

SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 2918

**KLASIFIKACIJA SRČANIH RITMOVA IZ SNIMAKA  
12-KANALNOG ELEKTROKARDIOGRAMA KORISTEĆI  
KOMBINACIJU PRISTUPA DUBOKOG UČENJA**

Marko Barišić

Zagreb, lipanj 2022.

SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 2918

**KLASIFIKACIJA SRČANIH RITMOVA IZ SNIMAKA  
12-KANALNOG ELEKTROKARDIOGRAMA KORISTEĆI  
KOMBINACIJU PRISTUPA DUBOKOG UČENJA**

Marko Barišić

Zagreb, lipanj 2022.

## DIPLOMSKI ZADATAK br. 2918

Pristupnik: **Marko Barišić (0036505569)**

Studij: Računarstvo

Profil: Računarska znanost

Mentor: izv. prof. dr. sc. Alan Jović

Zadatak: **Klasifikacija srčanih ritmova iz snimaka 12-kanalnog elektrokardiograma koristeći kombinaciju pristupa dubokog učenja**

### Opis zadatka:

Prema podacima Svjetske zdravstvene organizacije, kardiovaskularne bolesti najčešći su i najrašireniji uzrok smrti na svijetu. Uzrok su procijenjenih 32% smrti godišnje. Stoga je pravovremeno uočavanje srčanih aritmija izrazito bitno. Osnova za istraživanja u ovom području je elektrokardiogram (EKG), tj. snimka električne aktivnosti srca. Ustaljeni pristup je da stručnjaci snimaju i analiziraju EKG. Automatska klasifikacija srčanih ritmova nudi se kao moderni pristup ovom problemu s potencijalom poboljšanja dijagnostike i spašavanja života. U tu svrhu, potrebno je razviti model dubokog učenja za klasifikaciju srčanih ritmova iz snimaka 12-kanalnog EKG-a. Skup podataka će se prikupiti iz javno dostupnih izvora, a sastojat će se od snimaka EKG-a s označenim srčanim ritmom. Neuravnoteženost klasa u skupu podataka je učestali problem u ovom području, jer su neki srčani ritmovi rjeđi u snimkama od drugih. U svrhu rješavanja tog problema, primijenit će se postupak poboljšavanja podataka kako bi se, pomoću modela dubokog učenja, umjetno generirale vjerodostojne snimke EKG-a određenih srčanih ritmova. U radu će se koristiti kombinacija metoda konvolucijskog auto-enkodera i duboke neuronske mreže LSTM. Konvolucijski auto-inkoder će služiti za stvaranje sažetih snimaka EKG-a s naglašenih obilježjima. Takve sažete snimke EKG-a bit će korištene kao ulaz u duboku neuronsku mrežu LSTM, koja će služiti za konačnu klasifikaciju srčanih ritmova. Rješenje će se vrednovati na pripremljenom testnom skupu podataka te će se usporediti s rezultatima srodnih znanstvenih radova iz literature.

Rok za predaju rada: 27. lipnja 2022.

*Zahvaljujem se  
svom mentoru izv. prof. dr. sc. Alanu Joviću na suradnji zadnje tri godine i izvrsnom  
prijedlogu teme za završni rad koju sam uz njegovo mentorstvo dublje istražio za ovaj  
diplomski rad. Također, zahvaljujem se svojoj obitelji na potpori i povjerenju koje mi  
oduvijek ukazuju.*

*Hvala vam!*

# SADRŽAJ

<b>1. Uvod</b>	<b>1</b>
<b>2. Srce i elektrokardiogram</b>	<b>3</b>
2.1. Srce . . . . .	3
2.1.1. Anatomija srca . . . . .	3
2.1.2. Rad srca . . . . .	4
2.1.3. Sustav električne provodljivosti srca . . . . .	5
2.2. 12-kanalni elektrokardiogram . . . . .	6
2.2.1. Princip rada . . . . .	6
2.2.2. Analiza snimke . . . . .	8
2.3. Ritmovi i aritmije . . . . .	9
<b>3. Srodni radovi</b>	<b>12</b>
<b>4. Skup podataka i metode</b>	<b>15</b>
4.1. Skup podataka . . . . .	15
4.2. Predobrada snimke EKG-a – Pan-Tompkinsov algoritam . . . . .	18
4.3. Duboko učenje . . . . .	19
4.3.1. Konvolucijske mreže . . . . .	21
4.3.2. Konvolucijski auto-ekoder . . . . .	23
4.3.3. Povratne mreže . . . . .	24
4.3.4. Mreža duge kratkoročne memorije – LSTM . . . . .	26
4.3.5. Generativna suparnička mreža – GAN . . . . .	28
<b>5. Izvedba</b>	<b>31</b>
5.1. Pan Tompkinsov algoritam . . . . .	31
5.2. Izvedba dubokih konvolucijskih auto-ekodera . . . . .	35
5.3. Izvedba generativnih suparničkih mreža . . . . .	39
5.4. Poboljšavanje skupa podataka . . . . .	42

5.5. Izvedba duge kratkoročne memorije . . . . .	45
<b>6. Rezultati i rasprava</b>	<b>49</b>
6.1. Rekonstrukcija . . . . .	49
6.2. Klasifikacija . . . . .	51
<b>7. Zaključak</b>	<b>54</b>
<b>Literatura</b>	<b>55</b>

# 1. Uvod

Prema službenim podacima Svjetske zdravstvene organizacije, kardiovaskularne bolesti najčešći su uzrok smrti na svijetu [1]. Procjenjuje se da su uzrok 17.9 milijuna smrti godišnje, što predstavlja 32% svih smrti godišnje [1]. Znanstvena istraživanja imaju vrlo važnu ulogu u umanjivanju ovih brojki. Temelj znanstvenih istraživanja u području kardiologije je elektrokardiogram, uređaj koji bilježi električnu aktivnost srca. Prikupljeni podaci, u obliku snimaka elektrokardiograma, imaju široku upotrebu kako u znanstvenoj, tako i u praktičnoj primjeni.

U klasičnom pristupu, signali EKG-a se bilježe i promatraju kroz određeni period vremena kako bi konačno bili analizirani od strane stručne osobe. Danas pak, automatska klasifikacija srčanih ritmova nudi priliku za poboljšanje dijagnostike i spašavanje života. Takav se sustav može primijeniti za klasifikaciju srčanih ritmova u stvarnom vremenu u nosivoj tehnologiji, srčanim holterima i zdravstvenim ustanovama. Zadatak klasifikacije srčanih ritmova iz signala EKG-a općenito se može podijeliti u tri glavna koraka. Prvi korak je prikupljanje podataka, slično kao i u drugim zadacima strojnog učenja. Zatim dolazi izlučivanje značajki koje počinje detekcijom ključnih točaka u snimci EKG-a. Treći i konačni korak je klasifikacija srčanog ritma. Tradicionalni pristup zasnovan na pravilima nije učinkovit kada se bavimo velikim skupovima podataka jer zahtjeva značajnu analizu i ekspertno znanje [2]. Duboko učenje, s druge strane, pokazuje izvanredne rezultate u istraživanjima klasifikacije snimaka EKG-a u prošlim godinama [3]. Hijerarhijska arhitektura i ne-linearnost u dubokom učenju omogućuju izlučivanje značajki višeg reda (reprezentacije) te snažne sposobnosti pri izlučivanju značajki uvelike doprinose klasifikaciji [3].

Cilj ovog rada je iskoristiti kombinaciju pristupa dubokog učenja u svakom od navedenih koraka kako bismo implementirali rješenje s kraja na kraj (engl. *end-to-end solution*) za automatsku klasifikaciju pet srčanih ritmova (normalni sinusni ritam te četiri česte aritmije).

Rad je strukturiran na sljedeći način. U poglavlju **Srce i elektrokardiogram** proučava se područje kardiologije, rad srca, princip rada EKG-a, značajke ispisa EKG-a i

objašnjenje ritmova i aritmija promatranih u ovom radu. Nadalje, u poglavlju **Srodni radovi** proučavaju se srodni radovi iz ovog polja istraživanja. Poglavlje **Skup podataka i metode** opisuje korišteni skup podataka, metodu predobrade snimke EKG-a te je većina poglavlja posvećena teorijskoj strani korištenih modela dubokog učenja. Nakon slijedi poglavlje **Izvedba** koje detaljno opisuje na koji način je izveden svaki dio ovog diplomskog rada. Zatim, poglavlje **Rezultati i rasprava** predstavlja eksperimentalne rezultate i raspravlja ih. Konačno u poglavlju **Zaključak** osvrće se na čitav rad te se nude ideje za poboljšanje i nastavak istraživanja ove teme.



## 2. Srce i elektrokardiogram

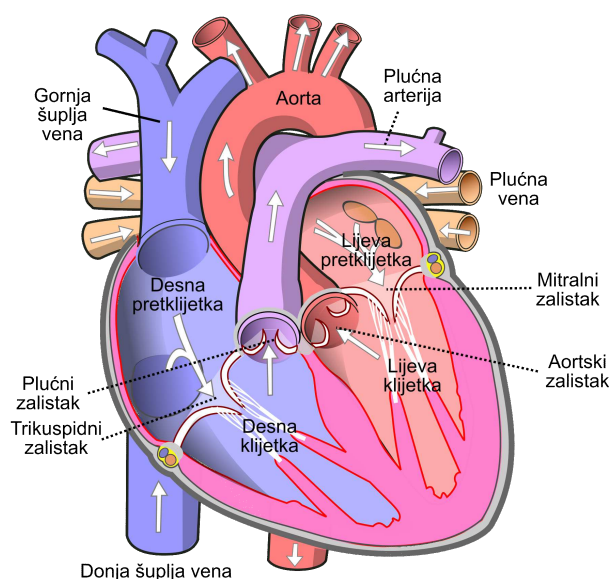
Znanje iz područja kardiologije ključno je za ovaj diplomski rad, prvenstveno za razumijevanje zadatka i rezultata. Ono je također važno za razumijevanje i primjenu algoritama objašnjenih u poglavlju 4. U nastavku ovog poglavlja proučava se srce, anatomija i rad srca, te zapis EKG-a, princip njegovog snimanja te analize snimke. Većina navedenoga preuzeta je iz [4].

### 2.1. Srce

Srce je mišić veličine šake smješten u prsnoj šupljini, najčešće na lijevoj strani, između dva plućna krila. Srce je ključni dio krvožilnog sustava čovjeka u kojem se ponaša kao pumpa.

#### 2.1.1. Anatomija srca

Srce je uzdužno podijeljeno na arterijski i venski dio. Arterijski dio obuhvaća lijevi dio srca, dok venski dio obuhvaća desni dio. Oba dijela su dalje podijeljena na još dva dijela, desnu i lijevu pretklijetku te desnu i lijevu klijetku. Drugi naziv za pretklijetku je atrij, a za klijetku ventrikul. Na slici 2.1 atriji su označeni kao pretkomore, a ventrikuli kao komore. Na prijelazu iz atrija u ventrikul nalaze se srčani zalisci. Bikuspidalni zalisci nalaze se na lijevoj, a trikuspidalni zalisci na desnoj strani. Srce je s ostatkom krvožilnog sustava povezano krvnim žilama. Vene su krvne žile koje dovode krv u srce, a arterije su krvne žile koje odvođe krv iz srca. Gornja i donja šuplja vena dovode deoksigeniranu krv iz cijelog tijela u desni atrij. Plućnim venama oksigenirana krv dolazi u lijevi atrij. Plućna arterija iz desnog ventrikula odvodi krv u pluća. Iz lijevog ventrikula izlazi najšira krvna žila, aorta. Plućna arterija i aorta također imaju zaliske, oni su polumjesečastog oblika. Zalisci imaju važnu ulogu u radu srca, oni sprečavaju vraćanje krvi iz žila u srce.



**Slika 2.1:** Anatomija ljudskog srca [5]

### 2.1.2. Rad srca

Srce se stalno steže i opušta. Stezanje srca se naziva sistola, a opuštanje dijastola. Kao što je navedeno u prethodnom potpoglavlju, vene dovode, a arterije odvođe krv iz srca. Postoje veliki i mali optok krvi, tj. plućni i sistemski optok krvi.

Mali optok krvi započinje sistolom, deoksigenirana krv plućnom arterijom odlazi u pluća. U plućima se ta krv oksigenira, oslobađa se ugljikov dioksid iz krvi te se krv obogaćuje kisikom. Iz pluća krv se vraća u lijevi atrij plućnim venama. Tada krv iz lijevog atrija odlazi u lijevi ventrikul od kuda se potiskuje u aortu.

Veliki optok krvi započinjem sistolom, oksigenirana krv aortom odlazi u cijelo tijelo. Iz aorte krv teče kroz sve manje i manje arterije. Konačno dolazi u kapilare kojima teče i izmjenjuje plinove sa stanicama. Krv predaje stanicama kisik, a od njih preuzima ugljikov dioksid. Time oksigenirana krv postaje deoksigenirana krv te putuje venulama, malim i velikim venama. Konačno preko gornje i donje šuplje vene dolazi do desnog atrija. Tu se sistolom potiskuje u desni ventrikul. To je kraj velikog optoka krvi i ponovno počinje mali optok krvi.

Količina krvi, koja se u periodu trajanja jedne sistole potisne u aortu, naziva se udarni volumen srca. Udarni volumen srca u prosjeku iznosi 70 ml.

### 2.1.3. Sustav električne provodljivosti srca

Srce radi samostalno. Središte rada srca je sustav električne provodljivosti srca. Glavna uloga tog sustava je stvaranje i prijenos električnih impulsa ostatku stijenke srčanog mišića (miokarda). Ovaj proces stvara električnu energiju koju očitavaju elektrode EKG-a, više o tome u poglavlju 2.2 Dijelovi tog sustava su sinus-atrijski čvor (S-A čvor), atrio-ventrikularni čvor (A-V čvor), Hissov snop te Purkinjeova vlakna. Svi dijelovi protežu se miokardom.

S-A čvor nalazi se u desnom atriju. Ovaj dio sustava zadužen je za određivanje frekvencije otkucaja srca, ima ulogu glavnog održavatelja ritma rada (pejsmejker) srca. S-A čvor održava frekvenciju u rasponu od 60 do 100 otkucaja u minuti, u prosjeku frekvencija iznosi 70 otkucaja/min. Usporeni rad srca, manje od 60 otkucaja/min, naziva se sinusna bradikardija, dok se ubrzani rad srca, više od 100 otkucaja/min, naziva sinusna tahikardija.

A-V čvor slijedno je sljedeći u sustavu električne provodljivosti srca. Nalazi se na prijelazu iz desnog atrija u desni ventrikul. A-V čvor također služi kao sporedni pejsmejker srca, održava rad srca u rasponu od 40 do 60 otkucaja u minuti. Ukoliko S-A čvor ne pokrene električni impuls, A-V čvor preuzima ulogu glavnog pejsmejker srca.

Hissov snop služi za prijenos impulsa od A-V čvora nadalje. Nalazi se dijelom u zidu desnog atrija i u pregradi između ventrikula. Ima desnu i lijevu granu koje se granaju prema odgovarajućim ventrikulima. Hissov snop jedini je komunikacijski kanal između atrija i ventrikula. Purkinjeova vlakna granaju se iz Hissovog snopa. Ona su skup stanica koje izravno podražuju desni i lijevi ventrikul.

Stanice miokarda u mirovanju imaju negativni električni potencijal na membrani, otprilike od -70 do -90 mV. Natrij-kalijeva pumpa polako smanjuje negativni naboj unutar stanice te povećava električni potencijal. U jednom se trenutku naboj unutar i izvan stanice ujednače. To se naziva depolarizacijom. Tada ioni kalcija ulaze u stanice i događa se stezanje miokarda. Zatim se stanica nastoji vratiti u stanje mirovanja i to čini smanjenjem pozitivnog naboja unutar sebe. Konačno dolazi do repolarizacije stanica i događa se opuštanje miokarda. Dakle, depolarizacija se očituje stezanjem miokarda, dok se repolarizacija očituje opuštanjem miokarda. Depolarizacija i repolarizacija stanica miokarda prenose se sustavom električne provodljivosti srca.

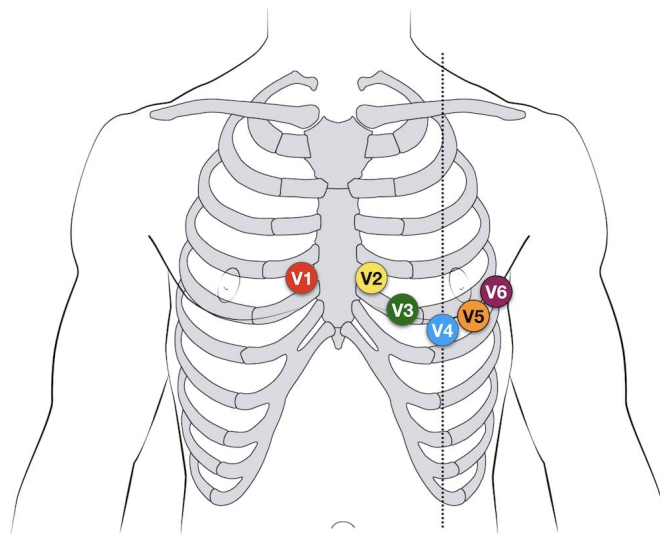
## 2.2. 12-kanalni elektrokardiogram

Elektrokardiograf je uređaj koji zapisuje električnu aktivnost srca u obliku valova elektrokardiograma. Elektrodama, kojima se spaja na pacijenta, detektira električnu aktivnost. U ovom poglavlju objašnjen je 12-kanalni EKG.

### 2.2.1. Princip rada

Elektrode EKG-a detektiraju električnu aktivnost srca koja se odvija ispod njih. Kada se pozitivno nabijeni električni impuls udaljava od elektrode, EKG ga zapisuje kao negativni val. Kada se takav električni impuls giba prema elektrodi, EKG ga zapisuje kao pozitivni val. Dakle, EKG pretvara električnu aktivnost detektiranu elektrodama u valove. Kanale se može shvatiti kao perspektivu iz koje se promatra električna aktivnost srca.

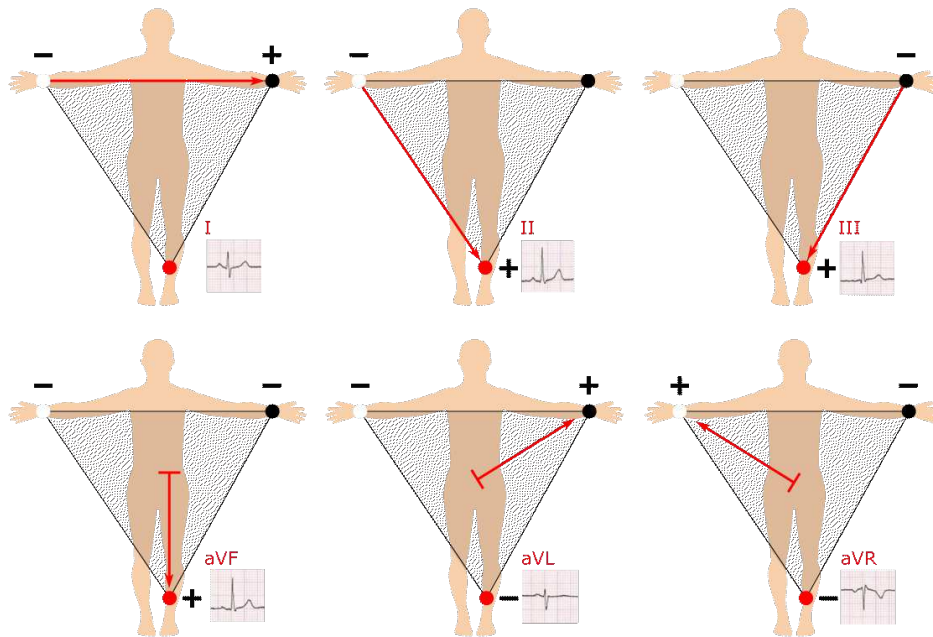
12-kanalni EKG ima 10 elektroda. Četiri elektrode postavljaju se na udove, obje ruke i noge. Za ove elektrode nije bitan točan položaj na udu, već je ključno da su udaljene barem 10 cm od srca. Ostalih šest elektroda postavlja se kako je prikazano na slici 2.2.



Slika 2.2: Položaj prekordijalnih elektroda [6]

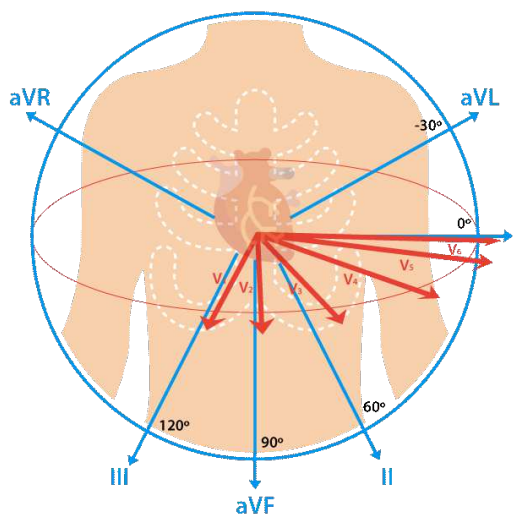
Elektrode na rukama imenovane su RA, LA, RL, LL gdje R i L označavaju desno (R) i lijevo (L), a A i L označavaju ruku (A) i nogu (L). Elektroda RL se ne prikazuje na očitavanju EKG-a, već ona služi kao svojevrsni stabilizator za smanjenje smetnji pri mjerenju EKG-om. Ostale tri elektrode čine točke Einthovenovog trokuta pomoću kojeg se tvori 6 kanala od njih 12. Ti kanali su I, II, III, aVF, aVL i aVR. Princip

Einthovenovog trokuta prikazan je na slici 2.3. Stvoreni kanali promatraju električnu aktivnost srca na poprečnom presjeku srca.



**Slika 2.3:** Einthovenov trokut [7]

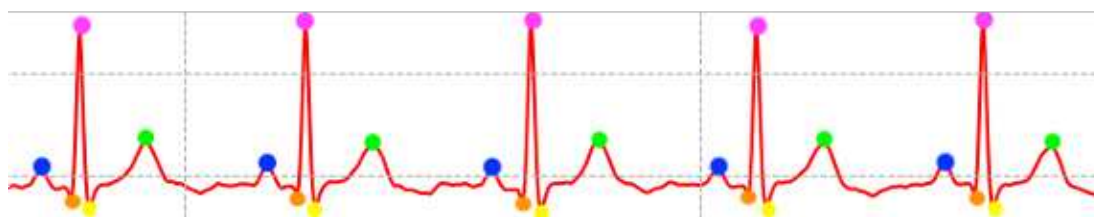
Prekordijalne elektrode stvaraju ostalih šest kanala. Kanali su istoimeni s nazivima odgovarajućih elektroda. Ti kanali promatraju električnu aktivnost srca na horizontalnom presjeku. Svih 10 elektroda zajedno tvori dvanaest kanala s kojima se dobiva potpuni prikaz električne aktivnosti srca. Slika 2.4 prikazuje 12 kanala stvorenih pomoću 10 elektroda.



**Slika 2.4:** Svih 12 kanala EKG-a [7]

## 2.2.2. Analiza snimke

Snimka EKG-a sastoji se od raznih valova. Svaki od tih valova predstavlja dio procesa rada srca. Na slici 2.5 prikazana je označena snimka EKG-a osobe s normalnim sinusnim ritmom. Plave, narančaste, roze, žute i zelene točke redom odgovaraju vrhovima P valova, Q valova, R valova, S valova i T valova. Slike 2.5 – 2.12 iscrtane su u sklopu programske izvedbe zadatka.



**Slika 2.5:** Označena snimka normalnog sinusnog ritma (kanal I)

P val obično je prvi val koji se pronalazi gledajući snimku EKG-a. On predstavlja depolarizaciju oba atrija. Ovaj val započinje kada S-A čvor okine. P val normalno traje od 0.08 do 0.11 sekundi. Nakon P vala dolazi Tp val koji predstavlja repolarizaciju atrija. Tp val je obrnuto orijentiran od P vala. Budući da se odvija u isto vrijeme kada i kompleks QRS često ga ne vidimo na snimci. Dio snimke između kraja P vala i početka kompleksa QRS naziva se PR odsječkom. PR odsječak predstavlja prijenos električnog impulsa depolarizacije kroz čvor A-V, Hissov snop te Purkinjeova vlakna. PR interval predstavlja vremenski period od početka P vala do početka kompleksa QRS. On predstavlja faze rada srca obuhvaćene P valom i PR odsječkom. Normalno trajanje PR intervala iznosi između 0.11 i 0.20 sekundi.

Kompleks QRS dolazi nakon P vala, tj. nakon PR segmenta. Sastoji se od dvaju ili više valova, uobičajeno od Q, R i S vala. kompleks QRS predstavlja depolarizaciju ventrikula. Q val prvi je negativni val kompleksa QRS. Val R prvi je pozitivni val nakon P vala. S val dolazi nakon i on je negativan. Kompleks QRS normalno traje od 0.06 do 0.11 sekundi. Od kraja kompleksa QRS do početka T vala proteže se ST odsječak. On predstavlja period mirovanja električnog naboja između depolarizacije i repolarizacije ventrikula. Točka na snimci EKG-a gdje kompleks QRS završava i ST odsječak započinje naziva se točka J.

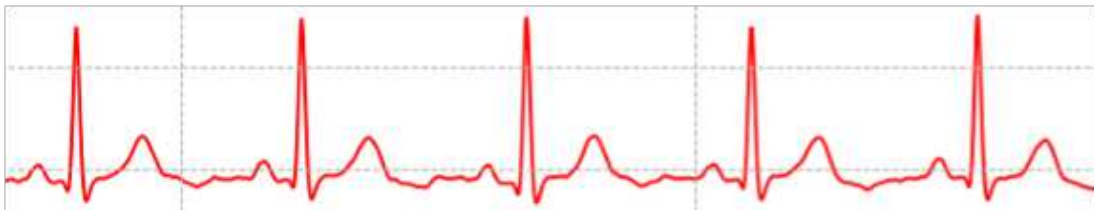
Val T pozitivno je orijentirani val koji dolazi nakon kompleksa QRS, tj. netom nakon ST odsječka. On predstavlja repolarizaciju ventrikula. Vremenski period od početka kompleksa QRS do kraja T vala naziva se QT intervalom. Unutar QT intervala odvijaju se svi događaji sistole i dijastole ventrikula. T valom završava jedan ciklus

i nakon njega dolazi novi ciklus s P valom na čelu. Ipak, između T vala i P vala ponekad se može vidjeti malen i dosta spljošten val. Taj se val naziva U valom i ima istu orijentaciju kao i njegov prethodnik, val T.

### 2.3. Ritmovi i aritmije

U ovom poglavlju opisani su svi ritmovi, tj. većinom aritmije koje su prisutne u skupu podataka korištenom u ovom završnom radu. Opisan je normalni sinus ritam te osam različitih aritmija. Sve referencirane snimke su iz korištenog skupa podataka. Iscrtane su unutar implementacije opisane u 5 poglavlju.

Normalni sinusni ritam predstavlja normalno stanje rada srca gdje je čvor S-A glavni pejsmejker. Vremenski intervali na snimci, PR interval, QT interval, R-R interval, su ravnomjerni i unutar normalnih granica. Na slici 2.6 prikazana je snimka normalnog sinusnog ritma. Frekvencija rada srca iznosi između 60 i 100 otkucaja u minuti. Ako frekvencija iznosi više od 100 otkucaja/min radi se o sinus bradikardiji. Ako frekvencija iznosi manje od 60 otkucaja/min radi se pak o sinusnoj tahikardiji.



Slika 2.6: Snimka normalnog sinusnog ritma (kanal I)

Blok desne grane snopa aritmija je pri kojoj se električni impuls uspješno provodi kroz lijevu granu Hissovog snopa do lijevog ventrikula, a prijenos impulsa prema desnom ventrikulu je usporen. To uzrokuje sporiju depolarizaciju što se na snimci EKG-a očituje produljenjem QRS intervala na 0.12 sekundi ili više. Produljeni i spori S valovi još jedna su značajka. Na snimci prekordijalnog kanala V1, slika 2.7, uočljiva je dodatna značajka ove aritmije. Kompleksa QRS izobličen je u takozvani oblik zečjih ušiju kao što je prikazano na slici 2.8.

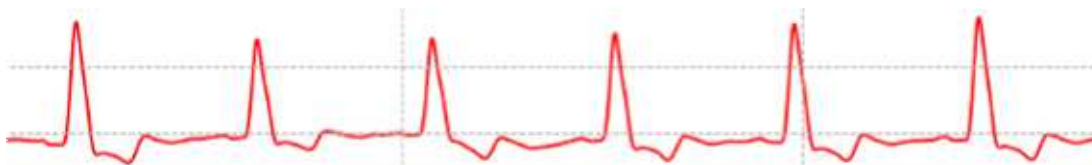


**Slika 2.7:** Snimka bloka desne grane snopa (kanal I)



**Slika 2.8:** Snimka bloka desne grane snopa (kanal V1)

Blok lijeve grane snopa, obrnuto od bloka desne grane snopa, aritmija je pri kojoj električni impuls prvo prolazi desnom granom Hissovog snopa. Depolarizacija desnog ventrikula prethodi depolarizaciji lijevog. Na snimci, slika 2.9, očituju se široki R valovi bez prethodećih Q valova. Također, na snimci kanala V6 vidljiva je zanimljiva značajka, vrh R vala izgleda kao slovo M. Ta se značajka vidi na slici 2.10.



**Slika 2.9:** Snimka bloka lijeve grane snopa (kanal I)

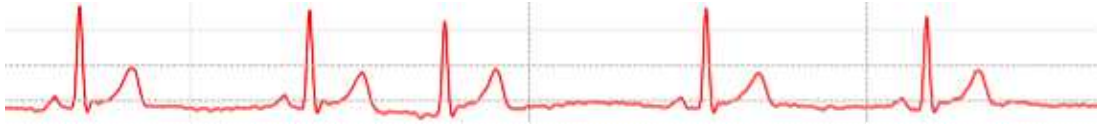


**Slika 2.10:** Snimka bloka lijeve grane snopa (kanal V6)

Preuranjena kontrakcija atrija aritmija je pri kojoj stanica s ulogom pejsmejкера u atriju okida većom frekvencijom nego čvor S-A. To rezultira uranjenom pojavom PQRS ciklusa u jednom trenutku. Nakon toga se čvor S-A resetira, ali pauza zbog



uranjenje kontrakcije atrija nije nadomještena. Na slici 2.11 očigledna je preuranjena kontrakcija atrija.



**Slika 2.11:** Snimka preuranjene kontrakcije atrija (kanal I)

Preuranjena kontrakcija ventrikula, kao i već opisana preuranjena kontrakcija atrija, događa se zbog bržeg okidanja stanice u ventrikulu nego čvora S-A. To uzrokuje uranjenju depolarizaciju ventrikula što ih stavlja u stanje oporavka srca. Kada normalni pejsmejker, čvor S-A, okine, ventirkuli nisu još repolarizirani. Zbog toga sistola ventrikula se ne odvija kada je predviđeno. Srce taj događaj ispravlja pauzom za kompenzaciju koja se na snimci EKG-a očituje vremenskim intervalom između dva vrha R valova između kojih se nalazi kompleks QRS bizarnog oblika.



**Slika 2.12:** Snimka preuranjene kontrakcije ventrikula (kanal I)

### 3. Srodni radovi

U ovom poglavlju su ukratko predstavljene srodni radovi koji rješavaju isti ili sličan zadatak kao onaj kojim se ovaj diplomski radi bavi. U uvodu je navedeno kako se zadatak klasifikacije srčanih ritmova iz snimke EKG-a općenito može podijeliti u tri glavna koraka.

Počevši s drugim korakom, tradicionalni pristup izlučivanju značajki iz snimke EKG-a svodi se na uporabu determinističkih algoritama. Pan Tompkinsov algoritam za detekciju kompleksa QRS [8], objavljen 1985. godine, ustalio se kao jedan od korištenijih algoritama u istraživanjima klasifikacije srčanih ritmova. Ovaj algoritam omogućava vrlo preciznu detekciju kompleksa QRS, točnije R vrhova, u snimkama EKG-a. Ovaj algoritam detaljnije je objašnjen u poglavlju 4.2. U nešto novijem radu od Elgendi et al. [9], predstavljen je algoritam za preciznu detekciju drugih ključnih točaka, P i T valova, na snimci EKG-a.

Unatoč značajnom napretku u tradicionalnom pristupu, u zadnje vrijeme pristupi koji koriste duboko učenje postižu odlične rezultate u izlučivanju značajki snimke EKG-a. Šarlija et al. [10] se u svome radu bave problemom detekcije kompleksa QRS koristeći 1D konvolucijsku neuronsku mrežu (engl. *1D Convolutional neural network*, *1D CNN*) za klasifikaciju. Arhitektura mreže sadrži dva konvolucijska sloja između kojih se nalazi jedan sloj sažimanja maksimalnom vrijednošću (engl. *max pooling layer*) iza kojeg slijede dva potpuno povezana sloja (engl. *fully connected layer*). Ova konvolucijska mreža služi kao pravilo odlučivanja koje označuje QRS komplekse s višestrukim pozitivnim izlaznim vrijednostima. Te izlazne vrijednosti se tada grupiraju i centralna mjera se koristi za konačnu odluku o detekciji kompleksa QRS. Skup podataka koji se koristi u ovom radu sadrži 44 snimke EKG-a iz baze podataka *MIT-BIH Arrhythmia Database* [11]. Ovaj pristup detekcije kompleksa QRS postiže senzitivnost (engl. *sensitivity*) od 99.81% i pozitivnu prediktivnu vrijednost (engl. *positive predictive value*) od 99.3%.

Nastavljajući s trećim korakom, klasifikacijom srčanih ritmova, najbolje rezultate često postižu pristupi strojnog učenja, posebice duboko učenje. Sanino et al. [12]

razvijaju model dubokog učenja za binarnu klasifikaciju srčanih ritmova koristeći arhitekturu sa sedam potpuno povezanih slojeva. Podatke su prikupili iz baze podataka *MIT-BIH Arrhythmia Database* [11] te ih označili oznakama *normalni* i *abnormalni*. Signali snimke EKG-a su prvo filtrirani, zatim se uklanja šum i segmentira snimku u otkucaje srca. Slijedi izlučivanje značajki koristeći deterministički algoritam nakon čega se otkucaj i njegove značajke šalju modelu dubokog učenja. Ulaz dubokog modela sačinjava 50 otkucaja srca i 4 temporalne značajke. Postižu točnost (engl. *accuracy*), senzitivnost i specifičnost (engl. *specificity*) od preko 99%.

Zbog mogućnosti modela dubokog učenja, posebice CNN-ova, duboko učenje se koristi za izlučivanje značajki i klasifikaciju. Podaci ne prolaze kroz algoritam za izlučivanje značajki, već se šalju direktno na ulaz dubokog modela, što rezultira rješenjem s kraja na kraj. Sljedeći rad, Riberio et al. [13], koristi četiri rezidualna bloka postavljena između konvolucijskih slojeva na početku i potpuno povezanih slojeva na kraju mreže. Skup podataka se sastoji od 2,322,513 snimaka EKG-a prikupljenih od više od milijun i pol pacijenata iz Brazila. Ovaj rad klasificira šest srčanih ritmova s F1-mjerom od preko 80% i specifičnošću od preko 99%.

Pyakillya et al. [14] predstavlja model dubokog učenja kao rješenje s kraja na kraj sa sedam 1D konvolucijskih slojeva za izlučivanje značajki popraćenih trima potpuno povezanim slojevima za klasifikaciju. U ovom radu se koriste podaci s natjecanja *PhysioNet/Computing in Cardiology Challenge 2017* [15]. Ovo rješenje klasificira snimke EKG-a u četiri razreda; normalni sinusni ritam, fibrilacija atrijska, drugi ritam i šumoviti ritam. Postignuta je točnost od 86%.

Vrlo zanimljivi rad, Yildirim et al. [16], koristi uzbudljivu kombinaciju pristupa dubokog učenja koja je motivirala ovaj diplomski rad. Temelj ovog rada je LSTM-klasifikator ((engl. *long short-term memory*)) klasifikator. Predložena je neuronska mreža s LSTM-slojem na početku, što prate dva potpuno povezana sloja sa slojem isključivanja neurona (engl. *dropout*) slojem između. Na izlazu drugog potpuno povezanog sloja nalazi se aktivacijska funkcija *softmax*. Klasifikator se koristi za klasifikaciju snimaka EKG-a u jedan od pet razreda. Kako bi se smanjio komputacijski trošak, implementirana je nelinearna struktura za kompresiju koja smanjuje veličinu ulaza u LSTM-klasifikator. Ta nelinearna struktura za kompresiju je ostvarena kao konvolucijski auto-ekoder, duboki model koji radi kompresiju (kodira) i dekompresiju (dekodira) signala snimke EKG-a. Komprimirani signal snimke EKG-a koristi se kao ulaz za LSTM-klasifikator čime se štedi na ukupnoj količini računskih resursa i vremena potrebnog za učenje. Ovaj rad postiže točnost od preko 99.0% i pokazuje kako se dodavanjem konvolucijskog auto-ekodera model ubrzao sedam puta, pri učenju i

klasifikaciji.

Naposljetku, duboko učenje se može primijeniti na prvi korak, prikupljanje podataka, povećanjem skupa podataka generiranjem umjetnih podataka. Termin poboljšanje podataka (engl. *data augmentation*) koristi se za svaku metodu koja umjetno povećava izvorni skup podataka za treniranje koristeći transformacije koje čuvaju oznake (engl. *label-preserving transformations*) [17]. Posljednji rad, Shaker et al. [18], bavi se problemom balansiranja skupa podataka generiranjem umjetnih podataka koristeći generativne suparničke mreže (engl. *generative adversarial network, GAN*). Pojedini model GAN-a je naučen za svaki razred srčanog ritma, jer su količine podataka dostupne za pojedini razred disproporcionalne. Ukoliko bi se samo jedan GAN naučio na cijelom skupu podataka, fokusirao bi se na generiranje primjeraka razreda koji dominiraju u distribuciji podataka u skupu i zapostavio slabije zastupljenje razrede. Korišteni skup podataka dolazi iz baze podataka *MIT-BIH Arrhythmia Database* [11]. Ovaj zanimljiv pristup čestom problemu i njegov pozitivan utjecaj na rezultate također su motivirali ovaj diplomski rad. U radu [18] je korišten CNN koji postiže točnost od preko 98%, preciznost (engl. *precision*) od preko 90%, specifičnost od preko 97.4% i senzitivnost od preko 97.7%.

## 4. Skup podataka i metode

U ovom diplomskom radu pokušava se riješiti problem klasifikacije srčanih ritmova iz snimaka EKG-a koristeći kombinaciju pristupa dubokog učenja. Koristi se kombinacija konvolucijskog auto-inkodera i LSTM-klasifikatora predstavljena u [16], nadalje ćemo ovu kombinaciju nazivati dubokim CAE-LSTM modelom. Također se koristi pristup poboljšanja skupa podataka generiranjem umjetnih podataka koristeći GAN-ove predstavljen u [18] u usporedbi s pristupom generiranja umjetnih podataka korištenjem konvolucijskog auto-inkodera. Dodatno, ovaj rad se razlikuje od [16] u dva pogleda, korišten je drugačiji skup podataka i metoda predobrade podataka. Posljedično, razlika je i u rezultatima, detaljnije o tome u poglavlju 6. Za izradu ovog poglavlja opisa metoda većina je preuzeta iz [19] i [20].

### 4.1. Skup podataka

U ovom diplomskom radu se koristi drugačiji skup podataka od onog korištenog u [16] kako bi se ispitalo kako se predstavljeni pristupi dubokog učenja ponašaju na drugačijem skupu podataka. Korišteni skup podataka dobiven je iz kolekcije triju baza podataka javno dostupnih od strane natjecanja *PhysioNet/Computing in Cardiology Challenge 2020* na [21]: *CPSC Database*, *CPSC-Extra Database* i *The Georgia 12-lead ECG Challenge (G12EC) Database*. Iako skup podataka sadrži snimke svih 12 kanala, korištene su samo snimke s drugog (II) kanala. Svaka snimka EKG-a je uzorkovana frekvencijom uzorkovanja od 500 Hz i sadrži jedan od srčanih ritmova iz tablice 4.2. Nadalje, svaka snimka je prošla predobradu i segmentirana je na pojedinačne otkucaje srca. Nakon toga skup podataka je podijeljen u skup za učenje, koji sadrži 70% nasumično odabranih primjeraka, i skup za testiranje, koji sadrži ostalih 30% primjeraka. Distribucija skupa podataka po srčanim ritmovima je prikazana u tablici 4.1.

Srčani ritam	Broj snimaka	Broj otkucaja	Broj otkucaja u skupu za učenje
NSR	2670	37909	26537
LBBB	467	6872	4811
RBBB	2399	38593	27016
PAC	1852	20610	14427
VPB	387	4068	2848
<b>UKUPNO</b>	<b>7775</b>	<b>108052</b>	<b>75639</b>

**Tablica 4.1:** Distribucija izvornog skupa podataka

Svaki primjerak iz skupa podataka sastoji se od binarne datoteke *MATLAB v4* (.mat) i tekstne datoteke u obliku *WFDB header* (.hea). Binarne datoteke .mat sadrže snimku signala EKG-a, dok tekstne datoteke .hea sadrže opis snimke signala i značajke pacijenta. U tekstnom isječku 4.1 prikazan je primjerak zapisa iz datoteke *A0010.hea*:

```

E00410.mat 12 500 5000 05-May-2020 14:50:55
E00410.mat 16+24 1000/mV 16 0 -29 -23559 0 I
E00410.mat 16+24 1000/mV 16 0 0 -6104 0 II
E00410.mat 16+24 1000/mV 16 0 29 17318 0 III
E00410.mat 16+24 1000/mV 16 0 14 14517 0 aVR
E00410.mat 16+24 1000/mV 16 0 -29 -20058 0 aVL
E00410.mat 16+24 1000/mV 16 0 14 5573 0 aVF
E00410.mat 16+24 1000/mV 16 0 78 5363 0 V1
E00410.mat 16+24 1000/mV 16 0 136 -16711 0 V2
E00410.mat 16+24 1000/mV 16 0 87 7002 0 V3
E00410.mat 16+24 1000/mV 16 0 19 -22028 0 V4
E00410.mat 16+24 1000/mV 16 0 -39 4923 0 V5
E00410.mat 16+24 1000/mV 16 0 -39 -2070 0 V6
#Age: 66
#Sex: Male
#Dx: 164909002
#Rx: Unknown

```

#Hx: Unknown

**Tekstni isječak 4.1:** sadržaj datoteke A0010.he

Iz prvog retka zapisa očituje se sljedeće. Identifikacijska oznaka snimke je *E00410*, dakle datoteka *E00410.mat* odgovara ovoj datoteci *.hea*. Snimka je snimljena na EKG-u s 12 kanala. Frekvencija uzorkovanja iznosi 500 Hz, a snimka sadrži 5000 uzoraka, tj. signala. Naposljetku je zapisana vremenska oznaka snimanja.

Sljedećih 12 redaka odgovara svakom od 12 kanala EKG-a, a na kraju svakog retka zapisan je kanal kojem redak odgovara. Redak počinje nazivom odgovarajuće datoteke *.mat*. Zatim slijedi informacija o zapisu signala, gdje je vidljivo da je ovaj signal zapisan s pomoću 16 bitova s posmakom od 24 bita. Amplituda razlučivosti snimke iznosi 1000 jedinica po milivoltu (*mV*). Razlučivost analogno-digitalnog pretvarača iznosi 16 bita. Vrijednost stvarnog signala od 0 predstavljena je u zapisu kao vrijednost 0. Zatim je zapisana prva vrijednost signala, u slučaju kanala *III* iznosi 29. Nakon prve vrijednosti signala slijedi kontrolni zbroj, u slučaju kanala *II* iznosi -6104.

Iz zadnja tri retka datoteke *.hea* isčitava se sljedeće. Pacijent je šezdeset i šestoro godišnji muškarac, što se vidi iz redaka *Age* i *Sex*. Redak *Dx* označava dijagnosticirani srčani ritam pomoću šifriranog zapisa. Koristeći tablicu 4.2 otkrivamo da se radi o bloku lijeve grane snopa (LBBB). Za ovog pacijenta liječnički recept (*Rx*) i povijest bolesti (*Hx*) su nepoznati.

Ciljna oznaka u ovom klasifikacijskom problemu je dijagnoza pacijenta (*Dx*). Vrijednosti koje ta oznaka može sadržavati su u korištenom skupu podataka su sljedeće:

Srčani ritam	Kratica	Šifra
Normalni sinusni ritam	NSR	426783006
Blok lijeve grane snopa	LBBB	164909002
Blok desne grane snopa	RBBB	59118001
Preuranjena kontrakcija atrijske	PAC	284470004
Preuranjena kontrakcija ventrikula	VPB	17338001

**Tablica 4.2:** Moguće oznake u skupu podataka (stupac šifra) i njihovo značenje

## 4.2. Predobrada snimke EKG-a – Pan-Tompkinsov algoritam

Svaka snimka EKG-a iz skupa podataka prolazi predobradu gdje se prvo pomoću Pan-Tompkinsovog algoritma segmentira na pojedinačne otkucaje. Točnije, Pan-Tompkinsov algoritam pronalazi komplekse QRS, tj. locira R vrhove, te *reže* dijelove snimke veličine 260 uzoraka (0.52 s) svaki. Locirani R vrh koristi se kao orijentacijska točka te se nalazi na stotom uzorku segmenta. Segmenti s ubraznim srčanim ritmom, gdje je otkucaj srca brži od 0.52 s, često imaju devijaciju u vidu zahvaćanja prethodnog vala T ili idućeg vala P. Takvi segmenti nisu uklonjeni iz skupa podataka. Konačno, segmentirane snimke normiraju se na vrijednost između 0 i 1. Ova metoda segmentacije otkucaja srca razlikuje se od metode u [16], u tom radu otkucaji su segmentirani uz pomoć nekoliko kardiologa. Dalje u ovom potpoglavlju je detaljno predstavljen Pan-Tompkinsov algoritam.

Jiapu Pan i Willis J. Tompkins autori su znanstvenog rada pod nazivom “A Real-Time QRS Detection Algorithm” iz 1985. godine [8]. U radu je opisan algoritam za pouzdanu detekciju kompleksa QRS iz EKG-signala u stvarnom vremenu. Izvorno je ovaj algoritam napisan u strojnom jeziku i implementiran na mikroprocesoru.

Algoritam koristi tri linearne transformacije signala te jednu nelinearnu kako bi što bolje riješio problem šuma u signalima. Obrada signala sastoji se od filtriranja pojavnim filtrom, diferenciranja signala, uprosječivanja signala integratorom pomičnog prozora te kvadriranjem signala.

Pojasno propusni filtar smanjuje utjecaj šuma od strane mišića, interferencije T valova i lutajuće bazne linije. Propusni pojas poželjno je veličine 5 – 15 Hz. Izvorno je pojasni filtar implementiran kaskadom niskopropusnog i visokopropusnog filtra kako bi se dobio propusni pojas veličine oko 5 – 15 Hz. Nakon toga signal se diferencira kako bi se dobila informacija o nagibu kompleksa QRS. Tada se signal kvadrira što signal pretvara u potpuno pozitivan signal te naglašava vrhove. Zadnji korak obrade signala je integracija pomičnim prozorom. Svrha toga je dobivanje informacija o obliku valova te detekcija vrhova R valova. Širina prozora bi trebala biti otprilike široka kao i najširi kompleks QRS na snimci. Ako je prozor preuzak, za neke komplekse QRS će se prikazati više vrhova nego je potrebno. Ako je prozor preširok, oblik koji integracija provede će obuhvaćati i T valove u QRS komplekse.

Iz prikaza generiranog integracijom pomičnim prozorom očituju se oblici specifični za taj prikaz. Rastući brid tog oblika odgovara kompleksu QRS, tj. trajanje ras-



tućeg brida oblika jednako je širini kompleksa QRS. Vrh tog oblika odgovara vrhu R vala. Sljedeći korak algoritma je postavljanje granica kako bi se što točnije detektirali kompleksi QRS na snimci.

Određuju se dva skupa granica, jedan koji će se primjeniti na snimci generiranoj integracijom pomičnim prozorom, drugi koji će se primjeniti na snimci filtriranoj pojasnim filtrom. Svaki skup sastoji se od sljedeće 4 granice: granica za vrh signala, granica za vrh šuma, viša granica te niža granica. Viša granica određena je vrijednostima granice za vrh signala i granice za vrh šuma. Kada algoritam naiđe na vrh, točka koja ima veću vrijednost od prethodne i nadolazeće točke uspoređuje njegovu vrijednost sa snimke sa višom granicom. Ako je vrijednost vrha veća od više granice, točka se klasificira kao vrh. To potiče preračunavanje granice za vrh signala. Ako unutar određenog vremenskog intervala nije klasificiran niti jedan vrh, taj se interval ponovno pretražuje, no ovaj se put koristi niža granica koja je upola manja od više granice. Kako bi se identificirao kompleks QRS, njegov vrh mora biti klasificiran u obje snimke. Vrlo važno je vremensko trajanje oporavka srca, točnije 200 ms. Dakle, vremenska udaljenost dvaju R vrhova mora biti barem veća od 200 ms.

Pored svega navedenoga, algoritam također održava dvije prosječne vrijednosti R-R intervala. Jedna je prosječna vrijednost osam zadnje viđenih R-R intervala, otkucaja. Druga je prosječna vrijednost osam zadnje viđenih R-R intervala koji pripadaju određenom vrijednosnom intervalu. Razlog ovoj tehnici je mogućnost brzog prilagođavanja nepravilnim frekvencijama otkucaja srca. Ukoliko su za zadnjih osam otkucaja obje prosječne vrijednosti R-R intervala iste, radi se o pravilnom srčanom ritmu.

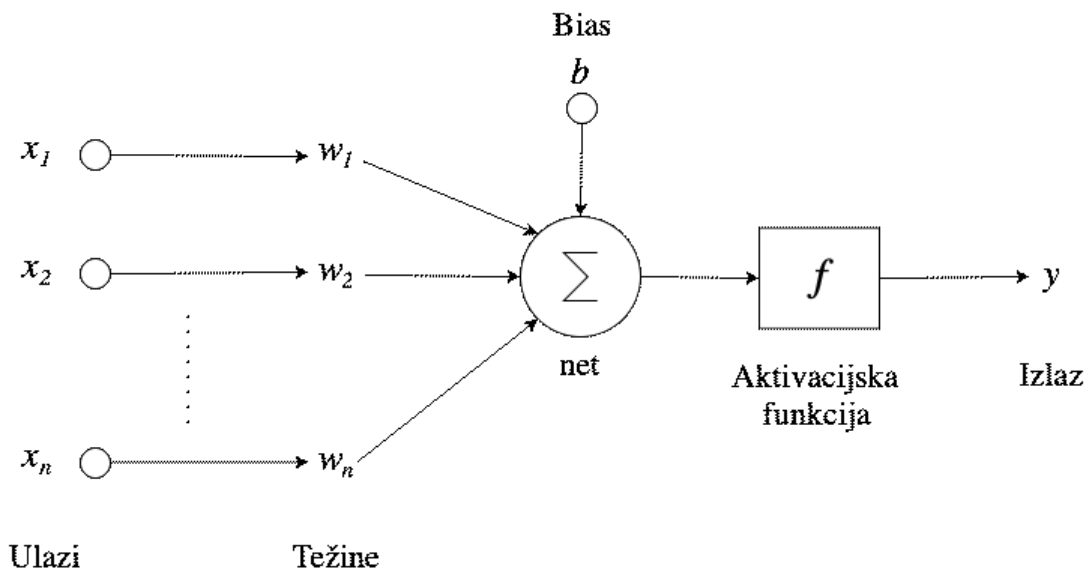
Također, algoritam ima sigurnosnu provjeru kod klasifikacije kompleksa QRS. Ukoliko je interval R-R kraći od 360 ms, ali naravno dulji od 200 ms, procjenjuje se radi li se o T valu ili o kompleksu QRS. Ukoliko je najveći nagib oblika u prikazu signala integracijom pomičnog prozora manji od polovice prijašnjeg, oblik se klasificira kao val T. U suprotnom klasificira se kao kompleks QRS.

### **4.3. Duboko učenje**

Duboko učenje spada u polje umjetne inteligencije pod nazivom strojno učenje. Zasniva se na dubokim umjetnim neuronskim mrežama. Umjetne neuronske mreže sastoje se od slojeva neurona, ulaznih, izlaznih i skrivenih slojeva. Duboke neuronske mreže imaju više skrivenih slojeva nego obične neuronske mreže.

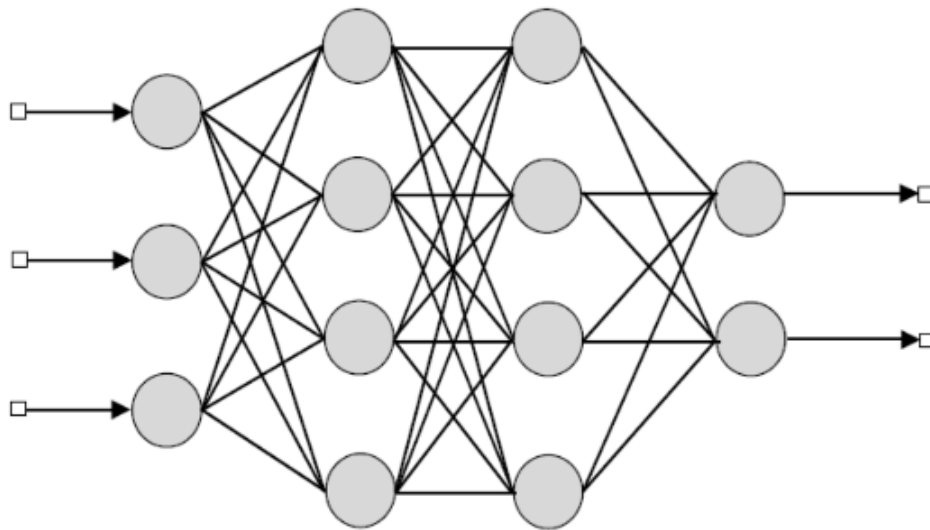
Temeljne sastavnice svake umjetne neuronske mreže su neuroni. Nastali po uzoru na model ljudskog mozga, neuroni se u umjetnoj neuronskoj mreži raspoređuju u slo-

jeve i međusobno su povezani vezama kojima se dodjeljuje težina. Na slici 4.1 prikazan je osnovni model neurona. Sastoji se od ulaza  $x_1, x_2 \dots x_n$  i odgovarajućih težina  $w_1, w_2 \dots w_n$  koje određuju u kojoj mjeri određeni ulazi *pobuđuju* neuron. Umnošci odgovarajućih ulaza i težina se sumiraju u sumu koja se često naziva *net*. Sumi se često dodaje jedna težina bez odgovarajućeg ulaza, naziva pristranost ili *bias*. *Net* se koristi kao argument u aktivacijskoj funkciji koja šalje na izlaz rezultat obrade vrijednost *net*. Postupkom učenja optimiraju se težine u neuronima.



Slika 4.1: Model neurona u umjetnoj neuronskoj mreži

Postoje različite strukture umjetnih neuronskih mreža, a temeljna je struktura unaprijedna, potpuna povezana neuronska mreža (engl. *feedforward network*). Ova mreža najčešće se koristi za klasifikaciju. Ona se sastoji od jednog ulaznog sloja čiji broj neurona odgovara dimenziji ulaznih podataka, jednog izlaznog sloja čiji broj neurona odgovara broju mogućih klasa te jednog ili više skrivenih slojeva koji obrađuju podatke dobivene iz prethodnog sloja i šalju obrađene podatke u sljedeći sloj. Pritom se koristi potpuna povezanost, što znači da je svaki neuron idućeg sloja povezan težinama sa svim neuronima prethodnog sloja. Unaprijedna mreža nema ciklusa. Ovakva struktura također ima ime višeslojni perceptron (engl. *multilayer perceptron*).



Slika 4.2: Unaprijedna mreža s potpuno povezanim slojevima [22]

### 4.3.1. Konvolucijske mreže

Konvolucijska mreža (engl. *Convolutional neural network, CNN*) vrsta je neuronske mreže koja je specijalizirana za obradu podataka čija topologija sličí rešetki (engl. *grid-like topology*). Takvi podaci mogu biti jednodimenzionalni, primjerice vremenski slijedovi. Također mogu biti dvodimenzionalni kao slike. Snimke EKG-a su 1D podaci, stoga je u ovom dijelu objašnjenja 1D konvolucijska mreža kao primjer. Konvolucijske mreže su neuronske mreže koje sadrže barem jedan konvolucijski sloj. U takvom sloju koristi se matematička operacija konvolucije, odakle dolazi i naziv sloja. Konvolucija je operacija koja koristeći dvije funkcije, vraća preoblikovanu verziju prve funkcije s obzirom na preklapanje s drugom funkcijom. U strojnom učenju operacija konvolucije jednaka je operaciji unakrsne korelacije.

$$S(t) = (w * x)(t) = \int w(a)x(t + a)da \quad (4.1)$$

U praksi su učestaliji diskretni podaci, tada se koristi sljedeći oblik unakrsne korelacije:

$$S(t) = (w * x) = \sum_{a=a_{min}}^{a_{max}} w(a)x(t + a) \quad (4.2)$$

Funkcija  $x$  je ulaz, a funkcija  $w$  je jezgra. Izlazi dobiveni iz neurona u konvolucijskom sloju tvore mapu značajki,  $S$  predstavlja mapu značajki. Dakle u slučaju 1D

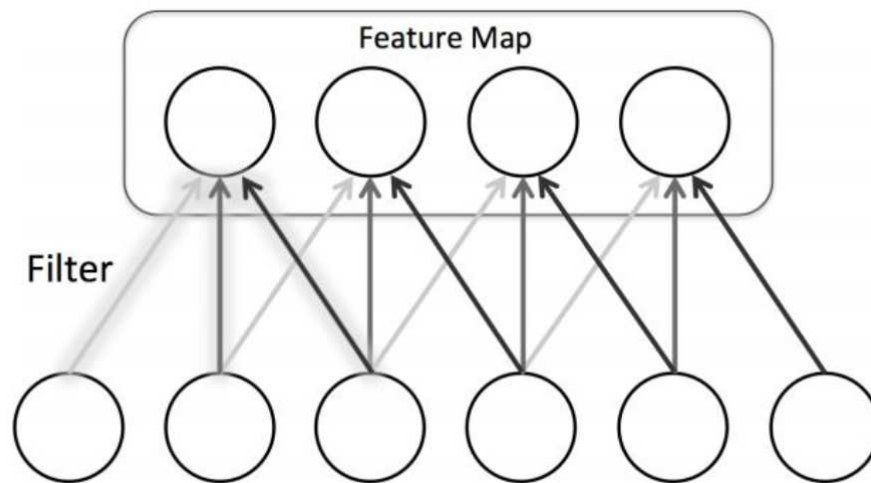
konvolucijske mreže  $x$  predstavlja ulaz u obliku vremenskog slijeda, npr. signal EKG-a zapisan u vremenu  $T$ . Jezgra  $w$  može se zamisliti kao filtar duljine  $t$  koji se koristi kako bi se njime prešlo u  $N$  koraka preko čitavog signala.

$$t = a_{max} - a_{min} \quad (4.3)$$

, konvolucijski sloj će imati  $N$  neurona gdje će svaki biti povezan s  $t$  ulaznih neurona. Slika 4.3 prikazuje kako je konvolucijski sloj povezan s ulaznim slojem.

Iz dosadašnjeg objašnjenja i slike 4.3 vidi se razlika između konvolucijskog sloja i potpuno povezanog sloja unaprijedne mreže. Kao prvo, neuroni konvolucijskog sloja nisu povezani sa svim neuronima prethodnog sloja. Također, konvolucijski sloj ostvaruje svojstvo dijeljenja parametara jer više veza među neuronima ima jednaku težinu. Svaki neuron ima aktivacijsku funkciju, pa tako imaju i neuroni konvolucijskog sloja. Najčešće korištena aktivacijska funkcija neurona konvolucijskog sloja je aktivacijska funkcija zglobnice – ReLu (engl. *Rectified Linear Unit*). Ona je definirana kao:

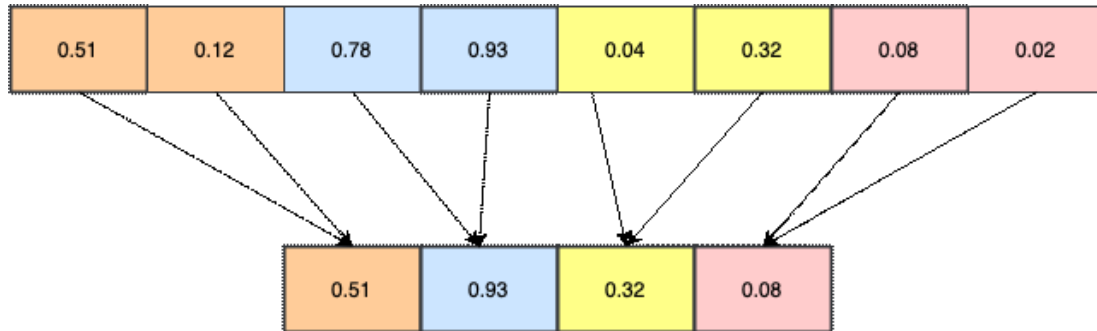
$$ReLU(x) = \max(0, x) \quad (4.4)$$



**Slika 4.3:** Ulazni sloj povezan s 1D konvolucijskim slojem [23]

Nakon konvolucijskog sloja uobičajeno dolazi sloj sažimanja (engl. *pooling layer*). Sloj sažimanja služi za smanjenje dimenzije mape značajki dobivene iz konvolucijskog sloja uz što manji gubitak informacija. Također, sloj sažimanja povećava invarijantnost modela na pomak. Različite funkcije sažimanja mogu biti implementirane u sloj sažimanja, no najčešće se koristi funkcija sažimanja maksimalnom vrijednošću

(engl. *max-pooling*). Ta funkcija se primjenjuje na područje mape značajki, pri čemu se uzima nekoliko susjeda i zamjenjuje se to područje s najvećom vrijednošću iz istog područja. Funkcija sažimanja najčešće se primjenjuje bez preklapanja područja. Na slici 4.4 prikazano je sažimanje izlaza 1D konvolucijskog sloja.



**Slika 4.4:** Sloj sažimanja maksimalnom vrijednošću nakon 1D konvolucijskog sloja

Konvolucijska mreža na svoj ulaz prima podatke s *grid-like* topologijom, zatim konvolucijski sloj obrađuje taj ulaz koristeći filter tako da operacijom konvlucije, tj. unakrsne korelacije, na svom izlazu stvori mapu značajki. Ta se mapa značajki dalje sažme u sloju sažimanja. Nakon nekoliko konvolucijskih slojeva i slojeva sažimanja, u konvolucijskoj mreži dolazi se do potpuno povezanih slojeva koji obavljaju klasifikaciju koristeći značajke dobivene iz zadnjeg sloja sažimanja.

### 4.3.2. Konvolucijski auto-ekoder

Auto-ekoder je vrsta duboke neuronske mreže koja se uči kako bi kopirala svoj ulaz na svoj izlaz. Arhitektura mreže se može promatrati kao spoj dvaju dijelova:

1. koderske funkcije koja uzima ulaz  $x$  i stvara *kod*  $h$  koji predstavlja ulaz

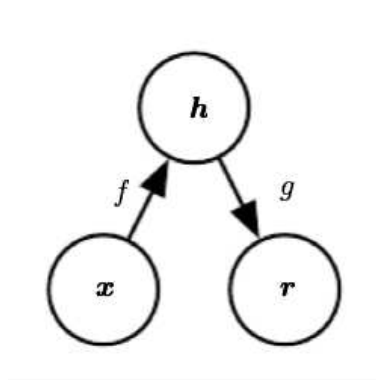
$$h = f(x) \quad (4.5)$$

2. dekodera, koji stvara rekonstrukciju  $r$  iz *koda*

$$r = g(h) \quad (4.6)$$

Slika 4.5 simbolično prikazuje arhitekturu auto-ekodera. Kada bi auto-ekoder uspio savršeno naučiti  $g(f(x)) = x$  za svaki  $x$  ne bi bio naročito koristan. Umjesto toga, auto-ekoderi su dizajnirani kako nebi mogli naučiti savršeno kopirati ulaz na izlaz. Ograničeni su kako bi mogli samo približno precizno kopirati i to ulaze koji nalikuju podacima koji su mu predstavljeni u fazi učenja. Kako je tada model primoran

prioretizirati koje značajke ulaza kopirati često uči korisne osobine i značajke podataka. Stoga moderni auto-ekoderi generaliziraju ideja *enkodera* i *dekodera* i predstavljaju determinističke funkcije 4.5 i 4.6 kao stohastička mapiranja  $p_{enkoder}(\mathbf{h}|\mathbf{x})$  i  $p_{dekoder}(\mathbf{x}|\mathbf{h})$ . Ta mapiranja, tj. ekoder i dekoder, u praksi su implementirani kao slojevi duboke neuronske mreže. Auto-ekoderi se uspješno koriste u smanjivanju dimenzionalnosti (engl. *dimensionality reduction*) i dohvat u informacija (engl. *information retrieval*).



**Slika 4.5:** Simbolično prikazana auto-ekoderska arhitektura [19].

Konvolucijski auto-ekoder (engl. *convolutional autoencoder*, *CAE*) vrsta je auto-ekodera koja u ekoderu i dekoderu koristi konvolucijske slojeve koji se uče nenadziranim učenjem. CNN-ovi i CAE-ovi koriste konvolucijske slojeve, ali CAE-ovi se ne uče za klasifikaciju ili detekciju. Cilj faze učenja kod CAE-a je minimizirati *rekonstrukcijsku pogrešku* (engl. *reconstruction error*) koja predstavlja razliku između rekonstruiranog i izvornog primjerka.

### 4.3.3. Povratne mreže

Povratne neuronske mreže (engl. *Recurrent Neural Networks*, *RNN*) obitelj su neuronskih mreža za obradu podataka sa strukturom slijeda (engl. *sequential data*). Kako su konvolucijske mreže specijalizirane za obrađivanje podataka s topologijom rešetke kao što su slike, tako su povratne mreže specijalizirane za obrađivanje slijedova vrijednosti  $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(\tau)}$ . Većina povratnih mreža također može obrađivati slijedove promjenjive duljine. Ključni dio dizajna povratnih mreža su dijeljeni parametri. Dijeljenje parametara omogućuje obrađivanje slijedova različite duljine. Pojednostavljeno možemo reći da povratne mreže obrađuju slijed koji sadrži vektore  $\mathbf{x}^{(t)}$  gdje se indeks vremenskog koraka  $t$  kreće od 1 do  $\tau$ . Indeks vremenskog koraka se ne mora nužno odnositi na prolazak vremena u stvarnom svijetu, već samo na položaj vrijednosti u slijedu.

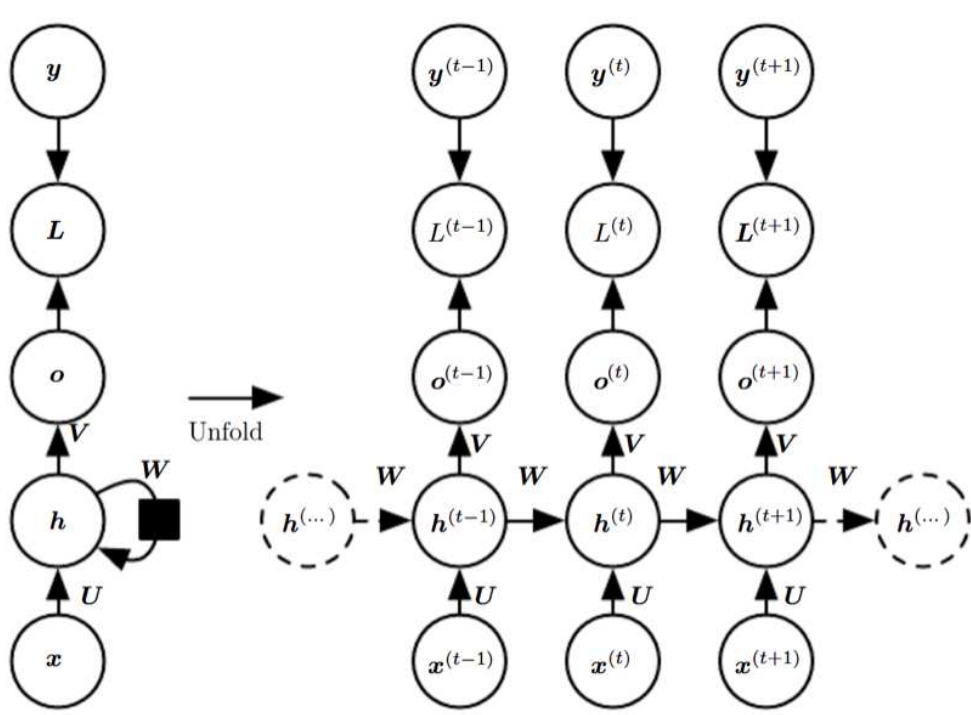
Slika 4.6 (desno) predstavlja računski graf (engl. *computational graph*) za računanje gubitka tijekom učenja povratne mreže koja preslikava ulazni slijed  $x$  na odgovarajući izlazni slijed  $o$ . Gubitak  $L$  mjeri razliku (udaljenost) između članova slijeda  $o$  i odgovarajućih članova slijeda oznaka  $y$ . Parametri povratne mreže nalaze se u trima matricama težina:  $U$ ,  $W$  i  $V$ . Matrica  $U$  sadrži težine s kojima se množi ulaz  $x^{(t)}$ , težine u matrici  $W$  odnose se na prethodno skriveno stanje  $h^{(t-1)}$ , a matrica  $V$  sadrži težine s kojima se množi skriveno stanje  $h^{(t)}$  pri računanju izlaza  $o^{(t)}$ . Uz navedene matrice parametri se nalaze i u vektorima pristranosti  $b_h$  koji se koristi pri računanju skrivenog stanja i  $b_o$  koji se koristi za računanje izlaza. U praksi se za nelinearnost pri ažuriranju skrivenog stanja najčešće koristi tangens hiperbolni (tanh), no dobre alternative su sigmoida i zglobnica. Dakle, povratna mreža je definirana sljedećim jednadžbama:

1. Ažuriranje skrivenog stanja

$$h^{(t)} = \tanh(W h^{(t-1)} + U x^{(t)} + b_h) \quad (4.7)$$

2. Projekcija u izlazni sloj

$$o^{(t)} = V h^{(t)} + b_o \quad (4.8)$$



**Slika 4.6:** Vizualizacija povratne neuronske mreže pomoću računskog grafa (engl. *computational graph*) odmotanog kroz vremensku dimenziju, svaki čvor odgovara jednom vremenskom indeksu [19].

Duboka povratna neuronska mreža je povratna neuronska mreža koja ima više skrivenih slojeva (slojeva u kojima se računaju skrivena stanja  $\mathbf{h}$ ), tj. ima vertikalnu dubinu veću od jedan. U slučaju duboke povratne neuronske mreže, jednadžbe koje ju definiraju izgledaju ovako:

1. Ažuriranje skrivenog stanja

$$h_1^{(t)} = \tanh(W_1 h_1^{(t-1)} + U_1 x^{(t)} + b_{h1}), n = 1 \quad (4.9)$$

$$h_n^{(t)} = \tanh(W_n h_n^{(t-1)} + U_n x^{(t)} + b_{hn}), n > 1 \quad (4.10)$$

2. Projekcija u izlazni sloj

$$o^{(t)} = V h_n^{(t)} + b_o \quad (4.11)$$

Također, propagacija unatrag (engl. *backpropagation*) primjenjena na povratne neuronske mreže zove propagacija unatrag kroz vrijeme (engl. *backpropagation through time*, *BPTT*). BPTT konceptualno nije različit od obične propagacije unatrag, samo ima dodatni horizontalni smjer propagacije unatrag kroz skriveni sloj.

Propagacija unatrag u horizontalnom smjeru uzrokuje akumulaciju gradijenata na parametre u matrici  $\mathbf{W}$  i čini optimizaciju jako nestabilnom. Uzrok toga je dijeljenje parametara kroz veliki broj uzastopnih slojeva, tj. uzastopno množenje matricom  $\mathbf{W}$ . Taj problem pri propagaciji unatrag često uzrokuje eksplodirajući i nestajući gradijent.

#### 4.3.4. Mreža duge kratkoročne memorije – LSTM

Na kraju prethodnog potpoglavlja predstavljen je glavni problem kod učenja povratnih neuronskih mreža, no što bi bilo poželjno poboljšati kako bi se taj problem riješio? Poželjno bi bilo rasteretiti parametre povratne neuronske mreže, matrice  $\mathbf{U}$  i  $\mathbf{W}$  imaju teške zadatke kao što su filtriranje nepotrebnih informacija, pamćenje dosad viđenih ulaza i projekcija novih elemenata ulaza u skriveno stanje. Dakle, skriveno stanje  $\mathbf{h}$  služi za izračun izlaza i pamćenje informacija. Kao rješenje predstavlja se ćelija s dugom kratkoročnom memorijom (engl. *Long Short-Term Memory*, *LSTM*).

Povratna mreža LSTM spada u vrstu povratnih neuronskih mreža s propusnicama (engl. *gated RNNs*). LSTM povratna mreža sastoji se od *LSTM-ćelija* koje imaju iste ulaze i izlaze kao i u običnom RNN-u, ali imaju više parametara i sustav propusnica koje kontroliraju protok informacija. LSTM uvodi nove komponente, za ćeliju  $i$  i vremenski indeks  $t$ : vektor stanja ćelije  $s_i^{(t)}$  (engl. *state*), propusnicu zaboravljanja  $f_i^{(t)}$



(engl. *forget gate*) i propusnicu novog ulaza  $g_i^{(t)}$  (engl. *input gate*). Slika 4.7 prikazuje arhitekturu jedne LSTM-ćelije sa svim komponentama. Bitno je naglasiti kako svaka propusnica ima svoje parametre u obliku odgovarajućih matrica  $W$ ,  $U$  i vektora pristranosti  $b$ .

Propusnica zaboravljanja  $f_i^{(t)}$  služi za *zaboravljanje* određenog dijela informacija iz prošlog skrivenog stanja  $h^{(t-1)}$ .

$$f_i^{(t)} = \sigma(W^f h^{(t-1)} + U^f x(t) + b^f) \quad (4.12)$$

Propusnica ulaza  $g_i^{(t)}$  služi za propuštanje određenog podskupa informacija iz ulaza  $x^{(t)}$ .

$$g_i^{(t)} = \sigma(W^g h^{(t-1)} + U^g x(t) + b^g) \quad (4.13)$$

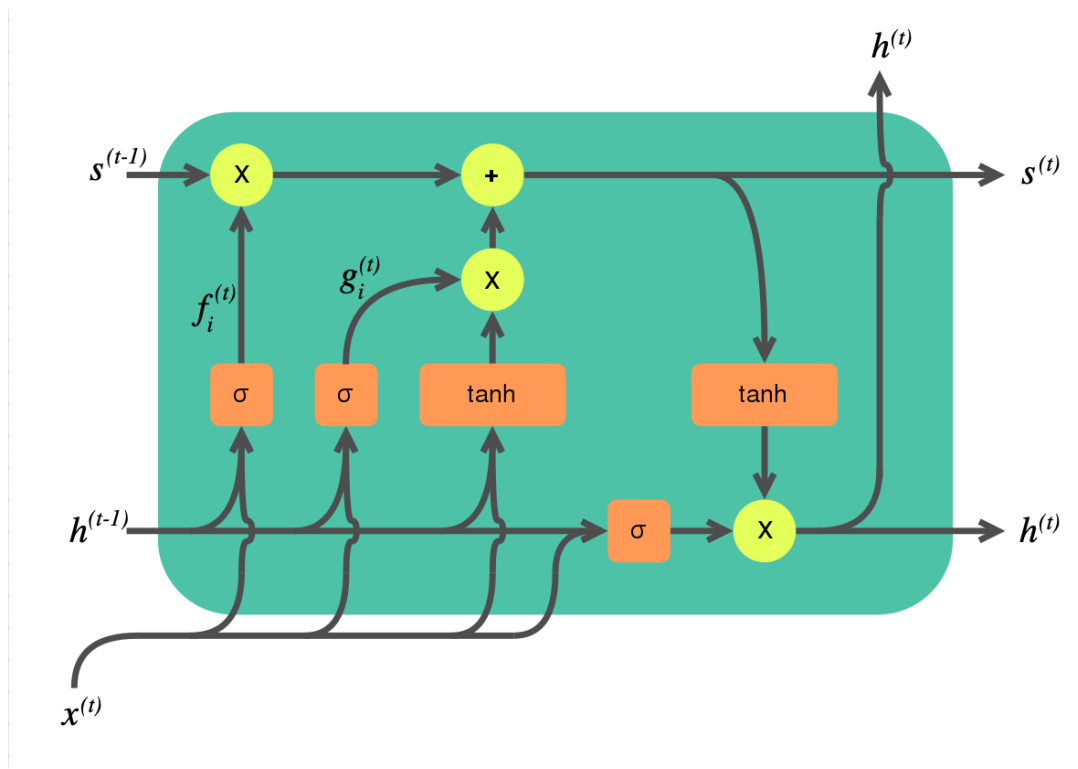
Propusnice dakle filtriraju informacije koje dolaze u ćeliju iz prethodne ćelije i s ulaza. Vektor stanja  $s_i^{(t)}$  se onda ažurira koristeći filtrirane informacije prema sljedećoj formuli:

$$s_i^{(t)} = f_i^{(t)} s_i^{(t-1)} + g_i^{(t)} \tanh(W h^{(t-1)} + U x^{(t)} + b_h) \quad (4.14)$$

Ažurirani vektor stanja  $s_i^{(t)}$  se tada koristi uz prošlo skriveno stanje  $h^{(t-1)}$  i ulaza  $x^{(t)}$  kako bi se dobilo trenutno skriveno stanje za ćeliju prema sljedećoj formuli:

$$h_i(t) = \tanh(s_i^{(t)}) \sigma(W h^{(t-1)} + U x^{(t)} + b_o) \quad (4.15)$$

Zbog ovako definirane strukture LSTM ćelije problem nestajućeg gradijenta je bitno smanjen, dok je eksplodirajući gradijent i dalje moguć kod unatražnog prolaza vezanog za  $h^{(t)}$ , no mnogo se rijede pojavljuje. Također je pokazano kako LSTM-mreže lakše uče dugoročne ovisnosti u usporedbi s jednostavnijim povratnim arhitekturama.



Slika 4.7: Struktura LSTM ćelije

### 4.3.5. Generativna suparnička mreža – GAN

Konvolucijske i povratne neuronske mreže najčešće koristimo za zadatak klasifikacije, tj. učimo preslikavanja primjerka iz skupa podataka na oznake. Ovakav tip učenja i modela naziva se diskriminativnim, npr. prepoznavanje radi li se o mački ili psu na danoj slici. Uz diskriminativne modele postoje i generativni modeli. Njihova zadaća je naučiti približnu distribuciju podataka iz skupa podataka za učenje te moći uspješno generirati umjetne primjerke podataka za koje se može vjerovati da dolaze iz istog skupa kao i oni za učenje.

Generativna suparnička mreža (engl. *Generative adversarial network, GAN*) moćan je razred generativnih modela koji su 2014. predstavili Goodfellow et al. u radu *Generative Adversarial Nets* [24]. Osnovni princip GAN-a je motiviran igrom nultog zbroja (engl. *zero-sum game*) s dva igrača gdje je ukupna dobit dva igrača jednaka nuli i dobit ili gubitak svakog igrača je uravnotežen dobiti ili gubitkom drugog igrača. GAN se sastoji od *generatora* i *diskriminatora* koji se istovremeno uče, to su dva igrača ove igre nultog zbroja. Generator nastoji naučiti potencijalnu distribuciju stvarnih podataka i stvara nove primjerke podataka. Diskriminator, koji je često binarni klasifikator, nastoji što točnije razlučiti između stvarnih i umjetno generiranih primjeraka podataka.

Postupak optimizacije GAN-a je *postupak minimax* kojem je cilj doći do trenutka kada se može smatrati da je generator naučio distribuciju stvarnih podataka, tj. postigao se Nashov ekvilibrij (engl. *Nash equilibrium*) [25].

Opišimo postupak učenja i optimizacije malo detaljnije. Diskriminator  $D$  je binarni klasifikator koji nastoji odlučiti je li primjerak  $x$  na njegovom ulazu stvarni podatak ili umjetno generirani podatak  $G(z)$ . Diskriminator na izlazu daje skalarnu predikciju  $o \in \mathbb{R}$  za ulaz  $x$  te primjenjuje sigmoidnu funkciju kako bi dobio vjerojatnost  $D(x) = 1/(1 + e^{-o})$ . Pretpostavimo da je oznaka  $y$  za stvarni podatak 1, a za umjetno generirani 0. Tada učimo diskriminator kako bi minimizirao gubitak binarne unakrsne entropije (engl. *binary cross-entropy loss, BCE*):

$$\min_D \{-y \log D(x) - (1 - y) \log (1 - D(x))\} \quad (4.16)$$

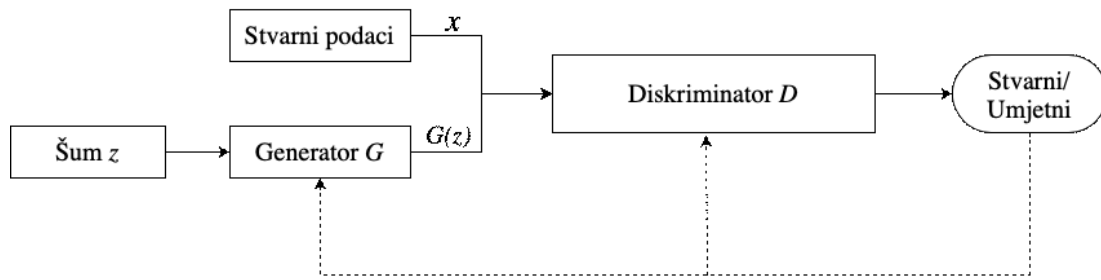
Generator na svoj ulaz dobiva vektor šuma  $z$ , tj. vektor nasumičnih vrijednosti iz normalne distribucije  $z \sim \mathcal{N}(0, 1)$ . Tada generator primjenjuje funkciju, tj. niz nelinearnih transformacija,  $G(z)$ . Cilj generatora je prevariti diskriminatora kako bi se  $G(z)$  klasificirao kao stvarni podatak, tj. generator želi  $D(G(z)) \approx 1$ . Može se dakle zaključiti da za diskriminator  $D$ , generator  $G$  ažurira svoje parametre kako bi maksimizirao gubitak unakrsne entropije kada je  $y = 0$ :

$$\max_G \{-(1 - y) \log (1 - D(G(z)))\} = \max_G \{-\log (1 - D(G(z)))\} \quad (4.17)$$

Konačno možemo prikazati postupak optimizacije i učenja GAN-a kao igru *minimax* između generatora  $G$  i diskriminatora  $D$ :

$$\min_D \max_G \{-E_{x \sim Dataset} \log D(x) - E_{z \sim Noise} \log(1 - D(G(z)))\} \quad (4.18)$$

Slika 4.8 prikazuje strukturu i postupak učenja GAN-a. Za kraj je bitno naglasiti kako generator  $G$  i diskriminator  $D$  mogu biti ostvareni kao bilo koja duboka neuronska mreža koja ima odgovarajuće dimenzije ulaza i izlaza.



**Slika 4.8:** Struktura i postupak učenja GAN-a. (*Pune strelice*) unaprijedni prolaz, (*iscrtane strelice*) prolaz unatrag.

## 5. Izvedba

U prethodnom poglavlju opisana je teoretska strana Pan-Tompkinsovog algoritma i metoda dubokog učenja koje su korištene u ovom diplomskom radu. U ovom je poglavlju opisana detaljna izvedba Pan-Tompkinsovog algoritma, konvolucijskog autoenkodera, generativne suparničke mreže, LSTM-a te postupka poboljšavanja skupa podataka. Opisane su arhitekture modela dubokog učenja, hiperparametri i grafovi gubitka tijekom faze učenja. Također je prikazan rad implementiranih modela uz primjerke njihovih izlaza i opisani su svi potrebni detalji izvedbe. Sve izvedbe su pisane u programskom jeziku *Python* [26] kotisteći biblioteku za duboko učenje *Pytorch* [27] i biblioteku za vizualizaciju *matplotlib* [28].

### 5.1. Pan Tompkinsov algoritam

Izvedba Pan-Tompkinsovog algoritma nalazi se u modulu *Pan\_Tompkins.py*, točnije u funkciji *detect\_qrs*. Funkcija od parametara prima snimku signala EKG-a, frekvenciju uzorkovanja i zastavicu za grafički prikaz podataka čija je pretpostavljena vrijednost *False*.

```
detect_qrs(ecg, fs, plot=False)
```

Pojasni filtar ostvaruje se korištenjem Butterworthovog pojasnog filtra prvog reda pojasa 5–15 Hz. Poziva se funkcija *bandpass\_filter*, prikazano u isječku koda 5.1, implementirana u modulu *Butterworth.py*, implementacija je prikazana u isječku koda 5.2.

```
from ..Butterworth import bandpass_filter
ecg_filtered = bandpass_filter(ecg, filter_lowcut,
                              filter_highcut, fs, filter_order)
```

**Isječak koda 5.1:** poziv metode *bandpass\_filter*

```

from scipy.signal import butter, lfilter
def bandpass_filter(data, lowcut, highcut,
                    fs, filter_order):
    nyquist_freq = 0.5 * fs
    low = lowcut / nyquist_freq
    high = highcut / nyquist_freq
    b, a = butter(filter_order, [low, high],
                  btype="band")
    y = lfilter(b, a, data)
    return y

```

**Isječak koda 5.2:** implementacija metode *bandpass\_filter* iz modula *Butterworth.py*

Ostale faze obrade signala opisane u trećem poglavlju odvijaju se jednako, prikazano je u isječku koda 5.3. Svaki signal se naknadno i normira.

```

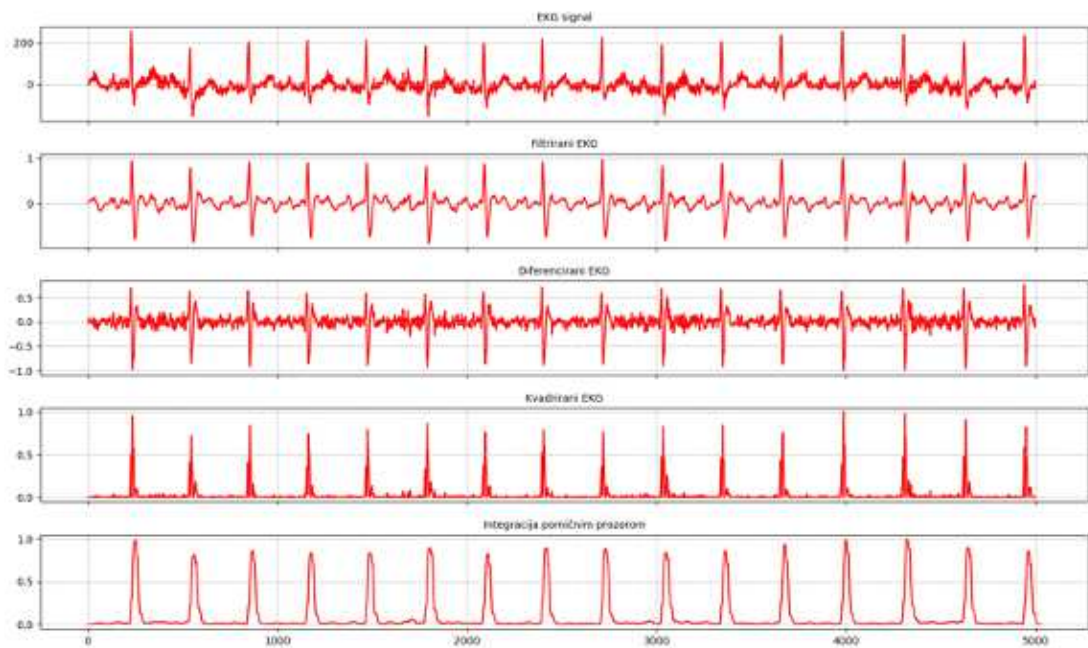
ecg_differentiated = np.ediff1d(ecg_filtered)
ecg_differentiated /= np.max(ecg_differentiated)

ecg_squared = ecg_differentiated ** 2
ecg_squared /= np.max(ecg_squared)

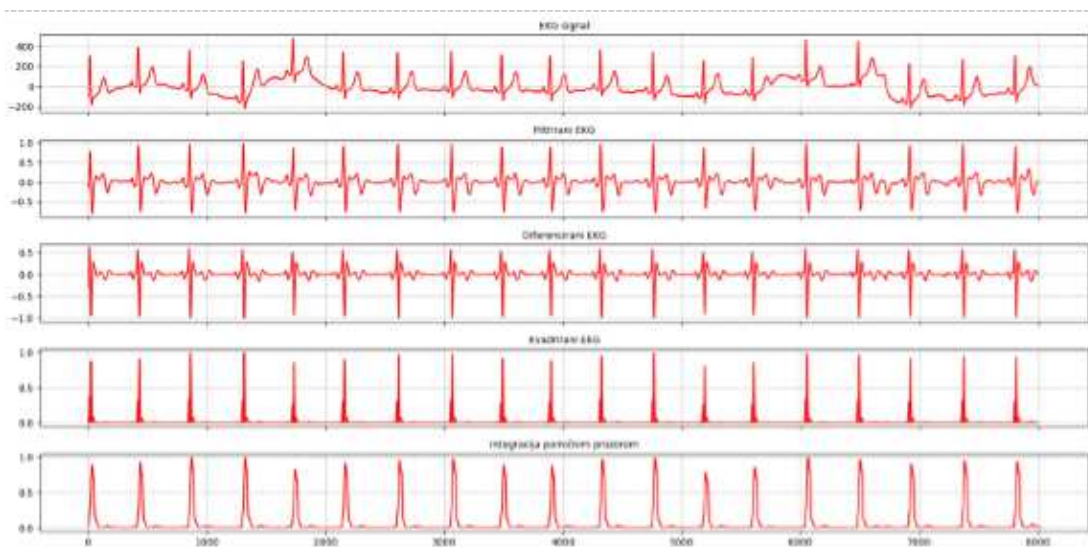
ecg_integrated = np.convolve(ecg_squared,
                             np.ones(integration_window) / integration_window)
ecg_integrated = ecg_integrated / np.max(ecg_integrated)

```

**Isječak koda 5.3:** faza obrade signala nakon filtriranja pojasnim filtrom



**Slika 5.1:** Transformacija izvornog signala u četiri koraka Pan-Tompkinsovog algoritma



**Slika 5.2:** Uklanjanje lutajuće bazne linije filtriranjem signala i ostala tri koraka transformacije

Tada se poziva funkcija  $find\_peaks(ecg, spacing, peak\_limit)$  koja pronalazi vrhove oblika dobivenih integracijom pomičnim prozorom. Pronalazi ih tako da prolazi signalom  $ecg$  vrijednost po vrijednost. Kada pronađe točku na signalu takvu da joj je vrijednost veća od prethodne i manja od sljedeće i veća od vrijednosti  $peak\_limit$  kandidira ju za vrh. Tada se pregledavaju sljedećih vrhovi u udaljenosti od  $spacing$  od kandidirane točke. Ako se pronađe točka koja zadovoljava iste uvjete, ali ima veću

vrijednost, ona postaje kandidat. Prije nove iteracije trenutni kandidat se sprema u polje vrhova, a iterator se povećava za *spacing*. Funkcija vraća polje vrhova *peak\_indices*. Ukoliko vraćeno polje nema dovoljno elemenata, ponovno se poziva funkcija ali ovoga puta s upola manjom vrijednosti *peak\_limit*.

Sljedeći korak je odluka o svakom vrhu u polju vrhova. Stvara se dodatno polje koje sadrži vrijednosti vrhova iz polja vrhova. Inicijaliziraju se granice: granica vrha signala, granica vrha šuma i viša granica. Niža granica nije korištena jer se njoj posao većinom riješio u prethodnom koraku. Iterira se u isto vrijeme po obje liste s informacijama o vrhovima pomoću ugrađene funkcije *zip*. Za svaki vrh provjerava se je li se već neki vrh detektirao kao kompleks QRS, a ako jest, je li njihov razmak veći od vremena oporavka srca. Ako su ovi uvjeti zadovoljeni ispituje se je li vrijednost vrha veća od gornje granice i time se određuje radi li se o kompleksu QRS ili o šumu. Na posljetku se označava početak, kraj i vrh kompleksa QRS koristeći polje detektiranih vrhova kompleksa QRS i signal dobiven integracijom pomičnog prozora. Implementacija ovog postupka prikazana je u isječku koda 5.4, a na slici 5.3 prikazan je rezultat algoritma, tj. demarkacija kompleksa QRS.

```
detected_qrs = np.array([], dtype=int)
detected_noise = np.array([], dtype=int)

for ind, value in zip(peak_indices, peak_values):
    if detected_qrs.size == 0 or
        ind - detected_qrs[-1] > refractory_period:

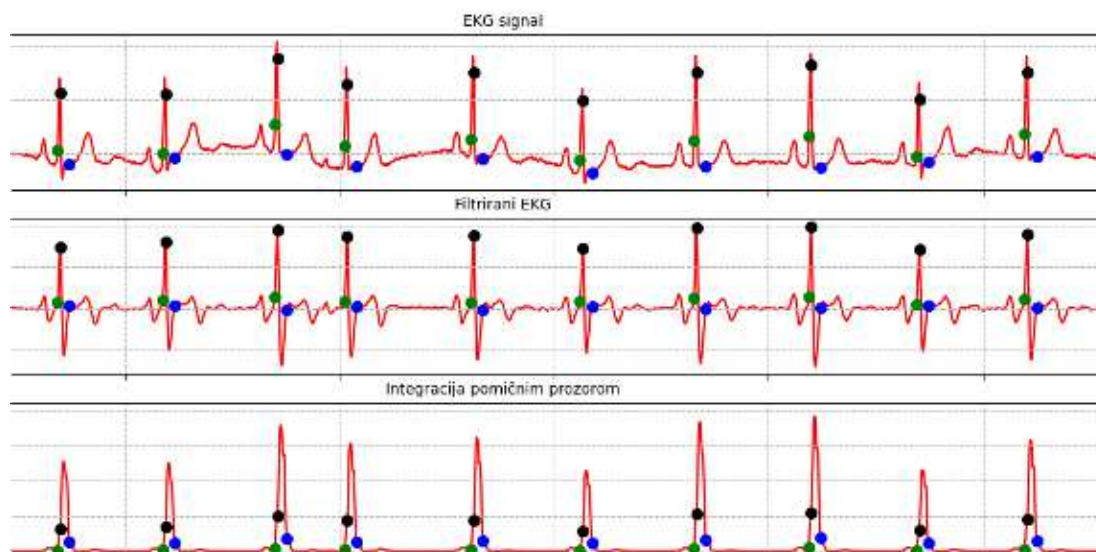
        if value > threshold_1:
            detected_qrs = np.append(detected_qrs, ind)
            signal_peak = 0.125 * value + 0.875 * _peak

        else:
            detected_noise = np.append(detected_noise,
                                       ind)
            noise_peak = 0.125*value + 0.875*noise_peak

    threshold_1 = noise_peak +
        0.25 * (signal_peak - noise_peak)
```

**Isječak koda 5.4:** izbor kompleksa QRS u Pan-Tompkinsovom algoritmu





Slika 5.3: Označene točke korištenjem Pan-Tompkinsovog algoritma

## 5.2. Izvedba dubokih konvolucijskih auto-ekodera

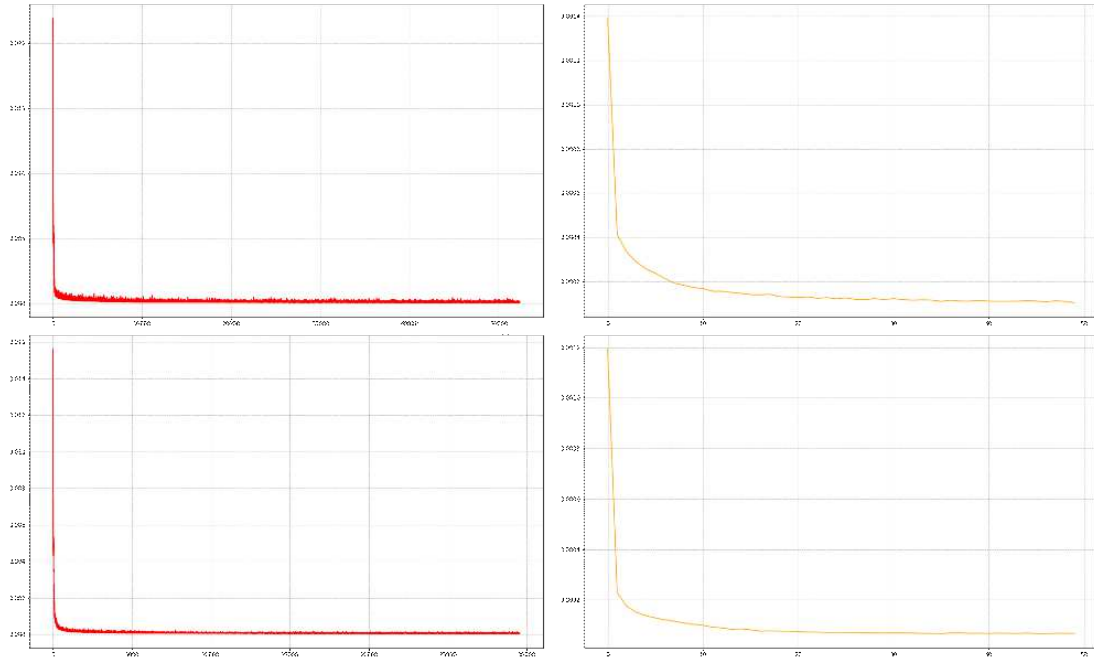
U ovom diplomskom radu naučena su dva duboka konvolucijska auto-ekodera (CAE) sa 16 slojeva, detaljna arhitektura je predstavljena u tablici 5.1. Jedan CAE je učen na izvornom skupu podataka, a drugi na poboljšanom skupu podataka korištenjem GAN-ova. U potpoglavlju 4.1 *Skup podataka* opisan je skup podataka te njegovi primjerci. Svaki primjerak u skupu podataka je snimka EKG-a duljine 260 uzoraka, tj. 0.52 s. Na ulazu u CAE dolaze snimke EKG-a dimenzija **260 x 1** koje predstavljaju pojedinačni otkucaj srca. Kao i svaki auto-ekoder, CAE se sastoji od ekodera i dekodera.

Ekoder se sastoji od slojeva 1-8 iz tablice 5.1 i ima zadaću sažimanja izvorne snimke EKG-a (**260 x 1**) u sažetu snimku duljine 32 uzorka (**32 x 1**). Izlaz iz sloja pod brojem 8 naziva se usko grlo (engl. *bottleneck*) i sadrži sažetu snimku EKG-a. Koristeći ekoder moguće je prikazati veću snimku EKG-a manjom sažetom snimkom.

Dekoder se sastoji od slojeva 9-16 iz tablice 5.1 i ima zadatak rekonstruiranja izvorne snimke EKG-a iz sažete snimke EKG-a. Rekonstruirana snimka EKG-a (**260 x 1**) uspoređuje se s izvornom snimkom tijekom faze učenja kako bi se dobila vrijednost gubitka. Također, te se dvije snimke uspoređuju tijekom faze ispitivanja kako bi se dobile rekonstrukcijske metrike (engl. *reconstruction metrics*). Slika 5.5 prikazuje izvornu snimku EKG-a, izlaz iz ekodera (sažeta snimka) i izlaz iz dekodera (rekonstruirana snimka).

Oba modela CAE su učena tijekom **50 epoha** s veličinom minigrupe (engl. *batch size*) **128**. Za optimizaciju je korišten optimizator **Adam** s vrijednostima beta (**0.9**,

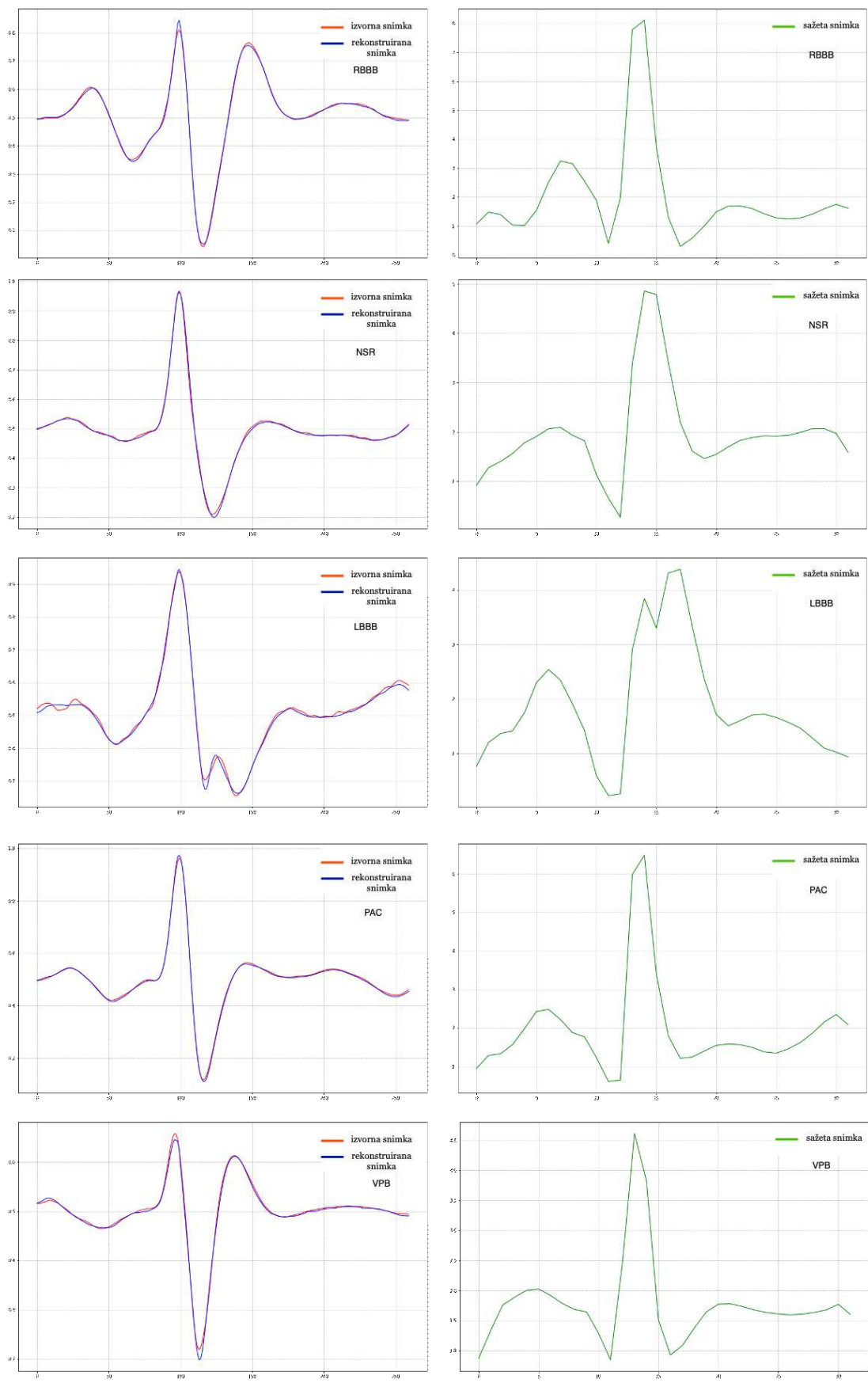
**0.999**) i regularizacijskim faktorom (engl. *weight decay*) **1e-05**. Za funkciju gubitka korištena je **srednja kvadratna pogreška** (engl. *mean squared error, MSE*). Jedina razlika je u koraku učenja, CAE učen na izvornom skupu podataka ima korak učenja **0.0004**, dok drugi CAE ima korak učenja **0.001**. Slika 5.4 prikazuje gubitak tijekom faze učenja za oba CAE-a.



**Slika 5.4:** Gubitak po koraku učenja (*lijevo*) i gubitak po epohi učenja (*desno*). (*Gornji red*) CAE učen na poboljšanom skupu podataka koristeći GAN-ove. (*Donji red*) CAE učen na izvornom skupu podataka.

	Naziv sloja (pytorch)	Dimenzije jezgre	Aktivacijska funkcija	Broj parametara	Dimenzije izlaza	
<b>Ulaz: izvorna snimka EKG-a 260 x 1</b>						
<b>E</b>	1	Conv1d	16 x 5	ReLU	96	260 x 16
<b>N</b>	2	MaxPool1d	2	–	0	130 x 16
<b>K</b>	3	Conv1d	64 x 5	ReLU	5,184	130 x 64
<b>O</b>	4	BatchNorm1d	–	–	128	130 x 64
<b>D</b>	5	MaxPool1d	2	–	0	65 x 64
<b>E</b>	6	Conv1d	32 x 3	ReLU	6,176	65 x 32
<b>R</b>	7	Conv1d	1 x 3	ReLU	97	65 x 1
	8	MaxPool1d	2	–	0	32 x 1
<b>Usko grlo: sažeta snimka EKG-a 32 x 1</b>						
<b>D</b>	9	ConvTranspose1d	1 x 3	ReLU	4	32 x 1
<b>E</b>	10	ConvTranspose1d	32 x 3	ReLU	128	32 x 32
<b>K</b>	11	Upsample	2	–	0	64 x 32
<b>O</b>	12	ConvTranspose1d	64 x 5	ReLU	10,304	64 x 64
<b>D</b>	13	Upsample	2	–	0	128 x 64
<b>E</b>	14	ConvTranspose1d	16 x 5	ReLU	5,136	128 x 16
<b>R</b>	15	Flatten	–	–	0	2048 x 1
	16	Linear	–	–	532,740	260 x 1
<b>Izlaz: rekonstruirana snimka EKG-a 260 x 1</b>						

**Tablica 5.1:** Arhitektura konvolucijskog auto-ekodera



**Slika 5.5:** primjerci izvorne (*crveno*), rekonstruirane (*plavo*) i odgovarajuće sažete snimke (*zeleno*) EKG-a za svaki srčani ritam.

### 5.3. Izvedba generativnih suparničkih mreža

U ovom diplomskom radu naučena su **tri** modela generativne suparničke mreže (GAN). Zašto su naučena tri modela, a ne jedan ili čak pet? U potpoglavlju 4.1 *Skup podataka* prikazana je nakrivljena distribucija (engl. *skewed distribution*) izvornog skupa podataka gdje NSR i RBBB svojom zastupljenošću dominiraju nad ostalim srčanim ritmovima. To predstavlja problem kod učenja GAN-a jer se tada generator, kojemu je cilj zavarati diskriminator generiranjem što realističnijih primjeraka, fokusira na generiranje primjeraka dominantnog razreda srčanih ritmova kako bi optimirao svoj gubitak. Time se izaziva urušavanje vjerojatnosti generiranja slabije zastupljenih razreda srčanih ritmova. Ovaj problem je poznat pod nazivom problema urušavanja vjerojatnosti generiranja (engl. *mode collapse problem*) [24]. Drugi problem je specifičan za slučajeve gdje je potrebno generirati označene podatke. Pretpostavimo da je moguće izbjeći prvi problem i uspješno naučiti jedan GAN za generiranje svih srčanih ritmova. Iako zvuči odlično, nailazimo na novi problem, neki razredi srčanih ritmova imaju slične primjerke i nije moguće sa 100%-tnom točnošću označiti sve generirane primjerke. Iz svega navedenoga da se zaključiti kako je najbolje rješenje naučiti pet GAN-ova, po jedan za svaki srčani ritam. Ipak su naučena tri, jer srčani ritmovi NSR i RBBB su već dovoljno zastupljeni u skupu podataka i ne zahtijevaju dodatne umjetne primjerke. Postupak poboljšavanja podataka primijenjen u ovom diplomskom radu opisan je u sljedećem potpoglavlju 5.4 *Poboljšavanje skupa podataka*.

Svaki GAN ima istu arhitekturu, jedine razlike su u hiperparametrima i skupu podataka za učenje. Detaljna arhitektura generatora prikazana je u tablici 5.3, a diskriminatora u tablici 5.4. Tablica 5.2 prikazuje postavke hiperparametara za model pojedinačnog srčanog ritma, svaki model koristi optimizator **Adam** s vrijednostima beta **(0.9, 0.999)** i regularizacijskim faktorom **1e-05**. Funkcija gubitka **binarne unakrsne entropije** (engl. *binary cross-entropy, BCE*) je korištena kao funkcija gubitka za svaki model.

Srčani ritam	Broj epoha	Veličina minigrupe	Korak učenja
LBBS	100	128	0.0005
PAC	50	64	0.0002
VPB	100	128	0.0005

**Tablica 5.2:** Postavke hiperparametara za model pojedinog srčanog ritma.

	Naziv sloja (pytorch)	Aktivacijska funkcija	Broj parametara	Dimenzije izlaza
<b>Ulaz: šum 100 x 1</b>				
1	Linear	-	12,800	128 x 1
2	BatchNorm1d	LeakyReLU	256	128 x 1
3	Linear	-	32,786	256 x 1
4	BatchNorm1d	LeakyReLU	512	128 x 1
5	Linear	-	131,072	512 x 1
6	BatchNorm1d	LeakyReLU	1,024	128 x 1
7	Linear	Sigmoid	133,120	260 x 1
<b>Izlaz: umjetna snimka EKG-a 260 x 1</b>				

**Tablica 5.3:** Arhitektura generatora.

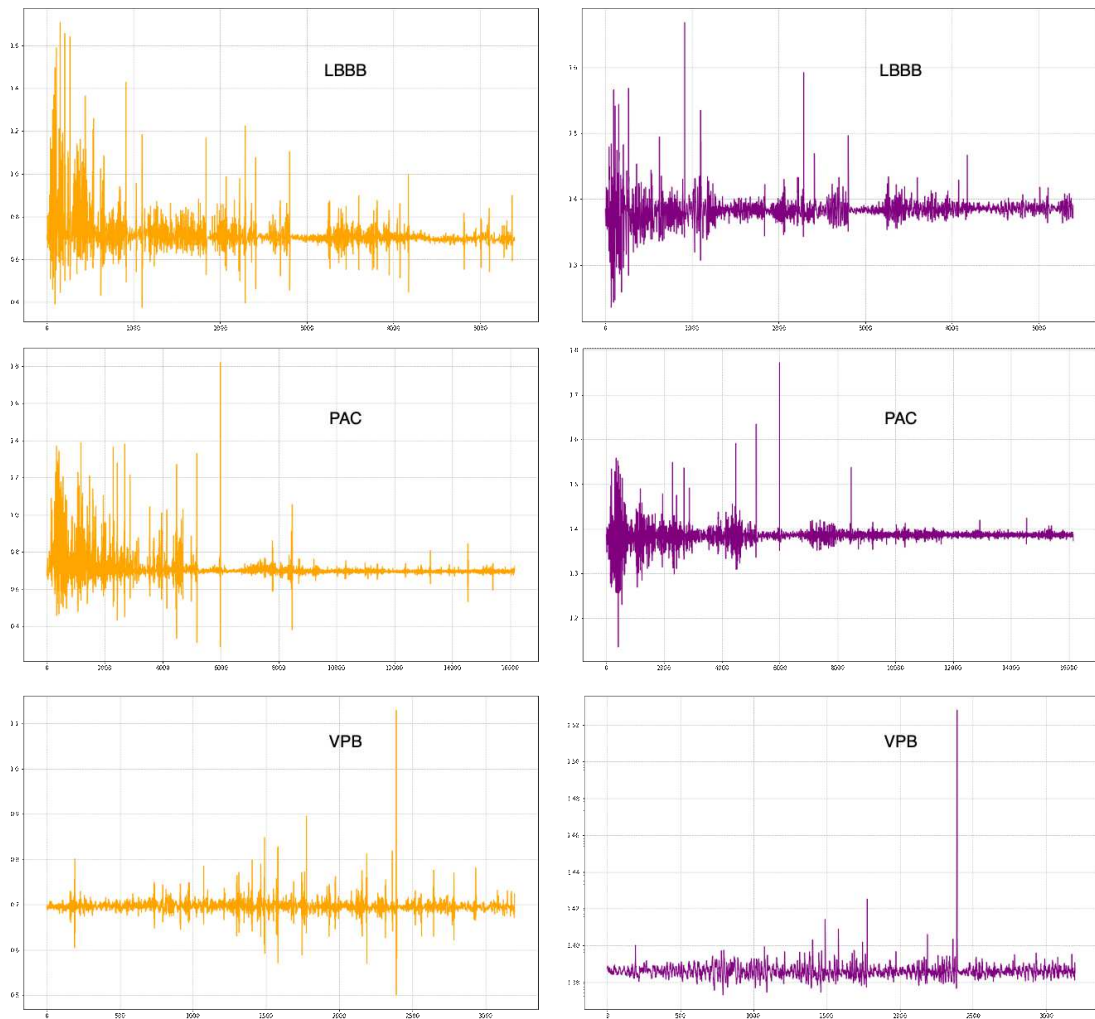
	Naziv sloja (pytorch)	Aktivacijska funkcija	Broj parametara	Dimenzije izlaza
<b>Ulaz: umjetno generirani ili stvarni primjerak 260 x 1</b>				
1	Linear	LeakyReLU	133,632	512 x 1
2	Linear	LeakyReLU	131,328	256 x 1
3	Linear	LeakyReLU	32,896	128 x 1
4	Linear	LeakyReLU	8,256	64 x 1
5	Linear	Sigmoid	65	1
<b>Izlaz: vjerojatnost radi li se o stvarnom ili umjetnom primjerku</b>				

**Tablica 5.4:** Arhitektura diskriminatora.

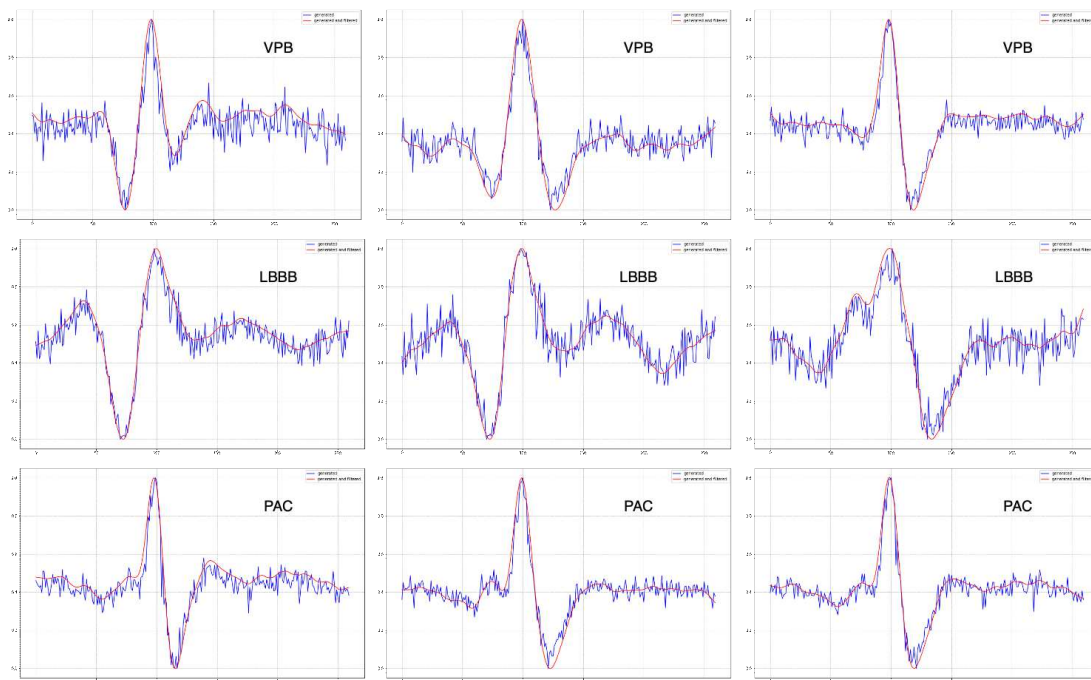
Slika 5.6 prikazuje gubitak generatora i diskriminatora tijekom faze učenja po koraku za svaki srčani ritam. Zbog načina na koji GAN funkcionira, opisano u poglavlju 4.3.5 *Generativna suparnička mreža – GAN*, grafovi gubitka imaju specifičan izgled. Za razliku od grafova gubitka CAE sa slike 5.4, gubitak generatora i diskriminatora ne konvergira u vrijednost blizu nule, već konvergira u vrijednost *ekvilibrija*.

Konačno, slika 5.7 prikazuje konačni proizvod naučenog generatora za svaki srčani ritam. Neposredni izlaz iz generatora (plava linija), iako je šumovit, ima oblik snimke EKG-a. Kako bi se neposredni izlaz zagladio i mogao koristiti za poboljšavanje skupa

podataka, mora proći kroz filtar Savitzky-Golay [29]. Zaglađivanjem filtrom se dobiva konačna umjetno generirana snimka EKG-a (crvena linija).



**Slika 5.6:** Gubitak tijekom faze učenja po koraku za svaki srčani ritam. Gubitak po koraku za generator (*lijevo*). Gubitak po koraku za diskriminator (*desno*).



**Slika 5.7:** Umjetno generirane snimke EKG-a za svaki srčani ritam. Neposredni izlaz iz generatora (*plavo*). Filtrirani izlaz iz generatora (*crveno*).

## 5.4. Poboljšavanje skupa podataka

U potpoglavlju 4.1 *Skup podataka* pomoću slike 4.1 opisan je problem neuravnoteženosti razreda srčanih ritmova unutar izvornog skupa podataka. Ritmovi LBBB, PAC i VPB nisu dovoljno zastupljeni u izvornom skupu podataka. Stoga se u ovom diplomskom radu primjenjuje metoda poboljšavanja skupa podataka (engl. *data augmentation*) kako bi se izvorni skup podataka uravnotežio dodavanjem umjetno generiranih primjeraka. Implementirana su dva načina poboljšavanja skupa podataka: koristeći CAE te koristeći GAN-ove. Oba dva načina dovode do nove i uravnotežene distribucije skupa podataka prikazane u tablici 5.5. Važno je napomenuti kako su umjetno generirani podaci korišteni **samo za poboljšavanje skupa podataka za učenje**. Skup podataka za testiranje sadrži samo izvorne podatke kako bi vjerodostojnost eksperimentalnih rezultata u poglavlju 6 *Rezultati i rasprava* ostala netaknuta.

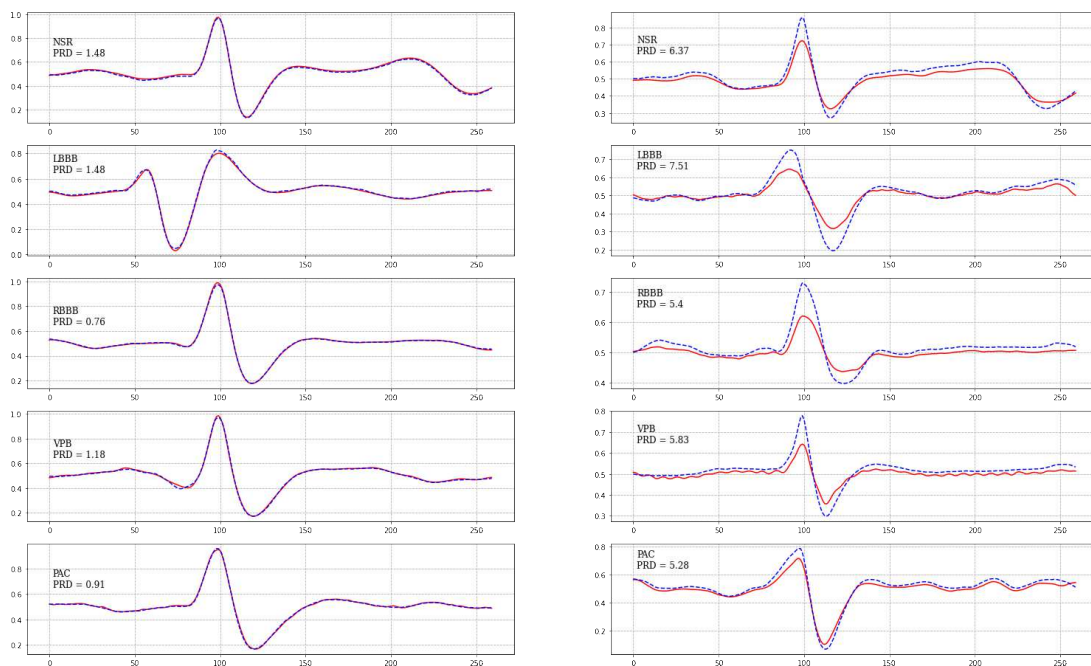
Prvi način poboljšavanja skupa podataka zasniva se na već naučenom konvolucijskom autonekoderu opisanom u potpoglavlju 5.2. CAE, dakle, na ulazu dobiva izvornu snimku, sažima ju te konačno rekonstruira. Fokusirajmo se na usporedbu izvorne i rekonstruirane snimke. Slika 5.8 pokazuje po dva para izvorne i rekonstruirane snimke za svaki srčani ritam. Na slici uz naziv srčanog ritma navedena je i metrika PRD,



ona je detaljnije objašnjena u potpoglavlju 6.1 *Rekonstrukcija*. Zasad je samo bitno da ta metrika govori o kvaliteti rekonstrukcije na način što je ona niža, rekonstrukcija je kvalitetnija. Prvi par za svaki srčani ritam (lijevi stupac) je primjer odlične rekonstrukcije izvorne snimke s niskom vrijednošću PRD-a. Drugi par svakog srčanog ritma (desni stupac) je primjer rekonstrukcije slabije kvalitete s višim vrijednostima PRD-a. Fokusrirajmo se na samo na druge parove. Vrlo su uočljive razlike rekonstruiranih snimaka u odnosu na izvorne, no ipak rekonstruirane snimke zadržavaju općeniti oblik izvorne snimke EKG-a. Ove snimke se razlikuju od izvornih, ali ostavljaju dojam stvarnih i jedinstvenih snimaka EKG-a čemu se teži pri stvaranju umjetnih primjeraka. Naravno, postoji granica za vrijednost PRD-a iznad koje rekonstruirane snimke izgledaju pomalo deformirano i nisu upotrebljive za poboljšavanje skupa podataka. Također, rekonstruirane snimke s preniskim PRD-om, koliko god to bilo odlično, su skoro pa identične replike izvornih snimaka što također nije poželjno jer stvara preveliku pristranost u skupu podataka. Donja granica vrijednosti PRD-a je postavljena na **1.5**, a gornja granica na **10**. Svakako bi se gornja granica PRD-a mogla sniziti što bi moglo utjecati na konačne rezultate, no u sklopu ovog rada takav eksperiment nije proveden. Generiranje umjetnih primjeraka je implementirano na sljedeći način. Već naučenom modelu CAE se pokazuju svi primjerci iz skupa podataka za učenje, model sažima ulaznu snimku i vraća rekonstruiranu snimku, dohvaća se vrijednost PRD-a i odlučuje dodaje li se rekonstruirana snimka u skup generiranih primjeraka. Ukoliko se nije generiralo dovoljno primjeraka za određeni srčani ritam, postupak se ponavlja, ali na skupu generiranih primjeraka za taj ritam s podešenom donjom i gornjom granicom vrijednosti PRD-a. Kada je dovoljno primjeraka generirano za svaki srčani ritam postupak poboljšavanja skupa podataka se obustavlja.

Srčani ritam	Generirani otkucaji	Broj otkucaja u skupu podataka	Broj otkucaja u skupu za učenje
NSR	0	37909	26537
LBBB	19244	26116	24055
RBBB	0	38593	27016
PAC	14427	35037	28854
VPB	22784	26852	25632
<b>UKUPNO</b>	<b>56455</b>	<b>164507</b>	<b>132094</b>

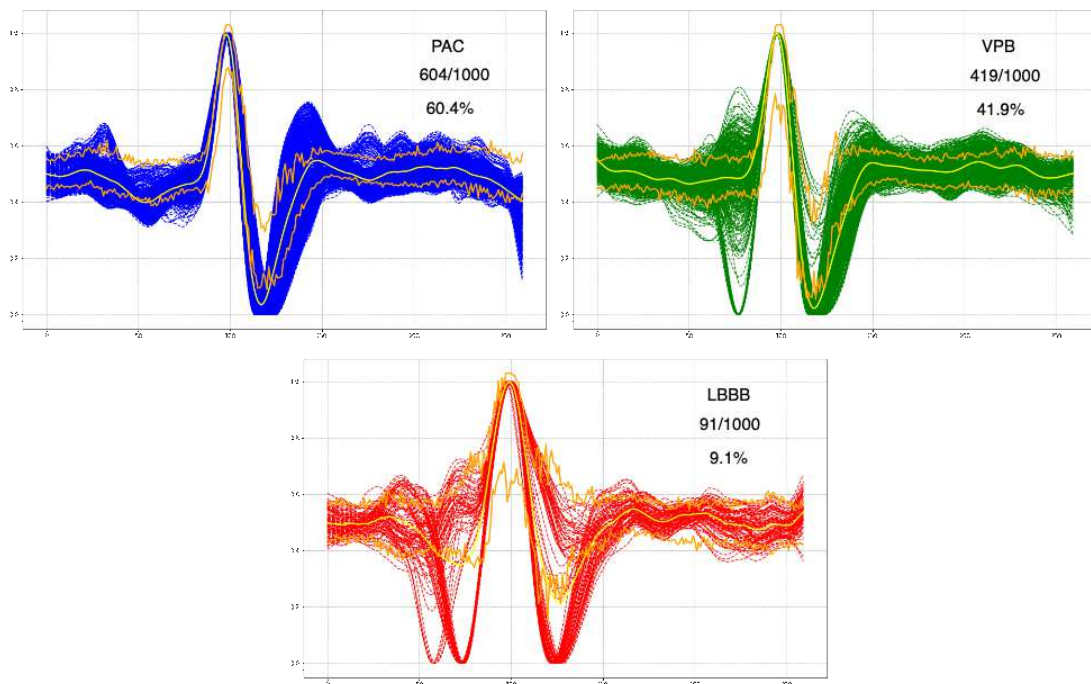
**Tablica 5.5:** Distribucija skupa podataka nakon poboljšanja.



**Slika 5.8:** Parovi izvorne snimke (*crvena linija*) i odgovarajuće rekonstruirane snimke EKG-a (*plava iscertana linija*) za svaki srčani ritam s naznačenim vrijednostima PRD-a. (*Lijevo*) primjeri odlične rekonstrukcije izvorne snimke, s nezamjetnim odstupanjima od izvorne snimke. (*Desno*) primjeri rekonstrukcije niže kvalitete (ali i dalje prihvatljive) s izraženijim odstupanjima od izvorne snimke, naročito na dijelu kompleksa QRS.

Drugi način poboljšavanja skupa podataka ostvaren je koristeći generatore naučenih GAN-ova, opisano u prethodnom potpoglavlju 5.3, za pojedini srčani ritam. Da ponovimo, naučena su tri GAN-a, točnije generatora, za srčane ritmove LBBB, PAC te VPB. Generator na ulazu dobiva vektor šuma, a na izlazu daje umjetnu snimku EKG-a koja se tada zaglađuje filtrom Savitzky-Golay kako bi se dobila konačna umjetno generirana snimka EKG-a. Slika 5.7 prikazuje izlaze iz generatora za pojedine srčane ritmove. Generatori ipak generiraju razne oblike EKG-a, neki od kojih bi samo unijeli nepotrebni šum u skup podataka za učenje. Kako bi se osiguralo da umjetno generirani primjerci dolaze iz iste distribucije, primijenjen je konsenzus slučajnog uzorka (engl. *Random sample consensus*, *RANSAC*). RANSAC je iterativna metoda koja se koristi za identificiranje stršećih vrijednosti (engl. *outliers*) u skupu vrijednosti. RANSAC se prvo primjenjuje u analizi izvornog skupa podataka, uzima se skup podataka za određeni srčani ritam i primjenjuje RANSAC na skupu vrijednosti za svaki od 260 uzoraka koji čine snimku EKG-a. Tom analizom otkriva se interval prihvatljivih vrijednosti (engl. *inlier interval*) za svaki od 260 uzoraka, što kada se spoji čini interval prihvatljivih primjeraka za skup podataka. Sada kada je interval prihvatljivih primje-

raka poznat primjenjuje se na generirane snimke kako bi osigurali da dolaze iz iste distribucije. Prvo se interval malo proširi kako ne bi bio previše ograničavajuć. Zatim se generira tisuću umjetnih snimaka EKG-a. Sada se prošireni interval prihvatljivih primjeraka koristi kako bi se odlučilo koji umjetno generirani primjerci se mogu koristiti za poboljšavanje skupa podataka. Ako zbroj odstupanja primjerka po svim uzorcima ne prelazi maksimalnu vrijednost od **8**, generirani primjerak je prihvaćen. Slika 5.9 prikazuje opisani postupak. Svaki graf prikazuje interval prihvatljivih primjeraka (narančaste linije), srednju vrijednost prihvaćenih primjeraka (žuta linija) te sve prihvaćene primjerke od tisuću generiranih (plavo, zeleno i crveno). Vidljivo je kako je najviše primjeraka prihvaćeno za PAC, pa za VPB i najmanje za LBBB s niskih 9.1% prihvaćenosti. No to nije problem, jer samo produžava postupak poboljšavanja skupa podataka koji se obustavlja kada se dovoljno umjetno generiranih podataka prihvati.



**Slika 5.9:** Postupak odlučivanja o prihvaćanju umjetno generiranih primjeraka za svaki srčani ritam. (Narančaste linije) interval prihvatljivih primjeraka koji se koristi kao temelj u odlučivanju. (Žuta linija) srednja vrijednost prihvaćenih primjeraka. (Plave, zelene i crvene linije) prihvaćeni umjetno generirani primjerci.

## 5.5. Izvedba duge kratkoročne memorije

U ovom diplomskom radu naučena su dvije LSTM-mreže za klasifikaciju srčanih ritmova iz **sažete** snimke EKG-a, detaljna arhitektura se nalazi u tablici 5.6. Ova se mreža

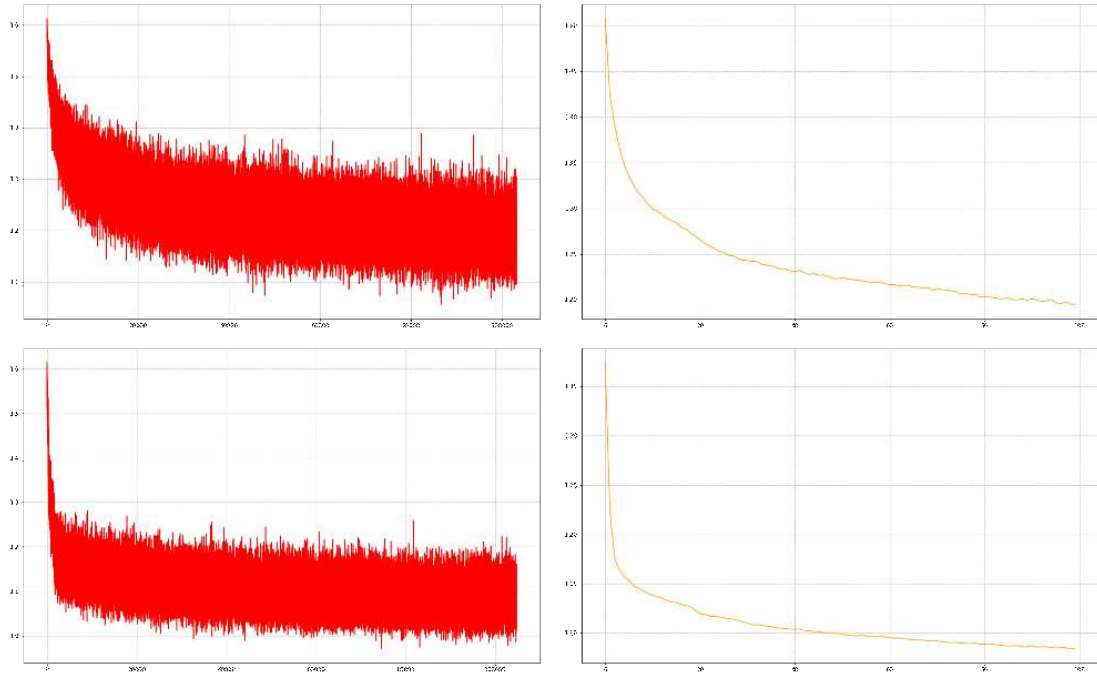
sastoji od pet slojeva, na ulazu primaju sažetu snimku EKG-a duljine 32 uzorka (0.52 s), a na izlazu aktivacijska funkcija *softmax* daje klasifikaciju s vjerojatnošću za svaku od 5 klasa (5 srčanih ritmova) ovog višeklasnog klasifikacijskog zadatka. Naučene su dvije *verzije* mreže, jedna kada se koristi CAE za poboljšavanje skupa podataka, a druga kada se koriste GAN-ovi. Arhitekture su im identične, razlikuju se u skupu podataka za učenje i hiperparametrima.

	Naziv sloja (pytorch)	Postavka sloja (pytorch)	Dimenzije izlaza
<b>Input: Compressed ECG signal 32 x 1</b>			
1	LSTM	batch first = True	32 x 1
2	Flatten	-	32
3	Linear	activation = ReLU	128
4	Dropout	p = 0.1	128
5	Linear	activation = softmax	5

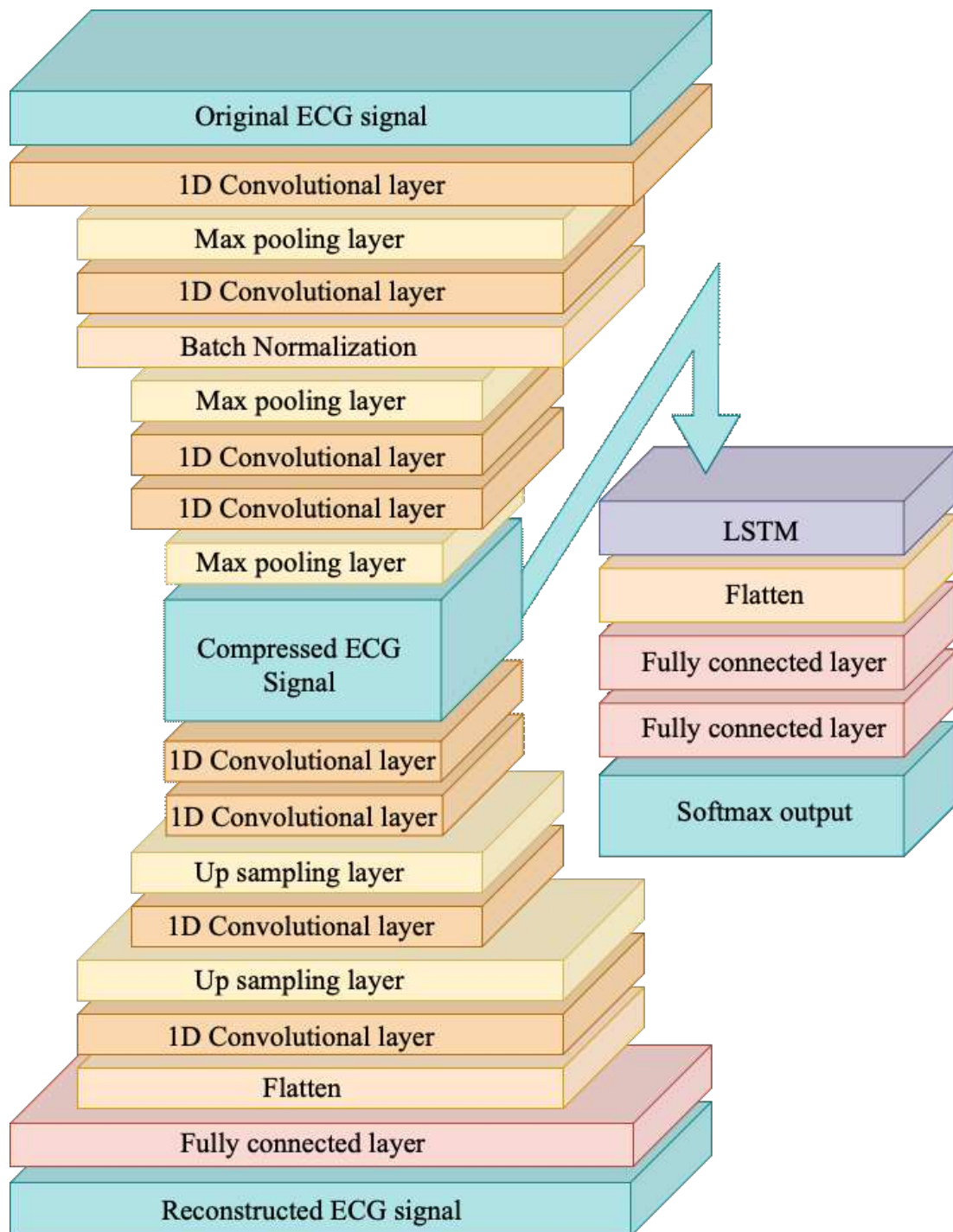
**Tablica 5.6:** Arhitektura LSTM-klasifikatora.

Objekti LSTM-mreže su učene kroz **100** epoha s veličinom minigrupe **128**. Kao optimizator je u oba slučaja korišten **Adam** s korakom učenja **0.005**, vrijednostima beta (**0.9**, **0.999**) i faktorom regularizacije **1e-05**. Također je u oba slučaja korištena **unakrsna entropija** (engl. *cross-entropy*) kao funkcija gubitka. Slika 5.10 prikazuje gubitak tijekom faze učenja za obje LSTM-mreže.

Konačno, opisane su izvedbe svih elemenata ovog klasifikatora CAE-LSTM. Kada se sve spoji, tok rada klasifikatora je sljedeći: na ulaz CAE dolazi snimka EKG-a duljine 260 uzoraka, tj. snimka jednog otkucaja, koja se sažima i šalje na ulaz klasifikatorske LSTM-mreže koja na svom izlazu daje klasifikaciju jednog od pet srčanih ritmova (NSR, RBBB, LBBB, PAC ili VPB). Slika 5.11 prikazuje blok dijagram dubokog klasifikatora CAE-LSTM. Tijekom klasifikacije se ne koristi dekoderski dio toka CAE-a.



**Slika 5.10:** Gubitak po koraku učenja (*lijevo*) i gubitak po epohi učenja (*desno*). (*Gornji red*) LSTM-mreža učena na poboljšanom skupu podataka koristeći CAE. (*Donji red*) LSTM mreža učena na poboljšanom skupu podataka koristeći GAN-ove



Slika 5.11: Blok dijagram arhitekture dubokog modela CAE-LSTM

## 6. Rezultati i rasprava

U ovom poglavlju pregledavaju se eksperimentalni rezultati koristeći **skup podataka za ispitivanje**. Rezultati su podijeljeni u dvije kategorije: rekonstrukcijski i klasifikacijski rezultati. Rekonstrukcijski rezultati dolaze iz evaluacije konvolucijskih auto-inkodera (CAE) i označuju kvalitetu sažimanja i rekonstrukcije snimaka EKG-a. Klasifikacijski rezultati dolaze iz evaluacije LSTM-mreža i označuju kvalitetu klasifikacije srčanih ritmova iz sažetih snimaka EKG-a.

### 6.1. Rekonstrukcija

Kako je već opisano, CAE se koristi za sažimanje i rekonstrukciju snimke EKG-a. Izvorna snimka duljine 260 uzoraka dolazi na ulaz, sažima se u sažetu snimku duljine 32 uzorka te na izlazu vraća rekonstruiranu snimku duljine 260 uzoraka. Tijekom ovog procesa jedan dio informacije izvorne snimke se gubi što se posljedično očituje u razlici između rekonstruirane i izvorne snimke, koliko god malena ta razlika bila. Kako bi se vrednovala kvaliteta rekonstrukcije, korištene su sljedeće rekonstrukcijske metrike: kvadratna sredina (engl. *root means squared, RMS*), postotna razlika kvadratne sredine (engl. *percentage root means squared difference, PRD*), normalizirani PRD (*PRDN*), ocjena kvalitete (engl. *quality score, QS*), omjer sažimanja (engl. *compression ratio, CR*) te odnos signala i šuma (engl. *signal-to-noise ratio, SNR*).

CR je jedina mjera koja je direktno povezana sa sažimanjem. Određuje se dijeljenjem veličine izvornog signala s veličinom sažetog signala. Veća vrijednost CR-a znači jače sažimanje. Kako bi se sažeti signal mogao što bolje rekonstruirati, bitno je odabrati dobru vrijednost CR-a. U ovom diplomskom radu koristi se vrijednost CR-a od 8.125.

Mjera QS označava kvalitetu rekonstruiranog signala, viša vrijednost QS-a ukazuje na kvalitetniju rekonstrukciju. RMS, PRDN i SNR ukazuju na količinu gubitka informacije i prisustvo komponenata šuma u sažetom signalu.

PRD je izrazito bitna rekonstrukcijska mjera koja označava količinu izgubljene in-

formacije u rekonstruiranom signalu. Niža vrijednost PRD-a znači manje izgubljene informacije što zauzvrat znači bolju rekonstrukciju. Slijede formule za korištene rekonstrukcijske mjere, gdje  $X_o$  označava izvorni signal,  $X_r$  rekonstruirani signal, a  $\bar{X}$  srednju vrijednost izvornog i rekonstruiranog signala:

$$PRD = 100 * \sqrt{\frac{\sum_{i=0}^{N-1} (X_o(i) - X_r(i))^2}{\sum_{i=0}^{N-1} (X_o(i))^2}} \quad (6.1)$$

$$PRDN = 100 * \sqrt{\frac{\sum_{i=0}^{N-1} (X_o(i) - X_r(i))^2}{\sum_{i=0}^{N-1} (X_o(i) - \bar{X})^2}} \quad (6.2)$$

$$RMS = \sqrt{\frac{\sum_{i=0}^{N-1} (X_o(i) - X_r(i))^2}{N - 1}} \quad (6.3)$$

$$SNR = 10 * \log \frac{\sum_{i=0}^{N-1} (X_o(i) - \bar{X})^2}{\sum_{i=0}^{N-1} (X_o(i) - X_r(i))^2} \quad (6.4)$$

$$QS = \frac{CR}{PRD} \quad (6.5)$$

Rekonstrukcijski rezultati su prikazani u tablicama 6.1 i 6.2. Tablica 6.1 odgovara slučaju kada je CAE učen na izvornom skupu podataka, dok tablica 6.2 slučaju kada je CAE učen na poboljšanom skupu podataka. Vidi se kako razred normalnog sinusnog ritma ima najbolje vrijednosti za svaku mjeru, što je i za očekivati jer je najzastupljeniji u skupu podataka. Razred PAC ga prati na drugom mjestu, razlog tomu je vjerojatno sličnost generalnog oblika primjeraka NSR-a i PAC-a. Štoviše, svi razredi imaju odlične rekonstrukcijske rezultate u objema tablicama. LBBB u tablici 6.2 postiže vrijednost PRD-a od 2.46 što pomalo strši, ali je i dalje dobar rezultat i nije razlog za brigu. Također, vidi se da nema značajnije razlike u rezultatima za dva modela CAE.

Srčani ritam	PRD	PRDN	RMS	SNR	QS
NSR	<b>1.31</b>	<b>5.72</b>	0.01	<b>58.21</b>	<b>6.71</b>
LBBB	1.94	7.61	0.01	53.17	4.69
RBBB	1.76	7.65	0.01	53.73	5.37
PAC	1.43	6.80	0.01	55.23	6.16
VPB	1.66	8.03	0.01	51.98	5.58
<b>Prosjek</b>	1.63	7.16	0.01	54.46	5.7

**Tablica 6.1:** Rekonstrukcijske metrike za CAE učen na izvornom skupu podataka.



Srčani ritam	PRD	PRDN	RMS	SNR	QS
NSR	<b>1.35</b>	<b>5.89</b>	0.01	<b>58.08</b>	<b>6.79</b>
LBBB	2.46	9.53	0.01	49.15	4.01
RBBB	1.95	8.49	0.01	52.51	5.22
PAC	1.62	7.42	0.01	54.15	5.99
VPB	1.86	8.98	0.01	50.57	5.39
<b>Prosjek</b>	1.85	8.06	0.01	52.89	5.48

**Tablica 6.2:** Rekonstrukcijske metrike za CAE učen na poboljšanom skupu podataka koristeći GAN-ove.

## 6.2. Klasifikacija

LSTM-klasifikator ima zadaću klasificiranja pet razreda srčanih ritmova iz sažete snimke EKG-a. Tablica 6.3 prikazuje klasifikacijske rezultate za LSTM-klasifikator učen na skupu podataka poboljšanim koristeći CAE, dok tablica 6.4 prikazuje klasifikacijske rezultate za LSTM-klasifikator učen na skupu podataka poboljšanim koristeći GAN-ove. Korištene su mjere točnosti (engl. *accuracy*), pozitivne prediktivne vrijednosti (engl. *positive predictive value, PPV*), senzitivnosti (engl. *sensitivity*), specifičnosti (engl. *specificity*) i negativne prediktivne vrijednosti (engl. *negative predictive value, NPV*). Ove mjere su odabrane zbog njihove uporabe u ispitivanjima dijagnostike u medicini [30]. Slijede formule za svaku od navedenih mjera:

$$Tocnost = \frac{TP + TN}{TP + FP + TN + FN} \quad (6.6)$$

$$Senzitivnost = \frac{TP}{TP + FN} \quad (6.7)$$

$$PPV = \frac{TP}{TP + FP} \quad (6.8)$$

$$Specifičnost = \frac{TN}{TN + FP} \quad (6.9)$$

$$NPV = \frac{TN}{TN + FN} \quad (6.10)$$

TP označuje ispravno klasificirane pozitivne primjerke (engl. *true positives*), a TN označuje ispravno klasificirane negativne primjerke (engl. *true negatives*). S druge

strane, FP označuje lažno pozitivne primjerke (engl. *false positives*), a FN lažno negativne (engl. *false negatives*). Također, senzitivnost je mogućnost ispravnog klasificiranja postojanja određenog srčanog ritma, dok je specifičnost mogućnost ispravnog klasificiranja nepostojanja određenog srčanog ritma.

Srčani ritam	Točnost	PPV	Senzitivnost	Specifičnost	NPV
NSR	88.4	54.81	81.34	96.84	89.5
LBBB	<b>96.65</b>	<b>94.16</b>	<b>88.23</b>	<b>97.2</b>	<b>98.68</b>
RBBB	91.58	72.01	84.52	96.61	93.07
VPB	93.91	84.63	84.09	96.15	96.29
PAC	82.94	80.97	57.81	83.48	94.01
<b>Average</b>	<b>90.7</b>	<b>77.32</b>	<b>79.2</b>	<b>94.06</b>	<b>94.31</b>

**Tablica 6.3:** Rezultati klasifikacije u slučaju poboljšanja skupa podataka koristeći CAE.

Srčani ritam	Točnost	PPV	Senzitivnost	Specifičnost	NPV
NSR	81.76	68.69	<b>88.23</b>	78.27	92.49
LBBB	<b>96.85</b>	74.41	76.90	98.20	<b>98.43</b>
RBBB	86.23	<b>81.69</b>	79.21	90.14	88.64
VPB	96.34	63.41	6.39	<b>99.86</b>	96.46
PAC	83.18	58.50	39.59	93.47	86.77
<b>Average</b>	<b>88.87</b>	<b>69.40</b>	<b>58.06</b>	<b>91.99</b>	<b>92.56</b>

**Tablica 6.4:** Rezultati klasifikacije u slučaju poboljšanja skupa podataka koristeći GAN-ove.

Iako nema tablice s klasifikacijskim rezultatima ostvarenim u slučaju kada nikakvo poboljšavanje skupa podataka nije korišteno, valja napomenuti kako su ti rezultati iznimno loši. Klasifikacijska sposobnost za PAC i VPB je, u tom slučaju, praktički nepostojeća s ekstremno niskim rezultatima. Uz to ni rezultati za ostale srčane ritmove nisu dobri te su značajno lošiji od rezultata predstavljenih u tablicama 6.3 i 6.4.

U slučaju poboljšanja skupa podataka koristeći CAE, iz tablice 6.3 vidi se kako srčani ritam LBBB postiže najbolje rezultate u svim kategorijama. U drugom slučaju, tablica 6.4, najbolji rezultati po kategorijama su raspršeni po srčanim ritmovima. Također, vrijednosti specifičnosti i NPV-a su bolje od vrijednosti senzitivnosti i PPV-a u objema tablicama. U praksi to znači da je ova rješenja kvalitetnije dijagnosticira da

pacijent **nema** određeni srčani ritam, nego da ima. Dodatno, implementirana rješenja postižu vrlo dobru sveukupnu točnost od 90.7% i 88.87%.

Ako se promotri tablica 6.4, uočavaju se zanimljive vrijednosti senzitivnosti, čak i PPV-a, za VPB i PAC. Ako se ove vrijednosti povežu s formulom senzitivnosti, uočava se kako je problem u prevelikom broju lažno negativnih primjeraka za ove razrede. Kad je riječ o vrijednosti PPV-a, NSR također ima nižu nego očekivanu vrijednost. Formula za PPV ukazuje na problem s količinom lažno pozitivnih primjeraka, pretpostavlja se kako je modelu teško sa stalnom točnošću razlučiti između NSR-a, PAC-a i VPB-a.

Vrijednosti PPV-a i senzitivnosti za NSR i PAC iz tablice 6.3 potvrđuju ovu tezu. Ako se promotri dualnost parova vrijednosti PPV-a i senzitivnosti za ove srčane ritmove, može se zaključiti kako model često PAC klasificira kao NSR. Ovaj problem proizlazi iz sličnosti oblika NSR-a i PAC-a.

U usporedbi s rezultatima iz rada [16], implementirana rješenja postižu lošije rezultate. Najveća razlika leži u korištenim skupovima podataka. Kao što je opisano u potpoglavlju 4.2., rad [16] za segmentaciju srčanih otkucaja koristi ekspertno znanje u vidu pomoći nekoliko kardiologa. Ova metoda i skup podataka korišten u [16] superiorni su predobradi i skupu podataka korištenima u ovom diplomskom radu. Kvaliteta skupa podataka posljedično dovodi i do kvalitetnijih rezultata koji dosežu točnost od 99.11%, dok ostvarene točnosti u ovom radu iznose **90.7%** i **88.87%**.

## 7. Zaključak

U ovom diplomskom radu implementiran je duboki model CAE-LSTM, predstavljen u [16], koji sažima ulaznu snimku EKG-a te klasificira sažetu snimku. U sklopu tog modela, implementiran je konvolucijski auto-ekoder (CAE) za sažimanje ulazne snimke i LSTM-mreža za klasifikaciju. Korišteni skup podataka pati od neuravnoteženosti primjeraka pa je primijenjena metoda poboljšanja skupa podataka (engl. *data augmentation*). Poboljšanje skupa podataka ostvareno je implementacijom GAN-ova za generiranje umjetnih snimaka EKG-a kao što je predstavljeno u [18]. Uz tu metodu ostvaren je i pristup poboljšanja skupa podataka korištenjem modela CAE. Konačni rezultati su zadovoljavajući, no nisu sustigli rezultate iz [16].

Rezultati ovog rada mogu se poboljšati na nekoliko načina. Prvi način leži u skupu podataka. Što se tiče veličine skupa podataka, bilo bi dobro nadodati još primjeraka iz drugih skupova podataka i pokušati tim pristupom što je više moguće uravnotežiti skup podataka. S druge strane, kvaliteta skupa podataka se također može unaprijediti kao što su to učinili u [16] uz pomoć kardiologa. Drugi način odnosi se na korištenje naprednije arhitekture GAN-a, npr. valjalo bi isprobati GAN-ove s konvolucijskim slojevima. Naravno, valjalo bi isprobati i usporediti posve drukčija rješenja od implementiranog.

Također, ovaj se rad može i nadograditi. Daljnji korak bi mogao biti ugradnja modela u programsko okruženje koje se može primijeniti u nosivoj tehnologiji ili uz nju.

Konačno, ovim zadatkom se bavim od svog završnog rada gdje sam koristio determinističke algoritme za analizu snimke, morfološku analizu, izlučivanje značajki i klasično strojno učenje. U zaključku tog rada sam predložio korištenje dubokog učenja za poboljšanje rezultata. Drago mi je da mogu potvrditi da sam ostvario tu ideju i kako su se rezultati doista poboljšali. Također mi je drago što sam mogao primijeniti znanje, koje sam stekao stvarajući svoj završni rad, u svom diplomskom radu. I dalje smatram da je ovaj zadatak zahtjevan, ali vrlo zanimljiv.

# LITERATURA

- [1] World Health Organization, “Cardiovascular diseases.” <https://www.who.int>. [Pristupljeno 5. lipnja 2022.].
- [2] Z. Ebrahimi, M. Loni, M. Daneshtalab, and A. Gharehbaghi, “A review on deep learning methods for ECG arrhythmia classification,” *Expert Systems with Applications: X*, vol. 7, p. 100033, 2020.
- [3] X. Liu, H. Wang, Z. Li, and L. Qin, “Deep learning in ECG diagnosis: A review,” *Knowledge-Based Systems*, vol. 227, p. 107187, 2021.
- [4] T. B. Garcia and N. E. Holtz, *12-Lead ECG The Art of Interpretation*. Jones and Bartlett Publisher, 2001.
- [5] Wikipedia, “Desno srce.” <https://hr.wikipedia.org>. [Pristupljeno 5. lipnja 2022.].
- [6] Life in the fast lane, “ECG Lead positioning.” <https://litfl.com>. [Pristupljeno 5. lipnja 2022.].
- [7] Cables and sensors, “12-lead ecg placement guide with illustrations.” <https://www.cablesandsensors.com>. [Pristupljeno 5. lipnja 2022.].
- [8] J. Pan and W. J. Tompkins, “A real-time QRS detection algorithm,” *IEEE Transactions on Biomedical Engineering*, no. 3, pp. 230–236, 1985.
- [9] M. Elgendi, B. Eskofier, and D. Abbott, “Fast T wave detection calibrated by clinical knowledge with annotation of P and T waves,” *Sensors*, vol. 15, no. 7, pp. 17693–17714, 2015.
- [10] M. Šarlija, F. Jurišić, and S. Popović, “A convolutional neural network based approach to QRS detection,” in *Proceedings of the 10th international symposium on image and signal processing and analysis*, pp. 121–125, IEEE, 2017.

- [11] G. B. Moody and R. G. Mark, “The impact of the MIT-BIH arrhythmia database,” *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.
- [12] G. Sannino and G. De Pietro, “A deep learning approach for ECG-based heartbeat classification for arrhythmia detection,” *Future Generation Computer Systems*, vol. 86, pp. 446–455, 2018.
- [13] A. H. Ribeiro, M. H. Ribeiro, G. M. Paixão, D. M. Oliveira, P. R. Gomes, J. A. Canazart, M. P. Ferreira, C. R. Andersson, P. W. Macfarlane, W. Meira Jr, *et al.*, “Automatic diagnosis of the 12-lead ECG using a deep neural network,” *Nature Communications*, vol. 11, no. 1, pp. 1–9, 2020.
- [14] B. Pyakillya, N. Kazachenko, and N. Mikhailovsky, “Deep learning for ECG classification,” in *Journal of Physics: Conference Series*, vol. 913, p. 012004, IOP Publishing, 2017.
- [15] G. D. Clifford, C. Liu, B. Moody, H. L. Li-wei, I. Silva, Q. Li, A. Johnson, and R. G. Mark, “AF classification from a short single lead ECG recording: The PhysioNet/computing in cardiology challenge 2017,” in *2017 Computing in Cardiology (CinC)*, pp. 1–4, IEEE, 2017.
- [16] O. Yildirim, U. B. Baloglu, R.-S. Tan, E. J. Ciaccio, and U. R. Acharya, “A new approach for arrhythmia classification using deep coded features and LSTM networks,” *Computer Methods and Programs in Biomedicine*, vol. 176, pp. 121–133, 2019.
- [17] L. Taylor and G. Nitschke, “Improving deep learning with generic data augmentation,” in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1542–1547, IEEE, 2018.
- [18] A. M. Shaker, M. Tantawi, H. A. Shedeed, and M. F. Tolba, “Generalization of convolutional neural networks for ECG classification using generative adversarial networks,” *IEEE Access*, vol. 8, pp. 35592–35605, 2020.
- [19] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.
- [20] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, *Dive into Deep Learning*. 2019. <https://d2l.ai/d2l-en.pdf>.

- [21] “Classification of 12-lead ECGs: the PhysioNet/Computing in Cardiology Challenge 2020.” <https://moody-challenge.physionet.org/2020/>. [Pristupljeno 19. veljače 2022.].
- [22] V. Labs, “Multilayer feedforward neural networks.” <https://virtuallavs.com>. [Pristupljeno 16. lipnja 2022.].
- [23] N. Buduma, *Fundamentals of Deep Learning*. O’Reilly Media, 2016.
- [24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” *Advances in neural information processing systems*, vol. 27, 2014.
- [25] L. J. Ratliff, S. A. Burden, and S. S. Sastry, “Characterization and computation of local nash equilibria in continuous games,” in *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 917–924, IEEE, 2013.
- [26] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.
- [27] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems 32*, pp. 8024–8035, Curran Associates, Inc., 2019.
- [28] J. D. Hunter, “Matplotlib: A 2d graphics environment,” *Computing in science & engineering*, vol. 9, no. 3, pp. 90–95, 2007.
- [29] R. W. Schafer, “What is a savitzky-golay filter?[lecture notes],” *IEEE Signal processing magazine*, vol. 28, no. 4, pp. 111–117, 2011.
- [30] R. Parikh, A. Mathai, S. Parikh, G. C. Sekhar, and R. Thomas, “Understanding and using sensitivity, specificity and predictive values,” *Indian Journal of Ophthalmology*, vol. 56, no. 1, p. 45, 2008.

# KLASIFIKACIJA SRČANIH RITMOVA IZ SNIMAKA 12-KANALNOG ELEKTROKARDIOGRAMA KORISTEĆI KOMBINACIJU PRISTUPA DUBOKOG UČENJA

## Sažetak

Uobičajeno je da se signali elektrokardiograma (EKG) zapisuju i motre kroz određeni vremenski period i konačno ih analizira stručnjak. Automatska klasifikacija srčanih ritmova ima potencijal poboljšati dijagnostiku. U ovom radu istražuje se korištenje učenja iz reprezentacije snimaka EKG-a za klasifikaciju srčanih ritmova. Skup podataka se sastoji od pet srčanih ritmova i kreiran je od baza podataka CPSC, CPSC-Extra i The Georgia 12-lead ECG Challenge Database. Koristi se sofisticirani pristup dubokog učenja za učenje iz reprezentacije i klasifikaciju, točnije kombinacija konvolucijskog auto-inkodera (CAE) i LSTM-klasifikatora. CAE je korišten za sažimanje ulazne snimke što služi kao ulaz u LSTM-klasifikator. Također je ostvareno poboljšanje skupa podataka (engl. *data augmentation*) za uravnotežavanje skupa podataka zasnovano na CAE te na GAN-ovima. Klasifikacijski rezultati tek prelaze 90% točnosti te pokazuju kako je korištenje ovog složenog pristupa dubokog učenja pogodno za rješavanje problema, ali i da ima mjesta za daljnji napredak.

**Ključne riječi:** klasifikacija srčanih ritmova, EKG, duboko učenje, konvolucijski auto-inkoder, LSTM, GAN, poboljšanje podataka.



# **CARDIAC ARRHYTHMIA CLASSIFICATION FROM 12-LEAD ELECTROCARDIOGRAM USING A COMBINATION OF DEEP LEARNING APPROACHES**

## **Abstract**

Traditionally, electrocardiogram (ECG) signals are recorded and monitored over a period of time and finally analyzed by an expert. Automatic classification of cardiac arrhythmias has the potential to improve diagnostics. In this work, we explore the use of representation learning from ECG signals for cardiac arrhythmia classification. The dataset consisting of five cardiac rhythm types was created from the CPSC, CPSC-Extra, and The Georgia 12-lead ECG Challenge databases. We use a sophisticated deep learning approach for representation learning and classification, namely a combination of a Convolutional Auto-Encoder (CAE) and a Long Short-Term Memory (LSTM) classifier. CAE was used to compress the input signal that serves as input to the LSTM classifier. We also implemented a CAE-based and a GAN-based data augmentation approach to balance the data distribution. The classification results reaching just above 90% accuracy show that the use of the complex deep learning approach is suitable for addressing the problem, but that there is still room for improvement.

**Keywords:** arrhythmia classification, ECG, deep learning, convolutional autoencoder, LSTM, GAN, data augmentation.