

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET

Marko Njirjak

**GENERATIVNI MODELI
UMJETNE INTELIGENCIJE ZA
OTKRIVANJE PEPTIDA
CILJANIH SVOJSTAVA**

DOKTORSKI RAD

Rijeka, 2026.

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET

Marko Njirjak

**GENERATIVNI MODELI
UMJETNE INTELIGENCIJE ZA
OTKRIVANJE PEPTIDA
CILJANIH SVOJSTAVA**

DOKTORSKI RAD

Mentor: Prof. dr. sc. Goran Mauša
Komentor: Prof. dr. sc. Ivan Štajduhar

Rijeka, 2026.

UNIVERSITY OF RIJEKA
FACULTY OF ENGINEERING

Marko Njirjak

**GENERATIVE ARTIFICIAL
INTELLIGENCE MODELS FOR
THE DISCOVERY OF
PEPTIDES WITH TARGET
PROPERTIES**

DOCTORAL THESIS

Supervisor: Prof. Goran Mauša, PhD
Co-supervisor: Prof. Ivan Štajduhar, PhD

Rijeka, 2026.

Mentor rada: Prof. dr. sc. Goran Mauša (Sveučilište u Rijeci, Tehnički fakultet)
Komentor rada: Prof. dr. sc. Ivan Štajduhar (Sveučilište u Rijeci, Tehnički fakultet)

Doktorski rad obranjen je dana _____ na Tehničkom fakultetu
Sveučilišta u Rijeci, Rijeka, Hrvatska, pred povjerenstvom u sastavu:

1. Prof. dr. sc. Kristijan Lenac (Sveučilište u Rijeci, Tehnički fakultet) – predsjednik
2. Prof. dr. sc. Jonatan Lerga (Sveučilište u Rijeci, Tehnički fakultet) – član
3. Prof. dr. sc. Branimir Bertoša (Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet) – član

ZAHVALE

Najveću zahvalu zavrjedili su moji roditelji, koji su me cijeloga života neizmjereno voljeli i usmjeravali da postanem osoba koja sam danas. Hvala vam na ljubavi, empatiji, razgovorima, zagrljajima, brizi, smijehu i svemu ostalom što ste mi nesebično dali. Neizmjereno vas volim.

Velika hvala mojim bakama i djedovima. Čuvali ste me, pazili, nasmijavali i poticali da učim i ulažem u svoje obrazovanje. Hvala što ste vjerovali u mene i kada sam nisam.

M&M, riječima vjerojatno nikada neću uspjeti izraziti koliko mi značiš. Hvala ti na svemu. Volim te!

T&G, hvala na svemu. Ovaj put je bio više od samog doktorata.

Eco, hvala ti na svim druženjima, razgovorima i idejama. Tvoje šale mi uvijek poprave dan.

Hvala mojim prijateljima na vremenu provedenom zajedno. Drago mi je da smo tu jedni za druge i u lijepim i u teškim trenucima.

Hvala i mojim mentorima, prof. Mauši i prof. Štajduharu, te prof. Kalafatović na vođenju, savjetima i zajedničkom radu.

SAŽETAK

Računalna karakterizacija i dizajn novih molekula unaprijedili su tradicionalne metode otkrivanja terapijskih spojeva koje se uvelike oslanjaju na eksperimentalni probir. Ispitivanjem molekula u računalu u početnim fazama istraživanja, moguće je identificirati perspektivne ligande uz značajno niži utrošak vremena i resursa. Stoga ne čudi da danas gotovo svaki projekt istraživanja novih molekula uključuje neki vid računalnog probira, pri čemu umjetna inteligencija dobiva sve više na značaju. Iako se navedeno polje ubrzano razvija, još uvijek postoje specifični nedostaci koji otežavaju širu primjenu strojnog učenja i ostalih tipova algoritama. To su primarno točnost modela i mogućnost generalizacije znanja, te skupa eksperimentalna validacija. Ova disertacija predlaže moguća rješenja kroz nekoliko slijednih faza istraživanja na primjeru dizajna peptida ciljanih svojstava.

Najprije je predložen novi način numeričke predstave peptida u računalu, nazvan shemom sekvencijalnih svojstava, koji kombinira informacije o fizikalno-kemijskim svojstvima peptida s informacijama o njegovoj primarnoj strukturi. Shema tako pruža detaljniji uvid u sekvence, što modelima strojnog učenja posljedično daje bolju priliku za postizanje viših razina točnosti. Osim za identifikaciju specifičnih svojstava u peptidu, dobro naučeni modeli strojnog učenja mogu se upotrijebiti i kao polazište za dizajn novih peptida. Tako su predložena dva modela strojnog učenja: za procjenu sklonosti peptida ka hidrolizi estera i za procjenu sklonosti samosastavljanja, pri čemu je potonji uz informacije o sekvencijalnim svojstvima koristio i informacije o sklonosti agregaciji gradivnih jedinica peptida: aminokiselina, dipeptida i tripeptida. Oba modela korištena su u genetskim algoritmima za uspješno stvaranje novih, dosad nepoznatih katalitičkih i sekvenci sa svojstvom samosastavljanja. Slične ideje primijenjene su i na pronalazak inhibitora enzima beta-sekretaze 1, gdje je korišten višekriterijski genetski algoritam s jedne strane vođen procijenjenom energijom vezanja peptida za enzim molekularnim pristajanjem, a s druge strane Wildman-

Crippen logP vrijednošću i procjenom vjerojatnosti prolaska krvno-moždane barijere postojećim modelom strojnog učenja iz literature. Posljednja nit istraživanja nastojala je pospješiti lakoću eksperimentalnog vrednovanja molekula kroz razdvajanje knjižnica peptida na korisnički zadan broj disjunktnih podskupova, za lakšu sintezu i dekonvoluciju. Upotrijebljen je višekriterijski genetski algoritam usmjeravan ka minimiziranju međusobnog preklapanja podskupova i preklapanja peptida po masi unutar svakog podskupa, te zadržavanju maksimalnog mogućeg broja peptida iz izvorne knjižnice.

Ostvareni rezultati potvrđuju uspješnost predstavljenih modela i algoritama, te daju značajan poticaj njihovoj primjeni u praksi. Time ova disertacija predlaže moguća rješenja za neke od istaknutih problema područja računalne karakterizacije i dizajna peptida, te daje doprinos njihovom rješavanju.

Ključne riječi: peptidi, strojno učenje, predviđanje funkcije peptida, evolucijski algoritmi, generativni modeli, knjižnice peptida

ABSTRACT

Computational characterisation and design of new molecules have improved traditional methods for the discovery of therapeutic compounds that heavily rely on experimental screening. By devising molecules in a computer during the initial stages of research, it is possible to identify promising ligands with significantly lower time and resource expenditures. Therefore, it is not surprising that today almost every research into new molecules includes some form of computational screening, with artificial intelligence gaining an increasing amount of importance. Although the aforementioned field is developing rapidly, there are still specific shortcomings that hinder the wider application of machine learning and other types of algorithms. These are primarily the accuracy of the models and their ability to generalise knowledge, as well as the expensive experimental validation. This dissertation proposes possible solutions through several successive research phases, exploring the design of peptides with targeted properties.

Firstly, a new numerical representation of peptides for machine learning models is proposed, referred to as the sequential properties representation scheme, which combines information about the physicochemical properties of peptides with information about their primary structure. The scheme thus provides a more detailed insight into the sequences, which consequently gives machine learning models a better chance of achieving higher levels of accuracy. In addition to identifying specific properties in a peptide, well-trained machine learning models can also be used as a starting point for the design of new peptides. Thus, two machine learning models have been proposed: for estimating the propensity of peptides to hydrolyse esters and for estimating the propensity for self-assembly, where the latter, in addition to information on sequential properties, also utilised information on the aggregation propensity of the peptidic building blocks: amino acids, dipeptides and tripeptides. Both models have been used in genetic algorithms for the successful creation of

new, previously unknown catalytic and self-assembling sequences. Similar ideas have been applied to the discovery of the beta-secretase 1 enzyme inhibitors, where a multi-criteria genetic algorithm was used, guided by the binding energy of the peptide to the enzyme estimated using molecular docking, Wildman-Crippen logP value and the probability of a peptide crossing the blood-brain barrier estimated using an existing machine learning model from the literature. The final part of the study sought to improve the ease of experimental evaluation of molecules by partitioning peptide libraries into a user-defined number of disjoint subsets for easier synthesis and deconvolution. A multi-criteria genetic algorithm was used, aimed at minimising the overlap between subsets and the overlap of peptides by mass within each subset, while retaining the maximum possible number of peptides from the original library.

The obtained results confirm the functionality of the presented models and algorithms, and provide a significant incentive for their application in practice. Thus, this dissertation proposes possible solutions to some of the prominent problems in the field of computational characterisation and design of peptides, and contributes to their resolution.

Keywords: peptides, machine learning, peptide function prediction, evolutionary algorithms, generative models, peptide libraries

SADRŽAJ

Zahvale	I
Sažetak	III
Abstract	V
1. Uvod	1
1.1. Strojno učenje	4
1.1.1. Modeli predviđanja	4
1.1.2. Generativni modeli	11
1.2. Evolucijski algoritmi	14
1.2.1. Stvaranje početne populacije	15
1.2.2. Mehanizam selekcije	16
1.2.3. Postupak rekombinacije	17
1.2.4. Mutacija genoma	18
1.2.5. Odabir jedinki za sljedeću generaciju	19
1.2.6. Evolucijski algoritmi kao generativni modeli	20
1.3. Hipoteze i znanstveni doprinosi	21
2. Shema sekvencijalnih svojstava	25
2.1. Metodologija	28
2.1.1. Skupovi podataka	28
2.1.2. Sheme predstave peptida u računalu	30
2.1.3. Modeli strojnog učenja i postupak usporedbe shema predstave	34
2.2. Rezultati i diskusija	39

3. Evolucijski pristup istraživanju katalitičkih peptida	43
3.1. Metodologija	44
3.1.1. Skup podataka	44
3.1.2. Model predviđanja	44
3.1.3. Evolucijski generativni model	45
3.2. Rezultati i diskusija	48
4. Generativni model za stvaranje novih peptida sa svojstvom samosastavljanja	53
4.1. Metodologija	55
4.1.1. Skup podataka	55
4.1.2. Predobrada podataka i priprema ulaznih vrijednosti modela predviđanja	57
4.1.3. Modeli predviđanja: učenje i vrednovanje	59
4.1.4. Generativni model i vrednovanje predloženih peptida	62
4.1.5. Ispitivanje konvergencije generativnog modela i analiza strukturnih motiva	66
4.2. Rezultati i diskusija	66
4.2.1. Performanse modela predviđanja	66
4.2.2. Performanse generativnog modela	71
4.2.3. Konvergencija genetskog algoritma i strukturni motivi	79
5. Evolucijski dizajn inhibitora beta-sekretaze 1	81
5.1. Metodologija	83
5.2. Rezultati i diskusija	84
6. Evolucijska optimizacija knjižnica peptida	89
6.1. Metodologija	90
6.1.1. Evolucijski pristup optimizaciji knjižnica peptida	90
6.1.2. Procjena broja nepreklapajućih peptida simuliranim kaljenjem	94
6.1.3. Parametrizacija algoritma	97
6.1.4. Ispitni slučajevi	98
6.2. Rezultati i diskusija	99

7. Zaključak	109
Literatura	113
Popis slika	145
Popis tablica	153
Popis algoritama	159
Popis kratica	161
Privitci	163
A. Peptidi s eksperimentalno ispitanim svojstvom samosastavljanja	165
B. Ispitivanje greške procjene preklapanja peptida po masi simuliranim kalje- njem	178
Životopis doktoranda	181
Popis znanstvenih radova	183

1. UVOD

Peptidi su kratki lanci aminokiselina povezani peptidnim vezama, uobičajene duljine između 2 i 50 gradivnih blokova (eng. residue) [142, 168, 171]. Njihove primjene uključuju dizajn molekula sa svojstvom spontanog samosastavljanja (eng. self-assembly) u strukture poput alfa-zavojnice (eng. alpha helix) i beta-ploče (eng. beta sheet) za stvaranje novih nanomaterijala [138], otkrivanje novih lijekova [180], antibiotika [28] i općenito spojeva koji mogu agonistički ili antagonistički djelovati na proizvoljnu metu, razvijanje novih katalitički aktivnih peptida za medicinske i industrijske primjene [40], i slično. Uvriježena literatura peptide veličinom svrstava između malih molekula (<900 Da [96]) i proteina, te upravo u odnosu na navedene grupe spojeva valja promotriti njihove prednosti i nedostatke. U usporedbi s malim molekulama, koje su obično rigidne strukturom, peptidi posjeduju veću fleksibilnost u sekundarnoj i tercijarnoj strukturi, te se stoga mogu bolje prilagoditi željenoj meti, minimizirajući pritom učinke van mete (eng. off-target effect) [135, 154]. Navedeno rezultira većom specifičnošću molekule prema meti i omogućuje stvaranje složenijih načina interakcija i djelovanja. S druge strane, sklonost peptida ka proteolitičkoj degradaciji čini njihovu primjenu u dizajnu lijekova izazovnom, pri čemu je u obzir potrebno uzeti čimbenike poput bioraspoloživosti molekule, vremena polueliminacije i ekskrecije [46, 177]. Peptidi također često imaju problema s prelaskom lipidnih membrana, što se može pripisati njihovoj veličini i manjku interakcije s hidrofobnim dvoslojem (eng. bilayer) [46], pri čemu je prolazak krvno-moždane barijere za doseg meta u mozgu ljudi posebno izazovan i obično uključuje modifikaciju primarne strukture peptida ili uporabu transportera (eng. shuttle) [194]. Nadalje, smanjena strukturna složenost peptida u odnosu na proteine obično rezultira manjom imunogenošću i nižim troškovima proizvodnje, ali s druge strane taj isti manjak složenosti peptide može ograničiti u postizanju stabilne trodimenzionalne konformacije i funkcijama koje mogu ispoljiti, čineći dizajn

umjetnih enzima i sličnih struktura posebno zahtjevnim [279]. Sukladno tome, peptidi predstavljaju svojevrsan kompromis između malih molekula i proteina, uvodeći veću fleksibilnost u dizajn specifičnih svojstava molekule, te istovremeno olakšavajući rukovanje.

Tradicionalni pristup dizajnu novih peptida temelji se na eksperimentalnoj validaciji svake pojedine sekvence [230]. U takvom postupku, istraživači se oslanjaju na znanje i intuiciju pri predlaganju sekvenci za koje pretpostavljaju da bi mogle posjedovati željena svojstva. Primjerice, poznato je da pozitivan naboj i hidrofobnost pridonose antimikrobnoj aktivnosti [1, 117], dok ravnoteža između hidrofobnih i hidrofilnih svojstava utječe na samosastavljanje [170]. Na temelju takvih spoznaja moguće je dizajnirati kandidate s povećanom vjerojatnošću postizanja ciljanih karakteristika. Osim toga, već opisane sekvence s potvrđenim svojstvima često se koriste kao polazišne točke za daljnji razvoj novih peptida [8, 36, 55, 110]. Iako takva strategija ubrzava otkrivanje novih kandidata, ona istovremeno ograničava istraživanje na uski dio kemijskog prostora i uvodi pristranost prema određenim tipovima sekvenci, zanemarujući potencijalno povoljnija rješenja izvan tog okvira. Dodatno, sinteza, purifikacija i ispitivanje aktivnosti peptida u laboratoriju koristeći instrumentalne metode kao što su masena spektrometrija i mikroskopija iziskuju mnogo vremena, resursa i specijaliziranu opremu [38, 107, 108]. Primjerice, pretklinička provjera inhibicije leptinskog receptora za 10-ak peptida može trajati nekoliko mjeseci, te iziskivati oko 80000 američkih dolara [198, 199]. S obzirom na to da se u tipičnom ciklusu stvaranja lijeka testira na tisuće molekula [166], ukupni troškovi istraživanja obično prijeđu milijardu američkih dolara, dok sam proces razvoja traje oko 10 godina [53]. Posljedično, otkrivanje novih molekula isključivo putem eksperimentalne validacije izrazito je skupo i vremenski zahtjevno kada se eksperimentalno validira svaka predložena sekvencija. Matematički gledano, dizajn peptida predstavlja NP-težak (eng. nondeterministic polynomial-time hard; NP-hard) kombinatorni problem, pri čemu broj mogućih sekvenci raste 20 puta sa svakom proteinogenom aminokiselinom u lancu [209]. Tako, primjerice, za peptid duljine 5 aminokiselina postoji oko 3 milijuna sekvenci, dok za duljinu 7 taj broj raste preko milijarde. Posljedično, eksperimentalna provjera svih spojeva u kemijskom prostoru kada peptid sadrži više od 3 aminokiseline u lancu [11] vrlo je zahtjevna i neizvediva u realnim okvirima budžeta i raspoloživog vremena, a uslijed očekivane oskudnosti korisnih molekula u takvim prostorima, nije niti isplativa [283].

S razvojem računalnog hardvera i naprednih algoritama, metode molekularne dinamike

(eng. molecular dynamics; MD) uvele su mogućnost simuliranja laboratorijskih eksperimenata *in silico*. MD se temelji na izračunu položaja atoma, skupina atoma ili molekula, ovisno o željenoj razini detalja i točnosti, koristeći Newtonove jednačbe gibanja [75]. Jednačbe se primjenjuju na svaki element sustava u vrlo malim vremenskim razmacima, obično femtosekunda ili nanosekunda, simulirajući pritom ponašanje sustava kroz vrijeme. Navedeno može dati informaciju o tome kako će sustav interagirati s određenom metom, hoće li se samosastaviti u pravilnu strukturu, i slično. Iako MD nadomješta potrebu za eksperimentima u početnoj fazi ispitivanja molekula, štedeći pritom resurse, simulacije još uvijek mogu trajati nekoliko dana ili tjedana [139], često zahtijevaju superračunalnu infrastrukturu, a rezultati uvelike ovise o postavkama simulacije (željeno polje sila, razina detalja, vrijeme simulacije [77, 249]) i greškama koje se akumuliraju kroz vrijeme simulacije. MD također uvodi dodatno ograničenje na veličinu sustava. Kako se Newtonove jednačbe moraju opetovano izračunavati za svaku gradivnu jedinicu, simuliranje velikih sustava kao što su lipidne membrane i krvno-moždana barijera i dalje je teško izvedivo čak i s velikom računalnom snagom. Sukladno tome, MD možemo promatrati kao grubu aproksimaciju laboratorijskih eksperimenata koja ne mora uvijek biti točna, ne omogućuje iscrpno ispitivanje velikih kemijskih prostora, no daje vrlo dobar uvid u ponašanje molekula i omogućuje njihovo relativno jeftino ispitivanje.

Uvođenjem dodatnih aproksimacija u računalne simulacije moguće je značajno skratiti vrijeme do rezultata, premda uz određen gubitak točnosti. Umjesto eksplicitnog rješavanja Newtonovih jednačbi gibanja, pristup molekularnog pristajanja (eng. molecular docking; MP) koristi optimizaciju, djelomično ograničava konformacijsku fleksibilnost liganda i receptora, te se oslanja na unaprijed poznate strukture molekula [70, 178]. Cilj MP-a jest predvidjeti orijentaciju liganda u odnosu na receptor pri formiranju stabilnog kompleksa, čime se dobiva uvid u jačinu međudjelovanja i moguće poze, što može biti ključno za razlikovanje tipa interakcije (agonizam, antagonizam). Iako je MP općenito manje točan u odnosu na MD, njegova je glavna prednost višestruko brže izvođenje, a uz primjenu paralelizacije, rezultati se mogu dobiti već unutar nekoliko minuta [255]. Međutim, postoji i rizik da MP ne uspije pronaći valjano rješenje. Različite varijante MP-a postavljaju specifična ograničenja na optimizaciju. Primjerice, postoje pristupi u kojima je ligand fleksibilan, a receptor fiksiran, pristupi s rigidnim molekulama, i slično. Uz odabir varijante, ključan je i izbor funkcije bodovanja (eng. scoring function) koja usmjerava

optimizaciju, pri čemu je to obično procjena slobodne energije kompleksa koja se nastoji minimizirati [152]. Unatoč brzini i mogućnosti šireg pretraživanja kemijskog prostora, niti MP ne omogućuje iscrpno istraživanje konformacijskog prostora.

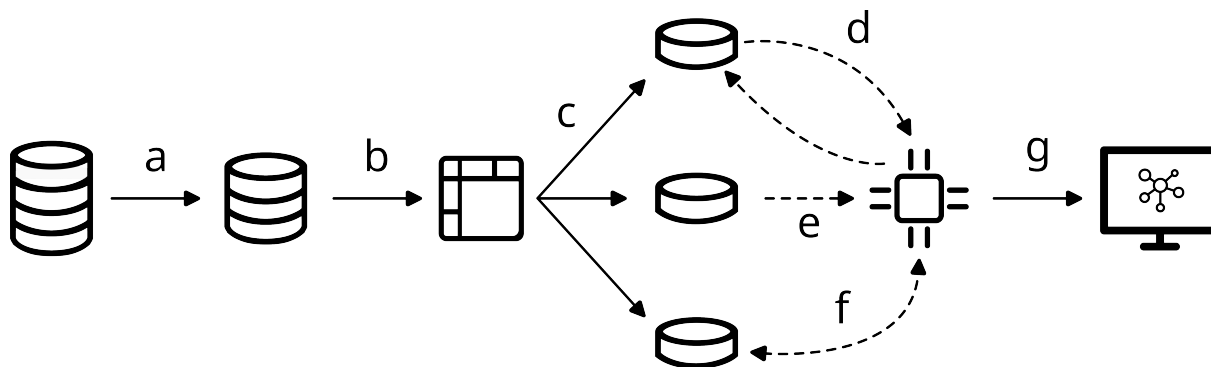
Usporedba laboratorijskih eksperimenata, MD-a i MP-a pokazuje da su brzina i točnost vrednovanja međusobno oprečni kriteriji – povećanje jednoga nužno dolazi na račun drugoga. Stoga eksperimenti ostaju neizostavni kao konačna potvrda svojstava molekule, dok se MP i MD primarno koriste u ranim fazama istraživanja za brzi probir (eng. screening) kandidata.

1.1. Strojno učenje

Uspjesi strojnog učenja (eng. machine learning; ML) u različitim domenama, od klasifikacije slika [106], igranja šaha [236], detekcije financijskih prijevara [12], preko računalnog vida i autonomne vožnje [207], pa sve do suvremenih i sveprisutnih velikih jezičnih modela (eng. large language models; LLMs) [216, 261], potaknuli su ideju o njihovoj primjeni u kemiji i farmaciji. Iako su prvi algoritmi strojnog učenja razvijeni tijekom druge polovice 20. stoljeća [74], njihova primjena u područjima kemije i farmacije dugo je bila ograničena zbog nedostatka velikih skupova podataka potrebnih za učenje modela [62]. Pojavom baza podataka kao što su Protein Data Bank [15], Enamine REAL Database [233] i ChEMBL [289], postalo je izvedivo razvijati i učiti modele namijenjene za predviđanje aktivnosti i strukture malih molekula, peptida i proteina.

1.1.1. Modeli predviđanja

Modeli predviđanja u strojnom učenju koriste se za procjenu vrijednosti ili određivanje pripadnosti klasi na temelju zadanih ulaznih podataka. Primjerice, pri izradi modela za predviđanje antimikrobne aktivnosti peptida, ulaz u model mogu činiti numerički opisane značajke peptida (npr. fizikalno-kemijska svojstva) [268]. Izlazna vrijednost iz modela može biti kontinuirana, poput minimalne inhibitorne koncentracije u odnosu na određenu bakteriju, što predstavlja problem regresije, ili pak decimalna između 0 i 1, koja označava vjerojatnost da će peptid imati antimikrobno djelovanje, što odgovara problemu klasifikacije.



Slika 1.1: Shematski prikaz razvoja modela strojnog učenja, uključujući (a) filtriranje podataka, (b) odabir i transformaciju značajki, (c) raspodjelu skupa podataka na podskup za učenje, validaciju i testiranje, (d) proces učenja, (e) optimizaciju hiperparametara na podskupu za validaciju, (f) konačno testiranje modela i (g) eksploataciju.

Prvi korak u izradi modela jest prikupljanje dovoljne količine podataka za učenje i vrednovanje. Svaki primjer u skupu podataka sastoji se od para ulaznih i izlaznih vrijednosti, a cilj modela jest naučiti predvidjeti izlaz na temelju pripadajućeg ulaza. Potreban broj primjera ovisi o domeni i složenosti problema, pri čemu višedimenzionalni podaci tipično zahtijevaju veće skupove za postizanje pouzdane točnosti [263]. Tako je za klasifikaciju slika u pravilu potrebno na tisuće primjera [292], dok se u specifičnim slučajevima, poput klasifikacije pojedinih skupina peptida, zadovoljavajući rezultati ponekad mogu dobiti već s nekoliko stotina molekula [57, 172]. Općenito je poželjno imati što veću količinu podataka jer će time potencijalna točnost modela biti viša. Baze podataka često sadrže nepotpune, nedosljedne ili udvostručene zapise koje je potrebno ukloniti (slika 1.1.a). Također, nisu svi prikupljeni primjeri nužno relevantni za problem koji se razmatra, pa ih je potrebno isključiti iz analize; primjerice, peptide koji sadrže neproteinogene aminokiseline. Nakon filtriranja podataka, pristupa se odabiru i transformaciji značajki koje će poslužiti kao ulaz u model (slika 1.1.b). Budući da modeli strojnog učenja rade isključivo s numeričkim podacima, sve karakteristike koje opisuju ulazne podatke moraju biti na odgovarajući način kvantificirane. U radu s peptidima često se koriste njihove fizikalno-kemijske značajke, pri čemu se svaki peptid opisuje skupom numeričkih vrijednosti koje sadrže njegov naboj, vrijednost $\log P$, izoelektričnu točku, i slično. Osim toga, moguće je uključiti i informaciju o prostornoj strukturi peptida, koja se može numerički predstaviti u obliku grafa koristeći matricu susjedstva čvorova. Općenito, u model se mogu unijeti raznovrsne značajke ulaznih podataka pod uvjetom da su prikladno numerički predstav-

ljene. Važno je pritom da značajke budu relevantne za problem koji se istražuje i da daju dovoljnu količinu informacija za zadatak predviđanja. Na primjer, modelu će biti teško predvidjeti sposobnost peptida za penetraciju stanične membrane na temelju učestalosti alanina u primarnoj strukturi. Međutim, ako mu se umjesto toga osigura informacija o hidrofobnosti i naboju svakog peptida, učenje postaje znatno točnije [284]. Nakon što se podaci pripreme i transformiraju u numeričke ulazno-izlazne parove, skup podataka uobičajeno se dijeli na tri podskupa: za učenje, validaciju i testiranje, pri čemu se za učenje odvađa najviše podataka, a za testiranje najmanje (slika 1.1.c). Podskup za učenje koristi se za učenje modela strojnog učenja, odnosno za uspostavljanje pravilnog preslikavanja ulaznih atributa u izlazne vrijednosti, bilo regresijske vrijednosti ili klasifikacijske vjerojatnosti (slika 1.1.d). Matematički gledano, model strojnog učenja može se promatrati kao složena funkcija f , određena skupom parametara θ , koja ulazne podatke λ preslikava u izlazne vrijednosti γ . Ako ciljne izlazne vrijednosti označimo s X , tada je cilj procesa učenja minimizirati funkciju gubitka δ izračunatu na temelju λ i X (izraz 1.1).

$$\min_{\theta}(\delta(f(\lambda, \theta), X)) \quad (1.1)$$

Funkcija gubitka kvantificira razliku između izlaza modela i očekivanog izlaza, te je temelj određivanju načina modifikacije parametara θ . Niže vrijednosti funkcije gubitka znače i bolje performanse modela, stoga modifikacija θ uvijek ide u smjeru minimizacije vrijednosti funkcije gubitka. Uobičajene funkcije gubitka za probleme klasifikacije su binarna unakrsna entropija (eng. binary cross-entropy), za binarne probleme, i kategorička unakrsna entropija (eng. categorical cross-entropy), za višeklasne probleme, dok se za probleme regresije obično koristi srednja apsolutna pogreška (eng. mean absolute error) ili srednja kvadratna pogreška (eng. mean squared error) [19].

Funkcija gubitka minimizira se iterativnom modifikacijom parametara θ na podskupu za učenje, dok se podskup za validaciju koristi za optimizaciju hiperparametara modela (npr. arhitekture modela; slika 1.1.e). Model se izravno prilagođava podacima za učenje, te uz dovoljno vremena i kapaciteta može postići apsolutno točno predviđanje na njima. Međutim, takva procjena sposobnosti predviđanja modela je pristrana i ne odražava očekivane performanse modela pri primjeni na nove, prethodno neviđene podatke, što nazivamo prekomjernom naučenošću (eng. overfitting). Zbog toga, procjena perfor-

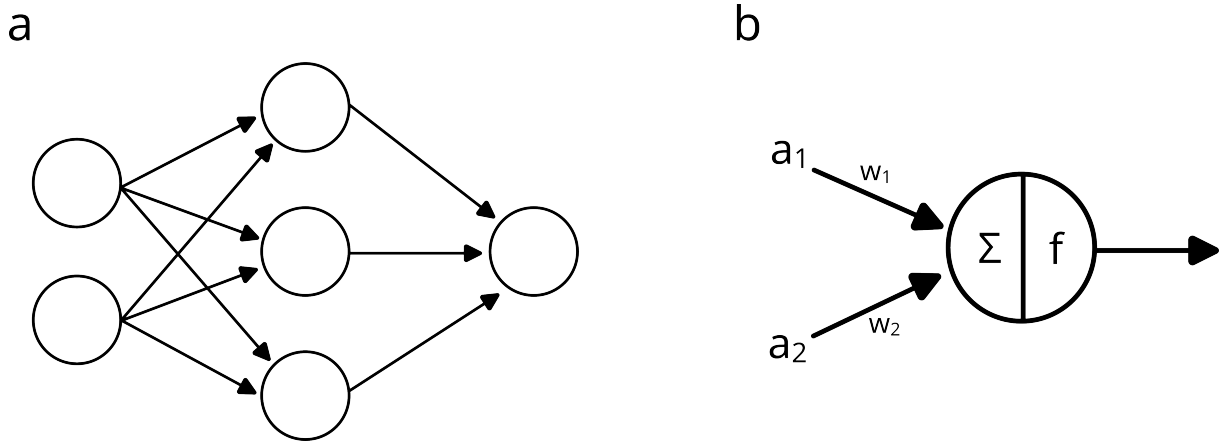
mansi i donošenje odluka o izmjeni hiperparametara odvija se na podskupu za validaciju, kojemu se model posredno prilagođava, preferirajući parametre modela koje na njemu postižu bolje rezultate. Krajnja procjena performansi nakon završetka procesa učenja odvija se na podskupu za testiranje, odnosno nad podacima s kojima model ni na koji način nije bio u doticaju prilikom samog procesa učenja i ugađanja (slika 1.1.f). Rezultati na podskupu za testiranje sačinjavaju završnu mjeru točnosti, jer se u toj fazi model više ne prilagođava, te odražavaju očekivanu sposobnost modela u primjeni na potpuno nove primjere. Po postizanju zadovoljavajuće razine točnosti, ovisno o prirodi problema, model se smatra spremnim za eksploataciju (slika 1.1.g).

Umjetne neuronske mreže

Umjetne neuronske mreže (eng. artificial neural networks) stvorene su na ideji oponašanja načina rada ljudskog mozga [6]. Iako to nisu uspjele u potpunosti ostvariti, ideja propagacija signala neuronima dala je matematičke temelje ovom modelu strojnog učenja. Arhitektura neuronskih mreža sastoji se od niza gradivnih jedinica – neurona, povezanih u slojeve (slika 1.2.a). Svi neuroni jednog sloja povezani su sa svim neuronima prethodnog i sljedećeg sloja. Propagacija signala odvija se od ulaznog sloja, koji prihvaća početne podatke, te se kroz niz ulančanih matematičkih operacija dolazi do izlaznog sloja, odnosno predviđanja modela. Iako neuronska mreža obavlja složenu zadaću, svaki njezin dio vrlo je jednostavan. Neuroni na svojem ulazu primaju n decimalnih vrijednosti x_i . Svaka vrijednost množi se pripadajućom težinom w_i , te se umnošci potom zbrajaju, nadodaje se pomak (eng. bias) w_0b , te se primjenjuje aktivacijska funkcija f (izraz 1.2, slika 1.2.b). Dobivena vrijednost predstavlja konačni izlaz neurona i može činiti vrijednost u izlaznom sloju ili ulaz u sljedeći neuron. Opisanim ulančavanjem dobije se model gdje svaka komponenta radi relativno jednostavan posao, dok mreža u cijelosti može rješavati složene zadatke.

$$f\left(\sum_{i=1}^n w_i x_i + w_0 b\right) \quad (1.2)$$

Aktivacijska funkcija neuronskim mrežama daje mogućnost modeliranja nelinearnih funkcija. Ako se ona ne bi primjenjivala na izlaz neurona, duboke neuronske mreže bile bi jednostavni linearni prediktori i nikako se ne bi mogli nositi sa zahtjevnim zadacima za



Slika 1.2: Prikaz (a) arhitekture neuronske mreže koja se sastoji od ulaznog sloja s dva neurona, središnjeg sloja s tri neurona, i izlaznog sloja s jednim neuronom. Neuron svakog sloja povezan je sa svim neuronima prethodnog i sljedećeg sloja. Vidljiv je i (b) detaljan presjek načina rada svakog neurona, pri čemu se svaka ulazna vrijednost a_i množi s pripadajućom težinom w_i , nakon čega slijedi sumacija i primjena aktivacijske funkcije.

kakve ih danas primjenjujemo. Iako aktivacijske funkcije mogu varirati po svakom pojedinom neuronu, obično se unutar cijele neuronske mreže koristi istovjetna, dok se aktivacije izlaznog sloja modificiraju kako bi izlazna vrijednost bila u skladu s očekivanim rasponom. Tako se u skrivenim, unutrašnjim slojevima neuronskih mreža često primjenjuje aktivacijska funkcija ReLU, dok se u izlaznom sloju primjenjuje sigmoidna za binarnu klasifikaciju, softmax za višeklasnu klasifikaciju ili linearna za regresiju [231].

Nakon propagacije podataka kroz mrežu, uz upotrebu funkcije gubitka i pravila ulančanog deriviranja, računaju se parcijalne derivacije i pripadajući gradijenti za svaku pojedinu težinu w_i u mreži [83]. Gradijenti se tada upotrebljavaju kako bi se težine mreže w_i prilagodile na način koji minimizira vrijednost funkcije gubitka L u tom trenutku (jednadžba 1.3), pri čemu je α obično pozitivan broj manji od 1. Time u svakoj iteraciji učenja dobivamo mrežu koja sve bolje obavlja svoju zadaću, s obzirom na to da je vrijednost funkcije gubitka korelirana s uspješnošću same mreže.

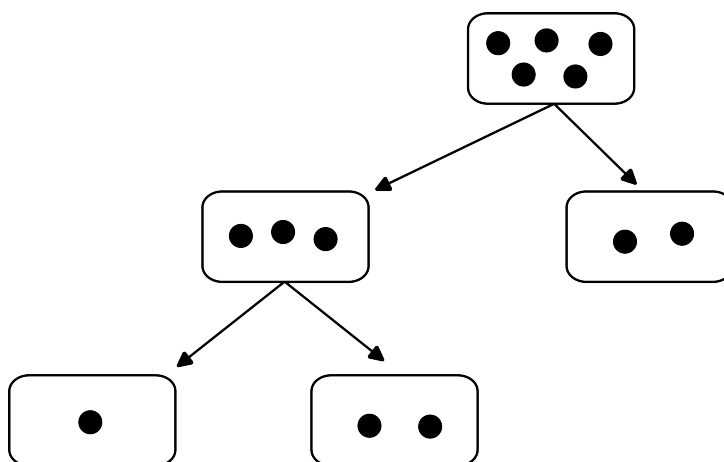
$$w_i^* = w_i - \alpha * \frac{\partial L}{\partial w_i}, \quad \forall i \in \{1, 2, \dots, n\} \quad (1.3)$$

Model nasumične šume

Model nasumične šume (eng. random forest) sastoji se od jednostavnijih modela strojnog učenja koja se nazivaju stablima odluke [27]. Svako stablo odluke iskazuje predviđanje

za pojedini ulazni primjer, a model nasumične šume tada donosi odluku koristeći sva dostupna predviđanja, najčešće odabirući opciju s najviše glasova kao konačnu.

Stablo odluke ima strukturu stabla i njegova gradnja kreće od korijenskog čvora, koji na početku učenja sadrži sve podatke. Potom se računa funkcija podjele čvora, pri čemu se obično koristi Gini koeficijent ili dobitak informacije (eng. information gain) [250], koja mjeri raznolikost podataka u čvoru. U cilju postizanja relativne homogenosti svakog čvora, korijenski čvor se dijeli na dva ili više čvorova, pri čemu svaki čvor preuzima podskup podataka, smanjujući time razlike u podacima po čvoru. Rekurzivna podjela novih čvorova nastavlja se sve dok se ne dosegne određena dubina stabla, vrijednost funkcije podjele se smanji ispod granične razine, ili slično (slika 1.3). Listovi stabla tada predstavljaju krajnje čvorove i opisuju kategorije u koje se podaci mogu klasificirati. Za novi primjer koji pristigne, tada se donosi niz odluka koje su ekvivalent kretanja po čvorovima, sve do dostizanja lista stabla.

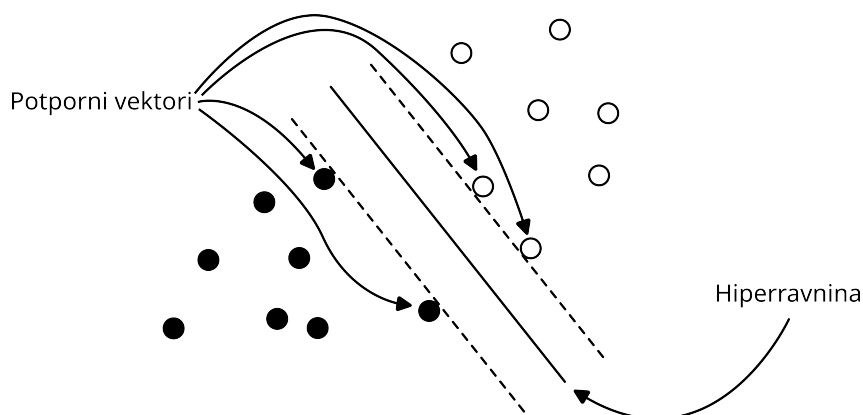


Slika 1.3: Prikaz procesa stvaranja stabla odluke. Čvorovi se rekurzivno dijele sve dok se ne dostigne uvjet za prestanak. Odluka o pripadnosti listu donosi se prolaskom od korijena do lista, iterativno prateći niz naučenih pravila.

Stroj potpornih vektora

Stroj potpornih vektora (eng. support vector machine, SVM) za cilj ima pronaći hiper-ravninu koja najtočnije odvaja dvije klase [45]. Krajnje točke pojedinih klasa, prostorno najbliže suprotnoj klasi, sačinjavaju potporne vektore, dok je hiperravnina maksimalno udaljena od njih (slika 1.4). Matematički, SVM nastoji n primjera x_i klasificirati u ispravnu klasu y_i minimizacijom izraza 1.4, pri čemu vektor normale w i skalar b definiraju

jednadžbu razdjelne hiperravnine, a C je regularizacijska konstanta koja određuje ravnotežu između udaljenosti hiperravnine od potpornih vektora i osiguravanja da se svaka točka x_i nalazi s odgovarajuće strane ravnine. Optimizacija modela tada se svodi na određivanje ravnine koja maksimalno udovoljava predstavljenim kriterijima. Prednost ovakvog modela jest što na definiciju hiperravnine u konačnici utječe samo nekolicina točaka, što smanjuje složenost optimizacijskog problema. U slučajevima kada je linearna odvojivost klasa teška za postići, SVM može najprije transformirati podatke u višedimenzionalni prostor koristeći nelinearnu jezgrenu funkciju (eng. kernel function) [206], te po postizanju linearne odvojivosti pristupiti pronalaženju hiperravnine.



Slika 1.4: Shematski prikaz pronalaženja hiperravnine pri učenju modela stroja potpornih vektora. Hiperravnina se nalazi između potpornih vektora dviju klasa.

$$\frac{1}{2}\|w\|^2 + C \left[\frac{1}{n} \sum_{i=1}^n \max(0, 1 - y_i(w^T x_i + b)) \right] \quad (1.4)$$

Primjena modela predviđanja u kemiji peptida

Daljnijim razvojem strojnog učenja i uspjesima modela poput AlphaFold-a [120], za što je dodijeljena i Nobelova nagrada [34], vidljiv je porast primjene modela predviđanja u području kemije peptida. Od 2019., zamjetno je povećanje frekvencije broja objavljenih znanstvenih radova koji u svojoj metodologiji koriste strojno učenje za računalno predviđanje terapijske aktivnosti [10]. Velikim dijelom to možemo pripisati brzini predviđanja modela ML, koja često iznosi manje od nekoliko desetaka milisekundi, i njihovoj točnosti kada se uče na velikim skupovima podataka, rješavajući tako glavne probleme MP-a i MD-a. Posljedično, znanstvenici su u modelima strojnog učenja prepoznali alate koji

omogućuju brz i točan probir peptida, a osim njihovog korištenja u predviđanju sastavljanja [11, 191], antimikrobne [109, 268, 280] i antiviralne aktivnosti [3, 148, 201], sposobnosti ulaska u stanicu [57, 169], prelaska krvno-moždane barijere u ljudi [39, 183], modeli se koriste i za predviđanje interakcije s proteinima [49, 153], dok je značajno istaknuti i njihovu primjenu u predviđanju prostorne strukture peptida [219]. Iako korisni, modeli strojnog učenja uvelike ovise o svojoj arhitekturi, ulaznim značajkama na temelju kojih vrše predviđanje i količini podataka dostupnim za njihovo učenje, stoga izrada kvalitetnih modela strojnog učenja, koji će u konačnici polučiti dobre rezultate, zahtijeva interdisciplinarno znanje kemije i računarstva. Kao i za MP i MD, konačna potvrda aktivnosti peptida koristeći eksperimentalnu validaciju nužna je i za ML, što niti jedna računalna metoda zasad ne može izbjeći.

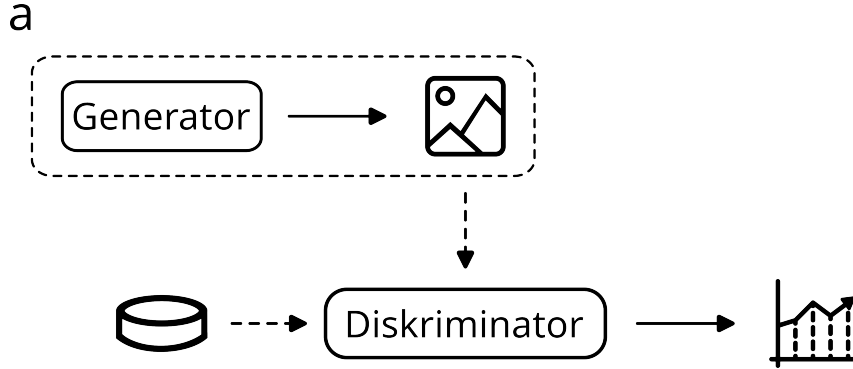
1.1.2. Generativni modeli

Nasuprot modelima predviđanja, generativni modeli za cilj imaju stvaranje novih primjera koji karakteristikama nalikuju na one koji su korišteni za učenje modela. Uobičajen primjer navedenog jest stvaranje slika određenog tipa [257]. Matematički gledano, svaki skup srodnih podataka slijedi određenu razdiobu i ima intrinzične pravilnosti, pa iako su principi funkcioniranja modela predviđanja i generativnih modela relativno slični, glavna razlika ogleda se u tome što modeliraju, pri čemu generativni modeli nastoje obuhvatiti osobitosti razdiobe podataka koristeći podatke za učenje. Uzorkovanjem iz naučene razdiobe i potom pravilnim dekodiranjem, ovi modeli mogu stvoriti nove podatke. Tipovi generativnih modela strojnog učenja razlikuju se u načinima na koje nastoje opisati razdiobu.

Generativne suparničke mreže

Generativne suparničke mreže (eng. generative adversarial network; GAN) učenje ostvaruju suprotstavljanjem dviju neuronskih mreža [84]. Prvi dio modela naziva se generator i koristi neuronsku mrežu kako bi n -dimenzionalni vektor nasumičnog šuma (označen sa z , obično uzorkovan iz Gaussove razdiobe) transformirao u korisne, m -dimenzionalne primjere ($G(z)$). Drugi dio modela naziva se diskriminator i na raspolaganju za učenje ima skup stvarnih podataka određene razdiobe (označen s x), te je njegov cilj točno pro-

cijeniti pripada li novi primjer navedenom skupu podataka ($D(x)$) ili je stvoren od strane generatora ($D(G(z))$). Sukladno tome, generator nastoji stvoriti podatke razdiobom što sličnije podacima iz skupa podataka za učenje kako bi u konačnici točnost diskriminatora bila jednaka nasumičnom pogađanju, odnosno 50%, nakon čega se generator koristi za stvaranje novih podataka (slika 1.5). Funkcija gubitka definirana je izrazom 1.5, pri čemu ju diskriminator nastoji maksimizirati, a generator minimizirati.

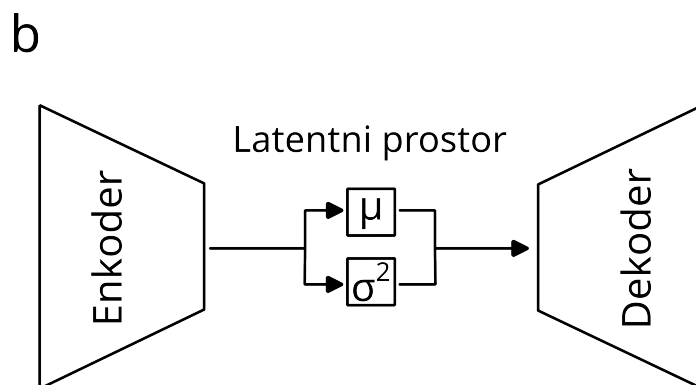


Slika 1.5: Shematski prikaz procesa razvoja modela generativnih suparničkih mreža. Posebno je označen dio generiranja novih podataka koji se odvađa od ostatka modela nakon uspješnog učenja i koristi u fazi eksploatacije.

$$\mathbb{E}_x[\log(D(x))] + \mathbb{E}_z[\log(1 - D(G(z)))] \quad (1.5)$$

Varijacijski autoenkoderi

Varijacijski autoenkoderi (eng. variational autoencoder, VAE) sastoje se od neuronske mreže s tzv. *uskim grlom*, pri čemu su dimenzije ulaznih i izlaznih podataka jednake (slika 1.6). Pri učenju, VAE najprije preslikava ulazne podatke u latentni prostor smanjenih dimenzija u sredini mreže, nastojeći pritom komprimirati podatke na najbitnije značajke i zadržati normalnu razdiobu latentnog prostora. Potom, procesom nasumičnog uzorkovanja i dekodiranja, VAE stvara nove podatke. Usporedbom ulaznih i izlaznih vrijednosti modela, pri čemu se koristi binarna unakrsna entropija, srednja kvadratna pogreška ili slično, te usporedbom razdiobe latentnog prostora modela s normalnom razdiobom (izraz 1.6, Kullback-Liebler divergencija [134]), dobije se informacija o uspješnosti modela i gradijentima pomoću kojih se modificiraju težine mreže [129]. Konceptualno, VAE se sastoji od tri dijela: enkodera, koji komprimira podatke, latentnog prostora, koji uči i uzorkuje razdiobu ulaznih podataka, i dekodera, koji nastoji rekonstruirati podatke.



Slika 1.6: Prikaz varijacijskog autoenkodera i njegova tri osnova dijela: enkodera, latentnog prostora i dekodera.

$$D_{KL}(X \sim N(\mu, \sigma^2) \parallel N(0, I)) \quad (1.6)$$

Primjena generativnih modela u kemiji peptida

Sposobnost prethodno opisanih generativnih modela strojnog učenja za pretragu velikih kemijskih prostora i predlaganje novih peptida sa specifičnim aktivnostima dovela je do njihovog povećanog korištenja u pretkliničkim fazama ispitivanja novih lijekova [267]. Varijacijski autoenkoderi već su korišteni za stvaranje antimikrobnih peptida koji su u eksperimentalnim vrednovanjima pokazali pozitivno djelovanje [53, 58, 59]. Dodatno, potreba za stvaranjem peptida sa karakterističnim svojstvima dovela je do razvoja modela VAE koji omogućuju uvjetovanje, odnosno pomoću dodatne parametrizacije moguće je povećano utjecati na svojstva peptida koji se generira, regulirajući tako primjerice sastav peptida, toksičnost, i slično [54, 87, 293]. Još od pojave prototipnog modela generativnih suparničkih mreža za dizajn peptida [222], GAN-ovi su korišteni za stvaranje antiviralnih [98, 246], antimikrobnih [136, 157] i peptida koji se vežu za specifične receptore [272]

Iako su se generativni modeli u dizajnu peptida počeli koristiti relativno nedavno, rezultati koje *in silico* stvorene molekule pokazuju u eksperimentalnim ispitivanjima daju naslutiti da bi se generativni modeli mogli iskoristiti za povećanje postotaka uspješnog otkrivanja molekula, koji trenutno iznose oko 1% [53]. Svaki model predviđanja i generativni model strojnog učenja ima svoju specifičnu primjenu i ne možemo tvrditi da je jedan općenito bolji od drugoga, sukladno teoremu „No Free Lunch” [277]. Iako modeli temeljeni na umjetnim neuronskim mrežama ulaze u sve više domena zadnjih godina, modeli

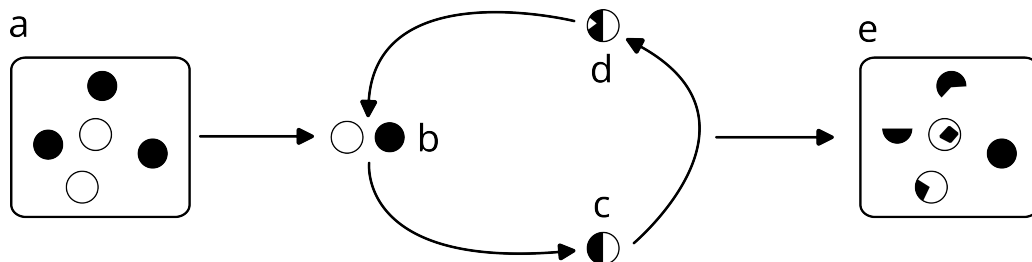
poput stabla odluke i dalje imaju prednost transparentnosti i inherentne objašnjivosti donošenja odluka. Sukladno tome, izbor modela za određeni problem možemo tretirati kao hiperparametar i svakako je potrebno ispitati koji tip modela će dati najbolje rezultate za određenu primjenu.

1.2. Evolucijski algoritmi

Priroda je milijunima godina usavršavala optimizacijski mehanizam kako bi živa bića prilagodila okruženju u kojem se nalaze. Danas taj proces nazivamo evolucija, a njegova snaga vidljiva je u samom ljudskom postojanju [52]. Od pojave prvih jednostaničnih organizama u Zemljinim oceanima, preko primitivnih višestaničnih nakupina, pa sve do ljudi, priroda konstantno razvija vrste koje su sve prilagođenije svojem neposrednom okolišu [37, 63]. Navedene činjenice svjedoče snazi evolucijskog procesa i mogućnosti njegove primjene u optimizaciji. Znanstvenici su snagu evolucije prepoznali još 1950-ih godina, kada su se javile prve ideje o oponašanju procesa evolucije u računalu [239, 258]. Cilj svih algoritama u ovom skupu jest pronaći parametre θ koji daju maksimalnu vrijednost funkcije f (izraz 1.7). U ovom slučaju, parametri θ su vrijednosti koje određuju rješenje i mogu predstavljati numerički kodiranu molekulu, sliku, vektor realnih brojeva, i slično, ovisno o problemu kojeg rješavamo. Funkcija f naziva se funkcijom dobrote i kvantificira kvalitetu rješenja. Evolucijski algoritmi ne zahtijevaju derivabilnost funkcije f , već mogu raditi na širem skupu problema i f promatrati kao crnu kutiju (eng. black box) [30], donoseći odluku o smjeru pretraživanja nove populacije isključivo na temelju izlaznih vrijednosti funkcije. Kako bi se navedeno ostvarilo i ovdje postoji sijaset algoritama koji će za određene tipove problema biti bolji ili gori, slično kao i kod strojnog učenja. Tako će, primjerice, algoritam neuroevolucije uspješno napraviti evoluciju arhitekture i težina neuronske mreže [244], dok će diferencijalna evolucija bolje odgovarati optimizaciji višedimenzionalnih kontinuiranih funkcija [245]. U ovoj disertaciji, fokus je stavljen na genetske algoritme, primarno zbog njihove svestranosti, mogućnosti primjene na razne tipove problema, te njihovog dosadašnjeg uspjeha u stvaranju i optimizaciji peptida [36, 55, 127, 210].

$$\max_{\theta}(f(\theta)) \tag{1.7}$$

Genetski algoritmi modificiraju rješenja ne bi li pronašli ono optimalno (slika 1.7). Počevši od skupa n nasumičnih rješenja θ_i , $\forall i \in 1, 2, \dots, n$ (slika 1.7.a), rješenja se kombiniraju i modificiraju kako bi u svakoj sljedećoj iteraciji populacija ostvarila napredak mjeren funkcijom f . Proces stvaranja nove populacije obuhvaća selekciju (slika 1.7.b), rekombinaciju (slika 1.7.c) i mutaciju (slika 1.7.d). Nakon određenog broja iteracija ili dostizanja unaprijed definirane kvalitete rješenja, proces se prekida i iz konačne populacije odabire se rješenje (slika 1.7.e). U slučaju jednokriterijske optimizacije, izbor je jasan, odabire se rješenje s najvišom vrijednošću funkcije dobrote, dok kod višekriterijske optimizacije izbor može ovisiti o željama korisnika. Način implementacije, tj. predstave rješenja u računalu i mehanizmi funkcioniranja selekcije, rekombinacije i mutacije uvelike ovise o promatranom problemu. Tako će, primjerice, u poznatom problemu putujućeg trgovca (eng. travelling salesman problem) rješenja biti predstavljena permutacijama gradova koje valja posjetiti [141], dok će kod optimizacije parametara matematičke funkcije rješenja biti predstavljena u obliku višedimenzionalnih vektora realnih brojeva [240]. Vrijednosti koje opisuju rješenje nazivaju se genomom.



Slika 1.7: Shematski prikaz načina rada genetskog algoritma. Proces najprije (a) generira nasumičnu populaciju, nakon čega slijedi ciklus iteracija (b) selekcije, (c) rekombinacije i (d) mutacije, nakon čega algoritam vraća (e) konačnu populaciju.

1.2.1. Stvaranje početne populacije

Ključni element prilikom stvaranja početne populacije u genetskom algoritmu jest da ona zaista bude nasumična, a njezini članovi podjednako zastupljeni u različitim dijelovima prostora rješenja koji pretražujemo. Bilo kakva pristranost prema dijelu domene pretraživanja može ograničiti evolucijski proces i dovesti do pronalaska suboptimalnih rješenja. Tako je prilikom stvaranja početne populacije peptida ključno imati zastupljene sekvence različitih duljina i sastava. Shodno tome, u većini slučajeva se za svaku jedinku

najprije odredi željena duljina, te se nakon toga za svaku poziciju bira aminokiselina iz dozvoljenog skupa koji se obično sastoji od 20 proteinogenih aminokiselina. Proces se ponavlja dok se ne stvori željeni broj jedinki (slika 1.8.a). Iako ju je poželjno izbjegavati, pristranost prema dijelu prostora pretrage nije uvijek loša. Primjerice, ako smo identificirali određeni motiv koji sekvenci osigurava antimikrobnu aktivnost, te želimo pronaći peptide koje imaju minimalnu toksičnost uz zadržavanje navedenog motiva, uvođenjem pristranosti te očuvanjem specifičnih aminokiselina, pretragu možemo usmjeriti na dio kemijskog prostora koji sadrži rješenja od interesa (slika 1.8.b). Dodatni parametar je veličina populacije, koja može jako varirati od slučaja do slučaja, ali se održava konstantnom za vrijeme cijele evolucije. Jednostavniji problemi mogu pronaći dobro rješenje već s nekoliko jedinki, dok je za višedimenzionalne probleme obično potrebno više desetaka ili stotina jedinki. Sve navedeno se mora uzeti u obzir u ovisnosti o problemu koji rješavamo.

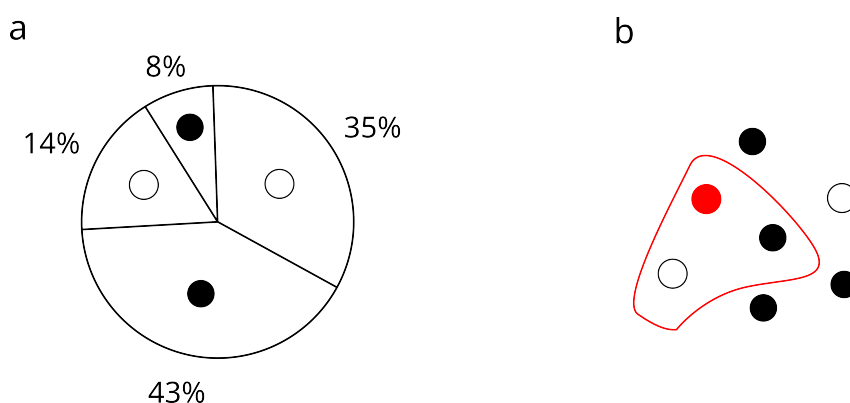
a	b
KKLFKKILKYLKK	MCE S ASSK
KSDLEESKEWIRRS	MCEQWG W K
RTLLLAGADKV	MCEYTGI I YK
SPYRCGSPDSRGSENTRCLIKK	MCE N IEYGGRGMSLEK D IK
ALGVATSAQITAAVALVEAKQAR	MCE S K
GSIPCGESCVFIPCISAIIGCSCSNKVCYKN	MCE W KKWKK W K
HGWGEDANLAMNPS	MCE L EKK
LRHSNIVKLY	MCE F K
HGQHGVHG	MCE F FK
KKWLKCLKLKK	MCE S WLHDIWDWVCIVLSK

Slika 1.8: Stvaranje (a) potpuno nasumične populacije peptida i (b) populacije koja zadržava motiv MCE...K u svim sekvencama.

1.2.2. Mehanizam selekcije

Svaka iteracija genetskog algoritma stvara m novih rješenja, pri čemu je prvi korak odabir roditelja čiji će se genomi rekombinirati. Cilj selekcije jest uvesti pristranost prema odabiru boljih rješenja kao roditelja, no još uvijek svim jedinkama dati mogućnost da budu roditelji. Stupanj pristranosti prema boljim rješenjima nazivamo selekcijskim pritiskom. Jedan od češćih način provođenja selekcije jest odabir kotačem sreće (eng. roulette wheel selection) [82]. U navedenom postupku, svim jedinkama u populaciji dodjeljuje se vjerojatnost odabira proporcionalna njihovoj kvaliteti izračunatoj funkcijom dobrote (slika 1.9.a). Potom se stohastički bira jedinka iz populacije. Ovako su ostvareni osnovni

uvjeti selekcijskog postupka, prednost je dana boljim jedinkama, ali niti jedna jedinka nije isključena iz mogućeg odabira. Druga često korištena metoda jest turnirska selekcija (eng. tournament selection) [21], gdje se najprije nasumično vrši odabir k jedinki iz populacije, te se među njima zadržava ona najbolja, sukladno vrijednosti funkcije dobrote (slika 1.9.b). Tako je također uvedena pristranost prema boljim jedinkama, ali na manjem podskupu, pa i dalje niti jednoj jedinki nije isključena mogućnost da postane roditelj. Variranjem vrijednosti k možemo utjecati na selekcijski pritisak. Primjerice, ako je k jednak veličini populacije, tada ćemo uvijek odabrati isključivo najbolju jedinku u populaciji.



Slika 1.9: Prikaz čestih postupaka provođenja selekcije roditelja: (a) odabir kotačem sreće, gdje svaka jedinka dobiva vjerojatnost odabira sukladno svojoj kvaliteti, (b) turnirska selekcija, gdje se naprije vrši nasumičan odabir k jedinki iz populacije (crveni obrub), te se među njima bira ona najbolja (crvena točka) sukladno vrijednosti funkcije dobrote.

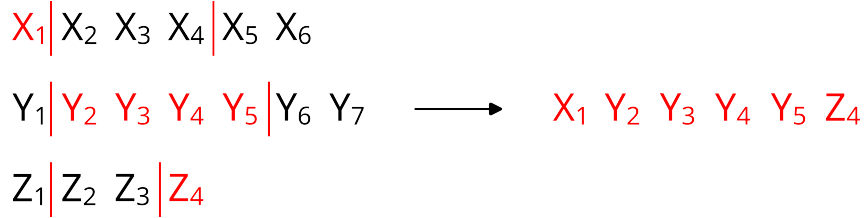
1.2.3. Postupak rekombinacije

Nakon odabira roditelja, pristupa se križanju njihovih genoma. Križanje nastoji stvoriti novu jedinku zadržavajući kvalitetna svojstva roditelja, te je ovisno o predstavi jedinke u računalu [179]. Uzmemo li genome koji se sastoje od jednodimenzionalnog vektora decimalnih brojeva, križanje je moguće napraviti jednadžbom 1.8, uzimajući aritmetičku sredinu za svaki član vektora koji se kombiniraju, pri čemu d_p predstavlja p -ti član vektora nove jedinke, n broj roditelja, a $r_{i,p}$ je p -ti član vektora i -tog roditelja. Također, vrijednosti se ne moraju kombinirati u jednakom omjeru, već je za svaki član vektora moguće nasumično odabrati decimalni broj ϵ_i koji diktira postotak udjela svakog roditelja u rješenju (jednadžba 1.9). Dodatno, vrijednosti genoma možemo promatrati kao granice raspona te nasumično odabrati broj iz istoga.

$$d_p = \frac{\sum_i^n r_{i,p}}{n} \quad (1.8)$$

$$d_p = \sum_i^n r_{i,p} * \epsilon_i, \quad \sum_i \epsilon_i = 1 \quad (1.9)$$

Za genome koji se sastoje od kategoričkih vrijednosti, obično se koristi križanje s k točaka prijeloma. U navedenom postupku, genom svakog roditelja razdvaja se na k dijelova, te se dijelovi izmjenično kombiniraju kako bi se stvorila nova jedinka (slika 1.10, primjer s 3 roditelja i $k = 2$). Pri provođenju selekcije bitno je uzeti u obzir što genom predstavlja. Primjerice, ako genom kodira redoslijed obilaska gradova, pri čemu se svaki grad smije pojaviti isključivo jednom, postupak rekombinacije ne bi smio iznjedriti genom koji krši navedeno. Također, ako govorimo o genomima definiranim decimalnim brojevima, valja obratiti pozornost postoje li granice vrijednosti svakog člana, kako se u genomu ne bi pojavile vrijednosti koje nemaju podlogu u problemu koji rješavamo; primjerice, postotak veći od 100%. Općenito, pri dizajniranju mehanizma rekombinacije, važno je obratiti pozornost na to što genom predstavlja. Ideja je kombiniranjem kvalitetnih roditelja iznjedriti još bolju jedinku.

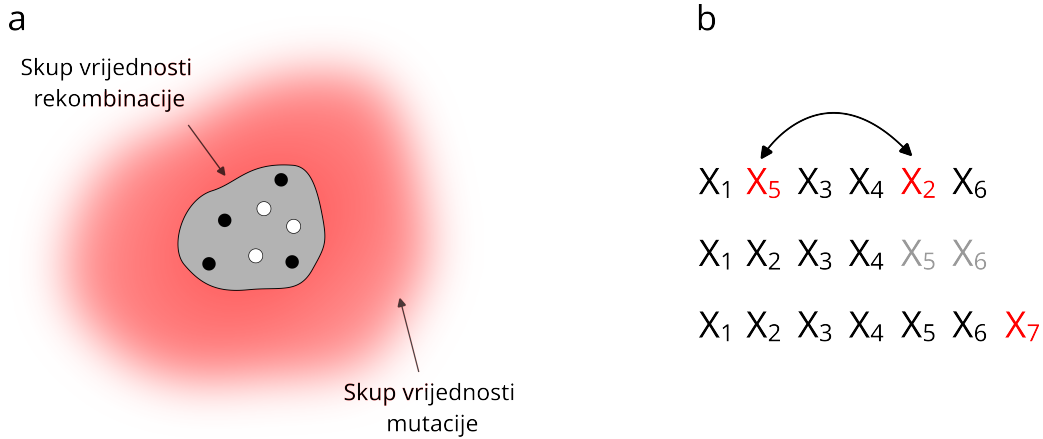


Slika 1.10: Prikaz križanja kategoričkih genoma triju roditelja s dvije točke prijeloma.

1.2.4. Mutacija genoma

Početni postupak stvaranja nasumične populacije indirektno definira raspon vrijednosti koje jedinke mogu poprimiti u genomu. Postupkom rekombinacije, vrijednosti se kombiniraju, no ostaju unutar specifičnog skupa definiranog genomom početne populacije. Postupak mutacije omogućava provjeru rješenja izvan tog skupa i potencijalni pronalazak kvalitetnijih jedinki, odnosno izbjegavanje lokalnih optimuma (slika 1.11.a) [225].

Općenito, mutacija vrši promjene genoma, pri čemu je cilj da one budu relativno



Slika 1.11: (a) Vizualizacija skupa vrijednosti kojeg je moguće postići rekombinacijom i mutacijom, te (b) primjer tri tipa mutacije kategoričkih genoma: zamjena mjesta dviju vrijednosti, skraćivanje genoma, te produljenje genoma.

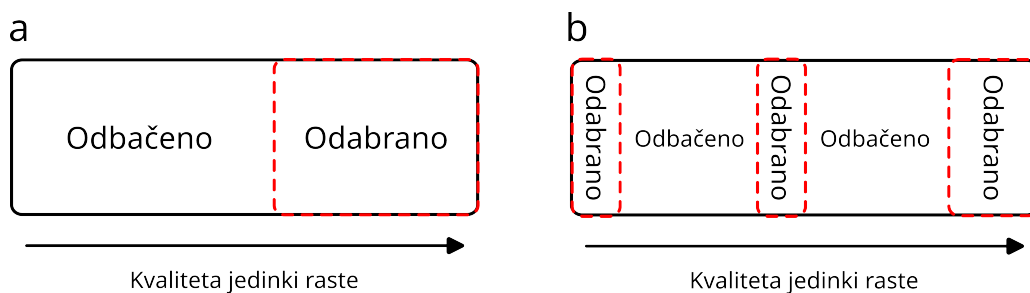
male, s uobičajenim vrijednostima izmjena između 1% i 5% genoma, kako se sva potencijalno dobra, prethodno dostignuta svojstva ne bi izgubila. Kao i kod postupka križanja, mutaciju također treba prilagoditi genomu i tipu problema koji rješavamo. Tako se aritmetičke vrijednosti genoma mogu mutirati zamjenom članova genoma nasumičnim brojem u predefiniranom rasponu. Također, svakom članu genoma d_p može se dodati nasumična perturbacija ϵ (jednadžba 1.10). Kategoričke vrijednosti, s druge strane, mogu zamijeniti mjesta u genomu ili se duljina genoma može modificirati dodajući/oduzimajući članove niza (slika 1.11.b).

$$d'_p = d_p + \epsilon, \quad \epsilon \in R \quad (1.10)$$

1.2.5. Odabir jedinki za sljedeću generaciju

Nakon što se mehanizmima selekcije, rekombinacije i mutacije stvori proizvoljan broj jedinki, pristupa se njihovom odabiru za sljedeću generaciju. Mehanizam odabira ima dvojaku svrhu. Najprije, uklanja jedinke koje su u odnosu na ostatak populacije ocijenjene lošima, što pridonosi pomicanju populacije prema kvalitetnijim rješenjima. Nadalje, održava veličinu populacije konstantnom. Naime, ako bi postupak opetovano stvarao nove jedinke bez mogućnosti njihovog uklanjanja, populacija bi u konačnici, uz dovoljan broj iteracija, dosegla veličinu koja prelazi hardverske mogućnosti pohrane memorije nasumičnog pristupa (eng. random access memory; RAM). Također, veći broj jedinki u popu-

laciji može nepotrebno produljiti vrijeme izvođenja algoritma, s obzirom na to da određeni mehanizmi selekcije, poput metode odabira kotačem sreće, računaju vjerojatnost odabira za svaku jedinku u populaciji. Kao i kod selekcije, u mehanizmu odabira također želimo uvesti pristranost prema boljim jedinkama, a stupanj pristranosti utječe na brzinu konvergencije i selekcijski pritisak. Jedna od najjednostavnijih metoda jest poredati jedinke prema kvaliteti i u iduću generaciju proslijediti isključivo one najbolje (slika 1.12.a). Ova metoda stvara snažnu pristranost prema boljim rješenjima, ali istovremeno može ukloniti kvalitetne dijelove genoma sadržane u nešto lošijim jedinkama. Modifikacija metode uključuje odabir određenog postotka najboljih jedinki, što se još naziva i elitizam, te nasumični odabir u preostalom dijelu populacije (slika 1.12.b). Navedeno smanjuje pritisak prema boljim jedinkama, može produljiti vrijeme do konvergencije, ali ima i veću vjerojatnost zadržati konstruktivne dijelove genoma, čak i ako se oni nalaze u nešto lošijim jedinkama. Odabir se može provoditi i prethodno opisanom metodom kotača sreće (slika 1.9.a). Sva-koj jedinki pridodijeli se vjerojatnost odabira izračunata na temelju njezine kvalitete, te se stohastičkim postupkom vrši odabir za sljedeću generaciju. Tako uvodimo pristranost prema boljim rješenjima, ali niti jedna jedinka nije isključena iz mogućeg odabira.



Slika 1.12: Prikaz čestih mehanizama odabira jedinki za sljedeću generaciju: (a) elitistički odabir isključivo najboljih jedinki i (b) elitizam uz zadržavanje nasumičnog odabira u preostalom dijelu populacije.

1.2.6. Evolucijski algoritmi kao generativni modeli

Veličina kemijskog prostora koje zauzimaju sve moguće molekule procijenjena je na 10^{60} [101]. Pretražiti i pronaći korisne molekule u navedenom prostoru vrlo je izazovno. Uzmemo li da vrednovanje po molekuli iziskuje samo 1 ms, pretraga peptida duljine 20 proteinogenih aminokiselina trajala bi $3,3 \cdot 10^{15}$ godina. Dodamo li pretrazi višestruke kriterije poput procjene mogućnosti sinteze, afiniteta vezanja za određenu metu, svojstava

ADME (apsorpcija, distribucija, metabolizam, izlučivanje), iscrpna pretraga čak i vrlo malih kemijskih prostora postaje praktično nemoguća. Usmjerenim pretraživanjem, evolucijski algoritmi mogu doskočiti navedenim ograničenjima tradicionalnih metoda i vrednujući mali dio prostora od interesa ponuditi kvalitetne kandidate. Dakako, stohastička priroda evolucijskih algoritama ne može ponuditi jamstvo pronalaska najboljih rješenja, no s obzirom na to da iscrpna pretraga prostora peptida vrlo brzo postaje nemoguća s rastom broja aminokiselina, evolucijski algoritmi su se isprofilirali kao vrijedan alat za dizajn novih molekula. Tako su korišteni za stvaranje novih baktericidnih peptida [48, 211, 291], automatiziranje otkrivanja novih vodećih spojeva (eng. lead compounds) [208, 241, 285], optimiziranje molekula predstavljenih strukturom grafova [116], te općenito za ubrzavanje pretrage molekula [36] i izbjegavanje uobičajenih kemijskih obrazaca koji se javljaju kao posljedica korištenja prethodno potvrđenih molekula kao polazišta za dizajn novih [191]. Sukladno tome, evolucijski algoritmi se mogu koristiti kao generativni modeli za stvaranje novih peptida sa specifičnim svojstvima.

1.3. Hipoteze i znanstveni doprinosi

Istraživanje koje ovaj doktorski rad obuhvaća provedeno je u nekoliko faza, te će kao takvo biti i predstavljeno u tekstu. Najprije je sistematičnom provjerom shema predstava peptida u računalu utvrđen nedostatak predstave koja bi istovremeno obuhvatila fizikalno-kemijska svojstva peptida i informaciju o njegovoj primarnoj strukturi, te tako dala kvalitetnije podatke modelima strojnog učenja. Po temeljito provedenoj analizi, predložena je i ispitana nova shema koja otklanja postojeće nedostatke (poglavlje 2.). Potom je pristupljeno proučavanju katalitičkih peptida i dizajnu genetskog algoritma za pretragu kemijskog prostora peptida sa svojstvom hidrolize estera (poglavlje 3.). U algoritam su osim standardnih evolucijskih mehanizama ugrađeni i mehanizmi za očuvanje raznolikosti molekula, te mogućnost postavljanja preferencije prema određenoj duljini molekule. Nakon detaljne analize algoritma i stvorenih peptida, utvrđeno je da bi svakako bilo korisno istražiti i validirati primjenu navedene evolucijske sheme i na drugim skupinama peptida. Sukladno tome, korištenjem prethodno razvijene sheme predstave peptida u računalu i idejama proizašlim pri proučavanju katalitičkih peptida, pristupljeno je stvaranju modela predviđanja i generativnog modela umjetne inteligencije za stvaranje novih peptida sa

svojstvom samosastavljanja (poglavlje 4.). Peptidi predloženi algoritmom validirani su postupkom molekularne dinamike i eksperimentalnom *in vitro* validacijom, te su dokazali 80%-tnu točnost generativnog modela. Fokus dijela istraživanja bio je i na problematici Alzheimerove bolesti. Po proučenim mehanizmima koji dovode do njezinog nastajanja i razvoja, enzim beta-sekretaze 1 identificiran je kao meta čijim bi se inhibiranjem moglo utjecati na koncentraciju amiloidnih nakupina u mozgu pacijenata, odnosno potaknuti njihovo smanjenje. Nastavno na to, pristupljeno je razvoju evolucijskog algoritma koji pronalazi peptidne inhibitore enzima beta-sekretaze 1, uz dodatni kriterij procjene prolaska krvno-moždane barijere (poglavlje 5.). Pritom su pronađene sekvence koje su u primarnoj računalnoj analizi pokazale povoljnija svojstva u odnosu na trenutno patentirane molekule iz literature. Posljednju nit istraživanja ovog doktorskog rada predstavlja optimizacija knjižnica peptida. Knjižnice peptida predstavljaju efikasan alat za istovremenu eksperimentalnu provjeru mnoštva peptida, no kasnija dekonvolucija i izdvajanje specifičnih molekula iz otopine mogu predstavljati problem. Zbog toga, razvijen je genetski algoritam koji izvornu knjižnicu od interesa razdvaja na određen broj podskupova uz zadržavanje većine spojeva i smanjenje preklapanja između njih (poglavlje 6.).

Sukladno navedenom, definirane su sljedeće hipoteze doktorskog rada:

1. Genetski algoritam vođen modelom predviđanja strojnog učenja može poslužiti kao osnova generativnog modela za otkrivanje peptida ciljanih svojstava.
2. Moguće je razviti generativni model za pronalaženje peptida s visokim afinitetom vezanja za metu i prolaska krvno-moždane barijere koristeći višekriterijski genetski algoritam.
3. Genetski algoritam može optimizirati knjižnice peptida koje su pogodnije za sintezu i dekonvoluciju.

Također, definirani su i sljedeći znanstveni doprinosi doktorskog rada:

1. Generativni model temeljen na genetskom algoritmu s operatorima prilagođenim evoluciji peptida, mogućnošću kontrole duljine sekvenci i raznovrsnosti populacije, te modelom predviđanja strojnog učenja kao funkcijom dobrote za otkrivanje peptida ciljanih svojstava.

2. Višekriterijski generativni model zasnovan na genetskom algoritmu za otkrivanje peptida s visokom vjerojatnošću prolaska krvno-moždane barijere i antagonističko djelovanje prema enzimu beta-sekretaze 1.
3. Model optimizacije knjižnica peptida za razdvajanje na podskupove pogodnije za dekonvoluciju nakon sinteze.
4. Lista peptida stvorena generativnim modelom s eksperimentalno potvrđenim svojstvom samosastavljanja.

Model predložen prvim doprinosom primijenjen je na stvaranje peptida s visokom vjerojatnošću za ispoljavanje:

- katalitičke aktivnosti pri hidrolizi estera. U okviru ovoga, razvijen je i model predviđanja za identifikaciju peptida s visokom vjerojatnošću za uzrokovanje hidrolize para-nitrofenil acetata, temeljen na povratnim neuronskim mrežama.
- svojstva samosastavljanja. U okviru ovoga, razvijen je i model predviđanja za identifikaciju peptida s visokom vjerojatnošću za samosastavljanje, temeljen na povratnim neuronskim mrežama, a koji koristi inovativnu shemu procjene sklonosti agregaciji kroz poznate sklonosti agregaciji aminokiselina, dipeptida i tripeptida.

2. SHEMA SEKVENCIJALNIH SVOJSTAVA

Rezultati istraživanja opisani u ovom poglavlju objavljeni su u sljedećem znanstvenom radu: E. Otović, M. Njirjak, D. Kalafatovic, and G. Mauša. Sequential properties representation scheme for recurrent neural network-based prediction of therapeutic peptides. Journal of Chemical Information and Modeling, 62(12):2961–2972, 2022. [197].

Računalno potpomognut probir molekula predstavlja alternativu skupoj i vremenski zahtjevnoj eksperimentalnoj validaciji [262]. Kvalitetno naučeni modeli strojnog učenja u ovoj domeni mogu doseći točnosti iznad 90% uz vrijeme predviđanja manje od 50 milisekundi po molekuli [24, 296]. Zbog toga, modeli se često koriste za predselekciju molekula prije provođenja eksperimentalnih studija, povećavajući tako vjerojatnost pozitivnog rezultata i usmjeravajući resurse tamo gdje oni mogu ostvariti najveću korist. Međutim, modeli strojnog učenja rade isključivo s numeričkim podacima, pa je peptide i druge molekule nužno tako i predstaviti. U literaturi postoje različite numeričke predstave peptida u računalu, a razlikuju se po tome na koja se svojstva molekule fokusiraju, pri čemu je osnovna podjela na kompozicijske i strukturne. Kompozicijske sheme karakteriziraju sastav i fizikalno-kemijska svojstva peptida i većinom su invarijantne na raspored aminokiselina u sekvenci. Primjerice, shema predstave pomoću aminokiseliniskog sastava (eng. amino acid composition; AAC) predstavlja svaki peptid kao vektor cijeli brojeva duljine 20, pri čemu svaki član vektora bilježi frekvenciju pojave određene aminokiseline u peptidu [224]. Na sličnom principu rade i ostale kompozitne sheme poput dipeptidnog sastava (eng. dipeptide composition, DPC) i zajedničkih trijada (eng. conjoint triad; CTF) [24].

Vrlo česta predstava peptida jest pomoću izračunatih fizikalno-kemijskih svojstava. Svaki peptid predstavljen je skupom vrijednosti poput hidrofnosti, izoelektrične točke, molekularne mase, i slično [212]. Glavni problem navedenih shema jest što u većini slučajeva ne uzimaju u obzir permutacije gradivnih elemenata unutar sekvence, što je vrlo bitno za funkciju [69]. Uzmemo li u obzir jednoslovne kodove kojima su aminokiseline predstavljene (tablica 2.1), peptidi QWR i WRQ imaju identičnu AAC predstavu i iz perspektive računala su istovjetni.

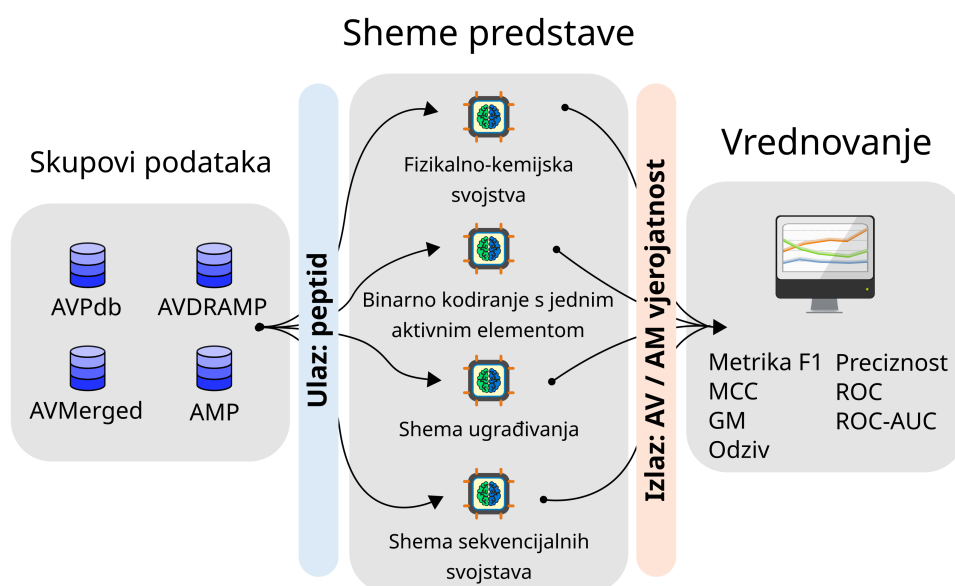
Tablica 2.1: Dvadeset proteinogenih aminokiselina zajedno s pripadajućim jednoslovnim kodovima za predstavu primarne strukture peptida/proteina.

Aminokiselina	Jednoslovni kod
Alanin	A
Arginin	R
Asparagin	N
Asparaginska kiselina	D
Cistein	C
Glutaminska kiselina	E
Glutamin	Q
Glicin	G
Histidin	H
Izoleucin	I
Leucin	L
Lizin	K
Metionin	M
Fenilalanin	F
Prolin	P
Serin	S
Treonin	T
Triptofan	W
Tirozin	Y
Valin	V

Strukturne sheme predstave peptida u računalu opisuju poredak gradivnih elemenata u sekvenci, poput aminokiselina, dipeptida i tripeptida. Svakoj aminokiselini može se dodijeliti jedinstvena numerička vrijednost ili vektor vrijednosti, pa je peptid predstavljen kao niz brojeva ili vektora, što se još naziva i binarno kodiranje s jednim aktivnim elementom (eng. one-hot encoding) [181]. U ovom slučaju, peptidi QWR i WRQ imaju različite predstave u računalu, što omogućuje algoritmima da ih uspješnije razlikuju. Često korištena shema jest i ugrađivanje (eng. embedding), koja dinamički mapira podatak u

prostor numeričkih vektora [242]. Prednost u odnosu na binarno kodiranje s jednim aktivnim elementom jest što se parametri mapiranja uče tijekom samog procesa učenja modela, a udaljenost između vektora predstavlja sličnost mapiranih podataka, što modelu strojnog učenja daje dodatnu informaciju i može biti korisno pri analizi i interpretaciji rezultata [269]. Ovakve predstave često se koriste u kombinaciji s povratnim neuronskim mrežama (eng. recurrent neural networks; RNN) ili transformerima [156, 288], te u sprezi s njima modelu strojnog učenja daju jasnu informaciju o poretku elemenata koji sačinjavaju sekvencu.

Međutim, niti jedna dosad poznata predstava peptida u računalu ne uzima u obzir obje karakteristike peptida, njegov sastav i strukturu. Sprega tih dviju vrsta parametara ključna je za određivanje karakteristika peptida i zaključivanje o njegovim funkcijama, pa tako i za podizanje točnosti i poboljšanja generalizacije modela strojnog učenja. U ovom poglavlju predložena je nova, hibridna shema predstave naziva „shema sekvencijalnih svojstava”, koja obuhvaća fizikalno-kemijske i strukturne karakteristike peptida, dizajnirana za predstavu peptida u računalu u svrhu provedbe strojnog učenja. Shema je uspoređena s tri predstave uvriježene u literaturi: predstava pomoću fizikalno-kemijskih svojstava peptida, binarno kodiranje s jednim aktivnim elementom i shema ugrađivanja, na problemu klasifikacije antiviralnih i antimikrobnih peptida (slika 2.1).



Slika 2.1: Shematski prikaz istraživanja. Svaka od 4 predstave peptida testirena je u dva-put ponovljenoj unakrsnoj validaciji u 10 preklopa za tri antiviralna i jedan antimikrobni skup podataka, te vrednovana pomoću sedam metrika. Preuzeto iz [197] i prevedeno.

2.1. Metodologija

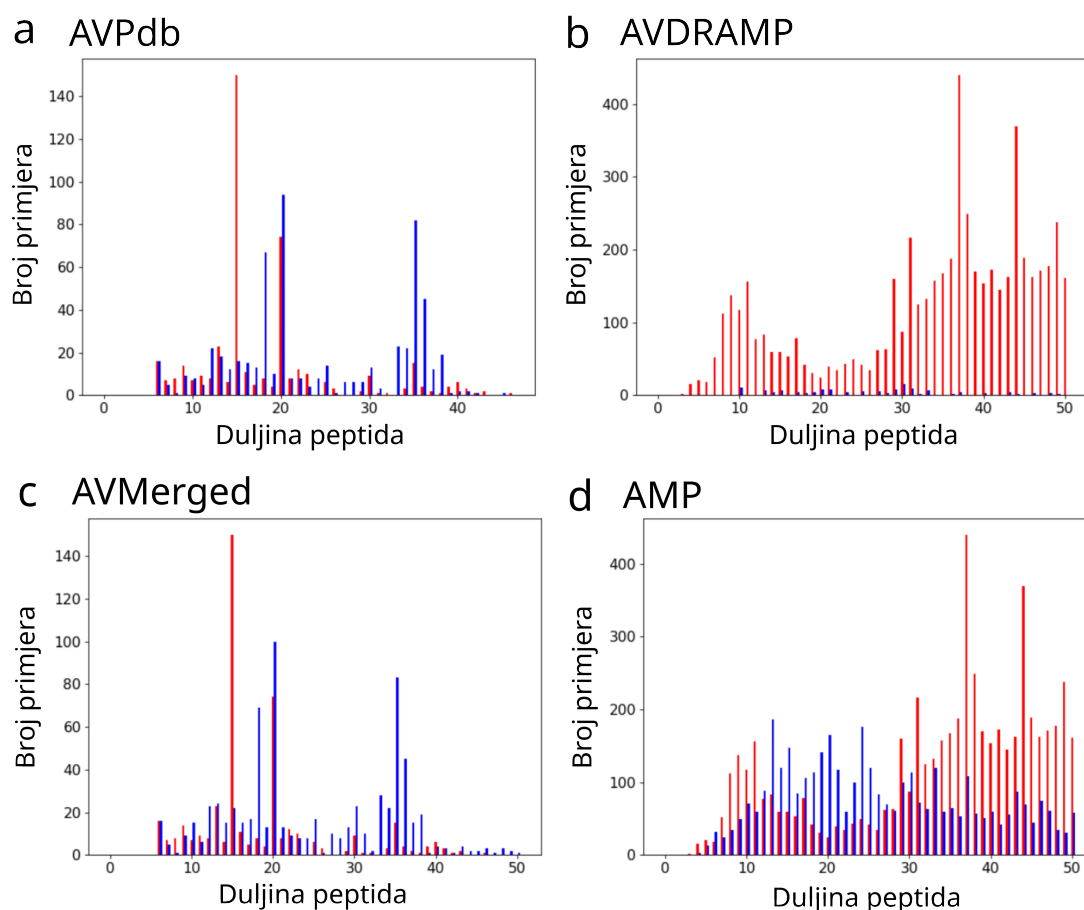
2.1.1. Skupovi podataka

U svrhu provedbe eksperimenata i usporedbe shema predstave peptida u računalu, korištena su četiri skupa podataka: AVPdb, AVDRAMP, AVMerged i AMP (tablica 2.2). Skupovi podataka stvoreni su kombiniranjem prijavljenih antimikrobnih i antiviralnih peptida iz literature [124, 215], te peptida iz baze Uniprot [253]. Podaci su dodatno filtrirani izostavljanjem peptida koji sadrže neproteinogene aminokiseline, sukladno činjenici da većina alata za računanje značajki podržava isključivo proteinogene aminokiseline, te onih čija duljina prelazi 50 aminokiselina kako bismo se usredotočili na peptidne sekvence i spriječili prelazak u domenu proteina. S obzirom na to da korišteni skupovi podataka sadrže isključivo pozitivne primjere, odnosno peptide koji su pokazali pozitivno djelovanje u odnosu na ispitanu aktivnost, uvriježen način izdvajanja negativnih podataka, nužnih za strojno učenje, jest dohvaćanje iz baze Uniprot [280]. Uniprot je visokokvalitetna baza podataka proteina i njihovih funkcija, te se često koristi u ovakvim tipovima istraživanja. Upit korišten za dobivanje negativnih podataka glasi: „NOT keyword:antimicrobial length:[0 TO 50] fragment:no AND reviewed:yes” i dohvaća isključivo cjelovite sekvence do 50 aminokiselina koje provjereno ne povezujemo s antimikrobnom aktivnošću.

Tablica 2.2: Pregled korištenih skupova podataka i izvora peptidnih sekvenci. Preuzeto iz [197] i prevedeno.

Skup podataka	Izvor	Broj pozitivnih peptida	Broj negativnih peptida
AVPdb	AVPdb [215]	598	444
AVDRAMP	DRAMP 2.0 (pozitivni peptidi) [124], Uniprot (negativni peptidi) [253]	149	5701
AVMerged	AVPdb (pozitivni i negativni peptidi) [215], DRAMP 2.0 (pozitivni peptidi) [124]	711	444
AMP	DRAMP 2.0 (pozitivni peptidi) [124], Uniprot (negativni peptidi) [253]	3708	5701

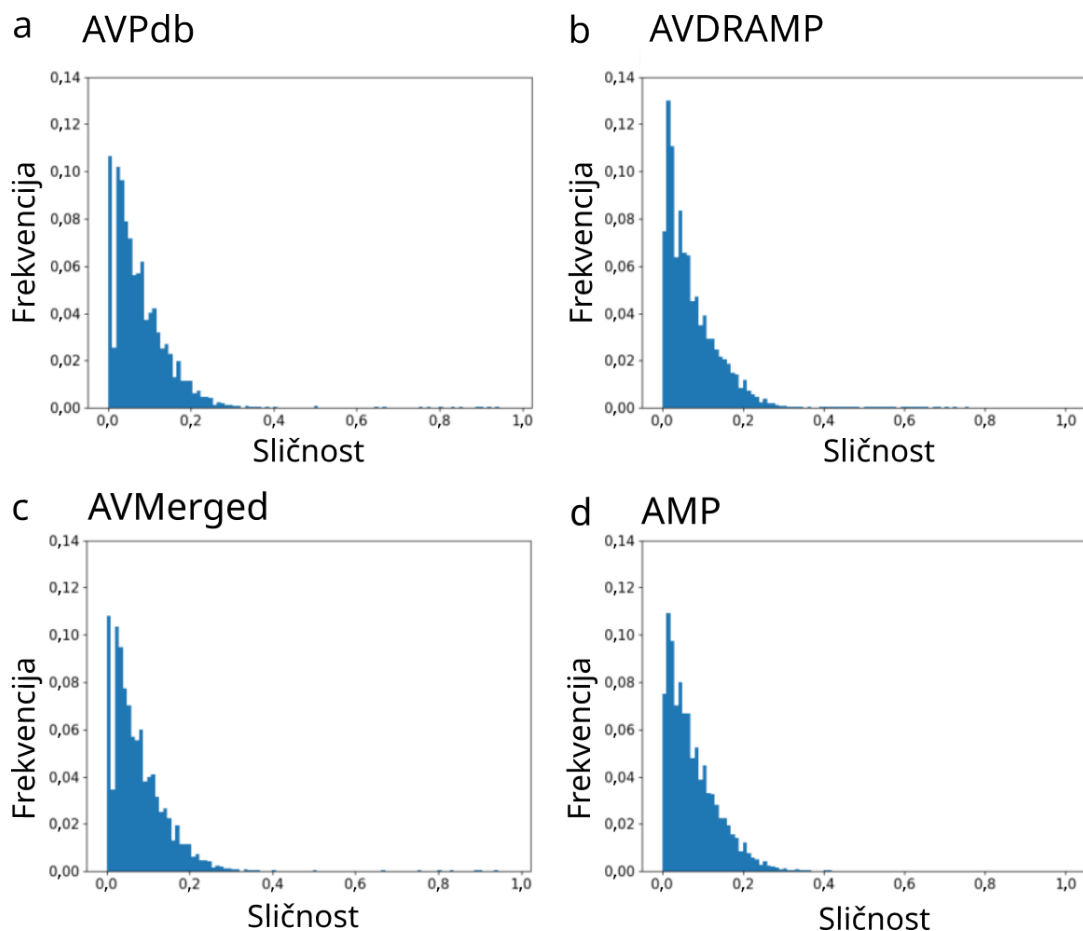
Važan aspekt prilikom učenja modela strojnog učenja jest i razdioba podataka. U slučaju binarne klasifikacije, teži se podjednako zastupljenosti obiju klasa. U slučaju nejednake razdiobe, klasa s većim brojem primjera ima snažniji utjecaj na prilagodbu modela prilikom učenja, te posljedično može smanjiti njegove generalizacijske sposobnosti. Razdioba podataka po skupovima korištenim u ovom istraživanju prikazana slici 2.2 ukazuje na ravnomjernu razdiobu u svim skupovima osim AVDRAMP. Međutim, kako bismo što točnije ocijenili kvalitetu modela strojnog učenja, i time indirektno usporedili količinu informacija koje sheme predstave dostavljaju modelima, uz standardne mjere točnosti klasifikacije (eng. accuracy) koristili smo različite metrike vrednovanja koje u obzir uzimaju neujednačenost u razdiobama klasa. Više o tome može se pročitati u potpoglavlju 2.1.3..



Slika 2.2: Razdioba duljina pozitivnih (antimikrobnih/antiviralnih) peptida označenih crveno i negativnih peptida označenih plavo u skupovima podataka korištenim u istraživanju. Preuzeto iz [197] i prevedeno.

Osigurana je i raznovrsnost podataka kako bi se spriječilo ispoljavanje visoke točnosti isključivo zbog sličnosti u podacima. Sličnost među parovima peptida izračunata je kao postotak identičnih aminokiselina na istim mjestima u peptidu nakon globalnog porav-

nanja koristeći algoritam Needleman-Wunsch [186]. Parametri algoritma ostavljeni su na predefiniranim verzijama iz paketa scikit-bio [227]. Zaključeno je da više od 98% peptida dijeli 30% ili manje sličnosti s ostalima, što predstavlja dovoljnu raznovrsnost za provođenje eksperimenata (slika 2.3).



Slika 2.3: Histogrami sličnosti peptida dobiveni koristeći globalno poravnanje Needleman-Wunsch. Preuzeto iz [197] i prevedeno.

2.1.2. Sheme predstave peptida u računalu

Uspoređena su četiri načina numeričke predstave peptida u računalu, jedan od kojih je shema sekvencijalnih svojstava predložena u okviru ovog istraživanja:

1. Fizikalno-kemijska svojstva

Peptid je predstavljen kao vektor od 28 fizikalno-kemijskih svojstava izračunatih iz sekvence.

2. Binarno kodiranje s jednim aktivnim elementom

Peptid je predstavljen nizom vektora, pri čemu svaki vektor sadrži 20 vrijednosti. Svaki član vektora predstavlja jednu od 20 proteinogenih aminokiselina. U svakom vektoru, vrijednost jedan je dozvoljena isključivo na jednom mjestu, dok ostala mjesta moraju sadržavati nulu.

3. Ugrađivanje riječi (eng. word embedding)

Peptid je predstavljen nizom N -dimenzionalnih vektora, pri čemu N označava odabranu veličinu sloja ugrađivanja u arhitekturi modela strojnog učenja. Vrijednosti vektora su decimalni brojevi koji se uče prilikom procesa učenja modela.

4. Shema sekvencijalnih svojstava

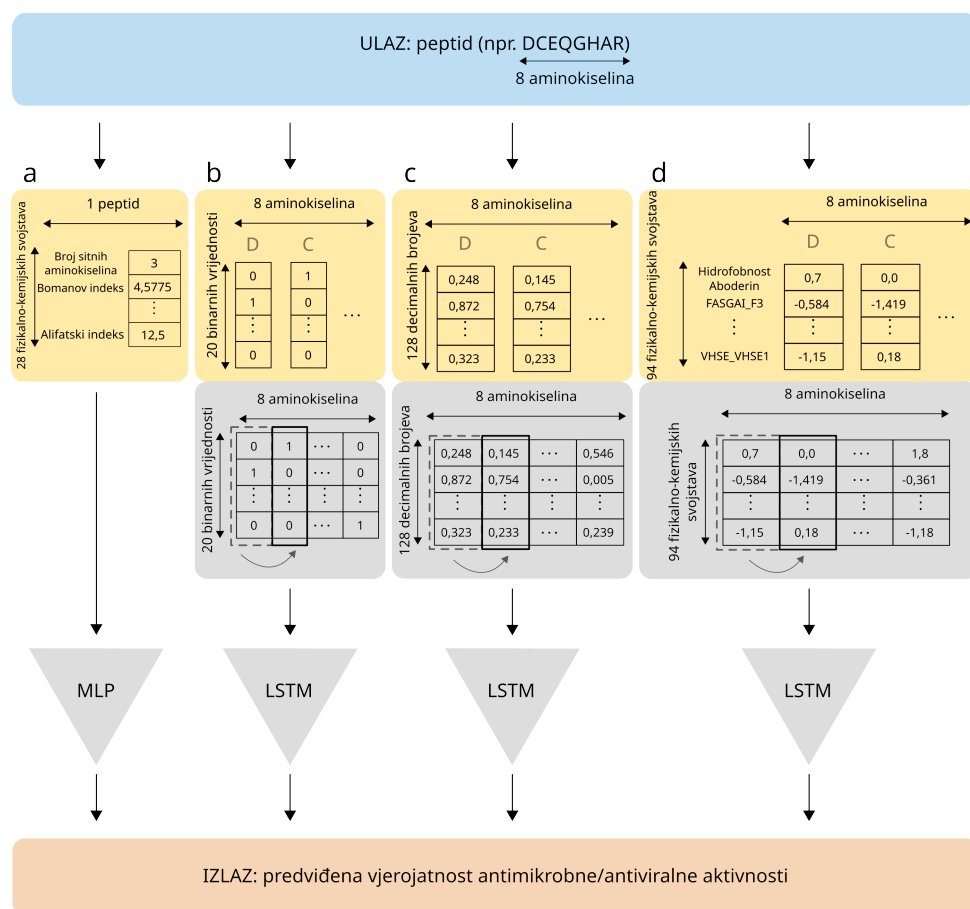
Peptid je predstavljen nizom vektora od 94 vrijednosti. Vrijednosti predstavljaju fizikalno-kemijska svojstva pojedine aminokiseline.

Pregled korištenih shema predstave dan je na slici 2.4.

Fizikalno-kemijska svojstva peptida

Svaki peptid predstavljen je s 28 fizikalno-kemijskih svojstava izračunatih koristeći programski paket *peptides.py* [140]. Svojstva uključuju hidrofobnost na Eisenbergovoj skali [65], tri svojstva Cruciani [47], hidrofobni moment s kutom rotacije od 100° i parcijalnom duljinom 11 [66], indeks nestabilnosti [91], alifatski indeks [113], naboj pri pH vrijednosti od 7,4 na Lehnigerovoj skali i izoelektričnu točku po Lehnigerovoj skali [149]. Za karakterizaciju aminokiselina, korištena je sljedeća podjela: sitne (A, C, G, S, T), male (A, C, D, G, N, P, S, T, V), alifatske (A, I, L, V), aromatske (F, H, W, Y), polarne (D, E, H, K, N, Q, R, S, T), nepolarne (A, C, F, G, I, L, M, P, V, W, Y), električno nabijene (D, E, H, K, R), bazične (H, K, R), kisele (D, E). Kvantificirana je pripadnost svake aminokiseline peptida pojedinoj klasi, te su apsolutne i relativne frekvencije klasa korištene kao podaci za karakterizaciju peptida.

Prednost ovakve sheme jest invarijantnost duljine predstave u odnosu na duljinu peptida, tj. svaki peptid, neovisno o duljini, bit će opisan pomoću fiksnog broja od 28 značajki. Sukladno tome, ovakva predstava pogodna je za obradu pomoću modela poput višeslojnog perceptrona (eng. multilayer perceptron; MLP), stabla odluke, i slično [2, 181, 223,



Slika 2.4: Pregled korištenih shema predstave peptida u računalu (označene žutim pravokutnicima) za peptid DCEQGHAR: (a) fizikalno-kemijska svojstva peptida, (b) binarno kodiranje s jednim aktivnim elementom, (c) ugrađivanje riječi (primjer s veličinom sloja od 128 decimalnih vrijednosti), (d) shema sekvencijalnih svojstava. S obzirom na svoju sekvencijalnu prirodu, podaci u (b), (c) i (d) se obrađuju povratnim neuronskim mrežama, odnosno slojevima duge kratkoročne memorije (označeni sivim trokutima), dok se podaci u (a) obrađuju višeslojnim perceptronom. Izlaz neuronske mreže predstavlja predviđenu vjerojatnost da ulazni peptid posjeduje antimikrobnu/antiviralnu aktivnost. Preuzeto iz [197] i prevedeno.

274, 287]. Manjkavost sheme oglada se u vrlo sličnim predstavama za peptide s bliskom razdiobom aminokiselina [69]. Primjerice, peptidi ACCD i ACDC imaju skoro identičnu fizikalno-kemijsku predstavu, s iznimkom hidrofobnog momenta, iako se zapravo mogu dosta drugačije ponašati u praksi (tablica 2.3).

Binarno kodiranje s jednim aktivnim elementom

Shema binarnog kodiranja s jednim aktivnim elementom zadržava informaciju o poretku aminokiselina u peptidu, no ni na koji način ne karakterizira njegova druga svojstva, primjerice fizikalno-kemijska. Svaka aminokiselina predstavljena je u obliku vektora od 21

Tablica 2.3: Usporedba fizikalno-kemijskih svojstava dvaju peptida s istim sastavom, ali drugačijim poretkom aminokiselina. Sva svojstva imaju identične vrijednosti, uz iznimku hidrofobnog momenta (podebljano). Preuzeto iz [197] i prevedeno.

Svojstvo	Peptid <i>ACCD</i>	Peptid <i>ACDC</i>
Svojstvo Cruciani 1	-0,2650	-0,2650
Svojstvo Cruciani 2	-0,6475	-0,6475
Svojstvo Cruciani 3	-0,0775	-0,0775
Indeks nestabilnosti	165,5	165,5
Bomanov indeks	1,0875	1,0875
Hidrofobnost (Eisenbergova skala)	0,075	0,075
Alifatski indeks	25	25
Izoelektrična točka (Lehningerova skala)	2,994992	2,994992
Naboj (Lehninger; pH=7,4)	-1,289589	-1,289589
Hidrofobni moment	0,2444636	0,3993655

člana [286]. Prvih 20 članova vektora predstavljaju proteinogene aminokiseline, svaki član jednu, dok zadnji član naznačuje kraj sekvence. Naime, prilikom učenja modela strojnog učenja, moguće je postići ubrzanje ako se podaci grupiraju u serije (eng. batch), te se iskoriste operacije grafičkih procesnih jedinica (eng. graphical processing unit; GPU) [85] poznate pod akronimom SIMT (eng. single instruction, multiple threads). Kako bismo to ostvarili, svi podaci u seriji moraju biti iste duljine, stoga se peptidi nadopunjuju neutralnim vrijednostima do duljine najduljeg peptida u skupu podataka. Neutralne vrijednosti ne utječu na predviđanje i ne daju dodatnu informaciju modelu strojnog učenja, već se isključivo koriste za ubrzanje procesa učenja. Posljedično, svaki peptid predstavljen je pomoću $M \cdot 21$ binarnih vrijednosti, pri čemu M označava duljinu najvećeg peptida u skupu podataka.

Ugrađivanje riječi

Peptidi također mogu biti predstavljeni kao vektori decimalnih vrijednosti. Sloj ugrađivanja, postavljen na ulaz neuronske mreže, preslikava diskretne vrijednosti u N -dimenzionalni geometrijski prostor, pri čemu se pravila preslikavanja uče prilikom samog učenja modela. Tako osim samog preslikavanja, algoritam istovremeno uči sličnosti i razlike među ulaznim vrijednostima, mapirajući ih bliže ili dalje u geometrijskom prostoru. Navedeno pomaže algoritmu strojnog učenja u uspješnijem razlikovanju klasa peptida. Ova tehnika uvedena je za učenje modela koji procesiraju prirodan jezik (eng. natural language processing; NLP) [121] i uspješno primijenjena za predviđanje aktivnosti peptida [287]. Kako

bismo uspješno primijenili tehniku ugrađivanja za predviđanje peptida, svaku aminokiselinu predstavili smo nasumičnim brojem iz skupa cijelih brojeva između 1 i 20, pri čemu dvjema aminokiselinama ne može biti dodijeljena ista vrijednost. Potom su sve sekvence nadopunjene nulama do duljine najduljeg peptida u skupu podataka, kako bismo uspješno mogli primijeniti prethodno opisane operacije SIMT i postići ubrzanje učenja. Prilikom učenja modela, sloj ugrađivanja mapira peptide u N -dimenzionalni vektorski prostor, pri čemu N predstavlja hiperparametar neuronske mreže.

Schema sekvencijalnih svojstava

Schema sekvencijalnih svojstava kombinira informacije o poretku aminokiselina i njihovim fizikalno-kemijskim svojstvima. Naime, svaki peptid predstavljen je listom 94-dimenzionalnih vektora, pri čemu svaki vektor sadrži informacije o fizikalno-kemijskim svojstvima jedne aminokiseline. Svojstva su dostupna kao dio skupa podataka `AAdatas` u programskom paketu *Peptides* [195]. Koristili smo isključivo svojstva definirana za 20 proteinogenih aminokiselina, a ako je svojstvo bilo dostupno na više različitih skala, primjerice hidrofobnost, korišteni su podaci za sve skale. Kako bismo označili kraj sekvence, te time omogućili nadopunu do duljine najduljeg peptida u skupu podataka i ubrzali učenje, u vektor je umetnuta dodatna vrijednost kao oznaka završetka peptida. Kao i u prethodnim slučajevima, navedeno omogućava iskorištavanje SIMT operacija GPU-ova koje ubrzavaju učenje. Tako je peptid duljine L predstavljen matricom vrijednosti $95 \times L$. Opisana hibridna shema predstave peptida u računalu trebala bi modelima strojnog učenja pružiti informacije o poretku aminokiselina, kao i o njihovim fizikalno-kemijskim svojstvima. Također, modeli strojnog učenja iz navedenih informacija mogu zaključivati o duljini sekvence, fizikalno-kemijskim značajkama podstruktura (aminokiseline, dipeptida, tripeptida), te o sastavu i strukturi cijelog peptida. Hipoteza jest da će navedeno podići točnost modela strojnog učenja za predviđanje antimikrobnih/antiviralnih peptida.

2.1.3. Modeli strojnog učenja i postupak usporedbe shema predstave

Usporedba shema i provjera hipoteze o informacijama sadržanim u sekvencijalnim svojstvima napravljena je pomoću dviju vrsta neuronskih mreža:

- Višeslojni perceptron

Koristi se za predviđanje temeljeno na predstavi shemom fizikalno-kemijskih svojstava.

- Konvolucijsko-povratna neuronska mreža

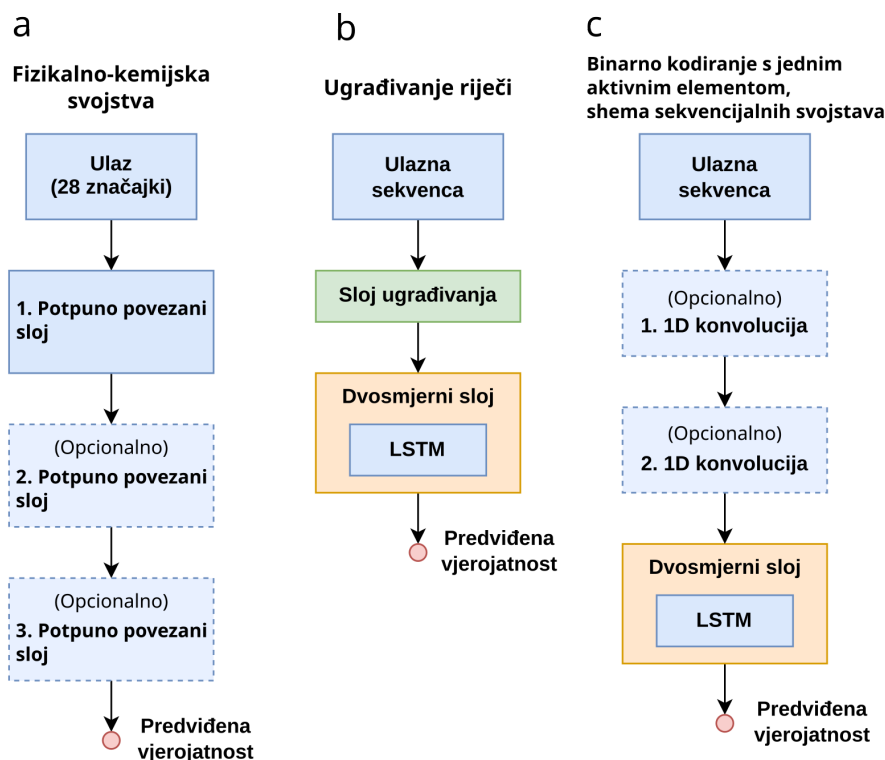
Koristi se za ostale sheme predstave peptida u računalu.

Nužnost prilagodbe arhitekture modela shemi predstave proizlazi iz načina na koji sheme enkodiraju peptide. Shema fizikalno-kemijskih svojstava predstavlja peptid kao cjelinu pomoću njegovih izračunatih svojstava, dok preostale sheme zasebno kodiraju gradivne jedinice peptida. Sukladno tome, konvolucijsko-povratne neuronske mreže, koje koračajno obrađuju podatke, nije uputno primijeniti na 28 fizikalno-kemijskih značajki jer između njih ne postoji sekvencijalna veza koja može pridonijeti točnosti modela. S obzirom na to da je višeslojni perceptron s jednim skrivenim slojem od 100 neurona i aktivacijskom funkcijom ReLU uspješno korišten za klasifikaciju antimikrobnih peptida, navedena arhitektura predstavljala je polazište za izradu modela strojnog učenja [13]. Optimizacija hiperparametara napravljena je pretraživanjem svih kombinacija koristeći sljedeće skupove mogućih vrijednosti:

- Broj skrivenih slojeva mreže: 1, 2, 3
- Broj neurona u svakom od skrivenih slojeva: 40, 80, 120, 160.

Ulaz u neuronsku mrežu je vektor od 28 fizikalno-kemijskih značajki, dok je izlaz sigmoidni neuron čija je vrijednost između 0 i 1, i predstavlja vjerojatnost da ulazna sekvenca posjeduje antimikrobnu/antiviralnu aktivnost (slika 2.5.a).

Arhitektura konvolucijsko-povratne neuronske mreže preuzeta je iz [287] (slika 2.5.b,c). Konvolucijski slojevi koriste se za izdvajanje obrazaca prisutnih u ulaznim podacima, dok povratni slojevi duge kratkoročne memorije (eng. long short-term memory; LSTM) [232] nastoje uočiti zavisnosti među obrascima i iskoristiti ih za predviđanje aktivnosti. Kako bi se spriječila prekomjerna naučenost modela, sloj za regularizaciju izbacivanjem neurona (eng. dropout layer) postavljen je između dvosmjernog sloja LSTM i izlaznog sigmoidnog neurona. Kao i u slučaju višeslojnog perceptrona, ulaz je numerički predstavljena peptidna sekvenca, a izlaz decimalna vrijednost između 0 i 1. Provedena je i optimizacija



Slika 2.5: Arhitekture modela korištenih neuronskih mreža za: a) višeslojni perceptron, b) povratnu neuronsku mrežu za shemu ugrađivanja riječi, c) konvolucijsko-povratnu neuronsku mrežu za sheme binarnog kodiranja s jednim aktivnim mjestom i sekvencijalna svojstva. Preuzeto iz [197] i prevedeno.

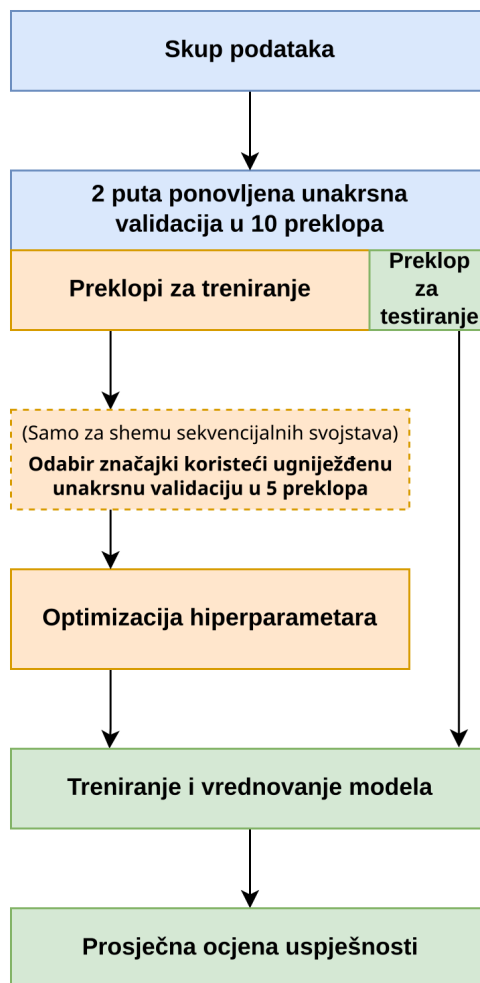
hiperparametara pretraživanjem svih kombinacija vrijednosti kako bi se pronašli najbolji parametri za svaki ispitni slučaj, koristeći sljedeće skupove mogućih vrijednosti:

- Veličina sloja ugrađivanja: 30, 50, 70, 90, 110, 130
- Broj konvolucijskih slojeva: 0, 1, 2
- Broj filtera u svakom od konvolucijskih slojeva: 16, 32 or 64
- Veličina jezgre u konvolucijskim slojevima: 4, 6, 8
- Broj elemenata u sloju LSTM: 64, 128, 256
- Postotak izbačenih neurona u sloju regularizacije: 10%, 20%, 30%.

Parametar veličine sloja ugrađivanja uzima se u obzir isključivo pri optimizaciji arhitekture mreže koja koristi sloj ugrađivanja, dok se parametri vezani za konvoluciju koriste isključivo u slučajevima binarnog kodiranja s jednim aktivnim elementom i sheme sekvencijalnih svojstava. Modeli su naučeni korištenjem stope učenja od 10^{-4} , veličine serije

32 peptida, optimizatora Adam [128], te u periodu od maksimalno 200 epoha. Također je upogonjen mehanizam ranijeg prestanka učenja za slučaj da se gubitak (eng. loss) na validacijskom skupu podataka ne smanji u 5 uzastopnih epoha. Normalizacija ulaznih vrijednosti neuronske mreže preporučuje se radi poboljšanja protoka gradijenta prilikom učenja [185, 237]. S obzirom na činjenicu da minimalne i maksimalne vrijednosti 28 fizikalno-kemijskih značajki peptida ne mogu biti unaprijed određene, korištena je metoda standardizacije podataka koja razdiobu svake značajke pretvara u razdiobu sa srednjom vrijednošću 0 i standardnom devijacijom 1. Kako su vrijednosti 94 fizikalno-kemijskih svojstava aminokiselina poznate unaprijed, navedene vrijednosti skalirane su u raspon $[-1, 1]$ koristeći tehniku min-max skaliranja [294].

Cjelokupni eksperiment sastoji se od 16 ispitnih slučajeva, budući da se svaka od 4 sheme predstave uči na 4 skupa podataka (slika 2.1). U svakom scenariju odvija se dvaput ponovljena stratificirana unakrsna validacija u 10 preklopa (eng. 10-fold cross-validation), pri čemu se 9 preklopa koristi za učenje modela i optimizaciju hiperparametara, a preostali preklap za konačno vrednovanje modela (slika 2.6). Svi preklopi imaju jednaku razdiobu pozitivne i negativne klase peptida, a podjela podataka identična je za sve ispitne slučajeve kako bi se omogućila što vjernija usporedba shema predstave. Unutar osnovne petlje unakrsne validacije, provodi se ugniježđena unakrsna validacija u 5 preklopa za optimizaciju hiperparametara i odabir značajki. Optimizacija hiperparametara provodi se testirajući sve kombinacije prethodno spomenutih vrijednosti parametara. Vrijednosti koje rezultiraju najvećom površinom ispod krivulje karakteristične funkcije prijemnika (eng. area under receiver operating characteristic curve; ROC-AUC) odabiru se kao konačni parametri modela. U ugniježđenoj petlji provodi se i odabir značajki za shemu sekvencijalnih svojstava. Postupkom koračajnog odabira značajki (eng. sequential feature selection) [73] iterativno se odabiru najinformativnije značajke kako bi se smanjilo računalno opterećenje prilikom učenja. Postupak se provodi isključivo za shemu sekvencijalnih svojstava zbog velikog broja značajki svake aminokiseline. U slučaju sheme fizikalno-kemijskih svojstava, odabir značajki nije potreban, dok se u slučajevima binarnog kodiranja s jednim aktivnim elementom i sheme ugrađivanja ne može primijeniti. Postupak odabira staje ako se vrijednost ROC-AUC ne poveća u tri uzastopne iteracije. Model se potom uči koristeći devet preklopa vanjske petlje unakrsne validacije i vrednuje na jednom preostalom preklopu.



Slika 2.6: Vizualizacija postupka provedbe eksperimenata na svakom od četiri skupa podataka. Preuzeto iz [197] i prevedeno.

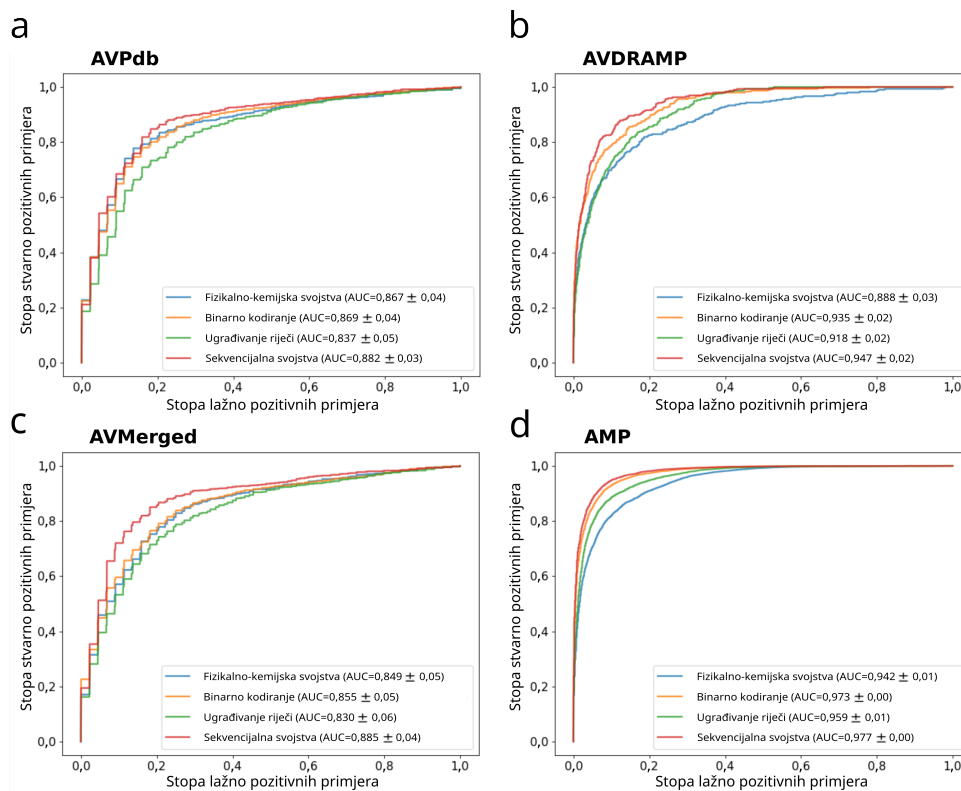
Opisanim postupkom vrednovanja na 10 preklopa vanjske petlje unakrsne validacije i 2 ponavljanja, dolazi se do 20 vrijednosti koje predstavljaju kvalitetu modela. Vrijednosti se tada usrednjuju, čime se dobiva prosječna kvaliteta za par shema predstave - skup podataka. Za usporedbu kvalitete modela, a time i shema predstave, korišteno je 6 metrika koje daju detaljan uvid u točnost modela čak i u slučajevima neuravnoteženih klasa: preciznost (eng. precision), odziv (eng. recall), metrika F1 [42], ROC-AUC [72], Matthewsov koeficijent korelacije (eng. Matthews correlation coefficient; MCC) [42] i geometrijska sredina (eng. geometric mean; GM) [133]. Postojanje statistički značajne razlike između performansi modela naučenog shemom sekvencijalnih svojstava i preostalih modela ispitano je neparametarskim Friedmanovim testom s razinom značajnosti od 0,01 [78]. Ako se utvrdi postojanje razlike, daljnja analiza Wilcoxonovim testom rangova (eng. Wilcoxon signed-ranks test) [44] s razinom značajnosti od 0,003 prilagođenom Bon-

ferroni korekcijom [22] provodi se u cilju utvrđivanja konkretnih parova shema predstave čije se performanse značajno razlikuju.

2.2. Rezultati i diskusija

Izlaz modela povezujemo s antimikrobnom/antiviralnom aktivnošću ili njenim izostankom korištenjem klasifikacijskog praga. Ako je izlaz modela veći od praga, tada ispitanu sekvencu pridodjeljujemo pozitivnoj klasi, odnosno smatramo da sekvenca posjeduje antimikrobnu/antiviralnu aktivnost. U suprotnom, sekvenca pripada negativnoj klasi. Krivulja karakteristične funkcije prijemnika variranjem praga opisuje kako se modeli ponašaju u ovisnosti o odabiru klasifikacijskog praga. Idealna krivulja ROC prolazi gornjim lijevim uglom grafa, odnosno ima stopu stvarno pozitivnih primjera (eng. true positive rate) 100%, stopu lažno pozitivnih primjera (eng. false positive rate) od 0%, te površinu ispod krivulje (ROC-AUC) od 1,0. Za preostale metrike vrednovanja, odabrani klasifikacijski prag jest 0,5.

Schema sekvencijalnih svojstava postigla je najviše vrijednosti ROC-AUC u sva četiri ispitna slučaja (slika 2.7), što indicira kako sadrži više informacija u odnosu na ostale načine predstave peptida. Također, prikazuje i robusnost same klasifikacije, pri čemu visoke vrijednosti AUC u ovom slučaju ukazuju na to da model nije samo točno klasificirao sekvencu, već je to učinio s visokom pouzdanošću. Slični zaključci mogu se donijeti i pregledom histograma vrijednosti F1 za provedene eksperimente (slika 2.8). Schema sekvencijalnih svojstava u svim ispitnim slučajevima daje najbolje rezultate. S obzirom na to da metrika F1 ne gleda puku točnost modela, već vrednuje i broj točno i krivo klasificiranih primjera po klasama, i ovdje je vidljiva robusnost i generalizacijska sposobnost klasifikatora temeljnog na shemi sekvencijalnih svojstava. Usporedbom vrijednosti ROC-AUC ostalih shema predstave, može se primijetiti da su modeli koji su imali informaciju o poretku aminokiselina (binarno kodiranje, ugrađivanje riječi, sekvencijalna svojstva) ostvarili bolje performanse u odnosu na predstavu fizikalno-kemijskim svojstvima na skupovima podataka AVDRAMP i AMP. Predstava fizikalno-kemijskim svojstvima ostvarila je bolje performanse u odnosu na ugrađivanje riječi na skupovima podataka AVPdb i AVMerged, ali i dalje je nešto lošija od binarnog kodiranja i sheme sekvencijalnih svojstava. Rezultati mogu ukazivati na to da, u slučajevima predviđanja antiviralne i antimi-

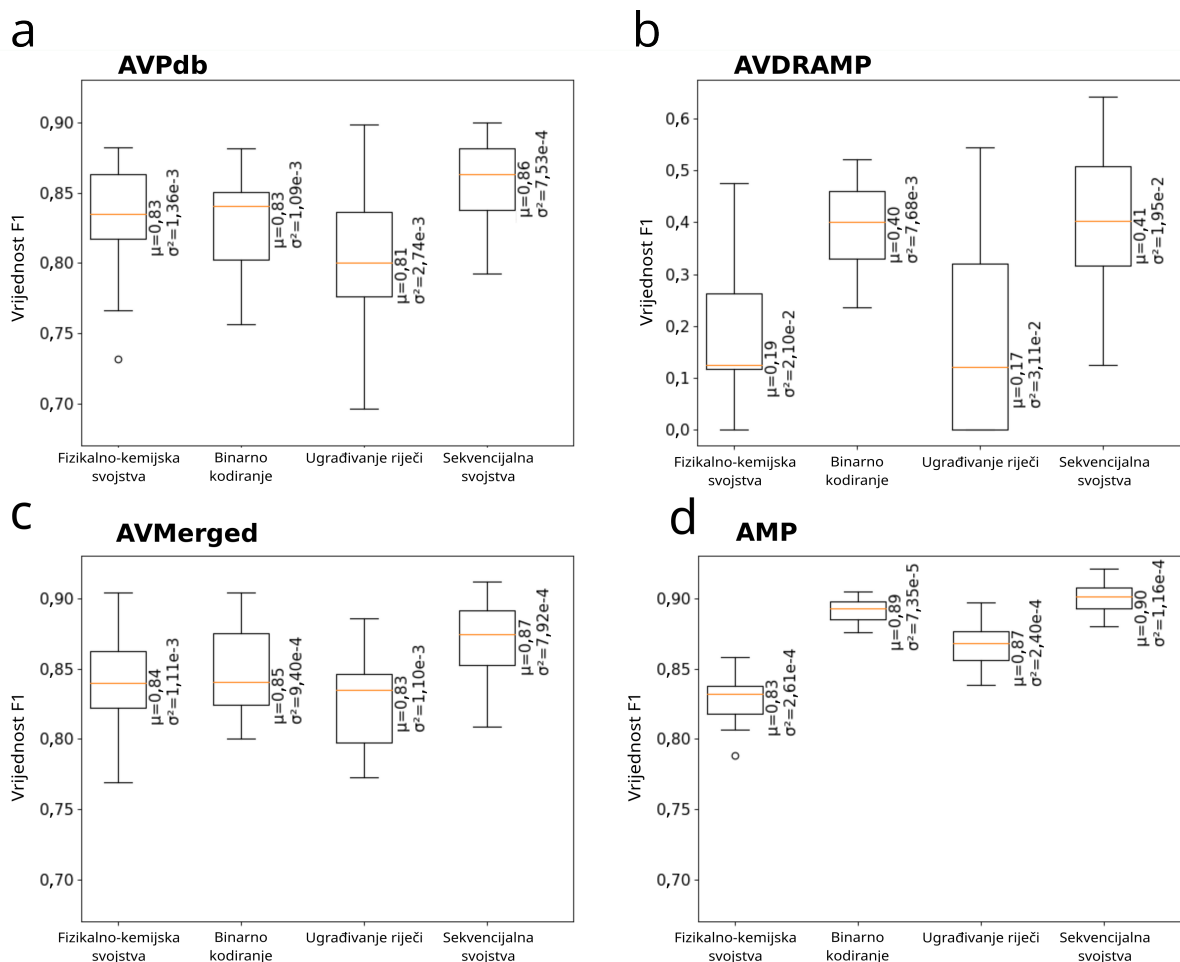


Slika 2.7: Krivulje ROC opisuju ponašanje modela na sva četiri skupa podataka: a) AVPdb b) AVDRAMP c) AVMerged d) AMP. Za svaku krivulju prikazana je srednja vrijednost i standardna devijacija. Preuzeto iz [197] i prevedeno.

krobne aktivnosti, raspored aminokiselina predstavlja informativniju značajku u odnosu na fizikalno-kemijska svojstva. Također, možemo zaključiti da su sheme binarnog kodiranja i sekvencijalnih svojstava ostvarile bolje rezultate u odnosu na shemu ugrađivanja riječi u provedenim ispitnim slučajevima.

Analiza statistički značajnih razlika među performansama modela prikaza je u tablici 2.4. Friedmanov test primijenjen na rezultate iskazane vrijednošću ROC-AUC pronašao je značajnu razliku na svakom skupu podataka, odnosno u 23 od 24 ispitna slučaja. Post-hoc analiza Wilcoxonovim testom utvrdila je značajne razlike između sheme sekvencijalnih svojstava i:

- svih ostalih načina predstave peptida u slučaju skupova podataka AVMerged i AMP,
- fizikalno-kemijskih svojstava i ugrađivanja riječi u slučaju skupa podataka AVDRAMP,
- ugrađivanja riječi u slučaju skupa podataka AVPdb.



Slika 2.8: Histogramima je prikazana razdioba vrijednosti F1 koje su modeli ostvarili na sva četiri skupa podataka: a) AVPdb b) AVDRAMP c) AVMerged d) AMP. Vrijednosti medijana prikazane su narančastim linijama, a srednje vrijednosti i standardne devijacije simbolima μ i σ^2 . Preuzeto iz [197] i prevedeno.

Analizom je utvrđeno da je shema sekvencijalnih svojstava ukupno pokazala bolje performanse u 55 od 72 usporedbe, dok u preostalih 17 slučajeva nije pronađena statistički značajna razlika između sheme sekvencijalnih svojstava i ostalih shema predstave (označeno zvjezdicom u tablici). U određenim slučajevima, modeli nisu predvidjeli niti jedan pozitivan primjer na podskupu za testiranje, pa metrika preciznosti nije mogla biti izračunata (slučajevi označeni tildom u tablici). U tim slučajevima, metrika preciznosti ostavljena je kao 0, što nužno ne održava stvarnu preciznost modela.

Slijedom navedenog, može se zaključiti da shema sekvencijalnih svojstava uspješno odgovara na nedostatke koji postoje u shemama za predstavu peptida u računalu. Spajanjem informacija o fizikalno-kemijskim svojstvima i poretku aminokiseline dobiva se sveobuhvatnija informacija o peptidima koja predstavlja odličan temelj za različite modele

Tablica 2.4: Rezultati vrednovanja na svim skupovima podataka; najbolje vrijednosti su podebljane. Vrijednosti metrika kod kojih nije pronađena statistički značajna razlika u odnosu na istu metriku sheme sekvencijalnih svojstava, označene su zvjezdicom. U određenim slučajevima, preciznost nije mogla biti pouzdano izračunata jer model na podskupu za testiranje nije predvidio niti jednu pozitivnu sekvencu, pa su takvi slučajevi označeni znakom tilde. Preuzeto iz [197] i prevedeno.

	Metrika F1	MCC	GM	Odziv	Preciznost	ROC-AUC	Skup podataka
Fizikalno-kemijska svojstva	0,833	0,588	0,782	0,870*	0,800	0,867*	AVPdb
Binarno kodiranje	0,829	0,610	0,805	0,814	0,845*	0,869*	
Ugrađivanje riječi	0,806	0,547	0,772	0,804	0,808	0,837	
<i>Sekvencijalna svojstva</i>	0,858*	0,671*	0,834*	0,855*	0,863*	0,882*	
Friedmanova p-vrijednost	2,55e-5	3,56e-5	5,34e-5	3,1e-6	1,73e-6	3,52e-4	
Fizikalno-kemijska svojstva	0,188	0,256	0,293	0,117	0,643*,~	0,888	AVDRAMP
Birano kodiranje	0,396*	0,440*	0,524*	0,282*	0,743*	0,935*	
Ugrađivanje riječi	0,168	0,205	0,237	0,108	0,427*,~	0,918	
<i>Sekvencijalna svojstva</i>	0,413*	0,440*	0,550*	0,322*	0,662*	0,947*	
Friedmanova p-vrijednost	1,5e-4	1,09e-3	6,94e-5	1,59e-5	4,05e-2	5,12e-8	
Fizikalno-kemijska svojstva	0,838	0,547	0,749	0,885*	0,797	0,849	AVMerged
Birano kodiranje	0,847*	0,577	0,766	0,887*	0,813	0,855	
Ugrađivanje riječi	0,827	0,502	0,711	0,889*	0,775	0,830	
<i>Sekvencijalna svojstva</i>	0,869*	0,652*	0,817*	0,885*	0,856*	0,885*	
Friedmanova p-vrijednost	9,63e-6	2,7e-7	3,62e-7	0,82	5,59e-8	8,48e-8	
Fizikalno-kemijska svojstva	0,828	0,723	0,855	0,801	0,856	0,942	AMP
Birano kodiranje	0,891	0,822	0,909	0,885*	0,898	0,973	
Ugrađivanje riječi	0,866	0,782	0,887	0,849	0,883	0,959	
<i>Sekvencijalna svojstva</i>	0,901*	0,839*	0,917*	0,890*	0,913*	0,977*	
Friedmanova p-vrijednost	2,64e-11	2,64e-11	3,87e-11	1,58e-9	2,66e-9	1,38e-11	

strojnog učenja. Kako se virtualni probir sve više koristi za inicijalni odabir molekula-kandidata prije daljnje optimizacije i eksperimentalnih ispitivanja, izgledno je da će ova shema u budućnosti pridonijeti povećanju točnosti *in silico* metoda. Shema je dostupna kao knjižnica funkcija u repozitoriju PyPI: <https://pypi.org/project/seqprops>.

3. EVOLUCIJSKI PRISTUP ISTRAŽIVANJU KATALITIČKIH PEPTIDA

Rezultati istraživanja opisani u ovom poglavlju objavljeni su u sljedećem znanstvenom radu: G. Mauša, M. Njirjak, E. Otović, and D. Kalafatovic. Configurable soft computing-based generative model: The search for catalytic peptides. MRS Advances, 8(19):1068–1074, 2023. [172].

Enzimi su ključni dionici raznih bioloških procesa, kao što su razgradnja laktoze [229], preveniranje oksidativnog stresa [251], sinteza RNA [25], dok novije primjene uključuju biodegradaciju polimera plastike [31], saniranje onečišćenja [218], biološke senzore [64], i slično. Prirodna evolucija sve teže prati korak s ubrzanim napretkom ljudske civilizacije, pa se u novije vrijeme za odgovaranje na različite izazove sve više proučava upotreba umjetnih, laboratorijski stvorenih enzima [182]. Enzimi su pretežno proteini prilagođeni djelovanju u specifičnim biološkim uvjetima i vrlo su osjetljivi na okolišne čimbenike, kao što su temperatura i pH vrijednost [20], a jedan način kako se tome može doskočiti jesu katalitički peptidi [298]. Katalitički peptidi su manji od proteina, lakše ih je sintetizirati, mogu biti stabilniji u odnosu na okolišne čimbenike, te su zahvalniji za računalno proučavanje s obzirom na to da je kraće sekvence lakše računalno karakterizirati i simulirati [167].

Modeli dubokog učenja pokazali su se vrlo uspješnima u predviđanju karakteristika peptida i generiranju novih sekvenci sa specifičnim svojstvima [41, 175]. Algoritmi poput VAE [58, 59], povratnih neuronskih mreža [181], i GAN [155] ubrzali su otkrivanje antitumorskih peptida [53, 256], funkcionalnih varijanata proteina [100, 125], lijekova temeljenih

na peptidima [181, 221], i malih molekula [16]. Generativni modeli mogu prilikom procesa učenja izdvojiti i naučiti karakteristike određene kategorije peptida, te posljedično stvarati nove peptide uzorkovanjem iz navedene razdiobe. Međutim, modele poput VAE i GAN karakterizira izostanak objašnjivosti i prilagodljivosti, što predstavlja izazov i svojevrsni nedostatak.

Ovo istraživanje predstavlja prvu studiju koja se bavi računalnom karakterizacijom i otkrivanjem katalitičkih peptida koristeći metode strojnog učenja i evolucijskih algoritama. Najprije je razvijen model RNN naučen na sekvencama s potvrđenom katalitičkom aktivnošću hidrolize estera. Potom je navedeni model upotrijebljen kao funkcija dobrote u genetskom algoritmu čiji je cilj optimizacijom pronaći nove katalitičke peptide.

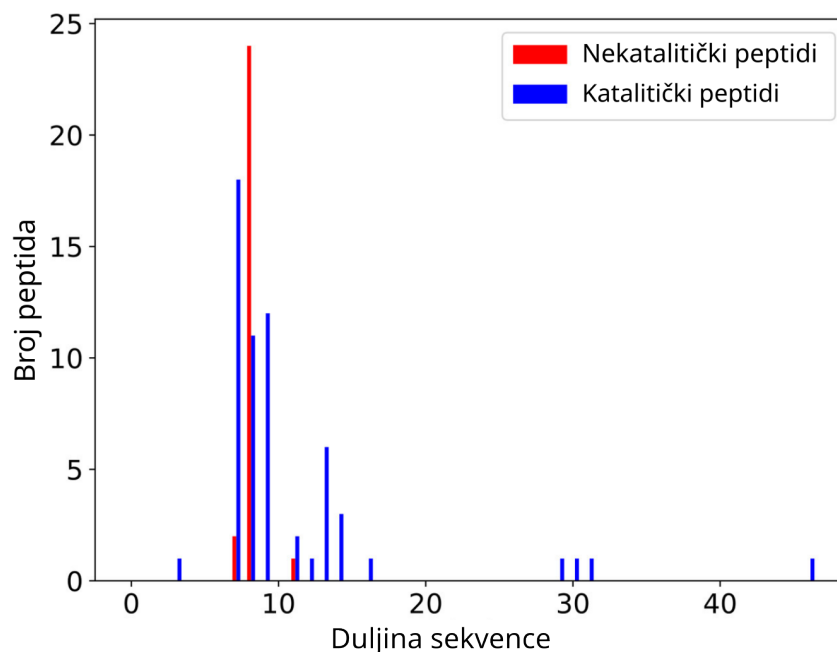
3.1. Metodologija

3.1.1. Skup podataka

Za učenje i procjenu točnosti modela strojnog učenja korišten je skup podataka eksperimentalno validiranih katalitičkih peptida iz literature, testiranih za hidrolizu para-nitrofenil acetata [114]. S obzirom na to da su neke sekvence ispitane više puta kako bi se povećala točnost rezultata, skup podataka može sadržavati višestruke unose pojedinih sekvenci. Posljedično, pripremom podataka uklonjeni su duplikati, te su sekvence podijeljene na katalitičke peptide koji predstavljaju pozitivnu klasu i one kod kojih katalitička aktivnost izostaje i predstavljaju negativnu klasu, sukladno evidentiranom mehanizmu u tablici podataka. Nakon obrade, korišteni skup podataka sadržavao je 56 pozitivnih i 29 negativnih sekvenci. Histogram duljina peptida u izvornom skupu podataka podijeljen po klasama prikazan na slici 3.1 ukazuje na koncentraciju sekvenci oko duljine od 8 aminokiselina.

3.1.2. Model predviđanja

Model predviđanja temeljen je na arhitekturi RNN, pri čemu je ulaz u model peptid predstavljen pomoću sheme sekvencijalnih svojstava (poglavlje 2.). Nakon ulaza, slijedi dvosmjerni sloj LSTM od 80 elemenata čiji je zadatak obuhvatiti sekvencijalne zavisnosti prisutne u podacima, odnosno u rasporedu aminokiselina u sekvenci. Slijede potpuno po-



Slika 3.1: Histogram broja pojavljivanja određenih duljina katalitičkih peptida (označeni plavo) i peptida kod kojih katalitička aktivnost izostaje (označeni crveno) u korištenom skupu podataka. Preuzeto iz [114] i prevedeno.

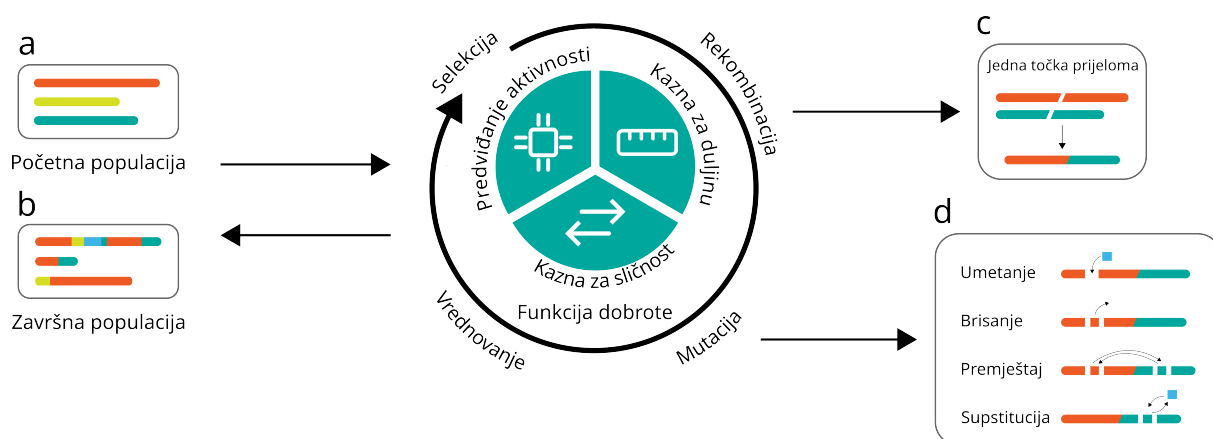
vezani slojevi neurona čiji se broj određuje prilikom optimizacije hiperparametara. Izlaz modela je sigmoidni neuron čija vrijednost odgovara procijenjenoj vjerojatnosti da ulazni peptid posjeduje katalitičku aktivnost. Model je naučen koristeći stratificiranu, dvaput ponovljenu unakrsnu validaciju u 10 preklopa, unutar koje je prisutna ugniježđena unakrsna validacija u 5 preklopa za optimizaciju hiperparametara s postupkom koračajnog odabira značajki. Učenje se provodilo do maksimalno 200 epoha, pri čemu je omogućen raniji završetak u slučaju stagnacije vrijednosti gubitka na validacijskim preklopima, te se koristio optimizator Adam sa stopom učenja od 10^{-4} . Sprječavanje prekomjerne naučenosti provedeno je slojem za regularizaciju izbacivanjem neurona s postotkom izbacivanja od 30%. Kako bi se učenje ubrzalo, korištena je veličina serije od 32 peptida. Model je vrednovan koristeći standardnu metriku točnosti klasifikacije, ali i vrijednost F1, MCC, geometrijsku sredinu između stope stvarno pozitivnih i stope stvarno negativnih primjera (eng. true negative rate), i ROC-AUC, kako bi se sveobuhvatno procijenile njegove performanse.

3.1.3. Evolucijski generativni model

Prethodno razvijeni model predviđanja uvršten je u genetski algoritam kao funkcija dobrote za procjenu katalitičke aktivnosti peptida. Algoritam simulira evoluciju pep-

tida predstavljenih zapisom FASTA, gdje je svaka aminokiselina označena jedinstvenim slovom [161], nastojeći pronaći sekvence sa što višom vjerojatnošću za katalitičku aktivnost. To čini počevši od populacije peptida nasumične duljine i sastava (slika 3.2.a), te primjenom iterativne optimizacije koja uključuje procjenu kvalitete peptida modelom predviđanja, rekombinaciju sekvenci, i mutaciju. Završna populacija tada sadržava visoko optimizirane sekvence s vrlo visokom vjerojatnošću za ispoljavanje katalitičke aktivnosti (slika 3.2.b).

Veličina populacije drži se konstantom nakon svake iteracije/generacije i iznosi 50. Broj aminokiselina koje sačinjavaju peptid u početnoj populaciji ograničen je na raspon [3, 50] kako bi se istovremeno osigurala raznolikost i spriječio prelazak u domenu proteina. Za odabir roditelja korištena je turnirska selekcija veličine 3, a pri samoj rekombinaciji upotrijebljena je jedna točka prijeloma (slika 3.2.c). Mutacija sekvence događa se s vjerojatnošću od 5%, te se mogu pojaviti 4 vrste mutacija (slika 3.2.d): umetanje, brisanje, premještanje i supstitucija, svaka s vjerojatnošću od 25%. Umetanje postavlja nasumičnu aminokiselinu na nasumično mjesto u peptidu, brisanje čini suprotno, odnosno briše nasumično odabranu aminokiselinu iz sekvence, premještanje mijenja mjesta dviju nasumično odabranih aminokiselina u sekvenci, dok supstitucija mijenja nasumičnu aminokiselinu novom. U svakoj od 30 optimizacijskih iteracija stvara se 30 novih peptida.



Slika 3.2: Shematski prikaz načina rada genetskog algoritma. Nasumična početna populacija podvrgava se ciklusima selekcije, rekombinacije i mutacije fiksni broj iteracija. Krajnja populacija optimizirana je prema kriterijima korištenim u vrednovanju, u ovom slučaju primarno prema predviđenoj vjerojatnosti za katalitičku aktivnost, a sekundarno prema duljini i sličnosti. Preuzeto iz [172] i prevedeno.

S obzirom na to da genetski algoritmi imaju tendenciju konvergencije u jedno najbolje rješenje, u optimizacijski proces je uvedena kaznena funkcija koja korigira vrijednost kvalitete rješenja sukladno njegovoj sličnosti ostatku populacije. Uzmemo li da P predstavlja skup rješenja u populaciji peptida, i indeks peptida čija se sličnost provjerava, $P_{a,b}$ frekvenciju b -te aminokiseline za a -to rješenje, kazna se računa sukladno jednadžbi 3.1. Tako se kvaliteta jedinki koje nalikuju ostatku populacije snižava, što im posljedično dodjeljuje manju vjerojatnost da budu izabrane kao roditelji ili da uspješno prijeđu u sljedeću generaciju.

$$f(i) = \frac{\left(\sum_{\substack{n=1 \\ n \neq i}}^{|P|} \left(1 - \sum_{j=1}^{20} \frac{|P_{i,j} - P_{n,j}|}{P_{i,j} + P_{n,j}} \right) \right)}{10 \cdot |P|} \quad (3.1)$$

Kraći peptidi mogu se lakše sintetizirati, te ih je jednostavnije simulirati koristeći metode molekularne dinamike. Kako bi se optimizacija usmjerila prema kratkim peptidima, uvedena je dodatna kaznena funkcija koja vrednuje duljinu peptida. Funkcija se koristi za snižavanje funkcije dobrote, a računa se sukladno jednadžbi 3.2, pri čemu L predstavlja broj aminokiselina od kojih se peptid sastoji, a R srednju vrijednost duljine peptida zadanog raspona. Preostale vrijednosti u jednadžbi 3.2 postavljene su tako da svako odudaranje duljine peptida u odnosu na srednju vrijednost zadanog raspona snižava vjerojatnost katalitičke aktivnosti za 5%, s maksimalnim sniženjem od 50%. Provedena su dva ispitna slučaja: slučaj bez željenog raspona duljina, te slučaj u kojem je za željenu duljinu sekvenci odabran raspon $[4, 12]$ koji vjerno obuhvaća najčešće duljine u skupu podataka za učenje modela predviđanja (slika 3.1). Korekcija vrijednosti kvalitete pojedinog rješenja jednadžbom 3.2 odvija se isključivo ako je duljina peptida izvan zadanog raspona.

$$f(L) = \min\{0, 05 \cdot |L - R|; 0, 5\} \quad (3.2)$$

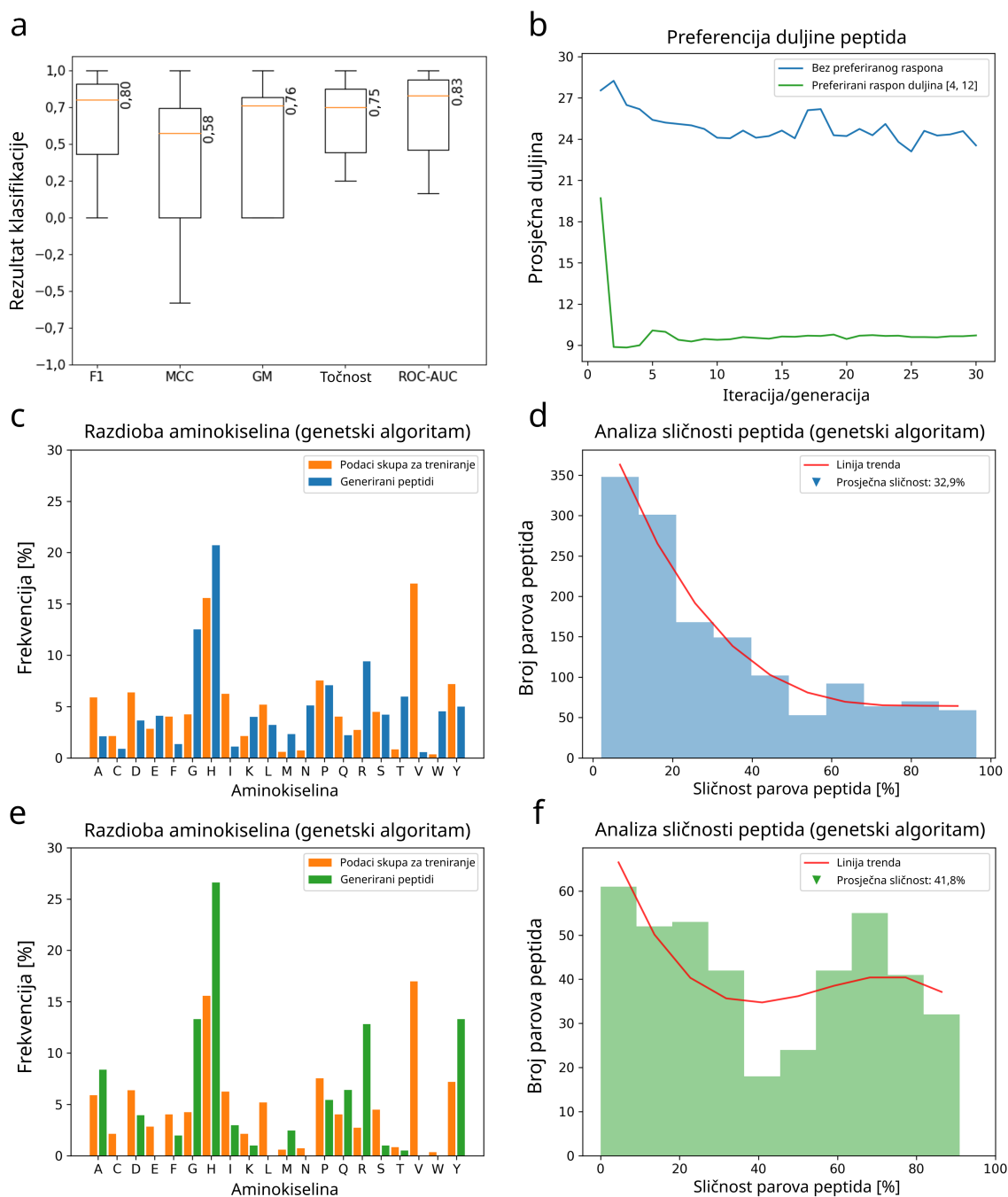
Generativni model trebao bi stvarati nove, raznolike sekvence među kojima se ne ponavljaju one koje su korištene za učenje modela predviđanja. Sukladno tome, napravljena je analiza čiji je cilj bio utvrditi koliko se peptidi međusobno razlikuju, te koliko se razlikuju u odnosu na sekvence za učenje. Kako je manjak metrika za procjenu jedinstvenosti peptida

već prepoznat kao neodgovoreno istraživačko pitanje [267], u ovom istraživanju predložili smo dvije metrike za analizu: usporedbu razdioba aminokiselina i procjenu sličnosti peptida. Kod razdiobe aminokiselina uspoređuju se relativne frekvencije pojavljivanja svih aminokiselina u skupu podataka za učenje, te u skupu podataka novogeneriranih peptida. S druge strane, sličnost peptida procjenjuje se isključivo na skupu generiranih peptida, brojanjem identičnih aminokiselina na odgovarajućim pozicijama unutar sekvence nakon globalnog poravnanja svih parova peptida algoritmom Needleman-Wunsch [186]. U procjeni sličnosti korištene su isključivo generirane sekvence s pripadnom vjerojatnošću za katalitičku aktivnost većom od 80%.

3.2. Rezultati i diskusija

Prag za pretvorbu vjerojatnosti modela predviđanja u kategoriju peptida dobiven je procedurom koja maksimizira vrijednost MCC i iznosi 0,55 [197]. Za sve peptide s vjerojatnošću iznad navedene vrijednosti smatramo da posjeduju katalitičku aktivnost, dok za preostale vrijedi suprotno. Iako se skup podataka za učenje i vrednovanje modela predviđanja sastojao od samo 85 peptida, model predviđanja postigao je vrlo dobre performanse, s medijanom točnosti klasifikacije od 75% (slika 3.3.a).

Dva ispitna slučaja genetskog algoritma, pri čemu u jednom nije određena duljina peptida, dok je u drugom slučaju željeni raspon duljina [4, 12], dala su vrijedne rezultate. Vidljivo je da kaznena funkcija uspješno usmjerava algoritam prema željenom rasponu duljina (slika 3.3.b), a zanimljivo je primijetiti sklonost algoritma ka stvaranju sekvenci prosječne duljine veće od 24 u slučaju kada kaznena funkcija za duljinu nije primijenjena. To predstavlja odstupanje od očekivanog s obzirom na to da medijan duljine peptida u skupu podataka za učenje iznosi 8. Navedeno može biti rezultat inicijalizacijskog procesa algoritma koji stvara sekvence nasumične duljine i sastava, no također može biti indikator kvalitetno naučenog modela predviđanja. Naime, ako je model predviđanja izveo obrasce iz podataka za učenje koji su općenito primjenjivi, tada genetski algoritam ima manju tendenciju stvaranja sekvenci sličnih onima u skupu podataka za učenje, već se vodi isključivo generalnim obrascima prisutnima u katalitičkim peptidima. U slučaju bez postavljenog raspona duljine, genetski algoritam favorizira glicin (G), arginin (R) i histidin (H), a minimizira uporabu valina (V) i izoleucina (I), što ne korelira s razdiobom



Slika 3.3: Vrednovanje generativnog modela koji koristi model predviđanja temeljen na arhitekturi RNN. (a) Rezultati vrednovanja i sve metrike modela predviđanja (medijan je prikazan narančastom linijom). (b) Prosječna duljina peptida u populaciji genetskog algoritma kroz generacije bez postavljene željene duljine (plavo) te s postavljenom željenom duljinom (zeleno). Prikazana je i usporedba razdioba aminokiselina u skupu podataka za učenje i (c) generiranih peptida bez postavljene željene duljine, te podataka za učenje i (e) generiranih peptida s postavljenom željenom duljinom. Analiza sličnosti također je provedena odvojeno za (d) slučaj bez postavljene željene duljine, te za (f) slučaj s postavljenim željenim rasponom duljina. Preuzeto iz [172] i prevedeno.

u podacima za učenje (slika 3.3.c). U slučaju s postavljenim rasponom duljine, algoritam počinje povećano favorizirati tirozin (Y), a minimizirati asparagin (N), triptofan (W) i glutaminsku kiselinu (E) (slika 3.3.e). Analiza sličnosti provedena je za 420 parova peptida u slučaju bez postavljenog raspona duljine (slika 3.3.d), te 1406 parova peptida u slučaju s postavljenim rasponom duljine (slika 3.3.f). S prosječnom sličnošću od 33% i 42%, možemo zaključiti da je većina generiranih peptida raznolika i ne podliježe nužno standardnim obrascima stvaranja katalitičkih molekula, gdje se postojeće aktivne sekvence koriste kao polazište za stvaranje novih. Popis najboljih sekvenci s pripadajućom predviđenom vjerojatnošću katalitičke aktivnosti prikazan je u tablici 3.1.

Tablica 3.1: Najboljih 15 peptida sukladno predviđenoj vjerojatnosti katalitičke aktivnosti za oba ispitna slučaja. Preuzeto iz [172] i prevedeno.

Željeni raspon duljina ^a		Bez postavljenog raspona duljina	
Peptid	Predviđena vjerojatnost	Peptid	Predviđena vjerojatnost
HRRRGGYIQ	98,1%	HHGMHPHLWHGPPPGTGWPKR	98,7%
HHGRGYIQ	98,1%	HHTRHLAQPWHGTGTGTWCPCGKR	98,7%
HHGRGYYYQ	98,0%	HHRGHPHLWHGTGTGTWCPCGKR	98,7%
HRGRRGYMYP	98,0%	HHGMHPHLAQPPPGTGWPKR	98,7%
HHHHHAADQ	98,0%	HHRGKYFFHKRYPPPGTGWPKR	98,6%
HHHHHAFADQ	97,9%	HHTRHLAQPWHGTGTGTWCPCGKR	98,6%
HHQHGHKFADQ	97,9%	HHGMHPHLWHGTGTGTWCPCGKR	98,6%
HHAHHAADQ	97,9%	HHTRMKPHLWHGTGTGTWCPCGKR	98,6%
HRRRRGYMYP	97,9%	HRRGMHPHLWHGTGTGTWCPCGKR	98,6%
HRGGRGYMYP	97,9%	HHTRHLAQPWHGTGTGTWCPSAR	98,6%
HHAHHAADQ	97,9%	HHGMHPHLWHGTGTGGTGWPGKR	98,6%
HRGRGYMYP	97,8%	HHGMHPHLAQPPPKFFDKPKR	98,6%
HHGRGYIYP	97,8%	HRRRGPHLWHGTGTGTWCPCGKR	98,6%
HHQHGHFADQ	97,8%	HRRGWLGREFGTGTVPGTGWPKR	98,5%
HHQHGHKIYQ	97,7%	HRRGWLGREFGTGTVPGTGWPKR	98,5%

^aŽeljeni raspon duljina postavljen je između 4 i 12.

Razvijena računalna metoda fleksibilna je na promjene i lako se može iskoristiti za stvaranje peptida drugih kategorija, kao što su antimikrobni, peptidi koji prolaze krvno-moždanu barijeru, samosastavljajući peptidi, i slično, dok stohastička priroda genetskog algoritma i implementirane kaznene funkcije omogućuju dobivanje raznolikih sekvenci i sklonost prema proizvoljno odabranim duljinama. S obzirom na to da je postupak istraživanja kemijskog prostora u algoritmu vođen isključivo kapacitetima predviđanja modela strojnog učenja, u velikoj mjeri je uklonjena pristranost prema određenom dijelu kemij-

skog prostora, što omogućuje istraživanje peptida izvan tradicionalnih predložaka koji se koriste za pojedine kategorije; primjerice, ravnoteža hidrofilnosti i hidrofobnosti za peptide sa svojstvom samosastavljanja [170]. Prethodni uspjesi i dodijeljena Nobelova nagrada za istraživanje enzima primjenom koncepata prirodne evolucije [9, 118, 220] dodatno potvrđuju da metodologija razvijena u ovom istraživanju uvelike može pomoći u otkrivanju novih katalitičkih peptida.

4. GENERATIVNI MODEL ZA STVARANJE NOVIH PEPTIDA SA SVOJSTVOM SAMOSASTAVLJANJA

Rezultati istraživanja opisani u ovom poglavlju objavljeni su u sljedećem znanstvenom radu: M. Njirjak, L. Žužić, M. Babić, P. Janković, E. Otović, D. Kalafatovic, and G. Mauša. Reshaping the discovery of self-assembling peptides with generative AI guided by hybrid deep learning. Nature Machine Intelligence, 6(12):1487–1500, 2024. [191].

Međumolekularno samosastavljanje molekula u nanostrukture predstavlja jedan od osnovnih procesa koji se događa u živim bićima [115, 137]. Molekule sa svojstvom samosastavljanja, vođene nekovalentnim interakcijama poput van der Waalsovih sila, spontano se slažu u pravilne strukture, uključujući alfa zavojnicu i beta-ploču [7]. Važnost takvih molekula proizlazi iz njihovih trodimenzionalnih struktura, koje utječu na svojstva kao što su kataliza kemijskih reakcija, molekularno prepoznavanje, i slično [40, 138, 150, 228]. Sukladno tome, navedeni spojevi mogu se upotrijebiti kao baza za razvoj novih nanomaterijala i umjetnih enzima.

U praksi, peptidi su se pokazali vrlo dobrima za razvoj novih molekula sa svojstvom samosastavljanja [76]. Njihova složenost i veći broj stupnjeva slobode u formiranju sekundarne strukture daje im veću mogućnost stvaranja novih nanostrukture u odnosu na male molekule, dok ih nešto niža složenost u odnosu na proteine čini lakšima za računalnu i eksperimentalnu karakterizaciju, te za usmjeravanje prema određenoj vrsti željene nanostrukture [271]. S druge strane, točno određivanje veze između sekvence, strukture i

funkcije peptida još je uvijek nedovoljno istraženo. Zbog toga, eksperimentalno otkrivanje novih peptida sa svojstvom samosastavljanja i dalje predstavlja vodeći pristup u dizajnu takvog tipa molekula [11]. Navedeni pristup može se okarakterizirati kao NP-težak kombinatorni problem gdje mogući prostor pretrage raste eksponencijalno za svaku novu poziciju u peptidu [209]. Uzevši u obzir da sinteza, pročišćavanje i karakterizacija peptida mogu trajati tjednima i zahtijevaju visoko specijaliziranu opremu, stručnjake i znanja [38, 107, 108, 126, 188, 282], iscrpna pretraga prostora peptida koji se sastoje od više od tri aminokiseline postaje gotovo nemogućom [11]. Zbog toga su osmišljene tehnike čiji je cilj drastično smanjiti prostor pretrage. Primjenom određenih molekularnih obrazaca dio sekvence peptida može se učiniti nepromjenjivim, te se optimizira samo manji broj preostalih aminokiselina, dok se uravnotežavanjem hidrofilnosti i hidrofobnosti nastoji utjecati na njegovu mogućnost samosastavljanja [170, 234]. Iako uspješno smanjuju prostor pretrage i omogućuju iscrpno testiranje složenijih peptida, ovakve procedure uvode neželjenu pristranost prema određenom dijelu kemijskog prostora, što ograničava mogućnost otkrivanja zaista novih sekvenci.

Metode molekularne dinamike uspješno se koriste za računalno simuliranje ponašanja kemijskih spojeva i njihovih međusobnih interakcija [75, 81, 90, 99, 146]. Iako još uvijek dosta ograničene u vidu veličine sustava koje mogu simulirati u određenom vremenu, njihova je upotreba omogućila provedbu virtualnih eksperimenata u računalu. Kako MD predstavlja aproksimaciju stvarnih laboratorijskih eksperimenata i oslanja se na iterativne simulacije ponašanja gradivnih dijelova molekule, atoma ili skupina atoma, neminovno uvodi određene greške u konačne rezultate, razina kojih djelomično ovisi o postavkama simulacije; primjerice razina detalja, vremenski razmak između dvaju trenutaka simulacije, veličina sustava [77, 249]. MD je uspješno primijenjen za procjenu sklonosti agregaciji (eng. aggregation propensity; AP), koja je dobar pokazatelj mogućnosti samosastavljanja, za dipeptide i tripeptide [76, 77]. Primjena sličnog pristupa na tetrapeptide ili ostale skupine peptida bila bi vrlo zahtjevna zbog eksponencijalnog rasta prostora pretraživanja [11, 200, 297].

Optimizacija i dizajn peptida pomoću strojnog učenja pokazali su se uspješnom alternativom MD-u [29, 53, 175, 268, 285, 290]. Po brzini predviđanja, ML je značajno brži u odnosu na MD, skraćujući vrijeme do rezultata s nekoliko dana na svega nekoliko sekundi. S druge strane, ML uvodi još jedan sloj aproksimacija, učeći iz dostupnih baza peptida

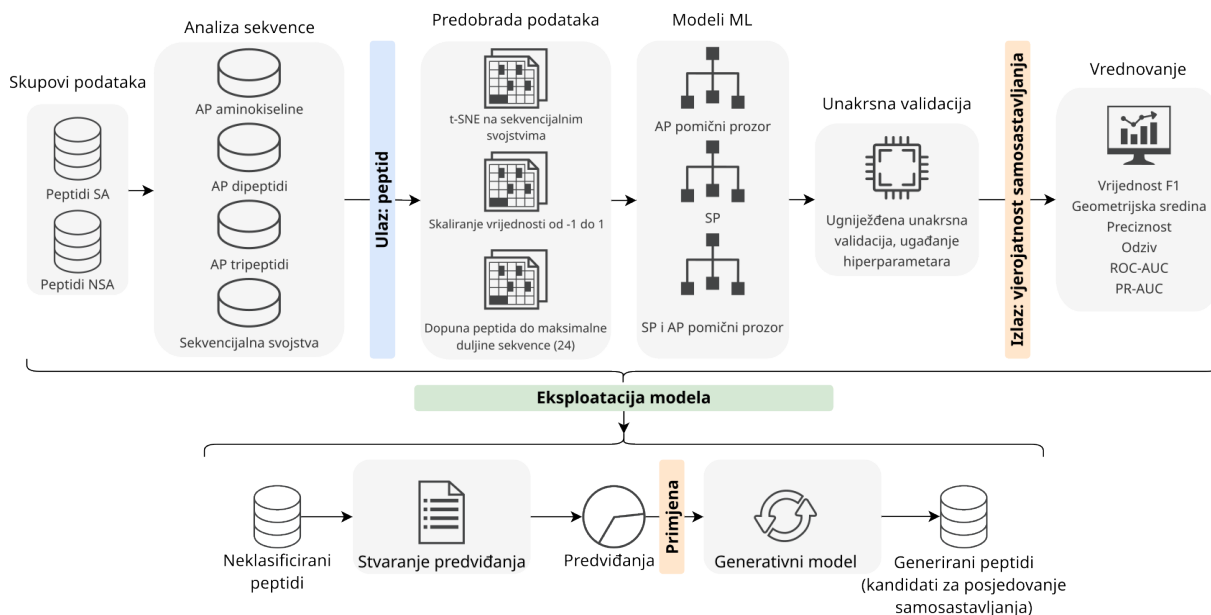
i pokušavajući generalizirati to znanje na previđanje novih sekvenci [97]. Zbog toga, ML može biti manje pouzdana metoda u odnosu na MD, no s druge strane omogućuje brzi probir molekula, čija točnost primarno ovisi o kvaliteti procesa učenja i dostupnim bazama podataka peptida od interesa. Iako se ML već koristio za pretragu novih peptida sa svojstvom samosastavljanja [11, 281], malo dostupnih i eksperimentalno potvrđenih sekvenci, te neravnoteža u klasama, ograničava njihovu uspješnost i primjenu [151]. Povećano istraživanje mogućnosti upotrebe strojnog učenja u područjima kemije i farmacije je očigledno [10, 88], no mnogi se pristupi još uvijek uvelike oslanjaju na MD kao glavnu stavku procesa probira, što ograničava njihovu primjenu na veće skupove molekula kao što je primjerice Enamine REAL knjižnica koja sadrži 10 milijardi molekula [68], ili eksploataciju u evolucijskim generativnim modelima kod kojih se uobičajeno treba provjeriti na tisuće spojeva [23].

U ovom istraživanju predložen je hibridni model povratne neuronske mreže za identifikaciju peptida sa svojstvom samosastavljanja, temeljen na modeliranju kontekstnih veza između gradivnih jedinica sekvence: aminokiselina [56], dipeptida [77], i tripeptida [76] (slika 4.1). Nakon učenja i temeljitog ispitivanja performansi modela, model je korišten kao vodilja genetskog algoritma čiji je cilj bio stvoriti nove peptide s visokom vjerojatnošću za posjedovanje svojstva samosastavljanja.

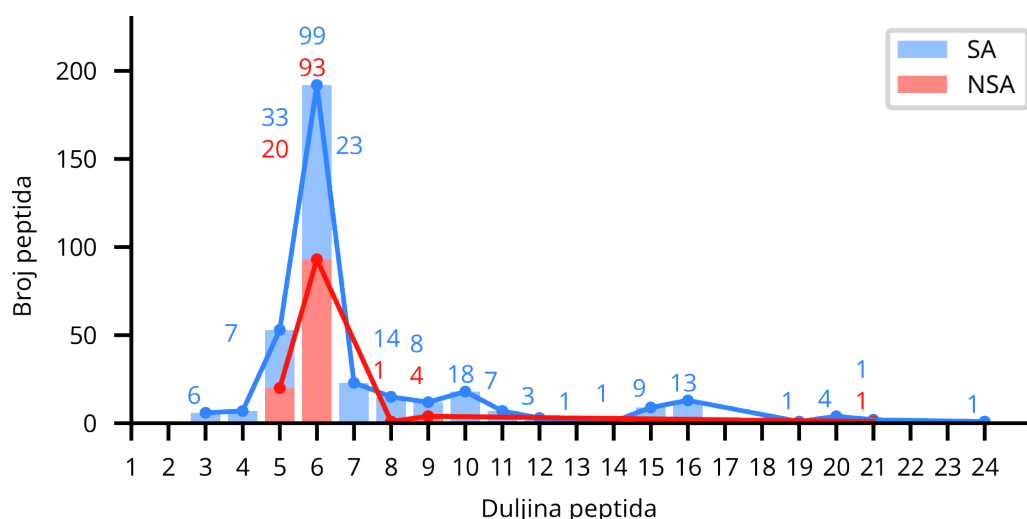
4.1. Metodologija

4.1.1. Skup podataka

Primjena metoda strojnog učenja na područje peptida sa svojstvom samosastavljanja dosad je bilo relativno izazovno s obzirom na nedovoljnu kvalitetu i veličinu dostupnih skupova podataka [297]. Stoga je u ovom istraživanju prikupljeno 368 peptida iz literature s eksperimentalno potvrđenim statusom posjedovanja svojstva samosastavljanja (tablica A.1). Razdiobe primjera po klasama (249 peptida sa svojstvom samosastavljanja, SA; te 119 preostalih peptida, NSA) i duljini peptida prikazane su na slici 4.2. Najveća koncentracija sekvenci je oko duljine od 6 aminokiselina, dok je također zamjetna i nejednaka razdioba po klasama, gdje pozitivna klasa obuhvaća 68% ukupnog skupa podataka.



Slika 4.1: Shematski prikaz metodologije istraživanja. Ulazni podaci modela su vrijednosti sklonosti agregaciji (AP) za aminokiseline, dipeptide, i tripeptide sekvence koja se procesira, uz dodatak izračunatih sekvencijalnih svojstava (SP). Predobrada podataka uključivala je primjenu algoritma t-SNE za smanjenje dimenzionalnosti podataka, skaliranje ulaznih vrijednosti u raspon $[-1, 1]$ za poboljšanje protoka gradijenata kroz mrežu i uspješnije učenje, te nadopunu sekvenci do duljine najveće sekvence u skupu podataka (24 aminokiseline) kako bi se ubrzalo učenje. Modeli su naučeni uz optimizaciju hiperparametara koristeći ugniježđenu unakrsnu validaciju, nakon čega je najbolji model primijenjen u generativnom genetskom algoritmu za dizajn novih peptida. Preuzeto iz [191] i prevedeno.

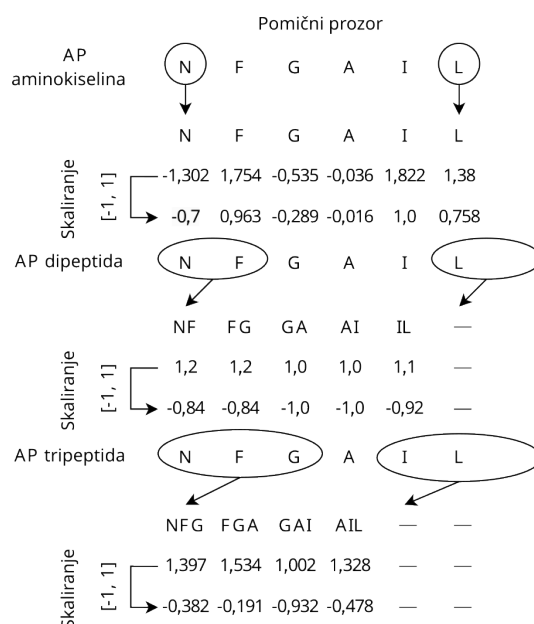


Slika 4.2: Razdioba peptida sa svojstvom samosastavljanja (označeni plavo) i peptida bez svojstva samosastavljanja (označeni crveno) u skupu korištenih podataka, prema duljini. Preuzeto iz [191] i prevedeno.

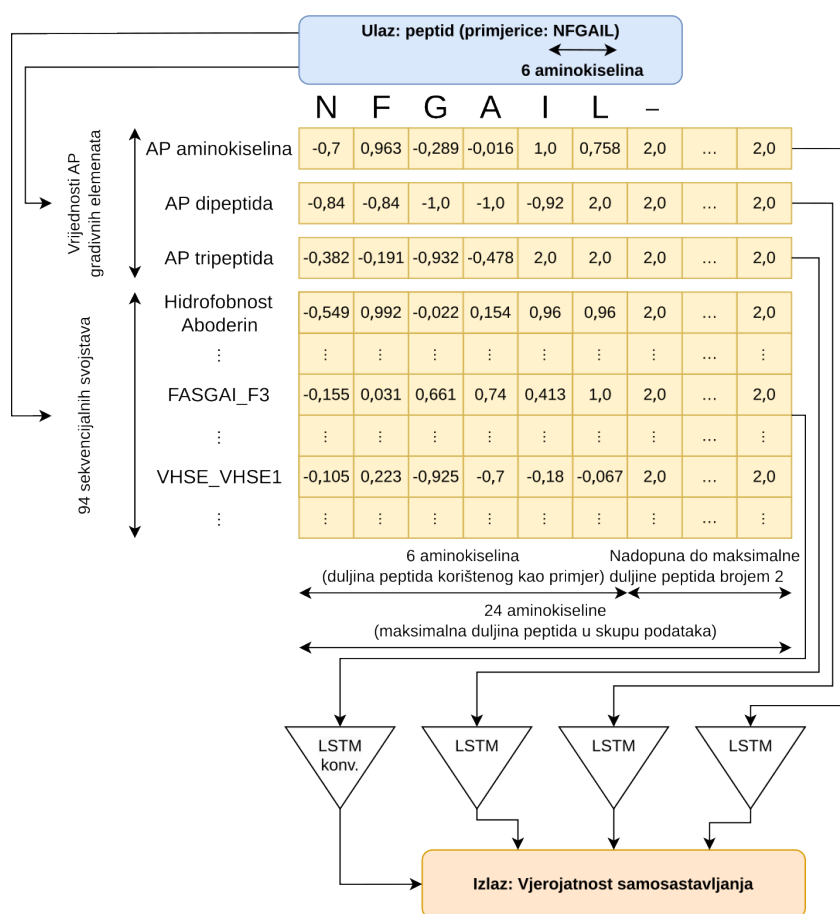
4.1.2. Predobrada podataka i priprema ulaznih vrijednosti modela predviđanja

Prethodne studije potvrdile su AP kao dobar prediktor aktivnosti samosastavljanja [11, 43, 260]. Međutim, računanje vrijednosti AP zahtijeva simulacije molekularne dinamike nakon kojih nam je već indikativno hoće li se peptid samosastaviti u nanos- strukturu, te korištenje tako izračunatih vrijednosti AP u modelu ML, koji bi trebao predvidjeti informaciju koju već znamo, predstavlja redundanciju. U ovom istraživanju problemu smo odlučili pristupiti na drugačiji način. Najprije, peptid od interesa rastavlja se na podskupove: aminokiseline, dipeptide i tripeptide, korištenjem tehnike pomičnog prozora (slika 4.3.a). Kako su agregacijske sklonosti za spomenute gradivne jedinice prethodno izračunate i dostupne u literaturi [56, 76, 77], navedene informacije koristimo kao značajke koje pomažu predvidjeti mogućnost samosastavljanja sekvence u cjelini. Time izbjegavamo nužnost provođenja dugotrajnih simulacija MD za svaku sekvencu čiju aktivnost želimo predvidjeti, što drastično skraćuje vrijeme do rezultata. Kako na mogućnost samosastavljanja utječu i drugi čimbenici, poput fizikalno-kemijskih svojstava peptida i rasporeda aminokiselina [115, 147, 163], uz informacije o vrijednostima AP, kao ulazne podatke odlučili smo postaviti i sekvencijalna svojstva (slika 4.3.b, opisano u poglavlju 2.). Dodatno, kako bismo pospješili protok gradijenata, svaka značajka zasebno skalirana je u raspon $[-1, 1]$ koristeći tehniku min-max skaliranja, a sekvence su također nadopunjene do duljine od 24 aminokiseline, što je maksimalna duljina prisutna u skupu podataka, kako bismo više peptida mogli grupirati u istu seriju i tako ubrzati učenje modela. Za nadopunu sekvenci odabran je broj 2, koji se nalazi izvan definiranog raspona vrijednosti značajki od $[-1, 1]$. Navedena vrijednost maskirana (eng. *masking*) je u svim modelima kako ni na koji način ne bi utjecala na njihovo djelovanje.

a



b

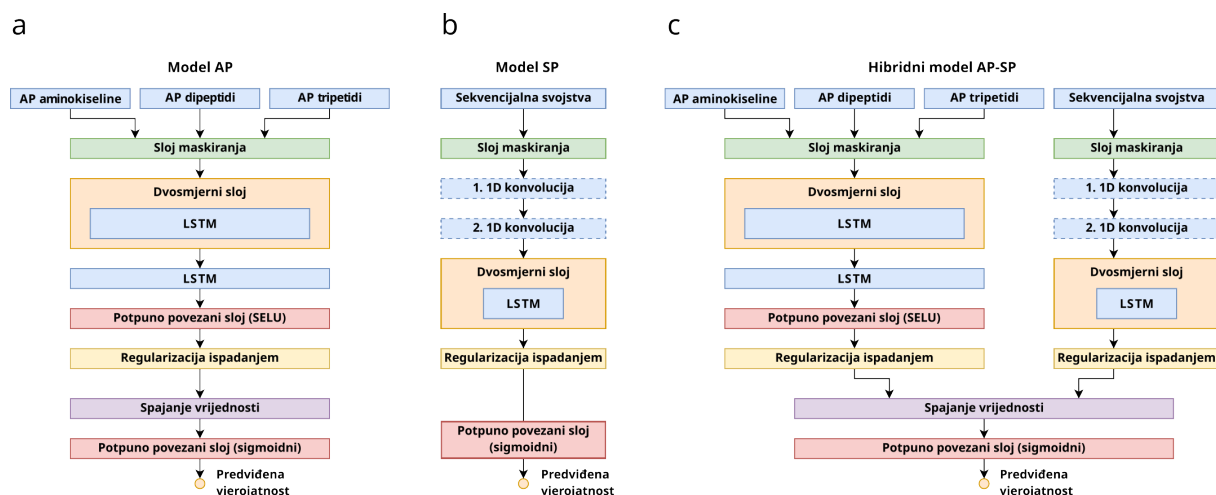


Slika 4.3: (a) Shematski prikaz načina rada tehnike pomičnog prozora za izdvajanje aminokiselina, dipeptida i tripeptida. (b) Prikaz ulaznih podataka za peptid NFGAIL i hibridni model AP-SP. Preuzeto iz [191] i prevedeno.

4.1.3. Modeli predviđanja: učenje i vrednovanje

Inicijalno su razvijene tri neuronske mreže: model koji za predviđanje koristi isključivo informacije o sklonosti agregaciji gradivnih elemenata peptida (model AP, slika 4.4.a), model koji koristi isključivo sekvencijalna svojstva (model SP, slika 4.4.b), te model nastao kombinirajući prethodna dva (hibridni model AP-SP, slika 4.4.c). Sekvencijalna svojstva sadrže 94 značajke za opis svake aminokiseline, što je značajno više u odnosu na tri vrijednosti sklonosti agregaciji koje se dobiju za svaki gradivni element peptida. Smatrali smo stoga da bi takav nejednak raspored u tipovima značajki mogao indirektno favorizirati upotrebu sekvencijalnih svojstava za donošenje odluka u modelima SP i AP-SP. Sukladno tome, a kako bismo izjednačili doprinose značajki, na vrijednosti SP primijenjeno je t-distribuirano stohastičko ugrađivanje u susjedstvo (eng. t-distributed stochastic neighbour embedding; t-SNE) [165], tehnika sažimanja dimenzionalnosti podataka koja se u literaturi već primjenjivala na peptide [58, 273]. Pomoću navedenog, 94 fizikalno-kemijska svojstva svake aminokiseline svedena su na 3 meta-svojstva. Time su stvorena dva dodatna modela: t-SNE SP s istom arhitekturom kao i model SP i smanjenim brojem ulaznih značajki, i t-SNE AP-SP s istom arhitekturom kao i model AP-SP i smanjenim brojem ulaznih značajki sekvencijalnih svojstava. Ukupno, u istraživanju je korišteno 5 modela strojnog učenja. Kao njihov temelj, odabrani su slojevi LSTM, koji su pri prvom pojavljivanju uvijek dvosmjerni. Slojevi LSTM vrlo se dobro nose s problemom iščezavajućih gradijenata [205] i osiguravaju koračajno procesiranje podataka, dok im dvosmjernost omogućava obradu sekvence u oba smjera, od N- prema C-terminusu peptida i obrnuto. Dijelovi neuronskih mreža koji procesiraju sekvencijalna svojstva opcionalno mogu još uključivati i konvolucijske slojeve za pronalaženje obrazaca u značajkama peptida. Svi modeli sadrže slojeve za regularizaciju izbacivanjem/ispadanjem neurona kako bi se spriječila pretjerana prilagodba podacima za učenje. Ovo je izuzetno važno s obzirom na to da se relativno složeni modeli primjenjuju na mali skup podataka. Krajnji izlaz definiran je sigmoidnim neuronom čija vrijednost predstavlja vjerojatnost da ulazni peptid posjeduje mogućnost samosastavljanja.

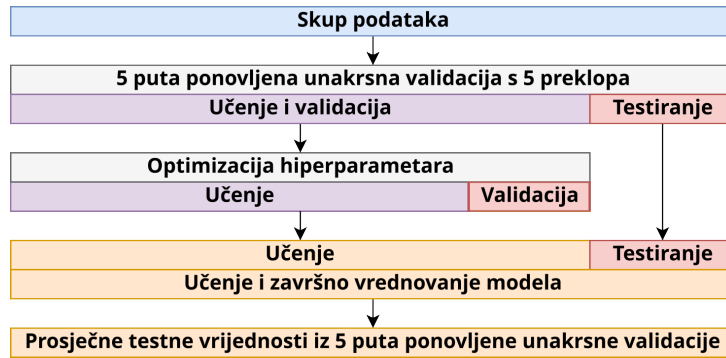
Modeli su učeni maksimalno 70 epoha, pri čemu je veličina serije postavljena na 600, što je osiguralo usrednjavanje gradijenata na cijelom skupu za učenje prije podešavanja težina, odnosno omogućilo brže i stabilnije učenje. Početna stopa učenja postavljena je



Slika 4.4: Arhitekture neuronskih mreža predloženih modela za (a) Model AP koji koristi isključivo informacije o gradivnim elementima peptida, (b) Model SP koji predviđanje formira na temelju vrijednosti sekvencijalnih svojstava, te (c) Hibridni model AP-SP koji kombinira prethodna dva. Preuzeto iz [191] i prevedeno.

na 0,01 i smanjivana 10% u svakoj epohi, počevši od desete. Proces učenja koristio je dvije petlje unakrsne validacije: vanjsku, 5 puta ponovljenu petlju u 5 preklopa za učenje i završno testiranje modela, te unutarnju, ugniježdenu unakrsnu validaciju u 5 preklopa za određivanje optimalnih hiperparametara modela na validacijskim preklopima. Vanjska petlja unakrsne validacije osigurala je stratificiranu podjelu podataka u 4 preklopa za učenje i jedan za testiranje, pri čemu je zadržan originalni odnos peptida SA i NSA od 2:1 (slika 4.2). Tako je u svakoj petlji učenja osigurano 236–237 peptida za učenje, 58–59 za validaciju, te 73–74 za testiranje modela. Procedura je shematski prikazana na slici 4.5. Moguće vrijednosti hiperparametara za broj ćelija u slojevima LSTM, veličinu jezgre konvolucijskih slojeva, i broj neurona u potpuno povezanim slojevima prikazane su u tablici 4.1. U svim potpuno povezanim slojevima, osim u zadnjem, korištena je aktivacijska funkcija SELU [131]. Osim navedenog, postotak izbacivanja neurona u slojevima regularizacije postavljen je na 50%. Uobičajene vrijednosti od 10%, 20% i 30% [243] su također isprobane izvan procedure pretrage hiperparametara, no nisu dale jednako dobre rezultate. U konačnici, hiperparametri koji su dali najniži prosječni gubitak na svim preklopima za validaciju, odabrani su za korištenje u iteraciji vanjske petlje unakrsne validacije.

Za detaljno vrednovanje predloženih modela strojnog učenja korišten je niz metrika čiji je cilj bio nepristrano i potpuno obuhvatiti njihove stvarne performanse. Izračunate



Slika 4.5: Učenje modela izvedeno je pomoću dviju petlji unakrsnih validacija, svaka u pet preklopa. Vanjska petlja obavlja učenje modela i krajnje testiranje na testom preklopu, dok unutarnja petlja provodi validaciju i odabir hiperparametara. Završni rezultati svih preklopa usrednjuju se, te se tako formiraju konačne performanse modela. Preuzeto iz [191] i prevedeno.

Tablica 4.1: Moguće vrijednosti za odabir pri optimizaciji hiperparametara modela. Preuzeto iz [191] i prevedeno.

Vrijednost	AP	SP	AP-SP	t-SNE SP	t-SNE AP-SP
Broj ćelija	-	32, 48, 64	32, 48, 64	32, 48, 64	32, 48, 64
Veličina jezgre	-	4, 6, 8	4, 6, 8	4, 6, 8	4, 6, 8
Broj neurona	64, 96, 128	-	64, 96, 128	-	64, 96, 128

su krivulje preciznosti i odziva (eng. precision-recall curve; PR) i krivulja ROC, te površine ispod njih (PR-AUC, ROC-AUC), točnost klasifikacije, vrijednost F1, i geometrijska sredina između stopa stvarno negativnih i stvarno pozitivnih primjera (gmean). Prilikom optimizacije hiperparametara, procijenjeno je nekoliko mogućih pragova za pretvaranje sigmoidnih izlaza neuronske mreže u klasifikaciju odluku. Kako bismo predložene modele stavili u okvir postojeće literature i provjerili njihovu kvalitetu, modele smo usporedili s modelom nasumične šume, s obzirom na to da je takav tip modela pokazao vrlo dobre performanse u prethodnim studijama [11, 35, 187]. Također, usporedba je napravljena s različitim oblicima neuronskih mreža koje su u svojim arhitekturama uključivale razne povratne slojeve [67], LSTM [103], dvosmjerne slojeve LSTM [226], višeslojni perceptron [144] i transformere [261], kao što je opisano u [162]. Svi modeli preuzeti iz [162] učili su se s početnom stopom učenja od 0,2 i veličinom rječnika za ugrađivanje od 21 (broj proteinogenih aminokiselina i numerička neutralna vrijednost), maksimalnom duljinom peptida od 24, veličinom serije od 1024, te maksimalnim brojem epoha od 100, s iznimkom modela MLP čiji je maksimalan broj epoha bio 50. U slučaju RF modela, korištena je

ugniježđena unakrsna validacija u pet preklopa za optimizaciju hiperparametara u svakoj iteraciji vanjske petlje. Optimizacija je modificirala broj stabala u šumi, birajući između 100, 200, 300, 400 i 500, te kriterij razdvajanja u kojem se koristio Gini indeks [71], entropija [112] ili logaritamski gubitak (eng. log-loss) [264]. Isprobane su maksimalne dubine stabala od 3, 6, 9, 12 i neograničeno mnogo razina, minimalan broj primjera potreban za razdvajanje čvora od 2, 5 i 10, te minimalan broj primjera potreban da se čvor smatra listom od 1, 2 i 4. U specifičnim slučajevima, provedeno je i ispitivanje statističke značajnosti razlika u klasifikaciji između modela pomoću McNemarovog dvostranog testa [173], koristeći razinu značajnosti $\alpha = 0,05$ i 1840 primjera. Pet predloženih modela dodatno je ispitano koristeći 20 eksperimentalno validiranih peptida iz [11], na temelju vrijednosti F1.

4.1.4. Generativni model i vrednovanje predloženih peptida

Predloženi hibridni model AP-SP korišten je kao funkcija dobre genetskog algoritma za stvaranje novih peptida sa svojstvom samosastavljanja. Algoritam je preuzet iz našeg prijašnjeg istraživanja [172] i prilagođen. Počevši s inicijalnom populacijom od 50 peptida nasumičnog sastava i duljine u rasponu od 3 do 24 aminokiseline, algoritam u svakoj od 30 iteracija stvara 30 novih sekvenci. Stvaranju svake sekvence prethodi izbor roditelja turnirskom selekcijom veličine turnira 3, rekombinacija s jednom nasumičnom točkom prijeloma, te 5% vjerojatnosti za primjenu jedne od četiri jednako vjerojatne mutacije: umetanje, brisanje, premještanje i supstitucija (vidjeti potpoglavlje 3.1.3.). Kako bismo stvaranje peptida u jednom ispitnom slučaju ograničili na peptide duljine 6, s obzirom na činjenicu da je to najčešća duljina u skupu podataka (slika 4.2), a u drugom slučaju na 5-10 aminokiselina, uzevši u obzir da 85% sekvenci u skupu podataka spada u navedeni raspon duljina, uvedena je kaznena funkcija koja snižava vrijednost dobre peptidima čija je duljina izvan željenog raspona. Raznolikost populacije održavana je dodatnom kaznenom funkcijom koja kvantificira sličnost svakog peptida ostatku populacije [172]. Kako bismo grafički prikazali i vizualno provjerili međusobnu sličnost generiranih peptida i sličnost odnosu na sekvence korištene za učenje modela predviđanja, konačni peptidi svakog eksperimenta predstavljeni su kao točke u dvodimenzionalnom prostoru, pri čemu njihova međusobna udaljenost indicira postotnu sličnost dobivenu globalnim poravnanjem

Needleman-Wunsch i usporedbom pripadajućih aminokiselina. Udaljenosti među točkama optimizirane su algoritmom stohastičkog gradijentnog spusta kako bi što vjernije odražavale sličnosti i razlike među peptidima. Korišteni skup podataka predstavljen je jednom točkom, a udaljenost bilo koje točke od njega indicira prosječnu sličnost sekvence u odnosu na sve sekvence u skupu podataka za učenje. Navedena analiza provedena je kako bi se utvrdilo stvara li algoritam doista nove peptide ili isključivo replicira sekvence prisutne u skupu podataka za učenje, te koliko uspješno održava različitost u populaciji.

Pet najboljih peptida iz svakog eksperimenta prema kriteriju vjerojatnosti posjedovanja samosastavljanja odabrani su za validaciju postupkom MD, koji je proveden sukladno metodologiji prethodnih studija [11, 76, 77, 260]. Korišteno je polje sila Martini verzije 2.2P, sa sekundarnom strukturom postavljenom na proširenu beta-ploču [76, 77, 260]. Simulacije su provedene u polariziranoj vodi. Kako bismo održali stalan broj aminokiselina u odnosu na veličinu simulacijskog prostora, u simulacijama heksapeptida nasumično je postavljeno 200 peptida, odnosno 120 peptida u slučaju decapeptida, u kocku veličine $20 \times 20 \times 20 \text{ nm}^3$, s resultantnom koncentracijom od 0,042M i 0,025M heksapeptida odnosno decapeptida. Simulacije su najprije podvrgnute postupku minimizacije koristeći 20000 iteracija uz korak od 20 fs , te nakon toga koristeći 50000 iteracija i korak od 10 fs . Potom je provedeno uravnotežavanje sustava koristeći termostat Bussi-Donadio-Parrinello [32] i barostat Berendsen [14], te nakon toga koristeći termostat Nosé-Hoover [104, 192] i barostat Parrinello-Rahman [202, 203, 204]. Navedeno je napravljeno kako bi se kombinirala stabilnost prve metode s preciznošću druge [111]. Faza Bussi-Donadio-Parrinello/Berendsen imala je 15000 iteracija i korak od 6 fs , dok je faza Nose-Hoover/Parrinello-Rahman imala 500000 iteracija i korak od 25 fs . Provedena su dva ispitna slučaja duljina simulacija 100 ns i 200 ns .

Omjer površine peptida dostupne otapalu (eng. solvent accesible surface area; AP_{SASA}) [145] na početku i kraju simulacija MD, međupeptidni kontakti (eng. interpeptide contacts; $AP_{kontakt}$) [252], te vizualna procjena koristili su se pri zaključivanju o mogućem samosastavljanju. Granična vrijednost AP_{SASA} određena je na temelju 20 eksperimentalno validiranih pentapeptida iz prethodne studije [11], minimizirajući broj lažno pozitivnih i lažno negativnih primjera pri svrstavanju peptida u binarne klase (tablica 4.2). Odabrana je vrijednost od 1,75, pri čemu za sve peptide čija je vrijednost AP_{SASA} iznad navedene smatramo da posjeduju mogućnost samosastavljanja, dok za preostale smatramo da istu

Tablica 4.2: Dvadeset pentapeptida iz [11] s eksperimentalno potvrđenim statusom samosastavljanja. Procjena praga samosastavljanja za daljnje analize napravljena je minimizacijom broja lažno pozitivnih (označeni crveno) i lažno negativnih primjera pri stvaranju peptida u binarne klase. Dodatno su označeni stvarno pozitivni podaci (zeleno) i stvarno negativni podaci (žuto). Preuzeto iz [191] i prevedeno.

Peptid	Predlagatelj	Status agregacije	AP _{SASA}	
PTPCY	AI	Ne	2,54	
PPPHY	AI	Ne	2,46	
SYCGY	AI	Da	2,42	
KWMDF	AI	Da	2,28	
FFEKF	AI	Da	2,24	
KWEFY	AI	Da	2,2	
FKFEF	Istraživač/ica	Da	2,12	
RWLDY	AI	Da	2,11	
WKPY Y	AI	Ne	2,11	
VVVVV	Istraživač/ica	Da	2,06	
FKIDF	AI	Da	2,03	
VKVFF	Istraživač/ica	Da	1,95	
KFFFE	Istraživač/ica	Da	1,95	
KFAFD	Istraživač/ica	Da	1,86	
VKVEV	Istraživač/ica	Da	1,84	Granična vrijednost za AP _{SASA} : 1,75
RVSVD	Istraživač/ica	Ne	1,66	
KKFDD	Istraživač/ica	Ne	1,64	
VKVKV	Istraživač/ica	Ne	1,35	
KVKVK	Istraživač/ica	Ne	1,19	
DPDPD	Istraživač/ica	Ne	1,01	

ne posjeduju. U slučaju međupeptidnih kontakata, uzeta je granična vrijednost od 0,5, iz literature [252]. U sprezi, ove dvije metrike mogu ukazati na mogućnost samosastavljanja te na oblik agregata koji se pojavljuje. Točnost generativnog modela dodatno je potvrđena stvaranjem sekvenci sa što nižom vjerojatnošću posjedovanja samosastavljanja. Navedene peptide iskoristili smo kao negativnu kontrolnu skupinu, pri čemu je cilj bio provjeriti postupa li generativni model zaista po kriterijima koji su mu zadani. Kao i u slučaju stvaranja peptida s visokom vjerojatnošću za samosastavljanje, najboljih 5 peptida iz svake kategorije podvrgnuto je simulacijama MD.

Po provedbi simulacija MD, pet peptida s visokom vjerojatnošću za samosastavljanje podvrgnuto je eksperimentalnoj validaciji. Odabrane peptide sintetizirala je tvrtka GeneCust, te je napravljen postupak izmjene pufera koristeći klorovodičnu kiselinu koncentracije 10 *mM* kako bi se uklonili eventualni ostaci trifluoroctene kiseline. Thioflavin

T (ThT) i deuterijev oksid dostavila je tvrtka Sigma-Aldrich. Liofilizirani prah svakog peptida otopljen je u natrijevoj lužini koncentracije 10 *mM*, pripremljenoj u vodi Milli-Q ili deuterijevom oksidu. Koristeći klorovodičnu kiselinu pripremljenoj u vodi Milli-Q ili deuterijevom oksidu, pH otopine je reguliran i postavljen na 7. Primjerci peptida su za daljnju karakterizaciju korišteni u pripremljenom obliku ili razrijeđeni. Zamućenost uzoraka korištena je kao indikator samosastavljanja. 100 μL uzorka postavljeno je u svaku od 96 jažica s koncentracijama od 5 *mM* do 0,039 *mM*, slijedom dvostrukih serijskih razrijeđenja, te je zabilježena apsorpcija pri valnoj duljini od 600 *nm* (OD600, čitač jažica Hidex Sense). Testovi spajanja s ThT-om također su napravljeni, koristeći koncentraciju peptida od 5 *mM*. Primjerci su inkubirani 15 minuta u otopini ThT-a pripremljenoj u metanolu, do završne koncentracije od 25 μM ThT-a. Potom je dodano 100 μL svakog primjerka u 96 jažica uz pobuđivanje na 450 *nm*, pri čemu je proučavan spektar emisija na 480 *nm* koristeći čitač jažica Tecan Infinite M200 PRO. Ispitani su i obrasci vezanja vodik, karakteristični za samosastavljanje peptida. Ispitivanje je provedeno proučavajući spektar peptida atenuirane totalne refleksije infracrvene spektroskopije s Fourierovom transformacijom (eng. attenuated total reflectance Fourier-transform infrared spectroscopy; ATR-FTIR) u deuterijevom oksidu koristeći FTIR instrument Agilent Technologies Cary 630 (Sigma-Aldrich), u rasponu od 650-4000 cm^{-1} i rezolucijom od 16 cm^{-1} . Također, peptidi koncentracije 5 *mM* obojeni su pomoću ThT-a koncentracije 25 μM . Potom su uzorci pripremljeni za mikroskopiju, pri čemu je svaki uzorak sadržavao 1 μL otopine. Mikroskopija je provedena koristeći invertirani fluorescentni mikroskop Olympus IX73 sa sljedećim staklima tvrtke Olympus: U-FUNA (plava): EX360-370, DM410, EM420-460; U-FBWA (zeleni): EX460-495, DM505, EM510-550; te U-FGW (crveni): EX530-550, DM570, EM575IF. Fotografije su prikupljene koristeći monokromnu kameru Olympus XM10 s U-FBWA (zelenim) filterom, povećanjem od 60 puta s numeričkom aperturom 1,42 i uljno-uronjenim objektivom, te programskim paketom CellSens Standard verzije 2,3. Fotografije su analizirane koristeći programski paket ImageJ verzije 1,54. Analizirane su i fotografije transmisijske elektronske mikroskopije (eng. transmission electron microscopy; TEM) dobivene koristeći mikroskop JEOL JEM-1400Flash tvrtke EMSIS, s XAROSA kamerom od 20 megapiksela i komplementarnim metal-oksidnim poluvodičem. Primjerci koncentracije 5 *mM* postavljeni su na bakrene ploče prekrivene strukturnom sondom Formvar, te su osušeni prije fotografiranja.

4.1.5. Ispitivanje konvergencije generativnog modela i analiza strukturnih motiva

Tri nezavisna eksperimenta provedena su u svrhu ispitivanja konvergencije genetskog algoritma korištenog u generativnom modelu. Naime, iako algoritam koristi kaznenu funkciju čiji je cilj održavanje raznolikosti u populaciji, intrinzično svojstvo genetskog algoritma jest konvergencija u jedno najbolje rješenje. Uvedena kaznena funkcija sprječava navedeno, ali ne i pojavu sekvenci koje dijele iste strukturne motive. U trima eksperimentima generirani su peptidi duljina između 5 i 10, s visokom vjerojatnošću za posjedovanje samosastavljanja. U svakom eksperimentu odabrana je drugačija inicijalna vrijednost generatora nasumičnih brojeva kako bi se osigurala nezavisnost svakog pokusa. Strukturni motivi prisutni u populaciji analizirani su koristeći algoritam poravnanja sekvenci Clustal Omega [150]. Prosječna sličnost sekvenci između različitih eksperimenata izračunata je kvantificiranjem sličnosti parova peptida u različitim populacijama koristeći algoritam Needleman-Wunsch, što je dosljedno s metodologijom u ostatku rada.

4.2. Rezultati i diskusija

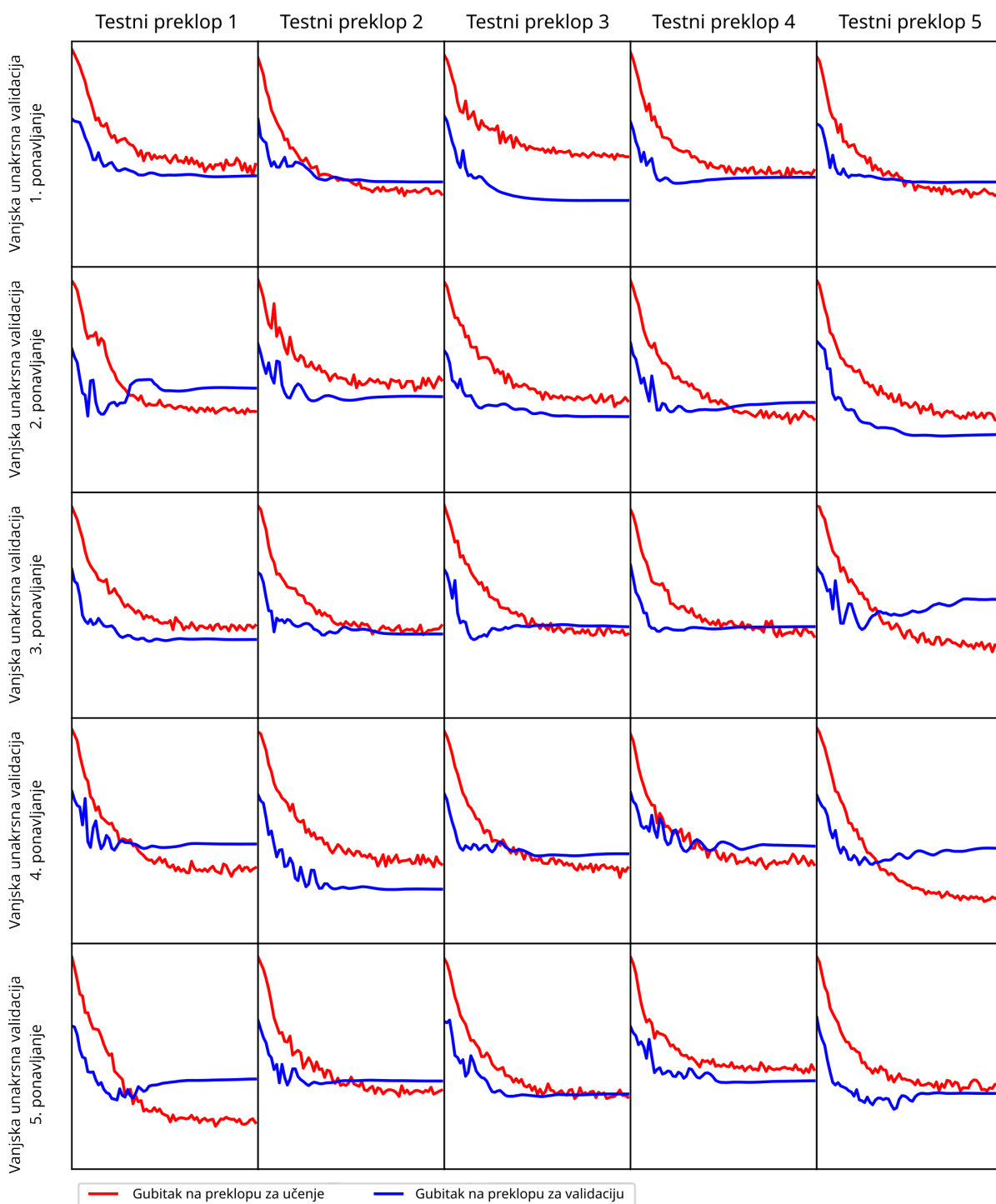
4.2.1. Performanse modela predviđanja

Prije provjere na testnom preklopu podataka, za svaki model određeni su najbolji hiperparametri prema kriteriju uspješnosti na validacijskom preklopu podataka, kao dio ugniježdene unakrsne validacije. Tablica 4.3 prikazuje najčešće odabrane vrijednosti hiperparametara za svaki model, zajedno s brojem koliko je puta navedeni parametar odabran u odnosu na ukupni broj pokusa. Primjetno je da je hibridni model AP-SP u većini slučajeva birao parametre koji rezultiraju manje složenom arhitekturom u odnosu na ostale modele. Navedeno možemo povezati s količinom informacija koje AP-SP model dobije iz ulaznih podataka. Informacije o AP vrijednostima dobivene pomičnim prozorom, upotpunjene informacijama o sekvencijalnim svojstvima peptida, predstavljaju informativne značajke na temelju kojih se točno predviđanje može dobiti i jednostavnijim modelom strojnog učenja. U tom smislu, podaci koje model dobije omogućuju mu da iskoristi jednostavniju arhitekturu za točno predviđanje svojstva samosastavljanja, što posljedično

Tablica 4.3: Najčešće odabrane vrijednosti hiperparametara pri učenju i optimizaciji modela. U zagradama je naveden broj slučajeva u kojima je navedeni parametar odabran u odnosu na ukupni broj slučajeva. Preuzeto iz [191] i prevedeno.

Model Parametar	AP	SP	AP-SP	t-SNE SP	t-SNE AP-SP
Broj ćelija u sloju LSTM	-	64 (13/25)	32 (12/25)	48 (10/25)	64 (10/25)
Veličina jezgre u konvolucijskom sloju	-	4 (10/25)	4 (13/25)	6 (12/25)	6 (11/25)
Broj neurona u potpuno povezanim slojevima	64 (12/25)	-	64 (12/25)	-	128 (10/25)

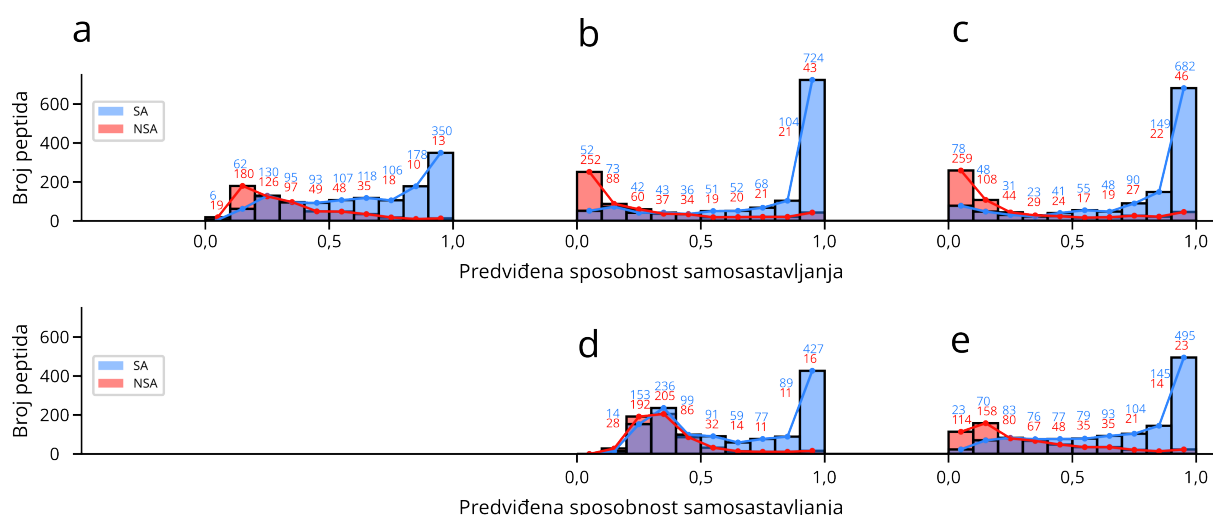
smanjuje veličinu skupa podataka potrebnu za uspješno učenje i povećava generalizacijske sposobnosti modela uz smanjenje mogućnosti za prekomjernu naučenost. Izbjegavanje prekomjerne naučenosti dodatno je potvrđeno usporedbom krivulja učenja i validacije modela AP-SP, koje u većini slučajeva koreliraju i konvergiraju u slične vrijednosti (slika 4.6). Pri likom učenja, model AP-SP postigao je najveću prosječnu točnost klasifikacije od 85,21%, te drugu najnižu stopu gubitka od 0,465, dok je model SP postigao slične performanse s drugom najvećom točnošću od 85,01% i najnižim gubitkom od 0,451 (tablica 4.4). Usporedbom histograma klasifikacija modela na preklopima za testiranje (slika 4.7), pri čemu idealna klasifikacija svim peptidima SA dodjeljuje vrijednost 1, a NSA 0, vidljivo je da modeli koji upotrebljavaju značajke sekvencijalnih svojstava uspješnije razlikuju kategorije peptida, što se vizualno manifestira kao niži broj sekvenci svrstan u središnji dio histograma. Navedeno može biti indikator da sklonost agregaciji valja oplemeniti drugim značajkama kako bi modeli strojnog učenja ostvarili puni potencijal. Varijante modela t-SNE (slika 4.7.d,e), koje su imale smanjen broj sekvencijalnih svojstava, ostvarile su rezultate slične onima modela AP, što dodatno daje naslutiti kako su sekvencijalna svojstva i njihovo slijedno procesiranje vrlo bitni za određivanje sposobnosti samosastavljanja peptida. Kod svih modela također je vidljiva i veća uspješnost u predviđanju pozitivne klase, što je očekivano s obzirom na razdiobu klasa prisutnu u skupu podataka za učenje (slika 4.2).



Slika 4.6: Usporedba krivulja gubitka na preklopima za učenje i validaciju u slučaju modela AP-SP. Krivulje su prikazane za svih 5 ponavljanja pokusa i svih 5 testnih preklopa. Preuzeto iz [191] i prevedeno.

Tablica 4.4: Prikaz vrijednosti metrika ostvarenih prilikom učenja modela. Uz maksimalne i minimalne vrijednosti točnosti i gubitka, navedene su i prosječne vrijednosti uz standardnu devijaciju. Preuzeto iz [191] i prevedeno.

Model Metrika	AP	SP	AP-SP	t-SNE SP	t-SNE AP-SP
Maksimalna točnost klasifikacije	77,11%	93,21%	93,88%	73,32%	84,01%
Minimalni gubitak	0,623	0,258	0,273	0,644	0,494
Prosječna točnost klasifikacije	71,71% (4,33%)	85,01% (7,63%)	85,21% (7,57%)	67,66% (5,54%)	76,74% (5,90%)
Prosječni gubitak	0,713 (0,073)	0,451 (0,155)	0,465 (0,149)	0,723 (0,068)	0,609 (0,104)



Slika 4.7: Histogram točnosti klasifikacije na agregiranim testnim podacima za modele (a) AP, (b) SP, (c) AP-SP, (d) t-SNE SP i (e) t-SNE AP-SP. Crvenom bojom označena je pozitivna klasa, odnosno peptidi sa svojstvom samosastavljanja (SA), dok su plavom bojom označene sekvence bez navedenog svojstva (NSA). Idealna klasifikacija svakoj molekuli SA dodjeljuje vrijednost 1, dok svakoj molekuli NSA dodjeljuje 0. Preuzeto iz [191] i prevedeno.

Što se klasifikacijskog praga tiče, isprobana je standardna vrijednost od 0,5, te pragovi procijenjeni na temelju krivulja ROC i PR. Iako se određena poboljšanja metrike gmean mogu ostvariti koristeći ROC ili prag od 0,5, pragovi PR rezultiraju najvišim vrijednostima F1 za sve modele (tablica 4.5). S obzirom na to da vrijednost F1 vjerno obuhvaća performanse modela i uzima u obzir neuravnoteženost klasa, što je ovdje slučaj, za daljnje analize odlučili smo koristiti pragove PR dobivene kao dio optimizacije hiperparametara. Model AP-SP postigao je najbolje rezultate sukladno svim metrikama i ispitnim slučajevima (tablica 4.5), na temelju čega možemo zaključiti da uspješno razlikuje peptide

sa svojstvom samosastavljanja od ostalih (točnost klasifikacije: 81,9%, vrijednost F1: 0,865). Slične performanse ostvario je i model SP (točnost klasifikacije: 80,4%, vrijednost F1: 0,856), što indicira da su sekvencijalna svojstva vrlo bitna pri klasifikaciji ovog tipa peptida. Međutim, McNemarov test statističke značajnosti ($P = 0.043$, $n = 1840$ peptida) ukazuje na značajne razlike u performansama između modela AP-SP i SP, pri čemu je AP-SP ostvario bolje rezultate. Model AP imao je nešto niže performanse, što je dosljedno s opažanjima o važnosti značajki stvorenih na temelju tablice 4.4 i slike 4.7. Zanimljivo za primijetiti jest da, iako smanjivanje broja značajki SP pomoću t-SNE nije iznjedrilo poboljšanja, model t-SNE AP-SP još uvijek je ostvario bolje rezultate u odnosu na model AP, potvrđujući važnost značajki sekvencijalnih svojstava.

Tablica 4.5: Prikaz performansi ispitanih modela na testnom skupu podataka za različite metrike i pragove. Pragovi ROC i PR određeni su na temelju udaljenosti do idealne točke klasifikacije na pripadajućim krivuljama, a ispitan je i standardni prag od 0,5. Preuzeto iz [191] i prevedeno.

Model Metrika	AP	SP	AP-SP	t-SNE SP	t-SNE AP-SP
ROC AUC	0,817 (0,004)	0,864 (0,006)	0,862 (0,017)	0,772 (0,010)	0,839 (0,014)
gmean (Prag ROC)	0,742 (0,005)	0,796 (0,009)	0,798 (0,018)	0,702 (0,008)	0,759 (0,013)
F1 (Prag ROC)	0,772 (0,006)	0,833 (0,009)	0,841 (0,008)	0,688 (0,010)	0,784 (0,011)
Acc (Prag ROC)	72,5% (0,6%)	79,0% (1,0%)	79,7% (1,1%)	66,2% (0,9%)	73,9% (1,2%)
PR AUC	0,896 (0,005)	0,919 (0,003)	0,920 (0,009)	0,890 (0,004)	0,911 (0,010)
gmean (Prag PR)	0,639 (0,013)	0,768 (0,010)	0,794 (0,016)	0,582 (0,021)	0,728 (0,013)
F1 (Prag PR)	0,824 (0,006)	0,856 (0,002)	0,865 (0,008)	0,796 (0,005)	0,830 (0,008)
Acc (Prag PR)	74,6% (0,8%)	80,4% (0,4%)	81,9% (1,1%)	70,4% (0,8%)	77,0% (1,0%)
gmean (0,5)	0,739 (0,005)	0,797 (0,012)	0,801 (0,017)	0,716 (0,007)	0,760 (0,017)
F1 (0,5)	0,771 (0,006)	0,844 (0,010)	0,853 (0,006)	0,717 (0,008)	0,800 (0,014)
Acc (0,5)	72,3% (0,6%)	79,9% (1,2%)	80,9% (0,9%)	68,2% (0,8%)	75,2% (1,6%)

Predloženi modeli uspoređeni su s postojećim modelima iz literature, pri čemu je cilj bio postaviti ih u kontekst i provjeriti ostvaruju li napredak, te ako da, u kojoj mjeri. Usporedba je napravljena s modelom nasumične šume, koji je dosad ostvario odlične rezultate u sličnim studijama [11, 35, 187], neuronskim mrežama temeljenim na povratnoj arhitekturi [67], slojevima LSTM [86, 226] i višeslojnom perceptronu [144], te transformatorom [261], preuzetima iz prethodne studije [162]. Svi modeli s kojima smo se usporedili slijedno procesiraju ulazne vrijednosti, s iznimkom modela nasumične šume čija arhitektura ne dozvoljava navedeno. Najbolje rezultate iznjedrilo je transformer (gmean: 0,819, vrijednost F1: 0,878, točnost klasifikacije: 83,7%), pri čemu je neuronska mreža sa slojevima LSTM ostvarila identičan rezultat u slučaju metrike F1 (tablica 4.6). S obzirom na

to da su razlike u točnosti klasifikacije svih modela unutar 5 postotnih bodova, provedena je analiza statističke značajnosti između AP-SP i ostalih modela strojnog učenja koristeći McNemarov test ($\alpha = 0,05$, $n = 1840$). Značajne razlike uočene su između modela AP-SP te modela RNN i MLP (označeno zvjezdicom u tablici 4.6), od kojih je AP-SP ostvario bolje rezultate. U preostalim slučajevima nije bilo značajnih razlika, indicirajući jednake performanse AP-SP i ostalih modela. S druge strane, usporedba modela napravljena na 20 eksperimentalno potvrđenih pentapeptida iz literature [11] (donji dio tablice 4.6), dodatno je ukazala na dobre performanse modela AP-SP i njegovu sposobnost da iznjedri točniju klasifikaciju u odnosu na istraživače i dosad najtočniji model iz literature [11].

Tablica 4.6: Usporedba modela iz literature i predloženog modela AP-SP. Modeli su uspoređeni na temelju metrike gmean, vrijednosti F1 i točnosti klasifikacije. Ispitano je postojanje statistički značajnih razlika (McNemarov test, $\alpha = 0,05$, $n = 1840$) između AP-SP i ostalih modela, a napravljena je i procjena performansi na 20 pentapeptida iz literature [11]. Preuzeto iz [191] i prevedeno.

Model Metrika	Transformer	RNN	LSTM	Dvosmjerni LSTM	MLP	RF	AP-SP	Performanse istraživača iz [11] ¹	Performanse modela AI iz [11] ²
gmean	0,819 (0,011)	0,746 (0,023)	0,805 (0,011)	0,797 (0,016)	0,748 (0,023)	0,772 (0,010)	0,794 (0,016)		
F1	0,878 (0,005)	0,852 (0,012)	0,878 (0,005)	0,873 (0,010)	0,848 (0,006)	0,875 (0,006)	0,865 (0,008)		
Točnost klasifikacije	83,7% (0,7%)	79,5% (1,7%)	83,4% (0,7%)	82,8% (1,4%)	79,2% (1,0%)	82,6% (0,8%)	81,9% (1,1%)		
P-vrijednost	0,052	0,025*	0,105	0,349	0,011*	0,464	Osnova usporedbe		
gmean	0,780 (0,025)	0,775 (0,100)	0,841 (0,075)	0,803 (0,034)	0,712 (0,073)	0,702 (0,038)	0,928 (0,041)	0	0
F1	0,789 (0,025)	0,828 (0,066)	0,861 (0,065)	0,824 (0,028)	0,791 (0,043)	0,817 (0,032)	0,930 (0,035)	0,706	0,800
Točnost klasifikacije	77% (2,4%)	79% (8,6%)	84% (7,3%)	80% (3,2%)	74% (5,8%)	76% (3,7%)	92% (4,0%)	55%	67%

¹ Performanse istraživača procijenjene su na temelju 11 peptida koje su predložili.

² Performanse modela AI procijenjene su na temelju 9 peptida koje je generirao.

4.2.2. Performanse generativnog modela

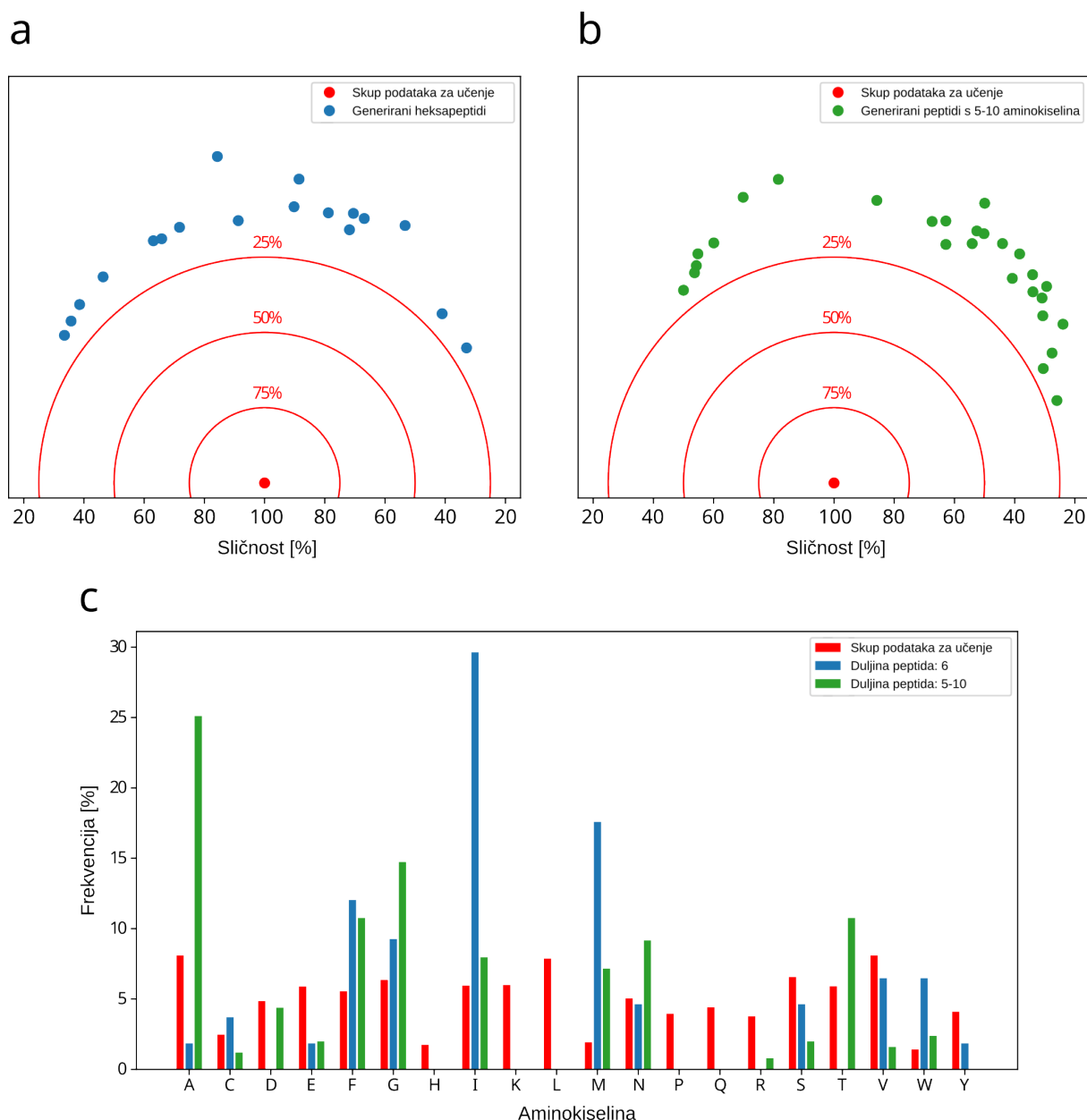
Cilj uporabe generativnog modela jest stvoriti nove, dosad nepoznate peptide koji posjeduju visoku vjerojatnost samosastavljanja. Pet najboljih peptida iz svakog od dvaju eksperimenata stvaranja heksapeptida i peptida s duljinama u rasponu 5-10 aminokiselina predstavljeno je tablicom 4.7. U oba slučaja, sličnost novostvorenih sekvenci u odnosu na peptide iz skupa podataka za učenje ($S_{učenje}$) niža je od 10%, što indicira da generativni model pretražuje različite regije kemijskog prostora i ne provodi preslikavanje primjera korištenih za učenje modela predviđanja. Međusobna sličnost generiranih peptida ($S_{generirani}$)

ipak je veća, no to je očekivano s obzirom na tendenciju genetskih algoritama konvergiranja u jedno najbolje rješenje (dodatno ispitano u potpoglavlju 4.2.3.). Postojanje razlika među sekvencama u konačnoj populaciji može se pripisati upotrebi kaznene funkcije za raznolikost populacije. Do sličnih zaključaka moguće je doći vizualizirajući generirane populacije točkama, pri čemu udaljenosti među njima predstavljaju međusobnu sličnost dvaju peptida (slika 4.8.a,b). Kako u ovakvom prikazu nije bilo moguće pronaći raspored točaka koji bi apsolutno točno odražavao međusobnu sličnost sekvenci i njihovu sličnost podacima za učenje, sličnost skupu za učenje uzeta je kao mjerodavnija metrika te je pogreška vizualizacije u tom pogledu zadržana ispod 6%. Provjerena je i razdioba aminokiselina, gdje se također mogu uočiti značajne razlike među generiranim i peptidima za učenje (slika 4.8.c).

Tablica 4.7: Pet najboljih peptida prema kriteriju vjerojatnosti samosastavljanja iz dva eksperimenta stvaranja heksapeptida i peptida duljine između 5 i 10 aminokiselina. Vjerojatnost samosastavljanja procijenjena je hibridnim modelom AP-SP. Metrike $S_{učenje}$ i $S_{generirani}$ ukazuju na sličnost primarne strukture peptida s peptidima korištenim za učenje modela predviđanja, odnosno ostalim generiranim peptidima. $AP_{kontakt}$ i AP_{SASA} izračunati su na temelju prvog i zadnjeg vremenskog trenutka simulacije molekularne dinamike. Preuzeto iz [191] i prevedeno.

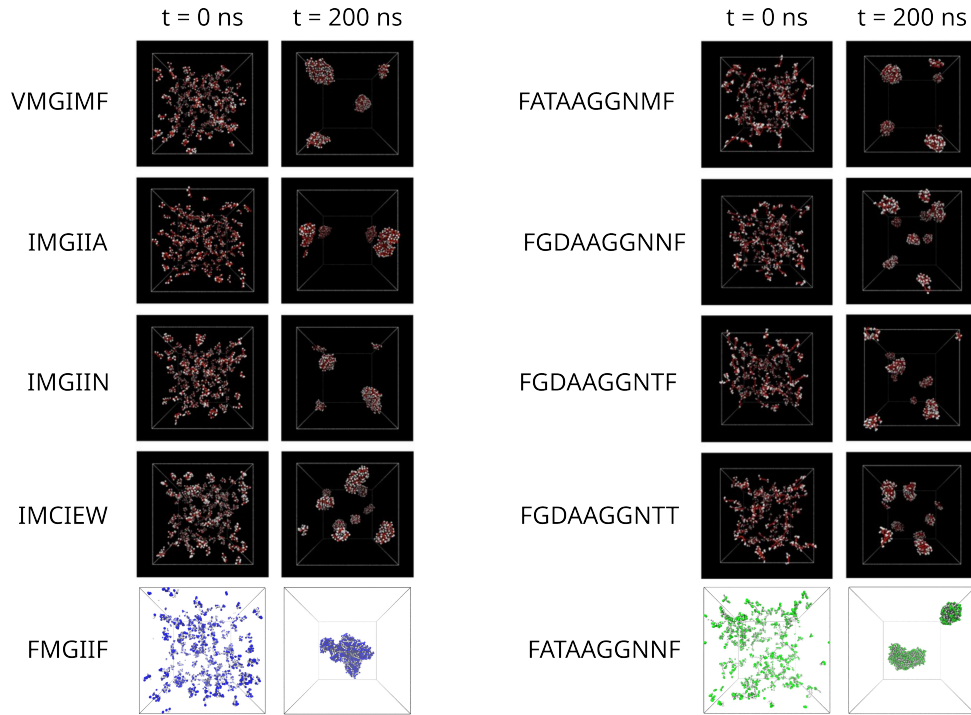
Peptid	Vjerojatnost samosastavljanja	$S_{učenje}$	$S_{generirani}$	$AP_{kontakt}$	AP_{SASA}
Heksapeptidi					
IMGIIA	99,4%	9,2%	62,5%	0,47	1,86
IMCIEW	99,0%	9,1%	41,7%	0,54	1,77
VMGIMF	98,9%	7,2%	50,0%	0,52	2,13
FMGIIF	98,8%	9,7%	58,3%	0,61	2,29
IMGIIIN	95,2%	9,0%	62,5%	0,70	2,00
Peptidi s 5-10 aminokiselina					
FGDAAGGNTT	99,9%	7,3%	74,3%	0,85	1,85
FATAAGGNNF	99,7%	7,2%	76,8%	0,92	2,38
FGDAAGGNNF	99,7%	6,6%	81,6%	0,82	1,88
FGDAAGGNTF	99,7%	7,1%	81,8%	0,85	1,90
FATAAGGNMF	98,3%	8,1%	74,5%	0,85	2,19

Kao dodatna provjera pojave samosastavljanja predloženih sekvenci, provedene su simulacije molekularne dinamike te su analizirane metrike $AP_{kontakt}$ i AP_{SASA} . U slučaju heksapeptida, svi peptidi osim IMGIIA prelaze zadane pragove za samosastavljanje od 0,5 za $AP_{kontakt}$ i 1,75 za AP_{SASA} , pri čemu IMGIIA ima vrijednost $AP_{kontakt}$ od 0,47. Dodatnom vizualnom analizom peptida na početku i kraju simulacija (slika 4.9) potvrđeno



Slika 4.8: Dvovimenzionalni prikaz generiranih sekvenci, njihovih međusobnih sličnosti, te sličnosti u odnosu na peptide korištene pri učenju modela predviđanja za (a) heksapeptide i (b) peptide duljine između 5 i 10 aminokiselina. Dodatno, napravljena je i (c) usporedba frekvencija aminokiselina među trima populacijama peptida. Preuzeto iz [191] i prevedeno.

je da svi formiraju agregate, pri čemu IMGIIA ispoljava nešto plosnatiju morfologiju u odnosu na ostale, što može biti razlog sniženoj vrijednosti $AP_{kontakt}$. U slučaju peptida duljine 5-10 aminokiselina, vidljiva je preferencija generativnog modela prema dekaeptidima, a sve sekvence prelaze zadane kriterije i formiraju agregate (slika 4.9). Stvaranje kontrolne skupine generativnim modelom rezultiralo je peptidima prikazanim u tablici 4.8.

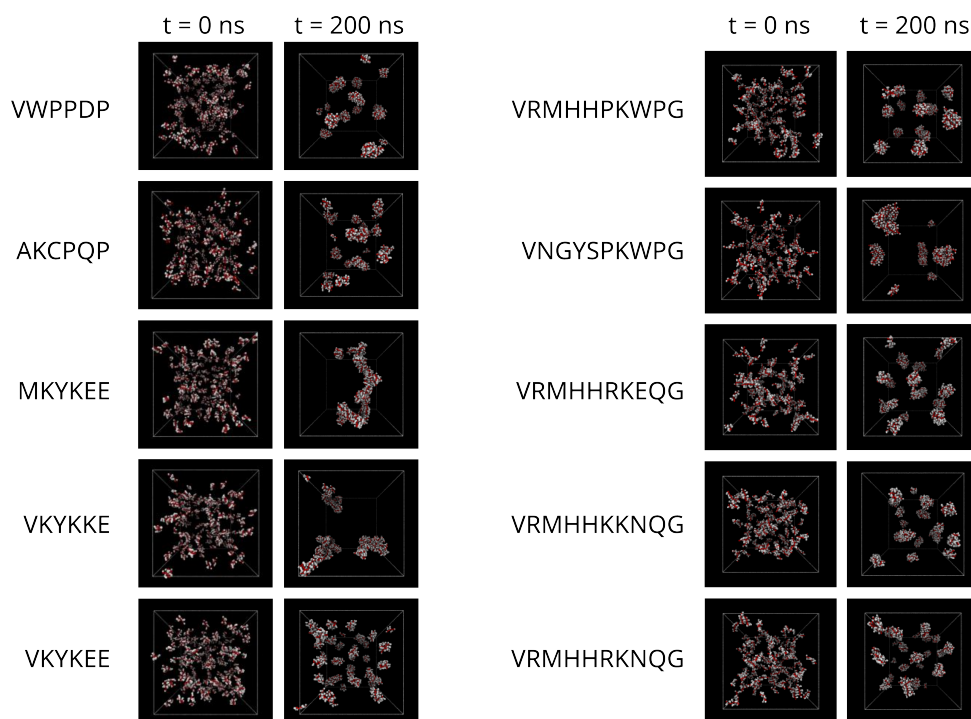


Slika 4.9: Prikaz početnih i završnih trenutaka simulacija molekularne dinamike za heksapeptide i peptide duljine između 5 i 10 aminokiselina koji imaju visoku vjerojatnost samosastavljanja. Preuzeto iz [191] i prevedeno.

Tablica 4.8: Pet najboljih peptida dviju kontrolnih skupina, stvorenih tako da imaju što nižu vjerojatnost samosastavljanja. Vjerojatnost samosastavljanja procijenjena je hibridnim modelom AP-SP. $AP_{kontakt}$ i AP_{SASA} izračunati su na temelju prvog i zadnjeg vremenskog trenutka simulacije molekularne dinamike. Preuzeto iz [191] i prevedeno.

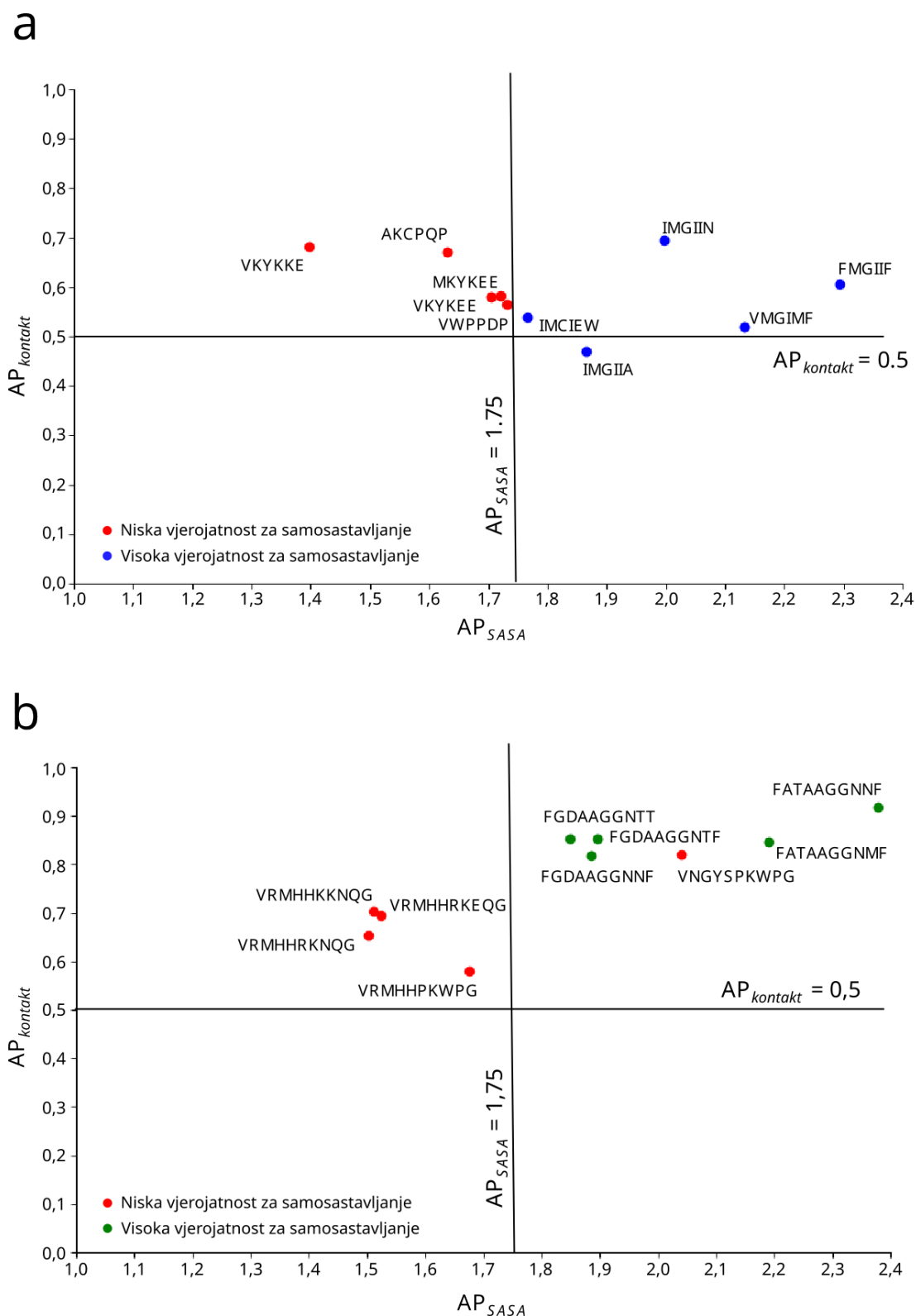
Peptid	Vjerojatnost samosastavljanja	$AP_{kontakt}$	AP_{SASA}
Heksapeptidi			
VWPPDP	0,3%	0,56	1,73
AKCPQP	0,4%	0,67	1,63
MKYKEE	0,4%	0,58	1,72
VKYKEE	0,4%	0,58	1,70
VKYKKE	0,4%	0,68	1,40
Dekapeptidi			
VRMHHPKWPG	1,1%	0,58	1,68
VNGYSPKWPG	1,2%	0,82	2,04
VRMHHRKEQG	1,2%	0,70	1,52
VRMHKKKNQG	1,3%	0,70	1,51
VRMHHRKNQG	1,3%	0,65	1,50

Iako svi prelaze pragove metrike $AP_{kontakt}$, koja ukazuje na morfologiju agregata, jedino peptid VNGYSPKWPG prelazi kriterij AP_{SASA} , koji ukazuje na svojstvo samosastavljanja. Također, vidljiva je i razlika u rasporedu molekula u odnosu na peptide koji su agregirali (slika 4.10). Usporedba generiranih peptida sa svojstvom samosastavljanja i kontrolne skupine pokazuje jasne razlike u njihovu smještaju u odnosu na kontrolne kriterije, pri čemu peptidi sa svojstvom samosastavljanja dominantno zauzimaju dio grafa koji predstavlja sekvence s vrijednošću AP_{SASA} iznad 1,75 (slika 4.11). Navedeni rezultati mogu se smatrati svojevrsnom potvrdom dobre korelacije modela predviđanja i simulacija MD.



Slika 4.10: Prikaz početnih i završnih trenutaka simulacija molekularne dinamike za heksapeptide i peptide duljine između 5 i 10 aminokiselina koji imaju nisku vjerojatnost samosastavljanja. Preuzeto iz [191] i prevedeno.

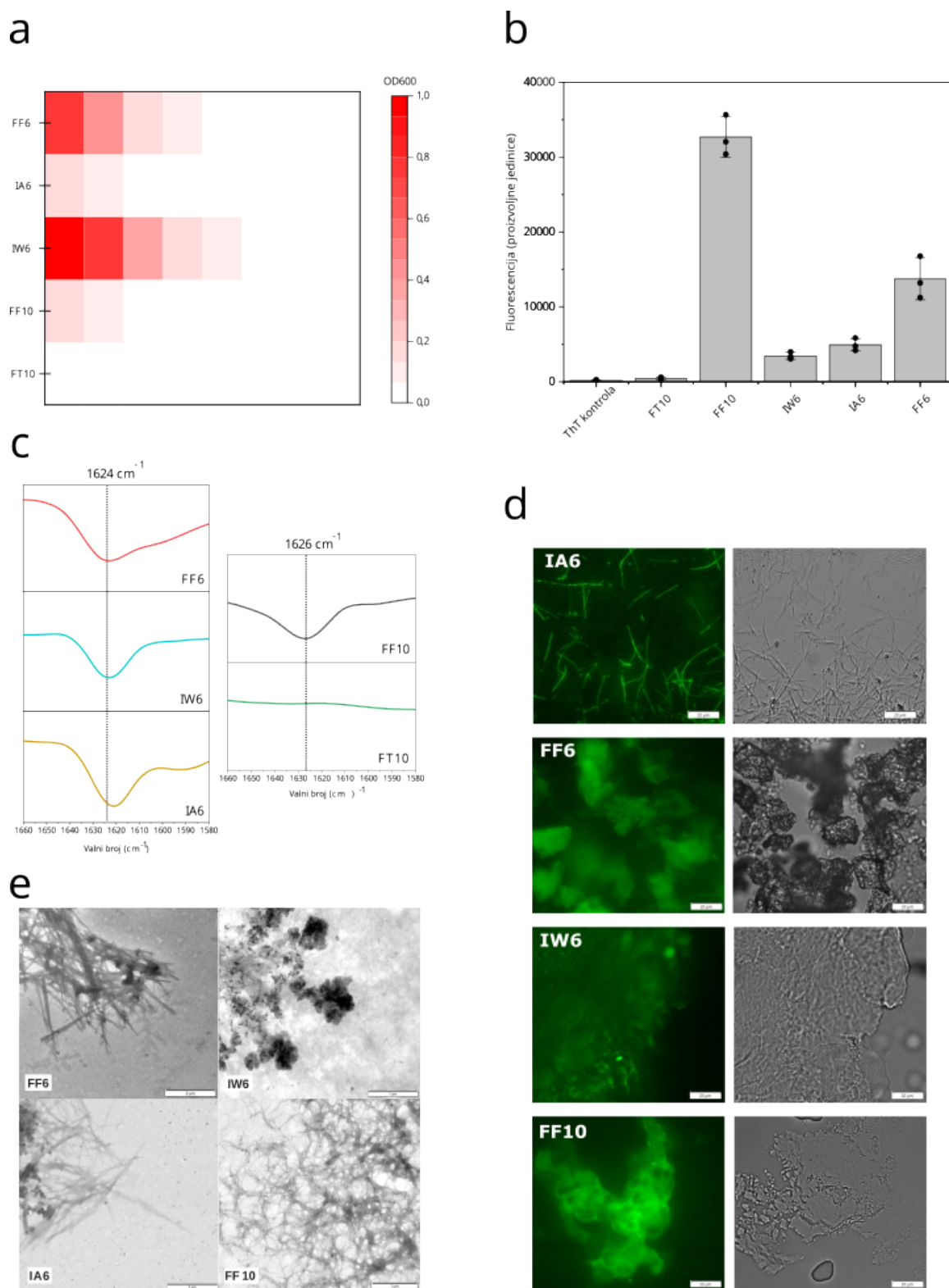
Eksperimentalno je validirano pet peptida, odabranih na temelju procijenjene vjerojatnosti samosastavljanja pomoću modela ML i vrijednosti metrike AP_{SASA} : FMGIIF (FF6), IMGIIA (IA6), IMCIEW (IW6), FATAAGGNMF (FF10) i FGDAAGGNTT (FT10). Peptidi FF10, IW6, IA6 i FF6 rezultirali su vrijednošću OD600 većom od 0,1. Usporedbe radi, vrijednost OD600 za vodu je 0,036, što indicira agregaciju prethodno navedenih peptida [11]. Nadalje, FF6 i IW6 rezultirali su zamućenim suspenzijama čestica, dok su IA6 i FF10 rezultirali prozirnijim, viskoznim suspenzijama. Otopina peptida FT10 ostala je



Slika 4.11: Dvodimenzionalni razmještaj (a) heksapeptida i (b) peptida duljine između 5 i 10 u odnosu na kriterije $AP_{kontakt}$ i AP_{SASA} , te njihove definirane pragove. Peptidi s visokom vjerojatnošću samosastavljanja označeni su plavo i zeleno, ovisno o duljini, dok su peptidi s niskom vjerojatnošću samosastavljanja označeni crveno. Preuzeto iz [191] i prevedeno.

prozirna za sve isprobane koncentracije (slika 4.12.a). Stvaranje struktura nalik beta-ploči potvrđeno je povećanjem fluorescencije ThT-a na 480 nm, pri čemu je najveće povećanje zabilježeno za FF10, a potom i za FF6, IA6 i IW6 (slika 4.12.b). Razlike u intenzitetima fluorescencije mogu se tumačiti i kroz prizmu afiniteta vezanja ThT-a za specifične strukturne mete amiloidnih vlakana [17], što može ukazivati na različitu morfologiju svakog od ispitanih peptida. Također, moguće je uočiti karakterističan vrh u spektru FTIR na 1624 cm^{-1} za heksapeptide i 1626 cm^{-1} za FF10, ukazujući na obrasce spajanja vodika karakteristične za beta-ploču (slika 4.12.c). Fluorescentna mikroskopija ponudila je vizualnu potvrdu vezanja ThT-a za agregate koje su peptidi formirali, od nakupina vlakana za IA6, do većih nakupina nedefinirane morfologije za FF6, IW6 i FF10 (slika 4.12.d), potvrđujući amiloidnu agregaciju. Mikroskopija TEM potvrdila je postojanje zamršenih nanovlakana za FF6 i IA6, mreža vlakana za FF10, te amorfnih agregata za IW6 (slika 4.12.e). Peptid FT10 nije pokazao tendenciju ka samosastavljanju niti u jednoj od primijenjenih metoda. Ova preliminarna eksperimentalna provjera ukazuje na 80%-tnu točnost generativnog modela, a time i modela predviđanja. Međutim, s obzirom na to da rezultati eksperimenata uvelike ovise o postavkama kao što su pH vrijednost i koncentracija peptida, u budućnosti bi svakako bilo poželjno provesti eksperimente s više sekvenci kako bi se dodatno ispitalo funkcioniranje predloženih modela, te detaljno ispitao potencijal generiranih peptida za formiranje novih peptidnih materijala.

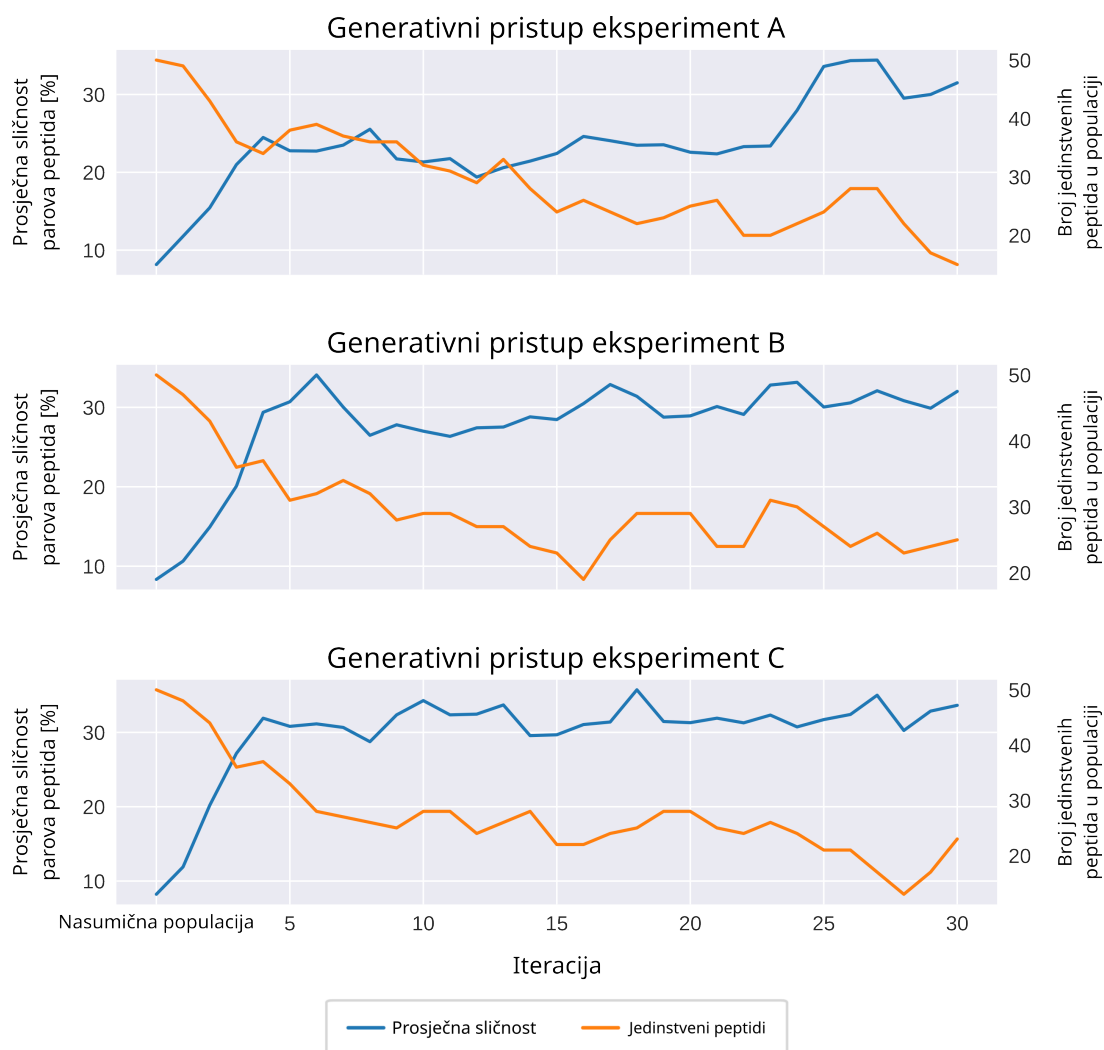
Izneseni rezultati potvrđuju dobre performanse generativnog modela i modela predviđanja. Vođen modelom predviđanja, generativni model uspješno je stvorio peptide koji su vrlo različiti u odnosu na one prijavljene u literaturi, te su u simulacijama molekularne dinamike i laboratorijskim eksperimentima pokazali tendenciju ka formiranju nanostrukture, odnosno samosastavljanju. Također, generativni model je nadišao točnost dosad najboljeg modela za stvaranje novih peptida sa svojstvom samosastavljanja i istraživača, postigavši 80%-tnu točnost [11]. Sukladno navedenom, predloženi generativni model nudi istraživačima mogućnost nepristrane pretrage širokih kemijskih prostora u svrhu pronalazjenja novih sekvenci s visokom vjerojatnošću za posjedovanje svojstva samosastavljanja. Izvorni kodovi modela predviđanja i generativnog modela, zajedno s naučenim modelima strojnog učenja, dostupni su u repozitoriju GitHub na sljedećoj internetskoj adresi: https://github.com/mnjirjak/ml_peptide_self_assembly.



Slika 4.12: Rezultati laboratorijskog ispitivanja pet najperspektivnijih peptida prema kriteriju vjerojatnosti samosastavljanja. (a) Prozirnost otopine izmjerena postupkom OD600. (b) Ispitivanje fluorescencije peptida. (c) Spektar FTIR za pet ispitanih peptida. (d) Prikaz provedene fluorescentne mikroskopije. (e) Mikroskopija TEM. Preuzeto iz [191] i prevedeno.

4.2.3. Konvergencija genetskog algoritma i strukturni motivi

Selekcijski pritisak koji se primjenjuje tijekom evolucije u obliku funkcije dobrote do-
vodi do konvergencije populacije u jedno najbolje rješenje, odnosno u jedan peptid. Na-
vedeno je posljedica osnovnih principa evolucije pri čemu se osobina koja daje evolucijsku
prednost, u ovom slučaju veća sklonost ka samosastavljanju, propagira kroz populaciju
i favorizira opstanak peptida s navedenom karakteristikom. Provedena su tri eksperi-
menta označena slovima A, B i C, čiji je cilj bio stvoriti peptide duljine između 5 i 10
aminokiselina s viskom vjerojatnošću samosastavljanja, te tako detaljnije ispitati ponaša-
nje predloženog generativnog modela. Tijekom evolucije, vidljivo je povećanje prosječne
sličnosti peptida u populaciji kroz generacije, te istovremeno smanjenje broja jedinstvenih



Slika 4.13: Prikaz rezultata triju nezavisnih eksperimenata konvergencije generativnog modela. Vidljivo je povećanje sličnosti peptida i smanjenje broja jedinstvenih sekvenci kroz iteracije. Preuzeto iz [191] i prevedeno.

sekvenci (slika 4.13). No, iako generativni model teži smanjenju različitosti populacije kroz generacije, kaznena funkcija još uvijek sprječava da sva rješenja postanu identična. Također, stohastička komponenta algoritma osigurava konvergenciju populacije u različite dijelove kemijskog prostora pri svakom pokretanju algoritma, što potvrđuje i prosječna sličnost sekvenci između populacija eksperimenata A i B od 9,8%, 11,4% između populacija eksperimenata A i C, te 10,7% između populacija eksperimenata B i C (tablica 4.9). Moguće je uočiti i različite strukturne motive prisutne u stvorenim sekvencama.

Tablica 4.9: Lista peptida stvorenih u trima eksperimentima ispitivanja konvergencije generativnog modela. Uz peptide navedena je i njihova procijenjena sklonost ka samosastavljanju, te strukturni motivi koji se pojavljuju. Pri višestrukome pojavljivanju iste sekvence u populaciji, u prikazu je zadržan isključivo jedan primjer. Preuzeto iz [191] i prevedeno.

Eksperiment A			Eksperiment B			Eksperiment C		
Motivi	Peptidi	Vjerojatnost samosastavljanja	Motivi	Peptidi	Vjerojatnost samosastavljanja	Motivi	Peptidi	Vjerojatnost samosastavljanja
FVLAEPF-M			FY--DGDYT	FYYIDGDYN	99,8%	MMYYEEMEQ	MMMFYETMRF	99,6%
				FYYIDGDYT	99,8%		MYMYEEMRF	99,6%
	FVLAEPFRD	99,7%		MIYIDGDYT	99,8%		MYMYEEMEQ	99,7%
	FVLAEPFRFT	99,8%		MYYIDGDYT	99,8%		MMMFYEEMEQ	99,8%
	FVLAEVMMVM	99,7%		FYGFDDGIN	99,8%		MMMYEEMEQ	99,9%
CCLNLNNT				FWGFDGDYT	99,8%	WLDLTIQVDQ	WLDLTIQVDQ	99,9%
				FYGFDDPT	99,8%		WLDLTIQVDQ	99,9%
				FYGFDDYT	99,8%		WLDLTIQVDQ	99,9%
	CCLNLNNYD	99,8%	MW-WVAGQWN	MMISVMMIN	99,9%		CCCLTIQVDQ	99,9%
	CCLNCNNYT	99,8%		FYGFVAGQYN	99,9%		VLDLTIQVDA	99,8%
CHHCSIMGGP				MWNVVMMWN	99,9%		MDLTIQVDQ	99,9%
	MVCSIMGGP	99,8%		MWLWEAGQYN	99,8%	WCCAKEEMEQ	WCCAKIEMRF	99,7%
	FHCSIMGGP	99,9%		FYGVVVGQWH	99,8%		CCCLKEEMEQ	99,8%
	CHHCSIMGGD	99,9%		MWNVVAGQWN	99,9%		WCCAKEEMEQ	99,9%
	CHHCSIMGGP	99,9%		MWNWVAGQWH	99,7%	CC-LDLTMRF	WLDLTLTMRA	99,5%
CQHCQIMMGGH				MWLWVAGQWH	99,8%		WLDLTLTMRF	99,8%
			MYFFEHLFHF	MWLWVVGQWH	99,7%		VDDLTLTMRF	99,6%
	CQHCQMYVVD	99,8%		MWLWEHCLIN	99,8%		VDDLTLTMRF	99,6%
	CQHCQMMVM	99,8%		MWLWEHCLFH	99,7%		CCCLYETMRF	99,5%
	CQHCQIMMGGH	99,9%		MWLWVHCLFH	99,6%		CCCLTLTMA	99,8%
	CHHCSIMMGGH	99,9%		FYFFEHLFHF	99,7%		CCCLTLTMRA	99,8%
	FHHCSIMMGGH	99,9%		FYFFEHLFHF	99,6%		CCCLTLTMRF	99,8%
				FYFFEHLCLYN	99,7%			
				MMFFEHLFHF	99,7%			
				MYFFEHLFHF	99,7%			

5. EVOLUCIJSKI DIZAJN INHIBITORA BETA-SEKRETAZE 1

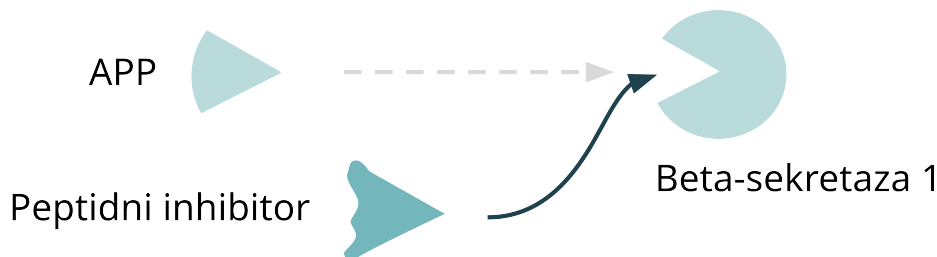
Rezultati istraživanja opisani u ovom poglavlju prezentirani su na konferenciji 37th European Peptide Symposium and 14th International Peptide Symposium u obliku postera i objavljeni: M. Njirjak, D. Kalafatovic, and G. Mauša. AI-driven design of β -secretase 1 inhibitors. In 37th European Peptide Symposium and 14th International Peptide Symposium, 2024. [189].

Alzheimerova bolest i ostali tipovi demencija postale su sedmi vodeći razlog smrtnosti u svijetu [278], a s obzirom na demografski trend starenja stanovništva [184], njihova zastupljenost u populaciji povećala se za 181% između 2000. i 2019. Projekcije Svjetske zdravstvene organizacije indiciraju da će do 2050. demencija postati vodeći uzrok invalidnosti i gubitka samostalnosti.

Alzheimerova bolest je vrlo složena i djelomično nastaje kao rezultat procesa starenja organizma [270]. Iako se istraživanja o točnim uzrocima i mogućim terapijskim smjerovima još uvijek intenzivno provode [26, 102, 217], vjeruje se da nakupljanje velikih količina proteina amiloid-beta i tau u obliku krivo savijenih amiloidnih nakupina uvelike pridonosi razvoju bolesti [247]. Starenjem se smanjuje sposobnost ljudskog tijela za razgradnju i uklanjanje nepotrebnih ili disfunkcionalnih proteina [270], pri čemu dolazi do povećanja koncentracije prethodno spomenutih amiloidnih nakupina u mozgu. Vjeruje se da nakupine mogu potaknuti neurodegeneraciju direktnom interakcijom s neuronima, ali i indirektno, potičući stvaranje reaktivnih vrsta kisika (eng. reactive oxygen species; ROS) i kroničnu neuroinflamaciju [266]. Kako ROS također uništava i neurone, a malo

je vjerojatno da će upalni proces završiti bez vanjske intervencije, dolazi do sve većeg gubitka moždane mase kako bolest napreduje [130]. U ranim stadijima bolesti osoba može ispoljavati probleme s pamćenjem i govorom, što se nekada krivo pridaje normalnim simptomima starenja [33, 248]. U kasnijim stadijima, osoba sve više gubi samostalnost, a motoričke sposobnosti se pogoršavaju, što otežava svakodnevno funkcioniranje. U kasnim stadijima bolesti dolazi do ozbiljnog gubitka neurona, halucinacija, potpune ovisnosti o trećim osobama, i smrti.

Dio nakupina nastaje razgradnjom amiloidnog prekursorskog proteina (eng. amyloid-beta precursor protein; APP) enzimima beta- i gama-sekretaze [295], pri čemu djelovanje na navedeni mehanizam predstavlja potencijalni način usporavanja napretka bolesti [95]. Premisa je slična kao u prethodnom istraživanju mamaca za šiljasti protein SARS-CoV-2 [158]. Stvaranjem peptida koji će se vezati za beta-sekretazu 1, onemogućuje se vezanje APP-a, što bi posljedično trebalo rezultirati nižom koncentracijom amiloidnih nakupina u mozgu (slika 5.1). Međutim, dizajnirati peptide koji imaju visok afinitet vezanja za metu te pritom uspješno prolaze iznimno selektivnu krvno-moždanu barijeru je vrlo izazovno. Posljedično, u ovom istraživanju, a ohrabreni prethodnim uspjesima (vidjeti poglavlja 3. i 4.), razvili smo genetski algoritam za dizajn peptidnih inhibitora enzima beta-sekretaze 1, vođen kriterijima afiniteta vezanja za metu, logP vrijednošću Wildman-Crippen, te procijenjenom vjerojatnošću za prolazak krvno-moždane barijere pomoću modela strojnog učenja.



Slika 5.1: Prikaz mehanizma funkcioniranja inhibitora enzima beta-sekretaze 1. Vezanjem za enzim, inhibitor sprječava vezanje amiloidnog prekursorskog proteina i njegovu razgradnju. Preuzeto iz [189] i prevedeno.

5.1. Metodologija

Za razliku od prethodnih istraživanja provedenih u sklopu ove disertacije, u ovom slučaju postoje višestruki optimizacijski kriteriji koje je potrebno istovremeno uzeti u obzir, počevši s afinitetom vezanja prema enzimu beta-sekretaze 1. Procjena energije vezanja provodi se tehnikom molekularnog pristajanja koristeći alat DOCKSTRING [80]. DOCKSTRING automatizira pripremu liganada kroz postupak stvaranja 3D struktura i njihovo poboljšanje koristeći polje sila MMFF94 [94], postupak protonacije ili deprotonacije koristeći OpenBabel [193], dok se za samu procjenu afiniteta vezanja u $kcal/mol$ koristi AutoDock Vina [255]. Negativnije vrijednosti koreliraju s jačom energijom vezanja. Postavke DOCKSTRING-a ostavljene su na zadanim vrijednostima. Za procjenu vjerojatnosti prelaska krvno-moždane barijere korištena su dva kriterija. Prvo je izračunata logP vrijednost Wildman-Crippen [276] kao indikator lipofilnosti peptida, odnosno njegove sposobnosti da se uspješno otopi u lipidima. Potom je korištena procjena prolaska krvno-moždane barijere postojećim modelom previđanja iz literature [51]. Model logističke regresije naučen je i testiran koristeći 238 peptida iz etabliranih baza kao što su Brainpeps [259], SATPdb [238], PepBank [235] i UniProt [253], pri čemu je ostvario točnost predviđanja od 79% na skupu podataka za testiranje.

Kako u ovom slučaju govorimo u istovremenoj optimizaciji više kriterija, prethodno razvijeni genetski algoritmi (vidjeti poglavlja 3. i 4.) nisu mogli biti korišteni. Zbog toga je kao okosnica generativnog modela upotrijebljen algoritam NSGA-II [61]. NSGA-II podržava proizvoljan broj kriterija za vrednovanje svakog rješenja, pri čemu usporedbu rješenja svodi na dva broja: rang prema kvaliteti i udaljenost grupiranja. Rangovi svrstavaju rješenja među kojima ne postoji relacija dominacije u zasebne nedominirane fronte, dok se udaljenosti grupiranja koriste pri usporedbi rješenja koja se nalaze na istoj nedominiranoj fronti. Navedeno omogućava višekriterijsku optimizaciju molekula prema tri kriterija istovremeno, pri čemu se lipofilnost i vjerojatnost prolaska krvno-moždane barijere nastoje maksimizirati, a procijenjena energija afiniteta vezanja minimizirati. Populacija veličine 100 peptida održava se konstantnom, a u svakoj od 35 iteracija stvara se 30 novih molekula. Korištena je rekombinacija s jednom točkom prijeloma. Veličina turnirske selekcije pri izboru roditelja i vjerojatnost mutacije varirane su u postupku određivanja hiperparametara algoritma, te su ispitane sljedeće vrijednosti:

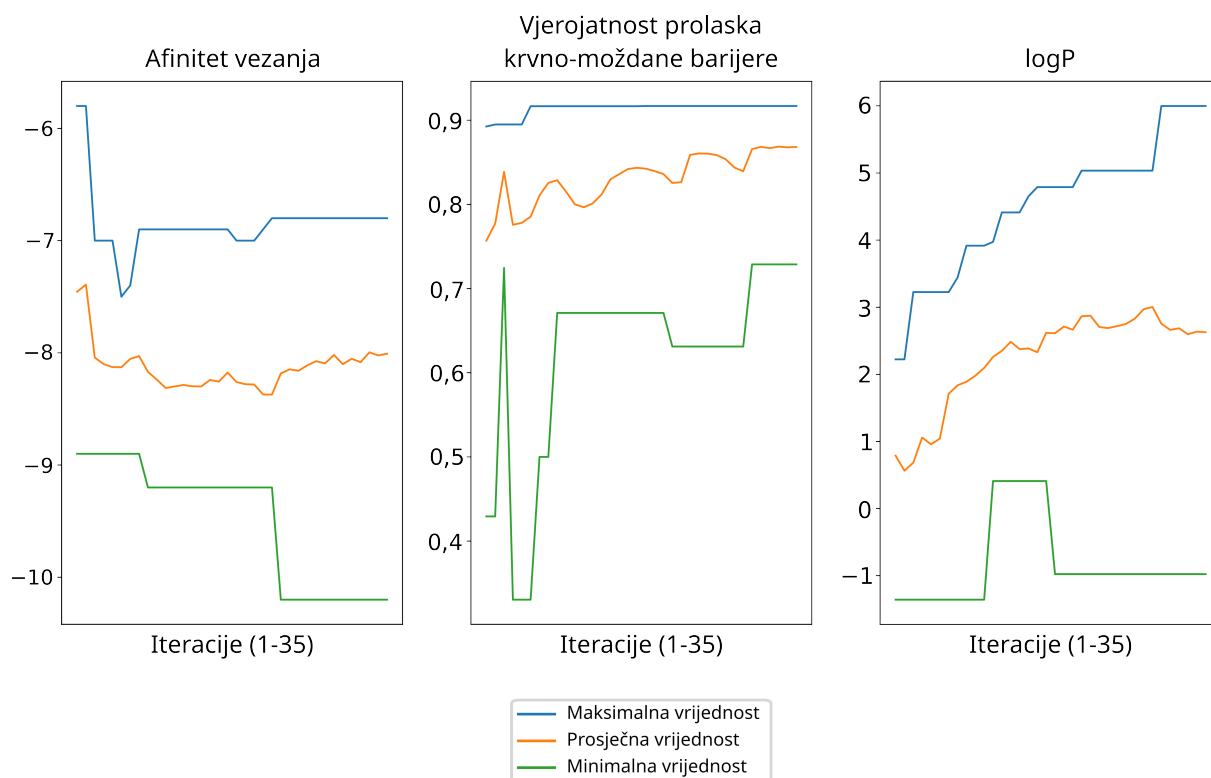
- Veličina turnirske selekcije: 2, 5
- Vjerojatnost mutacije: 1%, 5%, 20%.

Isprobane su sve moguće kombinacije parametara, što je rezultiralo sa šest ispitnih slučajeva. Moguća je provedba 5 vrsta mutacija, pri čemu su 4 preuzete iz prethodnog istraživanja (vidjeti potpoglavlje 3.1.3.), te je uveden dodatni tip mutacije u obliku zamjene nasumične aminokiseline alaninom, tzv. skeniranje alaninom (eng. alanine scan) [132]. Tijekom procesa evolucije, praćeni su podaci o minimalnoj, prosječnoj i maksimalnoj vrijednosti svakog od triju kriterija kao indikator napredovanja populacije u željenom smjeru. Ograničenje duljine peptida primijenjeno je isključivo pri stvaranju početne populacije, kada su stvarani peptidi duljina između 3 i 50 aminokiselina. U ostatku optimizacije nije primijenjeno nikakvo ograničenje na duljinu.

Za konačnu populaciju provjerena je razdioba sličnosti parova peptida nakon poravnania algoritmom Needleman-Wunsch [186], razdioba aminokiselina, te zastupljenost pojedinih duljina u populaciji, što je u skladu s metodologijom opisanom u ostatku ove disertacije. Zaključno, iz konačne populacije peptida nasumično su odabrane 4 molekule koje su uspoređene s patentiranim peptidnim inhibitorima beta-sekretaze 1 iz literature [143], te je provedena analiza u odnosu na Lipinskijevih 5 pravila [159, 160].

5.2. Rezultati i diskusija

Praćenjem krivulja napredovanja kroz iteracije (slika 5.2) i usporedbom kvalitete završnih populacija primijećeno je da veličina turnira 2 i stopa mutacije od 20% daju najbolje rezultate, pa su navedeni parametri primijenjeni u ostatku studije. Promotri li se prosječne vrijednosti kvalitete, vidljiva je tendencija povećanja vjerojatnosti prolaska krvno-moždane barijere i vrijednosti logP, dok je istovremeno primjetna tendencija ka smanjivanju procijenjene energije vezanja za metu, što odgovara željenom smjeru optimizacije molekula. Stvaranjem složenijih molekula možemo vrlo dobro prilagoditi afinitet vezanja za metu, no ta složenost istovremeno rezultira većim molekulama koje će teže prijeći krvno-moždanu barijeru. Postojanje takve oprjeke, gdje poboljšanje jednog kriterija nužno dolazi na štetu drugoga, vidljivo je u izmjeničnim periodima rasta i pada prosječnih vrijednosti kroz iteracije.



Slika 5.2: Prikaz napredovanja optimizacije kroz praćenje minimalne, maksimalne i prosječne vrijednosti svakog kriterija na cijeloj populaciji.

Četiri nasumično odabrana peptida iz konačne populacije pokazuju znatno bolji afinitet vezanja za enzim beta-sekretaze 1 u odnosu na patentirane inhibitore iz literature (tablica 5.1), povećavajući ga prosječno za 42 postotna boda. Također, vidljivo je i znatno poboljšanje vrijednosti logP i vjerojatnosti prolaska krvno-moždane barijere. Valja napomenuti kako postojeći patentirani inhibitori nisu nužno dizajnirani za samostalan prijelaz krvno-moždane barijere, pa to treba uzeti u obzir prilikom usporedbe. Prosječna sličnost parova peptida u populaciji iznosi 34,1% (slika 5.3), a vidljiva je i tendencija algoritma ka dizajniranju sekvenci s leucinom, prolinom i triptofanom, te minimizaciji korištenja glicina, arginina i treonina (slika 5.4). Iako tijekom optimizacije algoritam ni na koji način nije bio ograničen ili usmjeravan prema kraćim sekvencama, prosječna duljina peptida u konačnoj populaciji iznosi 6 aminokiselina (slika 5.5), što se može objasniti činjenicom da jednostavnije molekule lakše prolaze krvno-moždanu barijeru. Promotrimo li odabrane peptide u odnosu na Lipinskijeva pravila (tablica 5.2), jasno je da niti jedan peptid nema molekularnu masu ispod 500 Da, što je očekivano s obzirom na to da su peptidi veći i složeniji u odnosu na male molekule koje su se tradicionalno koristile kao lijekovi. Što se ostalih kriterija tiče, većinom su zadovoljeni, s iznimkom peptida WLWWWPF i vrijed-

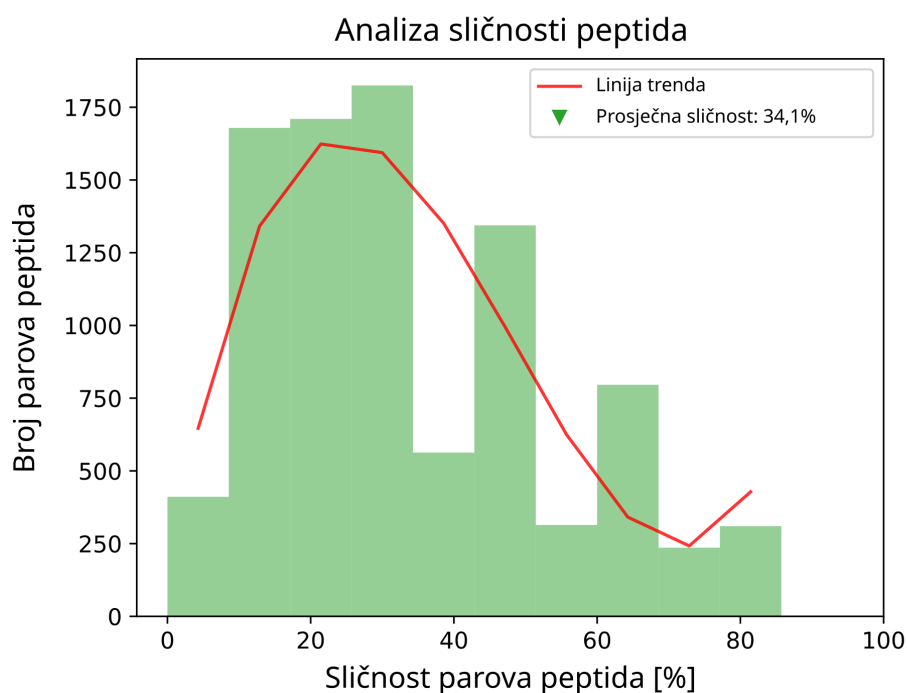
nosti logP, te peptida MHLAFKW i broja donora vodikovih veza.

Rezultati daju naslutiti kako su dizajnirani peptidi vrlo dobri kandidati za daljnje eksperimentalno proučavanje i optimizaciju, te predstavljaju kvalitetne vodeće spojeve za dizajn inhibitora beta-sekretaze 1.

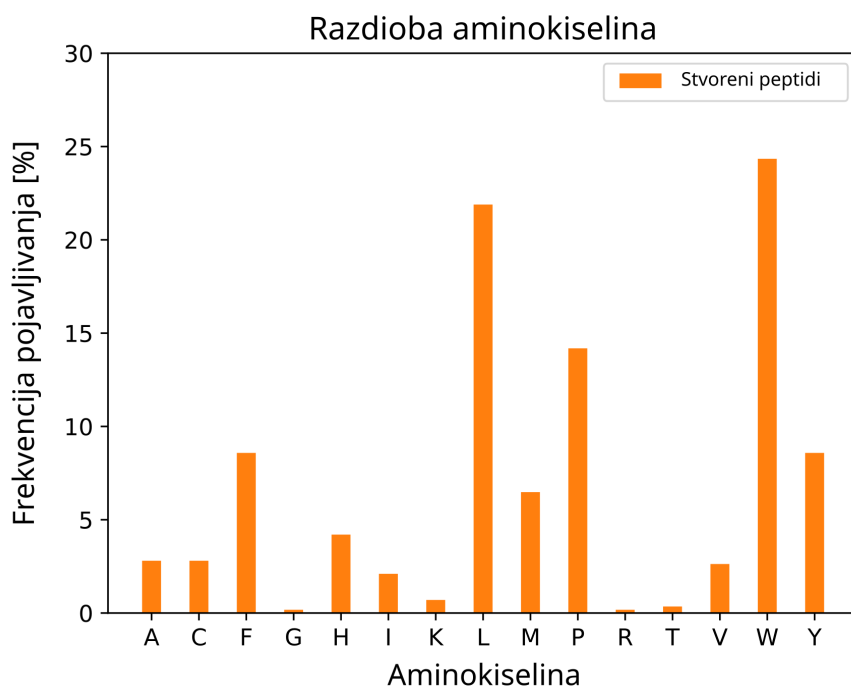
Tablica 5.1: Prikaz nasumično odabranih peptida s prve nedominirane fronte konačne populacije molekula i usporedba s patentiranim inhibitorima enzima beta-sekretaze 1 iz literature. Preuzeto iz [189] i prevedeno.

Peptid	Afinitet vezanja za beta-sekretazu 1 (↓)	Vjerojatnost prolaska krvno-moždane barijere (↑)	logP (↑)
WLWWWPF	-9,4	75,4%	6,00
WWWPF	-10,2	72,9%	4,26
GIHAYWT	-7,8	91,7%	-0,98
MHLAFKW	-7,9	89,7%	1,19
NEESMYCRLLGIGCG ¹	-6,1	14,8%	-7,36
PEESLYCRLLALGCG ¹	-6,3	10,6%	-5,13

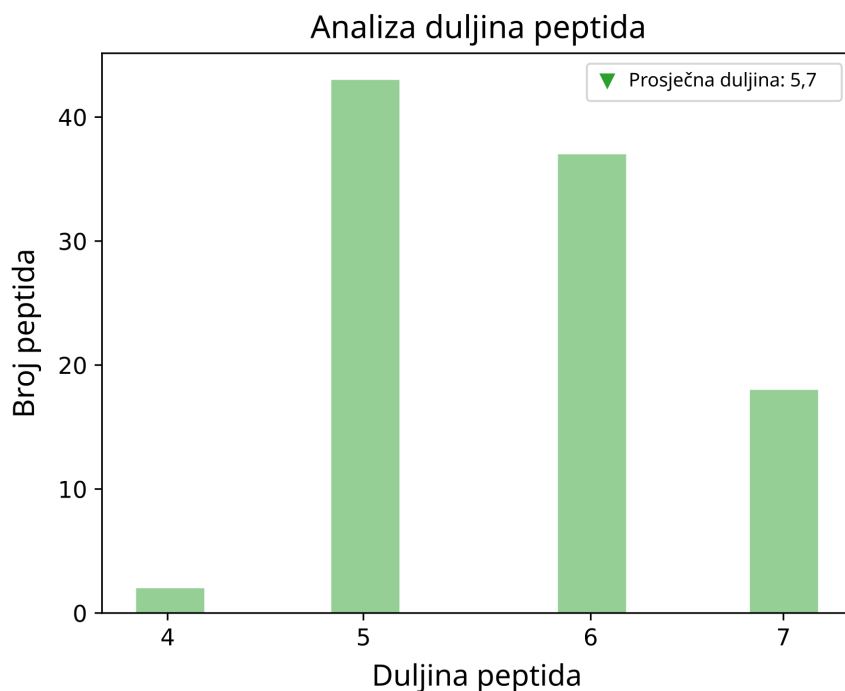
¹ Peptidni inhibitori beta-sekretaze 1 prijavljeni u patentu [143].



Slika 5.3: Analiza sličnosti parova peptida u konačnoj populaciji. Peptidi prosječno dijele manje od 35% sličnosti.



Slika 5.4: Razdioba aminokiselina u konačnoj populaciji. Vidljive su povećane koncentracije leucina, prolina i triptofana, te smanjeno korištenje glicina, arginina i treonina.



Slika 5.5: Histogram duljina peptida u konačnoj populaciji. Vidljiva je tendencija algoritma ka stvaranju sekvenci kraćih od 10 aminokiselina, pri čemu prosječna duljina iznosi 6 aminokiselina.

Tablica 5.2: Analiza odabranih peptida prema Lipinskijevim pravilima.

Peptid	Molekularna masa [Da] (≤ 500)	Donori vodikovih veza (≤ 5)	Akceptori vodikovih veza (≤ 10)	logP (≤ 5)
WLWWWPF	1102	4	0	6,00
WWWPF	803	3	0	4,26
GIHAYWT	829	5	5	-0,98
MHLAFKW	913	6	2	1,19

6. EVOLUCIJSKA OPTIMIZACIJA KNJIŽNICA PEPTIDA

Rezultati istraživanja opisani u ovom poglavlju objavljeni su u sljedećem znanstvenom radu: M. Njirjak, E. Otović, D. Kalafatovic, and G. Mauša. A multi-library approach to parallelised sequence space exploration: Streamlining library design and search space coverage. Computers in Biology and Medicine, 198:111207, 2025. [190].

Dizajniranje peptida koji će uspješno utjecati na određeni kemijski put iznimno je složen proces. Ako govorimo o dizajnu lijekova, u obzir valja uzeti ponašanje peptida u organizmu u ovisnosti o kriterijima ADME [265], njegovu potencijalnu toksičnost prema drugim metama [105], mogućnost peptida da uopće dođe do mete na koju bi se trebao vezati [194], i slično. Složenost također ne izostaje ni prilikom dizajna industrijskih molekula. Peptid mora imati primjereno djelovanje na određeni kemijski proces, stabilnu strukturu, dobru skalabilnost i odgovarajuću cijenu [93]. Općenito, dizajn bilo kakvog novog peptida za specifičnu primjenu je vrlo složen proces koji još uvijek zahtijeva iscrpnu eksperimentalnu provjeru tisuća molekula [254]).

Tradicionalni pristupi su skupoj eksperimentalnoj provjeri doskakali koristeći postojeće aktivne sekvence kao polazište za dizajn novih [36, 176], oslanjajući se na specifična pravila koja uvjetuju određenu aktivnost; primjerice, pozitivni naboj i hidrofobnost za antimikrobno djelovanje [1, 117]. Međutim, koliko god da navedeni pristup pomaže pri smanjenju skupa sekvenci koje je potrebno laboratorijski vrednovati, istovremeno može stvoriti neželjenu pristranost prema određenom dijelu kemijskog prostora. Dosad se testiranje novih peptida umnogome oslanjalo na sintezu i provjeru individualnih spojeva.

Međutim, kako dizajn peptida kombinatorno predstavlja NP-težak problem [209], iscrpno testiranje cijelog kemijskog prostora za određene duljine peptida vrlo brzo postaje neizvedivo, uzmemo li u obzir da za sekvencu s 15 proteinogenih aminokiselina postoji $\sim 3,3 \cdot 10^{19}$ mogućih peptida. Ideja grupiranja sličnih sekvenci u knjižnice stoga je donijela mnoge promjene u načinima sinteze i provjere spojeva [50, 89, 92, 119, 122, 174, 213, 214]. Naime, sinteza knjižnice peptida omogućuje da odjednom stvorimo na milijune spojeva u istoj otopini, koristeći tehniku sinteze podjele i miješanja (eng. split and pool synthesis) [79]. Tako je moguće istovremeno sintetizirati i provjeriti mnogostruko više spojeva nego kada bismo svakome pristupili zasebno. Navedeno može biti posebno korisno ako provodimo laboratorijsku evoluciju molekula, pri čemu je u svakoj iteraciji potrebno testirati različite varijante izvornog spoja [220]. Međutim, sinteza više sličnih molekula u istoj otopini, iako brža i učinkovitija, često stvara probleme u kasnijoj analizi rezultata, s obzirom na to da se ne može lako zaključiti koji konkretan spoj je uzrok uočenom efektu. Razdvajanje knjižnice peptida na više podskupova, pri čemu svaki podskup predstavlja jednu manju knjižnicu, može stvoriti ravnotežu između bržeg i učinkovitijeg ispitivanja spojeva, te uspješnije kasnije analize. Međutim, pronaći kvalitetan način za podijeliti knjižnicu u podskupove predstavlja izazov, s obzirom na kombinatornu prirodu problema.

Gradeći na prethodnim uspjesima genetskog algoritma kojeg je razvila istraživačka grupa čiji sam član [123], u ovom poglavlju predstavljen je poboljšani genetski algoritam za razdvajanje knjižnica peptida u više podskupova, koji uz minimizaciju preklapanja po masi unutar podskupova i maksimizacije pokrića izvorne knjižnice, nastoji još i minimizirati međusobno preklapanje podskupova. Dodatno, kako bi se omogućila šira primjena algoritma, razvijena je metoda procjene preklapanja peptida po masi unutar podskupa, temeljena na algoritmu simuliranog kaljenja. Navedeno je omogućilo primjenu algoritma na knjižnice peptida proizvoljne veličine, koje inače ne bi bilo moguće optimizirati.

6.1. Metodologija

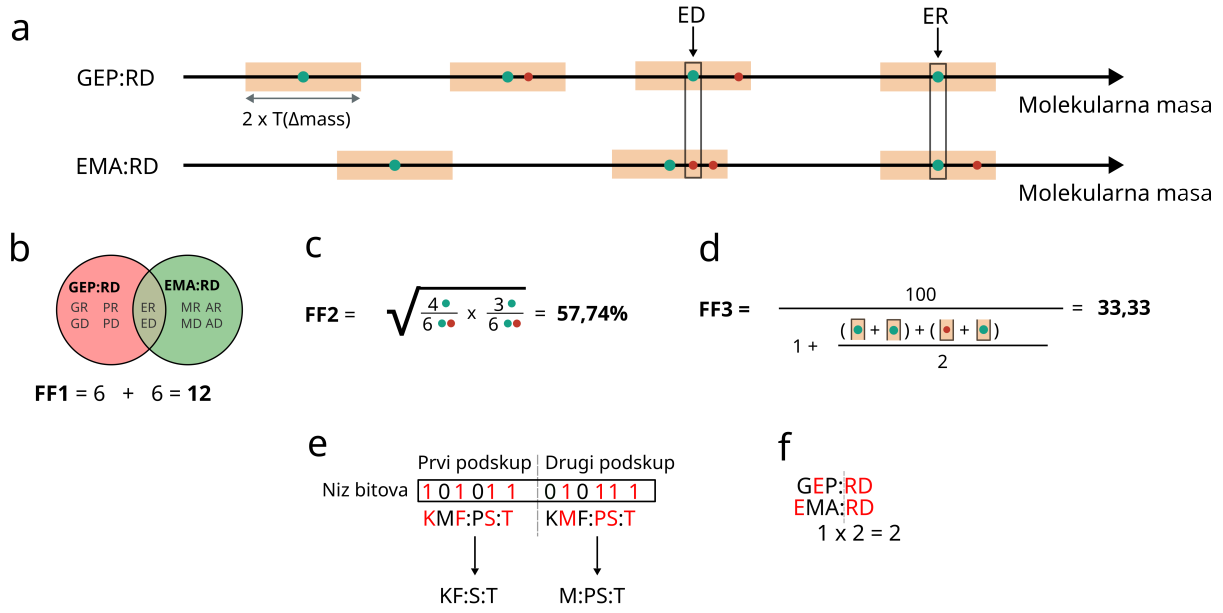
6.1.1. Evolucijski pristup optimizaciji knjižnica peptida

Kao jezgra pristupa, odabran je genetski algoritam NSGA-II [61] uz predstavu knjižnica peptida uvriježenim formatom FASTA [18], s dodatkom dvotočja za označavanje

različitih pozicija u sekvenci. Tako, primjerice, „GEP:RD” označava knjižnicu dipeptida u kojoj su na prvom mjestu sekvence moguće aminokiseline glicin, glutaminska kiselina i prolin, dok su na drugom mjestu moguće arginin i asparaginska kiselina. Sveukupno, knjižnica se sastoji od 6 peptida: GR, ER, PR, GD, ED i PD. Prethodno razvijeni pristup oslanjao se na optimizaciju jednog podskupa koristeći dva kriterija: stvaranje podskupa unutar kojeg se peptidi što manje preklapaju po masi, te zadržavanje što većeg broja peptida iz izvorne knjižnice [123]. Preklapanje po masi nastoji osigurati da peptidi budu lako razdvojivi u kasnijoj analizi kromatografijom uparenom s masenom spektrometrijom i matematički je definirano nejednadžbom 6.1, pri čemu M_i i M_j predstavljaju molekularne mase dvaju peptida koji se uspoređuju, a $T(\Delta mass)$ predstavlja minimalnu razliku masa, definiranu od strane korisnika algoritma. Veći $T(\Delta mass)$ implicira da se dvije molekule moraju više razlikovati u masi kako bi se smatralo da se ne preklapaju (više zelenih nego crvenih točaka na slici 6.1.a). Primjenom navedenog kriterija u Algoritmu 1, može se izračunati broj preklapanja u podskupu, a koristeći geometrijsku sredinu postupak se može generalizirati i primijeniti na proizvoljan broj podskupova (slika 6.1.c, jednadžba 6.2). Ako bi minimizacija preklapanja po masi bio jedini kriterij optimizacije, tada bi algoritam sasvim sigurno stvarao prazne podskupove, s obzirom na to da oni ne bi sadržavali niti jedno preklapanje i predstavljali bi globalni optimum. Međutim, takvi podskupovi nisu korisni, pa se kao protuteža kriteriju preklapanja po masi koristi procjena pokrivenosti izvorne knjižnice. Ovaj kriterij dodatno navodi optimizaciju u smjeru podskupova s većim brojem zadržanih peptida, nastojeći što više obuhvatiti kemijski prostor izvorne knjižnice koju optimiziramo (slika 6.1.b). Za proizvoljan broj podskupova, broj peptida se može izračunati jednadžbom 6.3, gdje K predstavlja broj podskupova, R je broj pozicija u sekvenci, a $r_{n,i}$ predstavlja broj aminokiselina-kandidata na i -toj poziciji u n -tom podskupu. Međutim, svodenje izvorne knjižnice peptida na jedan podskup neminovno dovodi do uklanjanja pojedinih sekvenci kako bi se smanjilo preklapanje i stvorilo podskupove pogodne za kasniju karakterizaciju, što potencijalno može ukloniti bitne molekule.

$$|M_i - M_j| < T(\Delta mass) \quad (6.1)$$

$$FF_2 = \sqrt[n]{k_1 \cdot k_2 \cdot \dots \cdot k_n} \quad (6.2)$$



Slika 6.1: Shematski prikaz (a) načina računanja broja peptida koji se ne preklapaju po masi unutar jednog podskupa (zelene točke nasuprot crvenim točkama), te broja identičnih peptida među dvama podskupovima (ED i ER). Prikazani su i postupci računanja (b) ukupnog broja peptida kojeg podskupovi sadrže, (c) srednjeg postotka peptida koji se ne preklapaju po masi, te (d) pokazatelja međusobne sličnosti podskupova. Vidljiv je i (e) način kodiranja knjižnica u algoritmu i njegova konverzija u podskupove, te (f) način računanja broja peptida koji se pojavljuju u dva podskupa. Preuzeto iz [190] i prevedeno.

Ulaz: Knjižnica peptida u posebnom formatu, npr. „GRS:EA:RA:DG”.
Tolerancija po masi $T(\Delta mass)$.

Izlaz: Broj peptida koji se ne preklapaju po masi.

```
niz_peptida ← stvori_peptide(knjižnica);
sortiraj_uzlazno_po_masi(niz_peptida);
zadnji_peptid ← niz_peptida[0];
brojač ← 1;
for i: 0 → veličina(niz_peptida) - 1 do
    if masa(niz_peptida[i]) - masa(zadnji_peptid) ≥ T(Δmass) then
        zadnji_peptid ← niz_peptida[i];
        brojač ← brojač + 1;
    end
end
vрати brojač;
```

Algoritam 1: Iscrpno pretraživanje broja peptida koji se ne preklapaju po masi u ulaznoj knjižnici peptida. Preuzeto iz [190] i prevedeno.

$$FF_1 = \sum_{n=1}^K \prod_{i=1}^R r_{n,i} \quad (6.3)$$

Kako bi se to spriječilo, predložen je algoritam za poboljšanu optimizaciju knjižnica peptida koji knjižnicu nastoji razdvojiti u više disjunktih podskupova. Uz korištenje prethodno opisanih dvaju kriterija optimizacije, uvodimo dodatni kriterij vrednovanja sličnosti između podskupova. Na taj način, peptidi koji bi inače bili odbačeni, mogu se razmjestiti u druge skupove i tako biti zadržani. Broj preklapanja među dvama podskupovima moguće je izračunati množeći brojeve identičnih aminokiselina koje se pojavljuju na istim mjestima u oba skupa. Tako, primjerice, podskupovi „GEP:RD” i „EMA:RD” imaju glutaminsku kiselinu na prvoj poziciji, te arginin i asparaginsku kiselinu na drugoj, što rezultira dvjema zajedničkim sekvencama: ER i ED (slika 6.1.b,f). U slučaju podskupova koji na makar jednoj od pozicija ne dijele aminokiseline, npr. „SKW:AD” i „SKW:GSK”, ne postoje zajedničke sekvence i broj preklapanja je 0. Prosječni broj preklapanja među svim parovima podskupova tada se obrađuje eksponencijalno padajućom funkcijom: $f(x) = 100/(1+x)$ i rezultira brojem između 0 i 100, pri čemu više vrijednosti indiciraju da se podskupovi manje prekapaju (slika 6.1.a,d). Eksponencijalna funkcija primjenjuje se kako bi se porast preklapanja kod manjih podskupova kaznio više nego kod većih. Primjerice, ako se broj preklapanja sekvenci poveća s 1 na 2, to predstavlja 100%-tno povećanje i treba se jako kazniti, dok je povećanje s 501 na 502 sekvence ipak manje značajno. U konačnici, prosječno preklapanje među podskupovima može se izračunati jednadžbom 6.4, pri čemu n predstavlja broj podskupova, a $p_{i,j}$ broj sekvenci koji je zajednički i -tom i j -tom podskupu. Navedeno je dodatno formalizirano u Algoritmu 2. Cilj je usmjeriti optimizaciju ka podskupovima koji se što manje prekapaju.

$$FF_3 = \frac{100}{1 + \frac{\sum_{i=1}^n \sum_{j=i+1}^n p_{i,j}}{n}} \quad (6.4)$$

Interna predstava rješenja kojom algoritam rukuje jest niz bitova, pri čemu je broj bitova definiran brojem aminokiselina u knjižnici peptida pomnožen brojem podskupova koji se traže (slika 6.1.e). Algoritam tada nastoji pronaći niz bitova koji, preklopljen preko izvorne knjižnice peptida, definira podskupove sa što povoljnijim svojstvima u odnosu na prethodno definirane kriterije vrednovanja. Veličina prostora pretrage, i posljedično složenost optimizacijskog problema, tada iznosi 2^n , gdje n predstavlja duljinu knjižnice peptida pomnoženu s brojem podskupova. Svaka pozicija podskupa uvijek mora sadržavati minimalno jednu aminokiselinu, pa u tom smislu gubitak pozicije u podskupu tijekom op-

Ulaz: Niz podskupova u posebnom formatu, npr. [„GRS:EA:RA:DG”, „RA:SW:E:G”].

Izlaz: Metrika preklapanja podskupova

```

for  $i$ :  $0 \rightarrow \text{veličina}(\text{niz\_podskupova}) - 1$  do
    for  $j$ :  $i + 1 \rightarrow \text{veličina}(\text{niz\_podskupova}) - 1$  do
         $\text{identični\_peptidi} \leftarrow \text{prebroji\_identične\_peptide}(\text{niz\_podskupova}[i],$ 
             $\text{niz\_podskupova}[j]);$ 
         $\text{dodaj\_u\_niz}(\text{niz\_za\_brojanje}, \text{identični\_peptidi});$ 
    end
end

 $\text{prosječno\_preklapanje} \leftarrow \text{suma}(\text{niz\_za\_brojanje}) / \text{veličina}(\text{niz\_za\_brojanje});$ 
vrați  $100 / (1 + \text{prosječno\_preklapanje});$ 

```

Algoritam 2: Računanje numeričkog pokazatelja preklapanja podskupova. Više vrijednosti indiciraju manji broj preklapanja. Preuzeto iz [190] i prevedeno.

timizacije nije dozvoljen. Tri opisana kriterija koja navode optimizaciju su u međusobnoj oprjeci, s obzirom na to da veća pokrivenost izvorne knjižnice peptida neminovno uvjetuje povećanje preklapanja po masi i veću međusobnu sličnost podskupova. Zbog toga, u većini slučajeva je nemoguće pronaći podjelu koja savršeno zadovoljava sva tri kriterija. Rješenja se stoga dijele u rangirane skupove koji se nazivaju nedominiranim frontama, a ona koja sadrži najkvalitetnija rješenja naziva se Paretova fronta [60]. Izlaz algoritma tada nije jedno predloženo rješenje, već skup rješenja različitih karakteristika između kojih korisnik može birati, u ovisnosti o potrebama istraživanja i zahtjevima eksperimenta.

6.1.2. Procjena broja nepreklapajućih peptida simuliranim kaljenjem

Izračun broja peptida koji se ne preklapaju po masi koristeći iscrpno pretraživanje opisano Algoritmom 1 zahtijeva izračunavanje masa svih peptida u podskupu, što za velike podskupove može biti vremenski i memorijski neizvedivo. To značajno ograničava veličinu knjižnica peptida na koju se algoritam može primijeniti. Kako bi se omogućila obrada knjižnica peptida proizvoljne veličine, osmislili smo surogat funkciju navedenog kriterija [4], formulirajući problem pronalaženja idućeg peptida koji se ne preklapa s trenutnim kao zasebni optimizacijski problem: *Neka S predstavlja podskup peptida izvorne knjižnice, a p predstavlja trenutni peptid od interesa. Pronađi peptid q takav da vrijedi $q = \arg \min_{x \in S} [\text{masa}(x) - \text{masa}(p) - T(\Delta \text{mass})] \wedge (\text{masa}(q) - \text{masa}(p) - T(\Delta \text{mass})) \geq 0$. Peptid p*

na početku predstavlja peptid s najmanjom masom u podskupu, čija se identifikacija može vrlo lako napraviti uzimajući najlakšu aminokiselinu na svakoj od pozicija. Iterativnom primjenom algoritma simuliranog kaljenja tada se traži peptid q , nakon čega p postaje q . Postupak se ponavlja sve dok više ne možemo pronaći q koji zadovoljava prethodno opisane uvjete (Algoritam 3).

Ulaz: Podskup peptida u posebnom formatu, npr. „GRS:EA:RA:DG”.

Tolerancija po masi $T(\Delta mass)$.

Početna temperatura `početna_temperatura`.

Stopa hlađenja R .

Izlaz: Procijenjeni broj peptida koji se ne preklapaju po masi.

```

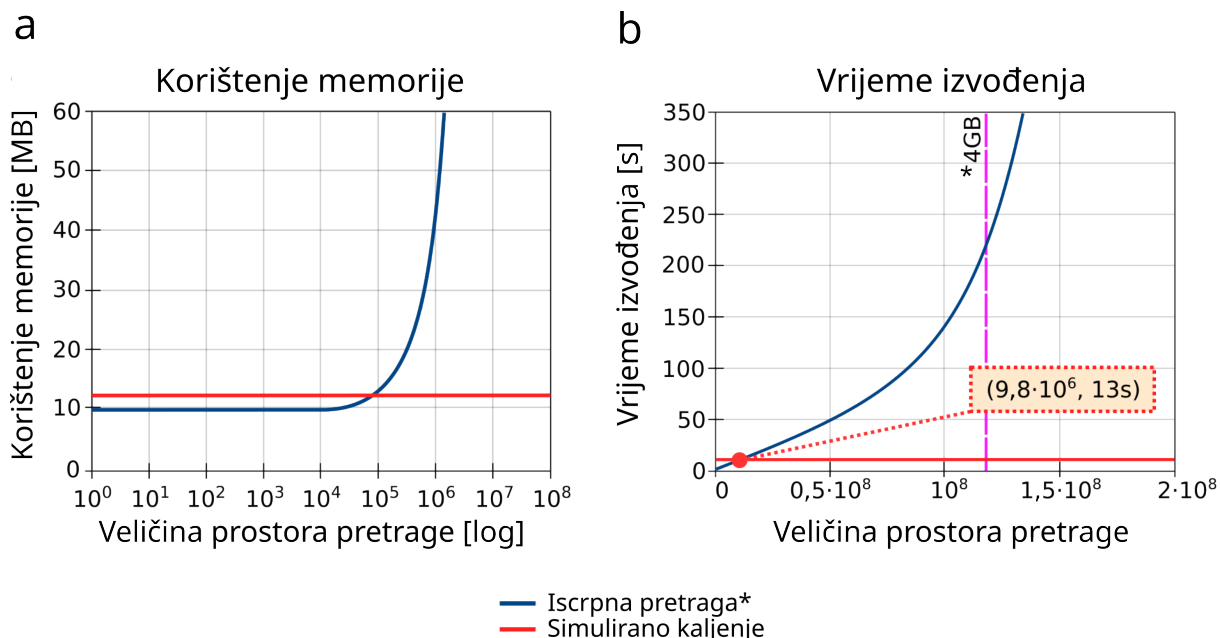
zadnji_peptid ← dohvati_najlakši_peptid(knjižnica);
brojač ← 1;
while true do
    trenutni_peptid ← dohvati_nasumični_peptid(knjižnica);
    trenutna_temperatura = početna_temperatura;
    while trenutna_temperatura ≥ 1,0 do
        modificirani_peptid ← modificiraj(trenutni_peptid);
        if vjerojatnost_prihvatanja(modificirani_peptid, trenutni_peptid,
            trenutna_temperatura, zadnji_peptid,  $T(\Delta mass)$ ) >
            nasumična_vrijednost(0, 1) then
            trenutni_peptid ← modificirani_peptid;
        end
        trenutna_temperatura = trenutna_temperatura * (1 - R);
    end
    if zadnji_peptid == trenutni_peptid then
        break;
    end
    zadnji_peptid ← trenutni_peptid;
    brojač ← brojač + 1;
end
vrati brojač;

```

Algoritam 3: Procjena broja peptida koji se ne preklapaju po masi metodom simuliranog kaljenja. Algoritam se koristi kao surogat model iscrpnom pretraživanju kako bi se omogućila optimizacija u slučajevima u kojima prostor pretrage sadrži više od $9,8 \cdot 10^6$ mogućih rješenja. Prosječna greška procjene u odnosu na stvarni broj preklapanja ne prelazi 3%. Preuzeto iz [190] i prevedeno.

S obzirom na to da iscrpno pretraživanje posjeduje vremensku i memorijsku složenost najgoreg slučaja $O(kn)$, pri čemu n predstavlja broj pozicija u sekvenci, a k broj aminokiselina-kandidata na svakoj od pozicija u peptidnom lancu, pretpostavili smo da

postoji granični slučaj veličine podskupa za koji vrijeme izvođenja i/ili količina potrebne memorije iscrpnog pretraživanja značajno nadilazi onu simuliranog kaljenja. Navedeno je istraženo varirajući n i k , te uspoređujući zahtjeve za resursima i točnost obaju algoritama (slika 6.2). Zahtjev za memorijom i vrijeme izvođenja iscrpne pretrage značajno ovise o



Slika 6.2: Ispitivanje (a) memorijskih i (b) vremenskih zahtjeva za algoritme izračunavanja broja peptida koji se ne preklapaju po masi metodama iscrpnog pretraživanja (plavo) i simuliranog kaljenja (crveno). Izbor metode sukladno kraćem vremenu izvršavanja osigurava ispunjavanje memorijskih zahtjeva. Preuzeto iz [190] i prevedeno.

veličini prostora pretrage, dok su kod simuliranog kaljenja oni poprilično konstantni. Proizvoljno odabrana granica količine memorije nasumičnog pristupa od 4 GB (označena ljubičastom linijom na slici 6.2.b) definira maksimalnu veličinu podskupa na koju se može primijeniti algoritam iscrpnog pretraživanja, s obzirom na to da je pristup ograničen količinom dostupne radne memorije. Pod istim uvjetima, algoritam simuliranog kaljenja nema ograničenja. Oba algoritma imaju vrijeme izvođenja od 13 sekundi na standardnoj konfiguraciji računala s 8 GB radne memorije nasumičnog pristupa i dvojezgrenom procesoru radne frekvencije oko 2 GHz kada veličina prostora pretrage sadrži $9,8 \cdot 10^6$ mogućnosti podjele. Daljnjim povećanjem veličine prostora pretrage, ušteda resursa od strane simuliranog kaljenja postaje sve izraženija. Posljedično, promotrimo li grafove ovisnosti na slici 6.2, izbor algoritma za izračunavanje preklapanja po masi može se napraviti isključivo na temelju kraćeg vremena izvođenja, što će osigurati i memorijske zahtjeve.

Sukladno tome, algoritam iscrpnog pretraživanja se koristi ako je složenost problema niža od $9,8 \cdot 10^6$ mogućih rješenja, a u suprotnom se koristi algoritam simuliranog kaljenja. S obzirom na to da algoritam simuliranog kaljenja predstavlja surogat model iscrpnog pretraživanja, on neminovno uvodi greške u proces izračuna broja peptida koji se ne preklapaju po masi. Analize su pokazale da prosječna greška ne prelazi 3% (tablica B.1), odnosno za podskup veličine 100 peptida, procijenjeni broj peptida koji se ne preklapaju po masi bi od stvarnog broja odudarao za najviše 3 peptida, što smatramo prihvatljivim. Izračun algoritmom simuliranog kaljenja omogućava optimizaciju knjižnica peptida proizvoljne veličine, što ne bi bilo moguće u slučaju isključivog korištenja algoritma iscrpne pretrage.

6.1.3. Parametrizacija algoritma

Optimizacijski slučajevi mogu široko varirati u složenosti, pri čemu veće knjižnice peptida koje treba razdvojiti u značajniji broj podskupova zahtijevaju i više računalnih resursa. Sukladno tome, postavljanje fiksni parametara algoritma za sve optimizacijske slučajeve bi u nekim situacijama rezultiralo bespotrebnim trošenjem računalnih resursa kada je dovoljno dobro rješenje već pronađeno, dok bi u drugim slučajevima optimizacija završila prije nego se kvalitetno rješenje uopće pronađe. Zbog toga, veličina populacije određuje se dinamički sukladno jednadžbi 6.5, pri čemu n predstavlja duljinu knjižnice peptida pomnoženu s brojem podskupova.

$$p(n) = \begin{cases} 100, & n \leq 1 \\ \lfloor 100 + 900 \cdot (n - 1)/200 \rfloor, & 2 \leq n \leq 200 \\ 1000, & n > 200 \end{cases} \quad (6.5)$$

Maksimalan broj iteracija algoritma postavlja se na dvostruku vrijednost veličine populacije, s gornjom granicom od 1000. Međutim, on ne mora uvijek biti i izvršen, s obzirom na to da je implementiran mehanizam ranog zaustavljanja ako kvaliteta populacije počne stagnirati. Naime, prilikom izvršavanja algoritma prate se kretanja veličina površina omeđenih trima kriterijima vrednovanja kako slijedi:

- Površina između ukupnog broja peptida i različitosti peptida po masi

- Površina između ukupnog broja peptida i sličnosti podskupova
- Površina između različitosti peptida po masi i sličnosti podskupova.

Neka je Q multiskup rješenja na Paretovoj fronti. Za par funkcija vrednovanja FF_a i FF_b , rješenja u Q najprije se sortiraju uzlazno po FF_a nakon čega se primjenjuje jednadžba 6.6 za izračunavanje površine ispod krivulje koristeći pravilo trapeza [275]. Ako se niti jedna od triju površina ne poveća u 100 uzastopnih iteracija algoritma, izvođenje se prekida.

$$P = \frac{1}{2} \sum_{i=2}^{|Q|} (Q_{i,FF_a} - Q_{i-1,FF_a})(Q_{i,FF_b} + Q_{i-1,FF_b}) + Q_{1,FF_a} Q_{1,FF_b} \quad (6.6)$$

U svakoj iteraciji, broj novostvorenih rješenja jednak je veličini populacije, a broj peptida u turnirskoj selekciji postavljen je na 5% veličine populacije. Posljedično, veličina populacije, maksimalan broj iteracija, broj novih peptida u svakoj iteraciji i veličina turnira direktno ovise o složenosti optimizacijskog problema. Preostali parametri uključuju vjerojatnost rekombinacije, koja iznosi 90% i vjerojatnost mutacije postavljenu na 5%. Rekombinacija se odvija koristeći jednu točku prijeloma, a u slučajevima kada se ona ne izvršava, provodi se kloniranje jednog od odabranih roditelja. Mutacija se provodi obrtanjem bita (eng. bit flip). Algoritam je parametriziran sukladno prethodnom istraživanju grupe čiji sam član [196].

6.1.4. Ispitni slučajevi

Kako bi se detaljno ispitala funkcionalnost predloženog algoritma i njegova sposobnost da razdvoji knjižnica peptida u više podskupova, provedeno je nekoliko ispitnih slučajeva. Najprije, pristupljeno je optimizaciji knjižnice „SER:WAG:IR” u tri ispitna slučaja: razdvajanje u jedan, dva i tri podskupa. Pri usporedbi rješenja, posebna pozornost obraćena je na broj rješenja koji se pojavljuju na Paretovoj fronti i na posebnosti svakog od njih u odnosu na ostale. Promotrene su i prednosti razdvajanja knjižnice u više podskupova u odnosu na ispitni slučaj u kojemu se knjižnica svodi isključivo na jedan podskup. Šire ispitivanje mogućnosti algoritma provedeno je kroz optimizaciju knjižnica peptida varirajućeg obima i složenosti sa sljedećim ulaznim knjižnicama:

- „OY:FVYE:VG:WDL:TC:QTAHPV”

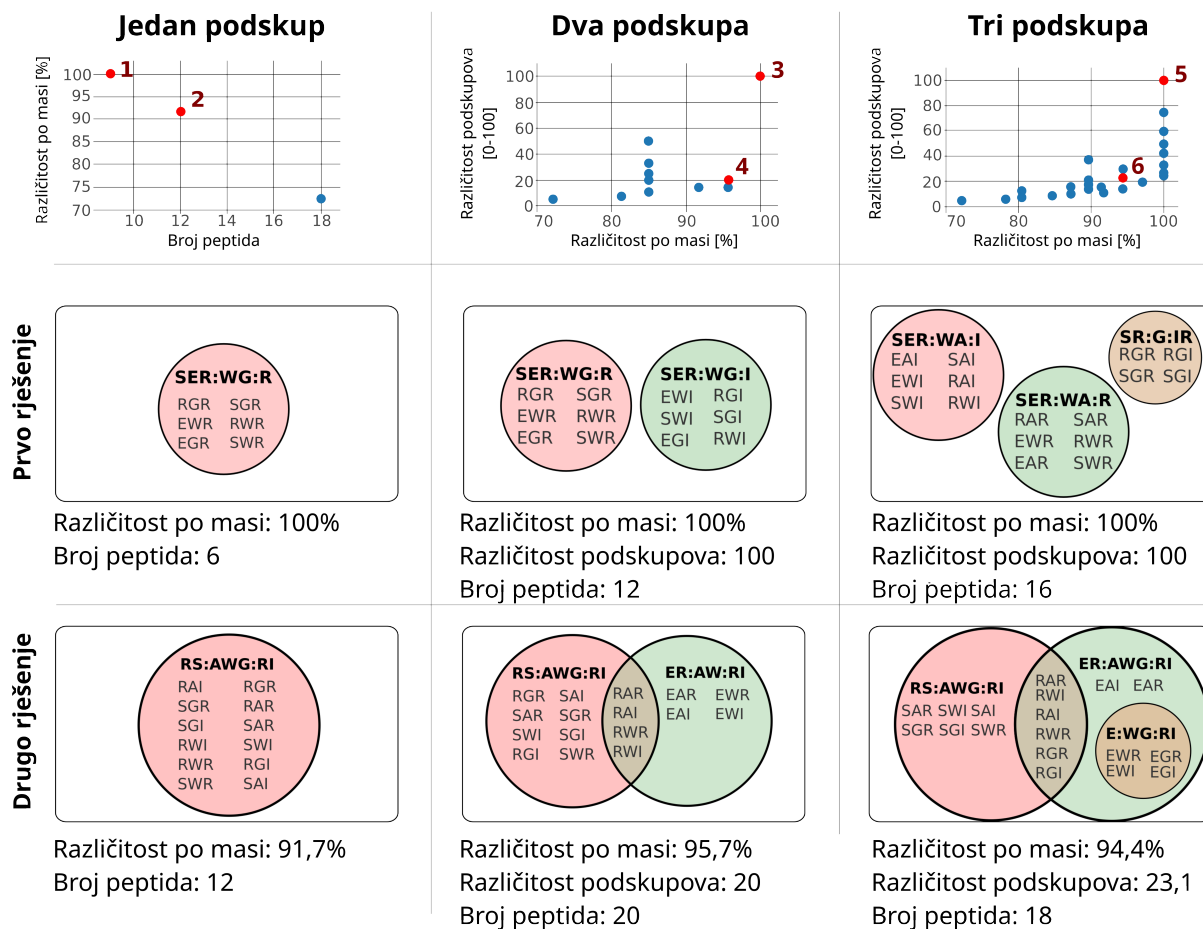
- „LWHTV:WP:HDICKSREAWQ:IGAHRFSVDY:IHMDVA:IQ”
- „CQYHRA:T:OAY:ENQLMD:CMQTILSRGEA:KCAQIENM:
PLQWIAYMRDVGSN:MFGS”
- „KHESLFPMQ:TKSIHYVMWQLNE:LDC:HE:CLSYTGVEQFAHMRKP:
VNMKG:SL:VFESWQCGYMD:FLANHMIDYRWSECKQPV:OTRM”.

Ispitana je mogućnost njihovog razdvajanja u 3, 4 i 5 podskupova, za svaki optimizacijski slučaj praćeni su parametri izvršavanja algoritma, a predstavljeno je i proizvoljno rješenje s Paretove fronte. Dodatno, napravljena je optimizacija knjižnice peptida „RYC:RAK:AGK–RWL:ARWQFPK:RWFAK:CRFG” koja sadrži 7560 peptida, 17 od kojih su sekvence iz baze podataka antimikrobnih peptida DRAMP [164].

Za svaki od ispitnih slučajeva navedenih na prethodnom popisu, provedena je i optimizacija naivnom metodom nasumičnog pretraživanja [5]. Navedeno je učinjeno kako bi se, kroz usporedbu s jednostavnijom metodom, rezultati algoritma stavili u kontekst. Kako bi usporedba dviju metoda bila vjerodostojna, metoda nasumičnog pretraživanja je u svakom ispitnom slučaju razmotrila isti broj rješenja kao i genetski algoritam. Potom su rješenja razdvojena u nedominirane fronte, nakon čega je zadržana isključivo Paretova fronta s koje je odabrano reprezentativno rješenje za usporedbu.

6.2. Rezultati i diskusija

Rezultati optimizacije za ispitne slučajeve knjižnice peptida „SER:WAG:IR” prikazani su na slici 6.3. Svaka točka u prvom redu slike 6.3 predstavlja optimizirano rješenje s Paretove fronte koje sadrži jedan, dva, ili tri podskupa. U slučaju optimizacije s jednim podskupom, možemo promatrati dva rješenja: „SER:WG:R” (točka „1”) i „RS:AWG:RI” (točka „2”). Oba su na Paretovoj fronti, no razlikuju se u karakteristikama. Podskup „SER:WG:R” zadržava 6 od 18 peptida iz izvorne knjižnice, no niti jedan peptid se ne preklapa po masi s nekima od ostalih. „RS:AWG:RI” zadržava nešto više peptida, njih 12, no određeni peptidi se preklapaju s ostalima po masi, stoga je različitost po masi 91,7%. Mana optimizacije jedne knjižnice je što se preklapanje po masi može izbjeći isključivo odbacivanjem sekvenci. U slučaju razdvajanja knjižnice u dva podskupa, moguće



Slika 6.3: Razdvajanje knjižnice peptida „SER:WAG:IR” u jedan, dva, i tri podskupa. Različitost po masi, međusobna različitost podskupova, te peptidi sadržani u podskupovima prikazani su za odabrana rješenja (označena crvenim točkama na grafovima). Vennovi dijagrami demonstriraju prednosti razdvajanje knjižnice peptida na veći broj podskupova. Preuzeto iz [190] i prevedeno.

je zadržati 12 peptida uz potpuno izbjegavanje preklapanja po masi i preklapanja između podskupova (točka „3”), što svakako predstavlja poboljšanje u odnosu na pristup s jednim podskupom. Moguće je zadržati i više sekvenci iz izvorne knjižnice, no uz prihvaćanje povećanja preklapanja (točka „4”). U ovom slučaju, broj podskupova je ograničavajući čimbenik, pa ako se on poveća s 2 na 3, vidimo da je moguće zadržati 16 peptida uz potpuno izbjegavanje oba tipa preklapanja (točka „5”). Naravno, i u ovom slučaju je moguće pronaći rješenje koje sadrži još više sekvenci, uz prihvaćanje smanjenja preostala dva kriterija (točka „6”). Pokrivenost izvorne knjižnice, odnosno broj zadržanih peptida, raste s porastom broja podskupova u koji se ona razdvaja. Sukladno tome, razvidno je da bolju pokrivenost izvorne knjižnice omogućuje povećanje u sličnostima po masi i međusobnim sličnostima podskupova, što je dosljedno s oprijekom u kriterijima opisanom

u potpoglavlju 6.1.1.. Na Paretovim frontama postoji mnogo rješenja pa istraživači mogu izabrati specifično rješenje sukladno potrebama eksperimenta.

Ispitivanje optimizacije složenijih knjižnica peptida predstavljenom je tablicama od 6.1 do 6.5. Složenost svakog ispitnog slučaja naznačena je veličinom prostora pretraživanja, odnosno mogućim brojem rješenja. Predstavljene su i informacije o parametrima koji se dinamički postavljaju, odnosno veličini populacije i maksimalnom broju iteracija, postotku korištenja svakog od dva tipa izračuna preklapanja po masi, te broju iteracija koji je algoritam zaista izvršio, sukladno mogućnosti uključivanja mehanizma ranijeg zaustavljanja (slika 6.4). Za svaki ispitni slučaj, odabrano je proizvoljno rješenje s Paretove fronte i predstavljene su njegove karakteristike. Rješenje je uspoređeno s proizvoljno odabranim rješenjem s Paretove fronte algoritma nasumičnog pretraživanja, kako bi se uočile razlike. Slučajevi u kojima je algoritam napravio manje od predviđenog broja iteracija zbog uključivanja mehanizma ranog zaustavljanja, označene su zelenom bojom. Povećavajući broj podskupova u koje se knjižnica peptida razdvaja, moguće je smanjiti preklapanje po masi uz zadržavanje niskog međusobnog preklapanja podskupova (tablica 6.2). Nadalje, povećanjem broja podskupova moguće je iznjedriti bolju pokrivenost izvorne knjižnice, odnosno podskupove koji sadrže veći broj peptida uz zadržavanje sličnih vrijednosti preklapanja (tablica 6.1). U određenim slučajevima, veći broj podskupova može istovremeno povećati sva tri kriterija (tablica 6.3), dok je u drugim slučajevima moguće značajno povećati jedan kriterij nauštrb preostala dva (tablica 6.4). Primjetno je i da u velikoj većini ispitnih slučajeva predloženi algoritam daje bolje rezultate u odnosu na optimizaciju nasumičnim pretraživanjem. Razmotrimo li obrasce korištenja različitih metoda za procjenu preklapanja po masi, moguće je uočiti da je iscrpno pretraživanje više korišteno u jednostavnijim ispitnim slučajevima (tablice 6.1, 6.2 i 6.5), dok je simulirano kaljenje korišteno u složenijima (tablice 6.3 i 6.4). Nadalje, optimizacija knjižnice peptida koja sadrži sekvence iz baze podataka DRAMP, ponudila je uvid u jedan mogući scenarij korištenja algoritma pri optimizaciji knjižnice koja sadrži antimikrobne peptide (tablica 6.5). U sva tri ispitna slučaja (3, 4 i 5 podskupova), predloženi algoritam je zadržao svih 17 antimikrobnih peptida, dok je nasumično pretraživanje u istim slučajevima zadržalo 9, 7 i 9 peptida. Uzevši u obzir podatke u tablicama od 6.1 do 6.5, mehanizam ranijeg zaustavljanja uštedio je izvođenje 3421 iteracije, što je ekvivalent uštedi resursa od 22%.

Tablica 6.1: Rezultati optimizacije knjižnice peptida koja sadrži 19 aminokiselina razdijeljenih na 6 pozicija, pri čemu se ona razdvajala u tri, četiri i pet podskupova. Naznačeni su i veličina prostora pretraživanja, veličina populacije, maksimalan i izveden broj iteracija, te postoci korištenja svake od dviju metoda za izračun različitosti po masi. Slučajevi u kojima je došlo do aktivacije mehanizma ranog zaustavljanja označeni su zeleno. Za svaki ispitni slučaj odabrano je proizvoljno rješenje s Paretove fronte i uspoređeno je s rješenjem pronađenim metodom nasumičnog pretraživanja. Preuzeto iz [190] i prevedeno.

Knjižnica peptida		OY:FVYE:VG:WDL:TC:QTAHPV				
Broj podskupova	Veličina prostora pretraživanja	Veličina populacije	Maksimalan broj iteracija	Simulirano kaljenje [%]	Iscrpno pretraživanje [%]	Broj iteracija koje su napravljene
3	5.7×10^{15}	352	704	0	100	497
Proizvoljno rješenje predloženog algoritma						
Broj peptida	Različitost po masi [%]	Različitost podskupova [0-100]	Podskup	Sekvenca podskupa		
240	100	100	1.	OY:FVE:VG:L:TC:QAH		
			2.	OY:FVY:VG:WD:TC:HV		
			3.	OY:FYE:V:WD:TC:QTP		
Proizvoljno rješenje nasumičnog pretraživanja						
117	100	100	1.	OY:FE:VG:WD:T:QTAHP		
			2.	OY:VYE:V:WL:C:QTP		
			3.	Y:E:V:W:C:V		
4	1.0×10^{21}	437	874	0	100	366
Proizvoljno rješenje predloženog algoritma						
268	100	100	1.	OY:FV:VG:L:TC:QTAH		
			2.	OY:FYE:G:WD:T:QTAHP		
			3.	OY:FVY:V:WD:T:QTAHPV		
			4.	OY:FVY:VG:WD:C:THV		
Proizvoljno rješenje nasumičnog pretraživanja						
120	100	100	1.	OY:FE:V:L:T:QAHPV		
			2.	Y:VY:G:D:TC:QHV		
			3.	OY:FY:G:WL:T:TAHPV		
			4.	OY:FYE:G:WL:C:TAPV		
5	1.8×10^{26}	523	1000	0	100	901
Proizvoljno rješenje predloženog algoritma						
312	100	100	1.	OY:FYE:G:L:T:QTAHPV		
			2.	OY:YE:G:WD:TC:QAHP		
			3.	OY:FV:G:WD:TC:QTAHP		
			4.	OY:FVY:V:WD:C:QTAHPV		
			5.	OY:FVY:V:WL:T:TAHPV		
Proizvoljno rješenje nasumičnog pretraživanja						
128	100	100	1.	Y:VYE:G:WL:TC:QA		
			2.	OY:FYE:V:L:C:QAHV		
			3.	OY:FVY:V:WL:T:PV		
			4.	O:V:G:WD:C:QAHP		
			5.	OY:FVY:V:WD:C:QTHV		

Tablica 6.2: Rezultati optimizacije knjižnice peptida koja sadrži 36 aminokiselina razdijeljenih na 6 pozicija, pri čemu se ona razdvajala u tri, četiri i pet podskupova. Naznačeni su i veličina prostora pretraživanja, veličina populacije, maksimalan i izveden broj iteracija, te postoci korištenja svake od dviju metoda za izračun različitosti po masi. Slučajevi u kojima je došlo do aktivacije mehanizma ranog zaustavljanja označeni su zeleno. Za svaki ispitni slučaj odabrano je proizvoljno rješenje s Paretove fronte i uspoređeno je s rješenjem pronađenim metodom nasumičnog pretraživanja. Preuzeto iz [190] i prevedeno.

Knjižnica peptida							LWHTV:WP:HDICKSREAWQ:IGAHRFSVDY:IHMDVA:IQ						
Broj podskupova		Veličina prostora pretraživanja		Veličina populacije		Maksimalan broj iteracija		Simulirano kaljenje [%]		Iscrpno pretraživanje [%]		Broj iteracija koje su napravljene	
3		5.0×10^{31}		581		1000		22		78		550	
Proizvoljno rješenje predloženog algoritma													
Broj peptida		Različitost po masi [%]		Različitost podskupova [0-100]		Podskup				Sekvenca podskupa			
424		86		100		1.		WHT:WP:SAW:GRFV:DA:IQ					
						2.		WTV:WP:KAW:AH:IM:I					
						3.		HV:WP:HDSQ:IS:DA:I					
Proizvoljno rješenje nasumičnog pretraživanja													
416		77		100		1.		L:WP:DEAW:S:M:I					
						2.		W:WP:EW:IAHRSD:A:I					
						3.		WT:WP:HICKSW:IHRFSVDY:HM:Q					
4		1.8×10^{42}		743		1000		14		86		629	
Proizvoljno rješenje predloženog algoritma													
424		92		100		1.		WV:WP:CSRAW:FS:HD:IQ					
						2.		WTV:WP:HEW:AFSY:IA:I					
						3.		W:WP:HISEAW:AY:IM:Q					
						4.		W:WP:DIC:RS:MVA:IQ					
Proizvoljno rješenje nasumičnog pretraživanja													
434		78		100		1.		HV:P:KA:I:IHMA:I					
						2.		H:W:CEW:RD:MVA:Q					
						3.		WT:WP:HDSEQ:IGHF:IHMD:I					
						4.		LHTV:P:IS:ASVDY:MV:I					
5		6.8×10^{52}		905		1000		15		85		1000	
Proizvoljno rješenje predloženog algoritma													
456		93		100		1.		LH:WP:HCSRAWQ:GF:HV:Q					
						2.		WT:WP:HR:IAY:IMV:Q					
						3.		WTV:WP:CAQ:FSVY:H:Q					
						4.		WV:WP:DCKSA:AR:HD:I					
						5.		LWH:WP:HISRA:VY:MV:I					
Proizvoljno rješenje nasumičnog pretraživanja													
435		77		100		1.		LW:W:HR:GRVDY:V:Q					
						2.		LW:WP:EW:IGAHVDY:HV:Q					
						3.		V:W:IWQ:IARFSVY:D:Q					
						4.		WHT:WP:KSW:FVY:IHA:I					
						5.		HT:W:CKRAQ:IGHRSD:MV:I					

Tablica 6.3: Rezultati optimizacije knjižnice peptida koja sadrži 53 aminokiselina razdijeljenih na 8 pozicija, pri čemu se ona razdvajala u tri, četiri i pet podskupova. Naznačeni su i veličina prostora pretraživanja, veličina populacije, maksimalan i izveden broj iteracija, te postoci korištenja svake od dviju metoda za izračun različitosti po masi. Slučajevi u kojima je došlo do aktivacije mehanizma ranog zaustavljanja označeni su zeleno. Za svaki ispitni slučaj odabrano je proizvoljno rješenje s Paretove fronte i uspoređeno je s rješenjem pronađenim metodom nasumičnog pretraživanja. Preuzeto iz [190] i prevedeno.

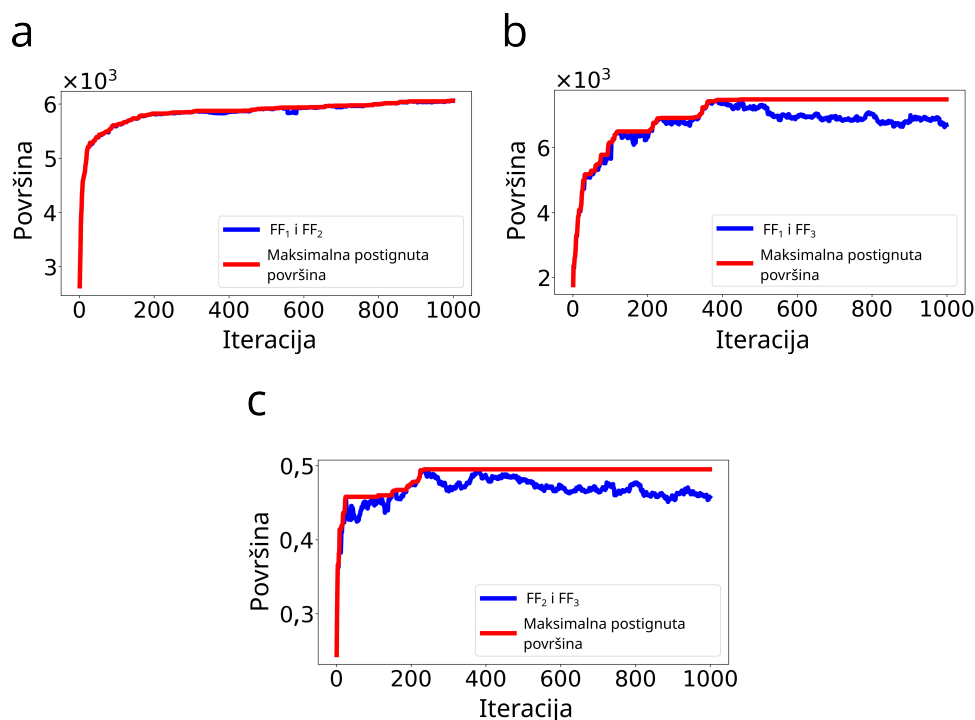
Knjižnica peptida		CQYHRA:T:OAY:ENQLMD:CMQTILSRGEA:KCAQIENM:PLQWIAYMRDVGSN:MFGS				
Broj podskupova	Veličina prostora pretraživanja	Veličina populacije	Maksimalan broj iteracija	Simulirano kaljenje [%]	Iscrpno pretraživanje [%]	Broj iteracija koje su napravljene
3	4.5×10^{46}	811	1000	33	67	829
Proizvoljno rješenje predloženog algoritma						
	Broj peptida	Različitost po masi [%]	Različitost podskupova [0-100]	Podskup	Sekvenca podskupa	
	3,648	36	43	1.	CQHRA:T:OY:QD:TLRG:KIM:WA:MFGS	
2.				C:T:OY:LM:CT:AI:QYM:M		
3.				CYR:T:OY:QL:CQISREA:AN:PYMGS:MG		
Proizvoljno rješenje nasumičnog pretraživanja						
	3,516	30	100	1.	A:T:OA:E:CTIRG:CA:PLR:G	
2.				YA:T:OAY:EQLM:MA:AQN:QWARDS:MF		
3.				HRA:T:OY:EN:MSRA:QINM:PQWAYRDVS:S		
4	1.6×10^{62}	1000	1000	26	74	493
Proizvoljno rješenje predloženog algoritma						
	4,152	44	75	1.	Y:T:AY:Q:QR:M:QWYS:FGS	
				2.	CYHR:T:OY:EQLMD:RG:AEM:PMWAYMDN:S	
				3.	CQYHR:T:OY:NQ:TIRGE:KCAN:WGS:FG	
				4.	C:T:A:Q:MTLSRA:M:LYDG:M	
Proizvoljno rješenje nasumičnog pretraživanja						
	4,428	27	33	1.	CYHR:T:AY:QLM:CMQTSG:CQN:PQYMV:S	
				2.	CYA:T:OY:EQLD:CLSA:AEN:LARDGS:G	
				3.	YR:T:OA:EL:CMTIRA:CA:LQMGS:S	
				4.	A:T:OY:EM:CMG:M:PLQVS:S	
5	5.7×10^{77}	1000	1000	22	78	1000
Proizvoljno rješenje predloženog algoritma						
	4,666	49	83	1.	QHRA:T:OY:N:MILRG:CAN:PWIMRV:FGS	
				2.	CQYHA:T:OY:LM:QTLSGA:CN:WIYVS:FG	
				3.	CQH:T:OY:E:L:I:WIYDVGS:MS	
				4.	QY:T:OY:QD:T:C:VN:G	
				5.	Q:T:O:QM:MTL:C:Y:F	
Proizvoljno rješenje nasumičnog pretraživanja						
	4,984	26	67	1.	H:T:A:NQMD:CMTSE:CAQI:LQAYN:MFGS	
				2.	Y:T:O:ED:MQLSA:N:IMRDGS:F	
				3.	RA:T:OAY:QM:MLRE:C:PAYVGS:G	
				4.	YHR:T:OY:MD:CLG:KCAQIEN:PLWIRDVGS:F	
				5.	QYRA:T:A:NLMD:IS:KAEN:WAMVGS:G	

Tablica 6.4: Rezultati optimizacije knjižnice peptida koja sadrži 83 aminokiselina razdijeljenih na 10 pozicija, pri čemu se ona razdvajala u tri, četiri i pet podskupova. Naznačeni su i veličina prostora pretraživanja, veličina populacije, maksimalan i izveden broj iteracija, te postoci korištenja svake od dviju metoda za izračun različitosti po masi. Slučajevi u kojima je došlo do aktivacije mehanizma ranog zaustavljanja označeni su zeleno. Za svaki ispitni slučaj odabrano je proizvoljno rješenje s Paretove fronte i uspoređeno je s rješenjem pronađenim metodom nasumičnog pretraživanja. Preuzeto iz [190] i prevedeno.

Knjižnica peptida						
KHESLFPMQ:TKSIHYVMWQLNE:LDC:HE:CLSYTGVEQFAHMRKP: VNMKG:SL:VFESWQCGYMD:FLANHMIDYRWSECKQPV:OTRM						
Broj podskupova	Veličina prostora pretraživanja	Veličina populacije	Maksimalan broj iteracija	Simulirano kaljenje [%]	Iscrpno pretraživanje [%]	Broj iteracija koje su napravljene
3	8.0×10^{73}	1000	1000	38	62	876
Proizvoljno rješenje predloženog algoritma						
Broj peptida	Različitost po masi [%]	Različitost podskupova [0-100]	Podskup	Sekvenca podskupa		
3,872	35	100	1.	EM:SH:C:H:GF:VNG:SL:QCD:IDYWE:OR		
			2.	LFPM:I:D:H:FK:K:L:S:NC:OT		
			3.	S:TKSYW:LC:E:CYAHRP:VG:L:WG:AMYEC:OR		
Proizvoljno rješenje nasumičnog pretraživanja						
78,492	1	100	1.	S:SHVN:C:E:LVA:KG:L:QMD:ANYRSECV:OTR		
			2.	KFQ:KIHVLN:LC:H:CLYTGVEQHRK:VMK:S:VFSQGM:AHIDYRWEV:M		
			3.	FP:KWQ:C:H:YTGEQHMP:N:S:VESWC:MDYSCKQV:T		
4	3.5×10^{98}	1000	1000	33	67	1000
Proizvoljno rješenje predloženog algoritma						
5,096	35	100	1.	M:MW:D:H:GR:G:L:M:MK:T		
			2.	PMQ:TKHM:L:E:LSEQ:NG:S:WGM:LMWV:OR		
			3.	KS:SWN:L:H:YG:MK:L:FG:LHMDP:OM		
			4.	HSFP:TYWQ:D:H:TA:G:SL:WD:FLAIDRWEV:OR		
Proizvoljno rješenje nasumičnog pretraživanja						
45,250	0	100	1.	LF:TKHYQN:D:H:LSTFAMP:NK:SL:FECD:LHDRWSCQK:TRM		
			2.	E:SIHVM:L:E:VEH:KG:L:ESWQCGM:LAMDRSEKQ:O		
			3.	SLFM:E:L:E:YGVFEFMRP:NK:L:QG:FLHYWKQV:R		
			4.	SLPQ:TQN:D:H:STHKP:MK:S:GM:LMDYEKP:TRM		
5	1.5×10^{123}	1000	1000	34	66	1000
Proizvoljno rješenje predloženog algoritma						
86,838	29	83	1.	HLM:L:LD:H:CSGVF:V:S:WM:M:R		
			2.	SQ:I:L:H:YGFHMP:N:L:WCGD:S:T		
			3.	H:TI:L:E:CYTGMP:VK:L:FW:RCP:OR		
			4.	E:N:D:H:STQFARP:K:L:Q:ANWSEV:M		
			5.	KHEFM:KIHVMWNE:LD:HE:CLVRP:VNG:L:EWQ:LAHDRW:OM		
Proizvoljno rješenje nasumičnog pretraživanja						
77,984	0	100	1.	HLQ:KVMWE:C:E:YTGEQFMK:VKG:L:VSW:FLHIRECPV:M		
			2.	KEFMQ:KSVQN:LDC:E:P:M:S:WM:FLANIDYWEP:T		
			3.	HSLFPMQ:SIHYLNE:C:E:LTGQAHRP:NM:L:VEQCD:ANHMIDYRWSECK:R		
			4.	KHSLFQ:TVMWE:L:H:LYTGVEAMR:G:S:VESWY:FANDRWSCKV:M		
			5.	KHFP:VL:C:H:LGFAMK:VNG:L:VG:FMDYSKQP:T		

Tablica 6.5: Rezultati optimizacije knjižnice peptida koja sadrži 28 aminokiselina razdijeljenih na 6 pozicija, pri čemu se ona razdvajala u tri, četiri i pet podskupova. Naznačeni su i veličina prostora pretraživanja, veličina populacije, maksimalan i izveden broj iteracija, te postoci korištenja svake od dviju metoda za izračun različitosti po masi. Slučajevi u kojima je došlo do aktivacije mehanizma ranog zaustavljanja označeni su zeleno. Za svaki ispitni slučaj odabrano je proizvoljno rješenje s Paretove fronte i uspoređeno je s rješenjem pronađenim metodom nasumičnog pretraživanja. Preuzeto iz [190] i prevedeno.

Knjižnica peptida		RYC:RAK:AGKRWL:ARWQFPK:RWFAK:CRFG				
Broj podskupova	Veličina prostora pretraživanja	Veličina populacije	Maksimalan broj iteracija	Simulirano kaljenje [%]	Iscrpno pretraživanje [%]	Broj iteracija koje su napravljene
3	6.1×10^{24}	473	946	19	81	839
Proizvoljno rješenje predloženog algoritma						
Broj peptida	Različitost po masi [%]	Različitost podskupova [0-100]	Podskup		Sekvenca podskupa	
7,562	6	60	1.	R:K:A:Q:K:F		
			2.	RYC:RAK:AGKRWL:ARWQFPK:RWFAK:CRFG		
			3.	Y:R:W:R:W:C		
Proizvoljno rješenje nasumičnog pretraživanja						
4,116	6	3	1.	RY:K:AGRWL:RK:WAK:RFG		
			2.	RYC:RAK:AGKRWL:ARWQFK:RWFA:CRF		
			3.	RC:A:AGR:RP:WK:RG		
4	1.1×10^{33}	599	1000	10	90	743
Proizvoljno rješenje predloženog algoritma						
7,595	12	15	1.	RY:K:KW:ARW:A:G		
			2.	Y:R:K:P:R:CFG		
			3.	RY:K:G:A:RWFAK:RF		
			4.	RYC:RAK:AGKRWL:ARWQFPK:RWFAK:CRFG		
Proizvoljno rješenje nasumičnog pretraživanja						
3,351	10	50	1.	Y:A:AGKWL:ARF:RWFAK:CRG		
			2.	RC:RAK:AGRWL:ARWFP:RWFAK:CRFG		
			3.	C:K:AGL:AQK:WA:C		
			4.	RYC:RK:K:ARWFPK:F:CRG		
5	2.0×10^{41}	725	1000	10	90	560
Proizvoljno rješenje predloženog algoritma						
7,594	19	23	1.	Y:R:L:WFK:R:CF		
			2.	YC:A:AW:Q:A:C		
			3.	C:A:AGL:RP:R:RF		
			4.	C:R:K:RK:AK:RFG		
			5.	RYC:RAK:AGKRWL:ARWQFPK:RWFAK:CRFG		
Proizvoljno rješenje nasumičnog pretraživanja						
3,240	15	28	1.	RYC:RA:AGKRWL:RWFP:RWFAK:CRFG		
			2.	RC:K:AGR:ARWF:RFK:CR		
			3.	YC:K:AGKL:AQK:WAK:RG		
			4.	RYC:A:AW:AK:FA:C		
			5.	RY:A:AKR:ARQF:WF:F		



Slika 6.4: Prikaz oscilacija površina ispod parova krivulja funkcija vrednovanja za vrijeme optimizacije. Naznačena je površina u svakoj iteraciji, te maksimalna površina postignuta do iteracije. Ako se površina ne poveća u 100 uzastopnih iteracija, izvođenje algoritma se prekida. Grafovi prikazuju površine za krivulje omeđene vrijednostima: (a) ukupnog broja peptida (FF_1) i različitosti po masi (FF_2), (b) ukupnog broja peptida (FF_1) i međusobne različitosti podskupova (FF_3), i (c) različitosti po masi (FF_2) i međusobne različitosti podskupova (FF_3). Podaci su zabilježeni pri optimizaciji knjižnice peptida „LWHTV:WP:HDICKSREAWQ:IGAHRFSVDY:IHMDVA:IQ” i traženju 5 podskupova (tablica 6.2). Preuzeto iz [190] i prevedeno.

Sukladno rezultatima, može se zaključiti da razdvajanje knjižnica peptida u veći broj podskupova donosi specifične prednosti u vidu zadržavanja sekvenci koje bi se u suprotnom uklonile. Predloženi algoritam, vođen kriterijima različitosti po masi, različitosti podskupova, te brojem zadržanih peptida, svakako može pomoći u tome. Uzme li se u obzir da je ispitivanje većeg broja sekvenci nužno pri dizajniranju novih lijekova i terapijskih molekula općenito, predloženi algoritam uvelike može pospješiti planiranje sinteze i pridonijeti većoj učinkovitosti procesa, olakšavajući dekonvoluciju knjižnice peptida nakon sinteze podjele i miješanja, i provedenih eksperimenata. Dodatno je razvijena i aplikacija s interaktivnim grafičkim sučeljem kroz koju je moguće koristiti opisani algoritam, dostupna na sljedećoj internetskoj adresi: <https://deshpet.riteh.hr>, olakšavajući tako pristup algoritmu osobama koje možda nisu toliko vične programiranju.

7. ZAKLJUČAK

Ovaj doktorski rad istražio je tematiku računalnog stvaranja i karakterizacije peptida, te se na temelju predstavljenih rezultata može zaključiti kako su definirane hipoteze potvrđene, a doprinosi ispunjeni. Predloženi su generativni modeli temeljeni na genetskom algoritmu s operatorima prilagođenim evoluciji peptida, mogućnošću kontrole duljine sekvenci i raznovrsnosti populacije, te modelom predviđanja strojnog učenja kao funkcijom dobrote za otkrivanje peptida s katalitičkom aktivnošću hidrolize estera i za stvaranje peptida s visokom vjerojatnošću samosastavljanja. Predstavljena je i lista peptida s eksperimentalno potvrđenim svojstvom samosastavljanja, a predloženi su i višekriterijski genetski algoritmi za pronalazak peptidnih inhibitora enzima beta-sekretaze 1 koji samostalno prolaze krvno-moždanu barijeru, te za optimizaciju knjižnica peptida i njihovo razdvajanje na podskupove pogodnije za dekonvoluciju nakon sinteze. Istraživanje je provedeno u nekoliko slijednih faza.

Najprije je pregledom literature ustanovljeno nepostojanje numeričke predstave peptida koja modelima strojnog učenja istovremeno pruža informacije o fizikalno-kemijskim svojstvima i primarnoj strukturi. S obzirom na to da informativne značajke predstavljaju temelj za izgradnju kvalitetnih modela strojnog učenja, predložena je nova shema predstave peptida u računalu, nazvana shemom sekvencijalnih svojstava. Predložena shema peptide numerički predstavlja pomoću 94 fizikalno-kemijska svojstva svake aminokiseline od kojih se peptid sastoji, što modelima strojnog učenja daje više informacija u odnosu na tradicionalne predstave uvriježene u literaturi. Usporedbom sheme sekvencijalnih svojstava na četiri skupa podataka s tri sheme uvriježene u literaturi u domeni računalnog predviđanja aktivnosti peptida, zaključeno je da shema sekvencijalnih svojstava modelima strojnog učenja daje potpuniju informaciju o sekvenci i time podiže razine njihovih točnosti. Detaljnijom analizom ROC-AUC vrijednosti utvrđeno je da je predložena shema

postigla najviše vrijednosti u odnosu na ostale načine predstave u slučaju sva četiri skupa podataka, dok su Friedmanov test i Wilcoxonov test rangova te zaključke dodatno osnažili i potvrdili postojanje statistički značajnih razlika između sheme sekvencijalnih svojstava i ostalih shema predstava peptida na sva 4 skupa podataka, odnosno u 23 od 24 ispitna slučaja.

Shema sekvencijalnih svojstava korištena je za predstavu primjera pomoću kojih je naučen model povratne neuronske mreže za identifikaciju peptida s katalitičkom aktivnošću hidrolize para-nitrofenil acetata. Model je postigao dobre rezultate s medijanom točnosti klasifikacije od 75%, iako se skup podataka za učenje i vrednovanje modela sastojao od svega 85 primjera. Model je korišten u genetskom algoritmu kao funkcija dobrote za pronalazak novih, dosad nepoznatih peptida s katalitičkom aktivnošću. Mehanizam održavanja raznolikosti populacije, kao i mehanizam navođenja algoritma prema peptidima s proizvoljnom duljinom sekvence, učinili su algoritam svestranijim. Nakon provedbe dvaju ispitnih slučajeva, algoritam je pronašao peptide čija je procijenjena vjerojatnost za katalitičku aktivnost vrlo visoka, pri čemu je za 30 najboljih peptida iz oba pokusa uvijek iznad 97%. Usporedba razdioba aminokiselina skupa podataka za učenje i novostvorenih peptida ukazala je na razlike u njihovu sastavu, što može biti indikator generalizacijske sposobnosti algoritma i uspješne pretrage kemijskog prostora. Pritom je uočena tendencija generativnog modela ka stvaranju peptida s glicinom, argininom, histidinom i tirozinom, što je u skupu podataka za učenje manje izraženo.

Slični principi primijenjeni su na pronalazak peptida sa svojstvom samosastavljanja, s obzirom na to da se oni mogu koristiti kao temelj razvoja novih nanomaterijala ili umjetnih enzima. Sklonost peptida ka samosastavljanju predviđena je hibridnim modelom strojnog učenja koji je sekvencu promatrao kao niz kemijskih gradivnih jedinica: aminokiselina, dipeptida i tripeptida. Izdvajanjem vrijednosti AP tehnikom pomičnog prozora, uz primjenu sheme sekvencijalnih svojstava, dobivena su predviđanja sklonosti samosastavljanja s prosječnom točnošću klasifikacije od 80%, što predstavlja značajan napredak u odnosu na dosad najtočniji model u literaturi. Po primjeni modela u generativnom genetskom algoritmu, zaključeno je da prosječna sličnost 10 najboljih peptida u odnosu na peptide iz skupa podataka za učenje iznosi manje od 10%, pri čemu sve sekvence imaju procijenjenu vjerojatnost samosastavljanja višu od 95%. Navedeno ukazuje na sposobnost generalizacije znanja modela predviđanja, te sposobnost generativnog modela da pronade

kvalitetne sekvence izvan tradicionalnog prostora rješenja omeđenog sekvencama prijavljenim u literaturi. Daljnja ispitivanja simulacijama molekularne dinamike i laboratorijskim testovima pokazala su da se četiri od pet ispitanih peptida zaista spontano slaže u neki oblik nanostrukture, što je dosljedno s točnošću modela predviđanja na testnom skupu podataka i predloženom generativnom modelu dodatno daje na važnosti i vjerodostojnosti.

Evolucijski dizajn novih molekula primijenjen je i na traženje inhibitora enzima beta-sekretaze 1 u mozgu čovjeka, čije inhibiranje predstavlja potencijalni način reguliranja napretka Alzheimerove bolesti. Genetskim algoritmom vođenim procjenom vezanja peptida za enzim molekularnim pristajanjem, Wildman-Crippen logP vrijednošću i procjenom penetracije krvno-moždane barijere postojećim modelom strojnog učenja iz literature, dobivene su sekvence po svojstvima kvalitetnije u odnosu na patentirane peptidne inhibitore beta-sekretaze 1 iz literature. Prosječni afinitet vezanja dizajniranih peptida za beta-sekretazu 1 poboljšao je za 42 postotna boda, a procijenjena vjerojatnost prolaska krvno-moždane barijere viša je od 72%.

Kao posljednja nit istraživanja ispitana je mogućnost korištenja evolucijskog pristupa u razdvajanju knjižnica peptida pri izradi plana sinteze. Knjižnice peptida sintetizirane tehnikom podjele i miješanja koristan su pristup istovremenom ispitivanju velikog broja peptida, no također kasniju dekonvoluciju knjižnice i identifikaciju spojeva koji su uzrok ustanovljenom učinku čine složenijom. Predstavljeni pristup kao temelj koristi genetski algoritam koji knjižnicu peptida razdvaja na podskupove i vođen je trima kriterijima: zadržavanjem što većeg broja peptida iz izvorne knjižnice, minimiziranjem preklapanja peptida po masi unutar podskupova, te minimiziranjem međusobnog preklapanja podskupova. Time se optimizacija knjižnice navodi ka zadržavanju izvornog kemijskog prostora kroz više disjunktnih podskupova. U okviru istraživanja, razvijen je i pristup evaluaciji preklapanja peptida po masi simuliranim kaljenjem za knjižnice za koje iscrpna pretraga nije moguća zbog ograničenja vremena izvođenja i dostupne memorije računala. Ispitivanjem algoritma u 5 ispitnih slučajeva, u kojima su provjerene knjižnice različitih duljina i složenosti, utvrđeno je da algoritam uspješno dijeli knjižnice nastojeći što je više moguće uvažiti zadana optimizacijska ograničenja, dok provedena usporedba s nasumičnim pretraživanjem daje rezultatima dodatni kontekst.

Ova disertacija time čini zaokruženu cjelinu i daje doprinos području računalne karakterizacije i dizajna peptida. Zaključno, predloženi algoritmi i modeli mogu se uspješno

primijeniti u istraživanju različitih skupina peptida kako bi se pronalazak vodećih spojeva ubrzao i učinio pristupačnijim.

LITERATURA

- [1] T. A. Ahmed and R. Hammami. Recent insights into structure–function relationships of antimicrobial peptides. *Journal of Food Biochemistry*, 43(1):e12546, 2019.
- [2] S. Akbar, A. Ahmad, M. Hayat, A. U. Rehman, S. Khan, and F. Ali. iAtbP-Hyb-EnC: Prediction of antitubercular peptides via heterogeneous feature representation and genetic algorithm based ensemble learning model. *Computers in Biology and Medicine*, 137:Article number: 104778, 2021.
- [3] F. Ali, H. Kumar, W. Alghamdi, F. A. Kateb, and F. K. Alarfaj. Recent advances in machine learning-based models for prediction of antiviral peptides. *Archives of Computational Methods in Engineering*, 30(7):4033–4044, 2023.
- [4] R. Alizadeh, J. K. Allen, and F. Mistree. Managing computational complexity using surrogate models: A critical review. *Research in Engineering Design*, 31(3):275–298, 2020.
- [5] S. Andradóttir. A review of random search methods. *Handbook of Simulation Optimization*, pages 277–292, 2014.
- [6] M. A. Arbib. *The handbook of brain theory and neural networks*. MIT press, 2003.
- [7] K. Ariga, J. P. Hill, M. V. Lee, A. Vinu, R. Charvet, and S. Acharya. Challenges and breakthroughs in recent research on self-assembly. *Science and Technology of Advanced Materials*, 2008.
- [8] F. H. Arnold. The nature of chemical innovation: New enzymes by evolution. *Quarterly Reviews of Biophysics*, 48(4):404–410, 2015.

- [9] F. H. Arnold. Directed evolution: Bringing new chemistry to life. *Angewandte Chemie International Edition*, 57(16):4143, 2017.
- [10] M. Attique, M. S. Farooq, A. Khelifi, and A. Abid. Prediction of therapeutic peptides using machine learning: Computational models, datasets, and feature encodings. *IEEE Access*, 8:148570–148594, 2020.
- [11] R. Batra, T. D. Loeffler, H. Chan, S. Srinivasan, H. Cui, I. V. Korendovych, V. Nanda, L. C. Palmer, L. A. Solomon, H. C. Fry, et al. Machine learning overcomes human bias in the discovery of self-assembling peptides. *Nature Chemistry*, 14(12):1427–1435, 2022.
- [12] O. A. Bello, A. Ogundipe, D. Mohammed, F. Adebola, O. A. Alonge, et al. AI-driven approaches for real-time fraud detection in US financial transactions: Challenges and opportunities. *European Journal of Computer Science and Information Technology*, 11(6):84–102, 2023.
- [13] J. A. Beltran, L. Aguilera-Mendoza, and C. A. Brizuela. Optimal selection of molecular descriptors for antimicrobial peptides classification: An evolutionary feature weighting approach. *BMC Genomics*, 19(S7), 2018.
- [14] H. J. Berendsen, J. v. Postma, W. F. Van Gunsteren, A. DiNola, and J. R. Haak. Molecular dynamics with coupling to an external bath. *The Journal of Chemical Physics*, 81(8):3684–3690, 1984.
- [15] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic Acids Research*, 28(1):235–242, 2000.
- [16] Y. Bian, J. Wang, J. J. Jun, and X.-Q. Xie. Deep convolutional generative adversarial network (dcGAN) models for screening and design of small molecules targeting cannabinoid receptors. *Molecular Pharmaceutics*, 16(11):4451–4460, 2019.
- [17] M. Biancalana, K. Makabe, A. Koide, and S. Koide. Molecular mechanism of thioflavin-T binding to the surface of β -rich peptide self-assemblies. *Journal of Molecular Biology*, 385(4):1052–1063, 2009.

- [18] P.-A. Binz, J. Shofstahl, J. A. Vizcaíno, H. Barsnes, R. J. Chalkley, G. Menschaert, E. Alpi, K. Clauser, J. K. Eng, L. Lane, et al. Proteomics standards initiative extended FASTA format. *Journal of Proteome Research*, 18(6):2686–2692, 2019.
- [19] C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.
- [20] H. Bisswanger. Enzyme assays. *Perspectives in Science*, 1(1-6):41–55, 2014.
- [21] T. Blicke. Tournament selection. *Evolutionary Computation*, 1(181-186):188, 2000.
- [22] C. Bonferroni. Class statistical theory and probability calculation. *Royal Institute of Economic and Commercial Sciences of Florence*, 8:3–62, 1936.
- [23] K. Boone, C. Wisdom, K. Camarda, P. Spencer, and C. Tamerler. Combining genetic algorithm with machine learning strategies for designing potent antimicrobial peptides. *BMC Bioinformatics*, 22(1):239, 2021.
- [24] V. Boopathi, S. Subramaniam, A. Malik, G. Lee, B. Manavalan, and D.-C. Yang. mACPpred: A Support Vector Machine-Based Meta-Predictor for Identification of Anticancer Peptides. *International Journal of Molecular Sciences*, 20(8):Article number: 1964, 2019.
- [25] S. Borukhov and E. Nudler. RNA polymerase: The vehicle of transcription. *Trends in Microbiology*, 16(3):126–134, 2008.
- [26] Z. Breijyeh and R. Karaman. Comprehensive review on Alzheimer’s disease: Causes and treatment. *Molecules*, 25(24):5789, 2020.
- [27] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.
- [28] K. Browne, S. Chakraborty, R. Chen, M. D. Willcox, D. S. Black, W. R. Walsh, and N. Kumar. A new era of antibiotics: The clinical potential of antimicrobial peptides. *International Journal of Molecular Sciences*, 21(19):7047, 2020.
- [29] S. N. H. Bukhari, J. Webber, and A. Mehbodniya. Decision tree based ensemble machine learning model for the prediction of Zika virus T-cell epitopes as potential vaccine candidates. *Scientific Reports*, 12(1):7810, 2022.

- [30] M. Bunge. A general black box theory. *Philosophy of Science*, 30(4):346–358, 1963.
- [31] T. Burgin, B. C. Pollard, B. C. Knott, H. B. Mayes, M. F. Crowley, J. E. McGeehan, G. T. Beckham, and H. L. Woodcock. The reaction mechanism of the *Ideonella sakaiensis* PETase enzyme. *Communications Chemistry*, 7(1):65, 2024.
- [32] G. Bussi, D. Donadio, and M. Parrinello. Canonical sampling through velocity rescaling. *The Journal of Chemical Physics*, 126(1), 2007.
- [33] D. Cabral and T. O. Obisesan. Detection and diagnosis of early symptomatic Alzheimer’s disease in primary care. *Federal Practitioner*, 41(Suppl 6):S7, 2024.
- [34] E. Callaway. Chemistry Nobel goes to developers of AlphaFold AI that predicts protein structures. *Nature*, 634(8034):525–526, 2024.
- [35] A. Capecchi, X. Cai, H. Personne, T. Köhler, C. van Delden, and J.-L. Reymond. Machine learning designs non-hemolytic antimicrobial peptides. *Chemical Science*, 12(26):9221–9232, 2021.
- [36] A. Capecchi, A. Zhang, and J.-L. Reymond. Populating chemical space with peptides using a genetic algorithm. *Journal of Chemical Information and Modeling*, 60(1):121–132, 2019.
- [37] T. Cavalier-Smith. Cell evolution and Earth history: Stasis and revolution. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 361(1470):969–1006, 2006.
- [38] K. H. Chan, W. H. Lee, M. Ni, Y. Loo, and C. A. Hauser. C-terminal residue of ultrashort peptides impacts on molecular self-assembly, hydrogelation, and interaction with small-molecule drugs. *Scientific Reports*, 8(1):1–14, 2018.
- [39] P. Charoenkwan, P. Chumnanpuen, N. Schaduengrat, P. Lio’, M. A. Moni, and W. Shoombuatong. Improved prediction and characterization of blood-brain barrier penetrating peptides using estimated propensity scores of dipeptides. *Journal of Computer-Aided Molecular Design*, 36(11):781–796, 2022.
- [40] A. Chatterjee, A. Reja, S. Pal, and D. Das. Systems chemistry of peptide-assemblies for biochemical transformations. *Chemical Society Reviews*, 51(8):3047–3070, 2022.

- [41] X. Chen, C. Li, M. T. Bernards, Y. Shi, Q. Shao, and Y. He. Sequence-based peptide identification, generation, and property prediction with deep learning: A review. *Molecular Systems Design and Engineering*, 6(6):406–428, 2021.
- [42] D. Chicco and G. Jurman. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21(1):Article number: 6, Jan 2020.
- [43] O. Conchillo-Solé, N. De Groot, F. Avilés, J. Vendrell, X. Daura, and S. Ventura. AGGRESKAN: A server for the prediction of “hot spots” of aggregation in polypeptides. *BMC Bioinformatics*, 8:65, 02 2007.
- [44] W. J. Conover. *Practical nonparametric statistics*. john wiley and sons, 1999.
- [45] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.
- [46] D. J. Craik, D. P. Fairlie, S. Liras, and D. Price. The future of peptide-based drugs. *Chemical Biology and Drug Design*, 81(1):136–147, 2013.
- [47] G. Cruciani, M. Baroni, E. Carosati, M. Clementi, R. Valigi, and S. Clementi. Peptide studies by means of principal properties of amino acids derived from MIF descriptors. *Journal of Chemometrics*, 18(3-4):146–155, 2004.
- [48] J. Cruz, P. Rondon-Villarreal, R. G. Torres, M. Urquiza, F. Guzman, C. Alvarez, M. A. Abengozar, D. A. Sierra, L. Rivas, R. Fernandez-Lafuente, et al. Design of bactericidal peptides against Escherichia coli O157: H7, Pseudomonas aeruginosa and methicillin-resistant Staphylococcus aureus. *Medicinal Chemistry*, 14(7):741–752, 2018.
- [49] J. M. Cunningham, G. Koytiger, P. K. Sorger, and M. AlQuraishi. Biophysical prediction of protein–peptide interactions and signaling networks using machine learning. *Nature Methods*, 17(2):175–183, 2020.
- [50] A. W. Czarnik and J. D. Keene. Combinatorial chemistry. *Current Biology*, 8(20):R705–R707, 1998.

- [51] R. Dai, W. Zhang, W. Tang, E. Wynendaele, Q. Zhu, Y. Bin, B. De Spiegeleer, and J. Xia. BBPpred: Sequence-based prediction of blood-brain barrier peptides with feature representation learning and logistic regression. *Journal of Chemical Information and Modeling*, 61(1):525–534, 2021.
- [52] C. Darwin. *On the origin of species: A facsimile of the first edition*. Harvard University Press, 1964.
- [53] P. Das, T. Sercu, K. Wadhawan, I. Padhi, S. Gehrmann, F. Cipcigan, V. Chenthamarakshan, H. Strobel, C. Dos Santos, P.-Y. Chen, et al. Accelerated antimicrobial discovery via deep generative models and molecular dynamics simulations. *Nature Biomedical Engineering*, 5(6):613–623, 2021.
- [54] P. Das, K. Wadhawan, O. Chang, T. Sercu, C. dos Santos, M. Riemer, V. Chenthamarakshan, I. Padhi, and A. Mojsilovic. Pepcvae: Semi-supervised targeted design of antimicrobial peptide molecules. *arXiv preprint: 1810.07743*, 2018.
- [55] A. M. Davis, A. T. Plowright, and E. Valeur. Directing evolution: The next revolution in drug discovery? *Nature Reviews Drug Discovery*, 16(10):681–698, 2017.
- [56] N. S. De Groot, I. Pallarés, F. X. Avilés, J. Vendrell, and S. Ventura. Prediction of „hot spots” of aggregation in disease-linked polypeptides. *BMC Structural Biology*, 5(1):18, 2005.
- [57] E. C. L. de Oliveira, K. Santana, L. Josino, A. H. Lima e Lima, and C. de Souza de Sales Júnior. Predicting cell-penetrating peptides using machine learning algorithms and navigating in their chemical space. *Scientific Reports*, 11(1):7628, 2021.
- [58] S. N. Dean, J. A. E. Alvarez, D. Zabetakis, S. A. Walper, and A. P. Malanoski. PepVAE: Variational autoencoder framework for antimicrobial peptide generation and activity prediction. *Frontiers in Microbiology*, 12:725727, 2021.
- [59] S. N. Dean and S. A. Walper. Variational autoencoder for generation of antimicrobial peptides. *ACS Omega*, 5(33):20746–20754, 2020.

- [60] K. Deb. Multi-objective optimisation using evolutionary algorithms: An introduction. In *Multi-objective evolutionary optimisation for product design and manufacturing*, pages 3–34. Springer, 2011.
- [61] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
- [62] E. Dražić, D. Jelušić, P. Janković Bevandić, G. Mauša, and D. Kalafatovic. Using machine learning to fast-track peptide nanomaterial discovery. *ACS Nano*, 19(22):20295–20320, 2025.
- [63] S. D. Dyllal, M. T. Brown, and P. J. Johnson. Ancient invasions: From endosymbionts to organelles. *Science*, 304(5668):253–257, 2004.
- [64] A. Economou, S. K. Karapetis, G.-P. Nikoleli, D. P. Nikolelis, S. Bratakou, and T. H. Varzakas. Enzyme-based sensors. *Advances in Food Diagnostics*, pages 231–250, 2017.
- [65] D. Eisenberg, E. Schwarz, M. Komaromy, and R. Wall. Analysis of membrane and surface protein sequences with the hydrophobic moment plot. *Journal of Molecular Biology*, 179(1):125–142, 1984.
- [66] D. Eisenberg, R. M. Weiss, and T. C. Terwilliger. The hydrophobic moment detects periodicity in protein hydrophobicity. *Proceedings of the National Academy of Sciences*, 81(1):140–144, 1984.
- [67] J. L. Elman. Finding structure in time. *Cognitive Science*, 14(2):179–211, 1990.
- [68] Enamine Inc. Enamine REAL database, Accessed: November 25th, 2025.
- [69] I. Erjavac, D. Kalafatovic, and G. Mauša. Coupled encoding methods for antimicrobial peptide prediction: How sensitive is a highly accurate model? *Artificial Intelligence in the Life Sciences*, 2:100034, 2022.
- [70] J. Fan, A. Fu, and L. Zhang. Progress in molecular docking. *Quantitative Biology*, 7(2):83–89, 2019.

- [71] F. A. Farris. The Gini index and measures of inequality. *The American Mathematical Monthly*, 117(10):851–864, 2010.
- [72] T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.
- [73] F. J. Ferri, P. Pudil, M. Hatef, and J. Kittler. Comparative study of techniques for large-scale feature selection. In *Machine Intelligence and Pattern Recognition*, volume 16, pages 403–413. Elsevier, 1994.
- [74] A. L. Fradkov. Early history of machine learning. *IFAC-PapersOnLine*, 53(2):1385–1390, 2020.
- [75] P. W. Frederix, I. Patmanidis, and S. J. Marrink. Molecular simulations of self-assembling bio-inspired supramolecular systems and their connection to experiments. *Chemical Society Reviews*, 47(10):3470–3489, 2018.
- [76] P. W. Frederix, G. G. Scott, Y. M. Abul-Haija, D. Kalafatovic, C. G. Pappas, N. Javid, N. T. Hunt, R. V. Ulijn, and T. Tuttle. Exploring the sequence space for (tri-) peptide self-assembly to design and discover new hydrogels. *Nature Chemistry*, 7(1):30–37, 2015.
- [77] P. W. Frederix, R. V. Ulijn, N. T. Hunt, and T. Tuttle. Virtual screening for dipeptide aggregation: Toward predictive tools for peptide self-assembly. *The Journal of Physical Chemistry Letters*, 2(19):2380–2384, 2011.
- [78] M. Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, 32(200):675–701, 1937.
- [79] Y. Gao and T. Kodadek. Split-and-pool synthesis and characterization of peptide tertiary amide library. *Journal of Visualized Experiments: JoVE*, /(88):51299, 2014.
- [80] M. García-Ortegón, G. N. Simm, A. J. Tripp, J. M. Hernández-Lobato, A. Bender, and S. Bacallado. DOCKSTRING: Easy molecular docking yields better benchmarks for ligand design. *Journal of Chemical Information and Modeling*, 62(15):3486–3502, 2022.

- [81] G. Gocheva, K. Peneva, and A. Ivanova. Self-assembly of doxorubicin and a drug-binding peptide studied by molecular dynamics. *Chemical Physics*, 525:110380, 2019.
- [82] D. E. Goldberg. *Genetic algorithms*. pearson education India, 2013.
- [83] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.
- [84] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.
- [85] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint: 1706.02677*, 2017.
- [86] A. Graves. Long short-term memory. *Supervised Sequence Labelling With Recurrent Neural Networks*, pages 37–45, 2012.
- [87] J. G. Greener, L. Moffat, and D. T. Jones. Design of metalloproteins and novel protein folds using variational autoencoders. *Scientific Reports*, 8(1):16189, 2018.
- [88] P. S. Gromski, A. B. Henson, J. M. Granda, and L. Cronin. How to explore chemical space using algorithms and automation. *Nature Reviews Chemistry*, 3(2):119–128, 2019.
- [89] B. Guixer, X. Arroyo, I. Belda, E. Sabidó, M. Teixidó, and E. Giralt. Chemically synthesized peptide libraries as a new source of BBB shuttles. Use of mass spectrometry for peptide identification. *Journal of Peptide Science*, 22(9):577–591, 2016.
- [90] C. Guo, Y. Luo, R. Zhou, and G. Wei. Triphenylalanine peptides self-assemble into nanospheres and nanorods that are different from the nanovesicles and nanotubes formed by diphenylalanine peptides. *Nanoscale*, 6(5):2800–2811, 2014.
- [91] K. Guruprasad, B. B. Reddy, and M. W. Pandit. Correlation between stability of a protein and its dipeptide composition: A novel approach for predicting in vivo

- stability of a protein from its primary sequence. *Protein Engineering, Design and Selection*, 4(2):155–161, 1990.
- [92] P. J. Hajduk, W. R. Galloway, and D. R. Spring. A question of library design. *Nature*, 470(7332):42–43, 2011.
- [93] M. Hajfathalian, S. Ghelichi, P. J. García-Moreno, A.-D. Moltke Sørensen, and C. Jacobsen. Peptides: Production, bioactivity, functionality, and applications. *Critical Reviews in Food Science and Nutrition*, 58(18):3097–3129, 2018.
- [94] T. A. Halgren. Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94. *Journal of Computational Chemistry*, 17(5-6):490–519, 1996.
- [95] H. Hampel, J. Hardy, K. Blennow, C. Chen, G. Perry, S. H. Kim, V. L. Villemagne, P. Aisen, M. Vendruscolo, T. Iwatsubo, et al. The amyloid- β pathway in Alzheimer’s disease. *Molecular Psychiatry*, 26(10):5481–5503, 2021.
- [96] X. Hao, W. Song, Y. Wang, J. Qin, and Z. Jiang. Recent advancements in electrochemical sensors based on MOFs and their derivatives. *Small*, 21(4):2408624, 2025.
- [97] M. M. Hasan, N. Schaduangrat, S. Basith, G. Lee, W. Shoombuatong, and B. Manavalan. HLPpred-Fuse: Improved and robust prediction of hemolytic peptide and its activity by fusing multiple feature representation. *Bioinformatics*, 36(11):3350–3356, 2020.
- [98] K. Hasegawa, Y. Moriwaki, T. Terada, C. Wei, and K. Shimizu. Feedback-AVPGAN: Feedback-guided generative adversarial network for generating antiviral peptides. *Journal of Bioinformatics and Computational Biology*, 20(06):2250026, 2022.
- [99] C. A. Hauser, R. Deng, A. Mishra, Y. Loo, U. Khoe, F. Zhuang, D. W. Cheong, A. Accardo, M. B. Sullivan, C. Riekel, et al. Natural tri- to hexapeptides self-assemble in water to amyloid β -type fiber aggregates by unexpected α -helical intermediate structures. *Proceedings of the National Academy of Sciences*, 108(4):1361–1366, 2011.

- [100] A. Hawkins-Hooker, F. Depardieu, S. Baur, G. Couairon, A. Chen, and D. Bikard. Generating functional protein variants with variational autoencoders. *PLoS Computational Biology*, 17(2):e1008736, 2021.
- [101] E. S. Henault, M. H. Rasmussen, and J. H. Jensen. Chemical space exploration: How genetic algorithms find the needle in the haystack. *PeerJ Physical Chemistry*, 2:e11, 2020.
- [102] M. T. Heneka, W. M. Van der Flier, F. Jessen, J. Hoozemanns, D. R. Thal, D. Boche, F. Brosseron, C. Teunissen, H. Zetterberg, A. H. Jacobs, et al. Neuroinflammation in Alzheimer disease. *Nature Reviews Immunology*, 25(5):321–352, 2025.
- [103] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
- [104] W. G. Hoover. Canonical dynamics: Equilibrium phase-space distributions. *Physical Review A*, 31(3):1695, 1985.
- [105] K. A. Houck and R. J. Kavlock. Understanding mechanisms of toxicity: Insights from drug discovery research. *Toxicology and Applied Pharmacology*, 227(2):163–178, 2008.
- [106] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7132–7141, 2018.
- [107] K. Hu, Y. Jiang, W. Xiong, H. Li, P.-Y. Zhang, F. Yin, Q. Zhang, H. Geng, F. Jiang, Z. Li, et al. Tuning peptide self-assembly by an in-tether chiral center. *Science Advances*, 4(5):eaar5907, 2018.
- [108] K. Hu, W. Xiong, C. Sun, C. Wang, J. Li, F. Yin, Y. Jiang, M.-R. Zhang, Z. Li, X. Wang, et al. Self-assembly of constrained cyclic peptides controlled by ring size. *CCS Chemistry*, 2(1):42–51, 2020.
- [109] J. Huang, Y. Xu, Y. Xue, Y. Huang, X. Li, X. Chen, Y. Xu, D. Zhang, P. Zhang, J. Zhao, et al. Identification of potent antimicrobial peptides via a machine-learning

- pipeline that mines the entire space of peptide sequences. *Nature Biomedical Engineering*, 7(6):797–810, 2023.
- [110] M. A. Huffman, A. Fryszkowska, O. Alvizo, M. Borra-Garske, K. R. Campos, K. A. Canada, P. N. Devine, D. Duan, J. H. Forstater, S. T. Grosser, et al. Design of an in vitro biocatalytic cascade for the manufacture of islatravir. *Science*, 366(6470):1255–1259, 2019.
- [111] P. H. Hünenberger. Thermostat algorithms for molecular dynamics simulations. *Advanced Computer Simulation: Approaches for Soft Matter Sciences I*, pages 105–149, 2005.
- [112] M. Idhammad, K. Afdel, and M. Belouch. Detection system of HTTP DDoS attacks in a cloud environment based on information theoretic entropy and random forest. *Security and Communication Networks*, 2018(1):1263123, 2018.
- [113] A. Ikai. Thermostability and aliphatic index of globular proteins. *The Journal of Biochemistry*, 88(6):1895–1898, 1980.
- [114] P. Janković, E. Otović, G. Mauša, and D. Kalafatovic. Manually curated dataset of catalytic peptides for ester hydrolysis. *Data in Brief*, 48:109290, 2023.
- [115] P. Janković, I. Šantek, A. S. Pina, and D. Kalafatovic. Exploiting peptide self-assembly for the development of minimalistic viral mimetics. *Frontiers in Chemistry*, 9:723473, 2021.
- [116] J. H. Jensen. A graph-based genetic algorithm and generative model/Monte Carlo tree search for the exploration of chemical space. *Chemical Science*, 10(12):3567–3572, 2019.
- [117] M. Jindal, C. Le, M. Mohd Yusof, and S. Sekaran. Net charge, hydrophobicity and specific amino acids contribute to the activity of antimicrobial peptides. *J. Health Transl. Med*, 17:1–7, 2014.
- [118] C. W. Jones. Another nobel prize for catalysis: Frances Arnold in 2018, 2018.
- [119] L. H. Jones and M. E. Bunnage. Applications of chemogenomic library screening in drug discovery. *Nature Reviews Drug Discovery*, 16(4):285–296, 2017.

- [120] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, et al. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, 2021.
- [121] D. Jurafsky and J. H. Martin. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Prentice Hall PTR, USA, 1st edition, 2000.
- [122] D. Kalafatovic, G. Mauša, D. Rešetar Maslov, and E. Giralt. Bottom-up design approach for OBOC peptide libraries. *Molecules*, 25(15):3316, 2020.
- [123] D. Kalafatovic, G. Mauša, T. Todorovski, and E. Giralt. Algorithm-supported, mass and sequence diversity-oriented random peptide library design. *Journal of Cheminformatics*, 11(1):25, 2019.
- [124] X. Kang, F. Dong, C. Shi, S. Liu, J. Sun, J. Chen, H. Li, H. Xu, X. Lao, and H. Zheng. DRAMP 2.0, an updated data repository of antimicrobial peptides. *Scientific Data*, 6(1):148, 2019.
- [125] M. Karimi, S. Zhu, Y. Cao, and Y. Shen. De novo protein design for novel folds using guided conditional Wasserstein generative adversarial networks. *Journal of Chemical Information and Modeling*, 60(12):5667–5681, 2020.
- [126] J. Kim, T. H. Han, Y.-I. Kim, J. S. Park, J. Choi, D. G. Churchill, S. O. Kim, and H. Ihee. Role of water in directing diphenylalanine assembly into nanotubes and nanowires. *Advanced Materials*, 22(5):583–587, 2010.
- [127] M. D. King, T. Long, T. Andersen, and O. M. McDougal. Genetic algorithm managed peptide mutant screening: Optimizing peptide ligands for targeted receptor binding. *Journal of Chemical Information and Modeling*, 56(12):2378–2387, 2016.
- [128] D. P. Kingma. Adam: A method for stochastic optimization. *arXiv preprint: 1412.6980*, 2014.
- [129] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint: 1312.6114*, 2013.

- [130] M. Kiraly, J. Foss, and T. Giordano. Neuroinflammation, its role in Alzheimer’s disease and therapeutic strategies. *The Journal of Prevention of Alzheimer’s Disease*, 10(4):686–698, 2023.
- [131] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter. Self-normalizing neural networks. *Advances in Neural Information Processing Systems*, 30, 2017.
- [132] T. Kortemme, D. E. Kim, and D. Baker. Computational alanine scanning of protein-protein interfaces. *Science’s STKE*, 2004(219):pl2–pl2, 2004.
- [133] M. Kubat and S. Matwin. Addressing the curse of imbalanced training sets: One-sided selection. In *International Conference on Machine Learning*, volume 97, pages 179–186. Citeseer, 1997.
- [134] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.
- [135] S. La Manna, C. Di Natale, D. Florio, and D. Marasco. Peptides as therapeutic agents for inflammatory-related diseases. *International Journal of Molecular Sciences*, 19(9):2714, 2018.
- [136] C.-W. Lai, C.-Y. Lin, M.-C. Tsai, W.-J. Chen, C.-C. Hsieh, Z.-J. Lin, L.-J. Shen, Y.-L. Chen, L.-J. Lai, S.-H. Chen, et al. From AI to action: Antimicrobial peptides engineered by generative adversarial networks (GANs)- A novel approach to combat resistant bacteria. *Chemical Engineering Journal*, 519:164905, 2025.
- [137] A. Lampel. Biology-inspired supramolecular peptide systems. *Chem*, 6(6):1222–1236, 2020.
- [138] A. Lampel, R. Ulijn, and T. Tuttle. Guiding principles for peptide nanotechnology through directed discovery. *Chemical Society Reviews*, 47(10):3737–3758, 2018.
- [139] A. Langham and Y. N. Kaznessis. Molecular simulations of antimicrobial peptides. In *Antimicrobial peptides: Methods and protocols*, pages 267–285. Springer, 2009.
- [140] M. Larralde. Peptides.py. <https://github.com/althonos/peptides.py> (pristupljeno 30. ožujka 2022.), 2021.

- [141] P. Larranaga, C. M. H. Kuijpers, R. H. Murga, I. Inza, and S. Dizdarevic. Genetic algorithms for the travelling salesman problem: A review of representations and operators. *Artificial Intelligence Review*, 13(2):129–170, 1999.
- [142] P. W. Latham. Therapeutic peptides revisited. *Nature Biotechnology*, 17(8):755–757, 1999.
- [143] R. A. Lazarus, W. Wang, and Y. Zhang. Peptide inhibitors of BACE1 (WO2013056054A2), Apr 2013.
- [144] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.
- [145] B. Lee and F. M. Richards. The interpretation of protein structures: Estimation of static accessibility. *Journal of Molecular Biology*, 55(3):379–IN4, 1971.
- [146] O.-S. Lee, V. Cho, and G. C. Schatz. Modeling the self-assembly of peptide amphiphiles into fibers using coarse-grained molecular dynamics. *Nano Letters*, 12(9):4907–4913, 2012.
- [147] S. Lee, T. H. Trinh, M. Yoo, J. Shin, H. Lee, J. Kim, E. Hwang, Y.-b. Lim, and C. Ryou. Self-assembling peptides and their application in the treatment of diseases. *International Journal of Molecular Sciences*, 20(23):5850, 2019.
- [148] N. Lefin, L. Herrera-Belen, J. G. Farias, and J. F. Beltrán. Review and perspective on bioinformatics tools using machine learning and deep learning for predicting antiviral peptides. *Molecular Diversity*, 28(4):2365–2374, 2024.
- [149] A. L. Lehninger, D. L. Nelson, and M. M. Cox. Lehninger principles of biochemistry. *Macmillan*, 2005.
- [150] A. Levin, T. A. Hakala, L. Schnaider, G. J. Bernardes, E. Gazit, and T. P. Knowles. Biomimetic peptide self-assembly for functional materials. *Nature Reviews Chemistry*, 4(11):615–634, 2020.
- [151] F. Li, J. Han, T. Cao, W. Lam, B. Fan, W. Tang, S. Chen, K. L. Fok, and L. Li. Design of self-assembly dipeptide hydrogels and machine learning via their chemical

- features. *Proceedings of the National Academy of Sciences*, 116(23):11259–11264, 2019.
- [152] J. Li, A. Fu, and L. Zhang. An overview of scoring functions used for protein–ligand interactions in molecular docking. *Interdisciplinary Sciences: Computational Life Sciences*, 11(2):320–328, 2019.
- [153] Z. Li, Q. Miao, F. Yan, Y. Meng, and P. Zhou. Machine learning in quantitative protein–peptide affinity prediction: Implications for therapeutic peptide design. *Current Drug Metabolism*, 20(3):170–176, 2019.
- [154] Y. Liang, X. Zhang, Y. Yuan, Y. Bao, and M. Xiong. Role and modulation of the secondary structure of antimicrobial peptides to improve selectivity. *Biomaterials Science*, 8(24):6858–6866, 2020.
- [155] E. Lin, C.-H. Lin, and H.-Y. Lane. Relevant applications of generative adversarial networks in drug design and discovery: Molecular de novo design, dimensionality reduction, and de novo peptide and protein design. *Molecules*, 25(14):3250, 2020.
- [156] T. Lin, Y. Wang, X. Liu, and X. Qiu. A survey of transformers. *AI Open*, 3:111–132, 2022.
- [157] T.-T. Lin, L.-Y. Yang, C.-Y. Lin, C.-T. Wang, C.-W. Lai, C.-F. Ko, Y.-H. Shih, and S.-H. Chen. Intelligent de novo design of novel antimicrobial peptides against antibiotic-resistant bacteria strains. *International Journal of Molecular Sciences*, 24(7):6788, 2023.
- [158] T. W. Linsky, R. Vergara, N. Codina, J. W. Nelson, M. J. Walker, W. Su, C. O. Barnes, T.-Y. Hsiang, K. Esser-Nobis, K. Yu, et al. De novo design of potent and resilient hACE2 decoys to neutralize SARS-CoV-2. *Science*, 370(6521):1208–1214, 2020.
- [159] C. A. Lipinski. Lead-and drug-like compounds: The rule-of-five revolution. *Drug Discovery Today: Technologies*, 1(4):337–341, 2004.
- [160] C. A. Lipinski, F. Lombardo, B. W. Dominy, and P. J. Feeney. Experimental and

- computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced Drug Delivery Reviews*, 23(1-3):3–25, 1997.
- [161] D. J. Lipman and W. R. Pearson. Rapid and sensitive protein similarity searches. *Science*, 227(4693):1435–1441, 1985.
- [162] Z. Liu, J. Wang, Y. Luo, S. Zhao, W. Li, and S. Z. Li. Efficient prediction of peptide self-assembly through sequential and graphical encoding. *Briefings in Bioinformatics*, 24(6):bbad409, 2023.
- [163] T. L. Lopez-Silva and J. P. Schneider. From structure to application: Progress and opportunities in peptide materials development. *Current Opinion in Chemical Biology*, 64:131–144, 2021.
- [164] T. Ma, Y. Liu, B. Yu, X. Sun, H. Yao, C. Hao, J. Li, M. Nawaz, X. Jiang, X. Lao, et al. DRAMP 4.0: An open-access data repository dedicated to the clinical translation of antimicrobial peptides. *Nucleic Acids Research*, 53(D1):D403–D410, 2025.
- [165] L. v. d. Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.
- [166] R. Macarron, M. N. Banks, D. Bojanic, D. J. Burns, D. A. Cirovic, T. Garyantes, D. V. Green, R. P. Hertzberg, W. P. Janzen, J. W. Paslay, et al. Impact of high-throughput screening in biomedical research. *Nature Reviews Drug Discovery*, 10(3):188–195, 2011.
- [167] Y. Maeda, O. V. Makhlynets, H. Matsui, and I. V. Korendovych. Design of catalytic peptides and proteins through rational and combinatorial approaches. *Annual Review of Biomedical Engineering*, 18(1):311–328, 2016.
- [168] I. Mahmood and M. Pettinato. Impact of intrinsic and extrinsic factors on the pharmacokinetics of peptides: When is the assessment of certain factors warranted? *Antibodies*, 11(1):1, 2021.
- [169] B. Manavalan, S. Subramaniam, T. H. Shin, M. O. Kim, and G. Lee. Machine-learning-based prediction of cell-penetrating peptides and their uptake efficiency with improved accuracy. *Journal of Proteome Research*, 17(8):2715–2726, 2018.

- [170] D. Mandal, A. N. Shirazi, and K. Parang. Self-assembly of peptides to nanostructures. *Organic and Biomolecular Chemistry*, 12(22):3544–3561, 2014.
- [171] S. Markus, E. Pirogova, and T. J. Piva. Evaluation of the use of therapeutic peptides for cancer treatment. *Journal of Biomedical Science*, 24(1):21, 2017.
- [172] G. Mauša, M. Njirjak, E. Otović, and D. Kalafatovic. Configurable soft computing-based generative model: The search for catalytic peptides. *MRS Advances*, 8(19):1068–1074, 2023.
- [173] Q. McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.
- [174] S. Melkko, J. Scheuermann, C. E. Dumelin, and D. Neri. Encoded self-assembling chemical libraries. *Nature Biotechnology*, 22(5):568–574, 2004.
- [175] M. C. Melo, J. R. Maasch, and C. de la Fuente-Nunez. Accelerating antibiotic discovery through artificial intelligence. *Communications Biology*, 4(1):1050, 2021.
- [176] K. Mishima, H. Kaneko, and K. Funatsu. Development of a new de novo design algorithm for exploring chemical space. *Molecular Informatics*, 33(11-12):779–789, 2014.
- [177] B. J. Moncla, K. Pryke, L. C. Rohan, and P. W. Graebing. Degradation of naturally occurring and engineered antimicrobial peptides by proteases. *Advances in BioScience and Biotechnology*, 2(6):404, 2011.
- [178] G. M. Morris and M. Lim-Wilby. Molecular docking. In *Molecular Modeling of Proteins*, pages 365–382. Springer, 2008.
- [179] H. Mühlenbein and D. Schlierkamp-Voosen. Analysis of selection, mutation and recombination in genetic algorithms. In *Evolution and Biocomputation: Computational Models of Evolution*, pages 142–168. Springer, 2005.
- [180] M. Muttenthaler, G. F. King, D. J. Adams, and P. F. Alewood. Trends in peptide drug discovery. *Nature Reviews Drug Discovery*, 20(4):309–325, 2021.

- [181] A. T. Müller, J. A. Hiss, and G. Schneider. Recurrent neural network model for constructive peptide design. *Journal of Chemical Information and Modeling*, 58(2):472–479, 2018.
- [182] V. Nanda and R. L. Koder. Designing artificial enzymes by intuition and computation. *Nature Chemistry*, 2(1):15–24, 2010.
- [183] A. Naseem, F. Alturise, T. Alkhalifah, and Y. D. Khan. BBB-PEP-prediction: Improved computational model for identification of blood–brain barrier peptides using blending position relative composition specific features and ensemble modeling. *Journal of Cheminformatics*, 15(1):110, 2023.
- [184] K. Navaneetham and D. Arunachalam. Global population aging, 1950–2050. In *Handbook of aging, health and public policy: Perspectives from Asia*, pages 99–116. Springer, 2025.
- [185] N. M. Nawi, W. H. Atomi, and M. Z. Rehman. The effect of data pre-processing on optimized training of artificial neural networks. *Procedia Technology*, 11:32–39, 2013.
- [186] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453, 1970.
- [187] M. Negovetić, E. Otović, D. Kalafatovic, and G. Mauša. Efficiently solving the curse of feature-space dimensionality for improved peptide classification. *Digital Discovery*, 3(6):1182–1193, 2024.
- [188] P. K. Nguyen, W. Gao, S. D. Patel, Z. Siddiqui, S. Weiner, E. Shimizu, B. Sarkar, and V. A. Kumar. Self-assembly of a dentinogenic peptide hydrogel. *ACS Omega*, 3(6):5980–5987, 2018.
- [189] M. Njirjak, D. Kalafatovic, and G. Mauša. AI-driven design of β -secretase 1 inhibitors. In *37th European Peptide Symposium and 14th International Peptide Symposium*, 2024.

- [190] M. Njirjak, E. Otović, D. Kalafatovic, and G. Mauša. A multi-library approach to parallelised sequence space exploration: Streamlining library design and search space coverage. *Computers in Biology and Medicine*, 198:111207, 2025.
- [191] M. Njirjak, L. Žužić, M. Babić, P. Janković, E. Otović, D. Kalafatovic, and G. Mauša. Reshaping the discovery of self-assembling peptides with generative AI guided by hybrid deep learning. *Nature Machine Intelligence*, 6(12):1487–1500, 2024.
- [192] S. Nosé. A unified formulation of the constant temperature molecular dynamics methods. *The Journal of Chemical Physics*, 81(1):511–519, 1984.
- [193] N. M. O’Boyle, M. Banck, C. A. James, C. Morley, T. Vandermeersch, and G. R. Hutchison. Open Babel: An open chemical toolbox. *Journal of Cheminformatics*, 3(1):33, 2011.
- [194] B. Oller-Salvia, M. Sánchez-Navarro, E. Giralt, and M. Teixidó. Blood–brain barrier shuttle peptides: An emerging paradigm for brain delivery. *Chemical Society Reviews*, 45(17):4690–4707, 2016.
- [195] D. Osorio, P. Rondón-Villarreal, and R. Torres. Peptides: A package for data mining of antimicrobial peptides. <https://github.com/dosorio/Peptides> (pristupljeno 30. ožujka 2022.), 2015.
- [196] E. Otović, M. Njirjak, I. Žužić, D. Kalafatovic, and G. Mauša. Genetic algorithm parametrization for informed exploration of short peptides chemical space. In *2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pages 1–3. IEEE, 2020.
- [197] E. Otović, M. Njirjak, D. Kalafatovic, and G. Mauša. Sequential properties representation scheme for recurrent neural network-based prediction of therapeutic peptides. *Journal of Chemical Information and Modeling*, 62(12):2961–2972, 2022.
- [198] L. Otvos Jr. Peptide-based drug research and development: Relative costs, comparative value. *Pharmaceutical Outsourcing*, 15:16–20, 2014.
- [199] L. Otvos Jr, I. Kovalszky, L. Scolaro, A. Sztodola, J. Olah, M. Cassone, D. Knappe, R. Hoffmann, S. Lovas, M. P. Hatfield, et al. Peptide-based leptin receptor antago-

- nists for cancer treatment and appetite regulation. *Peptide Science*, 96(2):117–125, 2011.
- [200] N. Palmer, J. R. Maasch, M. D. Torres, and C. de la Fuente-Nunez. Molecular dynamics for antimicrobial peptide discovery. *Infection and Immunity*, 89(4):e00703–20, 2021.
- [201] Y. Pang, L. Yao, J.-H. Jhong, Z. Wang, and T.-Y. Lee. AVPIden: A new scheme for identification and functional prediction of antiviral peptides based on machine learning approaches. *Briefings in Bioinformatics*, 22(6):bbab263, 2021.
- [202] M. Parrinello and A. Rahman. Crystal structure and pair potentials: A molecular-dynamics study. *Physical Review Letters*, 45(14):1196, 1980.
- [203] M. Parrinello and A. Rahman. Polymorphic transitions in single crystals: A new molecular dynamics method. *Journal of Applied Physics*, 52(12):7182–7190, 1981.
- [204] M. Parrinello and A. Rahman. Strain fluctuations and elastic constants. *The Journal of Chemical Physics*, 76(5):2662–2666, 1982.
- [205] R. Pascanu, T. Mikolov, and Y. Bengio. On the difficulty of training recurrent neural networks. In *International Conference on Machine Learning*, pages 1310–1318. Pmlr, 2013.
- [206] A. Patle and D. S. Chouhan. SVM kernel functions for classification. In *2013 International Conference on Advances in Technology and Engineering (ICATE)*, pages 1–9. IEEE, 2013.
- [207] Z. Peng, W. Luo, Y. Lu, T. Shen, C. Gulino, A. Seff, and J. Fu. Improving agent behaviors with RL fine-tuning for autonomous driving. In *European Conference on Computer Vision*, pages 165–181. Springer, 2024.
- [208] S. D. Pickett, D. V. S. Green, D. L. Hunt, D. A. Pardoe, and I. Hughes. Automated lead optimization of MMP-12 inhibitors using a genetic algorithm. *ACS Medicinal Chemistry Letters*, 2(1):28–33, 2011.
- [209] N. A. Pierce and E. Winfree. Protein design is NP-hard. *Protein Engineering*, 15(10):779–782, 2002.

- [210] W. F. Porto, L. Irazazabal, E. S. Alves, S. M. Ribeiro, C. O. Matos, Á. S. Pires, I. C. Fensterseifer, V. J. Miranda, E. F. Haney, V. Humblot, et al. In silico optimization of a guava antimicrobial peptide enables combinatorial exploration for peptide design. *Nature Communications*, 9(1):1490, 2018.
- [211] W. F. Porto, L. Irazazabal, E. S. F. Alves, S. M. Ribeiro, C. O. Matos, A. S. Pires, I. C. M. Fensterseifer, V. J. Miranda, E. F. Haney, V. Humblot, M. D. T. Torres, R. E. W. Hancock, L. M. Liao, A. Ladram, T. K. Lu, C. de la Fuente-Nunez, and O. L. Franco. In silico optimization of a guava antimicrobial peptide enables combinatorial exploration for peptide design. *Nature Communications*, 9(1):1490, 2018.
- [212] X. Qiang, C. Zhou, X. Ye, P.-f. Du, R. Su, and L. Wei. CPPred-FL: A sequence-based predictor for large-scale identification of cell-penetrating peptides by feature representation learning. *Briefings in Bioinformatics*, 21(1):11–23, 2020.
- [213] A. J. Quartararo, Z. P. Gates, B. A. Somsen, N. Hartrampf, X. Ye, A. Shimada, Y. Kajihara, C. Ottmann, and B. L. Pentelute. Ultra-large chemical libraries for the discovery of high-affinity peptide binders. *Nature Communications*, 11(1):3183, 2020.
- [214] P. Quesada Martinez and M. Sanchez Navarro. One-bead-one-compound (OBOC) combinatorial libraries for the discovery of ligands towards SIRPa and checkpoint CD47 proteins. In *Journal of Peptide Science*, volume 30. Wiley, 2024.
- [215] A. Qureshi, N. Thakur, H. Tandon, and M. Kumar. AVPdb: A database of experimentally validated antiviral peptides targeting medically important viruses. *Nucleic Acids Research*, 42(D1):D1147–D1153, 2014.
- [216] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al. Improving language understanding by generative pre-training. *OpenAI*, 2018.
- [217] M. S. Raffi and P. S. Aisen. Amyloid-lowering immunotherapies for Alzheimer disease: Current status and future directions. *Nature Reviews Neurology*, 21(9):490–498, 2025.

- [218] M. A. Rao, R. Scelza, R. Scotti, and L. Gianfreda. Role of enzymes in the remediation of polluted environments. *Journal of Soil Science and Plant Nutrition*, 10(3):333–353, 2010.
- [219] J. Rey, S. Murail, S. de Vries, P. Derreumaux, and P. Tuffery. PEP-FOLD4: A pH-dependent force field for peptide structure prediction in aqueous solution. *Nucleic Acids Research*, 51(W1):W432–W437, 2023.
- [220] P. A. Romero and F. H. Arnold. Exploring protein fitness landscapes by directed evolution. *Nature Reviews Molecular Cell Biology*, 10(12):866–876, 2009.
- [221] A. Rossetto and W. Zhou. Gandalf: Peptide generation for drug design using sequential and structural generative adversarial networks. In *Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 1–10, 2020.
- [222] A. M. Rossetto and W. Zhou. Gandalf: A prototype of a gan-based peptide design method. In *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 61–66, 2019.
- [223] S. Saha and G. P. S. Raghava. BcePred: Prediction of continuous B-cell epitopes in antigenic sequences using physico-chemical properties. In *International Conference on Artificial Immune Systems*, pages 197–204. Springer, 2004.
- [224] N. Schaduengrat, C. Nantasenamat, V. Prachayasittikul, and W. Shoombuatong. Meta-iAVP: A sequence-based meta-predictor for improving the prediction of anti-viral peptides using effective feature representation. *International Journal of Molecular Sciences*, 20(22):5743, 2019.
- [225] L. M. Schmitt. Theory of genetic algorithms. *Theoretical Computer Science*, 259(1-2):1–61, 2001.
- [226] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.
- [227] Scikit-bio development team. scikit-bio: A bioinformatics library for data scientists,

- students, and developers. <http://scikit-bio.org> (verzija 0.5.7; pristupljeno 30. ožujka 2022.), 2022.
- [228] N. C. Seeman. DNA in a material world. *Nature*, 421(6921):427–431, 2003.
- [229] L. Séguirel and C. Bon. On the evolution of lactase persistence in humans. *Annual Review of Genomics and Human Genetics*, 18:297–319, 2017.
- [230] K. Sharma, K. K. Sharma, A. Sharma, and R. Jain. Peptide-based drug discovery: Current status and recent advances. *Drug Discovery Today*, 28(2):103464, 2023.
- [231] S. Sharma, S. Sharma, and A. Athaiya. Activation functions in neural networks. *International Journal of Engineering Applied Sciences and Technology*, 4(12):310–316, 2020.
- [232] A. Sherstinsky. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena*, 404:Article number: 132306, 2020.
- [233] A. N. Shivanyuk, S. V. Ryabukhin, A. Tolmachev, A. Bogolyubsky, D. Mykytenko, A. Chupryna, W. Heilman, and A. Kostyuk. Enamine real database: Making chemical diversity real. *Chemistry Today*, 25(6):58–59, 2007.
- [234] K. Shmilovich, R. A. Mansbach, H. Sidky, O. E. Dunne, S. S. Panda, J. D. Tovar, and A. L. Ferguson. Discovery of self-assembling π -conjugated peptides by active learning-directed coarse-grained molecular simulation. *The Journal of Physical Chemistry B*, 124(19):3873–3891, 2020.
- [235] T. Shtatland, D. Guettler, M. Kossodo, M. Pivovarov, and R. Weissleder. PepBank—a database of peptides based on sequence text mining and public peptide data sources. *BMC Bioinformatics*, 8(1):280, 2007.
- [236] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419):1140–1144, 2018.

- [237] D. Singh and B. Singh. Investigating the impact of data normalization on classification performance. *Applied Soft Computing*, 97:105524, 2020.
- [238] S. Singh, K. Chaudhary, S. K. Dhanda, S. Bhalla, S. S. Usmani, A. Gautam, A. Tunknait, P. Agrawal, D. Mathur, and G. P. Raghava. SATPdb: A database of structurally annotated therapeutic peptides. *Nucleic Acids Research*, 44(D1):D1119–D1126, 2016.
- [239] A. Slowik and H. Kwasnicka. Evolutionary algorithms and their applications to engineering problems. *Neural Computing and Applications*, 32(16):12363–12379, 2020.
- [240] Y. Song, F. Wang, and X. Chen. An improved genetic algorithm for numerical function optimization. *Applied Intelligence*, 49(5):1880–1902, 2019.
- [241] J. O. Spiegel and J. D. Durrant. AutoGrow4: An open-source genetic algorithm for de novo drug design and lead optimization. *Journal of Cheminformatics*, 12(1):1–16, 2020.
- [242] I. Springer, H. Besser, N. Tickotsky-Moskovitz, S. Dvorkin, and Y. Louzoun. Prediction of specific TCR-peptide binding from large dictionaries of TCR-peptide pairs. *Frontiers in Immunology*, 11:1803, 2020.
- [243] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.
- [244] K. O. Stanley, J. Clune, J. Lehman, and R. Miikkulainen. Designing neural networks through neuroevolution. *Nature Machine Intelligence*, 1(1):24–35, 2019.
- [245] R. Storn. Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces. *Technical Report, International Computer Science Institute*, 11, 1995.
- [246] S. Surana, P. Arora, D. Singh, D. Sahasrabudde, and J. Valadi. Pandoragan: Generating antiviral peptides using generative adversarial network. *SN Computer Science*, 4(5):607, 2023.

- [247] C. Tackenberg, L. Kulic, and R. M. Nitsch. Familial Alzheimer’s disease mutations at position 22 of the amyloid β -peptide sequence differentially affect synaptic loss, tau phosphorylation and neuronal cell death in an ex vivo system. *PloS One*, 15(9):e0239584, 2020.
- [248] A. A. Tahami Monfared, M. J. Byrnes, L. A. White, and Q. Zhang. Alzheimer’s disease: Epidemiology and clinical progression. *Neurology and Therapy*, 11(2):553–569, 2022.
- [249] K. Takahashi, T. Oda, and K. Naruse. Coarse-grained molecular dynamics simulations of biomolecules. *AIMS Biophysics*, 1(1):1–15, 2014.
- [250] S. Tangirala. Evaluating the impact of GINI index and information gain on classification using decision tree classifier algorithm. *International Journal of Advanced Computer Science and Applications*, 11(2):612–619, 2020.
- [251] H. S. Tehrani and A. A. Moosavi-Movahedi. Catalase and its mysteries. *Progress in Biophysics and Molecular Biology*, 140:5–12, 2018.
- [252] S. Thapa, F. Clark, S. T. Schneebeli, and J. Li. Multiscale simulations to discover self-assembled oligopeptides: A benchmarking study. *Journal of Chemical Theory and Computation*, 20(1):375–384, 2023.
- [253] The UniProt Consortium. UniProt: The universal protein knowledgebase in 2025. *Nucleic Acids Research*, 53(D1):D609–D617, 2025.
- [254] I. Torjesen. Drug development: The journey of a medicine from lab to shelf. *The Pharmaceutical Journal*, 12:926–932, 2015.
- [255] O. Trott and A. J. Olson. AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of Computational Chemistry*, 31(2):455–461, 2010.
- [256] A. Tucs, D. P. Tran, A. Yumoto, Y. Ito, T. Uzawa, and K. Tsuda. Generating ampicillin-level antimicrobial peptides with activity-aware generative adversarial networks. *ACS Omega*, 5(36):22847–22851, 2020.

- [257] C. G. Turhan and H. S. Bilge. Recent trends in deep generative models: A review. In *2018 3rd International Conference on Computer Science and Engineering (UBMK)*, pages 574–579. IEEE, 2018.
- [258] A. Turing. Intelligent machinery. /, pages 1–21, 1948.
- [259] S. Van Dorpe, A. Bronselaer, J. Nielandt, S. Stalmans, E. Wynendaele, K. Aude-naert, C. Van De Wiele, C. Burvenich, K. Peremans, H. Hsuehou, et al. Brain-peps: The blood–brain barrier peptide database. *Brain Structure and Function*, 217(3):687–718, 2012.
- [260] A. van Teijlingen and T. Tuttle. Beyond tripeptides Two-step active machine learning for very large data sets. *Journal of Chemical Theory and Computation*, 17(5):3221–3232, 2021.
- [261] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.
- [262] S. Vatansever, A. Schlessinger, D. Wacker, H. Ü. Kaniskan, J. Jin, M.-M. Zhou, and B. Zhang. Artificial intelligence and machine learning-aided drug discovery in central nervous system diseases: State-of-the-arts and future directions. *Medicinal Research Reviews*, 41(3):1427–1473, 2021.
- [263] M. Verleysen and D. François. The curse of dimensionality in data mining and time series prediction. In *International Work-Conference on Artificial Neural Networks*, pages 758–770. Springer, 2005.
- [264] V. Vovk. The fundamental nature of the log loss function. In *Fields of Logic and Computation II: Essays Dedicated To Yuri Gurevich on the Occasion of His 75th Birthday*, pages 307–318. Springer, 2015.
- [265] J. Vrbancac and R. Slauter. ADME in drug discovery. In *A comprehensive guide to toxicology in nonclinical drug development*, pages 39–67. Elsevier, 2017.
- [266] L. C. Walker, D. G. Lynn, and Y. O. Chernoff. A standard model of Alzheimer’s disease? *Prion*, 12(5-6):261–265, 2018.

- [267] F. Wan, D. Kontogiorgos-Heintz, and C. de la Fuente-Nunez. Deep generative models for peptide design. *Digital Discovery*, 1(3):195–208, 2022.
- [268] F. Wan, F. Wong, J. J. Collins, and C. de la Fuente-Nunez. Machine learning for antimicrobial peptide identification and design. *Nature Reviews Bioengineering*, 2(5):392–407, 2024.
- [269] B. Wang, A. Wang, F. Chen, Y. Wang, and C.-C. J. Kuo. Evaluating word embedding models: Methods and experimental results. *APSIPA Transactions on Signal and Information Processing*, 8:e19, 2019.
- [270] H. Wang, J. A. Kulas, C. Wang, D. M. Holtzman, H. A. Ferris, and S. B. Hansen. Regulation of beta-amyloid production in neurons by astrocyte-derived cholesterol. *Proceedings of the National Academy of Sciences*, 118(33):e2102191118, 2021.
- [271] L. Wang, N. Wang, W. Zhang, X. Cheng, Z. Yan, G. Shao, X. Wang, R. Wang, and C. Fu. Therapeutic peptides: Current applications and future directions. *Signal Transduction and Targeted Therapy*, 7(1):48, 2022.
- [272] S. Wang, M. T. Bergman, C. K. Hall, and F. You. De Novo Design of Multiple Microplastic-Binding Peptides with a Protein Language Model-Guided Generative Adversarial Network. *Journal of Chemical Information and Modeling*, 2025.
- [273] L. Wei, X. Ye, T. Sakurai, Z. Mu, and L. Wei. ToxIBTL: Prediction of peptide toxicity based on information bottleneck and transfer learning. *Bioinformatics*, 38(6):1514–1524, 2022.
- [274] L. Wei, C. Zhou, H. Chen, J. Song, and R. Su. ACPred-FL: A sequence-based predictor using effective feature representation to improve the prediction of anti-cancer peptides. *Bioinformatics*, 34(23):4007–4016, 2018.
- [275] J. A. C. Weideman. Numerical integration of periodic functions: A few examples. *The American Mathematical Monthly*, 109(1):21–36, 2002.
- [276] S. A. Wildman and G. M. Crippen. Prediction of physicochemical parameters by atomic contributions. *Journal of Chemical Information and Computer Sciences*, 39(5):868–873, 1999.

- [277] D. H. Wolpert and W. G. Macready. Coevolutionary free lunches. *IEEE Transactions on Evolutionary Computation*, 9(6):721–735, 2005.
- [278] World Health Organization and others. Global status report on the public health response to dementia. *World Health Organization*, 2021.
- [279] W. Xiao, W. Jiang, Z. Chen, Y. Huang, J. Mao, W. Zheng, Y. Hu, and J. Shi. Advance in peptide-based drug development: Delivery platforms, therapeutics and vaccines. *Signal Transduction and Targeted Therapy*, 10(1):74, 2025.
- [280] J. Xu, F. Li, A. Leier, D. Xiang, H.-H. Shen, T. T. Marquez Lago, J. Li, D.-J. Yu, and J. Song. Comprehensive assessment of machine learning-based methods for predicting antimicrobial peptides. *Briefings in Bioinformatics*, 22(5):bbab083, 2021.
- [281] T. Xu, J. Wang, S. Zhao, D. Chen, H. Zhang, Y. Fang, N. Kong, Z. Zhou, W. Li, and H. Wang. Accelerating the prediction and discovery of peptide hydrogels with human-in-the-loop. *Nature Communications*, 14(1):3880, 2023.
- [282] X. Yan, Y. Cui, Q. He, K. Wang, J. Li, W. Mu, B. Wang, and Z.-c. Ou-yang. Reversible transitions between peptide nanotubes and vesicle-like structures including theoretical modeling studies. *Chemistry—A European Journal*, 14(19):5974–5980, 2008.
- [283] K. K. Yang, Z. Wu, and F. H. Arnold. Machine-learning-guided directed evolution for protein engineering. *Nature Methods*, 16(8):687–694, 2019.
- [284] L. M. Yin, M. A. Edwards, J. Li, C. M. Yip, and C. M. Deber. Roles of hydrophobicity and charge distribution of cationic antimicrobial peptides in peptide-membrane interactions. *Journal of Biological Chemistry*, 287(10):7738–7745, 2012.
- [285] M. Yoshida, T. Hinkley, S. Tsuda, Y. M. Abul-Haija, R. T. McBurney, V. Kulikov, J. S. Mathieson, S. G. Reyes, M. D. Castro, and L. Cronin. Using evolutionary algorithms and machine learning to explore sequence space for the discovery of antimicrobial peptides. *Chem*, 4(3):533–543, 2018.

- [286] M. Youmans, C. Spainhour, and P. Qiu. Long short-term memory recurrent Neural networks for antibacterial peptide identification. In *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 498–502, 2017.
- [287] L. Yu, R. Jing, F. Liu, J. Luo, and Y. Li. DeepACP: A novel computational approach for accurate identification of anticancer peptides by deep learning algorithm. *Molecular Therapy Nucleic Acids*, 22:862–870, 2020.
- [288] Y. Yu, X. Si, C. Hu, and J. Zhang. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation*, 31(7):1235–1270, 2019.
- [289] B. Zdrazil, E. Felix, F. Hunter, E. J. Manners, J. Blackshaw, S. Corbett, M. De Veij, H. Ioannidis, D. M. Lopez, J. F. Mosquera, et al. The ChEMBL Database in 2023: A drug discovery platform spanning multiple bioactivity data types and time periods. *Nucleic Acids Research*, 52(D1):D1180–D1192, 2024.
- [290] W.-F. Zeng, X.-X. Zhou, S. Willems, C. Ammar, M. Wahle, I. Bludau, E. Voytik, M. T. Strauss, and M. Mann. AlphaPeptDeep: A modular deep learning framework to predict peptide properties for proteomics. *Nature Communications*, 13(1):7238, 2022.
- [291] H. Zhang, Y. Wang, Y. Zhu, P. Huang, Q. Gao, X. Li, Z. Chen, Y. Liu, J. Jiang, Y. Gao, et al. Machine learning and genetic algorithm-guided directed evolution for the development of antimicrobial peptides. *Journal of Advanced Research*, 2024.
- [292] J. Zhang, Y. Xie, Q. Wu, and Y. Xia. Medical image classification using synergic deep learning. *Medical Image Analysis*, 54:10–19, 2019.
- [293] W. Zhao, K. Hou, Y. Shen, and X. Hu. A conditional denoising VAE-based framework for antimicrobial peptides generation with preserving desirable properties. *Bioinformatics*, 41(2):btaf069, 2025.
- [294] A. Zheng and A. Casari. *Feature engineering for machine learning: Principles and techniques for data scientists*. „O’Reilly Media, Inc.”, 2018.
- [295] H. Zheng and E. H. Koo. The amyloid precursor protein: Beyond amyloid. *Molecular Neurodegeneration*, 1(1):5, 2006.

- [296] J. Zhou, S. Chen, J. Xia, S. Sizhe Liu, T. Ling, W. Du, Y. Liu, J. Yin, and S. Z. Li. Novobench: Benchmarking deep learning-based De Novo sequencing methods in proteomics. *Advances in Neural Information Processing Systems*, 37:104776–104791, 2024.
- [297] P. Zhou, C. Yuan, and X. Yan. Computational approaches for understanding and predicting the self-assembled peptide hydrogels. *Current Opinion in Colloid and Interface Science*, page 101645, 2022.
- [298] O. Zozulia, M. Dolan, and I. Korendovych. Catalytic peptide assemblies. *Chemical Society Reviews*, 47(10):3621–3639, 2018.

POPIS SLIKA

- 1.1 Shematski prikaz razvoja modela strojnog učenja, uključujući (a) filtriranje podataka, (b) odabir i transformaciju značajki, (c) raspodjelu skupa podataka na podskup za učenje, validaciju i testiranje, (d) proces učenja, (e) optimizaciju hiperparametara na podskupu za validaciju, (f) konačno testiranje modela i (g) eksploataciju. 5
- 1.2 Prikaz (a) arhitekture neuronske mreže koja se sastoji od ulaznog sloja s dva neurona, središnjeg sloja s tri neurona, i izlaznog sloja s jednim neuronom. Neuron svakog sloja povezan je sa svim neuronima prethodnog i sljedećeg sloja. Vidljiv je i (b) detaljan presjek načina rada svakog neurona, pri čemu se svaka ulazna vrijednost a_i množi s pripadajućom težinom w_i , nakon čega slijedi sumacija i primjena aktivacijske funkcije. 8
- 1.3 Prikaz procesa stvaranja stabla odluke. Čvorovi se rekurzivno dijele sve dok se ne dostigne uvjet za prestanak. Odluka o pripadnosti listu donosi se prolaskom od korijena do lista, iterativno prateći niz naučenih pravila. 9
- 1.4 Shematski prikaz pronalaženja hiperravnine pri učenju modela stroja potpornih vektora. Hiperravnina se nalazi između potpornih vektora dviju klasa. 10
- 1.5 Shematski prikaz procesa razvoja modela generativnih suparničkih mreža. Posebno je označen dio generiranja novih podataka koji se odvajaju od ostatka modela nakon uspješnog učenja i koristi u fazi eksploatacije. 12
- 1.6 Prikaz varijacijskog autoenkodera i njegova tri osnova dijela: enkodera, latentnog prostora i dekodera. 13

1.7	Shematski prikaz načina rada genetskog algoritma. Proces najprije (a) generira nasumičnu populaciju, nakon čega slijedi ciklus iteracija (b) selekcije, (c) rekombinacije i (d) mutacije, nakon čega algoritam vraća (e) konačnu populaciju.	15
1.8	Stvaranje (a) potpuno nasumične populacije peptida i (b) populacije koja zadržava motiv MCE...K u svim sekvencama.	16
1.9	Prikaz čestih postupaka provođenja selekcije roditelja: (a) odabir kotačem sreće, gdje svaka jedinka dobiva vjerojatnost odabira sukladno svojoj kvaliteti, (b) turnirska selekcija, gdje se naprije vrši nasumičan odabir k jedinki iz populacije (crveni obrub), te se među njima bira ona najbolja (crvena točka) sukladno vrijednosti funkcije dobrote.	17
1.10	Prikaz križanja kategoričkih genoma triju roditelja s dvije točke prijeloma.	18
1.11	(a) Vizualizacija skupa vrijednosti kojeg je moguće postići rekombinacijom i mutacijom, te (b) primjer tri tipa mutacije kategoričkih genoma: zamjena mjesta dviju vrijednosti, skraćivanje genoma, te produljenje genoma. . . .	19
1.12	Prikaz čestih mehanizama odabira jedinki za sljedeću generaciju: (a) elitistički odabir isključivo najboljih jedinki i (b) elitizam uz zadržavanje nasumičnog odabira u preostalom dijelu populacije.	20
2.1	Shematski prikaz istraživanja. Svaka od 4 predstave peptida testirena je u dvaput ponovljenoj unakrsnoj validaciji u 10 preklopa za tri antiviralna i jedan antimikrobni skup podataka, te vrednovana pomoću sedam metrika. Preuzeto iz [197] i prevedeno.	27
2.2	Razdioba duljina pozitivnih (antimikrobnih/antiviralnih) peptida označenih crveno i negativnih peptida označenih plavo u skupovima podataka korištenim u istraživanju. Preuzeto iz [197] i prevedeno.	29
2.3	Histogrami sličnosti peptida dobiveni koristeći globalno poravnanje Needleman-Wunsch. Preuzeto iz [197] i prevedeno.	30

- 2.4 Pregled korištenih shema predstave peptida u računalu (označene žutim pravokutnicima) za peptid DCEQGHAR: (a) fizikalno-kemijska svojstva peptida, (b) binarno kodiranje s jednim aktivnim elementom, (c) ugrađivanje riječi (primjer s veličinom sloja od 128 decimalnih vrijednosti), (d) shema sekvencijalnih svojstava. S obzirom na svoju sekvencijalnu prirodu, podaci u (b), (c) i (d) se obrađuju povratnim neuronskim mrežama, odnosno slojevima duge kratkoročne memorije (označeni sivim trokutima), dok se podaci u (a) obrađuju višeslojnim perceptronom. Izlaz neuronske mreže predstavlja predviđenu vjerojatnost da ulazni peptid posjeduje anti-mikrobnu/antiviralnu aktivnost. Preuzeto iz [197] i prevedeno. 32
- 2.5 Arhitekture modela korištenih neuronskih mreža za: a) višeslojni perceptron, b) povratnu neuronsku mrežu za shemu ugrađivanja riječi, c) konvolucijsko-povratnu neuronsku mrežu za sheme binarnog kodiranja s jednim aktivnim mjestom i sekvencijalna svojstva. Preuzeto iz [197] i prevedeno. 36
- 2.6 Vizualizacija postupka provedbe eksperimenata na svakom od četiri skupa podataka. Preuzeto iz [197] i prevedeno. 38
- 2.7 Krivulje ROC opisuju ponašanje modela na sva četiri skupa podataka: a) AVPdb b) AVDRAMP c) AVMerged d) AMP. Za svaku krivulju prikazana je srednja vrijednost i standardna devijacija. Preuzeto iz [197] i prevedeno. 40
- 2.8 Histogramima je prikazana razdioba vrijednosti F1 koje su modeli ostvarili na sva četiri skupa podataka: a) AVPdb b) AVDRAMP c) AVMerged d) AMP. Vrijednosti medijana prikazane su narančastim linijama, a srednje vrijednosti i standardne devijacije simbolima μ i σ^2 . Preuzeto iz [197] i prevedeno. 41
- 3.1 Histogram broja pojavljivanja određenih duljina katalitičkih peptida (označeni plavo) i peptida kod kojih katalitička aktivnost izostaje (označeni crveno) u korištenom skupu podataka. Preuzeto iz [114] i prevedeno. . . . 45

- 3.2 Shematski prikaz načina rada genetskog algoritma. Nasumična početna populacija podvrgava se ciklusima selekcije, rekombinacije i mutacije fiksan broj iteracija. Krajnja populacija optimizirana je prema kriterijima korištenim u vrednovanju, u ovom slučaju primarno prema predviđenoj vjerojatnosti za katalitičku aktivnost, a sekundarno prema duljini i sličnosti. Preuzeto iz [172] i prevedeno. 46
- 3.3 Vrednovanje generativnog modela koji koristi model predviđanja temeljen na arhitekturi RNN. (a) Rezultati vrednovanja i sve metrike modela predviđanja (medijan je prikazan narančastom linijom). (b) Prosječna duljina peptida u populaciji genetskog algoritma kroz generacije bez postavljene željene duljine (plavo) te s postavljenom željenom duljinom (zeleno). Prikazana je i usporedba razdioba aminokiseline u skupu podataka za učenje i (c) generiranih peptida bez postavljene željene duljine, te podataka za učenje i (e) generiranih peptida s postavljenom željenom duljinom. Analiza sličnosti također je provedena odvojeno za (d) slučaj bez postavljene željene duljine, te za (f) slučaj s postavljenim željenim rasponom duljina. Preuzeto iz [172] i prevedeno. 49
- 4.1 Shematski prikaz metodologije istraživanja. Ulazni podaci modela su vrijednosti sklonosti agregaciji (AP) za aminokiseline, dipeptide, i tripeptide sekvence koja se procesira, uz dodatak izračunatih sekvencijalnih svojstava (SP). Predobrada podataka uključivala je primjenu algoritma t-SNE za smanjenje dimenzionalnosti podataka, skaliranje ulaznih vrijednosti u raspon $[-1, 1]$ za poboljšanje protoka gradijenata kroz mrežu i uspješnije učenje, te nadopunu sekvenci do duljine najveće sekvence u skupu podataka (24 aminokiseline) kako bi se ubrzalo učenje. Modeli su naučeni uz optimizaciju hiperparametara koristeći ugniježđenu unakrsnu validaciju, nakon čega je najbolji model primijenjen u generativnom genetskom algoritmu za dizajn novih peptida. Preuzeto iz [191] i prevedeno. 56
- 4.2 Razdioba peptida sa svojstvom samosastavljanja (označeni plavo) i peptida bez svojstva samosastavljanja (označeni crveno) u skupu korištenih podataka, prema duljini. Preuzeto iz [191] i prevedeno. 56

- 4.3 (a) Shematski prikaz načina rada tehnike pomičnog prozora za izdvajanje aminokiselina, dipeptida i tripeptida. (b) Prikaz ulaznih podataka za peptid NFGAIL i hibridni model AP-SP. Preuzeto iz [191] i prevedeno. 58
- 4.4 Arhitekture neuronskih mreža predloženih modela za (a) Model AP koji koristi isključivo informacije o gradivnim elementima peptida, (b) Model SP koji predviđanje formira na temelju vrijednosti sekvencijalnih svojstava, te (c) Hibridni model AP-SP koji kombinira prethodna dva. Preuzeto iz [191] i prevedeno. 60
- 4.5 Učenje modela izvedeno je pomoću dviju petlji unakrsnih validacija, svaka u pet preklopa. Vanjska petlja obavlja učenje modela i krajnje testiranje na testom preklopu, dok unutarnja petlja provodi validaciju i odabir hiperparametara. Završni rezultati svih preklopa usrednjuju se, te se tako formiraju konačne performanse modela. Preuzeto iz [191] i prevedeno. 61
- 4.6 Usporedba krivulja gubitka na preklopima za učenje i validaciju u slučaju modela AP-SP. Krivulje su prikazane za svih 5 ponavljanja pokusa i svih 5 testnih preklopa. Preuzeto iz [191] i prevedeno. 68
- 4.7 Histogram točnosti klasifikacije na agregiranim testnim podacima za modele (a) AP, (b) SP, (c) AP-SP, (d) t-SNE SP i (e) t-SNE AP-SP. Crvenom bojom označena je pozitivna klasa, odnosno peptidi sa svojstvom samosastavljanja (SA), dok su plavom bojom označene sekvence bez navedenog svojstva (NSA). Idealna klasifikacija svakoj molekuli SA dodjeljuje vrijednost 1, dok svakoj molekuli NSA dodjeljuje 0. Preuzeto iz [191] i prevedeno. 69
- 4.8 Dvodimenzionalni prikaz generiranih sekvenci, njihovih međusobnih sličnosti, te sličnosti u odnosu na peptide korištene pri učenju modela predviđanja za (a) heksapeptide i (b) peptide duljine između 5 i 10 aminokiselina. Dodatno, napravljena je i (c) usporedba frekvencija aminokiselina među trima populacijama peptida. Preuzeto iz [191] i prevedeno. 73
- 4.9 Prikaz početnih i završnih trenutaka simulacija molekularne dinamike za heksapeptide i peptide duljine između 5 i 10 aminokiselina koji imaju visoku vjerojatnost samosastavljanja. Preuzeto iz [191] i prevedeno. 74

4.10	Prikaz početnih i završnih trenutaka simulacija molekularne dinamike za heksapeptide i peptide duljine između 5 i 10 aminokiselina koji imaju nisku vjerojatnost samosastavljanja. Preuzeto iz [191] i prevedeno.	75
4.11	Dvodimenzionalni razmještaj (a) heksapeptida i (b) peptida duljine između 5 i 10 u odnosu na kriterije $AP_{kontakt}$ i AP_{SASA} , te njihove definirane pragove. Peptidi s visokom vjerojatnošću samosastavljanja označeni su plavo i zeleno, ovisno o duljini, dok su peptidi s niskom vjerojatnošću samosastavljanja označeni crveno. Preuzeto iz [191] i prevedeno.	76
4.12	Rezultati laboratorijskog ispitivanja pet najperspektivnijih peptida prema kriteriju vjerojatnosti samosastavljanja. (a) Prozirnost otopine izmjerena postupkom OD600. (b) Ispitivanje fluorescencije peptida. (c) Spektar FTIR za pet ispitanih peptida. (d) Prikaz provedene fluorescentne mikroskopije. (e) Mikroskopija TEM. Preuzeto iz [191] i prevedeno.	78
4.13	Prikaz rezultata triju nezavisnih eksperimenata konvergencije generativnog modela. Vidljivo je povećanje sličnosti peptida i smanjenje broja jedinstvenih sekvenci kroz iteracije. Preuzeto iz [191] i prevedeno.	79
5.1	Prikaz mehanizma funkcioniranja inhibitora enzima beta-sekretaze 1. Vežanjem za enzim, inhibitor sprječava vezanje amiloidnog prekursorskog proteina i njegovu razgradnju. Preuzeto iz [189] i prevedeno.	82
5.2	Prikaz napredovanja optimizacije kroz praćenje minimalne, maksimalne i prosječne vrijednosti svakog kriterija na cijeloj populaciji.	85
5.3	Analiza sličnosti parova peptida u konačnoj populaciji. Peptidi prosječno dijele manje od 35% sličnosti.	86
5.4	Razdioba aminokiselina u konačnoj populaciji. Vidljive su povećane koncentracije leucina, prolina i triptofana, te smanjeno korištenje glicina, arginina i treonina.	87
5.5	Histogram duljina peptida u konačnoj populaciji. Vidljiva je tendencija algoritma ka stvaranju sekvenci kraćih od 10 aminokiselina, pri čemu prosječna duljina iznosi 6 aminokiselina.	87

- 6.1 Shematski prikaz (a) načina računanja broja peptida koji se ne preklapaju po masi unutar jednog podskupa (zelene točke nasuprot crvenim točkama), te broja identičnih peptida među dvama podskupovima (ED i ER). Prikazani su i postupci računanja (b) ukupnog broja peptida kojeg podskupovi sadrže, (c) srednjeg postotka peptida koji se ne preklapaju po masi, te (d) pokazatelja međusobne sličnosti podskupova. Vidljiv je i (e) način kodiranja knjižnica u algoritmu i njegova konverzija u podskupove, te (f) način računanja broja peptida koji se pojavljuju u dva podskupa. Preuzeto iz [190] i prevedeno. 92
- 6.2 Ispitivanje (a) memorijskih i (b) vremenskih zahtjeva za algoritme izračunavanja broja peptida koji se ne preklapaju po masi metodama iscrpnog pretraživanja (plavo) i simuliranog kaljenja (crveno). Izbor metode sukladno kraćem vremenu izvršavanja osigurava ispunjavanje memorijskih zahtjeva. Preuzeto iz [190] i prevedeno. 96
- 6.3 Razdvajanje knjižnice peptida „SER:WAG:IR” u jedan, dva, i tri podskupa. Različitost po masi, međusobna različitost podskupova, te peptidi sadržani u podskupovima prikazani su za odabrana rješenja (označena crvenim točkama na grafovima). Vennovi dijagrami demonstriraju prednosti razdvajanje knjižnice peptida na veći broj podskupova. Preuzeto iz [190] i prevedeno. 100
- 6.4 Prikaz oscilacija površina ispod parova krivulja funkcija vrednovanja za vrijeme optimizacije. Naznačena je površina u svakoj iteraciji, te maksimalna površina postignuta do iteracije. Ako se površina ne poveća u 100 uzastopnih iteracija, izvođenje algoritma se prekida. Grafovi prikazuju površine za krivulje omeđene vrijednostima: (a) ukupnog broja peptida (FF_1) i različitosti po masi (FF_2), (b) ukupnog broja peptida (FF_1) i međusobne različitosti podskupova (FF_3), i (c) različitosti po masi (FF_2) i međusobne različitosti podskupova (FF_3). Podaci su zabilježeni pri optimizaciji knjižnice peptida „LWHTV:WP:HDICKSREAWQ:IGAHRFSVDY:IHMDVA:IQ” i traženju 5 podskupova (tablica 6.2). Preuzeto iz [190] i prevedeno. 107

POPIS TABLICA

2.1	Dvadeset proteinogenih aminokiselina zajedno s pripadajućim jednoslovnim kodovima za predstavu primarne strukture peptida/proteina.	26
2.2	Pregled korištenih skupova podataka i izvora peptidnih sekvenci. Preuzeto iz [197] i prevedeno.	28
2.3	Usporedba fizikalno-kemijskih svojstava dvaju peptida s istim sastavom, ali drugačijim poretom aminokiselina. Sva svojstva imaju identične vrijednosti, uz iznimku hidrofobnog momenta (podebljano). Preuzeto iz [197] i prevedeno.	33
2.4	Rezultati vrednovanja na svim skupovima podataka; najbolje vrijednosti su podebljane. Vrijednosti metrika kod kojih nije pronađena statistički značajna razlika u odnosu na istu metriku sheme sekvencijalnih svojstava, označene su zvjezdicom. U određenim slučajevima, preciznost nije mogla biti pouzdano izračunata jer model na podskupu za testiranje nije predvidio niti jednu pozitivnu sekvencu, pa su takvi slučajevi označeni znakom tilde. Preuzeto iz [197] i prevedeno.	42
3.1	Najboljih 15 peptida sukladno predviđenoj vjerojatnosti katalitičke aktivnosti za oba ispitna slučaja. Preuzeto iz [172] i prevedeno.	50
4.1	Moguće vrijednosti za odabir pri optimizaciji hiperparametara modela. Preuzeto iz [191] i prevedeno.	61

4.2	Dvadeset pentapeptida iz [11] s eksperimentalno potvrđenim statusom samosastavljanja. Procjena praga samosastavljanja za daljnje analize napravljena je minimizacijom broja lažno pozitivnih (označeni crveno) i lažno negativnih primjera pri svrstavanju peptida u binarne klase. Dodatno su označeni stvarno pozitivni podaci (zeleno) i stvarno negativni podaci (žuto). Preuzeto iz [191] i prevedeno.	64
4.3	Najčešće odabrane vrijednosti hiperparametara pri učenju i optimizaciji modela. U zagradama je naveden broj slučajeva u kojima je navedeni parametar odabran u odnosu na ukupni broj slučajeva. Preuzeto iz [191] i prevedeno.	67
4.4	Prikaz vrijednosti metrika ostvarenih prilikom učenja modela. Uz maksimalne i minimalne vrijednosti točnosti i gubitka, navedene su i prosječne vrijednosti uz standardnu devijaciju. Preuzeto iz [191] i prevedeno.	69
4.5	Prikaz performansi ispitanih modela na testnom skupu podataka za različite metrike i pragove. Pragovi ROC i PR određeni su na temelju udaljenosti do idealne točke klasifikacije na pripadajućim krivuljama, a ispitano je i standardni prag od 0,5. Preuzeto iz [191] i prevedeno.	70
4.6	Usporedba modela iz literature i predloženog modela AP-SP. Modeli su uspoređeni na temelju metrike gmean, vrijednosti F1 i točnosti klasifikacije. Ispitano je postojanje statistički značajnih razlika (McNemarov test, $\alpha = 0,05$, $n = 1840$) između AP-SP i ostalih modela, a napravljena je i procjena performansi na 20 pentapeptida iz literature [11]. Preuzeto iz [191] i prevedeno.	71
4.7	Pet najboljih peptida prema kriteriju vjerojatnosti samosastavljanja iz dva eksperimenta stvaranja heksapeptida i peptida duljine između 5 i 10 amino-kiselina. Vjerojatnost samosastavljanja procijenjena je hibridnim modelom AP-SP. Metrike $S_{učenje}$ i $S_{generirani}$ ukazuju na sličnost primarne strukture peptida s peptidima korištenim za učenje modela predviđanja, odnosno ostalim generiranim peptidima. $AP_{kontakt}$ i AP_{SASA} izračunati su na temelju prvog i zadnjeg vremenskog trenutka simulacije molekularne dinamike. Preuzeto iz [191] i prevedeno.	72

4.8	Pet najboljih peptida dviju kontrolnih skupina, stvorenih tako da imaju što nižu vjerojatnost samosastavljanja. Vjerojatnost samosastavljanja procijenjena je hibridnim modelom AP-SP. $AP_{kontakt}$ i AP_{SASA} izračunati su na temelju prvog i zadnjeg vremenskog trenutka simulacije molekularne dinamike. Preuzeto iz [191] i prevedeno.	74
4.9	Lista peptida stvorenih u trima eksperimentima ispitivanja konvergencije generativnog modela. Uz peptide navedena je i njihova procijenjena sklonost ka samosastavljanju, te strukturni motivi koji se pojavljuju. Pri višestrukom pojavljivanju iste sekvence u populaciji, u prikazu je zadržan isključivo jedan primjer. Preuzeto iz [191] i prevedeno.	80
5.1	Prikaz nasumično odabranih peptida s prve nedominirane fronte konačne populacije molekula i usporedba s patentiranim inhibitorima enzima beta-sekretaze 1 iz literature. Preuzeto iz [189] i prevedeno.	86
5.2	Analiza odabranih peptida prema Lipinskijevim pravilima.	88
6.1	Rezultati optimizacije knjižnice peptida koja sadrži 19 aminokiselina razdijeljenih na 6 pozicija, pri čemu se ona razdvajala u tri, četiri i pet podskupova. Naznačeni su i veličina prostora pretraživanja, veličina populacije, maksimalan i izveden broj iteracija, te postoci korištenja svake od dviju metoda za izračun različitosti po masi. Slučajevi u kojima je došlo do aktivacije mehanizma ranog zaustavljanja označeni su zeleno. Za svaki ispitni slučaj odabrano je proizvoljno rješenje s Paretove fronte i uspoređeno je s rješenjem pronađenim metodom nasumičnog pretraživanja. Preuzeto iz [190] i prevedeno.	102

- 6.2 Rezultati optimizacije knjižnice peptida koja sadrži 36 aminokiselina razdijeljenih na 6 pozicija, pri čemu se ona razdvajala u tri, četiri i pet podskupova. Naznačeni su i veličina prostora pretraživanja, veličina populacije, maksimalan i izveden broj iteracija, te postoci korištenja svake od dviju metoda za izračun različitosti po masi. Slučajevi u kojima je došlo do aktivacije mehanizma ranog zaustavljanja označeni su zeleno. Za svaki ispitni slučaj odabrano je proizvoljno rješenje s Paretove fronte i uspoređeno je s rješenjem pronađenim metodom nasumičnog pretraživanja. Preuzeto iz [190] i prevedeno. 103
- 6.3 Rezultati optimizacije knjižnice peptida koja sadrži 53 aminokiselina razdijeljenih na 8 pozicija, pri čemu se ona razdvajala u tri, četiri i pet podskupova. Naznačeni su i veličina prostora pretraživanja, veličina populacije, maksimalan i izveden broj iteracija, te postoci korištenja svake od dviju metoda za izračun različitosti po masi. Slučajevi u kojima je došlo do aktivacije mehanizma ranog zaustavljanja označeni su zeleno. Za svaki ispitni slučaj odabrano je proizvoljno rješenje s Paretove fronte i uspoređeno je s rješenjem pronađenim metodom nasumičnog pretraživanja. Preuzeto iz [190] i prevedeno. 104
- 6.4 Rezultati optimizacije knjižnice peptida koja sadrži 83 aminokiselina razdijeljenih na 10 pozicija, pri čemu se ona razdvajala u tri, četiri i pet podskupova. Naznačeni su i veličina prostora pretraživanja, veličina populacije, maksimalan i izveden broj iteracija, te postoci korištenja svake od dviju metoda za izračun različitosti po masi. Slučajevi u kojima je došlo do aktivacije mehanizma ranog zaustavljanja označeni su zeleno. Za svaki ispitni slučaj odabrano je proizvoljno rješenje s Paretove fronte i uspoređeno je s rješenjem pronađenim metodom nasumičnog pretraživanja. Preuzeto iz [190] i prevedeno. 105

- 6.5 Rezultati optimizacije knjižnice peptida koja sadrži 28 aminokiselina razdijeljenih na 6 pozicija, pri čemu se ona razdvajala u tri, četiri i pet podskupova. Naznačeni su i veličina prostora pretraživanja, veličina populacije, maksimalan i izveden broj iteracija, te postoci korištenja svake od dviju metoda za izračun različitosti po masi. Slučajevi u kojima je došlo do aktivacije mehanizma ranog zaustavljanja označeni su zeleno. Za svaki ispitni slučaj odabrano je proizvoljno rješenje s Paretove fronte i uspoređeno je s rješenjem pronađenim metodom nasumičnog pretraživanja. Preuzeto iz [190] i prevedeno. 106
- A.1 Skup peptida korišten pri učenju modela predviđanja za pronalazak peptida sa svojstvom samosastavljanja. Preuzeto iz [191] i prevedeno. 165
- B.1 Usporedba metoda za procjenu preklapanja peptida po masi. Simulirano kaljenje uspoređeno je s metodom iscrpnog pretraživanja koristeći ispitne slučajeve različitih složenosti. Svaki redak tablice predstavlja jedan ispitni slučaj, a njegova složenost indicirana je duljinom peptida, brojem aminokiselina-kandidata po svakoj poziciji, te veličinom prostora pretraživanja. Greška metode simuliranog kaljenja u odnosu na iscrpno pretraživanje promatrana je kao rezultat. Uzmemo li u obzir ispitne slučajeve u kojima se simulirano kaljenje koristi u algoritmu predloženom u poglavlju 6. (veličina prostora pretraživanja veća od $9,8 \cdot 10^6$, podebljano u tablici), prosječna greška iznosi manje od 3%. Preuzeto iz [190], prilagođeno i prevedeno. 178

POPIS ALGORITAMA

- 1 Iscrpno pretraživanje broja peptida koji se ne preklapaju po masi u ulaznoj knjižnici peptida. Preuzeto iz [190] i prevedeno. 92
- 2 Računanje numeričkog pokazatelja preklapanja podskupova. Više vrijednosti indiciraju manji broj preklapanja. Preuzeto iz [190] i prevedeno. . . . 94
- 3 Procjena broja peptida koji se ne preklapaju po masi metodom simuliranog kaljenja. Algoritam se koristi kao surogat model iscrpnom pretraživanju kako bi se omogućila optimizacija u slučajevima u kojima prostor pretrage sadrži više od $9,8 \cdot 10^6$ mogućih rješenja. Prosječna greška procjene u odnosu na stvarni broj preklapanja ne prelazi 3%. Preuzeto iz [190] i prevedeno. . . 95

POPIS KRATICA

AAC	aminokiselinski sastav (eng. amino acid composition)
ADME	apsorpcija, distribucija, metabolizam i izlučivanje (eng. absorption, distribution, metabolism and excretion)
AI	umjetna inteligencija (eng. artificial intelligence)
AP	sklonost agregaciji (eng. aggregation propensity)
APP	amiloidni prekursorski protein
ATR-FTIR	atenuirana totalna refleksija infracrvene spektroskopije s Fourierovom transformacijom (eng. attenuated total reflectance Fourier-transform infrared spectroscopy)
CTF	zajednička trijada (eng. conjoint triad)
DPC	dipeptidni sastav (eng. dipeptide composition)
GAN	generativna suparnička mreža (eng. generative adversarial network)
GM	geometrijska sredina (eng. geometric mean)
GPU	grafička procesna jedinica (eng. graphical processing unit)
LLM	veliki jezični model (eng. large language model)
LSTM	duga kratkoročna memorija (eng. long short-term memory)
MCC	Matthewsov koeficijent korelacije (eng. Matthews correlation coefficient)
MD	molekularna dinamika
ML	strojno učenje (eng. machine learning)
MLP	višeslojni perceptron (eng. multilayer perceptron)
MP	molekularno pristajanje
NLP	procesiranje prirodnog jezika (eng. natural language processing)

NP	eng. nondeterministic polynomial-time
NSA	peptidi bez svojstva samosastavljanja (eng. no self-assembly)
OD	optička gustoća (eng. optical density)
PR	preciznost i odziv (eng. precision-recall)
RAM	memorija nasumičnog pristupa (eng. random access memory)
RNA	ribonukleinska kiselina (eng. ribonucleic acid)
RNN	povratna neuronska mreža (eng. recurrent neural network)
ROC-AUC	površina ispod krivulje karakteristične funkcije prijemnika (eng. area under receiver operating characteristic curve)
ROS	reaktivne vrste kisika (eng. reactive oxygen species)
SA	peptidi sa svojstvom samosastavljanja (eng. self-assembly)
SASA	površina dostupna otapalu (eng. solvent accesible surface area)
SIMT	jedna instrukcija izvršena u više dretvi (eng. single instruction, multiple threads)
SP	sekvencijalna svojstva (eng. sequential properties)
SVM	stroj potpornih vektora (eng. support vector machine)
t-SNE	t-distribuirano stohastičko ugrađivanje u susjedstvo (eng. t-distributed stochastic neighbour embedding)
TEM	transmisijska elektronska mikroskopija (eng. transmission electron microscopy)
VAE	varijacijski autoenkoder (eng. variational autoencoder)

PRIVITCI

A. Peptidi s eksperimentalno ispitanim svojstvom samosastavljanja

Tablica A.1: Skup peptida korišten pri učenju modela predviđanja za pronalazak peptida sa svojstvom samosastavljanja. Preuzeto iz [191] i prevedeno.

Peptidi sa svojstvom samosastavljanja

Peptid	DOI znanstvenog rada
QEIRARLEQEIRARLEYEIRARLE	10.1002/anie.200604014
KFG	10.1002/anie.201307247
KLDLKLDLKLDL	10.1007/s11596-009-0424-6
QMAVHV	10.1016/j.bbrc.2016.12.011
VKVKVKVRVP'PTRVKVKVKV	10.1016/j.biomaterials.2012.08.046
VKVKVRVRVP'PTRVRVKVKV	10.1016/j.biomaterials.2012.08.046
VKVRVRVRVP'PTRVRVRVKV	10.1016/j.biomaterials.2012.08.046
VRVRVRVRVP'PTRVRVRVRV	10.1016/j.biomaterials.2012.08.046
GGVVIA	10.1016/j.bpj.2011.02.060
GNNQQNY	10.1016/j.bpj.2011.02.060
SSTSA	10.1016/j.bpj.2011.02.060
RADARADFRADARADA	10.1016/j.jconrel.2010.08.002
RADFRADARADARADA	10.1016/j.jconrel.2010.08.002
FGAIL	10.1021/acs.analchem.5b03454
NININI	10.1021/acs.analchem.5b03454
SFSFSF	10.1021/acs.analchem.5b03454
VQIVYK	10.1021/acs.analchem.5b03454
INHVGGTGGAIMAPVAVTRQLVGS	10.1021/acs.bioconjchem.9b00327
VVVVVVK	10.1021/acs.biomac.5b00850
FFSGHFDF	10.1021/acs.biomac.6b01169
FGFHFSFDF	10.1021/acs.biomac.6b01169
AAVVLLW	10.1021/acs.molpharmaceut.6b01003
GGGAAVVVHIKKK	10.1021/acsami.7b11681

IIIVVAAAGGGKKK	10.1021/acsami.7b11681
KKKAAAIIGGGVVV	10.1021/acsami.7b11681
KKKGGGAAAVVVIII	10.1021/acsami.7b11681
KKKIIIVVAAAGGG	10.1021/acsami.7b11681
VVVGGGIIIAAKKK	10.1021/acsami.7b11681
KLFFF	10.1021/bi8000616
AEAEAEAEAKAKAKAK	10.1021/bm0341374
AEAEAKAKAEAEAKAK	10.1021/bm0341374
AAVLLLLWEE	10.1021/bm0704267
AAVLLLLWEEEEEEEE	10.1021/bm0704267
KKSLSLSLSLSLK	10.1021/bm1010195
EFFAAE	10.1021/bm301141h
KFFAAK	10.1021/bm301141h
WVNAKQYWRILKRRW	10.1021/ja049477r
YVNAKQYYRILKRRY	10.1021/ja049477r
GAIL	10.1021/jp904251j
GFIL	10.1021/jp904251j
IIIIKK	10.1021/la801605b
LLLLLKK	10.1021/la801605b
GAVILRR	10.1021/la8025232
GGGGDD	10.1021/nl025563i
GGGGGGDD	10.1021/nl025563i
GGGGGGGGDD	10.1021/nl025563i
GGGGGGGGGGDD	10.1021/nl025563i
KYF	10.1038/nchem.2122
FKFEF	10.1038/s41557-022-01055-3
FKIDF	10.1038/s41557-022-01055-3
KFAFD	10.1038/s41557-022-01055-3
KFFFE	10.1038/s41557-022-01055-3
KWEFY	10.1038/s41557-022-01055-3

KWMDF	10.1038/s41557-022-01055-3
SYCGY	10.1038/s41557-022-01055-3
VKVEV	10.1038/s41557-022-01055-3
VKVFF	10.1038/s41557-022-01055-3
VVVVV	10.1038/s41557-022-01055-3
FFEKF	10.1038/s41557-022-01055-3
KFFDY	10.1038/s41557-022-01055-3
RWLDY	10.1038/s41557-022-01055-3
RATARAEARATARA	10.1039/c9tb02590a
FYLLYY	10.1073/pnas.0511298103
KDWSFY	10.1073/pnas.0511298103
KIVKWD	10.1073/pnas.0511298103
LLYYTE	10.1073/pnas.0511298103
LVEALY	10.1073/pnas.0511298103
LYQLEN	10.1073/pnas.0511298103
NHVTLS	10.1073/pnas.0511298103
VEALYL	10.1073/pnas.0511298103
SAVIIE	10.1073/pnas.2634884100
SDVIIE	10.1073/pnas.2634884100
SEVIIE	10.1073/pnas.2634884100
SFVIIE	10.1073/pnas.2634884100
SGVIIE	10.1073/pnas.2634884100
SIVIIE	10.1073/pnas.2634884100
SLVIIE	10.1073/pnas.2634884100
SMVIIE	10.1073/pnas.2634884100
SNVIIE	10.1073/pnas.2634884100
SQVIIE	10.1073/pnas.2634884100
SSVIIE	10.1073/pnas.2634884100
STEIIE	10.1073/pnas.2634884100
STFIIE	10.1073/pnas.2634884100

STLIIE	10.1073/pnas.2634884100
STNIIE	10.1073/pnas.2634884100
STQIIE	10.1073/pnas.2634884100
STSIIE	10.1073/pnas.2634884100
STVEIE	10.1073/pnas.2634884100
STVFIE	10.1073/pnas.2634884100
STVIFE	10.1073/pnas.2634884100
STVIIA	10.1073/pnas.2634884100
STVIID	10.1073/pnas.2634884100
STVIIE	10.1073/pnas.2634884100
STVIIF	10.1073/pnas.2634884100
STVIII	10.1073/pnas.2634884100
STVIIL	10.1073/pnas.2634884100
STVIIM	10.1073/pnas.2634884100
STVIIN	10.1073/pnas.2634884100
STVIIQ	10.1073/pnas.2634884100
STVIIS	10.1073/pnas.2634884100
STVIIT	10.1073/pnas.2634884100
STVIIV	10.1073/pnas.2634884100
STVIIW	10.1073/pnas.2634884100
STVIHY	10.1073/pnas.2634884100
STVIYE	10.1073/pnas.2634884100
STVLIE	10.1073/pnas.2634884100
STVNIE	10.1073/pnas.2634884100
STVTIE	10.1073/pnas.2634884100
STVWIE	10.1073/pnas.2634884100
STVYIE	10.1073/pnas.2634884100
SVVIIIE	10.1073/pnas.2634884100
SYVIIIE	10.1073/pnas.2634884100
TTVIIIE	10.1073/pnas.2634884100

VTVIIIE	10.1073/pnas.2634884100
WTVIIIE	10.1073/pnas.2634884100
ATVIIIE	10.1073/pnas.2634884100
DTVIIIE	10.1073/pnas.2634884100
ETVIIIE	10.1073/pnas.2634884100
FTVIIIE	10.1073/pnas.2634884100
GTVIIIE	10.1073/pnas.2634884100
ITVIIIE	10.1073/pnas.2634884100
LTVIIIE	10.1073/pnas.2634884100
MTVIIIE	10.1073/pnas.2634884100
NTVIIIE	10.1073/pnas.2634884100
QTVIIIE	10.1073/pnas.2634884100
YTVIIIE	10.1073/pnas.2634884100
NFGSVQ	10.1080/13506120412331272287
IKVAV	10.1126/science.1093783
DFY	10.1126/science.aal5005
FAGFT	10.21203/rs.3.rs-109949/v1
FFVDF	10.21203/rs.3.rs-109949/v1
FFVLK	10.21203/rs.3.rs-109949/v1
FGFDP	10.21203/rs.3.rs-109949/v1
FTISD	10.21203/rs.3.rs-109949/v1
ISDNL	10.21203/rs.3.rs-109949/v1
ITSVV	10.21203/rs.3.rs-109949/v1
LDFPI	10.21203/rs.3.rs-109949/v1
LIVGD	10.21203/rs.3.rs-109949/v1
YFTEF	10.21203/rs.3.rs-109949/v1
KLVFFAE	10.1529/biophysj.106.088542
CEIEIQI	10.1002/adfm.202009382
CEIEIQINM	10.1002/adfm.202009382
CEIEIQINMWQ	10.1002/adfm.202009382

CKAKAQA	10.1002/adfm.202009382
CKAKAQANM	10.1002/adfm.202009382
CKAKAQANMWQ	10.1002/adfm.202009382
CKFKFQF	10.1002/adfm.202009382
CKFKFQFNM	10.1002/adfm.202009382
CKFKFQFNMWQ	10.1002/adfm.202009382
CKIKIQI	10.1002/adfm.202009382
CKIKIQIN	10.1002/adfm.202009382
CKIKIQINMWQ	10.1002/adfm.202009382
CKIKQII	10.1002/adfm.202009382
CKIKQIINM	10.1002/adfm.202009382
CKIKQIINMWQ	10.1002/adfm.202009382
CSISIQI	10.1002/adfm.202009382
CSISIQINM	10.1002/adfm.202009382
KFKFQF	10.1002/adfm.202009382
KFKFQFN	10.1002/adfm.202009382
KFKFQFNMWQ	10.1002/adfm.202009382
KIKIQINM	10.1002/adfm.202009382
KIKIQINMWQ	10.1002/adfm.202009382
KIKQIINM	10.1002/adfm.202009382
KIKQIINMWQ	10.1002/adfm.202009382
MKFKFQF	10.1002/adfm.202009382
MKIKIQI	10.1002/adfm.202009382
KAKAQA	10.1002/adfm.202009382
KAKAQAN	10.1002/adfm.202009382
KAKAQANMWQ	10.1002/adfm.202009382
YFCFY	10.1002/advs.201801255
YYACAYY	10.1002/advs.201801255
MEMEMKMKMEMEMKMK	10.1002/bip.21340
QQEFQWQFRQQ	10.1002/bit.22274

RADARADARADARADA	10.1016/0142-9612(95)96874-Y
RARADADARARADADA	10.1016/0142-9612(95)96874-Y
GFFL	10.1016/j.biomaterials.2016.04.039
GFFLG	10.1016/j.biomaterials.2016.04.039
GFFLGLDD	10.1016/j.biomaterials.2016.04.039
FLVHSS	10.1016/S1047-8477(02)00630-5
AAKAAAKAAAKAAKA	10.1021/acs.jpcc.6b07673
PRFRFRFRF	10.1021/acs.langmuir.0c00198
RFRFRFRF	10.1021/acs.langmuir.0c00198
GYGYGYG	10.1021/acs.nano.8b00417
NYNNYN	10.1021/acs.nano.8b00417
NYNNYN	10.1021/acs.nano.8b00417
QYQQYQ	10.1021/acs.nano.8b00417
QYQQYQ	10.1021/acs.nano.8b00417
SYSSYS	10.1021/acs.nano.8b00417
SYSYSYS	10.1021/acs.nano.8b00417
FFALGLAGKK	10.1021/acs.nano.8b07401
FFGAAGLDGD	10.1021/acs.nano.8b07401
FFGAAGLKGG	10.1021/acs.nano.8b07401
FFGALGLDGD	10.1021/acs.nano.8b07401
FFGALGLKGG	10.1021/acs.nano.8b07401
FFGPAGLDGD	10.1021/acs.nano.8b07401
FFGPAGLKGG	10.1021/acs.nano.8b07401
FFGPLGLDGD	10.1021/acs.nano.8b07401
FFGPLGLKGG	10.1021/acs.nano.8b07401
FFPLGLAGDD	10.1021/acs.nano.8b07401
FFPLGLAGKK	10.1021/acs.nano.8b07401
NNQQNY	10.1021/jp4046287
NNVVNY	10.1021/jp4046287
NVQVVY	10.1021/jp4046287

NVVQIY	10.1021/jp4046287
NVVVVY	10.1021/jp4046287
YVIFL	10.1021/jp502473s
YVVFL	10.1021/jp502473s
YVVFV	10.1021/jp502473s
VVVVVVKK	10.1021/la026526+
FKFEFKFE	10.1021/la100623y
FKFEFKFEFKFE	10.1021/la100623y
AAAAAAK	10.1021/la802499n
IIK	10.1021/la804175h
VEVE	10.1021/nl802813f
VVEE	10.1021/nl802813f
PPPPPPPPPRKKRRQRRR	10.1039/B719868J
PPPPPPPPPRRR	10.1039/B719868J
LFF	10.1039/C2NR32006A
AAAAAAAAAAAAARR	10.1039/C3SM50303H
WRWRWR	10.1039/C3SM50764E
WRWRWRWR	10.1039/C3SM50764E
WRWRWRWRWR	10.1039/C3SM50764E
RATARAEFRATARA EA	10.1039/C4RA04393F
LLLLLLDD	10.1073/pnas.072089599
VVVVVVD	10.1073/pnas.072089599
VVVVVVDD	10.1073/pnas.072089599
AAAAAAD	10.1073/pnas.072089599
AIVAGD	10.1073/pnas.1014796108
AIVAGS	10.1073/pnas.1014796108
AIVAGT	10.1073/pnas.1014796108
IID	10.1073/pnas.1014796108
ILVAGD	10.1073/pnas.1014796108
ILVAGS	10.1073/pnas.1014796108

IVAD	10.1073/pnas.1014796108
IVD	10.1073/pnas.1014796108
LAVAGD	10.1073/pnas.1014796108
LIVAAD	10.1073/pnas.1014796108
LIVAD	10.1073/pnas.1014796108
LIVAGD	10.1073/pnas.1014796108
LIVAGE	10.1073/pnas.1014796108
LIVAGK	10.1073/pnas.1014796108
LIVAGS	10.1073/pnas.1014796108
LIVAGT	10.1073/pnas.1014796108
FFKLVFF	10.1074/jbc.271.15.8545
YTIAALLSPYS	10.1103/PhysRevLett.96.238301
KASAKAEAKAEAKASA	10.1177/0885328210379927
SAEAKAEAKAEASAEA	10.1177/0885328210379927
NFGAIL	10.1006/jmbi.1999.3422
AILSS	10.1021/jacs.3c11712

Peptidi bez svojstva samosastavljanja

Peptid	DOI znanstvenog rada
KIKQII	10.1002/adfm.202009382
DPDPD	10.1038/s41557-022-01055-3
EPYYK	10.1038/s41557-022-01055-3
KDHFY	10.1038/s41557-022-01055-3
KDPYY	10.1038/s41557-022-01055-3
KKFDD	10.1038/s41557-022-01055-3
KVKVK	10.1038/s41557-022-01055-3
PPPHY	10.1038/s41557-022-01055-3
PTPCY	10.1038/s41557-022-01055-3
RVSVD	10.1038/s41557-022-01055-3
VKVKV	10.1038/s41557-022-01055-3
WKPY Y	10.1038/s41557-022-01055-3

YDPKY	10.1038/s41557-022-01055-3
YEPYK	10.1038/s41557-022-01055-3
YTEYK	10.1038/s41557-022-01055-3
LIAGFN	10.1006/jmbi.1999.3422
NLGPVL	10.1006/jmbi.1999.3422
TGTHSTSDT	10.1021/acs.biomac.6b01169
TTSGHDTDT	10.1021/acs.biomac.6b01169
AASGHADAA	10.1021/acs.biomac.6b01171
AGAHASADA	10.1021/acs.biomac.6b01171
KTVIIE	10.1073/pnas.2634884100
KTVIIT	10.1073/pnas.2634884100
KTVIVE	10.1073/pnas.2634884100
KTVIYE	10.1073/pnas.2634884100
KTVLIE	10.1073/pnas.2634884100
PTVIIE	10.1073/pnas.2634884100
SPVIIE	10.1073/pnas.2634884100
STAIIE	10.1073/pnas.2634884100
STDHIE	10.1073/pnas.2634884100
STGIIE	10.1073/pnas.2634884100
STIHIE	10.1073/pnas.2634884100
STMHIE	10.1073/pnas.2634884100
STPIIE	10.1073/pnas.2634884100
STTHIE	10.1073/pnas.2634884100
STVAIE	10.1073/pnas.2634884100
STVDIE	10.1073/pnas.2634884100
STVGIE	10.1073/pnas.2634884100
STVIAE	10.1073/pnas.2634884100
STVIDE	10.1073/pnas.2634884100
STVIEE	10.1073/pnas.2634884100
STVIGE	10.1073/pnas.2634884100

STVIIP	10.1073/pnas.2634884100
STVILE	10.1073/pnas.2634884100
STVIME	10.1073/pnas.2634884100
STVINE	10.1073/pnas.2634884100
STVIPE	10.1073/pnas.2634884100
STVIQE	10.1073/pnas.2634884100
STVISE	10.1073/pnas.2634884100
STVITE	10.1073/pnas.2634884100
STVIVE	10.1073/pnas.2634884100
STVIWE	10.1073/pnas.2634884100
STVMIE	10.1073/pnas.2634884100
STVPIE	10.1073/pnas.2634884100
STVQIE	10.1073/pnas.2634884100
STVSIE	10.1073/pnas.2634884100
STVVIE	10.1073/pnas.2634884100
STYIIE	10.1073/pnas.2634884100
SWVIIE	10.1073/pnas.2634884100
KAAGG	10.21203/rs.3.rs-109949/v10
PFTTR	10.21203/rs.3.rs-109949/v13
PLVKA	10.21203/rs.3.rs-109949/v13
QIMRW	10.21203/rs.3.rs-109949/v13
KIKIQI	10.1002/adfm.202009382
AENGKS	10.1073/pnas.0511298103
DIEVDL	10.1073/pnas.0511298103
DLLKNG	10.1073/pnas.0511298103
EALYLV	10.1073/pnas.0511298103
EHSDLS	10.1073/pnas.0511298103
EKDEYA	10.1073/pnas.0511298103
ERGFFY	10.1073/pnas.0511298103
ERIEKV	10.1073/pnas.0511298103

EVDLLK	10.1073/pnas.0511298103
FFYTPK	10.1073/pnas.0511298103
FHPSDI	10.1073/pnas.0511298103
FSKDWS	10.1073/pnas.0511298103
FTPTEK	10.1073/pnas.0511298103
FVNQHL	10.1073/pnas.0511298103
FYTPKT	10.1073/pnas.0511298103
GERGFF	10.1073/pnas.0511298103
GFFYTP	10.1073/pnas.0511298103
GSHLVE	10.1073/pnas.0511298103
HLVEAL	10.1073/pnas.0511298103
HPAENG	10.1073/pnas.0511298103
IEKVEH	10.1073/pnas.0511298103
IQRTPK	10.1073/pnas.0511298103
IQVYSR	10.1073/pnas.0511298103
KSNFLN	10.1073/pnas.0511298103
KVEHSD	10.1073/pnas.0511298103
KWDRDM	10.1073/pnas.0511298103
LKNGER	10.1073/pnas.0511298103
LSFSKD	10.1073/pnas.0511298103
LSQPKI	10.1073/pnas.0511298103
NGERIE	10.1073/pnas.0511298103
NGKSNF	10.1073/pnas.0511298103
PKIQVY	10.1073/pnas.0511298103
PSDIEV	10.1073/pnas.0511298103
PTEKDE	10.1073/pnas.0511298103
QPKIVK	10.1073/pnas.0511298103
RGFFYT	10.1073/pnas.0511298103
RTPKIQ	10.1073/pnas.0511298103
RVNHVT	10.1073/pnas.0511298103

SDLSFS	10.1073/pnas.0511298103
SGFHPS	10.1073/pnas.0511298103
SHLVEA	10.1073/pnas.0511298103
SLYQLE	10.1073/pnas.0511298103
SRHPAE	10.1073/pnas.0511298103
TEFTPT	10.1073/pnas.0511298103
VKWDRD	10.1073/pnas.0511298103
VTLSQP	10.1073/pnas.0511298103
VYSRHP	10.1073/pnas.0511298103
WSFYLL	10.1073/pnas.0511298103
YQLENY	10.1073/pnas.0511298103
YVSGFH	10.1073/pnas.0511298103
YYTEFT	10.1073/pnas.0511298103
RITTR	10.1038/s41467-021-24597-5
RITWI	10.1038/s41467-021-24597-5
VKVKVKVK	10.1021/ja027993g
VKVKVKVKVPPTKVEVKVKV	10.1021/ja054719o

B. Ispitivanje greške procjene preklapanja peptida po masi simuliranim kaljenjem

Tablica B.1: Usporedba metoda za procjenu preklapanja peptida po masi. Simulirano kaljenje uspoređeno je s metodom iscrpnog pretraživanja koristeći ispitne slučajeve različitih složenosti. Svaki redak tablice predstavlja jedan ispitni slučaj, a njegova složenost indicirana je duljinom peptida, brojem aminokiselina-kandidata po svakoj poziciji, te veličinom prostora pretraživanja. Greška metode simuliranog kaljenja u odnosu na iscrpno pretraživanje promatrana je kao rezultat. Uzmemo li u obzir ispitne slučajeve u kojima se simulirano kaljenje koristi u algoritmu predloženom u poglavlju 6. (veličina prostora pretraživanja veća od $9,8 \cdot 10^6$, podebljano u tablici), prosječna greška iznosi manje od 3%. Preuzeto iz [190], prilagođeno i prevedeno.

Ulaz u algoritam			Rezultat		
Duljina peptida	Broj aminokiselina po poziciji	Veličina prostora pretraživanja	Rješenje iscrpnog pretraživanja	Rješenje simuliranog kaljenja	Apsolutna greška simuliranog kaljenja
9	1	1	1	1	0,00%
1	2	2	2	1	50,00%
1	3	3	1	2	100,00%
1	4	4	3	2	33,33%
1	5	5	4	1	75,00%
1	6	6	4	5	25,00%
1	7	7	4	4	0,00%
3	2	8	6	1	83,33%
2	3	9	4	3	25,00%
1	10	10	5	1	80,00%
2	4	16	9	1	88,89%
2	5	25	11	1	90,91%
3	3	27	9	12	33,33%
5	2	32	17	2	88,24%
2	6	36	12	1	91,67%
2	7	49	15	1	93,33%
6	2	64	12	16	33,33%
4	3	81	17	18	5,88%
2	10	100	15	20	33,33%

3	5	125	21	26	23,81%
7	2	128	22	26	18,18%
3	6	216	21	25	19,05%
5	3	243	27	30	11,11%
8	2	256	21	25	19,05%
3	7	343	27	1	96,30%
9	2	512	21	23	9,52%
4	5	625	28	32	14,29%
6	3	729	28	31	10,71%
3	10	1000	21	24	14,29%
10	2	1024	33	35	6,06%
4	6	1296	22	25	13,64%
7	3	2187	30	32	6,67%
4	7	2401	27	30	11,11%
5	5	3125	36	41	13,89%
4	8	4096	38	1	97,37%
4	9	6561	43	45	4,65%
5	6	7776	36	39	8,33%
4	10	10000	39	41	5,13%
6	5	15625	43	45	4,65%
7	4	16384	49	52	6,12%
5	7	16807	44	48	9,09%
9	3	19683	62	64	3,23%
5	8	32768	44	46	4,55%
6	6	46656	52	53	1,92%
10	3	59049	46	50	8,70%
8	4	65536	52	53	1,92%
7	5	78125	47	49	4,26%
5	10	100000	53	57	7,55%
6	7	117649	56	58	3,57%
9	4	262144	58	61	5,17%
7	6	279936	57	58	1,75%
8	5	390625	55	56	1,82%
6	9	531441	54	56	3,70%
7	7	823543	55	57	3,64%
6	10	1000000	59	60	1,69%
10	4	1048576	65	66	1,54%
8	6	1679616	68	71	4,41%

9	5	1953125	59	60	1,69%
7	8	2097152	65	66	1,54%
7	9	4782969	66	68	3,03%
8	7	5764801	65	66	1,54%
10	5	9765625	75	76	1,33%
7	10	10000000	63	65	3,17%
9	6	10077696	80	82	2,50%
8	8	16777216	75	77	2,67%
9	7	40353607	105	108	2,86%
8	9	43046721	72	74	2,78%
10	6	60466176	83	85	2,41%
8	10	100000000	87	90	3,45%
9	8	134217728	126	129	2,38%

ŽIVOTOPIS DOKTORANDA

Marko Njirjak, mag. ing. comp. rođen je 27. prosinca 1996. Prirodoslovno-matematički smjer Gimnazije Andrije Mohorovičića Rijeka završio je 2015., te iste godine upisuje prijediplomski studij računarstva koji završava 2018., a diplomski studij 2020. godine, oba na Tehničkom fakultetu Sveučilišta u Rijeci. Tijekom studija bio je stipendist grada Rijeke i dobitnik tri Dekanove nagrade za akademsku izvrsnost, a također je odradio i kratku stručnu praksu u tvrtki Huawei (NR Kina) u sklopu programa „Seeds for the Future”. Po završetku diplomskog studija, upisao je Doktorski studij iz područja tehničkih znanosti, polja računarstva, također na Tehničkom fakultetu Sveučilišta u Rijeci, u sklopu kojeg je istraživao primjene evolucijskih algoritama i strojnog učenja za pronalazak novih terapijskih molekula. Stručno se usavršavao u inozemstvu na Odjelu za kemijske i farmaceutске znanosti Sveučilišta u Trstu, Odjelu za biokemiju, mikrobiologiju, biologiju stanica i genetiku Sveučilišta La Laguna, te u Laboratoriju za bioinformatiku i humanu elektrofiziologiju Jonskog Sveučilišta. Trenutačno je zaposlen kao asistent na Zavodu za računarstvo Tehničkog fakulteta Sveučilišta u Rijeci.

POPIS ZNANSTVENIH RADOVA

- [1] M. Njirjak, E. Otović, D. Kalafatovic, and G. Mauša. A multi-library approach to parallelised sequence space exploration: Streamlining library design and search space coverage. *Computers in Biology and Medicine*, 198:111207, 2025.
- [2] M. Njirjak, D. Kalafatovic, and G. Mauša. AI-driven design of β -secretase 1 inhibitors. In *37th European Peptide Symposium and 14th International Peptide Symposium*, 2024.
- [3] M. Njirjak, L. Žužić, M. Babić, P. Janković, E. Otović, D. Kalafatovic, and G. Mauša. Reshaping the discovery of self-assembling peptides with generative AI guided by hybrid deep learning. *Nature Machine Intelligence*, 6:1487–1500, 2024.
- [4] G. Mauša, M. Njirjak, E. Otović, and D. Kalafatovic. Configurable soft computing-based generative model: The search for catalytic peptides. *MRS Advances*, 8:1068–1074, 2023.
- [5] E. Otović, M. Njirjak, D. Kalafatovic, and G. Mauša. Sequential properties representation scheme for recurrent neural network-based prediction of therapeutic peptides. *Journal of Chemical Information and Modeling*, 62:2961–2972, 2022.
- [6] M. Njirjak, E. Otović, D. Jozinović, J. Lerga, G. Mauša, A. Michelini, and I. Štajduhar. The choice of time–frequency representations of non-stationary signals affects machine learning model accuracy: A case study on earthquake detection from LEN-DB data. *Mathematics*, 10:965, 2022.
- [7] E. Otović, M. Njirjak, D. Jozinović, G. Mauša, A. Michelini, and I. Štajduhar. Intra-domain and cross-domain transfer learning for time series data—How transferable are the features? *Knowledge-Based Systems*, 239:107976, 2022.

- [8] E. Otović, M. Njirjak, I. Žužić, D. Kalafatovic, and G. Mauša. Genetic algorithm parametrization for informed exploration of short peptides chemical space. In 2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM), 2020.
- [9] M. Njirjak, E. Otović, M. Budimir, H. Vlahović, M. Tomić, and V. Marijančić. Smartphone based range of motion measurement in physiotherapy. In 2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO), 2020.