

Сравнение между модели за анализ на чувства, общоцелеви и специфични за домейна, върху данни от Tweeter

Стеван Велев, Ф.Н.: 62537

Софтуерно инженерство, IV курс

Софийски университет “Св. Климент Охридски”



Съдържание

I. Увод

II. Преглед на подходите за анализ на чувства в текст

III. Проектиране

IV. Реализация

V. Заключение

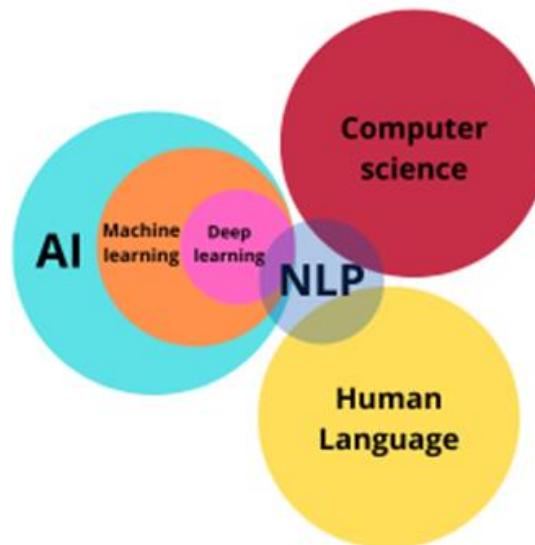


I. Увод

“Coding is today’s language of creativity.”
(Maria Klawe)

Обработка на естествен език (Natural Language Processing)

- **интердисциплинарна подобласт на компютърните науки и лингвистиката**
- **цели да даде способност на компютрите да поддържат и манипулират човешки език**
- **подходи за машинно самообучение:** базирани на правила или вероятностни подходи



Sentiment Analysis



My experience
so far has been
fantastic!

POSITIVE



The product is
ok I guess

NEUTRAL



Your support team is
useless

NEGATIVE



MonkeyLearn

Цел и задачи

Цел: да се направи **сравнение** между **точността** на няколко модела за извършване на **анализ на чувства**, поставени в **конкретен домейн** – комуникация чрез социалните мрежи и по-конкретно чрез съобщения в *Tweeter*

Задачи:

1. **Намиране на данни**, върху които ще се обучава модела, и тяхната **обработка** за по-лесното им разбиране от машина
2. **Трениране на собствени модели** чрез **Бернулиев наивен Байесов класификатор** и **логистична регресия** върху конкретно избрано множество от данни, което е преминало през предварителна обработка
3. **Дефиниране на метрики за оценка** на моделите – точност (*accuracy*), класификационен доклад (*classification report*), матрица на объркване (*confusion matrix*), ROC крива (*ROC curve*)
4. **Интегриране на готови модели с общо предназначение** (*nltk – Sentiment Intensity Analyzer*) и такива **специфични за домейна** (*transformers – Twitter roBERTa Sentiment Analyzer*)
5. **Извършване на оценка** на моделите спрямо дефинираните метрики
6. **Анализ и сравнение** на получените резултати



II. Преглед на подходите за решение |

“Simplicity is prerequisite for reliability.”
(Edsger Dijkstra)

Подходи за решение на задачата за анализ на чувствата и мненията на потребители

- От машинното самообучение – учене с учител и без учител
- Чрез лексикони – автоматично извършване на анализ на мнението въз основа на речници с предварително зададена оценка на думите
- Хибриден – комбинира действията, базирани на правила, и автоматичния подход, базиран на машинно самообучение



Анализ на мненията, базиран на правила

Техника (правило)	Описание
Лематизация (Lemmatisation)	Групиране на различните части на една дума в лема
Стеминг (Stemming)	Съкращаване на думата до нейния корен, което намалява размера на речника от думи в документа и подобрява резултатите, особено при по-малки множества от данни
Разделяне на текста на графични думи и пунктуационни знаци (Tokenisation)	Разделяне на текста на съставните му думи и пунктуационни знаци
Стоп думи или „шумови“ думи (Stopwords)	Премахване на думи, които не носят важно значение (цифри, знаци, местоимения, предлози и др.)
Филтриране на думи (Filtering Tokens)	Филтриране на думите по различни критерии (напр. дължина на думата)
Трансформиране на думи (Transforming Tokens)	Трансформиране на думите (напр. превръщане на главните букви в малки)
Автоматичен морфологичен анализ (Part-Of-Speech Tagging)	Маркиране на частите на речта в изречението съществителни, глаголи, наречия, съюзи и др.
Автоматичен синтактичен анализ (Parsing)	Генерира дърво на връзките между думите в изречението (синтактично дърво)
Многозначност (Word Sense Disambiguation)	Техники за отстраняване на многозначността на думите или фразите

Автоматичен анализ на чувствата

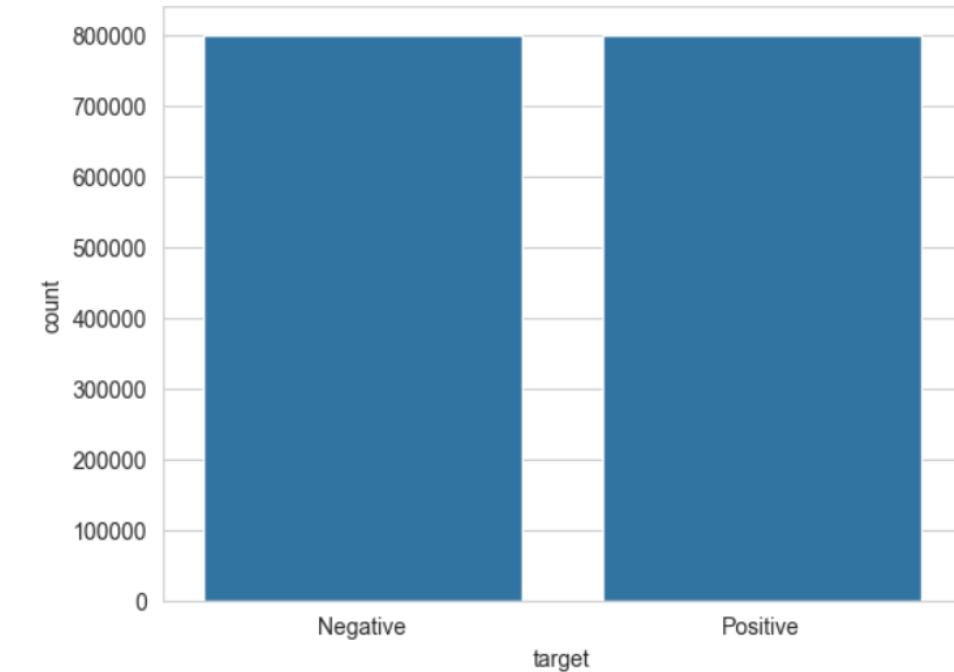
- **Процес на обучение** – моделът се учи да асоциира определен вход (текст) със съответния изход (таг) въз основа на тестовите примери, използвани за обучение
- **Процес на предсказване** – трансформиране на невиждани текстови входове във вектори с характеристики, които се подават на модела, който генерира предсказаните тагове (+-0)
- **Bag-of-words** и **bag-of-ngrams** техники
- **Класификационни алгоритми:**
 - **Наивен Байсов класификатор** – теорема на Байс
 - **Логистична регресия** – предсказване на Y при дадено X
 - **Машина с поддържащи вектори** – обучаващите примери като точки в n-мерно пространство
 - **Дълбоко самообучение** – вдъхновено от структурата и функцията на мозъка

III. Проектиране

“First, solve the problem. Then write the code.”
(John Johnson)

Модел на данните

- От областта на **социалната мрежа Twitter/X** – 1.6 miliona съобщения
- **Kaggle:** Sentiment140 dataset with 1.6 million tweets
- **Полета:**
 - **target (int64)** – оценъчния заряд на съобщението
(0 – негативни, 4 – положителни)
 - **ids (int64)** – уникалният идентификатор на съобщението
 - **date (object)** – датата на публикуване на съобщението
 - **flag (object)** – заявката към съобщението
 - **user (object)** – потребителят, публикувал съобщението
 - **text (object)** – текстът на съобщението



Предварителна обработка на данните

Селекция на
данните

Селекция на данните

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 1600000 entries, 0 to 1599999  
Data columns (total 6 columns):  
 #   Column   Non-Null Count   Dtype     
 ---  --      --      --      --  
 0   target   1600000 non-null  int64    
 1   ids      1600000 non-null  int64    
 2   date     1600000 non-null  object    
 3   flag     1600000 non-null  object    
 4   user     1600000 non-null  object    
 5   text     1600000 non-null  object    
 dtypes: int64(2), object(4)  
memory usage: 73.2+ MB
```



```
data = df[ [ 'text' , 'target' ] ]
```

Предварителна обработка на данните

Селекция



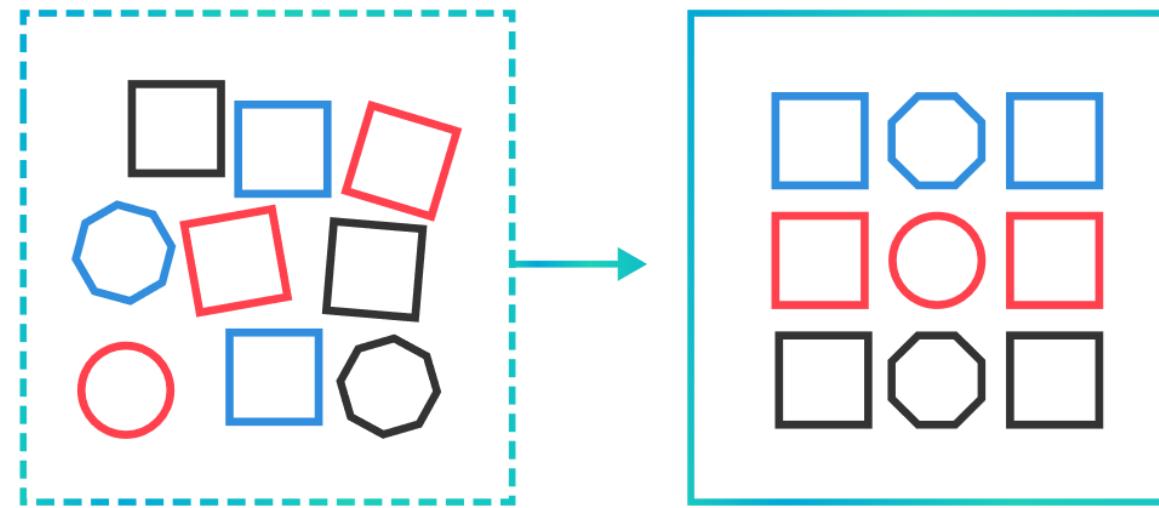
Трансформиране
на полета

Трансформиране на полета

```
df['target'].unique()
```

```
array([0, 4], dtype=int64)
```

```
data['target'] = data['target'].replace(4, 1)
```



Предварителна обработка на данните

Селекция



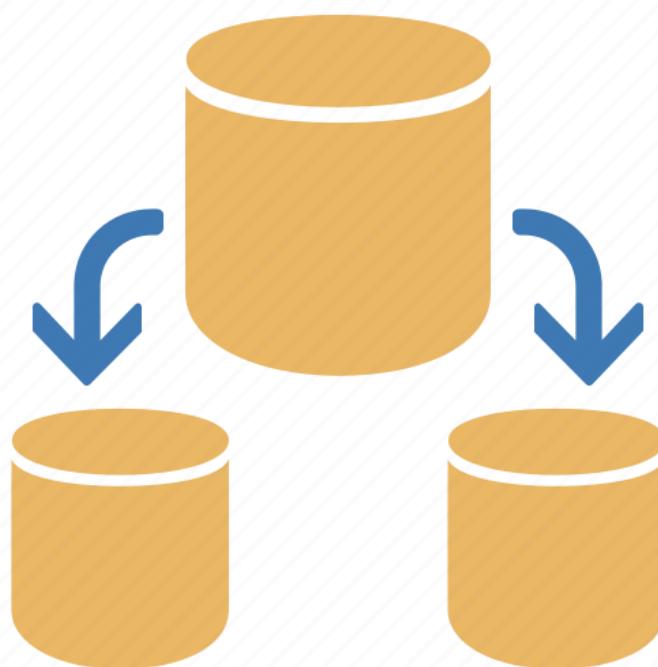
Трансформиране
на полета



Сегрегация на
данни

Сегрегация на данни

```
data_negative = data[data['target'] == 0]  
data_positive = data[data['target'] == 1]
```



Предварителна обработка на данните

Селекция



Трансформиране
на полета



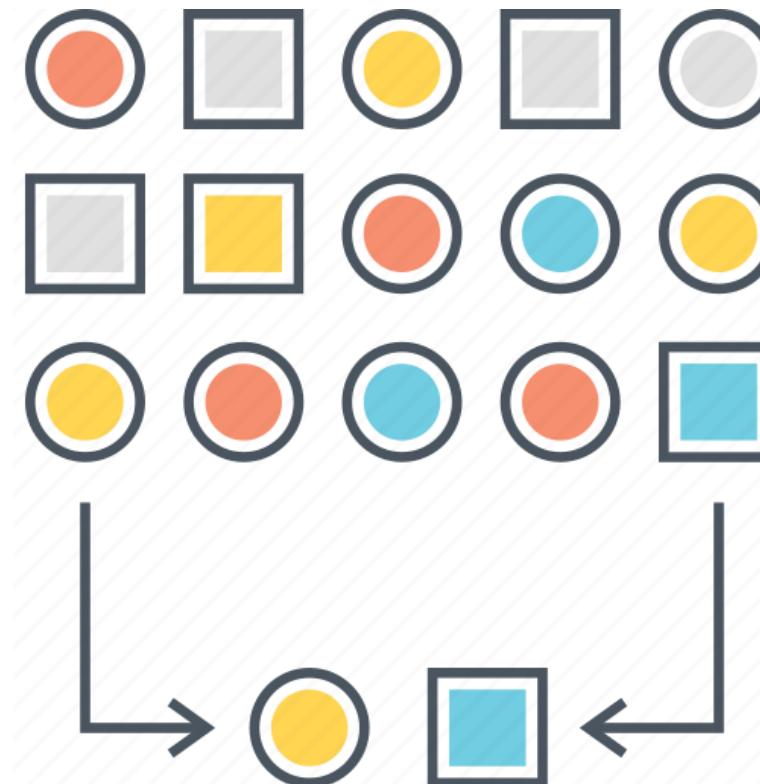
Сегрегация на
данни



Извадка от
данните

Извадка от данните

```
data_positive = data_positive.sample(n = 400000, random_state = 42)  
data_negative = data_negative.sample(n = 400000, random_state = 42)
```



Предварителна обработка на данните

Селекция

Трансформиране
на полета

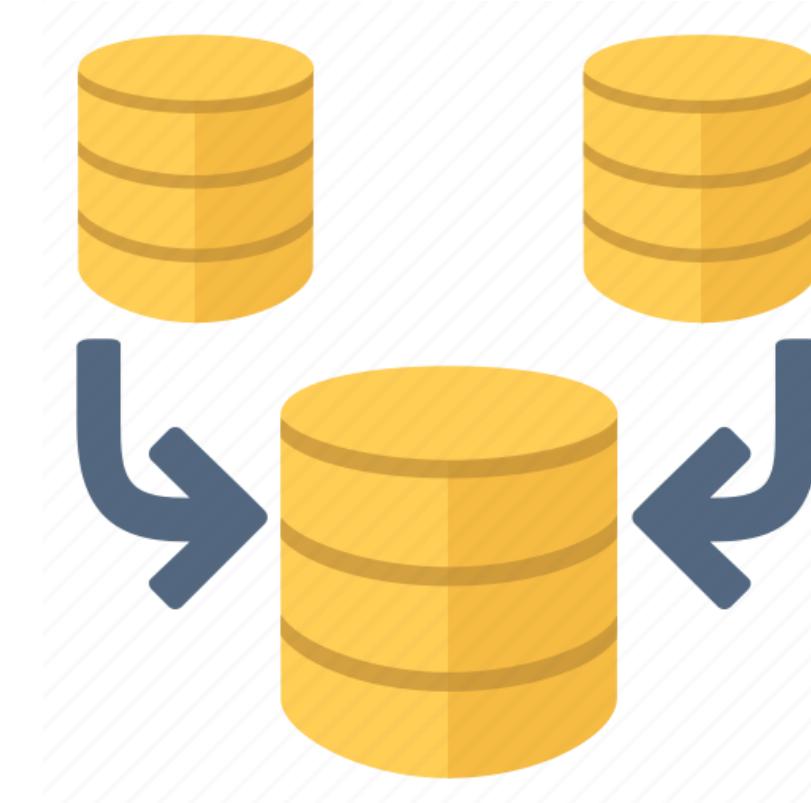
Сегрегация

Извадка от
данните

Обединяване на
данните

Обединяване на данните

```
dataset = pd.concat([data_positive, data_negative])
```



Предварителна обработка на данните



Преминаване към малки букви

```
dataset.head()
```

	text	target
0	Is lookin 4ward to a long weekend really dont...	1
1	#myweakness Is music and i live to meet the p...	1
2	figured out the Internet on my new iPod	1
3	@hillsongunited can't wait to worship with you...	1
4	@sillybeggar Congrats James !! I'm sure the bo...	1

```
dataset['text'] = dataset['text'].str.lower()
```

```
dataset.head()
```

	text	target
0	is lookin 4ward to a long weekend really dont...	1
1	#myweakness is music and i live to meet the p...	1
2	figured out the internet on my new ipod	1
3	@hillsongunited can't wait to worship with you...	1
4	@sillybeggar congrats james !! i'm sure the bo...	1

Предварителна обработка на данните



Премахване на „шумови“ думи

```
dataset.head()
```

	text	target
0	is lookin 4ward to a long weekend really dont...	1
1	#myweakness is music and i live to meet the p...	1
2	figured out the internet on my new ipod	1
3	@hillsongunited can't wait to worship with you...	1
4	@sillybeggar congrats james !! i'm sure the bo...	1

```
def remove_stopwords(text):
    return " ".join([word for word in str(text).split() if word not in stopwords_set])
dataset['text'] = dataset['text'].apply(lambda text: remove_stopwords(text))
```

```
dataset.head()
```

	text	target
0	lookin 4ward long weekend really dont want go ...	1
1	#myweakness music live meet people make	1
2	figured internet new ipod	1
3	@hillsongunited can't wait worship guys tonigh...	1
4	@sillybeggar congrats james !! i'm sure book g...	1

Предварителна обработка на данните



Премахване на URL адреси

```
print(dataset['text'].iloc[27034])
```

```
parents, here's website full free preschool computer games, enjoy http://bit.ly/pcql5 ms. destiny
```

```
def remove_URLs(text):
    return re.sub('((www\.[^s]+)|(https?://[^s]+))', ' ', text)
dataset['text'] = dataset['text'].apply(lambda text: remove_URLs(text))
```

```
print(dataset['text'].iloc[27034])
```

```
parents, here's website full free preschool computer games, enjoy s. destiny
```

Предварителна обработка на данните



Премахване на пунктуация

```
dataset.head()
```

	text	target
0	lookin 4ward long weekend really dont want go ...	1
1	#myweakness music live meet people make	1
2	figured internet new ipod	1
3	@hillsongunited can't wait worship guys tonigh...	1
4	@sillybeggar congrats james !! i'm sure book g...	1

```
def remove_punctuations(text):
    translator = str.maketrans('', '', punctuation_list)
    return text.translate(translator)
dataset['text'] = dataset['text'].apply(lambda text: remove_punctuations(text))
```

```
dataset['text'].head()
0    lookin 4ward long weekend really dont want go ...
1                  myweakness music live meet people make
2                      figured internet new ipod
3    hillsongunited cant wait worship guys tonight ...
4    sillybeggar congrats james im sure book going...
Name: text, dtype: object
```

Предварителна обработка на данните



Премахване на повтарящи се символи

dataset.head(10)

	text	target
0	lookin 4ward long weekend really dont want go ...	1
1	myweakness music live meet people make	1
2	figured internet new ipod	1
3	hillsongunited cant wait worship guys tonight ...	1
4	sillybeggar congrats james im sure book going...	1
5	debbybruck beautiful children smile world smiles	1
6	bethofalltrades s9 happy birthday	1
7	adinfinitum yes can ultimate vegan guide eric ...	1
8	getting writing mee fun nightt	1
9	noopman remote prefer site theres substitute w...	1

```
def remove_repeated_characters(text):
    return re.sub(r'(.+)\1+', r'\1', text)
dataset['text'] = dataset['text'].apply(lambda text: remove_repeated_characters(text))
```

dataset.head(10)

	text	target
0	lokin 4ward long wekend realy dont want go wor...	1
1	myweaknes music live met people make	1
2	figured internet new ipod	1
3	hilsongunited cant wait worship guys tonight i...	1
4	silybegar congrats james im sure bok going hug...	1
5	debbybruck beautiful children smile world smiles	1
6	bethofaltrades s9 hapy birthday	1
7	adinfinitum yes can ultimate vegan guide eric ...	1
8	geting writing me fun night	1
9	nopman remote prefer site theres substitute wo...	1

Предварителна обработка на данните



Премахване на числа

```
dataset.head()
```

	text	target
0	lokin 4ward long wekend realy dont want go wor...	1
1	myweaknes music live met people make	1
2	figured internet new ipod	1
3	hilsongunited cant wait worship guys tonight i...	1
4	silybegar congrats james im sure bok going hug...	1

```
def remove_numbers(text):
    return re.sub('[0-9]+', '', text)
dataset['text'] = dataset['text'].apply(lambda text: remove_numbers(text))
```

```
dataset.head()
```

	text	target
0	lokin ward long wekend realy dont want go work...	1
1	myweaknes music live met people make	1
2	figured internet new ipod	1
3	hilsongunited cant wait worship guys tonight i...	1
4	silybegar congrats james im sure bok going hug...	1

Предварителна обработка на данните



Разделяне на текста на графични думи

```
dataset.head()
```

	text	target
0	lokin ward long wekend realy dont want go work...	1
1	myweaknes music live met people make	1
2	figured internet new ipod	1
3	hilsongunited cant wait worship guys tonight i...	1
4	silybegar congrats james im sure bok going hug...	1

```
tokenizer = RegexpTokenizer(r'\w+')
dataset['text'] = dataset['text'].apply(tokenizer.tokenize)
```

```
dataset.head()
```

	text	target
0	[lokin, ward, long, wekend, realy, dont, want, ...]	1
1	[myweaknes, music, live, met, people, make]	1
2	[figured, internet, new, ipod]	1
3	[hilsongunited, cant, wait, worship, guys, ton...	1
4	[silybegar, congrats, james, im, sure, bok, go...	1

Предварителна обработка на данните



Стеминг – премахване на окончания

```
dataset.head()
```

	text	target
0	[lokin, ward, long, wekend, realy, dont, want,...]	1
1	[myweaknes, music, live, met, people, make]	1
2	[figured, internet, new, ipod]	1
3	[hilsongunited, cant, wait, worship, guys, ton...	1
4	[silybegar, congrats, james, im, sure, bok, go...	1

```
st = nltk.PorterStemmer()
def stemming(text):
    return [st.stem(word) for word in text]
dataset['text'] = dataset['text'].apply(lambda text : stemming(text))
```

```
dataset.head()
```

	text	target
0	[lokin, ward, long, wekend, reali, dont, want,...]	1
1	[myweakn, music, live, met, peopl, make]	1
2	[figur, internet, new, ipod]	1
3	[hilsongunit, cant, wait, worship, guy, tonigh...	1
4	[silybegar, congrat, jame, im, sure, bok, go, ...]	1

Предварителна обработка на данните



Лематизация

```
dataset.head(6)
```

	text	target
0	[lokin, ward, long, wekend, reali, dont, want,...]	1
1	[myweakn, music, live, met, peopl, make]	1
2	[figur, internet, new, ipod]	1
3	[hilsongunit, cant, wait, worship, guy, tonigh...	1
4	[silybegar, congrat, jame, im, sure, bok, go, ...]	1
5	[debybruck, beauti, children, smile, world, sm...	1

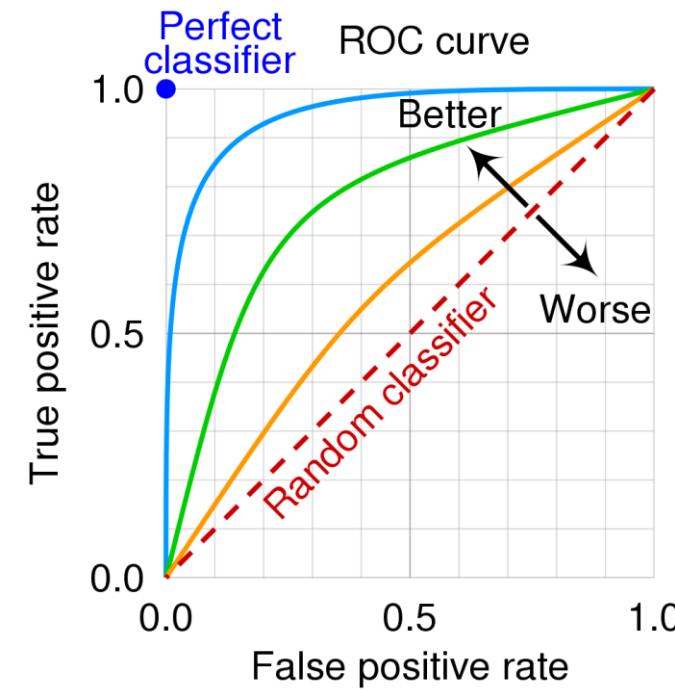
```
wnlm = WordNetLemmatizer()
def lemmatization(text):
    return [wnlm.lemmatize(word) for word in text]
dataset['text'] = dataset['text'].apply(lambda text: lemmatization(text))
```

```
dataset.head(6)
```

	text	target
0	[lokin, ward, long, wekend, reali, dont, want,...]	1
1	[myweakn, music, live, met, peopl, make]	1
2	[figur, internet, new, ipod]	1
3	[hilsongunit, cant, wait, worship, guy, tonigh...	1
4	[silybegar, congrat, jame, im, sure, bok, go, ...]	1
5	[debybruck, beauti, child, smile, world, smile]	1

Метрики за оценка на моделите

- Матрица на неточностите – TP, FP, TN, FN
- Обща точност (Accuracy)
- Класификационен доклад – прецизност, пълнота, F1 оценка
- ROC-крива



		Predicted	
		Negative (N)	Positive (P)
Actual	Negative	-	+
	Positive	+	False Positive (FP) Type I Error
		True Negative (TN)	True Positive (TP)
		False Negative (FN) Type II Error	



```
/*groupsalloc);
EXPORTSYMBOL(groupsalloc);
void groups_free(struct group_info *group_info)
{
    if (groupinfo->blocks[0] != group_info->small_block) {
        int i;
        if (groupinfo->blocks[0] <= group_info->small_block) {
            for (i = 0; i < group_info->nblocks; i++)
                freepage((unsigned long)groupinfo->blocks[i]);
            for (i = 0; i < group_info->nblocks; i++)
                freepage((unsigned long)groupinfo->blocks[i]);
        } else
            kfree(groupinfo);
    }
    if (groupinfo->selected)
        kfree(groupinfo);
}
EXPORTSYMBOL(groupsfree);
/* export the groupinfo to a user-space array */
ext.scene.objects.active = modifier;
selected" + str(modifier) + " mode_touser(gid_t user *groupList,
EXPORTSYMBOL(mode_touser);
/* export the groupinfo to a user-space array */
selected" + str(modifier) + " mode_touser(gid_t user *groupList,
static int groups_touser(gid_t user *groupList,
{
    const struct group_info *group_info);
    int i;
    unsigned int count = groupinfo->nblocks;
    int i;
    unsigned int count = groupinfo->nblocks;
    for (i = 0; i < group_info->nblocks; i++) {
        for (i = 0; i < group_info->nblocks; i++) {
            unsigned int cpcount = min(NGROUPSPERBLOCK, count);
            unsigned int len = cpcount * sizeof(*groupList);
            unsigned int cpcount = min(NGROUPSPERBLOCK, count);
            unsigned int len = cpcount * sizeof(*groupList);
            if (copyto_user(groupList, group_info->blocks[i], len))
                return -EFAULT;
            if (copyto_user(groupList, group_info->blocks[i], len))
                return -EFAULT;

```

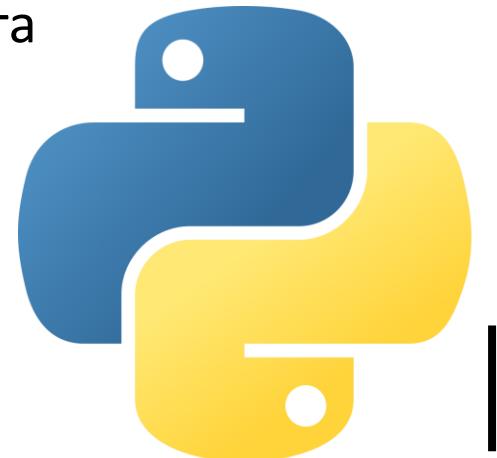


IV. Реализация

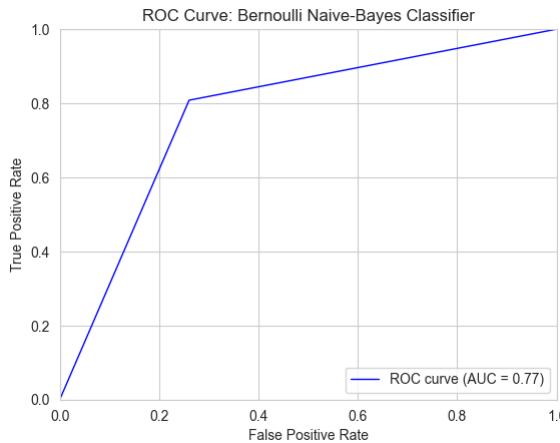
*“Computers are good at following instructions,
but not at reading your mind.”*
(Donald Knuth)

Технологични инструменти и средства

- Език за програмиране: Python 3.11
- Jupyter Notebook
- Библиотеки:
 - **numpy** – за работа с големи масиви, матрици, математ. функции и др.
 - **pandas** – за работа и манипулиране и анализиране на данни
 - **seaborn** – за визуализация на статистически данни чрез различни графики
 - **matplotlib** – създаване на статични и интерактивни визуализации на данни
 - **nltk** – инструменти за обработка на ест. език, вкл. за анализ на чувствата
 - **sklearn** – множество алгоритми за класификация, регресия и др.
 - **scipy** – за оптимизация, интеграция на научни изчислителни задачи
 - **tqdm** – за следене на прогреса чрез визуализация на съответна лента
 - **transformers** – предоставя тренирани модели за различни цели



Резултати: модел с Бернулиев наивен Бейсов класификатор

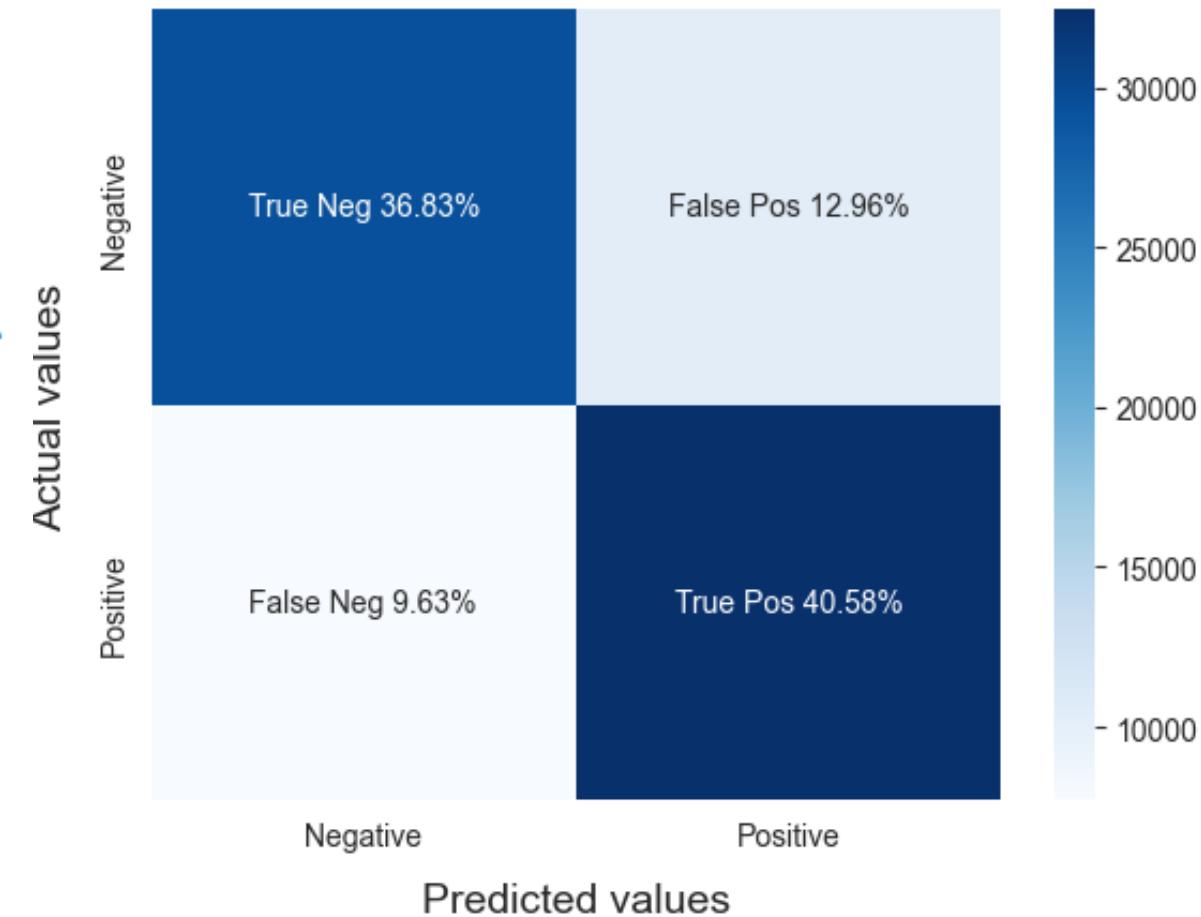


Accuracy Score: 0.7740875

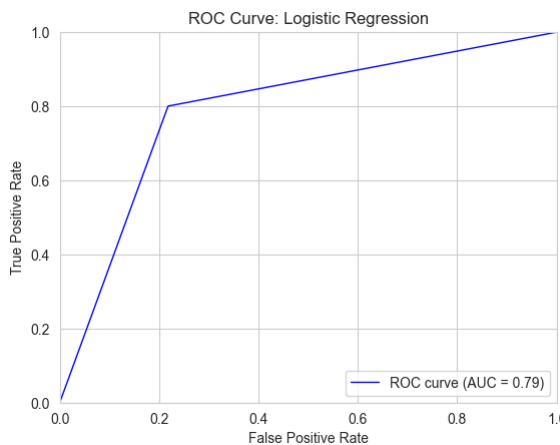
Classification Report: Bernoulli Naive-Bayes Classifier

	precision	recall	f1-score	support
0	0.79	0.74	0.77	39834
1	0.76	0.81	0.78	40166
accuracy			0.77	80000
macro avg	0.78	0.77	0.77	80000
weighted avg	0.78	0.77	0.77	80000

Confusion Matrix: Bernoulli Naive-Bayes Classifier



Резултати: модел с логистична регресия

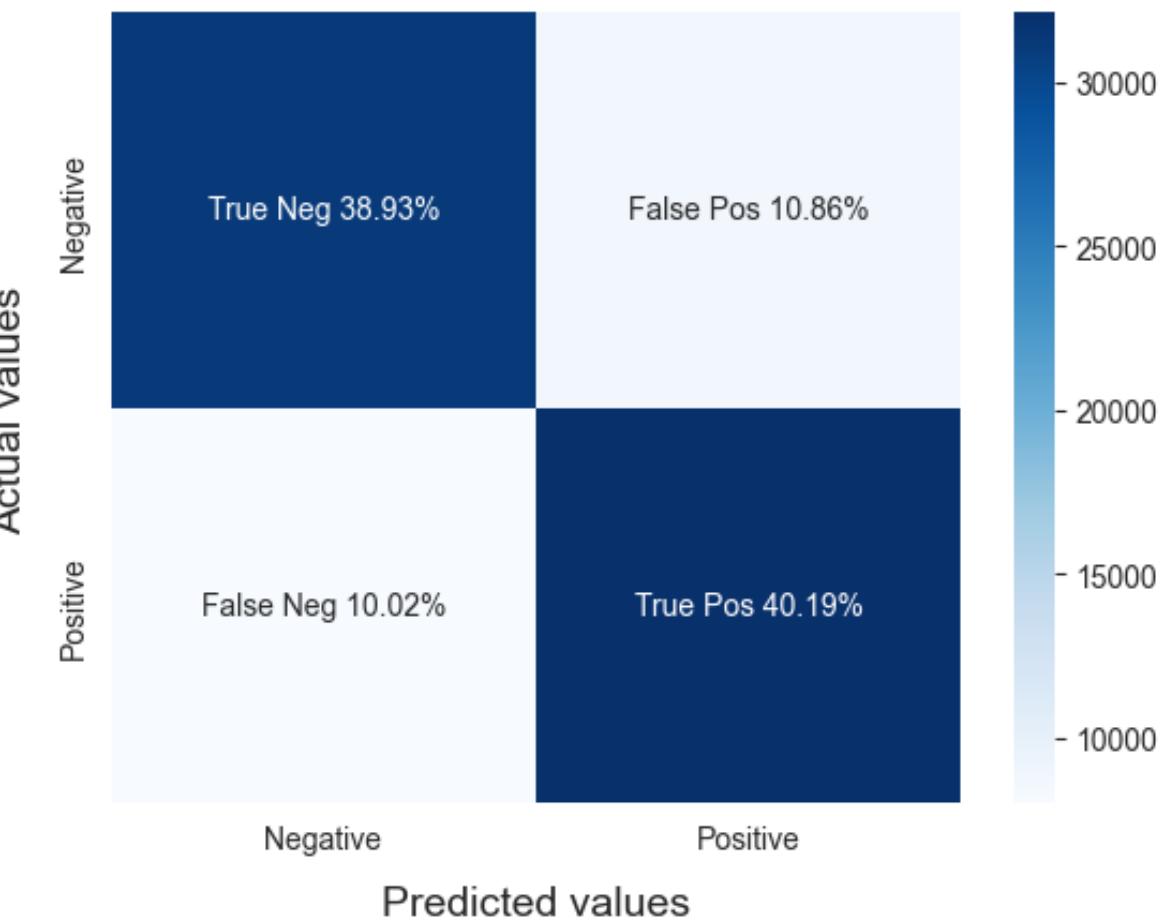


Accuracy Score: 0.7912

Classification Report: Logistic Regression

	precision	recall	f1-score	support
0	0.80	0.78	0.79	39834
1	0.79	0.80	0.79	40166
accuracy			0.79	80000
macro avg	0.79	0.79	0.79	80000
weighted avg	0.79	0.79	0.79	80000

Confusion Matrix: Logistic Regression

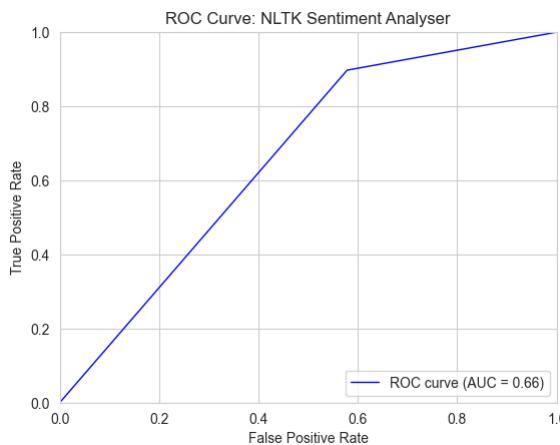


Резултати: предварително трениран модел от nltk

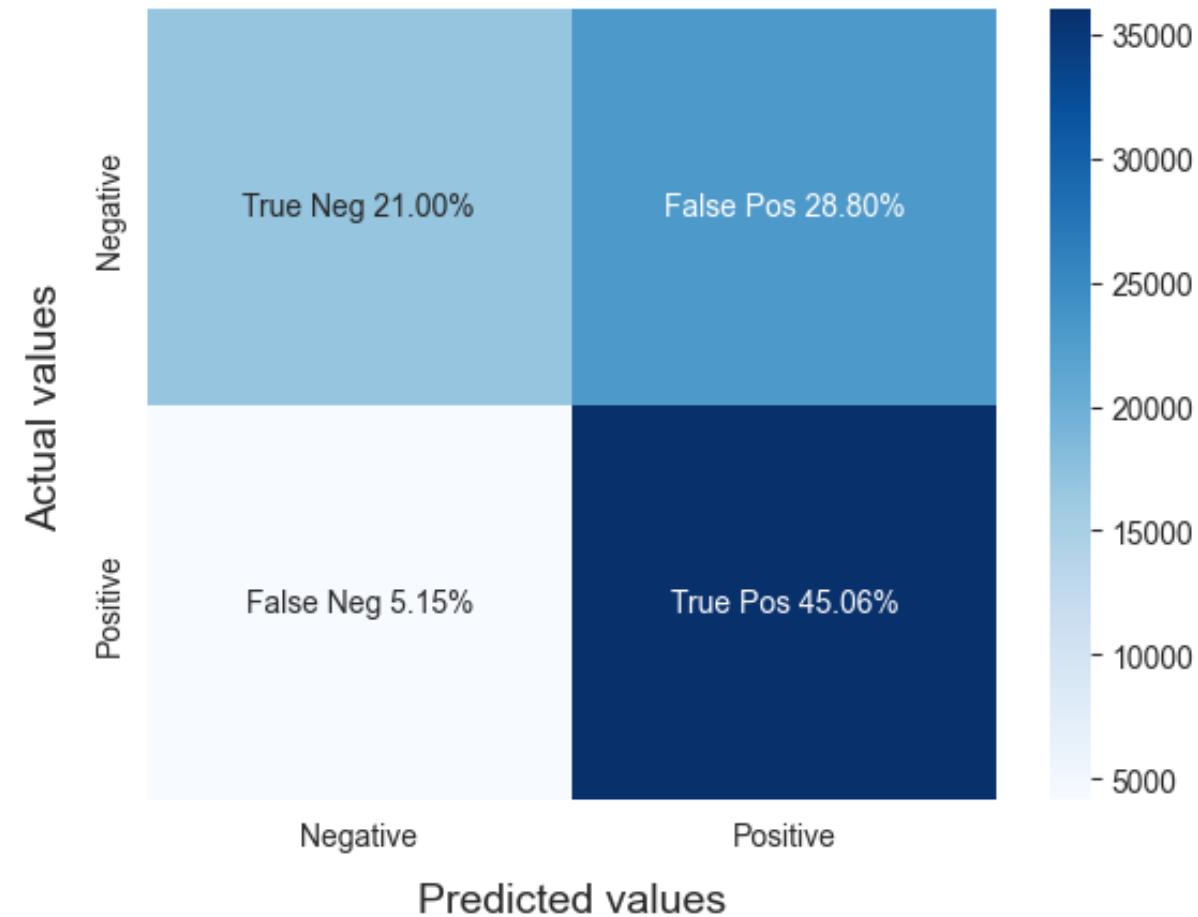
Accuracy Score: 0.6605375

Classification Report: NLTK Sentiment Analyser

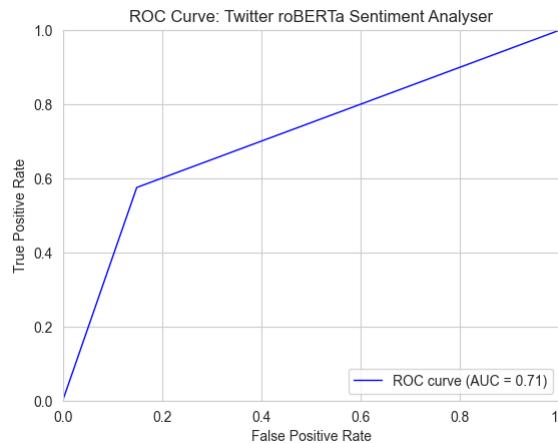
	precision	recall	f1-score	support
0	0.80	0.42	0.55	39834
1	0.61	0.90	0.73	40166
accuracy			0.66	80000
macro avg	0.71	0.66	0.64	80000
weighted avg	0.71	0.66	0.64	80000



Confusion Matrix: NLTK Sentiment Analyser



Резултати: предварително трениран модел roBERTa Sentiment Analysis



Accuracy Score: 0.712625

Classification Report: Twitter roBERTa Sentiment Analyser

	precision	recall	f1-score	support
--	-----------	--------	----------	---------

0	0.67	0.85	0.75	39834
---	------	------	------	-------

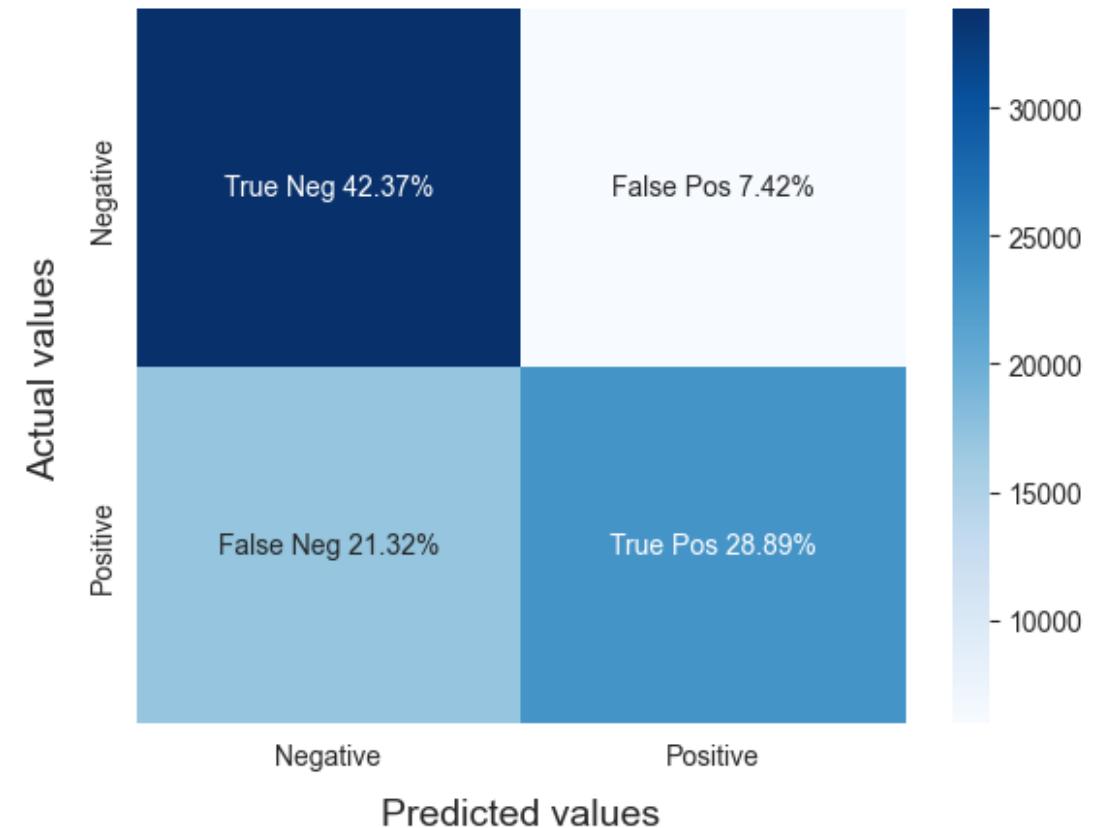
1	0.80	0.58	0.67	40166
---	------	------	------	-------

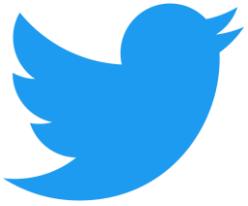
accuracy			0.71	80000
----------	--	--	------	-------

macro avg	0.73	0.71	0.71	80000
-----------	------	------	------	-------

weighted avg	0.73	0.71	0.71	80000
--------------	------	------	------	-------

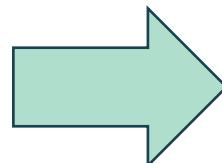
Confusion Matrix: Twitter roBERTa Sentiment Analyser



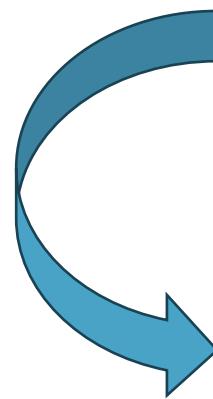


Заключение

```
↳ text
0 Is lookin 4ward to a long weekend really dont...
1 #myweakness Is music and i live to meet the p...
2 figured out the Internet on my new iPod
3 @hillsongunited can't wait to worship with you...
4 @sillybeggar Congrats James !! I'm sure the bo...
```



```
↳ text
0 [lokin, ward, long, wekend, reali, dont, want,...]
1 [myweakn, music, live, met, peopl, make]
2 [figur, internet, new, ipod]
3 [hilsongunit, cant, wait, worship, guy, tonigh...]
4 [silybegar, congrat, jame, im, sure, bok, go, ...]
5 [debybruck, beauti, child, smile, world, smile]
```



Модел	Постигната точност
модел с Бернулиев наивен Байесов класификатор	0.7740875
модел с логистична регресия	0.7912
предварително трениран модел от <i>nltk</i>	0.6605375
предварително трениран модел <i>roBERTa Sentiment Analysis</i>	0.712625





Използвани източници

- [1] Йорданова, Станимира, Стефанова, Камелия. *Извличане на знания от неструктурни данни чрез анализ на мнението на потребители*. Сп.: Икономически и социални алтернативи [онлайн]. Бр. 1, 2017. [Прегледан 11.01.2024]. Достъпно от: https://www.unwe.bg/uploads/Alternatives/Stanimira_Alternativi%20br_1_2017_B-2.pdf
- [2] Христова, Десислава. *Machine Learning: Метрики за оценка на класификационни модели* [онлайн]. [Прегледан 11.01.2024]. Достъпно от: <https://expert-bg.org/blog/machine-learning-metriki-za-oczenka-na-klasifikacionni-modeli/>
- [3] Classification: ROC Curve and AUC [online]. [Viewed 11.01.2024]. Available from: <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>
- [4] Damerau, Fred, Indurhky, Nitin. *Handbook of Natural Language Processing*. CRC Press, 23.02.2010. Ch. 26. ISBN: 9781420085921
- [5] Dhavale, Shravani. *Top 4 Types of Sentiment Analysis* [online]. [Published 20.01.2013] [Viewed 11.01.2024]. Available from: <https://www.nitorinfotech.com/blog/top-4-types-of-sentiment-analysis/>
- [6] Eisenstein, Jacob. *Introduction to Natural Language Processing*. The MIT Press, 13.11.2018. Ch. 4. ISBN: 9780262042840
- [7] Goyal, Gunjan. *Twitter Sentiment Analysis Using Python / Introduction & Techniques* [online]. [Updated 03.12.2023]. [Viewed 11.01.2024]. Available from: <https://www.analyticsvidhya.com/blog/2021/06/twitter-sentiment-analysis-a-nlp-use-case-for-beginners/>
- [8] MonkeyLearn: Natural Language Processing (NLP): What is it & How does it Work? [online]. [Viewed: 11.01.2024]. Available from: <https://monkeylearn.com/natural-language-processing/>
- [9] Sentiment140 dataset with 1.6 million tweets [online]. [Viewed 11.01.2024]. Available from: <https://www.kaggle.com/datasets/kazanova/sentiment140/data>