

Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky

Klasifikácia zákrytových premenných hviezd pomocou hlbokého učenia

Bakalárska práca

2022

Maximilián Revický

Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky

Klasifikácia zákrytových premenných hviezd pomocou hlbokého učenia

Bakalárska práca

Študijný program: Hospodárska informatika
Študijný odbor: Informatika
Školiace pracovisko: Katedra kybernetiky a umelej inteligencie (KKUI)
Školiteľ: doc. Ing. Peter Butka, PhD.
Konzultant: Ing. Viera Maslej Krešňáková, PhD.

Košice 2022

Maximilián Revický

Abstrakt v SJ

Jedným z problémov v astrofyzikálnom výskume je veľký objem dát, ktorý produkujú nové pozorovacie prístroje. Na uľahčenie ich spracovania sú čoraz častejšie využívané technológie umelej inteligencie. Cieľom tejto práce bola klasifikácia zákrytových premenných hviezd za použitia zvolených metód hlbokého učenia. Problém deficitu anotovaných dát pri klasifikácii svetelných kriviek zákrytových premenných hviezd bol riešený vygenerovaním syntetických dát pomocou softvéru Elisa. Práca predstavuje teoretický prehľad tejto problematiky. Analyzuje súčasný stav porovnaním dostupných riešení. Praktická časť postupuje podľa metodológie CRISP-DM. Ako najlepší bol aplikovaný model, ktorý vykázal výsledky na observačných dátach na úrovni 97%. Model dosiahol tento výsledok pomocou využitia 1D konvolučnej a LSTM rekurentnej neurónovej siete.

Kľúčové slová

hlboké učenie, astrofyzika, zákrytové hviezdy, klasifikácia

Abstrakt v AJ

The vast volume of data produced by new observational equipment is one of the important problems in astrophysics. Artificial intelligence technology is increasingly being used to assist them in processing information. The primary goal of this research was to use specific deep learning approaches to classify eclipsing binary stars. The problem of availability of annotated data for the classification of light curves of eclipsing binary stars was solved by generating synthetic data with the Elisa software. This thesis provides a theoretical overview of the subject as well as an overview of available solutions. We used the CRISP-DM methodology during the design and implementation of our solution. On observational data, the best model achieved 97% accuracy. A 1D convolutional and a LSTM recurrent neural network were used to achieve this result.

Klíčové slová v AJ

deep learning, astrophysics, eclipsing stars, classification

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY
Katedra kybernetiky a umelej inteligencie

ZADANIE
BAKALÁRSKEJ PRÁCE

Študijný odbor: **Informatika**
Študijný program: **Hospodárska informatika**

Názov práce:

Klasifikácia zákrytových premenných hviezd pomocou hlbokého učenia

Classification of eclipsing binary stars using deep learning

Študent: **Maximilián Revický**
Školiteľ: **doc. Ing. Peter Butka, PhD.**
Školiace pracovisko: **Katedra kybernetiky a umelej inteligencie**
Konzultant práce: **Ing. Viera Maslej Krešňáková, PhD.**
Pracovisko konzultanta: **Katedra kybernetiky a umelej inteligencie**

Pokyny na vypracovanie bakalárskej práce:

1. Podat' teoretický prehľad problematiky metód hlbokého učenia, s dôrazom na vhodné metódy pre automatickú klasifikáciu zákrytových premenných hviezd.
2. Analyzovať a predspracovať dostupnú dátovú množinu svetelných kriviek zákrytových premenných hviezd, navrhnúť postupy pre ich klasifikáciu pomocou zvolených metód hlbokého učenia.
3. Realizovať a vyhodnotiť navrhnuté postupy klasifikácie na danej množine dát.
4. Vypracovať dokumentáciu podľa pokynov katedry a vedúceho práce.

Jazyk, v ktorom sa práca vypracuje: slovenský
Termín pre odovzdanie práce: 27.05.2022
Dátum zadania bakalárskej práce: 29.10.2021



prof. Ing. Liberios Vokorokos, PhD.

dekan fakulty

Čestné vyhlásenie

Vyhlasujem, že som bakalársku prácu vypracoval samostatne s použitím uvedenej odbornej literatúry.

Košice 27. 5. 2022

.....

Vlastnoručný podpis

Podakovanie

Chcel by som sa poďakovať môjmu školiteľovi doc. Ing. Petrovi Butkovi, PhD. za cenné rady a pripomienky k práci. Taktiež veľká vďaka patrí konzultantke Ing. Viere Maslej Krešňákovovej, PhD. za čas, ochotu a trpezlivosť pri konzultáciách. Taktiež by som chcel poďakovať aj doc. Mgr. Štefanovi Parimuchovi, PhD. z UPJŠ za poskytnutie datasetov a spoluprácu.

Obsah

1	Hlboké učenie	3
1.1	Perceptrón	4
1.2	Aktivačné funkcie	4
1.2.1	ReLU	5
1.2.2	Softmax	6
1.3	Chybové funkcie	6
1.3.1	Krížová entropia	7
1.4	Algoritmus spätného šírenia chýb	8
1.5	Optimalizácia	8
1.5.1	Adam	8
1.6	Regularizácia	9
1.6.1	Dropout	10
1.7	Konvolučné neurónové siete	10
1.8	Rekurentné neurónové siete	13
1.8.1	Long short-term memory – LSTM	14
1.8.2	Gated Recurrent Unit – GRU	15
2	Analýza súčasného stavu – klasifikácia svetelných kriviek premen- ných hviezd	16
2.1	Svetelné krivky	16
2.2	Analýza súčasného stavu	19
3	Klasifikácia zákrytových premenných hviezd pomocou metód hlbo- kého učenia	21
3.1	Vybrané technológie	21
3.2	Hardvér	22
3.3	Metodika práce	22
3.4	Pochopenie cieľa	24

3.5	Pochopenie dát	24
3.6	Príprava dát	27
3.6.1	Syntetické dáta	27
3.6.2	Observačné dáta	29
3.7	Modelovanie	29
3.7.1	Model A	30
3.7.2	Model B	32
3.7.3	Tréning	34
3.7.4	Kritéria úspešnosti	34
3.7.5	Vyhodnotenie experimentov na testovacej množine syntetic- kých dát	36
3.8	Vyhodnotenie	37
3.9	Nasadenie	39
4	Záver	40
	Zoznam príloh	47
	Príloha A	48

Zoznam obrázkov

1–1	Príklad hlbkej neurónovej siete	3
1–2	Aplikovanie aktivačnej funkcie (Kuri-Morales, 2017) - upravené . .	5
1–3	Grafické zobrazenie aktivačnej funkcie ReLU	6
1–4	Grafické zobrazenie aktivačnej funkcie Softmax	6
1–5	Zobrazenie podučenej, optimálne natrénovanej a preučenej neurónovej siete (Trivedi et al., 2021) - upravené	9
1–6	Princíp regularizácie učenia pomocou dropoutu	10
1–7	Príklad konvolúcie	11
1–8	Príklad ukážky CNN (Alamsyah et al., 2019) - upravené	12
1–9	Hlboká CNN architektúra. (Hore) 1D CNN architektúra. (Dole) 2D CNN architektúra (Kussul et al., 2017)	13
1–10	Model architektúr jednoduchej RNN, LSTM a GRU (Sajjad et al., 2020) - upravené	14
2–1	Ukážka oddeleného binárneho systému, polodotykového binárneho systému a dotykového binárneho systému	18
3–1	CRISP DM	23
3–2	Vizualizácia dotykových (vľavo) a oddelených (vpravo) kriviek . . .	26
3–3	Vizualizácia dotykových (vľavo) a oddelených (vpravo) kriviek s kritickým sklonom dráhy	26
3–4	Vizualizácia škvrnitých hviezd	26
3–5	Vizualizácia Modelu A	31
3–6	Vizualizácia Modelu B	33
3–7	Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky v prípade klasifikácie dvojhviezd na oddelené a dotykové systémy . .	38
3–8	Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky v prípade klasifikácie na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy	39

A – 1	Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 1	48
A – 2	Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 2a (filter U)	48
A – 3	Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 2b (filter B)	49
A – 4	Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 2c (filter V)	50
A – 5	Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 2d (filter R)	50
A – 6	Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 2e (filter I)	51
A – 7	Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 3	51
A – 8	Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 4	52

Zoznam tabuliek

3–1	Hardwarové špecifikácie servera	22
3–2	Dataseťy syntetických dát	25
3–3	Dataseťy použité pri klasifikácii dvojhviezd na oddelené a dotykové	27
3–4	Dataseťy použité pri klasifikácii na dvojhviezdy s kritickým sklo- nom dráhy a škvrnité hviezdy	27
3–5	Ukážka jedného riadku z tabuľky Curves	28
3–6	Ukážka jedného riadku z tabuľky Parameters	29
3–7	Zhrnutie modelovania klasifikácie dvojhviezd na dotykové a odde- lené systémy	30
3–8	Zhrnutie modelovania klasifikácie na dvojhviezdy s kritickým sklo- nom dráhy a škvrnité hviezdy	30
3–9	Zhrnutie Modelu A	32
3–10	Zhrnutie Modelu B	34
3–11	Zhrnutie výsledkov klasifikácie dvojhviezd na oddelené a dotykové systémy	38
3–12	Zhrnutie výsledkov klasifikácie na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy	38
A – 1	Zhrnutie výsledkov Experimentu 1 na testovacej množine	48
A – 2	Zhrnutie výsledkov Experimentu 2a (filter U) na testovacej množine	49
A – 3	Zhrnutie výsledkov Experimentu 2b (filter B) na testovacej množine	49
A – 4	Zhrnutie výsledkov Experimentu 2c (filter V) na testovacej množine	49
A – 5	Zhrnutie výsledkov Experimentu 2d (filter R) na testovacej množine	50
A – 6	Zhrnutie výsledkov Experimentu 2e (filter I) na testovacej množine	51
A – 7	Zhrnutie výsledkov Experimentu 3 na testovacej množine	52
A – 8	Zhrnutie výsledkov Experimentu 4 na testovacej množine	52

Zoznam symbolov a skratiek

1D One-dimensional

2D Two-dimensional

Adam Adaptive Moment Estimation

AI Artificial Inteligance

BiLSTM Bidirectional Long Short-Term Memory

CNN Convolutional Neural Network

CRISP-DM Cross-industry Standard Process for Data Mining

DL Deep Learning

DNN Deep Neural Network

FN False Negative

FP False Positives

GAN Generative Adversarial Network

GPU Graphics Processing Unit

GRU Gated Recurrent Unit

HiTS High cadence Transient Survey

HW Hardware

LSTM Long Short-Term Memory

ML Machine Learning

MNRAS Monthly Notices of the Royal Astronomical Society

ReLU Rectifier Linear Unit

RNN Recurrent Neural Network

ROC Receiver Operating Characterist

TN True Negatives

TP True Positives

Úvod

Astronómia zaznamenala za posledné desaťročie exponenciálny nárast zozbieraných dát vďaka teleskopom s automatizovanými digitálnymi výstupmi. Na spracovanie sa v minulosti využívala náročná manuálna analýza. V dnešnej dobe aj astronómovia prijímajú rôzne pokročilé metódy na uľahčenie výskumu. Práve nadštandardný objem dát spôsobuje nutnosť využitia metód hlbokého učenia pri výskume. V súčasnosti sa v súvislosti s novými technikami spracovania údajov používajú okrem hlbokého učenia aj umelá inteligencia a strojové učenie.

Umelá inteligencia (skr. AI, angl. *Artificial Inteligance*) je akákoľvek technika, ktorej cieľom je umožniť počítačom napodobňovať ľudské správanie. Zahŕňa tak oblasti vrátane strojového učenia, spracovania prirodzeného jazyka, syntézy jazyka, počítačového videnia, robotiky, analýzy senzorov, optimalizácie a simulácie.

Strojové učenie (skr. ML, angl. *Machine Learning*) je podmnožina techník umelej inteligencie, ktorá umožňuje počítačovým systémom učiť sa z predchádzajúcich skúseností, pozorovaní údajov a zlepšovať svoje správanie pri danej úlohe.

Hlboké učenie (skr. DL, angl. *Deep Learning*) je podmnožinou strojového učenia. Typickými architektúrami DL sú hlboké neurónové siete (skr. DNN, angl. *Deep Neural network*), konvolučné neurónové siete (skr. CNN, angl. *Convolutional Neural Network*), rekurentné neurónové siete (skr. RNN, angl. *Recurrent Neural Network*), generatívne adverzné siete (skr. GAN, angl. *Generative Adversarial Network*) a mnohé ďalšie.

Na trénovanie neurónových sietí je požiadavka nadpriemerného počtu dát. V niektorých oblastiach astrofyziky je problém získať taký objem označených (anotovaných) dát, tak ako aj v našom prípade. Táto práca bola riešená v spolupráci s tímom astrofyzikov z Univerzity Pavla Jozefa Šafárika, ktorí nám dodali potrebné dáta. Trénovacie dáta vznikli na základe fyzikálneho modelu vygenerovaním pomocou softvéru Elisa pre simuláciu svetelných kriviek rôznych typov zákrytových premenných hviezd. Testovacie dáta sú observačné, teda dáta zozbierané z reálnych

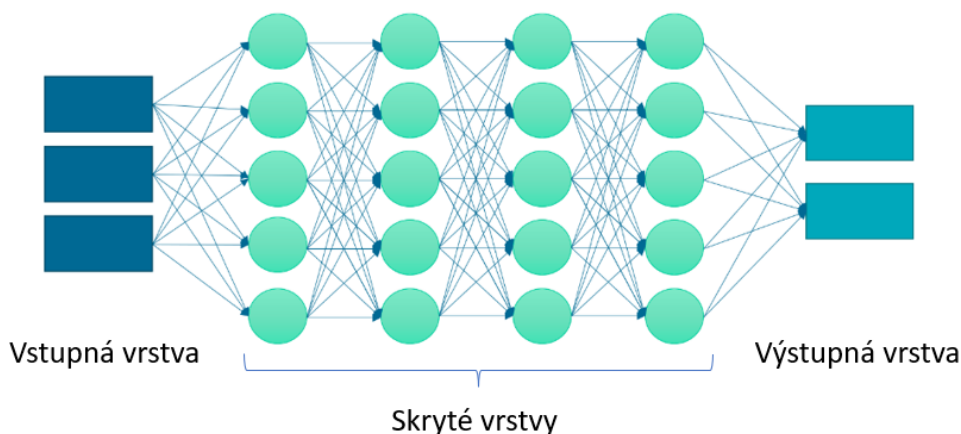
pozorovaní. V tejto bakalárskej práci riešime automatickú klasifikáciu pomocou hlbokého učenia. Klasifikujeme svetelné krivky zákrytových premenných hviezd pomocou konvolučných a rekurentných neurónových sietí.

Práca je členená do 3 kapitol. V kapitole 1 sa venujeme teoretickému prehľadu o hlbokom učení. Popíšeme si základnú štruktúru neurónových sietí, aktivačných funkcií, metódy učenia a optimalizácie, ako aj typy architektúr ako konvolučné a rekurentné neurónové siete. V kapitole 2 predstavíme základnú teóriu zákrytových svetelných hviezd. Poskytneme porovnanie a analýzu podobných riešení. V kapitole 3 sa zameriame na technológie a hardvér používaný pri našej práci. Ďalej predstavíme metodiku CRISP-DM, podľa ktorej postupujeme pri riešení klasifikačných úloh. Súčasťou metodiky je modelovanie pomocou konvolučných a rekurentných sietí, ako aj experimenty a ich vyhodnotenie.

1 Hlboké učenie

Hlboké učenie napodobňuje činnosť ľudského mozgu pri spracovávaní údajov a vytváraní vzorcov pre rozhodovanie. Dnes sa tento prístup používa v množstve rôznych aplikácií, ako napríklad na generovanie titulkov k videám, rozpoznávanie reči, rozpoznávanie tváre na fotografiách a pri samozajazdiacich autách.

Základom modelov hlbokého učenia je realizácia architektúr na báze neurónových sietí. Tieto sa často skladajú z vrstiev usporiadaných do rôznych typov architektúr. Tradične sa neurónová sieť skladá z troch typov vrstiev. Medzi vstupnou a výstupnou vrstvou sa nachádzajú skryté vrstvy. Základná neurónová sieť má jednu skrytú vrstvu, zatiaľ čo sieť s hlbokým učením má viac skrytých vrstiev (pozri obr. 1 – 1), čo pomôže zvýšiť presnosť siete a dosiahnuť zložitejšie klasifikácie dát. Na presnosť výsledku má vplyv množstvo dát, čím viac dát algoritmus dostane, tým bude presnejší. Viac vrstiev znamená aj to, že model bude vyžadovať viac parametrov a aj viac výpočtových zdrojov.



Obrázok 1 – 1 Príklad hlbkej neurónovej siete

Tréning pomocou veľkého množstva dát konfiguruje neuróny v neurónovej sieti. Výsledkom je natrénovaný model neurónovej siete, ktorý vyhodnocuje nové dáta. Modely hlbokého učenia prijímajú informácie z viacerých zdrojov dát a analyzujú dáta v reálnom čase bez potreby ľudského zásahu. Pri hlbokom učení je vhodné na

trénovanie modelov použiť grafické procesory (skr. GPU), pretože dokážu výrazne skrátiť čas tréningu v porovnaní s klasickým centrálnym procesorom počítača (Godfellow et al., 2016; Chollet, 2021; Oracle, n.d.).

1.1 Perceptrón

Perceptrón je jednoduchý binárny klasifikačný model, ktorý navrhol Frank Rosenblatt z Cornellovej univerzity. Pomáha rozdeliť súbor vstupných signálov na dve časti. Na rozdiel od mnohých iných klasifikačných prístupov bol však perceptrón modelovaný podľa základnej jednotky ľudského mozgu (neurónu). Perceptrón je veľmi jednoduchý príklad modelu strojového učenia. Dokáže prijať niekoľko vstupov, z ktorých každý má váhu označujúcu jeho dôležitosť, a vygenerovať výstup 0 alebo 1. Ak sa skombinuje s mnohými ďalšími perceptrónmi, vytvorí umelú neurónovú sieť (*Perceptrons and Multi-Layer Perceptrons*, n.d.; Vasilev et al., 2019). Perceptrón je matematicky definovaný nasledovne:

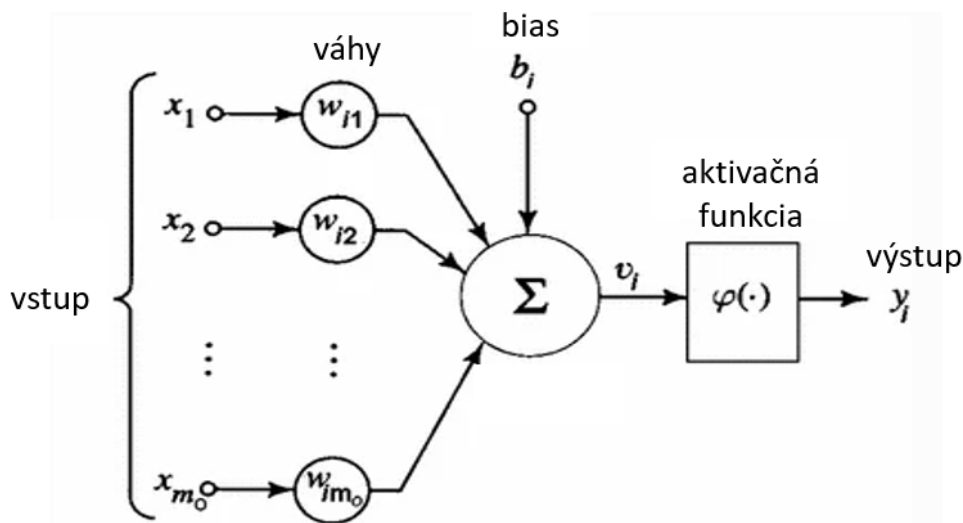
$$y = f\left(\sum_i x_i w_i + b\right) \quad (1.1)$$

Najprv sa vypočíta vážený súčet vstupov (x_i) a váh (w_i). K váženému súčtu sa pripočíta bias (b). Bias slúži ako ďalší parameter na zlepšenie výkonnosti modelu (Vasilev et al., 2019).

1.2 Aktivačné funkcie

Aktivačné funkcie sa používajú v hlbokých neurónových sieťach na transformáciu vstupného signálu na výstupný signál, ktorý prichádza ako vstup do ďalšej vrstvy. V hlbokej neurónovej sieti vypočítame súčet súčinov vstupov a ich zodpovedajúcich váh a nakoniec naň aplikujeme aktivačnú funkciu, aby sme získali výstup danej vrstvy a dodali ho ako vstup ďalšej vrstve. Aplikovanie aktivačnej funkcie je zobrazené na Obrázku 1–2.

Ak by sa v neurónovej sieti nepoužívala aktivačná funkcia, potom jej výstupom



Obrázok 1 – 2 Aplikovanie aktivačnej funkcie (Kuri-Morales, 2017) - upravené

by bol vážený súčet vstupov $\sum_i x_i w_i$, čo je lineárna funkcia. Neurónová sieť bez aktivačných funkcií funguje ako model jednoduchšej lineárnej regresie.

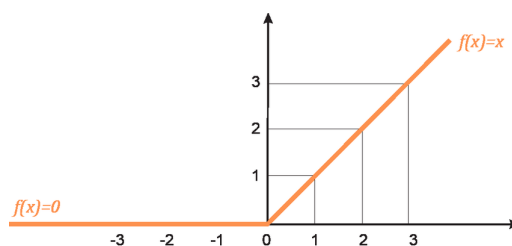
V praxi sa však ukázalo, že je potrebné, aby sa neurónová sieť nielen učila a počítala lineárnu funkciu, ale vykonávala aj zložitejšie úlohy. To je dôvod, prečo používame nelineárne aktivačné funkcie a hlboké neurónové siete, čím sa dosahuje kapacita pre vytváranie modelov schopných klasifikovať vysokorozmerné a nelineárne súbory údajov. V tom prípade má model viacero skrytých vrstiev a tiež komplexnú architektúru na extrahovanie znalostí (Sharma et al., 2020; Vasilev et al., 2019).

V tejto práci v skrytých vrstvách neurónovej siete používame aktivačnú funkciu ReLU a vo výstupnej vrstve aktivačnú funkciu Softmax.

1.2.1 ReLU

ReLU (angl. *Rectifier Linear Unit*) je často používanou aktivačnou funkciou v modeloch hlbokého učenia. Funkcia vráti 0, ak dostane akýkoľvek záporný vstup, ale pre akúkoľvek kladnú hodnotu x vráti túto hodnotu (pozri obr. 1 – 3). Možno ju teda zapísať ako (Brownlee, 2018):

$$f(x) = \max(0, x) \quad (1.2)$$



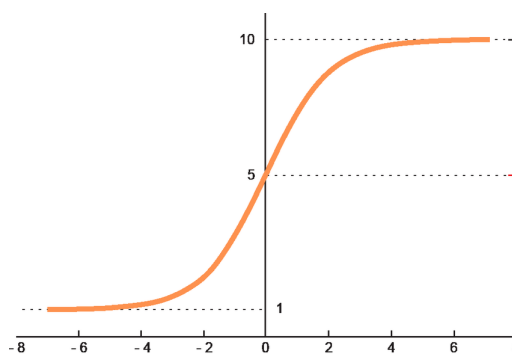
Obrázok 1 – 3 Grafické zobrazenie aktivačnej funkcie ReLU

1.2.2 Softmax

Softmax je matematická funkcia, ktorá prevádza vektor čísel na vektor pravdepodobností, kde pravdepodobnosti každej hodnoty sú úmerné relatívnej škále každej hodnoty vo vektore.

Vo všeobecnosti sa Softmax používa iba pre výstupnú vrstvu neurónovej siete. Hlavnou výhodou je schopnosť kategorizovať do viacerých tried. Normalizuje výstupy medzi 0 a 1 pre každú triedu a delí ich súčtom, čím dáva pravdepodobnosť, že vstupná hodnota je v danej triede (Brownlee, 2018).

$$f_i(\vec{x}) = \frac{e^{x_i}}{\sum_{j=1}^J e^{x_j}} \quad (1.3)$$



Obrázok 1 – 4 Grafické zobrazenie aktivačnej funkcie Softmax

1.3 Chybové funkcie

Cieľová funkcia optimalizácie môže byť minimalizačná alebo maximalizačná. Pri neurónových sieťach sa zvyčajne snažíme chybu minimalizovať. Na výpočet chyby

tak slúži funkcia ktorá minimalizuje chybu a nazýva sa chybová.

Táto funkcia charakterizuje úspešnosť algoritmu pri učení z daných údajov. Ak sa predikcia príliš líši od skutočného výsledku, chybová funkcia môže dosiahnuť vysokú hodnotu. Zníženiu chybovosti napomáha aplikovanie procesu optimalizácie počas učenia.

V tejto časti si predstavíme niekoľko chybových funkcií a ich aplikáciu pri hlbokom učení. Pre algoritmy hlbokého učenia nie je k dispozícii žiadna univerzálna chybová funkcia. Pri výbere chybovej funkcie pre konkrétny problém je potrebné zvážiť faktory, napr. zložitosť algoritmu.

V závislosti od typu úlohy možno chybové funkcie rozdeliť do dvoch kategórií, a to na (1) regresné chybové funkcie pre predikciu spojitých hodnôt a (2) klasifikačné chybové funkcie pre výstup z rôznych kategorických hodnôt (Goodfellow et al., 2016).

V práci riešime klasifikačnú úlohu, preto sme použili jednu z klasifikačných chybových funkcií, konkrétne krížovú entropiu.

1.3.1 Krížová entropia

Každá predikovaná hodnota sa porovnáva so skutočnou výstupnou hodnotou danej triedy (0 alebo 1) a vypočíta sa skóre, ktoré penalizuje pravdepodobnosť na základe vzdialenosti od očakávanej hodnoty. Penalizácia je logaritmická a poskytuje malé skóre za malé rozdiely (0,1 alebo 0,2) a veľké skóre za veľký rozdiel (0,9 alebo 1,0).

Za dokonalý model sa považuje model s krížovou entropiou 0,0. Krížovú entropiu môžeme vypočítať nasledovne:

$$CR = - \sum_i^M y_i \log(\hat{y}_i), \quad (1.4)$$

kde M predstavuje počet tried, y_i je skutočná výstupná hodnota a \hat{y}_i je predikovaná hodnota (Goodfellow et al., 2016).

1.4 Algoritmus spätného šírenia chýb

Algoritmus spätného šírenia chýb je bežne používaný pri tréňovaní neurónových sietí. Je to metóda optimalizácie váh neurónovej siete na základe chybovosti získanej v predchádzajúcej epoche (iterácii). Správne vyladenie váh umožňuje znížiť chybovosť, tým sa model stane spoľahlivejším. Váhy sa aktualizujú spätne od výstupu smerom k vstupu. Optimalizáciou váh možno minimalizovať chyby medzi pozorovanými dátami a predikciou (Saitoh, 2021; Purnawansyah et al., 2021).

1.5 Optimalizácia

Výrazné množstvo algoritmov hlbokého učenia potrebuje určitú verziu optimalizovania. Úlohou optimalizácie je minimalizácia alebo maximalizácia funkcie $f(x)$. Maximalizáciu možno dosiahnuť minimalizáciou $-f(x)$ prostredníctvom minimalizačného algoritmu.

Optimalizačná funkcia je časťou neurónovej siete, ktorá indikuje rýchlosť tréningového procesu a presnosť siete pri predikciách. Optimalizátor reaguje na výstup chybovej funkcie. Jednou z často používaných optimalizačných funkcií je Adam (Goodfellow et al., 2016).

1.5.1 Adam

Algoritmus Adam je metóda stochastickej optimalizácie funkcie založená na použití gradientu prvého rádu. Je vhodná na priamu implementáciu pre akýkoľvek model z hľadiska veľkých súborov údajov a parametrov. Z hľadiska hardvérových zdrojov vyžaduje menej pamäte a je výpočtovo veľmi efektívna.

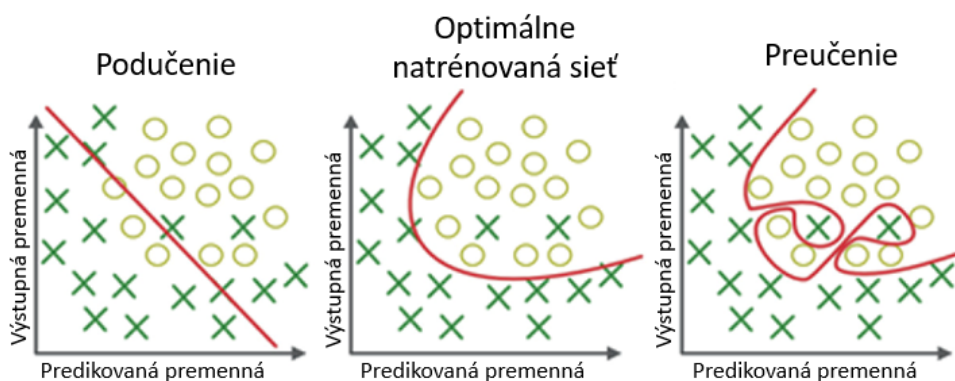
Adam je metóda, ktorá udržiava adaptívne miery učenia oddelené od každého parametra a udržiava jednu mieru učenia pre všetky aktualizácie váh, ktoré sa počas tréňovania nemenia (Kingma and Ba, 2014).

1.6 Regularizácia

Regularizácia je súbor stratégií, ktoré umožňujú učiacemu sa algoritmu lepšie generalizovať na nových dátach, často krátko na úkor zníženia výkonu na trénovacej množine. Generalizácia je schopnosť dobre klasifikovať údaje tej istej triedy ako učiace sa údaje, ktoré nikdy predtým nevidela (Goodfellow et al., 2016).

Pri trénovaní modelu neurónovej siete môže dôjsť k preučeniu (angl. *overfitting*) alebo podučeniu (angl. *underfitting*) modelu neurónovej siete (pozri obr. 1 – 5):

- Preučenie je stav, keď model produkuje dobré výsledky na trénovaných dátach, ale na testovacích dátach funguje zle. Stáva sa to preto, lebo model sa tak dobre prispôsobuje trénovacím dátam, že ponecháva veľmi malý alebo žiadny priestor na zovšeobecnenie nad novými dátami.
- Podučenie znamená, že model nedosahuje dobré výsledky z dôvodu príliš jednoduchého modelu alebo rôznorodosti datasetu. Takýto model nedokáže modelovať trénované údaje ani zovšeobecniť nové údaje (Stephen et al., 2014).

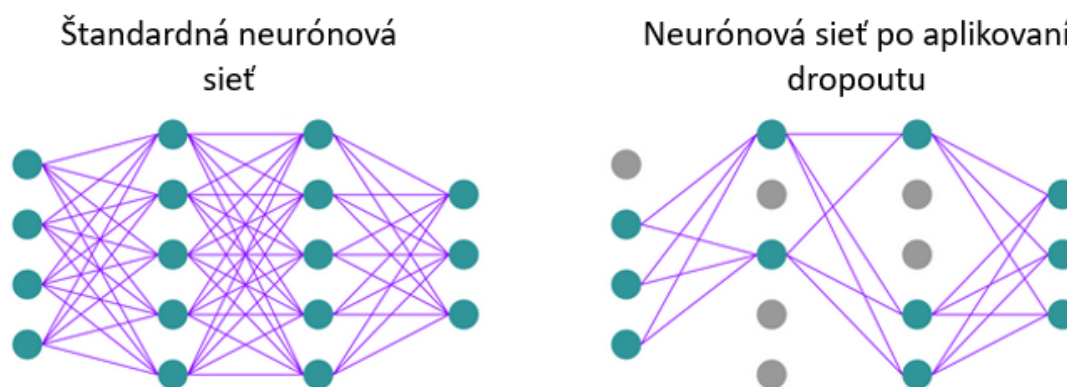


Obrázok 1 – 5 Zobrazenie podučenej, optimálne natrénovanej a preučenej neurónovej siete (Trivedi et al., 2021) - upravené

Aby sme zabránili preučeniu v našej práci, trénovanie bolo zastavené v prípade ak sa pri trénovaní modelu vo viacerých epochách po sebe nezlepšila `validation_accuracy`. Taktiež sme použili regularizačnú techniku dropout.

1.6.1 Dropout

Dropout je regularizačná technika, ktorá pomáha predchádzať preučeniu modelu neurónovej siete. Kľúčovou myšlienkou je náhodné vynechávanie neurónov a všetkých prisluchajúcich prepojení z neurónovej siete počas trénovania. Tým sa zabráni prílišnému prispôbovaniu jednotiek (Srivastava et al., 2014) (pozri obr. 1–6).



Obrázok 1–6 Princíp regularizácie učenia pomocou dropoutu

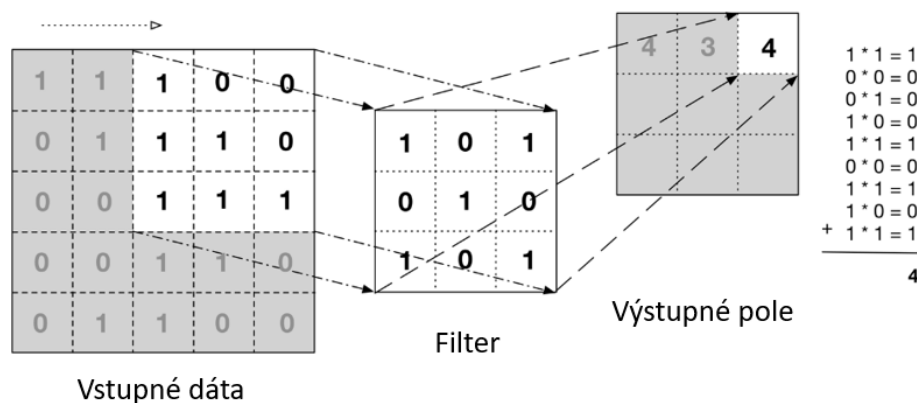
1.7 Konvolučné neurónové siete

Konvolučné neurónové siete sú veľmi podobné bežným dopredným neurónovým sieťam. Líšia sa v tom zmysle, že CNN explicitne predpokladajú, že vstupy sú obrázky, čo nám umožňuje zakódovať špecifické vlastnosti v architektúre, aby sme rozpoznali určité vzory v obrázkoch. CNN využívajú priestorový charakter údajov.

Na rozdiel od bežných neurónových sietí majú vrstvy CNN neuróny usporiadané v niekoľkých rozmeroch: kanály, šírka, výška a počet filtrov. Konvolučná neurónová sieť pozostáva z postupnosti vrstiev, kde každá vrstva transformuje aktivácie alebo výstupy predchádzajúcej vrstvy prostredníctvom inej diferencovateľnej funkcie. V CNN sa používa niekoľko takýchto vrstiev, ktoré budú bližšie popísané v nasledujúcich častiach. Najbežnejšie stavebné bloky, ktoré poznáme vo väčšine architektúr CNN sú vstupná vrstva, konvolučná vrstva, pooling vrstva a plne prepojené vrstvy (Sewak et al., 2018):

- **Vstupná vrstva** obsahuje dáta obrázku.
- **Konvolučná vrstva** používa konvolučné jadro ako filter pre vstup. Zvyčajne je takýchto filtrov veľa. Počas priameho prechodu sa filter posúva po vstupe a vypočíta aktivačnú mapu filtra v danom bode tak, že vypočíta bodový súčin jednotlivých hodnôt a ich sčítaním získa aktiváciu v danom bode. Príklad konvolúcie môžeme vidieť na obrázku 1 – 7.

Hlavným cieľom konvolúcie v súvislosti s CNN je extrahovať vlastnosti zo vstupného obrazu. Táto vrstva vykonáva väčšinu výpočtov v CNN.



Obrázok 1 – 7 Príklad konvolúcie

- **Pooling vrstva**, cieľom použitia pooling je vzorkovať vstupný obraz s cieľom znížiť výpočtové zaťaženie, spotrebu pamäte a počet parametrov. Pooling vrstva zvyčajne pracuje na každom vstupnom kanáli nezávisle, takže výstupná hĺbka je rovnaká ako vstupná hĺbka.

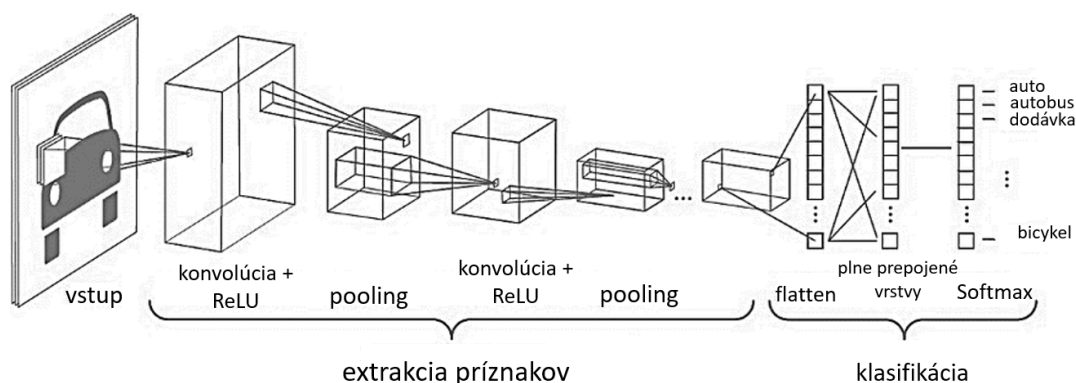
Rovnako ako konvolučné vrstvy, každý neurón je v pooling vrstve pripojený k výstupom obmedzeného počtu neurónov v predchádzajúcej vrstve, ktoré sa nachádzajú v malom obdĺžnikovom recepčnom poli.

Dve najbežnejšie pooling operácie sú *max-pooling* a *average-pooling*. Max-pooling vyberá maximum hodnôt v oblasti vstupnej mapy prvkov v každom kroku a average-pooling priemernú hodnotu hodnôt v oblasti. Výstupom v

každom kroku je jeden skalár, čo vedie k výraznému zmenšeniu veľkosti výstupu.

Ak filter zasahuje mimo obrazu, potom môžeme tieto neznáme hodnoty buď ignorovať, alebo ich nahradiť nulami. Tento jav je známy ako padding. Ak je prijateľné stratiť niekoľko hodnôt, môžeme nastaviť padding na *VALID*, ktorý sme využili aj v našej praktickej časti. V opačnom prípade nastavíme padding na *SAME*.

- **Plne prepojené vrstvy** spájajú každý neurón v jednej vrstve s každým neurónom v druhej vrstve. Ukážku CNN môžeme vidieť na obrázku 1 – 8.

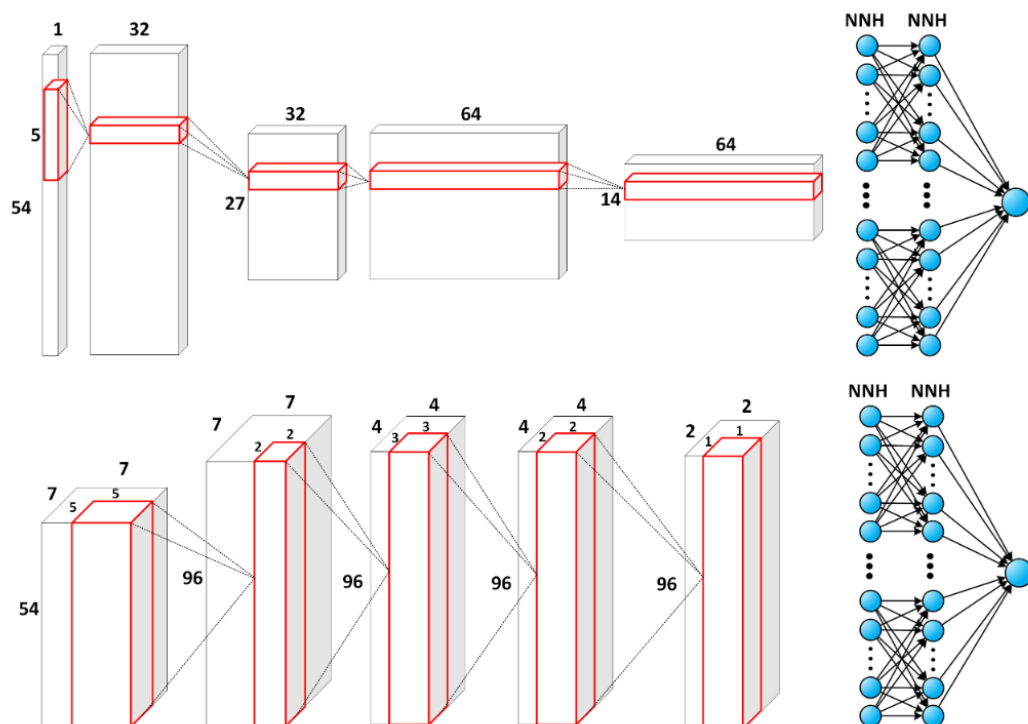


Obrázok 1 – 8 Príklad ukážky CNN (Alamsyah et al., 2019) - upravené

Konvolučná neurónová sieť, ktorú sme opísali, je príkladom dvojrozmernej CNN (2D CNN). V posledných rokoch sa však aplikácie CNN rozšírili aj do iných oblastí pomocou jednorozmerných údajov, napr. spektrálna analýza a analýza časových radov, spracovanie prirodzeného jazyka. Alternatívou k 2D CNN je jeho upravená verzia, ktorú používame aj v tejto práci, a to jednorozmerné CNN (1D CNN) (pozri obr. 1 – 9).

Hlavné výhody prístupu založeného na 1D CNN, umožňujúceho spracovávať časové rady údajov, sú (Kiranyaz et al., 2021):

- kompaktná konfigurácia architektúry, ktorá vykonáva len 1D konvolúcie, vďaka čomu je vhodná na detekciu a monitorovanie zmien v údajov v čase,



Obrázok 1 – 9 Hlboká CNN architektúra. (Hore) 1D CNN architektúra. (Dole) 2D CNN architektúra (Kussul et al., 2017)

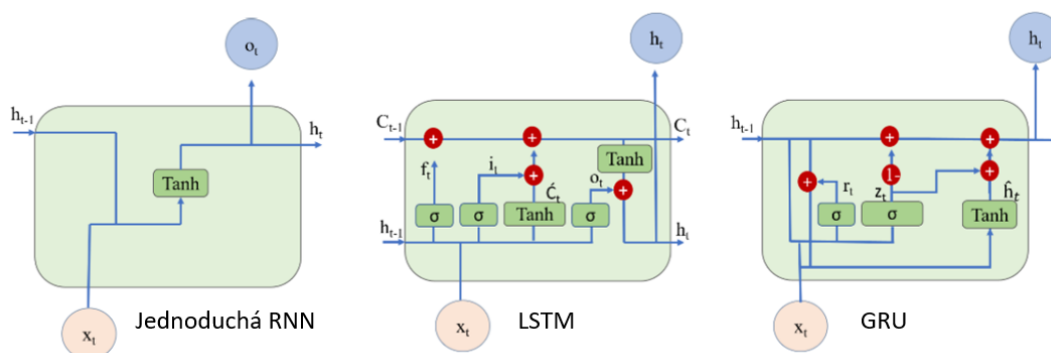
- nákladovo efektívna a praktická hardvérová implementácia v reálnom čase,
- schopnosť zabezpečiť efektívne tréňovanie klasifikátora s obmedzenou veľkosťou súboru tréňovacích dát.

1.8 Rekurentné neurónové siete

Rekurentné neurónové siete (skr. RNN, angl. *Recurrent Neural Network*) sú triedou neurónových sietí, ktoré sú prirodzene vhodné na spracovanie časových radov údajov a iných sekvenčných údajov. Prechod z dopredných neurónových sietí na rekurentné neurónové siete je koncepčne jednoduchý. Dopredné siete tradične mapujú vstupy pevnej veľkosti na výstupy pevnej veľkosti, napríklad na mapovanie z obrazu pevného priestorového rozsahu na jeho triedu alebo na segmentačnú mapu rovnakého priestorového rozsahu. Naopak rekurentné neurónové siete prirodzene pracujú na

vstupných sekvenciách s premenlivou dĺžkou a mapujú výstupné sekvencie s premenlivou dĺžkou, napríklad na mapovanie z obrazu na rôzne vety, ktoré tento obraz opisujú. Táto schopnosť sa dosahuje zdieľaním parametrov a transformácií v čase.

Špeciálny typ rekurentnej siete LSTM (*Long Short-Term Memory*) bol zavedený na zmiernenie problému miznúceho gradientu a stala sa jednou z najpopulárnejších architektúr RNN. Neskôr boli ako jednoduchšia alternatíva k LSTM zavedené rekurentné jednotky (skr. GRU), ktoré sa tiež stali pomerne populárnymi. LSTM a GRU zmierňujú tento problém veľmi podobným spôsobom. Obe RNN architektúry sme použili v praktickej časti tejto práce (Kostadinov, 2018; Zhou et al., 2020). Rozdiel medzi jednoduchou RNN, LSTM a GRU môžeme vidieť na obrázku 1 – 10.



Obrázok 1 – 10 Model architektúr jednoduchej RNN, LSTM a GRU (Sajjad et al., 2020) - upravené

1.8.1 Long short-term memory – LSTM

LSTM si dlhodobo pamätá vstupy pomocou skrytých jednotiek nazývaných pamäťové bunky. Tieto bunky fungujú ako únikové neuróny s bránou (angl. *gated leaky neurons*), ktoré sa v nasledujúcom kroku navzájom spájajú. V praxi to znamená, že k svojmu reálnemu stavu pridá vonkajší signál. Toto spojenie je viackrát zakódované iným neurónom, ktorý rozhodne, kedy obsah vymaže zo svojej pamäte.

Obojsmerná sieť LSTM (skr. BiLSTM, angl. Bidirectional Long Short-Term Memory) je špecifickým typom siete LSTM. BiLSTM pozostáva z dvoch samostatných

skrytých vrstiev. Prvá vrstva spracováva vstupnú sekvenciu v priamom smere, zatiaľ čo druhá skrytá vrstva spracováva sekvenciu v spätnom smere. Skryté vrstvy potom spájajú svoje výstupy vo výstupnej vrstve. V dôsledku toho má výstupná vrstva prístup k minulému aj budúcemu kontextu aktuálnej udalosti v sekvencii (Schuster and Paliwal, 1997; Smagulova and James, 2020). Dôležité časti LSTM sú:

- stav bunky (angl. *cell state*): dlhodobá pamäť,
- brána zabudnutia (angl. *forget gate*): brána, ktorá rozhoduje o tom, aká informácia sa má vynechať,
- vstupná brána (angl. *input gate*): brána, ktorá rozhoduje o tom, čo sa má postúpiť do ďalšej aktivácie.

1.8.2 Gated Recurrent Unit – GRU

GRU (angl. *Gated Recurrent Unit*) je vylepšená verzia štandardných rekurentných neurónových sietí, ktorej úlohou je znížiť počet parametrov, ktoré obsahuje blok LSTM. GRU sa skladá z dvoch brán: resetovacia brána a aktualizácia brána. Aktualizačná brána je zodpovedná za určenie množstva predchádzajúcich informácií, ktoré je potrebné odovzdať ďalej do ďalšieho stavu. Brána resetu rozhoduje o tom, či je predchádzajúci stav bunky dôležitý alebo nie (Saitoh, 2021).

2 Analýza súčasného stavu – klasifikácia svetelných kriviek premenných hviezd

V súčasnosti možno problém identifikácie hviezdnych parametrov v dvojhviezdnych systémoch označiť za problém veľkých dát. Nedávny vývoj pozorovacích techník viedol k získaniu obrovského množstva dát o dôležitých javoch vo vesmíre. Je dôležité spracovať dáta v reálnom čase. Ľudské zdroje to nedokážu, a preto hľadáme sofistikovanejšie riešenia. V súčasnosti výskumníci spracovávajú a analyzujú obrovské množstvo údajov ručne pomocou vyvinutých nástrojov, napr. Phoebe v1 (Prša and Zwitter, 2005), Phoebe v2 (Prša et al., 2016), ELC (Orosz and Hauschildt, 2000) atď.

Prvým krokom manuálneho prístupu je nájdenie morfolologickej klasifikácie. Následne sa identifikujú typy hviezd a všetky ďalšie súvisiace fyzikálne veličiny, ako je napr. gravitačné zrýchlenie a teplota. Zvyčajne sa vhodná implementácia vyvinutých nástrojov spolieha na zvolenú morfologickú klasifikáciu a vyžaduje ju na interné spracovanie údajov (Cokina et al., 2021). V práci sa zaoberáme vhodným riešením hlbokého učenia na určenie morfolologickej klasifikácie a klasifikácie na dvojhviezdy alebo škvrnité hviezdy. Klasifikáciu považujeme za nevyhnutný krok pri automatizácii analýzy svetelných kriviek.

2.1 Svetelné krivky

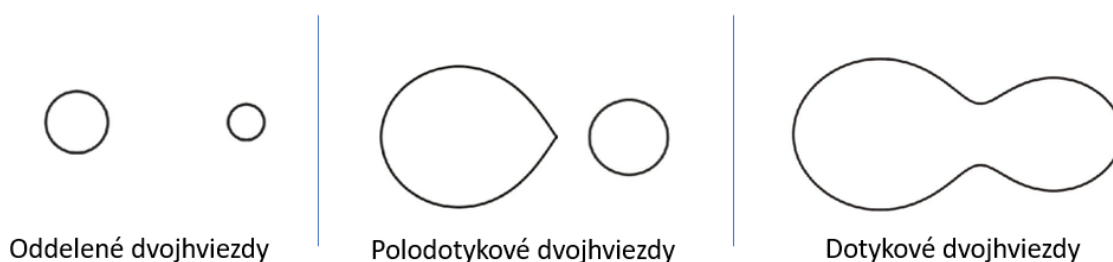
Vďaka fotometrickým prieskumom sa získalo veľké množstvo svetelných kriviek hviezd, ktoré obsahujú veľký počet zákrytových dvojhviezd. Zákrytové dvojhviezdy sú súčasťou širokej skupiny premenných hviezd. Premennosť je spôsobená vzájomným zatmením dvoch zložiek počas ich obežnej dráhy okolo spoločného ťažiska. K zákrytu dochádza vtedy, keď je jedna zložka prekrytá druhou. Tento jav spôsobuje pokles intenzity svetla vyžarovaného dvojhviezdny systémom. Výsledná svetelná krivka odráža procesy v systéme a ich vývoj v čase (Prsa, 2018).

V našej práci sa zaoberáme svetelnými krivkami, ktoré predstavujú základné vstupné dáta. Svetelné krivky sa zobrazujú ako graf, ktorý predstavuje jasnosť objektu za určité časové obdobie. Jasnosť môže byť zobrazená vo fyzikálnych jednotkách (napr. magnitúda) alebo normalizovaná na určitú hodnotu, teda bez jednotiek. Os x , ktorá predstavuje časový priebeh, sa zvyčajne vyjadruje v časových jednotkách ako juliánske dátumy alebo častejšie sa udáva v bezrozmernej veličine nazývanej fotometrická fáza. Pri štúdiu objektov, ktoré majú premenlivú intenzitu svetla v čase, ako sú napríklad zatmenia dvojhviezd alebo supernovy alebo novy poskytujú svetelné krivky cenný zdroj informácií a uľahčujú nám študovať takéto objekty (Hurley et al., 2002).

Dvojhviezdne systémy sú kľúčovými objektami v astrofyzike a predstavujú viac ako polovicu všetkých známych hviezd vo vesmíre (Abt, 1983). Štúdium dvojhviezd umožňuje lepšie pochopiť vývoj jednotlivých hviezd a vysvetliť mnohé pozorované javy vo vesmíre. Najjednoduchším zdrojom informácií o dvojhviezdnom systéme je jeho fotometrická svetelná krivka, pričom prvou informáciou, ktorú možno z takýchto údajov získať, je klasifikácia daného zdroja.

Zákrytové dvojhviezdy sú typom premenných hviezd, ktorých zmeny sú spôsobené vonkajšími faktormi. Môžu sa javiť ako jeden svetelný bod, ale pochádza z dvoch hviezd, ktoré okolo seba obiehajú. Zmeny intenzity svetla zatmenia dvojhviezd sú spôsobené tým, že jedna hviezda prechádza pred druhou (Percy, 2011). Ak predpokladáme, že hviezdy sú guľové a majú kruhové dráhy, potom môžeme ľahko odhadnúť, ako sa mení svetlo s časom a zatmením dvojhviezd (Wilson and Devinney, 1971; Wilson, 1979; Prša et al., 2016). V súčasnosti sa najčastejšie na spôsob klasifikácie používa všeobecný Rocheho model, ktorý nám poskytuje štyri možné konfigurácie: oddelené dvojhviezdy, polooddelené dvojhviezdy, dvojhviezdy s nadmerným kontaktom, dvojhviezdy s dvojitým kontaktom. Táto klasifikačná schéma popisuje množstvo vyplnenia Rocheových lalokov (Kallrath and Milone, 2009). Podrobnejšie popíšeme dva z týchto prípadov, ktoré predstavujú najčastejšie sa vyskytujúce systémy (Cokina et al., 2021):

- **Oddelené dvojhviezdy** predstavujú dvojicu hviezd, v ktorej sa každá hviezda nachádza vo vlastnom Rocheovom laloku (pozri obr. 2–1). Rocheov lalok je oblasť okolo hviezdy v dvojhviezdnom systéme, v ktorej je obiehajúci materiál gravitačne viazaný na túto hviezdu. Táto oblasť má tvar slzy, ktorej vrchol smeruje k druhej hviezde. V Rocheovej oblasti je gravitačná sila hviezdy väčšia ako gravitačná sila druhej hviezdy. Hviezdy sa navzájom výrazne neovplyvňujú a vyvíjajú sa nezávisle (Terrell, 2001; Malkov et al., 2007).
- **Dotykové dvojhviezdy** Dotykové dvojhviezdy sú také, pri ktorých obe zložky tvoriace dvojhviezdny systém presahujú svoje Rocheove laloky (pozri obr. 2–1). Materiál obiehajúci v Rocheových lalokoch oboch hviezd sa spojí a vytvorí obal nad celým dvojhviezdnym systémom. Trenie obiehajúceho materiálu v obale brzdí orbitálny pohyb, čo môže nakoniec viesť k splynutiu hviezd. Špecifickým podtypom dotykových systémom sú tzv. polodotykové dvojhviezdy, ktorý predstavuje systém na hrane dotykového systému s vyplneným Rocheovým lalokom minimálne jednej z hviezd.



Obrázok 2–1 Ukážka oddeleného binárneho systému, polodotykového binárneho systému a dotykového binárneho systému

Ďalším aspektom, ktorý vplýva na premenlivosť hviezd, je výskyt škvŕn na ich povrchu. Na vytvorenie škvŕny stačí, aby hviezda rotovala dostatočne rýchlo na to, aby sa sploštila. Fotometrické škvŕny sú oblasti na hviezde, ktoré svietia inak ako ostatné časti povrchu hviezdy. Môžu mať rôznu veľkosť a vyskytovať sa kdekoľvek na hviezde. U hviezd slnečného typu a chladnejších hviezd sa stretávame s fotomet-

rickými škvrnami s rôznymi efektívnymi teplotami, ktoré spôsobujú kontrast škvrny s jej okolím (Mikulášek and Zejda, 2013).

2.2 Analýza súčasného stavu

Všeobecná klasifikácia svetelných kriviek je dôležitou úlohou pri štúdiu rôznych astronomických objektov. Väčšinou súvisí s analýzou premenných hviezd, ale aj s analýzou objektov v slnečnej sústave alebo iných planetárnych systémoch.

Práca Carrasco-Davis et al. (2019), navrhuje nový model sekvenčnej klasifikácie astronomických objektov založený na rekurentnej konvolučnej neurónovej sieti (RCNN), ktorá ako vstupy používa sekvencie obrázkov. Tento prístup sa vyhýba výpočtu svetelných kriviek alebo rozdielových obrazov. Je to prvýkrát, čo sa sekvencie obrazov používajú priamo na klasifikáciu premenných objektov v astronómii. Druhým prínosom tejto práce je proces simulácie obrazu. Generuje syntetické sekvencie obrazov, ktoré zohľadňujú prístrojové a pozorovacie podmienky. Simulovaný súbor údajov sa používa na tréning klasifikátora RCNN. Tento prístup umožňuje vytvárať súbory údajov na tréning a testovanie modelu RCNN pre rôzne astronomické pozorovania. Okrem toho je použitie simulovaného súboru údajov rýchlejšie a lepšie prispôsobiteľné rôznym prieskumom a klasifikačným úlohám. Na otestovanie klasifikátora RCNN natrénovaného so syntetickým súborom údajov použili reálne údaje z prieskumu High cadence Transient Survey (HiTS), pričom dosiahli priemernú úspešnosť klasifikácie 85%, ktorá sa po vykonaní jemného ladenia s 10 reálnymi vzorkami na triedu zlepšila na 94%.

V práci Sarro et al. (2006) autori predstavili systém automatickej klasifikácie svetelných kriviek zákrytových dvojhviezd. Cieľom klasifikácie je klasifikovať zatmenia dvojhviezd podľa ich geometrickej konfigurácie v modifikovanej verzii použitej klasifikačnej schémy. Na klasifikáciu použili bayesovský súbor neurónových sietí natrénovaných na údajoch zozbieraných družicou Hipparcos. Súbor údajov pozostával zo siedmich rôznych kategórií, zahŕňal aj rozlíšenie excentrických dvojhviezdnych

systémov a dvoch typov morfológií pulzujúcich svetelných kriviek. Navrhovaný model sa pokúša riešiť problémy heterogenity tried a subjektivity tradičnej klasifikácie svetelných kriviek, ktorá do tej istej skupiny zahrňa systémy s rôznymi fyzikálnymi vlastnosťami. Percento straty pre ich najlepšiu architektúru bolo $6,9 \pm 1,3$ %.

Práca Cokina et al. (2021) je zameraná na automatickú klasifikáciu zákrytových dvojhviezd pomocou metód hlbokého učenia. Ich klasifikátor poskytuje nástroj na kategorizáciu svetelných kriviek dvojhviezd do dvoch tried: oddelené a dotykové. Práca sa zaoberala iba krivkami jedného filtra. Pomocou softvéru ELISa boli vytvorené syntetické dáta, ktoré boli použité na tréning klasifikátora. Na účely vyhodnotenia zhromaždili 100 svetelných kriviek pozorovaných dvojhviezd, aby mohli vyhodnotiť niekoľko klasifikátorov. Polodotykové zákrytové binárne hviezdy vyhodnotili ako oddelené dvojhviezdy. Najlepšie fungujúci klasifikátor kombinuje LSTM rekurentnú a jednorozmernú konvolučnú neurónovú sieť, ktorá na hodnotenom súbore dosiahla 98% úspešnosť klasifikácie (angl. accuracy).

Naša práca nadväzuje na prácu Cokina et al. (2021). Zaoberá sa klasifikáciou dvojhviezd na oddelené a dotykové systémy, avšak s krivkami 5 filtrov. Je rozšírená o klasifikáciu na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy.

3 Klasifikácia zákrytových premenných hviezd pomocou metód hlbokého učenia

V tejto kapitole sa venujeme návrhu vhodných klasifikačných modelov na riešenie automatickej klasifikácie zákrytových premenných hviezd. Pri riešení zvolených klasifikačných úloh v práci sme postupovali podľa metodológie CRISP-DM.

3.1 Vybrané technológie

V procese výberu technológií, sme si vybrali všestranný programovací jazyk Python verzie 3. Tento jazyk je medzi dátovými analytikmi veľmi rozšírený, jednoduchý a má ľahko použiteľnú syntax. Umožní rýchlejšie a efektívnejšie pracovať s dátami a ich úpravou. Python má širokú škálu aplikovateľnosti s nespočtým množstvom knižníc, ktoré rozširujú jeho funkcionality. V praktickej časti sme si pre analýzu vstupných dát, implementáciu metód hlbokého učenia a realizáciu experimentov zvolili:

- **Keras** je rámec (angl. *framework*) na riešenie problémov strojového učenia. Umožňuje rýchle experimentovanie a implementácie neurónovej siete. Disponuje viacerými modulmi vrátane vrstiev neurónovej siete, optimalizačných, aktivačných, chybových a regularizačných funkcií. Keras je API neurónovej siete, ktoré okrem iného beží nad TensorFlow. Ponúka oveľa jednoduchší prístup k vytváraniu modelov ako čistý TensorFlow.
- **TensorFlow** je komplexná open source platforma pre strojové učenie. Má komplexný, flexibilný systém nástrojov a knižníc. Dá sa použiť naprieč celým radom úloh, zameriava sa najmä na tréning hlbokých neurónových sietí.
- **Numpy** je softvérová knižnica, ktorá umožňuje vedecké výpočty v Pythone. Jeho najpoužívanejšia funkcia, v kontexte tejto práce je tvorba, úprava a výpočet N-rozmerných polí.

- **Pandas** je softvérová knižnica na štruktúrovanie údajov a manipuláciu s veľkými numerickými tabuľkami. Má vysoký výkon. Používa sa na predspracovanie vstupných údajov, teda ich štruktúrovanie a manipuláciu s tabuľkami.
- **Scikit-learn** je softvérová knižnica pre strojové učenie. Ponúka mnoho užitočných metód predspracovania údajov a vyhodnotenia výsledkov predikcie.

3.2 Hardvér

Trénovanie neurónových sietí je hardvérovo náročná operácia. Pri výskume sme využili zdieľaný server Technickej univerzity v Košiciach (TUKE) s webovým používateľským rozhraním JupyterLab. Čas tréovania je najviac ovplyvnený výkonom grafickej karty. Všetky tréovania neurónových sietí boli vykonané na serveri so špecifikáciami zobrazenými v tabuľke 3–1:

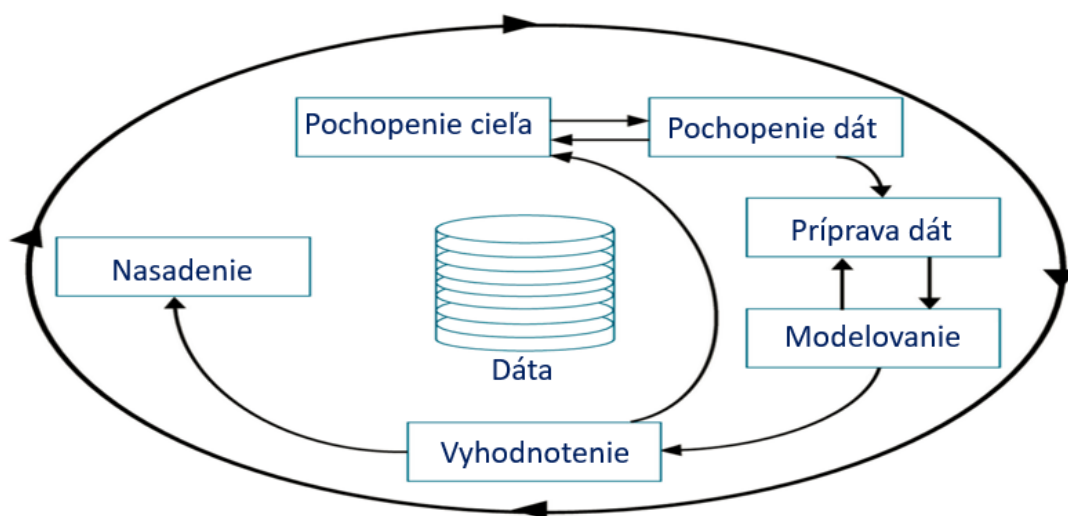
CPU	Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz
RAM	32GB
GPU	NVIDIA Tesla K40C 12G
OS	Ubuntu 18.04.5 LTS

Tabuľka 3–1 Hardwarové špecifikácie servera

3.3 Metodika práce

Pri riešení zadaných klasifikačných úloh v tejto práci sme postupovali podľa metodiky CRISP-DM (angl. *Cross Industry Standard Process for Data Mining*, (Paralič, 2003)). CRISP-DM je štandardný model procesu analýzy údajov. Cieľom tohto procesného modelu je konsolidovať prístupy a myšlienky súvisiace s údajmi prostredníctvom štandardného, organizovaného a štruktúrovaného prístupu. Tento model navrhuje vopred definovanú postupnosť fáz, ale môže byť potrebné opakovať niektoré z fáz v špecifických situáciách. Prehľad životného cyklu má pomôcť v dolovaní údajov

pochopiť celé údaje, proces analýzy a čo robiť v každej fáze tohto procesu. Ako je znázornené na obrázku 3 – 1, model procesu analýzy údajov navrhnutý v CRISP-DM je založený na cyklickom prístupe, ktorý zahŕňa šesť fáz. Šípky označujú najčastejšie sekvencie fáz, ktoré sa zvyčajne používajú v procese analýzy údajov, v niektorých prípadoch sa však očakáva pohyb tam a späť medzi fázami.



Obrázok 3 – 1 CRISP DM

- **Pochopenie cieľa** sa zameriava na pochopenie cieľov a požiadaviek projektu, ktoré sa snaží pretransformovať na konkrétnu definíciu úlohy.
- **Pochopenie dát** zahŕňa zber potrebných údajov a ich počiatočnú vizualizáciu/sumarizáciu s cieľom získať o nich prvé poznatky.
- **Príprava dát** zahŕňa všetky činnosti na vytvorenie konečného súboru dát z počiatočných nespracovaných údajov. Výstup fázy je súbor dát, ktoré sa použijú vo fáze modelovania.
- **Modelovanie**. V tejto fáze sa vyberie technika modelovania, návrh testovania, zostavenie modelu a posúdenie modelu.
- **Vyhodnotenie** zhromažďuje výsledky z modelovania a ich vyhodnotenie s cieľom overiť či spĺňajú požiadavky práce.

- **Nasadenie.** Hlavným účelom tejto záverečnej fázy je integrácia zvoleného riešenia do praxe.

3.4 Pochopenie cieľa

V nasledujúcich častiach bakalárskej práce sa zaoberáme klasifikačnými úlohami v astrofyzike, konkrétne na zákrytových premenných hviezdach. Zákrytové premenné hviezdy sú popísané v kapitole 2.

Naša práca nadväzuje na prácu Cokina et al. (2021), ktorá sa zaoberala binárnou klasifikáciou dvojhviezd na oddelené a dotykové krivky jedného filtra. Naším cieľom je vytvoriť všeobecnejší model, ktorý by dokázal klasifikovať dvojhviezdy na oddelené a dotykové krivky z ľubovoľných 5 základných filtrov.

Ďalším cieľom tejto práce je binárna klasifikácia na dvojhviezdy s kritickým sklonom dráhy a samostatné hviezdy so škvrnou, ktoré sa z pohľadu svetelných kriviek do určitej miery podobajú.

Doménoví experti požadovali, aby tieto klasifikácie boli čo najviac úspešné. Preto chceme, aby naše priemerné metriky (presnosť, návratnosť a F1 skóre) dosahovali pri klasifikácii dvojhviezd na oddelené a dotykové hodnoty nad 90%. Pri klasifikácii na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy, aby dosiahli hodnoty aspoň 80%.

3.5 Pochopenie dát

Jedným z problémov v oblasti neurónových sietí je potreba veľkého množstva vstupných dát. Získať dostatočný počet dát anotáciou pozorovacích údajov zákrytových premenných dvojhviezd je obtiažne. Tento problém nám pomohol vyriešiť tím astrofyzikov z Univerzity Pavla Jozefa Šafárika vygenerovaním syntetických dát pre svetelné krivky zákrytových premenných hviezd. Pre vygenerovanie bol použitý softvér ELISa (Fedurco et al., 2020; Cokina et al., 2021). Okrem syntetických dát nám dodali aj observačné dáta, ktoré boli použité na testovanie modelov. Softvér využíva

známe astrofyzikálne poznatky o zákrytových dvojhviezdach. Vo všeobecnosti rozvíja teóriu uvedenú Wilson (1979) a vylepšuje ju použitím geometrického prístupu a diskretizácie.

Poskytnuté datasety syntetických svetelných kriviek zákrytových premenných hviezd, sú zobrazené v tabuľke 3–2, kde `oddelené_krit` reprezentuje oddelené krivky s kritickým sklonom dráhy a `dotykové_krit` reprezentuje dotykové krivky s kritickým sklonom dráhy.

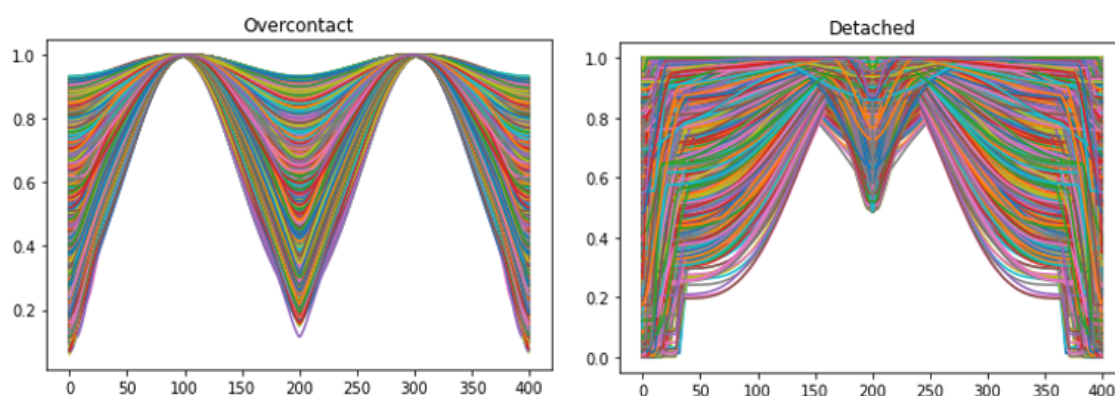
dataset	typ kriviek	veľkosť	počet kriviek
<code>detached_bellow_i_crit.db</code>	<code>oddelené_krit</code>	2.8 GB	66 648
<code>detached.db</code>	<code>oddelené</code>	94.9 GB	2 255 946
<code>overcontact_bellow_i_crit.db</code>	<code>dotykové_krit</code>	2.4 GB	55 914
<code>overcontact.db</code>	<code>dotykové</code>	3.9 GB	93 292
<code>single_spotty.db</code>	<code>škvrité</code>	295.3 MB	6 853

Tabuľka 3–2 Datasety syntetických dát

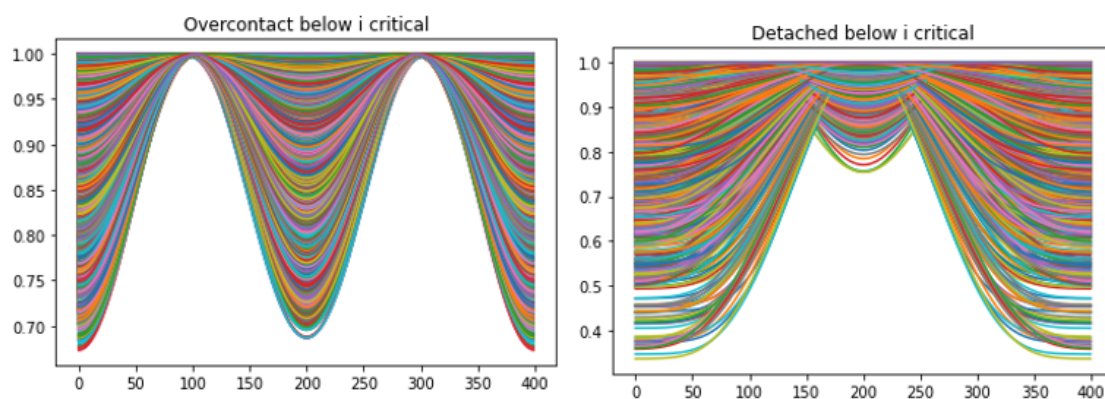
Na obrázku 3–2 je znázornená ukážka dotykových a oddelených zákrytových premenných hviezd. Tieto dáta sme použili pri klasifikácii dvojhviezd na dotykové a oddelené. Časť dotykových a oddelených kriviek s kritickým sklonom dráhy môžeme vidieť na obrázku 3–3. Vzorka škvritých hviezd je znázornená na obrázku 3–4. Dáta kriviek s kritickým sklonom dráhy a dáta kriviek škvritých hviezd boli použité pri klasifikácii na dvojhviezdy s kritickým sklonom dráhy a škvrité hviezdy.

Observačné dáta pre klasifikáciu dvojhviezd na oddelené a dotykové boli poskytnuté vo forme JSON súborov. Tvorilo ich 46 kriviek dotykových a 40 kriviek oddelených svetelných dvojhviezd. V rámci týchto testovacích dát sme pracovali s atribútom **Type** (oddelený alebo dotykový systém) a údajmi o svetelnej krivke.

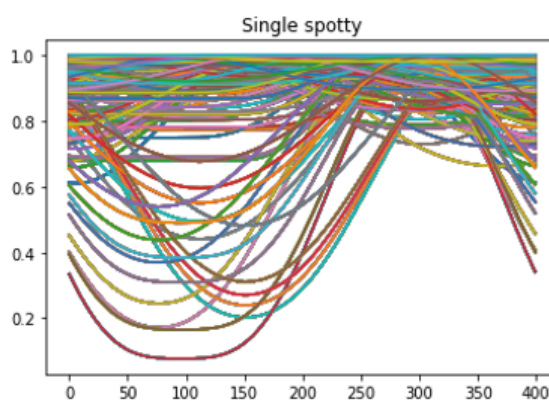
Observačné dáta pre klasifikáciu na dvojhviezdy s kritickým sklonom dráhy a škvrité hviezdy sme dostali v textových súboroch. Tvorilo ich 60 kriviek dvojhviezd s kritickým sklonom dráhy a 60 kriviek škvritých hviezd, ktoré obsahovali atribúty čas pozorovania, jasnosť a chybu. My sme pracovali s atribútom jasnosť.



Obrázok 3–2 Vizualizácia dotkových (vľavo) a oddelených (vpravo) kriviek



Obrázok 3–3 Vizualizácia dotkových (vľavo) a oddelených (vpravo) kriviek s kritickým sklonom dráhy



Obrázok 3–4 Vizualizácia škvrnitých hviezd

3.6 Príprava dát

Príprava dát je dôležitým krokom pri modelovaní neurónovej siete. Postup prípravy dát ovplyvňuje mnoho dôležitých parametrov. Znižuje chyby modelovania, urýchľuje proces trénovania neurónovej siete a vedie k zjednodušeniu systému ako celku. Výstupom tejto fázy je vytvorenie dátovej množiny, ktorá vstupuje do fázy modelovania.

3.6.1 Syntetické dáta

Keďže k dispozícii boli nerovnomerné datasety, tak ich bolo nutné vyvážiť. Tiež sme boli obmedzení aj HW a bolo potrebné, aby predspracovanie aj tréning boli ukončené v primeranom čase. Preto sme sa rozhodli pracovať s nasledovným počtom kriviek v jednotlivých klasifikačných úlohách (pozri tab. 3–3 a tab. 3–4).

	dataset	počet použitých kriviek	počet všetkých kriviek v datasete
Experiment 1	detached_random	50 000	2 255 946
	overcontact_random	50 000	93 292
Experiment 2	detached_random	50 000	2 255 946
	overcontact_random	50 000	93 292
Experiment 3	detached_random	50 000	2 255 946
	overcontact_random	50 000	93 292

Tabuľka 3–3 Datasety použité pri klasifikácii dvojhviezd na oddelené a dotykové

	dataset	počet použitých kriviek	počet všetkých kriviek v datasete
Experiment 4	detached_bellow_i_critical	10 000	66 648
	overcontact_bellow_i_critical	10 000	55 914
	single_spotty	6 853	93 292

Tabuľka 3–4 Datasety použité pri klasifikácii na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy

Predspracovanie prebiehalo nasledovne:

Všetky uvedené datasety boli dodané v .db formáte, pričom každý dataset sa skladal

z 3 tabuliek: *Parameters*, *Curves*, *Auxiliary*. Pre našu prácu sú potrebné tabuľky *Parameters* a *Curves*.

- Tabuľka *Curves* obsahuje nasledujúce atribúty: *id*, *Bessell_U*, *Bessell_B*, *Bessell_V*, *Bessell_R*, *Bessell_I*, *SLOAN_u*, *SLOAN_g*, *SLOAN_r*, *SLOAN_i*, *SLOAN_z*, *Kepler*, *GaiaDR2*, *TESS* (pozri tab. 3–5).

Z tabuľky *Curves* sme pracovali so základnými filtrami danej krivky: *Bessell_U*, *Bessell_B*, *Bessell_V*, *Bessell_R*, *Bessell_I*.

<i>id</i>	<i>Bessell_U</i>	<i>Bessell_B</i>	<i>Bessell_V</i>	<i>Bessell_R</i>	<i>Bessell_I</i>	<i>SLOAN_u</i>
5525038	b'\x93NUMPY\	b'\x93NUMPY\	b'\x93NUMPY\	b'\x93NUMPY\	b'\x93NUMPY\	b'\x93NUMPY\
<i>SLOAN_g</i>	<i>SLOAN_r</i>	<i>SLOAN_i</i>	<i>SLOAN_z</i>	<i>Kepler</i>	<i>GaiaDR2</i>	<i>TESS</i>
b'\x93NUMPY\	b'\x93NUMPY\	b'\x93NUMPY\	b'\x93NUMPY\	b'\x93NUMPY\	b'\x93NUMPY\	b'\x93NUMPY\

Tabuľka 3–5 Ukážka jedného riadku z tabuľky *Curves*

- Tabuľka *Parameters* obsahuje nasledujúce atribúty: *id*, *mass_ratio*, *primary_surface_potential*, *secondary_surface_potential*, *primary_t_eff*, *secondary_t_eff*, *inclination*, *critical_surface_potential*, *overcontact*, *primary_equivalent_radius*, *secondary_equivalent_radius*, *primary_filling_factor*, *secondary_filling_factor* (pozri tab. 3–6).

Z tabuľky *Parameters* sme pracovali iba s atribútom *overcontact*.

K jednotlivým krivkám z tabuľky *Curves* je priradený parameter *overcontact* tak, aby vznikla iba jedna tabuľka. Tabuľka je následne uložená vo formáte *.pk1*. Bolo to potrebné pre výrazné zrýchlenie načítania tabuliek, s ktorými sa pracuje v danej klasifikačnej úlohe.

Po načítaní tabuliek (v závislosti od klasifikačnej úlohy), sme dáta rozdelili na tréningovú a testovaciu množinu v pomere 80:20. Každá svetelná krivka je sa skladá

id	mass_ ratio	primary_ surface_potential	secondary_ surface_potential	primary_ t_eff	secondary_ t_eff	inclination
5525038	0.1	1.948052	1.948052	5500	5250	0.766994

critical_surface_ potential	overcontact	primary_ equivalent_radius	secondary_ equivalent_radius	primary_filling_ factor	secondary_ filling_factor
1.959104	1	0.585781	0.21126	0.169244	0.169244

Tabuľka 3 – 6 Ukážka jedného riadku z tabuľky Parameters

zo 400 hodnôt. Na krivky sme aplikovali funkciu šumu, ktorá bola dodaná výskumným tímom UPJŠ. Šum zabezpečí, aby sa syntetické dáta podobali viac na reálne (observačné) dáta. Ďalej boli vykonané menšie zmeny, aby mohol model pracovať s dátami.

Klasifikácia dvojhviezd na dotykové a oddelené sa líši od klasifikácie na dvojhviezdy s kritickým sklonom dráhy a škrvnité hviezdy v tom, že pri klasifikácii na dvojhviezdy s kritickým sklonom dráhy a škrvnité hviezdy sme po načítaní vstupných datasetov aplikovali funkciu náhodného zarovnania krivky, ktorá randomizuje zarovnanie svetelnej krivky z vopred určeného fázového intervalu na randomizovaný interval.

3.6.2 Observačné dáta

Observačné dáta boli namerané z reálnych pozorovaní a boli dodané vo forme JSON a textových súborov. Tieto dáta sme spracovali do formy .csv súboru. Podobne ako pri syntetických dátach, prebehla ich úprava do vhodnej formy pre testovanie. Dataset observačných kriviek sme použili na testovanie vytvorených modelov.

3.7 Modelovanie

V tejto podkapitole sa zameriame na vytvorenie modelov neurónovej siete, ktoré by mali dosiahnuť pri klasifikácii čo najlepšie výsledky. Použili sme 1-rozmerné konvolučné a LSTM rekurentné neurónové siete, ktoré sme už bližšie popísali v teoretic-

kej časti práce. Zdrojové kódy práce sa nachádzajú na github repozitári na adrese: github.com/MaxRevicky/Classification-of-eclipsing-binary-stars.

V tabuľke 3–7 a 3–8 sme zhrnuli, ktoré datasety a modely boli pri danom experimente použité, opísali sme vstup do modelov a priebeh evaluácie.

	dataset	vstup do modelu	model	evaluácia
Experiment 1	detached_overcontact	horizontálne spojenie 5	Model A	horizontálne spojenie 5
	overcontact_random	filtrov (UBVRI)		filtrov (UBVRI)
Experiment 2	detached_overcontact	1 typ filtra (U)	Model B	po jednom filtri (UBVRI)
	overcontact_random			
Experiment 3	detached_overcontact	vertikálne spojenie 5	Model B	vertikálne spojenie 5
	overcontact_random	filtrov (UBVRI)		filtrov (UBVRI)

Tabuľka 3–7 Zhrnutie modelovania klasifikácie dvojhviezd na dotykové a oddelené systémy

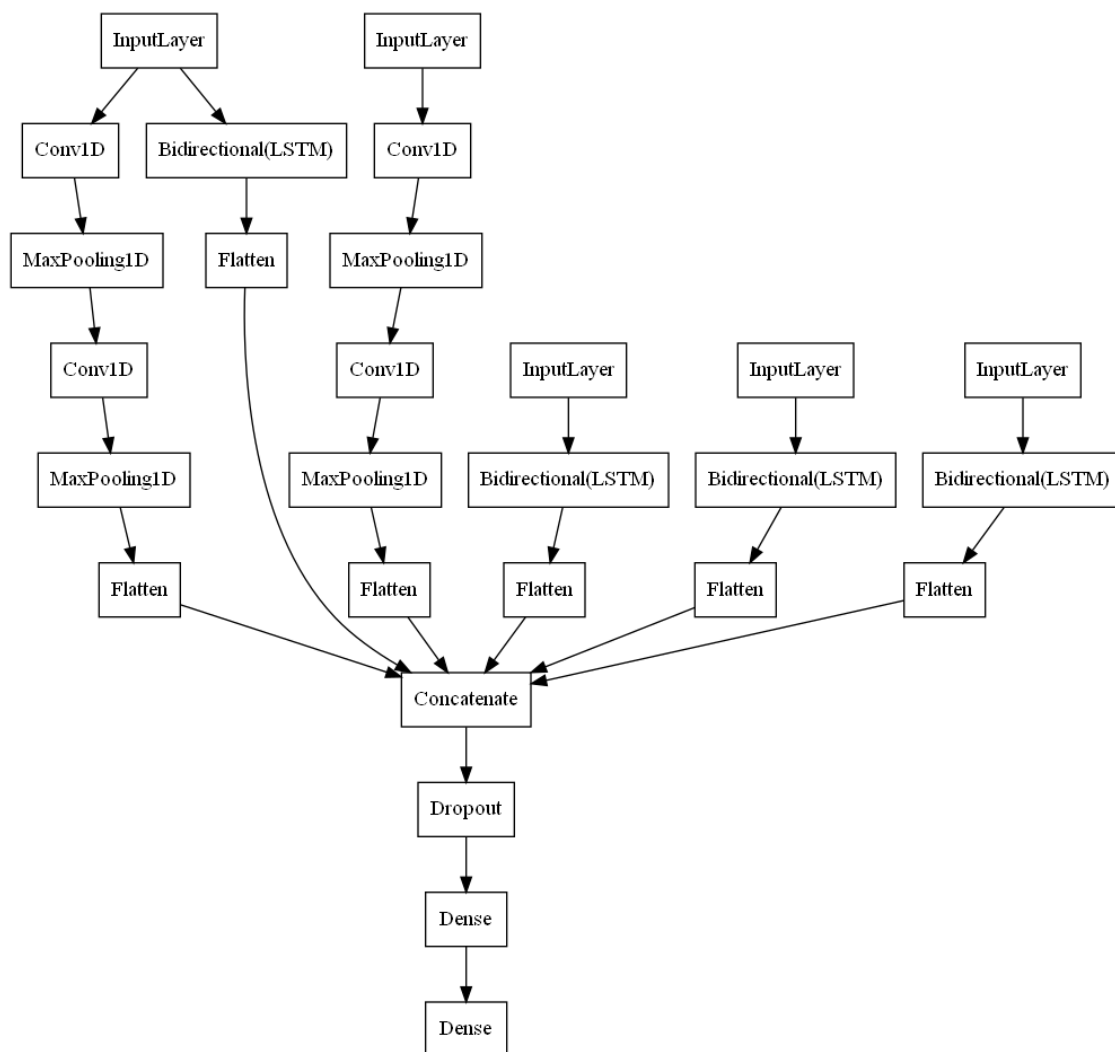
	dataset	vstup do modelu	model	evaluácia
Experiment 4	detached_below_i_crit	vertikálne spojenie 5 filtrov UBVRI	Model B	vertikálne spojenie 5 filtrov UBVRI
	overcontact_below_i_crit			
	single_spotty			

Tabuľka 3–8 Zhrnutie modelovania klasifikácie na dvojhviezdy s kritickým sklonom dráhy a škrvnité hviezdy

3.7.1 Model A

Model A má na vstupe 5 filtrov jednej krivky, ktoré sú spracované paralelne. Prvý filter je spracovávaný paralelne v 1D konvolučnom a LSTM bloku. Druhý filter je spracovaný sekvenčne v 1D konvolučnom bloku. Tretí, štvrtý a piaty filter sú spracované v LSTM blokoch. Výstupy týchto piatich blokov sa skombinovali do jednej vrstvy. Umožnia tak vstup pre plne prepojenú vrstvu s 32 skrytými neurónmi s ReLu aktivačnou funkciou. Regularizačnú metódu dropout sme nastavili na 20%. V modeli pozostávala výstupná vrstva z dvoch neurónov plne prepojených s predchádzajúcou časťou architektúry. Výstupná vrstva mala aktivačnú funkciu Softmax a

klasifikuje výstup do dvoch tried. Zhrnutie architektúry Modelu A je znázornené na obrázku 3–5 a v tabuľke 3–9.



Obrázok 3–5 Vizualizácia Modelu A

Layer (type)	Output Shape	Parameters
input_1 (InputLayer)	[(None, 400, 1)]	0
input_2 (InputLayer)	[(None, 400, 1)]	0
conv1d (Conv1D)	(None, 398, 32)	128
conv1d_2 (Conv1D)	(None, 398, 32)	128
max_pooling1d (MaxPooling1D)	(None, 199, 32)	0

Tabuľka 3 – 9 pokračovanie z predchádzajúcej strany

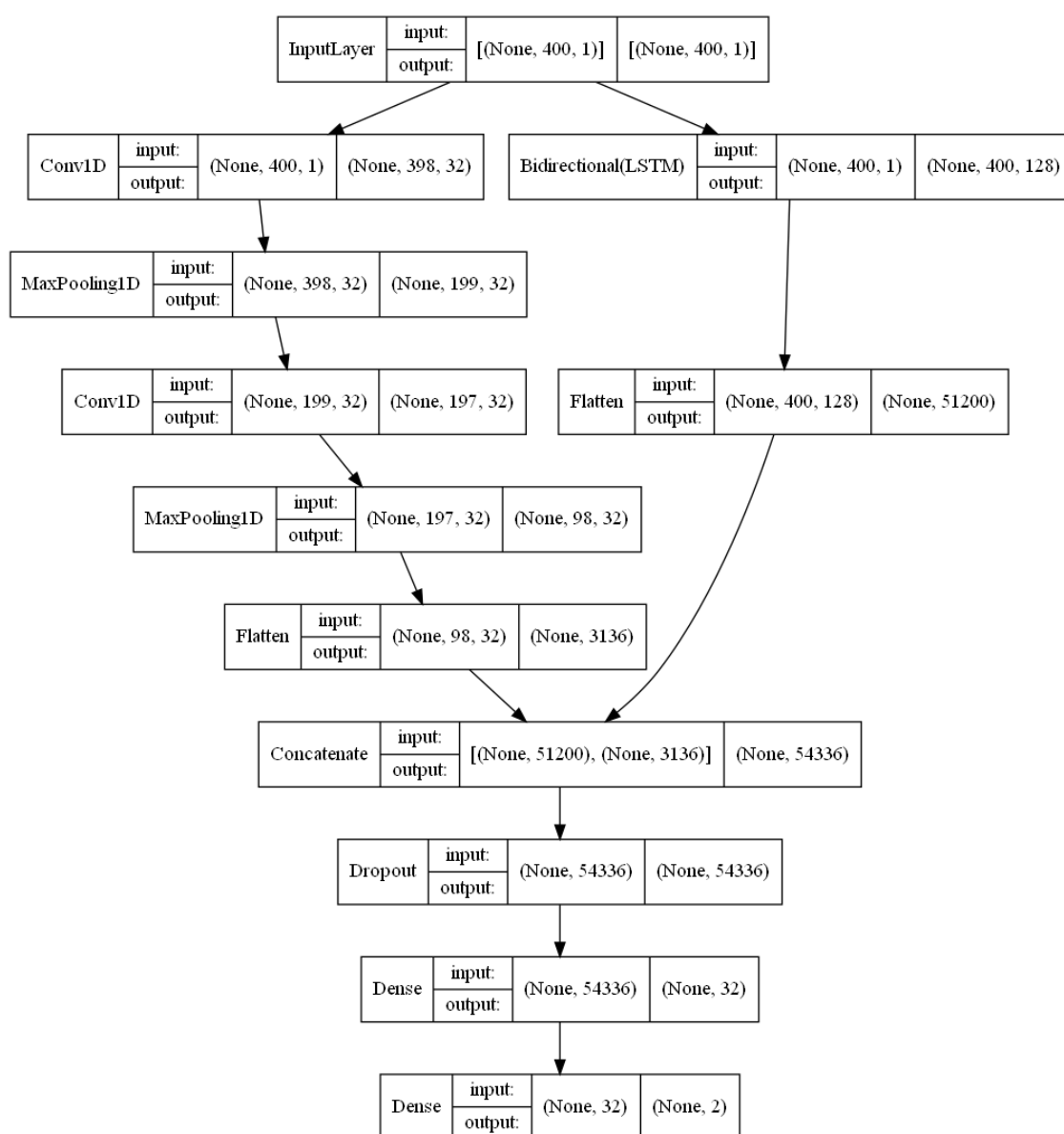
Layer (type)	Output Shape	Parameters
max_pooling1d_2 (MaxPooling1D)	(None, 199, 32)	0
conv1d_1 (Conv1D)	(None, 197, 32)	3104
conv1d_3 (Conv1D)	(None, 197, 32)	3104
input_3 (InputLayer)	[(None, 400, 1)]	0
input_4 (InputLayer)	[(None, 400, 1)]	0
input_5 (InputLayer)	[(None, 400, 1)]	0
bidirectional (Bidirectional)	(None, 400, 128)	33792
max_pooling1d_1 (MaxPooling1D)	(None, 98, 32)	0
max_pooling1d_3 (MaxPooling1D)	(None, 98, 32)	0
bidirectional_1 (Bidirectional)	(None, 400, 128)	33792
bidirectional_2 (Bidirectional)	(None, 400, 128)	33792
bidirectional_3 (Bidirectional)	(None, 400, 128)	33792
flatten (Flatten)	(None, 51200)	0
flatten_1 (Flatten)	(None, 3136)	0
flatten_2 (Flatten)	(None, 3136)	0
flatten_3 (Flatten)	(None, 51200)	0
flatten_4 (Flatten)	(None, 51200)	0
flatten_5 (Flatten)	(None, 51200)	0
concatenate (Concatenate)	(None, 211072)	0
dropout (Dropout)	(None, 211072)	0
dense (Dense)	(None, 32)	6754336
dense_1 (Dense)	(None, 2)	66
Total params: 6,896,034		
Trainable params: 6,896,034		
Non-trainable params: 0		

Tabuľka 3 – 9: Zhrnutie Modelu A

3.7.2 Model B

Na vstupe je krivka pozostávajúca zo 400 číselného vektora. V prvej paralelnej časti sme použili obojsmernú vrstvu LSTM, v druhej časti sme použili dva konvolučné

subbloky (Conv1D). Výstupy týchto dvoch blokov sa skombinovali, aby poskytli vstup pre plne prepojenú vrstvu s 32 skrytými neurónmi s aktivačnou funkciou ReLU. Pri tejto architektúre sme nastavili regularizačnú metódu dropout na 20%. Rovnako ako pri Modeli A, výstupná vrstva pozostávala z dvoch neurónov plne prepojených s predchádzajúcou časťou architektúry a aktivačnou funkciou Softmax. Zhrnutie architektúry Modelu B je znázornené na obrázku 3–6 a v tabuľke 3–10.



Obrázok 3–6 Vizualizácia Modelu B

Layer (type)	Output Shape	Parameters
input_6 (InputLayer)	[(None, 400, 1)]	0
conv1d_4 (Conv1D)	(None, 398, 32)	128
max_pooling1d_4 (MaxPooling1D)	(None, 199, 32)	0
conv1d_5 (Conv1D)	(None, 197, 32)	3104
bidirectional_4 (Bidirectional)	(None, 400, 128)	33792
max_pooling1d_5 (MaxPooling1D)	(None, 98, 32)	0
flatten_6 (Flatten)	(None, 51200)	0
flatten_7 (Flatten)	(None, 3136)	0
concatenate_1 (Concatenate)	(None, 54336)	0
dropout_1 (Dropout)	(None, 54336)	0
dense_2 (Dense)	(None, 32)	1738784
dense_3 (Dense)	(None, 2)	66
Total params: 1,775,874		
Trainable params: 1,775,874		
Non-trainable params: 0		

Tabuľka 3 – 10 Zhrnutie Modelu B

3.7.3 Tréning

Pre automatické ukladanie modelov sme monitorovali **validation accuracy**. Teda, ak v danej epoche model dosiahol vyššiu **validation accuracy** jednotlivé parametre modelu boli uložené. Pri trénovaní bol **batch size** nastavený na 128 a **validation split** na 0,1. Pri klasifikácii dvojhviezd na dotykové a oddelené tréning pozostával z 5 epoch. Už pri prvej epoche bola úspešnosť dosť vysoká. Pri klasifikácii na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy tréning pozostával z 10 epoch, pri prvých epochách sa úspešnosť zvyšuje výrazne, najlepšiu dosiahol pri 9 epoche.

3.7.4 Kritéria úspešnosti

Na interpretáciu kategorických výsledkov použijeme nasledovné hodnoty:

- **Kontingenčná tabuľka klasifikácie** pre binárne hodnoty:
 - **Skutočné pozitíva** (skr. TP, angl. *True Positives*): Hodnoty, ktoré sú v skutočnosti pravdivé boli aj predikované ako pravdivé. Čím väčšia hodnota tým lepšie.
 - **Falošné pozitíva** (skr. FP, angl. *False Positives*): Hodnoty, ktoré sú v skutočnosti nepravdivé, ale boli predikované ako pravdivé – (Chyba Typu 1). Čím menšia hodnota tým lepšie.
 - **Falošné negatíva** (skr. FN, angl. *False Negatives*): Hodnoty, ktoré sú v skutočnosti pravdivé, ale boli predikované ako nepravdivé – (Chyba Typu 2). Čím menšia hodnota tým lepšie.
 - **Skutočné negatíva** (skr. TN, angl. *True Negatives*): Hodnoty, ktoré sú v skutočnosti nepravdivé boli aj predikované ako nepravdivé. Čím väčšia hodnota tým lepšie.
- **Presnosť**: schopnosť klasifikátora neoznačiť vzorku, ktorá je v skutočnosti negatívna za pozitívnu. Čím bližšie k 1, tým lepšie.

$$\frac{TP}{TP + FP} \quad (3.1)$$

- **Návratnosť**: schopnosť klasifikátora nájsť všetky pozitívne prípady. Čím bližšie k 1, tým lepšie.

$$\frac{TP}{TP + FN} \quad (3.2)$$

- **F1 skóre**: harmonický priemer presnosti a návratnosti: Zahrňuje presnosť aj návratnosť, ale dáva im rovnakú váhu, čo nie je veľmi dobré v prípade nevyváženého cieľového atribútu alebo v prípade keď jeden typ chyby je preferovaný pred druhou. Čím väčšie skóre tým lepšie. Najlepšie skóre je 1,0 a najhoršie 0,0.
- **ROC oblasť** (angl. *Receiver Operating Characteristic*): ROC krivka ukazuje vzťah pomeru TP a FP. (Do grafu, kde x-ová os reprezentuje FP-rate a y-ová

TP-rate, zakreslí krivku pozostávajúcu z bodov predikcie). Dáva nám informáciu o tom aké TP/FP budeme mať pri danom FP/TP. Oblasťou sa rozumie oblasť pod krivkou.

3.7.5 Vyhodnotenie experimentov na testovacej množine syntetických dát

Pri vyhodnotení **klasifikácie dvojhviezd na oddelené a dotykové**, vyšlo pre metriky presnosť, návratnosť a F1 skóre, pri všetkých 3 experimentoch viac ako 98%. Výsledky sú pre zvýšenie prehľadnosti uvedené v prílohe A.

- **Experiment 1:** pri tomto experimente sme ako pri jedinom použili Model A (pozri 3.7.1). Tento model je robustný, ale je komplikovanejší. Každý filter vstupuje do inej vetvy modelu a čas tréovania bol pri použití tohto modelu najdlhší. Pri Modeli A potrebujeme na vstupe krivku v každom filtri, čo pri observačných dátach nemusí byť reálne. Výsledky z Experimentu 1 sú uvedené v tabuľke A – 1. Kontingenčná tabuľka klasifikácie a ROC krivku tohto experimentu môžeme vidieť na obrázku A – 1.
- **Experiment 2:** pri tomto experimente sme použili Model B (pozri 3.7.2). Model B je tiež robustný ale jednoduchší a čas tréovania je kratší. Výsledky z tohto experimentu sú rozdelené v 5 tabuľkách a 5 kontingenčných tabuľkách klasifikácie, keďže filtre boli evaluované po jednom. Kontingenčné tabuľky klasifikácie a ROC krivky môžeme vidieť na obrázku A – 2, A – 3, A – 4, A – 5, A – 6. Zhrnutie výsledkov metrík experimentu 2 sú zobrazené v tabuľkách A – 2, A – 3, A – 4, A – 5, A – 6.
- **Experiment 3:** najlepšie výsledky mal tento experiment, pri ktorom sme použili tiež Model B (pozri 3.7.2). Ako vstup do modelu a aj pri evaluácii boli všetky filtre, s ktorými pracujeme spojené vertikálne do jedného datasetu. Pri tomto experimente vyšli všetky metriky takmer 100%. (pozri tab. A – 7)

Nášmu modelu sa podarilo úspešne klasifikovať 49 620 z 50 000 dotykových kriviek a 49 973 z 50 000 oddelených kriviek (pozri obr. A – 7).

Pri **klasifikácií na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy**, sme dosiahli najlepšie výsledky pri použití modelu a vstupu do modelu z Experimentu 3 predošlej klasifikačnej úlohy. Tejto klasifikácii sa venuje Experiment 4, výsledok je zobrazený v prílohe A.

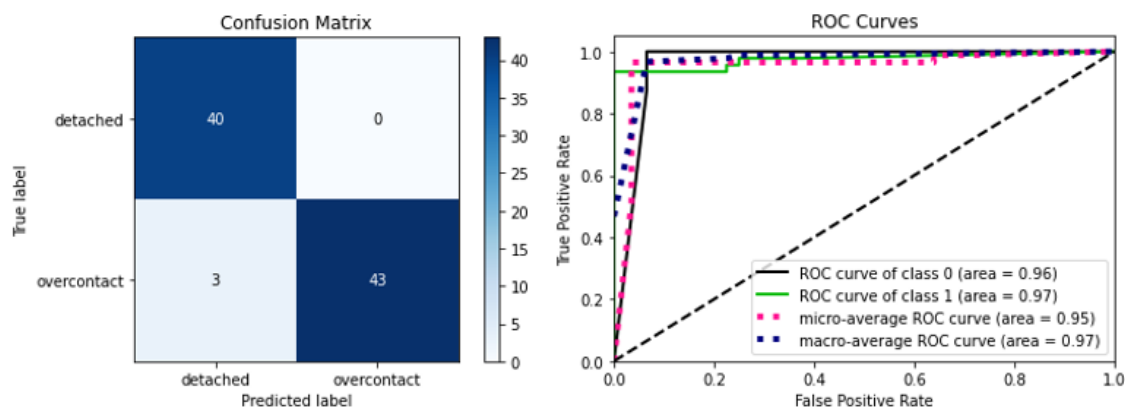
- **Experiment 4:** Model dosiahol priemerná presnosť 89%, návratnosť a F1 skóre 88% (pozri tab. A – 8). Náš model správne predikoval 18 740 z 19 940 kriviek binárnych hviezd a 5 937 zo 6 915 škvrnitých hviezd (pozri obr. A – 8). Domnievame sa, že výsledky pri tejto klasifikácii boli trochu slabšie, pretože krivky s kritickým sklonom dráhy a krivky škvrnitých hviezd sú veľmi podobné.

3.8 Vyhodnotenie

Uskutočnili sme viacero experimentov, vytvorili 2 modely konvolučnej a rekurentnej neurónovej siete. Skúšali sme rôzne typy vstupov do modelov. Úlohou týchto experimentov bolo vytvorenie modelu a zvolenie takého typu vstupu, ktorý by dosahoval najlepšie výsledky pri daných klasifikačných úlohách. Na základe výsledkov experimentov na syntetických dátach sme zistili, že najlepší model na riešenie klasifikácie je paralelná kombinácia 1-rozmerných konvolučných a BiLSTM rekurentných neurónových sietí, kde ako vstup do modelu sú všetky filtre, s ktorými pracujeme, spojené vertikálne do jedného datasetu. Keďže dosahoval najlepšie výsledky a je z našich modelov najrobustnejší, použili sme ho pri testovaní na observačných dátach.

Ako sme už spomínali v podkapitole Pochopenie dát 3.5, na otestovanie nášho modelu pre klasifikáciu dvojité hviezd na oddelené a dotykové, bolo poskytnutých 86 kriviek. Nášmu modelu sa nepodarilo správne klasifikovať iba 3 krivky, ktoré náš model predikoval ako oddelené krivky, ale správne mali byť predikované ako dotykové krivky. Testovanie observačných dát na tomto modeli dosiahlo celkovú presnosť, návratnosť a skóre F1 úroveň 97%, čo je o 7% viac ako sme si stanovili za cieľ.

Môžeme teda zhodnotiť, že klasifikáciu dvojhviezd na oddelené a dotykové hviezdy sme úspešne zvládli. Kontingenčnú tabuľku klasifikácie a ROC krivku môžeme vidieť na obrázku 3–7 a zhrnutie výsledkov klasifikácie v tabuľke 3–11.



Obrázok 3–7 Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky v prípade klasifikácie dvojhviezd na oddelené a dotykové systémy

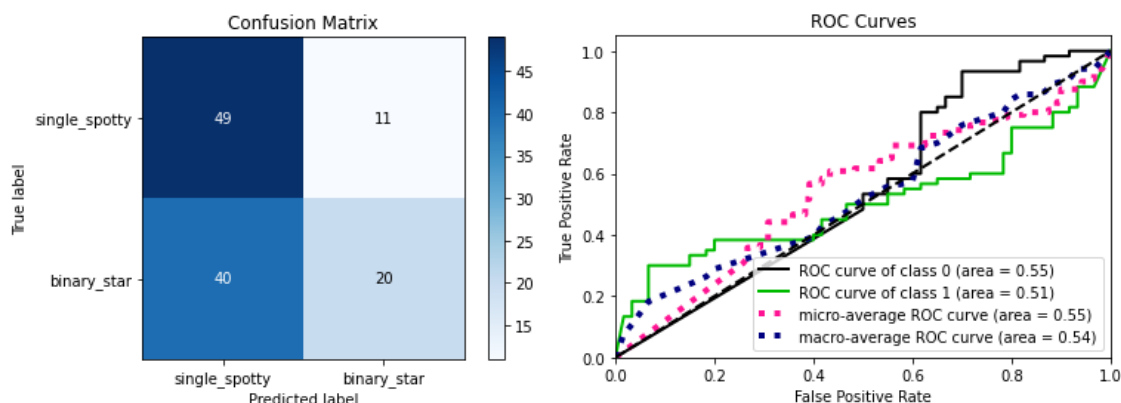
	Presnosť	Návratnosť	F1 skóre	Počet
Oddelené systémy	93%	100%	96%	40
Dotykové systémy	100%	93%	97%	46
macro priemer	97%	97%	97%	86

Tabuľka 3–11 Zhrnutie výsledkov klasifikácie dvojhviezd na oddelené a dotykové systémy

	Presnosť	Návratnosť	F1 skóre	Počet
Dvojhviezdy__krit	55%	82%	66%	60
Škvrnité hviezdy	65%	33%	44%	60
macro priemer	60%	57%	55%	120

Tabuľka 3–12 Zhrnutie výsledkov klasifikácie na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy

Na otestovanie modelu pre klasifikáciu na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy bolo použitých 120 kriviek. Aj keď na syntetických dátach metríky dosiahli okolo 90%, pri observačných tomu tak nebolo. Priemerná presnosť bola



Obrázok 3–8 Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky v prípade klasifikácie na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy

63%, návratnosť 58% a F1 skóre 54% (pozri tab. 3–12). Po konzultácii s doménovými expertmi sme zistili, že presnosť nedosahuje také výsledky ako so syntetickými dátami, lebo v reálnych dátach sa nachádzajú objekty presahujúce typ našich frézovacích dát, s ktorými si model neporadí. Kontingenčnú tabuľku klasifikácie a ROC krivku môžeme vidieť na obrázku 3–8.

3.9 Nasadenie

Najlepšie výsledky meraných metrík (presnosť, návratnosť a F1 skóre) sme získali pri navrhnutom a netrénovanom modeli hybridnej 1 rozmernej konvolučnej a LSTM rekurentnej neurónovej siete, kde vstup do modelu je vertikálne spojenie 5 filtrov UBVRI. Má realizovateľné využitie pri automatizácii procesov vo vedeckých výskumných centrách, ktoré sa venujú aj klasifikácii zákrytových premenných dvojhviezd na oddelené a dotykové. Bude slúžiť ako rozšírenie pre existujúci softvér Elisa. Pri klasifikácii svetelných kriviek na dvojhviezdy s kritickým sklonom dráhy a škvrnité hviezdy je potrebné pre model získať nové vhodne zozbierané dáta, aby umožnili relevantné testovanie a následne nasadenie.

4 Záver

Spolupráca medzi astrofyzikmi a informatikmi nám umožnila identifikáciu a spracovanie veľkých dát pomocou neurónovej siete aplikovaním umelej inteligencie.

V tejto bakalárskej práci sme použitím metód hlbokého učenia riešili úlohy automatickej klasifikácie v doméne zákrytových svetelných hviezd.

Prvým cieľom práce bolo podať teoretický prehľad problematiky metód hlbokého učenia, s dôrazom na vhodné metódy pre automatickú klasifikáciu zákrytových premenných hviezd. Tomuto cieľu sme sa venovali v kapitole 1, kde sme vysvetlili základné poznatky o hlbokom učení. Opísali sme rôzne funkcie a algoritmy vhodné pre túto oblasť. V kapitole 2 sme pokračovali detailnou analýzou klasifikácie svetelných kriviek a analýzou súčasného stavu riešení.

Ďalšími cieľmi bolo analyzovať a predspracovať dostupnú dátovú množinu svetelných kriviek zákrytových premenných hviezd. Navrhnuť postupy pre ich klasifikáciu pomocou zvolených metód hlbokého učenia a následne vyhodnotiť navrhnuté postupy na danej dátovej množine. Týmto cieľom sme sa venovali v kapitole 3. V praktickej časti sme postupovali podľa metodiky CRISP-DM. Zistili sme, že najlepšie výsledky dosiahneme použitím kombinácie paralelných 1-rozmerných konvolučných sietí a BiLSTM rekurentných sietí. Vytvorili sme 2 modely spomínaných architektur. Realizovali sme viacero experimentov, v ktorých sme menili buď vstupné dáta, modely alebo vstupy do modelov. Riešili sme 2 klasifikačné úlohy:

(1) Binárnu klasifikáciu dvojhviezd na oddelené a dotykové, pri ktorých priemerné merané metriky dosiahli pri vyhodnotení 97% na observačných dátach. Tým sa nám podarilo splniť zadaný cieľ. Ako možné rozšírenie tejto úlohy v budúcej práci je pridanie ďalších aj menej používaných filtrov a tým zväčšiť robustnosť modelu.

(2) Binárnu klasifikáciu na dvojhviezdy s kritickým sklonom dráhy a škrvnité hviezdy, ktoré dosiahli slabšie výsledky oproti syntetickým dátam. Nepodarilo sa nám dodržať stanovený cieľ aspoň 80%. Výsledky sme konzultovali s doménovými expertmi, ktorí po analýze zistili, že v reálnych dátach sa nachádzajú objekty, ktoré

sú typovo mimo našich trénovacích dát, s ktorými si tento model nevie poradiť. Ako riešenie pre zlepšenie výsledkov tejto úlohy vidíme možnosť zozbierania nových vhodne zatriedených dát, aby umožnili relevantné testovanie.

V kooperácii s tímom astrofyzikov z Univerzity Pavla Jozefa Šafárika, ktorí nám dodali dáta, sa na základe výborných výsledkov klasifikácie dvojhviezd na oddelené a dotykové systémy pripravuje publikácia pre vedecký časopis MNRAS (angl. *Monthly Notices of the Royal Astronomical Society*) zaoberajúci sa výskumom v astronómii a astrofyzike.

Literatúra

- Abt, H. A. (1983). Normal and abnormal binary frequencies, *Annual Review of Astronomy and Astrophysics* **21**(1): 343–372. <https://www.annualreviews.org/doi/10.1146/annurev.aa.21.090183.002015>.
- Alamsyah, A., Saputra, M. A. A. and Masrury, R. A. (2019). Object detection using convolutional neural network to identify popular fashion product, *Journal of Physics: Conference Series* **1192**.
- Brownlee, J. (2018). *Better Deep Learning: Train Faster, Reduce Overfitting, and Make Better Predictions*, Machine Learning Mastery.
- Carrasco-Davis, R., Cabrera-Vives, G., Förster, F., Estévez, P. A., Huijse, P., Protopapas, P., Reyes, I., Martínez-Palomera, J. and Donoso, C. (2019). Deep learning for image sequence classification of astronomical events, *Publications of the Astronomical Society of the Pacific* **131**(1004). <https://iopscience.iop.org/article/10.1088/1538-3873/aaef12>.
- Chollet, F. (2021). *Deep Learning with Python*, 2 edn, Manning Publications.
- Cokina, M., Maslej-Krešňáková, V., Butka, P. and Parimucha, (2021). Automatic classification of eclipsing binary stars using deep learning methods, *Astronomy and Computing* **36**. <https://linkinghub.elsevier.com/retrieve/pii/S2213133721000421>.
- Fedurco, M., Cokina, M. and Parimucha, (2020). Light curve modelling of close eclipsing binaries, *Contributions of the Astronomical Observatory Skalnaté Pleso* **50**(2). http://www.astro.sk/caosp/Edition/Abstracts/2020/Vol_50/No_2/pp563-564_abstract.html.
- Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep learning*, MIT press.

- Hurley, J. R., Tout, C. A. and Pols, O. R. (2002). Evolution of binary stars and the effect of tides on binary populations, *Monthly Notices of the Royal Astronomical Society* **329**(4): 897–928. <https://academic.oup.com/mnras/article-lookup/doi/10.1046/j.1365-8711.2002.05038.x>.
- Kallrath, J. and Milone, E. (2009). *Eclipsing Binary Stars: Modeling and Analysis*, Springer.
- Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization, *International Conference on Learning Representations*.
- Kiranyaz, S., Avcı, O., Abdeljaber, O., Ince, T., Gabbouj, M. and Inman, D. J. (2021). 1d convolutional neural networks and applications, *Mechanical Systems and Signal Processing* **151**. <https://linkinghub.elsevier.com/retrieve/pii/S0888327020307846>.
- Kostadinov, S. (2018). *Recurrent Neural Networks with Python Quick Start Guide*, Packt Publishing.
- Kuri-Morales, A. (2017). Closed determination of the number of neurons in the hidden layer of a multi-layered perceptron network, *Soft Computing* **21**(3): 597–609. <http://link.springer.com/10.1007/s00500-016-2416-3>.
- Kussul, N., Lavreniuk, M., Skakun, S. and Shelestov, A. (2017). Deep learning classification of land cover and crop types using remote sensing data, *IEEE Geoscience and Remote Sensing Letters* **14**(5): 778–782.
- Malkov, O. Y., Oblak, E., Avvakumova, E. A. and Torra, J. (2007). A procedure for the classification of eclipsing binaries, *Astronomy & Astrophysics* **465**(2): 549–556. <http://www.aanda.org/10.1051/0004-6361:20066103>.
- Mikulášek, Z. and Zejda, M. (2013). *Úvod do studia proměnných hvězd*, vyd. 1 edn, Masarykova univerzita, Brno.

-
- Oracle (n.d.). What is deep learning? <https://www.oracle.com/data-science/machine-learning/what-is-deep-learning/>.
- Orosz, J. A. and Hauschildt, P. H. (2000). The use of the nextgen model atmospheres for cool giants in a light curve synthesis code. <https://arxiv.org/abs/astro-ph/0010114>.
- Paralič, J. (2003). *Objavovanie znalostí v databázach*, 1 edn, Elfa, Košice.
- Perceptrons and Multi-Layer Perceptrons* (n.d.). <https://indiantechwarrior.com/perceptrons-and-multi-layer-perceptrons/>.
- Percy, J. R. (2011). *Understanding variable stars*, 1st paperback ed edn, Cambridge University Press, New York.
- Prsa, A. (2018). *Modeling and Analysis of Eclipsing Binary Stars*, IOP Publishing.
- Prša, A., Conroy, K. E., Horvat, M., Pablo, H., Kochoska, A., Bloemen, S., Giammarco, J., Hambleton, K. M. and Degroote, P. (2016). Physics of eclipsing binaries. ii. toward the increased model fidelity, *The Astrophysical Journal Supplement Series* **227**(2): 29. <https://iopscience.iop.org/article/10.3847/1538-4365/227/2/29>.
- Prša, A. and Zwitter, T. (2005). A computational guide to physics of eclipsing binaries. i. demonstrations and perspectives, *The Astrophysical Journal* **628**(1): 426–438. <https://iopscience.iop.org/article/10.1086/430591>.
- Purnawansyah, P., Haviluddin, H., Darwis, H., Azis, H. and Salim, Y. (2021). Backpropagation neural network with combination of activation functions for inbound traffic prediction, *Knowledge Engineering and Data Science* **4**(1): 14–28. <http://journal2.um.ac.id/index.php/keds/article/view/20008>.
- Saitoh, K. (2021). *Deep Learning from the Basics*, Packt Publishing.
-

-
- Sajjad, M., Khan, Z. A., Ullah, A., Hussain, T., Ullah, W., Lee, M. Y. and Baik, S. W. (2020). A novel cnn-gru-based hybrid approach for short-term residential load forecasting, *IEEE Access* **8**: 143759–143768.
- Sarro, L. M., Sánchez-Fernández, C. and Giménez, (2006). Automatic classification of eclipsing binaries light curves using neural networks, *Astronomy & Astrophysics* **446**(1): 395–402. <http://www.aanda.org/10.1051/0004-6361:20052830>.
- Schuster, M. and Paliwal, K. (1997). Bidirectional recurrent neural networks, *IEEE Transactions on Signal Processing* **45**(11): 2673–2681. <http://ieeexplore.ieee.org/document/650093/>.
- Sewak, M., Karim, M. R. and Pujari, P. (2018). *Practical Convolutional Neural Networks: Implement advanced deep learning models using Python*, Packt Publishing.
- Sharma, S., Sharma, S. and Athaiya, A. (2020). Activation functions in neural networks, *International Journal of Engineering Applied Sciences and Technology* **4**(12): 310–316. <https://www.ijeast.com/papers/310-316,Tesma412,IJEAST.pdf>.
- Smagulova, K. and James, A. P. (2020). Overview of long short-term memory neural networks, *Deep Learning Classifiers with Memristive Networks*, Springer International Publishing, Cham, pp. 139–153. http://link.springer.com/10.1007/978-3-030-14524-8_11.
- Srivastava, N., Hinton, G., Sutskever, Ruslan, Sutskever, I., Salakhutdinov, R. and Bengio, Y. (2014). Dropout: A simple way to prevent neural networks from overfitting, *Journal of Machine Learning Research* pp. 1929–1958. <https://www.jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf>.
- Stephen, J., Rohil, H. and Vasav, S. (2014). *Computer Science, Communication and Instrumentation Devices*.
-

- Terrell, D. (2001). Eclipsing binary stars: Past, present, and future, *Journal of the American Association of Variable Star Observers (JAAVSO)* pp. 1–15.
- Trivedi, U. B., Bhatt, M. and Srivastava, P. (2021). Prevent overfitting problem in machine learning, *Innovations in Information and Communication Technologies (IICT-2020)*, Springer International Publishing, Cham, pp. 345–349. https://link.springer.com/10.1007/978-3-030-66218-9_40.
- Vasilev, I., Slater, D., Spacagna, G., Roelants, P. and Zocca, V. (2019). *Python Deep Learning: Exploring deep learning techniques, neural network architectures and GANs with PyTorch, Keras and TensorFlow*, 2nd ed. edn, Packt Publishing.
- Wilson, R. E. (1979). Eccentric orbit generalization and simultaneous solution of binary star light and velocity curves, *The Astrophysical Journal* **234**. <http://adsabs.harvard.edu/doi/10.1086/157588>.
- Wilson, R. E. and Devinney, E. J. (1971). Realization of accurate close-binary light curves, *The Astrophysical Journal* **166**. <http://adsabs.harvard.edu/doi/10.1086/150986>.
- Zhou, K., and Fichtinger, G. (2020). *Handbook of medical image computing and computer assisted intervention*, Elsevier.

Zoznam príloh

Príloha A Zhrnutie výsledkov, vizualizácia kontingenčných tabuliek klasifikácie a ROC kriviek z podkapitoly 3.7.5 Vyhodnotenie experimentov na testovacej množine syntetických dát

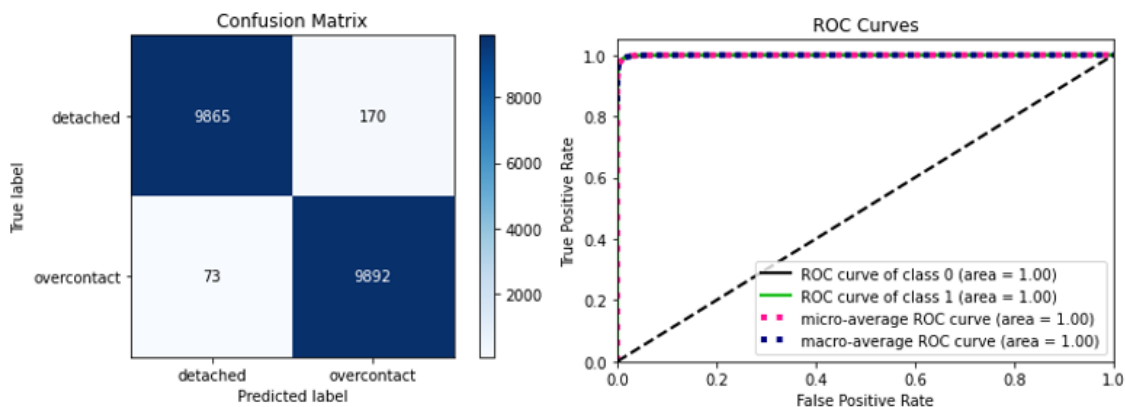
Príloha B CD médium – stav github repozitáru github.com/MaxRevicky/Classification-of-eclipsing-binary-stars v čase odovzdania:

- zdrojové kódy použité pri predspracovaní datasetov: `data-preparation`,
- datasey s observačnými dátami: `datasets`,
- zdrojové kódy použité pri vytvorení a vyhodnotení modelov: `modeling`,
- uložené natrénované modely: `models-checkpoints`,
- základné inštrukcie: `README.md`,
- systémovú príručku v elektronickej forme: `SP_Revicky_Maximilian.pdf`,
- používateľskú príručku v elektronickej forme: `PP_Revicky_Maximilian.pdf`,
- bakalárska práca v elektronickej forme: `BP_Revicky_Maximilian.pdf`.

Príloha C Používateľská príručka

Príloha D Systémová príručka

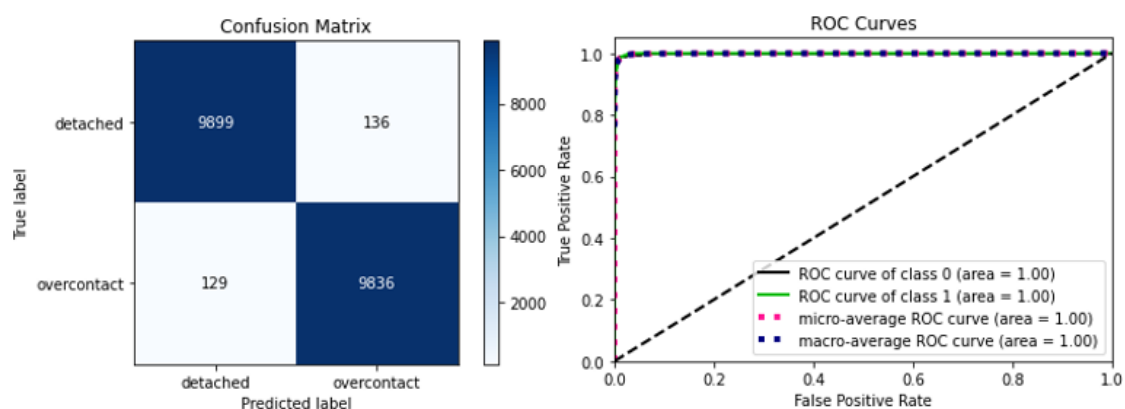
Príloha A



Obrázok A – 1 Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 1

Experiment 1	Presnosť	Návratnosť	F1 skóre	Počet
Oddelené systémy	99%	98%	99%	10 035
Dotykové systémy	98%	99%	99%	9 965
macro priemer	99%	99%	99%	20 000

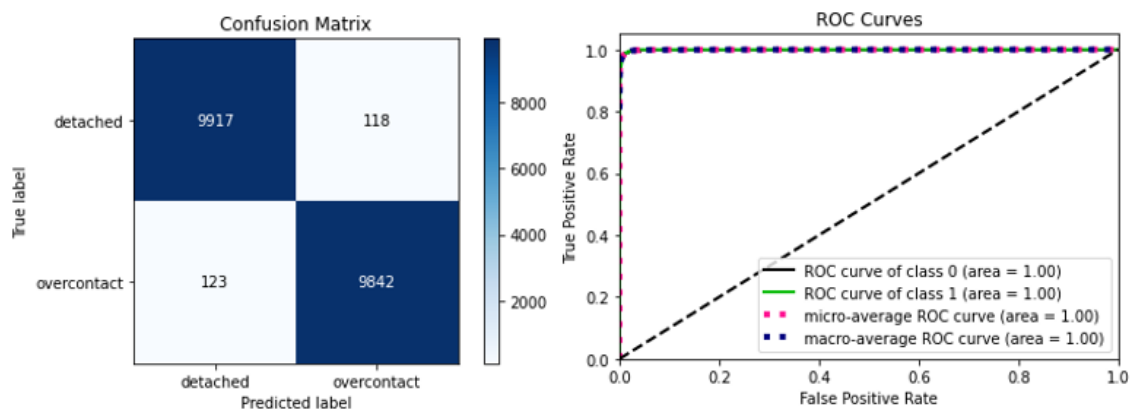
Tabuľka A – 1 Zhrnutie výsledkov Experimentu 1 na testovacej množine



Obrázok A – 2 Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 2a (filter U)

Experiment 2a	Presnosť	Návratnosť	F1 skóre	Počet
Oddelené systémy	99%	99%	99%	10 035
Dotykové systémy	99%	99%	99%	9 965
macro priemer	99%	99%	99%	20 000

Tabuľka A – 2 Zhrnutie výsledkov Experimentu 2a (filter U) na testovacej množine



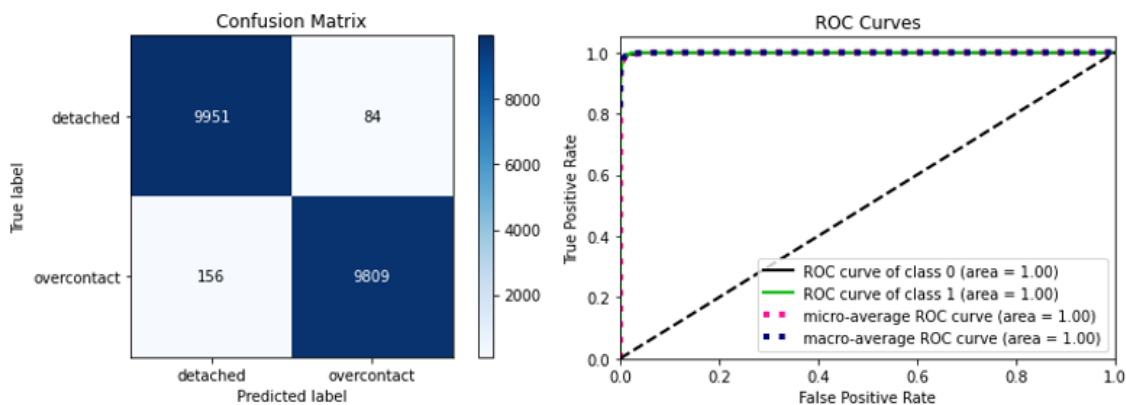
Obrázok A – 3 Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 2b (filter B)

Experiment 2b	Presnosť	Návratnosť	F1 skóre	Počet
Oddelené systémy	99%	99%	99%	10 035
Dotykové systémy	99%	99%	99%	9 965
macro priemer	99%	99%	99%	20 000

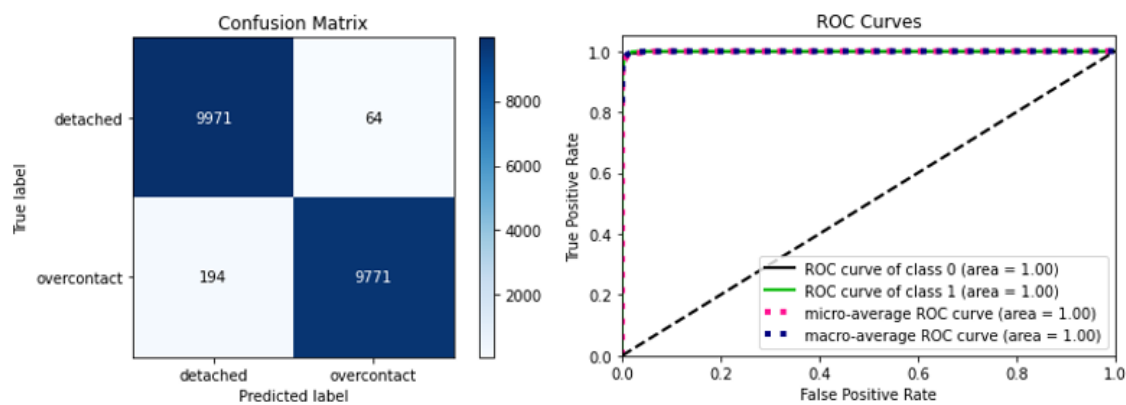
Tabuľka A – 3 Zhrnutie výsledkov Experimentu 2b (filter B) na testovacej množine

Experiment 2c	Presnosť	Návratnosť	F1 skóre	Počet
Oddelené systémy	98%	99%	99%	10 035
Dotykové systémy	99%	98%	99%	9 965
macro priemer	99%	99%	99%	20 000

Tabuľka A – 4 Zhrnutie výsledkov Experimentu 2c (filter V) na testovacej množine



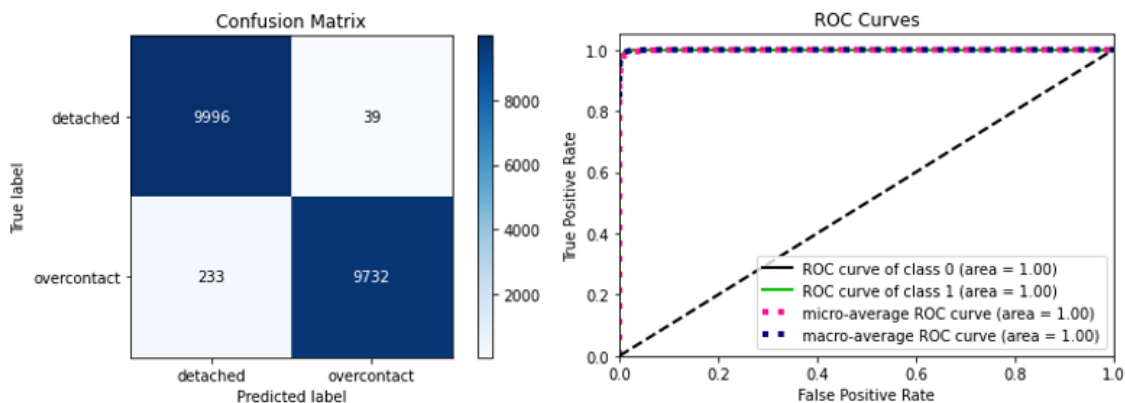
Obrázok A – 4 Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 2c (filter V)



Obrázok A – 5 Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 2d (filter R)

Experiment 2d	Presnosť	Návratnosť	F1 skóre	Počet
Oddelené systémy	98%	99%	99%	10 035
Dotykové systémy	99%	98%	99%	9 965
macro priemer	99%	99%	99%	20 000

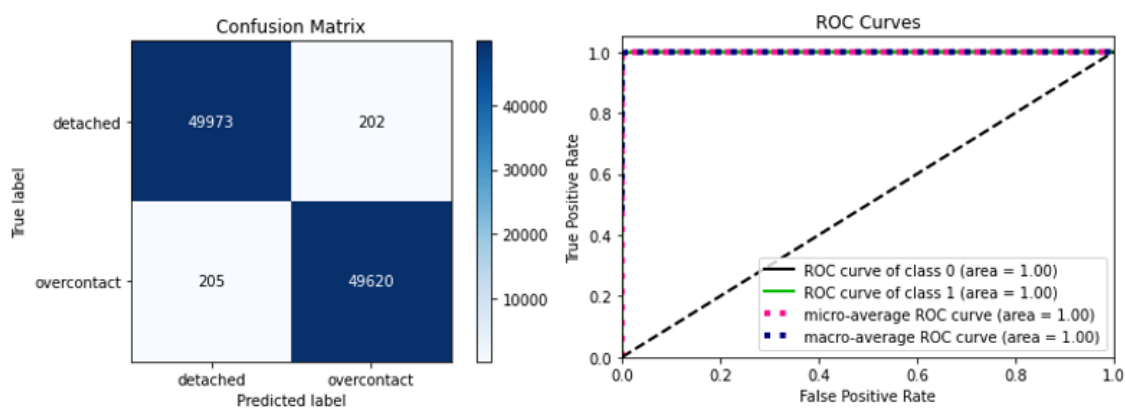
Tabuľka A – 5 Zhrnutie výsledkov Experimentu 2d (filter R) na testovacej množine



Obrázok A – 6 Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 2e (filter I)

Experiment 2e	Presnosť	Návratnosť	F1 skóre	Počet
Oddelené systémy	98%	100%	99%	10 035
Dotykové systémy	100%	98%	99%	9 965
macro priemer	99%	99%	99%	20 000

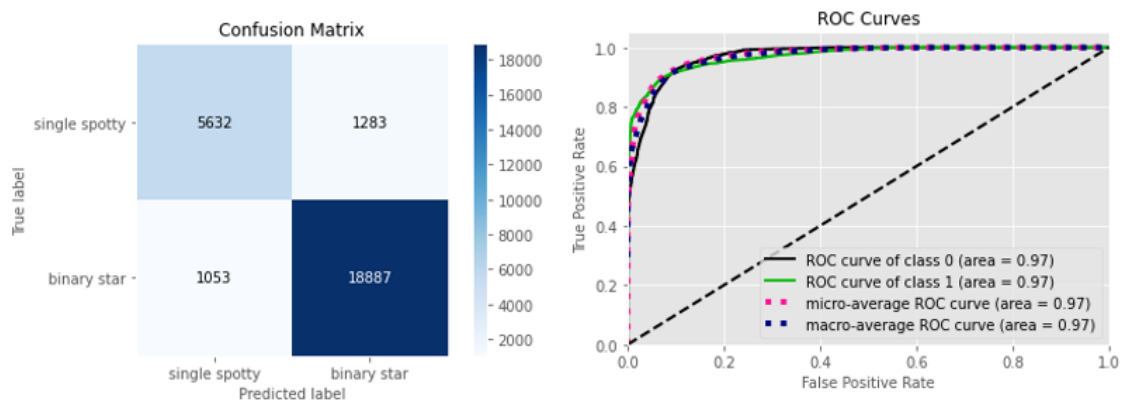
Tabuľka A – 6 Zhrnutie výsledkov Experimentu 2e (filter I) na testovacej množine



Obrázok A – 7 Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 3

Experiment 3	Presnosť	Návratnosť	F1 skóre	Počet
Oddelené systémy	100%	100%	100%	50 175
Dotykové systémy	100%	100%	100%	49 825
macro priemer	100%	100%	100%	100 000

Tabuľka A – 7 Zhrnutie výsledkov Experimentu 3 na testovacej množine



Obrázok A – 8 Vizualizácia kontingenčnej tabuľky klasifikácie a ROC krivky Experimentu 4

Experiment 4	Presnosť	Návratnosť	F1 skóre	Počet
Dvojhviezdy_krit	84%	81%	83%	6 915
Škvornité hviezdy	94%	95%	94%	19 940
macro priemer	89%	88%	88%	26 855

Tabuľka A – 8 Zhrnutie výsledkov Experimentu 4 na testovacej množine