



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

APLIKACE STROJOVÉHO UČENÍ PRO PREDIKTIVNÍ ÚDRŽBU V PRŮMYSLU 4.0

APPLICATIONS OF MACHINE LEARNING IN PREDICTIVE MAINTENANCE OF INDUSTRY 4.0

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Tadeáš Navrátil

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Karel Horák, Ph.D.

BRNO 2023

Diplomová práce

magisterský navazující studijní program **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

Student: Bc. Tadeáš Navrátil

ID: 211164

Ročník: 2

Akademický rok: 2022/23

NÁZEV TÉMATU:

Aplikace strojového učení pro prediktivní údržbu v Průmyslu 4.0

POKYNY PRO VYPRACOVÁNÍ:

Téma se zabývá metodami strojového učení v oblasti prediktivní údržby využívané v konceptu Průmyslu 4.0. Cílem práce je jednak zmapovat třídy metod strojového učení v oblasti Průmyslu 4.0 zahrnující mj. i vizuální inspekci (detekce defektů, anomálií, rozpoznání vzoru) a jednak především detailně rozpracovat oblast strojového učení pro prediktivní údržbu.

1. Nastudujte oblast I4.0 a sestavte zevrubnou rešerši metod strojového učení podle oblasti použití.
2. Po domluvě s vedoucím práce vyberte vhodnou úlohu typu prediktivní údržba pro demonstrativní aplikaci praktické části práce.
3. Pro úlohu prediktivní údržby převezměte volně dostupný nebo alternativně sestavte vlastní anotovaný dataset a navrhnete experiment ověřující aplikovatelnost vybrané konkrétní metody.
4. Re-implementujte popř. modifikujte podle potřeb experimentů vybranou metodu pro predikci anomálních stavů datasetu do uživatelsky schůdného prostředí GUI.
5. Proveďte a vyhodnoťte experiment včasné detekce anomálních stavů datasetu z hlediska přesnosti (čtyřpolní tabulka/AUC/F1) i výpočetní náročnosti.

DOPORUČENÁ LITERATURA:

1. Theissler A., Pérez-Velázquez J., Kettelgerdes M., Elger G. Predictive maintenance enabled by machine learning: Use cases and challenges in the automotive industry. Reliability Engineering & System Safety, 2021, ISSN 0951-8320.
2. One-class classification - Concept-learning in the absence of counter-examples. URL: <http://homepage.tudelft.nl/n9d04/thesis.pdf>

Termín zadání: 6.2.2023

Termín odevzdání: 17.5.2023

Vedoucí práce: Ing. Karel Horák, Ph.D.

doc. Ing. Petr Fiedler, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Práce zpracovává algoritmy strojového učení pro použití v konceptu Průmysl 4.0. Největší důraz je kladen na prediktivní údržbu a vizuální inspekci. V teoretické části je práce zaměřena na literární řešerši metod strojového učení v oblasti detekce anomálií v časových řadách a obrazových datech.

V praktické části se zabývá reimplementací vybraných metod a jejich zhodnocení pomocí matice záměn a metrik, které z ní vycházejí.

KLÍČOVÁ SLOVA

Strojové učení, Průmysl 4.0, Průmyslová revoluce, Prediktivní údržba, Vizuální inspekce, Detekce anomálií, Autoenkodéry

ABSTRACT

The thesis develops machine learning algorithms for use in the Industry 4.0 concept. The main focus is on predictive maintenance and visual inspection. In the theoretical part, the thesis focuses on a literature search of machine learning methods in the field of anomaly detection in time series and image data.

The practical part deals with the reimplementation of the selected methods and their evaluation using the confusion matrix and metrics based on it.

KEYWORDS

Machine Learning, Industry 4.0, Industrial Revolution, Predictive Maintenance, Visual Inspection, Anomaly detection, Autoencoders

NAVRÁTIL, Tadeáš. *Aplikace strojového učení pro prediktivní údržbu v Průmyslu 4.0*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2023, 134 s. Diplomová práce. Vedoucí práce: Ing. Karel Horák, Ph.D.

Prohlášení autora o původnosti díla

Jméno a příjmení autora: Bc. Tadeáš Navrátil
VUT ID autora: 211164
Typ práce: Diplomová práce
Akademický rok: 2022/23
Téma závěrečné práce: Aplikace strojového učení pro prediktivní údržbu v Průmyslu 4.0

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora*

* Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing.Karlovi Horákovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Obsah

Úvod	19
1 Průmysl 4.0	20
1.1 Historie	20
1.1.1 První průmyslová revoluce	20
1.1.2 Druhá průmyslová revoluce	21
1.1.3 Třetí průmyslová revoluce	22
1.1.4 Čtvrtá průmyslová revoluce	23
1.2 Představení konceptu Průmysl 4.0	23
1.3 Vědecké instituce	24
2 Strojové učení v průmyslu 4.0	26
2.1 Oblast prediktivní údržby	26
2.1.1 Předchůdci prediktivní údržby	26
2.1.2 Prediktivní údržba	27
2.1.3 Detekce anomálií v prediktivní údržbě	27
2.1.4 Metoda Long Short Term Memory	29
2.1.5 Metoda Sub-LOF	31
2.1.6 Metoda DWT-MLEAD	33
2.1.7 Metoda DeepAnT	34
2.1.8 Metoda Sub-IF	35
2.2 Oblast vizuální inspekce	36
2.2.1 Autoenkodéry	37
2.2.2 Metoda Generative Adversarial Network	38
3 Datasetsy	39
3.1 Typy datasetů	39
3.2 Zdroje datasetů	39
3.3 Představení konkrétních datasetů	40
3.3.1 Pump sensor data	40
3.3.2 NASA Bearing Dataset	44
3.3.3 Skoltech Anomaly Benchmark	44
3.3.4 Preventive to Predictive Maintenance	45
3.3.5 Ostatní datasetsy	45
3.4 Generátory datasetů	50
3.4.1 GutenTAG	50
3.4.2 MGAB	50

3.5	Generovaný dataset	51
4	Ověřovací experimenty	53
4.1	Metoda Long Short Term Memory	53
4.1.1	Pokusy s datasetem z generátoru GutenTAG	54
4.1.2	Adaptace algoritmu pro dataset Dataset pump sensor	56
4.2	Prediktivní metoda s využitím CNN	58
4.2.1	Pokusy s datasetem z generátoru GutenTAG	58
4.2.2	Adaptace algoritmu pro dataset Dataset pump sensor	60
4.3	Algoritmy Random forest a Linear regression	62
4.3.1	Pokusy s datasetem z generátoru GutenTAG	62
4.3.2	Adaptace algoritmu pro dataset Dataset pump sensor	62
4.4	Isolation Forest	64
4.4.1	Pokusy s datasetem z generátoru GutenTAG	64
4.4.2	Adaptace algoritmu pro dataset Dataset pump sensor	64
4.5	Local outlier factor	65
4.6	Autoenkodér	68
5	Implementace vybrané metody	69
5.1	Představení algoritmů	69
5.2	Hyperparametry autoenkodérů	72
5.3	Pokusy se základním nastavením	73
5.4	Vliv změny hyperparametrů	76
5.4.1	Vliv velikosti dávky (Batch size)	77
5.4.2	Vliv použitého optimizéru	78
5.4.3	Vliv použité ztrátové funkce (Loss function)	78
5.5	Citlivostní analýza	79
5.5.1	Ovlivnění proporcionální konstantou	80
5.5.2	Ovlivnění přídatným šumem	80
5.5.3	Ovlivnění filtrací signálu	81
5.6	Časová náročnost	82
6	Grafická aplikace pro detekci anomálií	84
6.1	Základní popis aplikace	84
6.2	Karta Dataset	84
6.3	Karta Train	86
6.4	Karta Test	87
	Závěr	89

Literatura	91
Seznam symbolů a zkratek	99
Seznam příloh	101
A Zdrojové kódy	102
A.1 Skripty pro generování testovacích signálů pomocí GutenTAG	102
B Vlivu hyperparametrů - tabulky a grafy	104
B.1 Vliv velikosti dávky	104
B.2 Vliv použitého optimizéru	110
B.3 Vliv ztrátové funkce	114
C Citlivostní analýza - tabulky a grafy	118
C.1 Vliv násobení konstantou	118
C.2 Vliv přidání šumu	123
C.3 Vliv filtrace signálu	129
D Obsah elektronické přílohy	133

Seznam obrázků

1.1	Parní stroj (obrázek z publikace od J. Janovského 1904)	20
1.2	Montážní linka na automobily	21
1.3	Reléové panely z počátku 60.let	22
1.4	Digitální továrna	24
1.5	Fotografie z otevření testbedu pro Průmysl 4.0 na CEITEC Brno	25
2.1	Typy údržby v průmyslu	26
2.2	Základní tři typy anomálií. Bodová (vlevo), Kolektivní(střed), Kontextová(vlevo)	28
2.3	Rozložení neuronu při trénování RNN	30
2.4	Schéma LSTM buňky	30
2.5	Průběhy signálů ze dvou datasetů (Senzor) s odpovídající věrohodností (p)	32
2.6	Dataset pro demonstraci LOF	32
2.7	Příklad zpracování signálu pomocí Haarovy vlnkové funkce	33
2.8	Vizualizace architektury DeepAnt	35
2.9	Rozdíl izolace normálního datového bodu (vlevo) a anomálie (vpravo)	36
2.10	Průměrná vzdálenost normálního datového bodu (modrá) a anomálie (červená)	36
2.11	Ilustrativní příklad použití autoenkodérů	37
2.12	Základní architektura GAN	38
3.1	Ilustrace signálů datasetu Pump sensor data se zvýrazněnými stavy	42
3.2	Korelační matice signálů datasetu Pump sensor data. Na horizontální ose jsou stejné popisky jako na vertikální	43
3.3	Ilustrace pokusu pro získání NASA Bearing Datasetu	44
3.4	Schéma pokusu pro vytvoření datasetu Preventive to Predictive Maintenance	45
3.5	Schéma experimentu pro získání E-coating ultrafiltration maintenance datasetu	49
3.6	Příklad signálu generovaného pomocí GutenTAG	50
3.7	Příklad sinusového signálu s anomáliemi generovaného pomocí GutenTAG	51
3.8	Příklad čtvercových pulzů s anomáliemi generovaných pomocí GutenTAG	52
3.9	Příklad Dirichletových vln s anomáliemi generovaných pomocí GutenTAG	52
4.1	Sinusový signál generovaný s pomocí GutenTAG s detekovanými anomáliemi pomocí LSTM	54

4.2	Obdélníkový signál generovaný s pomocí GutenTAG s detekovanými anomáliemi pomocí LSTM	55
4.3	Dirichletovy vlny generované s pomocí GutenTAG s detekovanými anomáliemi pomocí LSTM	55
4.4	Signál jednoho senzoru z datasetu Pump sensor data s detekovanými anomáliemi pomocí sítě LSTM odpovídajícím anomálním stavům zařízení	56
4.5	Signál jednoho senzoru z datasetu Pump sensor data s detekovanými anomáliemi pomocí sítě LSTM neodpovídajícím anomálním stavům zařízení	57
4.6	Signál jednoho senzoru z datasetu Pump sensor data s velmi malým počtem detekovaných anomálií detekovaných pomocí sítě LSTM . . .	57
4.7	Sinusový signál generovaný s pomocí GutenTAG s detekovanými anomáliemi pomocí CNN	58
4.8	Obdélníkový signál generovaný s pomocí GutenTAG s detekovanými anomáliemi pomocí CNN	59
4.9	Dirichletovy vlny generované s pomocí GutenTAG s detekovanými anomáliemi pomocí CNN	59
4.10	Signál jednoho senzoru z datasetu Pump sensor data s detekovanými anomáliemi pomocí sítě s CNN odpovídajícím anomálním stavům zařízení	60
4.11	Signál jednoho senzoru z datasetu Pump sensor data s detekovanými anomáliemi pomocí sítě s CNN neodpovídajícím anomálním stavům zařízení	61
4.12	Signál jednoho senzoru z datasetu Pump sensor data s velmi malým počtem detekovaných anomálií detekovaných pomocí sítě s CNN . . .	61
4.13	Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Random forest	63
4.14	Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Linear regression	63
4.15	Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Isolation Forest	65
4.16	Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Local outlier factor s počtem nejbližších sousedů nastaveným na 100	66
4.17	Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Local outlier factor s počtem nejbližších sousedů nastaveným na 1000	67

4.18	Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Local outlier factor s počtem nejbližších sousedů nastaveným na 5000	67
5.1	Výpis architektury autoenkodéru pro detekci anomálií s použitím plně propojených vrstev neuronové sítě	70
5.2	Výpis architektury autoenkodéru pro detekci anomálií s použitím LSTM vrstev neuronové sítě	71
5.3	Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí autoenkodéru založených na plně propojených vrstvách	74
5.4	Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí autoenkodéru založených na LSTM vrstvách	74
5.5	Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem založeným na plně propojených vrstvách neuronů	75
5.6	Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem založeným na LSTM vrstvách	75
6.1	Snímek obrazovky aplikace pro detekci anomálií s kartou Dataset	86
6.2	Snímek obrazovky aplikace pro detekci anomálií s otevřenou kartou Train	87
6.3	Snímek obrazovky aplikace pro detekci anomálií s otevřenou kartou Test	88
B.1	Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 1 trénované s velikostí dávky 10	108
B.2	Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 2 trénované s velikostí dávky 500	108
B.3	Detekce anomálního stavu zařízení z datasetu Pump sensor data s autoenkodérem 1 trénované s velikostí dávky 10	109
B.4	Detekce anomálního stavu zařízení z datasetu Pump sensor data s autoenkodérem 2 trénované s velikostí dávky 500	109
B.5	Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 1 trénované s optimizérem AdaMax	112
B.6	Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 2 trénované s optimizérem AdaMax	112
B.7	Detekce anomálního stavu zařízení z datasetu Pump sensor data s autoenkodérem 1 trénované s optimizérem AdaMax	113
B.8	Detekce anomálního stavu zařízení z datasetu Pump sensor data s autoenkodérem 2 trénované s optimizérem AdaMax	113
B.9	Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 1 trénované se ztrátovou funkcí MSE	115

B.10	Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 2 trénované se ztrátovou funkcí MAE	116
B.11	Detekce anomálního stavu zařízení z datasetu Pump sensor data s autoenkodérem 1 trénované se ztrátovou funkcí MSE	116
B.12	Detekce anomálního stavu zařízení z datasetu Pump sensor data s autoenkodérem 2 trénované se ztrátovou funkcí MAE	117
C.1	Matice záměn detekce anomálií datasetu Pump sensor data zeslabená vynásobením konstantou 0.00001 s autoenkodérem 1	119
C.2	Matice záměn detekce anomálií datasetu Pump sensor data zesílená vynásobením konstantou 10000 s autoenkodérem 1	119
C.3	Matice záměn detekce anomálií datasetu Pump sensor data zeslabená vynásobením konstantou 0.00001 s autoenkodérem 2	120
C.4	Matice záměn detekce anomálií datasetu Pump sensor data zesílená vynásobením konstantou 10000 s autoenkodérem 2	120
C.5	Detekce anomálního stavu zařízení z datasetu Pump sensor data zeslabená vynásobením konstantou 0.00001 s autoenkodérem 1	121
C.6	Detekce anomálního stavu zařízení z datasetu Pump sensor data zesílená vynásobením konstantou 10000 s autoenkodérem 1	121
C.7	Detekce anomálního stavu zařízení z datasetu Pump sensor data zeslabená vynásobením konstantou 0.00001 s autoenkodérem 2	122
C.8	Detekce anomálního stavu zařízení z datasetu Pump sensor data zesílená vynásobením konstantou 10000 s autoenkodérem 2	122
C.9	Matice záměn detekce anomálií datasetu Pump sensor data s přidaným šumem o SNR 20 dB pro autoenkodér 1	124
C.10	Matice záměn detekce anomálií datasetu Pump sensor data s přidaným šumem o SNR 20 dB pro autoenkodér 2	125
C.11	Matice záměn detekce anomálií datasetu Pump sensor data s přidaným šumem o SNR 5 dB pro autoenkodér 1	125
C.12	Matice záměn detekce anomálií datasetu Pump sensor data s přidaným šumem o SNR 5 dB pro autoenkodér 2	126
C.13	Detekce anomálního stavu zařízení z datasetu Pump sensor data s přidaným šumem o SNR 20 dB pro autoenkodér 1	126
C.14	Detekce anomálního stavu zařízení z datasetu Pump sensor data s přidaným šumem o SNR 20 dB pro autoenkodér 2	127
C.15	Detekce anomálního stavu zařízení z datasetu Pump sensor data s přidaným šumem o SNR 5 dB pro autoenkodér 1	127
C.16	Detekce anomálního stavu zařízení z datasetu Pump sensor data s přidaným šumem o SNR 5 dB pro autoenkodér 2	128

C.17	Matice záměn detekce anomálií datasetu Pump sensor data filtrovaná plovoucím průměrem o velikosti okna 100 pro autoenkodér 1	131
C.18	Matice záměn detekce anomálií datasetu Pump sensor data filtrovaná plovoucím průměrem o velikosti okna 100 pro autoenkodér 2	131
C.19	Detekce anomálního stavu zařízení z datasetu Pump sensor data filtrovaná plovoucím průměrem o velikosti okna 100 pro autoenkodér 1	132
C.20	Detekce anomálního stavu zařízení z datasetu Pump sensor data filtrovaná plovoucím průměrem o velikosti okna 100 pro autoenkodér 2	132

Seznam tabulek

5.1	Tabulka metrik detektoru anomálií datasetu Pump sensor data s autoenkodérem založeným na plně propojených vrstvách neuronů	76
5.2	Tabulka metrik detektoru anomálií datasetu Pump sensor data s autoenkodérem založeným na LSTM vrstvách	76
5.3	Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data pro různé velikosti dávky	77
5.4	Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data pro různé optimizéry	78
5.5	Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data pro různé ztrátové funkce	79
5.6	Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data vynásobená různými konstantami	80
5.7	Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data s přidáním šumem	81
5.8	Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data filtrovaná plovoucím průměrem s různou délkou okna	82
5.9	Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data filtrovaná plovoucím průměrem s různou délkou okna	83
B.1	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 1	104
B.2	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 10	104
B.3	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 50	105
B.4	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 100	105
B.5	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 500	105
B.6	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 1000	106
B.7	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 5000	106
B.8	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 10000	106

B.9	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 20000	107
B.10	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 50000	107
B.11	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaných pomocí optimizéru AdaMax	110
B.12	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaných pomocí optimizéru Adam	110
B.13	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaných pomocí optimizéru Adagrad	110
B.14	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaných pomocí optimizéru Adadelta	111
B.15	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaných pomocí optimizéru SGD	111
B.16	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným se ztrátovou funkcí MSE	114
B.17	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným se ztrátovou funkcí MAE	114
B.18	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným se ztrátovou funkcí Hubert loss	114
B.19	Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným se ztrátovou funkcí KL-divergence	115
C.1	Tabulka metrik detektorů anomálií datasetu Pump sensor data zeslabená vynásobením konstantou 0.00001	118
C.2	Tabulka metrik detektorů anomálií datasetu Pump sensor data zesílená vynásobením konstantou 10000	118
C.3	Tabulka metrik detektorů anomálií datasetu Pump sensor data s přidáním šumem o SNR 50 dB	123
C.4	Tabulka metrik detektorů anomálií datasetu Pump sensor data s přidáním šumem o SNR 20 dB	123
C.5	Tabulka metrik detektorů anomálií datasetu Pump sensor data s přidáním šumem o SNR 10 dB	123
C.6	Tabulka metrik detektorů anomálií datasetu Pump sensor data s přidáním šumem o SNR 5 dB	124
C.7	Tabulka metrik detektorů anomálií datasetu Pump sensor data filtrovaným filtrem plovoucího průměru s délkou okna 100 bodů	129
C.8	Tabulka metrik detektorů anomálií datasetu Pump sensor data filtrovaným filtrem plovoucího průměru s délkou okna 50 bodů	129

C.9	Tabulka metrik detektorů anomálií datasetu Pump sensor data filtrovaným filtrem plovoucího průměru s délkou okna 20 bodů	129
C.10	Tabulka metrik detektorů anomálií datasetu Pump sensor data filtrovaným filtrem plovoucího průměru s délkou okna 10 bodů	130
C.11	Tabulka metrik detektorů anomálií datasetu Pump sensor data filtrovaným filtrem plovoucího průměru s délkou okna 5 bodů	130

Seznam výpisů

code/sine_config_part.py	102
code/rec_config_part.py	102
code/dir_config_part.py	103

Úvod

V posledních několika dekádách dochází v průmyslu k obrovskému rozvoji v oblasti automatizace, digitalizace a tzv. chytrých technologií. Tento trend byl označen jako Čtvrtá průmyslová revoluce neboli tzv. Průmysl 4.0.

Průmysl 4.0 se snaží propojit zařízení, zaměstnance i výrobky do jedné informační sítě a tím zvyšovat efektivitu a flexibilitu výroby. Proto byly mimo jiné představeny Kyber-fyzikální systémy, Internet věcí, Internet služeb a Digitální dvojčata.

V průmyslu je zapojeno stále více a více zařízení, která produkují data. Cílem je tato data shromáždit, analyzovat a na jejich základě vytvářet tzv. chytrá řešení za účelem maximalizace efektivitu a zisku jednotlivých firem.

Jednou z možností, jak zvýšit efektivitu je zavést tzv. prediktivní údržbu, což je koncept, který sleduje kondici jednotlivých zařízení a odhaduje dobu, po kterou bude zařízení ještě schopné normálního provozu.

Cílem této práce je zmapovat metody strojového učení používané v konceptu Průmysl 4.0. Mezi hlavní oblasti zájmu patří vizuální inspekce (detekce anomálií, defektů v obrazových datech), prediktivní údržba (detekce anomálií v časových řadách) a příp. HMI interakce

Práce se skládá z teoretické a praktické části.

Teoretická část se věnuje literární rešerši konceptu Průmysl 4.0 a metod strojového učení, které se v ní dají použít. Metody strojového učení jsou rozdělené podle oblasti použití na část, která se věnuje vizuální inspekci a na prediktivní údržbu. V každé části je vždy rozebráno několik metod včetně způsobu použití.

Dále jsou v teoretické části zmíněné datasety a generátory datasetů, které jsou na internetu volně dostupné pro testování algoritmů na detekci anomálií.

V praktické části jsou některé z dříve uvedených metod testovány a zhodnocen jejich výkon. Do detailu je zde rozebrána detekce anomálií pomocí autoenkodérů, které jsou vytvořeny ve dvou implemenetacích. Pomocí plně propojených vrstev neuronů a pomocí vrstev s LSTM buňkami. Výkon těchto autoekodérů je dále zhodnocen pomocí matice záměn a metrik s ní spojených. Dále je zhodnocen vliv hyperparametrů a změn v signálu na výkon sítě včetně časové náročnosti na trénování a testování.

Tyto testy lze mimo jiné provádět pomocí grafické aplikace, která byla vytvořena v rámci této diplomové práce.

1 Průmysl 4.0

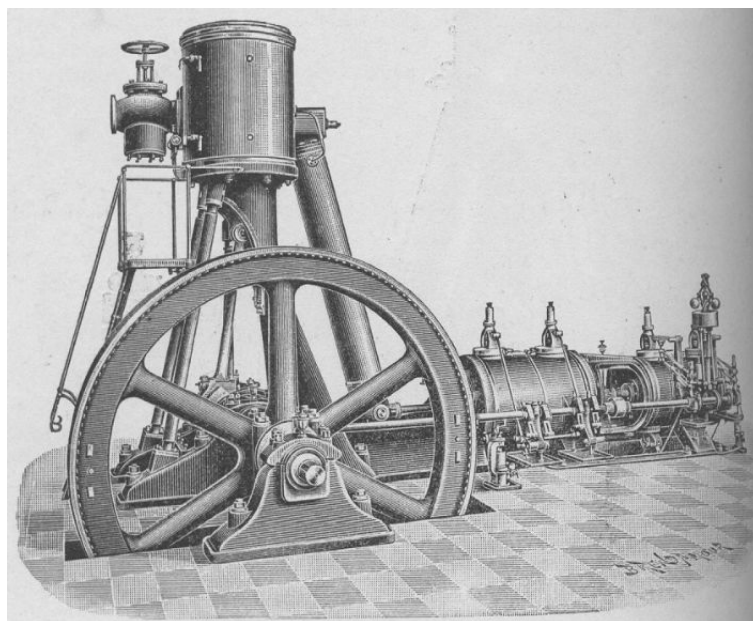
V této kapitole bude velmi stručně vysvětlen koncept Průmysl 4.0. Nejdříve budou shrnuty předchozí průmyslové revoluce a jejich význam při utváření současné podoby průmyslu. Další sekce bude zaměřena přímo na pojem Průmysl 4.0, na základní koncepty a předpokládaný vývoj tohoto fenoménu v budoucnosti.

1.1 Historie

V této sekci bude stručně představena historie průmyslových revolucí, hlavní objevy a jejich dopad na společnost.

1.1.1 První průmyslová revoluce

První průmyslová revoluce odstartovala ve Velké Británii ve druhé polovině 18. století. Její vznik zapříčinila industrializace a mechanizace průmyslu. Mezi největší vynálezy, které umožnily vznik této revoluce, patří parní stroj, který dokázal ve velké míře nahradit práci lidí a práci zvířat. Parní stroj dále umožnil rozvoj cestování a transportu průmyslových výrobků pomocí železnice. Parní stroj je znázorněn na obrázku 1.1



Obr. 1.1: Parní stroj (publikace J. Janovského 1904). Dostupné na[60]

Druhým důležitým vynálezem byl mechanický tkalcovský stav, který z Velké Británie dokázal udělat světovou velmoc ve vývozu bavlněných výrobků. V té době také dochází k rozvoji chemického průmyslu a k rozvoji zpracování železa a oceli.

První průmyslová revoluce měla celospolečenský vliv. Především dala vzniknout dělnické profesi. Velké množství lidí opustilo drobné zemědělství a začalo se stěhovat za prací do měst. Dochází také k postupnému zlepšení životního standardu, což vedlo k obrovskému nárůstu populace. [10, 13]

1.1.2 Druhá průmyslová revoluce

Druhá průmyslová revoluce začala ve druhé polovině 19. století a nesla se ve znamení elektrifikace. Elektrická energie se sice používala už v době první průmyslové revoluce, ale teprve ve druhé polovině 19. století dochází k jejímu masivnímu rozšíření. S popularizací elektrické energie také souvisí velké množství vynálezů jako žárovka, telefon či telegraf.

Jako další pilíř druhé průmyslové revoluce bývá uváděna pásová výroba. Jedna z prvních linek byla instalována v automobilovém závodě firmy Ford, čímž se výrazně zvýšila její efektivita. Pásovou výrobu automobilů je možné vidět na obrázku 1.2.



Obr. 1.2: Montážní linka na automobily. Dostupné na[61]

Během druhé průmyslové revoluce dochází po vynálezu spalovacího motoru k rozvoji automobilismu. Díky vynálezu letadel a dalšímu rozvoji železniční sítě dokáží

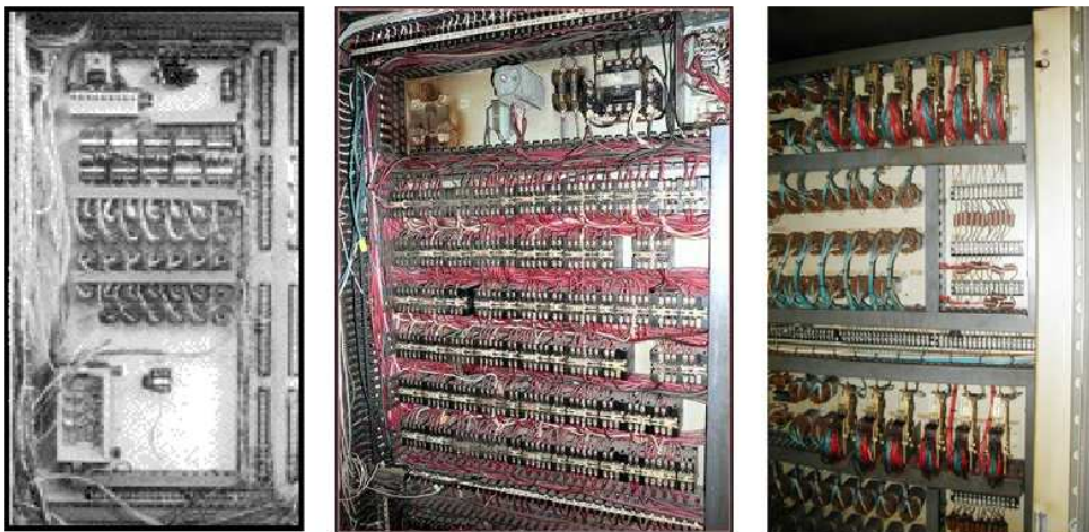
lidé cestovat na velmi dlouhé vzdálenosti v relativně krátkém čase. Nastává stále větší stěhování lidí do měst a k obecnému růstu životní úrovně. [11, 12, 13]

1.1.3 Třetí průmyslová revoluce

Třetí průmyslová revoluce se začíná rozvíjet po Druhé světové válce. Prvním významným milníkem byl v roce 1947 vynález tranzistoru, který později umožnil vznik digitálních integrovaných obvodů. Další roky se tedy nesly ve znamení vývoje na poli polovodičových technologií. Centrem pro výzkum v této oblasti se stala Americká Silicon Valley.

Během třetí průmyslové revoluce dochází také k rozvoji komunikačních technologií. V roce 1969 vzniká první předchůdce internetu tzv. ARPANET, který se postupně z Ameriky rozšířil i do části Evropy.

Spolu s rozvojem polovodičových a komunikačních technologií dochází také k postupnému rozvoji automatizace. Na konci 60. let minulého století bylo představeno první PLC. Tento milník je v některých literaturách označován jako začátek třetí průmyslové revoluce. Na obrázku 1.3 lze vidět reléové panely, předchůdce dnešních PLC.



Obr. 1.3: Reléové panely z počátku 60.let. Dostupné na[62]

Všechny nově vznikající technologie vedly k růstu efektivity, zkracování vzdáleností a postupně umožnily propojit informačně, ekonomicky i tržně celý svět. Tomuto trendu říkáme globalizace. [13, 14, 15]

1.1.4 Čtvrtá průmyslová revoluce

Termín Čtvrtá průmyslová revoluce neboli Průmysl 4.0 (*angl. Industry 4*) zazněl poprvé v roce 2011 na veletrhu Hannover Messe v Hamburku. Od té doby se z něho stal celosvětový pojem, který si klade za cíl využití moderních informačních technologií pro maximalizaci efektivity vývoje a výroby. [13, 14]

Blíže bude pojem Průmysl 4.0 představen v další části.

1.2 Představení konceptu Průmysl 4.0

Průmysl 4.0 je stále poměrně nový a rozvíjející se koncept. Šíře problematiky, kterou zahrnuje, je obrovská. Proto v této sekci budou představena pouze základní fakta, která jsou relevantní pro další práci.

Jedním z cílů Průmyslu 4.0 je vytvoření tzv. Chytré továrny. Jedná se o továrnu, která je založena na chytrých jednotkách, chytrých výrobcích a dalších chytrých řešeních, které spolu dokáží komunikovat, sdílet mezi sebou data např. pomocí Cloudového úložiště a domlouvat mezi sebou optimální strategii výroby.

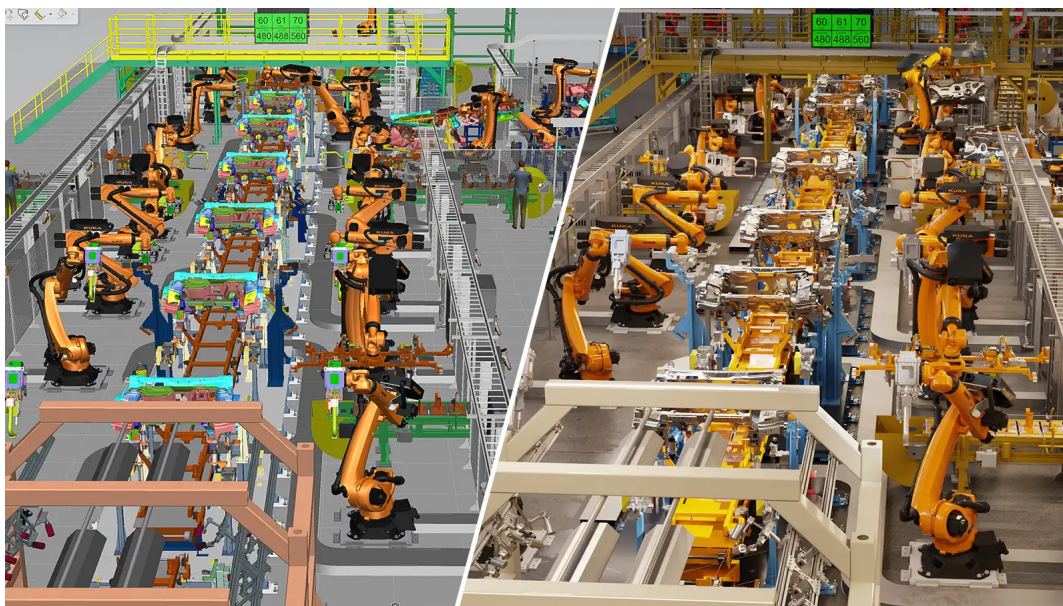
Aby chytrá zařízení (ať už se jedná o senzor, výrobní buňku, či systém řízení výroby) a případně i člověk mezi sebou dokázali komunikovat, musí být připojeni ke stejné síti. Tato síť se označuje jako Internet věcí (*angl. Internet fo things, IoT*). Přes tuto síť probíhá v reálném čase veškerá komunikace zařízení v dané firmě.

V průmyslu 4.0 jsou všechna fyzická zařízení natolik těsně propojena s digitálním světem, že je označujeme jako tzv. Kyber-fyzikální systémy *Cyber-physical systems (CPS)*. Každé fyzické zařízení (ať už se jedná o výrobní stroj či výrobek) má ještě svoje Digitální dvojče, což je digitální objekt, který zastupuje dané zařízení v digitálním světě. Pro ilustraci je přiložen obrázek 1.4, kde na pravé straně je skutečná továrna a na levé její Digitální dvojče.

Z předchozích odstavců je patrné, že v rámci Průmyslu 4.0 vzniká relativně velké množství dat, které je potřeba uložit a posléze i zpracovat. K tomu se dají použít tzv. Cloudy. Díky nim můžeme ukládat velké množství dat ze všech chytrých zařízení a ty následně analyzovat. Tzv. Big data analýza patří mezi klíčové koncepty a může přispět k dalšímu růstu efektivity.

Mezi další koncepty je simulace zařízení. Pokud plánujeme výrobu nového produktu nebo nějakou změnu na lince továrny, je vhodné danou změnu prvně simulovat ve virtuálním prostředí a až poté ji implementovat v reálném zařízení. Použití simulace je možné díky tomu, že každé zařízení má svoje Digitální dvojče. Tímto způsobem lze simulovat i celé závody.

Mezi velmi oblíbené technologie, které dále přispívají k rozvoji průmyslu 4.0, je tzv. aditivní výroba. Mezi její hlavní zástupce patří 3D tisk, což je technologie,



Obr. 1.4: Digitální továrna. Dostupné na[63]

kteřá umožňuje mimo jiné rychlou a relativně levnou výrobu prototypů strojních zařízení. Průmysl 4.0 se skládá z velkého množství dílčích konceptů. V této části byly vyjmenovány ty nejpodstatnější. Jedny z podstatných konceptů, které zmíněny nebyly, jsou prediktivní údržba a vizuální inspekce, kterým bude věnována další kapitola. [13, 16, 17, 18, 19]

1.3 Vědecké instituce

Ještě než se podíváme na oblasti strojového učení, které lze použít v Průmyslu 4.0, chtěli bychom se zmínit o institucích, které se věnují výzkumu v této oblasti.

Mezi nejvýznamnější české výzkumné ústavy, které se věnují fenoménu Průmysl 4.0, patří CIIRC ČVUT (Český institut informatiky, robotiky a kybernetiky). Tento institut byl založen v roce 2013 a v dnešní době má 8 výzkumných oddělení, které se věnují mimo jiné strojovému učení, počítačovému vidění, počítačové grafice, automatickému řízení a optimalizaci, robotice a dalším oblastem, které více méně všechny nějak souvisí s Průmyslem 4.0. Na ústavu se také nachází testbed pro Průmysl 4.0.

Vědeckým ředitelem ústavu je prof. Ing. Vladimír Mařík, DrSc., dr. h. c. Součástí CIIRC ČVUT je také Národní centrum pro průmysl 4.0, které funguje samostatně v rámci ČVUT.

CIIRC ČVUT (respektive jeho testbed) je společně s CEITEC (Středoevropský technologický institut resp. jeho testbed v Brně) součástí RICAIP (*angl. Research and Innovation Centre on Advanced Industrial Production*), což je výzkumné

a inovační centrum pokročilé průmyslové produkce. RICAIP dále tvoří 2 německé instituce. DFKI (*něm. Deutsches Forschungszentrum für Künstliche Intelligenz*) je německé výzkumné centrum pro umělou inteligenci a ZeMA (*něm. Deutsches Forschungszentrum für Künstliche Intelligenz*), což je německé centrum pro mechatniku a automatizaci.

Z německých institucí dále stojí za zmínku Platform Industrie 4.0, která stála u zrodu pojmu Průmysl 4.0. Jejím zakladatelem je prof. Wolfgang Wahlster, který je jeden z autorů celého konceptu Průmysl 4.0.

Pro ilustraci je připojena fotografie z otevření testbedu pro Průmysl 4.0 na pracovišti CEITEC v Brně.



Obr. 1.5: Fotografie z otevření testbedu pro Průmysl 4.0 na CEITEC Brno

2 Strojové učení v průmyslu 4.0

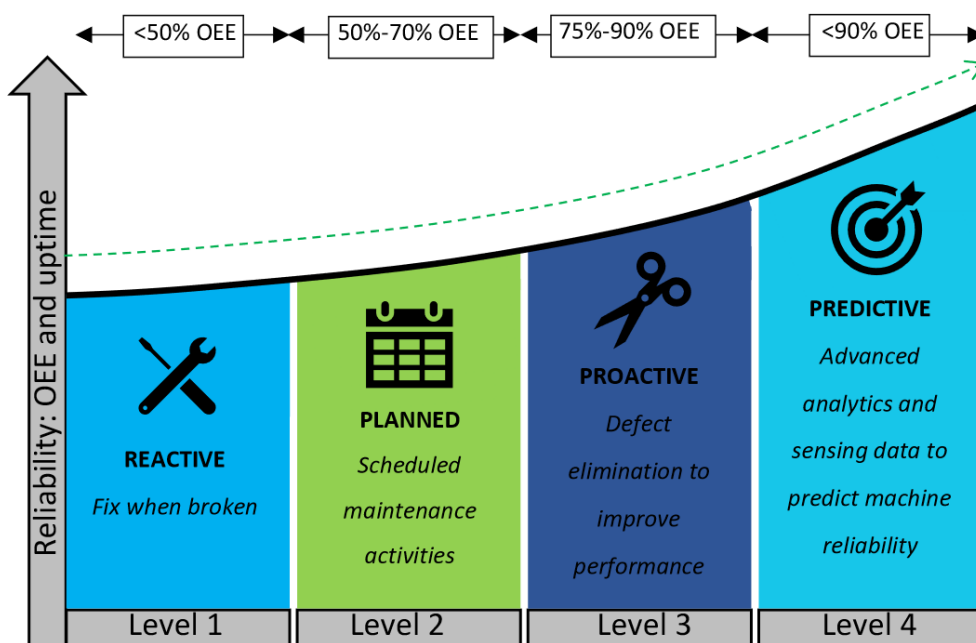
Tato kapitola je věnována algoritmům strojového učení, které lze použít v rámci Průmyslu 4.0. Zvláštní důraz je kladen na oblast prediktivní údržby a vizuální inspekce.

2.1 Oblast prediktivní údržby

Tato část se věnuje pojmu prediktivní údržba. Postupně zde budou představeny ostatní, méně efektivní systémy údržby, které jí předcházely. Poté bude rozebrána prediktivní údržba jako taková. V závěru budou představeny některé algoritmy, které jsou vhodné pro aplikaci na prediktivní údržbu.

2.1.1 Předchůdci prediktivní údržby

Tato část je zaměřena na metody údržby průmyslových zařízení, které předcházely Prediktivní údržbě podle [1, 2, 3]. Typy údržby shrnuje obrázek 2.1.



Obr. 2.1: Typy údržby v průmyslu. Dostupné na[1]

Úplně nejjednodušším konceptem v oblasti údržby je tzv. Reaktivní údržba. (angl. *Reactive maintenance, Run2Failure*). Tato metoda nechává zařízení pracovat, dokud nedojde k jeho selhání. Poté se provede údržba. Nevýhodou této metody

je zvýšený počet nepředpokládaných odstávek technologie, pokud musí být zastavena při selhání libovolného zařízení. Další nevýhodou jsou návazné chyby, které jsou způsobené především provozem defektního zařízení.

Koncept reaktivní údržby je v dnešní době poměrně nevhodný. Jednou z možností jeho náhrady je tzv. Preventivní údržba (*angl. Preventative maintenance*). Tento systém údržby zajišťuje pravidelnou výměnu komponentů technologie dřív, než by mohlo dojít k jejich selhání. Perioda výměny se dá určit například ze střední doby bezporuchového provozu. Výhodou tohoto systému je zejména snížení počtu neplánovaných odstávek technologie. Mezi nevýhody patří fakt, že zařízení vyměňujeme dřív, než bychom museli. Tím rostou náklady na údržbu a na celkový provoz technologie.

Abychom zbytečně nevyměňovali zařízení dřív než skutečně musíme, byl vyvinut systém Údržby založené na podmínce (*angl. Condition based maintenance*). Tento systém údržby je v některých člancích uváděn samostatně, i když některé ho řadí do Prediktivní údržby. Systém spočívá ve sledování provozních parametrů strojního zařízení a na základě pevně stanovených podmínek hlásí, kdy je potřeba danou technologii odstavit ještě předtím než dojde ke skutečné závadě. Výhodou tohoto přístupu je prodloužení užitečného života zařízení (*angl. Remaining usefull life*).

2.1.2 Prediktivní údržba

Nejvyspělejší metoda údržby zařízení je už několikrát zmíněná Prediktivní údržba (*angl. Predictive maintenance*). Její princip spočívá v kontinuálním sledování provozních parametrů zařízení (v závislosti na typu zařízení se může jednat o teplotu, tlak nebo např. vibrace), ale také parametrů okolí (jako je např. vnější teplota, vlhkost).

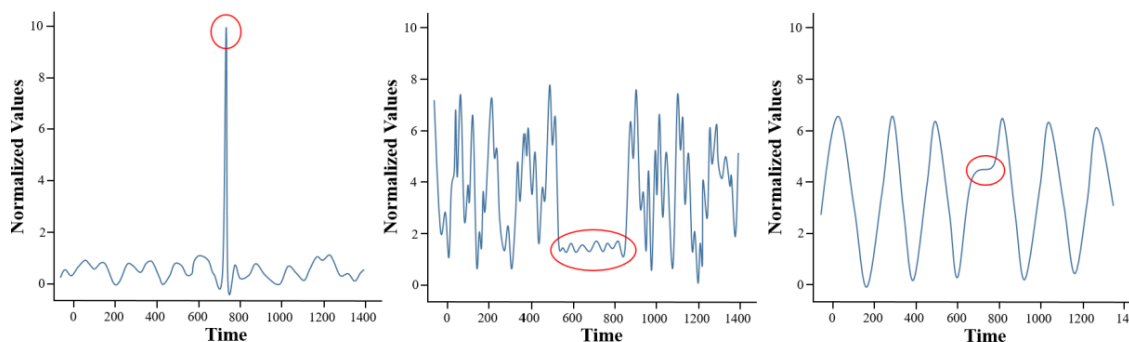
Cílem Prediktivní údržby je nejen detekce poruchy, ale i jiných nestandardních stavů, které mohou být potenciálním nebezpečím pro zařízení i když ještě nemusí znamenat přímo poruchu. Prediktivní údržba může tedy odhalit stav zařízení, který by mohl vést chybě, resp. selhání, ale díky včasnému zásahu je možné provést opravu, takže nemusí ani dojít k výměně zařízení. Tím se maximalizuje životnost jednotlivých komponentů, což vede ke snižování nákladů na údržbu.[1, 2, 3]

2.1.3 Detekce anomálií v prediktivní údržbě

Potenciálně rizikové stavy jsou odhalovány v provozních signálech jako tzv. anomálie. Anomálie můžeme definovat mnoha způsoby. Např. v [8] definují anomálii jako *data, která se velmi výrazně liší od ostatních dat ze stejného vzorku, ve kterém se všechna vyskytují*. Nebo v [7] definují anomálii jako *data pozorování, která se liší*

od ostatních dat tak významně, že vyvolávají podezření, že byly vygenerované jiným mechanismem.

Existuje také několik druhů anomálií. Článek [9] rozlišuje tři základní typy anomálií: bodové, kontextové a kolektivní. Za bodovou anomálii označíme takovou hodnotu nebo sekvenci hodnot signálu, která překračuje jeho normální hodnoty. Často bývají způsobeny šumem senzorů, chybou senzorů nebo nestandardním stavem systému. Za kontextovou anomálii označíme takovou hodnotu nebo sekvenci hodnot signálu, které sice nemusí překračovat normální hodnoty signálu, ale v kratším kontextu můžeme například pozorovat změnu standardního tvaru křivky. Za kolektivní anomálii označíme takovou sekvenci hodnot, z nichž každá sama o sobě by byla normální, ale jinak tyto hodnoty tvoří sekvenci, která anomální je. Příklady všech anomálií jsou uvedeny na obrázku 2.2



Obr. 2.2: Základní tři typy anomálií. Bodová (vlevo), Kolektivní(střed), Kontextová(vpravo). Dostupné na[59]

Každý z algoritmů pro detekci anomálií v signálech je vhodný pro jiný druh anomálií. V současné době neexistuje univerzální algoritmus vhodný pro detekci všech druhů anomálií na všech datasetech. [4]. Proto výběr algoritmu závisí především na dané aplikaci. Hlavní problém při výběru algoritmu je neznalost povahy anomálií, které chceme detekovat.

Algoritmy pro detekci anomálií můžeme také rozlišit podle jejich vnitřní podstaty, kterou detekují anomálie. Podle článku [4] je můžeme rozdělit na 6 typů: Prediktivní algoritmy, Rekonstrukční metody, Kódovací metody, Metody založené na vzdálenosti, Metody založené na rozložení a Izolační stromy.

Prediktivní metody předpovídají další datový bod v sekvenci a ten porovnávají s realitou. Rekonstrukční metody využívají systém s úzkým hrdlem, do kterého zakódují a znovu z něj dekódují data. Anomálie se v tomto případě hodnotí na základě podobnosti originálních a dekódovaných dat. Kódovací metody počítají přímo

skóre anomaly ze zakódované reprezentace původního signálu. Na rozdíl od Rekonstrukčních metod nemají dekodér. Metody založené na vzdálenosti převádějí původní signál do jiného stavového prostoru, kde má smysl počítat vzdálenost jednotlivých datových bodů a na základě toho hledat anomálie. Metody založené na statistickém rozložení odhadují, s jakou věrohodností patří daná data do rozložení normálního signálu. Izolační stromy jsou založené na principu, podle kterého jsou anomálie málo časté a snadno odlišitelné od ostatních dat. Při tvorbě izolačních stromů se statisticky dříve oddělí než normální data a mají tedy kratší průměrnou vzdálenost od kmene stromu.

V následující části této práce bylo vybráno několik různorodých metod strojového učení, které můžeme použít pro detekci anomálií v časových řadách a tudíž i v prediktivní údržbě.

2.1.4 Metoda Long Short Term Memory

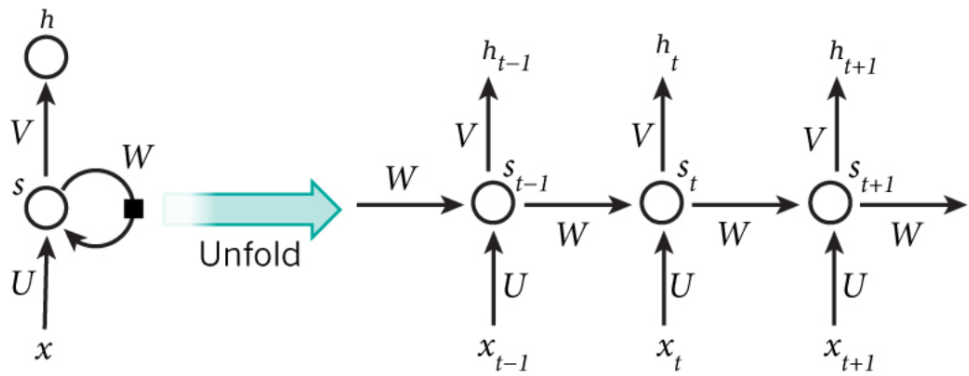
Metoda LSTM vynikla jako rozšíření Rekurentních neuronových sítí (RNN). Ty byly navrženy přímo pro práci se sekvenčními daty. V tom tkví jejich výhoda oproti ostatním algoritmům strojového učení. Většina algoritmů totiž zpracovává celou datovou sekvenci najednou. To přináší problémy hlavně z důvodu nutnosti volby velikosti okna, který vystupuje jako další důležitý hyperparametr. Pokud se zvolí moc krátké, algoritmus nemusí vnímat některé závislosti v signálu. Pokud se zvolí příliš dlouhé, je algoritmus ovlivněn šumem. [21]

LSTM a RNN zpracovávají datové řady bod po bodu, čímž se vytrácí nutnost časového omezení signálu. Díky tomu můžeme zpracovávat libovolně dlouhé signály a není potřeba mít apriorní informaci o závislostech v signálu. LSTM a RNN tedy umožňuje ze své podstaty online zpracování dat.[21]

Princip RNN je velmi podobný jako u sítě s algoritmem Error back propagation. Líší se pouze tím, že si každý neuron udržuje a s každým novým datovým prvkem aktualizuje svůj vektor skrytých vnitřních stavů (*angl. hidden state vector*), díky kterému je schopen reagovat i na předchozí vstupy.

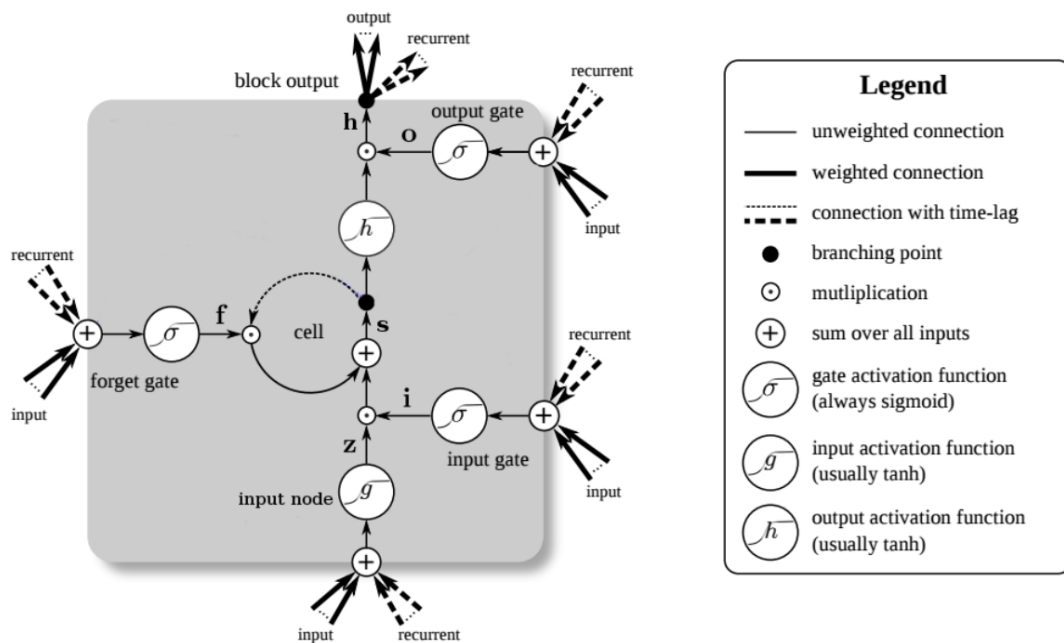
Učení probíhá jako u algoritmu Error back propagation, kdy jeden neuron rozložíme (*angl. unfold*) a této sekvenci neuronů postupně předkládáme trénovací data. (viz. obrázek 2.3) Zde se ale objevuje problém, který postihuje všechny rekurentní neuronové sítě. Pokud násobíme chybový gradient stále stejnou vahou (na čemž je tento algoritmus učení založen), tak dochází k jeho vynulování (*tzv. vanishing gradient*) nebo jeho růstu nade všechny meze (*angl. exploding gradient*). Z toho důvodu RNN často selhávají.[20]

Růstu gradientu nade všechny meze se dá zabránit tzv. gradient clippingem, což je podobný algoritmus jako L1 a L2 normalizace.



Obr. 2.3: Rozložení neuronu při trénování RNN Dostupné na[20]

Nulování gradientu lze předejít, pokud místo klasického neuronu, jaký používají RNN, použijeme LSTM buňku, kterou můžeme vidět na obrázku 2.4.



Obr. 2.4: Schéma LSTM buňky Dostupné na[20]

LSTM buňka se skládá z CEC (*angl. Constant error carousel*), vstupní brány, výstupní brány a brány zapomnění (*angl. forget gate*).

CEC je hlavní jednotka LSTM buňky. Má rekurentní spojení s předchozími vstupy a funguje tudíž jako paměť předchozích stavů.

Vstupní brána je multiplikatívni jednotka, která chrání CEC před rušením irrelevantními vstupy. Během trénování se vstupní brána učí, kdy nechat vstupní informaci projít do CEC. Dokud má vstupní brána hodnotu nula, tak se žádná informace nedostane dovnitř.

Výstupní brána je multiplikatívni jednotka, která chrání ostatní LSTM jednotky před rušením od CEC. Stejně jako vstupní i výstupní brána se učí, kdy informaci pustit ven. Pokud jsou obě brány zavřené, tak může chybový signál zůstat v buňce, aniž by narazil na problém vanishing gradientu.

Brány zapomnění jsou volitelnou součástí LSTM. Jejich funkce umožňuje resetovat CRC a tím udržovat vnitřní stav LSTM, aby jeho hodnota nerostla nade všechny meze.[20]

LSTM sítě se v detekci anomálií používají na odhad dalšího datového bodu v sekvenci. Předpovídaný bod se porovná s jeho skutečnou hodnotou a na základě rozdílu můžeme vyhodnotit skóre anomaly (*angl. anomaly score*).

Skóre anomaly lze vyhodnotit pomocí velkého množství metod. Například absolutní hodnotu chyby nebo mapování chybového vektoru na normální rozložení pomocí Maximum Likelihood Estimation (MLE).[21] Důležité je určit práh, který rozhoduje, zda daná chyba, resp. určitá věrohodnost patří ještě k normálu nebo už je anomální. V [21] tento problém vyřešili použitím dvou validačních datasetů, z nichž jeden byl určen k nastavení daného prahu.

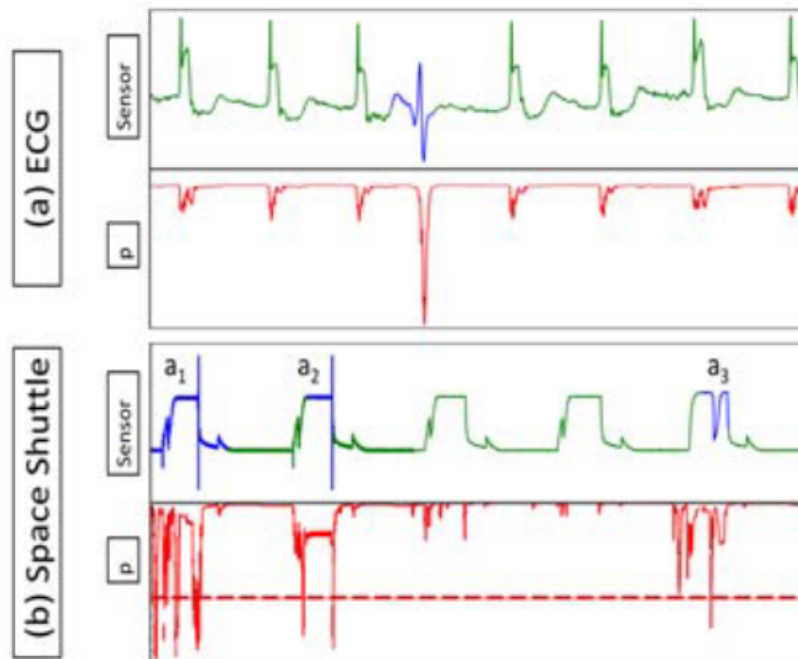
Pro ilustraci je přiložen obrázek 2.5, na kterém jsou vidět časové průběhy ze dvou datasetů: EEG a Space shuttle. Pod každým průběhem je uvedena odpovídající věrohodnost, která se používá jako ukazatel anomaly při použití MLE.

2.1.5 Metoda Sub-LOF

Článek [22] představuje koncept, který řeší nejen, jestli je vzorek anomální nebo ne, ale také, jak moc je anomální. Z toho důvodu zavádí pojem LOF (local outlier factor). LOF je metrika, která se přiřazuje každému datovému bodu a na základě její hodnoty můžeme rozhodnout, jak moc je daný vzorek anomální. Její hodnota je určena na základě polohy datového bodu vůči ostatním bodům a clusterům ve stavovém prostoru.

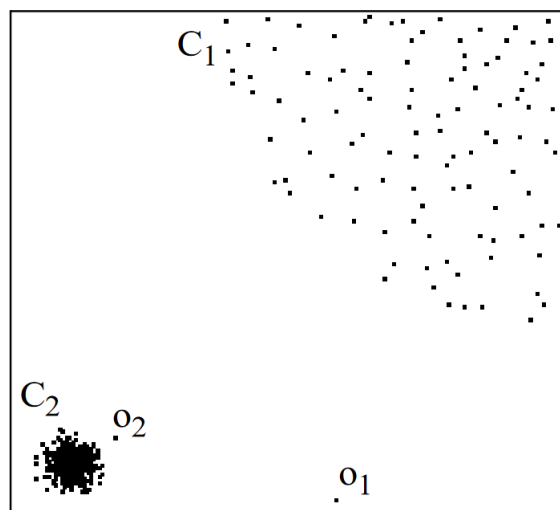
Metoda LOF hodnotí anomalitu jednotlivých bodů z lokálního pohledu, díky čemuž se dokáže adaptovat na různou velikost a hustotu clusterů. Jako jeden z hlavních hyperparametrů slouží MinPts, což je počet bodů v okolí, které definují lokální okolí objektu.

Výhody použití LOF lze snadno demonstrovat na obrázku 2.6. Na tomto obrázku vidíme ve stavovém prostoru data tvořící clustery C1 a C2. Na tomto obrázku jsou



Obr. 2.5: Průběhy signálů ze dvou datasetů (Sensor) s odpovídající věrohodností (p)
Dostupné na[21]

také datové body o_1 a o_2 , u kterých máme rozhodnout, jestli se jedná o anomálie, či nikoliv. U o_1 by měla patřit do C_1 a o_2 do C_2 .



Obr. 2.6: Dataset pro demonstraci LOF Dostupné na[22]

Pokud bychom brali v úvahu pouze globální pohled, tak o_1 bychom mohli zařadit

mezi anomálie, protože je podstatně dál ode všech datových bodů C1. O zařazení bodu o2 lze z globálního pohledu rozhodnout také jednoznačně. Jeho vzdálenost od nejbližšího bodu clusteru je menší než vzdálenost jednotlivých bodů C1. Proto patří do C2 a není anomální.

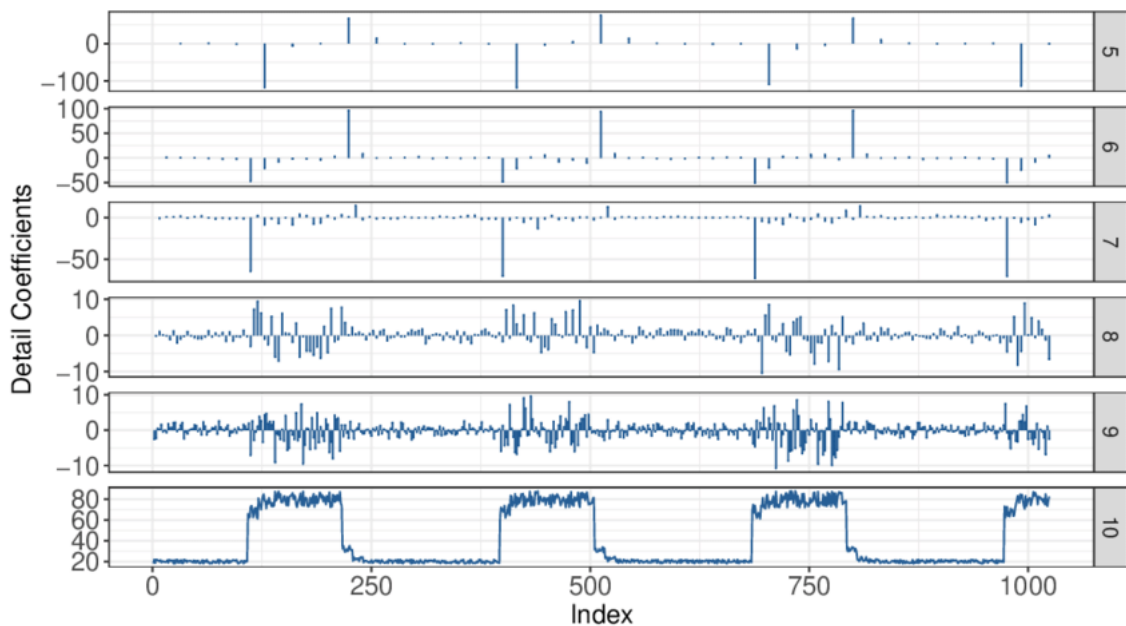
Při opětovném pohledu na obrázek se není možné ubránit pocitu, že vzhledem k hustotě ostatních bodů v C2 by bylo možné o2 vyhodnotit jako anomálii. Tento posudek lze ale získat pouze na základě lokálního pohledu. Z toho důvodu je vhodné použít LOF místo globálního hodnocení anomálií.

V článku [22] není bohužel zmíněna možnost aplikace na sekvenční data, která byla zmíněna v [4].

2.1.6 Metoda DWT-MLEAD

Celý název DWT-MLEAD je Discrete Wavelet Transform and Maximum Likelihood Estimation for Anomaly Detection in time series.

Vstupní signál prochází diskretní vlnkovou transformací. V článku [23] je použita Haarova funkce. Vlnková transformace umožňuje detekci krátkodobých i dlouhodobých anomálních úkazů v signálu. Příklad decimace signálu pomocí Haarovi funkce můžete vidět na obrázku 2.7. Originální signál je na uveden na spodu obrázku.



Obr. 2.7: Příklad zpracování signálu pomocí Haarovy vlnkové funkce Dostupné na[23]

Algoritmus DWT-MLEAD při učení zjišťuje parametry μ a kovarianční matici

vícerozměrného Gaussova rozdělení pro DWT vstupního signálu. Při zjišťování anomálnosti daného vzorku se pomocí MLE zjišťuje věrohodnost, se kterou daný vzorek patří do daného rozdělení.

Výsledek MLE každého signálu zpracovaný vlnkou se zapisuje do stromové struktury. Ze vzniklé struktury se smažou listy, v jejichž struktuře je méně než 2 anomálie. Jednotlivé listy stromu tvoří clustery. Pokud suma anomálií v daném clusteru překročí předem stanovený práh, vyhodnotí se cluster jako anomální. Tedy jsme detekovali anomálii.

Zahazování listů, které mají menší počet anomálních událostí než 2, se zdá podle článku efektivní obrana proti šumu v datech.

Jedná se o metodu učení bez učitele. Funguje pouze na offline datech, protože zpracovává datové sekvence jako celek ohraničený oknem. Tuto nevýhodu lze snadno překonat, pokud použijeme semi-online přístup, kdy online data předkládáme v balících.[23]

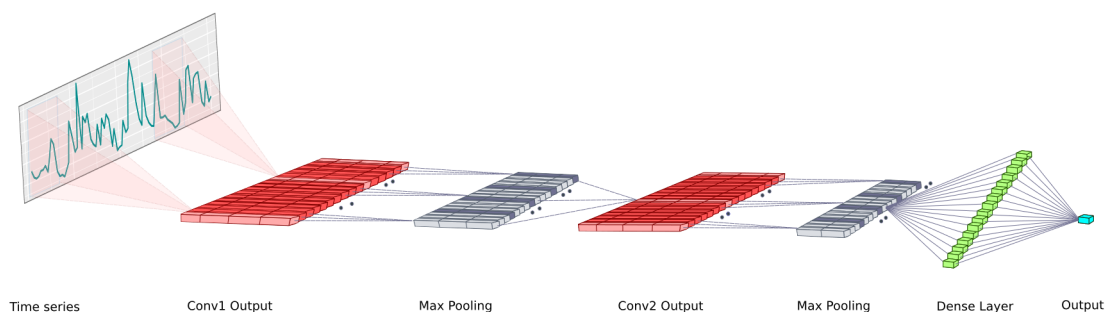
2.1.7 Metoda DeepAnT

Metoda DeepAnt má velmi podobnou architekturu jako LSTM. Skládá se ze dvou částí. První z nich je modul pro předpovědi (*angl. time series predictor*), který slouží pro předpověď dalšího datového bodu v sekvenci. Rozdíl oproti LSTM spočívá v použití Konvoluční neuronové sítě (CNN) místo LSTM jednotek. Předpověď následujícího datového bodu je následně předána do druhého modulu, což je detektor anomálií. Ten na základě odhadu CNN a skutečného datového bodu rozhodne, jestli se jedná o anomálii nebo ne.

Jedná se o metodu používající učení bez učitele. Podle článku [24] nemusíme síť trénovat pouze na normálních datech, ale klidně ji můžeme předložit i data s anomáliemi, pokud zaručíme, že jejich četnost bude do 5% z celkového počtu. Další výhodou této metody je relativně nízký počet trénovacích dat nutný pro naučení sítě. Autoři článku porovnávali počet nutných dat s architekturou LSTM. Datová úspora činila cca 40% trénovacích dat. Tato metoda (stejně jako LSTM) navíc dokáže přesně označit anomální bod/skupinu bodů v signálu. Neoznačí tedy pouze celou datovou řadu jako anomálii, ale dokáže anomálii detekovat přesněji.

V implementačním řešení metody vycházející z článku [24] se modul předpovědi skládá ze dvou konvolučních vrstev následovaných max-pooling vrstvou. Na ně navazuje plně propojená vrstva. Pro učení je použit SGD (Stochastic gradient descend). Každá konvoluční vrstva má 32 kernelů následovaných ReLU aktivační funkcí. Jako detektor anomálií slouží eukleidovská vzdálenost datových bodů. Práh anomaly se určuje z validačních dat.

Vizualizaci architektury z článku [24] můžete vidět na obrázku 2.8



Obr. 2.8: Vizualizace architektury DeepAnt. Dostupné na[24]

2.1.8 Metoda Sub-IF

Metoda Isolation forest je založena na tvorbě rozhodovacích stromů. Cílem je vytvořit rekurzivně stromovou strukturu, kdy v každém uzlu rozdělujeme vstupní data podle náhodně zvoleného atributu, dokud nedojdeme do stavu, že každá instance je izolována v samostatném listu. Pokud vytvoříme velké množství náhodně generovaných stromů, vznikne tzv. Isolation forest.

Princip metody Isolation forest spočívá ve vlastnosti anomálií. Anomálií je v datech obecně málo a od ostatních dat se velmi liší. Díky tomu se anomální data ve vzniklé stromové struktuře snadno a rychle izolují. Anomální instance jsou tudíž podstatně blíže kořenu než instance normální. Normální instance jsou izolovány podstatně dále od kořene.

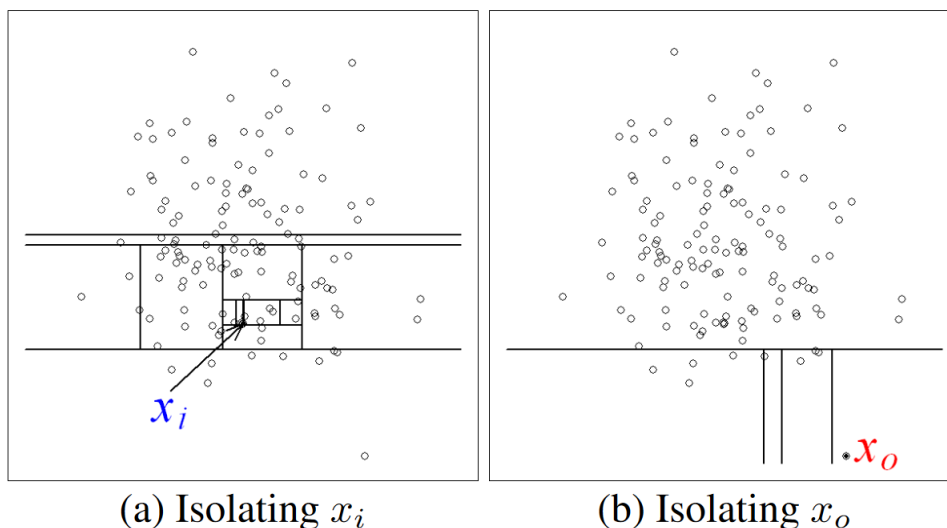
Rozdíl v náročnosti izolace normálních dat a anomálií demonstruje obrázek 2.9.

Vzhledem k velkému počtu náhodných stromů ve struktuře Isolation forest je tedy jako ukazatel anomality použita střední vzdálenost dané instance od kořenu jednotlivých stromů.[25]

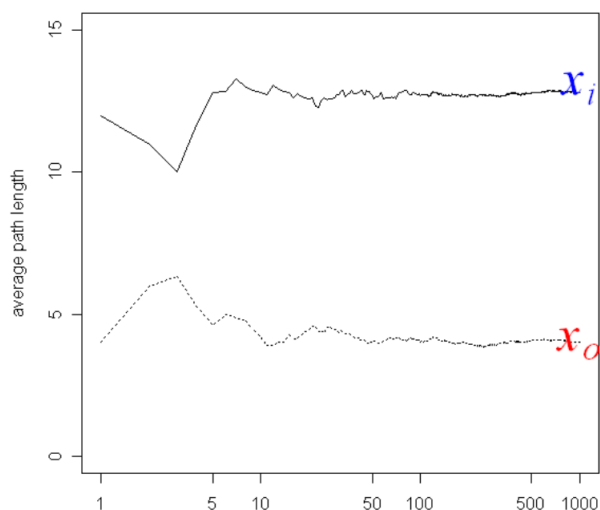
Příklad závislost střední hodnoty vzdálenosti normálního datového bodu a anomálie na počtu použitých stromů je uveden na obrázku 2.10. Z obrázku je vidět, že už několik desítek stromů dává velice stabilní výsledek.

Výhoda této metody spočívá v malé výpočetní a paměťové náročnosti. Další optimalizaci výpočetní náročnosti můžeme provést díky tomu, že stromy nemusíme budovat celé, protože anomálie se, jak je zmíněno výše, oddělí dříve než normální instance.[25]

V článku [25] nebyla přímo uvedena implementace pro časové řady, resp. sekvencní data.



Obr. 2.9: Rozdíl izolace normálního datového bodu (vlevo) a anomálie (vpravo) Dostupné na[25]



Obr. 2.10: Průměrná vzdálenost normálního datového bodu (modrá) a anomálie (červená) Dostupné na[25]

2.2 Oblast vizuální inspekce

Jedním ze zásadních oblastí Průmyslu 4.0 je i vizuální inspekce, která je pouze jedním z módů sběru dat. Tato sekce diplomové práce bude věnována získávání informací o anomáliích např. výrobků na základě jejich obrazových dat.

Metod pro detekci anomálií v této oblasti je velké množství. Dají se použít různé

varianty metod strojového učení pro zpracování obrazových dat jako například Konvoluční neuronové sítě (CNN) a Generativní sítě (GAN) nebo obecné metody strojového učení jako Bayesovské sítě a Rozhodovací stromy.

Vzhledem k tomu, že práce by měla být věnována spíše prediktivní údržbě, bude tato sekce podstatně méně obsáhlá než předchozí. Navíc některé metody uvedené v sekci prediktivní údržba (např. Isolation forest) by mohly být použity také v oblasti vizuální inspekce a naopak.

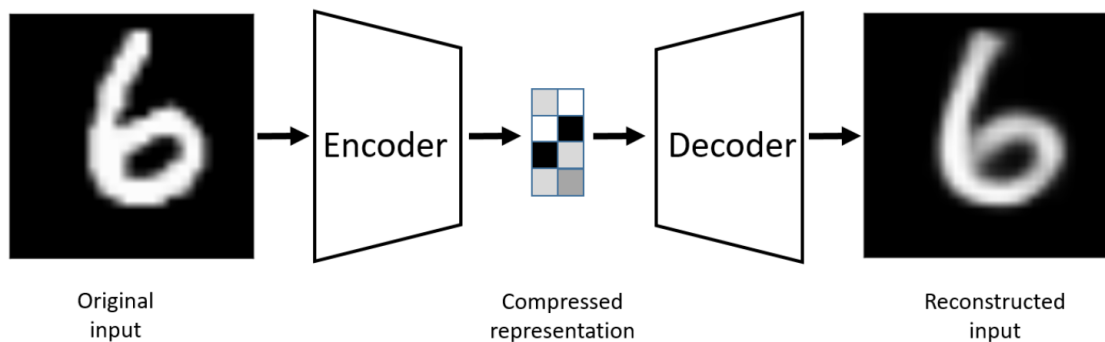
2.2.1 Autoenkodéry

Autoenkodéry se skládají ze dvou neuronových sítí. První z nich je enkodér, který převede vstupní data (obrázek) do takzvané latentní reprezentace, kdy jsou vstupní data reprezentována menším počtem datových bodů než původně obsahovala.

Druhá část autoenkodérů je tzv. dekodér, který je schopen z latentní reprezentace získat v ideálním případě původní data (obrázek). Enkodér i dekodér mohou být různé neuronové sítě, např CNN nebo klasická dopředná neuronová síť s algoritmem error back propagation. [26]

Při učícím procesu se autoenkodér snaží nastavit váhy enkodéru i dekodéru tak, aby dokázal mít na vstupu i výstupu stejná data. Autoenkodér dokáže zrekonstruovat pouze data, na které byl natrénován. Toho se dá využít při detekci anomálií, kdy autoenkodér dokáže zrekonstruovat správně pouze normální vzorky, na které byl naučen. [26]

Princip autoenkodérů můžete vidět na obrázku 2.11



Obr. 2.11: Ilustrativní příklad použití autoenkodérů. Dostupné na[26]

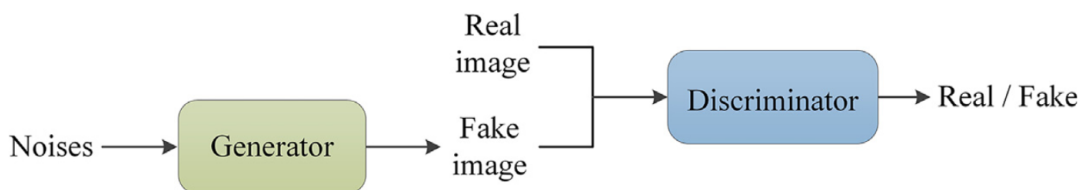
2.2.2 Metoda Generative Adversarial Network

Generative adversarial network nebo také GAN je architektura, která původně vznikla pro generování obrazových trénovacích dat pro CNN. Skládá se ze dvou jednotek. Generator G a Diskriminátor D spolu hrají hru nulovým součtem. Generátor se snaží naučit rozložení vstupního datasetu a Diskriminátor se snaží rozpoznat, jestli data, která dostává, jsou ze vstupního datasetu nebo jsou vygenerovány Generátorem. [27, 28]

Cílem G je co nejlépe napodobovat data ze vstupního datasetu a D se snaží co nejvíce zlepšit v detekci vygenerovaných a vstupních dat. G a D mohou mít různou architekturu může se jednat například o CNN nebo RNN. Pokud naučíme G a D pouze na normálních datech, můžeme získat velmi zajímavý detektor anomálií.[27, 28]

Článek [28] pracuje s dvěma přístupy. G-skóre vychází z rekonstrukční chyby generátoru, který je schopen zrekonstruovat pouze normální vzorky. D-skóre používá pro určení anomálního skóre Diskriminátor, který pozná normální data, na která byl naučen a abnormality, které mají jiné rozložení než normální data.

Základní architekturu GAN můžete vidět na obrázku 2.12



Obr. 2.12: Základní architektura GAN. Dostupné na[27]

3 Datasets

Tato kapitola bude věnována datasetům, z nichž některé budou použity v implementační části této práce. Datasetů lze v dnešní době najít na internetu velké množství. V této kapitole budou některé představeny a bude zhodnocena vhodnost jejich použití.

3.1 Typy datasetů

Datasets můžeme dělit podle různých kritérií. Jako první se nabízí rozdělení na základě původu dat. Na základě toho můžeme data rozdělit na syntetická a reálná. Výhoda reálných datasetů je, že reprezentují skutečný proces, kdežto výhoda syntetických datasetů je možnost tvorba libovolných anomálií v libovolném množství s relativně nízkými náklady.

Další dělení můžeme uskutečnit podle problému, který data představují. Při vizuální inspekci používáme především obrazová data, kdežto při prediktivní údržbě se využívají časové řady nebo pouze "jednorázová" vstupní data bez časové závislosti.

Také může rozdělit datasets na jednorozměrné a vícerozměrné podle počtu datových bodů, které najednou přicházejí na vstup algoritmu strojového učení. Některé algoritmy lépe zpracovávají vícerozměrná, jiné zase jednorozměrná data. Pokud vybíráme vhodný algoritmus strojového učení, je potřeba důkladně zvážit tvar vstupních dat.

3.2 Zdroje datasetů

V dnešní době je velké množství datasetů volně dostupných na internetu. V článcích se pro zaručení reprodukovatelnosti výsledků práce nacházejí odkazy, z nichž lze použité datasets získat.

Jedním z míst pro ukládání datasetů je <https://www.kaggle.com>. Jedná se o portál, který je zaměřen na datové vědy a strojové učení. Tato stránka mimo jiné nabízí uživatelům soutěže v datových vědách, umožňuje sdílet datasets a s nimi související kódy pro algoritmy strojového učení. Dále poskytuje prostor pro uživatelské diskuze. Jedna z velmi významných částí tohoto portálu je sekce pro výuku strojového učení.

Dále lze nalézt velké množství datasetů na stránkách <https://archive.ics.uci.edu>, což je repozitář zaměřený na strojové učení a inteligentní systémy nebo na internetových stránkách univerzit a výzkumných institucí.

3.3 Představení konkrétních datasetů

V této části budou představeny datasety, které by bylo teoreticky možné použít v oblasti detekce anomálií v časových řadách. U každého bude zhodnocena vhodnost pro použití v této diplomové práci.

Názvy sekcí v této části budou pojmenovány anglicky přímo podle jednotlivých datasetů. Ty jsou orientačně seřazeny podle předpokládané použitelnosti v další práci.

3.3.1 Pump sensor data

Tento dataset obsahuje časovou řadu provozních parametrů vodní pumpy, která se nachází ve stavech: normální, porucha a zotavení z poruchy. Jeden záznam tvoří data 52 blíže nespecifikovaných senzorů. Jedná se o jednu časovou řadu, kterou tvoří cca 220 000 záznamů popisujících 4 měsíce provozu. Tento dataset bude použit pro pokusy v praktické části, proto bude rozebrán více do detailu než ostatní. Dále také bude rozebráno předzpracování dat a jejich rozdělení na testovací a trénovací část.

Dataset je anotovaný. Vodní pumpa se nachází ve třech stavech (NORMAL, BROKEN, RECOVERING). Stav BROKEN se nachází v datasetu pouze 7x a je velmi krátký. Proto je za anomální stav je považován i stav RECOVERING, který vždy následuje bezprostředně za stavem BROKEN.

Jednotlivé časové řady jsou zobrazeny na obrázku 3.1. Anomální úseky jsou zde zvýrazněny. Stav BROKEN je zvýrazněn červeně a stav RECOVERING je zvýrazněn žlutě.

Data bylo nutné předzpracovat. Jedním z hlavních procesů byla náhrada 'Nan' hodnot v datasetu. Na základě počtu 'Nan' hodnot v datasetu byly senzory s indexem 0, 15, 50 a 51 vyřazeny z datasetu.

Zbylé hodnoty 'Nan' byly nahrazeny poslední hodnotou v časové řadě. Tento způsob náhrady byl zvolen, protože minimalizuje vznik nových anomálií v existujícím signálu. Tento předpoklad lze uplatnit, pokud počet 'Nan' hodnot v signálu není příliš veliký. Toto je naštěstí možné, protože, tyto časové řady byly v předchozím kroku odstraněny.

Dataset byl také podroben statistické analýze. Byla vytvořena korelační matice. Z ní je dobře poznat, že některé signály v Pump sensor dataset mají mezi sebou silnou korelaci. Korelační matici můžete vidět na obrázku 3.2.

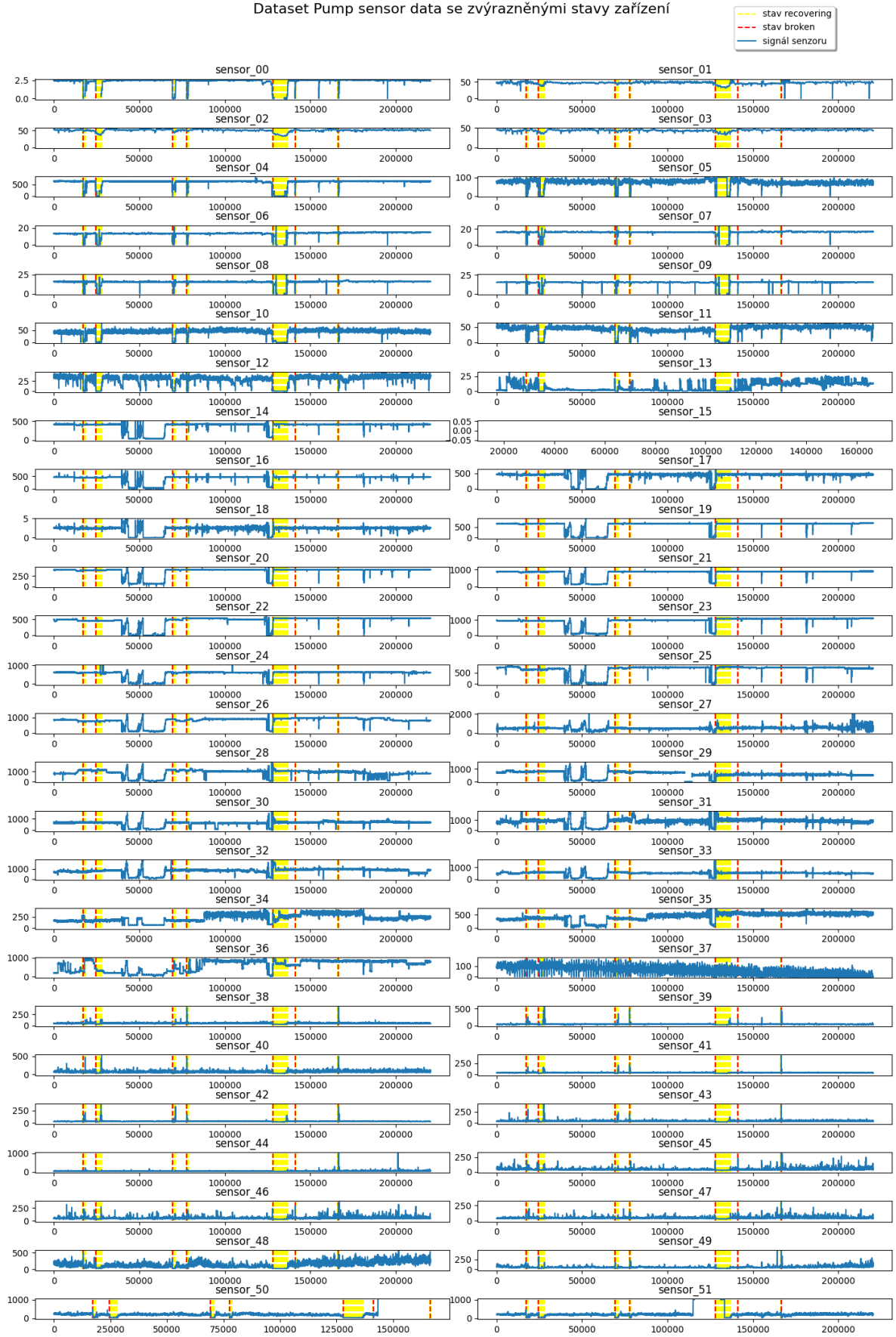
Jak už bylo zmíněno výše, dataset obsahuje cca 220 000 časových záznamů. Pro trénování neuronových sítí, které na vstup potřebují průběh signálu bez anomálií, bylo vyčleněno posledních 50 000 záznamů, které jsou bez anomálií. Zbytek datasetu byl použitý jako testovací. Trénovací soubor je dále dělen na trénovací a validační.

Jejich poměr je nastaven pomocí knihovních funkcí knihoven použitých pro strojové učení.

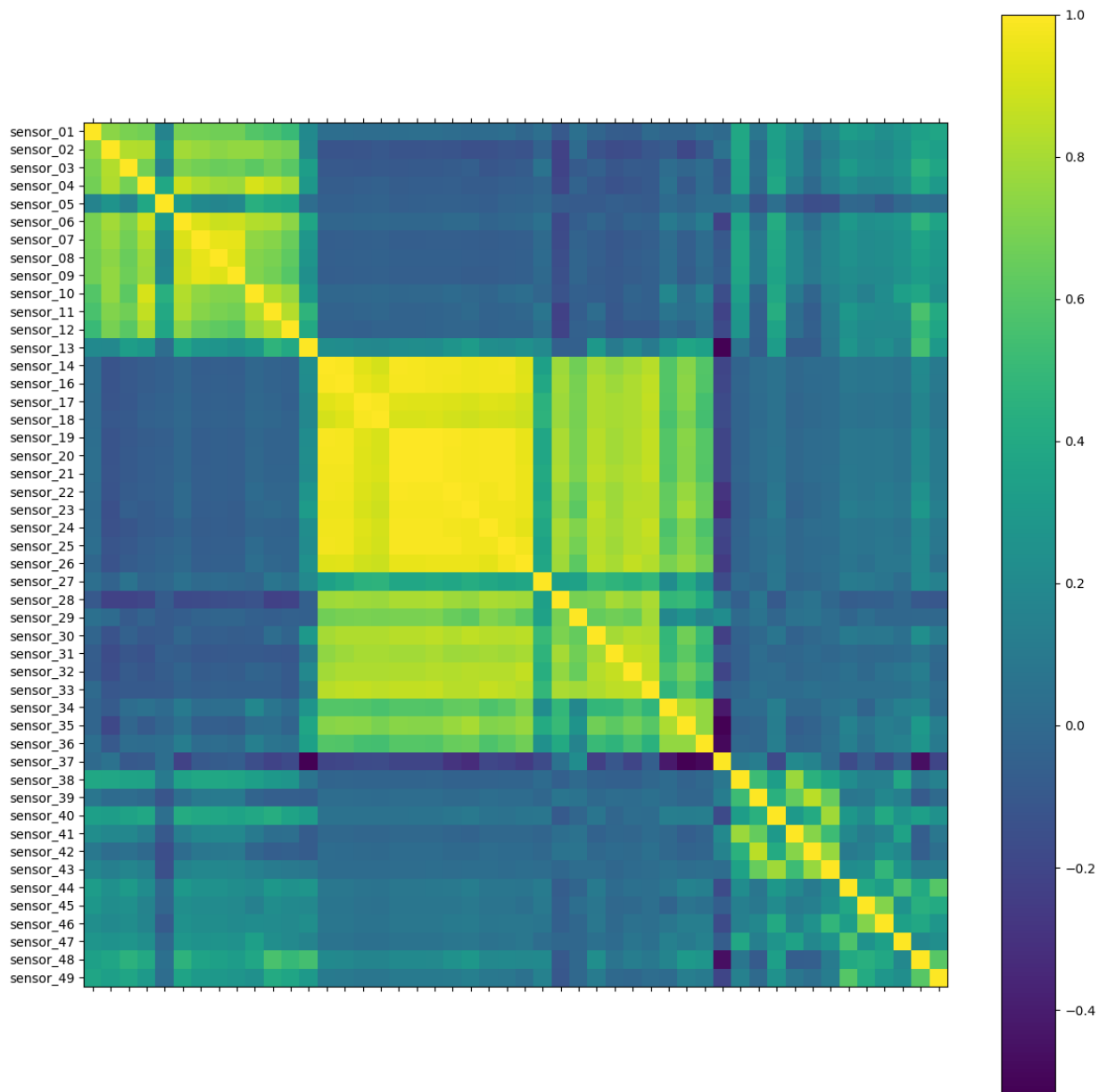
Některé algoritmy použité pro detekci anomálií mohou vyžadovat pro trénování anotovaná data, která obsahují určité procento anomálních vzorků. V tomto případě bylo pro trénování použito 50 000 vzorků z počátku datasetu.

Více informací včetně samotného datasetu lze nalézt na [49].

Dataset Pump sensor data se zvýrazněnými stavy zařízení



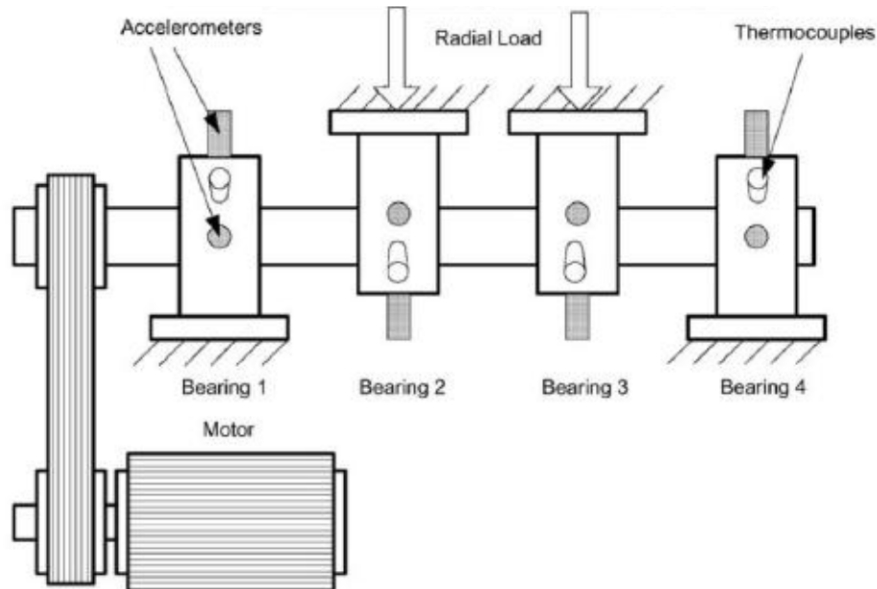
Obr. 3.1: Ilustrace signálů datasetu Pump sensor data se zvýrazněnými stavy zařízení



Obr. 3.2: Korelační matice signálů datasetu Pump sensor data. Na horizontální ose jsou stejné popisky jako na vertikální

3.3.2 NASA Bearing Dataset

Tento dataset obsahuje čtyři testy čtyř ložisek, kdy na každém jsou v pravidelných intervalech (každých pět resp. deset minut) měřeny vibrace (po dobu jedné sekundy se vzorkovací frekvencí 20 kHz). Celý experiment popisuje obrázek 3.3.



Obr. 3.3: Ilustrace pokusu pro získání NASA Bearing Datasetu. Dostupné na[31]

Každý test běžel tak dlouho, dokud alespoň jedno ze čtyř sledovaných ložisek neselhalo. Dataset sice není anotovaný, ale u každého testu zvlášť je známé, která ze čtyř ložisek selhala. Každý záznam se skládá celkem z 8 kanálů měření, kdy každé dva patří jednomu ložisku.

Vzhledem k tomu, že tento dataset obsahuje velké množství časových řad, mohl by být velmi vhodný pro další praktickou část této diplomové práce.

Další informace včetně datasetu lze nalézt na [31].

3.3.3 Skoltech Anomaly Benchmark

Tento dataset ve verzi 0.9 obsahuje 34 časových řad. V další verzi autoři slibují jeho masivní rozšíření. Dataset je zaměřen především na odhalení bodových a kontextových anomálií.

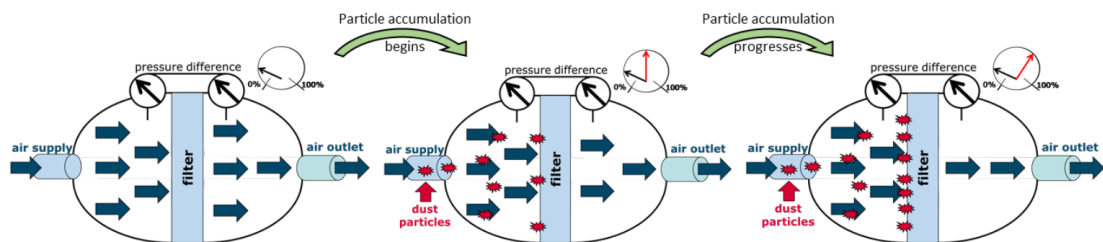
Každá z časových řad obsahuje několik atributů, na jejichž základě lze určit anomalitu konkrétního vzorku. Mezi atributy patří data z akcelerometrů, informace o tlaku, teplotě a dalších provozních parametrech.

Celý dataset obsahuje cca 40 000 záznamů, kde každý záznam má 8 atributů. Dataset je anotovaný, takže lze poznat, kdy nastala anomálie a to porovnat s našimi výsledky.

Tvůrci datasetu připravili také kód, který je schopný v datech detekovat anomálie. Dataset i s algoritmy tvůrců lze najít na [36].

3.3.4 Preventive to Predictive Maintenance

Tento dataset sleduje provoz filtru pevných částic od jeho uvedení do provozu až po jeho selhání. (*angl. run to failure*). Celý pokus schématicky znázorňuje obrázek 3.4



Obr. 3.4: Schéma pokusu pro vytvoření datasetu Preventive to Predictive Maintenance. Dostupné na[26]

Dataset je určen pro algoritmy, které se snaží na základě 5 provozních parametrů jako je diference tlaků před filtrem a za filtrem, průtoku, velikosti prachových částic a dalších zjistit RUL (*angl. remaining usefull life*).

Dataset obsahuje zvlášť trénovací a testovací data ve formě časové řady. Trénovací data obsahují 50 časových řad, které dohromady tvoří cca 39 000 záznamů. Testovací data obsahují zhruba stejný počet záznamů. Testovací část je anotovaná pomocí atributu RUL, trénovací nikoliv.

Tento dataset by mohlo být možné použít pro další práci.

Další informace lze nalézt na [33].

3.3.5 Ostatní datasety

V této části budou uvedeny datasety, které byly vzaty v úvahu, ale byly vyhodnoceny jako méně vhodné. Nemusí se jednat o nevhodné datasety, ale jsou méně vhodné než ty, kterým byla věnována samostatná sekce.

Dataset **One Year Industrial Component Degradation** popisuje rok provozu řezací čepele v průmyslu. Dataset obsahuje 519 souborů o 2048 záznamech (celkem tedy přes 1 000 000 záznamů), které s periodou 4 ms popisují celkem 8 provozních parametrů řezací čepele jako např. moment, aktuální rychlost, aktuální

poloha a další. Každý soubor má v názvu časovou značku začátku měření a popisuje vždy 8 sekund provozu stroje. Dataset není anotovaný, takže nemá popisky, kdy už je čepel opotřebovaná. Tento problém by mohl jít vyřešit pomocí následujícího datasetu, který popisuje totožný problém, ale obsahuje data pro novou i opotřebovanou čepel. Tento dataset obsahuje velké množství časových řad s relativně zajímavým počtem provozních parametrů a proto by mohl být použit v další práci. Více informací o datasetu lze najít na [40].

Stejnou situaci popisuje dataset **Vega shrink-wrapper component degradation**. Je dokonce vytvořen v rámci stejného projektu. Na rozdíl od předchozího datasetu se skládá pouze ze 6 souborů po 2048 záznamech z nichž 3 popisují novou a 3 opotřebovanou čepel. Tento dataset by se mohl hodit jako doplněk předchozího datasetu. Proto by mohl být využit v další práci. Více informací o datasetu lze najít na [41].

Dataset **Microsoft Azure Predictive Maintenance** vznikl jako součást Azure AI Notebooks for Predictive Maintenance, ale v dnešní době je dostupný právě na serveru <https://www.kaggle.com>. Na základě popisu nebylo možné určit, jestli se jedná o data syntetická nebo reálná.

Dataset se skládá z několika souborů, v nichž se nachází tato data: průměrné provozní parametry aktualizované každou hodinu jako napětí, rotace, vibrace aj. Dále obsahuje záznamy o chybách a selháních (*angl. errors and failures*) a také záznamy o údržbě. Každý soubor je různě obsáhlý. Např. soubor s provozními parametry obsahuje přes 870 000 záznamů a 5 atributů. Dataset je anotovaný, takže lze přesně poznat, kdy nastala chyba (*angl. error*) nebo selhání (*angl. failure*).

Tento dataset je přímo zaměřen na prediktivní údržbu a také obsahuje časové řady, které by se teoreticky mohly dát použít k detekci anomálií. I vzhledem k rozsahu ho považuji za relativně vhodný pro naše účely.

Více informací v čteně datasetu lze nalézt na [30].

Condition Based Maintenance of Naval Propulsion Plants Data Set pochází ze simulátoru námořního plavidla. Obsahuje 18 atributů, které popisují mimo jiné rychlost lodi, degradaci kompresoru a degradaci turbíny. Obsahuje cca 11000 záznamů a přestože neobsahuje časové značky, jedná se pravděpodobně o časové řady. Dataset je anotovaný pomocí koeficientu rozkladu kompresoru a turbíny. Jedná se o jeden časový záznam provozu zařízení do jeho selhání. Vzhledem k tomu, že se jedná pouze o jeden časový průběh, považuji tento dataset za méně vhodný pro další práci. Více informací o datasetu lze najít na [43].

NASA Turbofan Jet Engine Data Set obsahuje data ze simulátoru provozu 4 proudových motorů až do jejich selhání. Jeho cílem je předpověď RUL. Každý motor začíná svůj provoz s jiným stavem opotřebení. Dataset je rozdělen na 4 testovací a 4 trénovací soubory. Každý z nich obsahuje 26 atributů. Celkový počet záznamů

v 8 souborech je cca 250 000. V trénovacích datech vzrůstá opotřebením motoru lineárně až do jeho selhání. Testovací data anotovaná nejsou. Tento dataset by mohl být vhodný pro další práci. Více informací o lze najít na [44].

Oxford Battery Degradation Dataset 1 Tento dataset sleduje životnost lithium-iontových akumulátorů. Popisuje životní cyklus celkem 8 akumulátorů, u kterých sleduje teplotu, napětí a náboj. Každý akumulátor má životnost minimálně 5000 vybíjecích cyklů, z nichž je zaznamenaný každý stý. Jeden vybíjecí cyklus obsahuje cca 3500 záznamů. Tento dataset je velmi zajímavý. Otázka použitelnosti spočívá především v relativně nízkém počtu sledovaných atributů. Více informací lze nalézt na [47].

Condition monitoring of hydraulic systems Data Set obsahuje provozní parametry hydraulického systému. Dataset je anotovaný, takže víme, v jaké kondici se hydraulický systém nacházel. Dataset obsahuje 17 atributů a cca 2000 záznamů. Vzhledem k malému počtu dat je tento dataset poměrně nevhodný pro další práci. Více informací lze nalézt na [46]

Machine Predictive Maintenance Classification je syntetický dataset, který byl speciálně vyvinutý pro oblast prediktivní údržby. Dataset se skládá z 10 000 vzorků, z nichž každý obsahuje 14 atributů. Mezi ně patří: jednoznačný identifikátor, kvalita výrobku, teplota vzduchu, procesní teplota a další . . .

Dataset je anotovaný. Budto můžeme rozpoznávat, jestli chyba nastala nebo můžeme rozpoznávat přímo typ chyby. Tvůrci datasetu upozorňují, aby nebyl typ chyby zaměňován s atributem. To by mohlo vést ke zkreslení výsledků učícího mechanismu. Tento dataset bohužel neobsahuje časové řady provozních parametrů, ale pouze jednorázová data bez časové posloupnosti. Další problém by mohl být jeho rozsah. Proto považují tento dataset pro naše účely za méně vhodný. Další informace o tomto datasetu lze najít na [29]

Elevator Predictive Maintenance Dataset obsahuje data z německých závodů firmy Huawei pro výrobu výtahů. Cílem tohoto datasetu je předpovídat absolutní hodnotu vibrací na základě ostatních provozních parametrů. Dataset je anotovaný, takže známe cílovou hodnotu vibrace. Dataset obsahuje časovou řadu o cca 112 000 záznamech a cca 7 atributů. Vzhledem k počtu záznamů a tomu, že se jedná o časové řady, by tento dataset mohl vyhovovat našim účelům. Další informace včetně datasetu lze nalézt na [32].

Numenta Anomaly Benchmark (NAB) je kompilace několika časových řad z různých odvětví jako například data o dopravě, data o selháních klimatizačního systému a další. Některá data jsou naměřena v reálných situacích a některá jsou syntetická. Bližší informace i s návodnými algoritmy a doporučeným způsobem použití lze nalézt v jejich repozitáři na [37].

Jedná se o jednorozměrné časové řady, kdy každý časový průběh má řádově

několik tisíc řádků. Dataset není anotovaný, takže nelze přesně určit, kdy nastala anomálie. Vzhledem k tomu, že dataset obsahuje velké množství různých datasetů, nebude použit v další práci.

Predictive Maintenance: Naval Vessel Condition obsahuje data k prediktivní údržbě motorů námořních lodí. Dataset obsahuje necelých 12 000 záznamů s 16 atributy. Nejedná se sice o časová data a jednotlivé záznamy nejsou anotované z hlediska přítomnosti chyby na zařízení. Proto tento dataset nebude využit v další práci. Pro více informací lze najít na [38].

E-coating ultrafiltration maintenance dataset popisuje vývoj provozních parametrů stroje pro zpracování barvy v oblasti elektro-forézní malířství. Schéma experimentu je zobrazeno na obrázku 3.5 Dataset se skládá ze dvou částí, kdy každá má jinou originální vzorkovací frekvenci a jinou dobu sběru dat. Větší dataset popisuje sběr dat po dobu patnácti dní, kdy výsledná data byla převzorkována na vzorkovací periodu 30 minut.

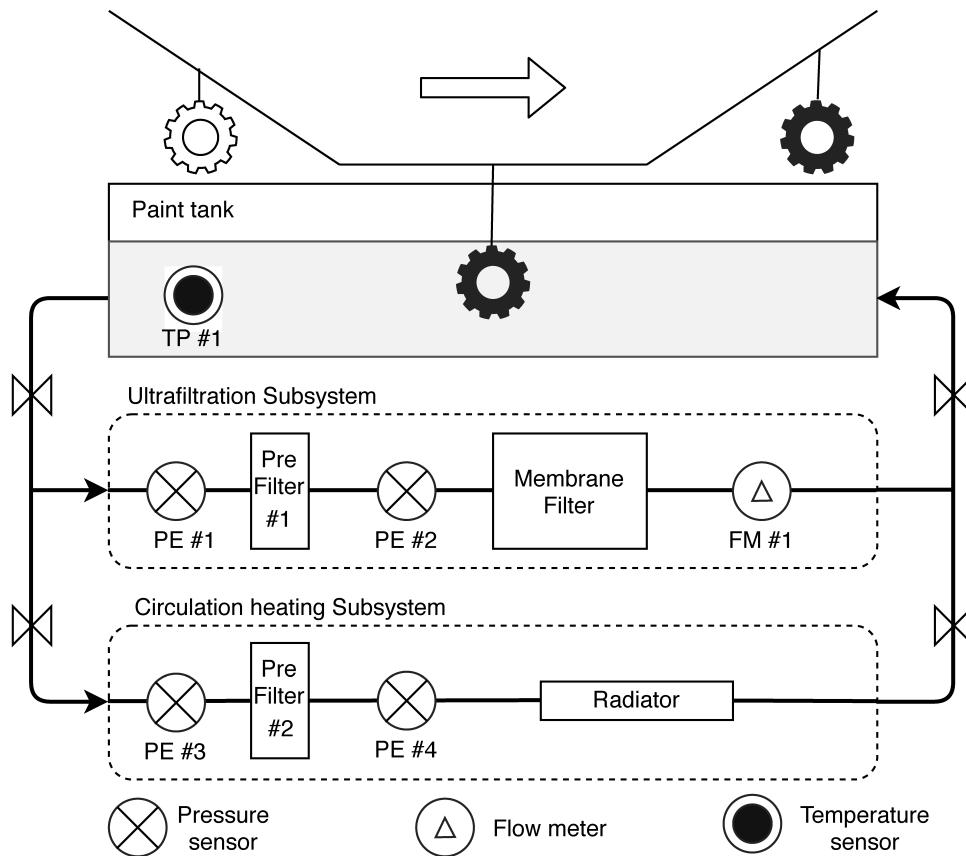
Dataset obsahuje cca 110 000 záznamů o 7 attributech jako jsou například tlak a teplota v modulech zařízení. Dataset není anotován z hlediska výskytu chyb na zařízení. Ty by šly sice z dat určit manuálně, ale výše máme uvedeny podstatně vhodnější datasety. Vzhledem k velikosti a skutečnosti, že se jedná o časové řady, by tento dataset mohl být vhodný pro další práci. Více informací na [39].

Genesis demonstrator data for machine learning Tento dataset popisuje chování přípravku, který na základě požadavků uživatele přesouvá materiál mezi čtyřmi stanicemi. Dataset obsahuje 2 části. Každá z nich popisuje samostatný provoz zařízení. Celkový počet atributů je 18. Celý dataset obsahuje cca 16 000 záznamů. Dataset není anotovaný, jestli došlo k chybě nebo ne. Tento dataset není vhodný pro další práci. Více informací o datasetu lze nalézt na [45].

Call2 Building People Counts Data Set Tento dataset popisuje vstupování lidí do budovy a jejich odchod z budovy po 15 týdnů. Skládá se z 5000 záznamů obsahujících 2 atributy. Každou půlhodinu se zaznamená kolik lidí vstoupilo dovnitř respektive odešlo z budovy. Cílem je odhadnout, kdy se v budově koná nějaká zvláštní událost (anomálie). Vzhledem k malému počtu atributů i relativně malému počtu dat je tento dataset zařazen mezi méně vhodné pro další práci. Více informací k danému datasetu lze nalézt na [48]

Healthcare Providers Data For Anomaly Detection Tento dataset, jak už název napovídá, nabízí data pro detekci podvodů v proplácení zdravotní péče. Vzhledem k tomu, že data nejsou označena, jsou vhodná pouze pro učení bez učitele.

Databáze obsahuje 100 000 záznamů o výkonech zdravotní péče. Bohužel se nejedná o časové řady, dataset není anotovaný a nejedná se ani o data z průmyslu. Proto tento dataset nebude v další práci použit. Další informace včetně datasetu lze nalézt na [34].



Obr. 3.5: Schéma experimentu pro získání E-coating ultrafiltration maintenance datasetu. Dostupné na[39]

Eighty years of Canadian climate data Tento dataset obsahuje data mapující počasí v Kanadě každý den od roku 1940 do roku 2020. Data o počasí jsou zjišťována na sedmi stanicích a od roku 1960 na třinácti stanicích umístěných ve významných kanadských městech. Jedná se o časové řady. Z toho důvodu tvůrci doporučují dataset použít pro předpověď dalších datových bodů v sérii. I přesto, že se nejedná o data z průmyslu, mohlo by být použití těchto dat pro zpracování např. LSTM velmi zajímavé. Další informace včetně datasetu lze nalézt na [35].

Energy of Las Vegas Street Tento dataset popisuje spotřebu elektrické energie v několika ulicích Las Vegas po dobu 2 let (2014-2016). Vzorkovací perioda časových řad je 15 minut. Dataset obsahuje 21 atributů a má přes 1 000 000 záznamů. Tento dataset pouze popisuje spotřebu energie. Neobsahuje žádné záznamy o selháních ani jinou anotaci, kterou by šlo využít v další práci. Více informací o datasetu lze najít na [42].

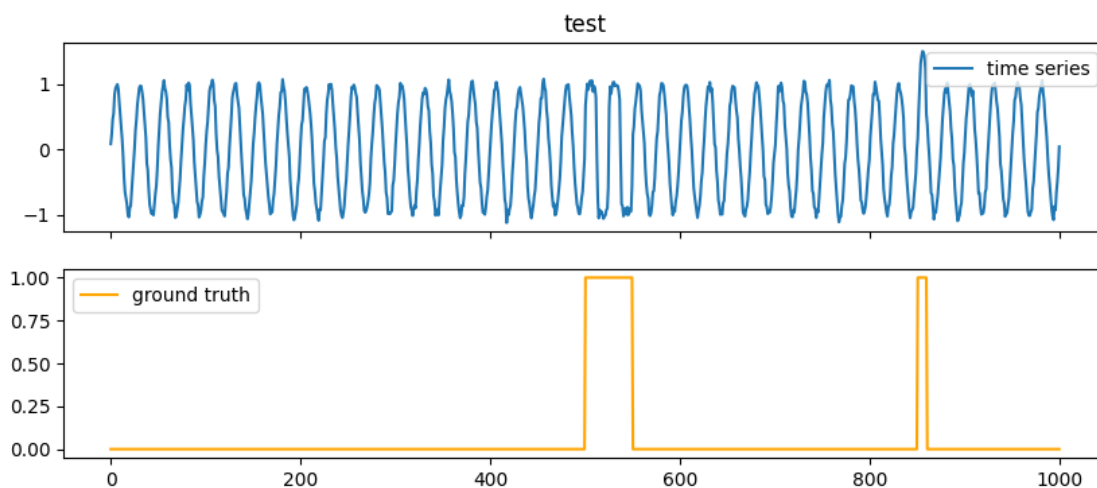
3.4 Generátory datasetů

Autoři článků, ve kterých jsou použity vlastní syntetické datasety, často nechávají svoje systémy ve volně dostupných repozitářích (např. github). Mezi tyto systémy patří např. GutenTAG nebo MGAB.

3.4.1 GutenTAG

GutenTAG (*z angl. A good Timeseries Anomaly Generator*) je volně šiřitelná knihovna napsaná v jazyce Python, díky které lze generovat jednorozměrné i vícerozměrné časové průběhy s různými druhy anomálií i bez nich.

Technický popis knihovny včetně návodu k použití lze nalézt v repozitáři na GitHubu na [50] Na obrázku 3.6 můžete vidět příklad vygenerovaného signálu včetně umístění anomálií.



Obr. 3.6: Příklad signálu generovaného pomocí GutenTAG. Dostupné na[50]

3.4.2 MGAB

MGAB (*z angl. The Mackey-Glass Anomaly Benchmark*) je vytvořen pomocí časových řad Mackey-Glass. Jedná se o časové řady, které jsou tvořeny pomocí nelineárních diferenciálních rovnic a jejich výsledkem je signál, ve kterém se vyskytují netriviální anomálie.

Technický popis včetně návodu k použití lze nalézt v repozitáři na GitHubu na [51] V tomto repozitáři se také nachází několik už vygenerovaných časových řad.

3.5 Generovaný dataset

Z generátoru GutenTAG budou v praktické části použity tři základní testovací signály: sinus, obdélníkové pulsy a Dirichletova vlna. Délka trénovacího i testovacího signálu je 10 000 datových bodů. Konfigurační soubory pro jednotlivé testovací signály jsou uvedeny v příloze.

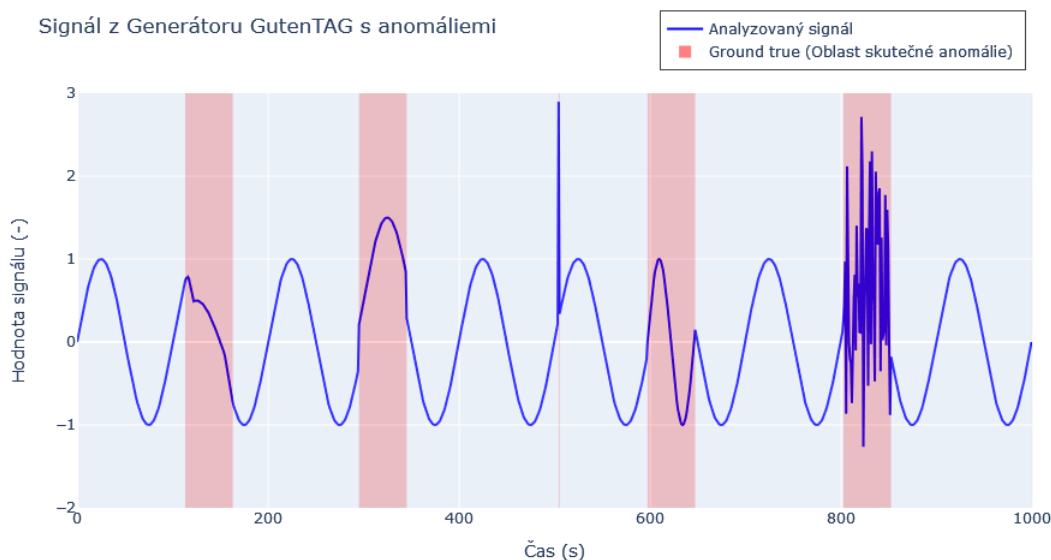
K sinusovému signálu byly přidány anomálie typu změna amplitudy, změna offsetu, lokální extrém, změna frekvence a výrazné zvýšení hladiny šumu. Příklad sinusového signálu s anomáliemi ve výše specifikovaném pořadí je zobrazen na obrázku 3.7.

K obdélníkovým impulsům byly přidány anomálie typu změna amplitudy, změna offsetu, lokální extrém, změna frekvence, změna střídy a výrazné zvýšení hladiny šumu. Příklad obdélníkového signálu s anomáliemi ve výše specifikovaném pořadí je zobrazen na obrázku 3.8.

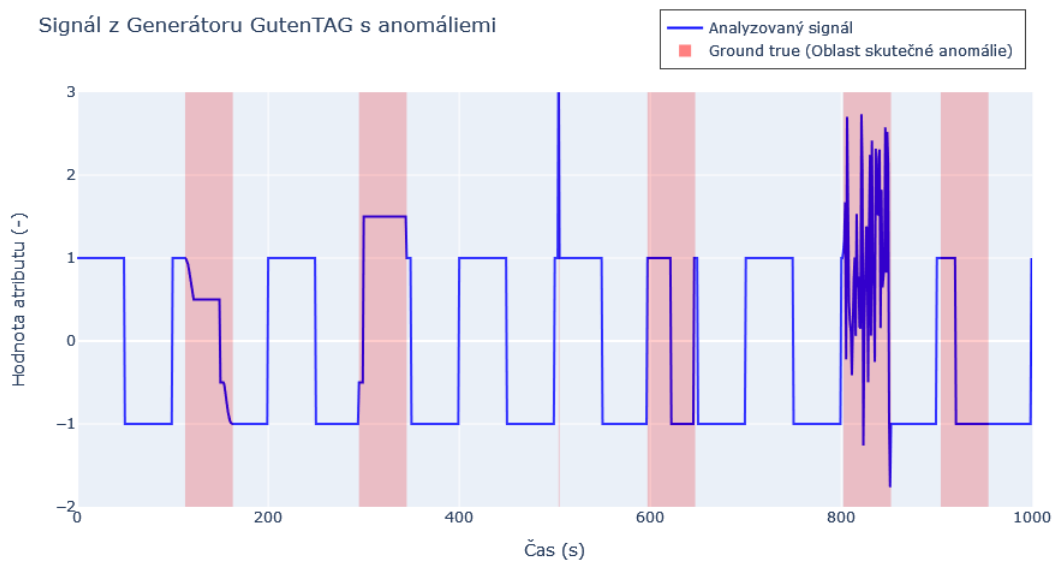
K Dirichletově vlně byly přidány anomálie typu změna amplitudy, změna offsetu, lokální extrém, změna frekvence a výrazné zvýšení hladiny šumu. Příklad obdélníkového signálu s anomáliemi ve výše specifikovaném pořadí je zobrazen na obrázku 3.9.

Ukázky testovacích signálů jsou záměrně zkráceny na 1000 datových bodů, aby na nich vynikl tvar křivky, resp. tvar anomálií.

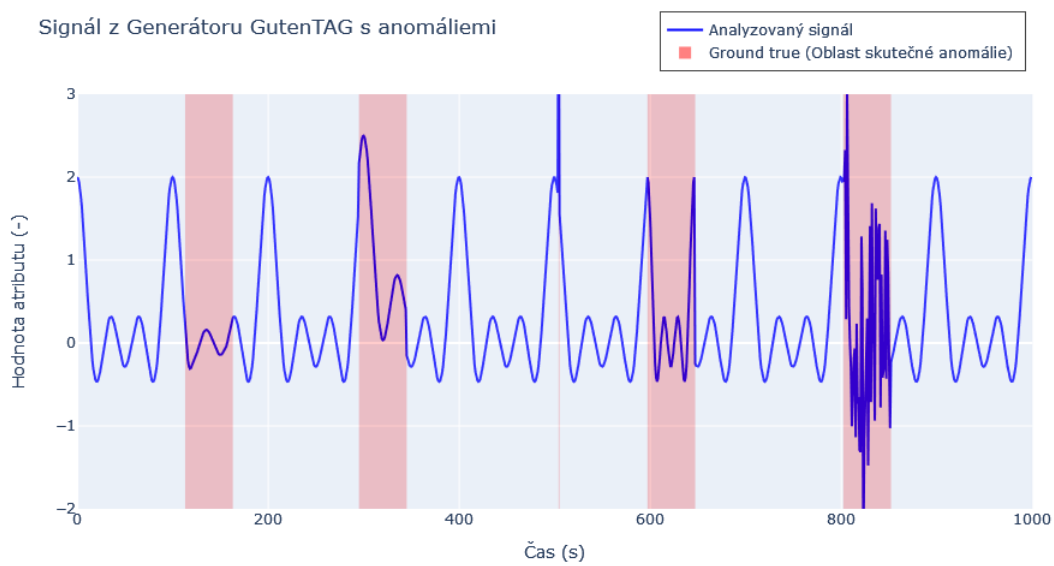
Ukázka části konfiguračního skriptu k jednotlivým testovacím signálům jsou uvedeny v příloze A.1.



Obr. 3.7: Příklad sinusového signálu s anomáliemi generovaného pomocí GutenTAG



Obr. 3.8: Příklad čtvercových pulzů s anomáliemi generovaných pomocí GutenTAG



Obr. 3.9: Příklad Dirichletových vln s anomáliemi generovaných pomocí GutenTAG

4 Ověřovací experimenty

V praktické části budou vybrány dostupné implementace metod uvedených v předchozích kapitolách pro detekci anomálií. Jednotlivé implementace byly pečlivě vybrány, aby vyhovovaly zadaným požadavkům. Cílem bylo najít adaptace, které umí pracovat s vícerozměrnými časovými řadami. Za tímto účelem bylo prohledáno několik desítek repositářů na portálu GitHub.

Tyto metody budou postupně testovány na jednoduchých datasetech z generátoru GutenTAG. Tím bude ověřeno, že algoritmus dokáže rozpoznat základní anomálie v jednoduchém signálu. Dále bude algoritmus otestován na datasetu Pump sensor data, který představuje vícerozměrnou časovou řadu reálných dat z průmyslové aplikace. Základní informace k datasetu jsou uvedeny v sekci 3.3.1.

4.1 Metoda Long Short Term Memory

Implementace metody LSTM použitá v této práci vychází z implementací získaných z repositářů Air-Pollution-Forecasting [52] a Timeseries-anomaly-detection-using-LSTM [53] na portálu GitHub.

V prvním repositáři se nachází implementace algoritmu pro predikci dalšího datového bodu jedné časové řady na základě několika předchozích hodnot všech časových řad datasetu pomocí neuronové sítě s využitím LSTM buněk. V druhém repositáři je implementována jedna z možných metod detekce anomálií, která je založena na absolutní chybě rekonstrukce predikovaného bodu. Předpokladem této metody je, že LSTM síť dokáže podstatně lépe predikovat hodnoty normálního průběhu signálu než anomálie. Důvodem je, že tato neuronová síť se učí pouze na normálních datech.

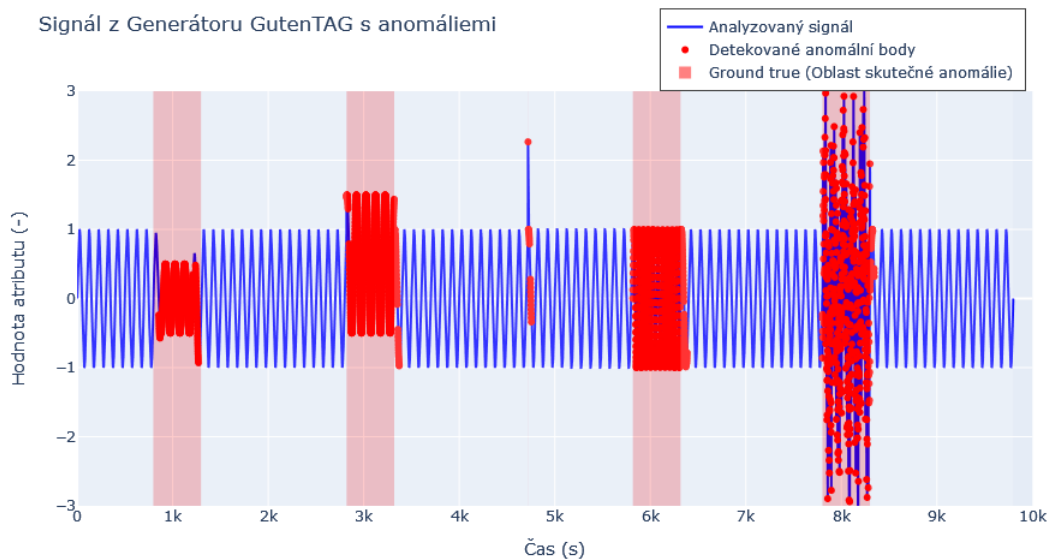
Díky algoritmu, který vznikl, dokáže detekovat anomálie v každé časové řadě zvlášť. Proto tato implementace není vhodná pro prediktivní údržbu, kde se předpokládá, že data budou vícerozměrná.

Algoritmus implementovaný v repositáři [52] má oproti algoritmu zmíněném ve 2. kapitole nevýhodu v podobě nutnosti volby velikosti okna. Učení probíhá tak, že dataset je převeden na problém učení s učitelem. To znamená, že se původní dataset rozdělí na úsek o délce zvoleného okna a další datový bod v řadě, který očekáváme na výstupu neuronové sítě.

Jak už bylo uvedeno dříve, délka okna patří mezi nejdůležitější hyperparametry, který přímo určuje nejen výpočetní náročnost ale hlavně schopnost sítě určit správně anomálie v signálu.

4.1.1 Pokusy s datasetem z generátoru GutenTAG

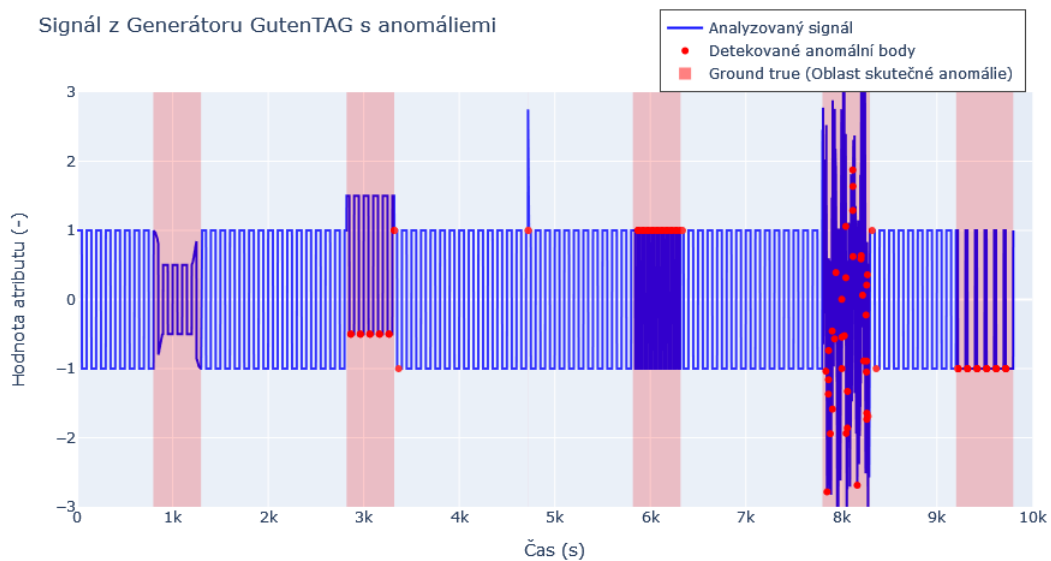
Nejdříve byl algoritmus otestován pomocí jednoduchých datasetů z generátoru GutenTAG. Jednotlivé testovací signály jsou popsány podrobně v sekci 3.5. Výsledky jednotlivých testovacích signálů jsou vidět na obrázcích 4.1, 4.2 a 4.3.



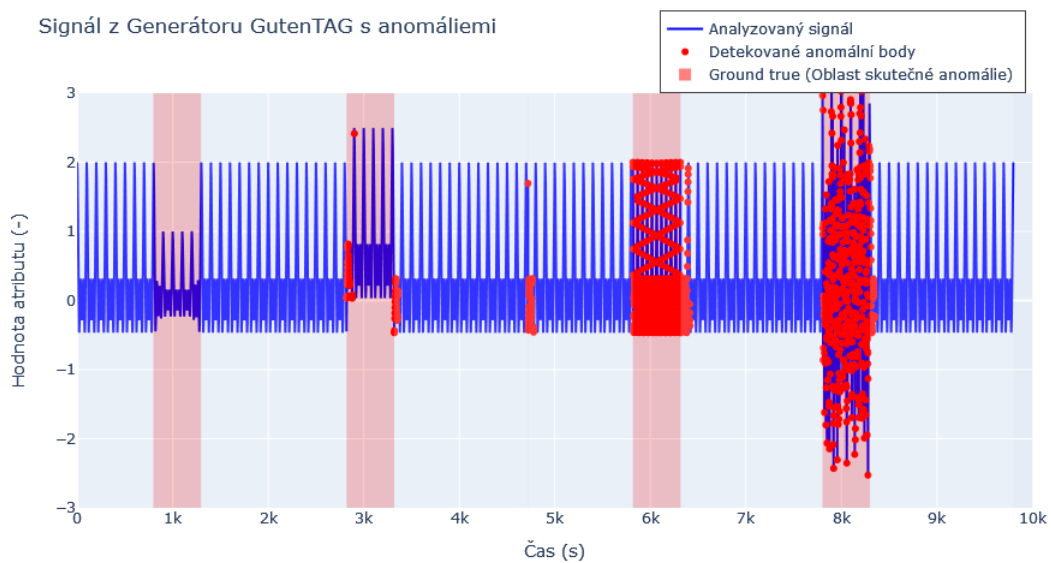
Obr. 4.1: Sinusový signál generovaný s pomocí GutenTAG s detekovanými anomáliemi pomoc LSTM

Z obrázků je vidět, že algoritmus poměrně spolehlivě detekuje anomální úseky v základních signálech. U některých průběhů algoritmus zaznamená pouze přechod do anomálie, což by nám ale v realitě mohlo stačit pro detekci anomálie. Bohužel algoritmus selhal při detekci anomálie "změna amplitudy" u 2 ze 3 signálů.

U neuronové sítě s použitím LSTM buněk byla tedy ověřena schopnost rozpoznat základní anomálie. Proto je možné přistoupit k testu na reálném datasetu.



Obr. 4.2: Obdélníkový signál generovaný s pomocí GutenTAG s detekovanými anomáliemi pomocí LSTM

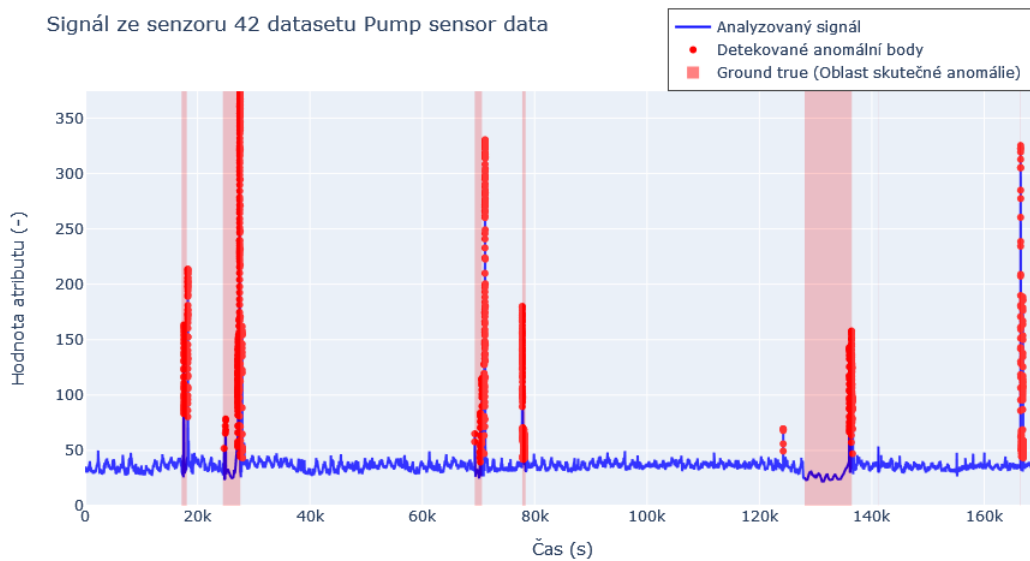


Obr. 4.3: Dirichletovy vlny generované s pomocí GutenTAG s detekovanými anomáliemi pomocí LSTM

4.1.2 Adaptace algoritmu pro dataset Dataset pump sensor

Jak už bylo výše uvedeno, neuronová síť s LSTM buňkami dokáže predikovat jednu časovou řadu na základě dat ze všech časových řad. Anomálie lze tedy detekovat pouze pro každý časový signál zvlášť.

Některé anomálie detekované v signálech datasetu velmi dobře korespondují se stavy BROKEN a RECOVERING. Ukázkou tohoto chování lze vidět například na obrázku 4.4. Anomálie detekované v některých signálech se stavy BROKEN a RECOVERING nekorespondují vůbec (např. obrázek 4.5). Na jiných signálech nebyly zaznamenány téměř žádné anomálie (obrázek 4.6).

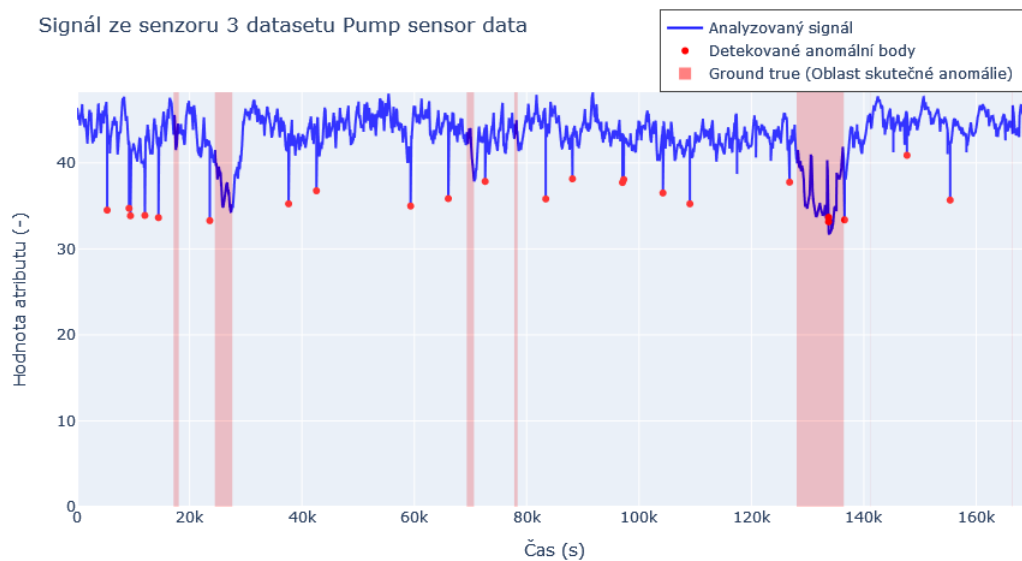


Obr. 4.4: Signál jednoho senzoru z datasetu Pump sensor data s detekovanými anomáliemi pomocí sítě LSTM odpovídajícími anomálním stavům zařízení

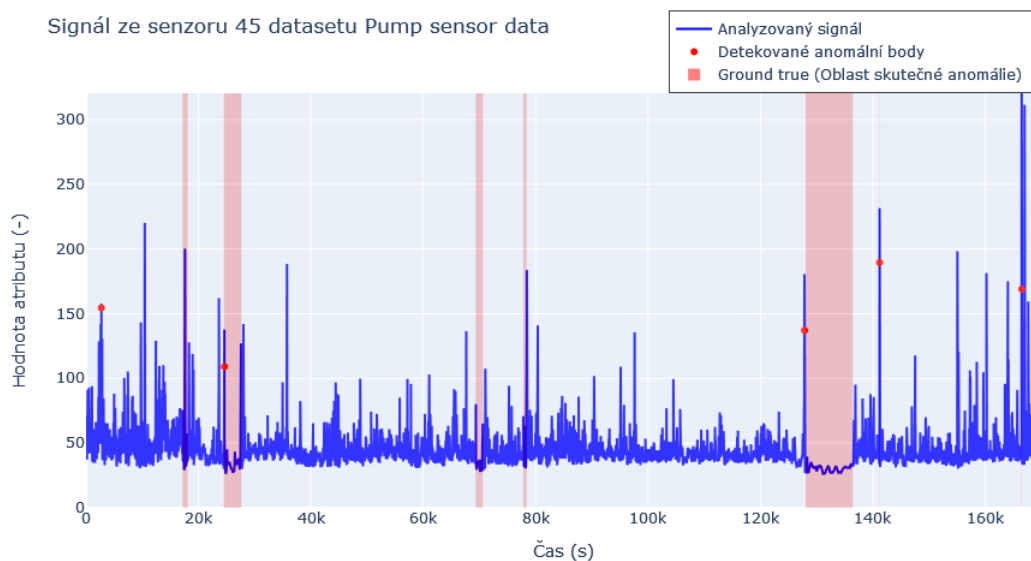
Otázkou je, zda na základě sledování anomálií v jednotlivých signálech lze stanovit anomální chování celého zařízení. Jednou z možností by bylo použití apriorní informace o veličině, kterou daný senzor sleduje a tím určit, jaké časové řady jsou pro detekci anomálie zařízení významné.

Dále by bylo možné pro detekci anomálií sledovaného zařízení dát dohromady informace o anomáliích v jednotlivých signálech a tím získat informaci o anomalitě vícerozměrného průběhu. Tento problém by mohlo vyřešit například tzv. rozhodování většiny (*angl. major voting*).

Tímto způsobem by pravděpodobně šlo pokrýt určité množství anomálií, ale značná část anomálií vychází právě ze vzájemného vztahu jednotlivých signálů. Na



Obr. 4.5: Signál jednoho senzoru z datasetu Pump sensor data s detekovanými anomáliemi pomocí sítě LSTM neodpovídajícím anomálním stavům zařízení



Obr. 4.6: Signál jednoho senzoru z datasetu Pump sensor data s velmi malým počtem detekovaných anomálií detekovaných pomocí sítě LSTM

základě konzultace s vedoucím této diplomové práce bylo upuštěno od metod spojující anomálie v jednotlivých signálech.

4.2 Prediktivní metoda s využitím CNN

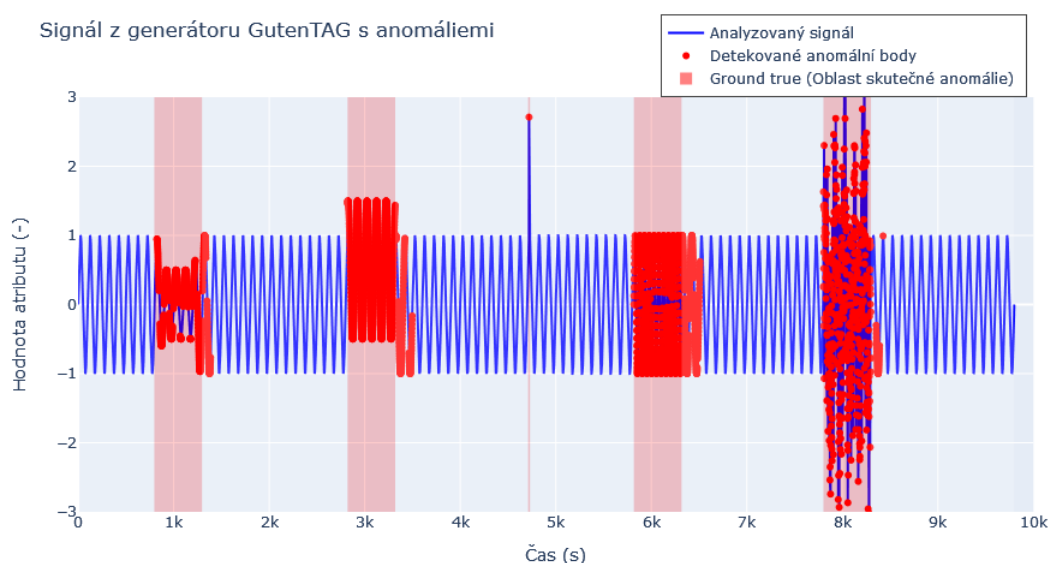
Tato metoda vychází z implementace detektoru anomálií z repozitáře `cnn-bilstm-attention` dostupného na portálu GitHub [54].

Princip metody detekce anomálií je velmi podobný předchozí síti LSTM. Základem je neuronová síť, jejímž cílem je naučit se predikovat další datový bod v časové řadě.

Na základě absolutní hodnoty chyby rekonstrukce je určena anomálnost datového bodu. Algoritmus pro určení anomálnosti datového bodu byl převzat z předchozího algoritmu. Jediným rozdílem oproti předchozímu algoritmu je tedy využití CNN místo LSTM.

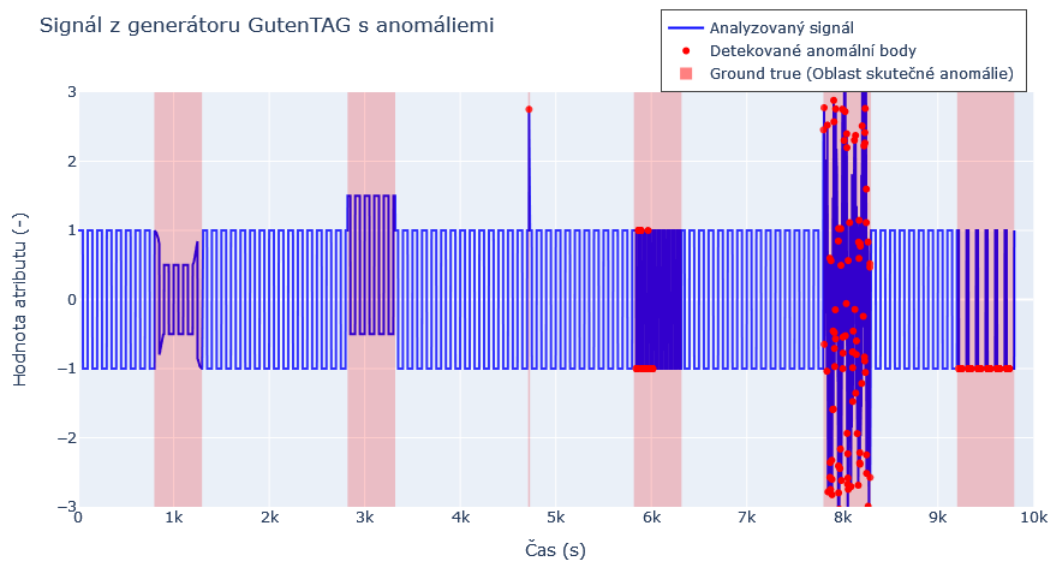
4.2.1 Pokusy s datasetem z generátoru GutenTAG

Nejdříve byl algoritmus otestován pomocí jednoduchých datasetů z generátoru GutenTAG. Jednotlivé stesovací signály jsou popsány podrobně v sekci 3.5. Výsledky detekce anomálií v jednotlivých testovacích signálech jsou vidět na obrázcích 4.7, 4.8 a 4.9.

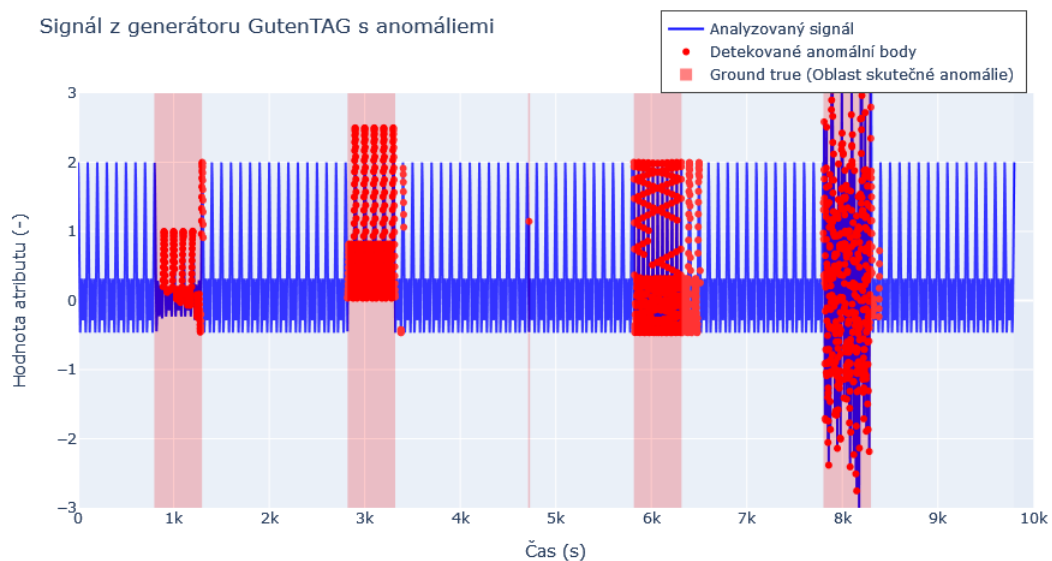


Obr. 4.7: Sinusový signál generovaný s pomocí GutenTAG s detekovanými anomáliemi pomocí CNN

Z obrázků je vidět, že tento algoritmus velice dobře pracuje s hladkými křivkami, tedy v našem případě s funkcí sinus a Dirichletovou vlnkou, u kterých se mu podařilo odhalit všechny anomálie správně. Podstatně horší výsledky vykazuje



Obr. 4.8: Obdélníkový signál generovaný s pomocí GutenTAG s detekovanými anomáliemi pomoc CNN



Obr. 4.9: Dirichletovy vlny generované s pomocí GutenTAG s detekovanými anomáliemi pomocí CNN

u obdélníkových pulzů, kde nedokázal detekovat ani změnu amplitudy ani změnu offsetu.

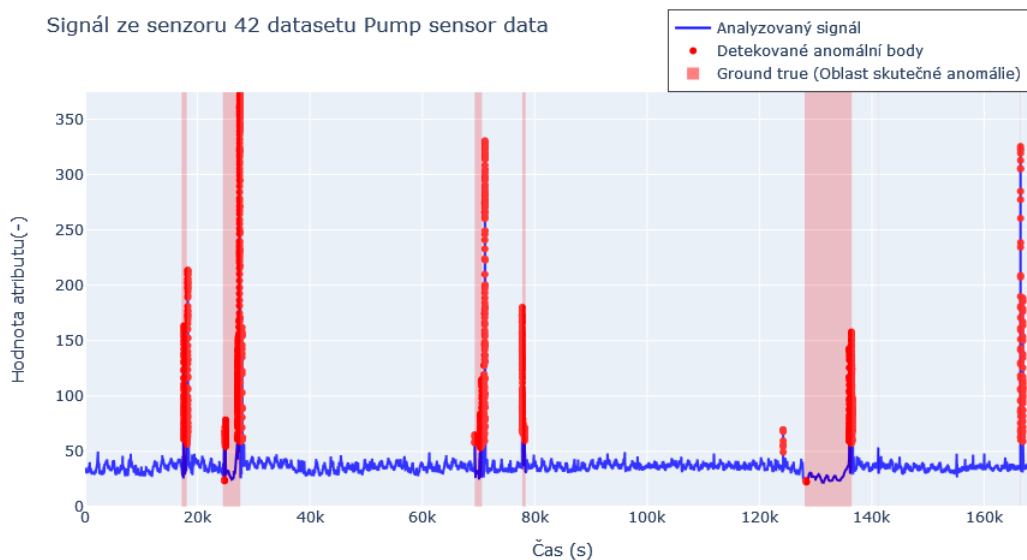
Z výsledků na jednoduchém datasetu je vidět, že i tento algoritmus strojového učení dokáže poměrně dobře rozpoznávat základní anomálie v signálech.

4.2.2 Adaptační algoritmu pro dataset Pump sensor data

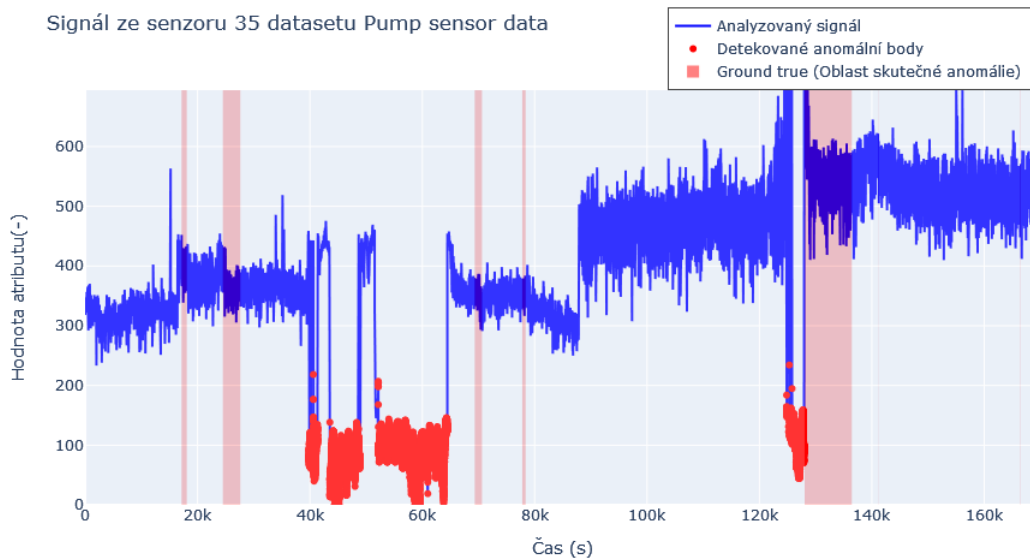
Jak už bylo uvedeno výše, tento algoritmus je velmi podobný algoritmu přechodnému. Liší se pouze jeho vnitřní implementací, která obsahuje kromě LSTM buněk také konvoluční vrstvu.

Tento algoritmus také dokáže detekovat anomálie pouze v jednom signálu na základě všech hodnot předchozích signálů. Vzhledem k problémům s agregací informací o anomáliích popsaných v sekci 4.1.2 je tento algoritmus méně vhodný pro naše účely.

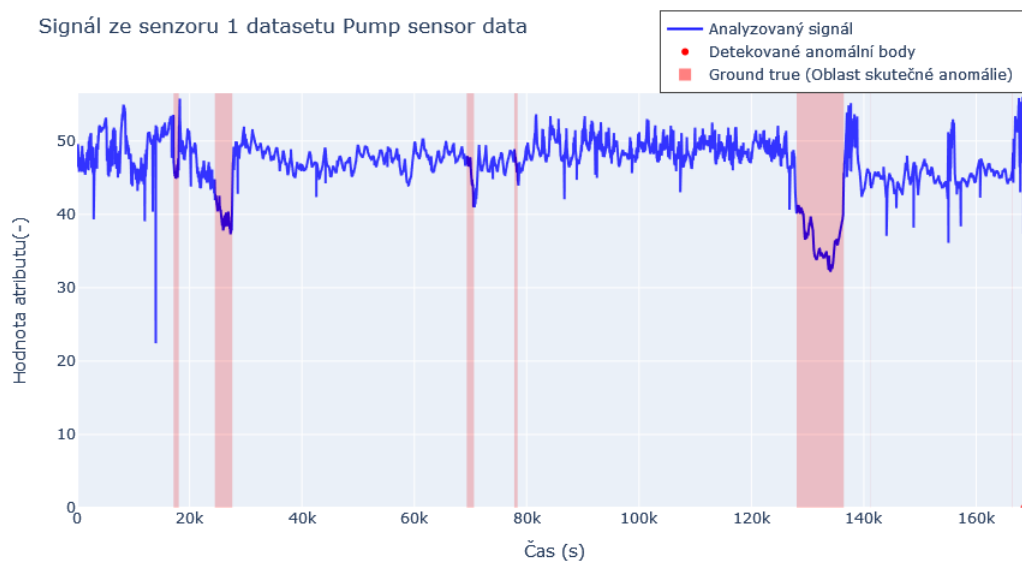
Výsledky u jednotlivých signálů jsou velmi podobné předchozímu algoritmu. Detekce anomálií u signálů některých senzorů (jako např. 4.10) velmi dobře koresponduje s anomálií stavu sledovaného systému. Některé s ním nekorespondují vůbec (obr. 4.11) a některé signály vůbec nevykazují anomální chování (obr. 4.12).



Obr. 4.10: Signál jednoho senzoru z datasetu Pump sensor data s detekovanými anomáliemi pomocí sítě s CNN odpovídajícím anomálním stavům zařízení



Obr. 4.11: Signál jednoho senzoru z datasetu Pump sensor data s detekovanými anomáliemi pomocí sítě s CNN neodpovídajícím anomálním stavům zařízení



Obr. 4.12: Signál jednoho senzoru z datasetu Pump sensor data s velmi malým počtem detekovaných anomálií detekovaných pomocí sítě s CNN

4.3 Algoritmy Random forest a Linear regression

Tyto dva algoritmy strojového učení jsou uvedeny ve stejné kapitoly, protože jejich implementace se nachází ve stejném repozitáři na portálu GitHub [55].

4.3.1 Pokusy s datasetem z generátoru GutenTAG

Algoritmus Random forest a Linear regression v této implementaci reagují na hodnoty signálů v samostatných časových úsecích. Proto jejich použití u jednorozměrného signálu postrádá smysl. Adaptace by šla provést tak, že by se nahradil vícerozměrný vstupní vektor časovým úsekem vstupního signálu o definované délce. Tím by se ale změnila podstata úlohy a proto tato adaptace nebude implementována.

4.3.2 Adaptace algoritmu pro dataset Dataset pump sensor

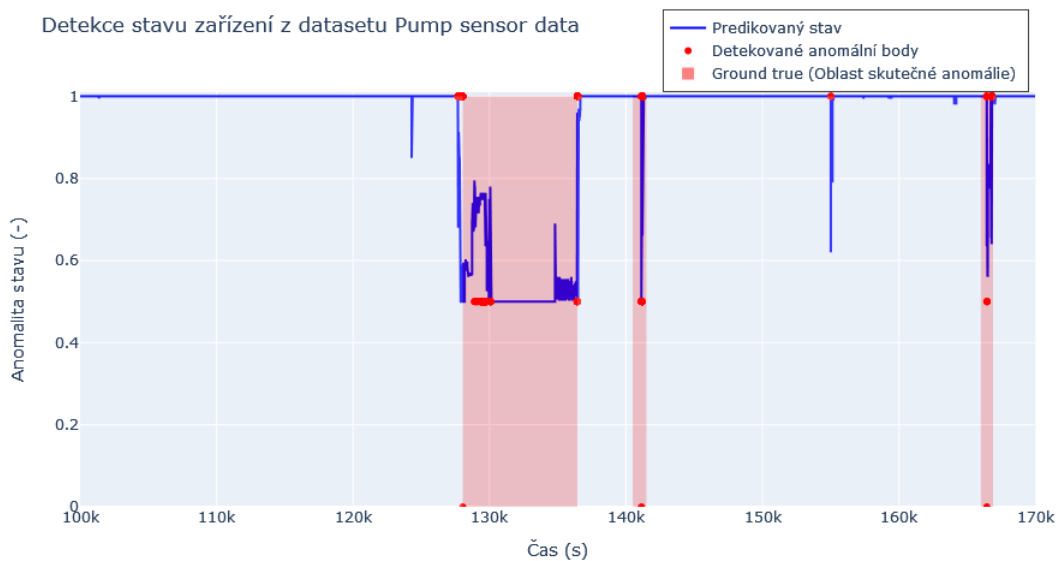
Autor repozitáře [55] testuje výše uvedené algoritmy strojového učení na datasetu Pump sensor data, který je použit i v této práci. Adaptace těchto algoritmů na dataset Pump sensor data tedy není vlastní prací autora této diplomové práce, ale je zde uvedena pouze pro doplnění práce o další metody. Pouze výstupní grafy jsou upravené, aby vyhovovali grafickému stylu této diplomové práce.

Oba algoritmy jsou navrženy jako jednoduchý problém učení s učitelem, kdy vstupem jsou aktuální hodnoty signálů (popř. posunuté o definovaný čas) všech senzorů a výstupem algoritmu je přímo informace o stavu zařízení reprezentovaná hodnotou výstupu, kdy 0 je stav BROKEN, 0.5 je stav RECOVERING a 1 je normální provozní stav bez anomálií.

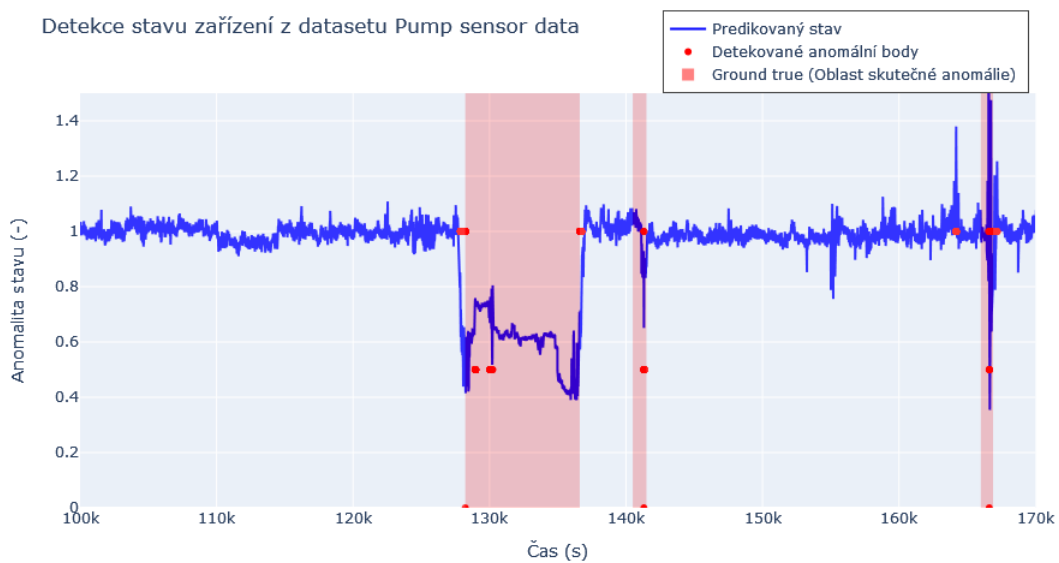
Dataset je rozdělen na trénovací a testovací data trochu jiným způsobem než u předchozích algoritmů, které se učily výhradně na datech bez anomálií. Algoritmy v tomto repozitáři se trénují na trénovacích i testovacích datech.

Tvůrce algoritmu prezentuje, že jeho algoritmus je vhodný nejen pro zhodnocení aktuálního stavu zařízení, ale i předpovědi jeho stavu v budoucnosti. Konkrétně pro předpověď 180, 1440, 2880 a 5760 minut dopředu. Autor uvádí, že jeho algoritmus se zdá být stejně spolehlivý jak pro detekci aktuálního stavu, tak pro dlouhodobou předpověď na zmíněném datasetu (až 4 dny dopředu). Dále také uvádí, že si nemůže být jistý, jestli tyto výsledky svědčí o spolehlivosti tohoto algoritmu.

Výsledky algoritmu Random forest jsou uvedeny na obrázcích 4.13. Výsledky algoritmu Linear regression jsou uvedeny na obrázcích 4.14.



Obr. 4.13: Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Random forest



Obr. 4.14: Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Linear regression

4.4 Isolation Forest

Implementace metody Isolation forest, která byla testována v této práci, pochází z repozitáře Credit-Card-Fraud-Detection-ML-Model umístěnému na portálu GitHub [56].

Při implementaci této metody byla využita implementace modelu Isolation Forest z knihovny Scikit-learn. Jedná se o učení bez učitele, kdy vstupem jsou aktuální hodnoty signálu. Nevýhoda použité implementace je, že nijak nereflkuje vstupní data jako časovou posloupnost. V každém časovém intervalu je anomálie dat hodnocena samostatně bez ohledu na předchozí stav senzorů. Anomálie nabývá pouze dvou hodnot. Pokud je stav detekovaný jako anomální, tak má výstup hodnotu 1, jinak je výstup algoritmu 0.

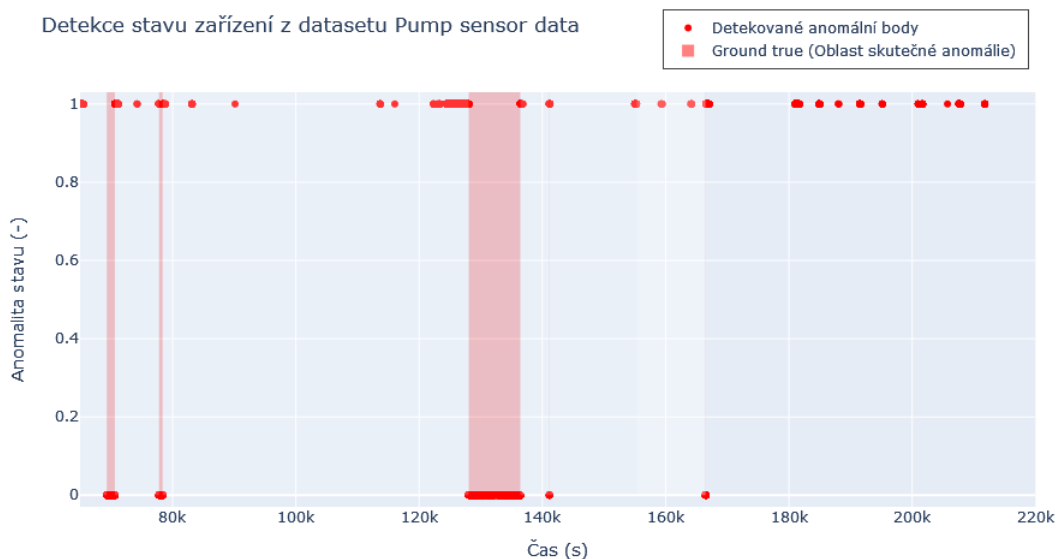
4.4.1 Pokusy s datasetem z generátoru GutenTAG

Vzhledem k tomu, že se stejně jako v předchozí sekci jedná o algoritmus, který reflektuje samostatně jednotlivé časové body, nemá smysl ho použít pro analýzu jednorozměrného signálu. Stejně jako v předchozím případě by šlo použít adaptaci, kde by se vstupní vícerozměrný vektor nahradil časovým úsekem jednorozměrného signálu, což by ale změnilo podstatu úlohy detekce anomálií. Proto tato adaptace nebude realizována.

4.4.2 Adaptace algoritmu pro dataset Dataset pump sensor

Při adaptaci metody pro dataset Pump sensor data bylo vyzkoušeno velké množství poměrů trénovacích a testovacích dat. Nejlepší výsledky algoritmus vykazoval při využití prvních 50 000 vzorků pro trénování a zbylá data pro testování. Výsledky výkonu algoritmu na testovacích datech jsou zobrazeny na obrázku 4.15. Červené pozadí grafu ukazuje skutečné anomální stavy zařízení. Červené tečky ukazují detekované anomální stavy. Pokud má anomálie hodnotu 1, tak se jedná o falešně pozitivní detekci. Pokud má anomálie hodnotu 0, tak se jedná o správně detekované anomálie.

Z obrázku 4.15 je vidět, že algoritmus velmi dobře detekoval všechny anomálie. Bohužel také detekoval relativně velké množství normálních stavů jako anomální. Při nasazení takového algoritmu do provozu by bylo potřeba zvážit, jak velký problém by představovaly falešně detekované anomálie.



Obr. 4.15: Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Isolation Forest

4.5 Local outlier factor

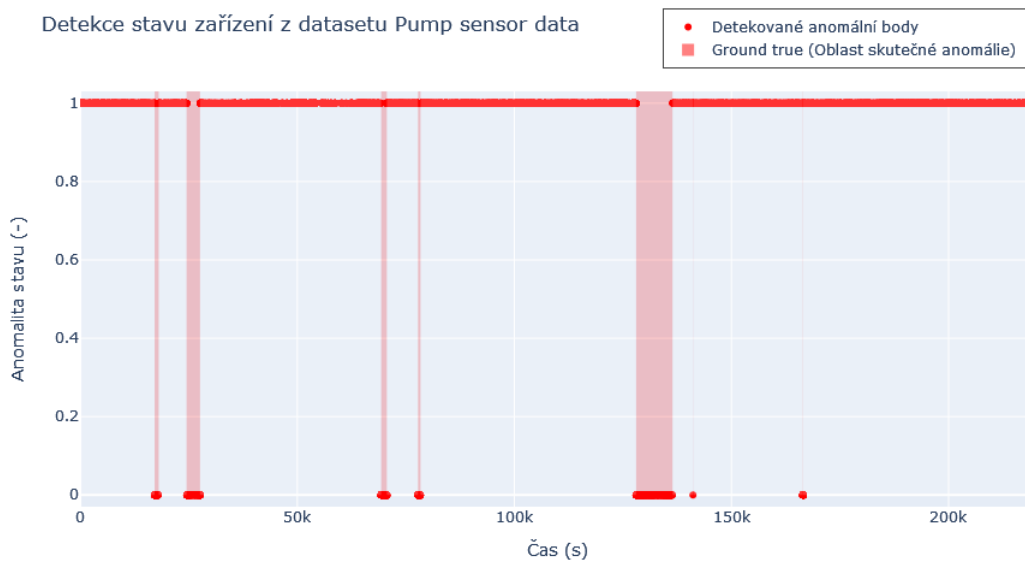
Implementace metody Local outlier factor, která je použita v této diplomové práci, pochází stejně jako výše zmíněná metoda Isolation Forest z repozitáře Credit-Card-Fraud-Detection-ML-Model dostupného na portálu GitHub [56].

Autor repozitáře [56] použil pro implementaci třídu LocalOutlierFactor z knihovny Scikit-learn. Pro učení a trénování byl použit celý dataset Pump sensor data. Vzhledem k tomu, že se jedná o metodu učení bez učitele, tak informace o anomalitě stavu zařízení byla použita až při vyhodnocení použitelnosti tohoto algoritmu. Nevýhodou tohoto algoritmu je stejně jako u metody Isolation Forest neschopnost pracovat s časovými řadami. To znamená, že stejně jako v předchozím případě posuzuje každý časový okamžik samostatně bez závislosti na předchozích hodnotách a stavech senzorů.

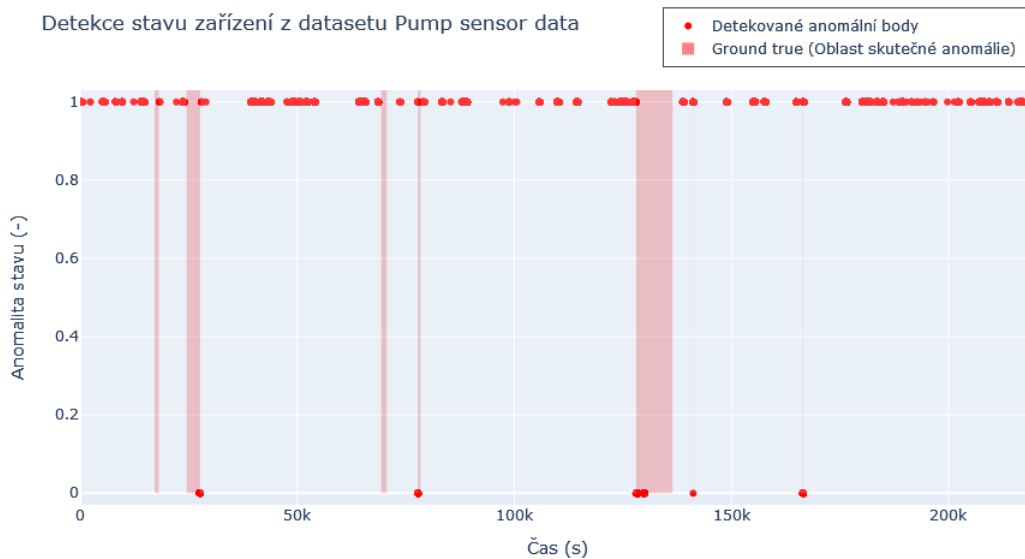
Výstupy algoritmu jsou 0 pro normální stav a 1 pro anomální stav. Proto ani nejsou hodnoty anomaly zobrazeny ve výstupních grafech.

Při testování tohoto algoritmu byl nastavován parametr počet nejbližších sousedů (*angl. number of local neighbours, ve funkci $n_neighbours$*). Toto číslo ovlivňuje, na kolik částí bude rozdělen stavový prostor, ve kterém hledáme anomálie. Výběr výsledků, který je uveden na obrázcích 4.16 4.17 4.18, ukazuje ovlivnění výsledků tímto parametrem.

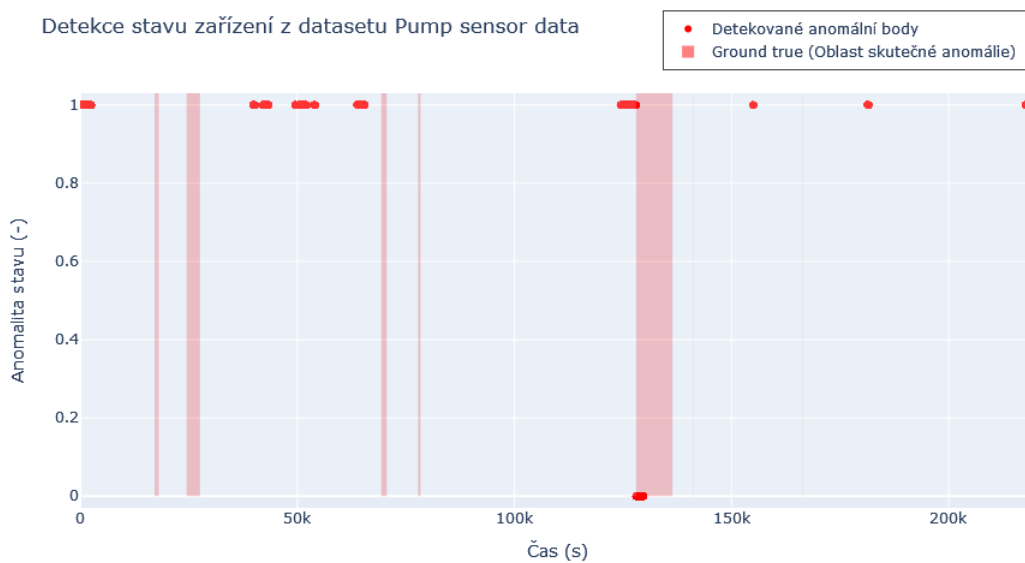
Z obrázků 4.16 4.17 4.18 je patrné, že pokud je číslo parametru $n_neighbours$ relativně malé, tak má model algoritmu LOF tendenci klasifikovat velkou část signálu jako anomální. Při zvětšování hodnoty parametru dochází k poklesu počtu falešně klasifikovaných anomálií, ale také klesá počet korektně klasifikovaných anomálií. Od určité hodnoty parametru začíná systém některé anomálie úplně ignorovat. Při hodnotě 5 000 zůstává většina anomálií nerozpoznána, ale některé normální části provozu zařízení jsou klasifikovány jako anomální. Větší hodnotu parametru nebylo možné vyzkoušet, vzhledem k velmi vysoké paměťové náročnosti metody.



Obr. 4.16: Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Local outlier factor s počtem nejbližších sousedů nastaveným na 100



Obr. 4.17: Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Local outlier factor s počtem nejbližších sousedů nastaveným na 1000



Obr. 4.18: Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí algoritmu Local outlier factor s počtem nejbližších sousedů nastaveným na 5000

4.6 Autoenkodér

Autoenkodér s libovolnou implementací je velmi intuitivní metoda pro detekci anomálií. Její princip je popsán v sekci 2.2.1. Implementace použitá v této diplomové práci vychází z implementace umístěné v repozitáři Sequential Anomaly detection na portálu GitHub [57]. Tato metoda bude v několika implementacích rozebrána podrobněji v další kapitole.

5 Implementace vybrané metody

Tato kapitola bude věnována zevrubnému zhodnocení dvou algoritmů založených na autoenkodérech pro detekci anomálií v multidimenzionálním signálu. Algoritmus bude testován na datasetu Pump sensor data, který je blíže představen v sekci 3.3.1. Nejprve budou představeny vybrané algoritmy a poté následně budou zhodnoceny jejich vlastnosti pomocí matice záměn a z ní odvozených výkonostních metrik.

5.1 Představení algoritmů

Algoritmy použité v této části vychází z repozitáře Sequential Anomaly detection z porálu GitHub [57]. V tomto repozitáři jsou uvedeny tři algoritmy pro detekci anomálií založené na autoenkodérech. V této práci budou testovány dva z nich.

První z nich je sestaven ze vstupní vrstvy, čtyř plně propojených skrytých vrstev (*angl. Dense*) s aktivační funkcí 'relu'. Na výstupu takto vzniklého autoenkodéru je opět plně propojená vrstva, tentokrát s aktivační funkcí sigmoida. Mezi jednotlivými plně propojenými vrstvami je umístěna vrstva BatchNormalization, která má za úkol udržet střední hodnotu signálu blízko 0 a směrodatnou odchylku blízko 1.

Počet neuronů jednotlivých plně propojených vrstev je odvozen od velikosti vstupního vektoru. První a druhá skrytá vrstva má vždy polovinu neuronů než vrstva předchozí. Čtvrtá vrstva má stejný počet neuronů jako vrstva třetí. Pátá a výstupní vrstva má zase dvojnásobný počet neuronů než předchozí vrstva, takže výstupní vrstva má stejný počet neuronů jako vrstva vstupní.

Tento autoenkodér bude dále v práci zmiňován jako autoenkodér 1.

Shrnutí architektury prvního autoenkodéru uzpůsobené pro dataset Pump sensor data můžete vidět na obrázku 5.1.

Druhý autoenkodér je sestaven ze vstupní vrstvy, čtyř skrytých vrstev, které jsou v tomto případě typu LSTM. Repeat vector vrstva funguje jako opakovač svého vstupu, který zopakuje n-krát. V tomto případě je počet opakování nastaven na počet vstupních parametrů (kanálů signálu). Na výstupu je stejně jako u předchozího modelu plně propojená vrstva.

Tento autoenkodér bude dále v práci zmiňován jako autoenkodér 2.

Shrnutí architektury druhého autoenkodéru uzpůsobené pro dataset Pump sensor data můžete vidět na obrázku 5.2.

Model: "model"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 48)]	0
dense (Dense)	(None, 32)	1568
batch_normalization (Batch Normalization)	(None, 32)	128
dense_1 (Dense)	(None, 16)	528
batch_normalization_1 (Batch Normalization)	(None, 16)	64
dense_2 (Dense)	(None, 16)	272
batch_normalization_2 (Batch Normalization)	(None, 16)	64
dense_3 (Dense)	(None, 32)	544
batch_normalization_3 (Batch Normalization)	(None, 32)	128
dense_4 (Dense)	(None, 48)	1584

=====
Total params: 4,880
Trainable params: 4,688
Non-trainable params: 192
=====

Obr. 5.1: Výpis architektury autoenkodéru pro detekci anomálií s použitím plně propojených vrstev neuronové sítě

Model: "model_1"

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 48, 1)]	0
lstm (LSTM)	(None, 48, 24)	2496
lstm_1 (LSTM)	(None, 24)	4704
repeat_vector (RepeatVector)	(None, 48, 24)	0
lstm_2 (LSTM)	(None, 48, 24)	4704
lstm_3 (LSTM)	(None, 48, 24)	4704
time_distributed (TimeDistrib uted)	(None, 48, 1)	25

=====
Total params: 16,633
Trainable params: 16,633
Non-trainable params: 0
=====

Obr. 5.2: Výpis architektury autoenkodéru pro detekci anomálií s použitím LSTM vrstev neuronové sítě

5.2 Hyperparametry autoenkodérů

V této sekci budou v krátkosti zmíněny hyperparametry, které mohou ovlivnit výkonost sítí pro detekci anomálií.

Na začátek budou zmíněny hyperparametry modelu, který reprezentuje neuronovou síť. Jedním z hlavních parametrů je tzv. optimizér, což je algoritmus pomocí kterého se aktualizují váhy modelu při učení. Jedním z nejběžnějších optimizérů je Adam, který je založen na metodě SGD (*Z angl. stochastic gradient descent*), který je rozšířen o odhad momentů prvního a druhého řádu. [58]. Mezi další optimizéry patří například již zmíněný SGD nebo RMSprop, Adamax, Adadelata a další.

Další hyperparametr navázaný přímo na model je ztrátová funkce (*angl. loss function*). Ztrátová funkce je měřítko, podle kterého se zjišťuje, jak moc se výstup modelu liší od požadované hodnoty, v případě učení s učitelem. (Učení autoenkodérů je případ učení s učitelem.) Na základě ztrátové funkce upravuje optimizér váhy modelu. Mezi ztrátové funkce patří například Průměr čtverců chyb (*angl. Mean square error*) nebo Binární křížová entropie (*angl. Binary cross-entropy*), ztrátová funkce Hindge a další.

V další části budou zmíněny parametry a funkce, které se týkají učení. Jako první z této části lze zmínit velikost várky (*angl. batch size*), což je parametr, který ovlivňuje, kolik vzorků projde učícím procesem než dojde k přepočítání vah modelu.

Jako další hyperparametr týkající se učení lze uvést počet epoch. Jedná se o parametr, který určuje, kolikrát projde celý trénovací dataset modelem než se prohlásí za naučený. Obecně lze říct, že do určité doby se zvětšováním počtu epoch klesá chyba daného modelu. Bohužel při neustálém zvětšování počtu epoch může dojít k takzvanému přeučení, což je stav, kdy se model adaptuje natolik na trénovací data, že klesá jeho výkonost na datech ostatních.

Aby nedošlo k přeučení, lze použít funkci objektu EarlyStopping, která sleduje nějaký parametr učení (například chybu na validačních datech) a pokud je splněna určitá podmínka (například tato hodnota neklesá, naopak s rostoucím počtem epoch mírně roste), tak dojde k přerušení učícího procesu.

Pokud je vyvolána podmínka EarlyStopping, znamená to, že model při učení v posledních několika krocích pravděpodobně zhoršil své vlastnosti. Abychom zachovali optimální parametry modelu, můžeme použít funkci objektu ModelCheckpoint, který ukládá na základě zvolených parametrů (například chyba na validačních datech) parametry nejlepšího modelu, které můžeme po ukončení učícího procesu znovu nahrát.

Anomálie se u autoenkodérů, jak je zmíněno v sekci 2.2.1, určují na základě správnosti rekonstrukce vstupního vektoru na výstupu. Čím více je rekonstrukce na výstupu podobná hodnotám na vstupu, tím je pravděpodobněji výsledek normální.

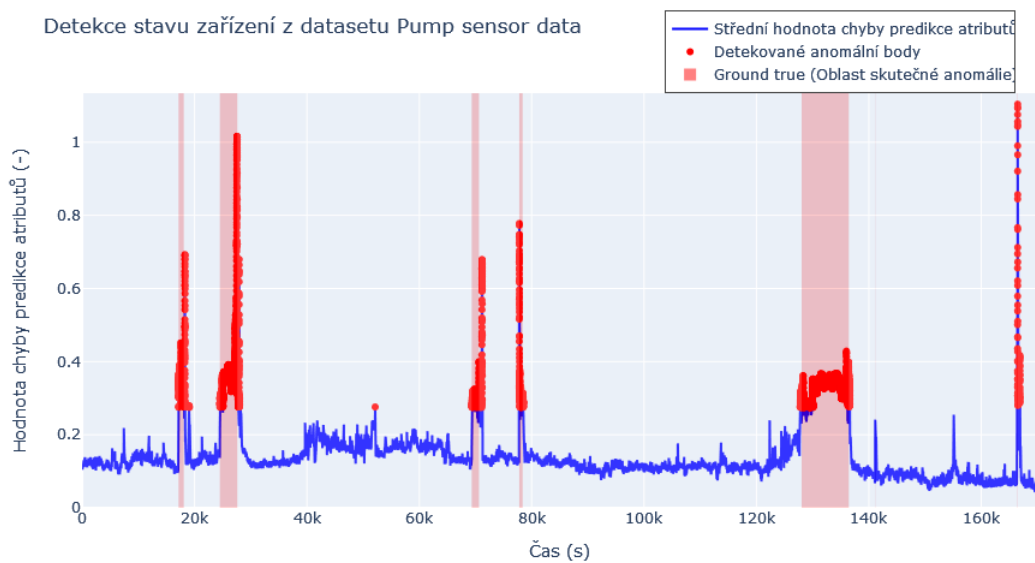
Chyba rekonstrukce se dá hodnotit pomocí tzv. ztrátové funkce, která byla zmíněná výše.

Důležité je určit prahovou hodnotu, kdy vzorek na vstupu označíme za anomální a kdy ještě ne. V této práci je použita metoda prahu, která vychází z maximální chyby rekonstrukce na normálních datech. Autoenkodér je model, který se učí pouze na normálních datech. Na konci učícího procesu stačí tedy použít maximum ztrátové funkce na trénovacím datasetu. Další možný přístup by byl použít dva testovací datasety, kdy jeden by sloužil k určení prahu a druhý k samotnému testování. K určení prahu by byla potřeba apriorní informace o výskytu anomálií v daném datasetu (kolik procent datasetu tvoří anomálie). Na základě toho by se určil percentil hodnot ztrátové funkce, který by odpovídal prahové hodnotě.

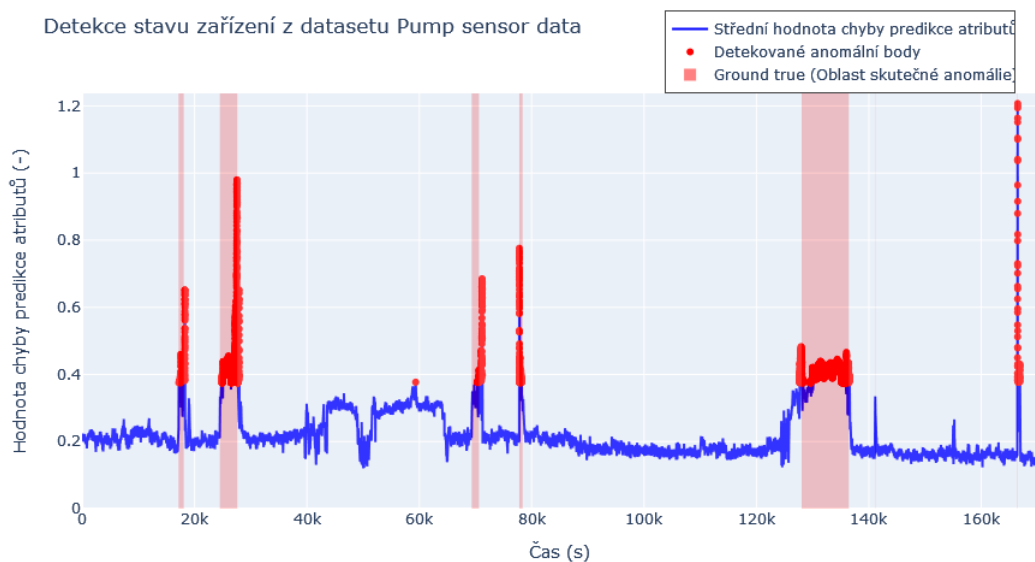
5.3 Pokusy se základním nastavením

Po představení algoritmu a jeho hyperparametrů lze přejít k testům jeho výkonosti a robustnosti. Jako první budou provedeny testy algoritmů pro základní nastavení hyperparametrů. Jako optimizér byl použit Adam. Jako ztrátová funkce byl použit Mean square error. Počet epoch je nastaven na 100, ale učení všech modelů skončilo maximálně vždy po několika desítkách iterací. Velikost várky byla nastavena na 10.

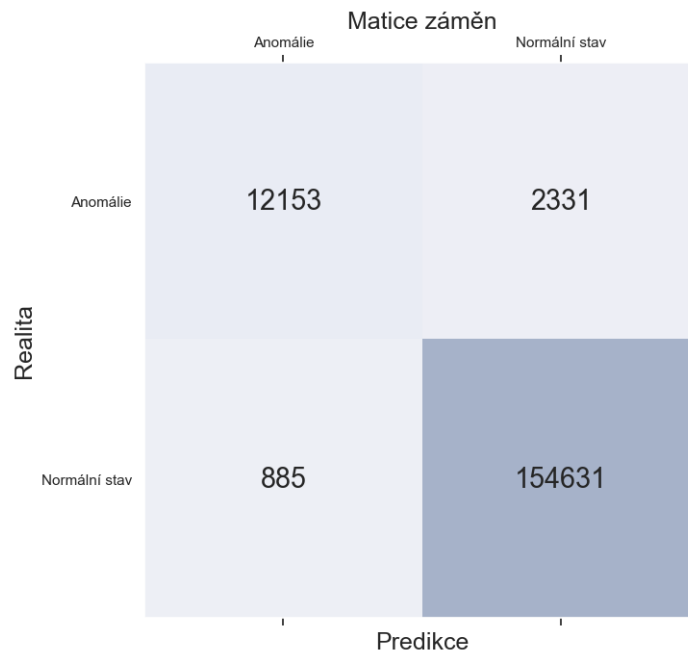
Výsledky detekčních algoritmů jsou zobrazeny na následujících obrázcích. Anomálie detekované v čase jsou zobrazeny na obrázcích 5.3 5.4. Vyhodnocení algoritmů podle matice záměn je zobrazeno na obrázcích 5.5 5.6 a vyhodnocení algoritmu podle různých metrik vycházejících z matice záměn je zobrazeno v tabulkách 5.1 5.2.



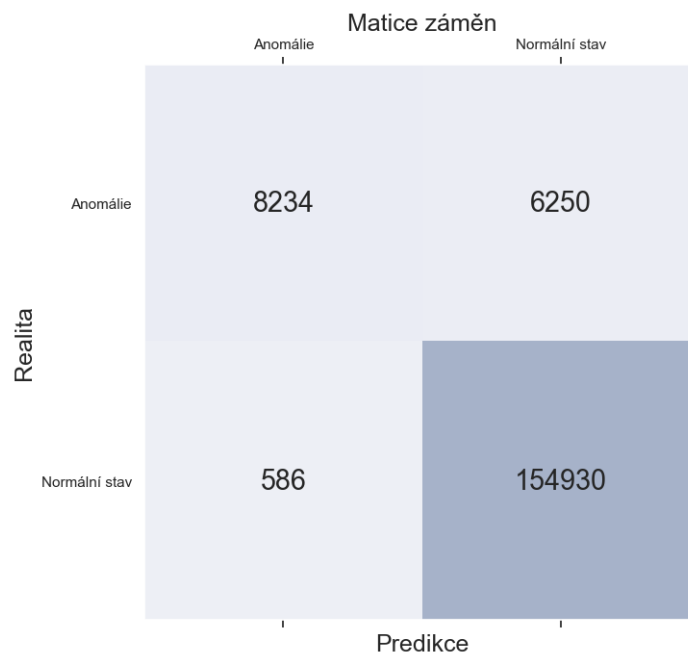
Obr. 5.3: Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí autoenkodéru založených na plně propojených vrstvách



Obr. 5.4: Detekce anomálního stavu zařízení z datasetu Pump sensor data pomocí autoenkodéru založených na LSTM vrstvách



Obr. 5.5: Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem založeným na plně propojených vrstvách neuronů



Obr. 5.6: Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem založeným na LSTM vrstvách

Tab. 5.1: Tabulka metrik detektoru anomálií datasetu Pump sensor data s autoenkodérem založeným na plně propojených vrstvách neuronů

Název metriky	Anglický název	Hodnota
Celková správnost	Accuracy	0.98108
Přesnost	Precision	0.93212
Senzitivita	Recall	0.83906
Specificita	Specificity	0.99431
F1 míra	F1 score	0.88314
AUC	AUC	0.91669

Tab. 5.2: Tabulka metrik detektoru anomálií datasetu Pump sensor data s autoenkodérem založeným na LSTM vrstvách

Název metriky	Anglický název	Hodnota
Celková správnost	Accuracy	0.95979
Přesnost	Precision	0.93356
Senzitivita	Recall	0.56849
Specificita	Specificity	0.99623
F1 míra	F1 score	0.70666
AUC	AUC	0.78236

5.4 Vliv změny hyperparametrů

Tato část bude věnována citlivosti modelu na změny hyperparametrů jako velikost várky, druh optimizéru a použitá ztrátová funkce při učení. Hodnoty hyperparametrů byly vybrány tak, aby odpovídaly řešenému problému. To znamená, že důraz bude kladen na optimizéry a ztrátové funkce vhodné pro učení generativního modelu, v našem případě autoenkodéru.

K testům budou uvedeny výkonostní metriky vycházející z matice záměn. Vzhledem k tomu, že se jedná o nevyvážený dataset, budou pro testy směrodatné metriky F1 míra a AUC. Platí, že se snažíme dosáhnout co nejvyšší hodnoty těchto parametrů. Anomálie v datasetu jsou pro účely této práce považovány za pozitivní třídu.

Každý pokus byl proveden pouze jednou a proto je zde přípustná náhodná odchylka od trendu, který na datech pozorujeme.

5.4.1 Vliv velikosti dávky (Batch size)

Jak už bylo zmíněno výše, parametr Velikost dávky (*angl. Batch size*) ovlivňuje, jak často dochází k úpravě vah trénovaného modelu. Tím se značně ovlivňuje nejen rychlost učení, ale taky výsledné váhy modelu.

V této části bude popsán vliv hodnoty tohoto parametru na výsledky trénovaného modelu. Hodnoty parametru byly vybrány 1, 10, 50, 100, 500, 1000, 5000, 10000, 20000, 50000.

Obecně lze říct, že s rostoucí velikostí dávky klesá čas potřebný pro naučení sítě při stejném výpočetním výkonu. Důvodem je menší počet operací změny vah, ke kterým během trénování dojde.

Z metrik uvedených v tabulce 5.3 lze vidět, že velikost dávky nijak výrazně výkonost autoenkodéru 2 při detekci anomálií neovlivňuje. K ovlivnění dochází až ve chvíli, kdy velikost dávky začíná být srovnatelná s velikostí datasetu. Příliš velká velikost dávky tedy ovlivňuje proces trénování sítě negativně. U autoenkodéru 1 bylo ovlivnění výraznější, ale až na velikost dávky 1000 a 5000, kdy se model skoro nenaučil, není toto ovlivnění významné. Pro autoenkodér 1 byla optimální velikost dávky 10 a pro autoenkodér 2 byla optimální hodnota tohoto parametru 500.

Tab. 5.3: Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data pro různé velikosti dávky

	Autoenkodér 1 (nn)		Autoenkodér 2 (LSTM)	
Velikost dávky	AUC	F1 míra	AUC	F1 míra
1	0.80371	0.45644	0.90991	0.51026
10	0.94821	0.64862	0.92765	0.56677
50	0.91541	0.52786	0.95802	0.70513
100	0.91634	0.53017	0.96090	0.86152
500	0.82067	0.46891	0.99028	0.93441
1000	0.63258	0.29725	0.98082	0.83912
5000	0.51322	0.05319	0.94660	0.91529
10000	0.79247	0.44689	0.91619	0.87789
20000	0.77769	0.43261	0.89057	0.77791
50000	0.82062	0.47294	0.64369	0.30391

Podrobnější výsledky všech experimentů lze najít v tabulkách B.1 B.2 B.3 B.4 B.5 B.6 B.8 B.9 B.10. Na obrázku B.1 je vidět matice záměn pro enkodér 1 trénovaný s velikostí dávky 10. Na obrázku B.2 je vidět časový průběh detekce anomálií pro odpovídající případ. Na obrázku B.3 je vidět matice záměn pro enkodér 2 trénovaný

s velikostí dávky 500. Na obrázku B.4 je vidět časový průběh detekce anomálií pro odpovídající případ.

5.4.2 Vliv použitého optimizéru

Jak už bylo zmíněno výše, optimizér je algoritmus, podle kterého se nastavují váhy na základě hodnoty ztrátové funkce. V této části bude testován vliv výběru optimizéru na výsledek učícího procesu.

Mezi testované optimizéry byly vybrány (Algoritmy budou uváděny pod jejich anglickými názvy, aby nedošlo k nechtěné záměně) Stochastic gradient descent jako základní optimalizační algoritmus, Adam jako často používaný optimizér, AdaGrad, AdaDelta a Adamax jako jeho varianty.

Podle metrik uvedených v tabulkách 5.4 můžeme říct, že nejlepších výsledků dosáhly enkodéry trénované pomocí optimizéru AdaMax. Ostatní algoritmy založené na algoritmu Adam dosáhli velmi podobných výsledků. Nejhorší výsledky měl model trénovaný s optimizérem Stochastic gradient descent.

Tab. 5.4: Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data pro různé optimizéry

	Autoenkodér 1 (nn)		Autoenkodér 2 (LSTM)	
Optimizér	AUC	F1 míra	AUC	F1 míra
AdaMax	0.962857	0.72241	0.98772	0.90538
Adam	0.91571	0.52821	0.95166	0.67006
Adagrad	0.91750	0.53714	0.81063	0.51681
Adadelta	0.85178	0.50140	0.80369	0.51216
SGD	0.81193	0.46614	0.82460	0.48308

Podrobnější výsledky všech experimentů lze najít v tabulkách B.11 B.12 B.13 B.14 B.15. Na obrázku B.5 je vidět matice záměn pro autoenkodér 1 trénovaný s optimizérem AdaMax. Na obrázku B.6 je vidět časový průběh detekce anomálií pro odpovídající případ. Na obrázku B.7 je vidět matice záměn pro autoenkodér 2 trénovaný s optimizérem AdaMax. Na obrázku B.8 je vidět časový průběh detekce anomálií pro odpovídající případ.

5.4.3 Vliv použité ztrátové funkce (Loss function)

Jak už bylo zmíněno v předchozí části, ztrátová funkce je algoritmus, který počítá chybu na výstupu algoritmu strojového učení při problému učení z učitelem. Výsledek tohoto algoritmu je poté použit jako vstup algoritmu optimizéru.

Mezi testované ztrátové byly vybrány tyto varianty: (Algoritmy budou uváděny pod jejich anglickými názvy, aby nedošlo k nechtěné záměně) Mean squared error (MSE) jako jedna z univerzálních ztrátových funkcí, Mean absolute error (MAE) jako obdoba MSE citlivější na anomálie, KL-divergence jako ztrátová funkce vhodná pro generativní sítě jako jsou například autoenkodéry a Hubert loss, která je kombinací MSE a MAE.

Podle metrik uvedených v tabulce 5.5 lze říct, že nejlepších výsledků dosáhl autoenkodér 1 trénovaný s použitím funkce MSE a autoenkodér 2 s použitím ztrátové funkce MAE. Nejhorších výsledků dosáhly modely trénované s použitím ztrátové funkce KL-divergence.

Tab. 5.5: Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data pro různé ztrátové funkce

	Autoenkodér 1 (nn)		Autoenkodér 2 (LSTM)	
Ztrátová funkce	AUC	F1 míra	AUC	F1 míra
MSE	0.92287	0.56401	0.94631	0.64844
MAE	0.72639	0.38173	0.96466	0.91529
Huber loss	0.92420	0.55623	0.95264	0.67387
KL- divergence	0.74975	0.40316	0.71857	0.37203

Podrobnější výsledky všech experimentů lze najít v tabulkách B.16 B.17 B.18 B.19. Na obrázku B.9 je vidět matice záměn pro autoenkodér 1 trénovaný se ztrátovou funkcí MSE. Na obrázku B.10 je vidět časový průběh detekce anomálií pro odpovídající případ. Na obrázku B.11 je vidět matice záměn pro autoenkodér 2 trénovaný se ztrátovou funkcí MAE. Na obrázku B.12 je vidět časový průběh detekce anomálií pro odpovídající případ.

5.5 Citlivostní analýza

V této části bude zkoumána robustnost modelu detektoru anomálií založeném na autoenkodérech. Vstupní data budou určitým způsobem upravena resp. poškozena. Poté bude sledována úspěšnost algoritmu při detekci anomálií. Aby byly výsledky různých architektur autoenkodérů srovnatelné, budou všechny při učení používat optimizér Adam a ztrátovou funkci MSE. Tyto výsledky lze porovnat s výsledky na základním datasetu, které lze najít v sekci 5.3.

Předmětem citlivostní analýzy budou tři testy. První bude reakce detekčního algoritmu na proporcionální zesílení resp. zeslabení všech signálů definovanou konstantou. Druhý test bude zkoumat reakci detektoru na přidání šumu do signálu a

třetí test se bude věnovat reakce detektoru na filtrování signálu pomocí filtru posuvného průměru. V následujících sekcích budou jednotlivé testy blíže popsány spolu s jejich výsledky.

Každý pokus byl proveden pouze jednou a proto je zde přípustná náhodná odchylka od trendu, který na datech pozorujeme.

5.5.1 Ovlivnění proporcionalní konstantou

Při zkoumání citlivosti algoritmu detekce anomálií na vynásobení signálu proporcionalní konstantou byl vždy celý dataset vynásoben jednou konstantou v intervalu od 0.00001 do 10000. Následně byly autoenkodéry na daný dataset naučeny a otestovány.

Na základě výsledných metrik uvedených v tabulce 5.6 lze říct, že vynásobení konstantou vůbec výkonnost algoritmu neovlivnilo. Výsledky pro jednotlivé datasety byly srovnatelné. Necitlivost algoritmu na zesílení resp. zeslabení signálu lze vysvětlit např. normalizací signálu, která proběhne před začátkem učícího procesu.

Tab. 5.6: Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data vynásobená různými konstantami

	Autoenkodér 1 (nn)		Autoenkodér 2 (LSTM)	
Násobící konstanta	AUC	F1 míra	AUC	F1 míra
0.00001	0.92682	0.56184	0.93536	0.60059
0.5	0.92094	0.54217	0.96080	0.88090
1 (Originální data)	0.91669	0.88314	0.78236	0.70666
1.5	0.91911	0.54090	0.98824	0.91501
10000	0.83879	0.48525	0.96579	0.73983

Všechny výsledné metriky jsou uvedeny pro dataset vynásobený krajními konstantami tzn. 0.00001 C.1 a 10000 C.2. Pro autoenkodér 1 je uvedená matice záměn pro konstantu 0.00001 C.1 a 10000 C.2. Jsou také uvedeny odpovídající časové průběhy detekce anomálií C.5 a C.6. Pro autoenkodér 2 je uvedená matice záměn pro konstantu 0.00001 C.3 a 10000 C.4. Jsou také uvedeny odpovídající časové průběhy detekce anomálií C.7 a C.8.

5.5.2 Ovlivnění přídavným šumem

V tomto testu byla zkoumána citlivost algoritmu na přídavný šum. Při tvorbě zašuměného datasetu byl ke každému kanálu přičten bílý šum podle zvoleného odstupů

signál-šum (*angl. SNR - Signal to noise ratio*). Cílem testu bylo zjistit při jaké hladině šumu už algoritmus není schopen rozpoznávat anomálie. Odstup signál šum byl volen v hodnotách 1, 5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 dB. Při testování nebyly vzhledem k výpočetní náročnosti testovány všechny varianty, ale postupovalo se metodou půlení intervalu, za předpokladu, že menší odstup signál šum znamená horší schopnost detekce.

Z metrik uvedených v tabulce 5.7 lze poznat, že oba autoenkodéry jsou schopné rozeznávat anomálie do rušení bílým šumem o SNR do 20 dB. V datasetu s rušením o SNR 10 dB byl zaznamenán silný pokles výkonosti prvního autoenkodéru. Pokles u druhého autoenkodéru nebyl tak výrazný, ale v datasetu s rušením o SNR o velikosti 5 dB ztratily oba algoritmy schopnost detekce naprosto.

Tab. 5.7: Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data s přidáním šumem

	Autoenkodér 1 (nn)		Autoenkodér 2 (LSTM)	
SNR (dB)	AUC	F1 míra	AUC	F1 míra
50	0.95343	0.67302	0.94029	0.61992
20	0.92875	0.59884	0.91183	0.60089
10	0.62119	0.32608	0.80562	0.64301
5	0.50658	0.02762	0.51941	0.07691

Podrobnější výsledky provedených experimentů lze najít v tabulkách C.3 C.4 C.5 C.6. Na obrázku C.9 je zobrazena matice záměn pro pro autoenkodér 1 zpracovávající dataset zašuměný šumem o SNR 20 dB. Odpovídající časový průběh časové detekce anomálií je zobrazen na obrázku C.13. Na obrázku C.10 je zobrazena matice záměn pro pro autoenkodér 2 zpracovávající dataset zašuměný šumem o SNR 20 dB. Odpovídající časový průběh časové detekce anomálií je zobrazen na obrázku C.14. Na obrázku C.11 je zobrazena matice záměn pro pro autoenkodér 1 zpracovávající dataset zašuměný šumem o SNR 5 dB. Odpovídající časový průběh časové detekce anomálií je zobrazen na obrázku C.15. Na obrázku C.12 je zobrazena matice záměn pro pro autoenkodér 2 zpracovávající dataset zašuměný šumem o SNR 5 dB. Odpovídající časový průběh časové detekce anomálií je zobrazen na obrázku C.16.

5.5.3 Ovlivnění filtrací signálu

V tomto testu byla zkoumána citlivost detektoru anomálií na filtraci signálu pomocí filtru typu klouzavý průměr s konstantní délkou okna. Cílem bylo zjistit, jestli určitá

délka okna může významně zhoršit schopnost detekce anomálií v signálu. Délka okna byla volena v následujících hodnotách: 5, 10, 20, 50, 100.

Z výsledných metrik uvedených v tabulce 5.8 lze vypožorovat, že se zvětšujícím se oknem klesá výkonost autoenkodéru. Tento pokles není nijak zásadní. Velký problém pro detekci představuje nevhodně zvolený práh, který se projevuje přítomností velkého počtu falešně detekovaných anomálií (viz. C.19 C.20). Pokud bychom zvolili práh jiný, vhodnějším způsobem, tak by detekce anomálií byla stejně kvalitní jako u neupraveného datasetu. Filtrování signálu tedy neovlivňuje samotný autoenkodér, ale aktuální systém nastavování prahu.

Tab. 5.8: Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data filtrovaná plovoucím průměrem s různou délkou okna

Délka okna	Autoenkodér 1 (nn)		Autoenkodér 2 (LSTM)	
	AUC	F1 míra	AUC	F1 míra
100	0.70876	0.24230	0.84609	0.37701
50	0.69903	0.23632	0.74191	0.26517
20	0.85089	0.38448	0.90999	0.50855
10	0.91699	0.53062	0.919625	0.53732
5	0.73409	0.25940	0.90842	0.50601

Podrobnější výsledky provedených experimentů lze najít v tabulkách C.7 C.8 C.9 C.10 C.11. Na obrázku C.17 je zobrazena matice záměn pro autoenkodér 1 zpracovávající dataset filtrovaný plovoucím průměrem s oknem o délce 100 bodů. Odpovídající časový průběh časové detekce anomálií je zobrazen na obrázku C.19. Na obrázku C.18 je zobrazena matice záměn pro autoenkodér 2 zpracovávající dataset filtrovaný plovoucím průměrem s oknem o délce 100 bodů. Odpovídající časový průběh časové detekce anomálií je zobrazen na obrázku C.20.

5.6 Časová náročnost

Pokud porovnáme časovou náročnost autoenkodéru 1 a autoenkodéru 2, tak zjistíme, že při trénování i testování je časová náročnost sítě s LSTM buňkami podstatně vyšší. Při měření času potřebného k natrénování sítě jsem bral v úvahu i počet epoch nutný k natrénování. Každý model měl nastavený pevný počet epoch na 100, ale u některých se uplatnila funkce již dříve zmíněného brzkého ukončení (*angl. Early stopping*). Proto byl výsledný čas trénování přepočítán na 100 učících epoch.

Výběr optimizéru i ztrátové funkce měl na dobu trénování pouze zanedbatelný vliv. Jediné, co ovlivnilo dobu trénování, byla velikost dávky. U autoenkodéru 1

byla doba trénování přepočítaná na 100 epoch téměř 950 sekund (asi 15 minut). S rostoucí velikostí dávky tento čas klesal až k 17 sekundám při dávce 5000. Dále se už doba pro trénování nezkracovala. U autoenkodéru 2 byla doba potřebná pro naučení modelu přepočítaná na 100 epoch skoro 19000 sekund, což odpovídá cca 5,3 hodinám. Tento čas klesal až k 20 minutám, kterých trénovací proces modelu dosáhl při dávce 1000. Výsledky z testu závislosti doby trénování na velikosti dávky shrnuje tabulka 5.9.

Při vybavování byla časová náročnost vždy srovnatelná. U autoenkodéru byla doba vybavování cca 9 sekund a u autoenkodéru 2 cca 90 sekund.

Autoenkodéry byly testovány na datasetu Pump sensor data. Modely by tréno- vány na počítači s konfigurací AMD Ryzen 5 4600H with Radeon Graphics a 16 GB RAM. K trénování nebyla využita grafická karta.

Tab. 5.9: Tabulka porovnání metrik AUC a F1 score detektorů anomálií datasetu Pump sensor data filtrovaná plovoucím průměrem s různou délkou okna

	Autoenkodér 1 (nn)			Autoenkodér 2 (LSTM)		
Vel. dávky	Čas (s)	Epochy	Čas 100 epoch (s)	Čas (s)	Epochy	Čas 100 epoch (s)
10	84,95	9	943,95	1327,43	7	18963,67
50	18,15	8	226,93	738,99	18	4105,54
100	12,88	10	128,89	556,38	24	2318,26
500	6,13	11	55,75	749,30	53	1413,78
1000	7,44	22	33,85	849,59	69	1231,29
5000	10,00	56	17,85	233,53	19	1229,11
10000	17,21	100	17,21	1350,53	100	1350,53
20000	17,99	100	17,99	636,42	49	1298,83
50000	17,34	100	17,34	228,88	16	1430,53

6 Grafická aplikace pro detekci anomálií

V rámci diplomové práce byla vyvinuta aplikace pro testování autoenkodérů použitých v předchozí kapitole. V této kapitole budou představeny její vlastnosti a způsob jejího použití.

6.1 Základní popis aplikace

Aplikace pro detekci anomálií se skládá ze tří částí. Části pro práci s datasetem, části pro trénování a části pro testování. Mezi nimi lze poměrně jednoduše přepínat, protože jsou umístěny na samostatných kartách. V pravé části se nachází konzole, na kterou se vypisují při běhu programu základní informace, díky kterým je uživatel schopen vyhodnotit stav programu, popř. reagovat na chyby. Uživatelské rozhraní je vytvořeno v angličtině.

Celá aplikace je naprogramovaná v jazyce Python. Grafické rozhraní je vytvořeno pomocí knihovny PySide6, což je obdoba známější PyQt6 s jinými licenčními podmínkami. Aplikace používá více vláken, díky čemuž je její ovládání responzivní a grafické rozhraní okamžitě reaguje na vstupy od uživatele.

Aplikace předpokládá, že pokud je dataset anotovaný, obsahuje pouze 2 třídy a to anomální a normální, kdy anomální je označená číslem 1 a normální číslem 0.

V další sekci si představíme jednotlivé karty aplikace.

6.2 Karta Dataset

Karta Dataset slouží k zobrazení a úpravě datasetu. Na obrázku 6.1 je vidět snímek obrazovky, kde je otevřena. Karta je rozdělena do několika skupin ovládacích prvků, které jsou zeleně označeny a očíslovány.

Sekce 1 slouží pro načítání datasetu z .csv souboru. Po stisknutí tlačítka **Load dataset** se otevře okno, ve kterém uživatel může vybrat soubor, který chce načíst. Po načtení souboru je pod tlačítkem vidět jeho systémová cesta. Pokud uživatel nechce načíst první sloupec datasetu (např. jedná se o index řádku nebo časovou značku, která pro daný pokus nemá význam), může zabránit jeho načtení pomocí zaškrtačacího políčka **Load without first column**.

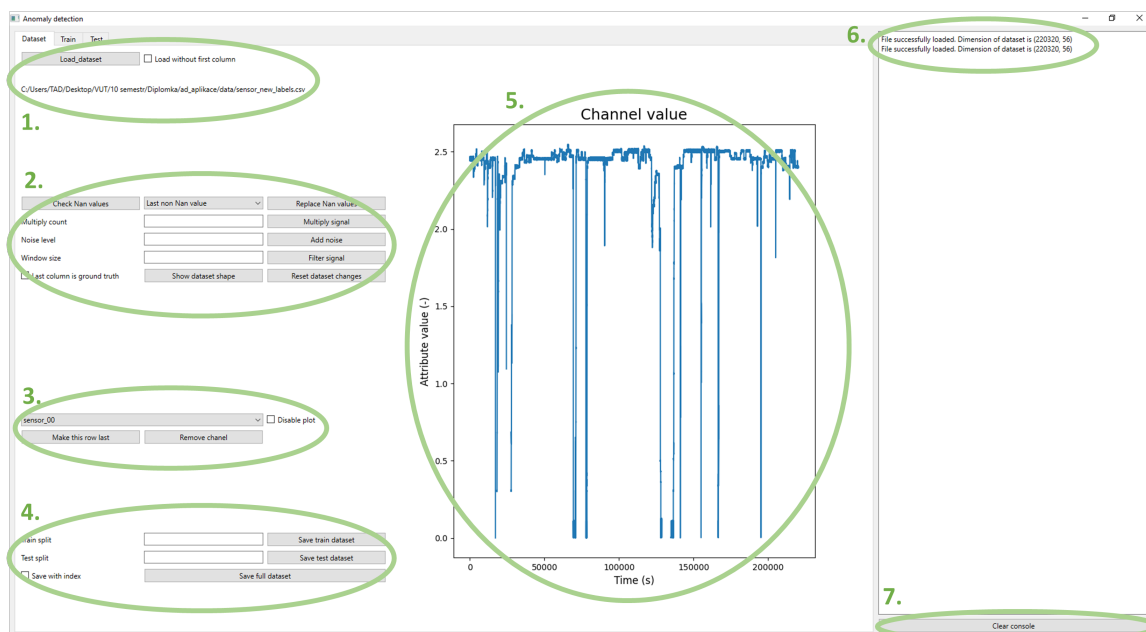
Sekce 2 slouží pro úpravu datasetu. Uživatel může kontrolovat přítomnost 'Nan' hodnot v datasetu a tyto hodnoty odstranit. Program podporuje odstranění pomocí dopředné (poslední platná hodnota v daném sloupci) i zpětné (nejbližší následující platná hodnota v daném sloupci) metody a také náhradu hodnotou 0. Dále lze v této sekci násobit celý dataset konstantou, přidávat šum s definovaným SNR nebo signál filtrovat pomocí klouzavého průměru s definovanou velikostí okna. Tyto změny lze

okamžitě sledovat v grafu v sekci 5. Pokud uživatel zašrtne políčko `Last column is ground truth`, nebudou žádné změny uplatněny na poslední sloupec. V této sekci je dále tlačítko pro zobrazení rozměrů datasetu a vrácení zpět všech provedených změn.

Sekce 3 obsahuje combo box pro procházení datasetu. Uživatel si může vybrat kanál (sloupec datasetu) podle jeho názvu a ten se následně vykreslí v sekci 5. Pokud by uživatel nechtěl zobrazovat jednotlivé kanály, může jejich vykreslování zastavit pomocí zaškrtačacího políčka. Tlačítko `Make this row last` posune daný kanál na konec datasetu. Program s ním následně může počítat jako s informací o anomaliě daného vzorku (*angl. ground truth*). Jednotlivé kanály lze odstranit pomocí tlačítka `Remove channel`.

Sekce 4 umožňuje ukládání zpracovaného datasetu. Pomocí políček `train split` a `test split` lze základní dataset rozdělit na části a ty uložit pomocí příslušných tlačítek. Do políček se píšou indexovací výrazy, které podporují standardní notaci jazyka Python pro indexování seznamu (*angl. list*). Např. výraz `10:20` vybere všechny prvky od indexu 10 do indexu 19. (Pozor Python indexuje od nuly). Výraz `20:` vybere všechny prvky od indexu 20 do konce seznamu. Uživatel si může vybrat pomocí zaškrtačacího políčka, jestli chce do souboru uložit i index řádků nebo ne. Uložení příslušného souboru se provede kliknutím na tlačítko a následným nastavením názvu souboru ve vyskakovacím okně.

Sekce 5 představuje graf pro zobrazení časového průběhu jednotlivých kanálů. Sekce 6. představuje výpisy do konzole při načítání souboru. Sekce 7. představuje tlačítko pro mazání obsahu konzole.



Obr. 6.1: Snímek obrazovky aplikace pro detekci anomálií s kartou Dataset

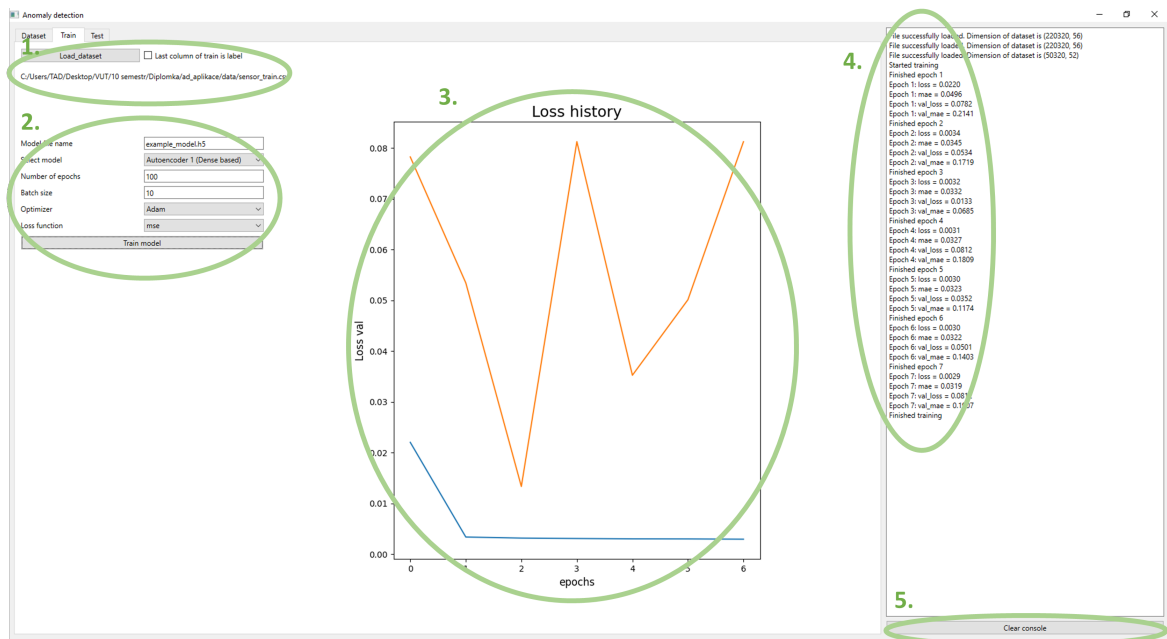
6.3 Karta Train

Karta Train slouží pro trénování a následné uložení natrénovaného modelu. Na obrázku 6.2 je vidět snímek obrazovky, kde je otevřena. Stejně jako v předchozí sekci je karta na obrázku rozdělena do několika skupin ovládacích prvků, které jsou zeleně označeny a očíslovány.

Sekce 1 slouží pro načtení trénovacího souboru. Pomocí zaškrtnutí tlačítka lze zvolit, jestli se má daný dataset načíst bez prvního sloupce, který může představovat index. Po načtení souboru je vidět jeho systémovou cestu.

Sekce 2 slouží pro nastavení parametrů trénovaného modelu. Uživatel vybírá název souboru pro trénovaný model. Název souboru se zadává bez přípony. Dále uživatel nastavuje model autoenkodéru, který chce použít. Na výběr má buď model založený na plně propojených vrstvách nebo model založený na LSTM buňkách. Dále uživatel může nastavit počet trénovacích epoch, velikost dávky, optimizér a ztrátovou funkci. Trénovací proces se spustí tlačítkem `Train model`. Nejlepší model se uloží automaticky po ukončení trénovacího procesu.

V sekci 3 je zobrazen průběh hodnot ztrátové funkce během trénování modelu. V sekci 4 je zobrazen ilustrační výpis z průběhu trénování. Sekce 5. představuje tlačítko pro mazání obsahu konzole.



Obr. 6.2: Snímek obrazovky aplikace pro detekci anomálií s otevřenou kartou Train

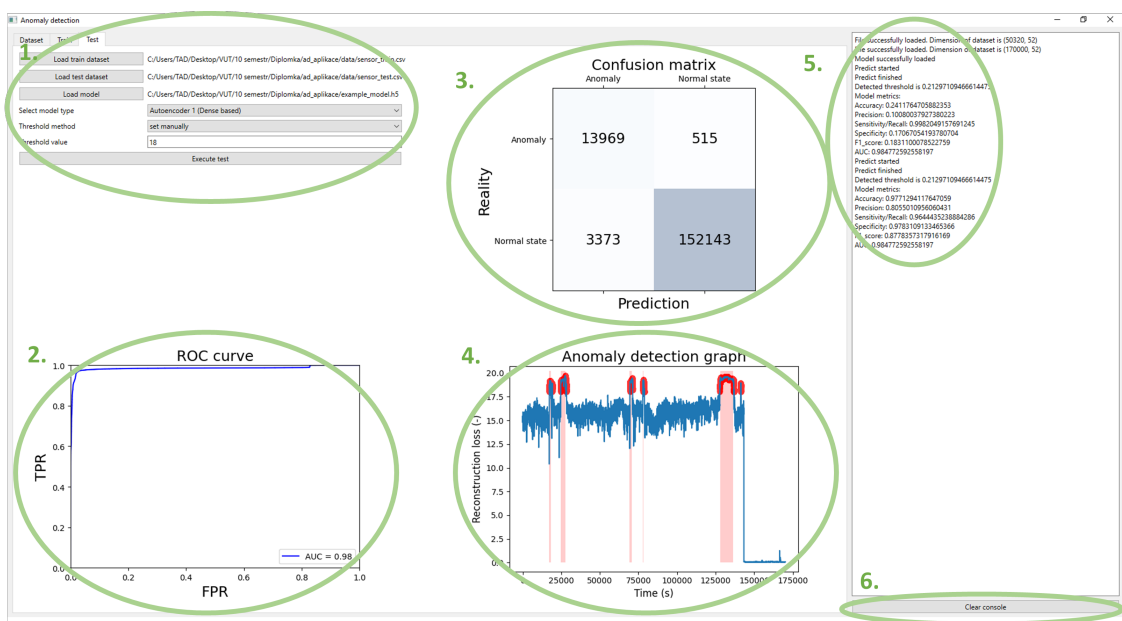
6.4 Karta Test

Karta Test slouží pro testování modelu natrénovaného na kartě Train. Na obrázku 6.3 je vidět snímek obrazovky, kde je otevřena. Stejně jako v předchozích sekcích je karta na obrázku rozdělena do několika skupin ovládacích prvků, které jsou zeleně označeny a očíslovány.

Sekce 1 slouží pro načtení trénovacího, testovacího datasetu a natrénovaného modelu. U obou datasetů předpokládá, že poslední sloupec dat představuje informaci o anomaliích vzorku (*angl. ground truth*), která slouží pro vyhodnocení výkonu daného modelu. Dále zde lze vybrat typ modelu, který musí korespondovat s načteným modelem, jinak dojde k chybě. Poté zde lze zvolit práh pro detekci anomálií. Tento práh může být buďto určen automaticky z trénovacího datasetu nebo manuálně zadanou hodnotou. Tlačítkem `Execute test` se spustí test.

Sekce 2, 3 a 4 obsahují grafy pro zhodnocení výkonosti daného modelu. Je zde zobrazena ROC křivka, matice záměn a časový průběh detekce anomálií.

V sekci 5 je výpis metrik daného testu jako například přesnost, F1 míra a další. Sekce 6. představuje tlačítko pro mazání obsahu konzole.



Obr. 6.3: Snímek obrazovky aplikace pro detekci anomálií s otevřenou kartou Test

Závěr

Cílem diplomové práce bylo provést rešerši metod strojového učení v oblasti Průmyslu 4.0 s důrazem především na prediktivní údržbu a vizuální inspekci. Dále zmapovat oblast detekce anomálií pro použití a v oblasti prediktivní údržby navrhnout experiment, který by ověřil funkci dané metody. Dále tento experiment prezentovat ve formě grafické aplikace a jeho výsledky následně vyhodnotit pomocí matice záměn a metrik z ní vycházejících.

První kapitola je věnována pojmu Průmysl 4.0. Ve stručnosti jsou zde zmíněny všechny průmyslové revoluce, jejich zásadní objevy a celospolečenské důsledky. Dále je zde představen koncept Průmysl 4.0 včetně souvisejících základních pojmů jako jsou Internet věcí, Kyberfyzikální systémy, Cloudy, Digitální dvojčata a další . . . V závěru první kapitoly byly představeny instituce, které se Průmyslem 4.0 aktivně zabývají.

Druhá kapitola je rozdělena do dvou částí. První se věnuje pojmu prediktivní údržba. Nejprve je zmíněn vývoj údržby zařízení v průmyslu, který postupně vedl až k systémům prediktivní údržby. Poté je vysvětlen samotný pojem prediktivní údržba a jeho přímá souvislost s detekcí anomálií v časových řadách. Dále jsou zmíněny různé metody vhodné pro detekci anomálií v časových řadách včetně jejich základního principu.

Druhá část druhé kapitoly je věnována vizuální inspekci. Prvně je stručně představen samotný pojem vizuální inspekce a následně jsou zmíněné typické metody použitelné v této oblasti. Na závěr je zmíněno, že metody použitelné pro vizuální inspekci lze obecně adaptovat i pro oblast prediktivní údržby.

V třetí kapitole jsou představeny datasey, které jsou prezentovány jako vhodné pro úlohy prediktivní údržby, resp. obecně detekce anomálií. Každý dataset je zde stručně představen a následně je zhodnocena vhodnost jeho použití v této práci. Podstatně detailněji je zde představen dataset Pump sensor data, který je použit v praktické části této práce. Dále jsou zde představeny generátory, které slouží pro vytvoření umělých datasetů. V závěru kapitoly je představen jednoduchý dataset, který sloužil pro základní ověření některých metod pro detekci anomálií v praktické části této práce.

Ve čtvrté kapitole jsou provedeny pokusy s několika metodami pro detekci anomálií. Pro základní ověření funkčnosti byl použit dataset z generátoru GutenTAG a poté dataset Pump sensor data. U každého algoritmu byl zhodnocen jeho výkon, který je vyjádřen graficky pomocí grafu časové detekce anomálií.

V páté kapitole byla do detailu rozebrána možnost použití autoenkodérů pro detekci anomálií v multidimenzionálních časových datech a předvedena na datasetu Pump sensor data. Pro experimenty byly použity dva autoenkodéry. Jeden byl za-

ložen na plně propojených vrstvách a druhý na LSTM buňkách. Experimenty byly navrženy tak, aby odpovídaly problémům v prediktivní údržbě. Hledali jsme tedy anomální stavy v časových datech technického zařízení a ty porovnávali s anotací datasetu. Výsledky všech experimentů byly vyhodnoceny pomocí matice záměn a metrik z ní vycházejících. Jako směrodatné metriky byly zvoleny AUC a F1 míra. Pro lepší názornost byl přidán i graf ukazující detekované anomální body v čase.

V rámci experimentů byl testován vliv některých hyperparametrů na výkon modelu a na časovou náročnost učení. Mezi testovanými hyperparametry byla velikost dávky, zvolený optimizér a ztrátová funkce. Z výsledků vyplývá, že optimální velikost dávky z hlediska výsledků byla do jednoho procenta trénovacího datasetu. U autoenkodéru 1 to bylo 10 a u autoenkodéru 2 to bylo 500. Nejlepší výsledky z testovaných optimizérů měl u obou autoenkodérů optimizér AdaMax. Při testování ztrátových funkcí dosáhl nejlepších výsledků autoenkodér 1 s použitím ztrátové funkce MSE a autoenkodér 2 se ztrátovou funkcí MAE.

Dále byla provedena citlivostní analýza, která měla za cíl zkoumat vliv změn datasetu na výkon daných modelů. Byl testován vliv násobení všech signálů konstantou, vliv přidaného šumu a vliv filtrace signálu pomocí plovoucího průměru. V případě násobení signálu konstantou nebyl zaznamenán žádný vliv. V případě ovlivnění přidáním bílého šumu fungovaly oba autoenkodéry do SNR 20 dB. Poté jejich výkonnost začala mírně klesat (U autoenkodéru 1 podstatně významněji). Při SNR 5 dB ztratili oba autoenkodéry schopnost detekce naprosto. Ovlivnění délkou okna bylo detekovatelné, ale bylo spíše způsobeno špatným určením prahu pro detekci anomálie. Výkonnost samotných autoenkodérů nebyla filtrací významně ovlivněna.

Na závěr páté kapitoly byla zhodnocena časová náročnost obou autoenkodérů. Časová náročnost autoenkodéru 2 byla podstatně vyšší. Trénovací časy se daly výrazně ovlivnit velikostí dávky. Při jejím zvětšování klesal čas potřebný k naučení sítě.

Na závěr byla představena aplikace pro zpracování datasetu, trénování modelů pro detekci anomálií a jejich testování.

Literatura

- [1] ÇINAR, Zeki Murat, Abubakar ABDUSSALAM NUHU, Qasim ZEESHAN, Orhan KORHAN, Mohammed ASMAEL a Babak SAFAEI. *Machine Learning in Predictive Maintenance towards Sustainable Smart Manufacturing in Industry 4.0. Sustainability* [online]. 2020, 5. 10. 2020, 12(19) [cit. 2022-11-17]. ISSN 2071-1050. Dostupné z: <[doi:10.3390/su12198211](https://doi.org/10.3390/su12198211)>.
- [2] ACHOUCH, Mounia, Mariya DIMITROVA, Khaled ZIANE, Sasan SATTAR-PANAH KARGANROUDI, Rizck DHOUIB, Hussein IBRAHIM a Mehdi ADDA. *On Predictive Maintenance in Industry 4.0: Overview, Models, and Challenges*. *Applied Sciences* [online]. 2022, 12(16) [cit. 2022-11-17]. ISSN 2076-3417. Dostupné z: <[doi:10.3390/app12168081](https://doi.org/10.3390/app12168081)>.
- [3] KAMAT, Pooja, Rekha SUGANDHI, S.B. SANAP, V.K. BHOJWANI a MATHEW V K. *Anomaly Detection for Predictive Maintenance in Industry 4.0- A survey*. *E3S Web of Conferences* [online]. 2020, 170 [cit. 2022-11-17]. ISSN 2267-1242. Dostupné z: <[doi:10.1051/e3sconf/202017002007](https://doi.org/10.1051/e3sconf/202017002007)>.
- [4] SCHMIDL, Sebastian, Phillip WENIG a Thorsten PAPENBROCK. *Anomaly detection in time series*. *Proceedings of the VLDB Endowment* [online]. 2022, 15(9), 1779-1797 [cit. 2022-11-17]. ISSN 2150-8097. Dostupné z: <<https://www.vldb.org/pvldb/vol15/p1779-wenig.pdf>>.
- [5] MALHOTRA, Pankaj, Lovekesh VIG, Gautam M. SHROFF a Puneet AGARWAL. *Long Short Term Memory Networks for Anomaly Detection in Time Series*. *ESANN* [online]. 2015, 2015 [cit. 2022-11-17]. Dostupné z: <<https://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2015-56.pdf>>.
- [6] AKASH, Singh. *Anomaly Detection for Temporal Data using Long Short-Term Memory (LSTM)*. Stockholm, 2017. Master thesis. KTH, School of Information and Communication Technology (ICT). [cit. 2022-11-17]. Dostupné z: <<https://www.diva-portal.org/smash/get/diva2:1149130/FULLTEXT01.pdf>>.
- [7] BARNETT, Vic a Toby LEWIS. *Outliers in Statistical Data, 3rd Edition.*, 3. vydání. Wiley, 1994. ISBN 978-0-471-93094-5.
- [8] HAWKINS, D. M. *Identification of Outliers: Monographs on Statistics and Applied Probability.*, Ilustrované vydání. Springer Science & Business Media, 2013. ISBN 9789401539944.

- [9] CHOI, Kukjin, Jihun YI, Changhwa PARK a Sungroh YOON. *Deep Learning for Anomaly Detection in Time-Series Data: Review, Analysis, and Guidelines*. IEEE Access [online]. 2021, 9, 120043-120065 [cit. 2022-11-17]. ISSN 2169-3536. Dostupné z: <doi:10.1109/ACCESS.2021.3107975>.
- [10] MOHAJAN, Haradhan. *The First Industrial Revolution: Creation of a New Global Human Era* [online]. 18.10.2019, 5(5), 377-387 [cit. 2022-11-19]. Dostupné z: <https://mpra.ub.uni-muenchen.de/96644/1/MPRA_paper_96644.pdf>.
- [11] Neznáný autor *The Second Industrial Revolution: The Technological Revolution* [online]. 16.5.2022 [cit. 2022-11-19]. Dostupné z: <https://richmondvale.org/blog/second-industrial-revolution/#What_was_the_Second_Industrial_Revolution_and_what_caused_it>.
- [12] ENGELMAN, Ryan. *The Second Industrial Revolution, 1870-1914* [online]. [cit. 2022-11-19]. Dostupné z: <https://ushistoryscene.com/article/second-industrial-revolution/>.
- [13] WARD, Kimberley. *Timeline of Revolutions*. Manufacturing Data Summit [online]. 2019, 18. 2. 2019 [cit. 2022-11-19]. Dostupné z: <https://manufacturingdata.io/newsroom/timeline-of-revolutions/>.
- [14] KNELL, Mark. *The digital revolution and digitalized network society* [online]. 13. 2. 2021, 9-25 [cit. 2022-11-22]. Dostupné z: <doi:https://doi.org/10.1007/s43253-021-00037-4>.
- [15] ROBERTS, Brian. *The Third Industrial Revolution: Implications for Planning Cities and Regions*. Working Paper Urban Frontiers [online]. 2015, 15.6.2015, 1(1) [cit. 2022-11-22]. Dostupné z: <https://www.researchgate.net/publication/278671121_The_Third_Industrial_Revolution_Implications_for_Planning_Cities_and_Regions>.
- [16] VAIDYA, Saurabh, Prashant AMBAD a Santosh BHOSLE. *Industry 4.0 – A Glimpse*. Procedia Manufacturing [online]. MIT Aurangabad, Maharashtra, INDIA, 2017, 11.12.2017, 20(20), 233-238 [cit. 2022-11-22]. ISSN 2351-9789. Dostupné z: <doi:https://doi.org/10.1016/j.promfg.2018.02.034>.
- [17] SULEIMAN, Zhanybek, Sabit SHAIKHOLLA, Dinara DIKHANBAYEVA, Essam SHEHAB, Ali TURKYILMAZ a Zude ZHOU. *Industry 4.0: Clustering of concepts and characteristics*. Cogent Engineering [online]. 2022, 9(1) [cit. 2022-11-22]. ISSN 2331-1916. Dostupné z: <doi:10.1080/23311916.2022.2034264>.

- [18] ROBLEK, Vasja, Maja MEŠKO a Alojz KRAPEŽ. *A Complex View of Industry 4.0*. SAGE Open [online]. 2016, 6(2) [cit. 2022-11-22]. ISSN 2158-2440. Dostupné z: <doi:10.1177/2158244016653987>.
- [19] GHOBAKHLOO, Morteza. *Industry 4.0, digitization, and opportunities for sustainability*. Journal of Cleaner Production [online]. 2020, 252 [cit. 2022-11-22]. ISSN 09596526. Dostupné z: <doi:10.1016/j.jclepro.2019.119869>.
- AKASH, Singh. Stockholm, 2017. Master thesis. KTH, School of Information and Communication Technology (ICT).
- [20] AKASH, Singh. *Anomaly Detection for Temporal Data using Long Short-Term Memory (LSTM)*. Stockholm, 2017. Master thesis. KTH, School of Information and Communication Technology (ICT).[cit. 2022-12-23]. Dostupné z: <https://www.diva-portal.org/smash/get/diva2:1149130/FULLTEXT01.pdf>.
- [21] MALHOTRA, Pankaj, Lovekesh VIG, Gautam M. SHROFF a Puneet AGARWAL. *Long Short Term Memory Networks for Anomaly Detection in Time Series*. ESANN [online]. 2015, 2015 [cit. 2022-11-17]. Dostupné z: <https://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2015-56.pdf>.
- [22] BREUNIG, Markus M., Hans-Peter KRIEGEL, Raymond T. NG a Jörg SANDER. *LOF*. Proceedings of the 2000 ACM SIGMOD international conference on Management of data - SIGMOD '00 [online]. New York, New York, USA: ACM Press, 2000, 2000, 93-104 [cit. 2022-12-23]. ISBN 1581132174. Dostupné z: <doi:10.1145/342009.335388>.
- [23] THILL, Markus, Wolfgang KONEN a Thomas BÄCK. *Time Series Anomaly Detection with Discrete Wavelet Transforms and Maximum Likelihood Estimation* [online]. 1.1.2019 [cit. 2022-12-23]. Dostupné z: <https://www.researchgate.net/publication/330400907_Time_Series_Anomaly_Detection_with_Discrete_Wavelet_Transforms_and_Maximum_Likelihood_Estimation>.
- [24] MUNIR, Mohsin, Shoaib Ahmed SIDDIQUI, Andreas DENGEL a Sheraz AHMED. *DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series*. IEEE Access [online]. 2019, 7, 1991-2005 [cit. 2022-12-23]. ISSN 2169-3536. Dostupné z: <doi:10.1109/ACCESS.2018.2886457>.

- [25] LIU, Fei Tony, Kai Ming TING a Zhi-Hua ZHOU. *Isolation Forest*. 2008 Eighth IEEE International Conference on Data Mining [online]. IEEE, 2008, 2008, 413-422 [cit. 2022-12-25]. ISBN 978-0-7695-3502-9. Dostupné z: <doi:10.1109/ICDM.2008.17>.
- [26] KOENIGSTEIN, Dor, Bank KOENIGSTEIN, Noam RAJA a Giryes RAJA. *Autoencoders* [online]. arXiv, 2020 [cit. 2022-12-27]. Dostupné z: <doi:10.48550/ARXIV.2003.05991>.
- [27] XIA, Xuan, Xizhou PAN, Nan LI, Xing HE, Lin MA, Xiaoguang ZHANG a Ning DING. *GAN-based anomaly detection: A review*. Neurocomputing [online]. 2022, 493, 497-535 [cit. 2022-12-27]. ISSN 09252312. Dostupné z: <doi:10.1016/j.neucom.2021.12.093>.
- [28] WANG, Huan-gang, Xin LI a Tao ZHANG. *Generative adversarial network based novelty detection using minimized reconstruction error* [online]. 2018, 19(1), 116-125 [cit. 2022-12-27]. ISSN 2095-9184. Dostupné z: <doi:10.1631/FITEE.1700786>.
- [29] *Machine Predictive Maintenance Classification: Dataset to predict machine failure (binary) and type (multiclass)*. <https://www.kaggle.com> [online]. [cit. 2023-01-13]. Dostupné z: <<https://www.kaggle.com/datasets/shivamb/machine-predictive-maintenance-classification>>.
- [30] *Microsoft Azure Predictive Maintenance: Data for predictive Maintenance*. <https://www.kaggle.com> [online]. [cit. 2023-01-13]. Dostupné z: <<https://www.kaggle.com/datasets/arnabbiswas1/microsoft-azure-predictive-maintenance>>.
- [31] *NASA Bearing Dataset: Prognostic Dataset for Predictive/Preventive Maintenance*. <https://www.kaggle.com> [online]. [cit. 2023-01-13]. Dostupné z: <<https://www.kaggle.com/datasets/vinayak123tyagi/bearing-dataset>>.
- [32] *Elevator Predictive Maintenance Dataset: Elevator predictive maintenance datasets from Huawei German Research Center*. <https://www.kaggle.com> [online]. [cit. 2023-01-13]. Dostupné z: <<https://www.kaggle.com/datasets/shivamb/elevator-predictive-maintenance-dataset>>.
- [33] *Preventive to Predictive Maintenance: Extensive data set for machine learning on remaining useful life prognosis (PHM)*. <https://www.kaggle.com> [online]. [cit. 2023-01-13]. Dostupné z: <<https://www.kaggle.com/datasets/prognosticshse/preventive-to-predictive-maintenance>>.

- [34] *Healthcare Providers Data For Anomaly Detection: Healthcare Provider Fraud Detection Using Unsupervised Learning*. <https://www.kaggle.com> [online]. [cit. 2023-01-13]. Dostupné z: <<https://www.kaggle.com/datasets/prognosticshse/preventive-to-predictive-maintenance>>.
- [35] *Eighty years of Canadian climate data: Daily temperatures and precipitation at 13 Canadian centres*. <https://www.kaggle.com> [online]. [cit. 2023-01-13]. Dostupné z: <<https://www.kaggle.com/datasets/tamilsel/healthcare-providers-data>>.
- [36] *Skoltech Anomaly Benchmark (SKAB): Time-series data for evaluating Anomaly Detection algorithms*. <https://www.kaggle.com> [online]. [cit. 2023-01-13]. Dostupné z: <<https://www.kaggle.com/datasets/yuriykatser/skoltech-anomaly-benchmark-skab>>.
- [37] *Numenta Anomaly Benchmark (NAB): Dataset and scoring for detecting anomalies in streaming data*. <https://www.kaggle.com> [online]. [cit. 2023-01-13]. Dostupné z: <<https://www.kaggle.com/datasets/boltzmannbrain/nab>>.
- [38] *Predictive Maintenance: Naval Vessel Condition: Lever Position, Ship Speed, and GT Measures at Steady State*. <https://www.kaggle.com> [online]. [cit. 2023-01-13]. Dostupné z: <<https://www.kaggle.com/datasets/thedevastator/improving-naval-vessel-condition-through-machine>>.
- [39] *E-coating ultrafiltration maintenance dataset: E-paint coating process including degradation progressions for multiple filters*. <https://www.kaggle.com> [online]. [cit. 2023-01-24]. Dostupné z: <https://www.kaggle.com/datasets/boyangs444/process-data-for-predictive-maintenance?select=iiot_30min_norm.csv>.
- [40] *One Year Industrial Component Degradation: Degradation of a cutting blade*. <https://www.kaggle.com> [online]. [cit. 2023-01-24]. Dostupné z: <https://www.kaggle.com/datasets/boyangs444/process-data-for-predictive-maintenance?select=iiot_30min_norm.csv>.
- [41] *Vega shrink-wrapper component degradation: New vs worn cutting blade data*. <https://www.kaggle.com> [online]. [cit. 2023-01-24]. Dostupné z: <<https://www.kaggle.com/datasets/inIT-OWL/vega-shrinkwrapper-runtofailure-data>>.

- [42] *Energy of Las Vegas Street (Time-series): Two Year Time-Series Data*. <https://www.kaggle.com> [online]. [cit. 2023-01-24]. Dostupné z: <<https://www.kaggle.com/datasets/thedevastator/optimizing-energy-conservation-of-las-vegas-stre>>.
- [43] *Condition Based Maintenance of Naval Propulsion Plants Data Set*. UCI: Machine learning repository [online]. [cit. 2023-01-24]. Dostupné z: <<http://archive.ics.uci.edu/ml/datasets/condition+based+maintenance+of+naval+propulsion+plants>>.
- [44] *NASA Turbofan Jet Engine Data Set: Run to Failure Degradation Simulation*. <https://www.kaggle.com> [online]. [cit. 2023-01-24]. Dostupné z: <<https://www.kaggle.com/datasets/behrad3d/nasa-cmaps>>.
- [45] *Genesis demonstrator data for machine learning: Data with normal behaviour and data with anomalies*. <https://www.kaggle.com> [online]. [cit. 2023-01-24]. Dostupné z: <<https://www.kaggle.com/datasets/inIT-OWL/genesis-demonstrator-data-for-machine-learning>>.
- [46] *Condition monitoring of hydraulic systems Data Set*. UCI: Machine learning repository [online]. [cit. 2023-01-24]. Dostupné z: <<https://archive.ics.uci.edu/ml/datasets/Condition+monitoring+of+hydraulic+systems#>>.
- [47] *Oxford Battery Degradation Dataset 1: Long term battery ageing tests of 8 Kokam (SLPB533459H4) 740 mAh lithium-ion pouch cells*. Oxford University Research Archive [online]. [cit. 2023-01-24]. Dostupné z: <<https://ora.ox.ac.uk/objects/uuid:03ba4b01-cfed-46d3-9b1a-7d4a7bdf6fac>>.
- [48] *CalIt2 Building People Counts Data Set*. UCI: Machine learning repository [online]. [cit. 2023-01-24]. Dostupné z: <<https://archive.ics.uci.edu/ml/datasets/CalIt2+Building+People+Counts>>.
- [49] *Pump sensor data: Pump sensor data for predictive maintenance*. <https://www.kaggle.com> [online]. [cit. 2023-01-24]. Dostupné z: <<https://www.kaggle.com/datasets/nphantawee/pump-sensor-data>>.
- [50] *GutenTAG: A good Timeseries Anomaly Generator*. [online]. GitHub, 2022 [cit. 2023-01-24]. Dostupné z: <<https://github.com/HPI-Information-Systems/gutentag>>.
- [51] *MGAB: The Mackey-Glass Anomaly Benchmark* [online]. GitHub, 2022 [cit. 2023-01-24]. Dostupné z: <<https://github.com/MarkusThill/MGAB>>.

- [52] GUPTA, Jyoti. *Air-Pollution-Forecasting*. GitHub [online]. 12. 8. 2020 [cit. 2023-04-03]. Dostupné z: <<https://github.com/jyoti0225/Air-Pollution-Forecasting>>.
- [53] TRAJCESKI, Jovan. *Timeseries-anomaly-detection-using-LSTM*. 2021, 25. 6. 2021 [cit. 2023-04-03]. Dostupné z: <<https://github.com/trajceskijovan/Timeseries-anomaly-detection-using-LSTM#timeseries-anomaly-detection-using-lstm>>.
- [54] YAMEHMET, Sarika. *Cnn-bilstm-attention*. GitHub [online]. 2020, 6. 10. 2020 [cit. 2023-04-03]. Dostupné z: <<https://github.com/sarikayamehmet/cnn-bilstm-attention>>.
- [55] KRUTILIN, Andrey. *Ds-predictive-maintenace*. GitHub [online]. 2023, 18.2.2023 [cit. 2023-04-08]. Dostupné z: <<https://github.com/andrey101010/ds-predictive-maintenace>>.
- [56] SINGH, Ajit Kumar. *Credit-Card-Fraud-Detection-ML-Model*. GitHub [online]. 2019, 26. 10. 2019 [cit. 2023-04-10]. Dostupné z: <<https://github.com/ajitsingh98/Credit-Card-Fraud-Detection-ML-Model>>.
- [57] LEE, Jeong Yup. *Sequential Anomaly detecion*. GitHub [online]. 2021, 3.9.2021 [cit. 2023-04-14]. Dostupné z: <https://github.com/forallx94/Sequential_Anomaly_detecion>.
- [58] *Tf.keras.optimizers.Adam: Optimizer that implements the Adam algorithm*. Tensorflow [online]. 2023, 2023 [cit. 2023-04-21]. Dostupné z: <https://www.tensorflow.org/api_docs/python/tf/keras/optimizers/Adam>.
- [59] <https://research.einar.partners/mystery-of-predictions-in-anomaly-detection/> Dostupné z: <<https://research.einar.partners/mystery-of-predictions-in-anomaly-detection/>>.
- [60] Anglická průmyslová revoluce a její rozšíření: Parní stroj [online]. [cit. 2023-01-23]. Dostupné z: <<https://www.dejepis.com/ucebnice/anglicka-prumyslova-revoluce-a-jeji-rozsireni/>>.
- [61] ZAMAN, Rokon. *Second Industrial Revolution in Perspective: Technologies like ICE, electric power and motor, vacuum tube, and radio waves unfolded creative waves: light bulb, automobile, radio and TV, and mass production*. The Waves: Technology, Society and Policy [online]. 2020, 16. 10. 2020 [cit. 2023-01-23]. Dostupné z: <<https://www.the-waves.org/2020/10/16/second-industrial-revolution-in-perspective/>>.

- [62] SEGOVIA, Vanessa Romero. *History of Control History of PLC and DCS* [online]. 2013 [cit. 2023-01-23]. Dostupné z: <<https://www.semanticscholar.org/paper/History-of-Control-History-of-PLC-and-DCS-Segovia/34cb56424212448031d71588c8d74f263e31c6fd>>.
- [63] SAYER, Peter. *Nvidia and Siemens to accelerate digital twins' development*. CIO [online]. 25. 5. 2022 [cit. 2023-01-23]. Dostupné z: <<https://www.cio.com/article/402420/nvidia-and-siemens-to-accelerate-digital-twins-development.html>>.
-

Seznam symbolů a zkratek

AUC	Plocha pod ROC křivkou – Area Under the ROC Curve
CEC	paměť LSTM buňky – Constant Error Carousel
CNN	Konvoluční neuronová síť – Convolutional neural network
CPS	Kyber-fyzikální systém – Cyber-physical systems
D	diskriminátor v GAN
DWT-MLEAD	metoda detekce anomálií – Discrete Wavelet Transform and Maximum Likelihood Estimation for Anomaly Detection
G	generátor v GAN
GAN	Generativní soupeřící síť – Generative adversarial network
GutenTAG	generátor nomálních časových řad – A good Timeseries Anomaly Generator
HMI	Rozhraní člověk-stroj – Human-Machine Interface
IF	metoda detekce anomálií – Isolation Forest
LOF	metoda detekce anomálií – Local Outlier Factor
LSTM	metoda detekce anomálií – Long Short-Term Memory
MAE	Průměrná absolutní chyba – Mean Absolute Error
MGAB	generátor anomálních časových řad – The Mackey-Glass Anomaly Benchmark
MLE	Odhad maximální věrohodnosti – Maximum Likelihood Estimation
PLC	Programovatelný logický automat – Programmable Logic Controller
RNN	Rekurentní neuronová síť – Recurrent Neural Network
ROC	metrika pro hodnocení klasifikátorů – Receiver Operating Characteristic curve
RUL	Zbývající užitečný život zařízení – Remaining Usefull Life
SGD	typ optimizéru – Stochastic Gradient Descent

SNR	Odstup signál-šum – Signal to Noise Ration
Nan	hodnota, která není konečné číslo – Not a number

Seznam příloh

A	Zdrojové kódy	102
A.1	Skripty pro generování testovacích signálů pomocí GutenTAG	102
B	Vlivu hyperparametrů - tabulky a grafy	104
B.1	Vliv velikosti dávky	104
B.2	Vliv použitého optimizéru	110
B.3	Vliv ztrátové funkce	114
C	Citlivostní analýza - tabulky a grafy	118
C.1	Vliv násobení konstantou	118
C.2	Vliv přidání šumu	123
C.3	Vliv filtrace signálu	129
D	Obsah elektronické přílohy	133

A Zdrojové kódy

A.1 Skripty pro generování testovacích signálů pomocí GutenTAG

Část zdrojového kódu skriptu pro generování sinusové vlny s anomáliemi.

```
1 config = {
2   "timeseries": [
3     {
4       "name": "test",
5       "length": 10000,
6       "semi-supervised": True,
7       "base-oscillations": [
8         {"kind": "sine", "variance": 0.0, "frequency": 1},
9       ],
10      "anomalies": [
11        {
12          "length": 500,
13          "channel": 0,
14          "exact-position": 1000,
15          "kinds": [{"kind": "amplitude", "amplitude_factor": .5}]
16        },
17        {
18          "length": 500,
19          "channel": 0,
20          "exact-position": 3000,
21          "kinds": [{"kind": "mean", "offset": .5}]
22        },
23        {
24          "length": 1,
25          "channel": 0,
26          "exact-position": 5000,
27          "kinds": [{"kind": "extremum", "min": False, "local": False, "context_window": 255}]
28        },
29        {
30          "length": 500,
31          "channel": 0,
32          "exact-position": 6000,
33          "kinds": [{"kind": "frequency", "frequency_factor": 2}]
34        },
35        {
36          "length": 500,
37          "channel": 0,
38          "exact-position": 8000,
39          "kinds": [{"kind": "variance", "variance": 1}]
40        },
41      ]
42    }
43 ]
44 }
```

Část skriptu pro generování obdélníkového signálu s anomáliemi.

```
1 config = {
2   "timeseries": [
3     {
4       "name": "test",
5       "length": 10000,
6       "semi-supervised": True,
7       "base-oscillations": [
8         {"kind": "square", "variance": 0.00, "frequency": 1, "duty": 0.5},
9       ],
10      "anomalies": [
11        {
12          "length": 500,
13          "channel": 0,
14          "exact-position": 1000,
15          "kinds": [{"kind": "amplitude", "amplitude_factor": .5}]
16        },
17        {
18          "length": 500,
19          "channel": 0,
```

```

20         "exact-position": 3000,
21         "kinds": [{"kind": "mean", "offset": .5}]
22     },
23     {
24         "length": 1,
25         "channel": 0,
26         "exact-position": 5000,
27         "kinds": [{"kind": "extremum", "min": False, "local": False, "context_window": 255}]
28     },
29     {
30         "length": 500,
31         "channel": 0,
32         "exact-position": 6000,
33         "kinds": [{"kind": "frequency", "frequency_factor": 2}]
34     },
35     {
36         "length": 500,
37         "channel": 0,
38         "exact-position": 8000,
39         "kinds": [{"kind": "variance", "variance": 1}]
40     },
41     {
42         "length": 500,
43         "channel": 0,
44         "exact-position": 9000,
45         "kinds": [{"kind": "pattern", "square_duty": 0.2}]
46     },
47 ]
48 }
49 ]
50 }

```

Část zdrojového kódu skriptu pro generování Dirichletovi vlny s anomáliemi.

```

1 config = {
2     "timeseries": [
3         {
4             "name": "test",
5             "length": 10000,
6             "semi-supervised": True,
7             "base-oscillations": [
8                 {"kind": "dirichlet", "frequency": 1, "amplitude": 2, "periodicity": 7}
9             ],
10            "anomalies": [
11                {
12                    "length": 500,
13                    "channel": 0,
14                    "exact-position": 1000,
15                    "kinds": [{"kind": "amplitude", "amplitude_factor": .5}]
16                },
17                {
18                    "length": 500,
19                    "channel": 0,
20                    "exact-position": 3000,
21                    "kinds": [{"kind": "mean", "offset": .5}]
22                },
23                {
24                    "length": 1,
25                    "channel": 0,
26                    "exact-position": 5000,
27                    "kinds": [{"kind": "extremum", "min": False, "local": False, "context_window": 255}]
28                },
29                {
30                    "length": 500,
31                    "channel": 0,
32                    "exact-position": 6000,
33                    "kinds": [{"kind": "frequency", "frequency_factor": 2}]
34                },
35                {
36                    "length": 500,
37                    "channel": 0,
38                    "exact-position": 8000,
39                    "kinds": [{"kind": "variance", "variance": 1}]
40                },
41            ],
42        }
43    ]
44 }
45 }

```

B Vlivu hyperparametrů - tabulky a grafy

B.1 Vliv velikosti dávky

Tab. B.1: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 1

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.84765	0.83677
Přesnost	Precision	0.32789	0.342746
Senzitivita	Recall	0.75075	0.99806
Specifická	Specificity	0.85667	0.82174
F1 míra	F1 score	0.45644	0.51026
AUC	AUC	0.80371	0.90991

Tab. B.2: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 10

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.90799	0.87015
Přesnost	Precision	0.480735	0.39593
Senzitivita	Recall	0.99669	0.99696
Specifická	Specificity	0.89973	0.85834
F1 míra	F1 score	0.64862	0.56677
AUC	AUC	0.94821	0.92765

Tab. B.3: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 50

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.84811	0.92926
Přesnost	Precision	0.35901	0.54675
Senzitivita	Recall	0.99655	0.99268
Specifická	Specificity	0.83429	0.92336
F1 míra	F1 score	0.52786	0.70513
AUC	AUC	0.91541	0.95802

Tab. B.4: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 100

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.84945	0.97412
Přesnost	Precision	0.36110	0.79161
Senzitivita	Recall	0.99696	0.94497
Specifická	Specificity	0.83571	0.97683
F1 míra	F1 score	0.53017	0.86152
AUC	AUC	0.91634	0.96090

Tab. B.5: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 500

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.84796	0.98812
Přesnost	Precision	0.33380	0.88243
Senzitivita	Recall	0.787765	0.99289
Specifická	Specificity	0.85357	0.98767
F1 míra	F1 score	0.46891	0.93441
AUC	AUC	0.82067	0.99028

Tab. B.6: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 1000

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.85145	0.96742
Přesnost	Precision	0.24898	0.72442
Senzitivita	Recall	0.36875	0.99696
Specifická	Specificity	0.89641	0.96468
F1 míra	F1 score	0.29725	0.83912
AUC	AUC	0.63258	0.98082

Tab. B.7: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 5000

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.91602	0.98581
Přesnost	Precision	0.67622	0.93182
Senzitivita	Recall	0.027685	0.89933
Specifická	Specificity	0.99876	0.99387
F1 míra	F1 score	0.05319	0.91529
AUC	AUC	0.51322	0.94660

Tab. B.8: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 100000

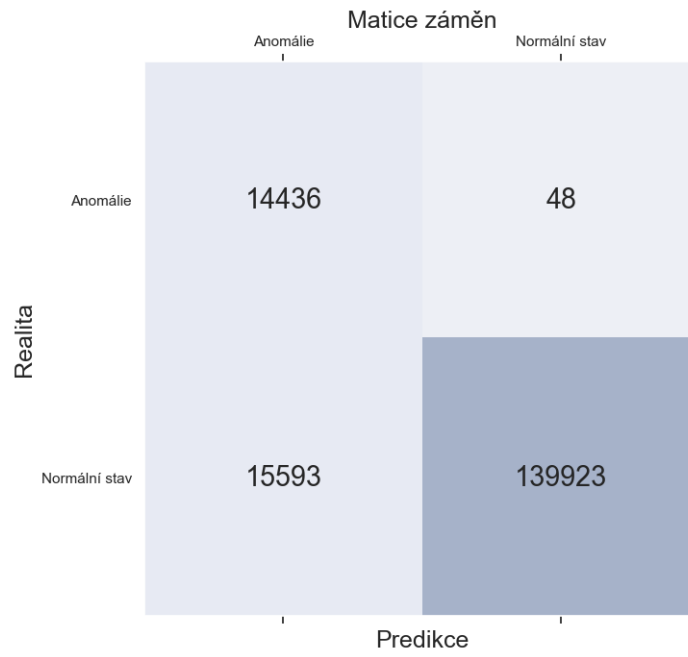
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.84661	0.98011
Přesnost	Precision	0.32251	0.92041
Senzitivita	Recall	0.72721	0.83913
Specifická	Specificity	0.85773	0.99324
F1 míra	F1 score	0.44689	0.87789
AUC	AUC	0.79247	0.91619

Tab. B.9: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 20000

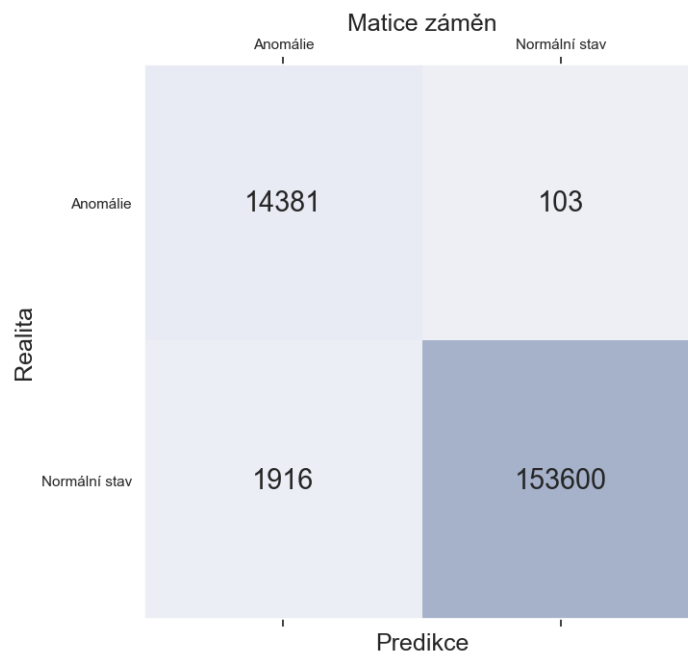
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.84408	0.96079
Přesnost	Precision	0.31351	0.75179
Senzitivita	Recall	0.69767	0.80592
Specifická	Specificity	0.85772	0.97521
F1 míra	F1 score	0.43261	0.77791
AUC	AUC	0.77769	0.89057

Tab. B.10: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným s velikostí dávky nastavenou na 50000

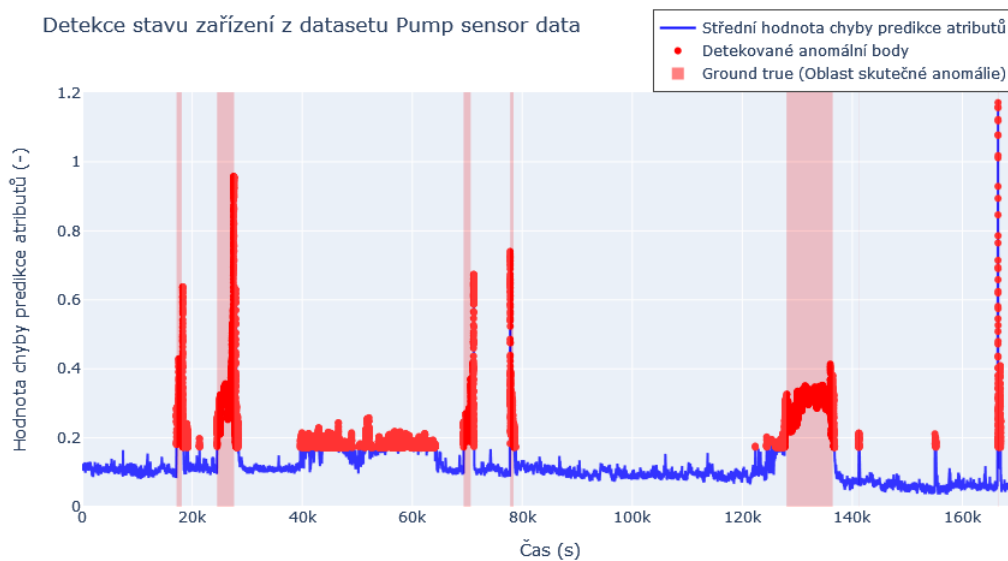
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.85115	0.84212
Přesnost	Precision	0.33863	0.24338
Senzitivita	Recall	0.78383	0.40452
Specifická	Specificity	0.85742	0.88288
F1 míra	F1 score	0.47294	0.30391
AUC	AUC	0.82062	0.64369



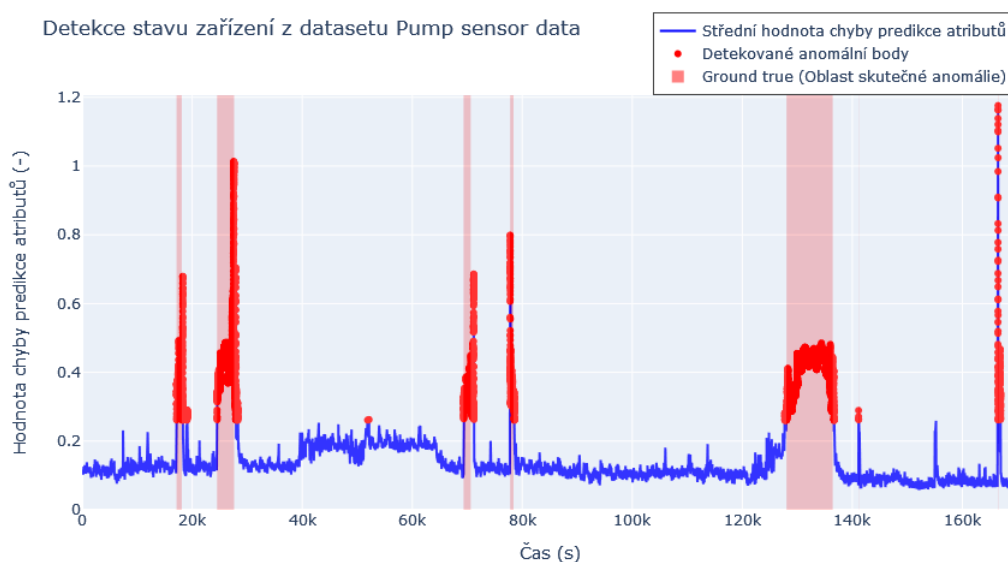
Obr. B.1: Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 1 trénované s velikostí dávky 10



Obr. B.2: Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 2 trénované s velikostí dávky 500



Obr. B.3: Detekce anomálního stavu zařízení z datasetu Pump sensor data s auto-
 enkodérem 1 trénované s velikostí dávky 10



Obr. B.4: Detekce anomálního stavu zařízení z datasetu Pump sensor data s auto-
 enkodérem 2 trénované s velikostí dávky 500

B.2 Vliv použitého optimizéru

Tab. B.11: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaných pomocí optimizéru AdaMax

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.93474	0.98229
Přesnost	Precision	0.56649	0.83108
Senzitivita	Recall	0.99675	0.99427
Specifická	Specificity	0.92895	0.98118
F1 míra	F1 score	0.72241	0.90538
AUC	AUC	0.962857	0.98772

Tab. B.12: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaných pomocí optimizéru Adam

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.84825	0.91661
Přesnost	Precision	0.35927	0.50539
Senzitivita	Recall	0.99703	0.99392
Specifická	Specificity	0.83439	0.909404
F1 míra	F1 score	0.52821	0.67006
AUC	AUC	0.91571	0.95166

Tab. B.13: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaných pomocí optimizéru Adagrad

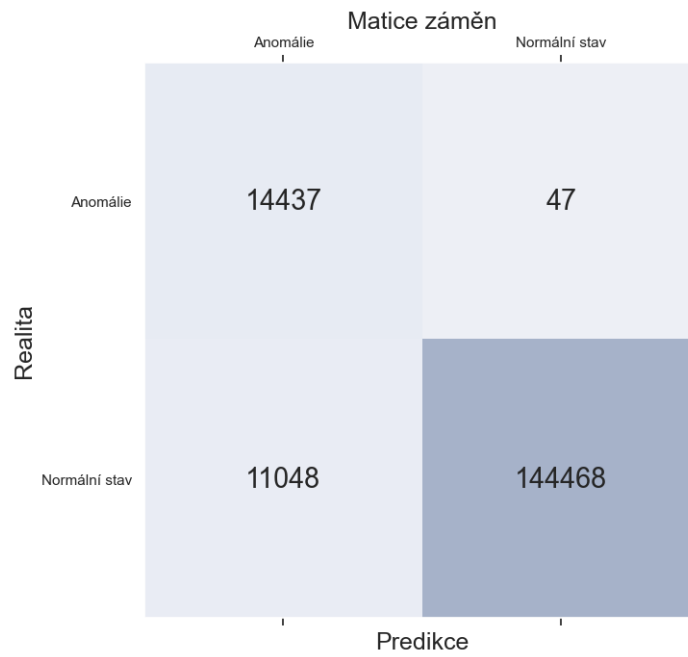
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.85405	0.88516
Přesnost	Precision	0.36800	0.40281
Senzitivita	Recall	0.99399	0.720796
Specifická	Specificity	0.84101	0.90047
F1 míra	F1 score	0.53714	0.51681
AUC	AUC	0.91750	0.81063

Tab. B.14: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaných pomocí optimizéru Adadelta

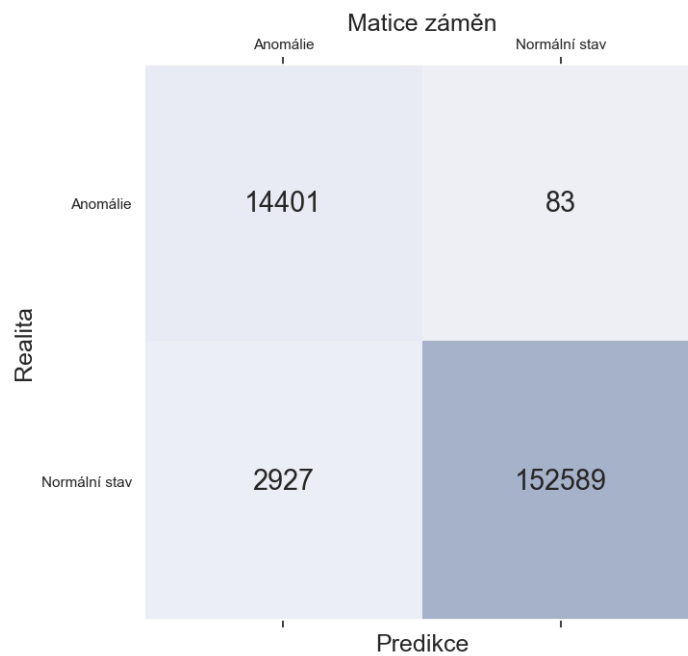
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.85666	0.88558
Přesnost	Precision	0.35629	0.40216
Senzitivita	Recall	0.84589	0.70498
Specifická	Specificity	0.85767	0.90240
F1 míra	F1 score	0.50140	0.51216
AUC	AUC	0.85178	0.80369

Tab. B.15: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaných pomocí optimizéru SGD

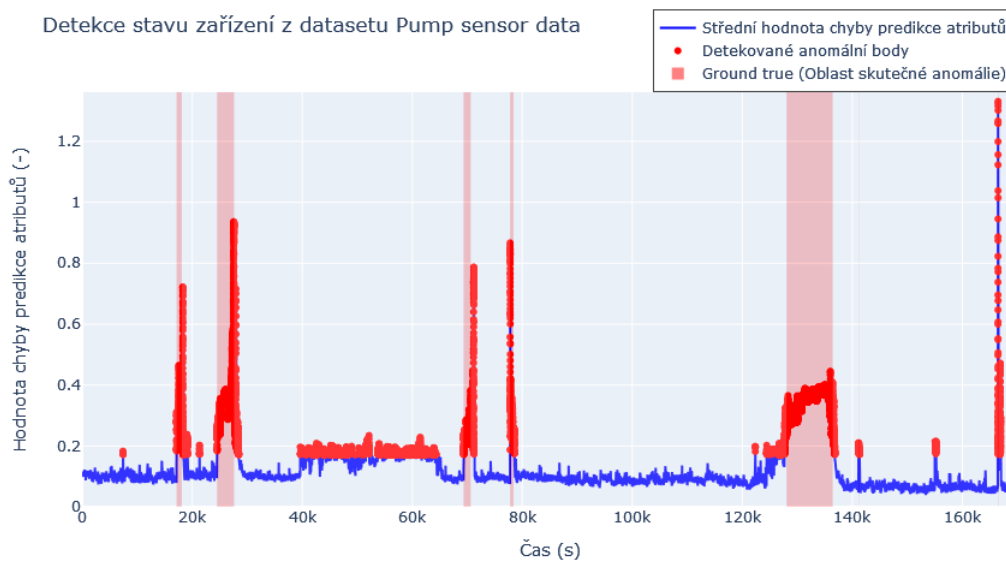
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.84825	0.85671
Přesnost	Precision	0.33514	0.34872
Senzitivita	Recall	0.76526	0.78590
Specifická	Specificity	0.85861	0.86330
F1 míra	F1 score	0.46614	0.48308
AUC	AUC	0.81193	0.82460



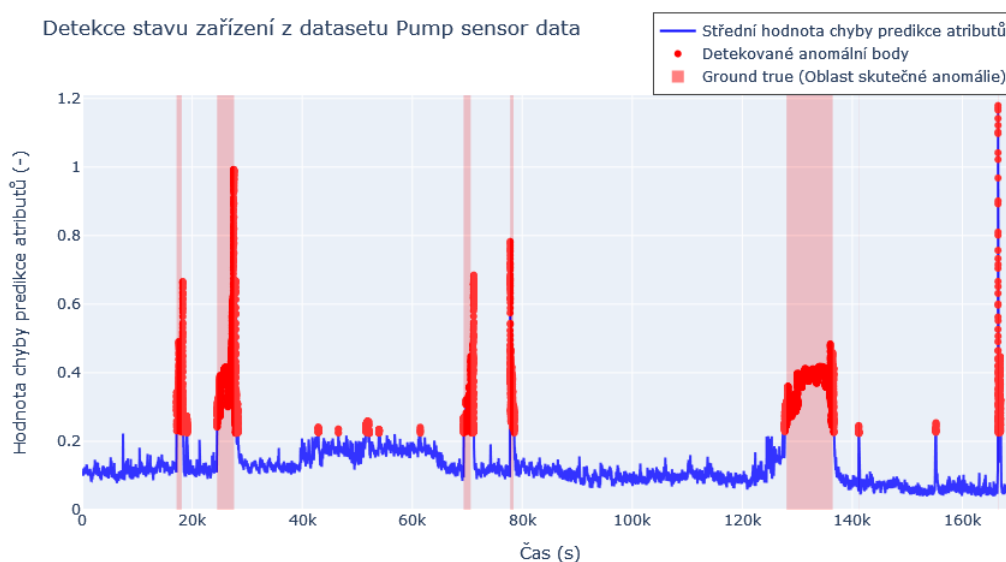
Obr. B.5: Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 1 trénované s optimizérem AdaMax



Obr. B.6: Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 2 trénované s optimizérem AdaMax



Obr. B.7: Detekce anomálního stavu zařízení z datasetu Pump sensor data s auto-
enkodérem 1 trénované s optimizérem AdaMax



Obr. B.8: Detekce anomálního stavu zařízení z datasetu Pump sensor data s auto-
enkodérem 2 trénované s optimizérem AdaMax

B.3 Vliv ztrátové funkce

Tab. B.16: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným se ztrátovou funkcí MSE

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.87005	0.90834
Přesnost	Precision	0.39488	0.48162
Senzitivita	Recall	0.98653	0.99206
Specifická	Specificity	0.85920	0.90055
F1 míra	F1 score	0.56401	0.64844
AUC	AUC	0.92287	0.94631

Tab. B.17: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným se ztrátovou funkcí MAE

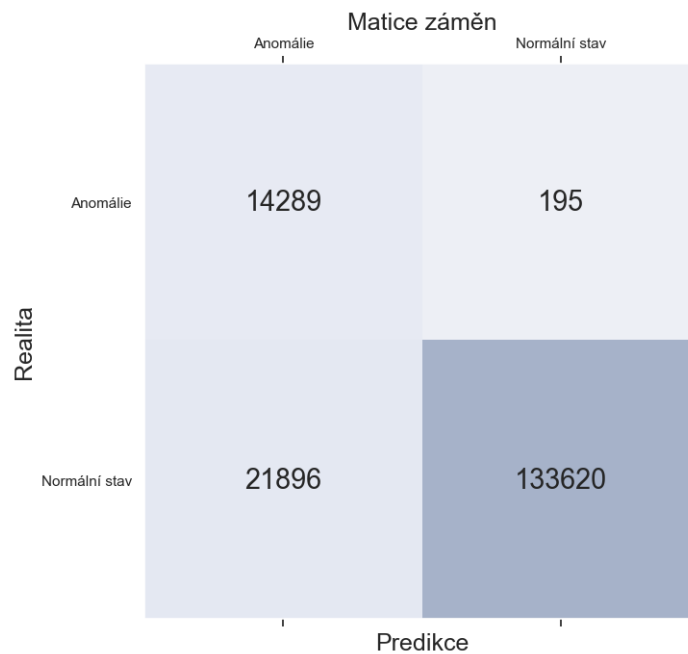
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.83597	0.98517
Přesnost	Precision	0.28116	0.89190
Senzitivita	Recall	0.59431	0.93993
Specifická	Specificity	0.85848	0.98939
F1 míra	F1 score	0.38173	0.91529
AUC	AUC	0.72639	0.96466

Tab. B.18: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným se ztrátovou funkcí Huber loss

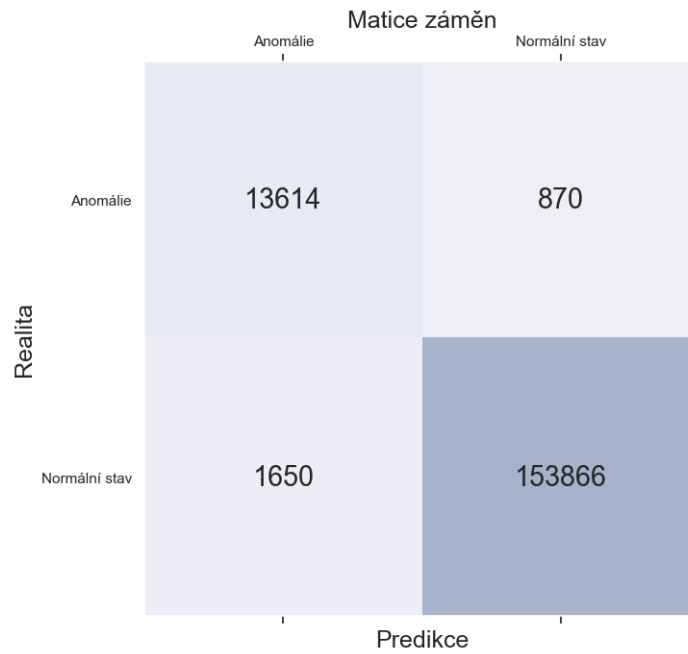
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.86459	0.91799
Přesnost	Precision	0.38585	0.50960
Senzitivita	Recall	0.99606	0.99441
Specifická	Specificity	0.85234	0.91088
F1 míra	F1 score	0.55623	0.67387
AUC	AUC	0.92420	0.95264

Tab. B.19: Tabulka metrik detektorů anomálií datasetu Pump sensor data trénovaným se ztrátovou funkcí KL-divergence

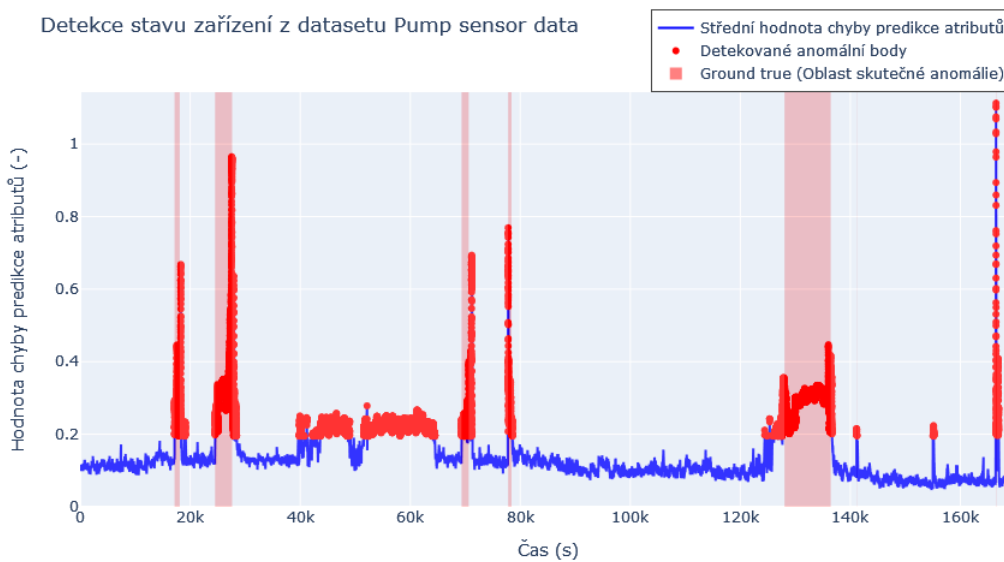
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.83758	0.83299
Přesnost	Precision	0.29346	0.27370
Senzitivita	Recall	0.64388	0.58064
Specifická	Specificity	0.85562	0.85649
F1 míra	F1 score	0.40316	0.37203
AUC	AUC	0.74975	0.71857



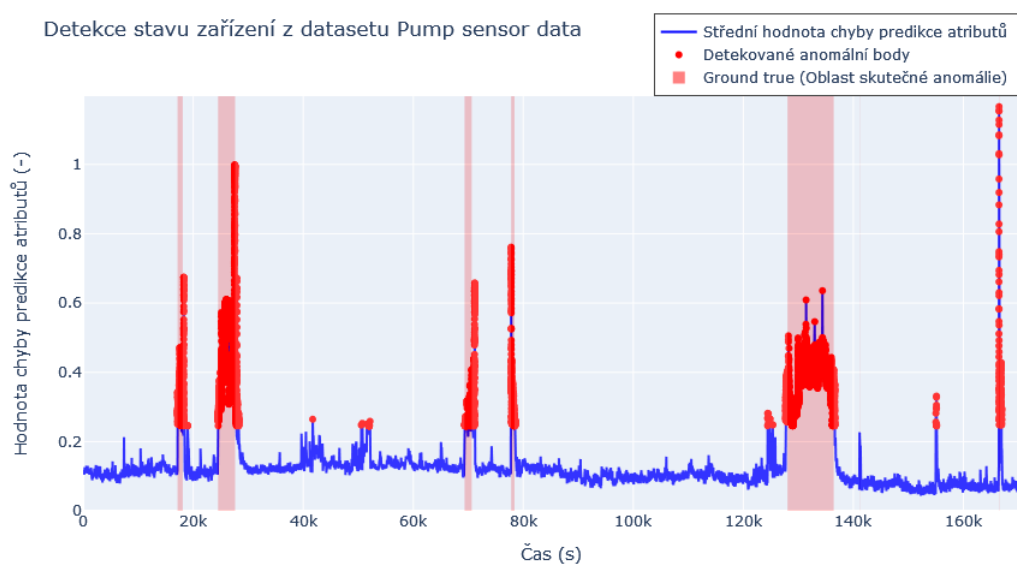
Obr. B.9: Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 1 trénované se ztrátovou funkcí MSE



Obr. B.10: Matice záměn detekce anomálií datasetu Pump sensor data s autoenkodérem 2 trénované se ztrátovou funkcí MAE



Obr. B.11: Detekce anomálního stavu zařízení z datasetu Pump sensor data s autoenkodérem 1 trénované se ztrátovou funkcí MSE



Obr. B.12: Detekce anomálního stavu zařízení z datasetu Pump sensor data s auto-
 enkodérem 2 trénované se ztrátovou funkcí MAE

C Citlivostní analýza - tabulky a grafy

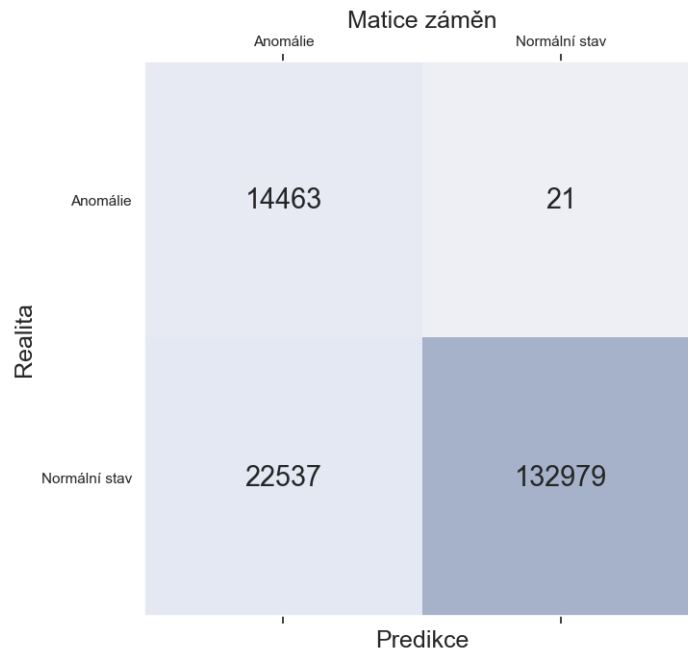
C.1 Vliv násobení konstantou

Tab. C.1: Tabulka metrik detektorů anomálií datasetu Pump sensor data zeslabená vynásobením konstantou 0.00001

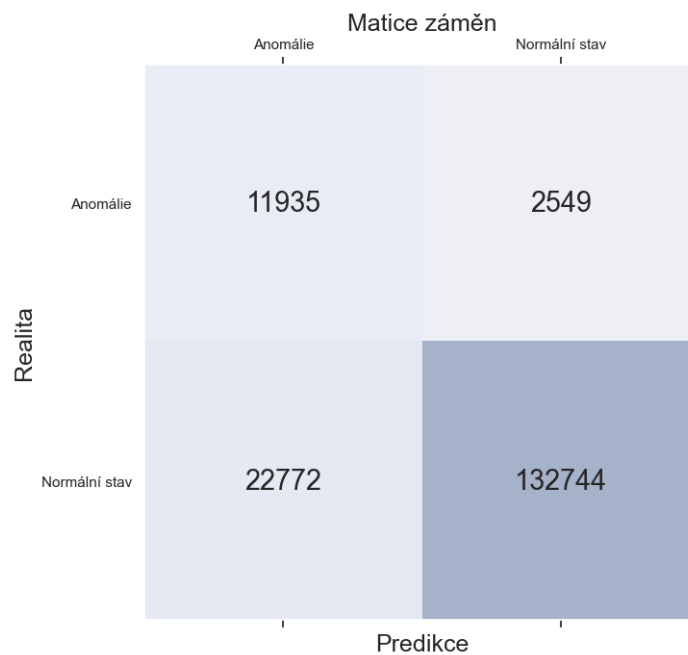
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.86731	0.88746
Přesnost	Precision	0.39089	0.43046
Senzitivita	Recall	0.99855	0.99309
Specifická	Specificity	0.85508	0.87763
F1 míra	F1 score	0.56184	0.60059
AUC	AUC	0.92682	0.93536

Tab. C.2: Tabulka metrik detektorů anomálií datasetu Pump sensor data zesílená vynásobením konstantou 10000

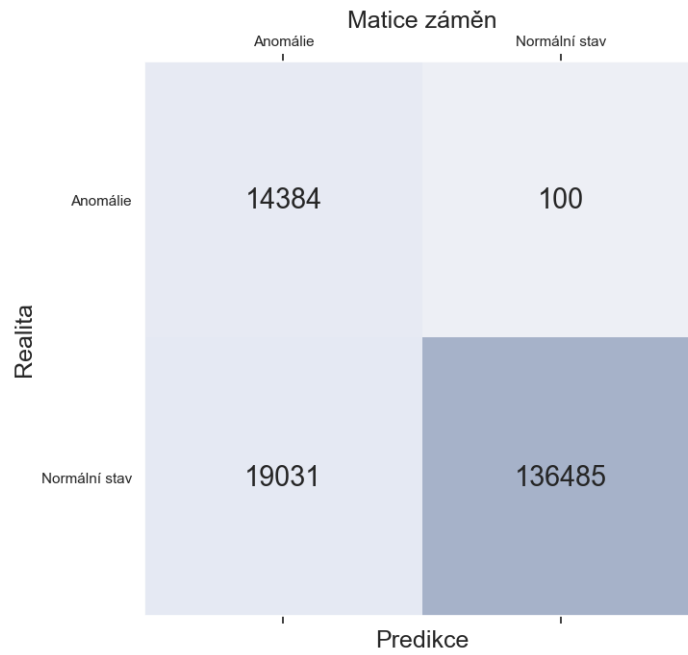
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.85105	0.94028
Přesnost	Precision	0.34387	0.58828
Senzitivita	Recall	0.82401	0.99655
Specifická	Specificity	0.85358	0.93504
F1 míra	F1 score	0.48525	0.73983
AUC	AUC	0.83879	0.96579



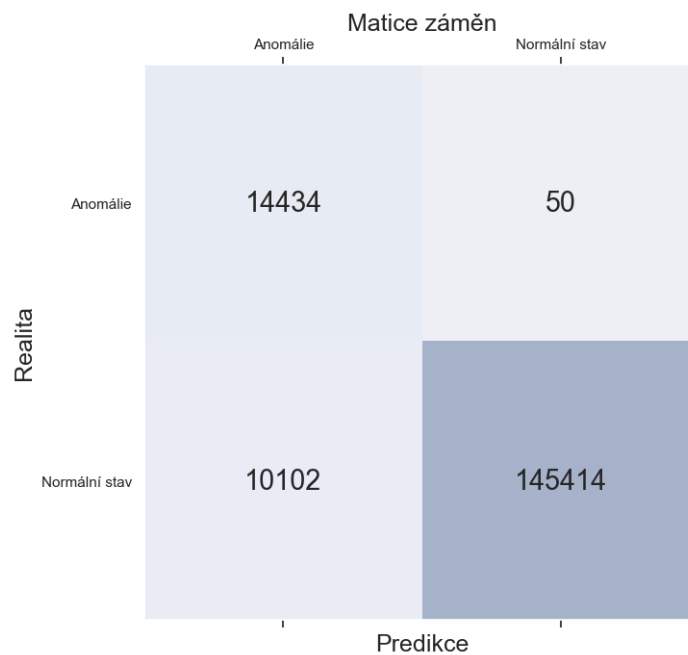
Obr. C.1: Matice záměn detekce anomálií datasetu Pump sensor data zeslabená vynásobením konstantou 0.00001 s autoenkodérem 1



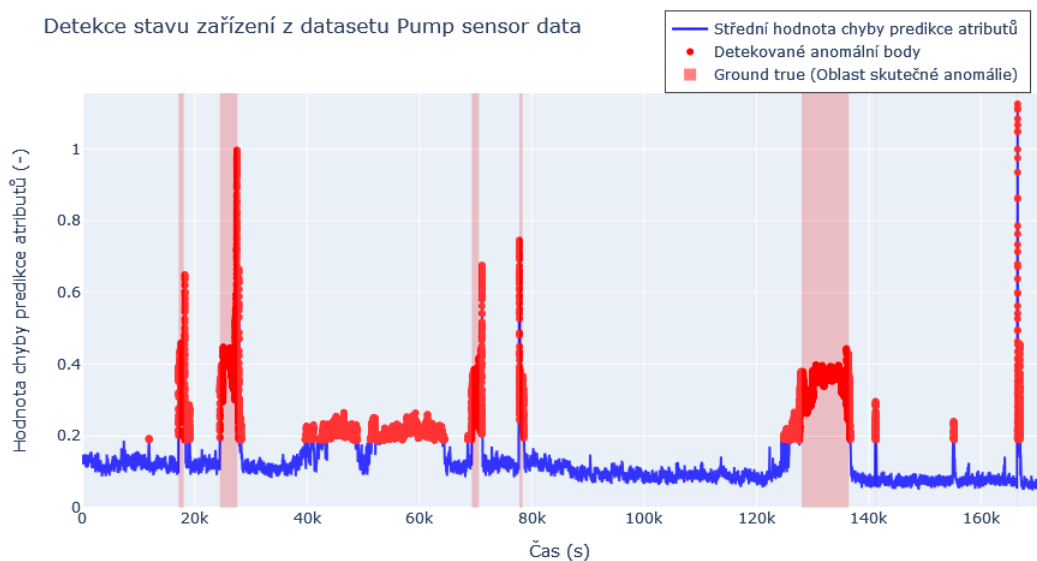
Obr. C.2: Matice záměn detekce anomálií datasetu Pump sensor data zesílená vynásobením konstantou 10000 s autoenkodérem 1



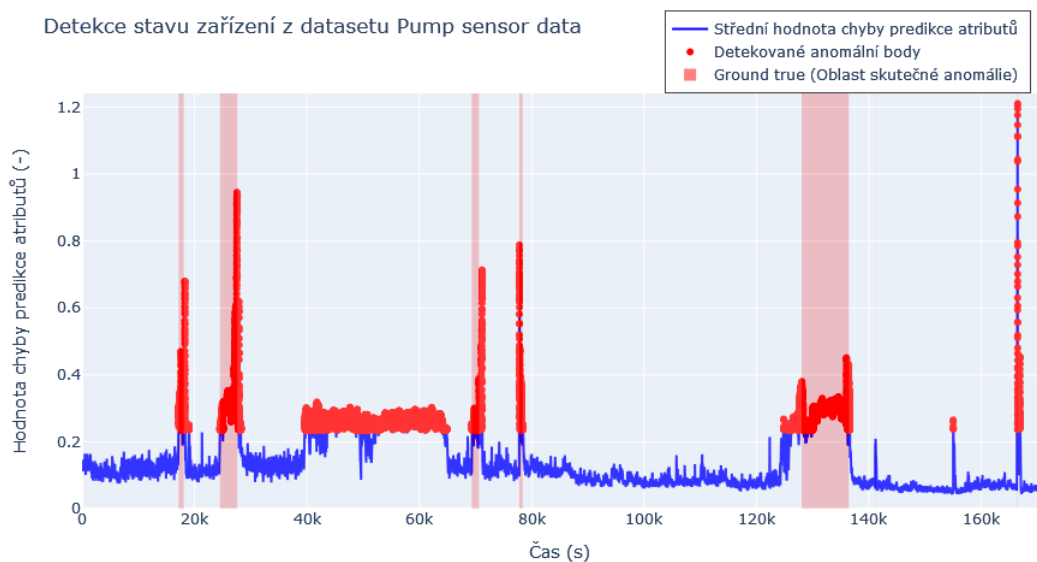
Obr. C.3: Matice záměn detekce anomálií datasetu Pump sensor data zeslabená vynásobením konstantou 0.00001 s autoenkodérem 2



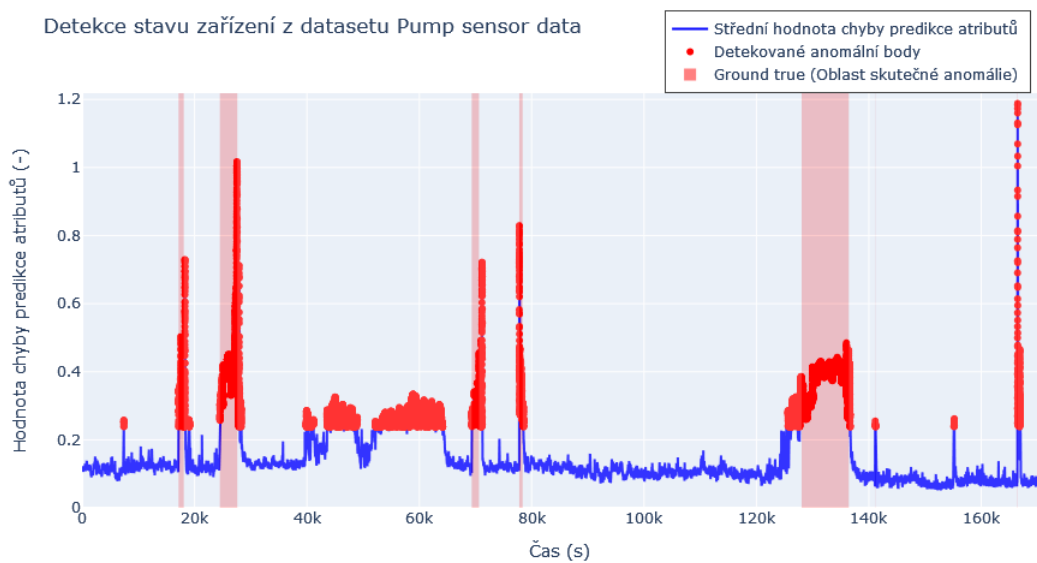
Obr. C.4: Matice záměn detekce anomálií datasetu Pump sensor data zesílená vynásobením konstantou 10000 s autoenkodérem 2



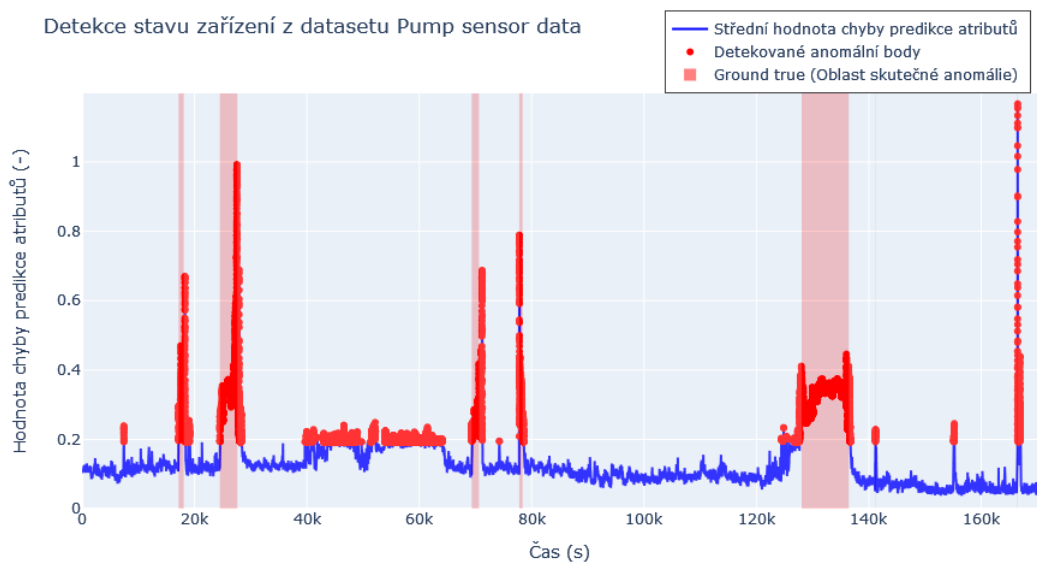
Obr. C.5: Detekce anomálního stavu zařízení z datasetu Pump sensor data zeslabená vynásobením konstantou 0.00001 s autoenkodérem 1



Obr. C.6: Detekce anomálního stavu zařízení z datasetu Pump sensor data zesílená vynásobením konstantou 10000 s autoenkodérem 1



Obr. C.7: Detekce anomálního stavu zařízení z datasetu Pump sensor data zeslabená vynásobením konstantou 0.00001 s autoenkodérem 2



Obr. C.8: Detekce anomálního stavu zařízení z datasetu Pump sensor data zesílená vynásobením konstantou 10000 s autoenkodérem 2

C.2 Vliv přidání šumu

Tab. C.3: Tabulka metrik detektorů anomálií datasetu Pump sensor data s přidáním šumem o SNR 50 dB

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.91748	0.89621
Přesnost	Precision	0.50802	0.45053
Senzitivita	Recall	0.99676	0.99344
Specifická	Specificity	0.91009	0.88716
F1 míra	F1 score	0.67302	0.61992
AUC	AUC	0.95343	0.94029

Tab. C.4: Tabulka metrik detektorů anomálií datasetu Pump sensor data s přidáním šumem o SNR 20 dB

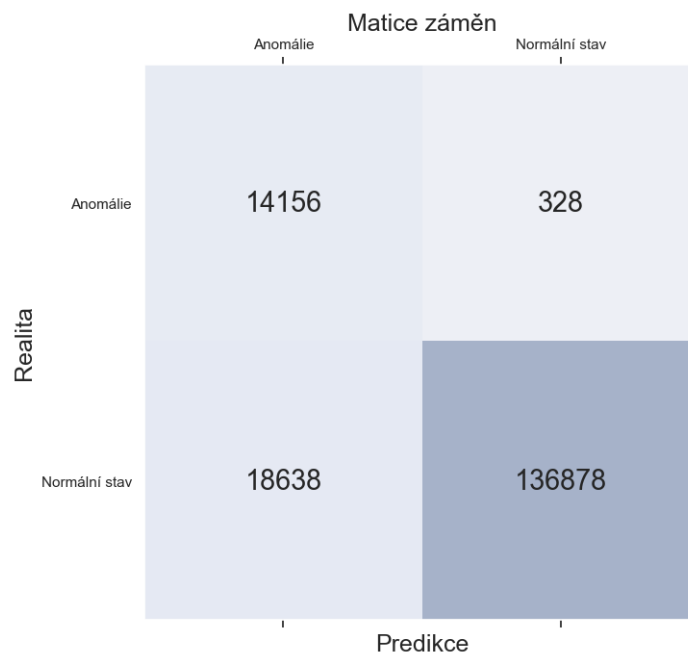
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.88843	0.89442
Přesnost	Precision	0.43166	0.44318
Senzitivita	Recall	0.97735	0.93282
Specifická	Specificity	0.88015	0.89084
F1 míra	F1 score	0.59884	0.60089
AUC	AUC	0.92875	0.91183

Tab. C.5: Tabulka metrik detektorů anomálií datasetu Pump sensor data s přidáním šumem o SNR 10 dB

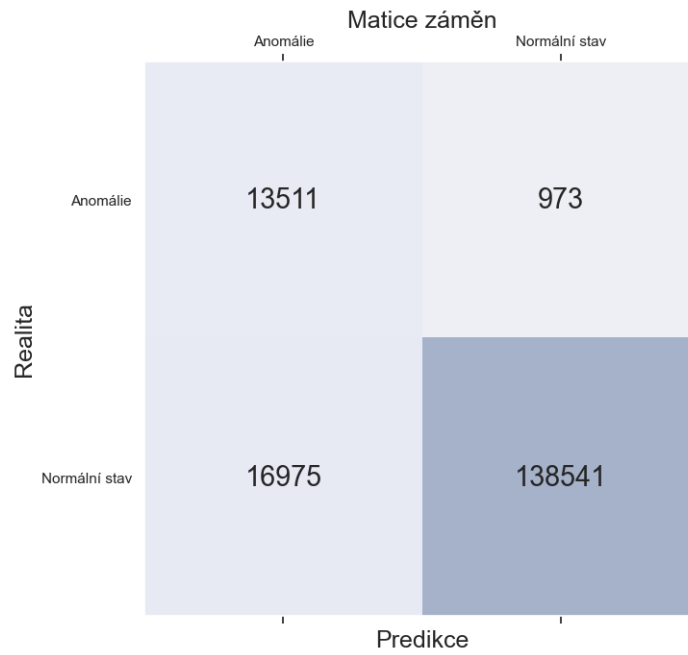
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.89929	0.93899
Přesnost	Precision	0.37927	0.64117
Senzitivita	Recall	0.28597	0.64485
Specifická	Specificity	0.95640	0.96639
F1 míra	F1 score	0.32608	0.64301
AUC	AUC	0.62119	0.80562

Tab. C.6: Tabulka metrik detektorů anomálií datasetu Pump sensor data s přidáním šumem o SNR 5 dB

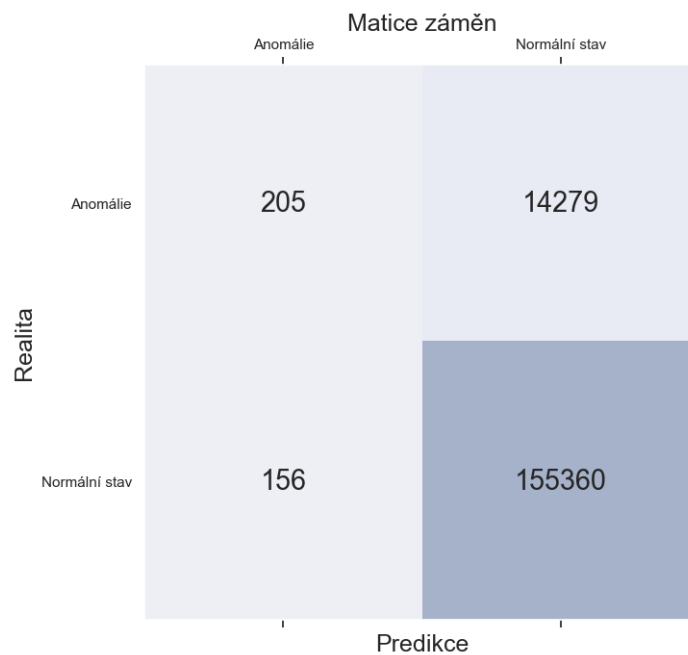
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.91509	0.91641
Přesnost	Precision	0.56786	0.64984
Senzitivita	Recall	0.01415	0.04087
Specifita	Specificity	0.99899	0.99795
F1 míra	F1 score	0.02762	0.07691
AUC	AUC	0.50658	0.51941



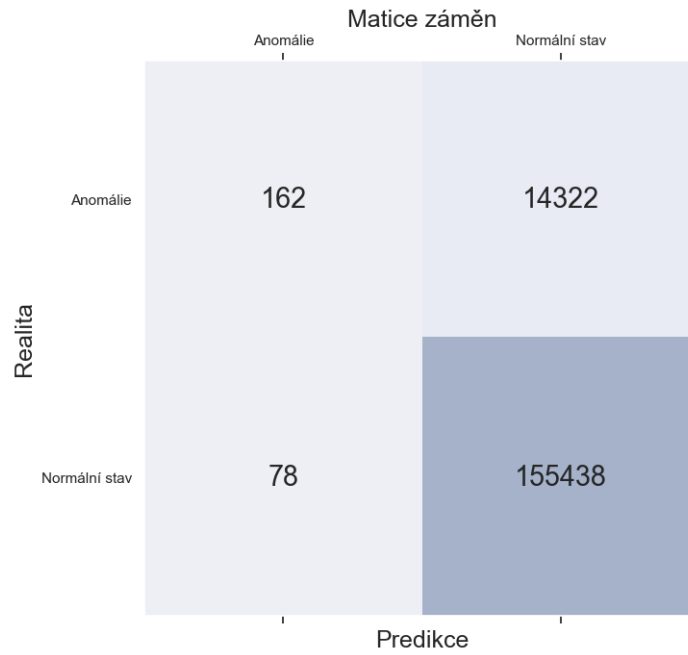
Obr. C.9: Matice záměn detekce anomálií datasetu Pump sensor data s přidáním šumem o SNR 20 dB pro autoenkodér 1



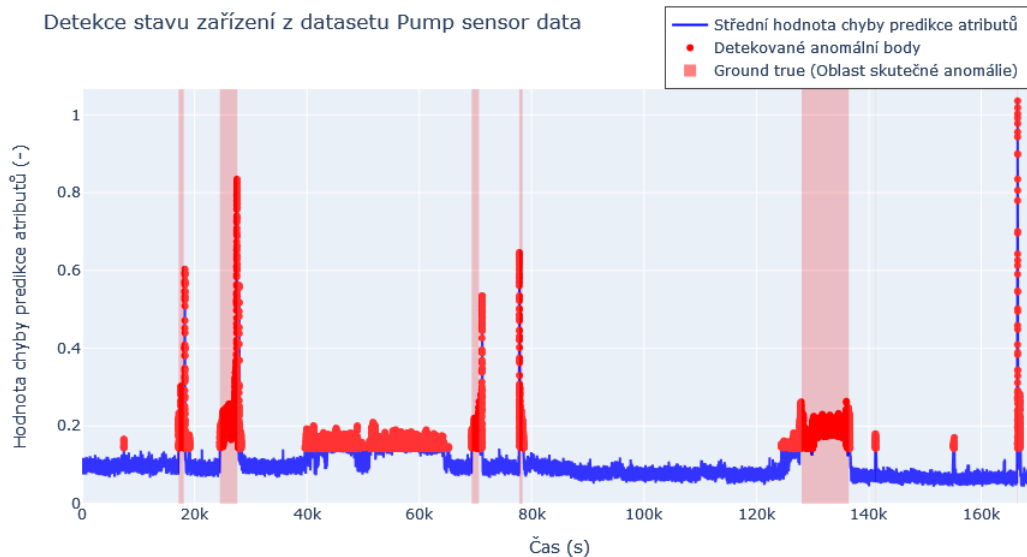
Obr. C.10: Matice záměn detekce anomálií datasetu Pump sensor data s přidaným šumem o SNR 20 dB pro autoenkodér 2



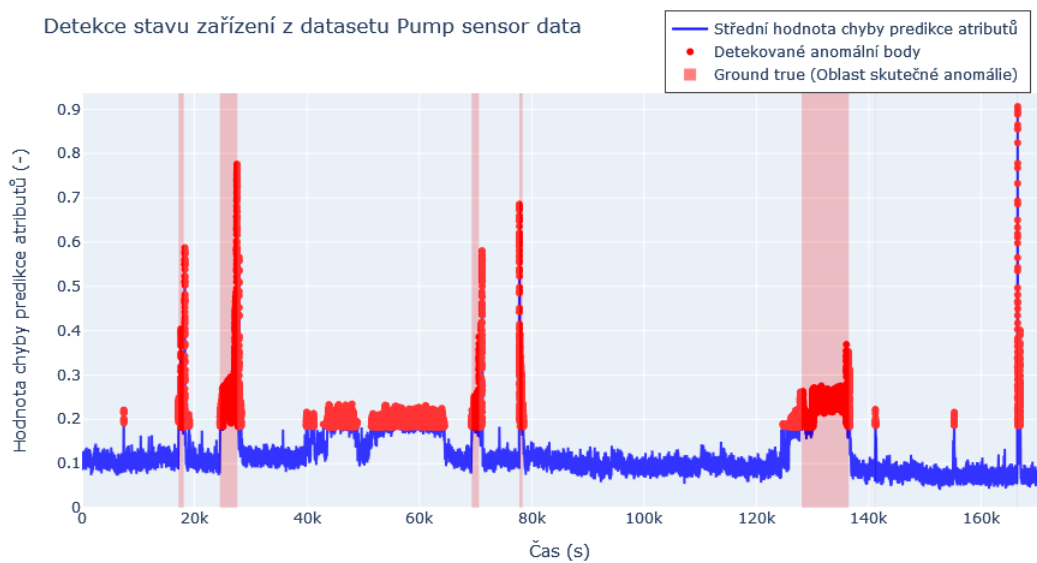
Obr. C.11: Matice záměn detekce anomálií datasetu Pump sensor data s přidaným šumem o SNR 5 dB pro autoenkodér 1



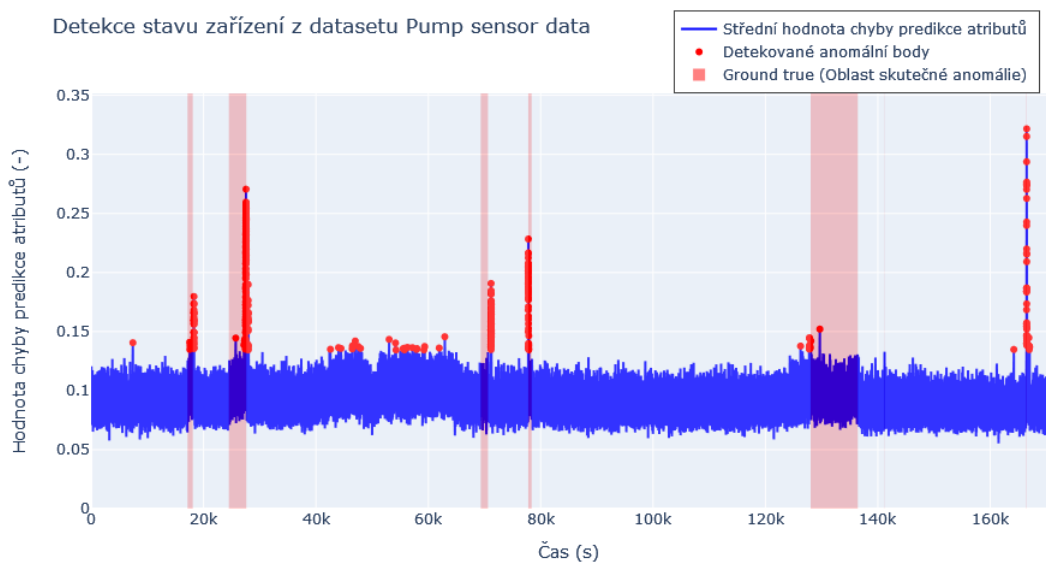
Obr. C.12: Matice záměn detekce anomálií datasetu Pump sensor data s přidaným šumem o SNR 5 dB pro autoenkodér 2



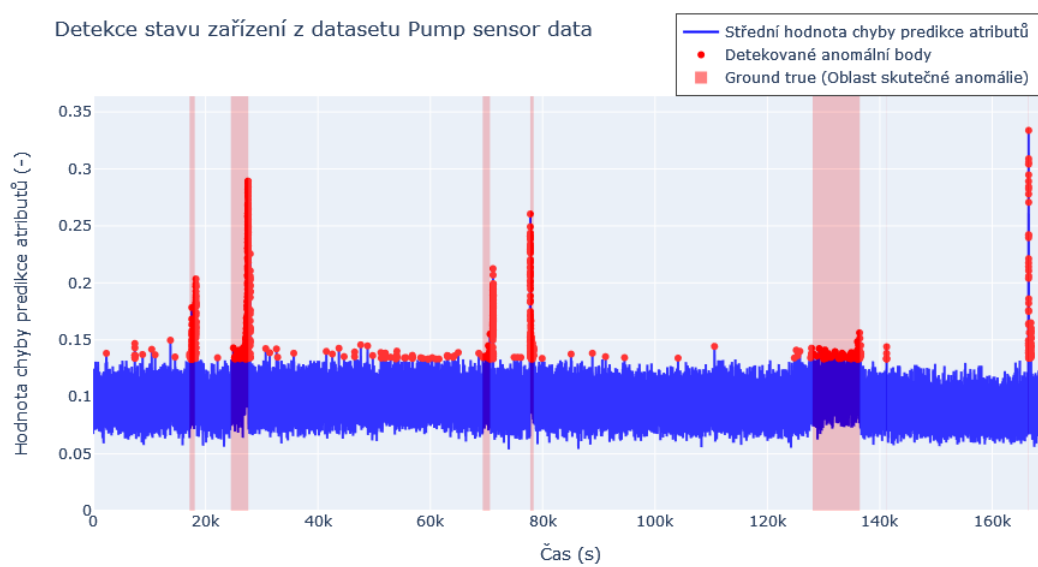
Obr. C.13: Detekce anomálního stavu zařízení z datasetu Pump sensor data s přidaným šumem o SNR 20 dB pro autoenkodér 1



Obr. C.14: Detekce anomálního stavu zařízení z datasetu Pump sensor data s přidáním šumem o SNR 20 dB pro autoenkodér 1



Obr. C.15: Detekce anomálního stavu zařízení z datasetu Pump sensor data s přidáním šumem o SNR 5 dB pro autoenkodér 1



Obr. C.16: Detekce anomálního stavu zařízení z datasetu Pump sensor data s přidáním šumem o SNR 5 dB pro autoenkodér 2

C.3 Vliv filtrace signálu

Tab. C.7: Tabulka metrik detektorů anomálií datasetu Pump sensor data filtrovaným filtrem plovoucího průměru s délkou okna 100 bodů

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.46716	0.71842
Přesnost	Precision	0.13785	0.23229
Senzitivita	Recall	1.0	1.0
Specifická	Specificity	0.41753	0.69219
F1 míra	F1 score	0.24230	0.37701
AUC	AUC	0.70876	0.84609

Tab. C.8: Tabulka metrik detektorů anomálií datasetu Pump sensor data filtrovaným filtrem plovoucího průměru s délkou okna 50 bodů

Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.44934	0.5278
Přesnost	Precision	0.13399	0.15285
Senzitivita	Recall	1.0	1.0
Specifická	Specificity	0.39805	0.48382
F1 míra	F1 score	0.23632	0.26517
AUC	AUC	0.69903	0.74191

Tab. C.9: Tabulka metrik detektorů anomálií datasetu Pump sensor data filtrovaným filtrem plovoucího průměru s délkou okna 20 bodů

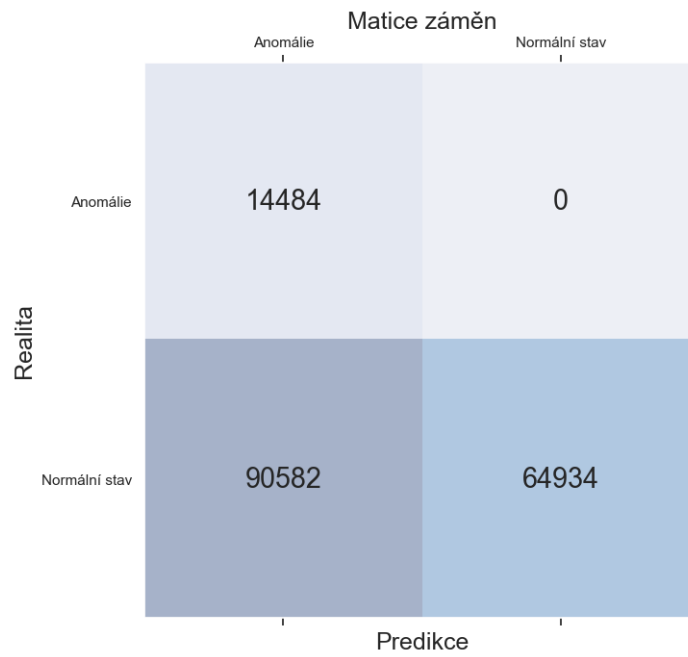
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.72721	0.83533
Přesnost	Precision	0.23799	0.34098
Senzitivita	Recall	1.0	1.0
Specifická	Specificity	0.701799	0.81999
F1 míra	F1 score	0.38448	0.50855
AUC	AUC	0.85089	0.90999

Tab. C.10: Tabulka metrik detektorů anomálií datasetu Pump sensor data filtrovaným filtrem plovoucího průměru s délkou okna 10 bodů

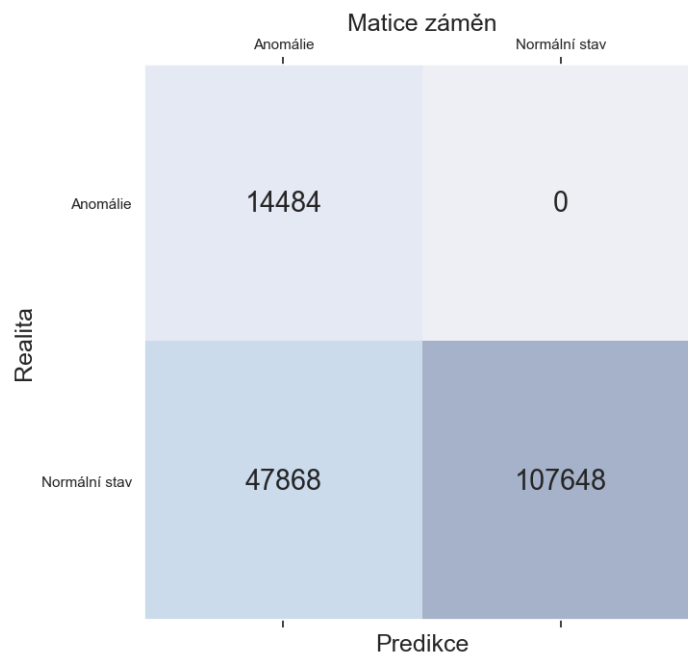
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.84951	0.85334
Přesnost	Precision	0.36133	0.36742
Senzitivita	Recall	0.99834	0.99951
Specifická	Specificity	0.83565	0.83973
F1 míra	F1 score	0.53062	0.53732
AUC	AUC	0.91699	0.919625

Tab. C.11: Tabulka metrik detektorů anomálií datasetu Pump sensor data filtrovaným filtrem plovoucího průměru s délkou okna 5 bodů

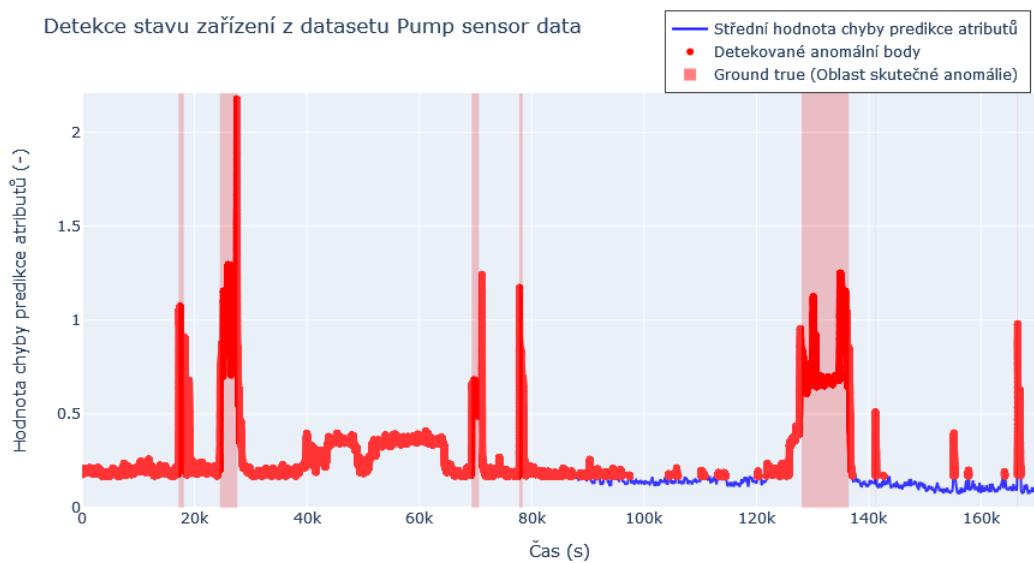
Název metriky	Anglický název	Autoenkodér 1 (nn)	Autoenkodér 2 (lstm)
Celková správnost	Accuracy	0.51350	0.83394
Přesnost	Precision	0.14903	0.33890
Senzitivita	Recall	1.0	0.99820
Specifická	Specificity	0.46819	0.81864
F1 míra	F1 score	0.25940	0.50601
AUC	AUC	0.73409	0.90842



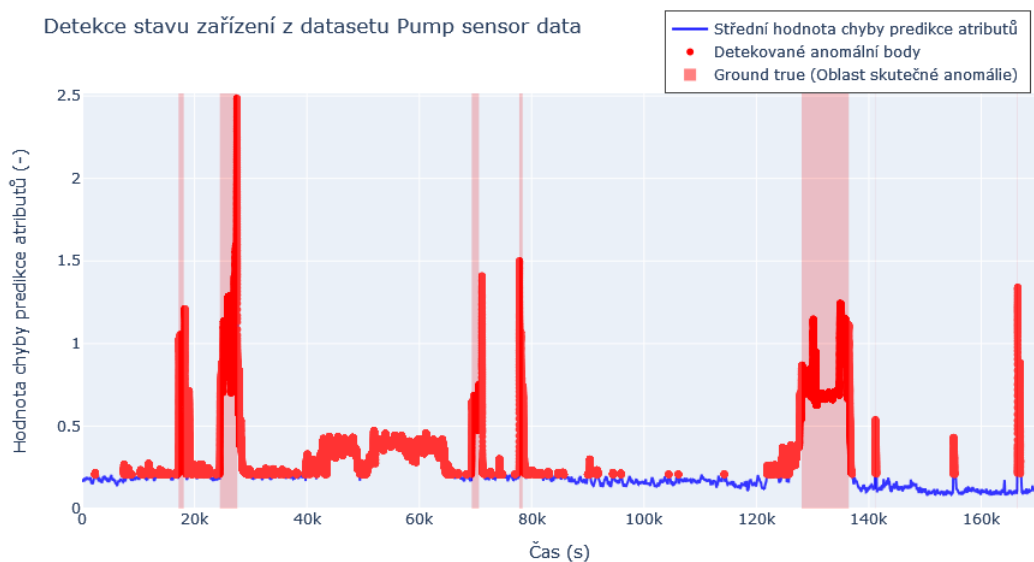
Obr. C.17: Matice záměn detekce anomálií datasetu Pump sensor data filtrovaná plovoucím průměrem o velikosti okna 100 pro autoenkodér 1



Obr. C.18: Matice záměn detekce anomálií datasetu Pump sensor data filtrovaná plovoucím průměrem o velikosti okna 100 pro autoenkodér 2



Obr. C.19: Detekce anomálního stavu zařízení z datasetu Pump sensor data filtrovaná plovoucím průměrem o velikosti okna 100 pro autoenkodér 1



Obr. C.20: Detekce anomálního stavu zařízení z datasetu Pump sensor data filtrovaná plovoucím průměrem o velikosti okna 100 pro autoenkodér 2

D Obsah elektronické přílohy

V této části bude popsán adresář přiloženého DVD.

- /.....kořenový adresář přiloženého archivu
 - 3_GutenTAG..... Skripty pro generování jednoduchého datasetu
 - dirichlet_config.py Skript pro generování datasetu s Dir. vlnkou
 - sine_config.py Skript pro generování datasetu se sinusovým signálem
 - square_config.py Skript pro generování datasetu s obdélníkovými pulzy
 - 4_Implementace Implementace vybraných metod (viz. kapitola 4)
 - CNN
 - GutenTAG Složka s GutenTAG datasetem
 - models2 Složka s natrénovanými modely
 - real_datasets Složka s datasetem
 - GutenTAG_ad.ipynb Skripty pro zobrazování výsledků s GutenTAG
 - Pump_dataset_CNN_train2.ipynb Skripty pro trénování modelů
 - Pump_sensor_CNN_ad.ipynb Skripty pro zobrazování výsledků
 - requirements.txt Požadované knihovny pro spuštění skriptu
 - LOF_IF
 - pump_sensor Složka s datasetem
 - IF_pump_data.ipynb Skripty pro model Isolation Forest
 - LOFF_pump_data_final.ipynb Skripty pro model Local outlier factor
 - requirements.txt Požadované knihovny pro spuštění skriptu
 - LSTM
 - GutenTAG Složka s GutenTAG datasetem
 - models Složka s natrénovanými modely
 - new_data Složka s datasetem
 - real_datasets Složka s datasetem
 - GutenTAG_ad.ipynb Skripty pro zobrazování výsledků s GutenTAG
 - GutenTAG_train.ipynb Skripty pro trénování modelů s GutenTAG
 - Pump_sensor_ad.ipynb Skripty pro zobrazování výsledků
 - Pump_sensor_graph.ipynb Skripty pro datasetu
 - Pump_sensor_train.ipynb Skripty pro trénování modelů
 - requirements.txt Požadované knihovny pro spuštění skriptu
 - Random_forest_Linear_reg
 - Custom_05_Pump_Sensors.ipynb Skripty pro model Random forest a Linear regresion
 - requirements.txt Požadované knihovny pro spuštění skriptu
 - README.txt Informace o požadovaném datasetu
 - 5_Autoenkoder Testy na autoenkodérech (viz. kapitola 5)
 - autoenkoder
 - model_xxx Složky s trénovanými modely
 - Pump_sensor_aug_xxx.ipynb Skripty pro úpravu základního datasetu
 - Pump_sensor_xxx_training.ipynb Skripty pro trénování modelů různých experimentů
 - Pump_sensor_xxx_graphs.ipynb Skripty pro zobrazování výsledků různých experimentů

├── ...	
├── obrázky Grafy a matice záměn z experimentů
├── ...	
├── 6_Aplikace GUI aplikace (viz. kapitola 6)
│ ├── data Data pro testování aplikace
│ │ ├── full_dataset.csv	
│ │ ├── test.csv	
│ │ └── train.csv	
│ ├── src Zdrojové soubory aplikace
│ │ ├── controller.py	
│ │ ├── gui.py	
│ │ ├── matplotlib_widget.py	
│ │ └── tf_logger.py	
│ ├── venv Virtuální prostředí pro spuštění aplikace
│ ├── main.py Skript pro spuštění aplikace
│ ├── model_e1.h5 Model pro testování aplikace
│ ├── model_e2.h5 Model pro testování aplikace
│ └── requirements.txt Požadované knihovny pro spuštění aplikace
├── Dataset	
│ ├── sensor.csv Dataset
│ └── new_sensor.csv Dataset bez Nan hodnot

Poznámka: Složky s datasetem většinou obsahují pouze README.txt, kde je popsáno, který soubor je nutné do dané složky umístit ze složky Dataset v kořenovém adresáři. Tato úprava byla provedena z důvodu šetření místa na DVD disku.