Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Nina Mrzelj

Globoko učenje na genomskih in filogenetskih podatkih

diplomsko delo

interdisciplinarni univerzitetni študijski program prve stopnje računalništvo in matematika

Mentor: prof. dr. Blaž Zupan

Ljubljana, 2016

Besedilo je oblikovano z urejevalnikom besedil LaTeX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Globoko učenje na genomskih in filogenetskih podatkih

Preučite uporabo tehnik globokega učenja v namene klasifikacije genskih zaporedij. V eksperimentalni študiji uporabite filogenetske razvrstitve bakterij in njihov referenčni del genoma. S tehnikami strojnega učenja zgradite modele, ki iz zaporedja nukleotidov v referenčnem delu DNA napovejo filogenetsko razvrstitev. Med seboj primerjajte napovedne uspešnosti konvolucijskih in rekurenčnih nevronskih mrež ter klasičnih postopkov strojnega učenja.

Izjava o avtorstvu zaključnega dela

Spodaj podpisana Nina Mrzelj, vpisna številka 63120248, avtorica pisnega zaključnega dela študija z naslovom:

Globoko učenje na genomskih in filogenetskih podatkih

IZJAVLJAM

1. da sem pisno zaključno delo študija izdelala samostojno pod mentorstvom prof. dr. Blaža Zupana;
2. da je tiskana oblika pisnega zaključnega dela študija istovetna elektronski obliki pisnega zaključnega dela študija;
3. da sem pridobila vsa potrebna dovoljenja za uporabo podatkov in avtorskih del v pisnem zaključnem delu študija in jih v pisnem zaključnem delu študija jasno označila;
4. da sem pri pripravi pisnega zaključnega dela študija ravnala v skladu z etičnimi načeli in, kjer je to potrebno, za raziskavo pridobila soglasje etične komisije;
5. soglašam, da se elektronska oblika pisnega zaključnega dela študija uporabi za preverjanje podobnosti vsebine z drugimi deli s programsko opremo za preverjanje podobnosti vsebine, ki je povezana s študijskim informacijskim sistemom članice;
6. da na UL neodplačno, neizključno, prostorsko in časovno neomejeno prenašam pravico shranitve avtorskega dela v elektronski obliki, pravico reproduciranja ter pravico dajanja pisnega zaključnega dela študija na voljo javnosti na svetovnem spletu preko Repozitorija UL;
7. dovoljujem objavo svojih osebnih podatkov, ki so navedeni v pisnem zaključnem delu študija in tej izjavi, skupaj z objavo pisnega zaključnega dela študija.

V Ljubljani, dne 8. septembra 2016 Podpis študentke:

Rada bi se zahvalila naslednjim osebam, ki so pomagale pri nastanku tega diplomskega dela ali pa so drugače zaslužne, da so bila moja študijska leta lepša:

- profesorju Blažu Zupanu - ker ste bili odzivni in vedno na voljo, ko sem potrebovala nasvete in komentarje pri pisanju diplomskega dela in ker ste ravno vi tisti, ki ste me s svojimi predavanji navdušili za področje strojnega učenja in analizo podatkov.

- Niku Colneriču - ker si mi bil s svojim znanjem o globokem učenju v veliko pomoč.

- mami in atiju - ker me moralno in finančno podpirata, me motivirata in sta mi vedno na voljo, ko vaju potrebujem. Hvala, ker verjameta vame; brez vaju ne bi bila tu, kjer sem.

- Niki - ker mi kupuješ letalske karte in mi sredi noči s svojimi idejami pomagaš pri reševanju problemov.

- Ernestu, Mixi, Darji, Iztoku, Matevžu, Nedimu, Evi, Žigu, Kristjanu, Luciju - ker ste poskrbeli, da študij ni bil le študij, pač pa tudi smeh in zabava.

- Primožu - ker me imaš rad tudi, ko sem tečna.

Mami.

Kazalo

Povzetek

Naslov: Globoko učenje na genomskih in filogenetskih podatkih

Metode globokega učenja v praksi dosegajo izjemne rezultate pri reševanju problemov na različnih področjih, med drugim tudi v genomiki. V diplomski nalogi smo se ukvarjali z razvrščanjem genskih zaporedij bakterij v taksonomske razrede. Cilj je bil zgraditi model, ki bo znal bakterijo na podlagi zaporedja njenega gena 16S rRNA razvrstiti v pravo deblo, razred, red, družino in rod. Z uporabo metod globokega učenja smo zgradili več klasifikacijskih modelov in vrednotili njihovo uspešnost na podlagi klasifikacijske točnosti in mere F1. Med seboj smo primerjali konvolucijske nevronske mreže, preproste rekurenčne nevronske mreže, dvosmerne rekurenčne nevronske mreže, kombinirane modele z rekurenčnimi in konvolucijskimi nevronskimi mrežami ter metodo naključnih gozdov. Eksperimente smo izvedli na dveh različno velikih množicah podatkov, preverili pa smo tudi, kako se modeli obnesejo pri klasifikaciji, če imajo na voljo le krajši del genskega zaporedja. Rezultati kažejo, da so za reševanje tovrstnih problemov najbolj primerne konvolucijske nevronske mreže.

Ključne besede: globoko učenje, klasifikacija, nevronske mreže.

Abstract

Title: Deep learning on genomic and phylogenetic data

Deep learning methods have been achieving amazing results in solving a variety of problems in many different fields, a very important one of them being genomics. In the thesis, deep learning methods have been used to classify bacterial DNA sequences into taxonomic ranks. The goal was to build a classification model based on the bacteria's 16S rRNA sequence and classify a bacteria by phylum, class, order, family and genus. The performance of five different models has been compared in terms of accuracy and F1 score. A model with convolutional neural networks, simple recurrent neural network, bidirectional neural network, a hybrid model that combines convolutional and neural network and a model using random forests have been built. Two experiments have been conducted. In the first one classification was based on the whole sequence. In the second one only a small sequence fragment was used. We evaluated the performance of the models based on two datasets of different sizes. Results show that convolutional neural networks outperformed other models in all the cases.

Keywords: deep learning, classification, neural networks.

Poglavje 1

Uvod

Genomika je relativno nova znanstvena disciplina, ki za raziskovalne namene uporablja zaporedje genoma. Genom je dedna informacija celotnega organizma, ki je zapisana v DNA (ang. deoxyribonucleic acid) ali pri nekaterih virusih v RNA (ang. ribonucleic acid). Leta 1990 sta Ari Patrinos (U.S. Department of Energy's Office of Science) in Francis Collins (National Institutes of Health, National Human Genome Research Institute) začela s projektom sekvenciranja človeškega genoma (ang. Human Genome Project - HGP) [2], katerega namen je bil pridobiti in razumeti celotno zaporedje človeške DNA. Poleg sekvenciranja človeškega genoma je ogromno narejenega tudi na področju odkrivanja zaporedij najrazličnejših organizmov. Pfeifferjev bacil (lat. Haemophilus influenzae) je bil prvi organizem iz narave, ki so mu leta 1995 določili zaporedje celotnega genoma [12]. Leta 1998 je bil objavljen genom prve živali, gliste Caenorhabditis elegans [11], do leta 2000 sta bila objavljena genoma prvega insekta - vinske mušice (lat. Drosophila melanogaster) [4] in prve rastline - navadnega repnjakovca (lat. Arabidopsis thaliana) [14]. Leta 2002 je bil prvič objavljen genom laboratorijske miši (lat. Mus musculus) [5], leta 2003 pa se je končal projekt sekvenciranja človeškega genoma, ki je trajal celih 13 let. Takrat je bilo sekvenciranje genoma počasno in drago. Z razvojem nove generacije visoko zmogljivega vzporednega sekvenciranja DNA [18] so se čas in stroški postopka zmanjšali, zanimanje pa močno povečalo.

Z raziskovanjem genskega zaporedja se lahko veliko naučimo tako o razvoju znotraj vrste kot o razvoju med vrstami. Podatkov je vse več, prav tako pa tudi idej, kako jih lahko uporabimo. Tehnološki napredek je omogočil, da lahko na podlagi genskih zaporedij gradimo evolucijska drevesa in razvrščamo organizme v taksonomske nize, prepoznavamo gene, ugotavljamo odziv organizma na zdravila, odkrivamo nove gene, ki povzročajo dedne bolezni, in še veliko več [15].

Pri analizi bioloških podatkov lahko uporabljamo metode strojnega učenja. Omogočajo nam boljše razumevanje človeškega genoma pa tudi odkrivanje novih znanj in zakonitosti. Strojno učenje delimo na tri veje: nadzorovano učenje (ang. supervised learning), nenadzorovano učenje (ang. unsupervised learning) in vzpodbujevalno učenje (reinforcement learning) [22]. V diplomski nalogi se bomo ukvarjali z nadzorovanim učenjem, kjer se sistem na podlagi vhodnih podatkov nauči vnaprej določenega tipa odgovorov. Primer problema je napovedovanje preostanka časa življenja pacienta z rakom. Na voljo imamo podatke o vrsti, lokaciji in razširjenosti raka, starosti pacienta in možnih načinih zdravljenja, cilj pa je zgraditi napovedni model, ki bo na podlagi teh podatkov znal napovedati preostali čas življenja. Pojem nadzorovano učenje se nanaša na dejstvo, da sistemu podamo podatke, kjer poznamo "pravi odgovor". Drugačen primer nadzorovanega učenja je na podlagi medicinskih podatkov ugotoviti, ali je tumor maligen (nevaren) ali benigen (manj nevaren). Take tipe vprašanj rešujemo s tehniko strojnega učenja imenovano klasifikacija (ang. classification) oziroma razvrščanje v skupine. Glavna naloga algoritmov je na podlagi značilnosti primera le-tega uvrstiti v nek vnaprej določen razred. Za reševanje tega problema poznamo več tehnik strojnega učenja: odločitvena drevesa, metoda podpornih vektorjev, Bayesov klasifikator, naključni gozdovi, nevronske mreže in druge [16].

Področje umetne inteligence, ki obravnava večslojne nevronske mreže, se imenuje globoko učenje (ang. deep learning)[17]. Nevronske mreže (ang. neural networks) so sestavljene iz umetnih nevronov in so zgrajene po vzoru človeških možganov. Podobno kot nevroni v človeških možganih so tudi ume-

tni nevroni v nevronskih mrežah organizirani v plasti. Učenje poteka v več nivojih tako, da se informacije iz ene plasti skozi nelinearno funkcijo prenesejo v višjo, bolj abstraktno plast. Z dovolj velikim številom plasti se tako lahko naučimo tudi zelo zapletenih relacij. Za uporabo tradiconalnih tehnik strojnega učenja je potrebno ročno določiti lastnosti objektov, na podlagih katerih nato poteka učenje. Postopek zahteva podrobno načrtovanje, predvsem pa veliko časa in področnega znanja. Glavna prednost globokega učenja pred ostalimi tehnikami strojnega učenja je, da med učenjem same avtomatsko določijo lastnosti in ugotovijo, katere so uporabne. Med procesom se nato naučijo pravila, ki povezuje vhodne podatke z izhodnimi. Globoko učenje je uspešno predvsem v primerih, ko imamo na voljo veliko število učnih podatkov.

Cilj diplomske naloge je preveriti, kako se globoko učenje obnese pri problemu klasifikacije bakterij v taksonomske nivoje. Pravilna razvrstitev bakterij v taksonomske nivoje ima pomemben vpliv na razumevanje klinične mikrobiologije in nalezljivih bolezni. Z analizo genoma lahko identificiramo vrsto bakterije ali pa ugotovimo, katerim že znanim je nova bakterija podobna. Problem je še posebej kritičen v primerih, ko je bakterija povzročiteljica bolezni - gre za patogene bakterije. Takrat ima pravilna klasifikacija pomembno vlogo pri identifikaciji bolezni in načinu zdravljenja [9]. V diplomski nalogi smo zgradili klasifikacijske modele, ki poizkušajo bakterijo razvrstiti v pravilno deblo, razred, red, družino in rod. Po zgledu iz članka [21] smo za učenje uporabili zaporedja gena 16S rRNA, klasifikacijo pa izvajali s pomočjo konvolucijskih nevronskih mrež. Zanimalo nas je tudi, kako se pri reševanju problema obnesejo rekurenčne nevronske mreže, eksperimente pa smo ponovili na dveh različno velikih množicah podatkov.

Vsebina diplomskega dela je razdeljena na 4 poglavja. Najprej opišemo, kakšne podatke smo uporabili za učenje in kje smo jih pridobili. Sledi predstavitev uporabljenih klasifikacijskih modelov in opis vrednotenja njihove uspešnosti. Zaključimo s prikazom rezultatov in sklepom.

Poglavje 2

Podatki

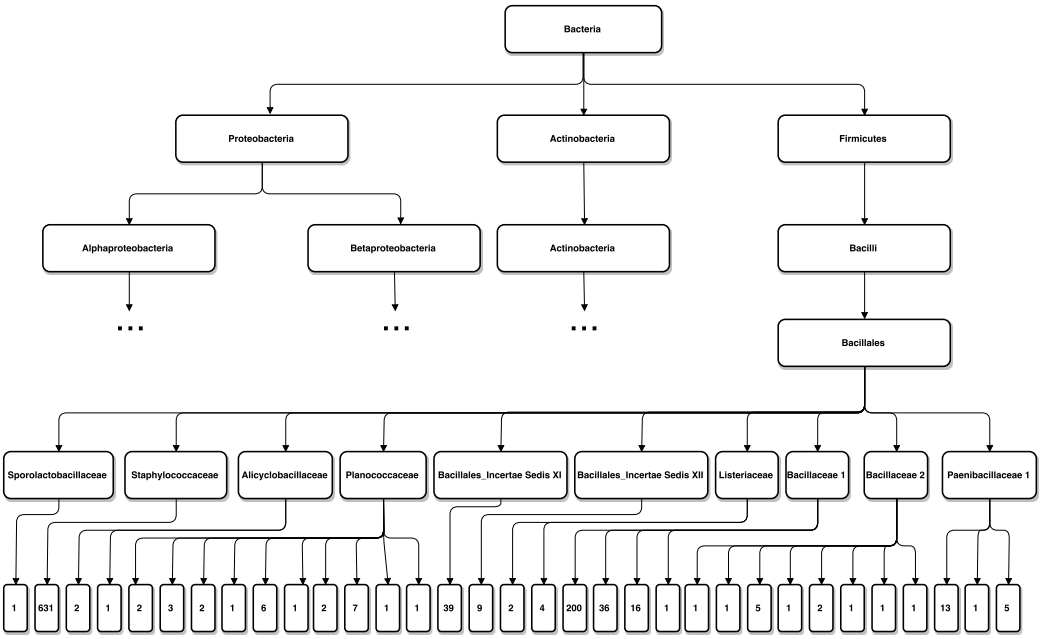
Podatke smo pridobili iz repozitorija RDP (Ribosomal Database Project) [10], izdaja 11.4. Repozitorij RDP vsebuje poravnana in označena genska zaporedja bakterij, arhej in gliv, skupaj z orodji za dodatno analizo teh zporedij.

Iz repozitorija smo pridobili datoteko v zapisu FASTA, ki je vključevala podatke o 1423984 neporavnanih genskih zaporedjih bakterij. Za vsako bakterijo smo imeli na voljo podatke o tem, katerim taksonomskim kategorijam pripada in pa njeno gensko zaporedje. Na voljo smo imeli informacije o deblu (ang. phylum), razredu (ang. class), redu (ang. order), družini (ang. family) in rodu (ang. genus) podanega zaporedja gena 16S rRNA. To je del bakterijskega genoma, ki vsebuje zapis za manjše podenote ribosomov (ang. small-subunit ribosomial RNA) in je uporaben kot splošni filogenetski označevalec. Pogosto se uporablja za ugotavljanje raznolikosti bakterij, identifikacije in njihove filogenetske podobnosti ter je osnova za molekularno taksonomijo [3]. Celotna množica podatkov je obsegala zaporedja gena 16S rRNA bakterij, ki so spadala v 30 različnih debel, 56 različnih razredov, 123 različnih redov, 316 različnih družin in 1830 različnih rodov.

Eksperimente smo izvajali na dveh različno velikih množicah podatkov. V prvem primeru smo po zgledu iz članka [21] uporabljali po 1000 naključno izbranih zaporedij gena 16S rRNA iz vsakega izmed treh najpogostejših debel

bakterij, Actinobacteria, Firmicutes, Proteobacteria, in tako skupaj zbrali 3000 zaporedij. Vsa zaporedja imajo dolžino več kot 1200bp (bazni par, ang. base pair), zato so klasificirane kot najboljši predstavniki svoje vrste in so s strani RDP baze potrjene kot kvalitetne. Slika 2.1 prikazuje delno strukturo uporabljenih podatkov. Koren drevesa predstavlja kraljestvo bakterij, ki se na naslednjem nivoju razdeli na tri debla - Proteobacteria, Actinobacteria in Firmicutes. Primeri bakterij iz debla Proteobacteria so v izbrani množici podatkov spadali v dva različna razreda - Alphaproteobacteria in Betaproteobacteria, vse bakterije iz debla Actinobacteria pa so pripadale enakoimenskemu razredu Actinobacteria. Struktura podatkov se v naslednjih korakih precej razveja, zato smo pri deblih Proteobacteria in Actinobacteria z vejanjem zaključili na nivoju razreda, medtem ko smo strukturo debla Firmicutes predstavili še malo bolj natančno. Vse bakterije tega debla iz izbrane množice podatkov so pripadale razredu Bacilli in redu Bacillales. Na naslednjem nivoju lahko opazimo, da se podatki razdelijo v 10 različnih družin. Na sliki lahko vidimo, da je v listih drevesa na zadnjem nivoju namesto imen rodov prikazano število primerov, ki spadajo v posamezen rod. Vsota vseh števil se torej sešteje v 1000, saj smo imeli na voljo 1000 primerov iz debla Firmicutes. Kot je razvidno iz strukture drevesa, so različne družine različno razcepljene na nivoju rodov, prav tako pa so različno zastopane v smislu števila primerov, ki jim pripadajo. Vsi primeri iz družine Staphylococcaceae spadajo v en sam rod, medtem ko se primeri iz družine Planococcaceae razdelijo na 10 rodov. Družini Staphylococcaceae in Sporolactobacillaceae sta na nivoju rodov enako razvejani - le en rod - a sta v naših podatkih zelo različno zastopani, na voljo imamo 631 primerov iz družine Staphylococcaceae in le en primer iz družine Sporolactobacillaceae.

V tabeli 2.1 so prikazani podatki o številu posameznih taksonomskih kategorij, na katere se razdelijo podatki iz vsakega debla manjše množice podatkov. Jasno je razvidno, da kljub enakomerni razvejanosti in zastopanosti podatkov na nivoju debla, množica podatkov v nižjih taksonomskih nivojih postane manj uravnotežena. Idealno bi bilo, da bi imeli na vseh nivojih



Slika 2.1: Delna struktura manjše množice podatkov

za vsako kategorijo enako število učnih primerov, kar se v našem primeru ne zgodi. Že na drugem nivoju se množica bakterij iz debla Actinobacteria razcepi na dva razreda, medtem ko bakterije iz debla Proteobacteria in Firmicutes ostanejo v enem razredu. Razlika je na nižjih nivojih še bolj opazna. Če bi bili primeri znotraj posameznega debla razporejeni enakomerno, bi imeli na nivoju rodov za bakterije debla Actinobacteria v povprečju na voljo okrog 14 primerov za vsak rod, za bakterije debla Firmicutes okrog 30 primerov za vsak rod in za bakterije debla Proteobacteria okrog 7 primerov za vsak rod. Že tukaj so razlike kar precejšne, a kot smo videli na sliki zgoraj, različne kategorije znotraj enega taksonomskega nivoja niso enakomerno zastopane. Tako imamo za nekatere rodove na voljo le en učni primer, za druge 2 ali 3, za najbolj zastopan rod v naši množici podtkov pa imamo na voljo kar 631 primerov. Skupno so podatki razporejeni v 3 debla, 4 razrede, 16 redov, 74 družin in 256 rodov.

Druga množica podatkov, na kateri smo izvajali eksperimente, je bila 10-krat večja od prve. Razlog za izvajanje na dveh množicah je globoko učenje,

Tabela 2.1: Taksonomska razporeditev manjše množice podatkov

Tri glavna bakterijska debla	Število kategorij na vsakem taksonomskem nivoju				
	deblo	razred	red	družina	rod
Firmicutes	1	1	1	10	33
Proteobacteria	1	2	13	35	153
Actinobacteria	1	1	2	29	70

za katerega je značilno, da je zelo uspešno pri klasifikacijiskih problemih, kjer imamo na voljo veliko količino podatkov. Podobno kot pri prvi množici smo tudi tokrat iz vsakega izmed treh najpogostejših debel naključno izbirali zaporedja. Za vsako izmed debel smo izbrali 10000 zaporedij in tako torej imeli na voljo skupno 30000 primerov. V tabeli 2.2 lahko vidimo, da so podatki tokrat še bolj kompleksni in bolj neenakomerno razporejeni, kot v primeru manjše množice. Če bi bili podatki znotraj taksonomskih kategorij na posameznih nivojih razporejeni enakomerno, bi imeli na nivoju rodov za bakterije debla Actinobacteria v povprečju na voljo okrog 70 učnih primerov za vsak rod, za bakterije debla Proteobacteria okrog 32 primerov za vsak rod in za bakterije znotraj debla Firmicutes okrog 159 primerov za vsak rod. Že v prejšjem primeru smo videli, da zastopanost kategorij znotraj nivoja ni enakomerna. V večji množici podatkov smo opazili, da imamo za nekatere rodove na voljo le eno zaporedje, za najbolj zastopanega pa kar 6423 zaporedij. Skupno so podatki večje množice razporejeni v 3 debla, 4 razrede, 19 redov, 88 družin in 519 rodov.

Tabela 2.2: Taksonomska razporeditev večje množice podatkov

Tri glavna bakterijska debla	Število kategorij na vsakem taksonomskem nivoju				
	deblo	razred	red	družina	rod
Firmicutes	1	1	1	12	63
Proteobacteria	1	2	16	38	313
Actinobacteria	1	1	2	38	143

Poglavje 3

Metode

V diplomski nalogi smo za klasifikacijo bakterij na podlagi zaporedij gena 16S rRNA uporabljali različne metode globokega učenja in jih primerjali z metodo naključnih gozdov. Podatke, ki smo jih imeli na voljo, smo najprej preprocesirali v primerno obliko za uporabljene metode strojnega učenja. Zgradili smo 4 različne klasifikacijske modele, ki za svoje delovanje uporabljajo nevronske mreže in opisali njihovo strukturo ter implementacijo. Na koncu smo uspešnost klasifikacijskih modelov primerjali med seboj.

3.1 Globoko učenje

Globoko učenje (ang. deep learning)[17] je relativno novo področje umetne inteligence, ki dosega izredne rezultate na področjih prepoznave govora, prepoznave slik, razumevanja besedila, prevajanja, pa tudi v drugih domenah, kot so odkrivanje zdravil in genomika. Globoko učenje odkriva zapletene strukture v ogromnih podatkovnih zbirkah z uporabo vzvratnega širjenja napake (ang. backpropagation) - model prilagaja notranje uteži za izračun predstavitve podatkov v trenutni plasti na podlagi izhodnih podatkov iz prejšnje plasti.

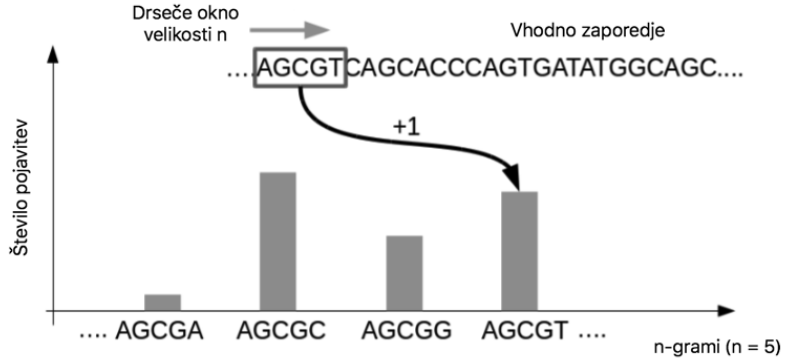
Klasifikacijski modeli, ki uporabljajo globoko učenje, imajo večplastno arhitekturo - sestavljeni so iz vhodne, skrite in izhodne plasti. Vsaka plast

je relativno preprosta računska enota, ki se poizkuša naučiti določeni nivo reprezentacije, med sabo pa so plasti povezane z nelinearnimi funkcijami. Izhod iz nižje plasti je tako vhod v višjo plast modela.

V diplomski nalogi smo gradili klasifikacijske modele s pomočjo konvolucijskih in rekurenčnih nevronskih mrež. Konvolucijske nevronske mreže (ang. convolutional neural networks) so bile sprva načrtovane za prepoznavanje vzorcev na slikah [20]. Običajno so sestavljene iz treh različnih tipov plasti: konvolucijska plast, združevalna plast in polno povezana plast. Kot že ime pove, ima glavno vlogo pri učenju konvolucijska plast. Konvolucija je matematična operacija med vhodnimi podatki in filtrom manjše dimenzije. S filtrom oziroma jedrom se premikamo po vhodnih podatkih in računamo skalarne produkte, na ta način pa proizvedemo nove podatke. Te podatke nato normaliziramo z vnaprej podano aktivacijsko funkcijo, dobljeni rezultat pa je vhod v nasledjo plast. Z vpeljavo konvolucije se časovna kompleksnost učenja poveča, zato uporabimo metodo združevanja maksimalnih vrednosti (ang. max pooling), ki zmanjšuje število parametrov in s tem poveča hitrost algoritma. Zadnja plast v mreži je polno povezana plast, ki povezuje vsako izhodno celico iz združevalne plasti z vsako izhodno celico modela.

Za naloge, ki vključujejo vhodne podatke v obliki zaporedja (prepoznava govora, procesiranje naravnega jezika) je pogosto boljše kot konvolucijske uporabiti rekurenčne nevronske mreže [13, 19]. Rekurenčne mreže procesirajo vhodne podatke po vrsti, enega na enkrat, v svojih skritih plasteh pa hranijo informacijo o zgodovini vseh prejšnjih stanj. Njihov glavni namen je, da se naučijo dolgoročnih odvisnosti, a se je v praksi izkazalo, da je informacije težko zadržati za dolgo časa. Kot rezultat so se pojavile mreže s spominom, med katerimi so najbolj znane LSTM (ang. long short-term memory) mreže. Uporabljajo posebne skrite celice, ki si vhodne podatke zapomnejo za daljši čas [17]. Po zmogljivosti se z LSTM celicami lahko primerjajo novejše GRU (ang. gated recurrent unit) celice [8], ki smo jih uporabili tudi pri gradnji klasifikacijskih modelov v tej diplomski nalogi.

Za gradnjo nevronskih mrež smo v naši nalogi uporabljali knjižnico keras



Slika 3.1: Spektralna predstavitev podatkov

[6], za pripravo podatkov in analizo modelov s stališča napovedne točnosti pa razvili program v jeziku Python.

Vhodni podatki v konvolucijsko nevronsko mrežo morajo biti v obliki večdimenzionalnega vektorja, zato smo genska zaporedja, ki smo jih imeli na voljo, najprej preoblikovali. Vsak organizem ima tipičen spekter n-gramov, ki se pojavlja v njihovem genskem zaporedju [7]. S pomočjo tega jih lahko ločimo od drugih organizmov. Na podatkih v spektralni obliki lahko uporabimo več tehnik strojnega učenja - metodo podpornih vektorjev, k-najbližjih sosedov, naključne gozdove..., zato jo pogosto uporabljamo pri klasifikacijskih problemih. Za dano število n je spektralna predstavitev genskega zaporedja vektor velikosti 4^n. Vsaka komponenta vektorja ustreza številu pojavitev pripadajočega n-grama v genskem zaporedju. Predstavitev sekvence v spektralni obliki dobimo torej tako, da se z drsečim oknom velikosti n premikamo po sekvenci in prištevamo po 1 komponenti vektorja, ki pripada trenutnemu opaženemu n-gramu (slika 3.1). V primeru, da v zaporedju naletimo na neznane znake, ta del podatkov zavržemo. V našem primeru smo zaporedje nukleotidov preoblikovali v spektralni prostor 5-gramov, torej so bili naši vhodni podatki v obliki vektorjev velikosti 1024.

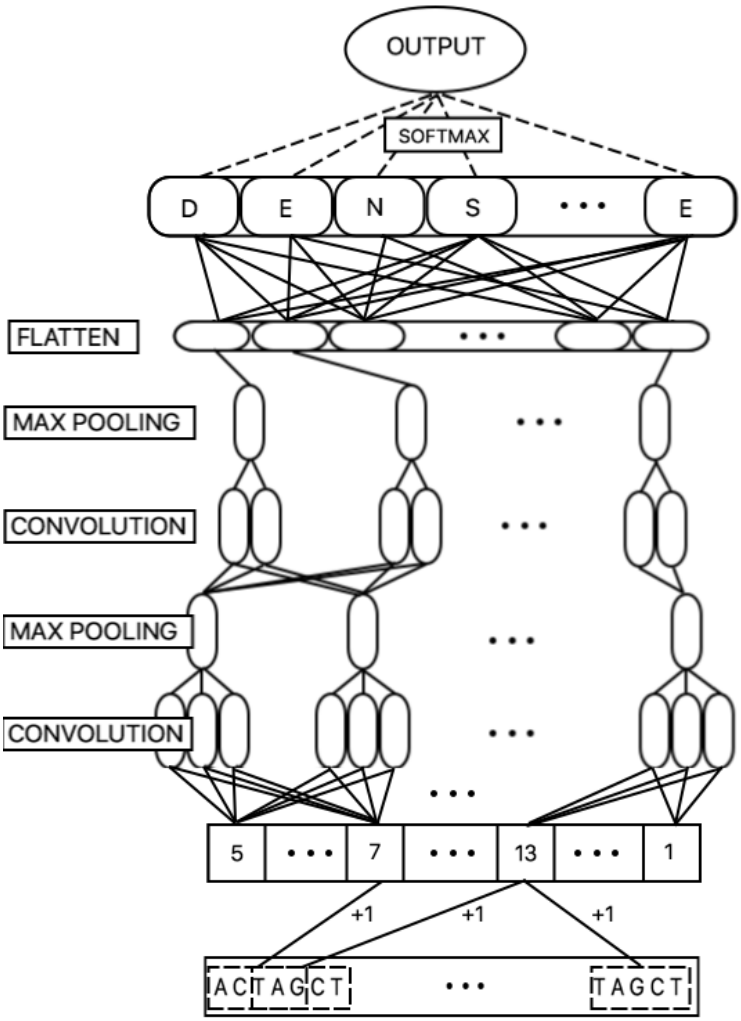
Program s slike 3.2 uporablja knjižnico keras in implementira konvolucijsko nevronsko mrežo, katere struktura je prikazana na sliki 3.3. Mreža je sestavljena iz dveh nivojev konvolucije. Prvi nivo je sestavljen iz 10-ih

```
model = Sequential()
model.add(Convolution1D(10, 5, border_mode="same",
                input_shape=(1024,1)))
model.add(Activation("relu"))
model.add(MaxPooling1D(pool_length=2, stride=None,
                border_mode="valid"))
model.add(Convolution1D(20, 5, border_mode="same"))
model.add(Activation("relu"))
model.add(MaxPooling1D(pool_length=2, stride=None,
                border_mode="valid"))
model.add(Flatten())
model.add(Dense(len(taxonomic_ranks)))
model.add(Activation("softmax"))
```

Slika 3.2: Implementacija konvolucijske nevronske mreže v okolju keras

filtrov, drugi pa iz 20-ih filtrov dimenzije 5. Po vsaki plasti konvolucije sledi aktivacija, nato pa še združevanje maksimalnih vrednosti. Vsak izhod iz konvolucijske plasti nato povežemo z vsako celico izhodne plasti modela. Število celic v izhodni plasti ustreza številu taksonomskih razredov, ki jih želimo potrditi oziroma zavrniti. Sledi še ena aktivacijska plast, ki izhod prejšne plasti spremeni v vrednosti med 0 in 1, ki se seštejejo v 1, zato jih lahko interpretiramo kot verjetnosti. Na koncu gensko zaporedje in s tem organizem, ki mu pripada, uvrstimo v tisti taksonomski razred, ki ima največjo verjetnost glede na izhodno plast modela.

S spektralno predstavitvijo podatkov, uporabljenih pri konvolucijski nevronski mreži, pri učenju ne upoštevamo celotnega genskega zaporedja, saj ignoriramo pozicije n-gramov. Da bi odpravili to pomanjkljivost, smo zgradili drugačen klasifikacijski model, kjer smo za učenje uporabili preprosto rekurenčno nevronsko mrežo (slika 3.4). V tem primeru smo namesto spektralne predstavitve genskega zaporedja za učenje uporabili kar celotno zaporedje.



Slika 3.3: Struktura konvolucijske nevronske mreže, ki smo jo uporabili za klasifikacijo bakterij

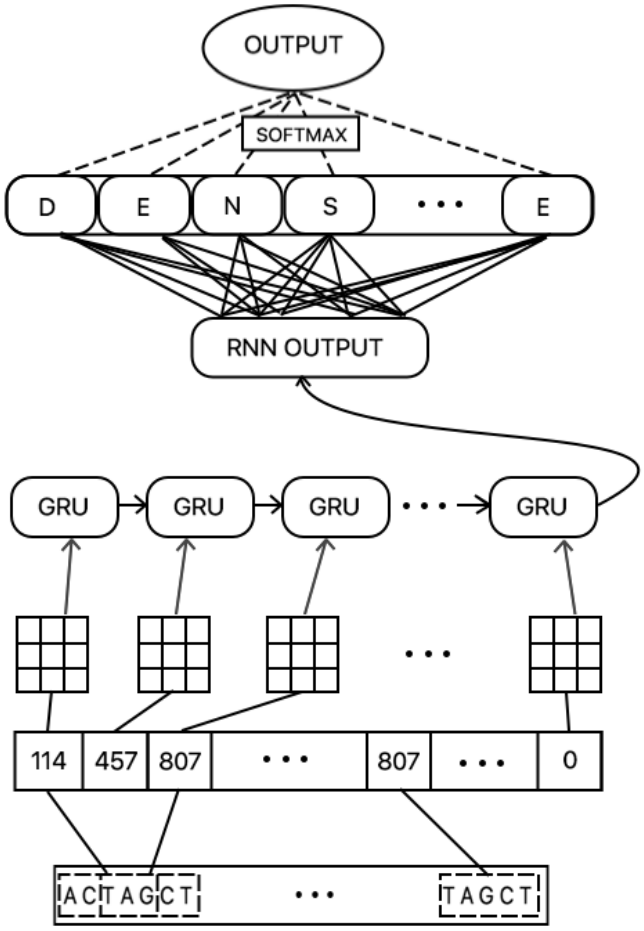
Po zaporedju smo se sprehajali z drsečim oknom velikosti 5 in beležili opažene 5-grame. Tako smo zaporedje nukleotidov dolžine n preslikali v vektor velikosti $n - 4$, ki namesto nukleotidov vsebuje indekse vsebovanih 5-gramov. Nato smo dobljene vektorje poravnali tako, da smo na začetku ali koncu dodali ničle. Na ta način smo vsa različno dolga genska zaporedja predstavili z vektorjem enotne dolžine, hkrati pa smo ohranili vse informacije, ki jih imamo v zaporedju.

Implementacija preproste rekurenčne nevronske mreže v okolju keras je predstavljena na sliki 3.5. Mreža je sestavljena iz ene plasti prej omenjenih celic GRU s 300 izhodnimi enotami. Podobno kot pri zgoraj opisanih konvolucijskih nevronskih mrežah, tudi tukaj povežemo rekurenčno plast z vsako enoto izhodne plasti, nato pa sledi še aktivacija, s katero izhodne vrednosti preslikamo v verjetnosti.

Pri rekurenčnih nevronskih mrežah lahko včasih opazimo, da zadnjemu delu zaporedja pripišemo večjo težo, saj pri vsakem koraku malo posplošimo prejšnja stanja. Da bi se izognili efektu pozabljanja, smo implementirali učenje z dvosmernimi rekurenčnimi nevronskimi mrežami (ang. bidirectional recurrent neural network) [23].

Struktura modela je prikazana na sliki 3.6, implementacija s knjižnico keras pa na sliki 3.7. Podatke, obdelane na enak način kot pri preprosti rekurenčni nevronski mreži, uporabimo v dveh modelih. Prvi model vhodna genska zaporedja obdeluje v običajnem vrstnem redu, torej od sprednjega proti zadnjemu koncu, medtem ko drugi model obdeluje zaporedje v nasprotno smer, od zadnjega proti sprednjemu koncu. Oba modela sta sestavljena iz ene plasti celic GRU s po 150 izhodnimi enotami, ki jih nato združimo. Enako kot pri prejšnih modelih tudi tukaj povežemo dobljeni združeni vektor z vsemi enotami izhodne plasti, nato pa temu dodamo še aktivacijsko plast.

Konvolucijske mreže so uspešne pri iskanju vzorcev, rekurenčne mreže pa upoštevajo zaporedje, zato smo implementirali tudi hibridni model (slika 3.6), ki poizkuša izkoristiti obe prednosti. Kot je razvidno iz programske kode 3.8, je model sestavljen iz dveh konvolucijskih plasti, sledi pa jima



Slika 3.4: Struktura rekurenčne nevronske mreže, ki smo jo uporabili za klasifikacijo bakterij

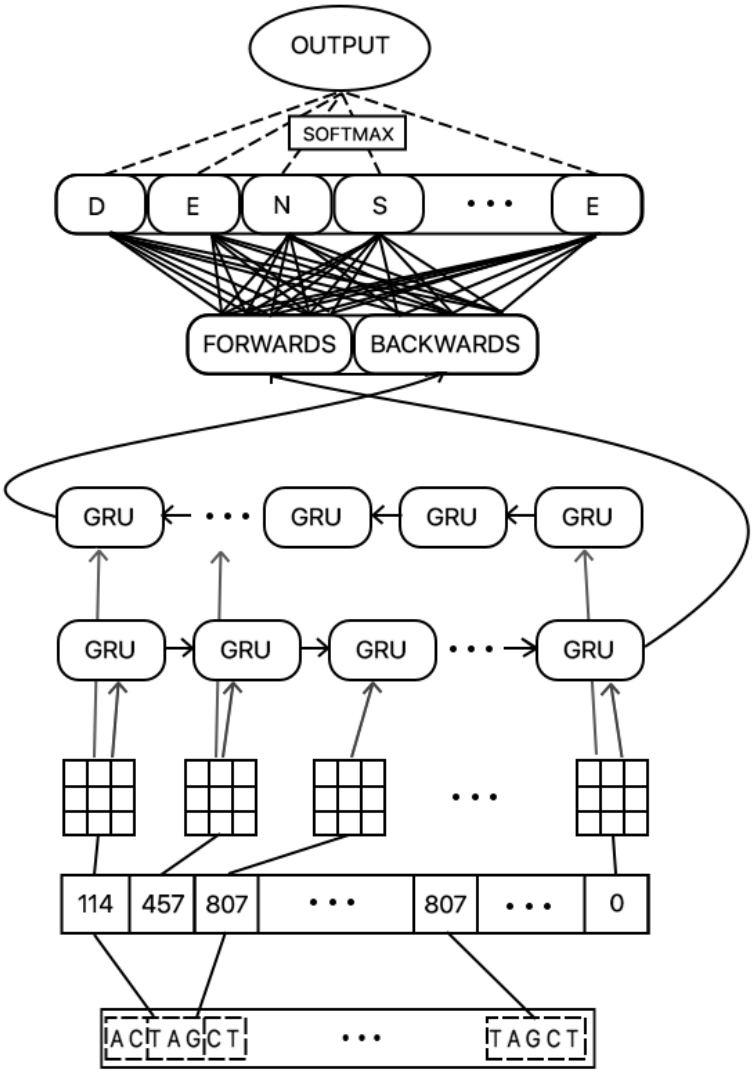
```
model = Sequential()
model.add(Embedding(1025, 100, input_length=sequence_length))
model.add(GRU(input_shape=(sequence_length, 1), output_dim=300))
model_rnn.add(Dense(len(taxonomic_ranks)))
model_rnn.add(Activation("softmax"))
```

Slika 3.5: Implementacija rekurenčne nevronske mreže v okolju keras

še rekurenčna plast. Konfiguracija konvolucijskih in rekurenčnih plasti je enaka kot pri prejšnjih primerih. Vhod v konvolucijsko plast so zaporedja, pridelana na enak način kot pri preprosti in dvosmerni rekurenčni nevronski mreži. Na ta način ohranimo vse informacije o zaporedju. Ideja je, da s pomočjo konvolucijskih mrež najprej poiščemo vzorce, nato pa se na želimo na teh poenostavljenih sekvencah z rekurenčnimi mrežami naučiti, v katere taksonomske razrede spadajo.

3.2 Vrednotenje točnosti

Za vrednotenje uspešnosti modelov moramo podatke najprej razdeliti na učno in testno množico. Učne primere uporabimo za razvoj klasifikacijskega modela, testne primere pa za ocenitov njegove uspešnosti. V diplomskem delu smo na podatkih izvajali 10-kratno prečno preverjanje, kar pomeni, da smo podatke razdelili na 10 enakih množic in od teh v vsakem koraku uporabili eno množico za testiranje, ostalih 9 pa za učenje. Uspešnost modelov v vsaki ponovitvi smo merili z dvema statistikama, klasifikacijsko točnostjo in mero F1, na koncu pa izračunali povprečje. Klasifikacijska točnost nam pove, kakšen delež vseh primerov smo uvrstili v pravilni razred. Mera F1 je predstavljena kot harmonično povprečje med natančnostjo (delež pravilno napovedanih primerov v nek razred) in občutljivostjo (delež pravilno napovedanih primerov nekega razreda). V našem primeru, ko imamo klasifikacijo v več razredov, mera F1 predstavlja povprečje mer skozi vse razrede.



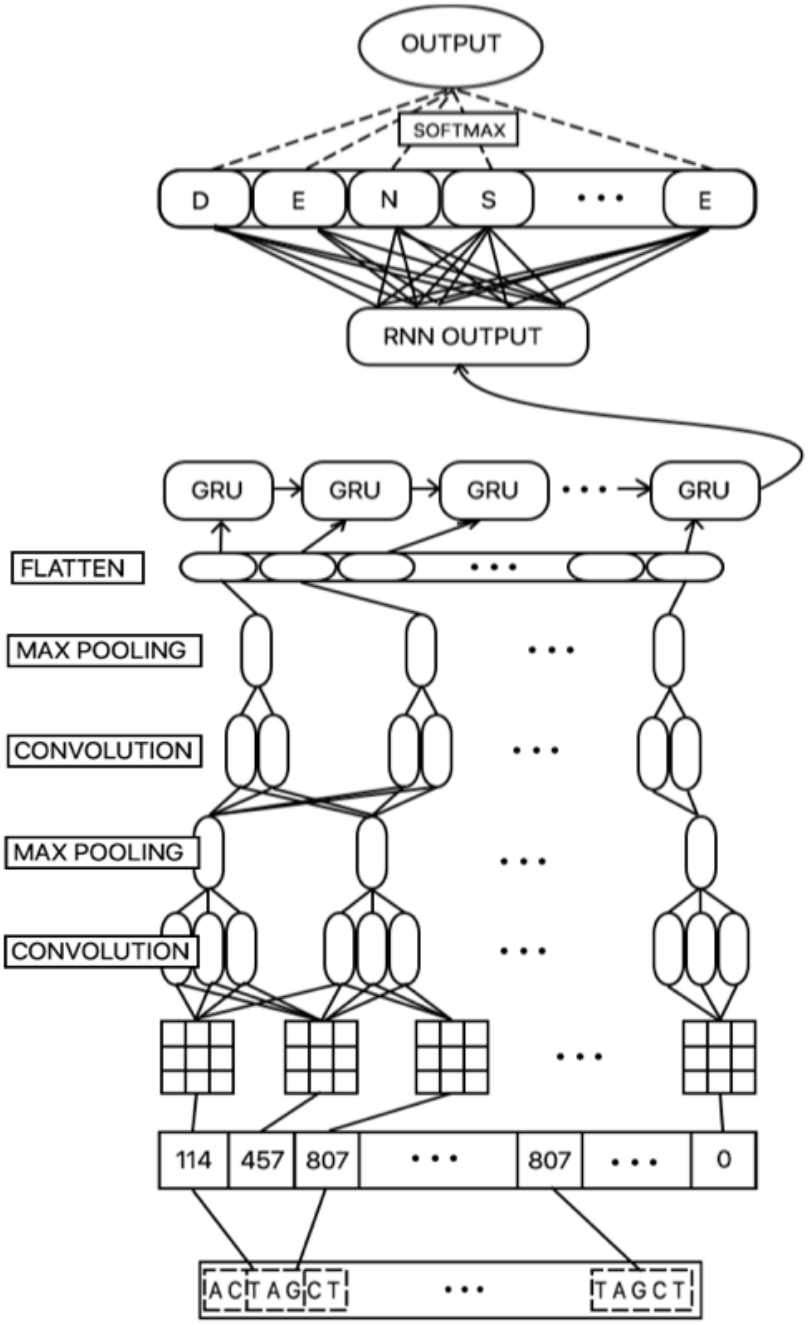
Slika 3.6: Struktura dvosmerne rekurenčne nevronske mreže, ki smo jo uporabili za klasifikacijo bakterij

```
forwards = Sequential()
forwards.add(Embedding(1025, 100, input_length=sequence_length))
forwards.add(GRU(input_shape=(sequence_length, 1), output_dim=150))

backwards = Sequential()
backwards.add(Embedding(1025, 100, input_length=sequence_length))
backwards.add(GRU(input_shape=(sequence_length, 1), output_dim=150,
                  go_backwards=True))

model = Sequential()
model.add((Merge([forwards, backwards], mode="concat")))
model.add(Dense(len(taxonomic_ranks)))
model.add(Activation("softmax"))
```

Slika 3.7: Implementacija dvosmerne rekurenčne nevronske mreže v okolju keras

```
model = Sequential()
model.add(Embedding(1025, 100, input_length=sequence_length))
model.add(Convolution1D(10, 5, border_mode="same"))
model.add(Activation("relu"))
model.add(MaxPooling1D(pool_length=2, stride=None,
                border_mode="valid"))
model.add(Convolution1D(20, 5, border_mode="same"))
model.add(Activation("relu"))
model.add(MaxPooling1D(pool_length=2, stride=None,
                border_mode="valid"))
model.add(GRU(output_dim=300))
model.add(Dense(len(taxonomic_ranks)))
model.add(Activation("softmax"))
```

Slika 3.8: Implementacija hibridnega modela s konvolucijsko in rekurenčno nevronsko mrežo v okolju keras



Slika 3.9: Struktura hibridnega modela (kombinacija konvolucijske in rekurenčne nevronske mreže), ki smo ga uporabili za klasifikacijo bakterij

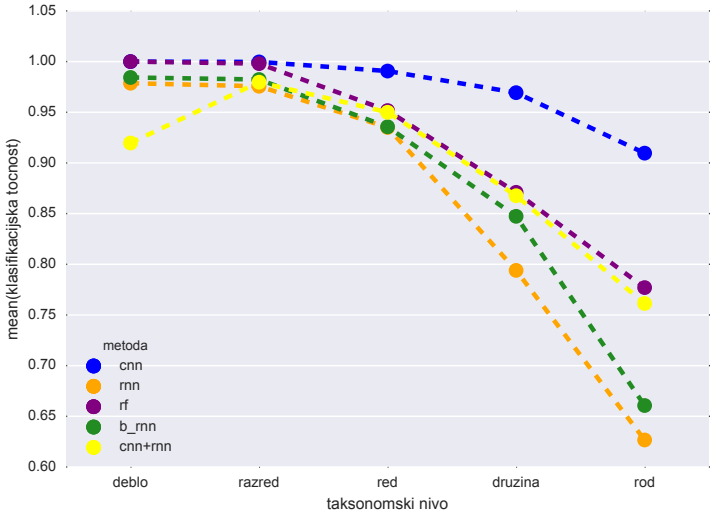
Poglavje 4

Rezultati in diskusija

Z uporabo podatkov in metod opisanih v prejšnjih poglavjih smo naredili dve vrsti eksperimentov. V prvem primeru smo s pomočjo 10-kratnega prečnega preverjanja merili učinkovitost napovedovanja za vsak taksonomski nivo (od debla do roda) posebej, pri čemer smo za vsak primer upoštevali njegovo celotno gensko zaporedje. V drugem primeru smo namesto celotne sekvence upoštevali le krajši, 500bp dolg del genskega zaporedja, ki smo ga pridobili tako, da smo iz celotnega zaporedja naključno izbrali 500 zaporednih nukleotidov. Namen je bil testirati, ali so metode sposobne klasificirati zaporedja v pravilne taksonomske razrede, tudi če imamo na voljo le manjši (500 bp) del vse genske informacije. Dva eksperimenta smo ponovili na manjši (3.000 primerov) in večji množici podatkov (30.000 primerov).

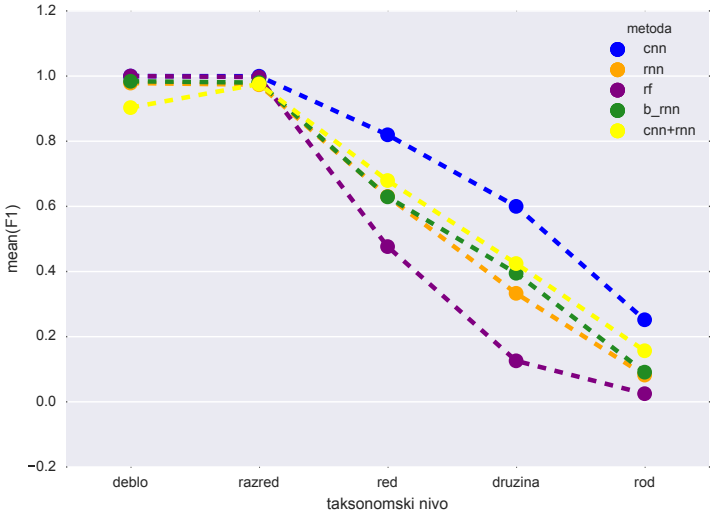
Med seboj smo primerjali rezultate klasifikacijskih modelov, ki smo jih na slikah označili z naslednjimi oznakami:

cnn	konvolucijska nevronska mreža
rnn	rekurenčna nevronska mreža
b_rnn	dvosmerna rekurenčna nevronska mreža
cnn+rnn	hibridni model - kombinacija konvolucijske in rekurenčne nevronske mreže
rf	metoda naključnih gozdov



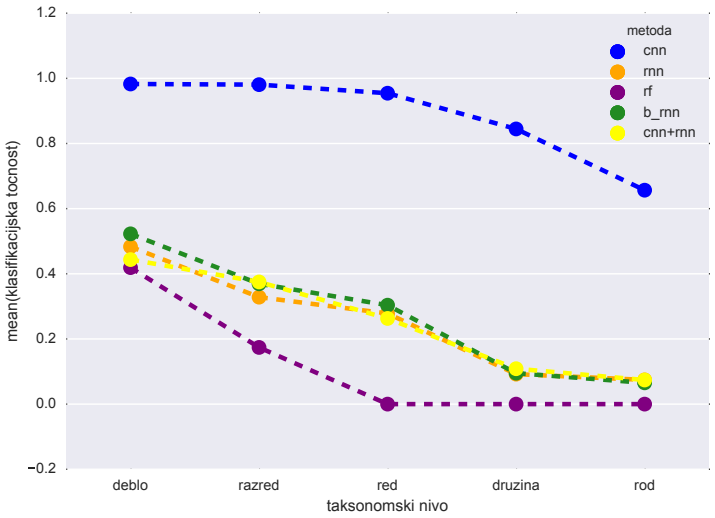
Slika 4.1: Primerjava uspešnosti metod pri klasifikaciji celotnih sekvenc - klasifikacijska točnost

Sliki 4.1 in 4.2 prikazujeta uspešnost modelov pri klasifikaciji primerov na manjši množici podatkov, kjer smo za klasifikacijo uporabljali celotno genško zaporedje. Iz obeh grafov je vidno, da so konvolucijske nevronske mreže dosegle najboljše rezultate. Pri razvrščanju zaporedij v začetne nivoje hierarhije (deblo, razred, red) razlike v klasifikacijski točnosti niso velike, na zadnjem nivoju, pri razvrščanju v rod, pa so konvolucijske nevronske mreže od druge najboljše metode uspešnejše za več kot 10%, od najslabše metode pa kar za več kot 25%. Zanimiva je krivulja hibridne metode - pri klasifikaciji zaporedij je na prvem nivoju občutno slabša od ostalih, nato pa se na naslednjih bolj kompleksnih nivojih izboljša in dosega skoraj enake klasifikacijske točnosti kot metoda naključnih gozdov. Na grafu mere F1 lahko opazimo, da pri vseh metodah vrednost hitro pada. To se verjetno zgodi zaradi večanja števila razredov in njihove neenakomerne porazdelitve. Krivulja hibridnega modela ponovno kaže, da je pri razvrščanje v razrede debla uspešnost slabša od ostalih modelov, a se že na naslednjem nivoju izboljša in se ji od vseh metod še najbolj uspe približati konvolucijskim nevronskim mrežam.

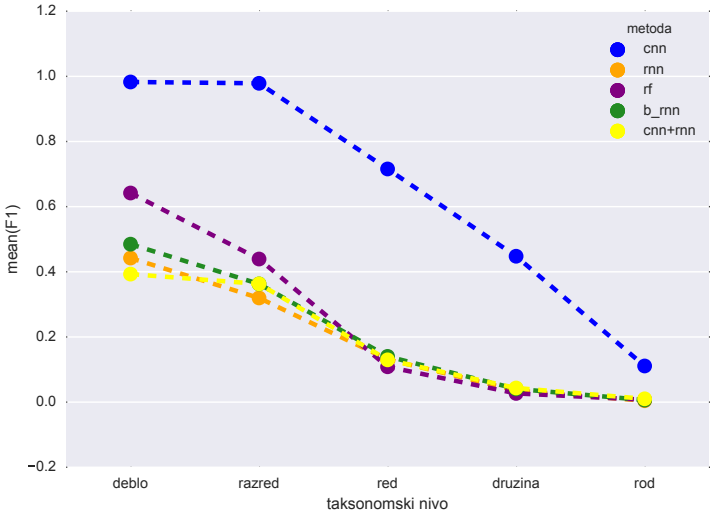


Slika 4.2: Primerjava uspešnosti metod pri klasifikaciji celotnih sekvenc - mera F1

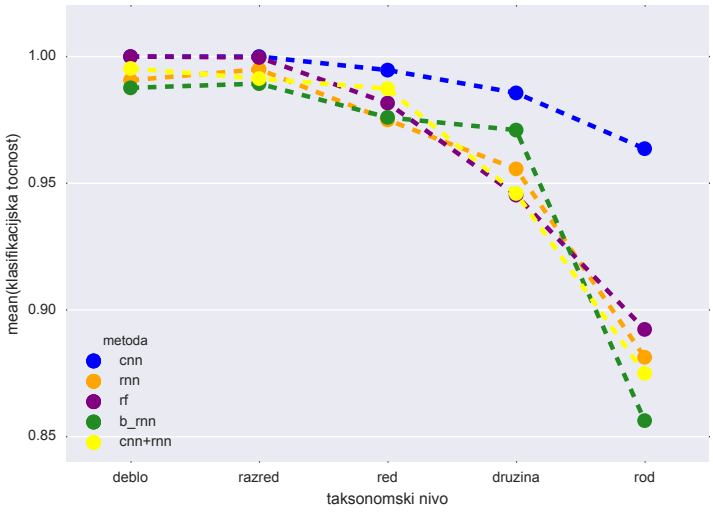
Na slikah 4.3 in 4.4 so prikazani rezultati uspešnosti modelov pri klasifikaciji krajših delov sekvenc (500 bp), kjer smo za učenje uporabljali manjšo množico podatkov. Še bolj kot prej je očitno, da so za to nalogo najbolj primerne konvolucijske nevronske mreže. Že pri razvrščanju v deblo je njihova klasifikacijska točnost od točnosti ostalih metod višja za več kot 40%, na zadnjem nivoju pa za več kot 50%. Tako vrednosti klasifikacijske točnosti kot tudi mere F1 kažejo, da ostale metode dosegajo zelo slabe rezultate, na zadnjem nivoju pa je njihova uspešnost skoraj na ničli. Pri modelih, ki vključujejo rekurenčne mreže, je možen vzrok za to prav dolžina sekvec. Modeli se namreč učijo na poravnanih sekvencah, ki so dolge več kot 1200bp. Ko želimo napovedati razred za sekvenco 500bp, jo hkrati dopolnimo z ničlami do dolžine ostalih sekvenc. Na ta način je več kot polovica sekvence, ki jo na koncu želimo klasificirati, neuporabna. Problem nevronskih mrež je tudi, da se obnašajo kot črna škatla. To pomeni, da ne vemo na kakšen način sprejemajo odločitve, zato je težko ugibati o vzrokih za njihovo neuspešnost (ali uspešnost).



Slika 4.3: Primerjava uspešnosti metod pri klasifikaciji sekvenc dolgih 500bp - klasifikacijska točnost



Slika 4.4: Primerjava uspešnosti metod pri klasifikaciji sekvenc dolgih 500bp - mera F1

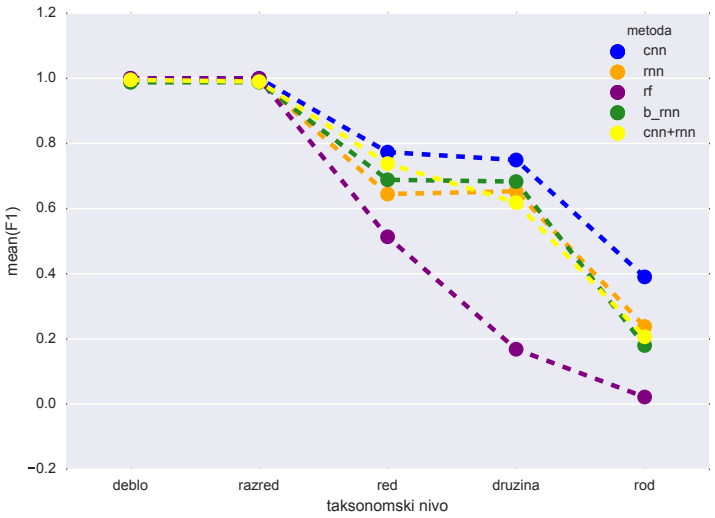


Slika 4.5: Primerjava uspešnosti metod pri klasifikaciji celotnih sekvenc ob učenju na večji množici podatkov - klasifikacijska točnost

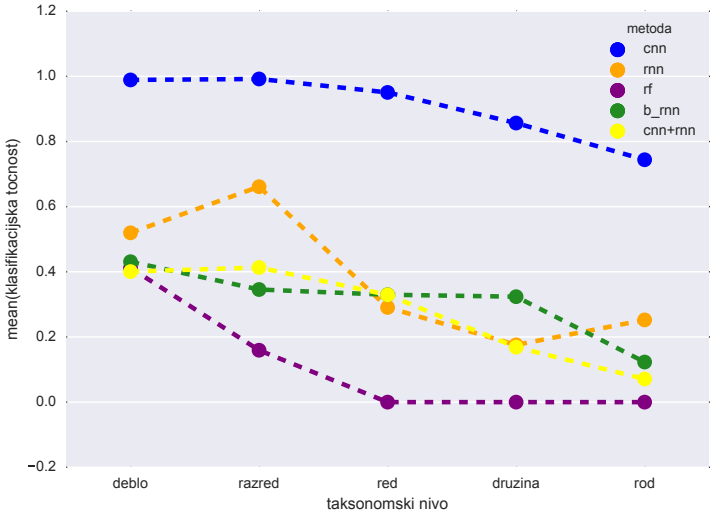
Metode globokega učenja so običajno bolj uspešne, če imajo na voljo več podatkov za učenje, zato smo eksperiment ponovili na večji množici učnih primerov. Kot je razvidno iz slik 4.5 in 4.6 so vsi klasifikacijski modeli dosegli boljše rezultate v primerjavi z učenjem na manjši množici podatkov. Še vedno je najboljše napovedi dosegala metoda s konvolucijskimi nevronskimi mrežami, a so razlike precej manjše. Pri klasifikacijski točnosti, kjer so bile v prvem primeru razlike med najboljšim in najslabšim modelom pri razvrščanju v rodov med 10% in 25%, so sedaj razlike le še okoli 10%, pri višjih nivojih pa so še manjše.

Sliki 4.7 in 4.8 prikazujeta rezultate uspešnosti klasifikacijskih modelov pri razvrščanju 500bp dolgih sekvenc v taksonomske razrede. Kot pri primeru učenja na manjši množici podatkov tudi tukaj opazimo, da konvolucijske nevronske mreže dosegajo veliko boljše rezultate od ostalih modelov.

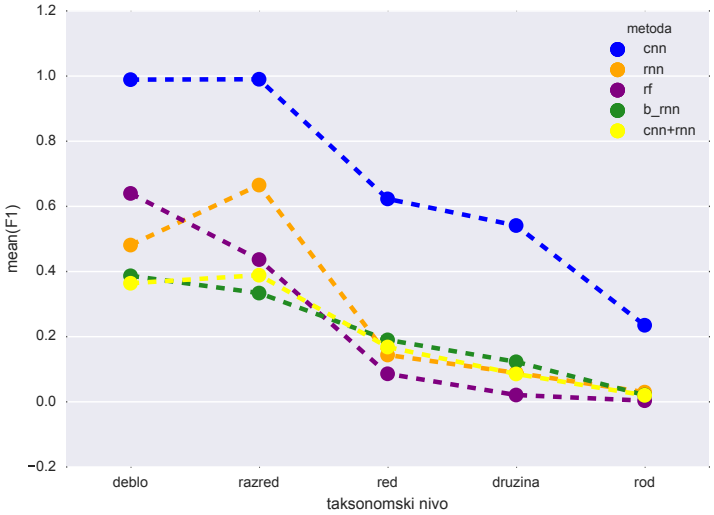
Zaradi naključnega izbora podatkov je prišlo do tega, da v drugem eksperimentu nimamo le večje množice primerov, temveč tudi večjo kompleksnost podatkov. Podatki so v drugem primeru bolj raznoliki in pripadajo večim ta-



Slika 4.6: Primerjava uspešnosti metod pri klasifikaciji celotnih sekvenc ob učenju na večji množici podatkov - mera F1



Slika 4.7: Primerjava uspešnosti metod pri klasifikaciji sekvenc dolgih 500bp ob učenju na večji množici podatkov - klasifikacijska točnost



Slika 4.8: Primerjava uspešnosti metod pri klasifikaciji sekvenc dolgih 500bp ob učenju na večji množici podatkov - mera F1

ksonomskim razredom. Sedaj razvrščamo sekvence v 3 debla (ostaja enako), 4 razrede (ostaja enako), 19 redov (prej 16), 88 družin (prej 74) in 519 rodov (prej 256). Kompleksnost se je torej na nivoju klasifikacije rodov povečala kar za dvakrat. Kljub temu lahko opazimo, da so se napovedni modeli izboljšali in dosegajo višje mere točnosti.

Poleg napovednih točnosti smo primerjali tudi čase učenja opisanih klasifikacijskih modelov. Programe smo poganjali na grafični kartici Geforce GTX Titan X. V tabeli 4.1 so prikazani tipični časi učenja posameznih metod na manjši (množica 1) in večji (množica 2) množici podatkov. Opazimo, da za učenje najmanj časa porabi metoda naključnih gozdov, največ časa pa dvosmerne rekurenčne nevronske mreže. Razlike med časi učenja med modeli so zelo velike - najhitrejši model na manjši množici podatkov za učenje porabi kar 900-krat manj časa od najpočasnejšega, na večji množici podatkov pa 240-krat manj časa. Opazimo lahko tudi, da čas pri učenju s konvolucijskimi nevronskimi mrežami raste približno linearno, pri metodah, ki vključujejo rekurenčne nevronske mreže, pa ne. Vzrok je v tem, da pri učenju konvo-

lucijskih nevronskih mrež vedno naredimo 200 ponovitev (saj se zgledujemo po modelu iz članka [21]), pri učenju z rekurenčnimi mrežami pa postopek učenja ustavimo, ko začne točnost na validacijskih podatkih padati zaradi prevelikega prilagajanja modela učnim podatkom. Število ponovitev učenja v tem primeru ni fiksno. V primeru manjše množice podatkov se rekurenčne mreže v eni iteraciji naučijo manj kot v primeru večje množice podatkov, zato je razumljivo, da več podatkov kot imamo na voljo, manj ponovitev učenja je potrebnih. Kot je razvidno iz tabele, so konvolucijske nevronske mreže veliko hitrejše od rekurenčnih nevronskih mrež, zato so tudi z vidika časovne zahtevnosti primernejša izbira za reševanje našega problema.

Tabela 4.1: Tipični časi učenja posameznih metod

Metoda učenja	Tipičen čas učenja v minutah	
	množica 1	množica 2
Konvolucijska nevronska mreža	3	35
Rekurenčna nevronska mreža	295	1545
Dvosmerna rekurenčna nevronska mreža	635	2660
Hibridni model	125	145
Metoda naključnih gozdov	0.7	11

Poglavje 5

Sklep

V diplomski nalogi smo razvili različne modele za klasifikacijo bakterij v taksonomske razrede in jih med sabo primerjali glede na klasifikacijsko točnost in mero F1. Za učenje in testiranje smo uporabili bakterijske 16S ribosomske RNA sekvence. Zgradili smo klasifikacijske modele, ki uporabljajo konvolucijske in rekurenčne nevronske mreže, primerjali pa smo jih tudi s klasifikacijskim modelom naključnih gozdov. Rezultat diplomske naloge je programska koda, dostopna v GitHub repozitoriju [1].

Uspešno smo replicirali rezultate iz članka [21], kjer so avtorji kot prvi predlagali uporabo konvolucijskih nevronskih mrež v namene taksonomske klasifikacije bakterij. Preizkusili smo tudi, kako se pri reševanju problema obnesejo rekurenčne nevronske mreže. Ker pri rešitvi s konvolucijskimi nevronskimi mrežami upoštevamo le petorke in njihove frekvence, ostalo zaporedje pa praktično zanemarimo, smo pričakovali, da bodo rekurenčne nevronske mreže izboljšale rezultat, saj upoštevajo zaporedje skozi celotno sekvenco. V nasprotju s pričakovanji so rezultati pokazali, da se modeli z rekurenčnimi nevronskimi mrežami obnesejo precej slabše pri vseh primerih in ne glede na to, katero mero točnosti uporabimo. Ker se s pomočjo globokega učenja lahko učimo na veliki količini podatkov, smo eksperiment ponovili še na večji množici. Rezultati glede uspešnosti modelov pri učenju na večji množici podatkov so pokazali, da se ob večjem številu učnih primerov napovedna točnost

modelov poveča, kljub temu da se poveča tudi kompleksnost podatkov. Ugotovili smo, da lahko 500bp dolge sekvence dokaj uspešno klasificirajo le konvolucijske nevronske mreže, uporabljeni modeli z rekurenčnimi mrežami pa za reševanje tega problema niso primerni.

Rezultati kažejo, da je prostora za izboljšave modelov še precej, predvsem pri klasifikaciji sekvenc dolgih 500bp. Morda bi pri tem lahko pomagalo več nivojev nevronskih mrež, bolj premišljena struktura modelov in drugačen izbor parametrov. Glede na rezultate eksperimenta na večji množici podatkov, bi se klasifikacijska točnost modelov povečala, če bi za učenje uporabili celotno množico podatkov, ki je na voljo v repozitoriju RDP. Problem je, da učenje nevronskih mrež, predvsem rekurenčnih, zahteva veliko časa, zato vseh možnosti ni mogoče preizkusiti, težavo pa predstavlja tudi narava nevronskih mrež, ki uporabniku ne omogoča vpogleda v njihovo sprejemanje odločitev.

Literatura

[1] Github repozitorij projekta. `https://github.com/ninamalina/sequence_classification`. Dostopano: 2016-08-28.

[2] Human genome project. `http://web.ornl.gov/sci/techresources/Human_Genome/`. Dostopano: 2016-08-28.

[3] Silvia G Acinas, Luisa A Marcelino, Vanja Klepac-Ceraj, and Martin F Polz. Divergence and redundancy of 16s rRNA sequences in genomes with multiple *rrn* operons. *Journal of Bacteriology*, 186(9):2629–2635, 2004.

[4] Mark D Adams, Susan E Celniker, Robert A Holt, Cheryl A Evans, Jeannine D Gocayne, Peter G Amanatides, Steven E Scherer, Peter W Li, Roger A Hoskins, Richard F Galle, et al. The genome sequence of drosophila melanogaster. *Science*, 287(5461):2185–2195, 2000.

[5] Asif T Chinwalla, Lisa L Cook, Kimberly D Delehaunty, Ginger A Fewell, Lucinda A Fulton, Robert S Fulton, Tina A Graves, LaDeana W Hillier, Elaine R Mardis, John D McPherson, et al. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420(6915):520–562, 2002.

[6] François Chollet. Keras. *GitHub repository: https://github.com/fchollet/keras*, 2015.

[7] Benny Chor, David Horn, Nick Goldman, Yaron Levy, and Tim Massingham. Genomic DNA k-mer spectra: models and modalities. *Genome Biology*, 10(10):1, 2009.

[8] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[9] Jill E Clarridge. Impact of 16s rrna gene sequence analysis for identification of bacteria on clinical microbiology and infectious diseases. *Clinical Microbiology Reviews*, 17(4):840–862, 2004.

[10] James R Cole, Qiong Wang, Jordan A Fish, Benli Chai, Donna M McGarrell, Yanni Sun, C Titus Brown, Andrea Porras-Alfaro, Cheryl R Kuske, and James M Tiedje. Ribosomal Database Project: data and tools for high throughput rRNA analysis. *Nucleic Acids Research*, 42(Database issue):D633–D642, 2013.

[11] Sequencing Consortium et al. {Genome sequence of the nematode *C.elegans*: A platform for investigating biology}. *Science*, 282:2012–2018, 1998.

[12] Robert D Fleischmann, Mark D Adams, Owen White, Rebecca A Clayton, et al. Whole-genome random sequencing and assembly of haemophilus influenzae rd. *Science*, 269(5223):496, 1995.

[13] Alex Graves, Abdel-Rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649. IEEE, 2013.

[14] Samir Kaul, Hean L Koo, Jennifer Jenkins, Michael Rizzo, Timothy Rooney, Luke J Tallon, Tamara Feldblyum, William Nierman, Maria Ines Benito, Xiaoying Lin, et al. Analysis of the genome sequence of the flowering plant arabidopsis thaliana. *Nature*, 408(6814):796–815, 2000.

[15] Daniel C Koboldt, Karyn Meltz Steinberg, David E Larson, Richard K Wilson, and Elaine R Mardis. The next-generation sequencing revolution and its impact on genomics. *Cell*, 155(1):27–38, 2013.

[16] Sotiris B Kotsiantis. Supervised machine learning: A review of classification techniques. In *Proceedings of the 2007 conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*, pages 3–24. IOS Press, 2007.

[17] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[18] Marcel Margulies, Michael Egholm, William E Altman, Said Attiya, Joel S Bader, Lisa A Bemben, Jan Berka, Michael S Braverman, Yi-Ju Chen, Zhoutao Chen, et al. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057):376–380, 2005.

[19] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Interspeech*, pages 1045–1048, 2010.

[20] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

[21] Riccardo Rizzo, Antonino Fiannaca, Massimo La Rosa, and Alfonso Urso. *A Deep Learning Approach to DNA Sequence Classification*, pages 129–140. Springer International Publishing, Cham, 2016.

[22] Stuart Jonathan Russell, Peter Norvig, John F Canny, Jitendra M Malik, and Douglas D Edwards. *Artificial intelligence: a modern approach*. Prentice hall Upper Saddle River, 2003.

[23] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.