Warsaw University of Technology

FACULTY OF
MATHEMATICS AND INFORMATION SCIENCE

# Bachelor's diploma thesis

in the field of study Computer Science and Information Systems

Thesis title

Author One

student record book number 123456

Author Two

student record book number 654321

thesis supervisor

title/degree Name Surname

consultation (optional)

academic title/degree Name and Surname

WARSAW 2022

**Abstract**

ENGLISH TITLE

Loremmmm ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumyeirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diamvoluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumyeirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diamvoluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

**Keywords:** keyword1, keyword2, ...

**Streszczenie**

POLISH TITLE

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumyeirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diamvoluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumyeirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diamvoluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

**Słowa kluczowe:** slowo1, slowo2, ...

# Contents

# Introduction

What is the thesis about? What is the content of it? What is the Author's contribution to it?

WARNING! In a diploma thesis which is a team project: Description of the work division in the team, including the scope of each co-author's contribution to the practical part (Team Programming Project) and the descriptive part of the diploma thesis.

# 1. Foundations

## 1.1. Time series classification

Time series is an ordered collection of observations indexed by time.

$$X = (x_t)_{t \in T} = (x_1, ..., x_T), \ x_t \in \mathbb{R}$$

The time index $T$ can represent any collection with natural order. It can relate to point in time when the measurement was observed, or it can represent a point in space measured along X axis. We assume that indices are spaced evenly in the set $T$. The realisation or observation $x_t$ in the times series is a numerical value describing the phenomena we observe, for example amplitude of a sound, stock price or y-coordinate. Time series classification is a problem of finding the optimal mapping between a set of time series and corresponding classes.

## 1.2. Transfer learning

# 2. Related works

In this chapter we would like to describe several algorithms used in time series classification. We will also recall theoretical definitions and distinctions used to describe transfer learning.

## 2.1. Dynamic Time Warping with k-Nearest Neighbour

The Dynamic Time Warping with k-Nearest Neighbour classifier uses a distance based algorithm with a specific distance measure. A DWT distance between time series $X^1$, $X^2$ of equal lengths is:

$$DTW(X^1, X^2) = \min\{\sum_{i=1}^{S} dist(x^1_{e_i}, x^2_{f_i}) : (e_i)_{i=1}^{S}, (f_i)_{i=1}^{S} \in 2^T\}$$

subject to:

- $e_1 = 1, f_1 = 1, e_S = N, f_S = N$
- $|e_{i+1} - e_i| \leq 1$, $|f_{i+1} - f_i| \leq 1$

The measure defined above, used in k-Nearest Neighbour classifier is often used as a benchmark classifier.

## 2.2. Multi Layer Perceptron

The Multi Layer Perceptron (MLP) is the first artificial neural network architecture proposed and can be used for time series classification task. The MLP network can be formally defined as a composition of *layer* functions. The output of the function is a vector that usually models the probability distribution over the set of classes.

$$MLP(X; \theta_1, \ldots, \theta_M, \beta_1, \ldots, \beta_M) = L_M(\ldots L_2(L_1(X; \theta_1, \beta_1); \theta_2, \beta_2); \theta_M, \beta_M)$$

Each layer $L_i : \mathbb{R}^M \to \mathbb{R}^N$ is a function that depends on the parameters $\theta \in r^{M \times N}, \beta in\, \mathbb{R}^N$

$$L_i(X, ; \theta_i, \beta_i) = f_i(X\theta_i + \beta_i)$$

Function $f_i : \mathbb{R}^N \rightarrow \mathbb{R}^N$ is an arbitrary chosen non-linear function. The number of layers and dimensions of weights are also an arbitrary choice, except for the weights is last and first later. The weights is last and first later have to match the dimensionality of input time series (the length of time series) and number of classes.

The disadvantage of using Multi Layer Perceptrons for time series classification is that the input size is fixed. All time series is the training data must have the same length. In transfer learning, this means that if we want to reuse the source network (or a set of first layers from the network), the target dataset must consists of time series of the same length.

The MLP architecture fails at understanding the temporal dependencies. Each input values in the time series is treated separately, because it is multiplied by a separate row in the weight matrix.

## 2.3. Convolutional Neural Networks

Convolutional Neural Networks are widely used in image recognition. A convolution applied for a time series can be interpreted as sliding a filter over the time series. A convolutional layer is a set of functions called convolutions or filters. The filter is applied at a given point, taking into account values that surrounds the point.

To define the convolution operation, let's assume the input is a matrix $X \in \mathbb{R}^{(N_1,\ldots,N_K)}$. In case of images, number of dimensions $K$ is often equal to 3 (height, width, channels), for univariate series we can assume just one dimension, and for multivariate time series wee need two dimensions - (feature, time). The filter consist of a matrix of weights $M \in \mathbb{R}^{(P_1,\ldots,P_K)}$. Usually, $P_i$ are odd numbers, so that we can index the matrix with symmetrical numbers: $(\frac{-P_i+1}{2}, \frac{-P_i+3}{2}, \ldots, 0, \ldots, \frac{P_i-1}{2})$. The 0 index marks the center of the matrix.

Finally the convolution $*$ is defined as follows:

$$(X * M)_{i_1,\ldots,i_M} = \sum_{k_1=\frac{-P_1+1}{2}}^{\frac{P_1-1}{2}} \cdots \sum_{k_M=\frac{-P_K+1}{2}}^{\frac{P_K-1}{2}} M_{k_1,\ldots,k_M} X_{i_1+k_1,\ldots,i_M+k_M}$$

In the first layer of convolutional neural networks used for univariate time series classification, the filter is one-dimensional. The output of the first for one time series layer has dimensions (length of time series - the length of the filter - 1, number of filters). Below we define the value of the output for filter $i$

$$y_{t,i} = f_i([\theta_1^i, \ldots, \theta_M^i] * [X_t, \ldots, X_{t+M-1}])$$

The function $f_i$ and weights $\theta = [\theta_1^i, \ldots, \theta_M^i]$ are different for each filter. The same filter is applied over the whole length of time series. This is called *weight sharing* and it enables the patterns regardless of the position in the time series.

The architecture of the convolutional layer is not dependent of the number of size of the input data. Regardless the size of input data, number of filters and size of filters remain the same, only the output sizes depends on the input size. Therefore, if the convolutional layer is succeeded by layers with the same property, like other convolutional layers or Global Pooling with Dense Layer (see section 2.4), the whole network may be invariant to the input sizes. Such networks may be interesting in terms of transfer learning, as the sizes of time series in the source task and in the target task do not have to match.

## 2.4. Fully Connected Networks

# Bibliography


[1] A. Author, *Title of a book*, Publisher, year, page–page.

[2] J. Bobkowski, S. Dobkowski, Title of an article, *Magazine X, No. 7*, year, PAGE–PAGE.

[3] C. Brink, Power structures, *Algebra Universalis 30(2)*, 1993, 177–216.

[4] F. Burris, H. P. Sankappanavar, *A Course of Universal Algebra*, Springer-Verlag, New York, 1981.