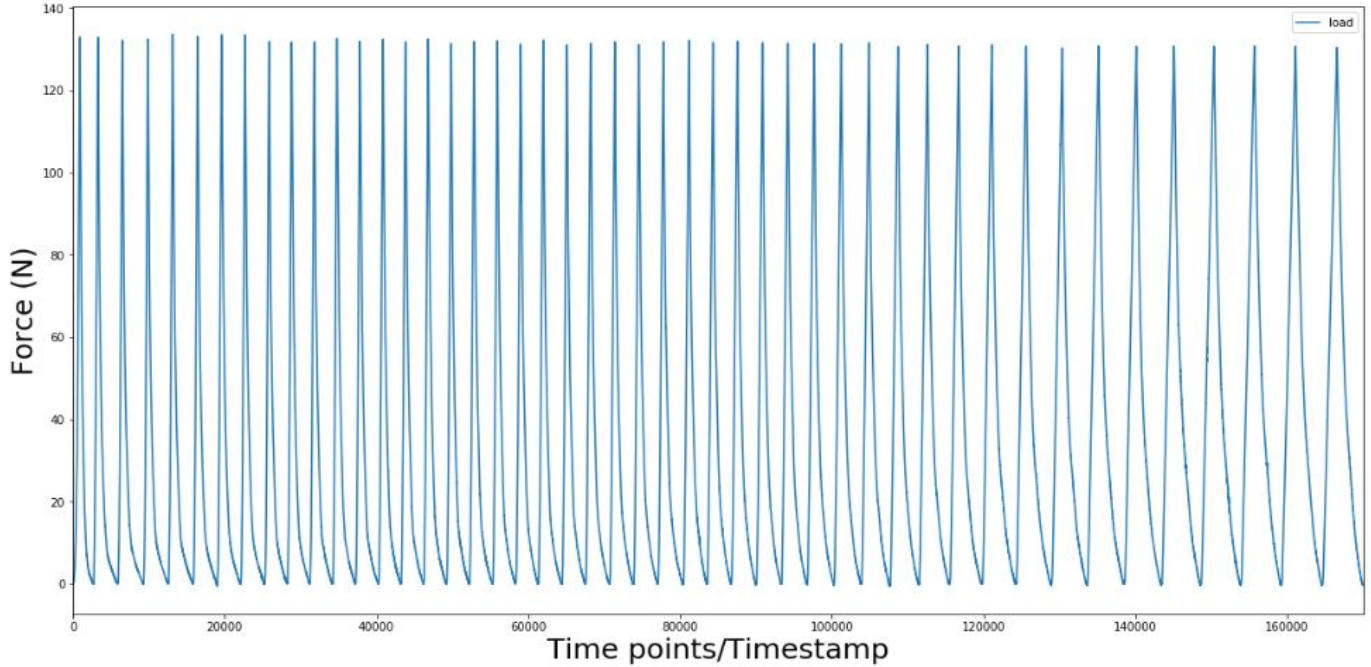
Time-Series Force Cycle Prediction using Machine Learning and Statistical Techniques

Dhananjay Shrouty

Goal

- The goal is to predict how the artificial muscle would behave when subject to cycles of force



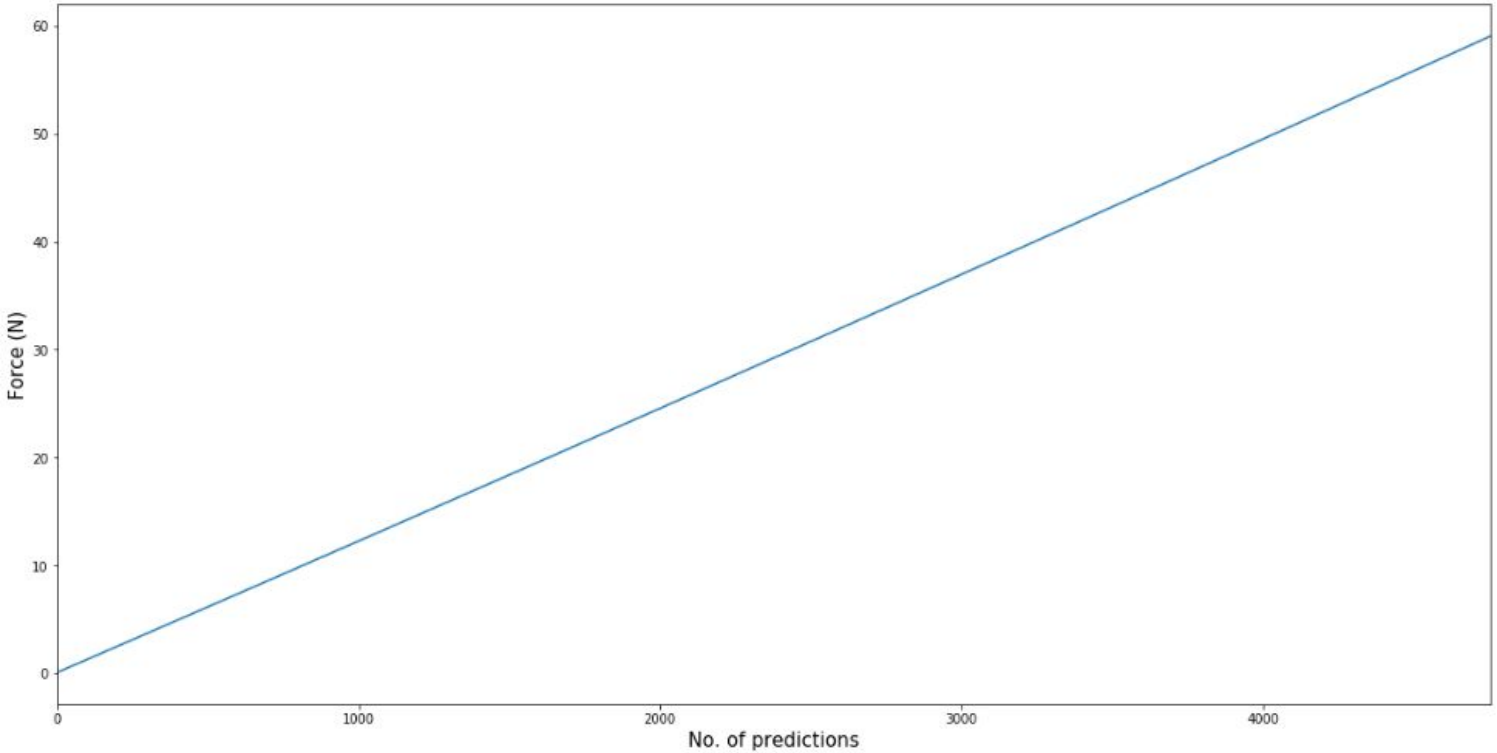
Approaches used

- The approaches used to tackle the problem were
  - Statistical Approaches
    - ARIMA (An upgrade of the ARMA technique)
  - Deep Learning Approaches
    - LSTM prediction
    - Encoder-Decoder prediction

ARIMA Overview

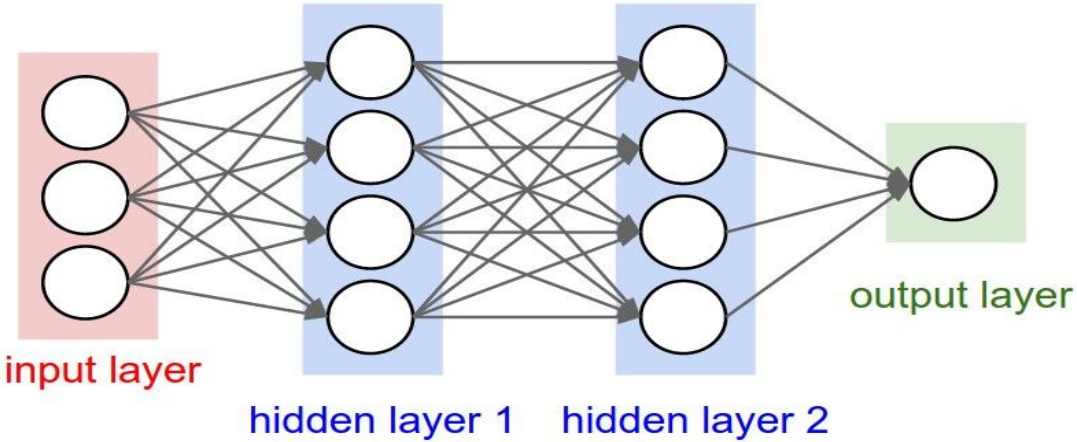
- Stands for Autoregressive Integrated Moving Average models
- A class of statistical models used mainly for Time Series analysis and prediction
- Finds heavy application in the area of short term forecasting
- Autoregressive Models
  - X(t) = A(1) * X(t-1) + A(2) * X(t-2) + E(t)
    - X(t) : time series under investigation
    - A(1), A(2) : autoregressive parameter of order 1 and 2
    - X(t-1), X(t-2) : time series lagged 1 and 2 period(s)
    - E(t) : error term of the model
- Moving Average Models
  - X(t) = -B(2) * E(t-2) -B(1) * E(t-1) + E(t)
    - B(1) : moving average parameter of order 1
    - E(t-1), E(t-2) : random error in the previous 1 and 2 period(s)

First Approach (ARIMA) Results



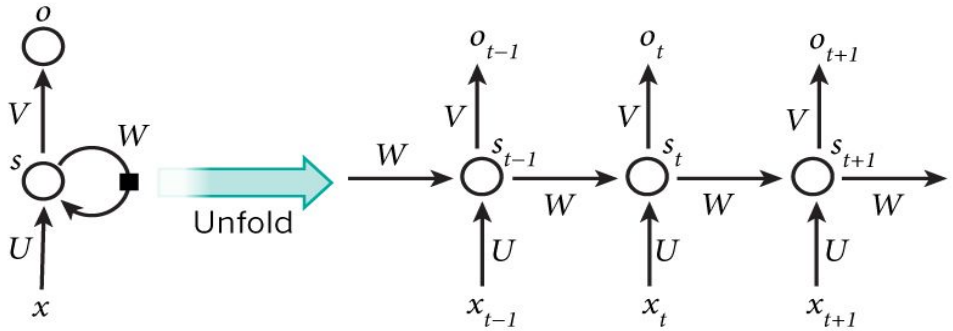
Deep Learning overview (Neural networks)

- Neural Networks are computing systems inspired by biological neural networks that constitute brains.
- Each connection between neurons can transmit a signal to another neuron (node)
- The biggest strength of neural networks is the ability to incorporate non-linearity in the predictions
- The non-linearity helps the neural network to predict the most complex functions
- Neural networks however usually require lots of data and high computation power



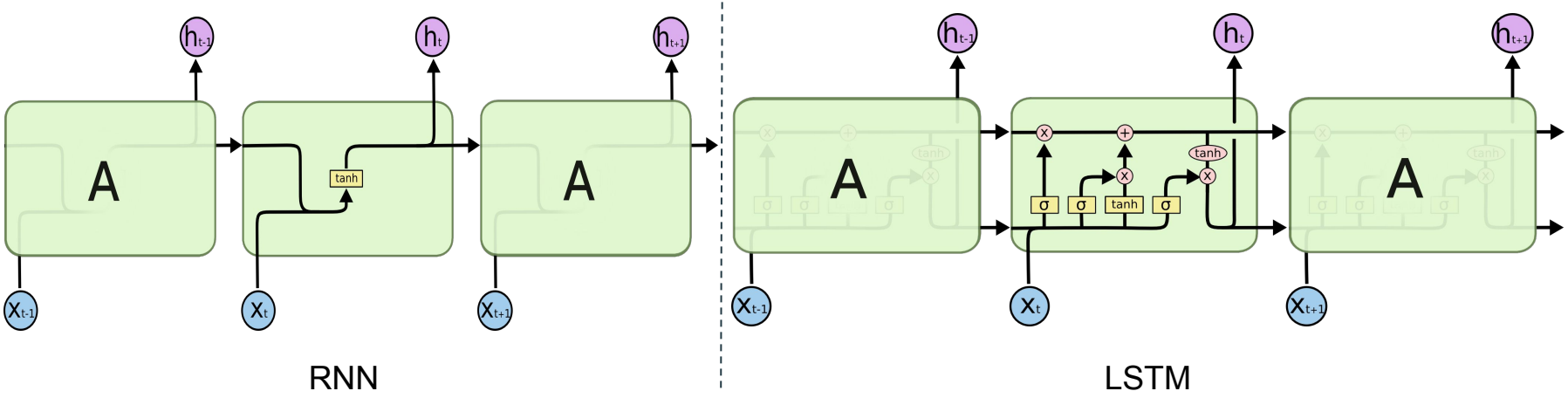
RNN Overview

- Deep learning for NLP (or sequences) and for computer vision use two very different kinds of neural network architectures
  - Computer Vision techniques usually use Convolutional Neural Networks
  - NLP and Sequence techniques usually use RNNs and LSTMs
- RNN (Recurrent Neural Networks)
  - RNN are capable of learning sequences
  - They are good for learning small sequences
  - Can be visualized as a network with memory



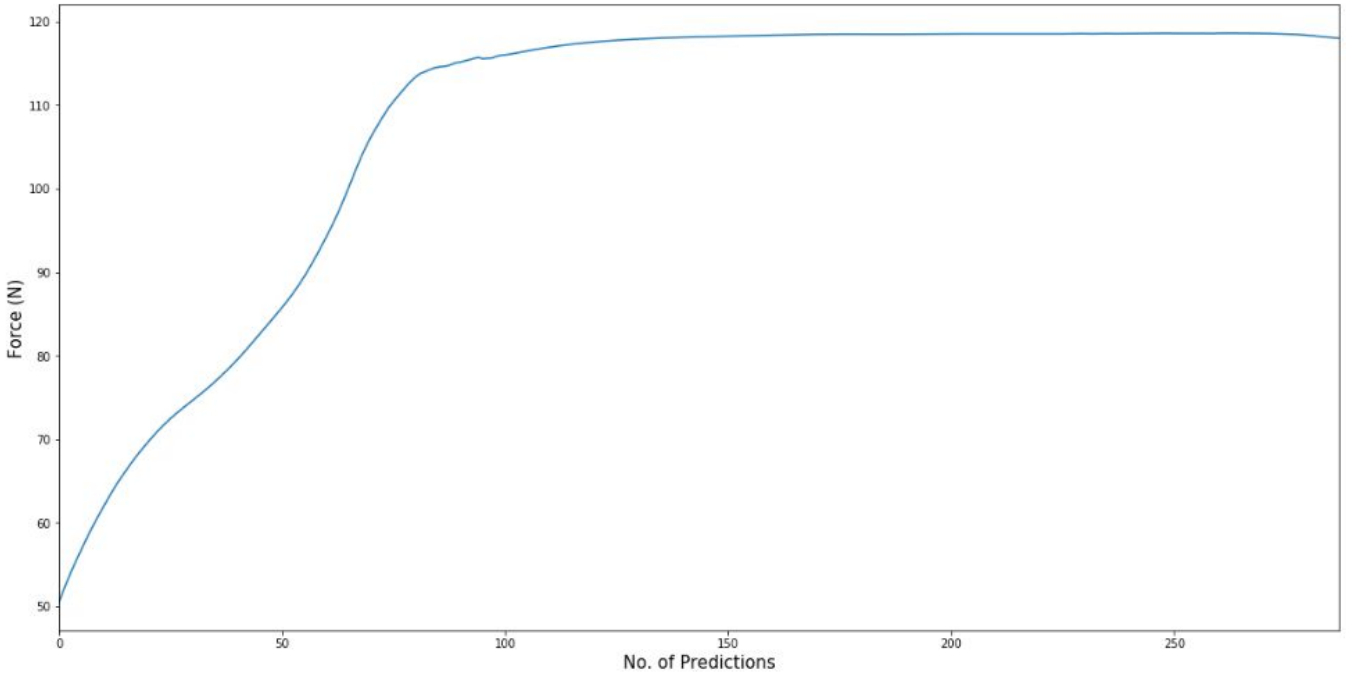
LSTM Networks

- RNN limitations can be that it can learn small sequences
- LSTMs overcome this by using a slightly different internal architecture
- LSTMs therefore perform good in learning longer sequences
- They however cannot overcome problems in multi-step ahead prediction which we'll discuss later



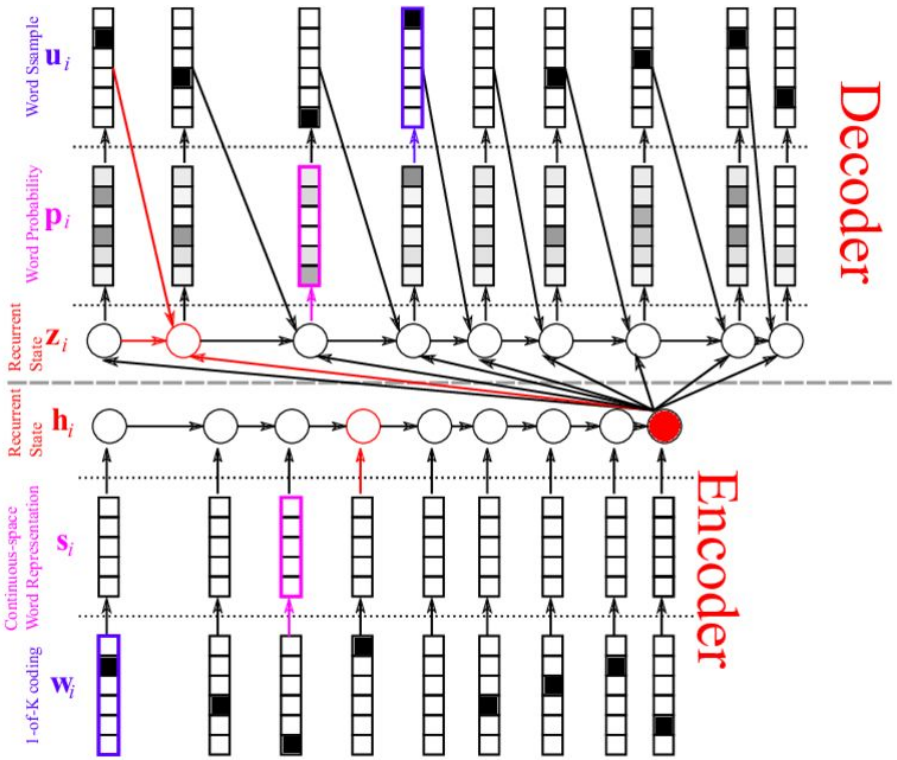
Second Approach (Single LSTM) Results

- The first approach included using a single LSTM layer and using it to predict the results further in the time series

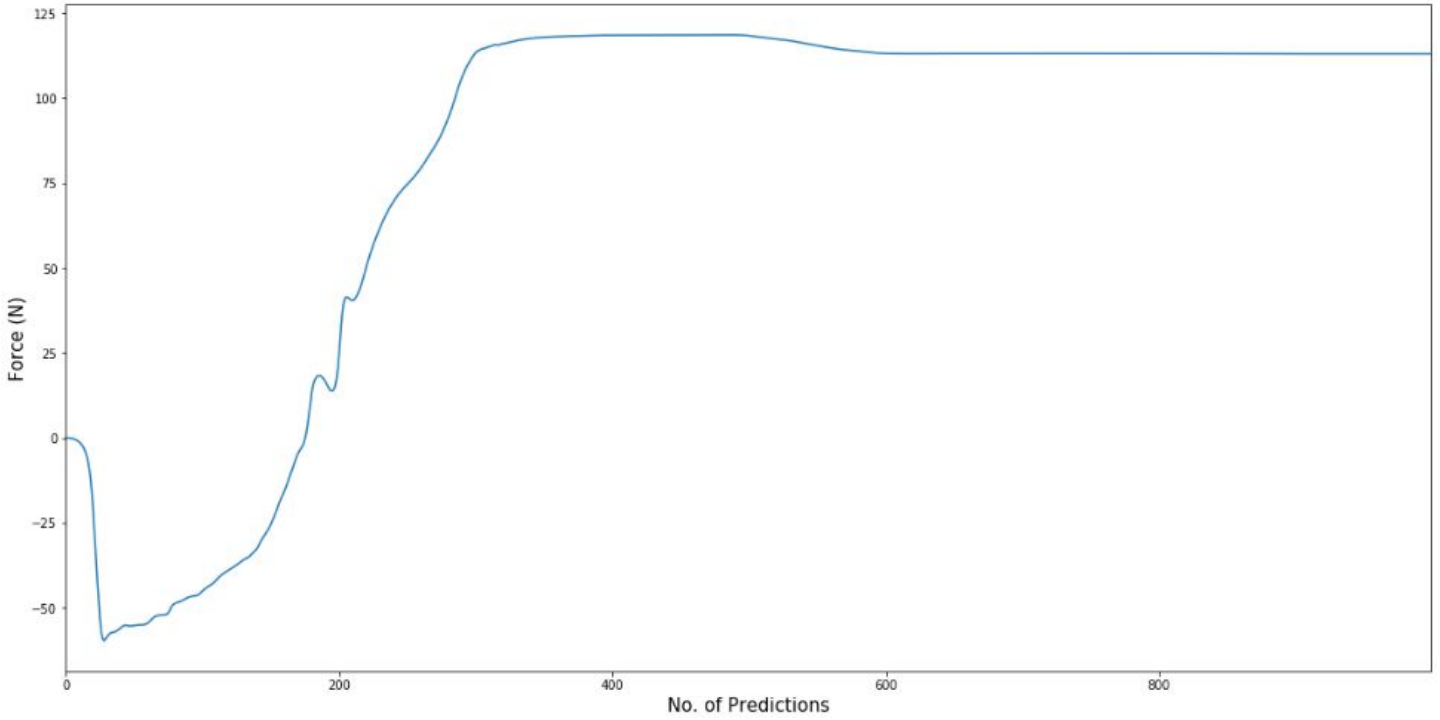


Encoder-Decoder

- A single LSTM is not very effective to predict sequences
- Sequences prediction (in neural machine translation) is done using Encoder-Decoder Approach
- Implementation of Encoder-decoder approach for time series



Encoder-Decoder Results



Improvements and Future work

- Time Series prediction is not an easy task
- Prediction of few steps ahead might be possible but multi-step ahead predictions flatten out as errors amplify when we make predictions out of predictions
- Better sequence to sequence algorithms although the ones used in the project were state of the art
- Choosing a sample from dataset which effectively represents the entire dataset (with some approximation errors) and use the dataset on smaller length prediction algorithms (even encoder/decoder)
- Teacher forcing or Professor forcing - but removes the notion of multi-step ahead prediction
- Multiple Encoder-decoder stacks

Questions?

Thanks!