ANALIZA SENTIMENTA

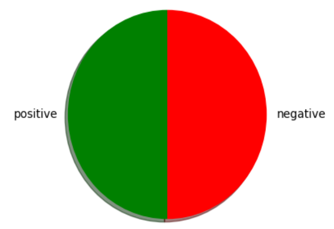
Završni projekat kursa Mašinsko učenje 2023. godine na Matematičkom fakultetu, Univerziteta u Beogradu

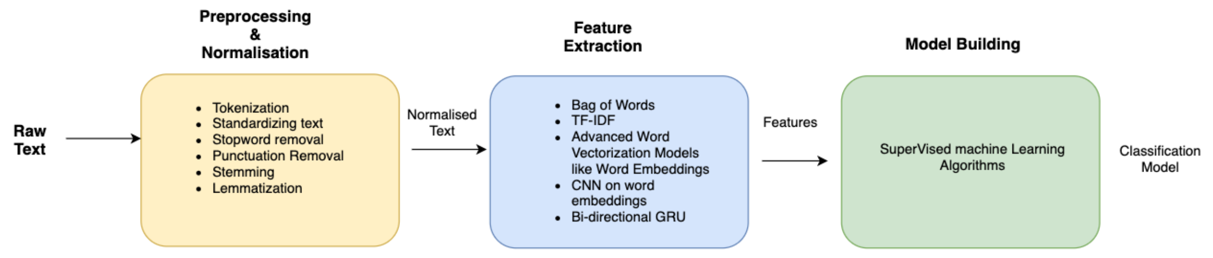
Pavle Savić

1075/2022

SKUP PODATAKA

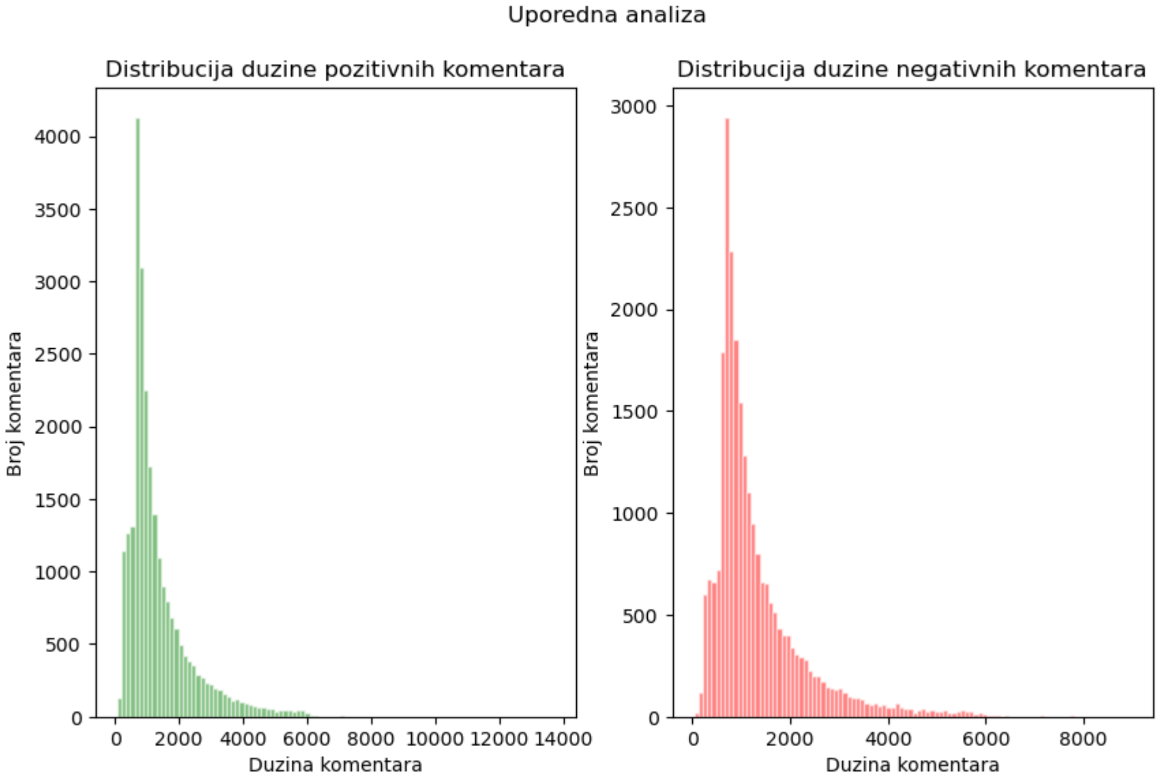
- 50 000 komentara na filmove sa IMDB sajta
- zadaci: osnovna analitika nad sirovim podacima, analiza prirodnog jezika i binarna klasifikacija komentara na pozitivne i negativne

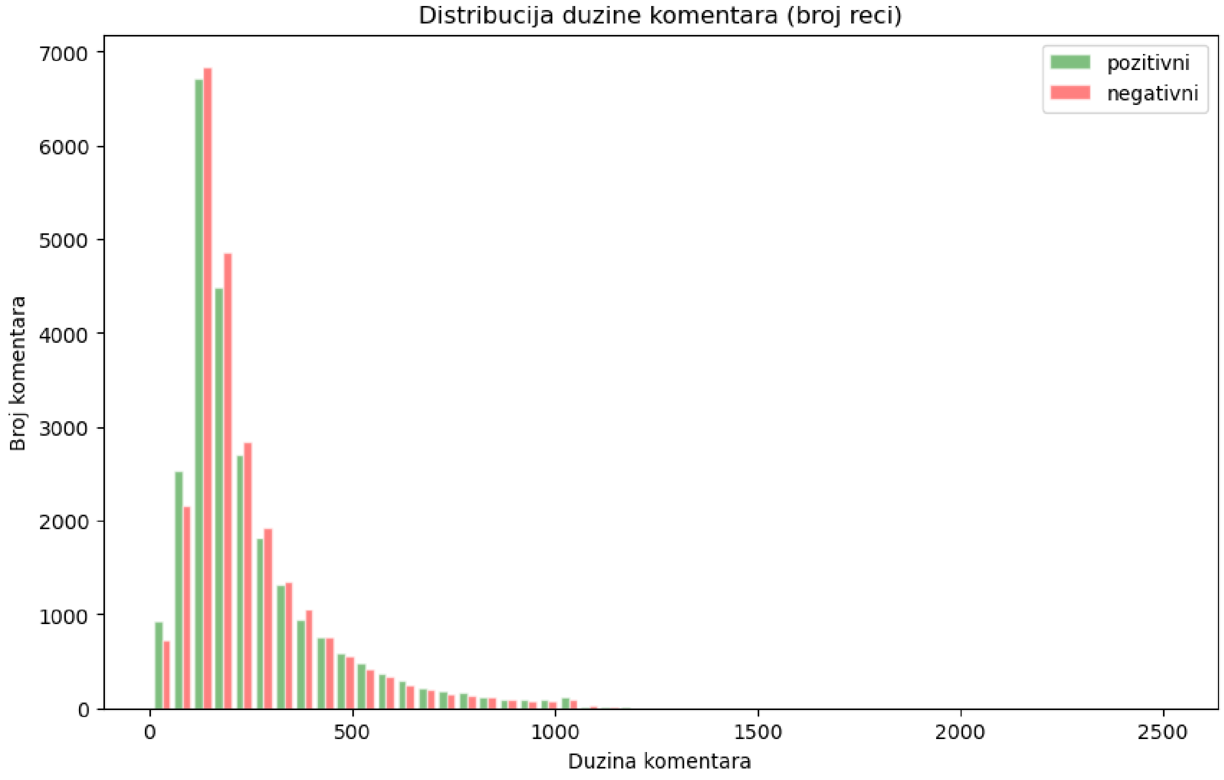




An NLP Model Creation Pipeline

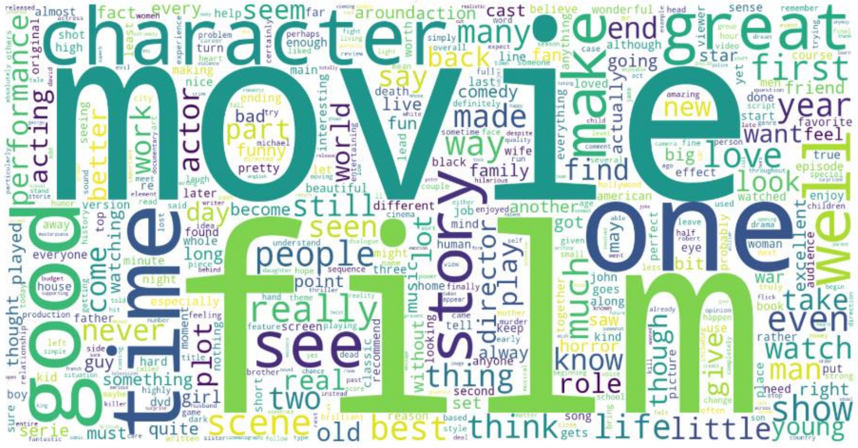
INICIJALNA ANALIZA SKUPA



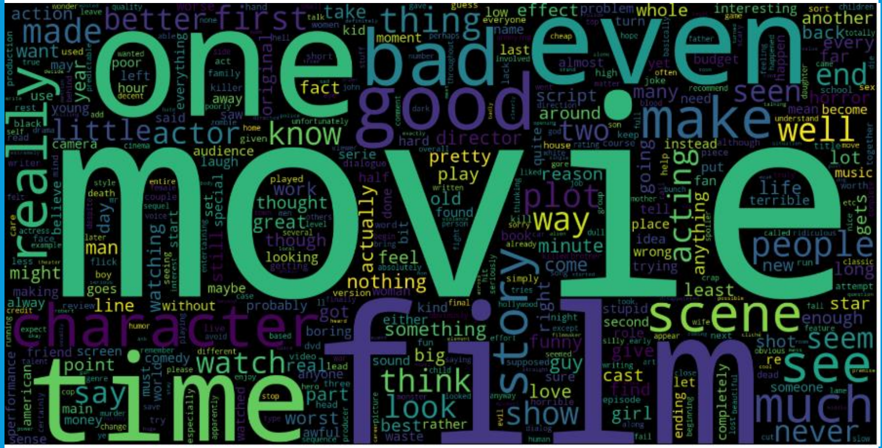


Statistika	Pozitivni	Negativni
Prosek	1324.80	1294.06
Medijana	968.00	973.00
Standardna devijacija	1031.47	945.87
Maks vrednost	13704.00	8969.00
Min vrednost	65.00	32.00

Statistika	Pozitivni	Negativni
Prosek	241.84	239.48
Medijana	179.00	182.00
Standardna devijacija	184.33	172.02
Maks vrednost	2515.00	1620.00
Min vrednost	10.00	6.00



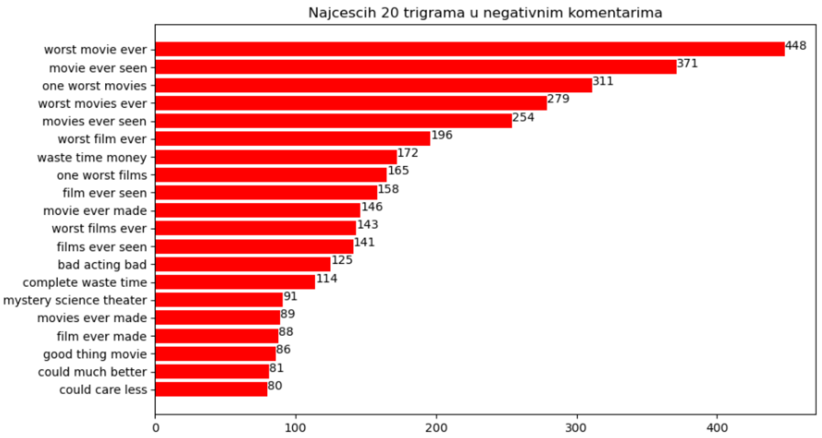
Pozitivni komentari

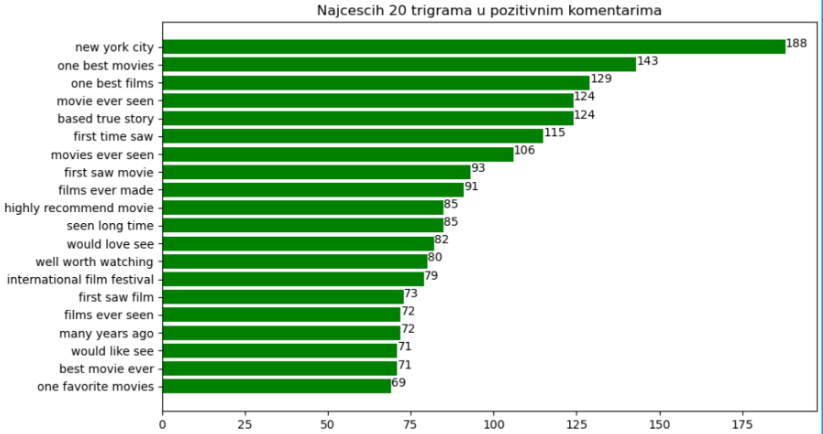


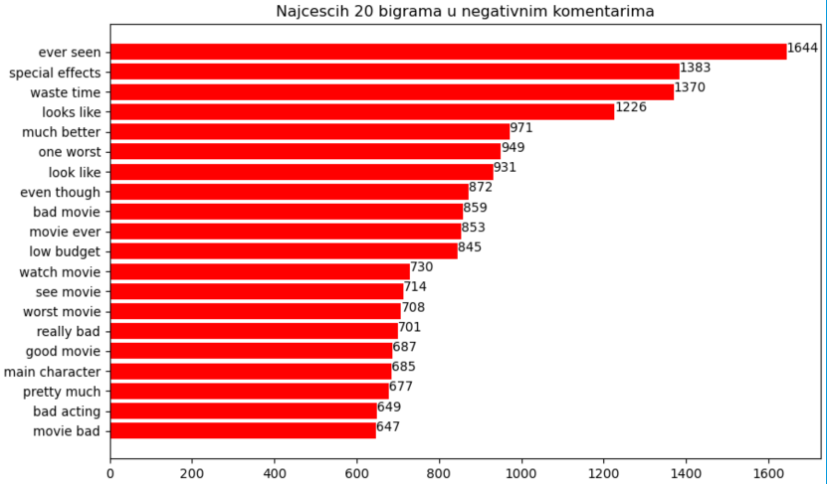
Negativni komentari

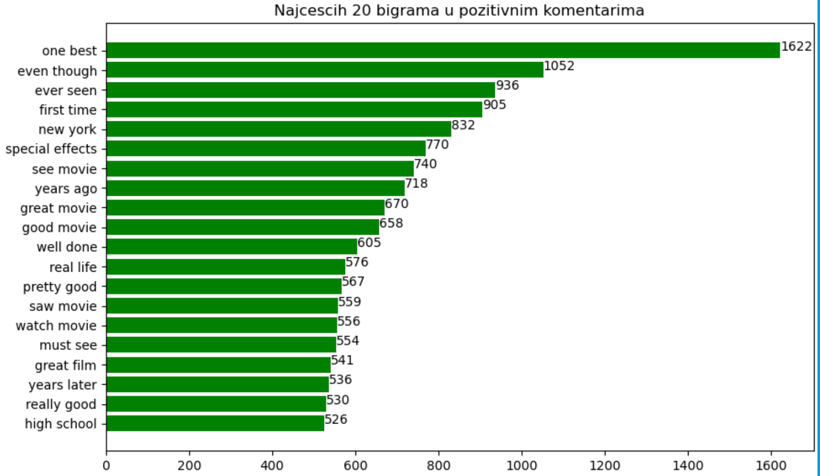
WORD CLOUD – NAJZASTUPLJENIJE REČI

ANALIZA BIGRAMA I TRIGRAMA









VEKTORIZACIJA

- Bag of Words
- tf-idf

Komentar → Preprocesiranje → Preprocesiran komentar → Tokenizacija → Tokeni → Vektorizacija → Vektor

```python
def simple_tokenization(review):
    tokens = nltk.tokenize.word_tokenize(review)
    tokens_without_punctuation = [token for token in tokens if token not in string.punctuation]
    return tokens_without_punctuation
```

```python
def review_preprocessor(text):
    text = text.lower()
    text = short_form_transform(text)
    text = strip_html(text)
    text = strip_url(text)
    text = full_stop_abbrev_elim(text)
    return text
```

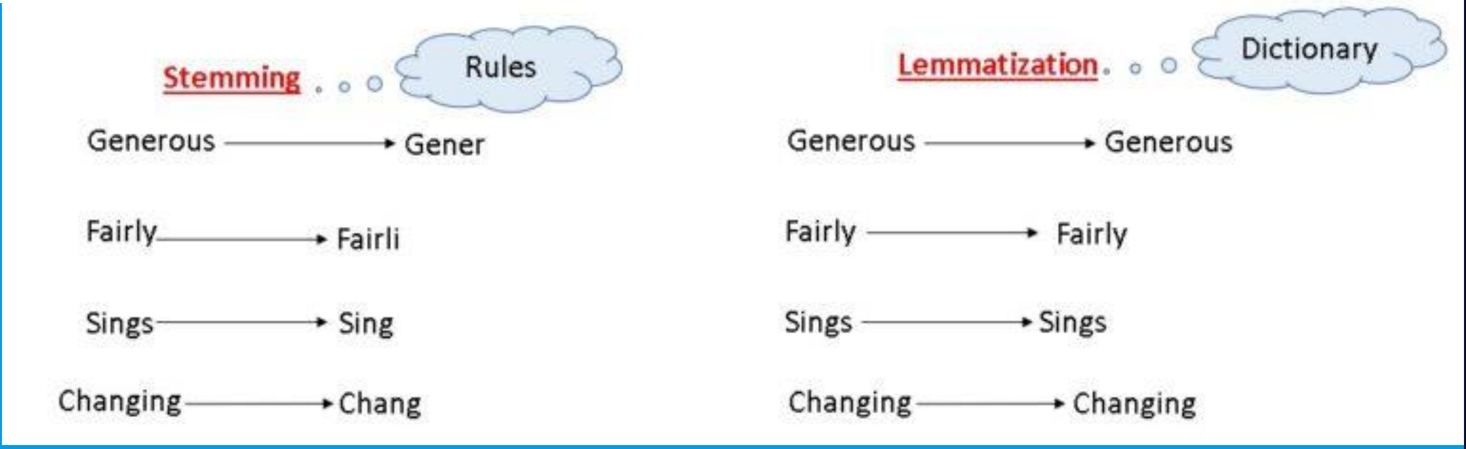
```python
def review_tokenizer(stemming, text):
    tokens = simple_tokenization(text)
    tokens = remove_stop_words(tokens)

    stems = []

    for token in tokens:

        token_pattern = re.compile(r'\b[^\W\d_]+\b')
        if not token_pattern.match(token) or len(token) <= 2:
            continue

        stem = stemming.stem(token)
        stems.append(stem)
    return stems
```

$$TF(t,d) = \frac{number\ of\ times\ t\ appears\ in\ d}{total\ number\ of\ terms\ in\ d}$$

$$IDF(t) = log\frac{N}{1+df}$$

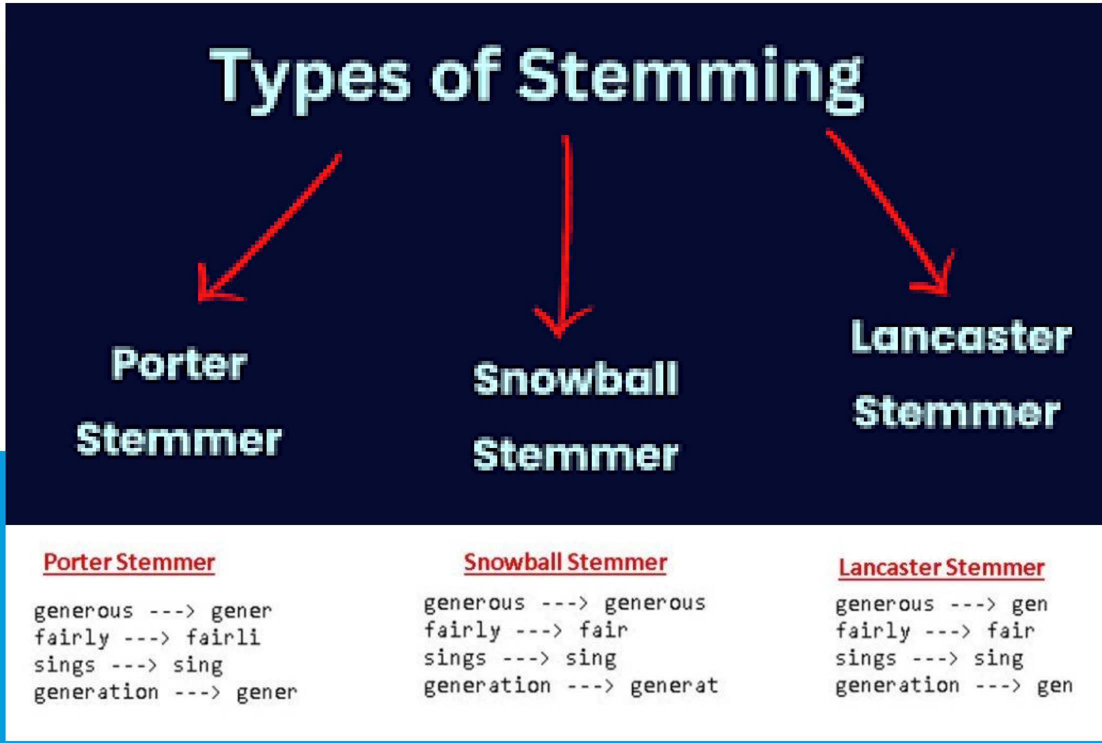
$$TF-IDF(t,d) = TF(t,d) * IDF(t)$$

STEMOVANJE ILI LEMATIZACIJA

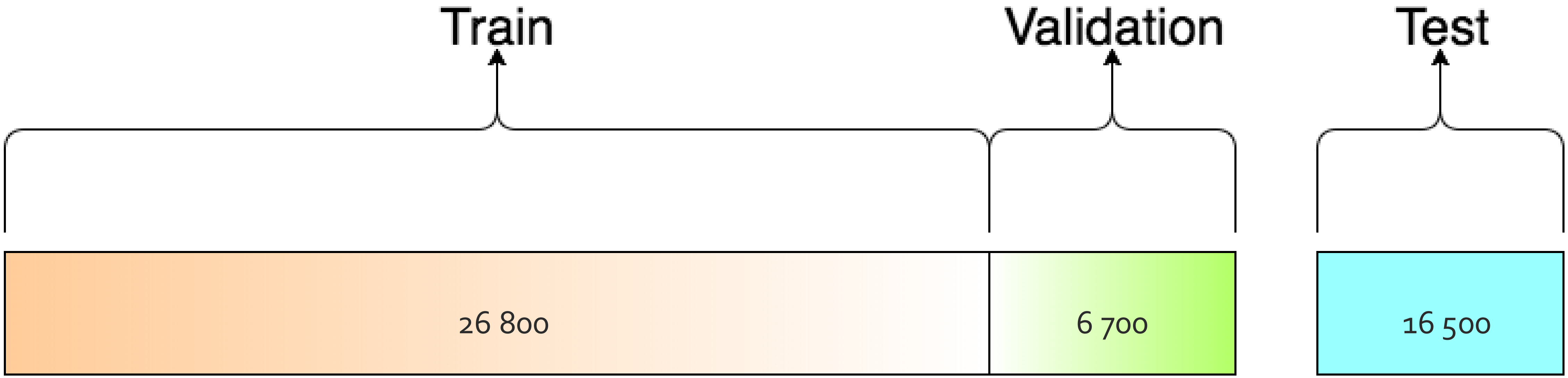


- Dosledna primena niza pravila kako bi se dobio `stem` - veštački koren reči
- Manja preciznost
- Značajno brže
- Stemovi mogu biti reči bez značenja (ograničena primena)

- Reči se pridružuje njen gramatički koren (lema)
- Veća preciznost
- Značajno sporije
- Leme zadržavaju značenje polazne reči



Povećanje agresivnosti i brzine



PODELA SKUPA I PODEŠAVANJE HIPERPARAMETARA

MODELI – NAJBOLJE VREDNOSTI HIPERPARAMETARA

Logistička regresija

```
Cs = np.array([10**i for i in range(-5,5)])
penalties = np.array(['l1', 'l2', 'elasticnet'])
l1_ratios = np.array([0.1 * i for i in range(1, 10)])
```

```
Najbolja vrednost regularizacionog hiperparametra: 1.0
Najbolja norma regularizacije: elasticnet
Najbolji l1_ratio: 0.4
Najbolji skor: 0.8365671641791045
```

Kernelizovani SVM

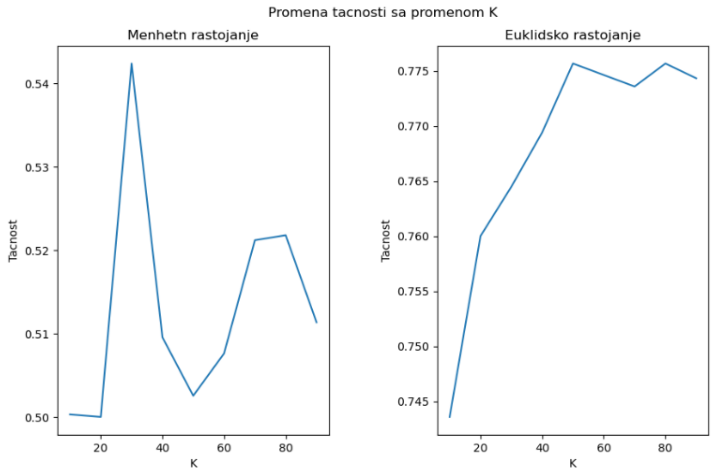
```
Cs = np.array([10**i for i in range(-3, 3)])
# scale - 1 / (n_features * X.var())
# auto - 1/ n_features
gammas = ['scale', 'auto']
kernels = ['linear', 'rbf', 'sigmoid']
```

```
Najbolja vrednost regularizacionog hiperparametra: 1.0
Najbolji tip kernela: rbf
Najbolji koeficijent kernela: scale
Najbolji skor: 0.8417910447761194
```

K najbližih suseda

```
Ks = np.array([10*i for i in range(1, 10)])
dist_metrics = ['manhattan', 'euclidean']
```

```
Najbolji broj suseda : 50
Najbolja metrika: euclidean
Najbolji skor: 0.7756716417910448
```

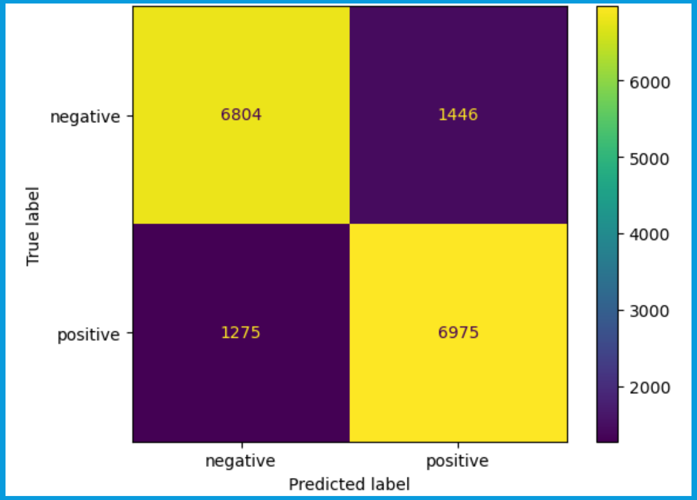


METRIKE - LOGISTIČKA REGRESIJA

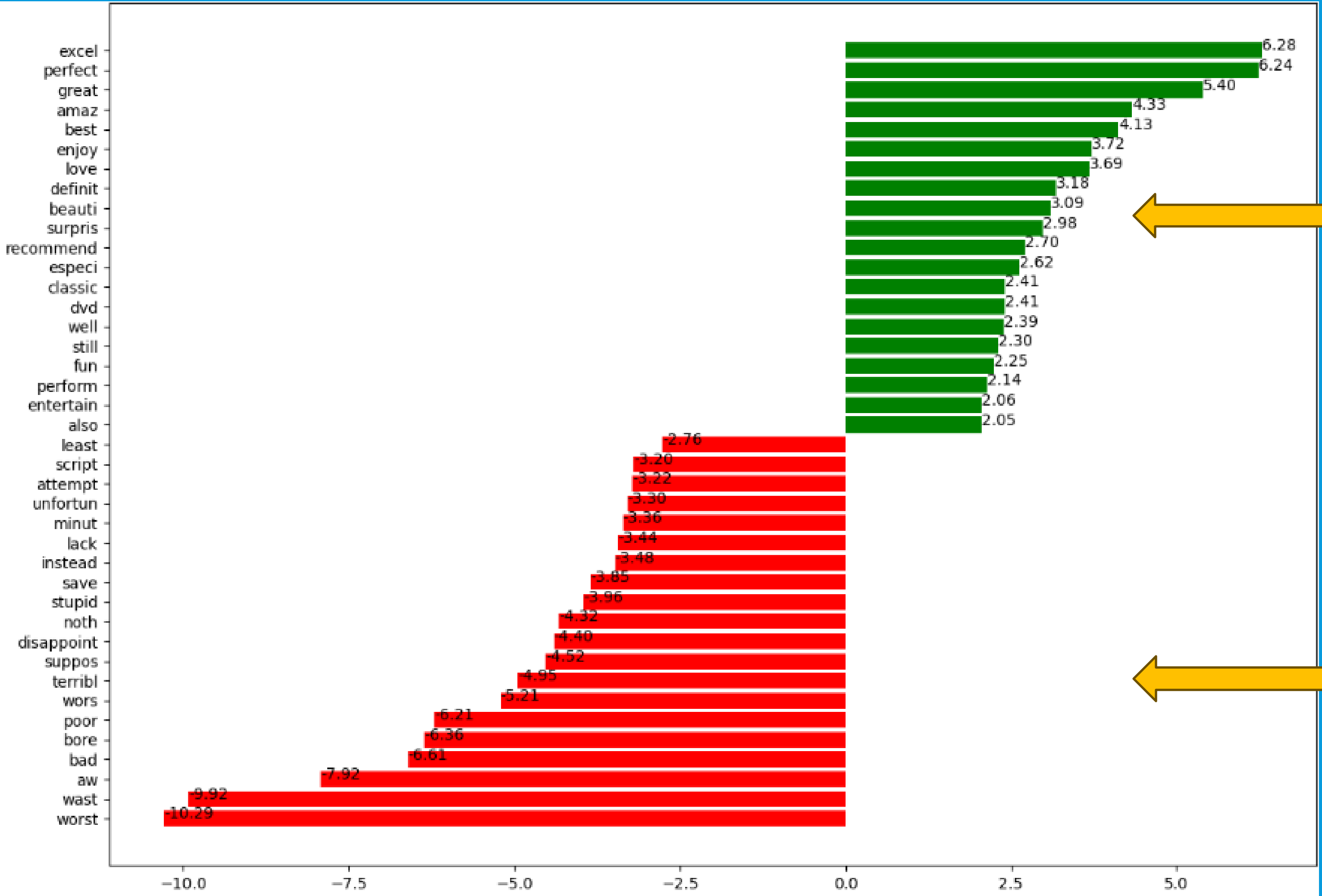
- Tačnost na train-val skupu: 0.8402089552238806
- Tačnost na test skupu: 0.8350909090909091

	precision	recall	f1-score	support
0	0.84	0.82	0.83	8250
1	0.83	0.85	0.84	8250
accuracy			0.84	16500
macro avg	0.84	0.84	0.84	16500
weighted avg	0.84	0.84	0.84	16500

Matrica konfuzije



LOGISTIČKA REGRESIJA – ANALIZA KOEFICIJENATA



20 reči vokabulara koje najviše sugerišu da je komentar pozitivan sa pridruženim koeficijentima

20 reči vokabulara koje najviše sugerišu da je komentar negativan sa pridruženim koeficijentima

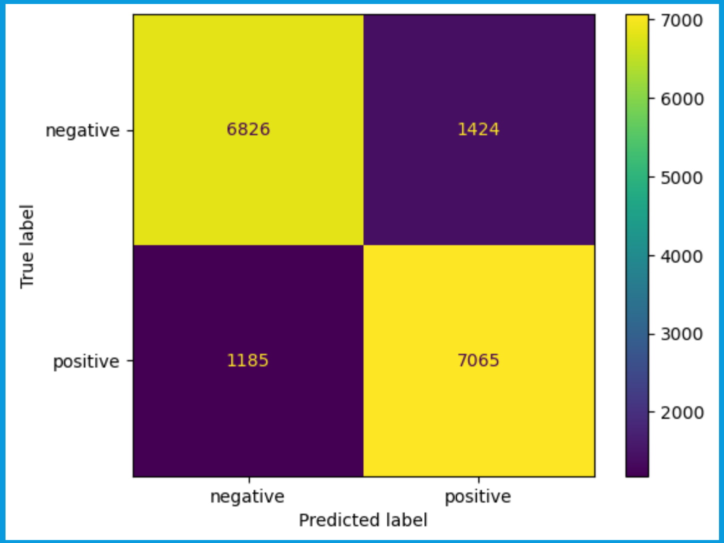
METRIKE – KERNELIZOVANI SVM

- Tačnost na train-val skupu: 0.9412238805970149
- Tačnost na test skupu: 0.8418787878787879

Overfitting!!!

	precision	recall	f1-score	support
0	0.85	0.83	0.84	8250
1	0.83	0.86	0.84	8250
accuracy			0.84	16500
macro avg	0.84	0.84	0.84	16500
weighted avg	0.84	0.84	0.84	16500

Matrica konfuzije

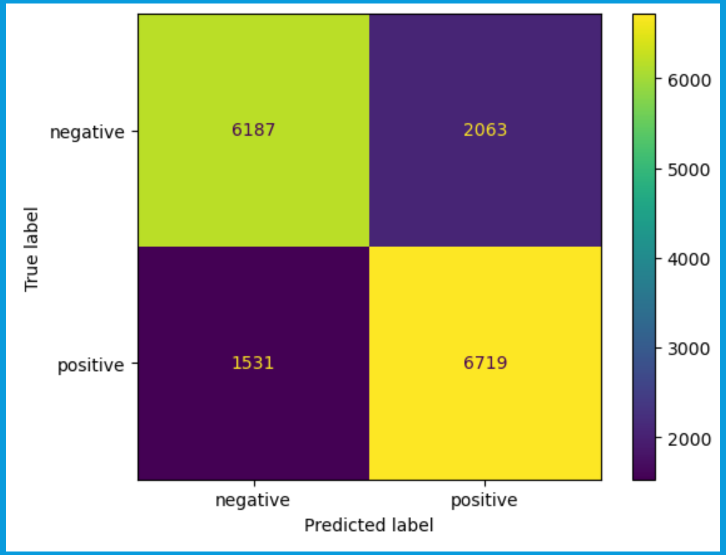


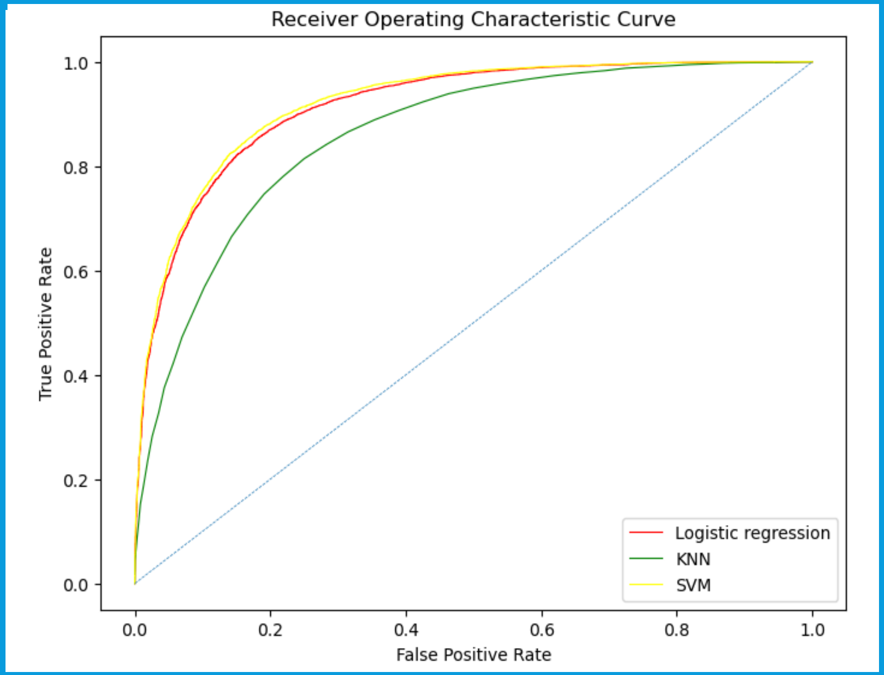
METRIKE – K NAJBLIŽIH SUSEDA

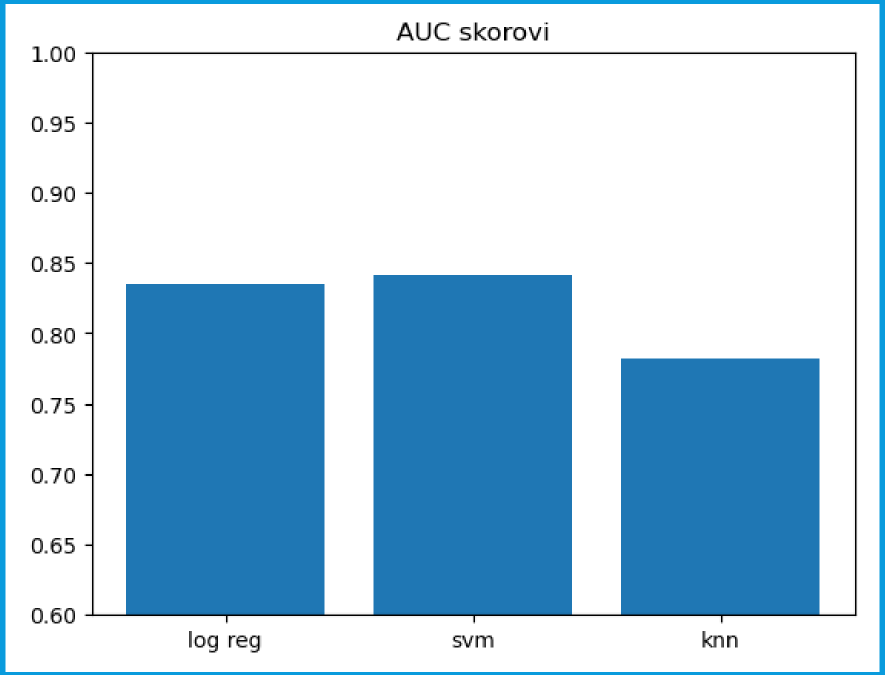
- Tačnost na train-val skupu: 0.7893432835820896
- Tačnost na test skupu: 0.7821818181818182

	precision	recall	f1-score	support
0	0.80	0.75	0.77	8250
1	0.77	0.81	0.79	8250
accuracy			0.78	16500
macro avg	0.78	0.78	0.78	16500
weighted avg	0.78	0.78	0.78	16500

Matrica konfuzije

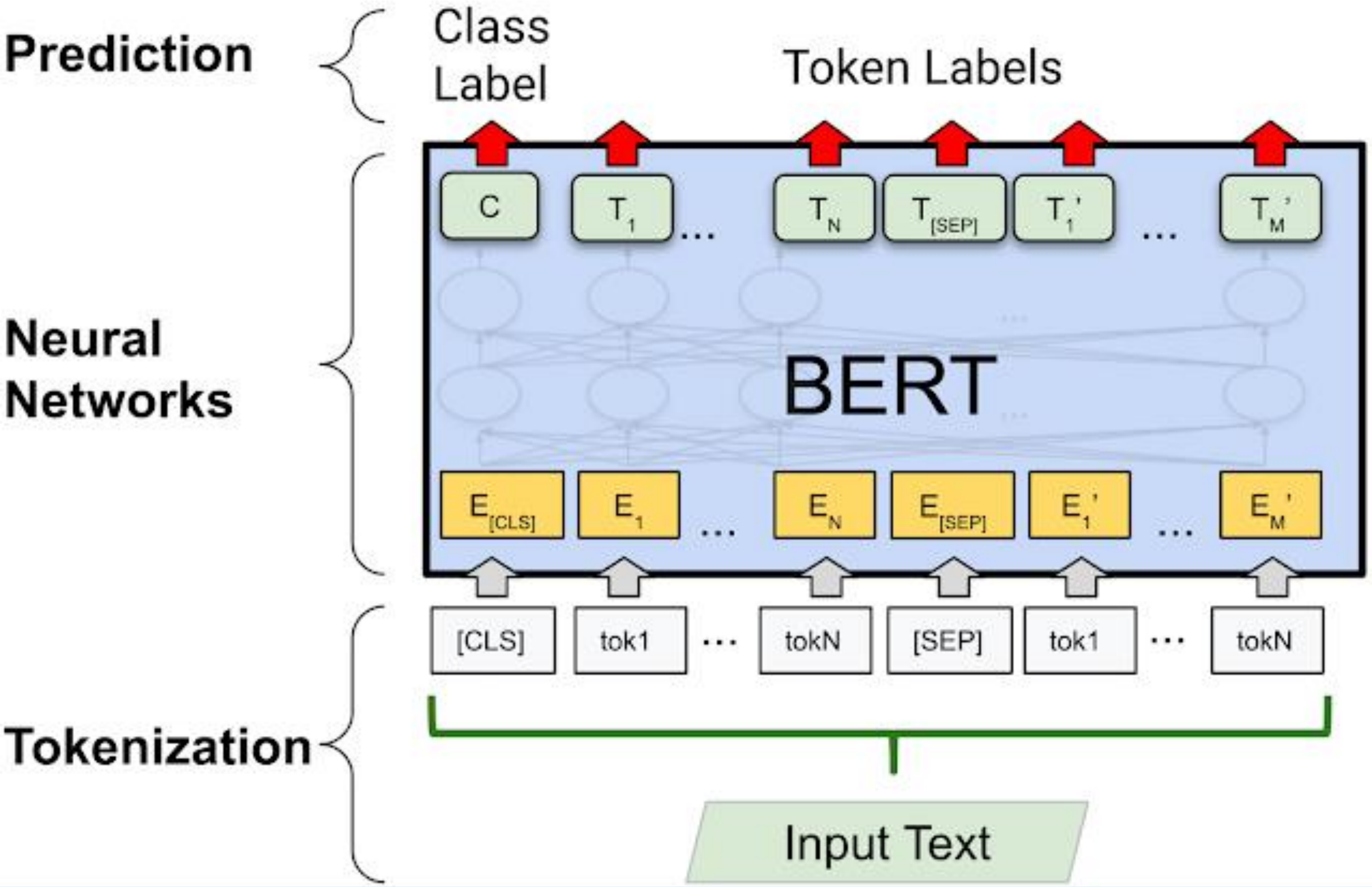


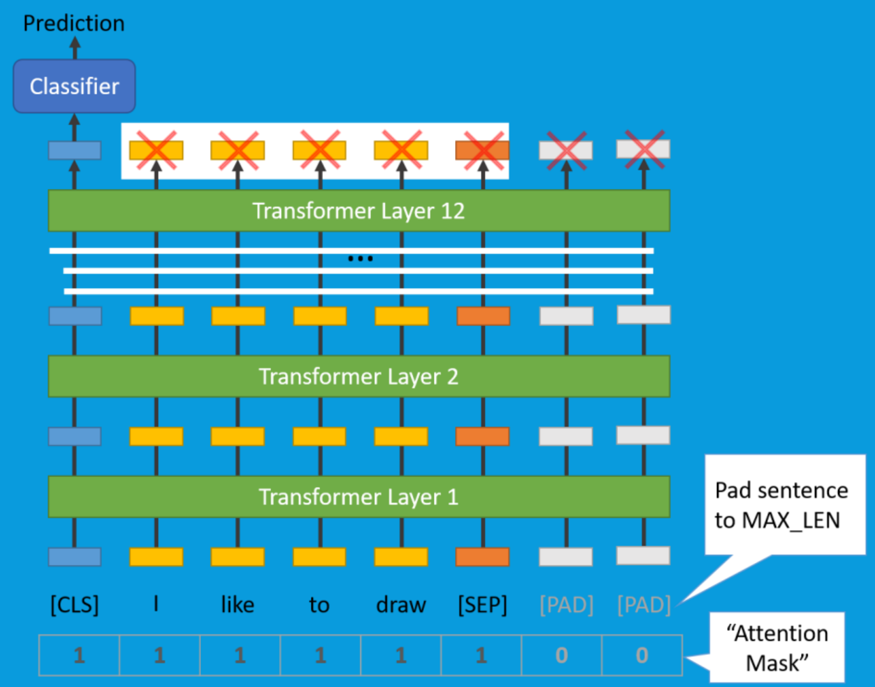




UPOREDNA ANALIZA

BERT TRANSFORMER



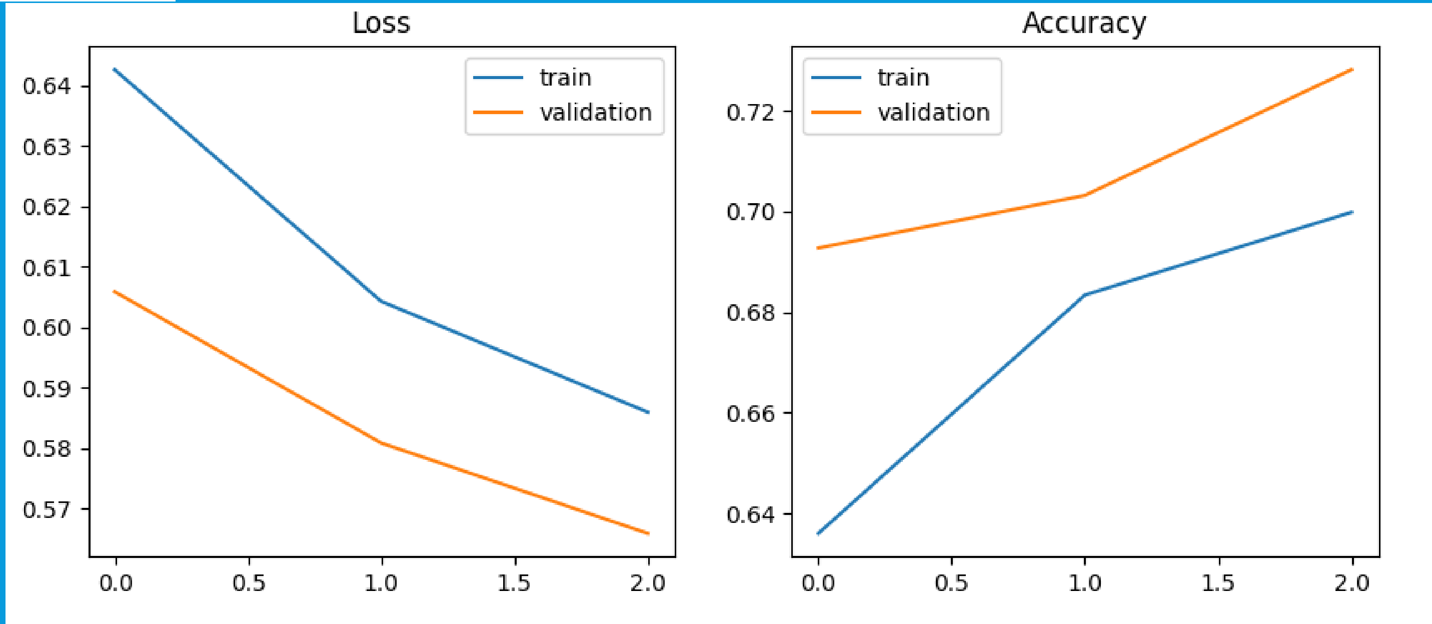


Specijalni tokeni:
[CLS] - classification task
[SEP] - separator token
[PAD] - padding token

Konfiguracija modela

Layer (type)	Output Shape	Param #	Connected to
input_ids (InputLayer)	[(None, 512)]	0	[]
attention_mask (InputLayer)	[(None, 512)]	0	[]
bert (TFBertMainLayer)	TFBaseModelOutputWithPoolingAndCrossAttentions(last_hidden_state=(None, 512, 768), pooler_output=(None, 768), past_key_values=None, hidden_states=None, attentions=None, cross_attentions=None)	108310272	['input_ids[0][0]', 'attention_mask[0][0]']
dense (Dense)	(None, 1024)	787456	['bert[0][1]']
dense_1 (Dense)	(None, 1)	1025	['dense[0][0]']

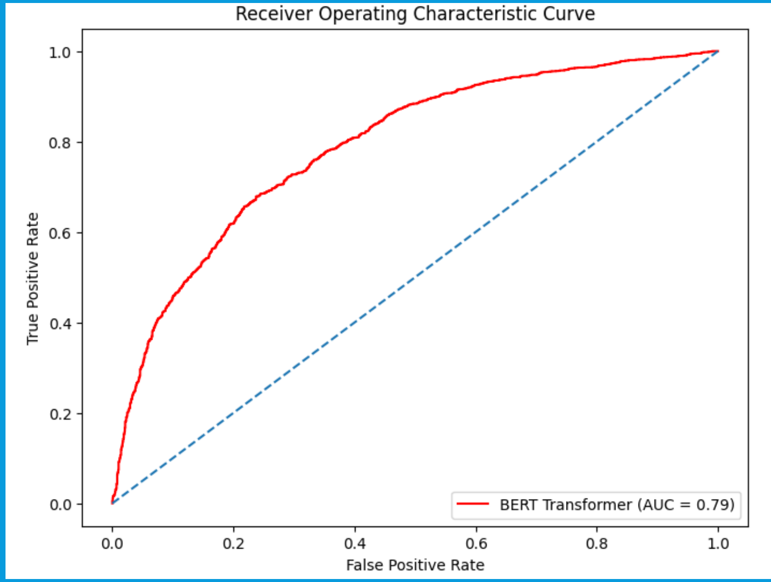
Obučavanje modela



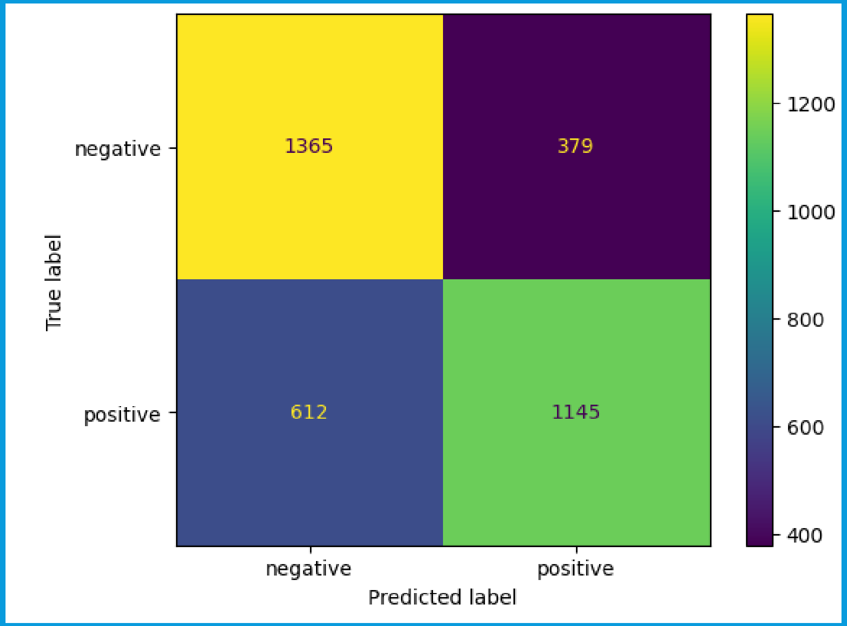
METRIKE - TRANSFORMER

- Tačnost na test skupu: 0.7169380177092259

	precision	recall	f1-score	support
0	0.69	0.78	0.73	1744
1	0.75	0.65	0.70	1757
accuracy			0.72	3501
macro avg	0.72	0.72	0.72	3501
weighted avg	0.72	0.72	0.72	3501



Matrica konfuzije



LITERATURA

- https://www.analyticsvidhya.com/blog/2021/11/an-introduction-to-stemming-in-natural-language-processing/
- https://www.datacamp.com/tutorial/stemming-lemmatization-python
- https://medium.com/@Mirza_Yusuf/using-a-bert-model-for-sentiment-analysis-6c6fcc106843
- https://www.geeksforgeeks.org/sentiment-classification-using-bert/