Modeliranje dnevnih koncentracije čestica u zrak pomoću konvolucijskih neuronskih mreža

Gudelj, Ivan

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Diplomski sveučilišni studij

MODELIRANJE DNEVNIH KONCENTRACIJA ČESTICA U ZRAKU POMOĆU KONVOLUCIJSKIH NEURONSKIH MREŽA

Diplomski rad

Ivan Gudelj

Osijek, prosinac 2023.

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

Obrazac D1: Obrazac za imenovanje Povjerenstva za diplomski ispit

Osijek, 04.12.2023.

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za diplomski ispit

Ime i prezime Pristupnika:	Ivan Gudelj
Studij, smjer:	Diplomski sveučilišni studij Računarstvo
Mat. br. Pristupnika, godina upisa:	D-1203R, 07.10.2021.
OIB studenta:	84116721025
Mentor:	izv. prof. dr. sc. Emmanuel Karlo Nyarko
Sumentor:	doc. dr. sc. Mario Lovrić
Sumentor iz tvrtke:	
Predsjednik Povjerenstva:	izv. prof. dr. sc. Ratko Grbić
Član Povjerenstva 1:	izv. prof. dr. sc. Emmanuel-Karlo Nyarko
Član Povjerenstva 2:	doc. dr. sc. Ivan Vidović
Naslov diplomskog rada:	Modeliranje dnevnih koncentracije čestica u zrak pomoću konvolucijskih neuronskih mreža
Znanstvena grana diplomskog rada:	**Umjetna inteligencija (zn. polje računarstvo)**
Zadatak diplomskog rada:	Cilj ovog rada je poboljšati predikciju dnevnih koncetracija čestica u zraku na gradskim mjernim mjestima. Pretpostavka je da su 1D CNN moćnije od algoritama poput Random Forests. Podaci za predikciju su u obliku vremenske multivarijatne vremenske serije. (Sumentor iz tvrtke: Doc. dr. sc. Mario Lovrić, KNOW-CENTER GmbH, Inffeldgasse 13/6, A-8010 Graz, Austrija)
Prijedlog ocjene pismenog dijela ispita (diplomskog rada):	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 2 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene od strane mentora:	04.12.2023.

Potvrda mentora o predaji konačne verzije rada:	*Mentor elektronički potpisao predaju konačne verzije.*
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

IZJAVA O ORIGINALNOSTI RADA

Osijek, 05.02.2024.

Ime i prezime studenta:	Ivan Gudelj
Studij:	Diplomski sveučilišni studij Računarstvo
Mat. br. studenta, godina upisa:	D-1203R, 07.10.2021.
Turnitin podudaranje [%]:	15

Ovom izjavom izjavljujem da je rad pod nazivom**: Modeliranje dnevnih koncentracije čestica u zrak pomoću konvolucijskih neuronskih mreža**

izrađen pod vodstvom mentora izv. prof. dr. sc. Emmanuel Karlo Nyarko

i sumentora doc. dr. sc. Mario Lovrić

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD

Zagađenje zraka predstavlja važan ekološki problem s velikim utjecajem na ljudsko zdravlje i ekosustav planeta. Samo zagađenje zraka se odnosi na prisutnost štetnih tvari u zraku, koje mogu biti prirodnog ili antropogenog podrijetla, odnosno izvori onečišćenja zraka kreću se od prirodnih fenomena poput vulkanskih erupcija i požara do ljudskih aktivnosti poput transporta, industrijskih procesa i proizvodnje energije. Jedan od najčešćih onečišćivača zraka jesu lebdeće čestice koje mogu biti različitih veličina i sastava, od grubih do finih i od organskih do anorganskih. Od svih lebdećih čestica koje onečišćavaju zrak, najzabrinjavajuće za ljudsko zdravlje su $PM_{2.5}$ (čestice aerodinamičkog promjera manjeg od 2.5 μm - fine lebdeće čestice) i PM_{10} (grube lebdeće čestice) [1]. $PM_{2.5}$ čestice su veoma male i mogu prodrijeti duboko u pluća, uzrokujući respiratorne probleme i kardiovaskularne bolesti, dok su PM_{10} čestice nešto veće te uzrokuju iritaciju očiju, nosa i grla. Osim navedenih lebdećih čestica, drugi parametri onečišćenja zraka uključuju dušikove okside (NO, NO_2), sumporov dioksid (SO_2), ugljikov monoksid (CO) i ozon (O_3). Sve navedene čestice klasificirane su kao kancerogena grupa 1 od strane Svjetske zdravstvene organizacije (WHO) i mogu imati značajan utjecaj na ljudsko zdravlje [2], okoliš i klimatske promjene. Potrebno je provoditi učinkovito upravljanje kvalitetom zraka i mjere kontrole kako bi se smanjili negativni učinci zagađenja zraka, te time zaštitilo zdravlje svih ljudi i okoliša. Napredak u strojnom učenju, posebice u konvolucijskim neuronskim mrežama (nadalje u tekstu CNN), omogućio je modeliranje i predviđanje koncentracije čestica u zraku. Uporaba CNN u modeliranju koncentracije čestica u zraku predstavlja inovativan pristup koji pruža precizna predviđanja i omogućava donošenje boljih odluka o mjerama kontrole kvalitete zraka. Diplomski rad će uključivati razvoj modela temeljenog na CNN za predviđanje koncentracije čestica u zraku na temelju već zadanog skupa podataka koji sadržava povijesna očitanja određenih veličina. Tijekom rada će se procijeniti točnost i učinkovitost modela uz primjenu različitih algoritama. Rezultati samog istraživanja doprinijeti će razvoju točnijeg sustava praćenja kvalitete zraka te samim time pomoći ekološkim agencijama u razvoju učinkovitijih politika i propisa za rješavanje problema zagađenja zraka.

Rad se sastoji od osam poglavlja, od kojih je prvo poglavlje teorijski uvod u temu i zadatak rada. U drugom poglavlju jeanalizirana aktualna istraživanja na zadanu temu, dok je u trećem poglavlju opisana teorijska pozadina neuronskih mreža i principi CNN , tj. objašnjeno je kako CNN funkcioniraju te koji je njihov potencijalni doprinos u predviđanju koncentracije čestica u zraku. Četvrto poglavlje opisuje korištenu tehnologiju i metodologiju kao i proces

prikupljanja podataka i pripremu podataka za model (to uključuje informacije o izvorima podataka, čišćenje podataka te pripremu podataka za trening i testiranje modela). Peto poglavlje opisuje postupak izrade i optimizacije CNN modela za predviđanje koncentracije čestica u zraku, što uključuje informacije o odabiru odgovarajuće arhitekture, obuci modela te optimizaciji hiperparametara istog. To uključuje evaluaciju performansi modela, usporedbu s drugim modelima te raspravu o ograničenjima i mogućim poboljšanjima modela. Za kraj, u zaključku će biti sumiran rad te naglašeni ključni doprinosi ovog istraživanja.

1.1. Zadatak diplomskog rada

Cilj ovog rada je poboljšati predikciju dnevnih koncetracija čestica u zraku na gradskim mjernim mjestima. Pretpostavka je da su 1D CNN moćnije od algoritama poput Random Forests. Podaci za predikciju su u obliku vremenske multivarijatne vremenske serije (eng *time-series*).

2. AKTUALNI DOSEZI U MODELIRANJU DNEVNIH KONCENTRACIJA ČESTICA U ZRAKU

Istraživanje provedeno u Sjedinjenim Američkim Državama tijekom 2011. godine [3] prikazalo je korištenje konvolucijske neuronske mreže kako bi se procijenile razine koncentracije $PM_{2.5}$ čestica. Podaci o optičkoj dubini aerosola, meteorološkim poljima i korištenju zemljišta integrirani su u ovaj model. Testiranje i evaluacija modela izvršeni su korištenjem različitih tehnika unakrsne validacije kako bi se osigurala pouzdanost rezultata. Dodatno, u istraživanju je razvijena inovativna metrika važnosti prediktora temeljena na interpretacijskoj metodi neuronske mreže. Istraživanje je otkrilo da je predloženi model temeljen na CNN-u nadmašio sve modele usporedbe u procjeni koncentracija $PM_{2.5}$. Model je sposoban uhvatiti kompleksne nelinearne odnose između koncentracija $PM_{2.5}$ i povezanih prediktora poput meteoroloških i zemljišnih faktora. U pristupu skupa, trenira se više CNN modela s različitim inicijalizacijama težina, a predviđanja se usrednjavaju, što rezultira boljom točnošću procjene. Također, istraživanje je analiziralo važnost prediktora pomoću *Layerwise Relevance Propagation* (LRP) i otkrilo da se redoslijed važnosti svakog prediktora mijenja ovisno o položaju prediktora u CNN-u i vrsti samog prediktora. (npr. vrijednost prediktora na središnjoj lokaciji smatra se relativno važnijom za sve prediktore, dok važnost drugih lokacija ovisi o vrsti prediktora). Konačno, istraživanje je izvijestilo da je predloženi model postigao visoku učinkovitost u procjeni koncentracija $PM_{2.5}$ te može biti koristan za aplikacije podrške odlučivanju gdje su potrebne točne prognoze koncentracija PM2.5 na velikim područjima.

Drugo istraživanje [4], fokusirano na prostornu predikciju koncentracije PM_{10} čestica u Ankari, Turska, koristilo je različite algoritme strojnog učenja. Posebna pažnja posvećena je identifikaciji ekoloških problema u urbaniziranim i industrijaliziranim područjima s ciljem postavljanja prioriteta za održive i življive prostore. Istraživanje je koristilo podatke o parametrima s 7 postaja u Ankari u razdoblju od 01.01.2009. do 31.12.2017.. Regionalna analiza provedena je korištenjem tehnika strojnog učenja kako bi se s visokom točnošću predvidjele regionalne varijacije onečišćenja zraka. Tehnike strojnog učenja korištene u istraživanju obuhvaćale su LASSO (eng. *Least Absolute Shrinkage and Selection Operator*), SVM (eng. *Support Vector Machine*), RF (eng. *Random Forests*), kNN (eng *k-Nearest Neighbors*), xGBoost (eng. *eXtreme Gradient Boosting*) i ANN (eng. *Artificial Neural Network*). Predviđene vrijednosti evaluirane su temeljem prostorne distribucije pomoću GIS-a (eng. *Geographic Information System*). Istraživanje upućuje na to da se kvaliteta zraka u Ankari može poboljšati

povećanjem zelenih površina, potrošnjom prirodnog plina i podržavanjem politika javnog prijevoza. Ovo istraživanje ima značaj za utvrđivanje održivih strategija prostornog planiranja za kvalitetu zraka u Ankari, analiziranjem rezultata regionalne analize zajedno s promjenama u korištenju zemljišta.

Istraživanjem promjene koncentracije zagađivača zraka tijekom COVID-19 lockdowna u Grazu, Austrija, korištenjem strojnog učenja, posebice regresije pomoću slučajnih suma, istražena su različita predviđanja i stvarne razine zagađenja [5]. Modeli su pokazali dobru generalizaciju, s PM_{10} i NO_2 pokazujući predviđene vrijednosti iznad izmjerenih tijekom lockdowna, dok je O_3 bio podcijenjen, povezano s smanjenjem NO_x emisija zbog manjeg prometa. U članku se navodi da je RF korišten kao algoritam strojnog učenja zbog svoje dobre generalizacije i jednostavnosti u vezi s učenjem heterogenih podataka. Modeli su optimizirani putem Bayesove optimizacije s deseterostrukom unakrsnom validacijom, a RMSE (eng. *Root Mean Squared Error*) je korišten kao funkcija troška za optimizaciju modela. Kvaliteta modela automatski je procijenjena na temelju RMSE vrijednosti dobivene tijekom deseterostruke unakrsne validacije. Korak odabira značajki uveden je pomoću permutacijske važnosti. Iako je PM_{10} pokazao umjereno smanjenje, istraživanje je ukazalo na kompleksne faktore koji utječu na PM koncentracije, uključujući prašinastu oluju iz Sahare koja je otežala mjerenja. Rezultati potvrđuju da je strojno učenje prikladno za analizu brzih promjena u emisijama zraka, ističući potrebu za širim studijama radi poboljšanja generalizacije modela i preciznijih procjena smanjenja zagađenja zraka tijekom događaja.

Dokument [6] razmatra metodologiju dubokog učenja koristeći GRU (eng. *Gated Recurrent Unit*) neuronsku mrežu i empirijsko načelo dekompozicije za prognoziranje koncentracije $PM_{2.5}$ na mjestima praćenja površinskih podataka. Metoda uključuje provjeru stacionarnosti serija koncentracije onečišćenja zraka, dekompoziciju serija koncentracije $PM_{2.5}$ pomoću EMD (eng. *Empirical Mode Decomposition*) te konstrukciju GRU neuronske mreže za višekoračnu prognozu. Metoda empirijske dekompozicije moda (EMD) doprinosi prognozi koncentracije $PM_{2.5}$ dekomponiranjem niza koncentracije $PM_{2.5}$ u više stacionarnih podnizova. Ova dekompozicija omogućuje bolje razumijevanje osnovnih uzoraka i karakteristika podataka o koncentraciji $PM_{2.5}$. Ubacivanjem tih stacionarnih podnizova zajedno s meteorološkim značajkama u prediktivni model, poput neuronske mreže sa stanicama za ponovno pozivanje (GRU), EMD-GRU model može učinkovitije uhvatiti nestacionarnost i vremensku ovisnost niza koncentracije $PM_{2.5}$, što dovodi do poboljšane točnosti u predviđanju koncentracije $PM_{2.5}$. U kontekstu višekoračne predikcije, GRU neuronska mreža uzima ulazne podatke trenutka koji se

trenutno obrađuje zajedno s vrijednošću skrivenog stanja iz prethodnog trenutka i koristi ih za izračunavanje izlazne vrijednosti za trenutačni trenutak. Ovaj postupak se ponavlja za svaki vremenski korak u nizu, omogućujući mreži rekurzivno predviđanje više budućih vrijednosti.

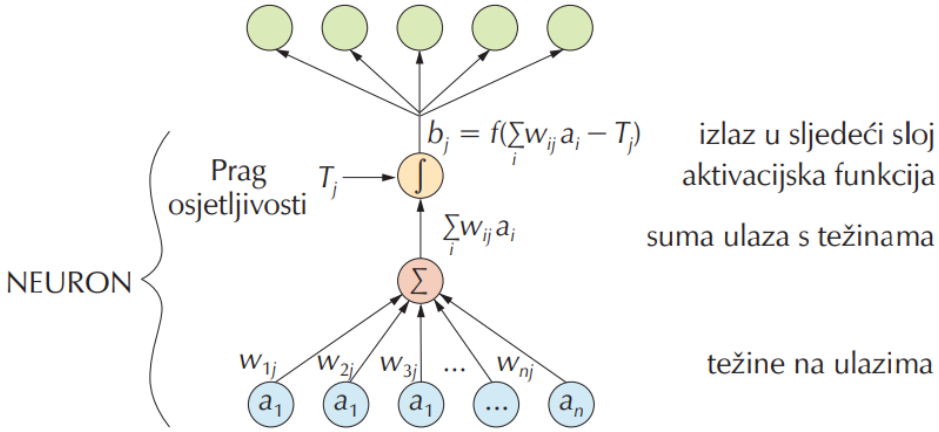
3. KONVOLUCIJSKE NEURONSKE MREŽE (CNN)

3.1. Uvod u neuronske mreže

Prvi poznati rad o umjetnim neuronskim mrežama objavljen je 1943. godine od strane autora McCulloh i Pitts [7], dok je prvi poznati prijedlog za mrežu koja se može trenirati dao Rosenblatt 1957. godine [8]. Neuronske mreže predstavljaju računalne modele koji su inspirirani strukturom i funkcijama biološkog mozga, odnosno neuronske mreže predstavljaju podskup algoritama strojnog učenja koji su dizajnirani kako bi uočili uzorke u podacima, učili iz istih, te predvidjeli buduća stanja bez da su eksplicitno programirani za specifične zadatke. Neuronske mreže su se pokazale kao veoma uspješan pristup u različitim područjima kao što su analiza i prepoznavanje slika, obrada jezika i prepoznavanje govora, no u današnje vrijeme moguće je vidjeti doprinos neuronskih mreža i u gotovo svim područjima koji uključuju automatizaciju.

Umjetni neuroni, također poznati kao čvorovi ili jedinice, čine jezgru neuronske mreže. Navedeni neuroni su organizirani u slojeve, tvoreći time arhitekturu mreže, pa se tako neuronska mreža sastoji od ulaznog sloja, skrivenog sloja i izlaznog sloja. Ulazni sloj predstavlja početnu točku mreže te prima neobrađene ulazne podatke. Svaki neuron u ulaznom sloju može predstavljati neku značajku (eng. *feature*) ulaznih podataka (npr., u zadatku prepoznavanja slike, svaki neuron može predstavljati vrijednost intenziteta piksela). Skriveni slojevi predstavljaju međuslojeve između ulaznog i izlaznog signala te igraju ključnu ulogu u izdvajanju i učenju složenih prikaza i značajki iz ulaznih podataka. Mreže koje se sastoje od više skrivenih slojeva se često mogu nazvati i dubokim neuronskim mrežama [9]. Izlazni sloj, ovisno o samom zadatku, daje kao rezultat konačna predviđanja odnosno odluke o klasifikaciji na temelju obrađenih podataka iz prethodnih slojeva. Broj neurona u izlaznom sloju ovisi o samom zadatku (npr. u problemu binarne klasifikacije, izlazni sloj može imati jedan neuron koji predstavlja vjerojatnost pripadnosti jednoj klasi, dok drugi neuron predstavlja vjerojatnost pripadnosti drugoj klasi).

Prema navedenome, neuronsku mrežu gledamo na sljedeći način; ulazni sloj poprima vrijednosti ulaznih veličina, zatim se svaka vrijednost ulaza množi određenom težinskom funkcijom w_i. Tako otežani, ulazni signali se zbrajaju, a njihov zbroj se uspoređuje s pragom osjetljivosti neurona, T_j (eng. *threshold*). Zatim skriveni sloj zbraja tako otežane ulaze pomoću zadane funkcije sumiranja Prijenosna funkcija može biti diskontinuirana skokovita funkcija ili neka kontinuirana funkcija kao što je sigmoida ili tangens-hiperbolna funkcija.



Sl. 3.1. Osnovni model perceptrona, [10]

Slika 3.1. prikazuje model perceptrona[1] s ulaznim podacima $a_1,a_2,a_3,...a_n \in \mathbb{R}$ i pripadajućim težinama $w_{1j},w_{2j},w_{3j},...w_{nj} \in \mathbb{R}$. Ukoliko neuron zadovoljava prag osjetljivosti (T_j), stvara se izlazna vrijednost b_j, gdje je $f : \mathbb{R} \rightarrow \mathbb{R}$ aktivacijska funkcija (3-1).

$$b_j = f(\sum_{i=1}^{n} w_{ij} a_i - T_j) \tag{3-1}$$

Prema [11] aktivacijska funkcija se koristi u umjetnim neuronskim mrežama za transformaciju ulaznog signala u izlazni signal, koji se dovodi kao ulaz u sljedeći sloj unutar neuronske mreže. Funkcionira na način da se izračunava zbroj umnožaka težina s pripadajućim ulaznim podacima, a konačno se nad tim zbrojem primjenjuje aktivacijska funkcija kako bi se dobio izlaz spreman za idući sloj neurona. U tablici 3.1. se nalaze najčešće korištene aktivacijske funkcije zajedno sa pripadajućim formulama, prednostima i nedostacima.

[1] Perceptron - osnovna gradivna jedinica u neuronskim mrežama

Tablica 3.1. Najčešće korištene aktivacijske funkcije.

Naziv	Formula	Prednosti	Nedostaci
Sigmoidna funkcija	$b(a) = \frac{1}{1 + e^{-a}}$	Glatka i daje vrijednost između 0 i 1, što je korisno za modeliranje vjerojatnosti. Koristi se u modelima gdje je potrebna binarna klasifikacija	Može dovesti do problema poznatih kao "sigmoidni nestajući gradijent" [12] kada se koristi u dubokim neuronskim mrežama
ReLU (Rectified Linear Unit) funkcija	$b(a) = \max(0, a)$	Jednostavna funkcija koja brzo konvergira tijekom učenja. Dobra sposobnost modeliranja nelinearnih veza i sprječava problem gradijentnog nestanka	Može uzrokovati "mrtve neurone" [13] jer sve negativne vrijednosti postaju nula (ne doprinose daljnjem učenju)
Tanh (Hiperbolni tangens) funkcija	$b(a) = \frac{e^{a} - e^{-a}}{e^{a} + e^{-a}}$	Ima vrijednost između -1 i 1, čime se postiže bolja normalizacija izlaza. Koristi se u modeliranju simetričnih podataka	Može dovesti do problema nedostajućeg gradijenta kao i sigmoidna funkcija
Softmax funkcija	$\mathrm{b}(\mathrm{a_i}) = \frac{e^{a_i}}{\sum_{j=1}^{K} e^{a_j}}$ Za j = 1,…K, K ∈ $\mathbb{R}$	Stvara vjerojatnosnu distribuciju preko više klasa (višeklasna klasifikacija), omogućava učinkovitu optimizaciju	Može biti osjetljiva na ekstremne ulazne vrijednosti, što utječe na relativne vjerojatnosti dodijeljene različitim klasama. Također ima tendenciju stvaranja nenultih vjerojatnosti za većinu klasa

Učenje samog modela (eng. *learning, training*) predstavlja iterativni postupak u kojem se vrijednosti težinskih faktora koji se primjenjuju na ulazne podatke optimiraju na osnovu pogreške između proračunate izlazne vrijednosti od strane modela (označeno sa $B' \in \mathbb{R}$) i poznate izlazne vrijednosti za isti ulaz (označeno s $B \in \mathbb{R}$). Mreža će tada, za dani ulaz a imati pripadajući točan izlaz $B(a)$ te dati izlaznu vrijednost:

$$B'(a, w)$$

Tada se skup za učenje definira kao skup uređenih parova $\{(a_i, B_i), i = (1,...,N)\}$. Sada je moguće uvesti funkciju L (3-2) koja predstavlja odstupanje izlaza od strane mreže $B'(a, w)$ za danu točnu vrijednost B:

$$L(B(a), B'(a, w)) \tag{3-2}$$

Dakle bitno je naglasiti da predikcija $B'(a, w)$ ovisi o ulaznim podacima a i pripadajućim težinama w. Funkcija gubitka (3-3) (eng. *cost function*) predstavlja skalarnu vrijednost koja je prosjek odstupanja izlaza modela od stvarnog izlaza nad cijelim skupom za učenje i definira se izrazom:

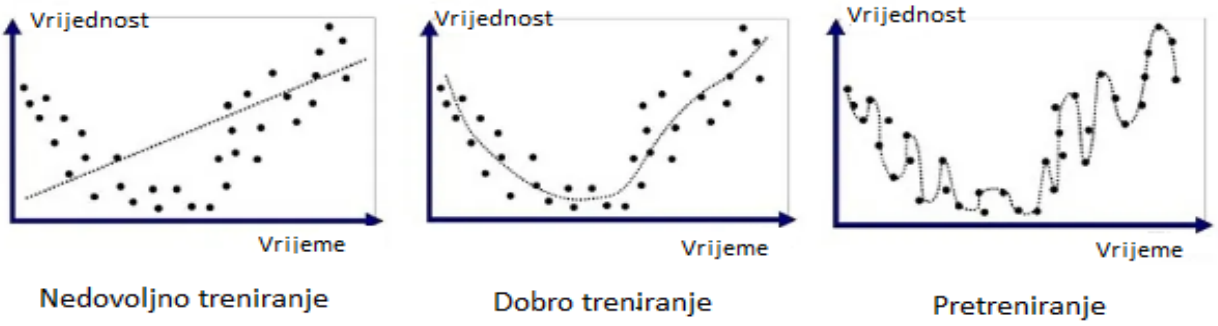
$$J(w) = \frac{1}{N}\sum_{i=1}^{N} L(B_i(a), B'(a, w)) \tag{3-3}$$

Pregled često korištenih funkcija gubitka pri rješavanju zadataka s regresijskim i klasifikacijskim problemima se može pronaći u Tablici 3.2. sam izbor funkcije gubitka ovisi o problemu koji je potrebno riješiti te modelu koji se odluči koristiti. Podešavanje samih parametara modela se vrši prema jednim od definiranih pravila učenja kao što je pravilo širenja unatrag odnosno algoritam unatražne propagacije (eng. *back propagation*) [14]. Daljnjim razvojem neuronskih mreža, omogućeni su brojni algoritmi i strukture učenja. Optimiranjem težinskih faktora, neuronska mreža uči predviđati stvarne vrijednosti, odnosno smanjuje se razlika između predviđenih i stvarnih vrijednosti izlaznih veličina. Sam kriterij pogreške govori o robusnosti i kvaliteti mreže. Provjera mreže se vrši uz pomoć novog skupa podataka – skup za provjeru.

Tablica 3.2. Najčešće korištene funkcije gubitka

Funkcija gubitka – klasifikacija	**Formula**
Cross Entropy (binarna klasifikacija)	$J(w) = -\sum_{i=1}^{N} B_i(\log(B_i') + (1 + B_i)\log(1 - B_i')$
Cross Entropy (klasifikacija – *k* klasa)	$\mathrm{L}(\mathrm{B_i}, \mathrm{B_i'}) = -\sum_{i=1}^{N} B_i \log(B_i')$ $J(w) = \frac{1}{N}\sum_{i=1}^{N} \mathrm{L}(\mathrm{B_i}, \mathrm{B_i'})$
Funkcija gubitka – regresija	**Formula**
Mean Squared Error (MSE)	$J(w) = \frac{1}{N}\sum_{i=1}^{N}(B_i - B_i')^2$
Root MSE	$J(w) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(B_i - B_i')^2}$
Mean Absolute Error (MAE)	$J(w) = \frac{1}{N}\sum_{i=1}^{N}\lvert B_i - B_i'\rvert$

Postoje dvije bitne pojave kod učenja neuronskih mreža a to su “pretreniranje” (eng. *overfitting*) i “nedovoljno treniranje” (eng. *underfitting*). Obje pojave predstavljaju jednu vrstu pogreške modela. Pretreniranje se javlja kada mreža pokaže dobre rezultate pri opisivanju vladanja podacima nad skupom podataka na kojem je i razvijena, a lošije rezultate izvan tog skupa. Kako bi se to spriječilo, potrebno je zaustaviti učenje neuronskih mreža u trenutku kada pogreška provjere počne rasti [15]. Nedovoljno treniranje znači da mreža nije dovoljno kompleksna za problem koji se rješava, odnosno ima premalo parametara ili nije dovoljno dobro trenirana, što rezultira lošim performansama i na skupu za učenje i na novim podacima. Slika 3.2. prikazuje primjer podataka s granicama odluke u pojedinim situacijama.



Sl. 3.2. Primjer nedovoljnog treniranja mreže, dobro trenirane mreže i pretreniranja mreže, [16]

Danas postoji veliki broj vrsta neuronskih mreža te ih je teže sustavno klasificirati. Najvažnija podjela je ona na unaprijedne (eng *feed-forward*) i povratne (eng. *feedback* ili *recurrent*) mreže. Signalima se dopušta putovati od ulaza prema izlazu neurona u unaprijednim mrežama, dok povratne veze dopuštaju signalima putovati u oba smjera. Također, bitno je napomenuti da je neuronske mreže moguće podijeliti s obzirom na metodu učenja koju koriste;

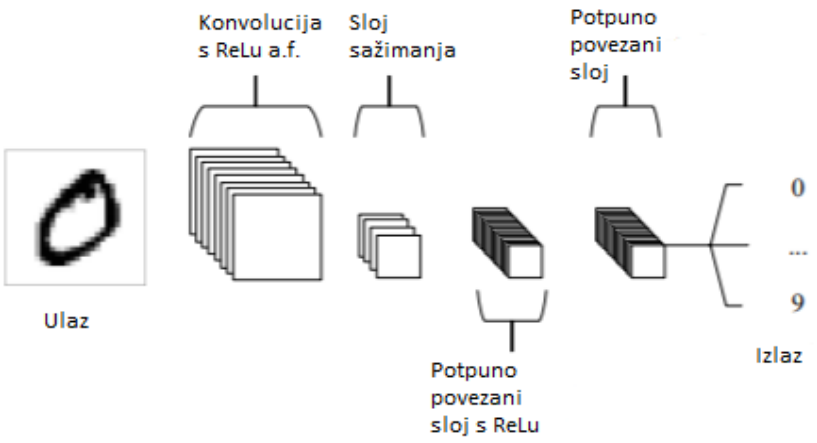
- Nadzirano učenje (eng. *supervised learning*) – temelji se na obuci uzorka podataka iz skupa podataka s već dodijeljenom ispravnom klasifikacijom. Mreža uči na temelju poznatih uzoraka tako da se na ulaz dovedu ulazni podaci, pa se izlaz mreže usporedi s poznatim rezultatom. Pogreška se koristi za promjenu parametara mreže (težine veza među neuronima). Postupak se višekratno ponavlja za sve ulazne podatke (korišteno u slučaju seminara).
- Nenadzirano učenje (eng. *unsupervised learning*) – mreža uči iz ulaznih podataka, tako da tokom učenja prepoznaje svojstva ili pravilnosti u ulaznim podacima. Koristi se za grupiranje podataka, izdvajanje značajki i prepoznavanje sličnosti. Mreža uči reagirati na različite ulazne uzorke različitim dijelovima mreže, tako da stvara internu reprezentaciju ulaznih podataka.
- Podržano učenje (eng. *reinforced learning*) – temelji se na učenju putem pokušaja i pogreške u interakciji s okolinom (dodjela nagrade/kazne). Cilj jest pronaći odgovarajući model djelovanja koji bi maksimizirao ukupnu kumulativnu nagradu agenta.

3.2. Osnove konvolucijskih neuronskih mreža

Prvi korak prema razvoju CNN smatra se objavom “The Ferrier Lecture” istraživačkog rada″ 1981. godine, o vizualnim korteksima majmuna i ptica autora Hubel i Wiesel u kojem

raspravljaju o funkcionalnoj arhitekturi vizualnog korteksa makakija[2] i njezinim implikacijama za razumijevanje vizualne percepcije. Zatim je, kasnijih 1980-ih godina, u području CNN, predstavljen postupak konvolucije od strane Kunihiko Fukushime pod nazivom neokognitron (eng. *neocognitron*) [17], koji je bio inspiriran radom Hubela i Wiesela. Međutim Yann Le Cunna je imao ključnu ulogu u dovođenju CNN na razinu na kojoj se nalazi danas [18], razvijanjem 7-razinsku CNN nazvanu LeNet-5.

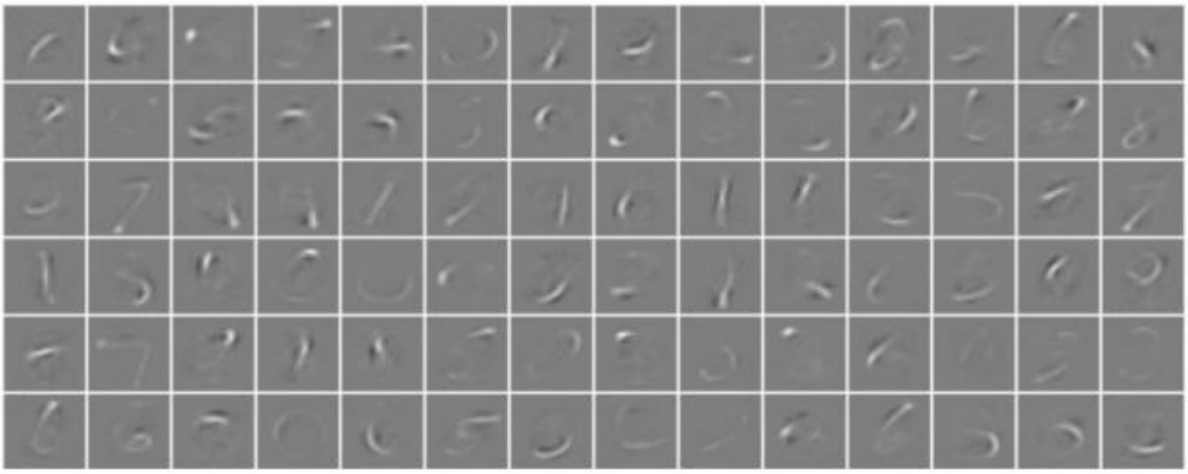
CNN se primarno razvio za obradu podataka sa slika, odnosno arhitektura CNN se postavlja tako da se očekuje slika kao podatak na ulazu. Jedna od ključnih razlika je u tome što su neuroni unutar slojeva CNN organizirani u 3 dimenzije; visina, širina i dubina (dubinom se smatra broj kanala na slici, npr. crno-bijela slika ima jedan kanal, slika s RGB formatom 3 kanala itd.). Za razliku od tradicionalnih umjetnih neuronskih mreža, CNN koristi tzv. konvolucijske slojeve (eng. *convolutional layers*), slojeve sažimanja (eng. *pooling layers*) i potpuno povezane slojeve (eng. *Fully-connected layers*) [19]. Uobičajena unutrašnja struktura klasične CNN se sastoji od nekoliko naizmjeničnih konvolucijskih slojeva i slojeva sažimanja. Na samom kraju se nalazi nekoliko potpuno povezanih slojeva koji su jednodimenzionalni, uključujući i izlazni sloj.



Sl. 3.3. Jednostavna CNN arhitektura od 5 slojeva, [20]

[2] eng. *Macaque* – vrsta majmuna

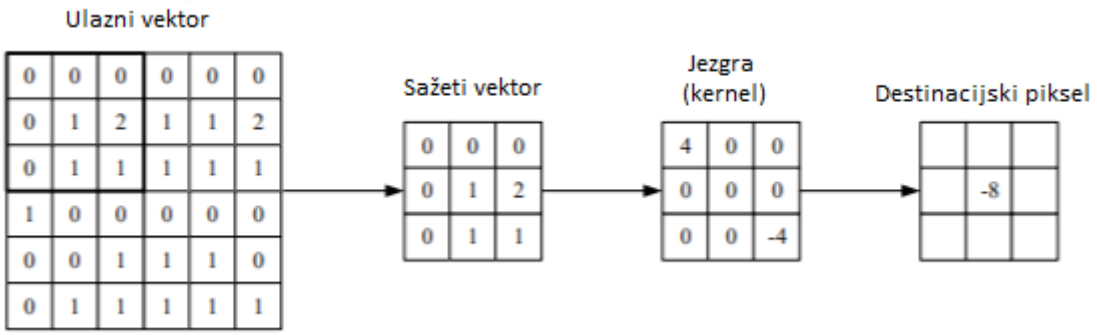
Primjer jednostavne konvolucijske neuronske mreže koja se koristi za MNIST klasifikaciju [21], moguće je vidjeti na slici 3.3. Konvolucijski sloj igra ključnu ulogu kod CNN. Parametri slojeva fokusirani su na korištenje jezgri/filtera (eng. *kernel*[3]*)* koje je moguće naučiti. Ove jezgre su uobičajeno male, ali se šire preko svih ulaza tj. ulaznih podataka. Kada podaci dođu do konvolucijskog sloja, sloj konvoluira svaki filter preko prostorne dimenzionalnosti ulaza kako bi izradio 2D aktivacijsku mapu. Ove mape se mogu vizualizirati kao što je prikazano na slici 3.4.



Sl. 3.4. Aktivacije preuzete iz prvog konvolucijskog sloja jednostavne duboke CNN mreže nakon obuke na MNIST bazi podataka, [22]

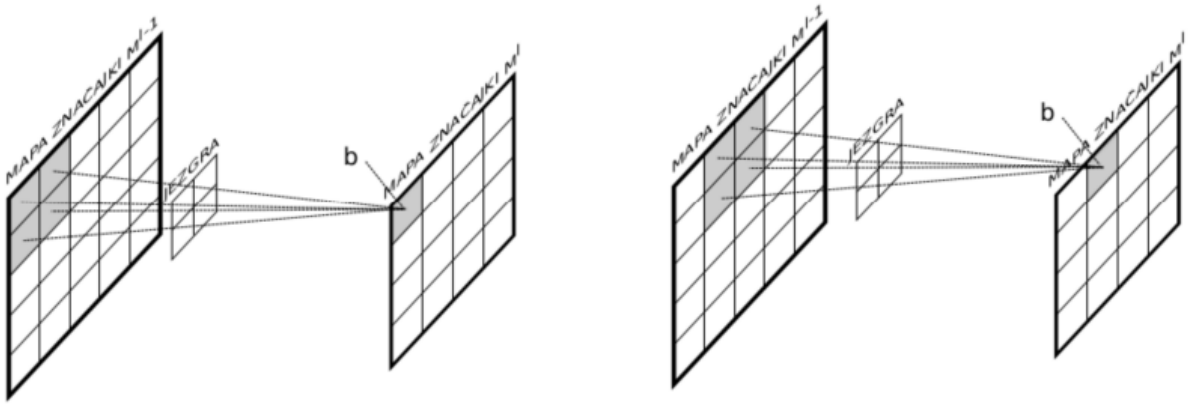
. Dok se prolazi kroz ulazne podatake, skalarni produkt se izračunava za svaku vrijednost u tom filteru (Slika 3.5.). Iz toga, mreža će naučiti filtere koji se aktiviraju kada vide određenu značajku na određenom prostornom položaju ulaza. Središnji element filtera je postavljen preko ulaznog vektora, koji se potom računa kao otežana suma sebe i susjednih pixela. Svaki filtar sadržava pripadajuću mapu značajki, koje će biti posložene tako da formiraju potpuni izlazni volumen konvolucijskog sloja.

[3] Kernel – Matrica određenih dimenzija koja predstavlja filter koji se provlači preko ulaznih podataka



Sl. 3.5. Prikaz konvolucijskog sloja, [23]

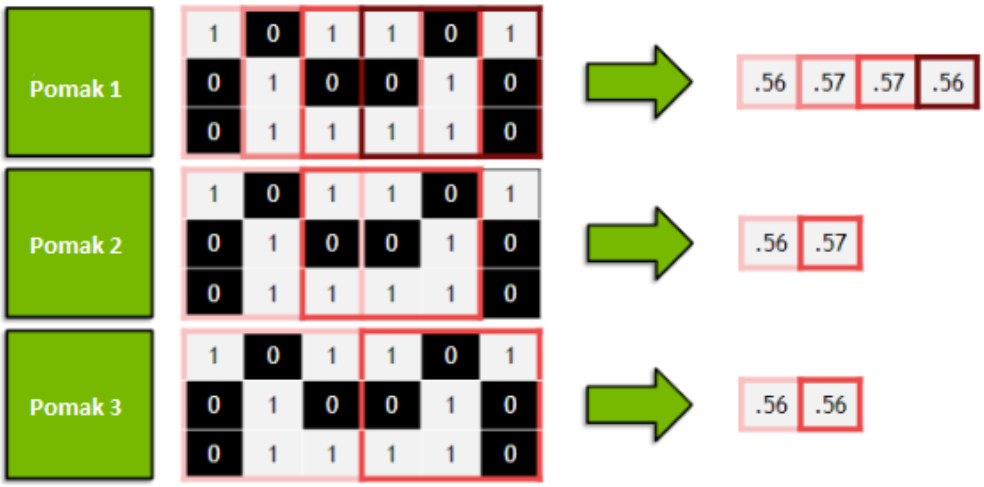
Učenje uobičajenih neuronskih mreža na ulazima kao što su slike, može rezultirati modelima koji su preveliki kako bi se učinkovito trenirali. To dolazi od potpuno povezanog načina rada standardnih neuronskih mreža te kako bi se to spriječilo, svaki neuron iz konvolucijskog sloja povezan je samo s malim dijelom ulaza u ulaznom volumenu podataka. Dimenzionalnost tog dijela obično se naziva veličina receptivnog polja neurona ili veličina filtera (Slika 3.6 – veličina filtera je 2x2).



Sl. 3.6. Primjer konvolucije unutar konvolucijskog sloja, [24]

Iz prethodne slike (Slika 3.6.) moguće je vidjeti da elementi (pikseli) izlazne mape značajki M^l ovise o lokalnom susjedstvu elemenata ulazne mape značajki M^{l-1}. Čime se drastično smanjuje broj parametara. Svaki pojedini element mape značajki se računa uz pomoć istog skupa parametara. Tri hiperparametra koja se koriste za definiranje dimenzije podataka pri izlazu iz konvolucijskog sloja jesu: dubina (eng. *depth*), pomak (eng. *stride*) i dodavanje nula (eng. *padding/zero-padding*). Dubina odgovara broju filtera koje želimo koristiti, pri čemu svaki filter uči prepoznati nešto drukčije u ulazu (Npr. ako prvi konvolucijski sloj koristi sirovu sliku kao ulaz, tada će se različiti neuroni duž dimenzije dubine aktivirati u prisutnosti različitih

usmjerenih rubova ili skupova boja). Drugi hiperparametar je pomak s kojim pomjeramo filter. Ukoliko je pomak postavljen na vrijednost1, tada se filter pomiče jedan po jedan piksel odnosno stupac u vektoru ulaznih podataka, ako je pomak postavljen na vrijednost 2, filter će se pomicati za 2 piksela nakon obrade podataka, kao na slici 3.7. (crveni okviri različitih nijansi predstavljaju filtere). Rijetko se koriste filteri s većim pomakom od 3.



Sl. 3.7. Primjer pomaka s vrijednostima 1, 2 i 3

Posljednji navedeni hiperparametar jest dodavanje nula (eng. *zero-padding*) (Slika 3.8.). Ukoliko želimo dovoljno podataka na ulazu, tj. kako bi osigurali da su svi pikseli iskorišteni u konvoluciji ili ukoliko želimo da na izlazu iz sloja slika bude iste dimenzije kao i na ulazu u sloj, možemo dodati još jedan rub elemenata vrijednosti nula oko naše slike. To je moguće izračunati po formuli (3.4), gdje V predstavlja dimenzije ulaznih podataka (visina x širina x dubina), R predstavlja veličinu filtera, Z postavljenu količinu dodanih nula, *S* predstavlja pomak. Također postoji i nešto što se zove zrcalno dodavanje (eng. *mirror padding*) gdje se kopira vrijednosti rubova i stvara novi rub s istim vrijednostima (Slika 3.9.)

Sl. 3.8. Primjer *zero-padding-a*

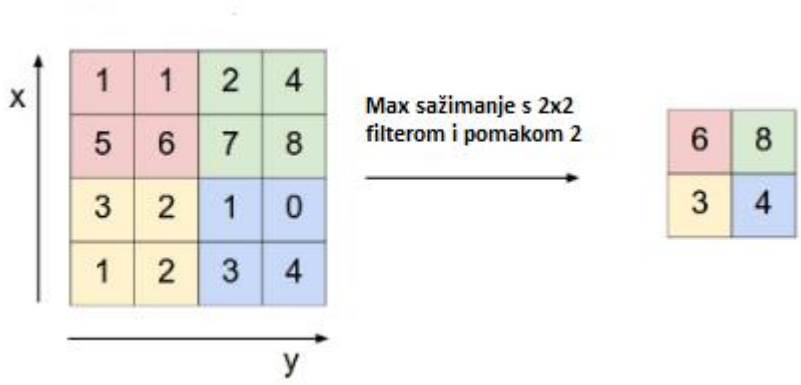
Sl. 3.9. Primjer *mirror-padding-a*

$$Dim = \frac{(V - R) + 2Z}{S + 1} \tag{2-4}$$

Ukoliko rezultat (2-4) nije cijeli broj, tada pomak nije dobro postavljen te dolazi do toga da se neki podaci s ulaza neće uzeti u obzir pri konvoluciji. Također postoji i nešto što se naziva "dijeljenje parametara". Funkcionira na sljedećoj pretpostavci; ako je jedna značajka područja korisna za računanje na određenom prostornom području obrađivanih podataka, vrlo vjerojatno će biti korisna i na drugom području. Ako ograničimo svaku pojedinu mapu značajki unutar izlaznog volumena na iste težine i *bias*[4], tada će biti moguće vidjeti ogromno smanjenje broja parametara koji se generiraju konvolucijskim slojem.

Gotovo uvijek između konvolucijskih slojeva postavljamo sloj sažimanja tijekom izrade modela. Funkcija sloja sažimanja jest postupno smanjivanje prostorne veličine reprezentacije podataka kako bi se smanjio broj parametara i izračuna u mreži te istovremeno kontrolirao sam postupak učenja (kako ne bi došlo do nedovoljnog treniranja ili pretreniranja). Sloj sažimanja nevoisno djeluje na svaki sloj dubine ulaznih podataka i smanjuje njegovu prostornu veličinu koirsteći operaciju „maksimalnog sažimanja" (eng. *max-pooling*). Najčešće korišten je sloj sažimanja s filterima veličine 2 x 2 s pomakom 2, koji uzorkuje svaki sloj smanjujući mu širinu i visinu za faktor 2 i odbacujući time 75% aktivacija. U ovom slučaju, operacija maksimalnog sažimanja uzima maksimum od 4 broja (2 x 2 područje), primjer ove operacije vidljiv je na slici 3.10. Osim maksimalnog sažimanja, koriste se također i sažimanje po prosjeku (eng. *Average pooling*) te L2-normirano sažimanje (eng. *L2-norm pooling*). [25]

[4] Bias – konstantna vrijednost koja se dodaje sumi produkta otežanih ulaznih podataka prije nego podaci prođu kroz aktivacijsku funkciju

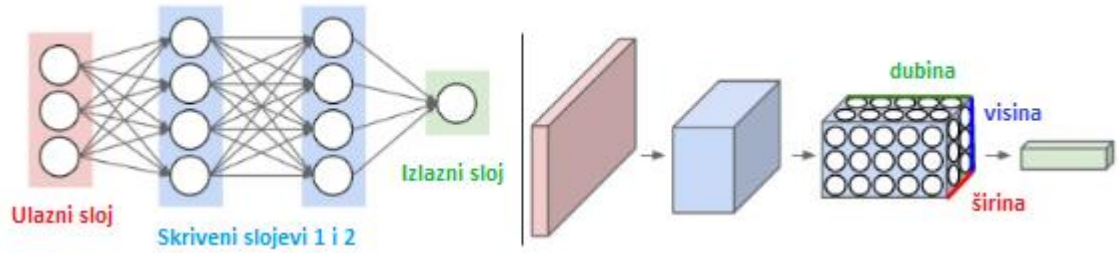


Sl. 3.10. Primjer *max pooling* operacije nad 4 x 4 podacima s primjenom 2 x 2 filtera, [26]

Osim konvolucijskog sloja i sloja sažimanja, CNN se odlikuju i korištenjem potpuno povezanog sloja podataka kao i *Dropout* sloja. Potpuno povezani sloj podataka je također sastavni dio arhitekture tradicionalnih neuronskih mreža. Svaki neuron je povezan sa svakim neuronom iz prethodnog sloja. Odnosno jedina razlika između konvolucijskih slojeva i potpuno povezanih slojeva jest ta da su neuroni konvolucijskog sloja spojeni samo na lokalno područje neurona iz prethodnog sloja te mnogi neuroni u konvolucijskoj mreži dijele parametre. *Dropout* sloj predstavlja metodu regularizacije koja pomaže pri sprječavanju pretreniranosti mreže. Pojam „*dropout*“ odnosi se na izostavljanje jedinica (skrivenih i vidljivih) u neuronskoj mreži. Kada se izostavi jedinica, privremeno se uklanja iz mreže zajedno sa svim ulaznim i izlaznim vezama. Odabir jedinica koje će se izostaviti je nasumičan. U najjednostavnijem slučaju, svaka jedinica se zadržava s fiksnom vjerojatnošću p neovisnom o drugim jedinicama, pri čemu se vrijednost p može odabrati pomoću validacijskog skupa ili jednostavno postaviti na 0,5, što se čini optimalnim za širok raspon mreža i zadataka [27]. Izostavljanje jedinica na neuronskoj mreži slično je uzorkovanju „prorijeđene" mreže iz nje. Prorijeđena mreža sastoji se od svih jedinica koje su preživjele izbacivanje . Neuronska mreža s n jedinica može se promatrati kao skup od 2^n mogućih prorijeđenih neuronskih mreža. Ove mreže dijele težine tako da ukupan broj parametara i dalje ostaje $O(n^2)$ ili manji. Za svaku prezentaciju svakog trening primjera, izvršava se uzorkovanje nove mreže koja se trenira. Stoga se učenje neuronske mreže s izostavljenim jedinicama može promatrati kao učenje skupa od 2^n prorijeđenih mreža s obilnim dijeljenjem težina, pri čemu se svaka prorijeđenih mreža rijetko ili uopće ne trenira.

U konačnici, uobičajene neuronske mreže kao ulaz zaprimaju vektor ulaznih podataka i transformiraju ga kroz niz skrivenih slojeva koji su sastavni dio arhitekture te neuronske mreže. Svaki skriveni sloj te mreže je potpuno povezan s prethodnim slojem te ti neuroni operiraju

potpuno neovisno o drugim neuronima, odnosno ne dijele nikakvu vezu (parametre). Posljednji sloj naziva se izlazni sloj te predstavlja vjerojatnost pripadnosti određenoj klasi. Poznati skup podataka CIFAR-10[5] sastoji se od slika dimenzija 32 x 32 x 3, pa se u slučaju korištenja obične neuronske mreže to svodi na 32*32*3=3072 težine za jedan neuron unutar skrivenog sloja, iako to zvuči kao "razumna" količina težina, za sliku 200 x 200 x 3, ta struktura nije optimalna budući da onda imamo neurone sa 200*200*3=120000 težina što vodi velikom broju podesivih parametara te bi to vrlo brzo dovelo do pretreniranosti. CNN iskorištavaju činjenicu da su ulazni podaci najčešće slike odnosno ulazni vektori s više dimenzija, te zbog toga drukčije organizira arhitekturu same mreže (Slika 3.11.), odnosno neuroni u CNN su postavljeni u 3 dimenzije (visina, širina, dubina). Također, budući da su neuroni skrivenih slojeva povezani s nekolicinom neurona iz prethodnog sloja (nisu potpuno povezani sa svima kao kod uobičajenih neuronskih mreža), smanjuje se broj mogućih podesivih parametara. Za slučaj korištenja CIFAR-10 skupa podataka, izlazni sloj će imati 1x1x10 dimenzije što predstavlja vektor vjerojatnosti pripadnosti jednoj od deset klasa.



Sl. 3.11. Lijevo: Uobičajena neuronska mreža sa potpuno povezanim slojevima, Desno: CNN s primjerom arhitekture neurona u 3 dimenzije, [28]

Različite arhitekture su rezultat napretka u istraživanju i primjeni konvolucijskih neuronskih mreža te predstavljaju temelj za mnoge uspješne primjene u području računalnog vida i strojnog učenja. Među najkorištenijim arhitekturama konvolucijskih neuronskih mreža su:

[5] CIFAR-10 – skup podataka koji se sastoji od 60000 slika te se koristi za testiranje i evaluaciju raznih modela strojnog učenja, prvenstveno modela za klasifikaciju i prepoznavanje objekata

- LeNet-5
- AlexNet
- ResNet50
- DenseNet
- ZFNet
- WaveNet
- ResNeXt
- VGG
- GoogLeNet
- Inception v1/v2/v3/v4
- SENet
- MobileNet

3.3. 1D CNN

Jedna od korištenih tehnika za analizu podataka predstavljenih u obliku vremenskog niza jest primjena 1D CNN čime se hvataju temporalne ovisnosti i otkrivaju se skriveni uzorci unutar podataka zadane vremenske serije. U samom središtu 1D CNN nalazi se koncept 1D konvolucije, koji uključuje klizanje jezgre duž vremenske osi vremenskog niza podataka. Ova operacija omogućuje mreži da prepozna lokalne uzorke unutar podataka. Množenjem jezgre sa sekvencijalnim podskupovima vremenskog niza podataka i zbrajanjem rezultata, mreža je u mogućnosti izvući bitne značajke. Primjenom aktivacijske funkcije uvodi se nelinearnost, omogućavajući modeliranje složenih odnosa između elemenata zadanog skupa podataka čime otkrivamo uzorke koji doprinose preciznim predviđanjima.

Arhitektura 1D CNN uobičajeno uključuje više slojeva kao i ostale CNN mreže (konvolucijski slojevi, slojevi sažimanja i potpuno povezani slojevi). Konvolucijski slojevi izvode operaciju 1D konvolucije, izvlačeći time lokalne značajke, zatim slojevi sažimanja smanjuju dimenzionalnost izvučenih značajki, "hvatajući" najbitnije informacije te na kraju potpuno povezani slojevi integriraju ove značajke u prediktivni model učeći temeljne odnose između podataka iz skupa. Dubina i složenost same mreže mogu varirati ovisno o konkretnoj primjeni te složenosti podataka. Recimo da imamo skup podataka *x*, dužine *N* i jezgru *h* dužine *K*, izlazna aktivacijska mapa *y* se računa prema formuli (3-5), gdje se rezultirajuća vrijednost predaje aktivacijskoj funkciji *f* (npr. ReLU) kako bi se unijela nelinearnost.

$$y = f\left(\sum_{i=0}^{i+K-1} x_i * h\right) \tag{3-5}$$

Primjer: Uzmite u obzir skup podataka x = [1, 2, 3, 4, 5] i primjenu jezgre h = [0.5, 1]. Primjena 1D konvolucije sa *korakom* 1, daje aktivacijsku mapu koju je moguće koristiti u nastavku analize ili prediktivnim zadacima:

$$y[0] = f(10.5 + 21) = f(2.5)$$

$$y[1] = f(20.5 + 31) = f(3.5)$$

$$y[2] = f(30.5 + 41) = f(4.5)$$

$$y[3] = f(40.5 + 51) = f(5.5)$$

Istraživači su pokušali pronaći potencijal 1D CNN u dekodiranju elektroencefalografije (EEG) signala, postižući dobre rezultate [29]. Osim toga, primjena višekanalnih arhitektura 1D CNN pokazala je obećavajuće rezultate u zadacima klasifikacije podataka u obliku vremenskog niza, gdje više ulaznih kanala hvata različite aspekte podataka [30] te time omogućava veoma precizno predviđanje.

U sljedećem poglavlju ćemo detaljno opisati postupak prikupljanja podataka, metode uzorkovanja i druge relevantne aspekte koji su primijenjeni u samoj obradi podataka kako bi se osigurali kvalitetni ulazni podatci za naš budući CNN model. Razumijevanje ovog procesa je ključno za interpretaciju i valjanost rezultata.

4. METODOLOGIJA I ANALIZA PODATAKA

U ovom poglavlju pruža se detaljan pregled postupaka prikupljanja i obrade podataka provedenih u istraživanju. Cilj ovog poglavlja je pružiti čitatelju uvid u izvore podataka, metode prikupljanja podataka i korake u obradi i pripremi podataka za daljnju analizu.

4.1. Primijenjene tehnologije za razvoj modela i analizu podataka

4.1.1. MATLAB

Primijenjen u fazi istraživanja i analize podataka, moćan instrument u arsenalu svakog tko se bavi analizom i obradom podataka. Upotrebom naprednih statističkih tehnika i metoda vizualizacije, MATLAB omogućava sveobuhvatno razumijevanje svih skupova podataka.

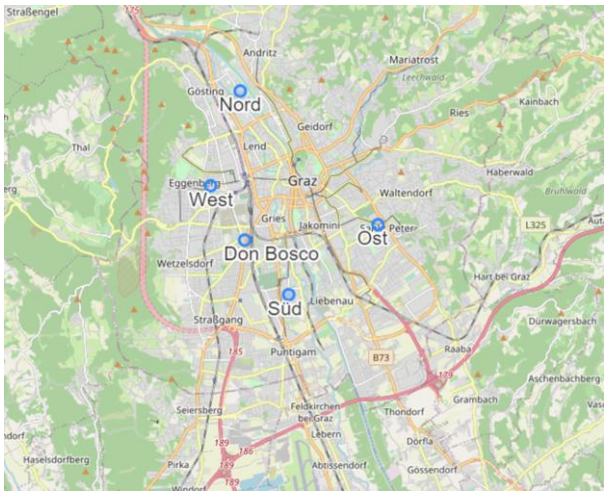
4.1.2. Python

Faza razvoja ″prediktivnog modela odvijala se na Python platformi, iskorištavajući bogati sustav biblioteka i okvira prilagođenih strojnom učenju. TensorFlow i Keras, dva temelja modernog dubokog učenja, su integrirani kako bi se izgradio CNN model. Ovaj model, osmišljeni za otkrivanje složenih odnosa unutar podataka, oblikovan je kroz iterativno fino podešavanje i rigorozno vrednovanje.

4.2. Prikupljanje, analiza i obrada podataka

4.2.1. Izvor podataka

Prije same obrade podataka, potrebno je nabaviti pouzdane i relevantne podatke. U ovom kontekstu, pažljivo su prikupljeni, katalogizirani i organizirani podaci o vremenskim varijablama i koncentracijama čestica u zraku s frekvencijom uzorkovanja od sat vremena. Podaci su preuzeti iz javnih izvora Austrijske vlade [31], te sadržavaju mjerenja prethodno spomenutih varijabli u razdoblju od 01.01.2014 do 17.03.2022.



Sl. 4.1. Prikaz karte s lokacijom mjernih stanica s kojih su skupljani podaci, preuzeto iz [5, str.3]

Sama mjerenja su dohvaćena iz Austrijskog grada Graz sa 5 mjernih stanica a to su; **Süd** (eng. *South*), **Nord** (eng. *North*), **West** (eng. *West*), **Ost** (eng. *East*) i **Don Bosco** kao što je prikazano na slici 4.1. Trenutno je i dalje najzagađeniji dio grada oko mjerne stanice Don Bosco koja se svake godine bori sa poštivanjem regulacija o česticama NO_2 te PM_{10} direktive 96/62/EC Europskog Vijeća. Detaljniji opis samih mjernih mjesta, slike i povijesni pregled moguće je vidjeti u [32]. Podaci su formatirani na način da se mjerena veličine sa pojedine stanice odvaja znakom "|" npr. mjerenje koncentracije NO čestica na mjernom mjestu Ost se označava kao "Ost|NO", mjerenje iste varijable na stanici Don Bosco bilo bi "DonBosco|NO" itd. (Sve ulazne i izlazne varijable korištene iz skupa podataka je moguće vidjeti kao P.4.1.).

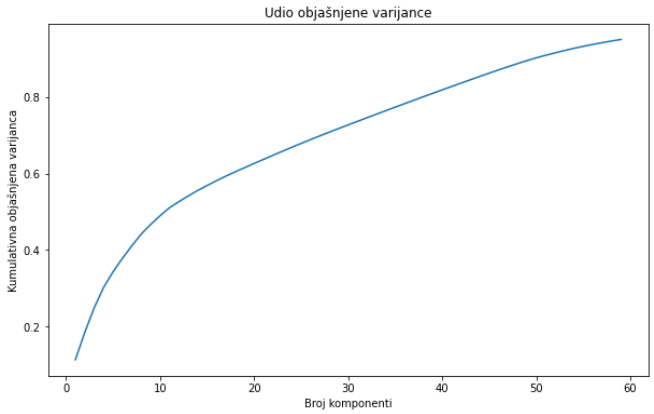
4.2.2. Analiza podataka

Analiza danog skupa podataka započinje temeljnim istraživanjem njegovih osnovnih karakteristika. Koristeći se statističkim metodama analize kao i vizualizacijama, uočena su osnovna obilježja podataka. Bitno je naglasiti da je iz podataka isključeno nekoliko mjerenja; prva dva dana u godini (1.siječnja i 2.siječnja svake godine) zbog vatrometa koji uzrokuje povećane koncentracije neželjenih čestica u zraku, kao i mjerenja između 26.03.2020 i 30.03.2020 zbog visoke koncentracije PM_{10} čestica uzrokovane Saharskom prašinom [33]. Kao i u [5], korištene su binarno kodirane temporalne vremenske varijable za sezonu, mjesec, dan u tjednu i dan u godini. Obrađeni podaci sastoje se od 71377 mjerenja (nakon izbacivanja navedenih mjerenja) što rezultira oko 2974 dana sa ukupno 64 ulazne vremenske varijable te 17 izlaznih varijabli koncentracija čestica u zraku koje je moguće pronaći na [34]. Detaljnim istraživanjem utvrđeno je kako je u podacima bilo nekoliko nedostajućih vrijednosti koje su

nadomještene vrijednostima dobivenim linearnom interpolacijom. Nadalje je potrebno analizirati same podatke te ovisnosti između istih.

4.2.3. Analiza glavnih komponenti (PCA)

Analiza glavne komponente (eng. *Principal Component Analysis*) predstavlja multivarijantnu tehniku za analiziranje tablice podataka u kojoj su opažanja opisana s nekoliko međusobno povezanih zavisnih varijabli [35]. Cilj ove tehnike jest izvući važne informacije iz tablice, prikazati ih kao skup novih ortogonalnih projekcija varijabli pod nazivom "glavne komponente" te prikazati obrazac sličnosti opažanja i varijabli kao točke na kartama. Kvalitetu PCA modela moguće je procijeniti koristeći tehnike unakrsne validacije poput *bootstrap*[6] metode ili *jackknife*[7] metode. Grafikon na slici 4.2. prikazuje vizualnu reprezentaciju udjela objašnjenih varijanci, razlučujući kumulativni doprinos svake glavne komponente. Tijekom samog izvršavanja PCA, traženo je da se objasni 95% varijance, što je učinjeno uz pomoć 59 varijabli.



Sl. 4.2. Prikaz udjela objašnjene varijance od 95%

PCA se može generalizirati kao analiza korespodencije kako bi se nosila s kvalitativnim varijablama i kao višestruka faktorska analiza (eng. *Multi-factor Analysis*) kako bi se nosila s heterogenim skupovima varijabli. Matematički PCA ovisi o svojstvenoj dekompoziciji pozitivno poludefiniranih matrica i o singularnoj dekompoziciji pravokutnih matrica.

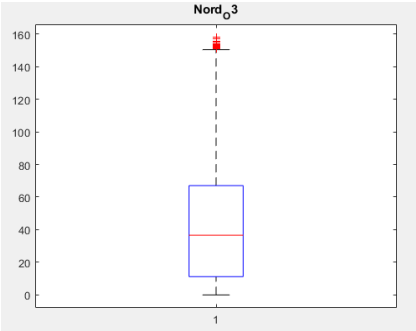
---

[6] *Bootstrap* metoda - tehnika ponovnog uzorkovanja koja se koristi za procjenu varijabilnosti statističkog procjenitelja ili za provjeru robusnosti modela.

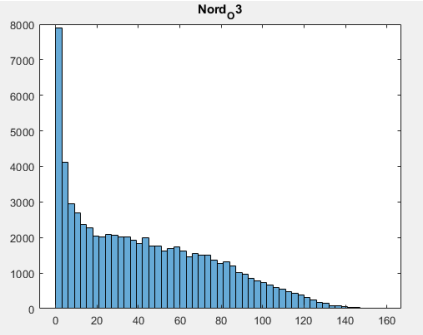
[7] *Jackknife* metoda - još jedna tehnika ponovnog uzorkovanja koja se koristi za procjenu pristranosti i varijance statističkog procjenitelja.

4.2.4. Vizualizacija distribucije podataka i veza među varijablama

Raspodjelu i intrinzične odnose unutar podataka moguće je vidjeti koristeći se nizom grafičkih prikaza kao što su box-dijagrami[8] (eng. *boxplot*), histogrami[9] i dijagrami vjerojatnosti. Navedeni prikazi nudili su nijansirane perspektive tendencija podataka kao i njihovu distribuciju i potencijalna odstupanja. Na slikama 4.3. i 4.4. moguće je vidjeti box-dijagrame i histogramski prikaz varijable Nord|O_3, odnosno koncentracije O_3 sa mjerne stranice Nord (box-dijagrami prikaze kao i histograme ostalih varijabli moguće je vidjeti kao P.4.2. odnosno P.4.3.). Sa slike 4.3. se može vidjeti kako se koncentracija središnjih 50% podataka nalazi između 10 i 65 (plavi pravokutnik), također koristeći se programskim alatima kao što je MATLAB, otkrili smo da navedeno mjerenje sadrži oko 19 stršećih vrijednosti (eng. *outlier*)[10], što je označeno crvenom bojom na vrhu slike. S druge strane slika 4.4. prikazuje distribuciju podataka odnosno učestalost pojavljivanja različitih vrijednosti. Moguće je vidjeti da ponajviše ima niskih mjerenja (iznosa 0-5) i to oko 8000 njih. Nakon analize svih vremenskih varijabli koje će se u nastavku koristiti kao ulaz u sam model za predviđanje koncentracije čestica u zraku, odrađena je i analiza svih ciljanih varijabli odnosno svih varijabli čestica u zraku koje treba predvidjeti na identičan način.



Sl. 4.3. Boxplot Nord|O_3 mjerenja



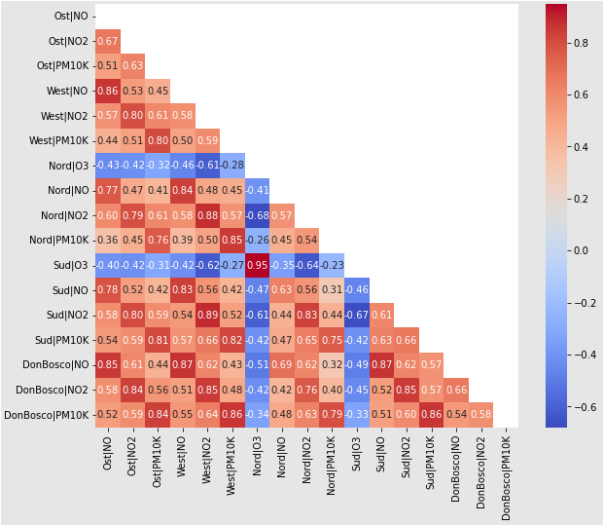
Sl. 4.4. Histogram Nord|O_3 mjerenja

4.2.5. Otkrivanje ovisnosti među podacima

Kako bi imali uvida u povezanost i ovisnost među podacima, potrebno je koristiti matricu korelacije[11]. Matrica na slici 4.5. prikazuje odnose između pojedinih ciljanih varijabli.

[8] Boxplot - grafička metoda koja prikazuje raspodjelu i osnovne karakteristike skupa podataka, kao što su medijan, kvartili te potencijalni izvanredni i ekstremni vrijednosti.
[9] Histogram -grafička reprezentacija distribucije numeričkih podataka, prikazujući učestalost pojavljivanja različitih vrijednosti u obliku niza uskih stupaca.
[10] Outlier - Podatak koji značajno odstupa od ostatka skupa podataka i može ukazivati na pogreške ili anomalije
[11] Matrica korelacije - Tablični prikaz koji prikazuje međusobne korelacije između varijabli u skupu podataka. Pomaže u identificiranju obrazaca, ovisnosti i potencijalnih veza između varijabli

Varijable s visokim pozitivnim ili negativnim korelacijama identificirane su, služeći kao vrijedni pokazatelj za naknadna razmatranja pri izradi modela za predviđanje. Također postoji i matrica korelacije svih varijabli koju je moguće pronaći kao P.4.4.



Sl. 4.5. Matrica korelacije ciljanih varijabli

Iz prikazane matrice korelacija vidimo kako gotovo sve vrijednosti osim O_3 polutanta imaju međusobnu ovisnost, drukčije rečeno, sve čestice mogu biti utjecane od strane istih faktora kao što su promet, industrijska emisija plinova i požari, kao i sve vremenske varijable koje su korištene za izradu rada. Analiza podataka provedena na ovaj način omogućava razumijevanje složenosti samog skupa podataka. Podaci su istraženi primjenom višedimenzionalog pristupa što je omogućilo rasvjetljenje strukture podataka i međusobne ovisnosti među istima. Daljnja poglavlja fokusiraju se na razvoj modela i obradu rezultata.

5. IZRADA, REZULTATI I ANALIZA MODELA

Za izradu ovog projekta najbitniji faktor je primjena 1D konvolucije nad podacima kako bi omogućili predviđanje ciljanih varijabli čestica u zraku (NO_2, NO, PM_{10} te O_3). Prvobitno je potrebno podijeliti podatke na skup podataka za učenje i skup podataka za testiranje. Uz pomoć testnog skupa će kasnije biti procjenjena točnost i preciznost modela. Podjela podataka u početku testiranja odrađena je na način da se prvih 80% podataka uzme kao skup za učenje (mjerenja od 01.01.2014 do 07.07.2020), a preostalih 20% podataka (od 07.07.2020 do 17.03.2022) uzme kao skup za testiranje. Podjelu podataka u obliku softverskog rješenja moguće je vidjeti kao P.5.1. koristeći se integriranom funkcijom *train_test_split()* iz *sklearn*[12] biblioteke. Nakon podjele, podatke je potrebno skalirati kako bi model brže konvergirao i time pružio veću preciznost odnosno bolje performanse. Samo skaliranje je također obavljeno uz pomoć funkcija *StandardScaler()* (za ulazne podatke modela) te *MinMaxScaler()* (za izlazne podatke modela) iz biblioteke *sklearn* što je također moguće vidjeti kao P.5.1.

Odabrana arhitektura modela je arhitektura veoma slična WaveNet arhitekturi koja se pokazala kao dobar izbor u nekim prethodnim istraživanjima pri regresijskim problemima koristeći se s 1D konvolucijama [36], [37] zbog mogućnosti hvatanja dugotrajnih ovisnosti među podacima. U svrhe testiranja modela (odnosno u svrhu finog podešavanja s različitim parametrima) kao argumenti funkcije *build_wavenet_model()* korišteni su; ulazni i izlazni oblik podataka, veličina filtera, broj filtera u konvolucijskim slojevima i brojem dilatacijskih slojeva[13]. Za samu izradu modela korištena je biblioteka *tensorflow*[14] odnosno njeno sučelje *keras*[15]. P.5.2. prikazuje softversko rješenje funkcije za izradu modela.

Optimiziranje parametara dovodi do veće točnosti modela pri predviđanju odnosno do manjih pogrešaka. Poznato je kako je za vrijeme trajanja COVID-19 pandemije došlo do

[12] Sklearn - široko korištena biblioteka Python-a koja omogućava alate za predobradu podataka, treniranje modela i njegovu procjenu

[13] Dilatacijski sloj - sloj unutar WaveNet arhitekture, vrsta konvolucijskog sloja u kojem se prostorna raznolikost filtera eksponencijalno širi, omogućujući modelu učinkovito hvatanje dugoročnih ovisnosti u sekvencijalnim podacima

[14] Tensorflow - Python biblioteka korištena za razvoj i treniranje modela strojnog i dubokog učenja

[15] Keras - Visokorazinsko sučelje za neuronske mreže osmišljeno kako bi olakšalo izradu i treniranje modela dubokog učenja

nekoliko *lockdown*[16] situacija u kojima je dobar postotak pučanstva bio primoran ostati u svojim domovima, čime se nešto smanjila proizvodnja i emisija ispušnih plinova, a samim time i koncentracija štetnih čestica u zraku što utječe na sposobnost modela da se nosi sa nepravilnostima u podatcima. Fino podešavanje parametara modela odrađeno je uz pomoć sljedećih opcija za pojedine parametre; broj epoha za trening modela (50/100/150/200), broj filtera (64/96/128/160/192/224/256/512), veličina filtera(2/3/4/5/6/7), broja dilatacijskih slojeva (5/6/7/8/9), optimizatora[17] (adam/sgd/rmsprop). Optimalni hiperparametri su određeni metodom pokušaja i pogreške. Ocijenjeno je više različitih konfiguracija hiperparametara, a odabrana je ona koja je dala najbolje rezultate.

Kao mjere evaluacije samog modela korištene su integrirane funkcije *sklearn* biblioteke: koeficijent determinacije (*r2_score()*) te srednja apsolutna pogreška (*mean_absolute_error()*), koja opisuje pogreške između apsolutne vrijednosti podataka testnog skupa i apsolutne vrijednosti predviđenih podataka (pogledati literaturu *skelarn* biblioteke [38]). P.5.3.prikazuje softversko rješenje uz pomoć kojega su stvarani modeli za predviđanje s različitim parametrima te potom testirani. Zaključak testiranja je da model sa parametrima kao u tablici 5.1. ostvaruje najbolje rezultate. Za nastavak analize rezultata i usporedbu s drugim poznatim modelima za predviđanje nad sekvencijalnim podacima, bit će korišten model s navedenim parametrima iz tablice.

Tablica 5.1. Parametri modela koji je ostvario najbolje rezultate

Parametar modela	**Vrijednost**
Omjer podataka (učenje/testiranje)	**80:20**
Broj epoha za treniranje/Ranije zaustavljanje	**150/114**
Veličina serije (eng. *Batch size*)	**24**
Broj dilatacijskih slojeva	**7**
Broj slojeva sažimanja	**1**

[16] Lockdown – Vrijeme tijekom kojeg su implementirane stroge mjere od strane vlasti kako bi se ograničilo kretanje i aktivnosti ljudi zbog zaustavljanja širenja virusa. Uobičajeno je to zatvaranje škola, javnih prostora i obrta koji nisu ekstremno nužni za funkcioniranje društva.

[17] Optimizator - služe za iterativno podešavanje parametara modela kako bi se minimizirala *loss* funkcija i poboljšale performanse modela

Broj filtera	**256**
Veličina filtera u konvolucijskom sloju	**3**
Veličina filtera u dilatacijskom sloju	**5**
Optimizator	**Adam**

5.1. Rezultati najboljeg modela

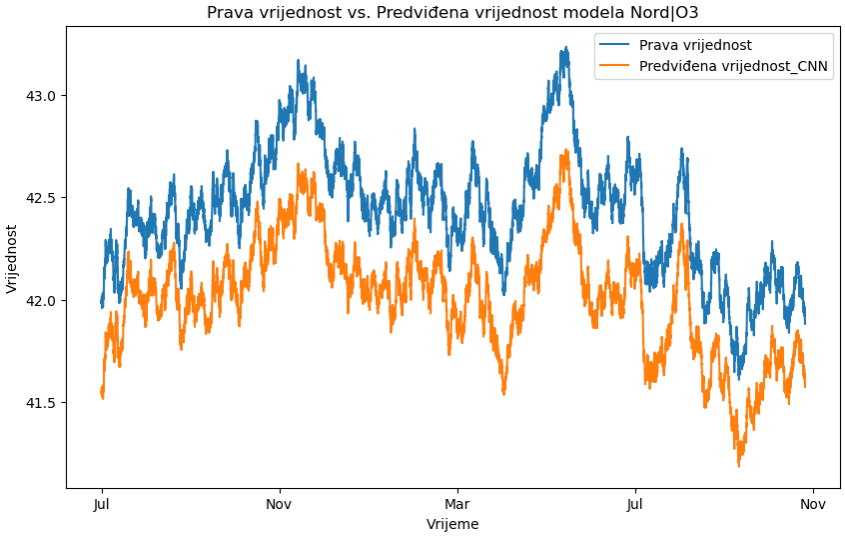
Kako bi olakšali razumijevanje o promjenama u koncentraciji štetnih čestica u zraku, učili smo navedeni CNN model na način da predviđa koncentraciju svih 17 ciljanih varijabli odjednom na temelju ulaznih podataka. Kroz razna testiranja i optimizaciju, model s parametrima iz tablice 5.1. je dao rezultate koji su prikazani u tablici 5.2. Rezultati su prikazani u smislu koeficijenta determinacije koji predstavlja statističku mjeru koja ocjenjuje koliko dobro model objašnjava varijaciju ciljanih varijabli u usporedbi s prosječnom vrijednošću ciljanih varijabli (vrijednost u intervalu [0,1], pri čemu veća vrijednost ukazuje na to da model bolje objašnjava varijaciju podataka; 1 - model savršeno objašnjava varijaciju, 0 - model se loše prilagođava podacima i loše opisuje varijaciju), MAE (eng. *Mean Absolute Error*) koji predstavlja mjeru koja kvantificira prosječno odstupanje između stvarnih i predviđenih vrijednosti u modeliranju (niža vrijednosti ukazuje na bolju usklađenost modela s podacima) te MdAPE (eng. *Median Absolute Percentage Error*).

Tablica 5.2. Rezultati najboljeg modela nad skupom za testiranje

Čestica	**R^2**	**MAE**	**MdAPE [%]**	**Medijan**	**Maksimum**	**Minimum**
Ost|NO	0.9008	6.2100	42.3956	7.0	474.0	0.0
Ost|NO_2	0.9064	3.5989	11.7627	23.0	125.4	0.0
Ost| PM_{10}	0.8182	5.5130	18.7712	21.9	271.2	0.0
West|NO	0.9137	4.4323	65.3414	2.0	359.5	0.0
West|NO_2	0.9144	3.1927	11.3429	21.0	105.0	0.0
West| PM_{10}	0.8222	4.3716	17.4234	18.9	179.3	0.0
Nord|O_3	0.9546	4.8649	11.2789	37.0	158.0	0.0
Nord|NO	0. 8913	3.3497	135.6126	1.0	314.4	0.0
Nord|NO_2	0.9191	2.6985	11.5743	17.0	86.0	0.0
Nord| PM_{10}	0.7094	5.1069	24.0225	17.2	197.9	0.0

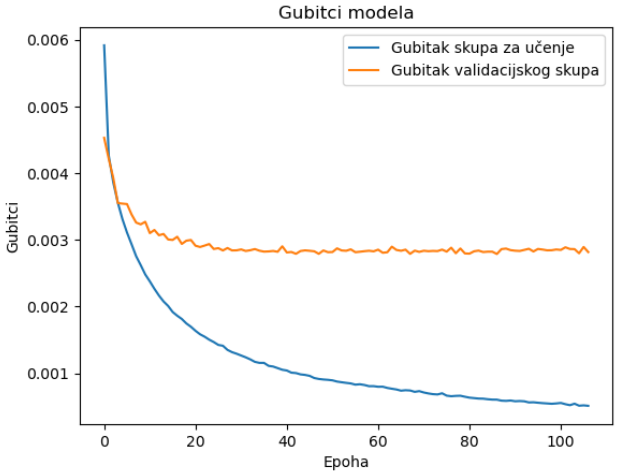
Sud|O_3	0.9602	4.5509	14.3128	29.0	163.1	0.0
Sud|NO	0.9103	7.2757	57.7369	5.0	522.0	0.0
Sud|NO_2	0.8737	4.5680	15.8798	23.0	131.9	0.4
Sud| PM_{10}	0.8723	4.3542	15.0246	20.5	176.9	0.0
DonBosco|NO	0.9159	10.5084	29.6286	22.0	688.0	0.0
DonBosco|NO_2	0.9062	4.0131	8.1764	36.0	136.8	0.0
DonBosco|PM_{10}	0.8158	5.1260	16.7144	22.5	360.8	0.0

Iz tablice 5.2. je moguće vidjeti kako izrađeni model pruža prilično dobre rezultate; koeficijent determinacije kvalitete predviđanja je prilično visok, pogotovo kada su u pitanju čestice NO, NO_2 i O_3 (gotovo sve vrijednosti su iznad 0.9000), dok je model bio nešto lošiji pri predviđanju PM_{10} čestica (u prosjeku je vrijednost približno 0.8075). To je posljedica dodatnih procesa kao što su prijenos na velike udaljenosti i sekundarna proizvodnja, čime se dodatno mjenja koncentracija PM_{10} čestica (navedeni procesi nisu toliko bitni u slučaju ostalih analiziranih zagađivača) [39]. Zbog navedenih procesa, model ima smanjenu izvedbu u slučaju čestica PM_{10}. Navedni rezultati ukazuju na to da mogućnost korištenja 1D konvolucije odnosno strojnog učenja olakšava razumijevanje stvarnog zagađenja zraka. Na slici 5.1. moguće je vidjeti prikaz koncentracije ciljane varijable O_3 sa mjernog mjesta „Nord“ na kojem su prikazane stvarne vrijednosti i predviđene vrijednosti od strane modela s pomičnim prosjekom od 4 mjeseca. Sa slike 5.1. moguće je vidjeti postojanje određenog odstupanja između predviđenih i stvarnih vrijednosti. Razlog tome je prisustvo vrijednosti koje znatno odstupaju od prosjeka unutar mjerenja koje ne bi trebali zanemariti kao i isključena mjerenja u periodu nove godine i događaja Saharske prašine.



Sl. 5.1. Prikaz stvarne vrijednosti i vrijednosti predviđene od strane modela za razdoblje od srpnja 2021. godine do studenog 2022. godine

Tijekom testiranja CNN modela korišten je i prikaz u kojem se vidi krivulja učenja koja prikazuje gubitke nad podacima za učenje i gubitke na validacijskim podacima (za validaciju modela koristi se 10% skupa za učenje tijekom samog procesa učenja modela). Ukoliko navedene krivulje prikazuju jasnu točku konvergencije modela, gdje se gubitak validacijskog skupa stabilizira, to ukazuje na to da model možda neće imati koristi od budućeg učenja nakon te točke. Navedene krivulje prikazane su na slici 5.2. Model je dostigao točku konvergencije u 114. iteraciji učenja te se uz pomoć jedne od funkcija *keras* biblioteke, *EarlyStopping()*, obustavlja učenje na način da se odredi „strpljenje“ koje, u slučaju da model određeni broj iteracija ne pokazuje napredak u učenju, obustavlja učenje.



Sl. 5.2. Prikaz krivulja gubitka skupa za učenje i gubitka validacijskog skupa

CNN model je treniran na sljedećoj hardverskoj konfiguraciji; CPU - AMD Ryzen 5 5600H (3301 MHz,6 fizičkih jezgri, 12 logičkih procesora, L1 Cache = 384KB, L2 Cache =

3.0MB, L3 Cache = 16MB), GPU – NVIDIA GeForce RTX 3050 (4GB VRAM GDDR6, 12GB Total, 1530MHz), RAM – 16GB (3200 MHz) uz OS – Windows 10 te je učenje modela trajalo 494 minute, odnosno nešto iznad 8 sati (114 epoha s prosjekom od 260s po epohi). Navedenimodel sadržava 3 314 705 parametara te su svi podesivi, što znači da se svakom iteracijom učenja, parametri iznova podešavaju kako bi se minimizirala funkcija gubitka. Model ne sadrži niti jedan fiksni odnosno statički parametar. Arhitekturu samog modela, zajedno sa ulaznim veličinama te brojem parametara po sloju moguće je vidjeti na slici 5.3. Bitno je i naglasiti samu veličinu modela koja iznosi 12.64 MB što je relativno kompaktan model.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d (Conv1D)             (None, 62, 256)           1024

 max_pooling1d (MaxPooling1  (None, 31, 256)           0
 D)

 conv1d_1 (Conv1D)           (None, 31, 256)           327936

 conv1d_2 (Conv1D)           (None, 31, 256)           327936

 conv1d_3 (Conv1D)           (None, 31, 256)           327936

 conv1d_4 (Conv1D)           (None, 31, 256)           327936

 conv1d_5 (Conv1D)           (None, 31, 256)           327936

 conv1d_6 (Conv1D)           (None, 31, 256)           327936

 conv1d_7 (Conv1D)           (None, 31, 256)           327936

 flatten (Flatten)           (None, 7936)              0

 dense (Dense)               (None, 128)               1015936

 dense_1 (Dense)             (None, 17)                2193

=================================================================
Total params: 3314705 (12.64 MB)
Trainable params: 3314705 (12.64 MB)
Non-trainable params: 0 (0.00 Byte)
```

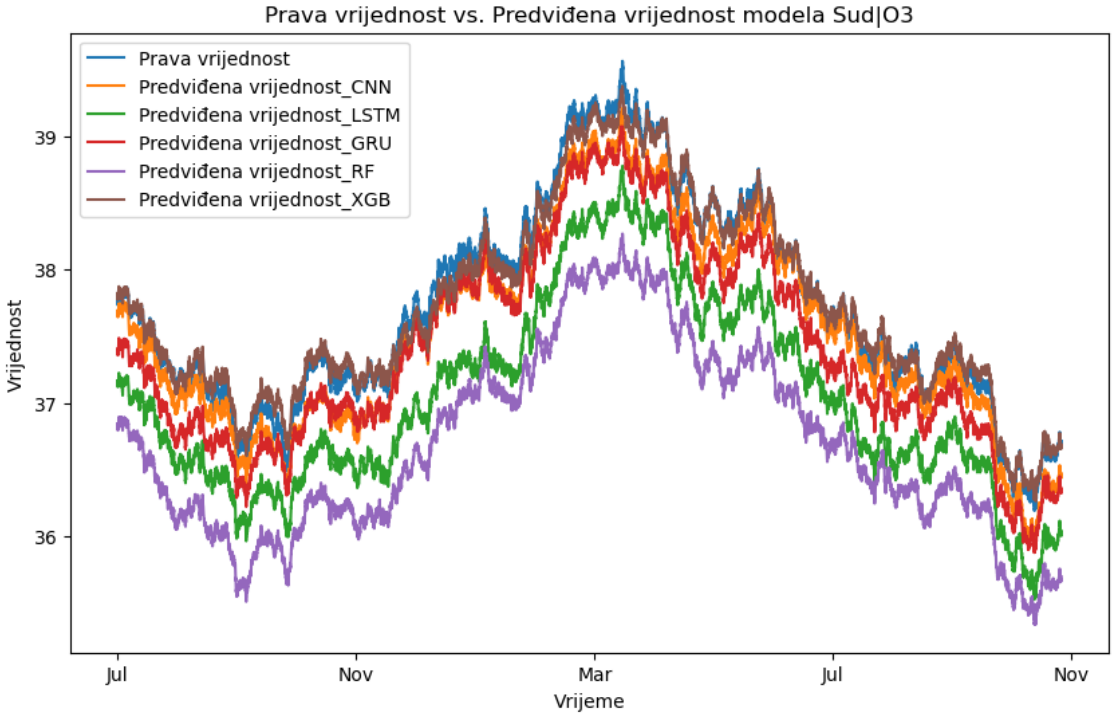
Sl. 5.3. Arhitektura najboljeg modela

Nakon optimizacije parametara i nakon završetka učenja, model sprema u *Hierarchical Data Format version 5* (HDF5 odnosno *.h5) formatu koji je široko korišten format za spremanje modela strojnog učenja gdje se sprema arhitektura modela, povezane težine i preostala konfiguracija. Spremanje modela u *.h5 formatu omogućuje njegovu kasniju upotrebu gdje ga je moguće koristiti za predviđanje bez ponovnog procesa učenja. Nakon spremanja CNN modela u navedenom formatu, novonastala datoteka *.h5 zauzima ≈ 40 MB.

5.2. Analiza i usporedba s često korištenim modelima za predviđanje nad sekvencijalnim podacima

Rezultati dobiveni od strane novostvorenog modela su uspoređeni s rezultatima LSTM modela (eng. *Long-Short Term Memory*) [40], GRU modela (eng. *Gated Recurrent Unit*) [41], *Random Forest* modela [42], te XGBoost (eng. *Extreme Gradient Boosting*) modela [43], [44] nad istim podacima. Performanse modela su, kao i u prethodnom poglavlju, prikazani u obliku koeficijenta determinacije kao i MAE za svaku ciljanu varijablu. Najbolji parametri navedenih

modela su određeni koristeći se funkcijom *GridSearchCV*() iz biblioteke *sklearn* koja stvara modele kombinirajući ranije definirane parametre pokušavajući minimizirati funkciju gubitka modela (korištene parametre za pronalazak najboljih modela moguće je vidjeti u P.5.4). U tablici 5.3. su prikazani navedeni rezultati. Također na slici 5.4. je prikazan je nivo koncentracije ciljane varijable O_3 sa mjernog mjesta „Sud“ na kojem su prikazane stvarne vrijednosti i predviđene vrijednosti svih navedenih modela s pomičnim prosjekom od 4 mjeseca.



Sl. 5.4. Prikaz stvarne vrijednosti i vrijednosti predviđene od strane svih modela za razdoblje od srpnja 2021. godine do studenog 2022. godine

Tablica 5.3. Rezultati najboljeg modela nad skaliranim skupom za testiranje

Čestica	R^2 1D CNN	MAE 1D CNN	R^2 LSTM	MAE LSTM	R^2 GRU	MAE GRU	R^2 RF	MAE RF	R^2 XGB	MAE XGB
Ost|NO	**0.9008**	6.2100	0.8835	6.7127	0.7981	8.3897	0.7664	10.2912	0.8401	7.3868
Ost|NO_2	**0.9064**	3.5989	0.8803	4.1327	0.8023	5.3265	0.7905	5.8821	0.8466	4.6886
Ost| PM_{10}	0.8182	5.5130	0.8071	5.3905	0.7039	6.7510	0.7366	6.5191	**0.8219**	5.0906
West|NO	**0.9137**	4.4323	0.9044	4.6856	0.8243	6.2447	0.7685	8.0933	0.8576	5.5190
West|NO_2	**0.9144**	3.1927	0.8947	3.6173	0.8332	4.7330	0.8171	5.2387	0.8522	4.4496
West| PM_{10}	0.8222	4.3716	0.7940	4.8000	0.6542	6.1470	0.6976	6.2122	**0.8290**	4.1441
Nord|O_3	**0.9546**	4.8649	0.9453	5.6343	0.9010	7.9393	0.9170	7.6297	0.9242	6.8803
Nord|NO	**0. 8913**	3.3497	0.8716	3.5162	0.7507	4.8289	0.7234	5.7668	0.8045	4.0485
Nord|NO_2	**0.9191**	2.6985	0.8932	3.3064	0.8180	4.4311	0.8122	4.7665	0.8372	4.1780

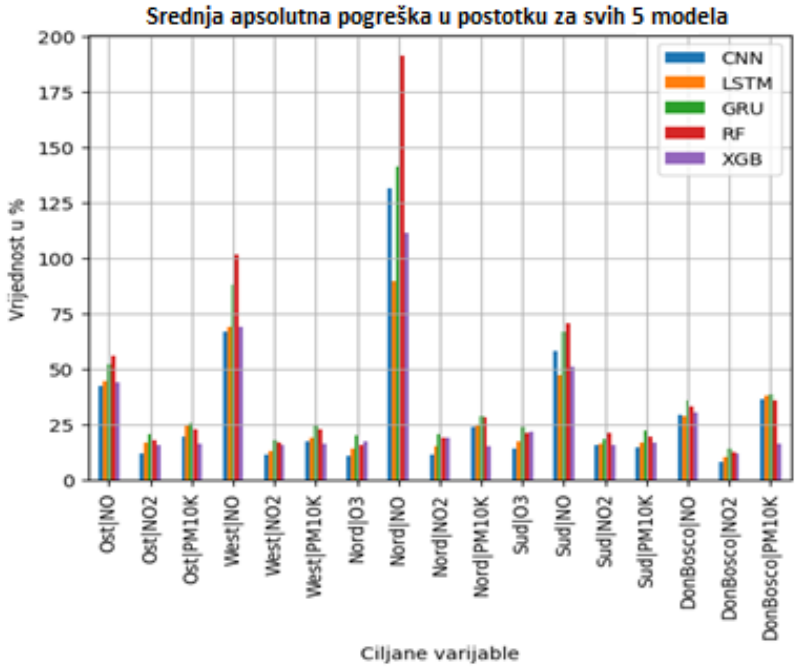
Nord| PM_{10}	0.7094	5.1069	0.6527	5.5235	0.5235	6.3713	0.6182	6.2155	**0.8353**	3.5537
Sud|O_3	**0.9602**	4.5509	0.9501	5.3614	0.9155	7.2405	0.9268	7.2333	0.9317	6.5069
Sud|NO	**0.9103**	7.2757	0.8972	7.7969	0.8290	10.7738	0.7685	13.3983	0.8531	9.8252
Sud|NO_2	**0.8737**	4.5680	0.8620	4.7530	0.8150	5.5443	0.7741	6.6125	0.8587	4.7012
Sud| PM_{10}	**0.8723**	4.3542	0.8527	4.8811	0.7491	6.5003	0.7545	6.5107	0.8450	4.9782
DonBosco|NO	**0.9159**	10.5084	0.9037	11.0673	0.8440	13.9953	0.7832	17.2618	0.8556	13.0624
DonBosco|NO_2	**0.9062**	4.0131	0.8727	4.8782	0.7887	6.4458	0.7779	6.7749	0.8375	5.6313
DonBosco|PM_{10}	0.8158	5.1260	0.8074	5.2686	0.7017	6.5032	0.7261	6.4453	**0.8229**	4.9263
PROSJEK	**0.9008**	**4.9255**	0.8824	5.3721	0.7969	6.9123	0.7778	7.3424	0.8429	5.8526

Iz tablice 5.3. moguće je vidjeti kako je CNN model nadmašio performanse ostalih modela i u pogledu koeficijenta determinacije i u pogledu apsolutne pogreške. Također sa slike 5.4., vidimo kako se predviđanje CNN modela ponajbolje poklapa sa stvarnim podacima. Svi navedeni modeli mogu biti dodatno optimizirani što bi vjerojatno donjelo bolje rezultate, no za to je potrebna kvalitetnija hardverska konfiguracija budući su kompleksniji modeli veoma resursno zahtjevni. U nastavku će biti opisana analiza predviđanja jedne varijable (Grafove svih varijabli koji prikazuju stvarne vrijednosti i predviđene vrijednosti od strane svih modela, zajedno sa koeficijentima determinacije, MAE i MdAPE rezultatima moguće je pronaći pod [45]).

Kao primjer uzmimo graf varijable Sud|O_3 (slika 5.4.). Možemo zaključiti da je koncentracija O_3 povećana ljeti, što se događa zato što O_3 nastaje fotokemijskom reakcijom između dušikovih oksida i hlapljivih organskih spojeva. Sunčeva svjetlost je potrebna za ovu reakciju, pa su koncentracije O_3 više u ljetnim mjesecima kada ima više sunčeve svjetlosti. Graf također prikazuje trend smanjenja koncentracija O_3 s vremenom. To je vjerojatno posljedica kombinacije čimbenika, uključujući smanjenje emisija dušikovih oksida i hlapljivih organskih spojeva, promjene klime i promjene u korištenju zemljišta. Opadajući trend koncentracija O_3 dobra je vijest, jer je O_3 štetan zagađivač koji može uzrokovati respiratorne probleme, bolesti srca i rak. Međutim, važno je nastaviti pratiti koncentracije O_3 i poduzimati mjere za smanjenje emisija O_3 , jer razine O_3 još uvijek mogu biti visoke, posebno u urbanim područjima.

Kako bi pobliže analizirali performanse svih modela, moguće je stvoriti prikaz MdAPE rezultata od strane svih modela koji prikazuje postotnu apsolutnu pogrešku svih modela za pojedinu varijablu. CNN model je nadmašio sve ostale modele u pogledu MdAPE rezultata za gotovo sve ciljane varijable. Graf koji prikazuje rezultate moguće je vidjeti na slici 5.5. Budući je NO vrlo reaktivan plin, njegova koncetracija u zraku može biti pod utjecajem različitih

prirodnih pojava kao što su munje, vulkanske erupcije i požari. Navedene pojave mogu dovesti do oslobađanja velikih količina NO u atmosferu, što može dodatno otežati mdoelima točno predviđanje koncentracija NO u zraku. Također, NO se emitira i iz različitih antropogenih izvora, kao što su motorna vozila, elektrane i industrijski pogoni. Navedene emisije, kao i prirodne pojave dodatno otežavaju predviđanje koncentracije NO.



Sl. 5.5. Prikaz MdAPE rezultata pojedine varijable od strane svih modela

S druge strane sa slike 5.5 moguće je vidjeti da su svi modeli ostvarili puno bolje rezultate pri predviđanju NO_2. Razlog tome jesu sami prirodni fenomeni čestice, odnosno NO_2 je puno manje reaktivan i ima stabilniju koncentraciju u atmosferi jer se zapravo stvara reakcijom NO i ozona. Dakle molekule NO_2 imaju tendenciju da ostanu u atmosferi duže od NO molekula. To dovodi do boljih MdAPE rezultata jer modeli imaju više vremena prilagođavanju promjena emisija NO_2. Osim toga NO_2 je ravnomjernije raspoređen po krajoliku što olakšava njegovo predviđanje.

Ukoliko usporedimo rezultate modela s rezultatima iz sličnog istraživanja [5], moguće je vidjeti kako je stvoreni 1D CNN model relativno uspješniji, budući da koeficijenti determinacije za korišteni RF model u prosjeku iznose 0.7000. Iako je novostvoreni modeli relativno precizan i pouzdan, potrebno je detaljnije istraživanje samih arhitektura i optimiranje hiperparametara navedenih modela kako bi dobili još bolje performanse modela. Model iz istraživanja [5] stvoren da previđa podatke tijekom *lockdown* perioda za vrijeme trajanja COVID-19 pandemije, čime se kvaliteta zraka znatno promjenila što je u konačnici rezultiralo na performanse modela. Tijekom

izrade modela u ovom radu, ti podatci su korišteni za učenje te se stoga preformanse modela ne mogu direktno usporediti.

6. ZAKLJUČAK

Ovo istraživanje prikazuje razvoj modela konvolucijske neuronske mreže u svrhu procjene koncentracije štetnih čestica u zraku odnosno, za predviđanje same kvalitete zraka. Korišteni su vremenski podaci Austrijske vlade u razdoblju od 01.01.2014 do 17.03.2022, sakupljeni sa 5 mjernih mjesta. Skup podataka u obliku vremenskog niza podijeljen je tako da je korišteno 80% podataka za trening te 20% podataka za testiranje. Istraživanje je provedeno nad 71377 uzoraka, s 64 ulazne varijable i 17 izlaznih varijabli. Cilj rada je bio analizirati podatke, pronaći ovisnosti među njima, te identificirati najučinkovitiji model 1D konvolucijske neuronske mreže, pronalaženjem optimalnih parametara i arhitekture modela. Najbolji rezultati su ostvareni kod čestica dušikovih oksida (NO, NO_2) gdje je koeficijent determinacije bio preko 90%, dok su najlošiji rezultati bili nad PM_{10} česticama (koeficijent determinacije je oko 80%).

Korištenjem različitih parametara, ostvaren je veoma pouzdan model sa sljedećim parametrima; broj epoha za trening: 150, veličina serije: 24, broj dilatacijskih slojeva: 7, broj slojeva sažimanja: 1, broj filtera po sloju: 256, veličina filtera u konvolucijskom sloju: 3x3, veličina filtera u dilatacijskom sloju: 5x5 te je korišten Adam optimizator. Rezultati modela nisu uspoređeni s nekim sličnim istraživanjima kao što je [5], jer je skup podataka nad kojima su testirani modeli, uključen u trening skup podataka modela iz rada (budući su podaci korišteni za testiranje modela iz drugih radova gotovo cijeli uključeni u podatke za učenje 1D CNN modela iz rada). Iako je 1D CNN model u radu pokazao veoma dobre rezultate, potrebno je detaljnije analizirati moguće arhitekture uz veći broj resursa na raspolaganju budući je proces učenja veoma računalno zahtjevan.

Zaključno, ovo istraživanje nastojalo je značajno doprinijeti području procjene kvalitete zraka koristeći konvolucijske neuronske mreže za modeliranje dnevnih koncentracija čestica u zraku. Razvijeni model dokazao je svoju učinkovitost u predviđanju razina štetnih čestica u zraku, ističući potencijal 1D konvolucije u rješavanju postavljenog izazova. Iako primjena 1D CNN-ova u tu svrhu ostaje relativno nedovoljno istražena, preciznost i pouzdanost u otkrivanju ponavljajućih uzoraka u podacima pružaju obećavajuće rješenje za precizno prognoziranje štetnih čestica u zraku. Ovo istraživanje ne samo da pruža vrijedne uvide u predviđanje kvalitete zraka, već također postavlja temelje za buduća istraživanja u ovoj domeni, potičući nastavak istraživanja i napredovanje tehnika modeliranja za praćenje okoliša. Za buduća istraživanja bi bilo veoma korisno detaljnije istražiti arhitekturu kao i optimizaciju hipermparametara modela za veću točnost i pouzdanost.

LITERATURA

[1] Watson, John G., Zhiqiang Lu, Douglas H. Lowenthal, Clifton A. Frazier, Paul A. Solomon, Richard H. Thuillier, and Karen Magliano. "Descriptive Analysis of PM2.5 and PM10 at Regionally Representative Locations during SJVAQS/AUSPEX." *A WMA International Specialty Conference on Regional Photochemical Measurements and Modeling* 30, no. 12 (June 1, 1996): 2079–2112., dostupno na: https://doi.org/10.1016/1352-2310(95)00402-5.

[2] Zhao, Suping, Ye Yu, Daiying Yin, Jianjun He, Na Liu, Jianjun Qu, and Jianhua Xiao. "Annual and Diurnal Variations of Gaseous and Particulate Pollutants in 31 Provincial Capital Cities Based on in Situ Air Quality Monitoring Data from China National Environmental Monitoring Center." *Environment International* 86 (2016): 92–106., dostupno na: https://doi.org/10.1016/j.envint.2015.11.003.

[3] Park, Sangsoo, Taesup Moon, Yongyun Li, and Shanhe Zhu. "Estimating PM2.5 Concentration of the Conterminous United States via Interpretable Convolutional Neural Networks." *Environmental Research Letters* 15, no. 4 (April 2020): 044013., dostupno na: https://doi.org/10.1088/1748-9326/ab6bd1.

[4] Bozdag, Asli, Yesim Dokuz, and Oznur Begum Gokcek. "Spatial Prediction of PM10 Concentration Using Machine Learning Algorithms in Ankara, Turkey." *Water, Air, & Soil Pollution* 231, no. 6 (2020): 1–14., dostupno na: https://doi.org/10.1007/s11270-020-04542-5.

[5] Lovrić, Mario, Kristina Pavlović, Matej Vuković, Stuart K. Grange, Michael Haberl, and Roman Kern. "Understanding the True Effects of the COVID-19 Lockdown on Air Pollution by Means of Machine Learning." *Environmental Pollution* 274 (2021): 115900., dostupno na: https://doi.org/10.1016/j.envpol.2020.115900.

[6] Huang, Guoyan, Xinyi Li, Bing Zhang, and Jiadong Ren. "PM2.5 Concentration Forecasting at Surface Monitoring Sites Using GRU Neural Network Based on Empirical Mode Decomposition." Science of The Total Environment 768 (2021): 144516. https://doi.org/10.1016/j.scitotenv.2020.144516.

[7] McCulloch, Warren S., and Walter Pitts. "A Logical Calculus of the Ideas Immanent in Nervous Activity." *The Bulletin of Mathematical Biophysics* 5, no. 4 (December 1, 1943): 115–33., dostupno na: https://doi.org/10.1007/BF02478259.

[8] Rosenblatt, Frank. “Perceptron Simulation Experiments.” *Proceedings of the IRE* 48 (1960): 301–9.

[9] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[10] Osnovni model perceptrona, dostupno na: https://repozitorij.fsb.unizg.hr/islandora/object/fsb%3A8996/datastream/PDF/view [12.7.2023]

[11] Sharma, S., Sharma S., Activation Functions in neural networks (2020), International Journal of Engineering Applied Sciences and Technology Vol. 4, Issue 12, ISSN No. 2455-2143, Pages 310-316 Sharma, Siddharth, Simone Sharma, and Anidhya Athaiya. “ACTIVATION FUNCTIONS IN NEURAL NETWORKS.” *International Journal of Engineering Applied Sciences and Technology* 04 (May 2020): 310–16., dostupno na: https://doi.org/10.33564/IJEAST.2020.v04i12.054.

[12] Roodschild, Matías, Jorge Gotay Sardiñas, and Adrián Will. “A New Approach for the Vanishing Gradient Problem on Sigmoid Activation.” *Progress in Artificial Intelligence* 9, no. 4 (December 1, 2020): 351–60., dostupno na: https://doi.org/10.1007/s13748-020-00218-y.

[13] Sharma, Ochin. “A New Activation Function for Deep Neural Network,” 84–86, 2019., dostupno na: https://doi.org/10.1109/COMITCon.2019.8862253.

[14] Li, Jing, Ji-hang Cheng, Jing-yuan Shi, and Fei Huang. “Brief Introduction of Back Propagation (BP) Neural Network Algorithm and Its Improvement.” In *Advances in Computer Science and Information Engineering*, edited by David Jin and Sally Lin, 553–58. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[15] Bolf N., Osvježimo znanje, Umjetne neuronske mreže, Zavod za mjerenja i automatsko vođenje procesa, Zagreb, 2019, dostupno na: https://hrcak.srce.hr/file/322233 [10.7.2023]

[16] Primjer nedovoljnog treniranja mreže, dobro trenirane mreže i pretreniranja mreže, dostupno na: https://www.kevindegila.com/images/over-underfitting.png

[17] Fukushima, Kunihiko. “Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position.” *Biological Cybernetics* 36, no. 4 (April 1, 1980): 193–202., dostupno na: https://doi.org/10.1007/BF00344251.

[18] Lecun, Yann, Leon Bottou, Y. Bengio, and Patrick Haffner. “Gradient-Based Learning Applied to Document Recognition.” *Proceedings of the IEEE* 86 (December 1998): 2278–2324., dostupno na: https://doi.org/10.1109/5.726791.

[19] Guo, Tianmei, Jiwen Dong, Henjian Li, and Yunxing Gao. “Simple Convolutional Neural Network on Image Classification.” *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, 2017, 721–24.

[20] Jednostavna CNN arhitektura od 5 slojeva, dostupno na: https://www.mdpi.com/electronics/electronics-09-01140/article_deploy/html/images/electronics-09-01140-g001-550.jpg

[21] Baldominos, Alejandro, Yago Saez, and Pedro Isasi. “A Survey of Handwritten Character Recognition with MNIST and EMNIST.” *Applied Sciences* 9, no. 15 (2019)., dostupno na: https://doi.org/10.3390/app9153169.

[22] Aktivacije preuzete iz prvog konvolucijskog sloja jednostavne duboke CNN mreže nakon obuke na MNIST bazi podataka, dostupno na: https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSzcKSxHmt_kaX8VIhuVsqMjwtcHX7HnyrFo7T0FqtcWYnjL7OI

[23] Prikaz konvolucijskog sloja, dostupno na: https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTKH57FoBuN5Hqznre758W-rrcinC9CaIdqoL7b1YDXSD3XgQtB

[24] Primjer konvolucije unutar konvolucijskog sloja, dostupno na: https://encrypted-tbn3.gstatic.com/images?q=tbn:ANd9GcSnGpcZKjHN90squGSpTec_dJUQATGuMbZmvS_TgBEzya8-vddD

[25] Konvolucijske neuronske mreže, dostupno na: https://cs231n.github.io/convolutional-networks/, CS231n Convolutional Neural Networks for Visual Recognition [12.7.2023]

[26] Primjer *max pooling* operacije nad 4 x 4 podacima s primjenom 2 x 2 filtera, dostupno na: https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTTB96i_SKjP1YSvlkewD6aS9kzb-UC9r7SbMJQlGWNDahT0Ww6

[27] Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. “Dropout: A Simple Way to Prevent Neural Networks from Overfitting.” *Journal of Machine Learning Research* 15, no. 56 (2014): 1929–58.

[28] Lijevo: Uobičajena neuronska mreža sa potpuno povezanim slojevima, Desno: CNN s primjerom arhitekture neurona u 3 dimenzije, dostupno na: https://encrypted-

tbn0.gstatic.com/images?q=tbn:ANd9GcSXjw2B-h-3NnmhFcr1koLPEi9zcbNDdHQWdLKNDghufYtOII90

[29] Schirrmeister, Robin Tibor, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. "Deep Learning with Convolutional Neural Networks for EEG Decoding and Visualization." *Human Brain Mapping* 38, no. 11 (August 2017): 5391–5420., dostupno na: https://doi.org/10.1002/hbm.23730.

[30] Zheng, Yi, Qi Liu, Enhong Chen, Yong Ge, and J. Leon Zhao. "Time Series Classification Using Multi-Channels Deep Convolutional Neural Networks." In *Web-Age Information Management*, edited by Feifei Li, Guoliang Li, Seung-won Hwang, Bin Yao, and Zhenjie Zhang, 298–310. Cham: Springer International Publishing, 2014., dostupno na: https://doi.org/10.1007/978-3-319-08010-9_33

[31] Podaci Austrijske vlade, dostupno na: https://www.umwelt.steiermark.at/cms/ziel/2060750/DE/ [12.7.2023]

[32] Moser, F., Kleb, U., Katz, H., 2019. Statistische Analyse der Luftqualitatin Graz anhand von Feinstaub und Stickstoffdioxid. Graz

[33] Federal Office: MeteoSwiss, 2020. Saharan dust events - MeteoSwiss. accessed 31.7.2023., dostupno na: https://www.meteoswiss.admin.ch/home/climate/the-climate-of-switzerland/specialties-of-the-swissclimate/saharan-dust-events.html

[34] Gudelj, I., "Meteorološki podaci iz grada Graz, Austrija". Zenodo, December 4, 2023., dostupno na: https://doi.org/10.5281/zenodo.10253451.

[35] Abdi, Hervé, and Lynne J. Williams. "Principal Component Analysis." *WIREs Computational Statistics* 2, no. 4 (2010): 433–59, dostupno na: https://doi.org/10.1002/wics.101.

[36] Dorado Rueda, Fernando, Jaime Durán Suárez, and Alejandro del Real Torres. "Short-Term Load Forecasting Using Encoder-Decoder WaveNet: Application to the French Grid." *Energies* 14, no. 9 (2021)., dostupno na: https://doi.org/10.3390/en14092524

[37] Borovykh, Anastasia, Sander Bohte, and Cornelis W Oosterlee. "Conditional Time Series Forecasting with Convolutional Neural Networks." *arXiv Preprint arXiv:1703.04691*, 2018.

[38] Selekcija modela, dostupno na:https://scikit-learn.org/stable/model_selection.html#model-selection, Model selection and evaluation [17.9.2023]

[39] Grange, S. K., D. C. Carslaw, A. C. Lewis, E. Boleti, and C. Hueglin. “Random Forest Meteorological Normalisation Models for Swiss \chemPM_10 Trend Analysis.” *Atmospheric Chemistry and Physics* 18, no. 9 (2018): 6223–39., dostupno na: https://doi.org/10.5194/acp-18-6223-2018.

[40] Diqi, Mohammad, Hamz, Indriana Hapsari, Leon Andretti Abdillah, and Adinda Novitasari. “Enhancing Weather Prediction Using Stacked Long Short-Term Memory Networks.” *Jurnal Teknik Informatika Dan Sistem Informasi* 10, no. 3 (2023): 519–30.

[41] Pirani, Muskaan, Paurav Thakkar, Pranay Jivrani, Mohammed Husain Bohara, and Dweepna Garg. “A Comparative Analysis of ARIMA, GRU, LSTM and BiLSTM on Financial Time Series Forecasting.” In *2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)*, 1–6, 2022., dostupno na: https://doi.org/10.1109/ICDCECE53908.2022.9793213.

[42] Regier, Peter, Matthew Duggan, Allison Myers-Pigg, and Nicholas Ward. “Effects of Random Forest Modeling Decisions on Biogeochemical Time Series Predictions.” *Limnology and Oceanography: Methods* 21, no. 1 (2023): 40–52, dostupno na: https://doi.org/10.1002/lom3.10523

[43] Luo, Junling, Zhongliang Zhang, Yao Fu, and Feng Rao. “Time Series Prediction of COVID-19 Transmission in America Using LSTM and XGBoost Algorithms.” *Results in Physics* 27 (2021): 104462, dostupno na: https://doi.org/10.1016/j.rinp.2021.104462

[44] Zamani Joharestani, Mehdi, Chunxiang Cao, Xiliang Ni, Barjeece Bashir, and Somayeh Talebiesfandarani. “PM2.5 Prediction Based on Random Forest, XGBoost, and Deep Learning Using Multisource Remote Sensing Data.” *Atmosphere* 10, no. 7 (2019), dostupno na: https://doi.org/10.3390/atmos10070373

[45] Gudelj, I., “Rezultati CNN, LSTM, RF, GRU, XGB modela za modeliranje dnevnih koncentracija čestica u zraku”. Zenodo, December 4, 2023., dostupno na: https://doi.org/10.5281/zenodo.10253479

[46] Gudelj, I. “Programska podrška korištena za analizu podataka, izradu modela te njegovo optimiziranje (modeliranje dnevnih koncentracija čestica u zraku pomoću konvolucijskih neuronskih mreža)”. Zenodo, December 4, 2023., dostupno na: https://doi.org/10.5281/zenodo.10253515

SAŽETAK

Ovaj rad fokusira se na unapređenje predikcije dnevnih koncentracija čestica u zraku na urbanim mjernim mjestima korištenjem konvolucijskih neuronskih mreža (CNN). Predviđeno je da 1D CNN, u usporedbi s algoritmima poput Random Forests, ima veću moć predviđanja. Podaci za predikciju oblikovani su kao vremenske multivarijatne vremenske serije.

Analiza podataka provodila se korištenjem Pythona i MATLAB-a, dok je testiranje parametara modela izvedeno u Pythonu kako bi se identificirali optimalni parametri i stvorio što precizniji model.

Rezultati pokazuju da je CNN model nadmašio performanse drugih modela, izraženo kroz koeficijente determinacije i apsolutne pogreške. Prikazuje se kako se predviđanja CNN modela usklađuju s stvarnim podacima, a svi modeli ukazuju na mogućnosti dodatne optimizacije. Daljnja analiza koncentracije ozona (O3) ilustrira sezonske varijacije i ukazuje na važnost praćenja koncentracija štetnih tvari u zraku. Iako su rezultati obećavajući, naglašava se potreba za nastavkom praćenja i smanjenjem emisija kako bi se očuvala kvaliteta zraka, posebno u urbanim područjima.

Ovo istraživanje ukazuje na uspješnost 1D CNN u predviđanju koncentracija čestica u zraku, pridonoseći novim saznanjima i potičući daljnja istraživanja na polju kvalitete zraka.

Ključne riječi: kvaliteta zraka, konvolucija, predviđanje zagađenja, umjetne neuronske mreže,

ABSTRACT

MODELING DAILY CONCENTRATIONS OF PARTICLES IN THE AIR USING CONVOLUTIONAL NEURAL NETWORKS

This study focuses on improving the prediction of daily air particle concentrations at urban monitoring sites using 1D convolutional neural networks (CNN). It is hypothesized that 1D CNN, compared to algorithms like Random Forests, possesses greater predictive power. The prediction data is structured as multivariate time series.

Data analysis was conducted using Python and MATLAB, and model parameter testing was performed in Python to identify optimal parameters and create the most efficient model.

Results indicate that the CNN model outperformed other models, as evidenced by coefficients of determination and absolute errors. The predictions of the CNN model align with actual data, with all models suggesting potential for further optimization. Further analysis of ozone (O3) concentrations illustrates seasonal variations and underscores the importance of monitoring harmful air pollutant concentrations. While the results are promising, there is a need for ongoing monitoring and emission reduction efforts to preserve air quality, especially in urban areas.

This research highlights the success of 1D CNN in predicting air particle concentrations, contributing new insights and encouraging further research in the field of air quality.

Keywords: air quality, artificial neural networks, convolution, pollution prediction

PRILOZI

Prilog 4.1. Tablica ulaznih vremenskih varijabli s njihovim opisima iz skupa podataka nad kojim je model treniran.

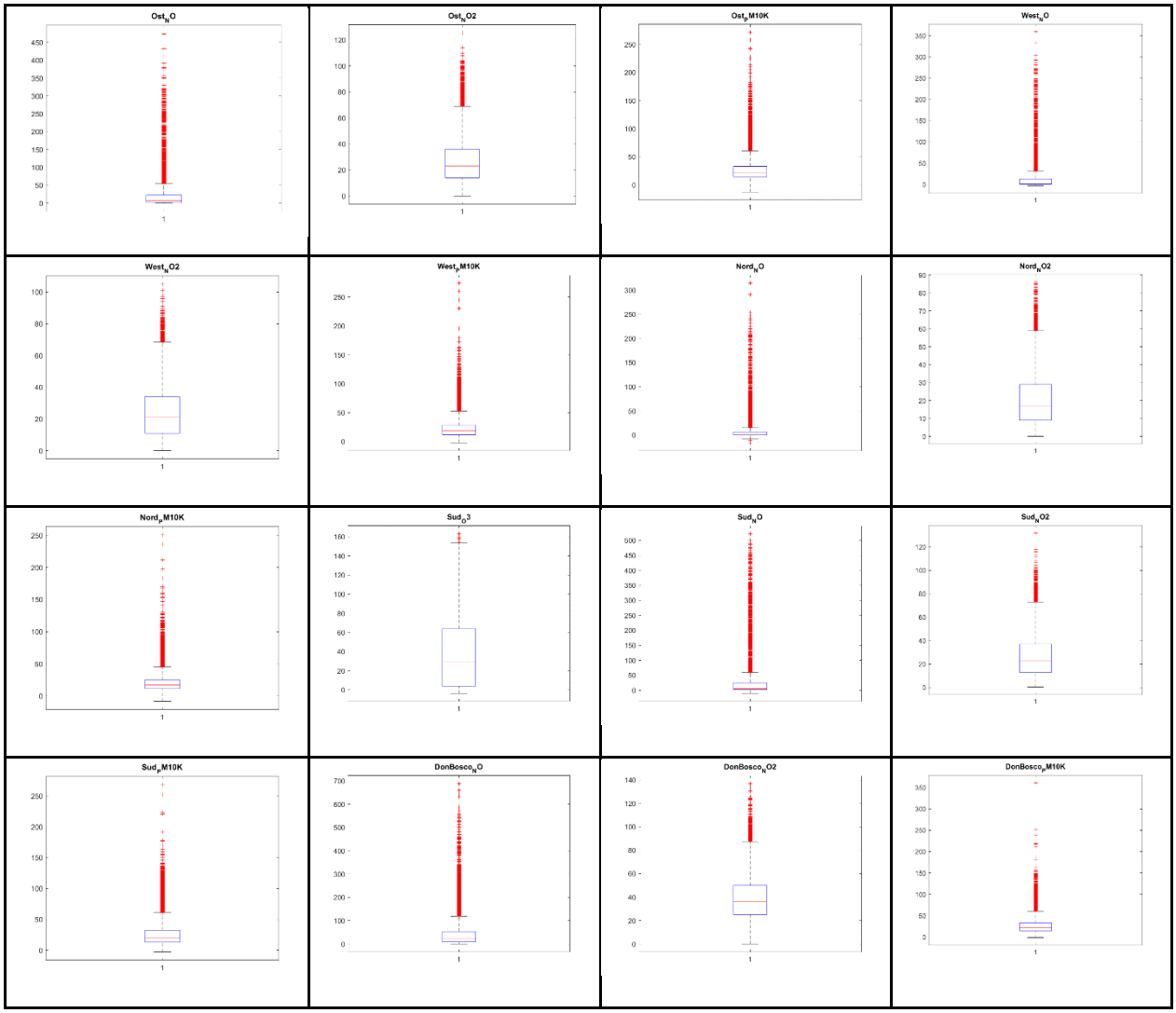
Varijabla	Opis varijable
Ulazne varijable	
sf	Površinski udio(udio površine prekriven određenom značajkom ili svojstvom)
slhf	Površinski latentni toplinski tok (brzina latentnog prijenosa topline između površine i atmosfere)
smlt	Otapanje snijega (stopa otapanja snijega)
snowc	Snježni pokrivač (opseg ili dubina snježnog pokrivača)
speed	Brzina vjetra (brzina vjetra)
sshf	Površinski osjetni toplinski tok (stopa osjetnog prijenosa topline između površine i atmosfere)
stl4	Razina temperature tla 4 (temperatura na određenoj dubini tla)
str	Kratkovalno zračenje reflektirano od vrha atmosfere (eng *Top of The Atmosphere* - TOA) (količina kratkovalnog zračenja reflektirana od Zemljine atmosfere)
strd	Kratkovalno zračenje vrha atmosfere (TOA) prema dolje (količina kratkovalnog zračenja koja doseže vrh Zemljine atmosfere)
tp	Ukupna količina padalina (količina oborina, uključujući kišu, snijeg itd.)
tsn	Snježne padaline (količina snijega)
u10	Istočna komponenta vjetra na 10 metara iznad površine
v10	Sjeverna komponenta vjetra na 10 metara iznad površine
rsn	Stopa padanja snijega
sd	Dubina snijega (dubina snijega na tlu)
sde	Isparavanje snijega (brzina isparavanja snijega)
dwm	Duboka vlažnost tla (sadržaj vlage u dubljim slojevima tla)
fal	Frakcija apsorbiranog fotosintetski aktivnog zračenja (PAR) korištenog za fotosintezu lišća

Lag_DonBosco (predstavlja zakašnjele vrijednosti) & DonBosco	RH	Relativna vlaga
	Temp	Temperatura
Lag_Nord & Nord	Percip	Količina padalina
	Pressure	Atmosferski tlak
	Radiation	Radijacija
	Winddir_cos*speed	Komponente smjera i brzine vjetra
	Winddir_sin*speed	
	Windspeed	Brzina vjetra
	Peek_wind_speed	Vrhunska brzina vjetra
Lag_Sud & Sud	Winddir_cos*speed	Komponente smjera i brzine vjetra
	Winddir_sin*speed	
	Windspeed	Brzina vjetra
Lag_Ost & Ost	Winddir_cos*speed	Komponente smjera i brzine vjetra
	Winddir_sin*speed	
	Windspeed	Brzina vjetra
Lag_West & West	Winddir_cos*speed	Komponente smjera i brzine vjetra
	Winddir_sin*speed	
	Windspeed	Brzina vjetra

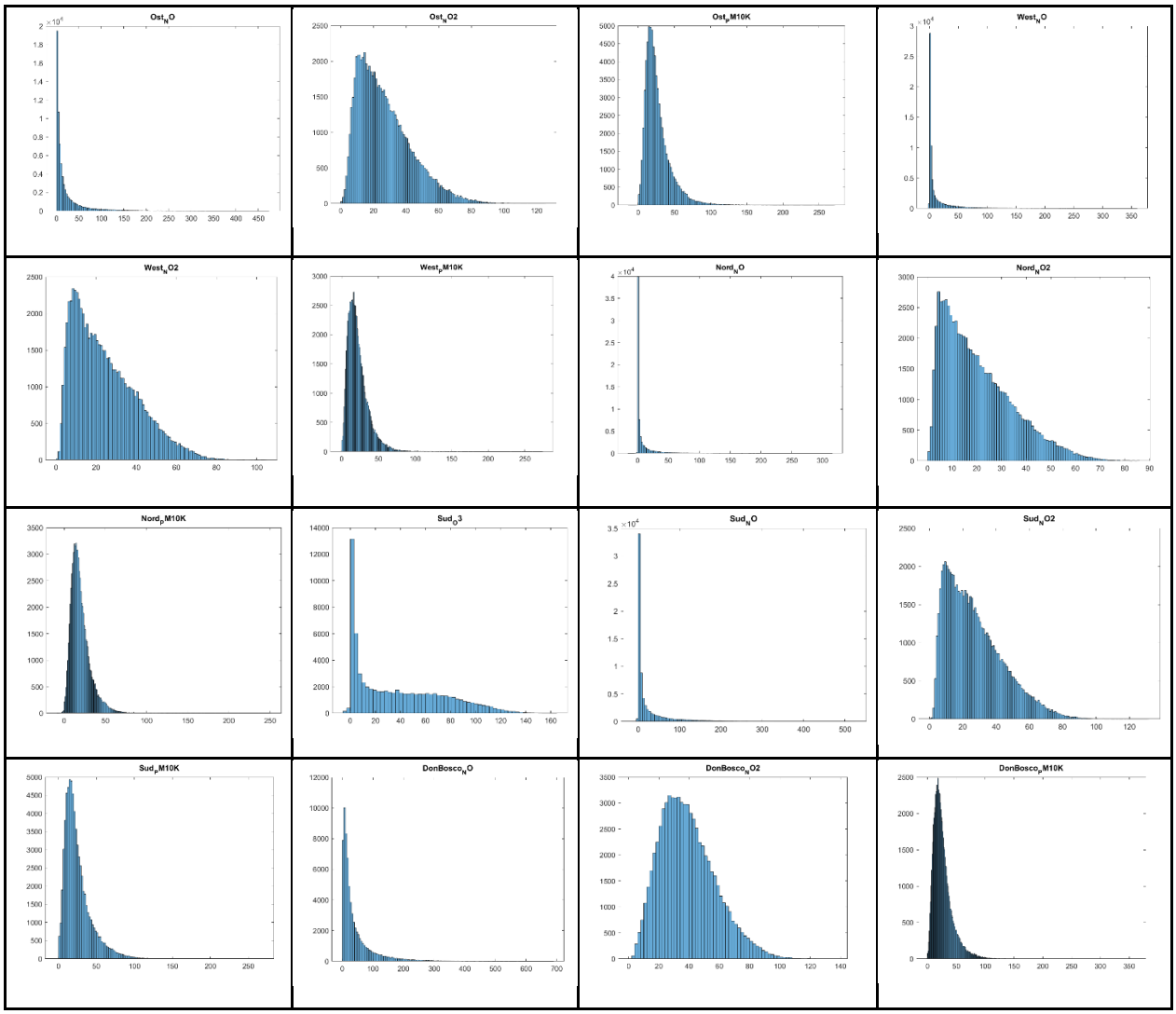
Binarno kodirane ulazne tempopralne varijable					
Holiday	Hour_c1	Hour_c2	Hour_c3	Hour_c4	Hour_c5
Month_Jan	Month_Feb	Month_Mar	Month_Apr	Month_May	Month_Jun
Month_Jul	Month_Aug	Month_Sep	Month_Oct	Month_Nov	Month_Dec
Weekday_Mon	Weekday_Tue	Weekday_Wed	Weekday_Thu	Weekday_Fri	Weekday_Sat

Season_spring	Season_summer	Season_fall
Izlazne varijable		
Ost|NO	Koncentracija NO na mjernoj stanici Ost	
Ost|NO_2	Koncentracija NO_2 na mjernoj stanici Ost	
Ost| PM_{10}	Koncentracija PM_{10} na mjernoj stanici Ost	
West|NO	Koncentracija NO na mjernoj stanici West	
West|NO_2	Koncentracija NO_2 na mjernoj stanici West	
West| PM_{10}	Koncentracija PM_{10} na mjernoj stanici West	
Nord|O_3	Koncentracija O_3 na mjernoj stanici Nord	
Nord|NO	Koncentracija NO na mjernoj stanici Nord	
Nord|NO_2	Koncentracija NO_2 na mjernoj stanici Nord	
Nord| PM_{10}	Koncentracija PM_{10} na mjernoj stanici Nord	
Sud|O_3	Koncentracija O_3 na mjernoj stanici Sud	
Sud|NO	Koncentracija NO na mjernoj stanici Sud	
Sud|NO_2	Koncentracija NO_2 na mjernoj stanici Sud	
Sud| PM_{10}	Koncentracija PM_{10} na mjernoj stanici Sud	
DonBosco|NO	Koncentracija NO na mjernoj stanici Don Bosco	
DonBosco|NO_2	Koncentracija NO_2 na mjernoj stanici Don Bosco	
DonBosco|PM_{10}	Koncentracija PM_{10} na mjernoj stanici DonBosco	

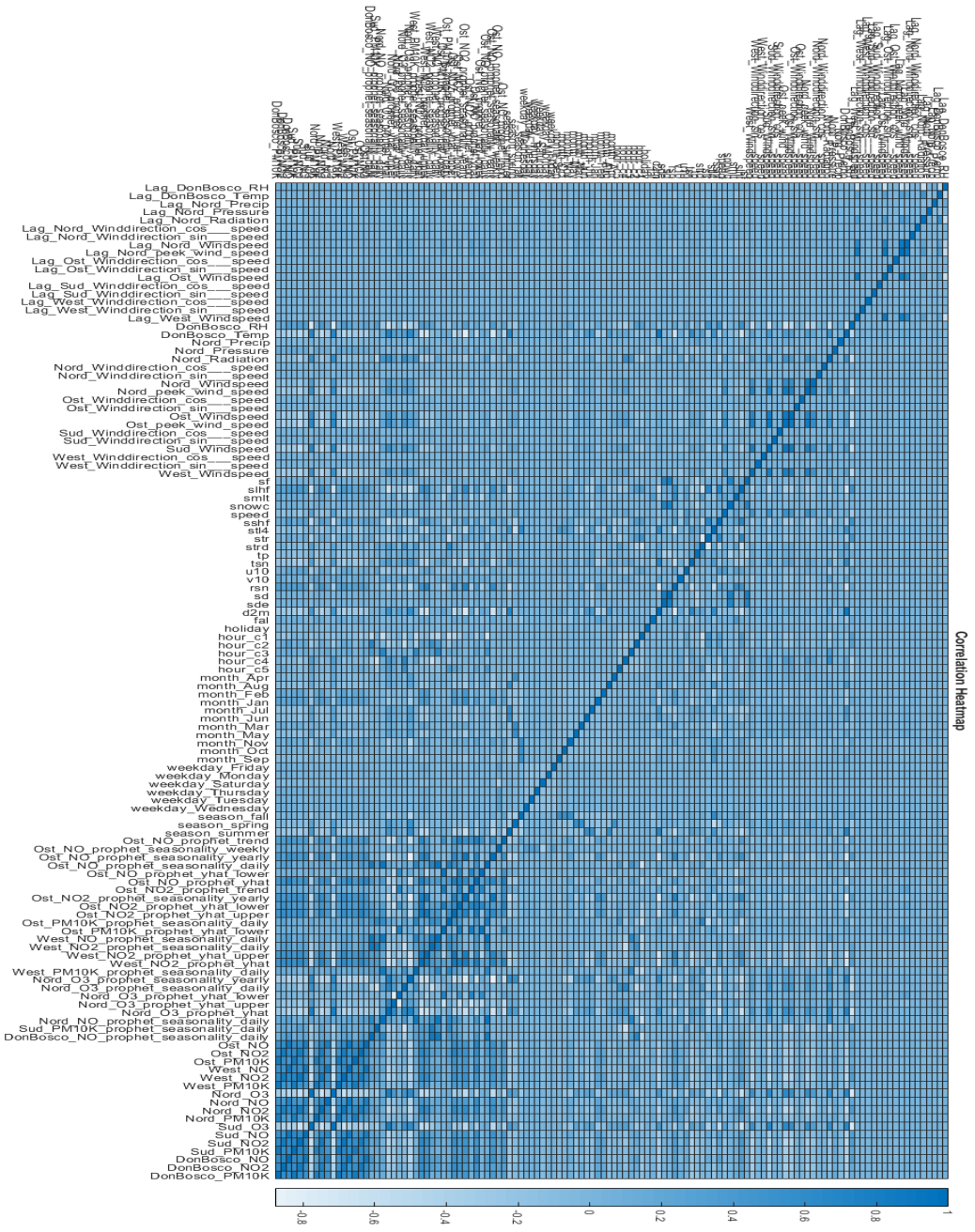
Prilog 4.2. Tablica s *Boxplot* prikazom pojedine izlazne varijable (bez Nord|O_3 jer je korištena kao primjer u radu).



Prilog 4.3. Tablica s Histogramskim prikazom pojedine izlazne varijable (bez Nord|O_3 jer je korištena kao primjer u radu).



Prilog 4.4. Matrica korelacije svih varijabli u skupu podataka (uključujući i ulazne podatke koji su izostavljeni).



Prilog 5.1. Podjela podataka na skup za učenje i skup za testiranje te skaliranje skupova

Linija ***Kod***

```
1:  #Splitting the data on train and test
2:  inputData_train, inputData_test, outputData_train, outputData_test =
    train_test_split(inputData, outputData, test_size=0.2)
3:  inputScaler = StandardScaler()
4:  outputScaler = MinMaxScaler()
5:  inputScaler.fit(inputData_train)
6:  outputScaler.fit(outputData_train)
7:  scaled_inputData_train = inputScaler.transform(inputData_train)
8:  scaled_inputData_test = inputScaler.transform(inputData_test)
9:  scaled_outputData_train = outputScaler.transform(outputData_train)
10: scaled_outputData_test = outputScaler.transform(outputData_test)
11: print("Scaled Input Train data shape:",scaled_inputData_train.shape)
12: print("Scaled Output Train data shape:
    ",scaled_outputData_train.shape)
13: print("-------------------------------------------")
14: print("Scaled Input Test data shape:",scaled_inputData_test.shape)
15: print("Scaled Output Test data shape:",scaled_outputData_test.shape)
```

Prilog 5.2. WaveNet 1D CNN model za predviđanje koncentracije čestica u zraku

Linija ***Kod***

```
1:  def build_wavenet_model(inputShape,outputShape,numOfDilatonLayers,
    numOfFilters,kernelSize, dillKernelSize):
2:      model = Sequential()
3:      # Prvi konvolucijski sloj i sloj sažimanja
4:      model.add(Conv1D(filters=numOfFilters, kernel_size= kernelSize,
        activation='relu', input_shape=inputShape))
5:      model.add(MaxPooling1D(pool_size=2))
6:      # Višestruko slaganje dilatacijskih slojeva
7:      dilation_rates = [2**i for i in range(numOfDilatonLayers)]
8:      for dilation_rate in dilation_rates:
9:          model.add(Conv1D(filters=numOfFilters,kernel_size=dillKernelSize,
            activation='relu',dilation_rate=dilation_rate, padding='causal'))
10:     # Zadnji slojevi
11:     model.add(Flatten())
12:     model.add(Dense(128, activation='relu'))
13:     model.add(Dense(outputShape, activation='linear'))
14:     return model
```

Prilog 5.3. Primjer testiranja WaveNet 1D CNN arhitekture modela za predviđanje koncentracije čestica u zraku

Linija Kod

```
1:  inputShape = (scaled_inputData_train.shape[1],1)
2:  outputShape = scaled_outputData_train.shape[1]
3:  print(scaled_inputData_train.shape)
4:  print(scaled_outputData_train.shape)
5:  numOfEpochs = 150 #Umetnuti željeni broj epoha za trening modela
6:  batchSize = 24 #Umetnuti željeni batch size
7:  dilationLayers =[7,8,9] #Umetnuti željene brojeve dilatacijskih slojeva
8:  filterSizes =[160,204,256,512] #Umetnuti željene veličine filtera
    patineceForCallback = 15 #Ranije zaustavljanje treninga modela
9:                        #(koliko epoha strpljenja za isti rezultat)
10: kernelSize = 3 #Umetnuti željenu veličinu filtera konvolucijskog sloja
11: dillKSize = 5 # Umetnuti željenu veličinu filtera dilatacijskog sloja
12: optimizers = [] #Umetnuti željene optimizatore)npr. 'adam','sgd'...)
13: for optim in optimizers:
14:     for numOfDilations in dilationLayers:
15:         for fSizes in filterSizes:
16:             print("=======================================")
                print(f"Model with dilation_rate={numOfDilations},
17:             filter_size={fSizes}, dillation_filter_size={dillKSize}")
18:             print("=======================================")
                currentModel = build_wavenet_model(inputShape,
19:             outputShape,numOfDilations,fSizes,kernelSize, dillKSize)
                currentModel.compile(optimizer=optim,
20:                              loss='mean_squared_error')
21:             currentModel.summary()
                early_stopping = EarlyStopping(monitor='val_loss',
22:             patience=patineceForCallback, restore_best_weights=True)
                history = currentModel.fit(scaled_inputData_train,
                scaled_outputData_train, epochs=numOfEpochs,
                batch_size=batchSize,validation_split=0.1, verbose=1,
23:             callbacks=[early_stopping])
                scaled_prediction = currentModel.predict
24:                               (scaled_inputData_test)
                mae = mean_absolute_error(scaled_outputData_test,
25:             scaled_prediction)
                mse = mean_squared_error(scaled_outputData_test,
26:             scaled_prediction)
27:             rmse = np.sqrt(mse)
28:             r2 = r2_score(scaled_outputData_test,scaled_prediction)
29:             print(f"Mean Absolute Error (MAE): {mae:.4f}")
30:             print(f"Mean Squared Error (MSE): {mse:.4f}")
31:             print(f"Root Mean Squared Error (RMSE): {rmse:.4f}")
32:             print(f"R-squared (R2) Score: {r2:.4f}")
                plt.plot(history.history['loss'],
33:             label='Gubitak trening skupa')
                plt.plot(history.history['val_loss'],
34:             label='Gubitak validacijskog skupa')
35:             plt.title('Gubitak modela')
36:             plt.xlabel('Broj epoha')
37:             plt.ylabel('Gubitak')
38:             plt.legend()
39:             plt.show()
```

Prilog 5.4. Tablica parametara iz kojih su pronađeni najbolji modeli (LSTM, GRU, RF, XGBoost) primjenjujući arhitekture prema [40], [41], [42], [43], [44] (**Podebljan** je najbolji rezultat pojedinog parametra).

LSTM		**GRU**		**RF**		**XGBoost**	
Parametar	**Vrijednost**	**Parametar**	**Vrijednost**	**Parametar**	**Vrijednost**	**Parametar**	**Vrijednost**
Hidden units	32, 64, **128**, 256	Hidden units	32, 64, **128**, 256	Broj stabala	100, 200, **300**, 400	Broj stabala	400, 500, **600**, 700
Optimizer	**adam**, rmsprop, sgd	Optimizer	**adam**, rmsprop, sgd	Maximalna dubina stabla	**None**, 10, 20, 30, 40	Maximalna dubina stabla	5, 6, 7, **8**, 9, 10
Learning rate	**0.001**, 0.0001	Learning rate	**0.001**, 0.0001	Minimalnoo za podjelu	**2**, 5, 10	Stopa učenja	0.01, **0.05**, 0.1
Dropout Layers	1, 2, **3,** 4, 5	Dropout Layers	0, 1, 2, **3**	Minimalno listova	**1**, 2, 4	Gamma	**0**, 0.1, 0.2
LSTM Layers	2, 3, 4, **5**, 6	GRU Layers	2, 3, 4, **5**, 6	Broj značajki pri djeljenju	**sqrt**, log2	Dio podatakaza obuku svakog stabla	**0.8**, 0.9, 1
						Dio značajki za uzorkovanje za svako stablo	0.8, **0.9**, 1
						Minimalna suma težina	1, 2, 3, **4**, 5, 6

ŽIVOTOPIS

Ivan Gudelj, rođen 18.studenog.1999. u Slavonskom Brodu, svoje obrazovanje započinje u Osnovnoj školi Vladimira Nazora u Novom Selu, BiH. Školovanje nastavlja u Srednjoj školi Pere Zečevića u Odžaku, BiH koju završava 2018. godine kao učenik generacije sa vrhunskim uspjehom. Daljnje educiranje obavlja na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku, smjer „Računarstvo“. Preddiplomski studiji je uspješno završio 2021.godine sa završnim radom na temu: „Vatrozidi i primjer njihove primjene“, nakon čega upisuje diplomski studiji na istom fakultetu, smjer „Robotika i umjetna inteligencija“.

U životopis ubraja širok spektar obavljanih poslova koji obuhvaća implementiranje sigurnosnih mjera i upravljanje mrežama u tvrtci Mlin-Majić, obavljanje više stručnih praksi u tvrtkama Ericsson Nikola Tesla, Barrage kao razvojni inžinjer te Atos, gdje i započinje svoju karijeru kao inžinjer internetske sigurnosti.

Osim akademskih obaveza, bavi se odbojkom (17 godina od kojih 10 profesionalno). Predstavljao je reprezentaciju BiH na Balkanijadi 2014. te Hrvatsku odbojkašku Sveučilišnu reprezentaciju dvaput na Europskim Akademskim Igrama (EUSA).

Ivan Gudelj
Potpis autora