

Primjena nadziranog strojnog učenja za povećanje energetske učinkovitosti

Robert Domgjonaj
Informacijski i poslovni sustavi
Fakultet organizacije i informatike

22. siječnja 2026.

1 Uvod

U ovom radu razmatra se primjena algoritama nadziranog strojnog učenja u svrhu povećanja energetske učinkovitosti i smanjenja negativnog utjecaja na okoliš. Poseban naglasak stavljen je na predviđanje potrošnje i proizvodnje električne energije u sustavima s obnovljivim izvorima energije.

Motivacija za ovaj rad proizlazi iz trenutačne globalne situacije nastale ljudskim djelovanjem na klimu planeta Zemlje kao i znatiželja oko mogućnosti razvijanja modela strojnog učenja za energetske učinkovitost.

Cilj rada je analizirati primjenjivost različitih algoritama strojnog učenja te procijeniti njihovu učinkovitost na stvarnom skupu podataka vezanih uz potrošnju energije.

1.1 Korišteni algoritmi strojnog učenja

U radu se razmatraju sljedeći algoritmi:

- Linearna regresija,
- kNN algoritam,
- Gradient Boosting regresija.

2 Algoritmi strojnog učenja

2.1 Linearna regresija

Svatko ima želju predvidjeti budućnost no nema svatko alate za tu namjeru. Jedan od alata koji nam može pomoći u predviđanju onog što će se dogoditi u budućnosti na temelju podataka jest linearna regresija. Linearna regresija pretpostavlja kako je zavisnost između dvije varijable linearna tj. povećanje i smanjenje jedne varijable utječe na smanjenje ili povećanje druge varijable. Na primjer kada bi htjeli znati da li unos proteina utječe na povećanje mišićne mase ili da li korištenje obnovljivih izvora energije utječe na smanjenje stakleničkih plinova u atmosferi koristili bi linearnu regresiju za modeliranje tih zavisnosti.

Formalno rečeno, kod regresije, hipoteza h preslikava primjere u numeričke oznake, odnosno

$$h : \mathbb{R}^n \rightarrow \mathbb{R},$$

što znači da hipoteza preslikava elemente iz ulaznog prostora (prostora primjera) u realne numeričke vrijednosti. Kod linearne regresije, izlaz je linearna kombinacija ulaznih varijabli [3]. Linearna regresija se modelira formulom

$$Y \approx \beta_0 + \beta_1 X$$

u kojoj je Y zavisna varijabla, a X nezavisna varijabla. β_0 i β_1 su dvije nepoznate konstante koje predstavljaju odsječak na osi i nagib u linearnom modelu. [2] Kada bi htjeli predvidjeti korištenje obnovljivih izvora energije u odnosu na stakleničke plinove u atmosferi, to bi mogli prikazati ovako

$$S \approx \beta_0 + \beta_1 O$$

gdje je S - staklenički plinovi u atmosferi, a O obnovljivi izvori energije.

Kako modeli regresije predviđaju na određenom skupu podataka korisno je imati funkciju gubitka koja nam govori koliko je određen model pogriješio u predviđanju za zadani uzorak, a to je u stvari razlika između predviđene vrijednosti modela i stvarne vrijednosti za taj uzorak. [5]. Funkcija gubitka trebala bi odražavati cilj koji model pokušava postići. Na primjer, kod regresijskih problema naš je cilj minimizirati razlike između predviđanja i ciljanih vrijednosti, dok je kod klasifikacije naš cilj minimizirati broj pogrešaka u klasifikaciji. [5] Kod regresije se za funkciju gubitka (označenu s L) najčešće koristi kvadratno odstupanje između ciljane i predviđene vrijednosti:

$$L(y, h(x)) = (y - h(x))^2.$$

U statistici se razlika između ciljane vrijednosti i predviđene vrijednosti, odnosno izraz $y - h(x)$, naziva *rezidual*. Prema tome, može se reći da je funkcija gubitka jednaka kvadratu reziduala. [3]

2.2 kNN algoritam

kNN(kNearestNeighbours ili kNajbližihSusjeda) algoritam se razlikuje od linearne regresije po tome što ne modelira funkciju predviđanja već nam govori u koju skupinu možemo klasificirati pojedini podatak na temelju blizine sa drugim podacima. [4]

Kada bismo morali prenijeti ovaj algoritam na ljudsko razmišljanje on je poprilično intuitivan. Mi ljudi svakodnevno klasificiramo objekte u zajedničke skupine [6], primjerice ako odemo na utakmicu Dinamo Zagreb - Hajduk Split grupirat ćemo ljude za plavim majicama u dinamovce, a one sa bijelim u hajdukovce, iako možda ljudi koji imaju plave majice navijaju za Hajduk Split i obrnuto.

Postoji puno načina na koji se može odrediti kojoj će skupini određeni podatak pripasti, a najčešće je to Euklidijska udaljenost [4]. Metrike koje nam kNN algoritam iznese nam mogu dati do znanja koliko nam povijesni podaci dobro ili loše mogu opisati sadašnjost što nam je od iznimne važnosti za analizu energetske učinkovitosti.

2.3 Gradient Boosting algoritam

Gradient Boosting je algoritam strojnog učenja koji gradi prediktivni model postupno, u više koraka. [8] Na početku se koristi vrlo jednostavan model koji daje grubu procjenu ciljne vrijednosti, ali pritom čini pogreške. Umjesto da se te pogreške zanemare, algoritam ih analizira i u sljedećem koraku gradi novi model čiji je zadatak ispraviti upravo te pogreške prethodnog modela. Svaki sljedeći model uči iz onoga što su prethodni modeli pogriješili i dodaje malu korekciju ukupnoj predikciji. Konačna predikcija dobiva se zbrajanjem doprinosa svih tih jednostavnih modela, čime se postupno poboljšava točnost. Na taj način Gradient Boosting uspijeva opisati i složenije, nelinearne odnose u podacima, iako se sastoji od velikog broja pojedinačno slabih modela.

Ono što Gradient Boosting algoritam u stvari radi jest da iterativno smanjuje funkciju gubitka (eng. loss function). To se događa na ovaj način. U početku započinjemo sa početnim modelom, najčešće stablom odlučivanja, čije su početne predikcije nasumice generirane. Nakon toga računamo rezidualnu grešku, razliku između stvarne i predviđene vrijednosti te potom provodimo postupak regularizacije. Postupak regularizacije uključuje smanjivanje utjecaja svakog novog slabog modela koji se dodaje skupu modela, ona se provodi kako bi spriječili prenaučenosti modela (eng. overfitting) odnosno kako se model ne bi previše prilagodio podacima. Nakon regularizacije treniramo sljedeći tako da su rezidualne pogreške iz prethodnog koraka ciljne vrijednosti za treniranje novog modela. U ovoj fazi procjenjuje se učinkovitost ažuriranog

skupa, uključujući novonaučeni model, najčešće korištenjem zasebnog testnog skupa podataka. Ako su rezultati na tom skupu zadovoljavajući, novi slabi model dodaje se skupu; u suprotnom može biti potrebno prilagoditi hiperparametre modela. Ovaj proces se iterativno ponavlja dok nije ostvaren kriterij kojeg definiramo u početku, kao što je dosezanje maksimalnog broja iteracija, postizanje željene točnosti ili ako dodatne iteracije više ne donose značajno poboljšanje. [8]

3 Podaci i pretprocesiranje

3.1 Opis skupa podataka

Skup podataka korišten u radu sadrži povijesne zapise potrošnje i proizvodnje električne energije, meteorološke podatke te vremenske značajke. Podaci sadrže metrike kao što su potrošnja energije i proizvodnja energije te su one već bile normalizirane u početku. Nadalje, ostale ključne varijable za analizu su količina svjetlosti i vjetera te temperatura koji nam mogu dati bitne naznake o prediktivnom modelu.

3.2 Pretprocesiranje podataka

Kako su podaci najbitniji dio modela nadziranog strojnog učenja bitno je točno pripremiti podatke kako bismo osigurali uspješnost i pouzdanost modela. U ovom radu pretprocesiranje uključuje normalizaciju podataka, obradu vremenskih značajki, korekciju pogrešnih zapisa te pripremu podataka za učenje modela.

Skup podataka sadrži velik broj numeričkih varijabli različitih raspona i mjernih jedinica, poput temperature zraka, insolacije, statističkih mjera potrošnje i vremenskih kodiranih značajki. Kako bi se osiguralo da niti jedna varijabla ne dominira procesom učenja zbog svoje numeričke skale, provedena je standardizacija ulaznih značajki.

Standardizacija se provodi transformacijom podataka na način da svaka značajka ima srednju vrijednost jednaku nuli i standardnu devijaciju jednaku jedan. [1] Ovaj postupak je osobito važan za algoritme temeljene na udaljenosti, poput modela k-najbližih susjeda (kNN), gdje neujednačeni rasponi značajki mogu značajno narušiti izračun sličnosti. Iako algoritmi poput Gradient Boostinga nisu osjetljivi na skalu podataka, provedba standardizacije omogućuje dosljednu i usporedivu primjenu svih modela. Standardizacija je omogućena pomoću Standard Scaler iz biblioteke SciKit-Learn.

Podaci o potrošnji i proizvodnji električne energije imaju obilježja vremenskih serija, pri čemu su prisutni dnevni, tjedni i sezonski obrasci. Modeliranje pomoću vremenskih serija je korištenje algoritama strojnog učenja i statističkih metoda za analiziranje podatkovnih elemenata koji se mijenjaju kroz vrijeme. [7] Umjesto korištenja diskretnih vremenskih oznaka, vremenske varijable u ovom skupu podataka kodirane su ciklički pomoću sinusnih i kosinusnih funkcija. Takav pristup omogućuje očuvanje kontinuiteta vremenskih ciklusa, primjerice prijelaza između kraja i početka dana ili godine.

Osim eksplicitnih vremenskih varijabli, korištene su i povijesne značajke poput pomaknutih vrijednosti `lag_96` te baznih vrijednosti iz prethodnog razdoblja `baseline_in`. Time se modelima omogućuje učenje ovisnosti trenutne potrošnje o prošlim stanjima sustava, što je značajno za predviđanje ponašanja modela strojnog učenja za energetske sustave.

3.3 Podjela skupa podataka

Podjela skupa podataka omogućuje objektivnu procjenu sposobnosti modela da generalizira na nove, prethodno neviđene podatke. U ovom radu skup podataka podijeljen je na skup za učenje (train set) i skup za testiranje (test set), pri čemu je korišten omjer 80 % podataka za učenje i 20 % podataka za testiranje.

Glavna svrha podjele skupa podataka jest razdvajanje procesa učenja i evaluacije modela. Skup za učenje koristi se za prilagodbu parametara modela i učenje odnosa između ulaznih značajki i ciljne varijable, dok se skup za testiranje koristi isključivo za procjenu kvalitete predikcija. Na taj način sprječava se da model bude evaluiran na istim podacima na kojima je treniran, čime se izbjegava optimistična i nerealna procjena njegove uspješnosti.

U kontekstu energetskih sustava, ovakav pristup simulira stvarnu situaciju u kojoj se model trenira na povijesnim podacima, a zatim primjenjuje na buduće vremenske korake, za koje stvarna potrošnja ili proizvodnja još nije poznata.

Prije same podjele, skup podataka razdvojen je na ulazne značajke (X) i ciljnu varijablu (y). Ciljna varijabla predstavlja normaliziranu potrošnju električne energije, dok ulazne značajke obuhvaćaju meteorološke varijable, vremenske značajke, povijesne vrijednosti potrošnje te statističke i diferencijalne značajke.

Nakon razdvajanja, provedena je slučajna podjela zapisa na skup za učenje i skup za testiranje. Slučajnost podjele određena je početnim sjemenjem (eng. seed) generatora slučajnih brojeva, čime je osigurana ponovljivost rezultata. Važan aspekt podjele skupa podataka jest redoslijed provedbe skaliranja značajki. Standardizacija ulaznih podataka provedena je isključivo na

skupu za učenje, dok je skup za testiranje transformiran koristeći iste parametre (srednju vrijednost i standardnu devijaciju) dobivene iz skupa za učenje. Ovakav pristup sprječava tzv. curenje informacija (engl. data leakage), pri kojem bi informacije iz testnog skupa neizravno utjecale na proces učenja. [3]

Time se osigurava realistična evaluacija modela, budući da u stvarnoj primjeni model nema pristup statističkim karakteristikama budućih podataka u trenutku učenja.

U kontekstu predviđanja energetske potrošnje, pravilna podjela skupa podataka ima posebno značenje. Energetski sustavi podložni su sezonalnim i dnevnim varijacijama, te je važno da model nauči opće obrasce ponašanja, a ne da se prilagodi specifičnim povijesnim slučajevima. Skup za testiranje omogućuje provjeru sposobnosti modela da ispravno reagira na različite vremenske i meteorološke uvjete koji nisu bili izravno prisutni tijekom učenja.

Pouzdana generalizacija modela omogućuje preciznije planiranje proizvodnje i potrošnje energije, što smanjuje potrebu za sigurnosnim rezervama i doprinosi povećanju energetske učinkovitosti te smanjenju negativnog utjecaja na okoliš.

4 Implementacija aplikacije

Aplikacija je implementirana u programskom jeziku Python te je koncipirana kao modularni sustav koji omogućuje jasan tijek obrade podataka, treniranje modela strojnog učenja, evaluaciju rezultata i njihovu interpretaciju.

4.1 Arhitektura aplikacije

Implementacija je podijeljena u više logičkih modula, pri čemu svaki modul ima jasno definiranu odgovornost. Glavna ideja arhitekture jest razdvajanje pojedinih faza strojnog učenja, čime se izbjegava spajanje funkcionalnosti i povećava čitljivost koda. Aplikacija se sastoji od sljedećih komponenti:

- učitavanje podataka,
- pretprocesiranje podataka,
- definiranje i treniranje modela,
- evaluacija modela,
- interpretacija rezultata,
- grafički prikaz rezultata.

Centralna ulazna točka aplikacije je datoteka `main.py`, iz koje se pozivaju sve ostale datoteke. Učitavanje podataka implementirano je u zasebnom modulu `data_loader.py`. Modul sadrži funkciju za učitavanje CSV datoteke u strukturu tipa `DataFrame`. Učitani podaci predstavljaju vremenske zapise energetske potrošnje i pripadajućih ulaznih značajki, koji se dalje prosljeđuju u fazu pretprocesiranja. Pretprocesiranje je implementirano u modulu `preprocessing.py`. U ovoj fazi podaci se pripremaju za primjenu algoritama nadziranog strojnog učenja. Pretprocesiranje obuhvaća razdvajanje ciljne varijable od ulaznih značajki, podjelu skupa podataka na skup za učenje i skup za testiranje te standardizaciju ulaznih podataka.

Modeli strojnog učenja definirani su u modulu `models.py`. U radu su implementirana tri algoritma nadziranog strojnog učenja: linearna regresija, model k-najbližih susjeda te Gradient Boosting regresija. Ovi modeli odabrani su kako bi se omogućila usporedba jednostavnog, interpretabilnog modela s naprednijim nelinearnim pristupima.

Linearni regresijski model koristi se kao referentni model te omogućuje analizu izravnih linearnih odnosa između ulaznih značajki i ciljne varijable. Model k-najbližih susjeda temelji se na lokalnoj sličnosti između zapisa te služi za ispitivanje postojanja ponavljajućih obrazaca u podacima. Gradient Boosting model primjenjuje višestruki pristup temeljen na nelinearnim regresijskim stablima, čime se omogućuje modeliranje složenijih odnosa i interakcija između varijabli.

Treniranje modela provodi se u glavnoj datoteci `main.py`, gdje se svaki model trenira isključivo na skupu za učenje. Time se osigurava dosljedan i usporediv proces učenja za sve primijenjene algoritme.

Evaluacija modela implementirana je u modulu `evaluation.py`. Za procjenu kvalitete predikcija korištene su metrike srednje apsolutne pogreške (MAE) i korijena srednje kvadratne pogreške (RMSE). MAE omogućuje procjenu prosječne pogreške predikcije, dok RMSE dodatno penalizira veća odstupanja te pruža uvid u rizik ekstremnih pogrešaka.

Evaluacija se provodi na skupu za testiranje, koji nije korišten tijekom treniranja modela. Time se osigurava objektivna procjena sposobnosti modela da generalizira na nove podatke. Dobivene metrike omogućuju usporedbu performansi različitih algoritama i analizu njihove prikladnosti za primjenu u energetske sustavima.

Interpretacija rezultata predstavlja ključni dio implementacije, budući da omogućuje razumijevanje utjecaja pojedinih varijabli na predikciju energetske potrošnje. Interpretacija linearne regresije provedena je analizom koeficijenata modela, pri čemu su identificirane varijable s najvećim apsolutnim vrijednostima koeficijenata. Ovakav pristup omogućuje uvid u smjer i jačinu utjecaja pojedinih značajki na ciljnu varijablu.

Za Gradient Boosting model korištena je analiza važnosti značajki, koja kvantificira doprinos svake varijable u procesu donošenja odluka modela. Time se identificiraju ključni meteorološki, vremenski i povijesni čimbenici koji imaju najveći utjecaj na predikciju potrošnje energije.

Model k-najbližih susjeda ne omogućuje izravnu interpretaciju u obliku koeficijenata ili važnosti značajki, no njegova primjena omogućuje indirektan uvid u strukturu podataka. Usporedba performansi kNN modela s ostalim modelima pruža informacije o stabilnosti i ponovljivosti obrazaca u energetskom sustavu.

Radi lakše interpretacije i usporedbe modela, implementiran je modul `visualization.py` za grafički prikaz rezultata. Modul omogućuje prikaz usporedbe pogrešaka modela korištenjem stupčastih dijagrama, prikaz važnosti značajki za Gradient Boosting model te usporedbu stvarnih i predviđenih vrijednosti pomoću raspršenih dijagrama.

Grafički prikazi dodatno olakšavaju analizu performansi modela i omogućuju vizualnu potvrdu zaključaka dobivenih numeričkom evaluacijom. Svi grafovi automatski se spremaju u zasebni direktorij.

5 Rezultati i analiza

Na grafu usporedbe pogreške predikcije prikazane su vrijednosti srednje apsolutne pogreške (MAE) i korijena srednje kvadratne pogreške (RMSE) za sva tri modela.

Rezultati pokazuju da linearna regresija ostvaruje najmanju MAE i RMSE vrijednost, što upućuje na to da u prosjeku proizvodi najtočnije predikcije. Također, razlika između MAE i RMSE kod linearnog modela relativno je umjerena, što znači da model nema izražene ekstremne pogreške u pojedinim vremenskim trenucima. Takvo ponašanje posebno je poželjno u energetskim sustavima, gdje su iznenadni veliki promašaji rizični zbog mogućeg preopterećenja mreže.

Model k-najbližih susjeda pokazuje veću pogrešku u odnosu na linearnu regresiju, pri čemu je RMSE znatno veći od MAE. To upućuje na postojanje povremenih velikih odstupanja u predikcijama, odnosno situacija u kojima model ne uspijeva pronaći dovoljno slične povijesne primjere.

Gradient Boosting model ostvaruje nešto lošije rezultate u odnosu na linearnu regresiju, ali pokazuje uravnotežen odnos između MAE i RMSE. Iako prosječna pogreška nije najmanja, ovaj model relativno dobro kontrolira ekstremna odstupanja, što ga čini pouzdanim u situacijama s izraženijim nelinearnim ponašanjem.

Zaključno, usporedba metrika pokazuje da jednostavniji linearni model u

ovom slučaju nadmašuje složenije algoritme, što upućuje na činjenicu da su odnosi u podacima dobro opisani linearnim i gotovo linearnim značajkama.

Graf važnosti značajki za Gradient Boosting model pruža uvid u to koje varijable imaju najveći doprinos u procesu predikcije. Najvažnije značajke uključuju promjene insolacije u vremenu, kratkoročnu varijabilnost potrošnje, sezonske obrasce te meteorološke varijable poput temperature zraka.

Visoka važnost varijable koja opisuje promjenu insolacije ukazuje na to da dinamičke promjene vremenskih uvjeta imaju veći utjecaj od njihovih apsolutnih vrijednosti. Drugim riječima, energetska potrošnja reagira na nagle promjene u okolišu nego na stabilna stanja. Slično tome, velika važnost kratkoročne standardne devijacije potrošnje potvrđuje da nestabilnost i fluktuacije u potrošnji predstavljaju ključni signal za predikciju budućih vrijednosti.

Sezonske varijable, poput sinusne komponente dana u godini, dodatno naglašavaju postojanje dugoročnih obrazaca povezanih s godišnjim dobima. Meteorološke varijable, prvenstveno temperatura, potvrđuju da klimatski uvjeti imaju značajan, ali ne i isključiv utjecaj na energetska potrošnja. Prisutnost varijable koja opisuje kapacitet sustava među važnijim značajkama pokazuje da model uzima u obzir i strukturne razlike između promatranih energetska jedinica.

Ovi rezultati zajedno ukazuju na to da je energetska potrošnja rezultat kombinacije ponašajnih, vremenskih i okolišnih čimbenika, pri čemu nelinearni model uspješno prepoznaje njihovu međusobnu interakciju.

Raspršeni dijagram koji prikazuje odnos stvarnih i predviđenih vrijednosti za Gradient Boosting model omogućuje vizualnu procjenu kvalitete predikcija. Idealno ponašanje modela odgovaralo bi točkama smještenima uz dijagonalu, koja predstavlja savršeno slaganje predikcije i stvarne vrijednosti.

Na prikazanom grafu vidljivo je da se većina točaka nalazi u blizini dijagonale, što potvrđuje da model u velikom broju slučajeva daje razumno točne predikcije. Ipak, uočljiva su i odstupanja, osobito kod viših vrijednosti potrošnje, gdje model pokazuje tendenciju podcjenjivanja stvarnih vrijednosti. To upućuje na ograničenu sposobnost modela da precizno predvidi ekstremne vršne potrošnje, što je čest problem u energetska sustavima zbog njihove inherentne nepredvidivosti.

Unatoč tim odstupanjima, raspršeni dijagram potvrđuje da Gradient Boosting model uspješno hvata opći trend potrošnje, dok su pogreške uglavnom umjerene i bez izraženih sistematskih pristranosti.

Analiza numeričkih metrika i grafičkih prikaza pokazuje da linearna regresija predstavlja najstabilniji i najprecizniji model u ovom slučaju, dok Gradient Boosting nudi dodatne uvide u nelinearne odnose i važnost pojedinih značajki. Model k-najbližih susjeda služi kao potvrda da energetska sustav

nije dominantno vođen lokalnim sličnostima, već globalnim i ponavljajućim obrascima.

Rezultati jasno upućuju na to da kvalitetno pretprocesirani podaci i pažljivo odabrane značajke mogu omogućiti visoku razinu prediktivne točnosti čak i uz primjenu jednostavnijih modela. Takva predvidivost energetske potrošnje omogućuje učinkovitije planiranje proizvodnje i distribucije energije, smanjuje potrebu za sigurnosnim fosilnim rezervama te time doprinosi povećanju energetske učinkovitosti i smanjenju negativnog utjecaja na okoliš.

6 Zaključak

U ovom seminarskom radu istražena je primjena algoritama nadziranog strojnog učenja u kontekstu povećanja energetske učinkovitosti i smanjenja negativnog utjecaja na okoliš. Polazište rada bila je pretpostavka da točne i pouzdane predikcije energetske potrošnje omogućuju učinkovitije planiranje i upravljanje energetske sustavima, čime se smanjuje potreba za obnovljivim izvorima energije tj. fosilnim gorivima. Za potrebe analize korišten je stvarni skup podataka koji sadrži energetske, meteorološke, vremenske i statističke značajke. Poseban naglasak stavljen je na kvalitetno pretprocesiranje podataka, koje je obuhvatilo standardizaciju numeričkih varijabli, očuvanje vremenske strukture pomoću cikličkog kodiranja te pravilnu podjelu skupa podataka na skup za učenje i skup za testiranje. Takav pristup omogućio je stabilno učenje modela i objektivnu procjenu njihove sposobnosti generalizacije.

U radu su implementirana i uspoređena tri algoritma nadziranog strojnog učenja: linearna regresija, model k-najbližih susjeda i Gradient Boosting regresija. Rezultati evaluacije pokazali su da linearna regresija ostvaruje najmanju prosječnu pogrešku predikcije, što upućuje na to da su odnosi u promatranim podacima u velikoj mjeri linearni i stabilni. Model k-najbližih susjeda ostvario je slabije rezultate jer se oslanja isključivo na sličnost pojedinačnih povijesnih slučajeva. Dobiveni rezultati pokazuju da takav pristup nije dovoljno učinkovit, što znači da se energetska potrošnja ne može pouzdano opisati samo pronalaženjem sličnih situacija u prošlosti, već da slijedi stabilne i ponavljajuće obrasce kroz vrijeme.

Iako Gradient Boosting model nije bio točniji od linearne regresije, on ipak donosi dodatnu vrijednost jer može opisati složenije odnose u podacima. Za razliku od linearne regresije, koja pretpostavlja da svaka varijabla utječe na potrošnju na jednostavan i neovisan način, Gradient Boosting može prepoznati situacije u kojima se učinak jedne varijable mijenja ovisno o vrijednosti druge.

Analiza važnosti značajki pokazala je da model ne promatra meteorološke i vremenske varijable odvojeno, već u kombinaciji. Na primjer, promjena insolacije ima veći značaj u određenim dijelovima godine ili u kombinaciji s temperaturom, što linearni model ne može u potpunosti opisati. Takvi uvidi pomažu u boljem razumijevanju ponašanja energetskeg sustava, iako u ovom konkretnom slučaju ne rezultiraju manjom pogreškom predikcije.

Zaključno, dobiveni rezultati jasno pokazuju da povećanje energetske učinkovitosti ne proizlazi isključivo iz tehnoloških poboljšanja infrastrukture, već i iz kvalitetnog korištenja podataka i analitičkih metoda. Primjena algoritama nadziranog strojnog učenja omogućuje preciznije predviđanje potrošnje energije, smanjenje neizvjesnosti u planiranju te izbjegavanje naglih vršnih opterećenja. Posljedično se smanjuje potreba za hitnim uključivanjem fosilnih izvora energije, čime se doprinosi smanjenju emisija stakleničkih plinova i negativnog utjecaja na okoliš.

Literatura

- [1] GeeksforGeeks, *ref1*, Normalization vs Standardization in Machine Learning, online članak. Dostupno: <https://www.geeksforgeeks.org/machine-learning/normalization-vs-standardization/>, pristupljeno: 21. siječnja 2026.
- [2] G. James, D. Witten, T. Hastie, R. Tibshirani, i J. Taylor, *An Introduction to Statistical Learning: with Applications in Python*, Springer Texts in Statistics, 1st ed., Springer, 2023. ISBN 978-3-031-38748-7.
- [3] J.Šnajder, *Linearna regresija*, predavanje iz kolegija Strojno učenje, Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu, 2022. Dostupno: https://www.fer.unizg.hr/_download/repository/SU1-2022-P03-LinearnaRegresija.pdf, pristupljeno: 20. siječnja 2026.
- [4] Pinecone, *k-Nearest Neighbor (kNN): An Introduction*, online članak. Dostupno: <https://www.pinecone.io/learn/k-nearest-neighbor/>, pristupljeno: 19. siječnja 2026.
- [5] R. Yehoshua, *Loss Functions in Machine Learning: Understand the Most Common Loss Functions and When to Use Each One*, Towards Data Science, 12. svibnja 2023. Dostupno: <https://towardsdatascience.com/loss-functions-in-machine-learning-9977e810ac02/>, pristupljeno: 21. siječnja 2026.

- [6] Z. Boban, *Homo sapiens – the KNN classifier: How our decisions are governed by the KNN approach?*, Towards Data Science, 7. rujna 2022. Dostupno: <https://towardsdatascience.com/homo-sapiens-the-knn-classifier-92261023249a/>, pristupljeno: 21. siječnja 2026.
- [7] IBM, *Time Series Model*, online članak. Dostupno: <https://www.ibm.com/think/topics/time-series-model>, pristupljeno: 21. siječnja 2026.
- [8] B. Clark and F. Lee, *What Is Gradient Boosting?*, IBM Think, online članak. Dostupno: <https://www.ibm.com/think/topics/gradient-boosting>, pristupljeno: 20. siječnja 2026.

ChatGPT je korišten za kreaciju kostura rada te za pretvaranje web-članaka u IEEE referencu.