Univerzitet u Nišu

Elektronski fakultet

**Analiza sentimenata na srpskom jeziku**

Procesiranje prirodnih jezika

Seminarski rad

Student: Mentor:

Barbara Mitić, 1639 Dr Suzana Stojković

Niš, 2024.

***Zadatak*:**

Skup podataka sadrži komentare prikupljene sa Booking.com platforme na srpskom jeziku. Analizirati koliko je ovaj skup podataka pogodan za obučavanje klasifikatora za analizu utisaka. Primeniti standardne algoritme klasifikacije, kao što su logistička regresija, Naive Bayes i SVM, ali i naprednije jezičke modele, poput višejezičnog RoBERTa modela.

***Opis skupa podataka*:**

Korišćeni skup podataka sadrži 2095 vrsta i 3 kolone, tj. 2095 prikupljenih komentara čiji su atributi:

* *positive -* pozitivni utisci korisnika o smeštaju
* *negative -* negativni utisci korisnika o smeštaju
* *rating -* ocena smeštaja (od 1 do 10)

Skup podataka nije potpuno kompletan, s obzirom na to da u nekim slučajevima nedostaju pozitivni ili negativni komentari. Iako je to prirodno u kontekstu korisničkih utisaka, predstavlja izazov za analizu sentimenata. Takođe, komentari nisu u svim slučajevima ispravno klasifikovani; na primer, komentar “Nemam nikakvih zamerki” prepoznat je kao negativan, iako izražava pozitivan stav. Zbog ovih nedoslednosti predlaže se objedinjavanje komentara, kako pozitivnih, tako i negativnih, u jedinstveni utisak koji će biti korišćen za dalje analize. Pored toga, u skupu podataka postoje i duplikati koje je potrebno ukloniti. Na posletku, skup podataka je neophodno dopuniti i atributom koji označava sentiment (pozitivan, neutralan ili negativan), a određuje na osnovu ocene.

Još jedan nedostatak u ovom skupu podataka je njegova neizbalansiranost, budući da je broj pozitivnih recenzija delako veći od broja neutralnih ili negativnih recenzija. Ova nebalansiranost može negativno uticati na proces klasifikacije i dodatno otežati analizu sentimenata. U takvim situacijama, modeli mogu favorizovati zastupljenije klase, što može rezultirati netačnim predikcijama one manje zastupljene. Neophodno je primeniti tehnike za balansiranje skupa podataka kako bi se poboljšala preciznost analize.

***Implementacija*:**

*Preprocesiranje podataka*

Problem nedostajućih vrednosti u kolonama za pozitivan ili negativan komentar je rešen jednostavnim objedinjavanjem ovih komentara. Duplikati su uklonjeni, a sentiment određen na osnovuocene (1-3 *negativan*, 4-6 *neutralan*, 7-10 *pozitivan*).

Za balansiranje skupa podataka kombinovane su dve tehnike: *Random Undersampling* i *Random Oversampling* i to u redosledu u kome su navedene*.* Određuje se broj uzoraka za svaku od kategorija (*pozitivan*, *neutralan*, *negativan*). Primećuje se da je broj uzoraka za kategoriju *pozitivan* višestruko veći od ostalih, tako da se najpre nasumičnim izborom uklanja (*undersampling*) veliki broj ovih uzoraka, kako bi sve kategorije imale približno isti broj uzoraka. Zatim se korišćenjem *oversampling* tehnike povećava broj instanci u ostalim (*neutralan, negativan*) kategorijama. Nedostatak ovog pristupa je u tome što se uklanjanjem pozitivnih komentara gubi značajni deo informacija, a oversampling funkcioniše po principu dupliranja nasumično odabranih, postojećih instanci, čime se ne dodaju nikakve nove informacije. Ipak, ukoliko se iskoristi samo *undersampling*, tada će u skupu podataka ostati vrlo mali broj instanci što nikako neće biti dovoljno za treniranje željenih modela i uspešnu analizu utisaka.

Korišćenjem *RandomUnderSampler* i *RandomOverSampler* iz *imblearn* biblioteke, izabrani su uzorci koji će biti korišćeni za dalje obučavanje modela. Rezultujući balansirani podaci su sačuvani u novom *DataFrame*-u:

under\_sampler = RandomUnderSampler(sampling\_strategy={  
'pozitivan': 109  
}, random\_state=42)

X\_under, y\_under = under\_sampler.fit\_resample(X, y)

under\_sampled\_data = pd.DataFrame(X\_under, columns=X.columns)

under\_sampled\_data['category'] = y\_under

over\_sampler = RandomOverSampler(sampling\_strategy='auto', random\_state=42)

X\_balanced, y\_balanced = over\_sampler.fit\_resample(under\_sampled\_data.drop('category', axis=1), under\_sampled\_data['category'])

balanced\_data = pd.DataFrame(X\_balanced, columns=X\_balanced.columns)

balanced\_data['category'] = y\_balanced

*Primena standardnih algoritama klasifikacije*

Metoda *apply\_standard\_classification\_algorithms* služi za primenu standardnih klasifikacionih algoritama, kao što su logistička regresija, Naive Bayes i SVM, na tekstove komentara u korišćenom skupu podataka. Prvi korak podrazumeva preprocesiranje komentara korišćenjem funkcije *preprocess\_comments*, koja pretvara tekst tako da bude napisan isključivo malim slovima, uklanja interpunkcijske znakove i brojeve, čime se priprema tekst za dalju analizu. Ovaj korak je važan zato što smanjuje varijabilnost podataka i poboljšava kvalitet ulaza za modele.

Biblioteka *Stanza* se koristi za *lematizaciju*, tj. za redukciju reči na osnovni oblik (lemu) uz pomoć modela za srpski jezik, budući da ne trenutno ne postoji predefinisani tokenizator za srpski jezik. *Lematizacija* je ključna zato što smanjuje broj jedinstvenih reči u skupu podataka, olakšavajući proces treniranja modela i povećavajući efikasnost. Takođe, uklanjanje uobičajenih reči (stopwords) pomaže da se fokusira na reči koje nose značenje, čime se dodatno poboljšava preciznost klasifikacije.

Nakon preprocesiranja, podaci se dele na podskupove za treniranje i testiranje u odnosu 80-20, što omogućava da se model obučava na jednom delu podataka, dok se njegov učinak procenjuje na drugom, osiguravajući generalizaciju modela. Zatim se koristi *LabelEncoder* za kodiranje ciljne promenljive, olakšavajući rad s kategorijskim podacima.

Nakon kodiranja ciljne promenljive, inicijalizuje se *TF-IDF* vektorizator, koji transformiše tekstualne podatke u numeričke vektore, čime se omogućava mašinskim modelima da bolje razumeju ulaze. Vrši se treniranje tri različita modela: logistička regresija, Naive Bayes i SVM, a zatim svaki od njih koristi za predikciju na testirajućem skupu podataka.

Rezultati su upoređeni sa *Counter* vektorizator koji broji koliko puta se svaka reč pojavljuje u nekom tekstu. Može biti koristan ako su frekvencije reči same po sebi dovoljni pokazatelji značaja reči u odnosu na izlaz (npr. kategoriju). Međutim, može naglasiti česte, ali neinformativne reči, što može negativno uticati na performanse modela.

Na posletku, prikupljaju se rezultati dobijeni za svaki od modela čime se formira zbirni izveštaj koji pruža uvid u učinak različitih klasifikacionih algoritama i pomaže u izboru najefikasnijeg modela za zadatak analize sentimenta.

*Primena višejezičnog XLM - RoBERTa modela*

*XLM-RoBERTa* model je odabran budući da ima podršku za rad sa tekstovima na srpskom jeziku. Metoda *apply\_big\_language\_models* prima jedan ulazni parametar, *data*, koji predstavlja *DataFrame* sa recenzijama, tj sa tekstovima komentara, konačnim ocenama i procenjenim sentimentima. Za obradu tekstualnih podataka je neophodno učitati *XLM-RoBERTa* tokenizator i model za *klasifikaciju sekvenci*. Komentari (iz skupa podataka) se “izvlače” u varijablu X, dok se *kategorije* kodiraju u numeričke vrednosti i smeštaju u varijablu y.

Vrši se podela podataka u podskupove za treniranje i testiranje u odnosu 80-20, što znači da se 80% podataka koristi za obučavanje modela, a 20% za testiranje. Komentari se zatim tokenizuju, što podrazumeva skraćivanje i dodavanje *padding*-a kako bi se omogućila pravilna obrada ulaza.

Definiše se klasa *SentimentDataset*, koja služi za kreiranje dataset-a, omogućavajući lakšu manipulaciju podacima. Zatim se konfigurišu argumenti za obuku, uključujući broj epoha i veličinu batch-a, kako bi se proces učenja optimizovao.

Model se trenira korišćenjem *Trainer* objekta, koji upravlja procesom obučavanja i evaluacije. Kada se treniranje završi, model se evaluira na testnom skupu, a rezultati se prikazuju u obliku različitih metrika, kao što su tačnost (eng. *accuracy*), preciznost (eng. *precision*), odziv (eng. *recall*) i F1 mera (eng. *F1-score*).

***Rezultati*:**

***Standardni algoritmi klasifikacije i nebalansirani skup podataka***

***Standardni algoritmi klasifikacije i balansirani skup podataka***

***XLM-RoBERTa model i nebalansirani skup podataka***

***XLM-RoBERTa model i balansirani skup podataka***

***Zaključak*:**