

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2024

Bc. Jakub Müller



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY

A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

DETEKCE ZAČÁTKU A KONCE KOMPLEXU QRS S VYUŽITÍM HLUBOKÉHO UČENÍ

DETEKCE ZAČÁTKU A KONCE KOMPLEXU QRS S VYUŽITÍM HLUBOKÉHO UČENÍ

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Jakub Müller

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Radovan Smíšek, Ph.D.

BRNO 2024

Diplomová práce

magisterský navazující studijní program **Biomedicínské inženýrství a bioinformatika**

Ústav biomedicínského inženýrství

Student: Bc. Jakub Müller

ID: 211581

Ročník: 2

Akademický rok: 2023/24

NÁZEV TÉMATU:

Detekce začátku a konce komplexu QRS s využitím hlubokého učení

POKyny PRO VYPRACOVÁNÍ:

1) Proveďte rešerši metod pro automatického měření trvání komplexu QRS se zaměřením na metody využívající hluboké učení. 2) Proveďte rešerši databází vhodných pro testování detektoru začátku a konce komplexu QRS. 3) Pomocí hlubokého učení natrénujte model pro detekci začátku a konce komplexu QRS. 4) Optimalizujte navržený algoritmus. 5) Otestujte výsledný algoritmus na alespoň pěti veřejně dostupných databázích a výsledky porovnejte s ostatními autory.

DOPORUČENÁ LITERATURA:

[1] A. Malali, S. Hiriyannaiah, et al. "Supervised ECG wave segmentation using convolutional LSTM", ICT Express, vol. 6, no. 3, pp. 166-169, 2020.

[2] V. Moskalenko, N. Zolotykh, G. Osipov, "Deep Learning for ECG Segmentation", Advances in Neural Computation, Machine Learning, and Cognitive Research III, vol. 856, pp. 246-254, 2020.

Termín zadání: 5.2.2024

Termín odevzdání: 22.5.2024

Vedoucí práce: Ing. Radovan Smíšek, Ph.D.

prof. Ing. Valentine Provazník, 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

Měření EKG je základním diagnostickým nástrojem zdravotního stavu srdce a automatizace jeho analýzy může přispět ve zdravotnictví k odlehčení pracovní zátěže personálu nebo ke zlepšení kvality automatické diagnostiky z nositelných zařízení. Tato práce se zaměřuje konkrétně na komplex QRS v signále EKG, s hlavním cílem aplikování metod hlubokého učení k detekci jeho počátku a konce. V teoretickém úvodu je čtenář seznámen s původem komplexu QRS a měřením EKG, umělými neuronovými sítěmi a hlubokým učáním. Pro implementaci vlastní metody byla zvolena architektura U-Net modifikovaná pro 1D signály. Data byly čerpány z pěti veřejně přístupných databází, jejich předzpracování proběhlo v prostředí Matlab. Následoval přesun do prostředí Python kde byly realizovány části modelu s použitím knihoven TensorFlow a Keras, následné trénování, testování modelu a vyhodnocení výsledků.

KLÍČOVÁ SLOVA

EKG signál, QRS komplex, QRS detekce, QRS počátek, QRS konec, databáze, neuronová síť, hluboké učení, konvoluce, U-Net, Python, TensorFlow, Keras

ABSTRACT

ECG measurement is an essential diagnostic tool for cardiac health, and automation of its analysis can aid our healthcare to relieve staff workload or improve the quality of automated diagnostics from wearable devices. This work focuses specifically on the QRS complex in the ECG signal, with the main goal of using deep learning methods to detect its onset and offset. In the theoretical introduction, the reader is introduced to the origin of the QRS complex and ECG measurements, artificial neural networks and deep learning. Modified architecture U-Net for 1D signals was chosen to implement the actual method. Data were extracted from five publicly available databases and preprocessed in Matlab. This was followed by moving to the Python environment where parts of the model were implemented using the TensorFlow and Keras libraries, subsequent training, testing of the model and evaluation of the results.

KEYWORDS

ECG signal, QRS complex, QRS detection, QRS onset, QRS offset, database, neural network, deep learning, convolution, U-Net, Python, TensorFlow, Keras

MÜLLER, Jakub. *Detekce začátku a konce komplexu QRS s využitím hlubokého učení*. Brno, 2024, 71 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství. Vedoucí práce: Ing. Radovan Smíšek, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Detekce začátku a konce komplexu QRS s využitím hlubokého učení“ jsem vypracoval samostatně pod vedením vedoucího diplomové 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové 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

PODĚKOVÁNÍ

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

Obsah

Úvod	19
1 Srdce a jeho aktivita	21
1.1 Funkce srdce	21
1.2 Srdeční činnost	23
1.3 Převodní systém srdeční	24
1.4 Měření EKG	26
2 Umělé neuronové sítě	29
2.1 Aktivační funkce	30
2.2 Učení sítě	31
2.3 Hluboké učení	32
3 Využití metod hlubokého učení k rozměřování délek komplexů QRS	35
3.1 Segmentace vln EKG pomocí sítě s architekturou U-Net	36
3.2 Segmentace vln EKG s využitím konvoluční LSTM sítě	38
4 Veřejně přístupné databáze	41
4.1 PhysioNet	41
4.2 Databáze QT	42
4.3 Databáze LUDB	43
4.4 Databáze ECGCIPA	44
4.5 Databáze ECGDMMLD	46
4.6 Databáze ECGRDVQ	47
5 Implementace vlastní metody s použitím hlubokého učení	49
5.1 Načítání dat	49
5.2 Předzpracování dat	50
5.3 Architektura modelu	52
5.4 Trénování a optimalizace parametrů modelu	55
5.5 Testování a zpracování výsledků	58
6 Výsledky vybraného modelu a jejich diskuze	61
Závěr	65
Literatura	67
Seznam symbolů, veličin a zkratk	71

Seznam obrázků

1.1	Podélný řez srdeční přepážkou - pohled frontální	22
1.2	Převodní systém srdeční	24
1.3	Pacemakerový potenciál buňky sinusového uzlu	25
1.4	Akční potenciál buňky pracovního myokardu	26
1.5	Popis křivky EKG s jednotlivými úseky	27
2.1	Schéma samotného neuronu	29
2.2	Struktura RNN	33
3.1	Architektura sítě, převzato z [14]	37
3.2	Schéma architektury sítě ConvLSTM	39
4.1	Ukázka QRS komplexu z QTDB	42
4.2	Ukázka dat z LUDB	43
4.3	Ukázka dat z ECGCIPA 1/2	44
4.4	Ukázka dat z ECGCIPA 2/2	45
4.5	Ukázka dat z ECGDMMLD	46
4.6	Ukázka dat z ECGRDVQ	47
5.1	Ukázka vynechání anotace z datasetu LUDB	50
5.2	Ukázka předzpracování na vzorku dat z LUDB, okno 500 ms	51
5.3	Ukázka okna z datasetu LUDB s anotací QRS	52
5.4	Architektura sítě 1D U-Net	55
5.5	Pokles validace za epochu při použití DICE (zelená), proti BKE (červená)	58
5.6	Interval tolerance kolem GT	59

Seznam tabulek

3.1	Výsledky při použití všech svodů [13]	36
3.2	Nejlepší dosažené výsledky autorů modelu z [14]	37
3.3	Dosažené míry úspěšnosti klasifikace modelem ConvLSTM [15]	39
5.1	Parametry použitých dat: F_s – vzorkovací frekvence	50
5.2	Počty záznamů určených pro testování a trénování	52
5.3	Rozdělení dat před trénováním	55
5.4	Hyperparametry modelu: BKE – binární křížová entropie	56
5.5	Parametry 3 nejlepších modelů a srovnání DICE s BKE	57
6.1	Výsledky segmentace QRS	61
6.2	Výsledky QR _{Son} , tolerance 5 ms	61
6.3	Výsledky QR _{Soff} , tolerance 5 ms	62
6.4	Srovnání segmentace QRS na datasetu QTDB s jinými modely	62
6.5	Srovnání výsledků QR _{Son} a QR _{Soff} na databázi QTDB s jinými modely	62
6.6	Srovnání výsledků QR _{Son} a QR _{Soff} na databázi QTDB s jinými modely	63

Seznam výpisů

5.1	Ukázka části kódu složené konvoluční vrstvy	53
5.2	Ukázka části kódu vrstvy podvzorkování	53
5.3	Ukázka části kódu vrstvy zpětného vzorkování	54
5.4	Příklad uložených parametrů jednoho modelu	57
5.5	Výpočet TP, FP a FN pro QR _{Son}	60
5.6	Výpočet TP, FP a FN pro QR _{Soff}	60

Úvod

Srdeční činnost je nedílnou součástí správného fungování našeho organismu a v současné době jsou její poruchy jednou z nejčastějších příčin úmrtí obyvatel západního světa. Proto je důležité mít přehled o stavu srdečního orgánu, který lze do jisté míry automatizovat, čehož lze využít ať už v rámci dlouhodobého nebo *real-time* monitoringu.

K diagnostice srdce se využívá měření potenciálů z pokožky pacienta pomocí řady EKG přístrojů, které se v dnešní době používají. Výstupem měření je záznam, neboli elektrokardiograf, jehož následná analýza slouží k diagnostice srdečního rytmu a zjišťování různých srdečních patologií. Právě zde může mít v klinické praxi potenciálního uplatnění automatizace procesu analýzy a anotace záznamů pomocí algoritmů strojového učení, ze kterých se vyčleňuje oblast hlubokého učení využívající neuronové sítě k modelování komplexních vzorů v datech, jež umožňují jejich pokročilou analýzu a zpracování.

Téma této práce se zabývá výše zmíněnou problematikou se zaměřením na automatickou detekci počátku a konce komplexu QRS v signálech EKG za použití hlubokého učení. Komplex QRS představuje v záznamu EKG depolarizaci srdečních komor.

V teoretických kapitolách práce je čtenář seznámen se základním popisem fyziologie srdečního orgánu, tvorby vzruchů v srdci, principů měření signálu EKG a stručným souhrnem patologií vycházejících z délky trvání QRS. V dalších částech následuje úvod do hlubokého učení, teorie umělých neuronových sítí, jejich architektury a metodiky trénování, spolu s příklady aplikace těchto principů k automatické segmentaci EKG.

V následující části řešerše jsou shrnuty poznatky od jiných autorů týkající se využití hlubokého učení k detekci počátku a konce komplexu QRS s využitím dat z veřejně dostupných databází. Těm se věnuje také následující kapitola shrnující charakteristiku databází použitých pro účely vývoje modelu hlubokého učení a jeho testování.

Praktická část se skládá z několika podkapitol dle postupu práce. Čtenář je seznámen s procesem implementace vlastní metody, počínaje zpracováním dat, návrhem architektury použitého modelu, trénováním, testováním a zpracováním výsledků, které jsou v poslední kapitole vyhodnoceny a porovnány.

1 Srdce a jeho aktivita

Nedílnou součástí fungování všech živých organismů je udržování stálosti vnitřního prostředí (homeostáza). Proces zachování homeostázy probíhá za nejrůznějších stavů, ať už přetrvání vlivů okolí nebo měnících se metabolických nároků organismu. Činnost srdce se spolu s krevním řečištěm stará o neustálou výměnu plynů a živin mezi tkáněmi a krví. Proces zvaný perfúze označuje cirkulaci krve v uzavřeném systému krevního řečiště. Samotnou výměnu plynů a živin, jenž probíhá v místech nejtenčích cév (kapilár) nazýváme difúze. Těmito procesy se srdce začleňuje do jednoho z nezbytných systémů potřebných pro udržení homeostázy [1].

Srdce tvoří spolu s cévami kardiovaskulární systém, ve kterém vykonává práci potřebnou k vypuzení krve. Proudění následuje tlakový spád z místa s tlakem vyšším do místa s tlakem nižším. Krev nejprve putuje směrem od srdce vysokotlakými tepnami, které jsou svou stavbou přizpůsobeny pulsním nárazům krve. Krev se dále dostává do arteriol (užší tepny), tlak řečiště klesá, až doputuje do kapilár, kde probíhá difúze. Zpětně se skrze venuly a vény (žíly) vrací nízkotlakou částí řečiště zpět k srdci.

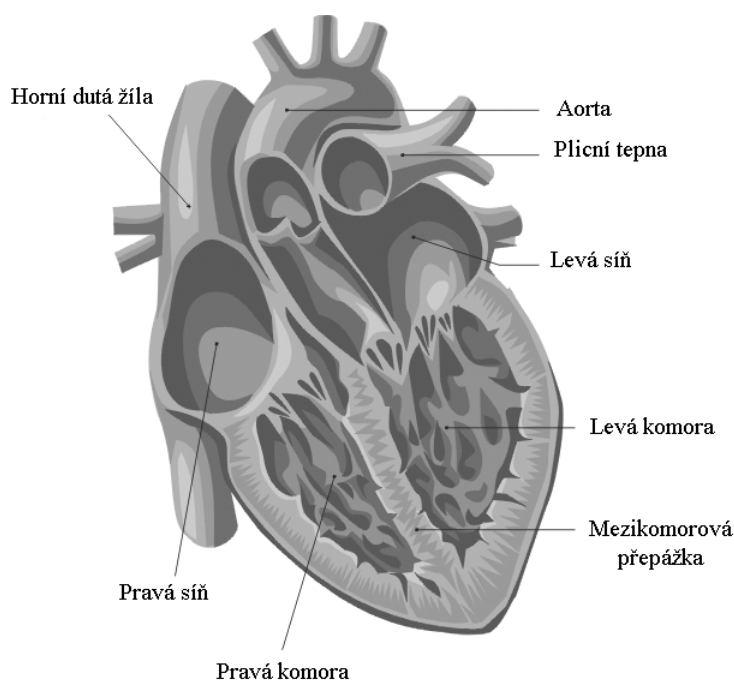
Samotný srdeční orgán tvoří dva páry síní a komor pracujících paralelně vedle sebe (Obr. 1.1). Aby bylo zabráněno zpětnému průtoku krve, jsou od sebe prostory vzájemně odděleny srdečními chlopněmi. Krev se v srdci přečerpává skrze pravou síň a komoru z oběhu velkého do malého (plicního). Zde dochází k jejímu okysličení, jež následuje zpětný přesun krve přes levou síň a komoru do aorty (srdečnice), která tvoří součást velkého tělního oběhu [1] [2].

1.1 Funkce srdce

Srdeční orgán se stará o přečerpávání krve v cévním systému těla. Můžeme tedy říci, že pracuje jako pumpa řízena vlastním autonomním nervovým systémem, jenž udává do pohybu srdeční svalovinu skládající se z kardiomyocytů (buňky srdeční svaloviny). Pro jeho správnou funkci je nezbytné souhlasné stahování pravé a levé síně následované oběma komorami.

Jak již bylo zmíněno, k usměrnění toku krve v srdci slouží systém chlopní. Síně od komor oddělují chlopně atrioventrikulární (cípaté) doplněny chlopněmi semilunárními (poloměsíčitě), jež se nacházejí mezi komorami a arteriemi [1].

Chlopně atrioventrikulární zabírají svými rozměry, při srovnání se semilunárními, větší plochu a krev skrze ně proudí pomaleji. Aby zůstaly během systoly (kontrakce srdce) uzavřené, mají na koncích svých cípů šlašinkami upevněné papilární svaly. Při kontrakci srdce zabraňují vyklenutí cípů do síně (prolaps chlopně). V případě



Obr. 1.1: Podélný řez srdeční přepážkou - pohled frontální, upraveno z [2]

jejich poškození dochází k nedomykavosti chlopní, což vede ke zpětnému toku – regurgitace [1] [3].

Poloměsíčné chlopně mají plochu menší, krev jimi protéká rychleji a pod větším tlakem. Jejich pohyb probíhá pasivně podle tlaku v komoře a arteriích. Při systole se v komoře vytvoří přetlak, chlopně se otevírají a krev je vypuzována ven do arterií. Naopak během diastoly komorový tlak poklesne a zpětným tokem krve dojde k uzavření. Oba páry chlopní svými pohyby vytvářejí také akustické vlnění, které tvoří součást srdečních ozev využívaných při vyšetření srdce poslechem, pomocí fonendoskopu (auskultace) [1].

1.2 Srdeční činnost

Srdce tedy přečerpává krev v pravidelných revolucích nazývaných srdeční cyklus. Během jednoho cyklu proběhne systola – kontrakce a diastola – relaxace srdeční svaloviny. Při relaxaci a kontrakci srdeční svaloviny dochází ke změnám tlaků v srdečních dutinách a v důsledku těchto změn je proudění krve během diastoly označováno jako plnění a při systole jako vypuzení krve. Páry síní a komor pracují se vzájemným časovým posunem, kdy na konec diastoly komor navazuje systola síní, jež plní komory krví. Nastává systola komor, krev je vypuzena do cév, síně mají diastolu a proudí do nich krev. Komory ukončí systolu, naplní se krví ze síní a celý cyklus se periodicky opakuje. V průběhů střídajících se kontrakcí a relaxací nastává děj izovolumické relaxace, kdy se objem krve v komoře nemění důsledkem uzavřených semilunárních chlopní tlakem zpětně proudící krve z arterií v kombinaci s uzavřenými chlopněmi atrioventrikulárními [1].

Samotnou diastolu komor lze rozdělit na dvě části dle rychlosti plnění komor krví. Z počátku se komory plní rychle, ale se zvyšujícím se tlakem rychlost klesá a nastává etapa pomalého plnění následována systolou síní. Systola komor se dělí na dvě různé fáze. Z počátku nastává izovolumická kontrakce, během níž se uzavírají cípate chlopně, tlak komor zůstává na nízkých hodnotách a semilunární chlopně jsou uzavřeny. Hodnota tlaku roste, dokud nepřevyší hodnotu tlaku v arteriích, kdy se otevírají semilunární chlopně a nastává ejekční fáze. Krev je vypuzena do arterií a to nejprve pomalým, pak rychlým vypuzováním. Tím kontrakce končí, semilunární chlopně se uzavírají, probíhá izovolumická relaxace a v komoře zůstává zbytkový objem [1].

Aby nedocházelo ke hromadění krve, mají levá i pravá komora, pokud je měříme desítky vteřin a více, stejnou hodnotu přečerpávaných objemů. Žilní návrat, jakožto objem krve vracející se zpět do komory, by měl být roven objemu, jež komora vypudí za jeden cyklus – srdeční výdej. Pro systolu odlišujeme objem systolický (tepový objem), který činí okolo 70 ml. Po ukončení systoly zůstane z objemu systolického, reziduální (zbytkový) objem, přibližně 50 ml [1].

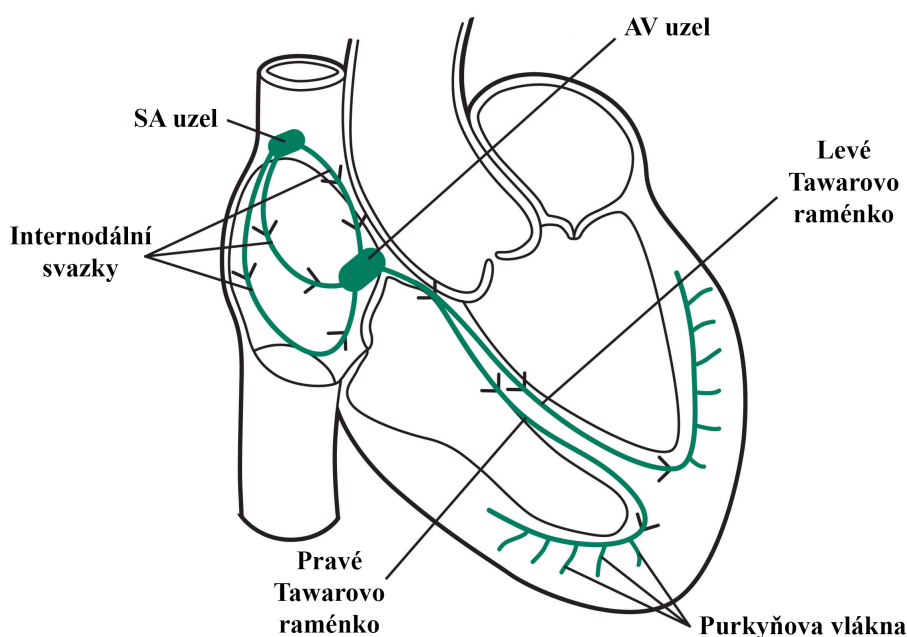
Pomocí těchto popisných veličin jsme schopni charakterizovat výkonnost srdce. Srdeční výdej představuje objem krve vypuzený během systoly, počítaný skrze součin systolického objemu a srdeční frekvence. Každý jedinec má odlišný výdej dle korespondujících proporcí postavy a věku. Vztažením srdečního výdeje na plochu těla dostáváme srdeční index, jež charakterizuje požadavky metabolismu tkání proti srdečnímu výdeji. Fyziologické hodnoty indexu rozsah se pohybuje v rozmezí 3-5 l/min/m² [1].

1.3 Převodní systém srdeční

Schopnost srdce tvořit vlastní vzruchy neboli autonomie, je spolu s automacií, a sice schopností střídání systoly a diastoly, jednou z jeho základních fyziologických vlastností. Podobně lze dělit i druhy svaloviny vykonávající jednotlivé akce. Pracovní svalovina vykonává stahy v závislosti na impulzech vytvořených a přivedených svalovinou převodního systému (Obr 1.2).

Jádro převodního systému srdečního tvoří dva uzly situované v pravé síni a vlákna vedoucí vzruchy z nich ke komorám. Spontánním generátorem vzruchů srdce je sinoatriální uzel (SA) uložený ve stěně pravé síně v blízkosti ústí horní duté žíly. Vzruchy z SA uzlu jsou vedeny třemi mezisvazky na uzel atrioventrikulární (AV), uložený u komorového septa v blízkosti trojcípé chlopně. Uzel AV umožňuje jednosměrný převod vzruchu ke kontrakci svaloviny komor, opožděný o 0,12-0,2 vteřin oproti stahu síní.

Hisův svazek je dalším vodivým spojením k převedení vzruchu na komorové septum do levé komory. Distální část svazku se dělí na dvě Tawarova raménka. Raménko pravé vede úsekem pravé komory ke komorovému septu, přičemž raménko levé zůstává na levé straně. V místě srdečního hrotu prostupují raménka dále stěnami komor jako Purkyňova vlákna, vedoucí vzruchy vyšší rychlostí (5 m/s) [1] [2].

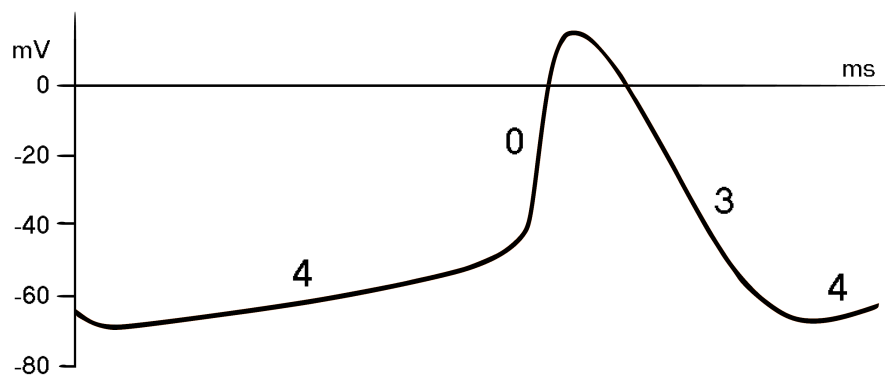


Obr. 1.2: Převodní systém srdeční

Jako všechny ostatní druhy svaloviny, potřebuje i svalovina srdeční k vykonání kontrakce akční potenciál (AP). Samovolná produkce vzruchů SA uzlu funguje díky

absenci klidového membránového potenciálu. Potenciál buněk uzlu kolísá od -60 do -40 mV a postupně se s nastupující depolarizací zvyšuje až do vzniku akčního potenciálu. Tento jev se nazývá spontánní diastolická depolarizace neboli diastolický prepotenciál (Obr. 1.3). Má přímý vliv na srdeční frekvenci a její rychlost řídí vegetativní nervový systém. Nový AP vzniká nejrychleji v SA uzlu a deceleruje s postupným průchodem jednotlivými svazky až k Purkyňovým vláknům. Proto je SA uzel přirozeným srdečním pacemakerem.

Při patologických jevech může být pacemakerem jakákoliv část myokardu a to i samotná pracovní svalovina. V případech nefunkčnosti se SA uzel nahrazuje srdečním kardiostimulátorem, tedy zařízením generujícím vzruchy umělé.

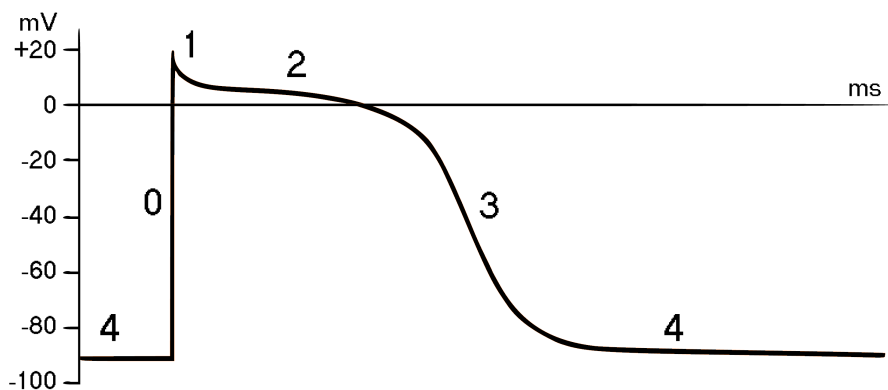


Obr. 1.3: Pacemakerový potenciál buňky sinusového uzlu. 0-depolarizace; 3-repolarizace; 4-diastolický prepotenciál

Pracovní svalovina se od svaloviny převodního systému liší ve vlastnostech buněčné membrány. Hlavní rozdíl je již v samotné hodnotě diastolického prepotenciálu okolo -90 mV, který označujeme jako klidový. Akční potenciál pracovního myokardu má delší průběh a liší se i svým tvarem (Obr. 1.4).

Na elektrické aktivitě svaloviny se podílí systém intra- a extracelulárních iontů s příslušnými kanálky na úrovni buněčné membrány, které jejich přecherpáním mění náboj uvnitř a vně buňky. Doputováním AP převodního systému k buňkám pracovního myokardu dochází k mírné depolarizaci, která aktivuje napětově řízené sodíkové kanály. Extracelulární ionty sodíku prostupují skrze kanálky dovnitř buňky a tím vyvolávají rychlou depolarizaci, která představuje výraznou změnu napětí. Aktivují se kanály draslíkové spolu s vápníkovými a vyvolají se iontové proudy. Nastává několika fázová repolarizace, která je zpočátku rychlá, pak přechází do fáze plateau, kdy dovnitř buněk vstupují ionty vápníku a udržují kladný náboj (potencují kontrakci).

Nastupuje fáze konečné repolarizace zapříčiněné prostupem draselných iontů. K navrácení iontových koncentrací pro opakování děje jsou ionty přečerpávány iontovými pumpami mezi intra- a extracelulárními prostory.



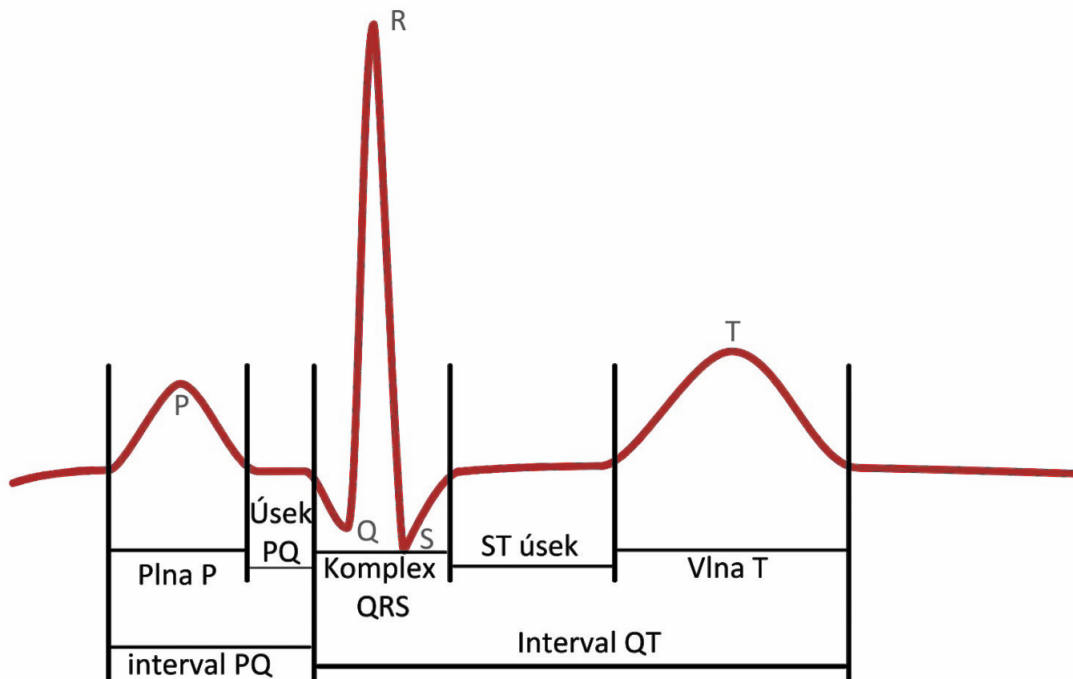
Obr. 1.4: Akční potenciál buňky pracovního myokardu. 0-depolarizace; 1-rychlá repolarizace; 2-fáze plateau; 3-konečná repolarizace; 4-klidový membránový potenciál

1.4 Měření EKG

Elektrokardiogram (EKG) je výsledný záznam z elektrokardiografu (EKG přístroj) měřícího srdeční potenciály. Výsledná křivka představuje sumu všech akčních potenciálů měřených pomocí elektrod z povrchu těla pacienta. V klinické praxi je EKG vyšetření důležitým diagnostickým nástrojem při zjišťování poruch srdečního rytmu.

Jako osvědčený standard měření EKG se v dnešní době používá 12 svodový měřicí systém, který lze rozdělit na svody hrudní, které berou srdeční potenciály v rovině horizontální a končetinové, beroucí potenciály v rovině frontální. Unipolární svody (Goldbergovy) jsou značeny jako aVR, aVL, aVF (v pořadí pravá ruka, levá ruka, levá noha). Goldbergovy svody měří potenciál proti inderferentní elektrodě neboli Wilsonově svorce, jež vzniká spojením všech tří končetinových elektrod a nese nulový potenciál. Končetinové svody bipolárního typu (Einthovenovy) zaznamenávají rozdíl potenciálů mezi elektrodami jsou značeny číslicemi I, II, III. Hrudní unipolární elektrody, nesoucí označení V1-V6, jsou postupně umístěny od 4. mezižebří pravé strany hrudní kosti (sternum) ke straně levé přes 4.-5. mezižebří a ve střední axilární čáře [1] [4].

Křivku EKG popisujeme dle jednotlivých, po sobě jdoucích zákmitů, vymezu-
cích určité úseky (Obr. 1.5). Začínáme vlnou P, která značí depolarizaci síní, následo-
vaná komplexem QRS, jež představuje depolarizaci komor. První zákmit v komplexu
je negativní, značen jako Q a stojí za depolarizací komorového septa. Následován po-
zitivním kmitem R, který zaznamenává maximální komorovou depolarizaci. Poslední
peak komplexu S má odklon od izolinie směrem dolů a zaznamenává depolarizaci
báze komor. Vše ukončuje vlna T, která reprezentuje opětovnou repolarizaci komor.



Obr. 1.5: Popis křivky EKG s jednotlivými úseky, upraveno z [2]

Jedno z dalších uplatnění EKG záznamů je analýza variability srdečního rytmu (HRV – Heart Rate Variability), jež využívá zákmitů R. Kmity slouží jako markery NN intervalů, což jsou všechny QRS komplexy v určitém záznamu, následující po P vlnách (RR intervaly). Analýza HRV nachází své uplatnění v diagnostice zvýšeného tlaku krve (hypertenze), spánkových režimů, vyšetření jedinců po infarktu, diagnostice plodu a dalších [5].

V klinické praxi se provádí analýza EKG z jednotlivých svodů na vytisknutém záznamu z elektrokardiografu nebo na počítači. Odoborník provádějící analýzu se zaměřuje na přítomnost, tvar, časový úsek, pravidelnost (sinusový rytmus), amplitudu a orientaci jednotlivých vln z každého svodu [6].

Z celkového záznamu lze dále vyhodnotit pravidelnost srdeční akce, hodnotu srdeční frekvence, která má fyziologické rozpětí 90-60 tepů/min. Frekvenci pod fyziologickou úrovní označujeme jako bradykardii (zpomalené) a nad úrovní jako tachykardii (zrychlené).

Při pohledu na jednotlivé časové intervaly se opět zaměřujeme na fyziologické hodnoty doby trvání. Pro interval PQ, představující dobu přenosu ze síní na komory, je normální hodnota 120-200 ms. Komorový komplex QRS, na který se zaměřuje téma této práce, má fyziologickou normu trvání 80-110 ms. Systolu reprezentuje interval QT, začínající kmitem Q nebo R, pokud Q není přítomno s normou trvání 340-420 ms [1] [2].

V instancích, kdy se hodnota délky trvání komplexu QRS dostane mimo zmíněný fyziologický interval jsme schopni popsat řadu onemocnění. Případy, kdy se délka QRS prodlužuje, mohou indikovat blokády ramének, elektrolytickou nerovnováhu, infarkt myokardu nebo zpoždění komorového vedení.

Blokáda Tawarova raménka se dále dělí na pravou a levou – BPRT, BLRT. V případě BLRT dochází k blokaci vzruchu s následkem opožděné depolarizace levé komory, důsledkem čehož je komora depolarizována pravým Tawarovým raménkem. Dále můžeme tuto patologii rozdělit na úplnou (kompletní) BLRT, při délce QRS nad 110ms, nebo inkompletní BLRT, která se drží v intervalu 60-110ms. Signalizací BLRT je zvýšení zátěže levé komory, vedoucí k navýšení rizika infarktu myokardu, srdečního selhání, náhlé srdeční smrti, či blokádě vedení atrioventrikulárního vzruchu.

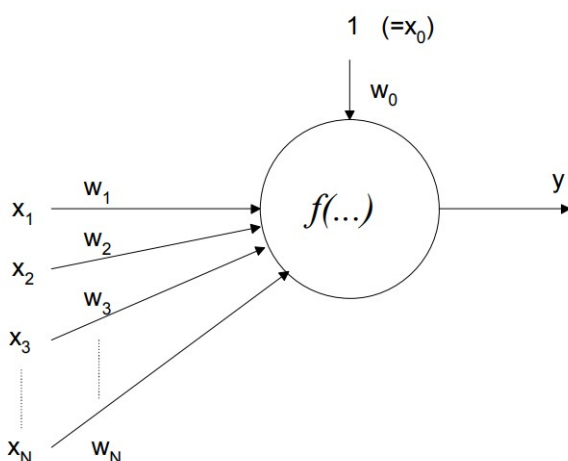
Následkem BPRT bývá opožděná depolarizace souhlasně pravé komory, mající za následek tlakové nebo objemové přetížení pravého srdce, ischemickou kardiomyopatii nebo kardiomyopatii způsobenou poškozením chlopní.

2 Umělé neuronové sítě

Počátek vývoje umělých neuronových sítí (UNS) datujeme od roku 1943, kdy se Warren McCulloch a Walter Pitts stali autory prvního matematického popisu neuronu. Myšlenka umělé sítě vychází z modelování fungování neuronů mozku živého organismu a schopnosti rozhodování. Základní stavební jednotkou UNS jsou jednotlivé umělé neurony, neboli perceptrony (Obr. 2.1). V roce 1962 se podařilo propojením základních prvků do jedné vrstvy vytvořit první síť s názvem The Perceptron, jejím autorem byl Frank Rosenblatt.

Kategoriálně se UNS zařazují mezi algoritmy strojového učení, neboli *Strojové učení* (angl. zkratka *Machine Learning*) (ML), které označuje schopnost strojů (počítačů) těžít a učit se na předložených datech za účelem následného rozpoznávání vzorů v datech nových, čímž se snaží simulovat schopnost rozhodování podobnou člověku. přiřazování do kategorií (klasifikace) či predikci vývoje číselných hodnot (regrese). V dnešní době je uplatnění UNS velmi rozsáhlé napříč všemi možnými obory, jež pracují s daty, k řešení klasifikačních nebo regresních úloh.

Základní architektura plytkých UNS se skládá ze tří propojených vrstev – vstupní, skryté a výstupní. Jednotlivé prvky (příznaky) vstupující do sítě ovlivňují počet neuronů ve vstupní vrstvě. Zde se projeví dimenzionalita používaného datasetu, jež vstupuje do sítě. Velikost výstupní vrstvy, neboli počet neuronů, ze kterých se skládá, je úměrný počtu požadovaných výstupů. Skrytá vrstva umožňuje nalézt vazby mezi vstupními a výstupními daty (učení s učitelem). Počet neuronů skryté vrstvy a samotných skrytých vrstev jsou jedny z hlavních nastavitelných hyperparametrů UNS, které ovlivňují komplexitu modelu a taktéž závisí na typu řešeného problému [7].



Obr. 2.1: Schéma samotného neuronu, převzato z [7]

Jednoduchý perceptron můžeme popsat rovnicí níže (2.1), spolu s její odvozenou variantou[7].

$$y = f \left[\sum_{i=1}^N w_i x_i - \vartheta \right] = f \left[\sum_{i=0}^N w_i x_i \right], \text{ pro } x_0 = 1, w_0 = -\vartheta \quad (2.1)$$

Představuje transformaci s N vstupy (x) na výstup (y), které jsou v rámci sumy jednotlivě násobeny příslušnými vahami (w), následně přivedeny na funkci, nazývanou aktivační funkcí neuronu. Existuje více typů lineárních, nelineárních či skokových aktivačních funkcí jako sigmoidální, hyperbolický tangens, softmax, ReLU a dalších. [?]

2.1 Aktivační funkce

Při vytváření modelů klasifikačních úloh se nejčastěji používají aktivační funkce nelineárního typu, jako softmax nebo *Reálná funkce jedné reálné proměnné* (angl. zkratka *Rectified Linear Unit*) (ReLU) (Vztah 2.3). Mezi další používané nelineární funkce můžeme zařadit funkci sigmoidální nebo hyperbolický tangens.

Funkce softmax transformuje vektor K vstupních hodnot na výstupní vektor, jehož suma se rovná 1 (Vztah 2.2) a na jednotlivé hodnoty lze nahlížet jako na pravděpodobnosti. Vstupní hodnoty mohou být kladné, záporné, či nulové, avšak výstup funkce bude vždy v intervalu od 0 do 1. Pokud má vstupní hodnota vysokou kladnou hodnotu, bude jí přiřazena vyšší pravděpodobnost na výstupu. V opačném případě u hodnot záporných nebo nulových jim připadne pravděpodobnost nízká. Vlastností softmax funkce lze využít jako normalizační vrstvy, do které vstupují prvky se rozdělením skóre nevhodným k interpretaci. Proto se softmax často používá v výstupní vrstvě UNS [8].

Základní předpisy funkcí softmax a ReLU:

$$\text{softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^K e^{x_j}} \quad (2.2)$$

$$\text{ReLU}(x) = \max(0, x) \quad (2.3)$$

2.2 Učení sítě

Dva základní přístupy k trénování UNS jsou metody učení s učitelem a bez učitele. V rámci předzpracování datasetu pro naučení UNS zpravidla rozdělíme celý používaný dataset na část trénovací, validační a testovací. Každá z částí tak dle svého názvu odpovídá příslušnému kroku tvorby sítě – trénování, validace a testování. Při zvolení metody učení s učitelem jsou síti předkládána vzorová data z trénovacího datasetu, jako například manuálně značené úseky, při klasifikaci z 1D signálových dat. Síť se pak učí na daných příkladech během transformace vstupních dat na požadovaný výstup [7] [9].

Schopnost iterativního učení sítě průchodem dat ze vstupu na výstup (dopředný směr) probíhá po epochách. Aktualizace vah sítě může probíhat po ukončení jedné epochy, kdy jsou předloženy všechny vektory trénovacích dat, nebo po iteracích, v závislosti na použitém trénovacím algoritmu.

K těmto účelům se zavádí určitá forma zpětné vazby, kdy se porovnávají výsledky dosažených výpočtů v UNS s očekávanými hodnotami. Odchytky jsou vypočteny pomocí zvolené kriteriální (ztrátová) funkce, která závisí na typu řešeného problému. Pro klasifikační úlohy s binárně kódovanými třídami ve vzorových datech se typicky používá ztrátová funkce křížové entropie:

$$CE(t, p) = - \sum_{i=1}^N [(t_i \log(p_i) + (1 - t_i) \log(1 - p_i))] \quad (2.4)$$

kde t_i jsou správné hodnoty kategorií nabývající hodnot 0 nebo 1 a p_i jsou pravděpodobnosti pro i -tou kategorii. Po výpočtu gradientu z hodnoty ztrátové funkce se upravují váhy jednotlivých neuronů skrze zpětné šíření chyby neboli *Backpropagation* [9].

Dalším blokem při trénování modelu je právě zvolení trénovacího algoritmu, který nám ovlivňuje proces aktualizace vah neuronů. Základní metoda gradientního sestupu *Gradientní sestup* (angl. zkratka *Gradient Descent*) (GD) upravuje váhy po dokončení jedné epochy trénování. V dnešní době se spíše nepoužívá, z důvodu pomalé rychlosti. Stochastický GD pracuje s vahami v rámci každé iterace, což sice zvyšuje výpočetní náročnost, ale zároveň poskytuje rychlejší hledání extrémů oproti standardnímu GD. Má lepší schopnost nalezení extrémů lokálních, ale potýká se s problémy přeskočení extrémů globálních. Mezi typem zmíněných algoritmů GD je tzv. *Mini-batch Gradient Descent*, kdy dochází k aktualizaci vah, po uplynutí zvoleného počtu iterací pohybuje se v řádech stovek až tisíc, v závislosti na dostupné paměti.

Dnes je nejčastěji používaným algoritmem *Adaptive Moment Estimation* (ADAM), který kombinuje exponenciální rozklad průměrů druhých mocnin předchozích gradientů a průměrů předchozích gradientů. Metoda ADAM tedy adaptivně upravuje krok učení v reálném čase, během trénování modelu pomocí exponenciálního zapomínání průměrů předchozích gradientů m_t a v_t kterými se následně upravují váhy modelu w_{it+1} [10].

$$m_t = \beta_1 m_{it} + (1 - \beta_1) \nabla_{w_{it}} L$$

$$v_t = \beta_2 v_{it} + (1 - \beta_2) (\nabla_{w_{it}} L)^2$$

$$w_{it+1} = w_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t$$

2.3 Hluboké učení

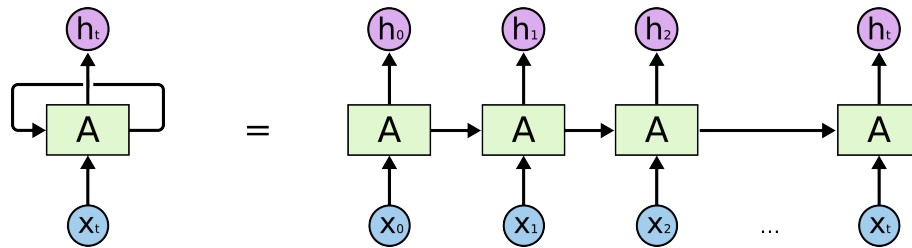
Hluboké učení (*Deep Learning*) je oblast algoritmů vyčleňujících se z ML, které se v posledních letech dostává stále větší pozornosti. Tohoto termínu se taktéž využívá ve spojení s neuronovými sítěmi, které se nazývají *Deep Neural Networks*, neboli v přímém překladu hluboké neuronové sítě. Jejich název odpovídá přímo velkému počtu vrstev, ve srovnání s klasickými (mělkými) neuronovými sítěmi. Větší komplexita jejich architektury se odráží ve vyšším počtu trénovaných parametrů, díky nimž mají schopnost se lépe namodelovat na vstupní data, detekovat skryté vazby a vzory v datech [11].

Další z rozdílů mezi mělkými a hlubokými UNS bývá v přípravě datasetu pro síť. Jedna z předností v případě sítí hlubokého učení je částečná absence extrakce příznaků z důvodů povahy architektury sítě, která v podstatě funguje jako extraktor. Pro příklad si můžeme představit vstupní 2D obrazová data procházející řadou konvolučních vrstev, jež slouží jako filtry postupně ořezávající obraz, čímž efektivně extrahují menší detaily (příznaky). Naopak nevýhodou mohou být nároky na větší objem dat potřebných ke kvalitnímu natrénování modelu, spolu s vyššími požadavky na paměť a výpočetní výkon [11].

Proces tvorby sítě hlubokého učení spočívá ve skládání základních stavebních prvků (bloků), dělí se dle prováděných operací. Jednorozměrná konvoluční vrstva slouží k rozšíření příznakového prostoru, počítáním konvoluce v podobě 1D filtru, který se pohybuje po vstupním signálu. Získáme z ní nové aktivační (příznakové) mapy, jakožto lineární kombinace příznaků vstupních s konvolučním kernelem. Mezi její nastavitelné parametry řadíme velikost kernelu, krok konvoluce a případné doplnění o nuly (padding) vstupních dat. Aktivační vrstva slouží k aplikování vybrané funkce jako ReLU nebo *softmax*. Pooling vrstva přebírá na vstup příznakovou mapu a provádí její zmenšení, dle zadaného kroku a okolí, vybírá maxima hodnot nebo

průměry. Plně propojená vrstva má nastavitelný počet neuronů, odpovídající požadovaným výstupním hodnotám. Ve vrstvě *DropOut* používané během trénování dochází v rámci optimalizace k deaktivaci neuronů konvolučních, či plně propojených vrstev. Blok *BatchNorm* provádí normalizaci vstupů mezi konvoluční nebo plně propojenou vrstvou [11].

Konvenční neuronové sítě se potýkají s problémem replikace lidského uvažování na úrovni aktivního používání již naučených parametrů, k vytváření nových vazeb umožňujících řešení komplexních problémů zejména u časových dat. Rekurentní sítě, neboli *Rekurentní neuronová síť* (angl. zkratka *Recurrent Neural Network*) (RNN) umožňují svou architekturou předejít těmto nedostatkům.



Obr. 2.2: Struktura RNN: A – část sítě, x – vstupy, h – výstupy, převzato z [12]

Sítě RNN obsahují cyklické vazby, které jsou schopny udržet již naučené informace (Obr. 2.2). Jak lze vidět na obrázku výše, natrénovaná data jsou postupně předávána napříč sítí. Princip fungování RNN si tak můžeme představit jako bychom předávali informaci mezi kopiemi totožných konvenčních sítí. Tento řetězový charakter dělá z RNN vhodné modely pro zpracování sekvencí – rozpoznávání řeči, překládání textu, modelování jazyka a dalších využití [12].

Speciálním typem rekurentní neuronové sítě (RNN) jsou sítě s dlouhou krátkodobou pamětí, neboli *Long Short-Term Memory* (LSTM), které jsou trénovány pomocí zpětného šíření chyby v čase a jsou schopny si dlouhodobě pamatovat naučené informace. Síť LSTM se hodí pro úlohy, které vyžadují modelování dlouhodobých závislostí v datech, jako je například překlad a přepis jazyka.

Skládají se z řady LSTM modulů, které jsou vzájemně propojeny a tvoří směrovaný acyklický graf. Schopnost LSTM přidávat nebo ubírat komponenty informací v rámci předávaného vektoru regulují brány modulu, tvořeny vrstvou sigmoidní aktivační funkce přecházející mezi stavy 1 a 0, která rozhodne o přijetí nebo odmítnutí příchozí komponenty. Každý LSTM modul obsahuje právě 3 takové brány k řízení svého stavu. Tato hradla umožňují buňce LSTM řídit jaké informace má uložit do své vnitřní paměti a jaké informace má zahodit.

V prvním kroku se informace dostávají skrze vyřazovací sigmoidální vrstvu na vrstvu vstupní, kde jsou funkcí hyperbolického tangens určeny hodnoty na přidání do následujícího stavu. Stavová aktualizace spočívá v zapomínání vybraných informací a přidání informací nových. To umožňuje modulu LSTM udržovat kontinuální výstup po mnoho časových kroků, i když se vstupní data rychle mění [12].

3 Využití metod hlubokého učení k rozměrování délek komplexů QRS

Jak již bylo popsáno v první kapitole této práce, elektrokardiogramy znázorňují periodicky se opakující křivky elektrické aktivity srdce, představující střídání systoly a diastoly srdeční svaloviny, mezi něž řadíme vlnu P, komplex QRS a vlnu T. Přesná analýza signálových amplitud, tvarů vln a délky trvání konkrétních úseků, umožňuje neinvazivní diagnostiku poruch srdečního rytmu a kardiovaskulárních onemocnění jako ischemie nebo infarktu myokardu. Odborní pracovníci během provádění anotací hledají opakující se prvky signálu, k rozpoznání jednotlivých regionů EKG, což v praxi zabírá značné množství času, obzvláště při analýze velkých objemů EKG dat z dlouhodobého monitoringu pacientů nebo výzkumných řešerů. Automatizace procesu segmentace a anotace by lékařům a výzkumníkům pomohla k rychlejšímu hledání podezřelých komplexů napříč větším počtem pacientů a zároveň tak ušetřila časové náklady.

Obecně se proces detekce důležitých bodů v signálech EKG a úseků mezi nimi, neboli EKG segmentace, taktéž nazývá delineaace signálu. Právě rozsáhlá morfologie signálů spolu s nestacionárním charakterem, přítomností artefaktů a šumu, představuje pro realizaci efektivní automatické delineaace hlavní překážky [15].

Dosavadní algoritmy přistupují k řešení automatizace segmentace signálů na různých úrovních. Jako elementární můžeme označit detektory QRS komplexů vycházející ze základních metod zpracování signálu, opírající se o amplitudu R peaku. K detekci hranic a vrcholů vln lze využít i jiných propracovanějších metod signálového zpracování, založených na vlnkové transformaci nebo skrytých Markovových modelech, které však obecně dosahují nízkých či průměrných výsledků a nejsou odolné vůči variacím v EKG datech, neboli zaostávají ve schopnosti generalizace.

Vzhledem k pokroku vývoje výpočetní techniky a příchodu éry algoritmů strojového učení, které v posledních letech nacházejí stále většího uplatnění a to i v oblasti segmentace signálů EKG. Tedy i ke zjištění délky trvání komplexu QRS a urychlení procesu diagnostiky. V následujících kapitolách budou uvedeny základní druhy architektur modelů hlubokého učení, které byly využity jinými autory k segmentaci EKG signálů.

3.1 Segmentace vln EKG pomocí sítě s architekturou U-Net

Autoři Moskalenko V., Zolotykh N., a Osipov G. z Lobačevské státní univerzity v Nižním Novgorodě navrhli metodu segmentace s využitím plně konvoluční sítě vycházející z architektury U-Net, za dosažení prvotřídních výsledků detekce počátku a konce komplexu QRS a vln P, T.

K předzpracování vstupních dat vytvořili autoři funkci k převzorkování signálů při použití jiných než jimi zvolených dat z databáze LUDB. Vzorkovací frekvence vstupního signálu je přepočítána použitím kubického splinu.

Vstup sítě je tedy vektor hodnot signálu o délce l z jednoho ze 12 svodů, každý svod vstupuje zvlášť. Výstupem je pak matice s rozměry $(4, l)$ s finálními skóre pro každou z kategorií P, QRS, T, nebo nedetekované. Architektura autory navrhované sítě se skládá ze čtyř bloků, z nichž každý obsahuje dvě konvoluční vrstvy s batch normalizací a ReLU aktivační funkcí. Bloky jsou sekvenčně propojeny MaxPooling vrstvy. Výstupy předchozích vrstev jsou skrze MaxPooling vrstvy navedeny na další bloky konvolučních vrstev. Výstupy z těchto vrstev projdou dekonvolucí a padding vrstvy na další konvoluční vrstvy v opačném pořadí. Všechny konvoluce prováděné v jednotlivých vrstvách mají parametry velikosti kernelu rovnou 9 a padding roven 4.

Hlavní rozdíly mezi sítí navrženou autory oproti původní U-Net síti sloužící ke zpracování 2D obrazových dat, je použití 1D konvolucí místo 2D, odlišné nastavení parametrů konvolučních vrstev a použití padding vrstev [13].

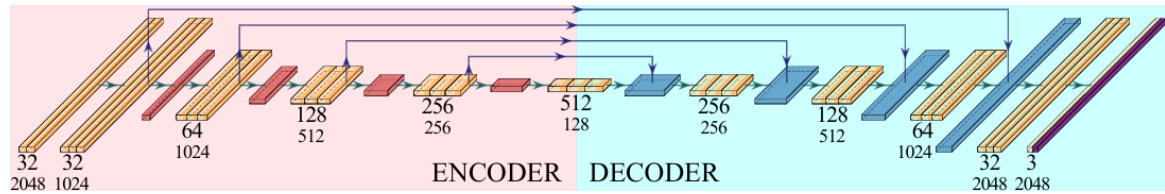
Dle výsledných hodnot statistik, publikovaných autory, získaných při testování navržené sítě na zbytku dat z LUDB vyhrazených k těmto účelům, se autorům podařilo dosáhnout téměř dokonalých výsledků (Tab. 3.1).

Kategorie	Senzitivita	PPV	Chyba	STD
QRS počátek	99,96	99,84	2,1	9,8
QRS konec	99,96	99,84	1,6	9,8

Tab. 3.1: Výsledky při použití všech svodů [13]

Autoři Guillermo Jimenez-Perez et al. se ve své práci rozhodli použít model také vycházející z architektury U-Net k jehož vývoji zvolili databázi QT (Kap. 4). Výsledná architektura U-Net se dle autorů skládá z 5 bloků enkodéru a dekodéru (Obr. 3.1). Poslední vrstva má 3 výstupní kanály pro segmentaci P, QRS, T a sigmoidální aktivační funkci. Pro potřeby této práce se zaměříme pouze na segmentaacci QRS Model byl implementován skrze knihovnu Keras s optimalizačním algoritmem

ADAM, kterému byly ponechány základní parametry. Jako jeden vstupní prvek autoři počítali jeden ze 105 záznamů QTDB, který byl následně rozdělen do oken s překryvem 2^{10} a 2^{11} vzorků [14].



Obr. 3.1: Architektura sítě, převzato z [14]

Výstup sítě jsou predikce jednotlivých segmentů v podobě binární masky, na které autoři aplikovali operaci známou u zpracování obrazů jako binární *closing*, v rámci post-processingu výsledků. Metriky pro vyhodnocení výkonu modelu zahrnují senzitivitu a *Pozitivní prediktivní hodnota* (angl. zkratka *Positive Predictive Value*) (PPV) k hodnocení kvality segmentace QRSa průměrnou chybu společně s směrodatnou odchylkou k hodnocení úspěšnosti detekce QRS počátku a konce [14](Tab. 3.2).

Kategorie	Senzitivita	PPV
QRS komplex	99,47	98,18
	Chyba	STD
QRS počátek	-3,2	22,4
QRS konec	7,9	21,9

Tab. 3.2: Nejlepší dosažené výsledky autorů modelu z [14]

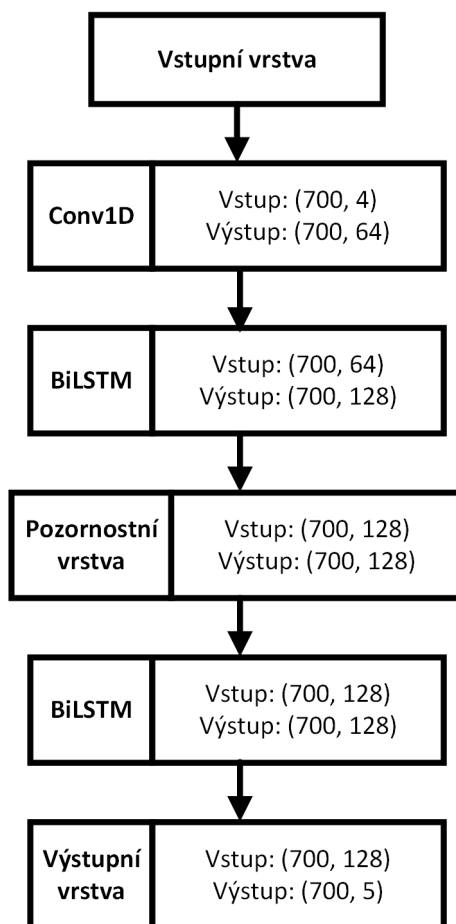
3.2 Segmentace vln EKG s využitím konvoluční LSTM sítě

Návrh autorů Malali A., Srinidhi H., Siddesh G.M., Srinivasa K.G., Sanjay N.T., uveřejněný v příspěvku *Supervised ECG wave segmentation using convolutional LSTM*, z roku 2020 představuje klasifikátor signálu EKG využívající síť hlubokého učení ConvLSTM, neboli kombinaci konvoluční a *Long Short-Term Memory* sítě. Síť ConvLSTM využívá k segmentaci jak časové tak i prostorové složky signálu [15].

Pro účely vývoje algoritmu použili autoři již zmíněnou veřejnou databázi QT v plném rozsahu 100 záznamů dvoukanálového EKG v 15-minutových úsecích. Každý záznam byl rozdělen na úseky o délce 700 vzorků s překryvem 50 vzorků mezi předchozím a následujícím segmentem. Oba kanály každého záznamu byly použity jako součást vstupních příznaků. Mezi další příznaky autoři zařadili signál vyhlazený interpolací kubickým splinem k odstranění přítomného šumu a první derivaci nezpracovaného signálu. Autoři původně plánovali i s použitím druhé signálové derivace, avšak později došli k závěru, že nijak nepřispívala k lepším výsledkům [15].

Použitá architektura rekurentní neuronové sítě je vyhovující pro segmentaci časové řadě (Obr. 3.2). Kombinace konvoluční vrstvy a LSTM modulů poskytuje dlouhodobé ukládání časových závislostí a eliminuje problém ztracení nebo přerůstání gradientu. Obousměrné LSTM (BiLSTM) používá dopřednou komponentu k výpočtu hodnot následujících skrytých stavů ze stavů předchozích. Druhá, zpětná komponenta, počítá hodnoty skrytých stavů obrácením chronologické posloupnosti vstupu [15].

Data ve formě vstupní matice, o rozměrech 700×4 , jsou na vstupu modelu přivedena na 1D konvoluční vrstvu operující v časové oblasti, která zároveň slouží jako extraktor příznaků. Autoři zmiňují, že použití konvoluce zároveň zjednoduší vícekanálová vstupní data, která jsou následně vedena do obousměrné LSTM vrstvy. Obousměrná vrstva LSTM obsahuje 64 buněk pro každý směr. Následuje self-attention vrstva soustřeďující se na části signálu při stanovení značky konkrétní kategorie. Opětovné použití LSTM přivádí data na výstupní hustou vrstvu, která v časové oblasti přiřazuje každé datové hodnotě konkrétní kategorii [15].



Obr. 3.2: Schéma architektury sítě ConvLSTM, přeloženo z [15]

Trénovací dataset – vstup při učení obsahoval 27 tisíc vzorků segmentů EKG o rozměrech 700×4 , shodných s maticí cílů. Trénování modelu probíhalo v 50 epochách s použitím batch normalizace k předejití přehlcení. Jako kritériální funkci autoři zvolili křížovou entropii a pro vyhodnocení úspěšnosti modelu použili F1 skóre, přesnost, specificitu a pozitivní prediktivní hodnotu (Tab. 3.3).

Kategorie	Přesnost	F1 skóre	PPV	Senzitivita
Vlna P	94,87	94,24	95,20	93,30
QRS komplex	96,66	96,70	95,83	95,57
Vlna T	92,73	93,04	91,56	94,56

Tab. 3.3: Dosažené míry úspěšnosti klasifikace modelem ConvLSTM [15]

4 Veřejně přístupné databáze

Základem vývoje jakéhokoliv ML modelu je potřeba adekvátního množství dat k jeho trénování a následnému testování. Díky pokrokům v oblasti informačních technologií a internetu jsme dnes schopni data získávat i skrze vzdálený přístup a nemusíme tak provádět vlastní, většinou časově náročné, měření. Pro tyto účely slouží taktéž veřejně přístupné databáze, poskytující různé druhy dat k podobným účelům, jako je náplň této práce.

4.1 PhysioNet

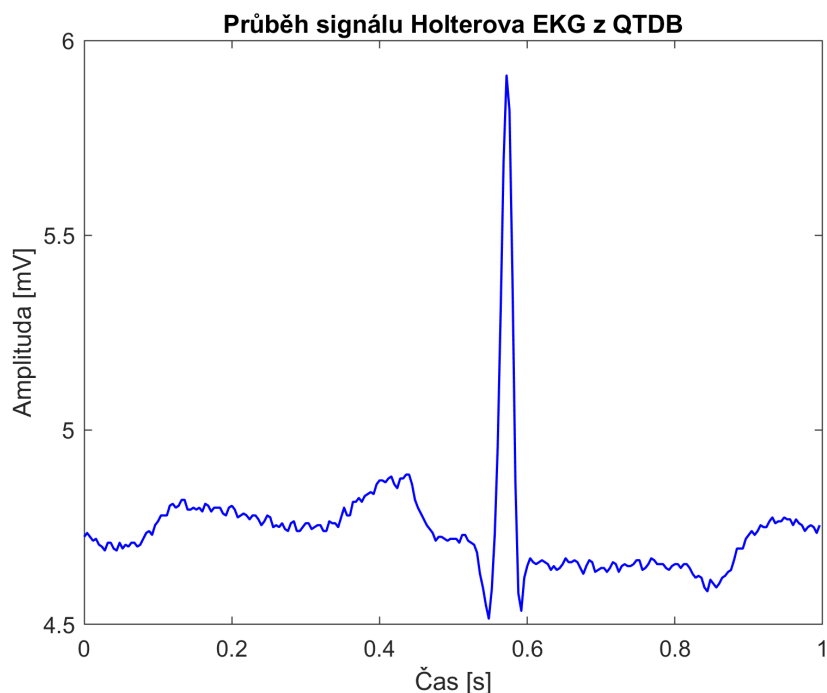
PhysioNet je on-line fórum poskytující služby uložení biomedicínských dat a softwarových nástrojů pro výzkum a vzdělávání ve zdravotnických vědách. Bylo založeno zároveň s archivem PhysioBank, obsahujícím široké spektrum digitálních záznamů fyziologických dat a knihovnu PhysioToolKit, která obsahuje open-source software k analýze a detekci fyziologicky důležitých jevů, jejich interaktivnímu zobrazení, simulaci signálů, kvantitativní evaluaci a srovnání analytických metod. Za vznikem stojí dlouholetá spolupráce mezi vědci a praktiky z široké škály přírodních a technických oborů, taktéž zahrnující řadu institucí jako Beth Israel Deaconess Medical Center, Harvard University, Massachusetts Institute of Technology (MIT) a další. [16]

Fórum PhysioNet tedy umožňuje volný přístup k rozsáhlému souboru, komentovaných, různorodých a dobře zdokumentovaných záznamů fyziologických signálů. Sbírkou dat zahrnují elektrokardiografii (EKG), elektromyografii (EMG), elektroencefalografii (EEG) a další biosignály spolu s klinickými a demografickými údaji, pro použití k edukativním nebo vývojovým potřebám. [16]

4.2 Databáze QT

Databáze QT vznikla v roce 1997 za účely evaluace algoritmů, k detekování hranic křivek signálů EKG. Současná verze databáze je od roku 1999 součástí PhysioNet a dodržuje standard, obdobný pro všechny ostatní EKG databáze. Celkem obsahuje 105 15-minutových záznamů dvoukanálového Holterova EKG, které obsahují širokou škálu QRS a ST-T segmentů (Obr. 4.1).

Většina naměřených záznamů obsažených v databázi byla vybrána na základě porovnání s již existujícími databázemi tak, aby ideálně reprezentovala různé morfologie srdeční aktivity a omezil se tak počet nechtěných artefaktů či kolísání nulové izolinie. Všechny signály v databázi jsou pořízeny se vzorkovací frekvencí 250 Hz. Hranice křivek byly v jednotlivých záznamech značeny experty skrze pracovní stanici SUN s interaktivním grafickým zobrazením WAVE, kde byly vždy vykresleny oba kanály zároveň a na nich následně značeny anotace. V každém záznamu bylo manuálně anotováno okolo 30 až 100 srdečních úderů. Anotace zahrnují začátek, peak a konec vlny P, komplexu QRS a vlny T. Pro 11 záznamů jsou vytvořeny dvě sady nezávislých anotací k umožnění pozorování variability značení mezi pozorovateli. [16] [17]

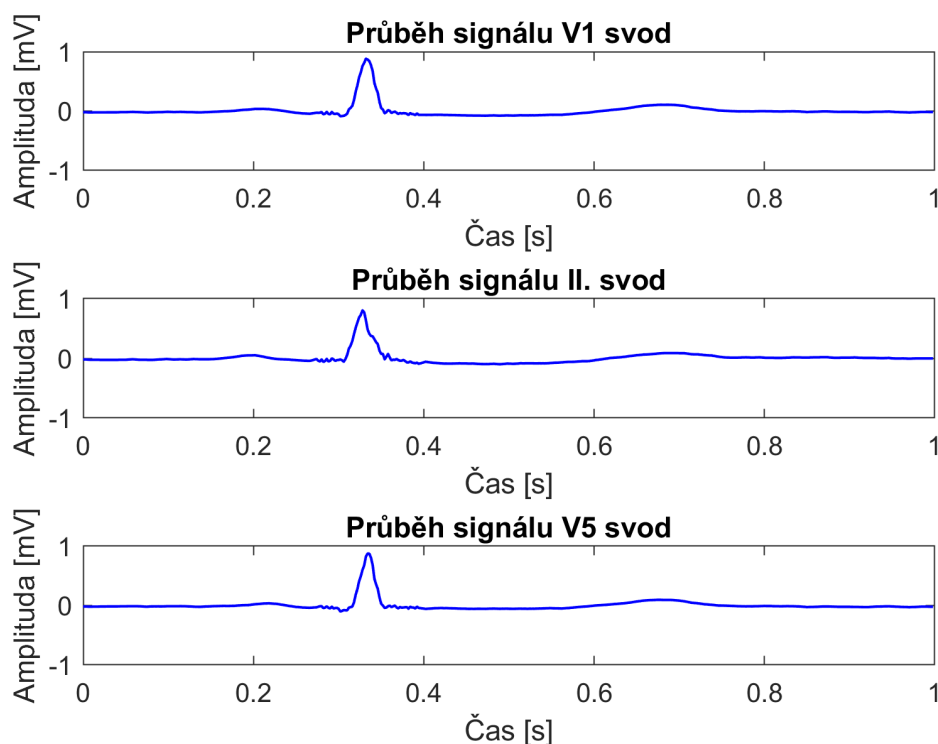


Obr. 4.1: Ukázka QRS komplexu z QTDB

4.3 Databáze LUDB

V rámci projektu Cyberheart na institutu informačních technologií, matematiky a mechaniky Lobačevské státní univerzity v Nižním Novgorodě, vznikla Lobachevsky University Database, zkráceně LUDB. Databáze obsahuje 200 záznamů 12-svodového EKG měřeného přístrojem *Schiller Cardiovit AT-101* na 200 subjektech. Signály byly měřeny jak na zdravých subjektech z řad dobrovolníků, tak na pacientech trpících různými kardiovaskulárními chorobami. Někteří z pacientů měli v době měření voperovaný kardiostimulátor. Věkový rozsah subjektů se pohybuje od 11 do 90 let, s průměrným věkem 52 let. Zastoupení pohlaví se skládá z 85 žen a 115 mužů.

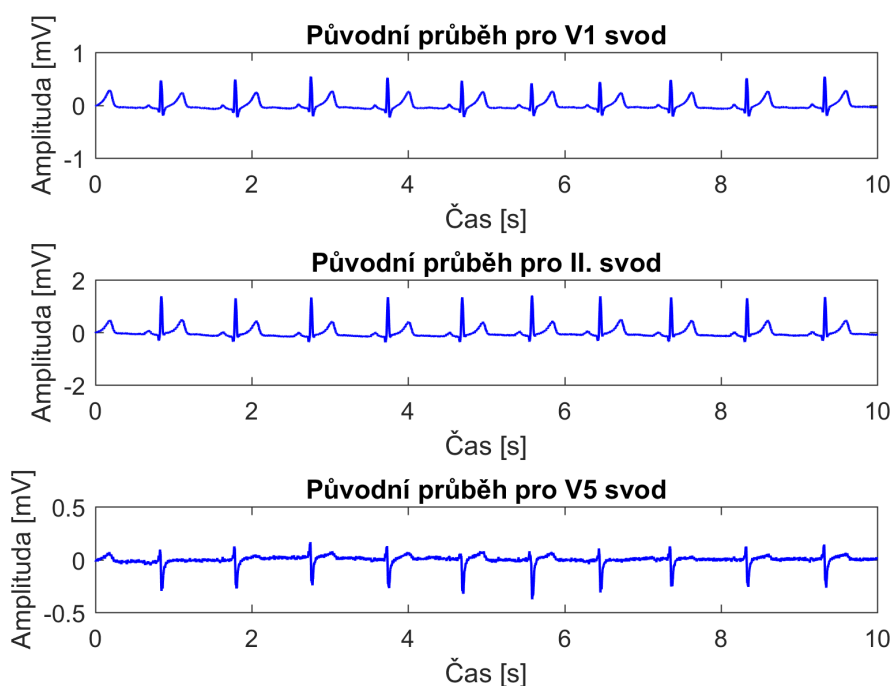
Délka jednoho záznamu činí 5000 vzorků, což při vzorkovací frekvenci 500 Hz odpovídá 10 vteřinám (ukázka 1 vteřiny na Obr. 4.2). Manuální anotace komplexů v signálech EKG a jejich následná diagnostika byla provedena pracovníky lékařských institucí v Nižním Novgorodě. Každý ze záznamů má kardiology anotovány meze a peak komplexu QRS, vlny P a T. Celkově je ze všech svodů zvláště anotováno 21966 komplexů QRS [18] [19].



Obr. 4.2: Ukázka dat z LUDB

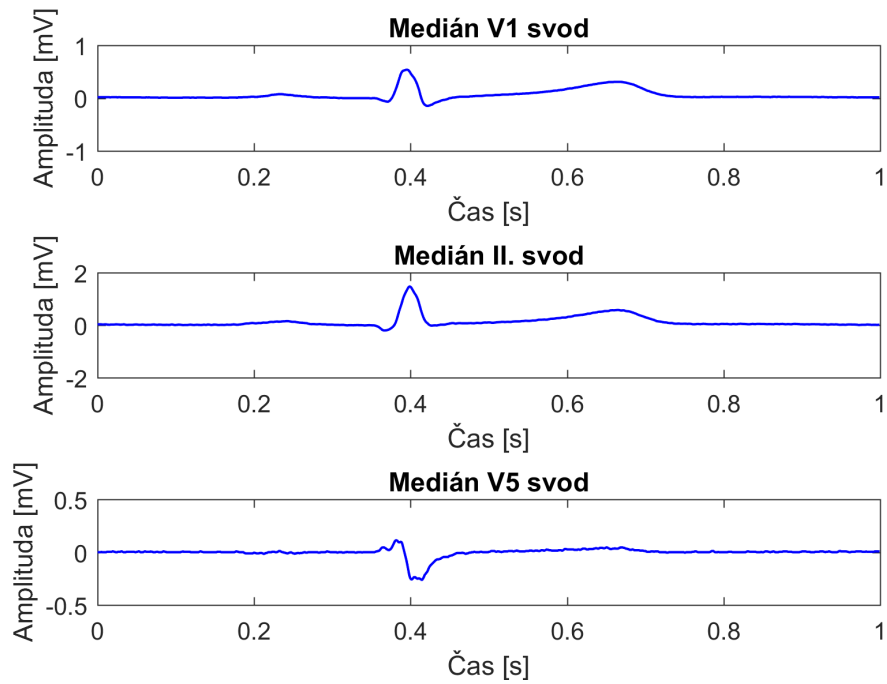
4.4 Databáze ECGCIPA

V rámci klinické studie CiPA ECG, zabývající se výzkumem účinků ranolazinu, verapamilu, lopinaviru, dofetilidu, diltiazemu a dalších medikamentů, byla vytvořena stejnojmenná databáze ECGCIPA. Její obsah tvoří vícekanálové EKG záznamy pořízené systémem *Mortara Surveyor system*, měřením 60 subjektů, jež se účastnili dané studie. Signály byly pořízeny se vzorkovací frekvencí 500 Hz s amplitudovým rozlišením $2,5 \mu\text{V}$. Měření každého ze subjektů bylo rozděleno do tří částí dle podávaných léků, s časovým intervalem od 1 hodiny před podáním, až 73 hodin po podání léků, přičemž z každého intervalu bylo pořízeno 28 záznamů (v daných časových okamžicích) o délce 10 vteřin (Obr. 4.3).



Obr. 4.3: Ukázka dat z ECGCIPA 1/2

Teoreticky předpokládaný rozsah dle 28 časových momentů pro 3 části u 60 subjektů by byl 5040 záznamů, avšak skutečný rozsah činí 5749 souborů, z důvodu opakovaných měření u některých ze subjektů. Každý záznam byl v rámci post-processingu převzorkován na 1000 Hz a zároveň byl pro každý ze 12 svodů v daném záznamu vypočítán mediánový průběh (Obr. 4.4). Na základě těchto mediánových záznamů byly vytvořeny semi-automatické anotace pomocí aplikace *CalECG*, následně upraveny dvěma nezávislými odborníky [20].



Obr. 4.4: Ukázka dat z ECGCIPA 2/2

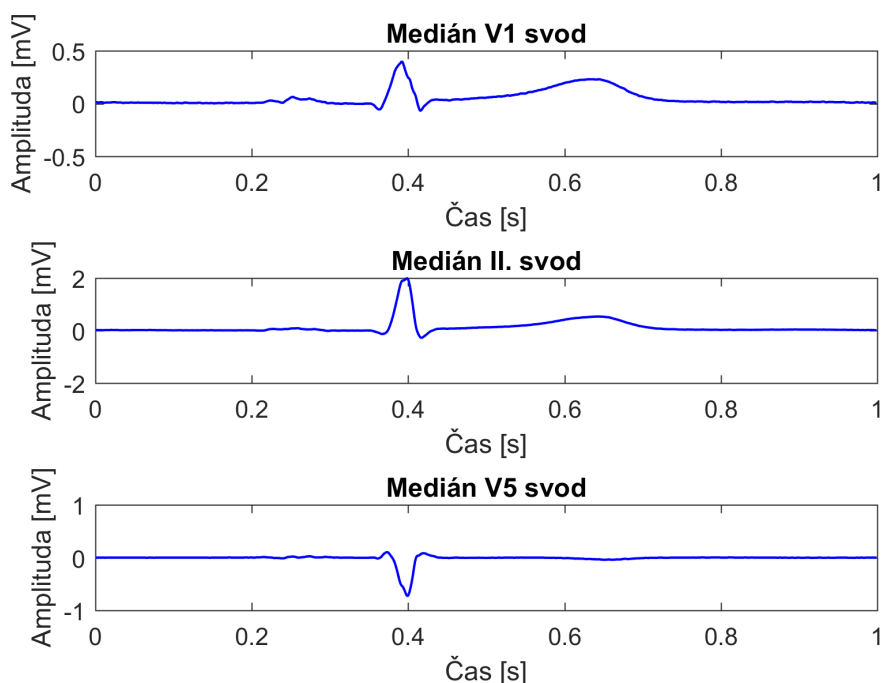
4.5 Databáze ECGDMLD

Databáze ECGDMLD obsahuje data z dvojité zaslepené klinické studie prováděné na 22 zdravých mužských a ženských subjektech ve věku 18 až 35 let. Cílem bylo porovnat elektrofyziologickou odpověď na podání léků blokujících hERG draslíkový kanál. Studie probíhala v 5 léčebných obdobích. V rámci každého léčebného období bylo odebráno 12 krevních vzorků pro farmakokinetické techniky spolu s odpovídajícími 12-kanálovými EKG záznamy.

Kontinuální EKG bylo snímáno s frekvencí 500 Hz a rozlišením amplitudy 2,5 μV pomocí systému *Mortara Surveyor*. Z kontinuálního záznamu byly extrahovány trojice 10sekundových EKG před odběrem každého farmakokinetického vzorku na 14 předdefinovaných časových bodech: 1 bod před dávkou (-0,5 h) a 13 bodů po dávce (1,5, 2, 2,5, 3, 6,5, 7, 7,5, 8, 12, 12,5, 13, 13,5 a 24 h), během kterých byl každý ze subjektů vleže po dobu 10 minut.

Následně byly z 14 časových bodů extrahovány a zároveň převzorkovány na 1000 Hz, 3 optimální 10sekundové záznamy se stabilní srdeční frekvencí a maximální kvalitou signálu pomocí softwaru *Antares*.

Celkový rozsah databáze činí 4211 signálů, uložených v surové podobě a v upravené verzi s vypočtenými mediány (Obr. 4.5). Pro účely této práce byly vybrány právě mediánové záznamy, kvůli z nich odvozených anotací, obsahujících pro nás důležitou, pozici počátku a konce komplexu QRS [21] [22].

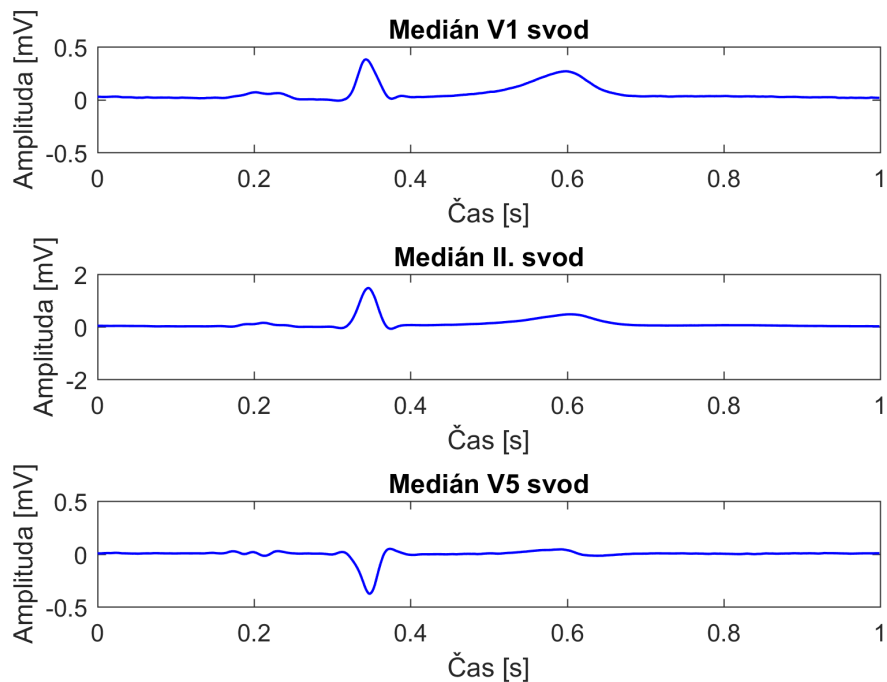


Obr. 4.5: Ukázka dat z ECGDMLD

4.6 Databáze ECGRDVQ

V rámci měření databáze ECGRDVQ byla každému subjektu podána řada látek zahrnujících dofetilid, chinidin síran, ranolazin, verapamil hydrochlorid a dávku placebo. Interval dávkování probíhal v pěti obdobích, kdy byla každá z dávek podána ráno, čímž započal 24 hodinový cyklus. Mezi každým obdobím pak trval 7denní interval k odbourání podaných látek. Na měření dat se podílelo 22 subjektů, kterým bylo měřeno 12svodové EKG v 16 časových bodech od podání medikamentů, podobně jako v databázi ECGDMLD. Signály byly snímány v 10 vteřinových intervalech se vzorkovací frekvencí 500 Hz a amplitudovým rozlišením 2,5 μV .

Celkový rozsah databáze činí 5232 nahrávek v základní nezpracované podobě. Obdobně jako u databází ECGCIPA nebo ECGDMLD byly záznamy převzorkovány na 1000 Hz a také odvozeny mediánové průběhy, pro které byly vytvořeny anotace [23](Obr. 4.6).



Obr. 4.6: Ukázka dat z ECGRDVQ

5 Implementace vlastní metody s použitím hlubokého učení

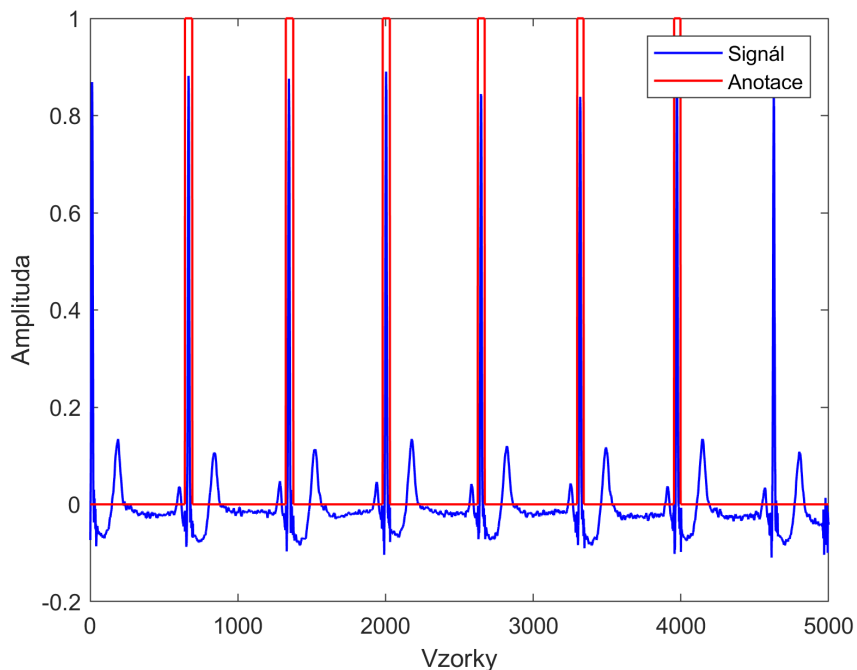
Pro návrh a implementaci vlastního postupu detekce začátku a konce QRS s použitím hlubokého učení bylo nejprve využito programového prostředí MATLAB spolu s balíčkem WFDB od PhysioNet pro načítání a předzpracování dat z databází. Při práci s balíčkem WFDB pro MATLAB bylo nutné nainstalovat starší verzi 2017 z důvodu nekompatibility u novějších verzí. Následná náplň práce, co se modelové části týče, byla provedena v prostředí Python s využitím knihoven TensorFlow a Keras ke konstrukci, trénování, testování a následnému zpracování výsledků modelu do požadovaných výstupů. Současně byla využita služba Weights & Biases (WANDB), která vznikla v roce 2018 a sloužila nám k monitorování průběhu trénování sady modelů v rámci optimalizace hyperparametrů [24].

5.1 Načítání dat

Pro účely trénování a testování modelů hlubokého učení v rámci této práce bylo vybráno 5 databází, mezi které patří již zmíněné QTDB, LUDB, ECGCIPA, ECGDMLD a ECGRDVQ. Data z posledních tří databází jsou podobného charakteru a disponují shodnými parametry co se týče vzorkovací frekvence 1000 Hz a amplitudového rozlišení 2,5 μV . Tyto datasety obsahují mediánové průběhy spolu s anotacemi, jak již bylo zmíněno v předchozím popisu jednotlivých databází.

Pro práci s daty v prostředí MATLAB bylo nejprve potřeba je načíst a převést z formátu *.dat*, k čemuž bylo využito struktury buňkového pole, která vhodně umožňuje ukládat data s libovolnou délkou. Signálové záznamy byly uloženy do jednotlivých řádků prvního sloupce pole. Následně byla ve druhém sloupci přidána příslušná anotace obsahující pouze úseky QRS značené kategoriální proměnnou 1, pokud je přítomen QRS komplex, nebo 0 pokud přítomen není. Úseky byly extrahovány z přiložených anotací, které byly v původním formátu značeny textovými znaky, doplněnými o indexy jejich pozic v signálu.

U databáze QTDB, jenž v původním rozsahu obsahuje 105 záznamů, bylo 8 záznamů vyřazeno z důvodu chybějících nebo nevyhovujících anotací. Tím se počet nahrávek použitých k následné extrakci oken snížil na 97. V dalším kroku byly z QTDB extrahovány pouze takové úseky jejichž značení provedli autoři zcela manuálně pouze na jednom ze dvou svodů, nikoliv s použitím dostupných softwarových nástrojů poskytujících semiautomatické anotace, což dále zredukovalo obsah použitých dat v časové oblasti. Obdobná redukce nastala i v případě dat z LUDB, kde autoři záměrně vynechali anotace pro QRS na počátku a konci signálu (Obr. 5.1).



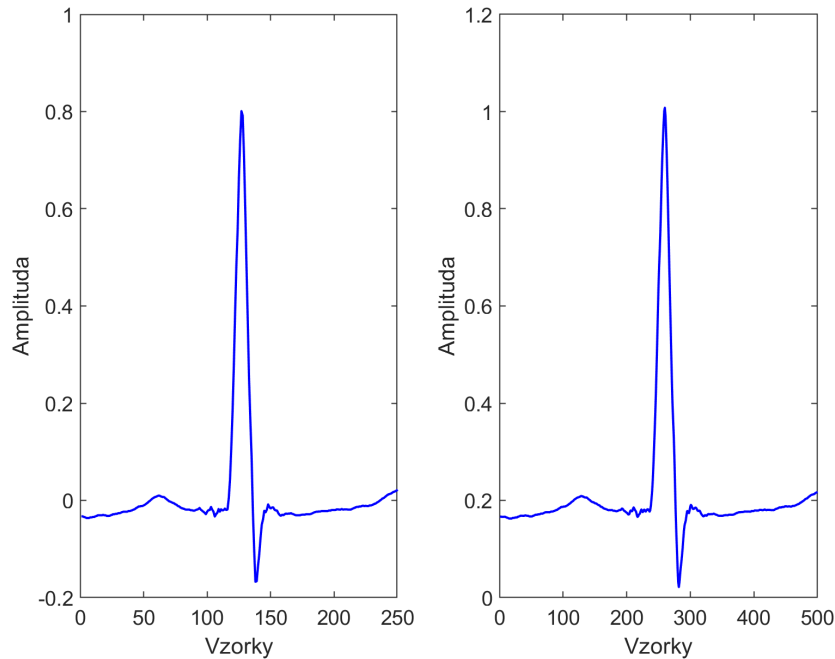
Obr. 5.1: Ukázka vynechání anotace z datasetu LUDB

5.2 Předzpracování dat

Výsledek prvního kroku načítání dat tvoří buňková pole pro každou databázi, kde jeden řádek odpovídá jednomu svodu jednoho záznamu. Ve druhé části přípravy a načítání dat byl proveden základní preprocessing k zajištění uniformity obsahu jednotlivých záznamů z důvodu lišících se rozsahů amplitud signálů mezi databázemi. V prvním kroku byly amplitudy záznamů normalizovány do rozsahu 0 až 1. Dalším krokem bylo zvolení společné vzorkovací frekvence 1000 Hz, která byla vybrána dle nejvyšší F_s z použitých databází k převzorkování signálů v databázích QTDB a LUDB, kde byla výchozí F_s 250 a 500 Hz (Obr. 5.2).

Databáze	F_s [Hz]	Záznamy	Svody	Délka [vzorky]	Počet oken
QTDB	250	97	1	225 000	3176
LUDB	500	200	12	5000	21 843
ECGCIPA	1000	5749	12	1200	68 988
ECGDMMLD	1000	4211	12	1200	50 412
ECGRDVQ	1000	5223	12	968 - 1876	62 652

Tab. 5.1: Parametry použitých dat: F_s – vzorkovací frekvence

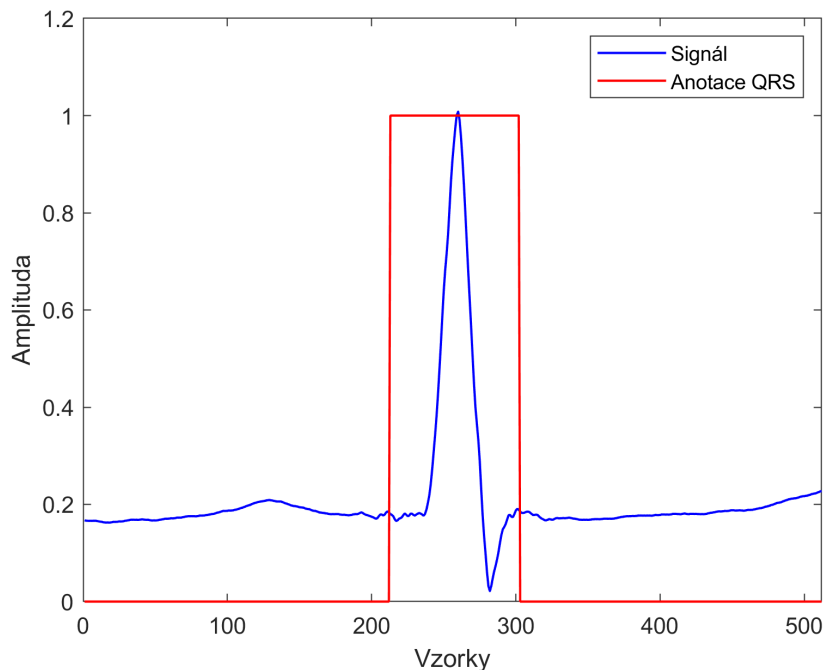


Obr. 5.2: Ukázka předzpracování na vzorku dat z LUIDB, okno 500 ms

Každý ze záznamů byl následně transformován pomocí plovoucího okna se středem kolem středu komplexu QRS do úseků o rozměrech 1×512 vzorků, tak aby každé okno obsahovalo právě jeden komplex QRS. Plovoucí okno bylo navíc doplněno o podmínku přidání pouze takových úseků, které obsahovaly úplné komplexy QRS, abychom se vyhnuli částem signálu na počátku a konci záznamu.

Anotace byly pro potřeby sítě taktéž upraveny formou *one-hot*, kdy došlo na každé z nich k rozdělení na anotaci značící nepřítomnost a přítomnost QRS. Každý takto transformovaný záznam byl následně přidán do matice dat o 3 dimenzích, z nichž první obsahuje signálová data, druhá anotace nepřítomnosti QRS a třetí anotace přítomnosti QRS. Rozměr matice pro každou databázi, odpovídající počtu oken, lze vidět v tabulce 5.1 (počet oken). Ukázka podoby jednoho výsledného zpracování záznamu s příslušnou anotací přítomnosti QRS lze vidět na Obr. 5.3.

V posledním kroku přípravy dat pro síť došlo k výběru trénovací a testovací sady z každé databáze. Do testovacích sad bylo náhodným výběrem ze 3 největších datasetů (ECGCIPA, ECGDMLD, ECGRDVQ) vybráno po 1000 záznamech, z LUIDB 500 záznamů a z QTDB 200 záznamů (Tab. 5.2). Tím bylo dosaženo počtu 5 testovacích setů pro splnění požadavku zadání k testování modelu na datech z 5 databází. Zbývající data určená k učení modelu byla sestavena do konečné matice, ze které se oddělila validační část pro optimalizaci hyperparametrů.



Obr. 5.3: Ukázka okna z datasetu LUDB s anotací QRS

Databáze	QTDB	LUDB	CIPA	DMMLD	RDVQ	Trénovací set
Testování	200	500	1000	1000	1000	
Trénování	2976	21 343	67 988	49 412	61 652	203 371

Tab. 5.2: Počty záznamů určených pro testování a trénování

5.3 Architektura modelu

Následující sekce praktické části práce se přesunují do prostředí Python. Zvoleným modelem hlubokého učení je neuronová síť typu U-Net, která dle autorů článků [13] a [14] nabízí velice slibné výsledky. K učení modelu byla připravena finální matice trénovacích dat popsána v předchozí kapitole obsahující nižší stovky tisíc záznamů (Tab. 5.2).

Architektura použité sítě tedy vychází přímo z architektury sítě použité autory Moskalenko V, et al.[13]. Na implementaci modelu byly využity knihovny TensorFlow a Keras, ze kterých byly čerpány základní bloky pro sestavení sítě. K tomu bylo taktéž potřeba použít tři speciální složené bloky. První z nich poskytuje na svém výstupu aktivační mapu a skládá se z bloků padding, 1D konvoluce, batch normalizace a aktivace (Výpis 5.1)[25].

Výpis 5.1: Ukázka části kódu složené konvoluční vrstvy

```

1 from tensorflow.keras import layers
2
3 class ConvolutionLayer(layers.Layer):
4     # Parametry: pocet filtru, aktivacni fce, padding
5     # Vychazi parametry: aktiv fce = ReLU a padding = 4
6     def __init__(self, num_filters, activation='relu',
7                 padding=4, **kwargs):
8         super(ConvolutionLayer, self).__init__(**kwargs)
9
10    # Jednotlive vrstvy
11        self.num_filters = num_filters
12        self.activation = activation
13        self.padding = padding
14
15        self.pad_layer =
16        layers.ZeroPadding1D(padding=self.padding)
17        self.conv_layer = layers.Conv1D(filters=
18        self.num_filters, kernel_size=9)
19        self.batch_norm = layers.BatchNormalization()
20        self.activation_func = layers.Activation(self.
21            activation)
22    # Pipeline vstupu pres bloky
23    def call(self, inputs):
24        padded = self.pad_layer(inputs)
25        conv_output = self.conv_layer(padded) # kernel=9
26        norm_output = self.batch_norm(conv_output)
27        activated_output = self.activation_func(norm_output)
28        return activated_output # Aktivacni mapa

```

Následující blok provádějící podvzorkování přijme výstup z bloku složené konvoluce ve formě aktivační mapy. Redukce dimenzionality vstupních dat má účel extrahovat hlubší abstraktní příznaky pro nalezení skrytých vazeb v datech. Konstrukce je podobná předchozímu bloku a znázorněna pomocí části kódu níže (Výpis 5.2). Vstupní aktivační mapa nejprve prochází blokem pooling s hodnotou 2 čímž je v časové oblasti zredukována o polovinu a vstupuje do série dvou konvolučních bloků popsaných v předchozím odstavci. Výstupem je opět aktivační mapa, avšak podvzorkovaná a obohacena o další příznaky z konvolučních filtrů [25].

Výpis 5.2: Ukázka části kódu vrstvy podvzorkování

```

1 # Pipeline vstupu pres bloky
2 def call(self, inputs):
3     pooled = self.pool(inputs)
4     conv1_output = self.conv1(pooled)
5     conv2_output = self.conv2(conv1_output)
6     return conv2_output # Aktivacni mapa

```

Poslední složený blok provádí sestavení segmentovaného výstupu zpětným vzorkováním, kterým zvyšuje rozlišení příznakových map v časové oblasti pomocí transponované konvoluce s pevně nastavenou velikostí kernelu 8 a krokem 2. Výstup transponované konvoluce je předán do bloku konkatenace, který jej spojí s výstupními příznaky bloku podvzorkování, což ve výsledku slouží ke zvýšení efektivity sítě a rekonstrukci detailů. Následně jsou takto propojená data opět předána sérii dvou konvolučních bloků (Výpis 5.1) a výstupem je tedy aktivační mapa (Výpis 5.3) [25].

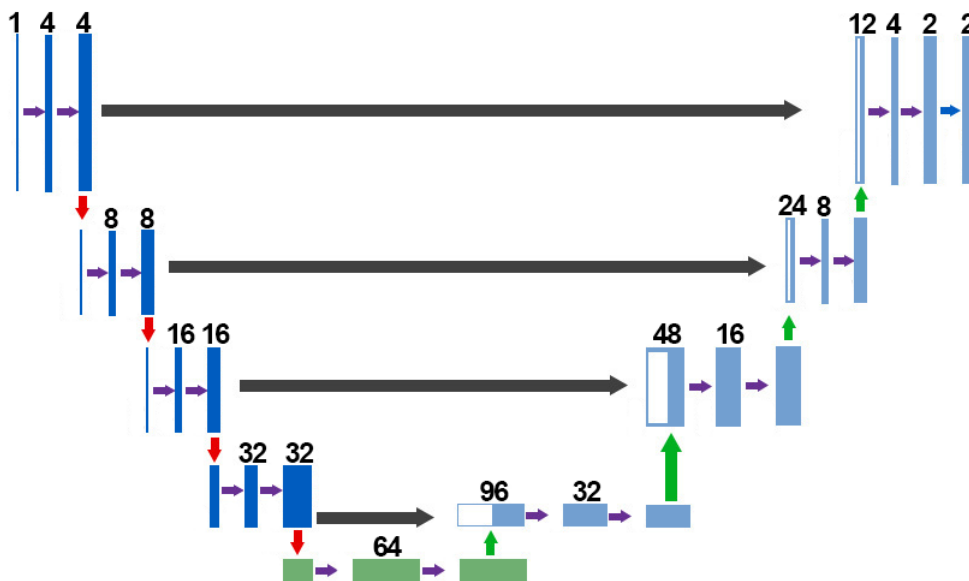
Výpis 5.3: Ukázka části kódu vrstvy zpětného vzorkování

```

1 # Pipeline vstupu pres bloky
2 def call(self, higher_features, lower_features):
3     upsampled = self.transposed_conv(lower_features)
4     concatenated = self.concat([higher_features,
5                                 upsampled])
6     conv1_output = self.conv1(concatenated)
7     conv2_output = self.conv2(conv1_output)
8     return conv2_output # Aktivacni mapa

```

Signálová vstupní data postupně procházejí sérií vrstev až do části maximálního zúžení modelu, neboli nejhlubší vrstvy (bottleneck), zodpovědné za zachycení co nejabstraktnějších rysů a detailů vstupních dat. Tato sekce zároveň přemostuje části kódovací a dekódovací.



Obr. 5.4: Architektura sítě 1D U-Net

Výše zmíněné bloky byly tedy sestaveny do finální architektury modelu dle vzoru autorů Moskalenko V, et al.[13]. Průchod dat strukturou si můžeme přiblížit čtením modelu zleva doprava – enkodér, bottleneck, dekodér. Poslední vrstvu sítě, tedy nejvyšší bod části dekodéru, tvoří finální konvoluční blok, který s počtem filtrů 2 a aktivační funkcí *softmax*. Z vizualizace architektury lze také rozpoznat označení typu modelu U-Net, kdy sestavené bloky jako celek připomínají tvar písmene "U"(Obr. 5.4).

5.4 Trénování a optimalizace parametrů modelu

Nyní se přesuneme k trénování samotného modelu. Nejprve byl načten trénovací set v podobě matice o rozměrech 203371×512 (Tab. 5.2 – Trénovací set), která byla náhodným výběrem v poměru 9:1 rozdělena na trénovací a validační data (Tab. 5.3). Spolu s ní byla shodným způsobem zpracována matice skutečně pravdivých anotací neboli GT.

Poměr	Data	Počet záznamů
90%	Trénovací	183 033
10%	Validační	20 338

Tab. 5.3: Rozdělení dat před trénováním

K trénování modelu spolu s optimalizací parametrů byla vytvořena trénovací smyčka, ve které se postupně vystřídaly kombinace z vybrané množiny hodnot nastavujících jednotlivé parametry, mezi které se řadí krok učení, počet epoch a velikost *batch* (Tab. 5.4).

Vybraným trénovacím algoritmem byl ADAM, který se v dnešní době využívá jako "zlatý standard" (Kap. 2.2). Koeficient DICE, který se běžně také využívá pro vyhodnocení zdatnosti algoritmů segmentace obrazů, byl vybranou ztrátovou funkcí. Podstata koeficientu odpovídá požadavkům na model a sice maximalizovat překryv mezi segmenty na výstupu modelu a segmenty anotovanými (GT). V čitateli můžeme vidět dvojnásobek absolutní hodnoty průniku dvou oblastí (X – predikce, Y – GT) a ve jmenovateli součet jejich absolutních hodnot. Zároveň si můžeme všimnout parametru *smooth*, který měl v našem případě hodnotu 10^{-6} a sloužil jako pojistka pro případ výskytu nulového jmenovatele (Vztah 5.1).

$$DICE = \frac{2 \times |X \cap Y| + smooth}{|X| + |Y| + smooth} \quad (5.1)$$

V trénovací smyčce se tedy vždy na začátku vybraly parametry, kterými byl pojmenován současně trénovaný model (příklad: *Model_lr0.0001_ep15_bs128*). Název následně sloužil k uložení do monitorování postupu trénování modelů ve službě WANDB. Po dokončení každého cyklu a načtení nových parametrů se uložil nový název modelu spolu s novou inicializací WANDB (spuštění monitorování).

Parametry	Hodnoty
Aktivační funkce	softmax
Algoritmus	ADAM
Ztrátová funkce	DICE, BKE
Krok učení	1e-3, 1e-4, 5e-4
Počet epoch	15, 18, 21, 23, 25
Batch size	64, 128, 256

Tab. 5.4: Hyperparametry modelu: BKE – binární křížová entropie

Jeden cyklus smyčky byl ukončen po dosažení maximálního počtu epoch dle nastavené hodnoty nebo při splnění podmínky předčasného zastavení (*Early Stopping*). Tento blok konstantně monitoroval hodnotu validační ztrátové proměnné a pokud se během tří po sobě jdoucích epoch hodnota nesnížila, byl model vyhodnocen jako natrénovaný a konkrétní cyklus byl ukončen. Na konci každého cyklu se model uložil jako samostatný soubor formátu *.h5*. Zároveň se také uložily jeho parametry do souboru formátu *.json* pro lepší orientaci mezi uloženými soubory (Výpis 5.4). Celkově bylo natrénováno přes 30 modelů s různými parametry dle Tab. 5.4.

Výpis 5.4: Příklad uložených parametrů jednoho modelu

```

1 {
2   "model": "1D-UNET",
3   "activation": "softmax",
4   "optimizer": "ADAM",
5   "loss": "DICE",
6   "learning_rate": 0.001,
7   "epochs": 21,
8   "batch_size": 128,
9   "best_validation_loss": 0.006893767509609461
10 }
```

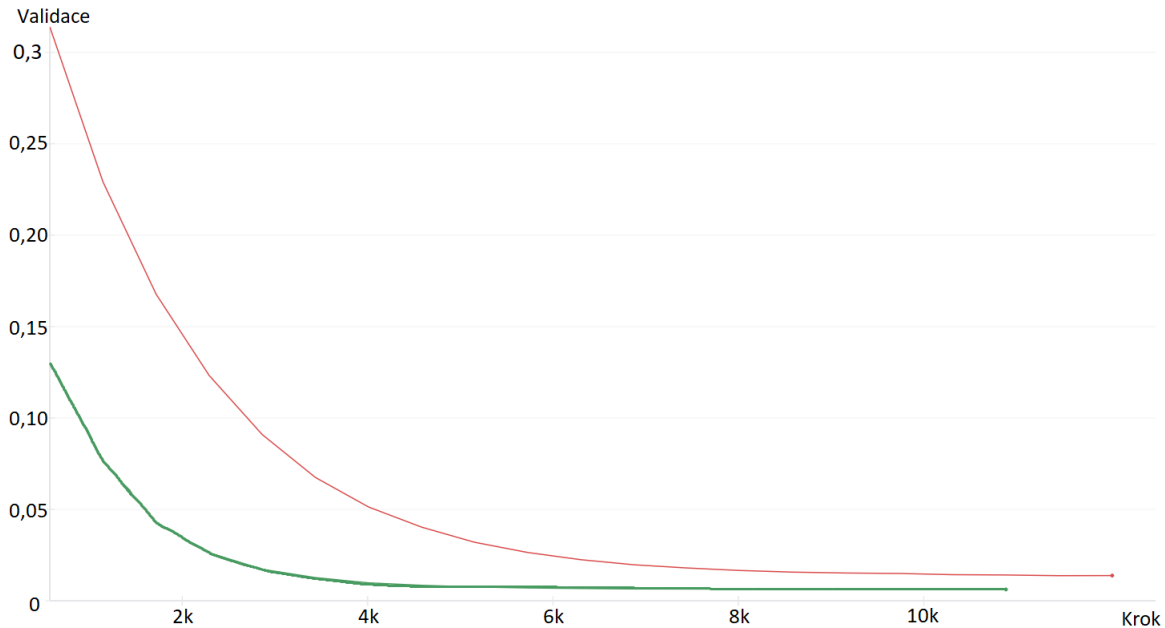
V závěru trénování byl proveden průchod soubory obsahující parametry modelů s účelem sestavení tabulky parametrů a nalezení co nejnižší hodnoty validační ztrátové funkce. Z výsledku bylo zřejmé, že nejlepší výkon podávají modely s krokem učení 0,0001, počtem epoch 21 až 25 a velikostí *batch* 64 až 128, přičemž používají ztrátovou funkci DICE.

$$BKE = -\frac{1}{N} \sum_{i=1}^N [y_i \log(x_i) + (1 - y_i) \log(1 - x_i)] \quad (5.2)$$

Pro demonstraci prospěšnosti použití DICE byly natrénovány modely se shodnými parametry, s rozdílem použití binární křížové entropie (BKE) jako ztrátové funkce, ve které jsou x – predikované hodnoty, y – GT a N – počet observací (Vztah 5.2, Obr. 5.5). Hodnoty validace se ve výsledku lišily v řádech 10^{-4} . Tři nejlepší modely pak postoupily do následující fáze testování (Tab. 5.5).

Ztrátová funkce	Krok učení	Počet epoch	Velikost batch	Validační ztrátová funkce
DICE	0,0001	25	64	0,00585
DICE	0,0001	25	128	0,00615
DICE	0,0001	21	128	0,00646
BKE	0,0001	21	64	0,01370

Tab. 5.5: Parametry 3 nejlepších modelů a srovnání DICE s BKE



Obr. 5.5: Pokles validace za epochu při použití DICE (zelená), proti BKE (červená)

5.5 Testování a zpracování výsledků

Další oddíl praktické části se věnuje testování vybraných modelů na určených množinách dat z 5 databází a zpracování výsledků. Vzhledem ke vzorkovací frekvenci 1000 Hz, kterou mají všechna data společnou, odpovídá jeden vzorek jedné milisekundě. Do testovací sekce postoupily 3 modely, které poskytovaly nejlepší výsledky na základě validační ztrátové funkce. Po načtení jednotlivých databází a predikování vybraným modelem na každé z nich byly predikce uloženy k následnému zpracování a vyhodnocení zvolených metrik.

Na výsledcích bylo nejprve potřeba provést post-processing kvůli výskytu nespojitých segmentů. Výstupní vektory byly naprahovány, jelikož jim funkce *softmax* v poslední vrstvě modelu přiřadila hodnotu z intervalu $[0, 1]$. Na ošetření výskytu nespojitých úseků byla vytvořena funkce *process_and_merge_qrs_segments*, která prochází binarizovaný vektor, ukládá jednotlivé segmenty, jež dosahují délky minimálně 5 ms a zároveň jsou od sebe vzdáleny maximálně 20 ms. Takto skládá segmenty do jednoho celku, dokud nejsou poskládány všechny nalezené, nebo pokud celková délka sestaveného celku přesáhne 150 ms.

Vybrané metriky hodnocení predikce dat se dále rozdělily na tři části. První z nich se zaměřila na komplex QRS jako celek a na úspěšnost jeho detekce a délky trvání v porovnání s GT. Vyhodnocení úspěšnosti detekce reprezentují senzitivita, pozitivní prediktivní hodnota (PPV) a koeficient DICE, který přímo porovnává po-

čet detekovaných vzorků segmentu proti GT. Odchylku v délce trvání (QRSd) od GT reprezentuje střední absolutní chyba (Vztah 5.3) .

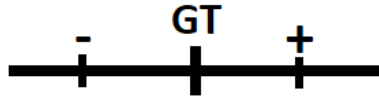
$$MAE = \frac{1}{N} \sum_{i=1}^N |QRSd_{GTi} - QRSd_i| \quad (5.3)$$

Zbývající dvě části hodnocení se zaměřily přímo na detekované počátky a konce komplexu (QRSon a QRSoﬀ), do kterých byly opět zařazeny senzitivita, PPV a průměrná chyba spolu se směrodatnou odchylkou chyby (STD) v [ms] (Vztah 5.4 a 5.5).

$$Chyba = \frac{1}{N} \sum_{i=1}^N (QRS_{ONi} - QRS_{ON,GTi}) \quad (5.4)$$

$$STD = \sqrt{\frac{1}{N} \sum_{i=1}^N (Chyba_i - \overline{Chyba})^2} \quad (5.5)$$

K vyhodnocení metrik senzitivity a PPV pro QRSON a QRSoﬀ byla napsána funkce, do které vstupují predikované hodnoty, GT a hodnota intervalu tolerance, jež udává časovou odchylku v ms o kterou se mohou detekované QRSON a QRSoﬀ lišit od svých skutečných hodnot (GT). To si můžeme představit jako interval hodnot, na kterém uprostřed leží hodnota z GT (QRSON nebo QRSoﬀ) kolem níž se na obě strany rozprostírá interval tolerance (Obr. 5.6).



Obr. 5.6: Interval tolerance kolem GT

Tento přístup se použil k označení predikovaných hodnot jako skutečně pravdivých (TP), falešně pozitivních (FP) a falešně negativních (FN). V případě QRSon byly jako TP počítány všechny predikované hodnoty spadající do intervalu tolerance. Pokud predikovaná hodnota byla mimo interval tolerance a zároveň její rozdíl s GT byl menší než 0, byla přičtena k FP. V opačném případě, tedy rozdíl byl větší než 0, byla přičtena k FN (Výpis 5.5).

Výpis 5.5: Výpočet TP, FP a FN pro QRSon

```
1 {
2 if np.abs(difference) <= tolerance:
3     TP += 1
4 elif np.abs(difference) > tolerance and difference < 0:
5     FP += 1
6 elif np.abs(difference) > tolerance and difference > 0:
7     FN += 1
8 }
```

Obdobným způsobem byly počítány TP, FP a FN pro QRSoFF. Pokud byla predikovaná hodnota mimo rozsah tolerance a její rozdíl s GT byl menší než 0, byla přičtena k FN, v opačném případě k FP (Výpis 5.6).

Výpis 5.6: Výpočet TP, FP a FN pro QRSoFF

```
1 {
2 if np.abs(difference) <= tolerance:
3     TP += 1
4 elif np.abs(difference) > tolerance and difference < 0 :
5     FN += 1
6 elif np.abs(difference) > tolerance and difference > 0:
7     FP += 1
8 }
```

6 Výsledky vybraného modelu a jejich diskuze

V této kapitole budou diskutovány nejlepší dosažené výsledky námi vybraného modelu pod pracovním názvem 1D U-Net s krokem učení 0,0001 počtem epoch 25 a velikostí *batch* 64. Jak již bylo zmíněno v závěru podkapitoly 5.4, do fáze testování postoupily 3 nejlepší modely, ze kterých byl vybrán jeden finální model, jež dosáhl nejlepších výsledků v porovnání s referenčními hodnotami (GT) dle metrik popsanych v podkapitole 5.5. Výsledky jsou rozděleny do 3 tabulek, dle hodnocených částí: QRS segmentace, QRSON a QRSoFF (Tab. 6.1, 6.2, 6.3).

Dataset	Se [%]	PPV [%]	QRSd MAE [ms]	DICE QRS
QTDB	80,96	87,81	21,58	0,84
LUDB	95,75	96,03	7,27	0,96
ECGCIPA	98,19	98,82	2,56	0,99
ECGDMMLD	98,63	99,13	1,72	0,99
ECGRDVQ	97,79	98,02	4,21	0,98

Tab. 6.1: Výsledky segmentace QRS

Dataset	Se [%]	PPV [%]	Chyba [ms]	STD [ms]
QTDB	36,57	71,91	7,18	11,94
LUDB	88,59	88,20	-0,26	5,65
ECGCIPA	99,99	99,80	0,28	1,70
ECGDMMLD	99,99	99,99	-0,05	1,16
ECGRDVQ	97,49	99,59	0,04	2,65

Tab. 6.2: Výsledky QRSON, tolerance 5 ms

U výše vypsanych výsledků pro QRSON a QRSoFF byl nastaven interval tolerance chyby 5 ms. Model se tedy v predikci QRSON a QRSoFF nemohl chybovat o více než 5 vzorků, aby se hodnota stále počítala jako správně určená. Obecně lze u výsledků průměrných chyb vidět tendenci k opožděné predikci (kladné hodnoty) QRSON a v opačném případě dřívější (záporné hodnoty) predikci QRSoFF (Tab. 6.2, 6.3).

Z výsledků první části, která hodnotí detekci QRS komplexu, lze vidět jasný rozdíl úspěšnosti mezi jednotlivými datasety (Tab. 6.1). S výjimkou datasetu QTDB jsou výsledky uspokojující s hodnotami senzitivity a PPV přes 95 %, což svědčí o robustnosti detekce QRS komplexu.

Dataset	Se [%]	PPV [%]	Chyba [ms]	STD [ms]
QTDB	42,86	32,14	-2,21	36,96
LUDB	85,19	90,51	-0,54	5,03
ECGCIPA	99,80	99,70	-0,30	1,80
ECGDMMLD	99,60	99,30	-0,48	1,55
ECGRDVQ	97,28	99,18	-0,20	2,72

Tab. 6.3: Výsledky QRSoft, tolerance 5 ms

Obecně nízká úspěšnost u QTDB je trend, který lze následně pozorovat i při vyhodnocení senzitivity a PPV detekce QRSoft a QRSoft, což s největší pravděpodobností souvisí s počtem dat zastupujících QTDB, který činí nižší tisíce vstupních vektorů v trénovacím datasetu (Tab. 5.2). Porovnání s jinými modely na bázi hlubokého učení u kterých autoři taktéž použili databázi QTDB (na trénování i testování) znázorňuje Tab. 6.4.

	1D U-Net	ConvLSTM [15]	U-Net [14]	CNN [26]
Se [%]	80,96	95,57	99,47	92,20
PPV [%]	87,81	95,83	98,18	93,00

Tab. 6.4: Srovnání segmentace QRS na datasetu QTDB s jinými modely

Nyní se zaměříme na výsledky průměrné chyby a STD počátku a konce QRS pro databázi QTDB ve srovnání s jinými modely (Tab. 6.5). Při komparaci směrodatné odchylky počátku QRS s modelem U-Net [14] dosahuje náš model nižších hodnot. Zároveň jsou tyto výsledky srovnatelné s modelem 1D ConvNet [27]. Průměrná chyba a STD pro QRSoft jsou však horší v porovnání s ostatními, což může souviset s počtem dat v trénovacím setu a také s odlišným přístupem k preprocessingu dat.

	1D U-Net	U-Net [14]	1D ConvNet [27]
QRSoft			
Chyba [ms]	7,18	-3,20	-2,60
STD [ms]	11,94	22,40	10,80
QRSoft			
Chyba [ms]	-2,21	7,90	4,40
STD [ms]	36,96	21,90	15,20

Tab. 6.5: Srovnání výsledků QRSoft a QRSoft na databázi QTDB s jinými modely

	1D U-Net	U-Net [13]
QRSon		
Se [%]	88,59	99,96
PPV [%]	88,20	99,84
Chyba [ms]	-0,26	2,10
STD [ms]	5,65	9,80
QRSoff		
Se [%]	85,19	99,96
PPV [%]	90,51	99,84
Chyba [ms]	-0,54	1,60
STD [ms]	5,03	9,80

Tab. 6.6: Srovnání výsledků QRSon a QRSoff na databázi QTDB s jinými modely

Další výsledky srovnatelné s jinými autory, kteří používali shodná data, jsou počátky a konce QRS pro databázi LUDB. V tomto případě autoři [13] poskytují i výsledky úspěšnosti detekce v podobě senzitivity a PPV. Náš model v porovnání s konkurenční U-Net[13] sice zaostává v detekci počátků a konců, avšak průměrná chyba a směrodatná odchylka dosahují menších rozdílů oproti referenčním hodnotám (Tab. 6.6).

Celkově náš model dosahuje téměř dokonalých výsledků při testování na větších databázích ECGCIPA, ECGMMLD a ECGRDVQ, jejichž QRS komplexy je na základě hodnot senzitivity dosahujících 99,28 % až 99,80 % a hodnot PPV 99,18 % až 99,70 % schopen kvalitně detekovat. Taktéž vysoká hodnota DICE 0,98 až 0,99 svědčí o kvalitě segmentace a nízká hodnota MAE o přesném rozměření délky trvání (Tab. 6.1). Zároveň vykazuje robustnost detekce počátku a konce QRS kde opět detekuje téměř dokonale s nízkou odchylkou dle průměrné chyby a směrodatné odchylky (Tab. 6.2, 6.3).

Vzhledem k odlišným povahám databází ECGCIPA, ECGMMLD a ECGRDVQ, které vycházejí z případových studií zabývajících se vlivem řady medikamentů na člověka, by bylo trénování na takto složeném datasetu dobré řešit trénovací smyčkou, která by modelu předkládala méně zastoupená data z QTDB či LUDB v určitém poměru, nebo častěji úměrně k datům zastoupeným ve větším počtu.

Závěr

Hlavním cílem této diplomové práce bylo vyřešení úkolu detekce počátku a konce komplexu QRS s využitím metod hlubokého učení. V počáteční kapitole teoretického úvodu byla nejprve věnována pozornost základům anatomie a fyziologie srdeční činnosti k uvedení čtenáře do problematiky vzniku srdečního vzruchu a pochopení principu jeho měření. Účelem kapitoly tedy bylo přiblížit propojení fyziologie srdečního orgánu a technického aspektu detekce komplexu QRS, ze kterého pramení zadání této práce.

Následující teoretická kapitola seznámila čtenáře se základními principy umělých neuronových sítí s krátkým úvodem do sítí rekurentních, které jsou součástí řešerše 3. kapitoly, jejich trénování pomocí základních metod gradientního sestupu nebo pokročilejší metody ADAM, jež se v dnešní době široce využívá. Následně byla popsána metodika hlubokého učení a hlubokých neuronových sítí spolu s procesem konstrukce jejich architektury ze základních bloků. Popsány jsou bloky 1D konvoluce, aktivační vrstvy ve formě ReLU nebo softmax a pooling vrstvy sloužící k manipulaci s prostorovými rozměry aktivačních map uvnitř sítě.

Kapitola řešerše metod vycházející z prvního bodu zadání se zabývá aplikací hlubokého učení na rozměrování délky komplexu QRS a jejich potenciální využití v praxi, zejména v rámci šetření času a snaze snížení počtu školených odborníků potřebných k manuální anotaci větších objemů záznamů EKG. Dále jsou popsány odlišné metody segmentace EKG od různých skupin autorů. Ve dvou popsáných přístupech si autoři zvolili model principu enkodér-dekodér architektury U-Net, jež vznikla k účelům klasifikace obrazových dat, avšak v upravené podobě ji lze uplatnit také ke zpracování 1D signálových dat. Dalším popsáným přístupem je použití kombinovaného modelu konvoluční LSTM sítě, která využívá 1D konvoluční vrstvu jako extraktor příznaků ze vstupních dat k získání aktivačních map, jež jsou následně předány LSTM vrstvám.

Druhým bodem zadání, kterému se věnuje následující kapitola, byla řešerše vhodných databází pro testování modelu na detekci začátku a konce komplexu QRS. V rámci kapitoly je popsáno on-line fórum PhysioNet, jež poskytuje služby jako úložiště veřejně přístupných databází biomedicínských dat a software k jejich analýze, převádění mezi datovými typy, řízení vzdáleného přístupu a další prostředky ke zpracování a využití dat. Na tento úvod navazuje technický popis pěti databází obsahujících anotované nahrávky EKG, které sloužily k vývoji a testování našeho modelu.

Předposlední kapitola nejprve popisuje použité programové prostředí, vztažené k postupu práce od načtení dat po vyhodnocení výsledků. Načítání dat a jejich transformace do testovacích množin a následné oddělení trénovací sady dle databází probíhala v prostředí MATLAB pro jeho lepší přehlednost a pohodlnější manipulaci s proměnnými ve *workspace*. V této fázi prošla signálová data také normalizací do intervalu $[0,1]$ a převzorkováním dat z QTDB a LUDB na 1000 Hz. Anotace bylo potřeba převést do podoby masek, které na místech QRS obsahují hodnotu 1. Výstupem fáze předzpracování byly matice obsahující okna s komplexy QRS o délce 512 vzorků dle modelu.

Následná konstrukce architektury modelu vycházejícího z U-Net[13] a součástí potřebných pro trénování a testování se přesunula do prostředí Python s použitím knihoven TensorFlow a Keras, které poskytují nástroje na práci s prvky strojového učení. Konstrukce jednotlivých bloků, ze kterých byl sestaven model jsou pro přehlednost popsány na přiložených částech kódu.

Model byl trénován optimalizačním algoritmem ADAM ve smyčce, která vystřídala řadu parametrů pro optimalizaci na validačním datasetu. K monitorování procesu bylo využito služby Weights & Biases, která nahrává proces učení na cloudové uložení a nabízí řadu nástrojů k vizualizaci uživatelem vybraných metrik, zejména sledování poklesu ztrátové funkce a monitoring zatížení hardware. Výstupem trénování byla sada modelů, ze které se následně vybral finální model 1D U-Net dle nejlepších dosažených výsledků testování, kde bylo přihlíženo zejména k výsledkům z databáze QT u které bylo největším problémem dosáhnout kvalitní detekce počátku a konce QRS.

Po celkovém shrnutí výsledků z testování modelu v poslední kapitole vyplývá celkově nízká přesnost detekce počátku a konce QRS pro databázi QTDB, ale i přes tento neuspokojivý výsledek model vykazuje průměrnou chybu detekce v řádu jednotek ms. U databáze LUDB dokázal model 1D U-Net dosáhnout srovnatelných výsledků průměrné chyby a STD počátku a konce s autory U-Net[13]. Pro zbývající 3 databáze, které měly největší zastoupení dat v trénovacím datasetu, model vykazoval velice vysoké hodnoty senzitivity 97,28 % až 99,99 % a PPV 98,18 % až 99,99 % při detekci začátku a konce QRS s nízkou průměrnou chybou -0,20 ms až -0,30 ms (brzká predikce) pro QRSoff a 0,04 ms až 0,28 ms (pozdější predikce) QRSON.

Literatura

- [1] ROKYTA, Richard. *Fyziologie. Třetí, přepracované vydání (první vydání v nakladatelství Galén)* [online]. Praha: Galén, 2016. [cit. 10. 11. 2023]. ISBN 978-80-7492-238-1.
- [2] *Přednášky z předmětu Fyziologie člověka – Srdce* Brno: MUNI, 2018. [cit. 10. 11. 2023].
- [3] IKEM - Institut Klinické a Experimentální Medicíny. *Aortální insuficience, regurgitace (nedomykavost aortální chlopně)* [online]. Praha, 2015 - 2022. [cit. 12. 11. 2023]. Dostupné z URL:
<<https://www.ikem.cz/cs/aortalni-insuficience-regurgitace-\nedomykavost-aortalni-chlopne/a-432/>>
- [4] UCEBNICE-EKG.CZ. *Teorie EKG* [online]. [cit. 15. 11. 2023]. Dostupné z URL:
<http://www.ucebnice-ekg.cz/index.php?option=com_content&view=article&id=36&Itemid=117>
- [5] KOZUMPLÍK, Jiří. *Přednášky z předmětu Zpracování biologických signálů (AABS)*. Brno, FEKT VUT, 2020. [cit. 12. 11. 2023].
- [6] MORNSTEIN, Vojtěch a kolektiv autorů. *Lékařská fyzika a biofyzika*. Brno: Masarykova univerzita, 2018. ISBN 978-80-210-8984-6. [cit. 15. 11. 2023].
- [7] KOZUMPLÍK, Jiří, Ivo PROVAZNÍK. *Umělá inteligence v medicíně*. Brno: VUT, 2007. [cit. 15. 11. 2023].
- [8] WOOD, Thomas.: *What is the Softmax Function?*. DeepAI [online]. [cit. 25. 11. 2023]. Dostupné z URL:
<<https://deepai.org/machine-learning-glossary-and-terms/softmax-layer>>
- [9] GARRETA, Raúl.: *A gentle guide to machine learning*. MonkeyLearn Blog [online]. [cit. 25. 11. 2023]. Dostupné z URL:
<<https://monkeylearn.com/blog/gentle-guide-to-machine-learning/>>
- [10] WOOD, Thomas.: *What is the Adam Optimization Algorithm?* [online]. DeepAI [cit. 30. 11. 2023]. Dostupné z URL:
<<https://deepai.org/machine-learning-glossary-and-terms/adam-machine-learning>>
- [11] CHMELÍK, Jiří. *Machine learning - Deep Learning 1, Brno: VUT, 2022*.

- [12] OLAH, Christopher. *Understanding LSTM Networks*. Colah's blog [online]. [cit. 2. 12. 2023]. Dostupné z URL:
<<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>>
- [13] MOSKALENKO, Viktor, Nikolai ZOLOTYKH a Grigory OSIPOV. *Deep Learning for ECG Segmentation. Advances in Neural Computation, Machine Learning, and Cognitive Research III*. [online]. Springer International Publishing, 2020, 2020-09-04, 246-254. Studies in Computational Intelligence. ISBN 978-3-030-30424-9. [cit. 8. 12. 2023]. Dostupné z URL:
<[doi:10.1007/978-3-030-30425-6_29](https://doi.org/10.1007/978-3-030-30425-6_29)>
- [14] JIMENEZ-PEREZ, Guillermo; ALCÁINE, Alejandro a CAMARA, Oscar. *U-Net Architecture for the Automatic Detection and Delineation of the Electrocardiogram*. [online]. 2019. [cit. 2024-05-21]. Dostupné z URL:
<<https://doi.org/10.22489/CinC.2019.284>>
- [15] MALALI, Aman, Srinidhi HIRIYANNAIAH, Siddesh G.M., Srinivasa K.G. a Sanjay N.T. *Supervised ECG wave segmentation using convolutional LSTM*. [online]. ICT Express. 2020, 6(3), 166-169. ISSN 24059595. [cit. 6. 12. 2023]. Dostupné z URL:
<<https://doi.org/10.1016/j.icte.2020.04.004>>
- [16] GOLDBERGER, Ary L., Luis A. N. AMARAL, Leon GLASS, et al.: *PhysioBank, PhysioToolkit, and PhysioNet*. [online]. Circulation. 2000, 101(23). ISSN 0009-7322. [cit. 26. 11. 2023]. Dostupné z URL:
<[doi:10.1161/01.CIR.101.23.e215](https://doi.org/10.1161/01.CIR.101.23.e215)>
- [17] LAGUNA P, Mark RG, Goldberger AL, Moody GB. *A Database for Evaluation of Algorithms for Measurement of QT and Other Waveform Intervals in the ECG*. [online]. Computers in Cardiology 24:673-676 (1997). [cit. 26. 11. 2023]. Dostupné z URL:
<<https://physionet.org/physiobank/database/qtdb/doc/index.shtml>>.
- [18] KALYAKULINA, A., YUSIPOV, I., MOSKALENKO, V., NIKOLSKIY, A., KOSONOGOV, K., ZOLOTYKH, N., IVANCHENKO, M. *Lobachevsky University Electrocardiography Database (version 1.0.1)* [online]. PhysioNet. [cit. 20. 12. 2023]. Dostupné z URL:
<<https://doi.org/10.13026/eegm-h675>>

- [19] KALYAKULINA, A., I. YUSIPOV, Viktor A. MOSKALENKO, Alexander V. NIKOLSKIY, Konstantin A. KOSONOGOV, Grigory V. OSIPOV, Nikolai Yu. ZOLOTYKH a Mikhail V. IVANCHENKO. *LUDB: A New Open-Access Validation Tool for Electrocardiogram Delineation Algorithms* [online]. IEEE Access. 2020, 8, 186181-186190. ISSN 2169-3536. [cit. 20. 12. 2023]. Dostupné z URL: <<https://doi.org/10.1109/ACCESS.2020.3029211>>
- [20] VICENTE J, ZUSTERZEEL R, JOHANNESSEN L, OCHOA-JIMENEZ R, MASON J, SANABRIA C, KEMP S, SAGER P, PATEL V, MATTA MK, LIU J, FLORIAN J, GARNETT C, STOCKBRIDGE N, STRAUSS DG *Assessment of Multi-Ion Channel Block in a Phase I Randomized Study Design: Results of the CiPA Phase I ECG Biomarker Validation Study* [online]. Design: Results of the CiPA Phase I ECG Biomarker Validation Study B.m.: Wiley. 2019, 1. [cit. 21. 12. 2023]. Dostupné z URL: <[doi:10.1002/cpt.1303](https://doi.org/10.1002/cpt.1303)>
- [21] *ECG Effects of Dofetilide, Moxifloxacin, Dofetilide+Mexiletine, Dofetilide+Lidocaine and Moxifloxacin+Diltiazem.* [online]. [cit. 22. 12. 2023]. Dostupné z URL: <<https://doi.org/10.13026/C2D016>>
- [22] JOHANNESSEN, L; VICENTE, J; MASON, JW; ERATO, C; SANABRIA, C et al. *Late sodium current block for drug-induced long QT syndrome: Results from a prospective clinical trial.* [online]. 2016, roč. 99, č. 2, s. 214-223. ISSN 0009-9236. [cit. 22. 12. 2023]. Dostupné z URL: <<https://doi.org/10.1002/cpt.205>>
- [23] JOHANNESSEN, L; VICENTE, J; MASON, J W; SANABRIA, C; WAITELABOTT, K et al. *Differentiating Drug-Induced Multichannel Block on the Electrocardiogram: Randomized Study of Dofetilide, Quinidine, Ranolazine, and Verapamil.* [online]. 2014, roč. 96, č. 5, s. 549-558. ISSN 0009-9236. [cit. 22. 12. 2023]. Dostupné z URL: <<https://doi.org/10.1038/clpt.2014.155>>
- [24] *Weights & Biases.* [online]. Wandb. 2018. [cit. 20. 05. 2023]. Dostupné z URL: <<https://wandb.ai/>>
- [25] byschii. *ecg-segmentation.* [online]. GitHub 2022. Dostupné z URL: <<https://github.com/byschii/ecg-segmentation>>

- [26] SODMANN, Philipp; VOLLMER, Marcus; NATH, Neetika a KADERALI, Lars. *A convolutional neural network for ECG annotation as the basis for classification of cardiac rhythms*. [online]. *Physiological Measurement*. 2018, roč. 39, č. 10. ISSN 1361-6579. [cit. 2024-05-21]. Dostupné z URL: <<https://doi.org/10.1088/1361-6579/aae304>>
- [27] CAMPS, Julià; RODRIGUEZ, Blanca a MINCHOLE, Ana. *Deep Learning Based QRS Multilead Delineator in Electrocardiogram Signals*. [online]. 2018. [cit. 2024-05-21]. Dostupné z URL: <<https://doi.org/10.22489/CinC.2018.292>>

Seznam symbolů, veličin a zkratek

<i>ml</i>	Jednotka veličiny objemu
<i>Hz</i>	Jednotka veličiny frekvence
<i>ms</i>	Jednotka veličiny času
UNS	Umělá neuronová síť
ML	Strojové učení (angl. zkratka <i>Machine Learning</i>)
ReLU	Reálná funkce jedné reálné proměnné (angl. zkratka <i>Rectified Linear Unit</i>)
GD	Gradientní sestup (angl. zkratka <i>Gradient Descent</i>)
ADAM	Adaptive Moment Estimation
LSTM	Long Short-Term Memory
RNN	Rekurentní neuronová síť (angl. zkratka <i>Recurrent Neural Network</i>)
LUDB	Lobachevsky University Database
WANDB	Weights & Biases
DICE	Dice koeficient
BKE	Binární křížová entropie
GT	Základní pravda (angl. zkratka <i>Ground Truth</i>)
MAE	Střední absolutní chyba (angl. zkratka Mean Absolute Error)
PPV	Pozitivní prediktivní hodnota (angl. zkratka Positive Predictive Value)
QRSd	Délka trvání QRS
QRSon	Počátek komplexu QRS
QRSoff	Konec komplexu QRS