

Analiza saobraćaja Srbije upotrebom snimaka video kamera

1st Pavle Glušac

Departman za računarstvo i automatiku

Fakultet tehničkih nauka

Novi Sad, Republika Srbija

glusac.r215.2023@uns.ac.rs

2nd Nevena Radešić

Departman za računarstvo i automatiku

Fakultet tehničkih nauka

Novi Sad, Republika Srbija

radesic.r22.2023@uns.ac.rs

Apstrakt—Protok saobraćaja predstavlja ključnu komponentu svakodnevnog života, kao i značajnog broja ljudskih delatnosti, te je uočavanje šablona u protoku saobraćaja nezaobilazan problem. Koristeći javno dostupne resurse na internetu, izvršili smo sakupljanje slika Takovske ulice u Beogradu, u intervalima od pet minuta, kao i podatke o vremenskim uslovima u zadatim trenucima. Koristeći YOLOv5 (*You Only Look Once*) pretrenirani model za detekciju objekata, izvršili smo brojanje vozila na prikupljenim slikama. Koristeći tako sakupljene podatke, modelovali smo ih *Prophet*, SARIMA (*Seasonal Auto-Regressive Integrated Moving Average*), ETS (*Exponential Smoothing*), SNAIVE (*Seasonal Naive*) i *Random Forest* algoritimima, nakon čega smo vršili njihovo poređenje. Najbolji rezultati postignuti su *Random Forest* algoritmom, pri čemu je postignut MSE od 14.2. Model *Prophet* je takođe vrlo dobro modelovao podatke postigavši metriku MSE od 25.2 na testnom skupu podataka. Analizom prikupljenih podataka primećena je velika pravilnost u podacima, što omogućava modelima da dobro modeluju vremensku seriju.

Ključne reči—saobraćaj, analiza vremenske serije, detekcija, prophet, snaive, ets, arima

I. UVOD

U današnjem ubrzanom tempu života, efikasnost i sigurnost saobraćaja postaju sve važniji aspekti urbanog planiranja i razvoja [1]. S obzirom na konstantan porast broja vozila na putevima, kao i na potrebu za optimizacijom saobraćajnih tokova, analiza saobraćaja postaje ključna komponenta u poboljšanju kvaliteta života u urbanim sredinama. Protok saobraćaja nije samo indikator mobilnosti stanovništva već i značajan faktor koji utiče na ekonomski razvoj, životnu sredinu i opštu sigurnost [2]. Stoga je razumevanje šablona u protoku saobraćaja neophodno za efikasno upravljanje saobraćajnim sistemima.

Cilj ovog projekta jeste analiza saobraćajnih tokova u Srbiji kroz primenu savremenih tehnologija obrade slika i mašinskog učenja. Upotrebom snimaka video kamera sa određenih lokacija i javno dostupnih resursa na internetu, projekat se fokusira na prikupljanje, analizu i modelovanje saobraćajnih podataka. Primarni cilj je identifikacija šablona u protoku saobraćaja i njihova korelacija sa vremenskim uslovima, sa ciljem unapređenja saobraćajnih sistema i planiranja infrastrukture.

Prethodna istraživanja na ovu temu su se uglavnom fokusirala na analizu saobraćaja u velikim urbanim centrima, koristeći tradicionalne metode prikupljanja podataka kao što su senzori na putu ili ankete. Međutim, ove metode su često skupocene, vremenski zahtevne i ne pružaju uvek sveobuhvatne podatke o saobraćajnim tokovima. Nasuprot tome, korišćenje video snimaka kao izvora podataka, u kombinaciji sa algoritmima mašinskog učenja, predstavlja inovativan pristup koji omogućava precizniju i efikasniju analizu.

U ovom radu predstavljamo metodologiju koja koristi YOLOv5 pretrenirani model za detekciju vozila na video snimcima [3], nakon čega sledi modelovanje saobraćajnih tokova korišćenjem više statističkih metoda i algoritama mašinskog učenja, uključujući *Prophet* [4], SARIMA [5], ETS [6], SNAIVE [7] i *Random Forest* [8]. Kroz upoređivanje različitih modela, projekat istražuje najefikasnije metode za predviđanje saobraćajnih tokova.

Ovaj rad pruža detaljan pregled izazova i rešenja povezanih sa analizom saobraćaja u Srbiji, uz poseban naglasak na doprinos upotrebe naprednih tehnika obrade slika i mašinskog učenja. U nastavku dokumenta, u poglavlju II, biće predstavljen kratak pregled literature sa već postojećim rešenjima ovog problema. Nakon toga, u poglavlju III biće opisana metodologija korištena u projektu. U okviru tog poglavlja, najpre će biti opisan proces prikupljanja podataka, zatim proces statičke analize, nakon čega će svaki od gorepomenutih pristupa biti detaljnije objašnjen. Poslednje, IV poglavlje posvećeno je sumarizaciji najbitnijih rezultata rada i daje kratak pregled potencijalnih pravaca budućeg rada.

II. RELEVANTNA LITERATURA

Pri istraživanju srodnih radova, rad koji je najbliži našem je rad [9]. Rad pokriva većinu modela koje i naš rad pokriva a to su ARIMA, ETS, SNAIVE, *Prophet*, kao i ansambl tih modela. Podaci koje modeluju su saobraćajni podaci Bangladeša, tačnije grada Daka. Koriste 66000 sakupljenih podataka kroz 30 dana, što znači da je učestalost sakupljanja podataka znatno niža od našeg pristupa. Još jedna od razlika jeste to što njihovi modeli modeluju više ulica, dok naši samo jednu. Zaključak rada [9] je to da je *Prophet* najbolji model među navedenim u

radu, što je naš rad potvrdio, međutim, mi smo bolji rezultat od *Prophet* modela dobili uvođenjem *Random Forest* modela.

Pristup rada [10] jeste to što se za modelovanje saobraćajnih gužvi koristi isključivo *Random Forest*, i saobraćajne gužve ne modeluju kao vremensku seriju. Obeležja koja su dostupna u podacima rada su:

- *Weather* (sunny, foggy, rainy, snowy)
- *Time* (peak/non-peak)
- *Holiday* (yes/no)
- *Special Condition* (yes/no)
- *Quality of Road* (yes/no)

a ciljna labela im je kategorična i može uzimati jednu od vrednosti *smooth*, *congestion*, *blockage*. Rad [10] je došao do zaključka da su vremenski uslovi značajan faktor pri modelovanju podataka, te se zato i ti podaci sakupljaju u našem radu, međutim *Random Forest* koristimo za regresiju, ne klasifikaciju, i takođe imamo različita obeležja.

Rad [11] bavi se estimacijom gustine saobraćaja korišćenjem YOLO i SORT [12] algoritama. Ideja rada jeste da koriste YOLOv4 koji će istrenirati na CCTV (*Closed-circuit television*) slikama puta na Novom Zelandu kako bi dalje koristili SORT algoritam koji bi vršio *real-time* brojanje vozila vodeći računa o pravcu njihovog kretanja. Naš rad takođe koristi YOLO model za detekciju saobraćajnih učesnika, međutim, opredelili smo se da ne vršimo *real-time* brojanje vozila jer je osnovna ideja rada modelovati saobraćaj, tj. sirove brojeve vozila.

Rad [13] se bavi poređenjem LSTM, *Prophet*, SARIMA i *XGBoost* modela. Skup podataka koji je korišten za rad čini ~ 2000 redova, sa 47 atributa koji obuhvataju senzorske podatke. Podaci su prikupljeni u Sjedinjenim Američkim Državama, što je razlika u odnosu na ovaj rad. Takođe, podaci prikupljeni u [13] su beleženi u intervalima od 15 minuta, dok su naši podaci sakupljeni u intervalima od 5 minuta. Rezultati rada [13] su slični našim rezultatima. Najbolje performanse postiže *XGBoost*, dok kod nas to čini *Random Forest*. Oba modela su specijalizovani za skupove podataka visoke dimenzionalnosti i nelinearnosti, te je očekivano da će imati slične performanse. Nakon *XGBoost* modela, *Prophet* postiže najbolje rezultate, što je i kod nas slučaj. Takođe, SARIMA ne postiže rezultate poput prethodna dva modela.

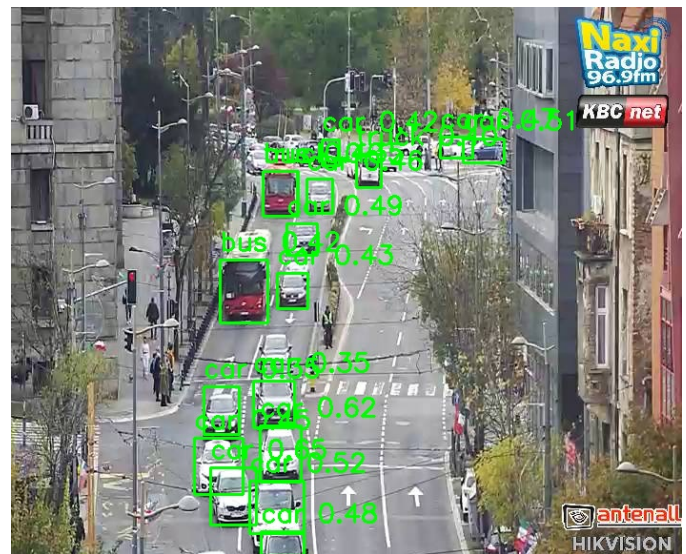
Rad [14] se bavi podacima sakupljenim u Čangša gradu, u Kini. Podaci su sakupljeni u intervalima od 5 minuta, slično našem radu, dok je ukupno vreme sakupljanja trajalo 10 dana, što je znatno manje. Metode koje su izabrali za modelovanje vremenske serije su SARIMA, LSTM i BPNN. Metode dubokog učenja pokazale su se znatno boljim od SARIMA modela. Rad nam ukazuje da se metode dubokog učenja mogu bolje prilagoditi nepravilnostima podataka od metoda klasične statistike.

III. METODOLOGIJA

Metodologija rada zasniva se na tri velike celine - sakupljanje podataka, njihova analiza i upotreba algoritama za njihovo modelovanje. Tekuće poglavlje će obraditi svaku od tih celina, u navedenom redosledu.

A. Sakupljanje podataka

Slike saobraćaja sakupljene su sa AMSS sajta kao i sajta 'Uživo kamere'. Pri izboru izvora podataka, bilo je neophodno manuelno nadzirati snimke kamere, kako njihov kvalitet varira i kako se neke od kamere rotiraju u određenim intervalima. Vodeći se tim kriterijumima, kamere Takovske ulice u Beogradu u najvećoj meri ispunjavaju navedene uslove, te su predmet posmatranja ovog rada. Primer jedne slike (sa već izvršenim detekcijama vozila) dat je na slici 1.

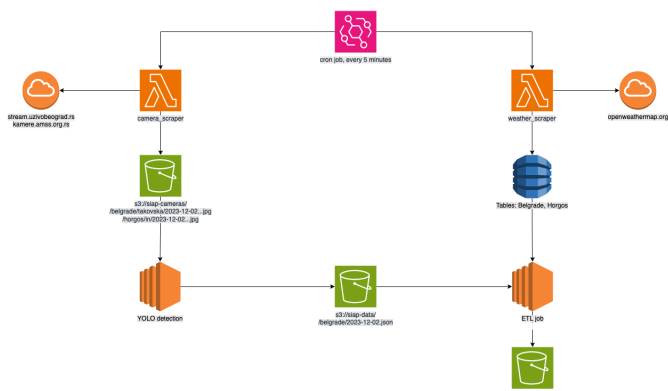


Slika 1. Primer rezultata detekcije

Za potrebe sakupljanja podataka, korišćena je Amazon AWS cloud platforma. Tok podataka započinje AWS *CloudWatch* događajem, koji je definisan kao *cron job* koji se okida na svakih 5 minuta. Taj događaj osluškuju dve lambda funkcije koje se izvršavaju istovremeno, to su lambda funkcija za povlačenje slika i lambda funkcija za prikupljanje informacija o vremenskoj prognozi. Lambda funkcija zadužena za sakupljanje slika kreira HTTP pozive ka odgovarajućim adresama, te rezultate skladišti na predefinisano S3 lokaciju. Detekcija vozila vrši se na EC2 instanci, koja sirove izlaze YOLO algoritma skladišti u vidu JSON formata na narednoj S3 lokaciji. Skripta zadužena za ETL (*Extract, Transform, Load*) operacije ovih sirovih podataka vrši *join* operaciju podacima o vremenskim uslovima koji se nalaze u *DynamoDB* bazi podataka i čuva ih u finalnom S3 sloju, gde su spremni za dalje korišćenje. Podaci o vremenskim uslovima sakupljeni su koristeći 'openweathermap' API, po istom principu. Arhitektura sistema za prikupljanje podataka prikazana je na dijagramu 2.

Rezultujući CSV dokument sadrži sledeće kolone:

Data Pipeline Overview

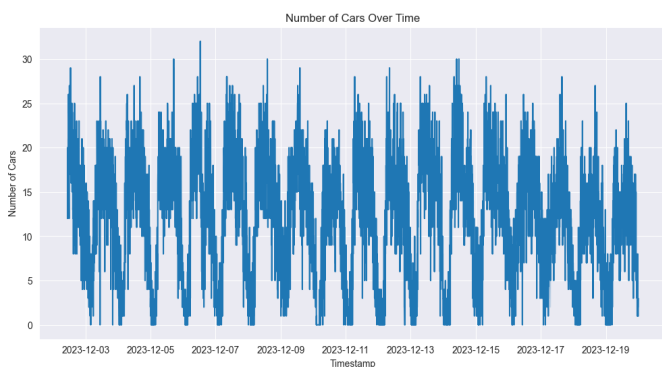


Slika 2. Arhitektura sistema za prikupljanje podataka

- *Index*: redni broj detekcije
- *Timestamp*: vreme prikupljanja podatka u ISO 8601 formatu
- *Detections*: broj detektovanih vozila
- *Weather Main*: vremenski uslovi (snežno, kišovito, oblačno, vedro)
- *Weather Description*: detaljniji opis vremena

Prikupljeni su podaci u intervalu između 02.12.2023. i 15.01.2024, što rezultuje približno 15 000 redova u finalnom CSV dokumentu.

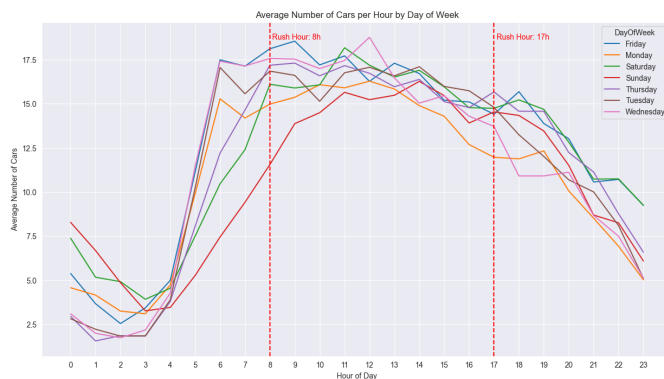
Prikupljeni podaci o broju vozila ilustrovani su na slici 3, gde je primećeno da podaci izražavaju značajnu sezonalnost. Takođe se može primetiti da postoji veća varijabilnost u podacima, što se oslikava u naglim skokovima u predikcijama.



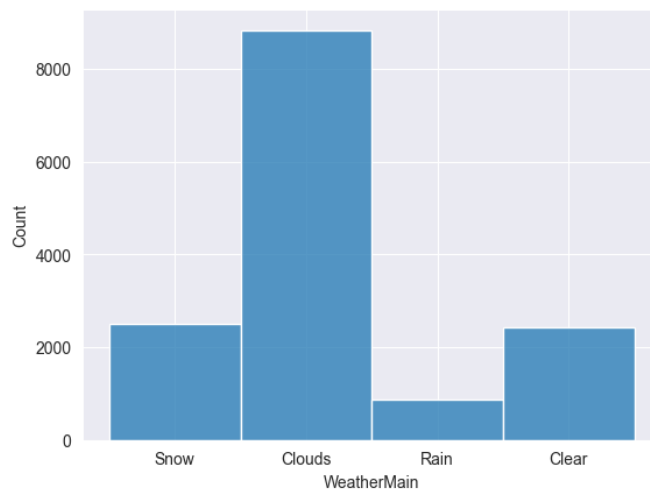
Slika 3. Prikaz broja vozila, od 02.12.2023. do 19.12.2023.

B. Analiza podataka

Sezonalni obrasci saobraćaja po danima u nedelji i satima u danu, prikazani na slici 4, ukazali su na jasno definisane maksimalne vrednosti tokom radnih dana, posebno tokom jutarnjih i popodnevničkih časova. Poređenje ovih obrazaca kroz različite delove dana u nedelji omogućilo nam je da utvrdimo značajne varijacije u saobraćaju koje se mogu povezati sa



Slika 4. Prosečne vrednosti broja detekcija, po danima u nedelji

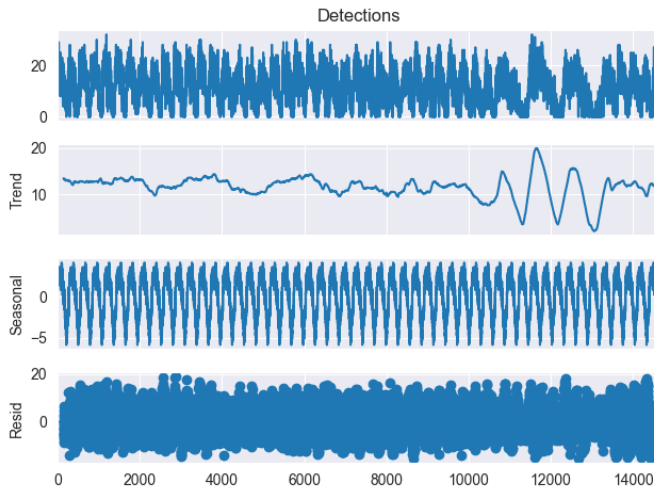


Slika 5. Distribucija vremenskih uslova u podacima

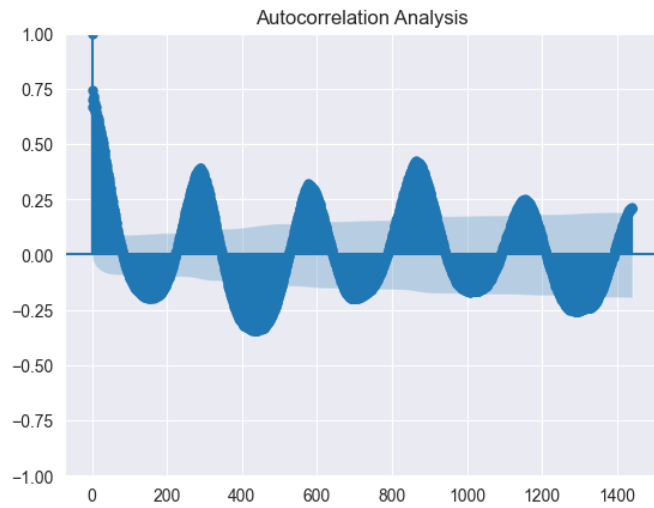
radnim navikama i aktivnostima populacije. Nedeljom postoji primetan pad u broju detektovanih vozila, što je donekle očekivano, pošto je većini građana nedelja neradan dan.

Distribucija vremenskih uslova u podacima pokazuje da je većinu vremena bilo oblačno vreme, daljim analiziranjem primećujemo značajan disbalans među ovim klasama, odnosno da klasi 'Clouds' pripada ~60% obeležja, što vidimo na grafikonu 5. Uzevši u obzir vremenski interval u kom su prikupljeni podaci, očekivana je ova distribucija, međutim, može se postaviti pretpostavka da ovo obeležje neće imati značajan uticaj na samu ciljnu labelu. Takođe, temperatura u prikazanom periodu nije imala značajne fluktuacije, tako da se ne može očekivati značajna korelacija.

Sezonska dekompozicija je statistički postupak kojim se podaci razlažu na tri komponente: trend, sezonsku i iregularnu (rezidualnu). Navedene komponente podataka ovog rada prikazane su na slici 6. Trend komponenta održava skoro konstantnu vrednost, sa fluktuacijama tokom novogodišnjih praznika. Sezonska komponenta ima u potpunosti pravilan šablon, koji pokazuje sezonalnost na dnevnom nivou. Rezidualna komponenta je ukazala na anomalije koje nisu objašnjene



Slika 6. Sezonska dekompozicija



Slika 7. Analiza autokorelacije

ni trendom ni sezonskim ciklusom, i mogu se pripisati šumu.

Analiza autokorelacije, koja je prikazana na slici 7, pokazala je značajnu periodičnost i zavisnost saobraćajnih podataka od prethodnih vremenskih intervala. Zapažena je visoka autokorelacija, što ukazuje na potencijalne obrasce koji se mogu koristiti za predviđanje budućih saobraćajnih tokova korišćenjem *lag* vrednosti.

Analizom matrice korelacije, koja je prikazana na slici 8, primećujemo sledeće:

- Postoji snažna pozitivna korelacija između "Detections" i "Lag_Detections" (0.65), što implicira da prethodne detekcije mogu biti snažan prediktor trenutnih detekcija.
- Umerena pozitivna korelacija između "Hour" i "Lag_Detections" (0.36) sugeriše da postoji veza između sata u toku dana i detekcija, što može ukazivati na dnevne obrasce u detekcijama.



Slika 8. Matrica korelacije

- Varijable "Temperature" i "WeatherMain" imaju veoma nisku korelaciju sa varijablom "Detections", što sugeriše da temperature i opšti vremenski uslovi nemaju značajan direktan uticaj na detekcije.

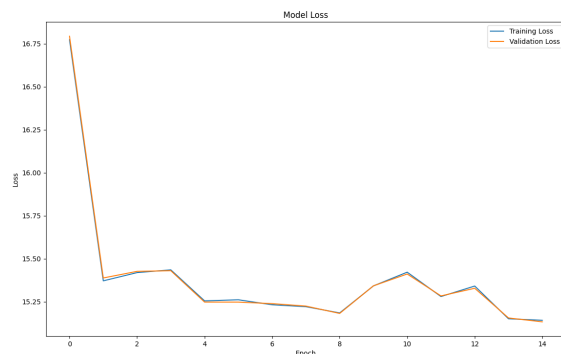
C. Neuronske mreže

Pri modelovanju podataka, iskoristiće se modeli dubokog učenja kako bi se modelovali podatci. Treniraće se tri arhitekture, LSTM (*Long Short-Term Memory*) [15], GRU (*Gated Recurrent Unit*) [16] i hibridna arhitektura ConvGRU (*Convolutional GRU*). Prve dve arhitekture će biti implementirane u Keras radnom okviru, koristeći Keras-tuner biblioteku za optimizaciju hiperparametara, dok će treća arhitektura biti implementirana u PyTorch radnom okviru, koristeći optuna biblioteku za optimizaciju hiperparametara. Keras-tuner koristi Bajesovu optimizaciju hiperparametara, dok optuna koristi TPE (*Tree-structured Parzen Estimator*) [17] za pretragu prostora hiperparametara. Za svaki model, trening-test odnos podataka će iznositi 90:10. Korišćena metrika je *Mean Squared Error*.

LSTM

LSTM arhitektura u većini zadataka mašinskog učenja zahteva veliku količinu podataka kako bi trening bio uspešan. Međutim, imajući na umu da su korišćeni podaci niske kompleksnosti, kao i to da prate vrlo jasan šablon, trening će biti izvršen bez tehnika proširivanja skupa podataka, poput augmentacija i slično. Pri treningu modela, gde je postavka takva da model uzima poslednjih 10 detekcija i vrši predikciju narednih 5, model postiže odličan MSE koji iznosi 15.3 na testnom skupu podataka. Trening i validaciona greška se mogu videti na slici 10.

Hiperparametri koji postižu najbolje performanse su



Slika 9. Treniranje LSTM modela, koji na osnovu prethodnih 10 merenja generiše narednih 5

- learning_rate: 6e-4
- recurrent_layers: 3
- hidden_size: 50

Ukoliko povećamo broj predikcija koje LSTM mora da predvidi, performanse značajno opadaju. U slučaju kad se model pohrani brojem detekcija za čitav dan (288) i zatraži se jednak broj predikcija, MSE opada na ~ 41 .

GRU

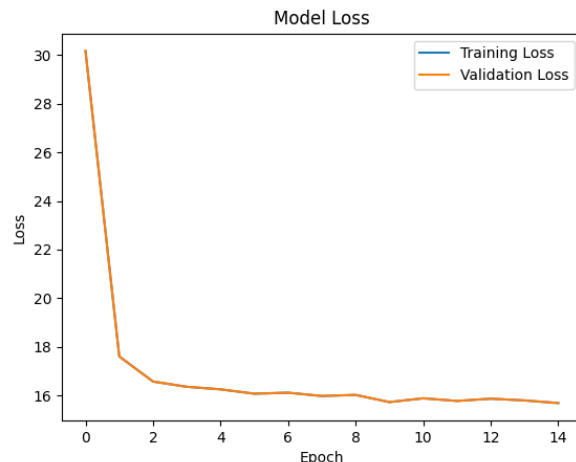
Za razliku od LSTM arhitekture, GRU je pojednostavljena varijanta modela koja sadrži manji broj trening parametara. Takođe, često daje bolje performanse na manjim skupovima podataka, dok suprotno važi za kompleksne i velike skupove podataka. Pri istoj postavci kao za LSTM, GRU postiže MSE od 15.2, a hiperparametri najboljeg modela su:

- learning_rate: 1e-3
- recurrent_layers: 4
- hidden_size: 100, 60, 70, 90

Kao kod LSTM-a, povećavanje broja predikcija značajno degradira performanse opadaju. U slučaju kad se model pohrani brojem detekcija za čitav dan (288) i zatraži se jednak broj predikcija, MSE opada na ~ 40 .

ConvGRU

Kao što je opisano, performanse LSTM i GRU modela značajno degradiraju povećavanjem veličine izlazne sekvence. Razlog zašto se to dešava je sama nepogodnost modela da generiše sekvence veće dužine. Kako bi taj problem izbegao, konstruisana je ConvGRU arhitektura. Na samom ulazu u model, nalazi se konvolutivni sloj koji smanjuje veličinu ulazne sekvence. Odabrano je da veličina kernela bude 24, što odgovara detekcijama u trajanju od dva sata, dok je za stride izabran 12, što je pomeraj kernela po intervalima od sat vremena. S' tim na umu, GRU jedinice barataju daleko kraćim sekvencama koje bi trebalo da sadrže sve relevantne



Slika 10. Treniranje GRU modela, koji na osnovu prethodnih 10 merenja generiše narednih 5

informacije o prethodnim detekcijama. Prikaz opisane arhitekture modela dat je na slici 11.

Treniranje ovakve arhitekture doprinelo je značajno boljim performansama za dugačke sekvence. MSE dobijen iznosi 16.03 na testnom skupu podataka. Hiperparametri najboljeg modela su:

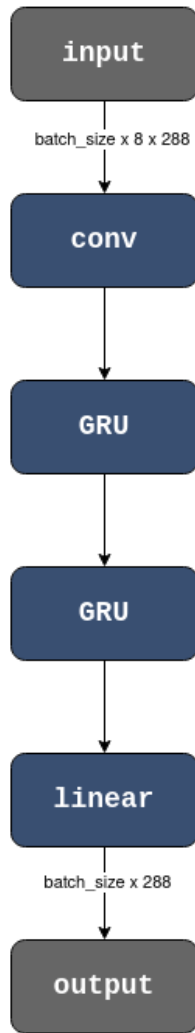
- learning_rate: 7e-4
- recurrent_layers: 3
- hidden_size: 96
- dropout: 0.1

Prikaz MSE pri treningu je prikazan na slici 12. Prikaz inference se može videti na slici 13.

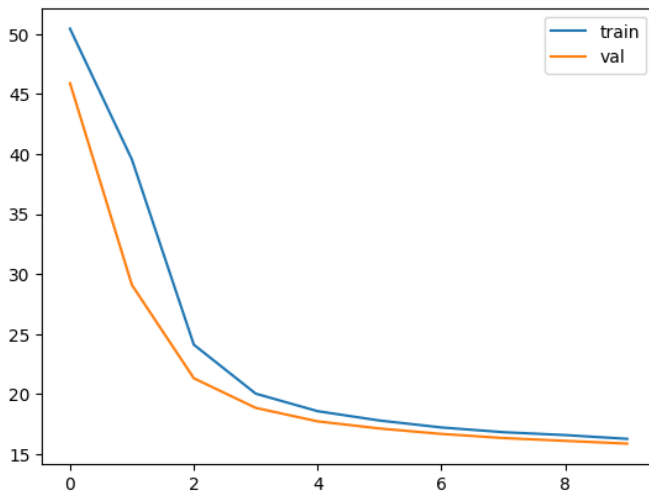
Dodatno, implementirana je još jedna ideja u ConvGRU arhitekturi inspirisana enkoder-dekoder arhitekturama, a to je dodavanje sloja transponovane konvolucije pred poslednji sloj mreže. Taj sloj služi da bi povećao dimenzionalnost nisko-dimenzionog signala sa izlaza rekurentnih jedinica. MSE dobijen ovom modifikacijom na testnom skupu podataka iznosi 18.1.

D. Modelovanje podataka

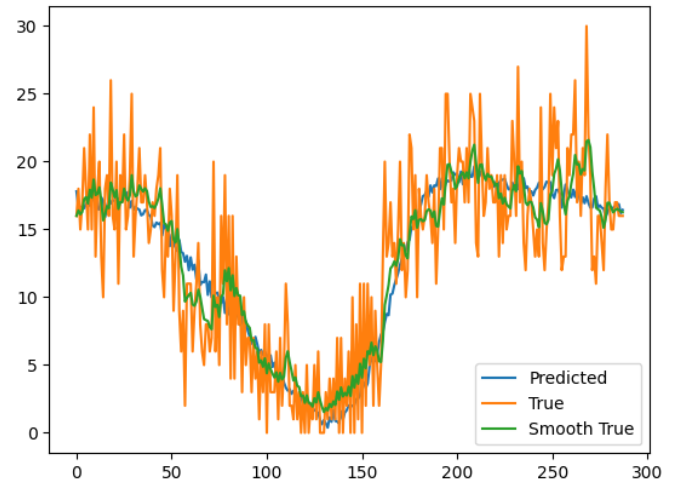
U ovom radu istražujemo efikasnost različitih metoda mašinskog učenja i statističkih modela za predikciju saobraćajnih tokova, koristeći kao kriterijum poređenja srednju kvadratnu grešku (MSE), koja je čest izbor u relevantnoj literaturi. Metode koje smo analizirali su *Random Forest*, *Prophet*, *SARIMA*, *SNAIVE*, i *ETS*. Pri treniranju modela, korišćena je 90:10 podela na trening i testni skup, zbog relativno velikog broja podataka. Za svaki od modela, korišćena je biblioteka *'optuna'* za optimizaciju hiperparametara. *Optuna* koristi *TPE (Tree-structured Parzen Estimator)* [17] za pretragu prostora hiperparametara.



Slika 11. Arhitektura ConvGRU modela



Slika 12. Trening ConvGRU modela



Slika 13. Enter Caption

Random Forest

Random Forest je ansambl metoda koja koristi više stabala odlučivanja za generisanje predikcija. Svako stablo je obučeno na slučajno odabranom podskupu podataka, čime se smanjuje varijansa i povećava robusnost modela. Ovaj algoritam je posebno efikasan u situacijama sa velikim i složenim skupovima podataka, gde tradicionalni pristupi mogu dovesti do preprilagođavanja. U našem istraživanju, *Random Forest* je ostvario MSE od 14.2, što ukazuje na visoku preciznost u predikciji saobraćajnih tokova. Hiperparametri koji su postigli najbolje performanse su:

- *n_estimators*: 70
- *min_samples_split*: 2
- *ccp_alpha*: 0.013
- *max_samples*: 0.5

Prophet

Prophet je metod razvijen od strane kompanije *Facebook* za predviđanje vremenskih serija, koji se posebno odlikuje svojom sposobnošću da efikasno obrađuje podatke sa jasnim trendovima i sezonskim varijacijama. *Prophet* se takođe odlikuje svojom sposobnošću da uzima u obzir praznike i druge specijalne događaje, modelujući ih zasebno. Ovaj model je dizajniran da bude fleksibilan i lako prilagodljiv, ima sposobnost da sam prilagođava parametre na osnovu podataka.

Prophet modeluje vremenske serije $y(t)$ kao kombinaciju tri glavne komponente: trend, sezonalnost, i praznike (*special events*), i formulisan je kroz sledeću aditivnu funkciju datu na jednačini 1.

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t \quad (1)$$

gde je:

- $g(t)$ funkcija trenda koja modeluje nelinearne trendove promene u vremenskoj seriji
- $s(t)$ sezonska komponenta koja pruža fleksibilnost modelu da se prilagodi sezonskim obrascima različitih perioda (na primer, godišnjih, nedeljnih)
- $h(t)$ komponenta koja predstavlja efekte praznika i specijalnih događaja koji mogu imati uticaj na predikciju
- ϵ_t predstavlja neobjašnjivi šum.

Prophet je ostvario MSE od 25.2, što pokazuje njegovu sposobnost da efikasno modeluje složene sezonske obrasce u saobraćajnim podacima. Hiperparametri koji su postigli najbolje performanse su:

- *seasonality_mode*: 'additive'
- *changepoint_prior_scale*: 0.04
- *seasonality_prior_scale*: 3.054

SARIMA

Seasonal AutoRegressive Integrated Moving Average model, poznat kao SARIMA, predstavlja sofisticiranu metodu za modelovanje i predikciju vremenskih serija koje pokazuju sezonske obrasce. Ova metoda integriše elemente autoregresije (AR), diferencijacije (I) radi stabilizacije srednje vrednosti serije, pokretnog proseka (MA), i sezonskih komponenti, omogućavajući time efikasno modelovanje složenih temporalnih struktura. Struktura SARIMA modela može se opisati nizom parametara $(p, d, q) \times (P, D, Q)$ s, gde:

- p označava red autoregresije (AR),
- d označava stepen diferencijacije (I),
- q označava red pokretnog proseka (MA),
- $P, D,$ i Q su sezonski ekvivalenti za AR, I, i MA komponente tim redosledom, dok s predstavlja dužinu sezonskog ciklusa.

SARIMA se odlikuje svojom sposobnošću da modeluje podatke sa jakim sezonskim fluktuacijama, kao i trendovima i cikličnim komponentama. Ovaj model je ostvario MSE od 45, što je dobar, ali svakako značajno lošiji rezultat od prethodna dva modela. Hiperparametri koji su postigli najbolje performanse su:

- p : 1
- d : 0
- q : 1
- P : 0
- D : 0
- Q : 1
- s : 288

SNAIVE

SNAIVE, ili sezonski naivni pristup, bazira se na pretpostavci da će buduće vrednosti biti jednake vrednostima iz istog perioda u prethodnom ciklusu. Na primer, ukoliko je detektovan broj automobila u ponedeljak u 14:00h bio 20, pretpostavka je

da će u utorak u 14:00h biti jednak broj automobila. Formalno, algoritam se zapisuje kao na jednačini 2.

$$\hat{Y}_{t+m} = Y_{t+m-s} \quad (2)$$

gde je t određen trenutak, m korak predikcije unapred, a s dužina ciklusa.

Iako jednostavan, ovaj metod može biti iznenađujuće efikasan za podatke sa snažnim i redovnim sezonskim obrascima. Sa MSE od 70, SNAIVE pruža koristan *benchmark* za upoređivanje sa složenijim metodama. Hiperparametri koji su postigli najbolje performanse su:

- *seasonal_lag*: 288

ETS

ETS (*Error, Trend, Seasonality*) model eksplicitno modeluje tri glavne komponente vremenskih serija: grešku, trend i sezonalnost. Idejno je sličan *Prophet* modelu, međutim, ETS je manje fleksibilan i ne detektuje automatski trendove poput *Prophet* modela. ETS se može prikazati jednačinom 3

$$Y_t = T_t + S_t + \epsilon_t \quad (3)$$

Sa MSE od 73.14, ETS model je pokazao da je manje precizan u našem kontekstu. Hiperparametri koji su postigli najbolje performanse su:

- *trend*: None
- *seasonal*: 'add'
- *seasonal_periods*: 288

IV. REZULTATI I DISKUSIJA

U ovom radu smo se fokusirali na analizu saobraćajnih tokova u Srbiji koristeći savremene tehnologije obrade slika i mašinskog učenja. Projekat je obuhvatio sakupljanje podataka iz javno dostupnih video snimaka, njihovu analizu i modelovanje koristeći algoritme mašinskog učenja i statističke modele kao što su *Random Forest*, *Prophet*, SARIMA, SNAIVE, i ETS. Osnovni cilj bio je identifikacija šablona u protoku saobraćaja i njihova korelacija sa vremenskim uslovima, sa krajnjim ciljem unapređenja saobraćajnih sistema i infrastrukture.

Analiza podataka pokazala je značajnu sezonalnost i varijabilnost u saobraćajnim tokovima, uz izražene dnevne i nedeljne obrasce. Korišćenje Amazon AWS *cloud* platforme omogućilo je efikasno sakupljanje i obradu podataka, dok su statistički postupci poput sezonske dekompozicije i analize autokorelacije ukazali na potencijalne obrasce koji se mogu koristiti za predviđanje budućih saobraćajnih tokova.

Rezultati pokazuju da *Random Forest* metoda pruža najbolje rezultate s obzirom na MSE (*Mean Squared Error*), što ukazuje na njegovu snažnu sposobnost da precizno modeluje složene saobraćajne tokove.

Prophet model, iako nije dostigao preciznost *Random Forest* modela, pokazao je značajnu fleksibilnost u modelovanju sezonskih varijacija i praznika. Njegova sposobnost da automatski detektuje promene u trendu i da se prilagodi atipičnim promenama čini ga izuzetno korisnim za podatke koji pokazuju nepravilne fluktuacije.

SARIMA model, tradicionalno korišćen u statističkoj analizi vremenskih serija, pokazao je solidne rezultate u predikciji. Međutim, zahtevnost u odabiru odgovarajućih parametara i potreba za prethodnom stacionarizacijom podataka mogu predstavljati izazov u modelovanju.

SNAIVE metoda, iako najmanje sofisticirana, pružila je korisne uvide za serije s jasnim i konzistentnim sezonskim obrascima, naglašavajući važnost osnovnih sezonskih komponenti u predikciji.

ETS model je jedan od jednostavnijih modela koji razlaže podatke na osnovne komponente (greška, trend i sezonalnost), i pokazao se kao manje precizan u poređenju sa ostalim modelima.

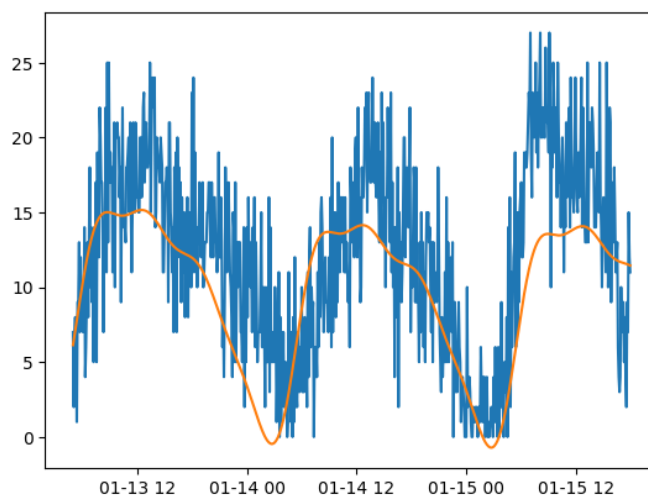
Diskusija ovih rezultata ukazuje na to da izbor metode za predikciju vremenskih serija treba zasnivati na specifičnim karakteristikama podataka i na ciljevima analize. Dok *Random Forest* pruža najveću preciznost u opštem slučaju, *Prophet* nudi značajne prednosti u fleksibilnosti i lakoći korišćenja, pogotovo za podatke sa kompleksnim obrascima. SARIMA i ETS ostaju značajni za tradicionalniju statističku analizu, dok SNAIVE služi kao dobar *benchmark* za proveru osnovnih sezonskih efekata.

Prikazivanjem predikcija *Prophet* modela i *Random Forest* modela na slikama 14 i 15, vidimo da se *Random Forest* bolje prilagodio podacima, što se može pripisati tome što *Random Forest* koristi više varijabli, dok *Prophet* koristi samo *Timestamp*. *Prophet* pruža veoma regularne predikcije, za šta je po dizajnu i predviđen.

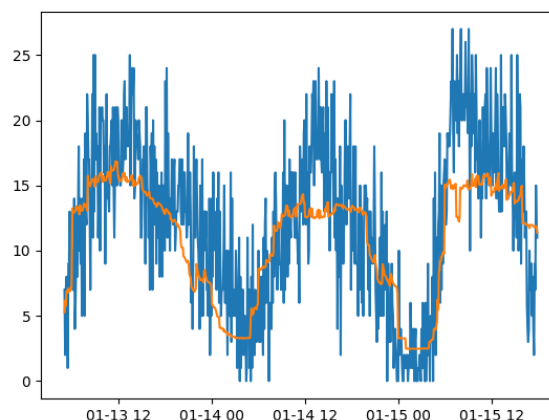
Uočeno je da se greške jednostavno najčešće dešavaju kada se desi značajnije odstupanje broja detektovanih vozila u odnosu na prethodni dan. Na slici 15 uočavamo da je greška najveća na trećoj periodi, gde važi da je broj vozila veći od prethodnih dana. Osim ovoga, nije uočen nijedan drugi šablon po kom se greške dešavaju.

Najveća prepreka pri modelovanju saobraćaja na ovaj način jeste što se pri sakupljanju podataka koristi YOLO model, koji ne vrši savršene predikcije. Kao što se može primetiti na slici 16, model nije detektovao sva vozila i efekat tih grešaka može biti šum koji se pojavljuje u podacima. Manuelnom inspekcijom primetili smo da je broj grešaka koje model napravi u najgorem slučaju ± 4 vozila.

Još jedna od stvari za razmatranje pri modelovanju jeste sam izbor ulice - autori rada [9] navode da izbor ulice značajno utiče na pristup modelovanju. Odabrana ulica, Takovska ulica u Beogradu, je jedna od prometnijih ulica u državi, što znači da možda neće prikazivati toliko nagle fluktuacije koliko bi



Slika 14. Predikcije *Prophet* modela (narandžasto) i tačne vrednosti (plavo)



Slika 15. Predikcije *Random Forest* modela (narandžasto) i tačne vrednosti (plavo)

recimo pokazala slepa ulica sa stambenim zgradama - tu bismo očekivali promene pri dolasku/odlasku stanovnika na posao, dok bismo tokom ostatka dana verovatno imali većinski konstantan broj vozila.

LITERATURA

- [1] Loo, B.P. and Chow, S.Y., 2006. Sustainable urban transportation: Concepts, policies, and methodologies. *Journal of urban planning and development*, 132(2), pp.76-79.
- [2] Levinson, H.S. and Falcocchio, J.C., 2011. Urban development and traffic congestion. In *Transportation and Development Institute Congress 2011: Integrated Transportation and Development for a Better Tomorrow* (pp. 948-956).
- [3] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A., 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).
- [4] Taylor, S.J. and Letham, B., 2018. Forecasting at scale. *The American Statistician*, 72(1), pp.37-45.



Slika 16. Greške YOLO modela pri smanjenoj vidljivosti

- [5] Box, G.E., Jenkins, G.M., Reinsel, G.C. and Ljung, G.M., 2015. Time series analysis: forecasting and control. John Wiley & Sons.
- [6] Hyndman, R., Koehler, A.B., Ord, J.K. and Snyder, R.D., 2008. Forecasting with exponential smoothing: the state space approach. Springer Science & Business Media.
- [7] Hyndman, R.J. and Athanasopoulos, G., 2018. Forecasting: principles and practice. OTexts.
- [8] Breiman, L., 2001. Random forests. Machine learning, 45, pp.5-32.
- [9] Shuvo, M.A.R., Zubair, M., Purnota, A.T., Hossain, S. and Hossain, M.I., 2021, January. Traffic forecasting using time-series analysis. In 2021 6th International Conference on Inventive Computation Technologies (ICICT) (pp. 269-274). IEEE.
- [10] Liu, Y. and Wu, H., 2017, December. Prediction of road traffic congestion based on random forest. In 2017 10th International Symposium on Computational Intelligence and Design (ISCID) (Vol. 2, pp. 361-364). IEEE.
- [11] Algiriyage, N., Prasanna, R., Stock, K., Hudson-Doyle, E., Johnston, D., Punchihewa, M. and Jayawardhana, S., 2021, May. Towards Real-time Traffic Flow Estimation using YOLO and SORT from Surveillance Video Footage. In ISCRAM (pp. 40-48).
- [12] Bewley, A., Ge, Z., Ott, L., Ramos, F. and Upcroft, B., 2016, September. Simple online and realtime tracking. In 2016 IEEE international conference on image processing (ICIP) (pp. 3464-3468). IEEE.
- [13] Mavani, M., Chaturvedi, P. and Manekar, S., 2023, December. Comparative Analysis of Time Series Models for Traffic Flow Prediction. In 2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON) (Vol. 10, pp. 788-794). IEEE.
- [14] Zheng, J. and Huang, M., 2020. Traffic flow forecast through time series analysis based on deep learning. IEEE Access, 8, pp.82562-82570.
- [15] Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. Neural computation, 9(8), pp.1735-1780.
- [16] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. and Bengio, Y., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.
- [17] Watanabe, S., 2023. Tree-structured Parzen estimator: Understanding its algorithm components and their roles for better empirical performance. arXiv preprint arXiv:2304.11127.