|  |  |  |
| --- | --- | --- |
| Unigrb | UNIVERZITET U NIŠU  ELEKTRONSKI FAKULTET | logo_1960_4 |

Danica Đorđević

**Topic i named entity recognition**

**Web majning**

Mentor: doc. Dr Miloš Bogdanović

Student: Danica Đorđević 1121

Niš, 2021. god.

Sadržaj

[1. UVOD 3](#_Toc66185120)

[2. DATA MINING 5](#_Toc66185121)

[2.1. Preprocesiranje podataka 6](#_Toc66185122)

[2.1.1. Čišćenje podataka 7](#_Toc66185123)

[2.1.2. Transformacija podataka 9](#_Toc66185124)

[2.1.3. Redukcija podataka 10](#_Toc66185125)

[2.2. Mašinsko učenje 11](#_Toc66185126)

[2.2.1. Nenadgledano učenje 13](#_Toc66185127)

[2.2.2. Nadgledano učenje 16](#_Toc66185128)

[2.2.2.1. Regresija 16](#_Toc66185129)

[2.2.2.2. Klasifikacija 18](#_Toc66185130)

[2.3. Stabla odluke 19](#_Toc66185131)

[2.4. Naivni Bajesov algoritam 25](#_Toc66185132)

[2.5. K-najbliži susedi algoritam 30](#_Toc66185133)

[2.6. Poređenje algoritama 32](#_Toc66185134)

[3. IMPLEMENTACIJA 33](#_Toc66185135)

[4. ZAKLJUČAK 44](#_Toc66185136)

[5. LITERATURA 45](#_Toc66185137)

# UVOD

Pojedinci i organizacije svakodnevno generišu gomilu podataka. Statistika tvrdi da je gotovo 80% postojećih tekstualnih podataka nestrukturirano, što znači da nisu organizovani na unapred definisan način, da se ne mogu pretraživati i gotovo je nemoguće njima upravljati. [1] Nestruktuisan tekst se javlja prilikom interakcija sa klijentima, razmene elektronske pošte, objavljivanja objava na društvenim mrežama, itd. Ručna analiza ovolike količine podataka, u današnje vreme, predstavlja veoma zahtevan, dugotrajan i obiman zadatak. Takođe, ručno sortiranje velikih količina podataka može dovesti do učestanijih grešaka i nedoslednosti.

Preduzeća koja vrše analizu velikih količina podataka, danas koriste sisteme veštačke inteligencije za obavljanje tog posla. Ovakvi sistemi koriste metode tekst majninga (*eng. text mining*), s obzirom da je on presudan za organizaciju, kategorizaciju i uzimanje relevantnih informacija iz sirovih podataka. Ovakvi sistemi imaju sposobnost detekcije teme teksta, kao i prepoznavanja imenovanih entiteta, kako bi se iz teksta izvukle relevantne informacije. Modeli analize tema omogućavaju pregled velike količine podataka i identifikacije najčešćih i najvažnijih tema na lak, brz i potpuno skalabilan način. [2] Pomoću prepoznavanja imenovanih entiteta mogu se izvući ključne informacije da bi se bolje razumelo o čemu se radi u tekstu ili se mogu koristiti za prikupljanje važnih informacija za čuvanje u bazi podataka. [3]

Ovaj rad će se baviti metodom detekcije teme teksta, kao i metodom detekcije imenovanog entiteta, u cilju boljeg razumevanja teksta i ekstrakcije korinih informacija iz istog.

# TEKST MAJNING

Tekst majning, takođe poznat i kao analiza teksta, je proces pretvaranja nestruktuiranog teksta u struktuirane podatke za jednostavnu analizu. Tekst mining koristi tehnike za obradu prirodnog jezika (*eng. Natural Language Processing - NLP*), time omogućavaju mašinama da razumeju ljudski jezik i automatski ga obrade.

Za preduzeća velika količina podataka koja se generiše svakodnevno predstavlja i veliku prednost i izazov. Podaci s jedne strane pomažu kompanijama da steknu pametan uvid u mišljenja ljudi o proizvodu ili usluzi. S druge strane, postoji dilema kako obraditi sve ove podatke. I tu tekst mining igra glavnu ulogu. Kompanije uglavnom koriste tekst majning za automatizaciju mnogih svojih procesa. Transformišući podatke u informacije koje mašine mogu da razumeju, tekst majning automatizuje proces klasifikacije tekstova prema osećanjima, temama i namerama. Zahvaljujući tekst majningu, preduzeća mogu da analiziraju složene i velike skupove podataka na jednostavan, brz i efikasan način. U isto vreme, kompanije koriste ovu moćnu alatku kako bi smanjile neke od svojih ručnih i ponavljajućih zadataka, štedeći svojim timovima dragoceno vreme. Tekst majning algoritam mogao bi pomoći da se prepoznaju najpopularnije teme koje se pojavljuju u komentarima kupaca i način na koji ljudi misle o njima: da li su komentari pozitivni, negativni ili neutralni? Takođe, mogle bi da se otkriju glavne ključne reči koje su kupci naveli u vezi sa datom temom. Ukratko, tekst majning pomaže kompanijama da maksimalno iskoriste svoje podatke, što dovodi do boljih poslovnih odluka zasnovanih na podacima.

Tekst majning postiže ove rezultate uz pomoć mašinskog učenja. Mašinsko učenje je disciplina povezana sa veštačkom inteligencijom, koja se fokusira na kreiranju algoritama koji omogućavaju da mašine nauče zadatke na osnovu primera. Modele mašinskog učenja treba trenirati nad određenim skupom podataka, koji se naziva trening skup. Nakon treninga model će vršiti predikcije sa određenim nivoom preciznosti. [1]

Važno je napraviti poređenje između sledećih pojmova: tekst majning, analiza teksta i analitika teksta. Tekst majning i analiza teksta se često koriste kao sinonimi. Međutim, analitika teksta je malo drugačiji koncept. I tekst majning i analitika teksta imaju za cilj rešavanje istih problema, ali korišćenjem različitih metoda. Oboje nameravaju da reše problem automatske analize sirovih tekstualnih podataka. Tekst majning identifikuje relevantne informacije u tekstu i stoga daje kvalitativne rezultate. Analitika teksta, međutim, fokusira se na pronalaženje obrazaca i trendova u velikim skupovima podataka, što rezultira više kvantitativnim rezultatima. Analitika teksta se obično koristi za kreiranje grafikona, tabela i drugih vrsta vizuelnih izveštaja. Tekst majning kombinuje pojmove statistike, lingvistike i mašinskog učenja da bi se stvorili modeli koji uče iz trening podataka i mogu predvideti rezultate na osnovu novih informacija, na osnovu njihovog prethodnog iskustva. Analitika teksta, s druge strane, koristi rezultate analiza izvedenih pomoću modela tekst majninga, za stvaranje grafikona i svih vrsta vizualizacija podataka. Izbor pravog pristupa zavisi od toga koja vrsta informacija je dostupna. U većini slučajeva, oba pristupa se kombinuju za svaku analizu, što dovodi do ubedljivijih rezultata. [1]

U nastavku rada će biti obrađenje različite metode i tehnike, koje tekst majning koristi.

# Metode i tehnike u tekst majningu

Postoje različite metode i tehnike koje se koriste u tekst majningu. Metode se mogu podeliti u osnovne i napredne metode. U osnovne metode tekst majninga se ubrajaju:

* **Učestalost reči** - Učestalost reči se može koristiti za identifikovanje termina ili koncepata koji se najčešće ponavljaju u skupu podataka. Pronalaženje najviše pomenutih reči u nestrukturiranom tekstu može biti posebno korisno kada se analiziraju recenzije kupaca, razgovori na društvenim mrežama ili povratne informacije kupaca. Na primer, ako se reči skupo, precijenjeno i precijenjeno često pojavljuju u recenzijama kupaca, to može ukazivati na to da trebate prilagoditi cijene (ili ciljno tržište!).
* **Kolokacija** (*eng. collocation*) - Kolokacija se odnosi na niz reči koji se često pojavljuju jedna blizu druge. Najčešći tipovi kolokacija su bigrami (par reči koje će se verovatno spojiti, poput početka, uštede vremena ili donošenja odluka) i trigrami (kombinacija tri reči, na primer, u neposrednoj blizini ili održati kontakt). Identifikovanje kolokacija - i njihovo brojanje kao jednu jedinu reč - poboljšava granularnost teksta, omogućava bolje razumevanje njegove semantičke strukture i, na kraju, dovodi do tačnijih rezultata rudarenja tekstom.
* **Konkordancija** (*eng. concordance*) - Konkordancijase koristi za prepoznavanje određenog konteksta ili instance u kojoj se pojavljuje reč ili skup reči. Ljudski jezik može biti dvosmislen, što znači da se ista reč može koristiti u mnogo različitih konteksta. Konkordancijamože pomoći u razumevanju tačnog značenja reči na osnovu konteksta.

Napredne metode tekst majninga uključuju klasifikaciju teksta. Klasifikacija teksta je postupak dodeljivanja kategorija ili oznaga (*eng. tags*) nestruktuiranim tekstualnim podacima. Ovaj osnovni zadatak obrade prirodnog jezika olakšava organizovanje i strukturiranje složenog teksta, pretvarajući ga u značajne podatke. Zahvaljujući klasifikaciji teksta, preduzeća mogu da analiziraju sve vrste informacija i na brz i isplativ način dobiju dragocene uvide. Najvažniji zadaci klasifikacije teksta predstavljaju:

* **Analizu tema** – pomaže pri razumevanju glavnih tema teksta i organizovanju tekstualnih podataka (na primer, podaci sa označenim temama se mogu klasifikovati po njima),
* **Analizu sentimenta** – bavi se analizom emocija koje leže u osnovi bilo kog teksta. Analiza sentimenata pomaže pri razumevanju mišljenja i emocija u tekstu i njegovoj klasifikaciji kao pozitivog, negativnog ili neutralnog. Ova analiza se može koristiti pri analizi objava na društvenim mrežama, pregleda recenzija, i mnogo toga,
* **Detekciju korišćenog jezika** – omogućava klasifikaciju teksta na osnovu jezika koji je u tekstu korišćen. Jedna od korisnih primena jeste aplikacija koja automatski preosledjuje tikete pravom geografskom pogručju,
* **Detekcija namera** – omogućava klasifikovanje teksta prema namere koja stoji iza njega ili svrhe iza teksta. Ovo omogućuje kompanijama da lakše prepoznaju potencijalne klijente koji su zainteresovani za njihove usluge.

Tehnika koja se često primenjuje kod tekst majninga jeste ekstrakcija teksta. Ekstrakcija teksta je tehnika analize teksta koja iz teksta izdvaja određene podatake, kao što su ključne reči, imena entiteta, adrese, e-mail adrese itd. Korišćenjem ekstrakcije teksta kompanije mogu izbeći ručno sortiranje podataka da bi izvukle ključne informacije. Najčešće, može biti najefikasnije kombinovati izdvajanje teksta i klasifikaciju teksta u istoj analizi. Glavni zadaci tehnike ekstrakcije teksta su:

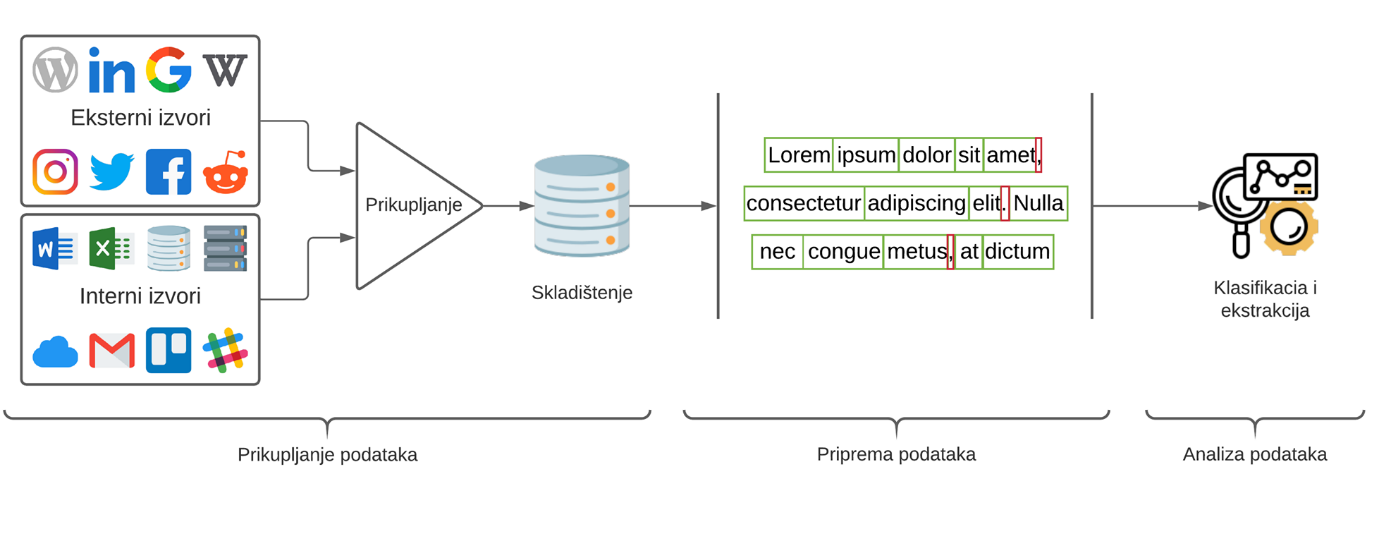
* **Izdvajanje klučnih reči** – omogućava indeksiranje podataka koji se pretražuju, rezimiranje sadržaja teksta ili označivanje teksta (tagovanje), s obzirom da ključne reči predstavljaju najrelevantnije pojmove u tekstu,
* **Prepoznavanje imenovanih entiteta** – omogućava identifikaciju i izvlačenje imena kompanije, organizacija, osoba i drugih imenovanih entiteta iz teksta,
* **Izdvajanje karakteristika** - pomaže u identifikovanju specifičnih karakteristika proizvoda ili usluga u skupu podataka. Na primer, ako se analiziraju opisi proizvoda, mogle bi da se izvuku karakteristike poput boje, marke, modela itd.

U nastavku će biti više reči o procesu tekst majninga.

# Proces tekst majninga

Prilikom procesa tekst majninga podaci prolaze kroz tri glavna koraka:

* **Prikupljanje podataka** – ovaj korak uključuje prikupljanje internih (razmenjene poruke, razmenjena elektronska pošta, ankete, itd.) i/ili eksternih (objave sa društvenih mreža, novinski članci ili tekst sa bilo koje Web stranice) podataka i njihovo skladištenje,
* **Priprema podataka** – prilikom rukovanja tekstom, primenjuju se tehnike poput tokenizacije, lematizacije, uklanjanja stop reči, stemovanja, kako bi se reči pripremile i propustile kroz model mašinskog učenja,
* **Analiza podataka** – primene klasifikacije i ekstrakcije teksta.



Slika 1. Proces tekst majninga

Na slici 1 je prikazan proces tekst majninga. U nastavku će biti obrađeno prepoznavanje imenovanih entiteta i detekcija tema.

# Detekcija tema

Detekcija tema je tehnika mašinskog učenja koja organizuje i razume velike kolekcije tekstualnih podataka, dodeljivanjem oznaka ili kategorija prema svakoj pojedinačnoj temi. Detekcija tema koristi obradu prirodnog jezika za razgradnju ljudskog jezika tako da može pronaći uzorke i otključati semantičke strukture u tekstovima kako bi izvukla uvide i pomogla u donošenju odluka na osnovu podataka. Dva najčešća pristupa za analizu tema uz mašinsko učenje su NLP modeliranje tema i NLP klasifikacija tema.

Modeliranje tema je tehnika mašinskog učenja bez nadzora. To znači da može da izvodi obrasce i grupiše slične izraze bez potrebe da prethodno definiše oznake teme ili obuči podatke. Ova vrsta algoritma se može primeniti brzo i lako, ali postoji i mana - prilično su netačni.

Klasifikacija teksta ili izdvajanje teme iz teksta, s druge strane, mora da zna teme teksta pre započinjanja analize, jer morate da označite podatke da biste obučili klasifikator teme. Iako je u pitanju dodatni korak, klasifikatori tema se dugoročno isplaćuju i mnogo su precizniji od tehnika klasterizacije.

S obzirom da je u praktičnoj implementaciji projekta korišćeno modeliranje tema, ono će biti dalje obrađeno u ovom radu.

# Modeliranje tema

Modeliranje tema rešava sledeću vrstu problema: imate skup tekstualnih dokumenata i želite da saznate različite teme koje oni pokrivaju i grupišete ih prema tim temama. Način na koji ovi algoritmi rade pretpostavlja da je svaki dokument sastavljen od mešavine tema, a zatim pokušava da otkrije koliko je jaka prisutnost svake teme u datom dokumentu. To se postiže grupisanjem dokumenata na osnovu reči koje sadrže i uočavanjem korelacije između njih. Ovaj rad će obraditi rad najčešće korišćenog algoritma: LSA.

# Latentna semantička analiza

Latentna semantička analiza je tradicionalna metoda za modeliranje tema. Zasnovana je na principu koji se naziva hipoteza o distribucij. Hipoteza o distibuciji nalaže da reči i izrazi koji se javljaju u sličnim delovima teksta imaju slična značenja. Algoritam je zasnovan na frekvencijama reči u skupu podataka. Opšta ideja je da za svaku reč u svakom dokumentu izbroji učestalost te reči i da se grupišu dokumenti koji imaju visoke frekvencije istih reči. Učestalost reči ili pojma u dokumentu je broj koji pokazuje koliko se često reč pojavljuje u dokumentu. Učestalost se može izračunati jednostavnim brojanjem - ako se reč mačka pojavi 10 puta u dokumentu, onda je njena učestalost 10. Ovaj pristup se pokazao pomalo ograničenim, pa se tf-idf obično koristi. Tf-idf uzima u obzir koliko je reč uopšte (u svim dokumentima) u odnosu na to koliko je uobičajena u određenom dokumentu, pa su češće reči rangirane više, jer se smatraju boljim „predstavljanjem“ dokumenta, čak i ako nisu najbrojniji.

Nakon proračuna učestalosti reči, ostaje nam matrica koja ima red za svaku reč i kolonu za svaki dokument. Svaka ćelija je izračunata učestalost za tu reč u tom dokumentu. Ovo je matrica dokumenata; odnosi dokumente na pojmove.

U njemu je skriveno ono što želimo: matrica teme dokumenta i matrica teme pojma, koji povezuju dokumente sa temama i termine sa temama. Ove matrice prikazuju informacije o temama tekstova.

Način na koji se generišu ove matrice je dekompozicijom matrice dokumenta na tri matrice upotrebom tehnike koja se naziva skraćeni SVD. Prvo, dekompozicija pojedinačne vrednosti (SVD) linearni je algebarski algoritam za faktorizaciju matrice u proizvod tri matrice U \* S \* V. Važno je da je srednja matrica S dijagonalna matrica singularnih vrednosti originala matrica. Za LSA, svaka pojedinačna vrednost predstavlja potencijalnu temu.

Skraćeni SVD bira najveće t singularne vrednosti i zadržava prvih t kolona U i prvih t redova V, smanjujući dimenzionalnost prvobitnog razlaganja. t će biti broj tema koje algoritam pronađe, pa je to hiperparametar kojem je potrebno podešavanje. Ideja je da se odaberu najvažnije teme, a U je matrica dokument-tema, a V matrica-tema.

Vektori koji čine ove matrice predstavljaju dokumente izražene temama i pojmove izražene temama; mogu se meriti tehnikama poput kosinusne sličnosti za procenu.

# Prepoznavanje imenovanih entiteta

Prepoznavanje imenovanih entiteta (NER) - takođe se naziva identifikacija entiteta ili izdvajanje entiteta - tehnika je obrade prirodnog jezika (NLP) koja automatski identifikuje imenovane entitete u tekstu i klasifikuje ih u unapred definisane kategorije. Entiteti mogu biti imena ljudi, organizacija, lokacija, vremena, količine, novčane vrednosti, procenti i još mnogo toga.



Pomoću prepoznavanja imenovanih entiteta možete izvući ključne informacije da biste razumeli o čemu se radi u tekstu ili ih jednostavno koristiti za prikupljanje važnih informacija za čuvanje u bazi podataka.

Kada čitamo tekst, prirodno prepoznajemo imenovane entitete kao što su ljudi, vrednosti, lokacije itd. Na primer, u rečenici „Mark Zuckerberg je jedan od osnivača Facebook-a, kompanije iz Sjedinjenih Država“ možemo identifikovati tri vrste entiteta:

„Ličnost“: Mark Zuckerberg

„Kompanija“: Facebook

„Lokacija“: Sjedinjene Države

Međutim, za računare moramo da im pomognemo da prvo prepoznaju entitete kako bi ih mogli kategorisati.

To se postiže mašinskim učenjem i obradom prirodnog jezika (NLP).

NLP proučava strukturu i pravila jezika i stvara inteligentne sisteme sposobne da izvuku značenje iz teksta i govora, dok mašinsko učenje pomaže mašinama da se vremenom uče i poboljšavaju.

Da bi naučio šta je entitet, NER model mora biti u stanju da detektuje reč ili niz reči koje čine entitet (npr. Njujork) i da zna kojoj kategoriji entiteta pripada.

Dakle, prvo moramo da kreiramo kategorije entiteta, kao što su Ime, Lokacija, Događaj, Organizacija itd., I unesemo NER model relevantne podatke o obuci. Zatim, označavanjem uzoraka reči i fraza njihovim odgovarajućim entitetima, na kraju ćete naučiti svoj NER model kako da sam otkriva entitete.

# IMPLEMENTACIJA

U ovom radu je odrađena detekcija tema i odrađeno je prepoznavanje imenovanih entiteta, u cilju razumevanja tekstualnih podataka Donalda Trampa i Hilari Klinton, koji su preuzeti sa Twitera. [4] Podaci su prikupljeni neposredno pre predsedničkih izbora u Americi 2016. godine. Ova analiza teksta pomaže u zapažanju najčešćih tema io soba o kojima su Donald Tramp i Hilari Klinton govorili pred izbore. Za potrebe implementacije su korišćene biblioteke:

* spaCy,
* nltk,
* matplotlib i
* pandas.

SpaCy biblioteka je korišćenja za prepoznavanje imenovanih entiteta jer ima bolje persormanse u odnosu na nltk biblioteku. SpaCy biblioteka ima više oznaka za imenovane entitete u odnosu na nltk biblioteku, te tako preciznije označava tekstualne podatke. Takođe, nltk biblioteka je u nekim primerima pogrešno klasifikovala organizacije kao što su Google, Facebook, itd. SpaCy biblioteka je navedene organizacije označila oznakom ORG, dok je nltk biblioteka ove organizacije označila oznakom PERSON, što prestavlja pogrešnu oznaku (iako obe biblioteke imaju oznake PERSON i ORG).

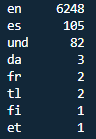
Skup podataka koji je korišćen u ovom projektu ima 8448 redova i 28 atributa. S obzirom da postoji veliki broj nerelevantnih atributa, izbačen je jedan deo njih. Izbačeni atributi su prikazani na slici.



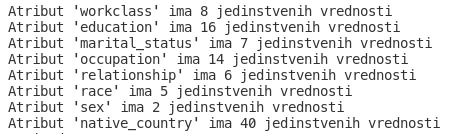
Svaki red sadrži sledeće preostale atribute:

* **id** – atribut koji jedinstveno identifikuje svaki tvit,
* **handle** - atribut koji oyna;ava da li je tvit postavio Donald Tramp ili Hilari Klinton,
* **text** - atribut koji predstavlja tekstualni sadržaj tvita,
* **is\_retweet** - atribut koji označava da li je tvit postavljen od strane neke druge osobe (osobe koja nije Hilari Klinton ili Donald Tramp),
* **original\_author** - atribut koji predstavlja ime osobe koja je postavila tvit, ako je is\_retweet atribut postavljen na *True*, ako je is\_retweet atribut postavljen na *False*, vrednost ovog atributa je prazna,
* **lang** - atribut koji predstavlja jezik u kojem je tvit napisan.

Treba uzeti u obzir činjenicu da Donald Tramp i Hilari Klinton postavljaju tvitove na engleskom jeziku, ali i na drugim jezicima. Na slici se može videti da ima tvitova napisanih na španskom, francuskom, danskom, finskom i ostalim jezicima.



Ovaj projekat je implementiran za potrebe analize teksta napisanom na engleskom jeziku, te tako iz skupa podataka treba izbaciti sve tvitove koji su napisani na nekom drugom jeziku.

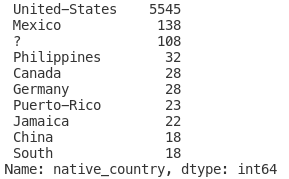


Slika 18. Prikaz broja jedinstvenih vrednosti za svaki od atributa

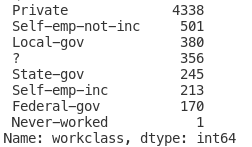
Prvo je izvršeno učitanje dataseta korišćenjem *pandas* biblioteke. Ova biblioteka učitane fajlove predstavlja u obliku DataFrame-a. DataFrame predstavlja tabelarnu reprezentaciju podataka. Podaci imaju dve dimenzije: redove i kolone. Zatim je izvršena podela tog DataFrame-a na dva dela:

* ***X*** - predstavlja redove sa svim karakteristikama, bez kolone za koju se vrši predikcija (bez klasnog atributa) tj. bez *income* kolone,
* ***y*** - predstavlja kolonu za koju se vrši predikcija i sadrži vrednosti kolone *income*.

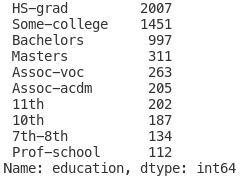
Za dalju obradu podataka, neophodno je da se uoče distribucije vrednosti kako bi se videlo koji atributi se mogu potencijalno modifikovati. Distribucije kategoričkih atributa su date na slikama 19, 20, 21, 22, 23, 24, 25, 26 i 27:



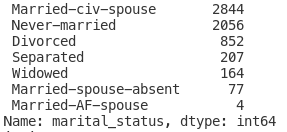
Slika 19. Distribucija vrednosti atributa *native\_country*



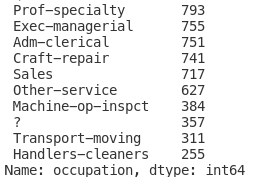
Slika 20. Distribucija vrednosti atributa *workclass*



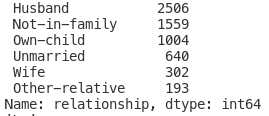
Slika 21. Distribucija vrednosti atributa *education*



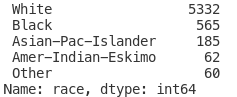
Slika 22. Distribucija vrednosti atributa *marital\_status*



Slika 23. Distribucija vrednosti atributa *occupation*



Slika 24. Distribucija vrednosti atributa *relationship*



Slika 25. Distribucija vrednosti atributa *race*

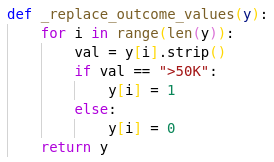


Slika 26. Distribucija vrednosti atributa *sex*



Slika 27. Distribucija vrednosti atributa i*ncome*

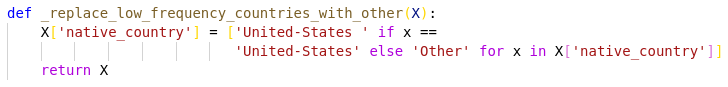
Pošto atribut income ima nenumeričke vrednosti, neophodno je te vrednosti prevesti u numeričke tipa int. Ovo je urađeno u funkciji prikazanoj na sledećoj slici (slika 28):



Slika 28. Prikaz funkcije koja vrši prevođenje tipa klasnog atributa u tip int

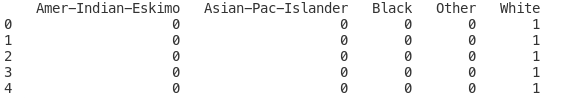
Ova funkcija prolazi kroz niz *y* i vrši zamenu vrednosti “>50K” jedinicom, a vrednosti “<=50K” nulom.

Na slici distribucije atributa *native\_country* se može videti da je većina ljudi (skoro 90%) iz Sredinjenih Američkih Država, pa se modifikacija može uraditi baš na ovom atributu. Cilj je smanjiti broj jedinstvenih vrednosti za ovaj atribut i umesto četrdeset jedinstvenih vrednosti, imati dve. Treba zameniti vrednosti sa niskom frekventnošću pojavljivanja labelom *Other*. Tako da sada postoje samo dve jedinstvene vrednosti za atribut *native\_country*: *Other* i *United-States*. Zamena je izvršena u funkciji prikazanoj na sledećoj slici (slika 29):



Slika 29. Funkcija koja vrši zamenu vrednosti sa niskom frekvencom pojavljivanja labelom *Other*

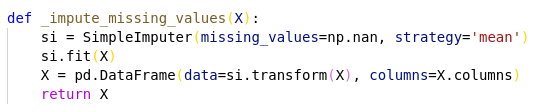
Ovaj dataset poseduje veliki broj kategoričkih atributa tj. atributa koji nemaju numeričku vrednost. Pošto modeli rade samo sa numeričim vrednostima, neophodno je ove atribute prevesti iz nenumeričke u numeričke oblike. Biblioteka *pandas* nudi rešenje za ovaj problem, korišćenjem *dummies* atributa. Ovaj metod za svaku jedinstvenu vrednost koju kolona poseduje, pravi nove kolone, a staru kolonu briše. Tako, na primer, ako se primeni ova metoda na kolonu *race*, dobiće se 5 novih kolona sa binarnim vrednostima, dok će stara *race* kolona biti izbrisana. Primer primenjivanja ove metode na kolonu *race* je dat na slici 30:



Slika 30. Primenjivanje *dummies* metode na kolonu *race*

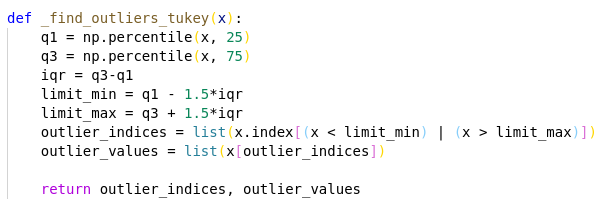
U ovom primeru se mogu videti 5 novih kolona, gde vrednost 1 u koloni označava da pojedinac pripada toj rasi, a vrednost 0 označava da pojedinac ne pripada toj rasi.

Korišćeni dataset ima nedostajućih vrednosti, pa je neophodno popuniti ta polja određenim vrednostima, u zavisnosti od korišćene strategije. Za popunjavanje nedostajućih vrednosti je korišćena strategija srednje vrednosti, što znači da su se nedostajuće vrednosti u kolonama popunjavale srednjom vrednošću tih kolona. Za popunjavanje polja je korišćen *SimpleImputer* iz *sklearn* biblioteke. Ovo je implementirano u funkciji prikazanoj na slici 31:



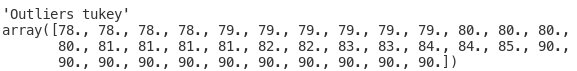
Slika 31. Funkcija koja vrši zamenu nedostajućih vrednosti

Zatim je izvršena detekcija outlier-a za određene kolone. Nije vršeno uklanjanje istih, već samo detekcija. Detekcija je izvršena *tukey* metodom u funkciji prikazanoj na slici 32:



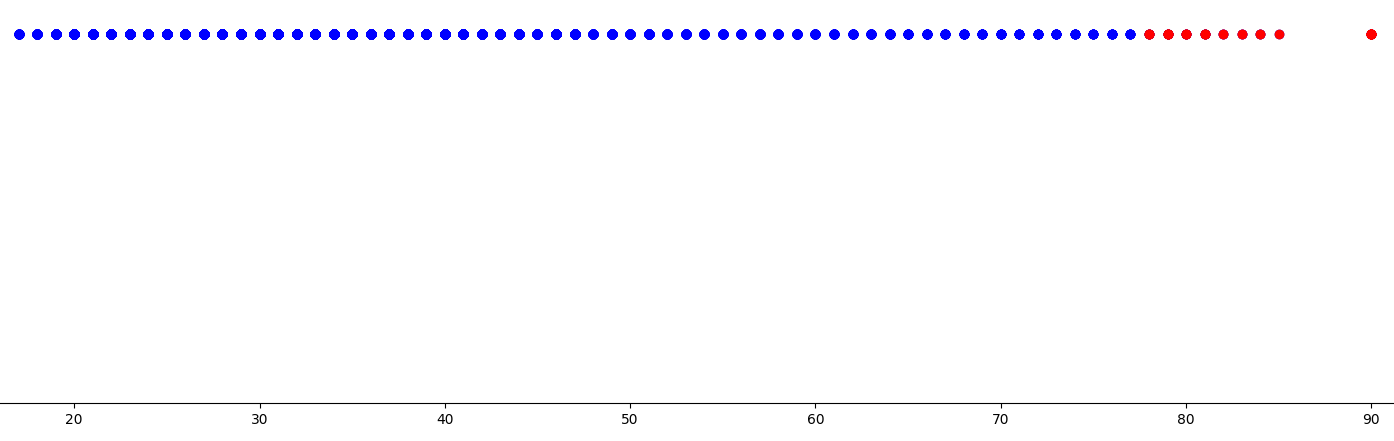
Slika 32. Funkcija koja vrši detekciju outlier-a

Vrednosti outlier-a kolone *age* su prikazani na sledećoj slici (slika 33):



Slika 33. Odstupanja tj. outlier-i za kolonu *age*

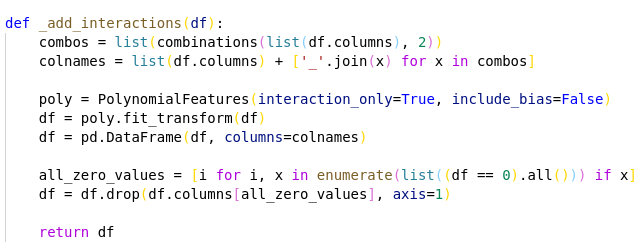
Vizuelni prikaz distribucije vrednosti outlier-a i svih vrednosti kolone *age* su prikazani na slici 34:



Slika 34. Vizuelni prikaz odstupanja vrednosti kolone *age*

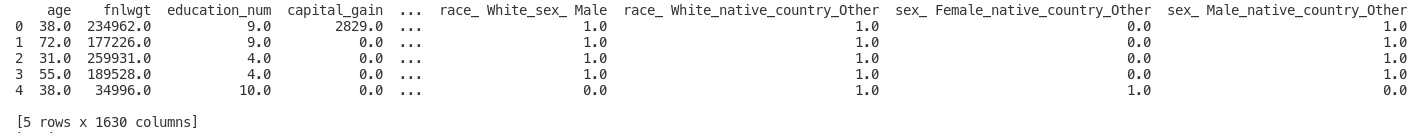
Na ovoj slici se mogu videti vrednosti izvan granica i one su prikazane crvenom bojom, dok su vrednosti koje ne odstupaju od uobičajenih vrednosti prikazane plavom bojom.

Često ulazne karakteristike interaguju na neočekivan i nelineralni način prilikom prediktivnog modeliranja. Te interakcije se modu identifikovati i modelirati pomoću algoritma za učenje. Drugi pristup je osmišljavanje novih karakteristika koje uočavaju ove interakcije i utvrđivanje da li poboljšavaju performanse modela. Pored toga, transformacije poput podizanja ulaznih promenljivih na stepen mogu pomoći u boljem uočavanju važnih odnosa između ulaznih promenljivih i ciljne promenljive tj. promenljive za koju se vrši predikcija. Ove karakteristike nazivaju se interakcije i polinomne karakteristike i omogućavaju upotrebu jednostavnijih algoritama za modeliranje, jer se tumačenja složenih ulaznih promenljivih i njihovih odnosa pomeraju u fazu pripreme podataka. Ponekad ove karakteristike mogu poboljšati performanse modeliranja, iako po cenu dodavanja hiljada ili čak miliona dodatnih ulaznih promenljivih. U implementaciji je izvršeno dodavanje interakcija i polinominalnih karakteristika. To je urađeno u funkciji prikazanoj na sledećoj slici (slika 35):



Slika 35. Funkcija koja vrši dodavanje interakcija

Dodate interakcije znatno povećavaju broj kolona. Prvih 5 redova dataframe-a sa dodatim interakcijama je prikazan na sledećoj slici (slika 36):



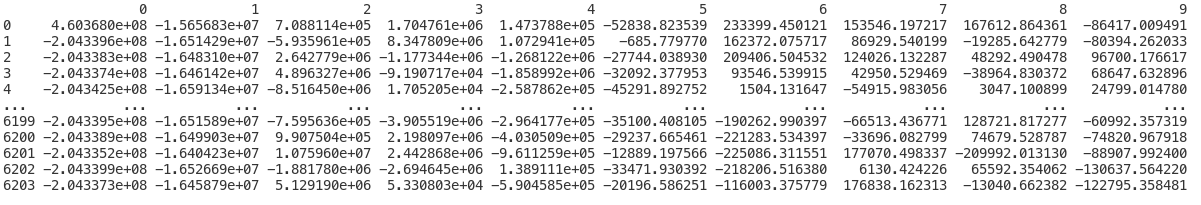
Slika 36. Izgled prvig 5 redova dataframe-a nakon dodavanja interakcija

Na ovoj slici se može videti da dataframe sada ima 1630 kolona, što je znatno više od početnih 15. Procesiranje tolikog broja kolona nije efikasno, pa je neophodno izvršiti i redukciju broja kolona tj. neophodno je izvršiti redukciju dimenzionalnosti. To je može uraditi korišćenjem PCA(Principal Component Analysis) metode iz biblioteke *sklearn*. Prikaz funkcije koja implementira ovu funkcionalnost je dat na sledećoj slici (slika 37):



Slika 37. Implementacija PCA metode

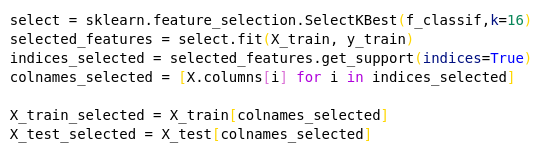
Dataframe nakon redukcije dimenzionalnosti ima 10 kolona, ali te kolone nisu više povezane sa prethodnim, već nove kolone dobijaju nove numeričke nazive. Izgled dataframe-a nakon redukcije dimenzionalnosti korišćenjem PCA metode je prikazan na slici 38:



Slika 38. Izgled kolona nakon redukcije PCA metodom

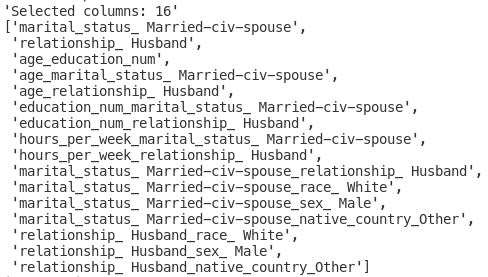
Na slici se može videti da imena novih kolona nemaju nikakvo značenje, a i vrednosti u kolonama je teško razumeti. Zbog teškog tumačenja atributa i njihovih vrednosti, odlučeno je da se u implementaciji ne koristi PCA metoda za redukciju dimenzionalnosti.

Za redukciju dimenzionalnosti je iskorišćena metoda *SelectKBest* iz biblioteke *sklearn*. Ova metoda pronalazi k najrelevantnijih atributa. Rangiranje se vrši korišćenjem *f\_classif* algoritma. Ovaj algoritam je nezavisan od prediktivnog metoda koji se koristi. Ovom metodom su izdvojeni samo 16 najbitnijih kolona. Kod koji vrši selekciju najbitnijih atributa je prikazan na sledećoj slici (slika 39):



Slika 39. Selekcija k najrelevantnijih atributa

Imena selektovanih kolona su prikazana na slici 40:



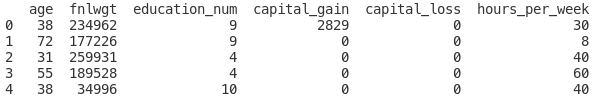
Slika 40. Rezultat izvršene selekcije k najrelevantnijih atributa

Takođe, izvršena je i kros validacija skupa podataka na trening i test skup. Veličina trening skupa predstavlja 90% celokupnog skupa podataka. Kros validacija je izvršena korišćenjem biblioteke *sklearn*, a kod je prikazan na slici 41:



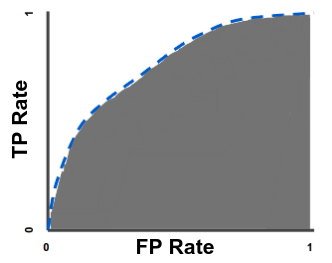
Slika 41. Kros validacija

Ovim je korak preprocesiranja podataka završen. Zatim se prelazi na treniranje modela i upoređivanje rezultata pre i nakon preprocesiranja podataka. Iz skupa neprocesiranih podataka su izbačene kolone sa nenumeričkim podacima, kao i redovi sa nedostajućim vrednostima. Nenumeričke podatke je neophodno izbaciti, da bi primenjivanje bilo kog algoritma bilo moguće. Izgled skupa neprocesiranih podataka je dat na sledećoj slici (slika 42):



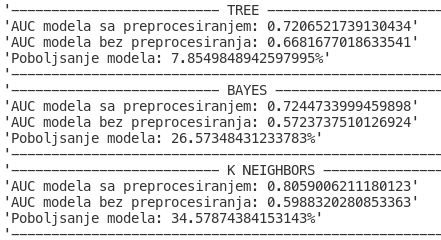
Slika 42. Izgled prvih 5 redova neprocesiranih podataka

Izvršena je klasifikacija korišćenjem tri algoritama. Za evaluaciju algoritama je korišćena AUC (*Area Under the ROC Curve*) mera. AUC meri celokupno dvodimenzionalno područje ispod cele ROC krive od (0,0) do (1,1). Izgled područja ispod ROC krive je prikazano na sledećoj slici (slika 43):



Slika 43. Područje ispod ROC krive

Jedan od načina tumačenja AUC je verovatnoća da model rangira slučajni pozitivni primer više od slučajnog negativnog primera. AUC se kreće u opsegu od 0 do 1. Model čija su predviđanja 100% pogrešna ima AUC od 0,0. Onaj model čija su predviđanja 100% tačna ima AUC 1,0. Rezultati klasifikacije pre i nakon preprocesiranja podataka su prikazani na slici 44:



Slika 44. Performanse sva tri algoritma pre i nakon preprocesiranja

Sa slike se vidi da su performanse preprocesiranih podataka u sva tri slučaja bolje. Najveće poboljšanje performansi je uočeno kod algoritma K najbližih suseda. Kod ovog algoritma je klasifikator takoreći nagađao vrednosti ciljnog atributa (atribut *income*) sa performansama od 0.598. Dok su se nakon preprocesiranja podataka performanse znatno poboljšale na 0.8. Ovi rezultati ukazuju na veliki značaj koji ima preprocesiranje podataka, kao i izbor algoritma za kreiranje modela.

# ZAKLJUČAK

U radu su najpre objašnjeni pojmovi data mining-a i procesa otkrivanja znanja. Detaljnije su opisane tehnike preprocesiranja podataka, kao i pojmovi nadgledano i nenadgledano učenje. Zatim su obrađena i tri algoritma za klasifikaciju. Klasifikacija predstavlja jednu od moćnijih metoda nadgledanog učenja, koja se često koristi u praksi. Na primer, model klasifikacije može se koristiti za identifikovanje podnosilaca zahteva za kredit kao niske, srednje ili visoke kreditne rizike. Različite metode klasifikacije daju različite rezultate. Ne postoji “savršen” model, kao ni model koji je superiorniji u odnosu na ostale. Klasifikacioni modeli se koriste u zavisnosti od reprezentacije podataka koji se koriste, vrste problema koji se rešava, vrste izlaznih rezultata i mnogih drugih faktora. U zavisnosti od tih faktora će modeli imati manju ili veću preciznost u radu. Stoga je dobra praksa sistematično procenjivati skup različitih algoritama kandidata i otkrivati šta dobro ili najbolje deluje na podatke. Poznato je da neki algoritmi rade lošije ako postoje ulazne promenljive koje su irelevantne. Postoje i algoritmi na koje će negativno uticati ako su dve ili više ulaznih promenljivih u visokoj korelaciji. U tim slučajevima možda treba identifikovati i ukloniti nebitne ili visoko korelirane promenljive ili koristiti alternativne algoritme.

Kao veoma važan faktor u procesu klasifikacije, se nalazi i preprocesiranje podataka. Kao što je u radu i dokazano, preprocesiranje može znatno poboljšati performanse modela i u praksi predstavlja veoma važan korak. Kolone skupa podataka mogu imati različite tipove atributa, te tako neke promenljive mogu biti numeričke, kao što su: celi brojevi, vrednosti sa pokretnom zarezom, procenti, itd. Ostale promenljive mogu biti imena, kategorije ili oznake predstavljene znakovima ili rečima, a neke mogu biti binarne, predstavljene sa 0 i 1 ili Tačno i Netačno. Problem je što algoritmi mašinskog učenja u svojoj osnovi rade na numeričkim podacima. Brojevi uzimaju kao ulaz, a broj predviđaju kao izlaz. Svi podaci se vide kao vektori i matrice, koristeći terminologiju iz linearne algebre. Kao takvi, neobrađeni podaci moraju se promeniti pre treninga, procene i upotrebe modela mašinskog učenja. Ponekad promenama podataka može interno upravljati algoritam mašinskog učenja. Čak i ako neobrađeni podaci sadrže samo brojeve, često je potrebna neka priprema podataka.

Postoji međusobna zavisnost između podataka i izbora algoritama. Algoritmi prvenstveno nameću očekivanja podacima, a poštovanje tih očekivanja zahteva da podaci budu na odgovarajući način pripremljeni. Suprotno tome, oblik podataka može pomoći u odabiru algoritama za procenu za koje postoji veća verovatnoća da će biti efikasni.

# LITERATURA

1. <https://monkeylearn.com/text-mining/>
2. <https://monkeylearn.com/topic-analysis/>
3. <https://monkeylearn.com/blog/named-entity-recognition/>
4. <https://www.kaggle.com/benhamner/clinton-trump-tweets/home>