
Klasifikacija EKG signala

— Data Mining —

MSc Andrijana Blečić

Definicija problema

- Primijenjena vještačka inteligencija
- Automatizovana analiza velikih količina podataka, čime se poboljšava tačnost dijagnoza i optimizuje proces kliničkog odlučivanja i liječenja
- Personalizovana medicina
- Smanjenje troškova i rizika u zdravstvu
- Detekcija zdravstvenih promjena u realnom vremenu

Proces mašinskog učenja

Enkodiranje

Obrada nenumeričkih atributa



Akvizicija i analiza podataka

Vizuelni prikaz, informacije, atributi

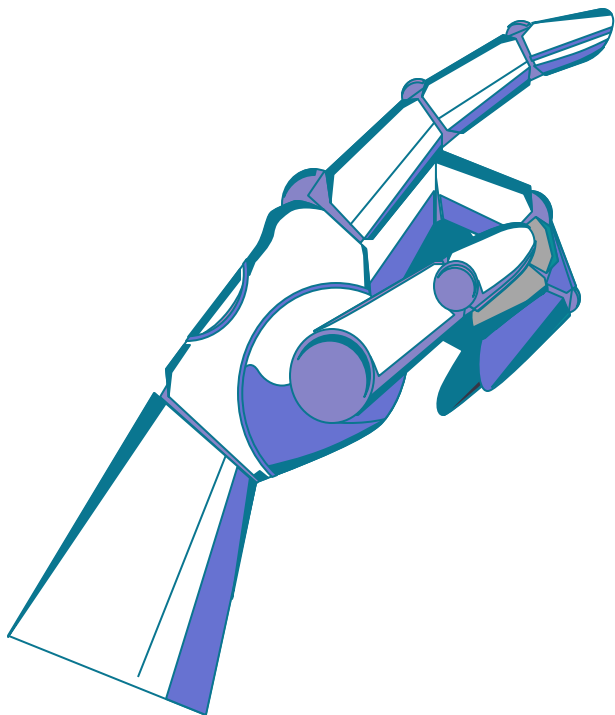
Skaliranje

Različite vrste skalera

Podjela, trening

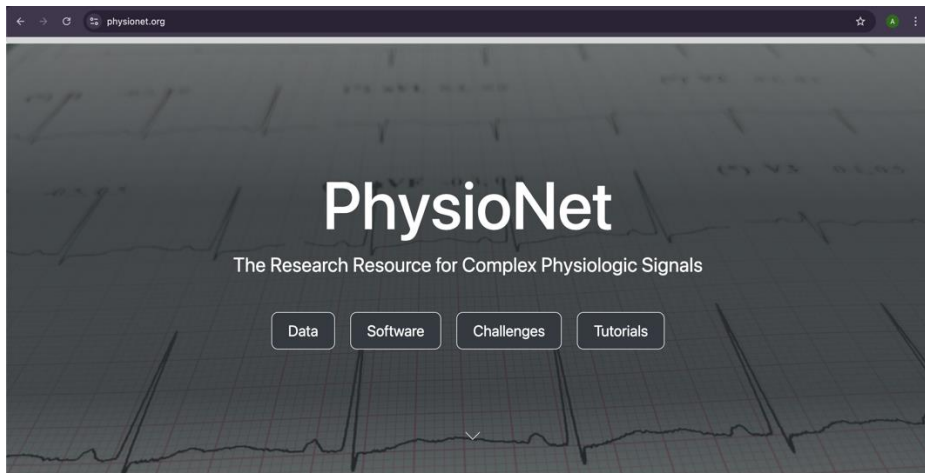
Dodatna obrada, novi pristupi u zavisnosti od rezultata

Akvizicija i analiza podataka



- Physionet
- Analiza dataseta
- Filtriranje podataka
- Filtriranje signala
- Ekstrakcija obilježja

Akvizicija i analiza podataka



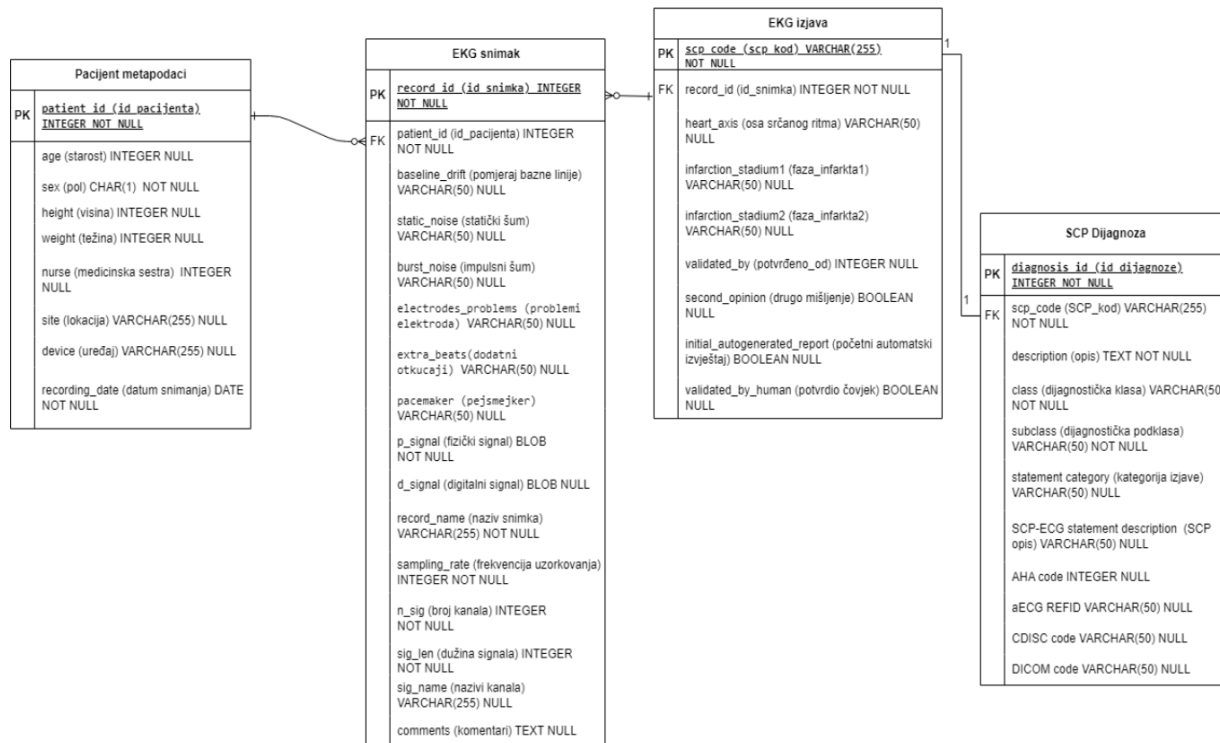
PhysioNet je online repozitorijum besplatno dostupnih fizioloških signala i odgovarajućih komponenti otvorenog koda.

<https://physionet.org/content/ptb-xl/1.0.3/>

PTB-XL EKG skup podataka - obuhvata 21799 kliničkih 12-kanalnih EKG signala u trajanju od 10 sekundi, prikupljenih od 18869 pacijenata.

- EKG snimci i pacijenti identifikovani su pomoću jedinstvenih identifikatora (ecg_id i patient_id).
- Generalni metapodaci: starost, pol, visina, težina, broj medicinske sestre, uređaj za snimanje i datum snimanja;
- Metapodaci snimaka: statična buka(šum), problemi sa elektrodama, broj suvišnih otkucaja.

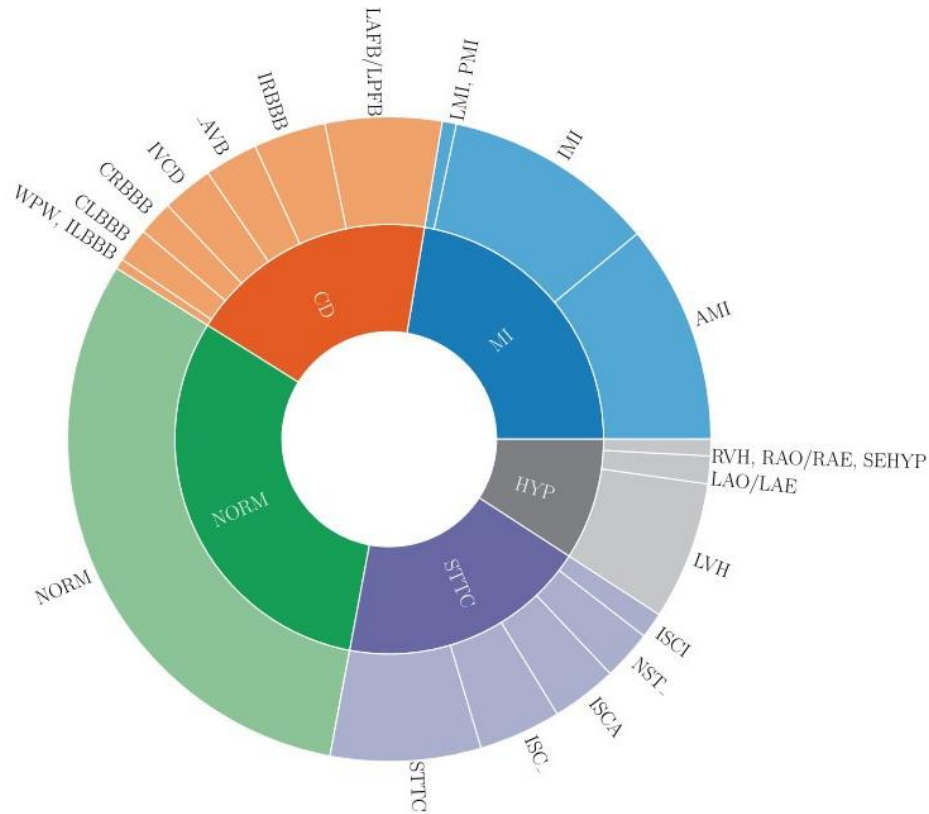
PTB-XL skup podataka



ER dijagram PTB-XL baze podataka, koji prikazuje ključne tabele i njihove međusobne veze, uključujući metapodatke pacijenata, EKG zapise, dijagnostičke izjave i standardizovane SCP dijagnoze

PTB-XL skup podataka

Broj snimaka	Superklasa	Opis
9514	NORM	Normalni EKG (eng. <i>Normal ECG</i>)
5469	MI	Srčani Udar (eng. <i>Myocardial Infarction</i>)
5235	STTC	Promjene na ST/ T segmentima (eng. <i>ST/ T Change</i>)
4898	CD	Poremećaj sprovodnog sistema (eng. <i>Conduction Disturbance</i>)
2649	HYP	Hipertrofija (eng. <i>Hypertrophy</i>)



Analiza i preprocesiranje podataka

WFDB (WaveForm Database) biblioteka

WFDB format podrazumijeva:

- WFDB signal fajlove (.dat) – binarni fajlovi u kojima su uzorci digitalizovanih signala (RECORDNAME.dat)
- WFDB zaglavlja (.hea) – kratki tekstualni fajlovi koji sadrže opis sadržaja odgovarajućeg signal fajla (RECORDNAME.hea).

Funkcije za učitavanje signala: `wfdb.rdrecord` i `wfdb.rdsamp`

```
Dimenzije signala 00001_lr: (1000, 12)
[[-0.119 -0.055  0.064 ... -0.026 -0.039 -0.079]
 [-0.116 -0.051  0.065 ... -0.031 -0.034 -0.074]
 [-0.12  -0.044  0.076 ... -0.028 -0.029 -0.069]
 ...
 [ 0.069  0.    -0.069 ...  0.024 -0.041 -0.058]
 [ 0.086  0.004 -0.081 ...  0.242 -0.046 -0.098]
 [ 0.022 -0.031 -0.054 ...  0.143 -0.035 -0.12  ]]
```

- `wfdb.rdsamp(record_name, sampfrom=0, sampto=None, channels=None, pn_dir=None, channel_names=None, warn_empty=False, return_res=64)`

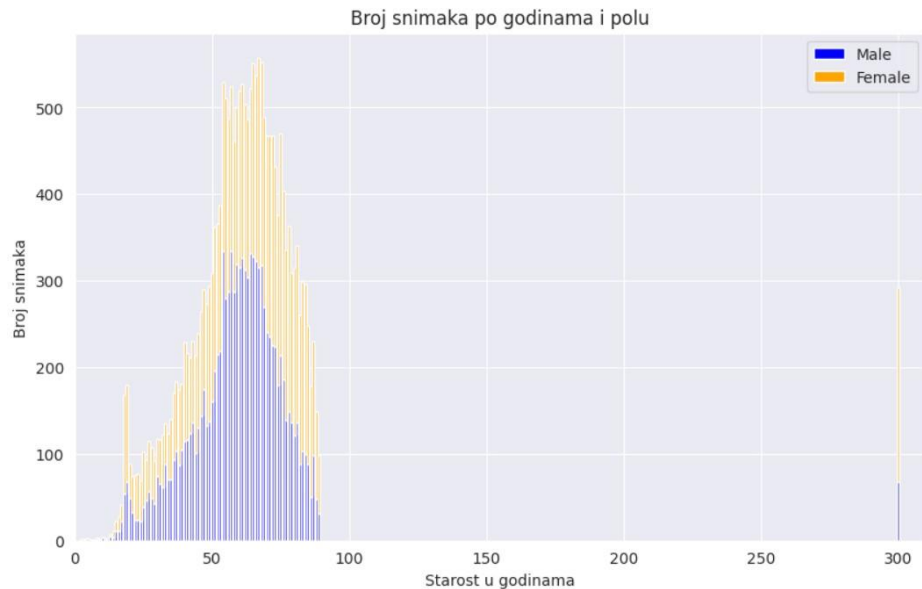
Rezultat ove funkcije su:

- signal: ndarray – 2D NumPy niz koji sadrži fizičke signale iz WFDB zapisa.
- polja: **dict**:
 - **fs**: Frekvencija zapisa (eng. *Sampling frequency*),
 - **units**: fizičke jedinice izmjerenog signala (u ovom slučaju, pošto je u pitanju EKG signal biće *mV*),
 - **sig_name**: naziv signala za svaki kanal,
 - **comments**: sadrži komentare koji su napisani u zaglavlju.

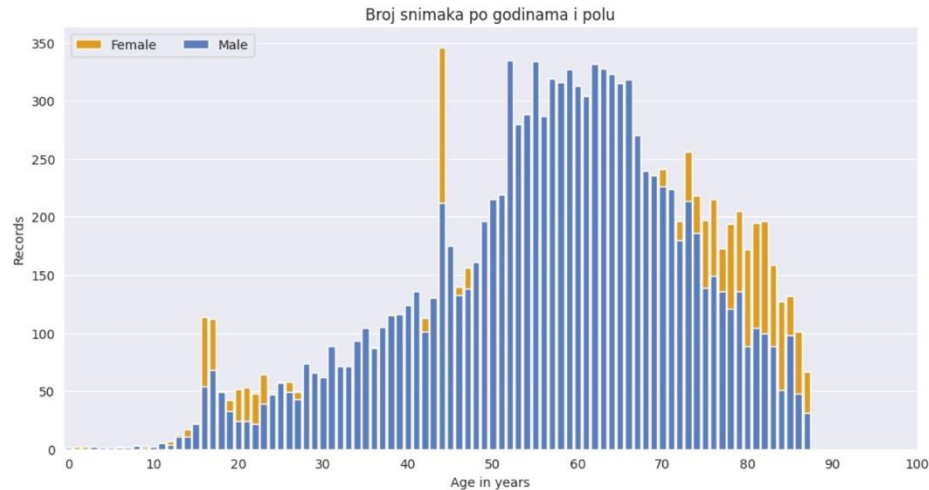
Učitavanje jednog zapisa iz fajla; prvi element rezultata poziva funkcije `wfdb.rdsamp("00001_lr")`.

Analiza i pretprocesiranje podataka

EKG signali su prikupljeni od ukupno 18869 pacijenata, od kojih je 52% muškog i 48% ženskog pola, starosti od 0 do 95 godina (sa prosjekom od 62 godine). Za pacijente starije od 90 godina, ovo polje je popunjeno brojem 300 u skladu sa *HIPAA (Health Insurance Portability and Accountability Act)* standardom.



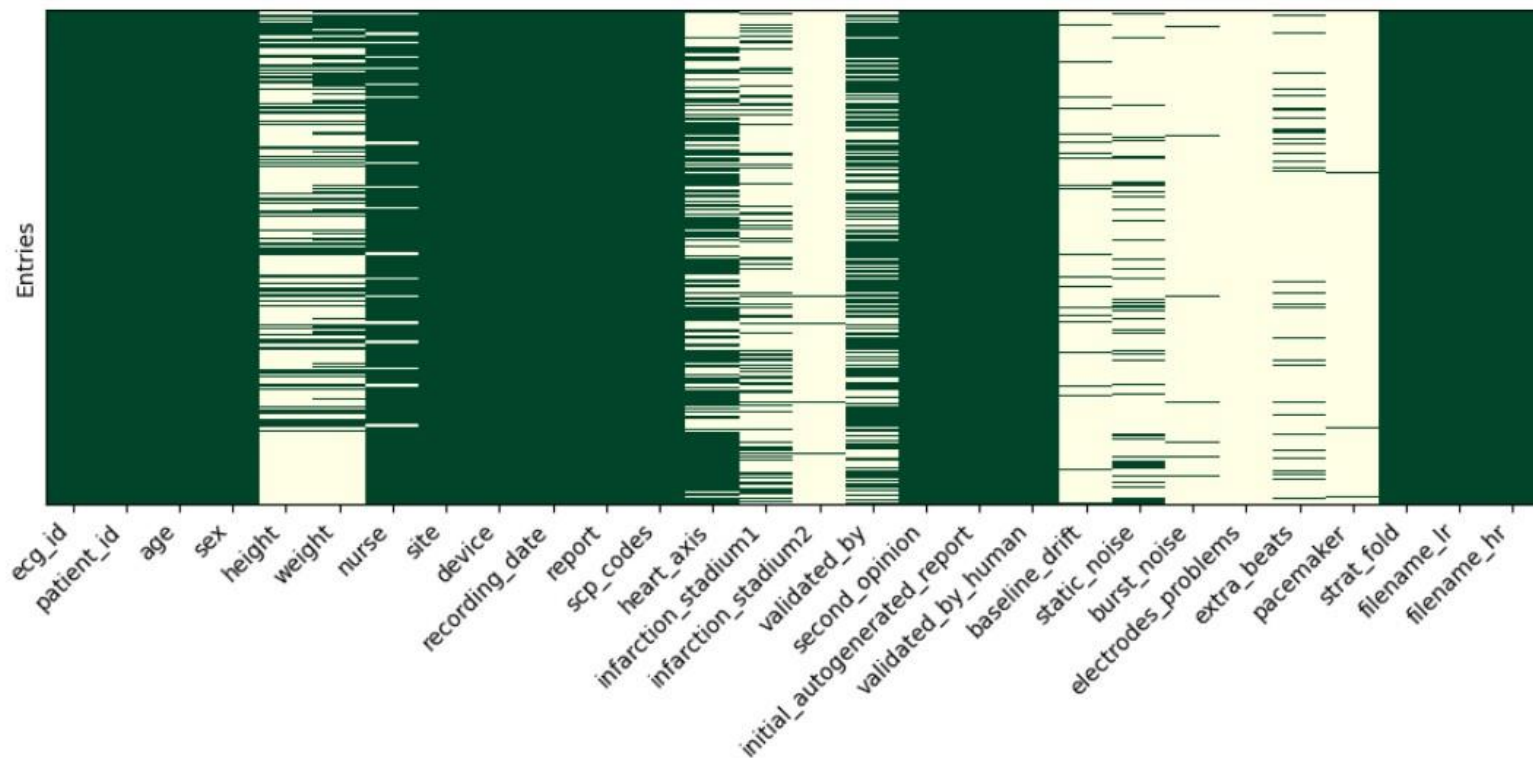
Distribucija broja snimaka po starosti i polu pacijenata. Grafikon prikazuje broj snimaka grupisanih prema godinama starosti za muškarce (plavo) i žene (žuto), pri čemu su najzastupljenije starosne grupe između 50 i 70 godina



Distribucija broja snimaka po starosti i polu pacijenata. Grafikon prikazuje broj snimaka grupisanih prema godinama starosti za muškarce (plavo) i žene (žuto), pri čemu je pacijentima kojima je dodijeljena vrijednost 300 za godine starosti sada dodijeljena prosječna vrijednost godina starosti.

Analiza i pretprocesiranje podataka

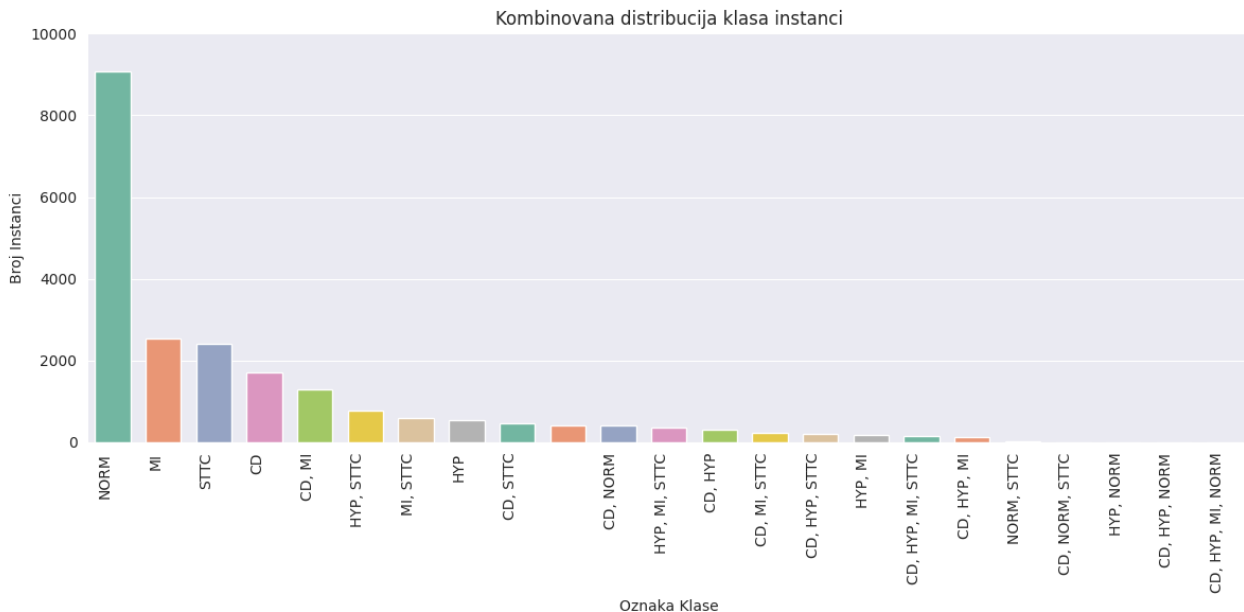
Gustina popunjenosti kolona tabele *ptbxxl_database* ilustrovana je na slici.



Analiza i pretprocesiranje podataka

Funkcija *aggregate_diagnostic* kombinuje oznake pojedinačnih signala dodijeljene na osnovu SCP izjava iz tabele *scp_statements*.

```
def aggregate_diagnostic(y_dic):  
    tmp = []  
    for key in y_dic.keys():  
        if key in agg_df.index:  
            tmp.append(agg_df.loc[key].diagnostic_subclass)  
    return list(set(tmp))
```



Grafički prikaz na slici jasno pokazuje da skup podataka nije izbalansiran (**neravnomjerna distribucija klasa**), tačnije klasa NORM je značajno najzastupljenija.

Filtriranje podataka

Svaki EKG snimak klasifikovan je u jednu ili više podklasa, pri čemu je svakoj podklasi dodijeljena određena vjerovatnoća, što čini jedan SCP kod ili oznaku iz tabele *scp_statements*.

Keywords	Weighting Factor (Confidence)
nicht auszuschliessen, cannot rule out, cannot be excluded	15%
möglicherweise, consider, suggest, likely	35%
wahrscheinlich, possible, maybe, probably, ablaufend, Verdacht auf	50%
Sonst, Bild	80%
Consistent with, Diagnose, Zustand nach...	100%

Izjave o vjerovatnoći dijagnostičkih tvrdnji, izvedene iz ključnih riječi u EKG izvještaju, kako je predstavljeno u EKG izjavama (scp_statements tabeli).

Značajan broj EKG snimaka označen je podklasama sa vjerovatnoćom od 0.0 ili neznatno višom.

- Podaci su filtrirani tako da se koriste samo klase označene sa vjerovatnoćom od 100%

Određene podklase su prisutne u malom broju instanci

- Uklonjene su podklase koje se pojavljuju u manje od 20 instanci

Na ovaj način je umjesto inicijalnih **21799** EKG snimaka, zadržano **17204** instanci.

Filtriranje signala

Butterworth bandpass filter – dizajniran je tako da propušta frekvencije unutar definisanog opsega, dok potiskuje frekvencije van tog opsega.

- Funkcija `butter_bandpass()` definiše parametre filtera određivanjem donje i gornje granične frekvencije (*lowcut* i *highcut*), zajedno sa frekvencijom uzorkovanja (*fs*).
- Funkcija `filter_signal()` primjenjuje ovaj filter na EKG podatke, čisteći signal od šuma ispod 0.5 Hz i iznad 50 Hz, što su tipične frekvencije za obradu EKG signala

```
def butter_bandpass(lowcut, highcut, fs, order):  
    nyquist = 0.5 * fs  
    low = lowcut / nyquist  
    high = highcut / nyquist  
    b, a = butter(order, [low, high], btype='band')  
    return b, a  
  
def filter_signal(data, fs, lowcut=0.5, highcut=50, order=5):  
    if highcut is None:  
        highcut = min(150, 0.8 * (fs / 2))  
    try:  
        b, a = butter_bandpass(lowcut, highcut, fs, order=order)  
        y = filtfilt(b, a, data)  
    except:  
        y = float('nan')  
    return y
```

Analiza i pretprocesiranje podataka



Grafički prikaz EKG signala označenog dijagnostičkom klasom NORM

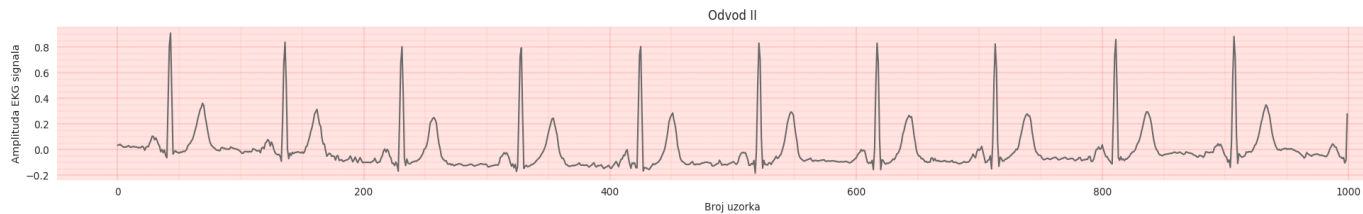


Grafički prikaz EKG signala označenog dijagnostičkom klasom MI

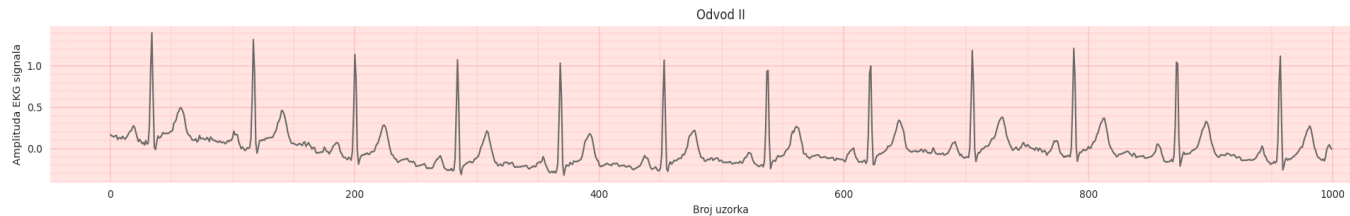
Analiza i preprocesiranje podataka



Grafički prikaz EKG signala označenog dijagnostičkom klasom HYP

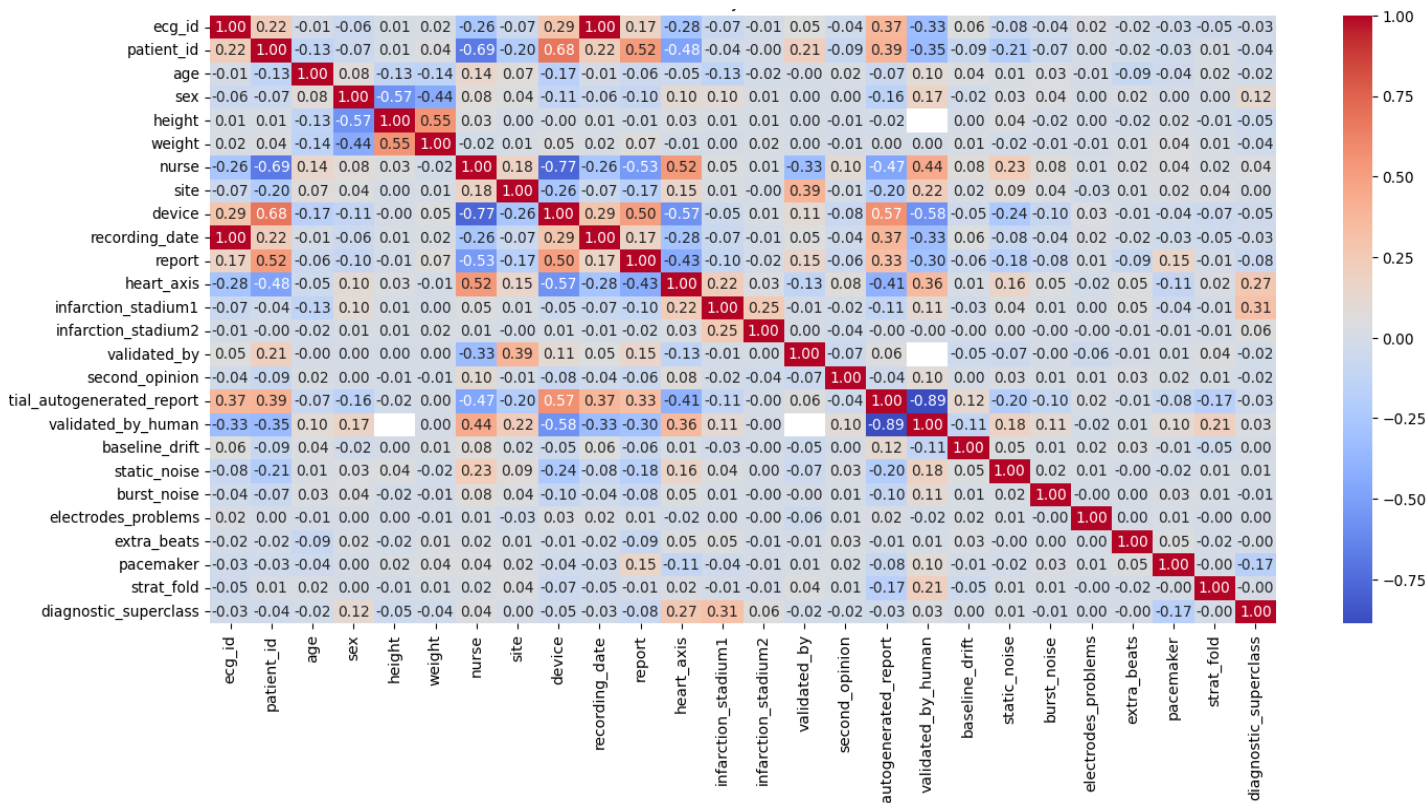


Grafički prikaz EKG signala označenog dijagnostičkom klasom STTC



Grafički prikaz EKG signala označenog dijagnostičkom klasom CD

Analiza i pretprocesiranje podataka

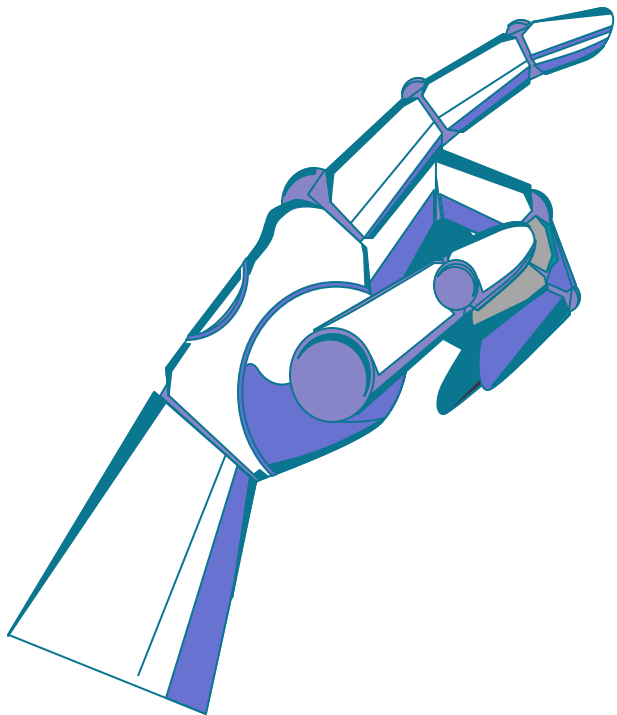


Matrica korelacije atributa tabele ptbxl_database

Analiza i pretprocesiranje podataka

- Uključivanje dodatnih atributa (na osnovu rezultata iz matrice korelacije) za poboljšanje performansi modela klasifikacije
- Labele i dodatni atributi, koji su u početku bili tipa `String`, enkodirani su u numerički tip podataka korišćenjem metode `LabelEncoder` iz biblioteke `sklearn.preprocessing`.
- Skupovi EKG snimaka i odgovarajuće labele podijeljeni su u tri skupa: za treniranje, testiranje i validaciju modela korišćenjem vrijednosti `strat_fold` iz tabele *ptbxxl_database*.
- Klase sa vrijednostima ovog parametra 10 i 9 su navedene kao najpreciznije utvrđene, pa se za validacioni skup koriste instance sa vrijednošću `strat_fold = 10`, a za testni skup one gdje je `strat_fold = 9`.

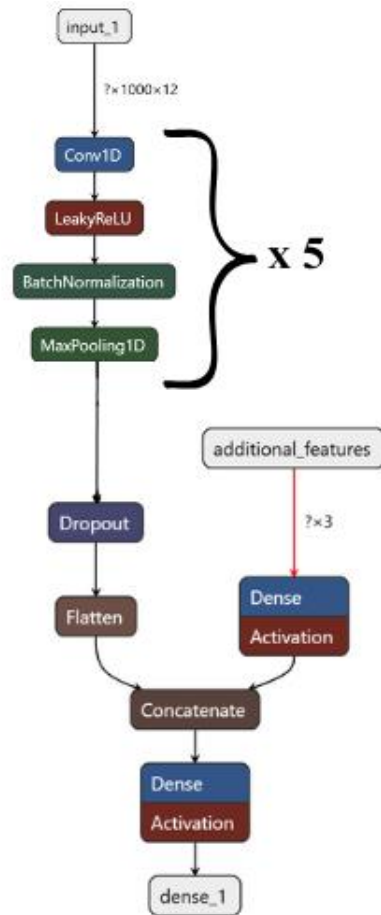
Klasifikacija EKG signala



- Konvolucione neuronske mreže, 1D CNN
- Rekurentne neuronske mreže, LSTM
- Objašnjiva vještačka inteligencija

Model dubokog učenja

- Arhitektura konvolucione neuronske mreže za binarnu klasifikaciju sa dodatnim atributima na ulazu (starost, srčana osovina, stadijum infarkta).
- Ulaz u mrežu (Input sloj) je 1D sekvenca sa oblikom (broj_uzoraka, broj_kanala) (`num_samp`, `num_leads`), gdje broj uzoraka predstavlja dužinu signala, koja zavisi od izabrane frekvencije (npr. 1000 za 100Hz ili 5000 za 500Hz), dok broj kanala označava broj izabranih kanala (*leads*) – u primjeru sa slike rađeno je sa svih 12 kanala.



Model konvolucione neuronske mreže za binarnu klasifikaciju

Mreža se sastoji iz dvije faze:

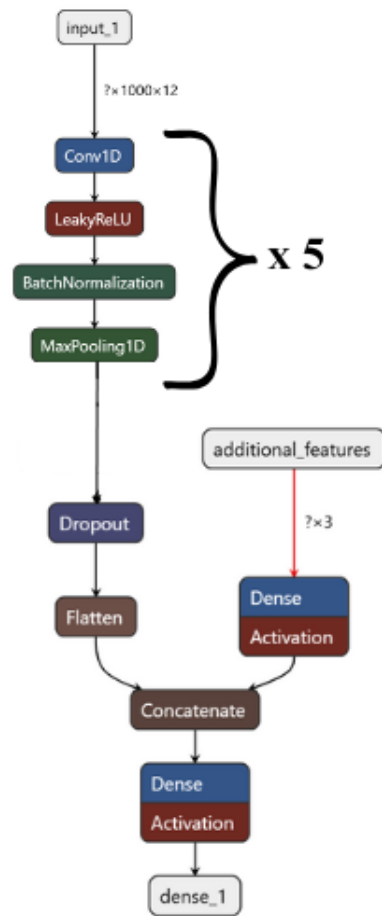
- konvolucionni blokovi za izdvajanje prostornih karakteristika i
- potpuno povezani slojevi za finalnu klasifikaciju.

Prvih pet slojeva modela su konvolucionni slojevi povezani sa slojevima za maksimizaciju (*MaxPooling*).

Konvolucionni slojevi primjenjuju jednodimenzionalne konvolucije sa hiperparametrom `stride=1`, što znači da se jezgro (*kernel*) pomjera za jedan uzorak po iteraciji. Jezgro veličine 5 služi za učenje prostornih obrazaca u signalima tokom procesa treniranja.

Za očuvanje integriteta nultog popunjavanja (*zero padding*) koristi se "same padding", što omogućava očuvanje dimenzija ulaza kroz sve slojeve.

Bias nije uključen u ove slojeve (`use_bias=False`), jer *batch* normalizacija preuzima ulogu korekcije.



Model konvolucione neuronske mreže za binarnu klasifikaciju

Da bi se smanjila dimenzionalnost ulaznog predstavljanja za polovinu nakon svakog konvolucionog sloja primjenjuje se *MaxPooling* sa filterom veličine 2 i bez preklapanja (`stride=2`).

- omogućava mreži da izdvoji relevantne karakteristike dok smanjuje rizik od prenaučenosti (*overfitting*).

Specifičnosti konvolucionih slojeva uključuju:

- Conv1D sloj 1: 32 filtera veličine jezgra 5;
- Conv1D sloj 2: 64 filtera veličine jezgra 5;
- Conv1D sloj 3: 64 filtera veličine jezgra 5;
- Conv1D sloj 4 i 5: 128 filtera veličine jezgra 5.

Svaki konvolucionni sloj je praćen *LeakyReLU* aktivacijom kako bi se izbjegao problem "mrtvih neurona" (engl. *dead neurons*), gdje svi izlazi neurona postaju nula.

```
input_layer = Input(shape=(num_samp, num_leads))

conv1 = Conv1D(filters=32, kernel_size=5, padding='same', use_bias=False)(input_layer)
leakyrelu1 = LeakyReLU()(conv1)
batch_norm1 = BatchNormalization()(leakyrelu1)
pooling1 = MaxPooling1D(pool_size=2)(batch_norm1)

conv2 = Conv1D(filters=64, kernel_size=5, padding='same', use_bias=False)(pooling1)
leakyrelu2 = LeakyReLU()(conv2)
batch_norm2 = BatchNormalization()(leakyrelu2)
pooling2 = MaxPooling1D(pool_size=2)(batch_norm2)

conv3 = Conv1D(filters=64, kernel_size=5, padding='same', use_bias=False)(pooling2)
leakyrelu3 = LeakyReLU()(conv3)
batch_norm3 = BatchNormalization()(leakyrelu3)
pooling3 = MaxPooling1D(pool_size=2)(batch_norm3)

conv4 = Conv1D(filters=128, kernel_size=5, padding='same', use_bias=False)(pooling3)
leakyrelu4 = LeakyReLU()(conv4)
batch_norm4 = BatchNormalization()(leakyrelu4)
pooling4 = MaxPooling1D(pool_size=2)(batch_norm4)

conv5 = Conv1D(filters=128, kernel_size=5, padding='same', use_bias=False)(pooling4)
leakyrelu5 = LeakyReLU()(conv5)
batch_norm5 = BatchNormalization()(leakyrelu5)
pooling5 = MaxPooling1D(pool_size=2)(batch_norm5)

dropout5 = Dropout(0.2)(pooling5)
flatten = Flatten()(dropout5)

dense = Dense(16, activation='relu')(flatten)
output_layer = Dense(1, activation='sigmoid')(dense)
model = Model(inputs=input_layer, outputs=output_layer)
```

Model konvolucione neuronske mreže za binarnu klasifikaciju

- **Batch normalizacija** primjenjuje se nakon svake *LeakyReLU* aktivacije, kako bi se stabilizovao proces treniranja i smanjila varijabilnost u podacima. Batch normalizacija uključuje parametre za pomjeranje koji se mogu učiti (*gamma*) i za skaliranje (*beta*), koji efektivno preuzimaju ulogu bias termina, čime se izbjegava redundantnost i ubrzava konvergencija.
- Kako bi se smanjio rizik od prevelikog uklapanja (*overfitting*), primijenjen je **Dropout** sloj nakon poslednjeg konvolucionog i *pooling* sloja sa stopom od 0.2 (20%).
- Potpuno povezani, **flatten**, sloj koji služi za pripremu karakteristika za finalnu fazu klasifikacije. *Flatten* slojevi uključuju:
 - *Dense* sloj (Gusto povezani sloj) sa 16 neurona i ReLU aktivacijom, koji služi za dodatnu ekstrakciju karakteristika;
 - Izlazni *Dense* sloj sa jednim neuronom i sigmoidnom aktivacijom za binarnu klasifikaciju, koja daje vjerovatnoću za binarnu klasifikaciju (0 ili 1) i ima samo jedan čvor na izlazu.

```
input_layer = Input(shape=(num_samp, num_leads))

conv1 = Conv1D(filters=32, kernel_size=5, padding='same', use_bias=False)(input_layer)
leakyrelu1 = LeakyReLU()(conv1)
batch_norm1 = BatchNormalization()(leakyrelu1)
pooling1 = MaxPooling1D(pool_size=2)(batch_norm1)

conv2 = Conv1D(filters=64, kernel_size=5, padding='same', use_bias=False)(pooling1)
leakyrelu2 = LeakyReLU()(conv2)
batch_norm2 = BatchNormalization()(leakyrelu2)
pooling2 = MaxPooling1D(pool_size=2)(batch_norm2)

conv3 = Conv1D(filters=64, kernel_size=5, padding='same', use_bias=False)(pooling2)
leakyrelu3 = LeakyReLU()(conv3)
batch_norm3 = BatchNormalization()(leakyrelu3)
pooling3 = MaxPooling1D(pool_size=2)(batch_norm3)

conv4 = Conv1D(filters=128, kernel_size=5, padding='same', use_bias=False)(pooling3)
leakyrelu4 = LeakyReLU()(conv4)
batch_norm4 = BatchNormalization()(leakyrelu4)
pooling4 = MaxPooling1D(pool_size=2)(batch_norm4)

conv5 = Conv1D(filters=128, kernel_size=5, padding='same', use_bias=False)(pooling4)
leakyrelu5 = LeakyReLU()(conv5)
batch_norm5 = BatchNormalization()(leakyrelu5)
pooling5 = MaxPooling1D(pool_size=2)(batch_norm5)

dropout5 = Dropout(0.2)(pooling5)
flatten = Flatten()(dropout5)

dense = Dense(16, activation='relu')(flatten)
output_layer = Dense(1, activation='sigmoid')(dense)
model = Model(inputs=input_layer, outputs=output_layer)
```

Rezultati

Metrike

Accuracy je mjera tačnosti koja se računa po formuli: $Acc = \frac{TN+TP}{TN+FP+FN+TP}$.

Metrika **f1-score** definiše se upotrebom pojmova:

- P – Preciznost (Precision): Odnos između pravih pozitivnih (TP) i zbira pravih pozitivnih i lažnih pozitivnih (FP): $P = \frac{TP}{TP+FP}$,
- R – Odziv (Recall) ili senzitivnost (sensitivity) - SN je odnos između pravih pozitivnih (TP) i zbira pravih pozitivnih i lažnih negativnih (FN): $SN = \frac{TP}{TP+FN}$.

F1 mjera je harmonijska sredina između preciznosti i odziva, balansirajući oba mjerenja, odnosno:

$$f1 - score = 2 * \frac{P*SN}{P+SN} = 2 * \frac{P*R}{P+R}.$$

Rezultati

Metrike

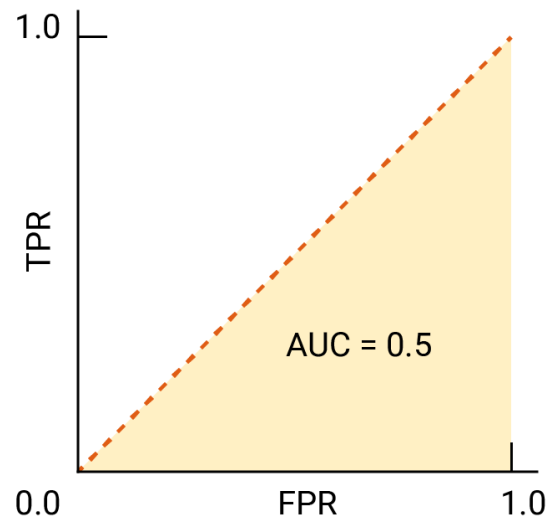
ROC kriva (Receiver Operating Characteristic Curve) prikazuje odnos između stope lažno pozitivnih (False Positive Rate, FPR) i stope pravih pozitivnih (True Positive Rate, TPR) za različite pragove (threshold) modela.

AUROC mjeri površinu ispod te krive (**Area Under the Receiver Operating Characteristic Curve**), gdje je vrijednost 1 idealan rezultat, a 0.5 označava slučajno pogađanje.

Formula za AUROC mjeru je integral TPR u odnosu na FPR

pri različitim pragovima:

$$AUROC = \int_0^1 TPR(FPR) d(FPR)$$



Rezultati

Klasifikacija NORM i DIAGNOSIS klasa

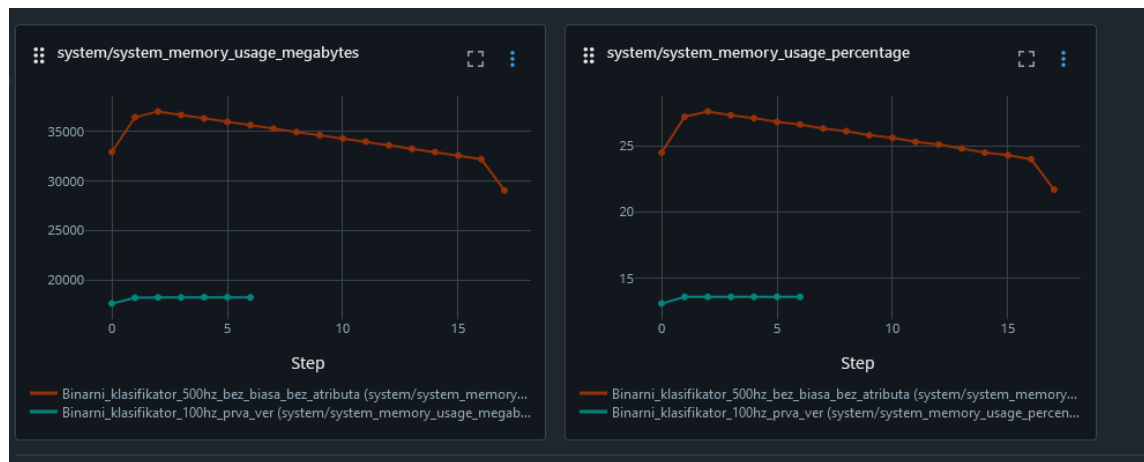
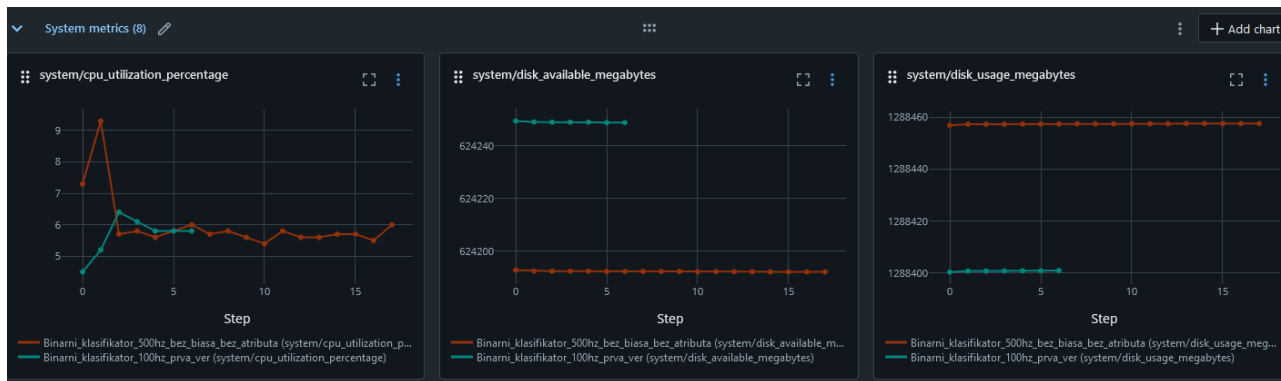
Klasa 0 je **NORM**, dok je klasa 1 **DIAGNOSIS** (kombinovane ostale dijagnostičke klase).

1. Korišćena je prva verzija modela (**Konvoluciona neuronska mreža sa jednim ulaznim slojem za sirove EKG signale**) i EKG signali sa frekvencijom od **100Hz** (`sampling_rate=100`). Performanse modela pokazuju F1-score od 89.71%, tačnost od 88.08% i AUROC score od 94.84%.
1. Korišćena je prva verzija modela (**Konvoluciona neuronska mreža sa jednim ulaznim slojem za sirove EKG signale**) i EKG signali sa frekvencijom od **500Hz** (`sampling_rate=100`). Performanse modela pokazuju F1-score od 88.80%, tačnost od 87.14% i AUROC score od 94.53%.

Zaključak: Na osnovu prikazanih rezultata, model koji koristi EKG signale sa frekvencijom uzorkovanja od 500Hz pokazuje slabije performanse u odnosu na model sa frekvencijom od 100Hz na svim metrikama procjene (F1-score, tačnost i AUROC score). Osim slabijih performansi, resursi potrebni za treniranje modela sa frekvencijom od 500Hz značajno su veći.

Rezultati

Klasifikacija NORM i DIAGNOSIS klasa



Rezultati

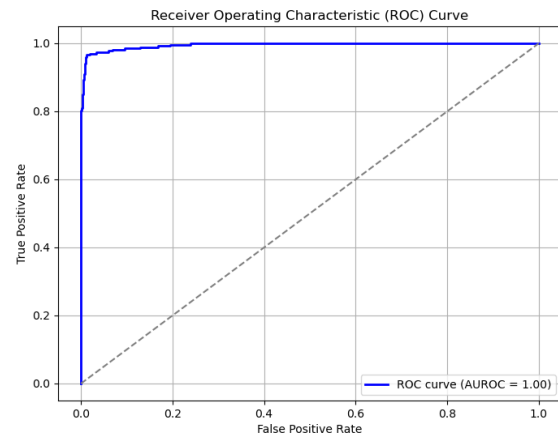
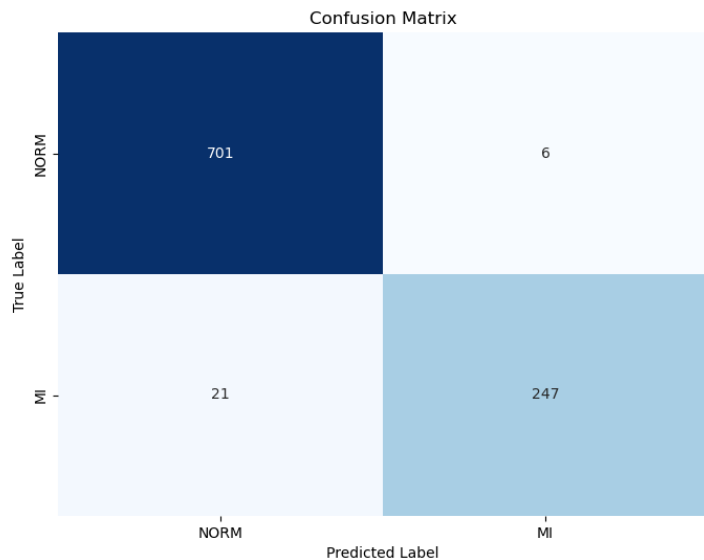
Klasifikacija NORM i DIAGNOSIS klasa

Metrike	NORM	MI	CD	HYP	STTC
Tačnost(Accuracy)	88.08	96.00	86.50	92.75	86.67
F1 score	89.71	92.60	74.75	42.06	74.27
AUROC score	94.84	99.34	90.16	85.67	91.45

Metrike	NORM	MI	CD	HYP	STTC
Tačnost(Accuracy)	87.14	97.23	85.16	91.58	86.71
F1 score	88.80	94.82	73.85	41.86	75.12
AUROC score	94.53	99.52	90.01	84.95	91.85

Rezultati

Klasifikacija MI i NORM klasa



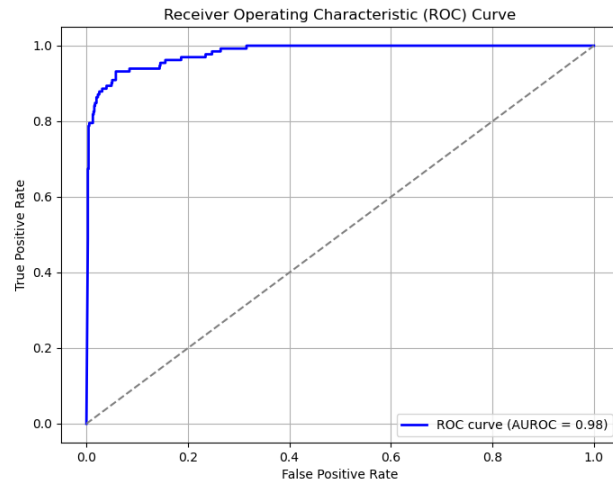
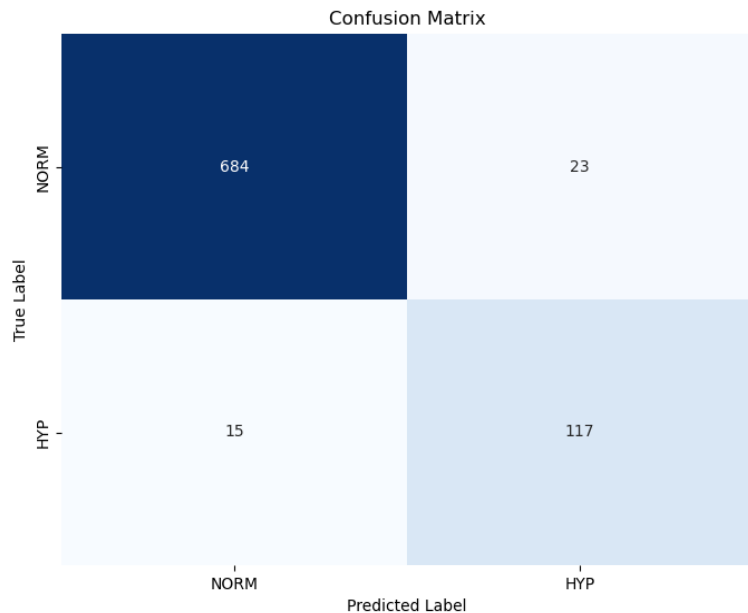
	precision	recall	f1-score	support
0	0.97	0.99	0.98	707
1	0.98	0.92	0.95	268
accuracy			0.97	975
macro avg	0.97	0.96	0.96	975
weighted avg	0.97	0.97	0.97	975

F1 Score: 94.82%
AUROC Score: 0.9952

Rezultati binarne klasifikacije MI-NORM. drugog modela, sa dodatnim ulazom za atribut.

Rezultati

Klasifikacija HYP i NORM klasa (undersampling)



	precision	recall	f1-score	support
0	0.98	0.97	0.97	707
1	0.84	0.89	0.86	132
accuracy			0.95	839
macro avg	0.91	0.93	0.92	839
weighted avg	0.96	0.95	0.96	839

Rezultati binarne klasifikacije HYP-NORM. drugog modela, sa dodatnim ulazom za atribut, nakon primjene undersampling tehnike.

AUROC Score: 0.9807
F1 Score: 86.03%

Model konvolucione neuronske mreže za multiklasifikaciju

- Upotreba ***softmax*** aktivacione funkcije na izlazu modela, nakon gustog (dense) sloja, kako bi model vratio vektor vjerovatnoća dužine broja klasa (5 klasa).

```
processed_additional_input = Input(shape=(3,), name='additional_features')
features_dense = Dense(8, activation='relu')(processed_additional_input)
concatenated = concatenate([flatten, features_dense])
dense = Dense(units=5)(concatenated)

output_layer = Softmax()(dense)
model = Model(inputs=[input_layer, processed_additional_input], outputs=output_layer)
```

- Za rješavanje problema višestrukog označavanja u višeklasnoj klasifikaciji, implementirani su prilagođeni težinski koeficijenti za klase, kao i prilagođena funkcija greške `weighted binary crossentropy`.
- Funkcija `calculate_label_weights()` koristi frekvenciju klasa u obuci kako bi odredila težinske koeficijente za svaku klasu, a zatim se te težine koriste u funkciji `weighted_binary_crossentropy()`.

Rezultati

Multiklasifikacija 5 klasa

Dijagnostičke klase korišćene u eksperimentu su 'CD', 'HYP', 'MI', 'NORM' i 'STTC', redom označene brojevima 0, 1, 2, 3 i 4.

U ovom eksperimentu korišćeni su EKG signali sa frekvencijom uzorkovanja od 100Hz.

	precision	recall	f1-score	support
0	0.81	0.53	0.64	483
1	0.64	0.46	0.54	132
2	0.77	0.63	0.69	268
3	0.85	0.87	0.86	721
4	0.75	0.55	0.64	456
micro avg	0.80	0.66	0.73	2060
macro avg	0.76	0.61	0.67	2060
weighted avg	0.79	0.66	0.72	2060
samples avg	0.80	0.72	0.75	2060

F1 Score: 71.79%

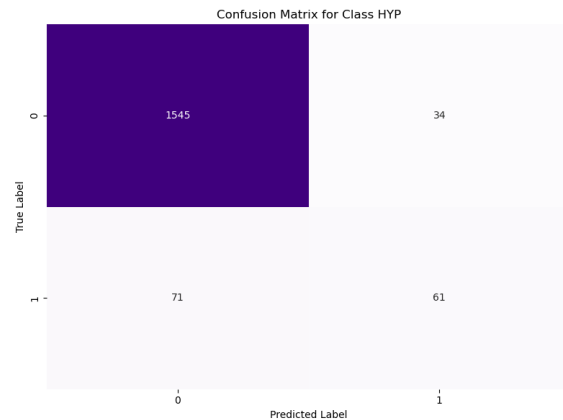
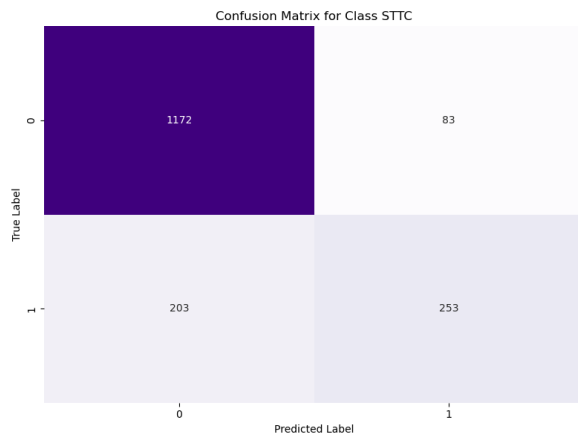
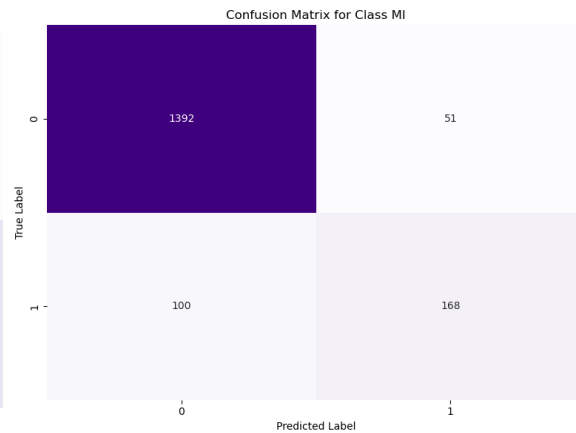
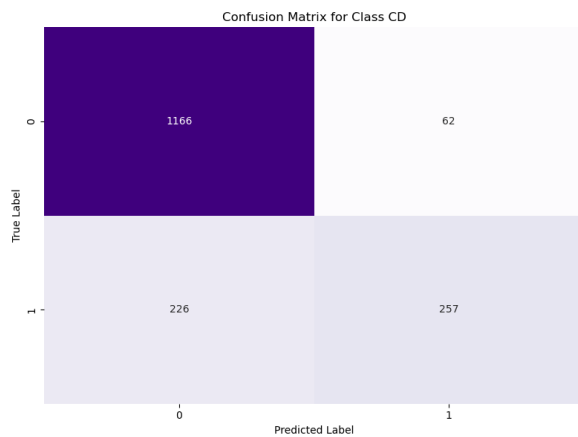
AUROC Score: 0.8654

Klasifikacioni izvještaj testiranja druge verzije modela, sa dodatnim ulazom za atribut, za problem klasifikacije sa više oznaka.

Model je testiranjem pokazao *F1-score* od 71.79%, tačnost od 80% i *AUROC* score od 86.54%. Analiza po klasama pokazuje da je klasa 3 (NORM) imala najbolji *F1-score* (87%), dok je klasa 1 (HYP) ostvarila najniži *F1-score* (54%) zbog niže preciznosti i odziva.

Rezultati

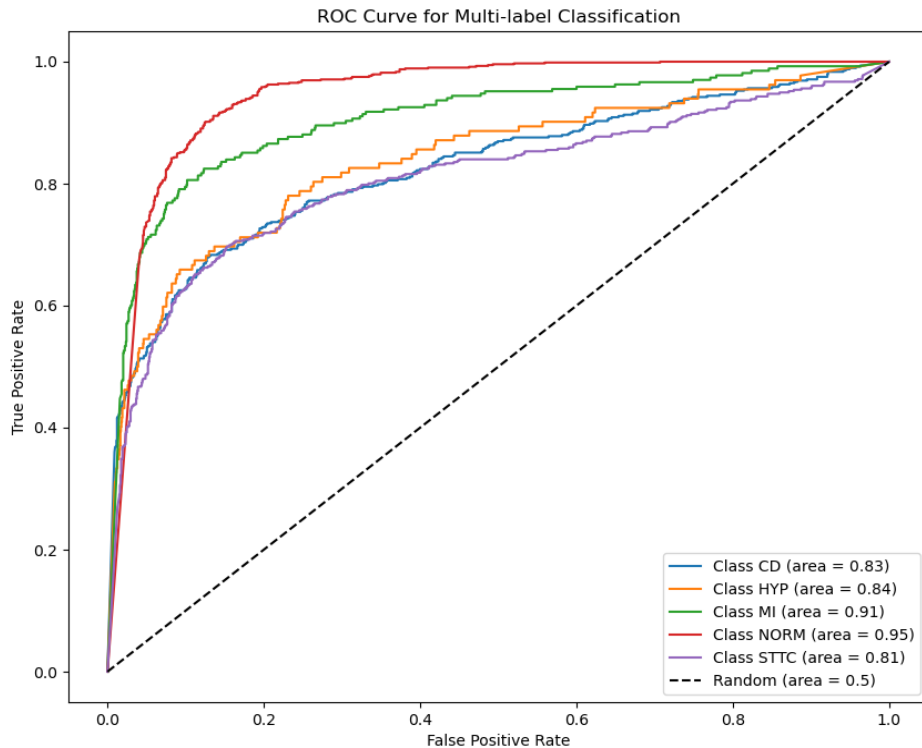
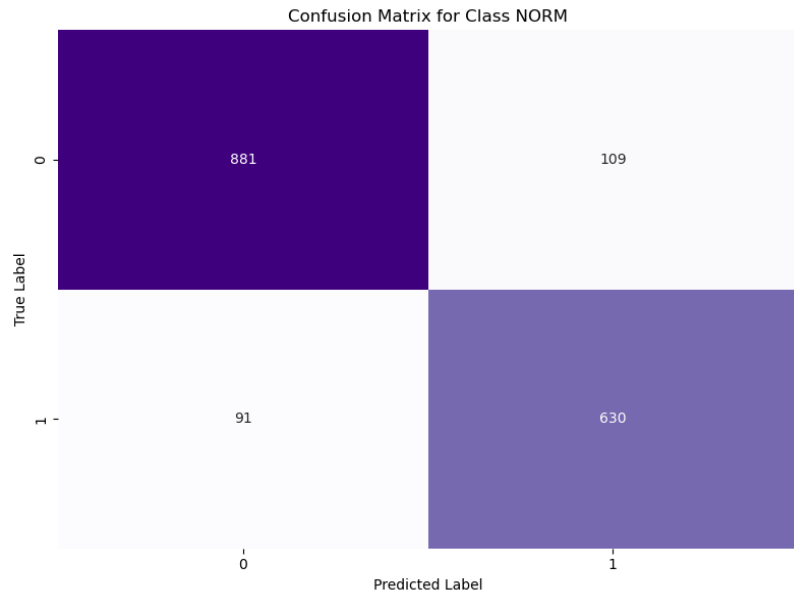
Multiklasifikacija



Prikaz metrika performansi druge verzije modela, sa dodatnim ulazom za atribute, za problem klasifikacije sa više oznaka.

Rezultati

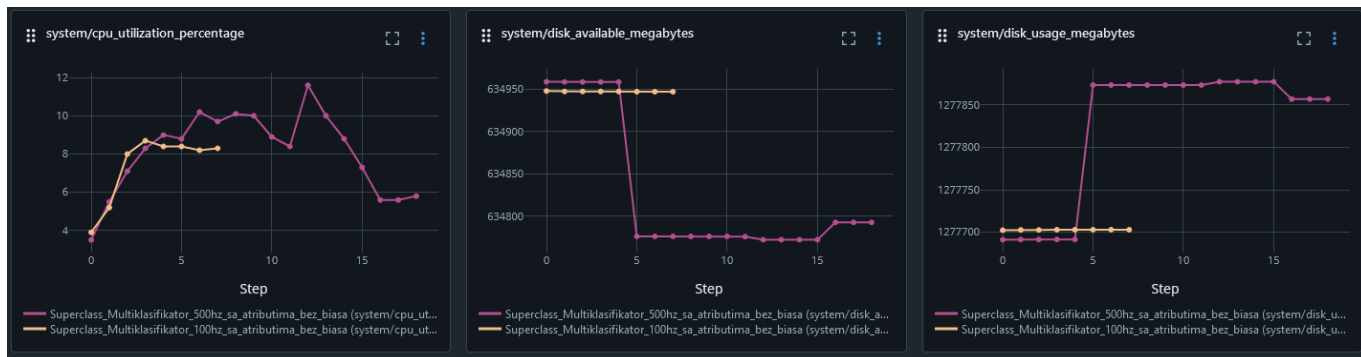
Multiklasifikacija



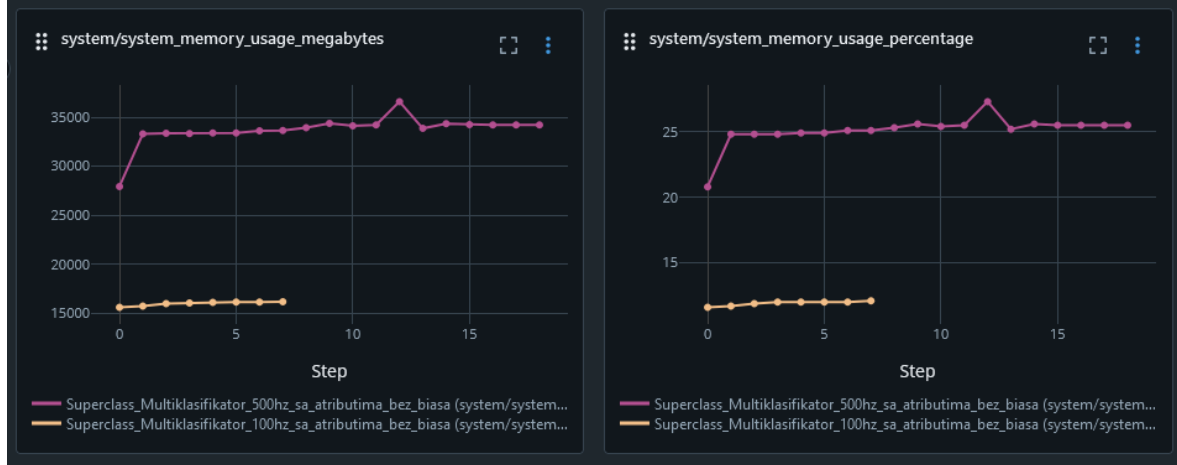
Prikaz metrika performansi druge verzije modela, sa dodatnim ulazom za atribute, za problem klasifikacije sa više oznaka.

Multiklasifikacija

Rezultati



Resursi potrebni za treniranje modela sa frekvencijom od 500Hz značajno su veći.

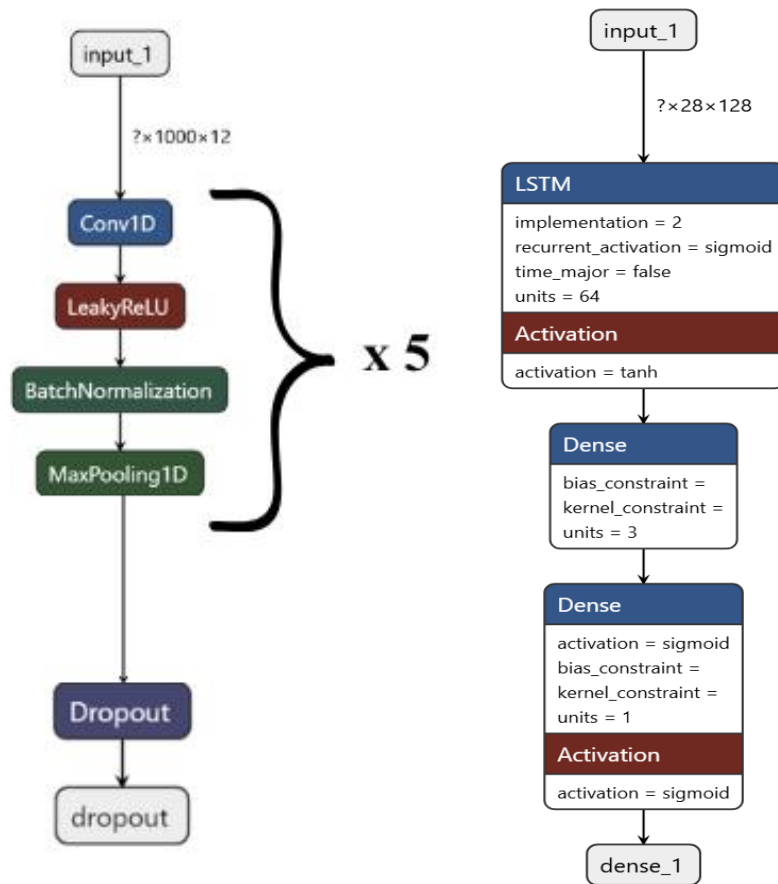


Metrike iskorišćenja resursa (CPU, memorija i disk) pokazuju da model sa frekvencijom od 500Hz troši daleko više procesorskih i memorijskih kapaciteta poređenju sa modelom koji koristi frekvenciju od 100Hz.

Roza linija predstavlja model treniran nad podacima sa frekvencijom od 500Hz, dok žuta linija predstavlja model treniran nad podacima sa frekvencijom od 100Hz. Svaki grafikon prikazuje različitu metriku sistemskih resursa: iskorišćenost CPU-a (%), dostupan prostor na disku (MB), korišćenje diska (MB), memoriju u megabajtima (MB) i procenat korišćenja memorije (%).

Model konvolucione neuronske mreže sa dodatnim LSTM slojem za binarnu klasifikaciju

- Arhitektura koja kombinuje model KNM za ekstrakciju karakteristika (lijevo) i LSTM sloj za klasifikaciju (desno).

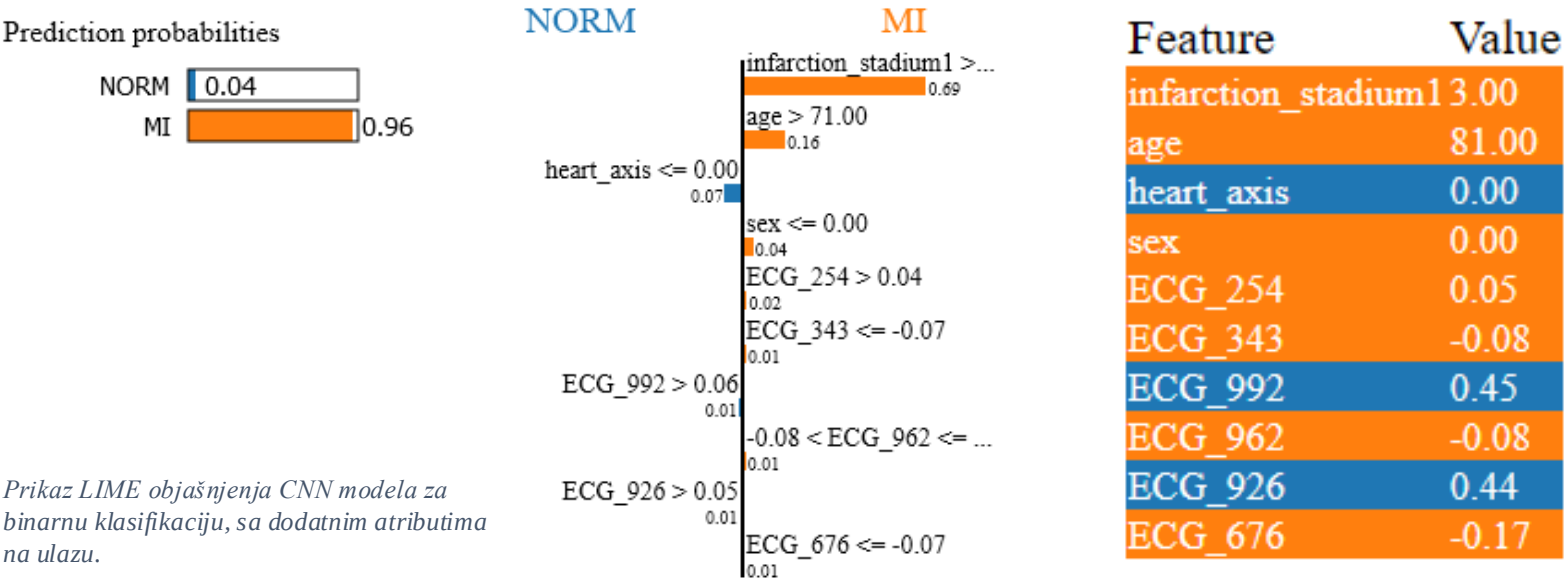


Odluka modela VI – objašnjivost

LIME

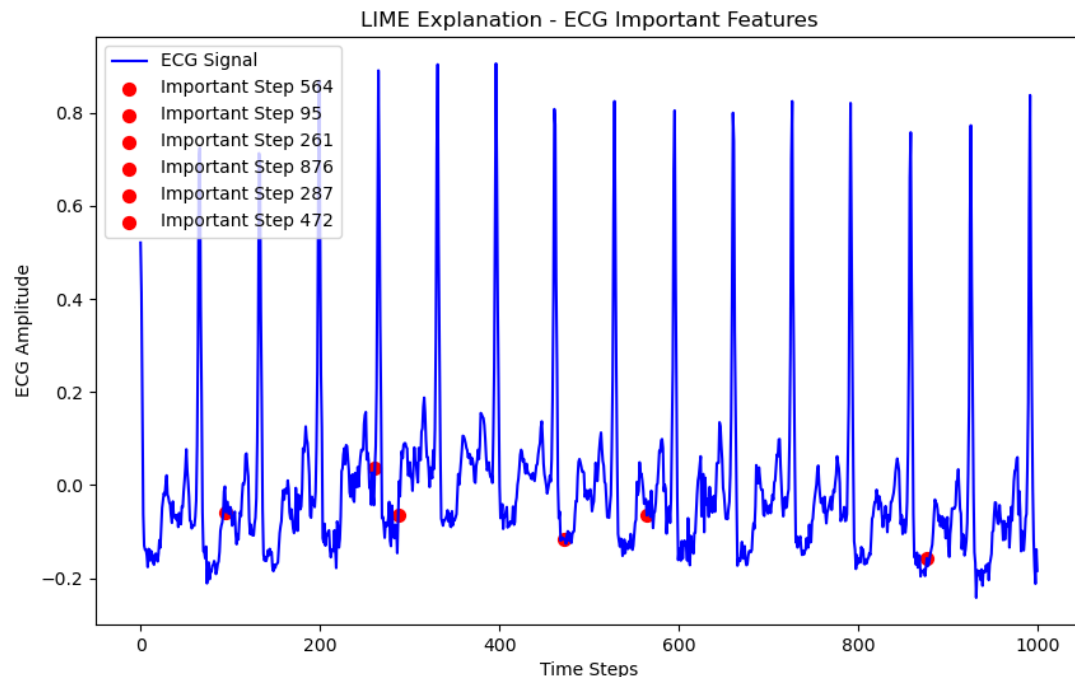
Local interpretable model-agnostic explanations (LIME) je oznaka za lokalna interpretabilna model-agnostička tumačenja.

Fokus je na lokalnim surogatnim modelima (jednostavniji, interpretabilniji modeli poput linearne regresije ili stabla odlučivanja), koji objašnjavaju predikcije za određeni podatak, umjesto da stvaraju globalni surogatni model. Surogatni modeli služe kao aproksimacije ponašanja osnovnog modela koji želimo objasniti.



Odluka modela VI – objašnjivost

LIME



Na slici tablični LIME prikaz pokazuje kako su dodatni podaci i pojedinačne tačke EKG-a uticali na predikciju koja je procijenjena sa 96% vjerovatnoće za MI i 4% za NORM.

Ključne EKG tačke koje je LIME metod istakao označene su na signalu.

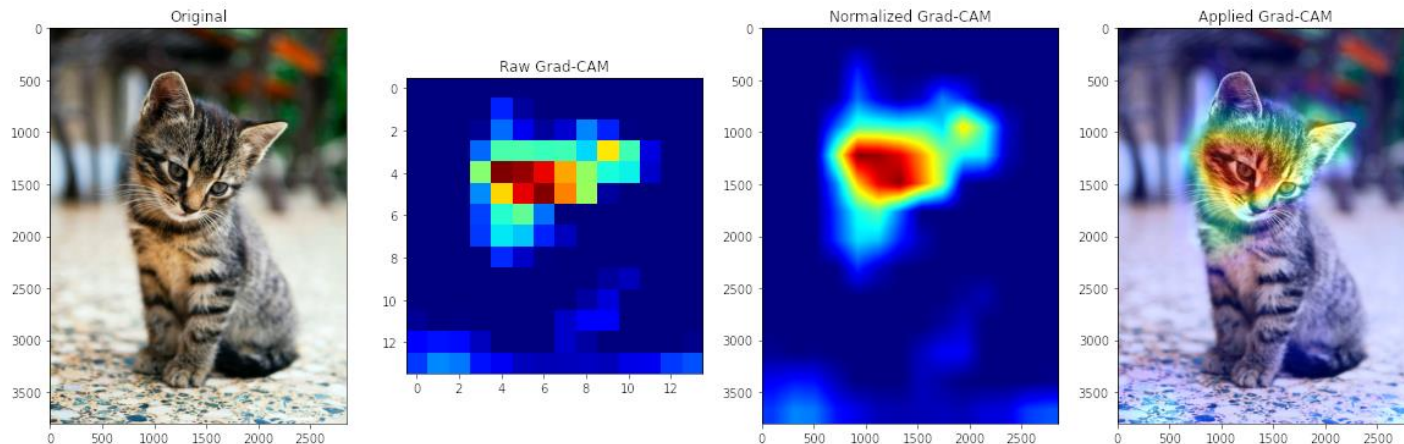
Prikaz LIME objašnjenja CNN modela za binarnu klasifikaciju, sa dodatnim atributima na ulazu.

Odluka modela VI – objašnjivost

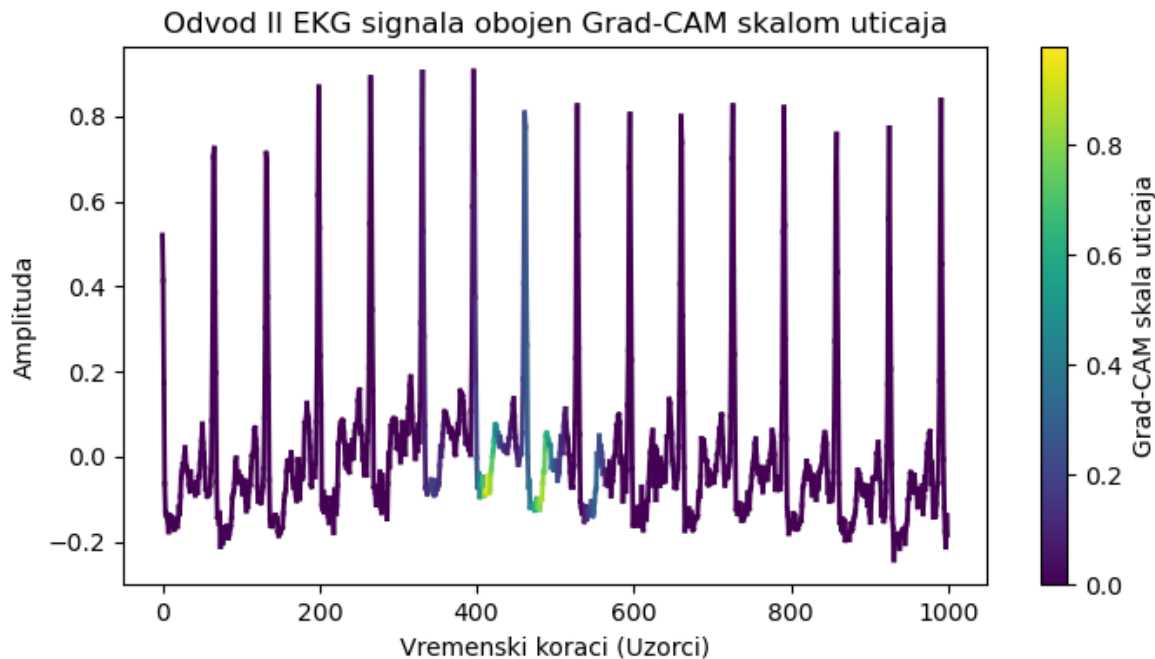
Grad-CAM

Grad-CAM (Gradient-weighted Class Activation Mapping) je tehnika za interpretaciju dubokih konvolucionih neuronskih mreža (CNN) koja pomaže vizualizaciji i razumijevanju odluka modela.

- Koristi informacije iz završnih konvolucionih slojeva mreže, gdje se pomoću gradijenata povezanih sa ciljanom klasom određuje koji dijelovi ulaznog podatka najviše doprinose predikciji.
- Grad-CAM izračunava srednje vrijednosti gradijenata kroz kanale završnog konvolucionog sloja kako bi dobio *heatmap*-u koja prikazuje najvažnije regije slike u odluci modela.



Odluka modela VI - objašnjivost



Prikaz rezultata primjene grad-CAM metoda za predikcije CNN binarnog klasifikatora za MI-NORM klase.

AIHeal

- Medicinska industrija, kardiologija
- Istraživanja iz oblasti vještačke inteligencije
- Hardver
- Mobilna aplikacija
- UCG / OneAI

Hardver - senzori

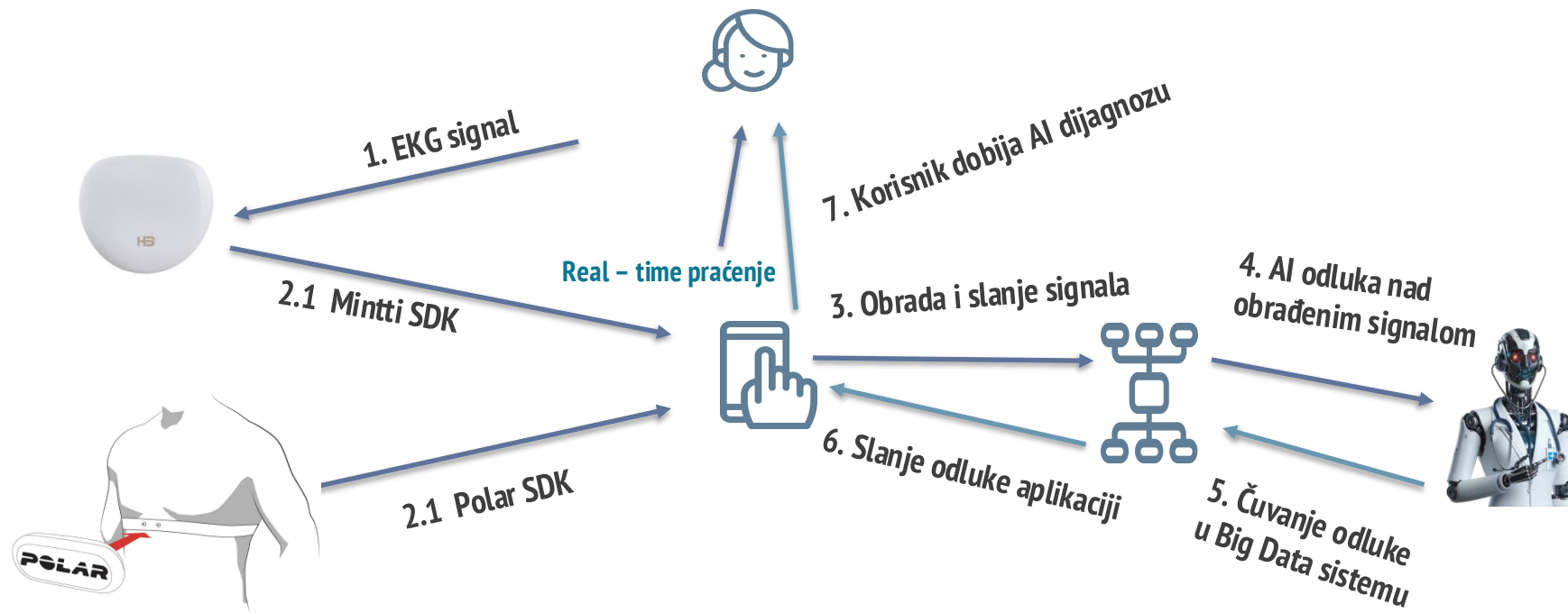


- Signal
- Broj otkucaja
- SDK - Kotlin



- Signal
- Broj otkucaja
- Položaj tijela
- Broj udisaja
- SDK - Java

AIHeal mobilna aplikacija



Hvala na pažnji!