*Proceedings of 8th Transport Research Arena TRA 2020, April 27-30, 2020, Helsinki, Finland*

Title of your paper: Capitalize first letter of title and subtitle

# First Authora, Second Authorb[[1]](#footnote-2), Third Authorb

aFirst affiliation, Address, City and Postcode, Country

bSecond affiliation, Address, City and Postcode, Country

**Abstract**

Start abstract text here. Overall, these guidelines explain how a paper must be prepared for publication in the TRA2020 Conference Proceedings. The paper must be no more than 10 pages, including the front page, and placed on A4 format (210 mm × 297 mm) sheets, with margins of 25 mm on all sides. The front page must contain only the title, the authors’ names and affiliations, abstract and keywords. The abstract must be less than 150 words, using Times New Roman 10pt. Submission file type can be MS Word or be Portable Document Format (PDF). The paper should be written in English only. This template must be strictly respected.

*Keywords:* up to six keywords complementing title and abstract separated by semicolons; using Times New Roman 10pt.

**PLEASE NOTE YOUR WHOLE CONFERENCE PAPER CANNOT HAVE MORE THAN 10 PAGES. PLEASE USE TIMES NEW ROMAN, FONT SIZE IS 10 pt.**

* + 1. Nomenclature

A radius of

B position of

C additional nomenclature continue

# Introduction

The main objective of this paper is the development of fault detection schemes that can accurately

estimate possible abnormal operational patterns in railway systems leading to a reduction in LCC. Two

achieve this, a set of anomaly detection schemes have been developed able to identify voltage variations,

the fault location etc.

Anomaly detection refers to the process of finding unusual patterns that do not conform to expected

behavior. These unexpected behaviors are known as anomalies or outliers. There are three basic types of

anomalies, which are the point, the contextual and the collective anomalies. Specifically, if an individual

data instance is too far off from the rest of the dataset, then it is termed as point anomalous. Moreover, if

a data instance is anomalous in a specific context (usually in time series datasets), it is termed as

contextual anomaly. Finally, if a set of related data is anomalous with respect to the whole dataset, then it

is known as collective anomaly.

We study the use of neural based network autoencoders to identify anomalies in time series data. An autoencoder

is an unsupervised learning method that fits well with the purpose of identifiying anomalies. In particular, an autoencoder learns a representation of the input data in a large feature space, and then performs dimensionality reduction, capturing the most representanive features, and then reconstruct the input data based on the features from the reduced feature space. Since anomalies often correspond to non-representative features, an autoencoder trying to reconstruct these anomalies results in a large reconstruction error. Therefore data points with large reconstruction error are considered anomalies.

# **Preliminaries in Neural Networks**

Basic Neural Theory

* + 1. Multivariate Time Series

A multivariate timeseries is a sequence of vectors describing the state of a system at a specific time point.

More formaly an M-dimensional time series X is a sequence of m-dimensional vectors , where is an M-dimensional vector describing the state of the system at time , for . Each dimension corresponds to a feature.

* + 1. Problem Definition

Given a time series , where describing a feature of our multivariate time series, we aim at assigning each point an anomaly score such that the higher the outlier score is the more likely that point is an anomaly. Based on the anomaly score we can set a threshold to distinquish the anomalous from the normal points.

* + 1. AutoEncoders

An autoencoder

* + 1. Anomaly detection Using AutoEncoders
    2. Deep AutoEncoders

1. **Application of Nns in Railway Systems**
   1. **Scenario**
   2. **Implementation**
      1. Preprocessing
      2. Training

Our multivariate time series dataset is standarized so that every feature has a mean of zero and variance of 1. Also we use a rolling average of 10 samples to reduce our input noise. To train our models we will use overlapping multivariate sequences of 100 samples with a step of one as input data,(This choice is made based on the distance of the stations), and as output data we will use the corresponding sequences of the feature we want to reproduce.Also 15% of the training set was excluded for validation.

We trained our models for 100 epochs using the popular Adam optimization algorithm to minimize the our loss function. The loss function we chose for the training of our models was the Huber Loss function which is quadratic near zero and linear elsewhere, in an effort to minimize the effects of outliers in our training loss.

* + - 1. Deep N eural Network Based Autoencoders

We implemented and compared three autoencoders. One based on reccurent neural networks using GRU\* cells,

one using 1D covolutions and one using 1D convolutions with MaxPooling regularization.

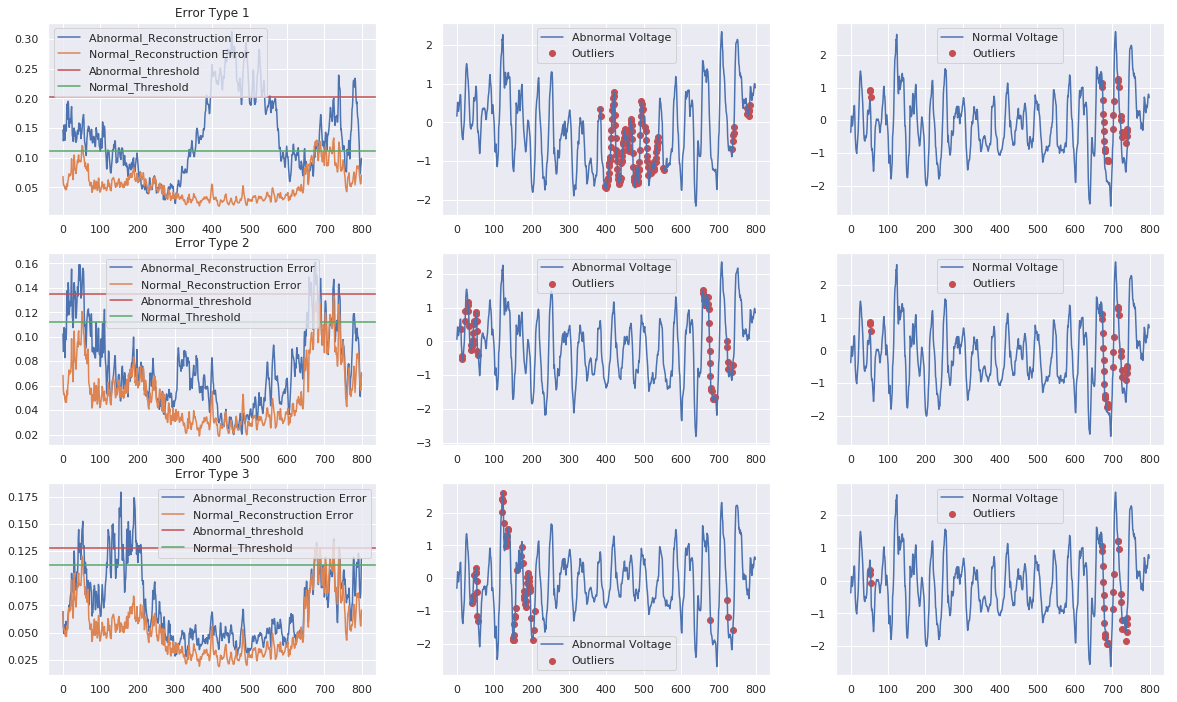
* + - * Reccurent autoencoder

Recurrent neural networks often shortened to RNNs\cite{rnn}, are a class of neural networks which excibit temporal behavior due to directed connections between units of an individual layer. Recurrent neural networks maintain a hidden vector h, which is updated at time step as, where tanh is the hyperbolic tangent function, **W** is the recurrent weight matrix and **I** is the projection matrix. The hidden state h is used to make a prediction , softmax provides a normalized distribution over the possible cases, **σ** is the logistic sigmoid function and **W** is a weight matrix. By using **h** as the input to another RNN, we can stack RNNs, creating deeper architecturesIn the context of our reccurent autoencoder implementation we will use a variance of the recurrent neural model the Gated Recurrent Unit(GRU) which addeses the vanishing gradient problem, commonly occuring in ordinary RNNs. The architecture of our reccurent autoencoder is presented at figure.

* + - * Convolutional Autoencoder
    1. Experiments

We tested our neural network autoencoder on simulated datasets where one of the available substations is in outage. That means that our voltage timeseries present lower than expected values around the area of the faulty substation. For our experiments we performed the same preprocessing of the dataset as in the training set.

As a reconstruction error we use the mean squaered error of our predictions. Below we plot the reconstruction error for each type of simulated error and the detected anomalies well as the anomalies detected in normal and abnormal sequences. The anomaly threshold is set at the mean plus half standard deviation of the reconstruction error.



1. \* Corresponding author. Tel.: +000-000-000-0000;

   *E-mail address:* author@institute.xxx [↑](#footnote-ref-2)