Klasifikacija žanra glasbe z uporabo globokega učenja

***Abstract*—V nalogi smo implementirali program s preprostim grafičnim vmesnikom za klasifikacijo žanra glasbe. V ta namen smo zgradili tri modele različnih vrst nevronskih mrež, jih primerjali in najboljšega uporabili v programu. Izbran model se je izkazal za dokaj uspešnega in na testnih vzorcih dosegal 87 odstotno uspešnost klasifikacije.**

I. Uvod

Uporaba umetne inteligence, specifično globokega učenja, postaja vedno bolj pogosta. Ena izmed možnosti uporabe, s katero se ukvarjamo v nadaljevanju, je analiza zvoka. Eden izmed problemov s področja zvočne analize je klasifikacija žanra glasbe. Tega želimo rešiti z uporabo modelov nevronskih mrež. V ta namen izdelamo program, ki nam poljuben zvočni posnetek analizira in klasificira njegov žanr.

Prvo sledi pregled področja. V osrednjem poglavju predstavimo pripravo podatkov, grajenje modelov in njihovo optimizacijo, analizo rezultatov in implementacijo programa. Na koncu sledi zaključek.

II. Pregled področja

Med raziskanimi primeri klasifikacije žanra glasbe z uporabo strojnega učenja prevladuje uporaba globokega učenja. Najboljše rezultate dosegajo modeli konvolucijskih nevronskih mrež, kar se je izkazalo tudi v našem primeru. Podatke za učenje modelov se v veliki večini preoblikuje v obliko Mel frekvenčnih kepstralnih koeficientov[4]. Tak pristop smo uporabili tudi mi.

III. Metode

Reševanje problema je potekalo v štirih delih. Pri vseh je bil uporabljen programski jezik Python. Prvo smo izbrano zbirko podatkov pripravili za uporabo v modelih. Sledilo je grajenje in optimizacija modelov izbranih nevronskih mrež. Nato smo dobljene modele analizirali in izbrali najboljšega. Na koncu je sledila priprava programa z grafičnim vmesnikom.

A. Priprava podatkov

Izbrana podatkovna zbirka, GTZAN[1], vsebuje 1000 zvočnih posnetkov dolžine 30 sekund v formatu .wav. Enakomerno so porazdeljeni med 10 žanrov glasbe: blues, classical, country, disco, hiphop, jazz, metal, pop, reggae ter rock.

Posnetke smo z uporabo knjižnice Librosa[3], namenjene zvočni analizi, preoblikovali v Mel frekvenčne kepstralne koeficiente (angl. Mel-frequency cepstral coefficients (MFCCs)). Uporabljenih je bilo 13 koeficientov. Med grajenjem in analizo modelov se je izkazalo, da je 1000 vzorcev premalo za dobro klasifikacijo. Vsak posnetek smo razdelili na več krajših. Poizkusili smo z delitvijo na 3 in 10 delov. Slednja delitev se je izkazala za najboljšo. Množico 10000 posnetkov smo prvo razdelili na učno in testno množico v razmerju 3:1. Nato smo učni odvzeli petino vzorcev za potrebe validacije.

Dobljene vzorce smo s pripadajočimi oznakimi shranili v skupno datoteko oblike json.

B. Grajenje modelov

Pri grajenju modelov smo uporabili tri različne nevronske mreže. Uporabljena je bila knjižnica TensorFlow[7] s pripadajočo knjižnico Keras[2]. Optimizacija modelov je potekala iterativno s spreminjanjem števila nivojev, njihovih parametrov in oblike vhodnih podatkov (število segmentov vsakega zvočnega posnetka). V nadaljevanju so predstavljene končne oblike modelov. Pri vseh modelih je bila uporabljena optimizacijska metoda Adam s stopnjo učenja 0.0001.[9], [8]

1) Preprosta nevronska mreža: Prvi model je bil zgrajen s preprosto naprej povezano nevronsko mrežo iz šestih gosto povezanih nivojev. Prvi nivo vsebuje 1024 izhodov, drugi 512, tretji 256, četrti 128 in peti 64. Pri vseh je bila uporabljena aktivacijska funkcija ReLU (Rectified Linear Unit). Šesti nivo vsebuje toliko izhodov, kot imamo žanrov glasbe, torej 10. Uporabljena je bila aktivacijska funkcija softmax.

Pri grajenju modela je prihajalo do pretiranega prilagajanja (angl. overfitting), zato smo uporabili dve tehniki za preprečevanje. Prva je bila naključno odstranjevanje nevronov (angl. Dropout). Najboljše rezultate je model dosegal pri 20 odstotni verjetnosti. Druga tehnika je bila regularizacija tipa L2. Pri tem se je vrednost 0.01 izkazala za najprimernejšo. Obe tehniki sta bili uporabljeni na prvih petih nivojih.

2) Konvolucijska nevronska mreža: Drugi model je sestavljen iz treh konvolucijskih in dveh gosto povezanih nivojev. Prvi in drugi konvolucijski nivo uporabljata 64 in 128 jeder dimenzije 3 x 3. Tretji uporablja 256 jeder dimenzije 2 x 2. Prvi gosto povezan nivo vsebuje 64 izhodov, drugi 10. Pri prvih štirih nivojih je uporabljena aktivacijska funkcija ReLU, pri zadnjem ponovno softmax.

Pri tem modelu sta bili prav tako uporabljeni dve tehniki za preprečevanje pretiranjega prilagajanja. Pristotni sta na četrtem nivoju. Pri naključnem odstranjevanju nevronov je bila ponovna uporabljena 20 odstotna verjetnost. Pri regularizaciji tipa L2 se je vrednost 0.1 izkazala za primernejšo.

3) Rekurentna nevronska mreža: Tretji model predstavlja posebna vrsta rekurentnih nevronskih mrež, LSTM (Long short-term memory). Sestavljen je iz treh LSTM in dveh gosto povezanih nivojev. Prvi LSTM nivo vsebuje 256 enot, drugi

128 in tretji 64. Prvi gosto povezan nivo vsebuje 64 izhodov, drugi 10. Pri prvih dveh gosto povezanih nivojih je uporabljena akcivacijska funkcija ReLU, pri zadnjem softmax.

Za preprečevanje pretiranega prilagajanja je na prvem gosto povezanem nivoju dodano naključno odstranjevanje nevronov z 20 odstotno verjetnostjo.

C. Analiza modelov

Modeli so bili prvo primerjani glede na natančnost klasifikacije testne množice.

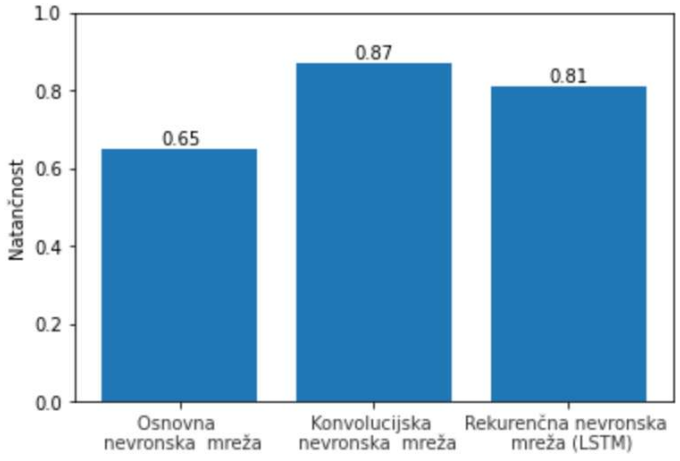

Fig. 1. Natančnost klasifikacije modelov

Glede na prvi kriterij sta se modela konvolucijskih in rekurenčnih nevronskih mrež izkazala za znatno boljša, pri čemer se prvi obnese najbolje.

Nato smo modele primerjali še z uporabo ROC krivulj in AUC metrike.

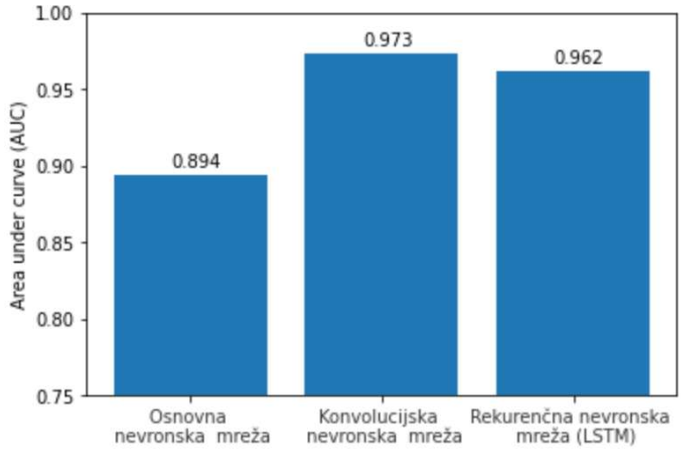

Fig. 2. AUC vrednosti modelov

Modela konvolucijskih in rekurenčnenih nevronskih mrež sta tudi glede na drugi kriterij najuspešneša. Glede na obe primerjavi modelov, se je model konvolucijske nevronske mreže izkazal za najuspešnejšega in bil izbran za uporabo v programu za klasifikacijo žanra glasbe.

Pri primerjavi ROC krivulj posameznih modelov se je izkazalo da imajo modeli v povprečju največ težav s klasifikacijo country in rock glasbe, najmanj pa s klasifikacijo klasične in metal glasbe.

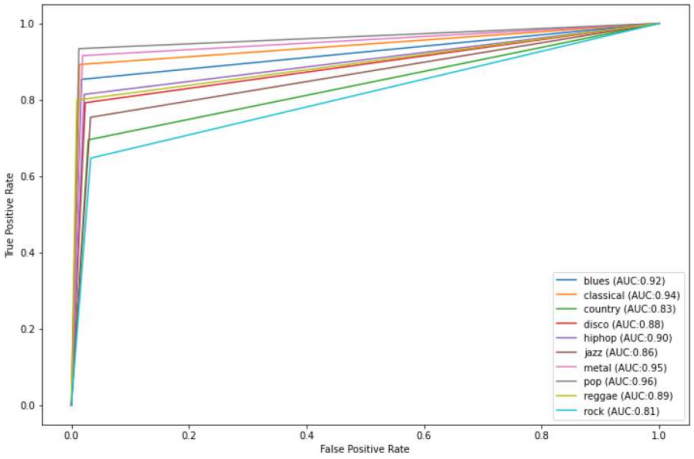

Fig. 3. ROC krivulje modela preproste nevronske mreže

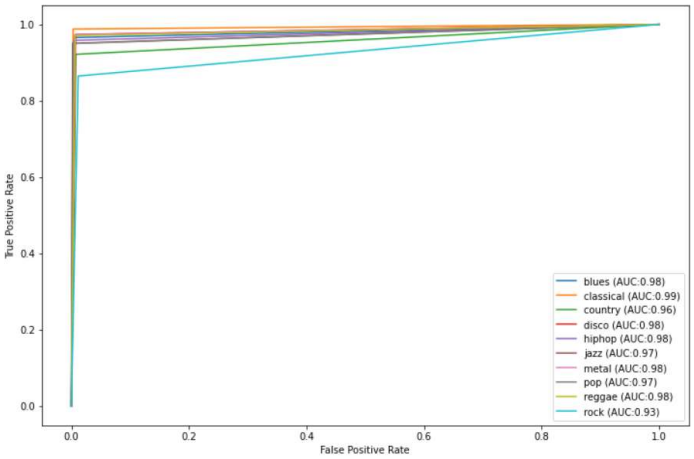

Fig. 4. ROC krivulje modela konvolucijske nevronske mreže

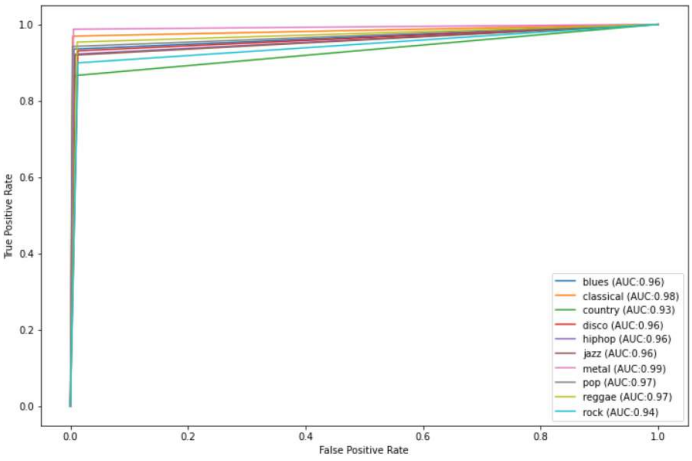

Fig. 5. ROC krivulje modela rekurenčne nevronske mreže

D. Možne izboljšave

Preizkušenih je bilo kar nekaj različic predstavljenih modelov a obstaja še veliko možnih kombinacij raznih nivojev in njihovih parametrov. Učna množica, ki smo jo izbrali se je že kmalu izkazala za premajhno in jo je bilo potrebno razdeliti na več krajših vzorcev. Možna izboljšava bi bila uporaba večje zbirke zvočnih posnetkov, ki bi lahko bili lahko daljši.

E. Program

Program za klasifikacijo žanra glasbe je bil prav tako narejen v programskem jeziku Python. Za grafični vmesnik je bila uporabljena knjižnica PySimpleGUI[6]. Program uporabniku omogoča izbiro poljubnega zvočnega posnetka. Ta je s pomočjo knjižnice Librosa razdeljen na segmente dolžine 3 sekunde. Ti so preoblikovani na enak način kot vzorci zbirke v prvem koraku. Program nato s pomočjo naloženega modela izvede klasifikacijo segmentov posnetka. Verjetnosti žanrov za vsak segment so nato seštete in deljene z številom segmentov, da dobimo povprečje za celoten posnetek. Uporabniku se izpiše klasificiran žanr in prikaže graf s tremi najverjetnimi žanri. Program je bil v izvršljivo datoteko zapakiran s prevajalnikom Nuitka[5]. Izbran model se mora nahajati v enakem direktoriju kot program, v svojem direktoriju z imenom model.

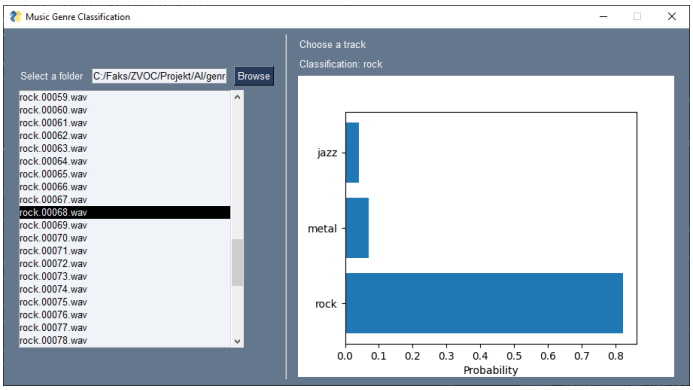

Fig. 6. Grafični vmesnik programa

IV. Zaključek

V nalogi smo reševali problem klasifikacije žanra glasbe. Cilj naloge je bilo ustvariti program z grafičnim vmesnikom za klasifikacijo poljubne glasbe. V ta namen smo uporabili metode globokega učenja in zgradili tri modele globokih nevronskih mrež. Primerjali smo jih glede na natančnost klasifikacije ter z uporabo ROC krivulj in AUC metrike. Med modeli se je najbolje izkazal model konvolucijske nevronske mreže. Dosegel je 87 odstotno natačnost klasifikacije in AUC vrednost 0.973.

References

[1] Gtzan genre collection. Dosegljivo: http://marsyas.info/downloads/datasets.html. [Dostopano: 02. 01. 2022].
[2] Keras: the python deep learning api. Dosegljivo: https://keras.io/. [Dostopano: 02. 01. 2022].
[3] Librosa : audio and music processing in python. Dosegljivo: https://librosa.org/. [Dostopano: 02. 01. 2022].
[4] Mel-frequency cepstrum - wikipedia. Dosegljivo: https://en.wikipedia.org/wiki/Mel-frequency$_cepstrum$. [$Dostopano$: 02.01.2022].
[5] Nuitka the python compiler. https://nuitka.net/.
[6] Pysimplegui. Dosegljivo: https://PySimpleGUI.org. [Dostopano: 02. 01. 2022].
[7] Tensorflow. Dosegljivo: https://www.tensorflow.org/. [Dostopano: 02. 01. 2022].
[8] Athulya K M and Sindhu S. Deep learning based music genre classification using spectrogram (july 10, 2021). proceedings of the international conference on iot based control networks intelligent systems - icicnis 2021. Dosegljivo: https://papers.ssrn.com/sol3/papers.cfm?abstract$_id$ = 3883911. [$Dostopano$: 28.11.2021].
[9] Marc Saint-Félix. Music genre detection with deep learning. Dosegljivo: https://towardsdatascience.com/music-genre-detection-with-deep-learning-cf89e4cb2ecc. [Dostopano: 02. 01. 2022].