Exposé for a Studienprojekt
A reproducibility study of state-of-the-art classifiers for deep
learning on time series

Simon Maximilian Friedo
April 18, 2024

1 Introduction

The importance of deep learning in various domains has been indisputable as it has become the
focus of a steadily increasing amount of research publications. A time series is a sequence of
real valued data points that are typically collected or recorded over time. The sources of the time
series data sets used as benchmarks vary widely, from instance image data, sensor data or sim-
ulated/synthetic data, showing the wide range of use cases for time series classification. A time
series may be either univariate, a single value $x_i$ is associated with each time stamp, or multivari-
ate, $m$ values $x_i = (x_i^{(1)}, \ldots, x_i^{(m)})$ are associated with each time stamp, where $m$ is the number
of channels.

Time series classification (TSC) is an important problem for the analysis of a time series and
uses machine learning models to predict a class for a time series. TSC finds applications in vari-
ous fields, including finance [TPT+17a, TPT+17b], healthcare [RR14, ROC+18], and industrial
automation [TZT+21, ZWZ+21], among others. Effective time series classification enables data-
driven decision-making, anomaly detection, and predictive analytics in diverse domains.

Recent developments in the domain of TSC have reported innovative new achievements that define a
new standard on the UCR time series benchmark archive [DKK+18]. However, this unprecedented
surge in research publications has also unveiled some troubling trends. Despite the remarkable
progress there have been some research projects that report seemingly unattainable benchmark
results, while having flawed methodologies, like insufficient model training or overfitting the test
data.

One reason for this is the tendency to prioritize maximizing test accuracy over multiple epochs,
leading to overfitting and unreliable outcomes, which unfairly favor the perceived advantage over
conventional models. This causes reasonable doubts in the reported results.

Another concerning observation is the selective evaluation of models, using only subsets of the
benchmark datasets without a plausible justification. Unjustified selective sampling can be in-
terpreted as “cherry-picking” the datasets and only using those that are favorable for the models’
performance. Without transparently disclosing the reason behind dataset selection, the validity
and generalizability of the results remain questionable.

Furthermore, the lack of availability of source code poses a significant obstacle to reproducibility
in the field. Many papers fail to provide access to their code base, despite relying on widely-used
frameworks like TensorFlow and PyTorch. This impedes the ability to verify the reported results
and build upon existing work.

In light of these worrisome trends, we will review multiple state-of-the-art publications with regard
to their reproducibility and potential pitfalls. The chosen publications all provide access to their
source code, which is an essential requirement for their reproducibility. This study project will
provide an insight into the current state of deep learning for time series classification and address
potential issues with the reproducibility of the reviewed publications.

1.1 State-of-the-art classifiers

In this study project, we aim to assess the reproducibility of five published research papers, that
claim to outperform state-of-the-art approaches for classifying multivariate time series (MTS). All
of those papers have made their source code public, which is essential for assessing the reproducibil-
ity and correctness of the implementation. They are an excerpt of the papers surveyed by Navid
Mohammadi Foumani in “Deep Learning for Time Series Classification and Extrinsic Regression:
“A New Attention Mechanism to Classify Multivariate Time Series” [HC20] by Yifan Hao and Huiping Cao proposes a novel Cross Attention Stabilized Fully Convolutional Neural Network (CA-SFCN) for classifying multivariate time series. The self-proclaimed goal was to capture long-range dependencies of the time-series sequences to represent interactions of multiple variables in features. To achieve this goal, they used a temporal attention mechanism to extract short- and long-term memory and designed a variable attention algorithm to select relevant variables at each time step. To evaluate the approach, they did extensive experiments on 14 different real-life MTS datasets [KMDH19] and compared their approach to 16 different approaches. They claim to have shown that their approach outperforms state-of-the-art classification methods and that the cross-attention mechanism achieves the best performance across the tested approaches.

Seyed Navid Mohammadi Foumani et al. proposed a classifier consisting of multiple disjoint Convolutional Neural Networks (CNNs) in combination with a novel “1+1D” filter block, which emphasizes the interaction between dimensions. In their paper “Disjoint-CNN for Multivariate Time Series Classification” [FTS21], they claim to outperform all the tested established CNN-based classifiers for multivariate time series in regard to accuracy and rival state-of-the-art MTSC models. For comparison, they tested six CNN-based and two MTSC models and used 26 of the 30 UEA Multivariate time series archive datasets [BDL18].

The main contribution of the MACNN architecture, introduced by Wei Chen and Ke Shi in “Multi-scale Attention Convolutional Neural Network for time series classification” [CS21] consists of the Multi-scale Attention Mechanism (MAM), that is a strategy which enhances useful feature maps and suppresses less useful ones by learning the importance of each feature map, and the MACNN architecture, which is built on stacked Multi-scale Attention (MA) modules consisting of a multi-scale and an attention block. The goal of this architecture is to capture different scales of information with the multi-scale block and to focus on the important feature maps using the attention block, which is developed on channel dimension. Sufficient experiments have been conducted and are evaluated with different metrics. For the experiments, they chose 85 datasets from the UCR archive in accordance to the experiment settings described by Bagnall et al. [BLB17]. The achieved results show that MACNN achieves remarkable performance and outperforms other approaches by a large margin.

The work presented in “Paying Attention to Astronomical Transients: Introducing the Time-series Transformer for Photometric Classification” [AJM21] by Tarek Allam Jr. and Jason D. McEwen makes use of an architecture called transformer that was introduced by Vaswani et al. [VSP17]. They presented a time-series transformer architecture they call “t2” that has several key differences to the original transformer architecture. They opted to remove the decoder, as a single transformer-block is sufficient for classification. Two additional layers have been introduced prior to the positional encoding unit consisting of Gaussian Process Interpolation and Convolutional Embedding. A Concatenation layer that is able to add an arbitrary number of additional features to the network works in conjunction with these two layers. With those layers in combination with a typical transformer-block they claim to be able to achieve results comparable to the state-of-the-art in classification. A direct comparison to other methods is understandably not possible since each classifier has been evaluated on different conditions and data. They emphasize their performance on imbalanced data, which is one aspect in which the t2 architecture does well.

The Task-Aware Reconstruction Network (TARNet) introduced in “TARNet: Task-Aware Reconstruction for Time-Series Transformer” [CZS22] is a new Model using transformers to learn task-aware data reconstruction. A data driven masking strategy that uses self-attention score distribution from end task training to identify timestamps that are important to the end task. The data at those timestamps is then masked out and reconstructed by the reconstruction network. This is done in an effort to make the reconstruction network task-aware. Their experiments have shown that the TARNet outperforms state-of-the-art baseline models across all evaluation metrics. For evaluation, they used the accuracy and Root Mean Squared Error metric and datasets that are available in the UEA Archive [BDL18], UCI Machine Learning Repository [DG17, HOWC13] and the Monash University, UEA, UCR Time Series Regression Archive [TBPW20].
2 Project Goal

The goal of this study project is to conduct a comprehensive examination of the five aforementioned papers concerning their reproducibility and correctness. The methodologies outlined in each paper will be reviewed, with emphasis placed on verifying the correctness of their implementation. The primary aim is to find any discrepancies or deviations from the described behavior, as well as searching for signs of inadequate model training, such as mixing the test and validation data. Additionally, the experimental settings and results will be analyzed to discover any potential sources of error, such as cherry-picking of datasets and mismatches in evaluation metrics among competing methodologies. The overreaching goal is to collect all of those potential issues.

3 Approach

To assess the reproducibility of a paper, we will analyze the methodology and corresponding source code for potential pitfalls.

One of those pitfalls concerns the wrong selection of train/validation/test split, potentially leading to inadequate model training. The UCR/UEA datasets used for training have fixed train and test splits and do not provide a validation split of the data. The purpose of a validation split in training a classifier is to assess the performance of the model during the training process and to tune hyperparameters effectively. It is essential that the evaluation split is an independent dataset that is not used for training the classifier. The validation split is also used to prevent overfitting of the classifier to the training data. The correct way to obtain a validation split from the provided data sets would be to use a percentage of the train data. Many of the datasets only provide a few dozen samples, causing some researches have passed the test data as validation data, which results in overfitting. To increase the proportion of the train data, some researches have reshuffled the test and train data which in turn causes the results to be incomparable to other papers.

Another pitfall some researchers have fallen victim to is the pitfall of faulty model selection. This pitfall is selecting and saving the best accuracy and model when testing the model on the test data during training on multiple epochs. This will also result in overfitting the model to the test data. The correct way to select the model would be to select the model that performs best on training/validation data.

The third pitfall involves hyperparameter selection, whereby hyperparameters are hardcoded based on the dataset rather than learned from the validation data, potentially skewing test performance. This is problematic because it could indicate that optimal hyperparameters were chosen to achieve the best test performance. This pitfall is avoided by having fixed hyperparameters that are independent of the data set or learned form the validation split.

The next pitfall consists of omitting datasets without clear reasoning and only using a subset of the datasets on which the proposed classifier performs best, misrepresenting the classifier’s accuracy. If there isn’t a clear reasoning for the selection of the subset, the impression of “cherry-picking” arises, leaving reasonable doubt in the reported performance comparison to other models, as this misrepresents the performance of the classifier.

The final pitfall concerns the choice of metrics, which should be standardized for comparability with other studies. Using different metrics renders cross-study comparisons unreliable. To be comparable to other papers using different metrics the results must be rerun under the new metric.

The initial stage in analyzing a paper for the aforementioned pitfalls consists of conducting a thorough examination of the methodology as described within the paper. We need to understand why certain datasets were left out and whether the reasons are justified. Additionally, it’s important to look into the choice of the metric they used to evaluate the results and whether the same metric was used across all the studies used for comparison. After examining the methodology as described in the paper for potential pitfalls, it is crucial to analyze the provided source code. This examination should consider aspects such as model selection, the choice of the train/validation/test data split and the selection of hyperparameters. Furthermore, we will try to run the provided code with the goal of replicating the results of the paper.
References


