ANOML

Time-Series Forecasting using Deep Learning Techniques for Performance Parameters of Enterprise Software Systems

Project URL: https://github.com/AnoML/**MultivariateTimeSeriesForecasting**

CSE Final Year Project Group No: 18 (Team AnoML)

W. G. T. Ashen	140048P
P. D. Geethika	140176F
M. G. R. Y. Gunarathne	140181P
T. A. S. Y. Jayathilaka	140260H

Introduction

Time-Series Forecasting Models | Introduction

- Different statistical, probabilistic, and deep learning models in Literature
    - Auto-Regression Models & EMA (ARMA, ARIMA, GARCH)
    - Deep Neural Networks
        - Recurrent Neural Networks
            - Long Short Term Memory (LSTMs)
            - Gated Recurrent Unit (GRUs)
    - Adaptive Short Term Forecasting
    - Adaptive Auto-Regression
    - Adaptive Model Selection
    - Adaptive Model Composition
    - Density Forecast
    - Quantile Regression
    - .....

- In general, statistical models are more suitable for:
    - Identifying linear dependencies or specific nonlinear patterns of past data
    - Forecasting univariate time-series

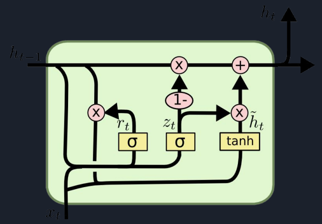
- **Most suitable models for multivariate time-series: LSTMs, GRUs**

LSTMs & GRUs | Introduction

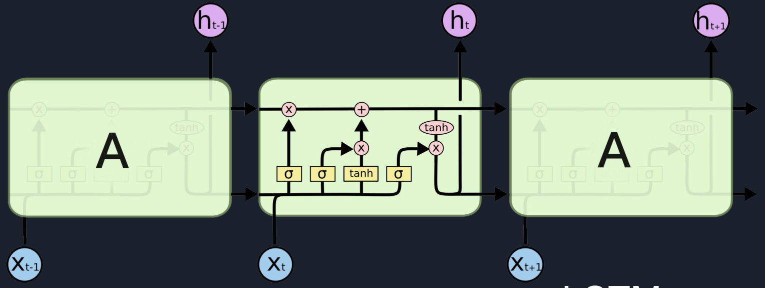
- Standard RNNs are difficult to train for long-term temporal dependencies
  - Gradient of the loss function decays exponentially with time (Vanishing Gradient problem)

- LSTM RNNs solve above problem.
  - Include a memory cell that maintains info in memory for long periods of time
  - Include set of gates to control when information enters, outputs, and forgets from the memory
  - Capable of learning longer-term dependencies

- GRUs are similar to LSTMs.
  - More simplified structure than LSTMs
  - Include set of gates to control when information enters, outputs, and forgets
  - Use fewer gates than LSTMs
  - Computationally more efficient than LSTMs



GRUs



LSTMs

ML Project Goal | Introduction

ANOML

Our goal is to apply and evaluate deep learning techniques for multivariate time-series data in performance domain and provide a comparative analysis

Methodology

1. Data Collection | Methodology

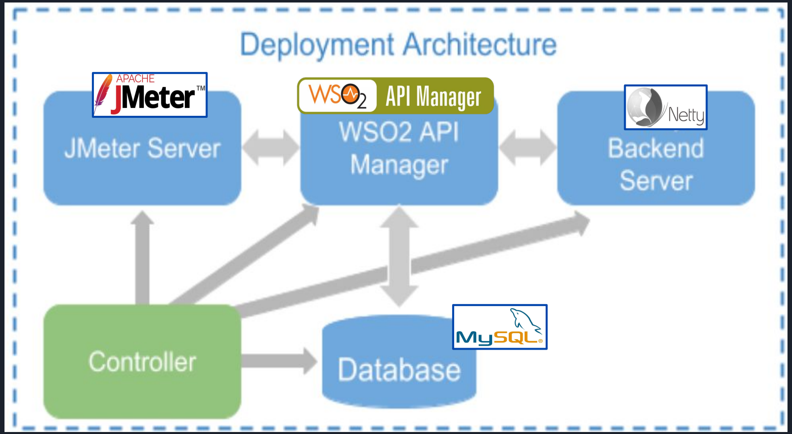
ANOML

Steps in Data Collection

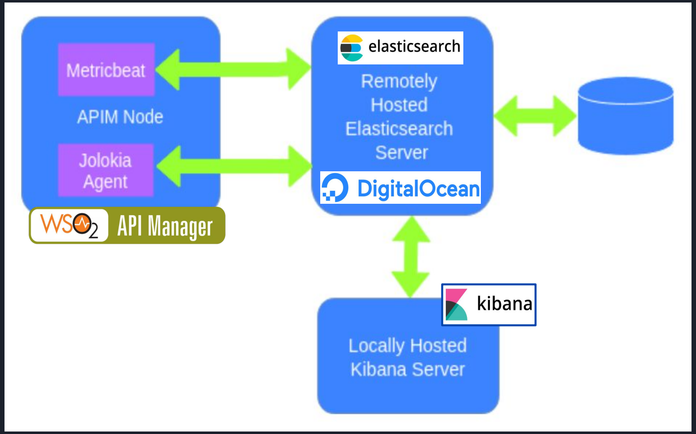
- Deploy WSO2 API Manager locally
- Setup system tools to collect performance data from JVM and server nodes
  - Metricbeat - to collect system parameters
  - Jolokia JVM Agent - to collect JVM parameters
- Simulate normal behaviour of the system using scripts
- Collect data over time

Characteristics of Dataset

- Multivariate
- Time-series
- Unlabeled
- Synthetically-generated
- Read from OS & JVM
- 1 min constant time interval



Deployment Architecture



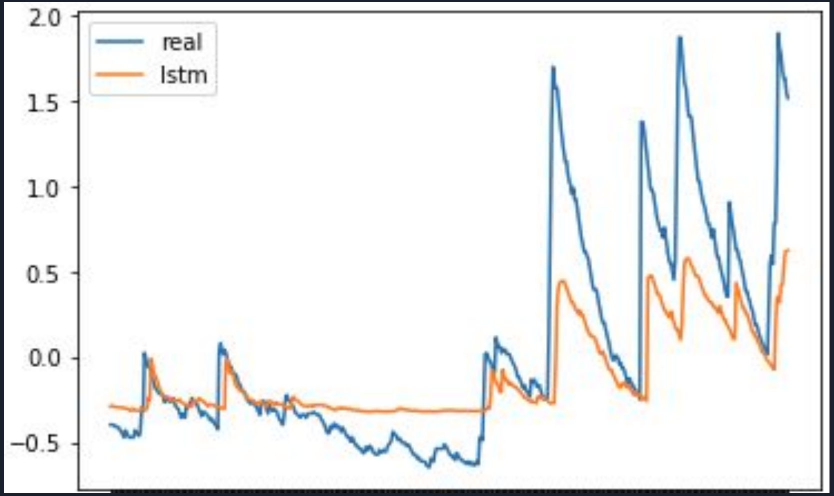
Data Collection Setup in API Manager Node

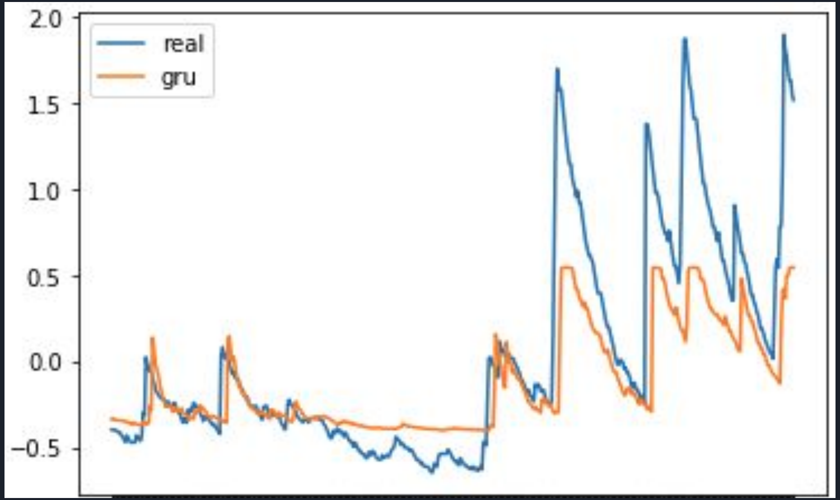
2. **Time-Series Forecasting** | Methodology

- Only System Load related parameters (total 0f 10) are used in this ML project.
  - Describes about System Load Averages
- Dataset is split into 2 proportions for training and testing purposes
  - Training **:** Testing ratio = 2 **:** 1
- Shift the model and feed future values as outputs of model
  - For supervised-training the models
- No feature reduction step in deep learning models
  - Better to have large number of parameters
  - Better to have large number of data points
- Normalize the features to get values into one scale
- Tune hyperparameters of LSTM model using an exhaustive search
  - Find out values for the parameters of the model such as activation function, recurrent activation function, optimizer function, and loss function
- Train models
- Plot actual and predicted behaviours
- Calculate mean absolute error of prediction models w.r.t. actual behaviour

Results & Analysis

Results & Analysis





Prediction using LSTMs

Mean Absolute Error (MAE) : 0.27157166756650 76

Prediction using GRUs

Mean Absolute Error (MAE) : 0.24688981519815 91

- LSTMs and GRUs have relatively low errors in time-series forecasting.
- GRUs take a lower time for training than LSTMs due to simplified architecture.
- According to MAE, GRUs outperform LSTMs (But in general, they perform similarly)

Conclusion

Conclusion

- Time-series forecasting algorithm has the ability to extrapolate patterns outside of the domain of training data.
    - Not many ML algorithms have this capability. Hence, not suitable for time-series forecasting.
- In general, time-series forecasting does not require critical & time-consuming feature engineering techniques.
    - Save time & efforts.
- LSTMs & GRUs require supervised training.
    - Transform time-series by shifting data, so that the observation at the previous time step is used as an input to forecast the observation at the current time time step.
- Hyperparameter tuning is important in LSTMs and GRUs.
- Results show that GRUs outperforms LSTMs.
    - LSTM remember longer sequences than GRUs and outperform them in tasks requiring modeling long-distance relations.
    - But since the considered time series has a short term trend GRU performs better in the considered scenario.
- GRUs are computationally more efficient than LSTMs because of the simplified nature.

Thank You!