ANALIZA VELIKIH PODATAKA

školska 2024/2025 godina

Vežba 11: Analiza teksta i osnove NLP

U savremenom digitalnom dobu, tekstualni podaci predstavljaju jedan od najzastupljenijih oblika informacija. Svakog dana se generišu ogromne količine tekstova putem mejlova, društvenih mreža, komentara, recenzija, novinskih članaka, blogova i mnogih drugih izvora. Međutim, većina ovih podataka je **nestrukturisana**, što ih čini teškim za direktnu analizu klasičnim statističkim metodama.

Upravo tu na scenu stupa **NLP – Natural Language Processing**, odnosno obrada prirodnog jezika.

NLP nalazi široku primenu u brojnim oblastima, uključujući:

- 🔎 **Pretragu i ekstrakciju informacija** iz velikih tekstualnih baza (npr. prepoznavanje ključnih fraza u dokumentima)
- 🧠 **Prepoznavanje entiteta** (eng. *Named Entity Recognition*) – identifikacija imena, lokacija, organizacija u tekstu
- 😀 **Analizu sentimenta** – otkrivanje emocionalnog tona izraženog u tekstu (pozitivno, negativno, neutralno)
- 🌐 **Automatsko prevođenje jezika** (npr. Google Translate)
- 🤖 **Generisanje teksta** – chatbotovi, virtuelni asistenti, automatsko pisanje izveštaja, odgovaranje na mejlove

U poslovnom kontekstu, NLP se koristi za:

- Automatizaciju korisničke podrške
- Analizu povratnih informacija korisnika
- Prikupljanje podataka sa društvenih mreža
- Razumevanje tržišta i stavova potrošača

🧻 2. Obrada teksta: predprocesiranje (preprocessing)

Da bismo tekst koristili u mašinskom učenju ili statističkoj analizi, moramo ga transformisati iz sirovog oblika u **strukturisani** i **čist** niz karakteristika. Ovo podrazumeva sledeće korake:

🖌️ 2.1. Čišćenje teksta

Uklanjanje:

- velikih slova (pretvaranje u mala radi ujednačenosti)
- linkova, brojeva, specijalnih znakova
- višestrukih razmaka

➡️ Ovo omogućava bolju obradu u kasnijim fazama.

🔪 2.2. Tokenizacija

Tokenizacija je proces **razbijanja teksta na manje jedinice**, obično reči ili rečenice (tzv. "tokeni").

📌 Primer:

Ulaz: "Ovo je odličan proizvod!"

Tokeni: ["Ovo", "je", "odličan", "proizvod", "!"]

🚫 2.3. Uklanjanje stop-reči

Stop-reči su najčešće reči jezika koje imaju malu informativnu vrednost u analizi sentimenta ili teme (npr. "i", "da", "ali", "je", "u").

➡️ Njihovo uklanjanje smanjuje "šum" i poboljšava fokus modela na ključne reči.

🧬 2.4. Stemming i Lemmatizacija (opciono)

- **Stemming** svodi reči na korenski oblik (npr. “driving” → “drive”)
- **Lemmatizacija** koristi leksičke baze da vrati osnovni oblik reči u kontekstu.

💡 3. Sentiment analiza

Sentiment analiza je poddomen NLP-a koji se bavi klasifikacijom **emocionalnog tona** izraženog u tekstu.

➡️ Na osnovu sadržaja teksta, možemo proceniti da li je:

- **pozitivan** (npr. "Sjajan proizvod!")
- **negativan** (npr. "Potpuno razočaranje.")
- **neutralan** (npr. "Stiglo je juče.")

📊 3.1. VADER (Valence Aware Dictionary and sEntiment Reasoner)

VADER je korisni alat za sentiment analizu razvijen u okviru NLTK biblioteke. Namenjen je **analizi sentimenta u društvenim medijima**, ali je pogodan i za druge tekstualne izvore.

🔍 Karakteristike:

- Ne koristi trenirane modele → zasniva se na rečniku reči sa dodeljenim sentiment vrednostima.
- Dobro detektuje *intenzitet*, *negacije*, *emotikone* i *slang*.
- Brz i jednostavan za primenu.

📈 3.2. Metod rada VADER-a

Za svaki tekst, VADER vraća četiri skora:

Skor	Značenje
pos	Verovatnoća da je tekst pozitivan
neu	Verovatnoća da je neutralan
neg	Verovatnoća da je negativan
compound	Opšti sentiment skor (od -1 do 1)

📌 **Compound skor** se koristi za klasifikaciju:

- compound ≥ 0.05 → **pozitivan**
- compound ≤ -0.05 → **negativan**
- Inače → **neutralan**

🧪 4. Praktične primene sentiment analize

Ova tehnika je posebno korisna kada imamo **velike količine podataka** koje je teško ručno pregledati.

📍 Tipični scenariji primene:

- **Analiza korisničkih recenzija proizvoda i usluga**
  Kompanije koriste sentiment analizu da bi automatski klasifikovale recenzije sa sajtova poput Amazon, Yelp, TripAdvisor ili IMDB, kako bi uočile zadovoljstvo korisnika i eventualne probleme.
- **Praćenje imidža brenda na društvenim mrežama**
  Analiza Twitter objava, Facebook komentara ili Reddit diskusija omogućava kompanijama i političarima da brzo reaguju na promene javnog mišljenja.
- **Automatsko filtriranje toksičnih komentara**
  U online zajednicama, sentiment analiza može pomoći u prepoznavanju i uklanjanju uvredljivih, mrzilačkih ili provokativnih komentara (tzv. trolovanje).

🗃️ 5. Vizualizacija rezultata

Nakon što klasifikujemo tekstove po sentimentu, sledeći korak je **vizualna analiza** kako bismo bolje razumeli obrasce u podacima. Vizualizacije omogućavaju brz uvid u raspodelu mišljenja i najčešće teme koje se pojavljuju.

🔧 Najčešće metode vizualizacije:

- **Broj recenzija po sentiment klasi**
  Histogrami koji prikazuju odnos pozitivnih, negativnih i neutralnih komentara.
- **Bar-plot najčešćih reči**
  Poređenje učestalosti reči u pozitivnim i negativnim recenzijama pomaže da se uoče dominantne teme,
- **Wordcloud (oblak reči)**
  Vizualni prikaz najčešće korišćenih reči, gde je veličina fonta proporcionalna učestalosti reči. Lako uočavamo ključne termine.
- **Sentiment kroz vreme (vremenske serije)**
  Ako imamo vremensku dimenziju, možemo analizirati kako se sentiment menjao tokom meseci ili godina (npr. tokom marketinške kampanje ili nakon skandala).

🧩 6. Ograničenja i izazovi

Iako je sentiment analiza moćan alat, ona ima svoja **ograničenja** i ne treba je koristiti bez razumevanja njenih slabosti.

❗ Glavni izazovi:

- **Sarkazam i ironija**
  „Baš super, pokvarilo se posle 5 minuta.“ – ovakav komentar može zbuniti jednostavne algoritme, jer reč „super“ nosi pozitivnu konotaciju, iako je ton poruke negativan.
- **Nedostatak konteksta**
  Alati poput VADER-a koriste unapred definisana pravila i rečnike, ali ne razumeju širi kontekst rečenice. Ne mogu da "pamte" prethodne rečenice niti prate značenje kroz ceo pasus.
- **Višejezičnost**
  VADER je optimizovan isključivo za **engleski jezik**. Za druge jezike potrebno je koristiti druge alate ili trenirati posebne modele.

Za sofisticiranije zadatke koriste se transformerski modeli kao što su BERT, RoBERTa, DistilBERT, koji koriste kontekstualne reprezentacije reči. Ovi modeli su mnogo precizniji, ali zahtevaju više memorije, vremena za treniranje i tehničkog znanja.

Prilikom izbora metode sentiment analize, važno je uravnotežiti tačnost, brzinu i kompleksnost alata.

PRAKTIČNI PRIMER

📂 1. Uvoz biblioteka i priprema

```
import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import nltk

from nltk.sentiment import SentimentIntensityAnalyzer
```

```
from nltk.corpus import stopwords

from nltk.tokenize import word_tokenize

from collections import Counter

import re

# Preuzimanje potrebnih resursa iz NLTK

nltk.download('vader_lexicon')

nltk.download('punkt')

nltk.download('stopwords')
```

📄 2. Priprema podataka

Ovde koristimo **simulirani dataset recenzija**. U praksi možeš koristiti skup sa Kaggle-a (npr. Amazon, Yelp, IMDB review dataset).

```
# Simulirani skup podataka (možeš zameniti CSV-om sa recenzijama)

data = {

    'review': [

        "I absolutely loved this product! It was fantastic!",

        "This is the worst thing I have ever bought.",

        "Not bad, but could be better.",

        "Excellent service and great quality.",

        "Terrible experience, very disappointed.",

        "Okay product. Nothing special.",

        "I’m really happy with my purchase!",

        "Would not recommend this to anyone.",

        "Best value for money. Highly recommended!",

        "Disgusting. Broke after one use."

    ]}
```

```
df = pd.DataFrame(data)

df.head()
```

🖌️ 3. Čišćenje teksta

```
def clean_text(text):
    # 1. Male slova
    text = text.lower()
    # 2. Uklanjanje linkova, brojeva i spec. znakova
    text = re.sub(r"http\S+|www\S+|[^a-z\s]", "", text)
    # 3. Tokenizacija (deljenje na reči)
    tokens = word_tokenize(text)
    # 4. Uklanjanje stop-reči
    stop_words = set(stopwords.words('english'))
    filtered = [word for word in tokens if word not in stop_words]
    # 5. Spajanje u nazad u tekst
    return " ".join(filtered)

# Primena na sve recenzije
df['cleaned'] = df['review'].apply(clean_text)
df.head()
```

🔍 4. Sentiment analiza pomoću VADER-a

```
# Inicijalizacija VADER analizatora
sia = SentimentIntensityAnalyzer()
```

```
# Dodavanje sentiment skorova
df['scores'] = df['review'].apply(lambda x: sia.polarity_scores(x))
df = pd.concat([df.drop(['scores'], axis=1), df['scores'].apply(pd.Series)],
     axis=1)

# Dodavanje kolone sa klasifikacijom (pozitivno, negativno, neutralno)
def classify_sentiment(compound):
    if compound >= 0.05:
        return 'positive'
    elif compound <= -0.05:
        return 'negative'
    else:
        return 'neutral'

df['sentiment'] = df['compound'].apply(classify_sentiment)
df[['review', 'compound', 'sentiment']]
```

📊 5. Vizualizacija rezultata

```
# Broj recenzija po sentimentu
sns.countplot(x='sentiment', data=df, palette='coolwarm')
plt.title('Distribucija sentimenta u recenzijama')
plt.xlabel('Sentiment')
plt.ylabel('Broj recenzija')
plt.show()
```

📈 6. Analiza reči po sentimentu (ekstra deo)

```
# Grupisanje po sentimentu

positive_words = []

negative_words = []


for i, row in df.iterrows():

    tokens = word_tokenize(row['cleaned'])

    if row['sentiment'] == 'positive':

        positive_words.extend(tokens)

    elif row['sentiment'] == 'negative':

        negative_words.extend(tokens)


# Najčešće reči po sentimentu

pos_freq = Counter(positive_words).most_common(10)

neg_freq = Counter(negative_words).most_common(10)


# Prikaz kao bar-plot

def plot_freq(freq_data, title):

    words, counts = zip(*freq_data)

    plt.figure(figsize=(8,4))

    sns.barplot(x=list(counts), y=list(words), palette='crest')

    plt.title(title)

    plt.xlabel("Frekvencija")

    plt.ylabel("Reč")

    plt.show()

plot_freq(pos_freq, "Najčešće reči u pozitivnim recenzijama")

plot_freq(neg_freq, "Najčešće reči u negativnim recenzijama")
```

Zadatak za samostalni rad:

Na osnovu obrađenih primera i teorije, studenti treba samostalno da primene naučeno na većem, realnom skupu podataka. Preporuka je da se koristi **Amazon Reviews dataset** dostupan na Kaggle-u ili sličan skup tekstualnih recenzija.

1. **Primeni ovu vežbu na realan dataset**
   Preuzmi, pročisti i pripremi podatke za analizu sentimenta.
2. **Izračunaj procenat pozitivnih i negativnih recenzija**
   Uporedi brojnost različitih klasa sentimenta (pozitivno, negativno, neutralno) i predstavi ih numerički i grafički (bar-plot).
3. **Sortiraj recenzije po najsnažnijem pozitivnom i negativnom sentimentu**
   Iskoristi compound skor iz VADER-a da izdvojiš recenzije sa ekstremnim vrednostima (blizu -1 i +1).
4. **Kombinuj sentiment sa dodatnim atributima**
   Ako dataset sadrži i druge informacije, poput **ocena zvezdicama**, uporedi tekstualni sentiment sa numeričkim ocenama. Pokušaj da otkriješ neusklađenosti (npr. pozitivne recenzije sa niskim ocenama i obrnuto).
5. **Vizualizuj dobijene rezultate**
   Napravi barem još jednu jednostavnu vizualizaciju (npr. bar-plot distribucije sentimenta, box-plot sentimenta po oceni, ili wordcloud najčešćih reči u pozitivnim i negativnim recenzijama).