Predviđanje aktivnosti peptida modelom iterativnog učenja

Polić, Romano

SVEUČILIŠTE U RIJECI

TEHNIČKI FAKULTET

Diplomski sveučilišni studij računarstva

Diplomski rad

Predviđanje aktivnosti peptida modelom iterativnog učenja

Rijeka, 2021.

Romano Polić

0069077854

SVEUČILIŠTE U RIJECI

TEHNIČKI FAKULTET

Diplomski sveučilišni studij računarstva

Diplomski rad

Predviđanje aktivnosti peptida modelom iterativnog učenja

Mentor: doc. dr. sc. Goran Mauša

Rijeka, 2021.

Romano Polić

0069077854

Umjesto ove stranice umetnuti zadatak za diplomski rad

IZJAVA

Izjavljujem da sam samostalno izradio ovaj rad pod vodstvom mentora doc. dr. sc. Gorana Mauše.

U Rijeci, _____

Romano Polić

ZAHVALA

Prije svega zahvalio bi se obitelji na podršci i motivaciji koju su mi pružali tokom studija.

Zahvaljujem se i svim svojim prijateljima koji su bili uz mene kroz sve faze studija. Naravno zahvaljujem se i svojim kolegama koji su ovaj put učinili mnogo lakšim i zabavnijim. Zahvaljujem se i kolegi Eriku Otoviću koji je također pratio tijek pisanja rada te je svojim savjetima i prijedlozima svakako bio važan dio cijele priče. Nikako ne bi propustio priliku zahvaliti se svim profesorima koji su se na bilo koji način našli u ulozi mojeg mentora jer smatram da sam od njih primio najviše znanja i mudrih savjeta.

Posebne zahvale svakako idu mojem mentoru, doc. dr. sc. Goranu Mauši, koji ne samo da je bio uvijek dostupan za pitanja i pomoć već je samu izradu rada učinio mnogo lakšom i zanimljivijom. Sve riječi su suvišne no svakako veliko hvala mojem mentoru.

Sadržaj

Popis slika

Popis tablica

Poglavlje 1

Uvod

Infekcije uzrokovane *antimikrobno otpornim bakterijama* (AMR) postale su ozbiljan problem globalne zdravstvene zaštite. Procjenjuje se da najmanje 700 000 ljudi umire od AMR infekcija svake godine. [2] Pojava i širenje višestruko otpornih tzv. *superbuba* uzrokuju sve veću potrebu za novim antimikrobnim lijekom. Kao buduće oružje za borbu protiv antimikrobno otpornih infekcija, *antimikrobni peptidi* (AMP) sve češći su predmet istraživanja i smatraju se ključnim čimbenikom u njihovoj prevenciji.

Dominantan pristup u analizi kemijskih spojeva poput proteina i peptida temeljio se na iscrpnim i računalno iznimno zahtjevnim numeričkim metodama molekularne dinamike (eng. *molecular dynamics*). Za provedbu takvih metoda potrebno je unaprijed znati fizikalne i kemijske zakone koji upravljaju procesima te ih pretvoriti u složeni lanac računskih operacija. Razvojem računala i računalnog inženjerstva, rodila se mogućnost računalnog učenja na temelju empirijskih podataka u cilju oblikovanja algoritama za predviđanje koji postaju daleko brži i jednostavniji te otvaraju mogućnost analize pojava za koje nisu unaprijed poznati prirodni zakoni.

Uz dovoljno veliki izvorni skup podataka računalo može s visokom vjerojatnošću odrađivati razne predviđanja no ako je taj skup malen i pri tome sadrži podatke koji nisu unutar područja primjene ta predviđanje postaje znatno otežano.

Ovaj diplomski rad je dio uspostavnog istraživačkog projekta Hrvatske zaklade za znanost pod naslovom "Dizajn katalitički aktivnih peptida i peptidnih nanostruktura" pod oznakom UIP-2019-04-7999. Cilj ovog rada je proučavanje metoda predviđanja u uvjetima male količine podataka, s fokusom na tehniku iterativnog učenja i višeosposobljenog učenja. Prikazat će se razne metode obrade podataka poput analize glavnih komponenti, poduzorkovanje, i druge, a s ciljem što bolje pripreme podataka za konačan postupak treniranja i prevenciju pogrešnih izlaznih rezultata.

Kao krajnji rezultat ovog projekta očekuje se izgradnja modela iterativnog učenja uz korištenje podataka izvan domene primjene te njegova usporedba s tradicijskim modelom učenja korištenjem odgovarajućeg testa statističkog zaključivanja. Tradicijski model razlikuje se od modela iterativnog učenja po tome što koristi podatke isključivo iz domene primjene odnosno uvijek radi samo na jednom skupu podataka. Pod podacima izvan domene primjene misli se na podatke s kojima se model susreće po prvi puta odnosno prethodno nije imao znanja o njima te se nadamo da će mu to znanje koristiti u boljoj kategorizaciji ostatka aktivnosti.

Poglavlje 2

Peptidi

Radi boljeg shvaćanja projekta i njegovog krajnjeg cilja, kratko će se opisati pojam peptida, što su oni i kako se kategoriziraju.

2.1 Aminokiseline

Jednostavno rečeno, aminokiseline su molekule koje sačinjavaju peptid. Postoji oko 500 poznatih amino kiselina, dok postoji 20 prirodnih amino kiselina koje sačinjavaju proteine ljudskog tijela i većine živućih organizama (eng. *proteinogenic amino acids*). [3] Svaki od peptida ima svoje posebnosti te se razlikuju u svojstvima, funkcijama, što je prikazano u tablici 2.1.

- Hidrofobnost - svojstvo koje posjeduju hidrofobne aminokiseline, ne vole boraviti u okruženjima sličnim vodi za razliku od hidrofilnih koje borave u takvim okruženjima,
- Aromatičnost - svojstvo koje posjeduju aromatične aminokiseline, sadrže aromatski prsten, stabilne cikličke strukture,
- Naboj - svojstvo koje posjeduju aminokiseline, može biti neutralan, pozitivan ili negativan,

- Polaritet - svojstvo koje posjeduje polarna aminokiselina, sadrži neravnomjernu raspodjelu elektrona po molekuli i može sadržavati neutralni, pozitivni ili negativni naboj, dok nepolarna aminokiselina ima ravnomjernu raspodjelu elektrona i neutralni naboj,
- Alifacitet - svojstvo koje posjeduje alifatska aminokiselina, može biti nepolarna i hidrofobna,
- Amfifilnost - svojstvo koje posjeduje amfifilna aminokiselina koja je također lipofilna (*voli masti*) i hidrofilna.

Kategorija peptida	**Naziv peptida**	**Jednoslovna oznaka**	**Troslovna oznaka**
Polarno pozitivni	Histidin	H	His
	Lizin	K	Lys
	Arginin	R	Arg
Polarno negativni	Aspartinska kiselina	D	Asp
	Glutaminska kiselina	E	Glu
Polarno neutralni	Serin	S	Ser
	Treonin	T	Thr
	Asparagin	N	Asn
	Glutamin	Q	Gin
Nepolarno alifatski	Alanin	A	Ala
	Valin	V	Val
	Leucin	L	Leu
	Izoleucin	I	Ile
	Metionin	M	Met
Nepolarno aromatski	Fenilalanin	F	Phe
	Tirozin	Y	Tyr
	Tritofan	W	Trp
	Prolin	P	Pro
	Glicin	G	Gly
	Cistein	C	Cys
Posebna svojstva	Asparagin \Aspartate	B	Asx
	Glutamin \Glutamate	Z	Glx

Tablica 2.1: Aminokiseline koje se pojavljuju u genetskom kodu, grupirane prema poznatim svojstvima Izvor: [1]

Svaka aminokiselina sadrži dvije funkcionalne skupine:

- amin, $-NH_2$, početna točka aminokiseline (N-terminus),
- karboksil, $-COOH$, završna točka aminokiseline (C-kraj),

- između spomenutih skupina je bočni lanac specifičan za svaku aminokiselinu koji određuje njezina svojstva.

Kao što je spomenuto, aminokiseline su molekule koje sačinjavaju peptid, tj. peptidi su kratki lanci aminokiselina povezanih peptidnim vezama, gdje se peptidna veza stvara između N-kraja jedne aminokiseline i C-kraja prethodne aminokiseline u lancu. Polipeptidi su duži, nerazgranati peptidni lanci do 50 aminokiselina. Ako polipeptidni lanac sadrži više od 50 aminokiselina, tada govorimo o proteinu.

2.2 Klasifikatori i deskriptori

Kako se aminokiseline razlikuju, tako se razlikuju i peptidi. Peptidi se kategoriziraju u različite skupine na temelju njihovih funkcija i izvora. Neki od njih uključuju gljivične peptide, biljne peptide, bakterijske ili antibiotske peptide, cjepivne peptide, peptide otrova i slično.

U ovome radu fokus će biti na predviđanju jedanaest kategorija peptida odnosno poboljšanju performansi predviđanja manjinske klase. Klase peptida korištene u sklopu projekta [4]:

- Antibakterijski - čine velik dio AMP-a i imaju široki inhibitorni učinak na uobičajene patogene bakterije,
- Antigljivični - podrazred AMP-a koji se bave gljivičnim infekcijama s povećanom otpornošću na lijekove,
- Antivirusni - snažan ubijajući učinak na viruse,
- Antiparazitni - ubijajući učinak na parazite koji uzrokuju bolesti poput malarije i lišmanije,
- Antikancerogeni - primjenjuju antikancerogene mehanizme regrutiranjem imunoloških stanica (poput dendritičnih stanica) za ubijanje tumorskih stanica, induciranjem nekroze ili apoptoze stanica raka, inhibiranjem angiogeneze kako bi se eliminirala pre-

hrana tumora i spriječilo metastaziranje, i aktiviranje određenih regulatornih funkcionalnih proteina čime ometaju transkripciju gena i translaciju tumorskih stanica,

- Antimikrobni,
- Antitumorski,
- Antiprotozojski,
- Insekticidni,
- Antigram minus,
- Antigram plus.

Osim podjele u klase, peptidi se mogu opisati nizom svojstava koja će se koristiti tijekom cijelog projekta i koja će biti jedan od parametara prilikom izgradnje modela. Neka od korištenih svojstava su sljedeća [5]:

- Ostaci - aminokiseline ugrađene u peptide,
- Duljina slijeda - duljina peptida, broj aminokiselina,
- Molekularna težina - ukupna molekularna težina aminokiselinskog slijeda,
- Sekvencijski naboj - neto naboj peptidnog slijeda. Može se izračunati na različitim pH vrijednostima na različitim skalama pKa, u ovom projektu korištena je Lehningerova skala,
- Bomanov indeks - Potencijalni indeks interakcije proteina, ova mjera opisuje sposobnost interakcije proteina, [6]
- Alifatski indeks - relativni volumen koji zauzimaju alifatski bočni lanci (Alanin, Valin, Izolevcin i Leucin). Alifatske aminokiseline odgovorne su za toplinsku stabilnost proteina, [7]
- Cruciani svojstva - širok skup deskriptora koji se temelje na interakciji svakog aminokiselinskog ostatka s nekoliko kemijskih skupina (ili "sondi"), poput nabijenih iona, metilnih, hidroksilnih skupina i tako dalje,

- Indeks hidrofobnosti - hidrofobnost se mijenja ovisno o otapalu u kojem se protein nalazi i važna je stabilizacijska sila u presavijanju proteina. U ovom se projektu koristila Eisenbergova ljestvica,
- Hidrofobni moment - mjera amfifilnosti okomite na os bilo koje periodične peptidne strukture,
- Izoelektrična točka - pH pri kojem određena molekula ili površina nema neto električni naboj,
- Indeks nestabilnosti - pokazuje je li peptid stabilan ili nestabilan na temelju njegove aminokiselinskog slijeda.

Poglavlje 3

Strojno učenje

Strojno učenje grana je umjetne inteligencije i njen glavni cilj je oponašanje ljudskog ponašanja kroz niz algoritama. Cilj svakog algoritma je razvijanje što većeg postotka točnosti oponašanja kroz postupak treniranja. U ovom poglavlju objasnit će se pojmovi neuronskih mreža, višeosposobljenog učenja te iterativnog učenja koji predstavljaju sastavne dijelove konačnog rješenja ovog diplomskog rada.

3.1 Neuronske mreže

Karakteristika neuronskih mreža je u tome što se sastoje iz više slojeva te se informacije prenose iz sloja u sloj. Važno kod svakog od slojeva je da radi isključivo s podacima koje je primio od prethodnog sloja bez dodatnih izmjena te da sačuva naučeno znanje, proširi ga i prenese ga u sljedeći sloj. Ako se desi da je informacija izgubljena u nekom od slojeva, svi slojevi koji slijede nakon sloja koji je izgubio određenu informaciju neće joj moći pristupiti, tj. informacija je tada zauvijek izgubljena. Svaka neuronska mreža se sastoji od ulaznih i izlaznih slojeva.

Neuronske mreže zapravo su skup algoritama koji pokušavaju oponašati ljudski mozak koji uči na temelju ulaznih podataka. Cilj oponašanja ljudskog mozga je u stjecanju sposobnosti rješavanja složenih problema i uočavanja uobičajenih obrazaca pojave u raznim

područjima poput medicine, poljoprivrede, tehničke industrije i slično.

Svaki čvor ili neuron jednog sloja ima pridijeljenu odgovarajuću težinu i prag (pristranost) te ako je izlazna vrijednost pojedinog čvora veća od praga, čvor se aktivira i prenosi informaciju sljedećem čvoru u mreži. Uloga težina svakog čvora je da se čvoru dodjeli njegova važnost što bi značilo da čvorovi s većim težinama imaju veći utjecaj na konačnu izlaznu vrijednost iz mreže.

Prilikom treniranja svake od neuronskih mreža, u svakom koraku želimo evaluirati točnost odnosno gubitak informacija. Cilj je minimizirati gubitak te ostvariti što veću točnost što bi značilo da model s određenom točnošću zna razlikovati i predviđati pojedine probleme.

Primjena neuronskih mreža je velika te se javlja u raznim aplikacijama poput predviđanja razvoja tržišta, otkrivanje prevara na granicama, razne predviđanja u području medicine i drugo.

3.2 Višeosposobljeno učenje

S obzirom na problematiku kategorizacije više aktivnosti (jedanaest), jedna od tehnika strojnog učenja koja se primijenila u ovome radu je i višeosposobljeno učenje. Ovom tehnikom nastoji se moći kategorizirati koju aktivnost posjeduje peptid proslijeđen kao ulazni parametar.

Često se miješaju pojmovi višerazrednog i višeobilježnog kategoriziranja podataka no ta dva pojma u potpunosti su različita i ne mogu se koristiti u kombinaciji. Razlika između ta dva načina kategorizacije je u tome što se smatra skupinom u koju se pojedini problem svrstava. Tako se kod višerazredne kategorizacije problem koji se razmatra želi kategorizirati točno u određenu skupinu dok kod višeobilježnog kategoriziranja nastoji se svrstati problem u neku od kategorija skupine kojoj pripada.

Različitost između višerazrednog i višeobilježnog kategoriziranja podataka možemo objasniti na primjeru gledanja filma. Filmovi imaju različite žanrove i svaka osoba ima drugačiji ukus, neki vole komedije, neki akciju i slično. Najčešće kod odabira filma gledamo

da udovoljimo čim većem broju ljudi odnosno tražimo film koji zadovoljava više žanrova te tu govorimo o višeobilježnoj kategorizaciji. Za razliku od višeoblježne kategorizacije, višerazredna kategorizacija nastojat će naći film koji pripada isključivo jednom od žanrova.

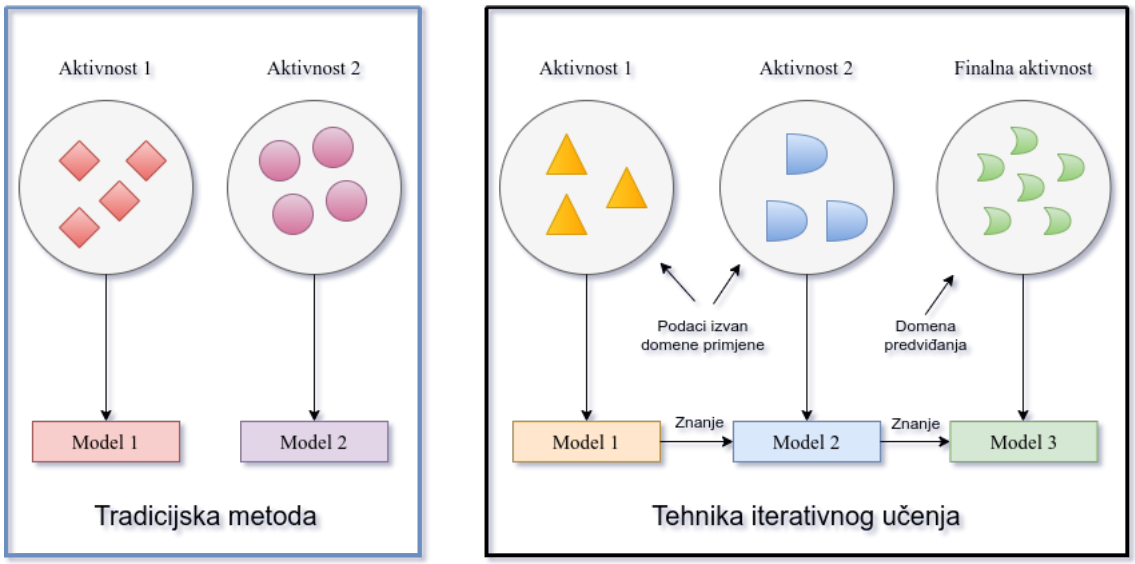
3.3 Iterativno učenje

Treniranje modela složen je i dugotrajan postupak, a svaka izmjena postojećeg modela zahtjeva ponovno izvršavanje algoritma. Kako bi se odgovorilo na uobičajene probleme treninga modela te prilagodbe modela na nove probleme javila se metoda iterativnog učenja. Primjenom metode iterativnog učenja proces treniranja modela znatno je kraći i nudi mogućnost proširivanja postojećeg modela na novom, prethodno nepoznatom problemu.

Princip rada metode iterativnog učenja je da se znanje pretreniranog modela iskoristi na nekom drugom, ali relevantnom problemu (slika 3.1). Pritom dolazi do prijenosa znanja iz pretreniranog modela u potpuno novi model. Dodatna prednost prijenosa znanja je što se blokovi prethodno izgrađenog modela mogu dodatno mijenjati kroz postupak zamrzavanja te se tako otvara mogućnost da se znanje modela proširi unutar samog tijela pretreniranog modela. Ono što se želi postići metodom iterativnog učenja je iskorištavanje stečenog znanja na jednom zadatku kako bi se poboljšala i olakšalo učenje na nekom drugom problemu.

Dodatna motivacija za primjenu metoda iterativnog učenja je u količini izvornog skupa podataka. Naime, ako želimo raditi na problemu za koji nemamo puno podataka, znanje stečeno na problemu za koji smo imali dovoljan broj podataka može pomoći u kategorizaciji novog problema. Ovime ne samo da se omogućava učenje o problemu s malim skupom podataka već se i štedi na samom vremenu učenja modela pošto ne moramo raditi učenje modela iz nule već započinjemo s već naučenim znanjem.

Važno je da prilikom primjene metode iterativnog učenja značajke koje se koriste budu generalne odnosno da se i kod treniranja novog modela koriste u istom formatu i da budu u istom kontekstu. Također, ulazni skup podataka mora imati isti oblik kao i izvorno trenirani model, inače je potrebno napraviti odgovarajuću prilagodbu ulaznih podataka.



Slika 3.1: Tehnika iterativnog učenja

3.4 Razvijeni modeli

Treniranje modela provedeno je korištenjem dva različita pristupa. Prvi pristup je kao skup ulaznih podataka prosljeđivao informaciju o svojstvima peptida, njih 32, nad kojima se radio opisani postupak *analize glavnih komponenti*. Drugi pristup odbacio je svojstva peptida kao dio ulaznih podataka te je fokus bio na peptidnim sljedovima.

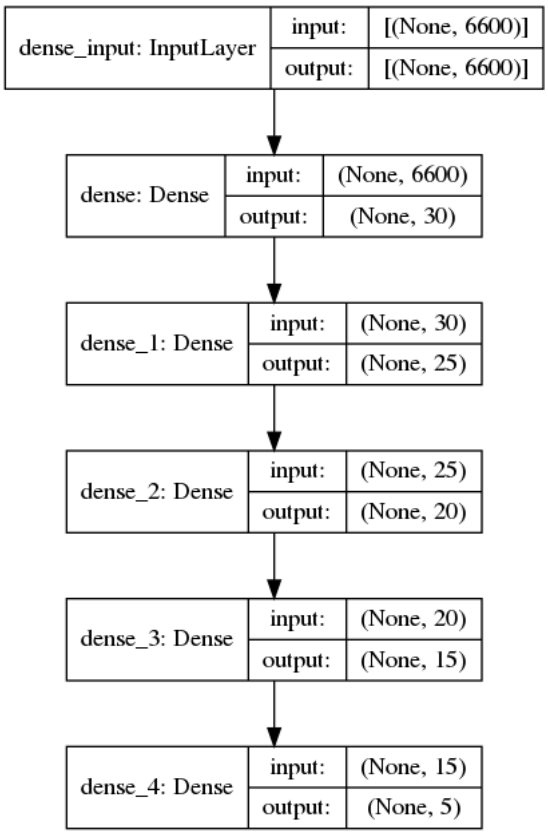
U cilju treniranja modela koji najbolje kategorizira podatke razvijena su dva modela od kojih je jedan jednostavna neuronska mreža s par jednostavnih slojeva te drugi složenija mreža koja sadrži više blokova te se zasniva na konvolucijskim slojevima.

3.4.1 Jednostavna neuronska mreža

Ulazni podaci sastoje se od jedne dimenzije stoga je prvi korišteni pristup bio izgradnja jednostavnog modela koji zna raditi samo s jednom dimenzijom. Iako su se koristila dva pristupa prilikom modeliranja ulaznih podataka, te se oni također razlikuju u dimenzionalnosti, jednostavna neuronska mreža početni je model koji bi mogao dobro raditi za oba pristupa.

Model se započinje graditi iz baznog *Sequential* modela koji je po strukturi linearni

stog slojeva koji se u njega dodaju. U bazni model dodan je jedan blok kojega čine *gusti* (Dense) slojevi te model završava krajnjim slojem koji je namijenjen za kategorizaciju kako je prikazano slikom 3.2.



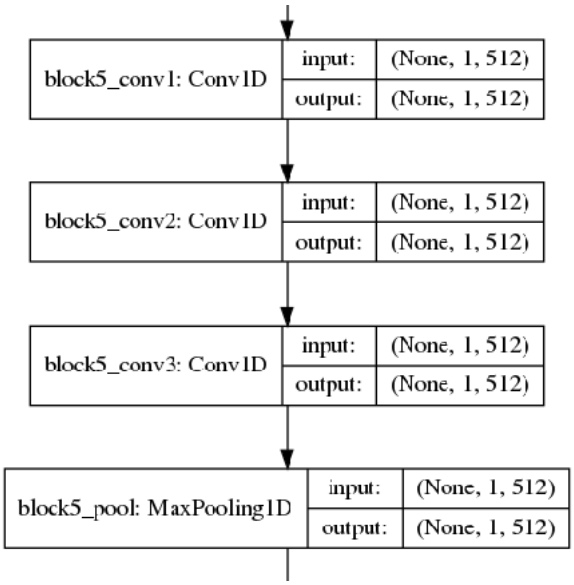
Slika 3.2: Grafički prikaz slojeva jednostavne neuronske mreže za predviđanje pet kategorija

Važno je da se ulazni podaci prosljeđuju modelu u istom obliku kod provođenja tehnike iterativnog učenja pošto se u suprotnom ne može definirati model.

3.4.2 Konvolucijska 1D mreža

U cilju treniranja modela koji bi učio na temelju ulaznih sljedova peptida razvijena je i konvolucijska 1D mreža pošto se sljedovi peptida podvrgavaju postupku *One-hot* kodiranja 5.4 čime se povećava dimenzionalnost ulaznih podataka u mrežu. Prema tome ovaj model strukturom je puno složeniji od modela koji se koristio u cilju predviđanja aktivnosti peptida kada su ulazne značajke bile svojstva peptida.

Razvijeni model, koji radi na principu konvolucijskih 1D mreža, sastoji se od pet



Slika 3.3: Primjer jednog bloka konvolucijske 1D mreže

blokova (slika 3.3) koji sadrže konvolucijske (Conv1D) i *polling* (MaxPooling1D) slojeve te završnog bloka koji je namijenjen za kategorizaciju ovisno o ulaznom broju kategorija. Bazni model na kojemu se gradio završni model također je linearni stog slojeva, *Sequential*.

3.5 Alati

Strojno učenje koristi se sve češće u cilju razvoja modela te postoji širok skup alata koji olakšavaju rad. U nastavku će se opisati korišteni alati koji su pomogli u izgradnji modela te evaluaciji dobivenih rezultata.

3.5.1 Python

Python je programski jezik opće namjene, interpretiran i visoke razine kojeg je stvorio Guido van Rossum 1990. godine (prva javna inačica objavljena je u veljači 1991. godine), ime dobiva po televizijskoj seriji Monty Python's Flying Circus. Objektno orijentirano, strukturno i aspektno orijentirano programiranje stilovi su dopušteni korištenjem Pythona te ova fleksibilnost čini Python programski jezik sve popularnijim. Python se najviše koristi na Linuxu, no postoje i inačice za druge operacijske sustave.

Unutar IT zajednice česte su kritike Pythona na račun njegove sporosti. Pošto je Python interpreterski jezik, programi napisani u njemu vrše se malo sporije za usporedbu od kompajlerskih jezika, kao što su C, C++ i slični. Međutim, unatoč toj brzinskoj manjkavosti, u industriji se Python poprilično koristi (ponajviše kao back-end programski jezik).

Python se često uspoređuje s Javom. Oboje su interpreterski jezici, i oboje imaju gotovo nikakvu podršku za višejezgreno izvođenje programa, pošto i Python i Java koriste samo jednu procesorsku jezgru. Java je kao jezik puno primjenjenija u izradi mobilnih aplikacija i interaktivnog web sadržaja, dok je Python gospodar PC svijeta. Što se tiče brzine izvođenja programa, Java i Python su približno jednaki. [8]

3.5.2 Tensorflow i Keras

Tensorflow i Keras jedni su od najpopularnijih modula za provođenje postupka strojnog učenja u područjima računarstva i podatkovne znanosti. Tensorflow je radni okvir (*framework*) dok je Keras knjižnica koja je dostupna u okviru Tensorflow-a. Keras se razvio u cilju kreiranja slojeva za kreiranje neuronskih mreža na konceptima oblika i matematičkih detalja.

Koraci prilikom izgradnje modela korištenjem Keras knjižnice su sljedeći:

- Učitavanje podataka,
- Obrada učitavanih podataka,
- Definiranje modela,
- Kompajliranje modela,
- Treniranje odabranog modela,
- Vrednovanje modela,
- Provođenje predviđanja nad novoistreniranim modelom,
- Spremanje rezultirajućeg modela.

3.5.3 Scikit-learn (Sklearn)

Scikit-learn (Sklearn) je besplatna softverska knjižnica strojnog učenja namijenjena za programski jezik Python. Sadrži razne algoritme kategorizacije, regresije i klasterizacije, uključujući potporne vektorske strojeve (SVM), slučajne šume, k-sredinu i DBSCAN, a dizajniran je za interakciju s Python numeričkim i znanstvenim knjižnicama NumPy i SciPy. [9]

Poglavlje 4

Podaci

S obzirom na to da je područje peptida i istraživanje njihovih aktivnosti tek u ranoj fazi te nema puno prikupljenih i testiranih podataka, izvorni skup podataka je ograničen. Prikupljanje podataka nije jednostavan pothvat pošto ono iziskuje određena znanja iz područja biotehnologije i kemije te zahtjeva razne proračune kako bi se konstruiralo prihvatljivo rješenje nakon čega treba proći i fazu testiranja.

Izvorni skup podataka stoga se uzimao iz postojećih repozitorija prethodno prikupljenih podataka te će se ovim poglavljem opisati izvor podataka te kako su oni distribuirani.

4.1 DRAMP

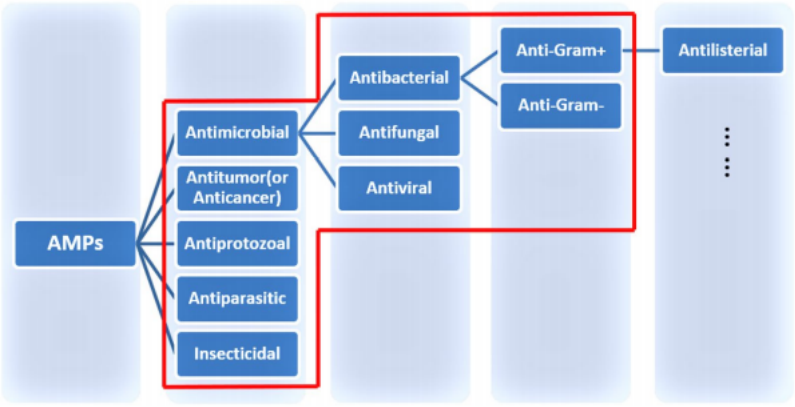
Repozitorij podataka antimikrobnih peptida (eng. *Data repository of antimicrobial peptides* - DRAMP) je baza podataka koja je javno dostupna te sadrži patentirane i kliničke *antimikrobne peptide* (AMP). Iako baza sadrži tisuće AMP-ova, samo je nekoliko dijelova ušlo u kliničku fazu. [2] Sa službene stranice DRAMP [1], vidljivo je kako se baza stalno ažurira što je motivacija za buduća istraživanje pošto će izvor podataka rasti, a to svakako znači više informacija koje mogu biti od velikog značaja prilikom razvoja modela strojnog učenja.

[1] http://dramp.cpu-bioinfor.org/

DRAMP dijeli podatke u četiri dijela:

- Prema porijeklu,
- Prema izvoru podataka,
- Prema taksonomiji,
- Prema aktivnosti.

Svaki od navedenih dijelova baze DRAMP sadrži različite značajke peptida koje se mogu analizirati. Fokus ovog diplomskog rada bit će upravo aktivnosti *antimikrobnih peptida* za koje baza DRAMP sadrži čak jedanaest kategorija 2.2. Sve aktivnosti u bazi DRAMP čine aktivnosti *antimikrobnih peptida*, koji predstavljaju jednu od glavnih skupina peptida, te one unutar sebe imaju dodatne podijele na podkategorije i podaktivnosti. Među podkategorijama možemo odrediti i ovisnosti među njima. Primjerice, antibakterijska, antigljivična i antivirusna aktivnost podkategorija su antimikrobijske aktivnosti dok antibakterijska aktivnost roditeljska je anti-gram aktivnostima što je prikazano slikom 4.1.



Slika 4.1: Prikaz ovisnosti između pojedinih aktivnosti *antimikrobnih peptida*. Izvor: [2]

4.2 Distribucija podataka

Iako u korištenom izvornom skupu podataka svi peptidi imaju značajke antimikrobnih peptida, na temelju svojstava se napravila detaljnija raspodjela kako je prikazano

tablicom 4.1. Unutar svake od kategorija dodatno postoje dvije skupine podataka, pozitivni uzorci odnosno uzorci za koje se utvrdilo da stvarno pripadaju promatranoj kategoriji i negativni uzorci koji mogu sadržavati i svojstva neke druge klase.

Aktivnost	**Ukupno podataka**	**Prosječna duljina peptidnog slijeda**
Antimikrobna	10341	33
Antibakterijska	8871	32
Antigljivična	7228	34
Antivirusna	5911	33
Antiparazitna	5745	33
Antikancerogena	5778	33
Antitumorska	5711	33
Antiprotozojska	5718	33
Insekticidna	5797	33
Antigram minus	7361	31
Antigram plus	7539	31

Tablica 4.1: Distribucija podataka po aktivnostima (kategorijama)

Iz distribucije prikazane tablicom 4.1 možemo primijetiti kako aktivnosti s većim skupom podataka zapravo čine one aktivnosti koje se dodatno dijele u podkategorije stoga je za očekivati da se među tim skupom podataka nalaze i druge aktivnosti. Pored toga svaka od kategorija aktivnosti u sebi sadrži i negativne uzorke koji će se promatrati kao posebna kategorija prilikom predikcije aktivnosti.

Poglavlje 5

Metodologija

Prije provedbe postupka treniranja potrebno je odraditi niz koraka kako bi osigurali vjerodostojne rezultate i ogradili se od mogućih pogrešaka. U tom smislu opisat će se postupci čišćenja i obrade podataka, validacije rezultata te kako se provodio postupak treniranja koristeći različite skupove ulaznih podataka. Dodatno će se opisati koraci prilikom primjene tehnike iterativnog učenja.

5.1 Organizacija podataka

S obzirom na to da se izvorni skup podataka sastoji od aktivnosti koje mogu biti ovisne o nekim drugim aktivnostima (4.1) i pri tome u svakom skupu podataka imamo pozitivne i negativne uzorke potrebno je odraditi svojevrsno *čišćenje* i organiziranje podataka.

Kako bi se radilo samo nad podacima koji pripadaju pojedinoj kategoriji napravljena je podjela na pozitivne i negativne uzorke. Ovaj postupak značajno je smanjio početni skup podataka no postupak je neophodan kako bi izbjegli usmjeravanje modela u pogrešnom smjeru tijekom treninga. Raspodjelom kategorija na pozitivne i negativne uzorke nastao je novi skup podataka, a to je skup podataka koji sadrži nekategorizirane podatke odnosno taj skup podataka se promatra kao nova kategorija. Ovime je početni problem od kategorizacije jedanaest aktivnosti dobio dodatnu skupinu, klasu neutralnih aktivnosti odnosno onih koje

ne pripadaju specifično nekoj od izvornih kategorija.

Uvođenje nove kategorija trebalo bi pomoći postupku treniranja jer ako je model nesiguran kojoj kategoriji pridružiti određeni peptid, neutralna kategorija bi mogla biti opcija za kategorizaciju.

Osim raspodjele podataka na dvije skupine, podaci se između kategorija preklapaju te postoje ovisnosti između njih. Detaljnom analizom i praćenjem distribucije podataka utvrđeno je kako aktivnosti koje su nadređene nekim drugim aktivnostima sadrže iste podatke čime se složenost kategorizacije i razina zbunjenosti modela tijekom učenja povećava.

Kako bi se odgovorilo na ovaj problem napravljeno je odbacivanje ovisnih aktivnosti iz nadređene klase. Dodatan izazov u ovom postupku bio je i sam broj podataka kojega kategorija sadrži pošto se podjelom podataka na pozitivne i negativne uzorke izvorni skup podataka dosta smanjio. Prema tome, trebalo je uzeti u obzir broj podataka klase kako se ne bi došlo u situaciju istrebljivanja pojedine klase odnosno trebalo je osigurati postojanje svih kategorija.

Temeljem svega navedenoga jasno je kako je krajnji skup podataka znatno smanjen te je to dodatna stavka koja je ovaj cjelokupan proces učinila znatno kompliciranijim nego što li je to uobičajeno kada se radi postupak strojnog učenja i kategorizacije podataka.

5.2 Analiza glavnih komponenata

Principal Component Analysis (PCA) nenadzirana je, neparametarska statistička tehnika koja se primarno koristi za smanjenje dimenzionalnosti u strojnom učenju. [10] Glavna motivacija za korištenje tehnike *analize glavnih komponenti* je u smanjenju broja značajki koje se koriste prilikom izgradnje završnog modela. Ulazni skup podataka sastoji se od sveukupno 32 značajke od kojih neke mogu biti veoma važne odnosno temelj znanja prilikom treniranja modela dok ostatak može biti beznačajan te unositi određenu razinu šuma u cjelokupan proces.

Jedna od prednosti primjene PCA tehnike je u prevenciji pretreniranosti modela

što je čest slučaj ako se koristi veliki broj značajki. Svakom dodatnom značajkom smanjuje se varijabilnost modela i uvodi se nepotreban šum stoga se primjenom ove tehnike smanjuje šum i algoritam radi s čišćim podacima koji ga neće usmjeravati u pogrešnom smjeru.

Prije provedbe ove metode potrebno je provesti odgovarajuću normalizaciju podataka pošto PCA pronalazi komponente s najvećom varijancom, a ako podaci nisu skalirani na određene jedinice to može dovesti do pogrešne usporedbe varijance među značajkama. Nakon provedene analize PCA tehnikom, samo 10 značajki pokazalo se kao zaista značajnima za proces treniranja te su ostale značajke odbačena.

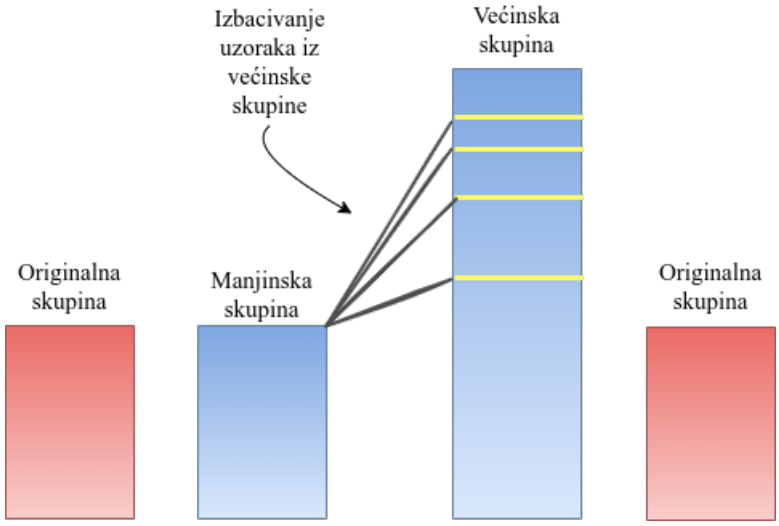
5.3 Neuravnoteženost podataka

Ulazni skupovi podataka nakon odrađenog čišćenja i organiziranja podataka rezultirali su različitim brojem uzoraka odnosno došlo je do neuravnoteženosti među podacima. Problem neuravnoteženih podataka je da ako je jedna kategorija većinska i dominira nad ostalim podacima može usmjeravati proces treninga i model u krivom smjeru. Bolje rečeno, neuronska mreža dobivat će više znanja o većinskoj kategoriji što će rezultirati lošijim performansama nad manjinskom klasom.

Kako bi se odgovorilo na problem neuravnoteženosti među podacima postoje metode poduzorkovanja i redundantnog otipkavanja. Obje metode rade na sličan način, jednoj je u cilju smanjiti broj podataka većinske klase na broj podataka manjinske dok se kod druge nastoje povećati uzorci manjinske klase kako bi model dobio određeno znanje i o njima.

5.3.1 Poduzorkovanje

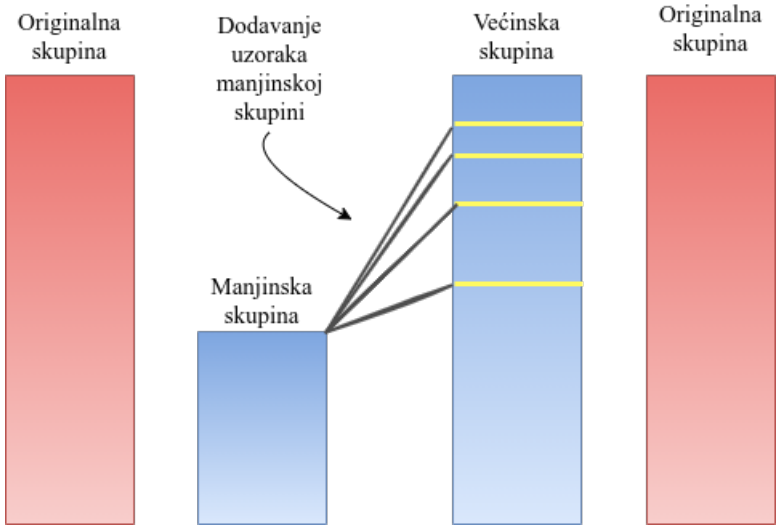
Ovim postupkom nastoji se smanjiti veličina skupa podataka većinske klase na razinu manjinske klase (slika 5.1). Jedna od mana ovog pristupa je što se smanjuje ulazni skup podataka pošto odbacujemo uzorke većinske klase, a podaci su veoma važni u postupku treniranja modela.



Slika 5.1: Tehnika poduzorkovanja

5.3.2 Redundantno otipkavanje

U cilju izbjegavanja loših performansi nad manjinskom klasom proveden je i postupak redundantnog otipkavanja koji radi na principu stvaranja duplikata iz skupa manjinske klase s približno sličnim svojstvima dupliciranog uzorka (slika 5.2).



Slika 5.2: Tehnika redundantnog otipkavanja

Isprobane su dvije tehnike redundantnog otipkavanja:

- Synthetic Minority Oversampling Technique (SMOTE),
- Adaptive Synthetic Sampling Approach - ADASYN.

SMOTE radi na principu pronalaženja najbližih susjeda među uzorcima manjinske klase. Nakon pronalaska susjeda povlači liniju između njih te generira nasumični sintetički uzorak koji predstavlja novi uzorak manjinske klase. [11]

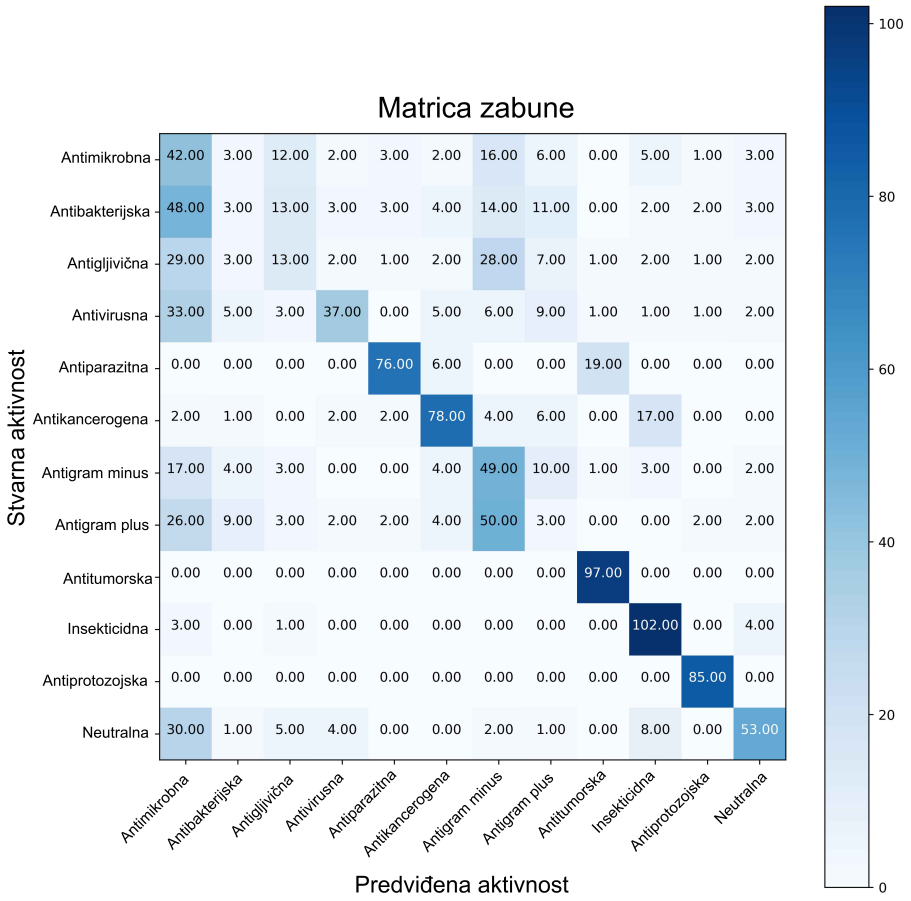
ADASYN je poboljšana inačica tehnike SMOTE. Poboljšanje u odnosu na SMOTE je što svakom novokreiranom uzorku pridodaje malu nasumičnu vrijednost čime taj uzorak postaje puno realniji pošto dodanom vrijednošću stvara uzorak sa svojstvima približno jednakima stvarnim, izvornim uzorcima. Dodavanjem nasumične vrijednosti, novokreirani uzorci manjinske klase dobivaju veći stupanj međusobne varijance. [11]

5.3.3 Pretreniranost

Ako se ne provede pravilna obrada podataka moguće je da treniranje modela rezultira "jako dobrim" rezultatima. Naime, prilikom treninga modela važno je samo da je proslijeđen skup ulaznih podataka nad kojima će se zasnivati razvoj znanja modela no provjera podataka kao takva ne postoji. Iz tog razloga priprema ulaznih podataka za proces treninga isključivo je odgovornost programera te ako se loše odradi rezultati mogu biti iznenađujući ili nemogući. Postoje različiti faktori koji utječu na pretreniranost modela te je važno njihovo pravovremeno uočavanje i prevencija kako bi se izbjegli netočni podaci u kasnijim analizama.

Jedan od mnogih faktora su sami podaci koji ulaze u postupak treniranja. Naime, nakon raspodjele podataka na skupove za trening odnosno test važno je da su podaci jedinstveni u oba skupa, tj. da se ne preklapaju. Ako se u skupu pripremljenom za trening javi neki od testnih podataka model će steći znanje o tim testnim podacima te prilikom validacije rezultata ti testni podaci neće odgovarati realnim uvjetima pošto se u realnosti očekuju podaci koje model vidi po prvi puta.

Kako je prikazano slikom 5.3, matrica zabune može se činiti veoma dobrom i ako se pravovremeno ne identificira da je razlog tome pretreniranost, možemo pogrešno predstavljati rezultate. Također, ako se radi temeljni model za neko veće istraživanje, cijelo istraživanje može biti nevažeće jer se temeljilo na pogrešnim podacima.



Slika 5.3: Pretreniranost modela kod predikcije svih dvanaest aktivnosti

Osim podjele ulaznih podataka važno je paziti i na dužinu treniranja modela jer dugo treniranje također može dovesti do promatranog problema.

Pretreniranost se može spriječiti korištenjem što većeg skupa podataka za treniranje, no često je teško prikupiti velike skupove podataka čime se još više stavlja naglasak na kvalitetno razumijevanje podataka i njihovu obradu.

5.4 Postupak *one-hot* kodiranja

Da bi se uspješno primijenilo strojno učenje na sljedove aminokiselina (peptide), prvo je bilo potrebno odraditi odgovarajuće kodiranje peptida. Korištena je metoda *prorije-*

đenog kodiranja, poznata i pod nazivom binarno kodiranje. U *prorijeđenom kodiranju*, svaka aminokiselina predstavljena je kao jedan vrući vektor (*one-hot*) duljine 20, pri čemu je svaka pozicija, osim jedne, postavljena na 0 pošto se zapravo radi o vektoru binarnih vrijednosti. Primjerice, u vektoriziranom formatu aminokiseline alanin i valin, koje predstavljaju prvu i posljednju aminokiselinu po abecednom poretku, kodiraju se kao 10000000000000000000 i 00000000000000000001. Na primjer, aminokiselinski slijed GHKARVLAEAMSQVTGSA-AVM, p2 peptid, kodiran je u matricu A kako je prikazano na slici 5.4.

$$A = \begin{array}{c} \\ G \\ H \\ \vdots \\ V \\ M \end{array} \begin{array}{c} \begin{array}{cccccccccccccccccccc} A & R & N & D & C & E & Q & G & H & I & L & K & M & F & P & S & T & W & Y & V \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

Slika 5.4: Primjer *one-hot* vektora za peptid GHKARVLAEAMSQVTGSAAVM

5.5 Stopa učenja

S obzirom na to da je treniranje neuronske mreže odnosno modela složen i problematičan optimizacijski zadatak potrebno je utjecati na njegovo pojednostavljenje. Osim pojednostavljenja složenog problema cilj je postići visoke performanse i veću brzinu treniranja u čemu nam pomaže stopa učenja koju je također moguće mijenjati i tijekom samog postupka treniranja. [12]

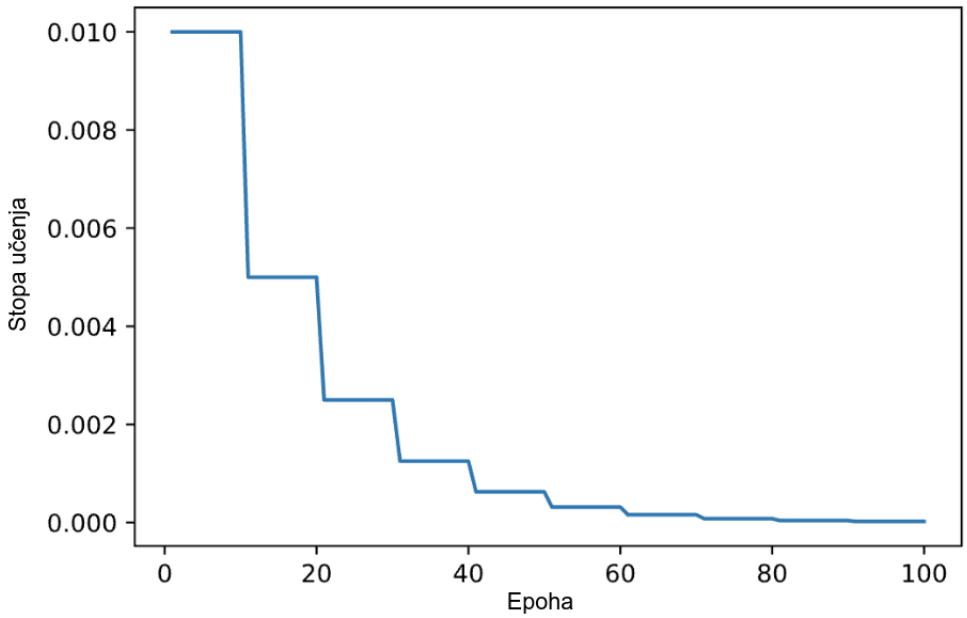
Korištenje principa rasporeda stope učenja u iteracijama se ažuriraju težine čvorova mreže kroz svaku epohu. Ovaj postupak naziva se kaljenje stope učenja ili prilagodljive stope učenja. Često korištene tehnike prilagodbe stope učenja nastoje smanjivati stopu učenja tijekom vremena treniranja. Prednost ovog pristupa je davanje slobode modelu da uči jako puno na početku procesa treninga uz veliku stopu učenja te ograničavanje učenja u kasnijoj fazi treninga sa što manjim izmjenama težina čvorova korištenjem manje stope učenja.

Primjenom ovog pristupa želi se ostvariti brzo učenje i osiguranje dobrih težina čvorova uz kasnije fino uglađivanje stečenog znanja.

Postoje dvije metode rasporeda stope učenja:

- Postepeno smanjivanje stope učenja kroz epohe,
- Smanjivanje stope učenja velikim padovima u određenim epohama.

U sklopu ovog diplomskog rada primijenila se tehnika smanjivanja stope učenja (*Drop-Based Learning Rate Schedule*) koja radi na principu skokovitog smanjivanja stope učenja u određenoj epohi. Kod ove tehnike započne se određenom stopom učenja te se ona smanjuje za zadani faktor svakih X epoha (slika 5.5).



Slika 5.5: *Drop-Based Learning Rate Schedule*

Stopa učenja se primjenom ove tehnike računa na sljedeći način:

$$stopa\_ucenja = pocetna\_stopa\_ucenja * faktor\_smanjivanja^{floor(epoha/korak)} \quad (5.1)$$

pri čemu je:

- *pocetna_stopa_ucenja* - zadana stopa učenja sa kojom se započinje tehnika smanjiva-

nja stope učenja,

- *faktor_smanjivanja* - vrijednost za koju smanjujemo stopu učenja u određenoj epohi,
- *epoha* - trenurna epoha za koju se računa nova vrijednost stope učenja,
- *korak* - zadani korak u kojemu se očekuje smanjenje stope učenja.

Funkcija koja računa smanjivanje stope učenja može se proslijediti u sam proces treniranja kroz *LearningRateScheduler* povratni poziv (*callback*) koji se brine da se stopa učenja mijenja u svakoj od zadanih epoha.

Kod tehnika izmjene stope učenja postoji skup praksi kojima se može utjecati na stopu učenja u cilju ostvarivanja što boljih rezultata. Veoma je važno krenuti s velikom početnom stopom učenja pošto će se stopa učenja svakako smanjivati tijekom vremena, a time ćemo dobiti velike modifikacije težina čvorova na početku i moći ćemo raditi fino uglađivanje kasnije s manjom stopom učenja. Potrebno je paziti na odabir faktora smanjivanja pošto ne želimo naglo izgubiti mogućnost da model razvije znanje. Zadnji, ali ne manje važan parametar je korak u kojemu se odlučimo raditi smanjivanje stope učenja. S ovim parametrom treba raditi pažljivo jer mali korak znači brže smanjivanje stope učenja što također može dovesti do slabijih rezultata.

5.6 Rano zaustavljanje algoritma

Još jedna funkcija povratnog poziva koja je korištena u procesu treniranja modela je funkcija ranog zaustavljanja algoritma, *EarlyStopping*. Glavna motivacija za korištenje ove funkcije je da se osigura prestanak treninga modela kada model dosegne određenu razinu zasićenja. Ovaj postupak također može pomoći u odabiru najboljeg mogućeg modela pošto se kao parametri mogu proslijediti koje metrike su nam bitne prije nego li prekinemo postupak treninga.

Jedna od metrika koja se prati u sklopu izgradnje modela u ovom radu je gubitak te je postavljen uvjet da se treniranje modela zaustavi ako ne postoji poboljšanje u vidu

gubitaka kroz tri epohe. Na ovaj način daje se prostor modelu za poboljšanje kroz tri epohe te ako poboljšanja nema, model je u zasićenju te se proces treniranja zaustavlja.

Moguće je da se u četvrtoj epohi javi novo poboljšanje, ali isto tako postoji i mogućnost pogoršanja performansi stoga je potrebno prije postavljanja funkcije povratnog poziva dobro znati ponašanje modela te prema tome postaviti parametre i metrike koje se gledaju.

5.7 Unakrsna validacija u k preklopa validacija

Treniranje modela samo po sebi može se pokazati kao jako dobro ili jako loše. Primjenom unakrsne validacija u k preklopa nastoji se osigurati da dobre ili loše performanse modela nisu slučajne. U tom vidu kroz k koraka odabiru se nasumični uzorci iz skupova za trening i test te se provodi postupak treninga modela. Važno je da se težine modela u svakom koraku vrate na početnu vrijednost kako se ne bi radilo dotreniravanje težina već prethodno izgrađenog modela. Osim podjele ulaznih skupova na k skupina koje se u koracima primjenjuju na isti model čije težine nakon svakog koraka vraćamo na početnu vrijednost možemo iznova trenirati model kroz k koraka te odabrati onaj s najboljim performansama.

Ovim postupkom dobiti ćemo bolji uvid u stabilnost modela te možemo testirati razne parametre poput omjera na koji ćemo dijeliti skupove podataka za trening i test koji nam mogu pomoći u odabiru najboljeg modela kojega možemo uzeti u konačnom treniranju. Korišteni broj preklopa za sve provedene analize u ovom radu je deset.

Glavni cilj evaluacije modela kroz provedbu validacije je vidjeti njegovo ponašanje nad različitim testnim podacima jer će oni biti važni za ocjenjivanje performansi modela u buduće kada bude zaista radio s podacima koje prethodno nije imao priliku vidjeti.

5.8 Treniranje modela

Nakon provedbe prethodno opisanih postupaka čišćenja i obrade podataka, može se započeti s postupkom treniranja modela. Postupak treniranja modela provodio se nad dva različita modela, modela koji se zasniva na jednostavnoj neuronskoj mreži (3.4.1) te modela čiji su temelj konvolucijske 1D mreže (3.4.2).

Osim korištenja različitih modela koristili su se i različiti skupovi ulaznih podataka. Peptidi su sljedovi amino kiselina te posjeduju različita svojstva. U sklopu treniranja modela nije moguće kombinirati tekst kao ulazni podatak i svojstva koja su brojčane vrijednosti. Iz tog razloga opisani su postupci treniranja modela kada ulazne podatke čine svojstva peptida, odnosno peptidni sljedovi.

5.8.1 Treniranje na temelju svojstva peptida

Kod odabira svojstava peptida kao ulaznih podataka bilo ih je potrebno proučiti kako bi mogli poduzeti odgovarajuće korake. Provedenom analizom glavnih komponenti, utvrđeno je kako nisu sva svojstva jednako značajna te su odabrana samo ona najznačajnija. Prema tome početni ulazni oblik u model koji je imao dimenziju *(broj_podataka, 32)*, pri čemu broj 32 predstavlja broj svojstava kojima je određen oblik, sveo se na *(broj_podataka, 10)* jer se samo 10 svojstava pokazalo kao značajno.

5.8.2 Treniranje na temelju peptidnih sljedova

Osim svojstava, sljedovi peptida kao ulazni podaci mogu usmjeravati trening modela na drugačiji način od svojstava te su stoga i oni uzeti u razmatranje. S obzirom na to da računalo radi primarno s brojevima potrebno je ulazne sljedove peptida provesti kroz postupak *One-hot* kodiranja.

Ulazni oblik u model nešto je drugačijih dimenzija nego li kod svojstava kao ulaznih podataka (dimenziju određuje duljina polja nakon što se višedimenzionalno polje svede na

jednu dimenziju).

5.9 Tehnika iterativnog učenja

Prije provođenja treninga modela tehnikom iterativnog učenja bilo je potrebno osigurati dobre performanse inicijalnog modela čije bi se znanje iskoristilo tako da bi se prenijelo u idući model.

Modeli se sastoje od blokova i slojeva te postoji mogućnost manipuliranja njima poput zamrzavanja, dodavanje novih blokova i slično. U tom vidu isprobani su različiti pristupi kod nadogradnje postojećeg modela:

- Zamrzavanje svih blokova inicijalnog modela osim jednoga,
- Zamrzavanje svih blokova inicijalnog modela te dodavanje novog bloka,
- Zamrzavanje svih blokova inicijalnog modela osim jednoga te dodavanje novog bloka.

Postoje i mnoge druge kombinacije koje se mogu primijeniti, a glavna ideja je da se očuva već stečeno znanje modela te da se omogući proširivanje toga znanja novim skupom podataka u istoj domeni.

Modifikacija prethodno izgrađenog bloka može rezultirati gubitkom informacije koju je model stekao u tom bloku stoga je u ovom radu iskorištena metoda zamrzavanja svih blokova prethodno treniranog modela uz dodavanje potpuno novog bloka s ciljem da novi blok proširi znanje cjelokupnog modela s novim skupom podataka.

Postupak treniranja konačnog modela primjenom tehnike iterativnog učenja uključivao je postepeno dodavanje aktivnosti. U prvom koraku iskoristio se inicijalni model koji je znao kategorizirati posjeduje li peptid jednu od aktivnosti ili ne. U svakom od sljedećih koraka uzimao bi se prethodno dobiveni model proširen dodatnim blokom te bi se trenirao na potpuno novoj aktivnosti.

Poglavlje 6

Rezultati

Razvijeni modeli trenirani su na dvije različite skupine podataka, jedna je uključivala svojstva peptida kao ulazne značajke dok se u drugoj kao ulazni parametar gledao peptidni slijed. S obzirom na dva različita načina izgradnje modela ovisno o ulaznim podacima, provedena je analiza posebno za svaki od njih.

Osim same analize i usporedbe jednog načina na prema drugome, glavni naglasak je na provjeri uspijeva li tehnika iterativnog učenja rezultirati boljim rezultatima nego li tradicijski postupak treniranja gdje se mreži proslijedi samo jedna specifična skupina ulaznih podataka.

Cilj analize je i pokazati kako se tehnike strojnog učenja snalaze u slučaju malog broja podataka te utječe li to značajno na performanse. Početna pretpostavka prije analize rezultata glasi da se očekuju bolji rezultati kada model nadograđuje svoje znanje te mali ulazni skup podataka ne bi trebao utjecati na znanje koje model već posjeduje. Sve analize provedene su na temelju treninga modela u trajanju od 50 epoha uz mogućnost ranijeg završetka preko funkcije povratnog poziva (pojašnjeno u poglavlju 5.6).

6.1 Svojstva peptida kao ulazni podaci

Prvi od pristupa koji se koristio za izgradnju modela koji bi mogao znati kategorizirati različite aktivnosti peptida je pristup kod kojega su ulazni podaci svojstva peptida, njih 32 (detaljno obrazloženo u poglavlju 2.2). Rezultati su podijeljeni u kategorije po modelima, pri čemu prva kategorija uključuje rezultate provedene korištenjem jednostavne neuronske mreže kao baznog modela, a druga je rađena korištenjem modela koji radi na principu konvolucijskih 1D mreža.

Obje kategorije dijele se u šest analiza:

- Predviđanja posjeduje li peptid određenu aktivnost ili ne,
- Pokušaj poboljšanja predviđanja jedne od aktivnosti kada se tehnikom iterativnog učenja prenosilo znanje iz modela u model,
- predviđanja svih jedanaest aktivnosti bez primjene tehnike iterativnog učenja,
- Pokušaj poboljšanja predviđanja svih jedanaest aktivnosti uz primjenu tehnike iterativnog učenja,
- Pokušaj predviđanja samo onih aktivnosti sa najvećim skupovima podataka bez primjene tehnike iterativnog učenja,
- Pokušaj predviđanja samo onih aktivnosti sa najvećim skupovima podataka uz primjenu tehnike iterativnog učenja.

Kako bi se moglo dati odgovor na pitanje poboljšavaju li se performanse modela kada se jednom naučeno znanje prenosi u sljedeći postupak treniranja važno je razumjeti kako radi inicijalno predviđanje koristeći tradicijske tehnike. Iz tog razloga, napravljena su predviđanja pomoću kojih se može vidjeti posjeduje li peptid određenu aktivnost ili ne. Provedbom postupka treniranja generirano je jedanaest izlaznih izvještaja kroz koje se pratila točnost i performanse.

6.1.1 Model na principu jednostavne neuronske mreže

U analizi modela temeljenog na principu jednostavne neuronske mreže, za svaku od jedanaest aktivnosti provedena je unakrsna validacija u deset preklopa, gdje prosječne vrijednosti kroz deset koraka možemo vidjeti u tablici 6.1.

Aktivnost	**Točnost (%)**	**Gubitak**
Antimikrobna	79.89 (± 0.55)	0.44
Antibakterijska	81.04 (± 1.53)	0.46
Antigljivična	74.72 (± 3.88)	0.57
Antivirusna	55.0 (± 5.73)	0.70
Antiparazitna	49.44 (± 10.38)	0.73
Antikancerogena	50.94 (± 9.17)	0.73
Antigram minus	80.51 (± 2.46)	0.51
Antigram plus	78.45 (± 1.97)	0.51
Antitumorska	50.0 (± 15.81)	0.69
Insekticidna	55.99 (± 9.94)	0.69
Antiprotozojska	50.0 (± 11.18)	0.71

Tablica 6.1: Točnosti i gubitci unakrsne validacije u deset preklopa kada su se koristila svojstva peptida kao ulazni podaci prilikom treniranja modela koji se zasniva na jednostavnoj neuronskoj mreži

Vidljivo je kako model ne predviđa jednako točno sve aktivnosti, a razlog tome je što za neke skupove aktivnosti postoji vrlo mali skup podataka te model na temelju njih ne uspijeva skupiti dovoljno znanja. Vrijednosti prikazane u tablici odgovaraju točnostima predviđanja nad testnim podacima.

Nakon provedene analize modela kroz unakrsnu validaciju u deset preklopa pristupilo se postupku treniranja modela te su rezultati prikazani u tablici 6.2.

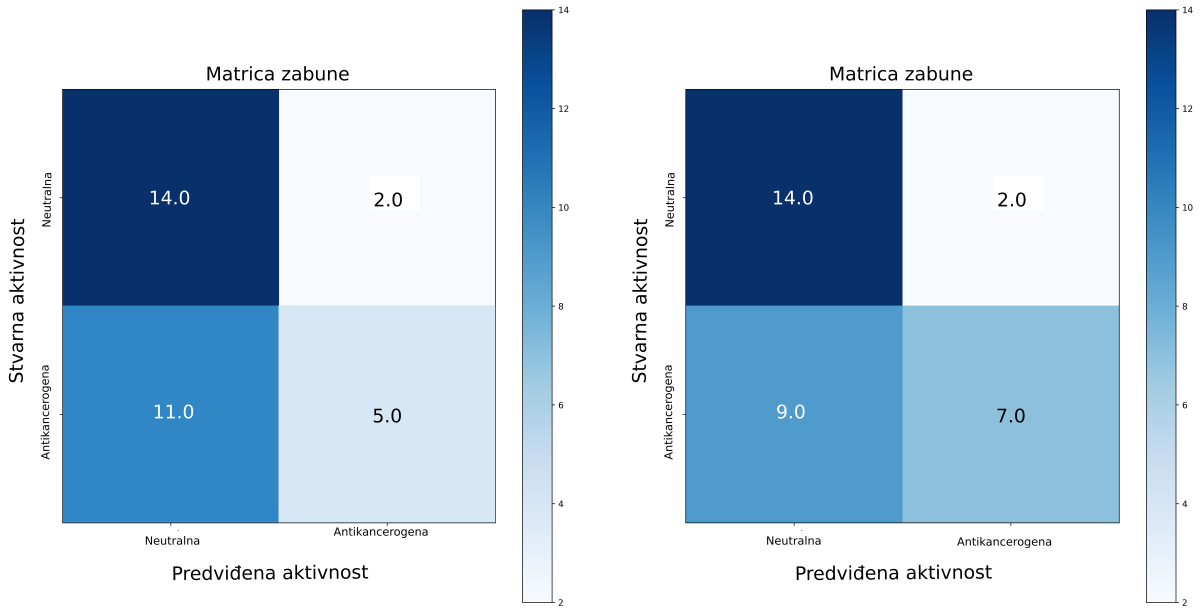
Zbog izrazito lošijih performansi nad aktivnostima čiji su ulazni skupovi podataka značajno manji, tehnikom iterativnog učenja pokušala se postići veća točnost. Za primjer je uzeta antikancerogena aktivnost. Kroz analizu unakrsne validacije u deset preklopa dobivena je prosječna vrijednost točnosti od čak 63% (62.5 ± 6.55) što je bolje od prethodnih 51% (50.94 (± 9.17)).

Kako se ne bi oslanjali samo na analizu unakrsne validacije u deset preklopa, analizirana je i točnost konačno izgrađenog modela nakon što se je u fazama izvornom modelu

Aktivnost	Točnost (%) - trening	Točnost (%) - test
Antimikrobna	80.32	79.74
Antibakterijska	79.65	81.47
Antigljivična	75.08	74.67
Antivirusna	53.57	44.047
Antiparazitna	50.0	55.6
Antikancerogena	53.33	68.75
Antigram minus	77.660	77.560
Antigram plus	76.45	74.04
Antitumorska	41.6	75.0
Insekticidna	50.87	42.50
Antiprotozojska	55.5	37.5

Tablica 6.2: Pregled točnosti izgrađenog modela koji zna kategorizirati posjeduje li peptid određenu aktivnost kada su se koristila svojstva peptida kao ulazni podaci prilikom treniranja modela koji se zasniva na jednostavnoj neuronskoj mreži

dodavalo znanje o ostalim aktivnostima tehnikom iterativnog učenja. Izvorni skup podataka za antikancerogene aktivnosti je značajno manji od ostatka, no završni model uspio je sa 66% (slika 6.1b) točnosti kategorizirati posjeduje li peptid značajke antikancerogene aktivnosti, što je također bolje od izvorne predviđanja bez primjene tehnike iterativnog učenja koja je uspjela kategorizirati peptid kao antikancerogen s točnošću od 59% (slika 6.1a).

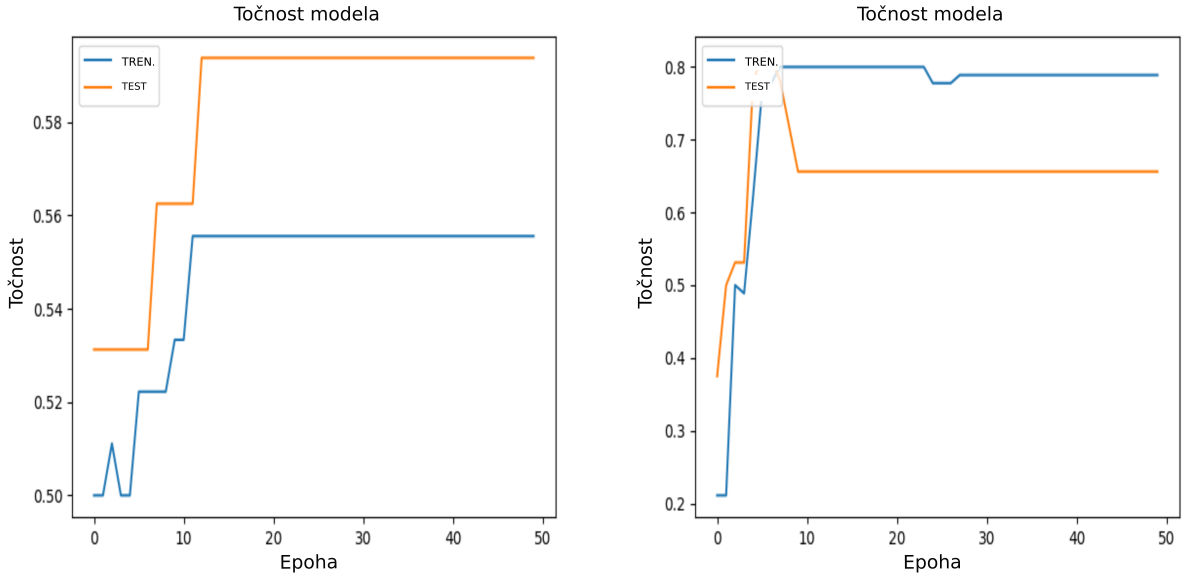


(a) Tradicijska metoda treniranja

(b) Tehnika iterativnog učenja

Slika 6.1: Prikaz matrica zabune kod predviđanja antikancerogene aktivnosti kada su se koristila svojstva peptida kao ulazni podaci prilikom treniranja modela koji se zasniva na jednostavnoj neuronskoj mreži

Kako bi pokazali kako je tehnika iterativnog učenja zaista rezultirala boljim rezultatima nego tradicijska metoda treniranja modela, možemo se poslužiti rezultatima točnosti kroz epohe. Mali skup ulaznih podataka može značiti da je točnost izgrađenog modela tradicijskom metodom zapravo slučajna pošto model nema dovoljno znanja kojim bi se učvrstila vjerodostojnost rezultata. To je također vidljivo iz sljedećih grafova, prikazanih na slikama 6.2a i 6.2b.



(a) Tradicijska metoda treniranja

(b) Tehnika iterativnog učenja

Slika 6.2: Prikaz kretanja točnosti tokom treninga kroz epohe kod predviđanja antikancerogene aktivnosti

Graf koji prikazuje kako se kretala točnost modela kroz epohe korištenjem tehnike iterativnog učenja ima manji broj skokova i konstantni uspon u prvih 10 epoha dok je trening modela korištenjem tradicijske tehnike podložniji većim skokovima u prvih 10 epoha i periodima bez napretka točnosti u kasnijim epohama.

S obzirom na neravnomjernu raspodjelu podataka i veoma male količine ulaznih podataka, treniranje modela u cilju predviđanja svih jedanaest aktivnosti odjednom veoma je složen zadatak. Primjenom tradicijske metode nije moguće postići zadovoljavajuće rezultate budući da model nema dovoljno informacija na temelju kojih bi znao kategorizirati aktivnosti peptida.

Provedbom validacije pokazano je kako tehnika iterativnog učenja u prosjeku daje

točnost kategorizacije od 7% (± 2.76) dok tradicijska metoda daje točnost od 8% (+- 3.93). Obje metode imaju izrazito loše rezultate predviđanja te iako naizgled tradicijska metoda dominira nad tehnikom iterativnog učenja, niti jedna od njih ne može se proglasiti boljom ili lošijom jer su korišteni mali skupovi podataka.

Konačno istrenirani modeli pokazali su kako je tradicijska metoda u mogućnosti kategorizirati svih jedanaest aktivnosti s točnošću od 8% (slika A.1 iz priloga) dok je predviđanje modela dobivenog tehnikom iterativnog učenja rezultiralo jednakim postotkom (slika A.2 iz priloga). Točnost modela kroz postupak iterativnog učenja dosta smanjuju podaci čiji su ulazni skupovi mali pošto u već izgrađeno znanje unose dosta šuma.

Odbacivanjem aktivnosti peptida za koje je pripadajući skup ulaznih podataka malen i uz odabir samo onih aktivnosti s najvećim skupovima podataka želi se dokazati da tehnika iterativnog učenja zaista dominira nad tradicijskom metodom treniranja. Rezultati također nisu na razini solidnih odnosno prihvatljivih razina točnosti no daju neku osnovnu motivaciju i prostor za poboljšanje ako se skupovi podataka povećaju. Istrenirani model primjenom tehnike iterativnog učenja zaista je rezultirao većom točnošću te je uspio kategorizirati pet aktivnosti s konačnih 37% dok je tradicijska metoda uspjela kreirati model koji je u mogućnosti točno predvidjeti pet kategorija s 35 postotnom točnošću.

6.1.2 Model na principu konvolucijske 1D mreže

Analiza i vrednovanje performansi provedena je i nad složenijim modelom koji se sastoji od konvolucijskih blokova. Za razliku od modela koji se zasniva na jednostavnim neuronskim mrežama, kod ovog modela postupak treniranja znatno je duži jer se sastoji od više blokova, a samim time uključuje više parametara.

Treniralo se tri različita tipa modela, (i) model koji zna predvidjeti svih dvanaest kategorija istovremeno, (ii) model koji zna predvidjeti posjeduje li peptid jednu, određenu aktivnost ili ne te (iii) model koji zna kategorizirati aktivnosti kada se u obzir uzimaju samo one s najvećim skupovima podataka odnosno radi se predviđanje nad pet kategorija istovremeno. U ovoj analizi također su se uspoređivale performanse tradicijske metode treniranja i metode koja uključuje tehniku iterativnog učenja.

Treniranjem modela za predviđanje jedne, određene aktivnosti koju posjeduje peptid koristeći tradicijsku metodu dobiveni su sljedeći rezultati:

Aktivnost	**Točnost (%) - trening**	**Točnost (%) - test**
Antimikrobna	99.5	85.6
Antibakterijska	98.5	87.46
Antigljivična	98.90	86.76
Antivirusna	95.6	66.6
Antiparazitna	90.38	72.2
Antikancerogena	91.1	84.4
Antigram minus	99.25	88.898
Antigram plus	97.69	85.19
Antitumorska	99.9	99.9
Insekticidna	85.08	87.5
Antiprotozojska	94.4	75.0

Tablica 6.3: Performanse izgrađenog modela koji zna kategorizirati posjeduje li peptid određenu aktivnost kada su se koristila svojstva peptida kao ulazni podaci prilikom treniranja modela koji se zasniva na konvolucijskoj 1D mreži

Performanse istreniranih modela znatno su bolje od onih kada se trening radio uz pomoć jednostavne neuronske mreže uz istu primjenu tradicijske metode. Zanimljivo je kako se kod predviđanja insekticidne aktivnosti javlja bolja točnost kod predviđanja testnih podataka te se to može tumačiti tako da znanje koje je model ostvario treningom nije pouzdano i vrijedilo bi pokušati ponovno istrenirati model i usporediti rezultate. Isto tako kod antimikrobne aktivnosti može se uočiti velika razlika u točnosti predviđanja nad podacima treninga i podacima testa te se može pomisliti kako se radi o pretreniranosti. Pod pojmom pretreniranosti ne mora značiti da se je model učio pogrešno na podacima koji će mu se kasnije dati za validaciju već jednostavno slučaj može biti da se je toliko dobro naučio na trening podacima te ne zna ništa izvan toga. U ovom slučaju veoma visokih performansi modela koji su trenirani tradicijskom metodom zanimljivo je vidjeti unosi li tehnika iterativnog učenja neko poboljšanje. Analizirat će se pokušaj poboljšanja performansi kod predviđanja posjeduje li peptid antikancerogenu aktivnost.

U tablici 6.4 možemo vidjeti kako se kretala točnost modela koji se gradio iz prethodno izgrađenog modela. Početni model znao je kategorizirati samo posjeduje li peptid antimikrobnu aktivnost ili ne, te se njegovo znanje nadograđivalo tako da su mu se postepeno dodavale nove aktivnosti koje je novi model trebao naučiti kategorizirati. Aktivnosti se

nisu dodavale prema nekom kriteriju te se nije vodilo računa o tome koliko podataka sadrži koja aktivnost.

Aktivnost	**Znanje o aktivnostima**	**Točnost (%)**
Antimikrobna	-	85.6
Antibakterijska	Antimikrobna	88.95
Antigljivična	Antimikrobna, Antibakterijska	90.8
Antivirusna	Antimikrobna, ..., Antigljivična	85.7
Antiparazitna	Antimikrobna, ..., Antivirusna	88.8
Antigram minus	Antimikrobna, ..., Antiparazitna	90.6
Antigram plus	Antimikrobna, ..., Antigram minus	88.6
Antitumorska	Antimikrobna, ..., Antigram plus	99.8
Insekticidna	Antimikrobna, ..., Antitumorska	55
Antiprotozojska	Antimikrobna, ..., Insekticidna	87.5
Antikancerogena	Sve prethodno navedene aktivnosti	81.25

Tablica 6.4: Koraci prilikom kreiranja završnog modela za kategoriziranje antikancerogene aktivnosti tehnikom iterativnog učenja kada su se koristila svojstva peptida kao ulazni podaci prilikom treniranja modela koji se zasniva na konvolucijskoj 1D mreži

Konačni model dobiven tehnikom iterativnog učenja prethodno je vidio sve aktivnosti osim antikancerogene (u tablici 6.4 pomoću tri točkice zamijenjeno je pisanje svih aktivnosti koje su se prethodno predavale modelima, tzv. viđene aktivnosti, dok se nije izgradio konačni model) i kao izlaz zna kategorizirati posjeduje li peptid antikancerogenu aktivnost s točnošću od 81%. Možemo primijetiti kako je točnost nešto manja nego li kod tradicijske metode, a to možemo pripisati aktivnostima s manjim skupovima podataka. Manji skupovi podataka unose šum u model koji već ima znanje o nekim aktivnostima te ne uspijeva dobro kategorizirati novu aktivnost za koju ima jako malo podataka.

Kako bi se uvjerili da manji skupovi podataka zaista jesu uzrok lošijih rezultata konačnog modela, proveo se trening tehnikom iterativnog učenja tako da se nisu uzele u obzir zadnje tri aktivnosti koje su imale dosta manje skupove ulaznih podataka. Točnost modela nakon treninga zaista se povećala te je približno jednaka točnosti dobivenoj pomoću tradicijske metode i iznosi 84.5% (prikazano u prilogu A.5). Ova točnost bila bi još veća ako se izbace one aktivnost s izrazito malim skupovima podataka ili uz pravilan redoslijed dodavanja novog znanja prethodnom modelu tijekom provedbe tehnike iterativnog učenja.

6.2 Peptidni sljedovi kao ulazni podaci

Kao i u prethodno spomenutoj analizi rezultata kada su se kao ulazni podaci koristila svojstva peptida i u sklopu ovih rezultata obradit će se šest analiza (navedenih u poglavlju 6.1) podijeljenih na dvije kategorije ovisno o korištenom modelu. Za razliku od prijašnje analize, sada se za skup ulaznih podataka koriste sljedovi peptida koji zahtijevaju dodatnu obradu, kao što je objašnjeno u poglavlju 5.4.

S obzirom na promjenu ulaznih podataka zanimljivo je vidjeti razlikuju li se rezultati nad istim provedenim postupcima treniranja modela.

6.2.1 Model na principu jednostavne neuronske mreže

Primjenom tradicijske metode treniralo se jedanaest aktivnosti i za svaku od njih se provela unakrsna validacija u deset preklopa, gdje su prosječne vrijednosti kroz deset koraka prikazane u tablici 6.5.

Aktivnost	**Točnost (%)**	**Gubitak**
Antimikrobna	88.57 (± 0.27)	0.256
Antibakterijska	88.43 (± 0.59)	0.27
Antigljivična	86.14 (± 0.96)	0.31
Antivirusna	61.79 (± 7.76)	0.65
Antiparazitna	48.89 (± 2.22)	0.698
Antikancerogena	60.0 (± 8.927)	0.67
Antigram minus	86.91 (± 1.06)	0.311
Antigram plus	86.236 (± 0.968)	0.33
Antitumorska	55.0 (± 15.0)	0.69
Insekticidna	62.75 (± 11.589)	0.66
Antiprotozojska	57.5 (± 10.0)	0.689

Tablica 6.5: Točnosti i gubitci unakrsne validacije u deset preklopa kada su se koristili slijedovi peptida kao ulazni podaci prilikom treniranja modela koji se zasniva na jednostavnoj neuronskoj mreži

Tradicijski model veoma dobro zna kategorizirati aktivnosti čiji su skupovi ulaznih podataka veći, no, kao i kod svojstava kao ulaznih podataka, loše radi kod aktivnosti s manjim skupovima. Nakon provedene analize modela kroz unakrsnu validaciju u deset preklopa

pristupilo se postupku treniranja modela te su rezultati dani tablicom 6.6.

Aktivnost	**Točnost (%) - trening**	**Točnost (%) - test**
Antimikrobna	92.74	88.79
Antibakterijska	92.43	88.8
Antigljivična	92.3	85.78
Antivirusna	85.71	75
Antiparazitna	88.461	72.22
Antikancerogena	76.66	53.125
Antigram minus	90.61	86.2
Antigram plus	89.88	85.59
Antitumorska	75.0	75.0
Insekticidna	80.70	69.99
Antiprotozojska	94.44	62.5

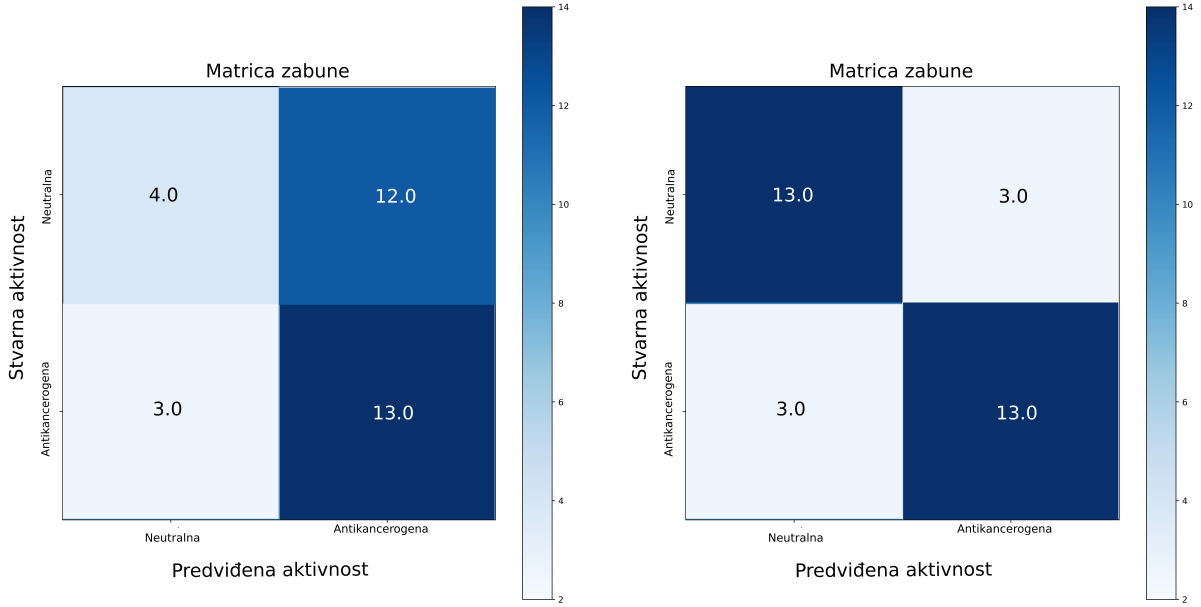
Tablica 6.6: Pregled točnosti izgrađenog modela koji zna kategorizirati posjeduje li peptid određenu aktivnost kada su se koristili slijedovi peptida kao ulazni podaci prilikom treniranja modela koji se zasniva na jednostavnoj neuronskoj mreži

U cilju poboljšanja predviđanja aktivnosti s manjim skupom ulaznih podataka izabrana je antikancerogena aktivnost za koju se pretpostavljaju bolji rezultati predviđanja nakon primjene tehnike iterativnog učenja. Kroz analizu unakrsne validacije u deset preklopa dobivena je prosječna vrijednost točnosti od čak 85% (84.687 ± 11.73) što je bolje od primjene tradicijske metode treniranja čija točnost iznosi 60% (60.0 ± 8.927).

Konačni model primjenom tehnike iterativnog učenja uspio je uz 81% točnosti (slika 6.3b) kategorizirati posjeduje li peptid značajke antikancerogene aktivnosti što je također bolje od izvorne predviđanja bez primjene tehnike iterativnog učenja koja je uspjela kategorizirati peptid kao antikancerogen s točnošću od 53% (slika 6.3a).

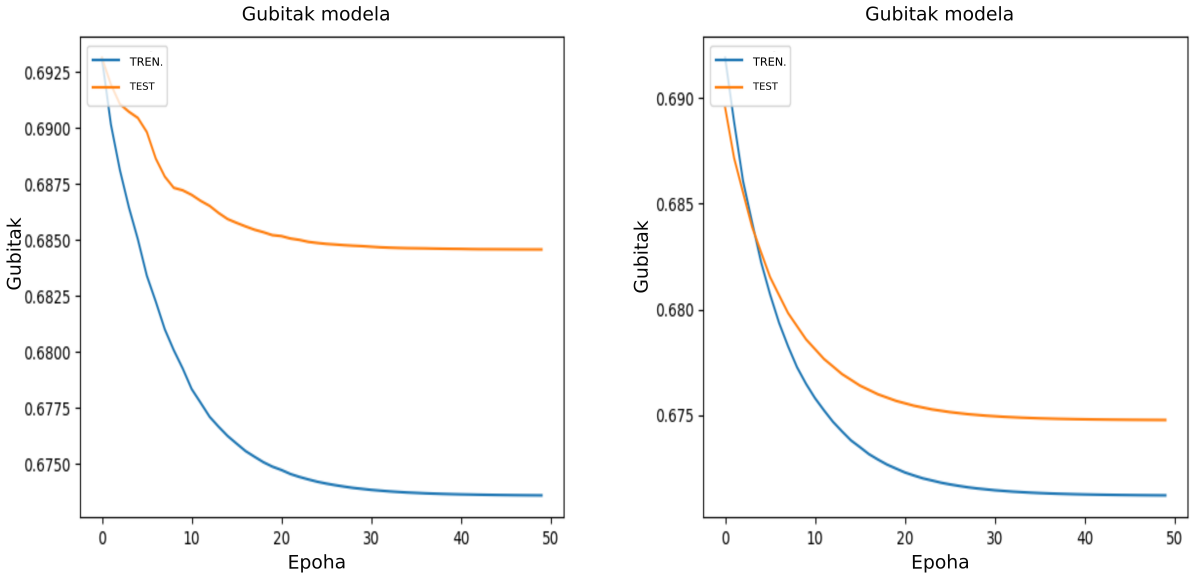
Vjerodostojnost prikazanih matrica zabune potvrđujemo grafovima na kojima se može vidjeti kako se kretao gubitak kroz epohe tijekom postupka treniranja modela. Putanja gubitaka bolje prati krivulju treninga u slučaju tehnike iterativnog učenja čime se može zaključiti da je rezultirajući model vjerodostojniji nego li kada se za trening koristi tradicijska metoda što nam pokazuju grafovi na slikom 6.4a i 6.4b.

Neovisno o tipu ulaznih podataka, ni tehnika iterativnog učenja, ni tradicijska metoda treniranja nisu uspjele polučiti prihvatljive rezultate za problem predviđanja svih jedanaest aktivnosti. Konačni model dobiven tradicijskom metodom treninga s točnošću od 8%



(a) Tradicijska metoda treniranja (b) Tehnika iterativnog učenja

Slika 6.3: Prikaz matrica zabune kod predviđanja antikancerogene aktivnosti kada su se koristili slijedovi peptida kao ulazni podaci prilikom treniranja modela koji se zasniva na jednostavnoj neuronskoj mreži



(a) Tradicijska metoda treniranja (b) Tehnika iterativnog učenja

Slika 6.4: Prikaz kretanja gubitka tokom treninga kroz epohe kod predviđanja antikancerogene aktivnosti

može kategorizirati kojoj aktivnosti pripada peptid. Model dobiven tehnikom iterativnog učenja daje bolje rezultate te je točnost predviđanja aktivnosti peptida 13%.

Kako bi se smanjio šum koji unose aktivnosti s malim skupovima ulaznih podataka u postupak treniranja modela tehnikom iterativnog učenja, napravljena je analiza kategorizacije samo onih aktivnosti s najvećim skupovima podataka.

Model dobiven tradicijskom metodom treniranja s točnošću od 35% može kategorizirati koju aktivnost posjeduje peptid dok model dobiven tehnikom iterativnog učenja za isti problem ima točnost od 36%. Točnosti oba modela su slične, slike A.3 i A.4 iz priloga, no model dobiven tehnikom iterativnog učenja ima više znanja što je vidljivo iz raznolikosti matrice zabune.

6.2.2 Model na principu konvolucijske 1D mreže

Posljednja provedena analiza uključuje sljedove peptida kao ulazni skup podataka te se trening provodio nad modelom koji se temelji na konvolucijskim 1D mrežama. U usporedbi s ostalim provedenim analizama, treniranje ovakvih modela pokazalo se najdužima, čak i do par sati dulje nego li kada su se kao ulazni skupovi podataka koristila svojstva peptida i kada se koristio model na principu jednostavne neuronske mreže.

S obzirom na to da su ulazni skupovi podataka sljedovi peptida te je proveden postupak *one-hot* kodiranja očekuje se da konvolucijski blokovi daju bolje rezultate u odnosu na jednostavniji model. Razlog ovakvog očekivanja je što se nakon provedenog postupka kodiranja povećava dimenzionalnost ulaznog skupa podataka, a s takvim podacima se očekuje da bolje rade konvolucijski blokovi koji se sastoje od više parametara za treniranje. U cilju testiranja performansi proveden je trening modela koji bi znao kategorizirati posjeduje li peptid određenu aktivnost ili ne koristeći tradicijsku metodu te su rezultati prikazani u tablici 6.7.

Zbog izrazito niske točnosti predviđanja antikancerogene aktivnosti iskoristila se tehnika iterativnog učenja u cilju poboljšanja performansi. Ovdje je također zanimljivo pogledati kako se kretala točnost modela nakon što se u svakom koraku dodavala nova aktivnost koju model još prethodno nije vidio (prikazano u tablici 6.8).

Usporedbom rezultata dobivenih tradicijskom metodom i onih dobivenih tehnikom

Aktivnost	**Točnost (%) - trening**	**Točnost (%) - test**
Antimikrobna	96.9	86.9
Antibakterijska	50.0	50.0
Antigljivična	50.0	50.0
Antivirusna	61.5	54.76
Antiparazitna	50.0	50.0
Antikancerogena	50.0	50.0
Antigram minus	75.6	75.0
Antigram plus	50.0	50.0
Antitumorska	91.67	50.0
Insekticidna	92.98	72.5
Antiprotozojska	83.3	62.5

Tablica 6.7: Performanse izgrađenog modela koji zna kategorizirati posjeduje li peptid određenu aktivnost kada su se koristili slijedovi peptida kao ulazni podaci prilikom treniranja modela koji se zasniva na konvolucijskoj 1D mreži

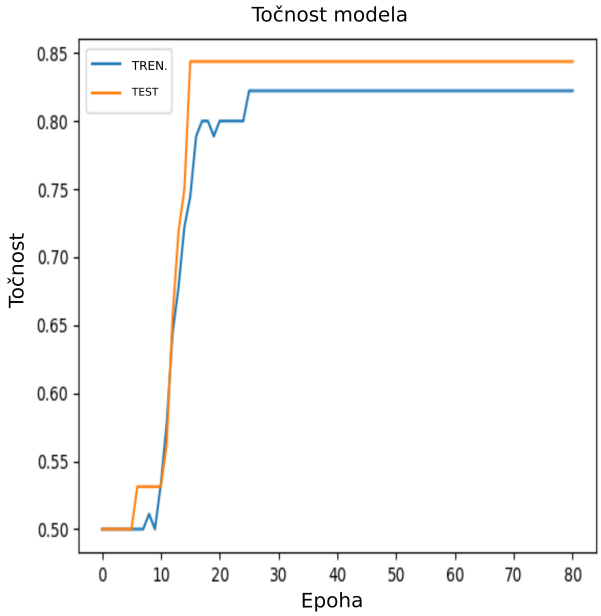
Aktivnost	**Znanje o aktivnostima**	**Točnost (%)**
Antimikrobna	-	86.9
Antibakterijska	Antimikrobna	89.8
Antigljivična	Antimikrobna, Antibakterijska	88.4
Antivirusna	Antimikrobna, ..., Antigljivična	83.3
Antiparazitna	Antimikrobna, ..., Antivirusna	61.1
Antigram minus	Antimikrobna, ..., Antiparazitna	87.04
Antigram plus	Antimikrobna, ..., Antigram minus	87.7
Antitumorska	Antimikrobna, ..., Antigram plus	99.9
Insekticidna	Antimikrobna, ..., Antitumorska	82.4
Antiprotozojska	Antimikrobna, ..., Insekticidna	99.9
Antikancerogena	Sve prethodno navedene aktivnosti	68.75

Tablica 6.8: Koraci prilikom kreiranja završnog modela za kategoriziranje antikancerogene aktivnosti tehnikom iterativnog učenja kada su se koristili slijedovi peptida kao ulazni podaci prilikom treniranja modela koji se zasniva na konvolucijskoj 1D mreži

iterativnog učenja vidljivo je kako su performanse znatno bolje uz primjenu tehnike iterativnog učenja te je točnost s 50% porasla na 68.75%.

Detaljnijom analizom kretanja krivulje točnosti kroz epohe (prikazano slikom A.6 u prilogu) može se primijetiti stagnacija kada se model trenira tradicijskom metodom te iako se treniranje provelo nad manjim brojem epoha (20), nema naznaka da bi se moglo postići značajnije poboljšanje. Isto tako može se primijetiti kako krivulja točnosti treninga ne prati najbolje krivulju testa što je još jedan dokaz kako kod malog skupa ulaznih podataka tradicijska metoda jednostavno ne može steći kvalitetno znanje za kategorizaciju aktivnosti.

S druge strane, kretanje krivulje točnosti kod primjene tehnike iterativnog učenja pokazuje strmi uspon, što se može vidjeti u prilogu A.7, te nakon spomenutog broja epoha (20) daje znatno bolju točnost (spomenutih 68.75%). Strmi uspon je znak da model ima potencijala za eventualna poboljšanja uz primjenu većeg broja epoha. Kako bi se uvjerili da je strmi uspon zaista značio potencijal za poboljšanje performansi, model se trenirao u 100 epoha (slika 6.5). Vidljivo je kako se predviđanje pokazalo boljim te je model uspio s točnošću od 84% kategorizirati posjeduje li peptid antikancerogenu aktivnost ili ne.



Slika 6.5: *Kretanje krivulje točnosti kroz 100 epoha kod predviđanja posjeduje li peptid antikancerogenu aktivnost ili ne*

Ako se bolje pogledaju krivulje točnosti za trening tradicijskom metodom i za primjenu tehnike iterativnog učenja, može se primijetiti kako kod tehnike iterativnog učenja krivulje treninga i testa prate jedna drugu. Na temelju ovog zapažanja može se reći kako model treniran uz primjenu tehnike iterativnog učenja ima više znanja, zbog prethodno ubačenih podataka, te pouzdanije može predvidjeti posjeduje li peptid antikancerogenu aktivnost ili ne. S druge strane, model treniran tradicijskom metodom radi samo sa znanjem skupa podataka koji su specifični za tu klasu i kada tome pridodamo problem male količine podataka razumljivo je da je pouzdanost modela manja.

Poglavlje 7

Rasprava

7.1 Usporedba korištenih pristupa

U provedenim analizama fokus je bio na podacima koji se šalju kao ulaz u proces treniranja modela. U tom smislu analizirana su svojstva i sljedovi peptida koji predstavljaju dvije različite skupine, ne samo prema broju značajki već i prema tipu podataka. Primjerice, sljedovi peptida su znakovni zapis gdje svako slovo predstavlja jednu aminokiselinu dok su svojstva peptida brojčane vrijednosti. Nad oba skupa ulaznih podataka provedeni su postupci obrade, tehnikom analize glavnih komponenti provedeno je smanjivanje dimenzionalnosti ulaznih značajki, a nad sljedovima peptida se proveo postupak *one-hot* kodiranja.

Kod oba pristupa analizirana su dva modela: model koji se zasniva na jednostavnoj neuronskoj mreži i model koji se zasniva na konvolucijskim 1D mrežama. Provedbom treninga inicijalnog modela tradicijskom metodom već se može primijetiti kako svojstva peptida daju lošije rezultate nego sljedovi peptida bez obzira koji na korišteni model (rezultati treniranja prikazani su u tablicama 6.1 i 6.5). Analizom rezultata treniranja inicijalnog modela izabrana je jedna aktivnost čiji su se rezultati pokazali daleko lošijim od ostatka te se za tu aktivnost, kao svojevrsna provjera inovativnog koncepta, nastojalo poboljšati performanse predviđanja primjenom tehnike iterativnog učenja.

Tehnika iterativnog učenja pokazala se djelotvornijom u slučaju sljedova peptida

gdje su poboljšanja vidljiva dok se kod svojstva peptida uspio ostvariti vrlo mali pomak. Primjerice, kod treniranja modela koji se zasniva na jednostavnoj neuronskoj mreži, tehnikom iterativnog učenja uspjelo se ostvariti predviđanje posjeduje li peptid antikancerogenu aktivnost s točnošću od 85% (84.687 ± 11.73) što je bolje od tradicijske metode koja je rezultirala točnošću od 60% (60.0 ± 8.927). U slučaju treninga nad modelom koji se zasniva na konvolucijskim 1D mrežama, tehnika iterativnog učenja također je bila uspješnija kada su se kao ulazni podaci proslijedili sljedovi peptida dok su performanse u slučaju svojstava kao ulaznih podataka bile u granicama performansi modela dobivenih tradicijskom metodom treniranja što je prikazano u 7.1.

Metoda	**Ulazni podaci**	**Točnost (%)**
Tradicijska	Svojstva peptida	84.4
Iterativno učenje	Svojstva peptida	84.5
Tradicijska	Slijedovi peptida	50.0
Iterativno učenje	Slijedovi peptida	84.0

Tablica 7.1: Usporedba rezultata treniranja tehnikom iterativnog učenja i tradicijskom metodom kod modela za predviđanje antikancerogene aktivnosti koji se zasniva na konvolucijskoj 1D mreži

Osim poboljšanja samo jedne aktivnosti pokušalo se raditi predviđanje svih dvanaest klasa odjednom no kod primjene tradicijske metode ili tehnike iterativnog učenja performanse su jednako loše zbog malih ulaznih skupova podataka. Tehnika iterativnog učenja ipak daje nešto bolje performanse konačnog modela, a razlog tome je što posjeduje prethodno znanje o ostalim aktivnostima čime bolje kategorizira novu aktivnost.

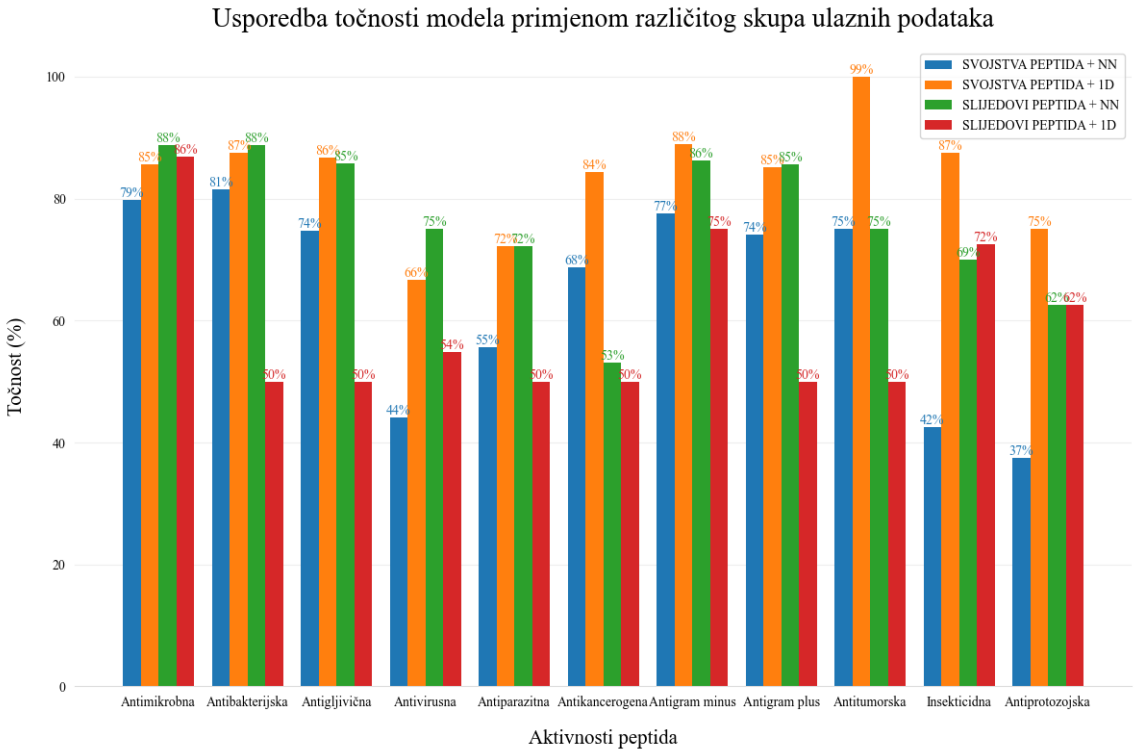
Kako bi se dalo odgovor na pitanje koliko jako utječu mali skupovi ulaznih podataka na sam izlazni model odradila se analiza predviđanja samo onih aktivnosti s najvećim skupom podataka te su rezultati prikazani u tablici 7.2.

Metoda	**Ulazni podaci**	**Točnost (%)**
Tradicijska	Svojstva peptida	35
Iterativno učenje	Svojstva peptida	37
Tradicijska	Slijedovi peptida	35
Iterativno učenje	Slijedovi peptida	36

Tablica 7.2: Usporedba rezultata treniranja tehnikom iterativnog učenja i tradicijskom metodom kod predikcije aktivnosti sa najvećim skupovim podataka na temelju modela koji se zasniva na jednostavnoj neuronskoj mreži

Opet se pokazalo da tehnika iterativnog učenja daje malo bolje rezultate no motivacija za njeno korištenje proizlazi iz rezultata matrica zabune gdje se može primijetiti veću raspršenost u odnosu na tradicijsku metodu što je znak da model posjeduje određeno znanje o svakoj aktivnosti odnosno o svakom prethodno viđenom problemu (bolji uvid u raspršenost rezultata prikazan je matricom zabune na slici A.3).

Ukupan pregled točnosti modela ovisno o skupu ulaznih podataka prikazan je slikom 7.1 gdje se može vidjeti i usporedba modela koji se zasniva na jednostavnoj neuronskoj mreži i onoga koji se zasniva na konvolucijskoj 1D mreži.



Slika 7.1: Pregled točnosti različitih modela i različitih skupova ulaznih podataka kod predikcije aktivnosti peptida

7.2 Izazovi

Cjelokupan projekt predstavlja relativno novo područje u svijetu biotehnologije i računarstva. Peptidi i računanje njihovih značajki složen je i dugotrajan proces te postoji vrlo malo radova koji se bave ovim problemom. S obzirom na navedeno jasno je kako postoje

vrlo mali skupovi podataka te je jako teško stvoriti potrebno okruženje za razvoj kvalitetnijih modela i predviđanja.

Svojstva peptida vrlo su slična za pojedine sljedove unutar istih skupina te samim time kategorizacija aktivnosti postaje teža jer isti peptid može biti kategoriziran u više skupina te tu nailazimo na problem ovisnosti i preklapanja 5.1.

Osim sličnosti svojstava u sklopu izgradnje modela strojnog učenja, nisu sva svojstva jednako značajna te mogu voditi model u krivome smjeru. Iz toga slijedi kako razumijevanje svojstava i odbacivanje onih manje značajnih može imati veliku ulogu za postizanje željenih rezultata.

Pravilno provođenje postupka kategorizacije aktivnosti od velikog je značaja zbog očiglednih prepreka i izazova koji su trenutno prisutni u području istraživanja i proučavanja peptidnih sljedova, njihovih aktivnosti i ostalih značajki.

7.3 Budući rad

Ovim radom učinjeni su prvi koraci u istraživanju širokog područja koje svakako ima potencijala za dodatan razvoj. Veći skupovi ulaznih podataka znatno mogu poboljšati predviđanja kreiranih modela, a samim time povećati performanse kod primjene tehnike iterativnog učenja.

Postoje i razne kombinacije provođenja treninga modela tehnikom iterativnog učenja koje nisu obrađene ovim radom te su vrijedne testiranja. Osim toga postoje razni drugi parametri čijom se modifikacijom također može utjecati na konačne rezultate.

Sama obrada podataka ima veliki utjecaj na konačni model, razvojem tehnologija svakako se očekuje poboljšanje i u tom segmentu te bi bilo od velikog značaja isprobati različite načine. Kao predmet razmišljanja također može biti kreiranje ulaznih podataka kojim bi se izbjegao dugotrajan i složen proces ručnog proučavanja uzoraka i računanja značajki pojedinih peptida. Svakako za ostvarenje ove ideje potrebno je razviti odgovarajuće modele koji bi na temelju prethodnog znanja kako određeni peptid, koji posjeduje specifično

svojstvo, treba izgledati mogli dati potrebne informacije i pomoći prilikom konstruiranja novih ulaznih podataka.

Radom je pokrivena izrada modela koji bi trebao znati kategorizirati koju aktivnost posjeduje određeni peptid gdje je aktivnost peptida jedan od parametara koji bi možda bio značajan u kreiranju novog ulaznog skupa podataka.

Poglavlje 8

Zaključak

Peptidi su sve popularniji u raznim istraživanjima no ipak za sada nema dovoljno prikupljenih podataka o njihovim karakteristikama niti postoji metoda kojom bi se ubrzao proces obrade. Čest je i slučaj kada se neki zadatak ne može riješiti zbog nedovoljno znanja ili nedostatka ključnog dijela za njegovu implementaciju. Ovim radom istražuje se mogućnost predviđanja aktivnosti koju posjeduje peptid čime se želi stvoriti svojevrsna pomoć znanstvenicima u pretraživanju novih spojeva, a s ciljem poboljšanja i ubrzanja postojećih procedura u tom području. Također, želi se omogućiti rješavanje problema uvođenjem podataka izvan domene primjene kako bi se osnažilo znanje koje već postoji o relevantnom problemu.

Pokazano je kako postoje različiti načini na koje se mogu trenirati modeli te se rezultati razlikuju i ovisno o ulaznim podacima koji se prosljeđuju u proces treniranja. Mali ulazni skupovi podataka predstavljaju problem kod treniranja modela tradicijskom metodom što je vidljivo iz točnosti koju dobivamo na konačnom modelu. Tehnika iterativnog učenja pokazala se kao odlična metoda za poboljšanje performansi prethodno treniranog modela te isto tako skraćuje i vrijeme potrebno za trening. Nadalje, ovom tehnikom moguće je prethodno treniranom modelu dodati znanje o nekom novom problemu u cilju poboljšanja performansi čime se stvara i mogućnosti da model proširi svoje znanje s novim, ali relevantnim problemom.

Konkretno poboljšanje ostvareno ovim radom može se vidjeti iz usporedbe točnosti

modela koji zna kategorizirati posjeduje li peptid određenu aktivnost. Točnost kategorizacije kancerogene aktivnosti primjenom jednog od postupaka uporabom tradicijske metode iznosi 53% dok uporabom tehnike iterativnog učenja točnost iznosi 81%. Također, pokazano je kako sljedovi peptida kao ulazni podaci rezultiraju boljim performansama nego li svojstva peptida što se može pripisati činjenici da model teže uči kada ima veći broj ulaznih značajki. U nekim postupcima pokazano je i kako tehnika iterativnog učenja ne povećava točnost za jako veliki postotak no ostvarivanje i malog poboljšanja nam govori da je stvarno moguće prenošenje znanja iz modela u model te dotreniravanje modela sa skupom podataka kojega model prethodno nije vidio.

Bibliografija

[1] Medicine Faculty of Biology and Health. peptide-properties. https://www.bmh.manchester.ac.uk/. [Online; accessed 3-Jul-2021].

[2] Xinyue Kang, Fanyi Dong, Cheng Shi, Shicai Liu, Jian Sun, Jiaxin Chen, Haiqi Li, Hanmei Xu, Xingzhen Lao, and Heng Zheng. Dramp 2.0, an updated data repository of antimicrobial peptides. *DRAMP*, 8 2019.

[3] Michael J. Lopez and Shamim S. Mohiuddin. Biochemistry, essential amino acids. *StatPearls*, 3 2021.

[4] Yuchen Huan, Qing Kong*, Haijin Mou, and Huaxi Yi. Antimicrobial peptides: Classification, design, application and research progress in multiple fields. *Front. Microbiol.*, 10 2020.

[5] Daniel Osorio. R Documentation, Peptides package. https://www.rdocumentation.org/packages/Peptides/versions/2.4.1. [Online; accessed 19-Jun-2021].

[6] Wang, G. Li, X., Wang, and Z. Apd3: The antimicrobial peptide database as a tool for research and education. *Nucleic Acids Research*, 44:D1087–D1093, 2016.

[7] Ikai and Atsushi. Thermostability and aliphatic index of globular proteins. *Journal of Biochemistry*, 88, 01 1981.

[8] Wikipedia. Python (programming language). https://en.wikipedia.org/wiki/Python_%28programming_language%29. [Online; accessed 21-Jun-2021].

[9] Wikipedia. scikit-learn. https://en.wikipedia.org/wiki/Scikit-learn. [Online; accessed 26-Jun-2021].

[10] Harsha Goonewardana. Pca: Application in machine learning. *Apprentice Journal*, 2 2019.

[11] Indresh Bhattacharyya. Smote and adasyn (handling imbalanced data set). *Medium*, 8 2018.

[12] Jason Brownlee. Using learning rate schedules for deep learning models in python with keras. *Deep Learning*, 6 2016.

Sažetak

Moderna istraživanja u domeni kemije peptida iziskuju pouzdane modele predviđanja koji će olakšati proces eksperimentalne pripreme i dublje razumijevanje njihovih svojstava. Cilj ovog diplomskog rada je povećanje pouzdanosti takvih modela koji se grade na relativno malim skupovima podataka upotrebom tehnike iterativnog učenja. Izgrađena su dva modela od kojih se jedan zasnivao na jednostavnim neuronskim mrežama, a drugi na konvolucijskim 1D mrežama.. Rezultati provedenih analiza potvrdili su potencijal tehnike koja koristi podatke izvan domene primjene te dovodi do poboljšanja točnosti u rasponu od 1% do 35% u odnosu na tradicijski postupak.

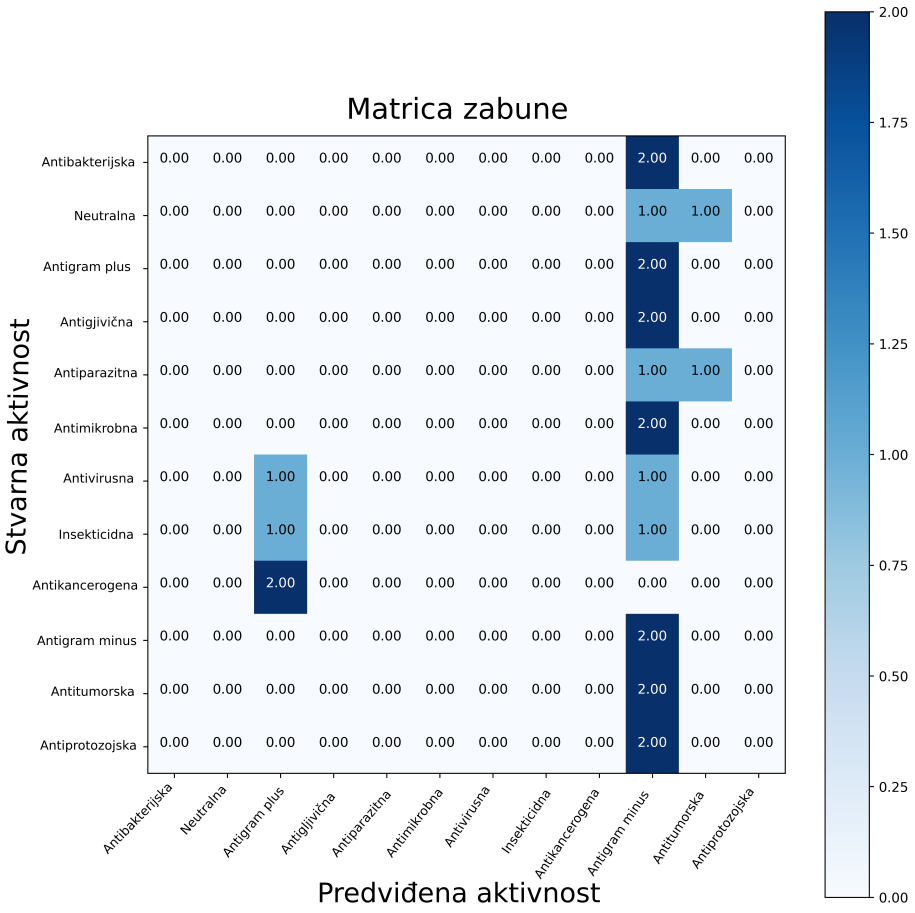
Ključne riječi: peptid, jednostavna neuronska mreža, konvolucijska 1D mreža, aktivnost peptida, tehnika iterativnog učenja, podaci izvan domene primjene.

Abstract

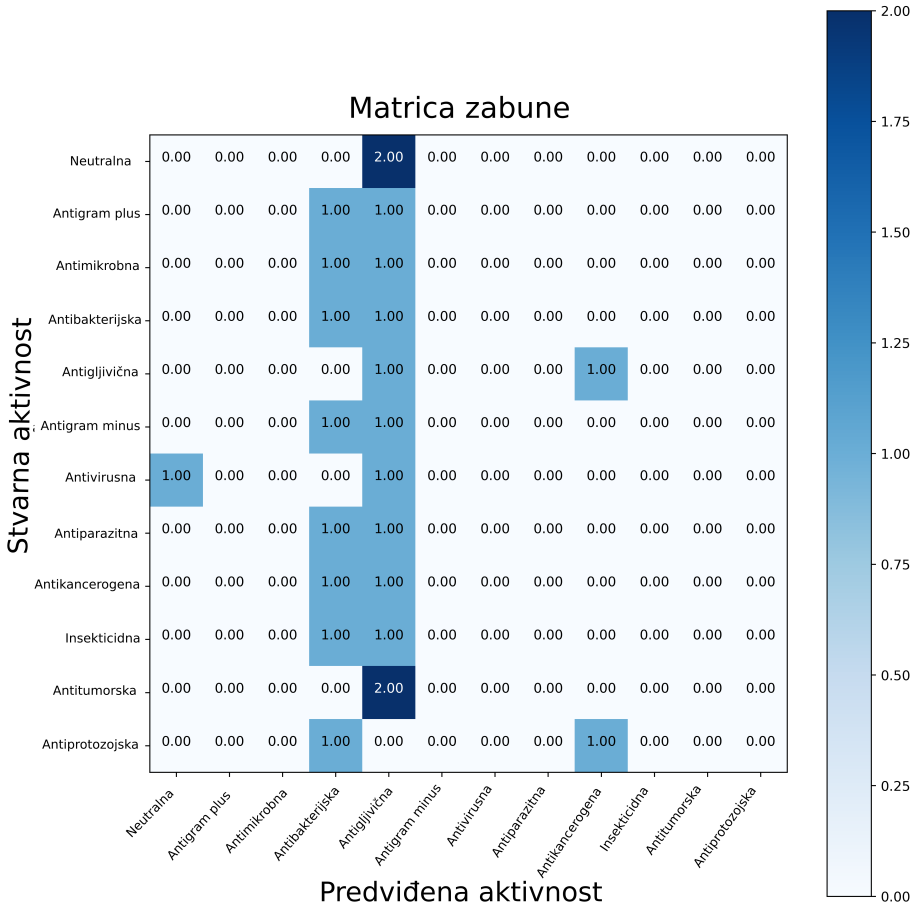
Modern research in the field of peptide chemistry requires reliable prediction models that will facilitate the process of experimental preparation and a deeper understanding of their properties. This thesis aims to increase the reliability of such models that are built on relatively small data sets using transfer learning. Two models were built, one based on simple neural networks and the other on convolutional 1D networks. The results of the conducted analyses confirmed the potential of the technique that uses data outside the domain of application and leads to an improvement in accuracy in the range from 1 % to 35 % compared to the traditional procedure.

Keywords: peptide, simple neural network, convolutional 1D network, peptide activity, transfer learning, unseen data.

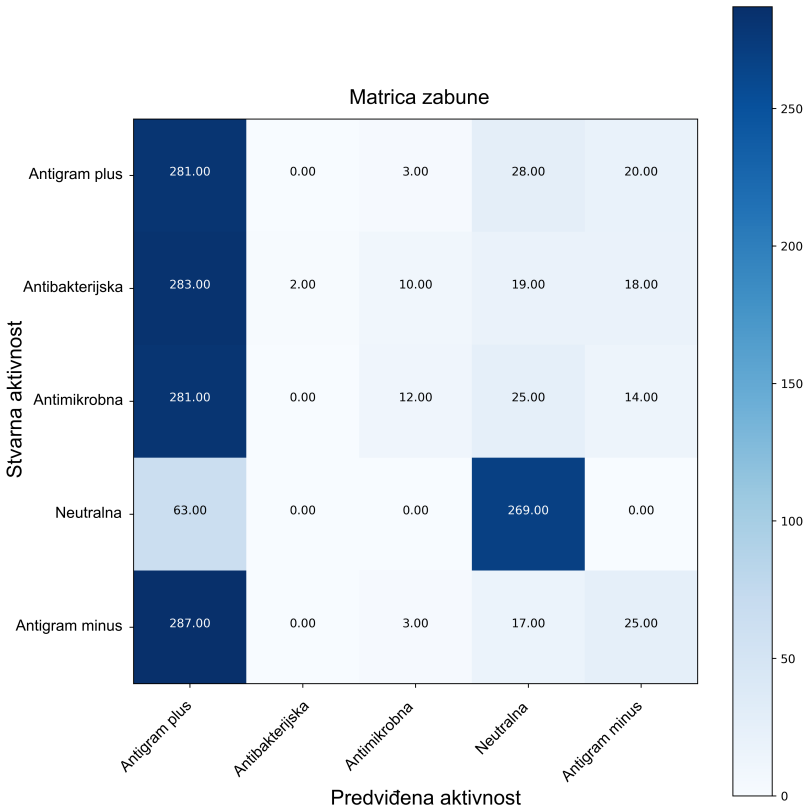
Prilozi



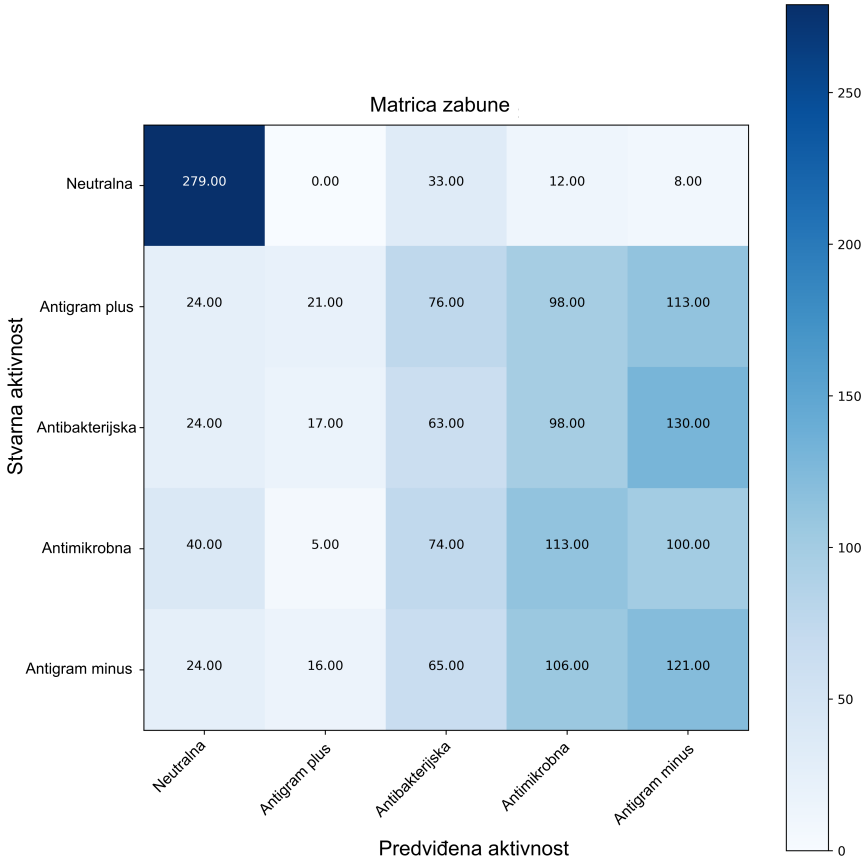
Slika A.1: Prikaz matrice zabune kod predviđanja svih jedanaest aktivnosti primjenom tradicijske metode treniranja



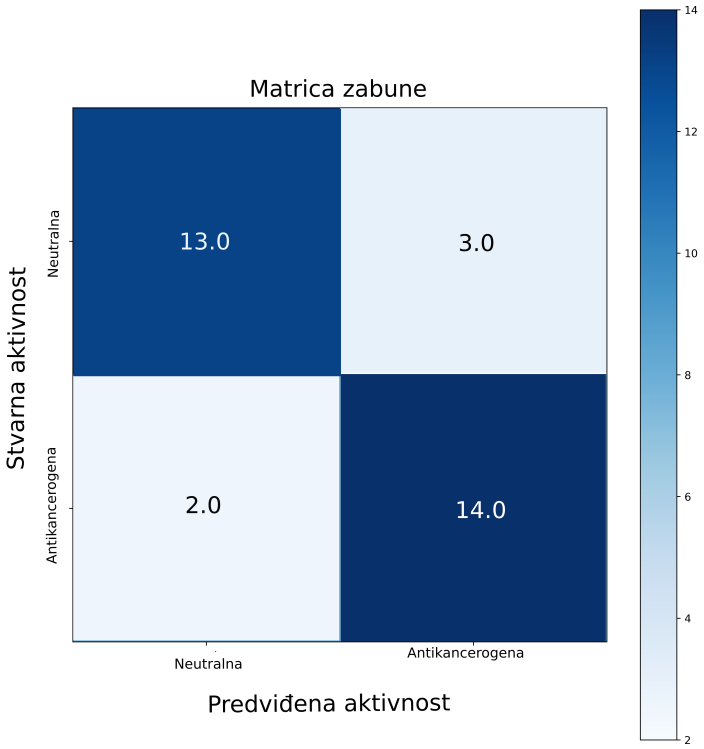
Slika A.2: Prikaz matrice zabune kod predviđanja svih jedanaest aktivnosti primjenom tehnike iterativnog učenja



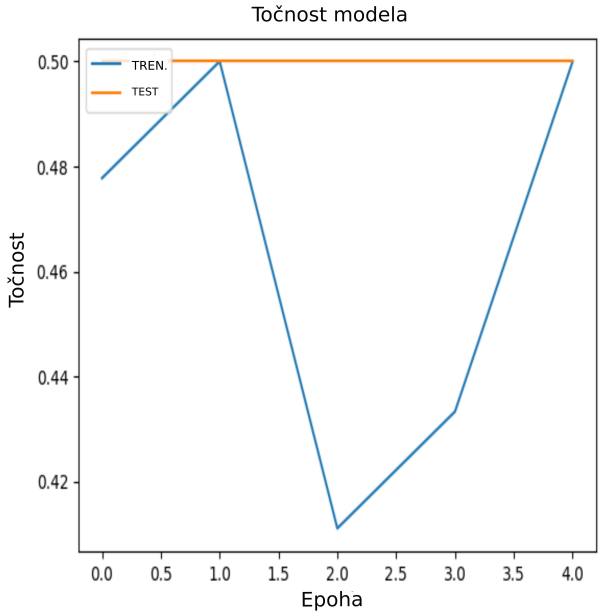
Slika A.3: Prikaz matrice zabune kod predviđanja pet aktivnosti sa najvećim skupom podataka dobivene tradicijskom metodom



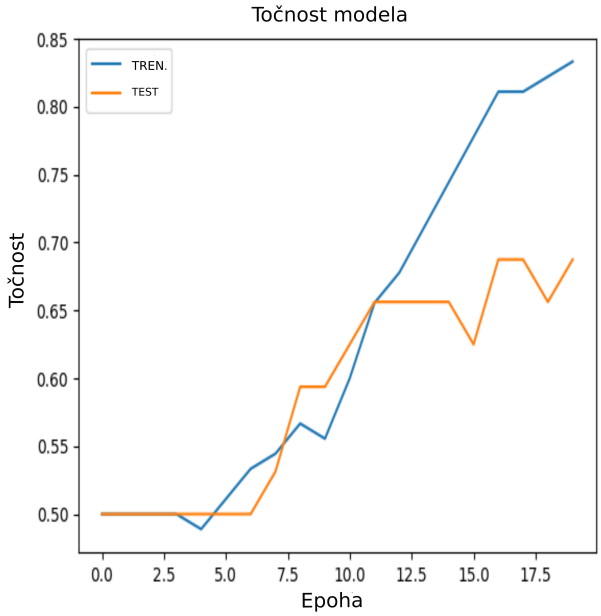
Slika A.4: Prikaz matrice zabune kod predviđanja pet aktivnosti sa najvećim skupom podataka dobivene tehnikom iterativnog učenja



Slika A.5: *Matrica zabune kod modela za predviđanje antikancerogene aktivnosti dobivenog tehnikom iterativnog učenja bez aktivnosti sa malim skupovima podataka*



Slika A.6: *Stagnacija krivulje točnosti kroz epohe kod predviđanja posjeduje li peptid antikancerogenu aktivnost ili ne*



Slika A.7: *Kretanje krivulje točnosti kroz 20 epoha kod predviđanja posjeduje li peptid anti-kancerogenu aktivnost ili ne*