Московский государственный технический университет им. Н.Э. Баумана.

Факультет «Информатика и управление»

Кафедра ИУ5. Курс «Методы машинного обучения»

Отчет по лабораторной работе №5

«Предобработка текста»

Выполнил:
студент группы ИУ5-23Б
Белоусов Евгений
Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю. Е.
Подпись и дата:

Москва, 2022 г.

Описание задания

1. Для произвольного предложения или текста решите следующие задачи:
   - Токенизация.
   - Частеречная разметка.
   - Лемматизация.
   - Выделение (распознавание) именованных сущностей.
   - Разбор предложения.
2. Для произвольного набора данных, предназначенного для классификации текстов, решите задачу классификации текста двумя способами:
   - Способ 1. На основе CountVectorizer или TfidfVectorizer.
   - Способ 2. На основе моделей word2vec или Glove или fastText.
   - Сравните качество полученных моделей.

Для поиска наборов данных в поисковой системе можно использовать ключевые слова "datasets for text classification".

```
1 import numpy as np
2 import pandas as pd
```

```
1 !unzip /content/drive/MyDrive/Colab_data/MMO/fake_news.zip
```

```
Archive:  /content/drive/MyDrive/Colab_data/MMO/fake_news.zip
replace news_articles.csv? [y]es, [n]o, [A]ll, [N]one, [r]ename: n
```

```
1 data = pd.read_csv('news_articles.csv')
2 data.head()
```

	author	published	title	text	language	site_
0	Barracuda Brigade	2016-10-26T21:41:00.000+03:00	muslims busted they stole millions in govt ben...	print they should pay all the back all the mon...	english	100percentfedup.c
1	reasoning with facts	2016-10-29T08:47:11.259+03:00	re why did attorney general loretta lynch plea...	why did attorney general loretta lynch plead t...	english	100percentfedup.c
2	Barracuda Brigade	2016-10-31T01:41:49.479+02:00	breaking weiner cooperating with fbi on hillar...	red state \nfox news sunday reported this mor...	english	100percentfedup.c
3	Fed Up	2016-11-01T05:22:00.000+02:00	pin drop speech by father of daughter kidnappe...	email kayla mueller was a prisoner and torture...	english	100percentfedup.c
4	Fed Up	2016-11-01T21:56:00.000+02:00	fantastic trumps point plan	email healthcare reform to make	english	100percentfedup.c

```
1 data = data[data['language']=='english']
```

```
1 data['language'].unique()
```

```
array(['english'], dtype=object)
```

```
1 data.keys()
```

```
Index(['author', 'published', 'title', 'text', 'language', 'site_url',
       'main_img_url', 'type', 'label', 'title_without_stopwords',
       'text_without_stopwords', 'hasImage'],
      dtype='object')
```

```
data = data.drop(columns = ['author', 'published', 'title', 'language', 'site_url',
       'main_img_url', 'type', 'title_without_stopwords',
       'text_without_stopwords', 'hasImage'])
```

```
sentence = data.iloc[0]['text']
```

```
sentence
```

```
'print they should pay all the back all the money plus interest the entire family a
nd everyone who came in with them need to be deported asap why did it take two year
s to bust them \nhere we go again another group stealing from the government and ta
xpayers a group of somalis stole over four million in government benefits over just
```

токенизация

```
import nltk
nltk.download('punkt')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
True
```

```
from nltk import tokenize
nltk_tk_1 = nltk.WordPunctTokenizer()
nltk_tk_1.tokenize(sentence)
```

```
 'why',
 'did',
 'it',
 'take',
 'two',
 'years',
 'to',
 'bust',
 'them',
 'here',
 'we',
 'go',
 'again',
 'another',
 'group',
 'stealing',
 'from',
 'the',
 'government',
 'and',
 'taxpayers',
 'a',
 'group',
 'of',
 'somalis'
```

```
 somalis',
'stole',
'over',
'four',
'million',
'in',
'government',
'benefits',
'over',
'just',
'months',
'weve',
'reported',
'on',
'numerous',
'cases',
'like',
'this',
'one',
'where',
'the',
'muslim',
'refugeesimmigrants',
'commit',
'fraud',
'by',
'scamming',
'our',
'systemits',
'way',
'out',
'of',
'control',
'more',
'related']
```

```
# Токенизация по предложениям
nltk_tk_sents = nltk.tokenize.sent_tokenize(sentence)
print(len(nltk_tk_sents))
nltk_tk_sents
```

```
1
['print they should pay all the back all the money plus interest the entire family a
```

частеречная разметка

```
from spacy.lang.en import English
import spacy
nlp = spacy.load('en_core_web_sm')
```

```
spacy_text1 = nlp(sentence)
for token in spacy_text1:
   print('{} - {} - {}'.format(token.text, token.pos_, token.dep_))
```

```
to - PART - aux
bust - VERB - xcomp
them - PRON - dobj
```

```
them - PRON - dobj

 - SPACE -
here - ADV - advmod
we - PRON - nsubj
go - VERB - ROOT
again - ADV - advmod
another - DET - det
group - NOUN - nsubj
stealing - VERB - acl
from - ADP - prep
the - DET - det
government - NOUN - pobj
and - CCONJ - cc
taxpayers - VERB - conj
a - DET - det
group - NOUN - nsubj
of - ADP - prep
somalis - ADJ - pobj
stole - VERB - ROOT
over - ADP - quantmod
four - NUM - compound
million - NUM - dobj
in - ADP - prep
government - NOUN - compound
benefits - NOUN - pobj
over - ADP - prep
just - ADV - advmod
  - SPACE -
months - NOUN - pobj

 - SPACE -
we - PRON - nsubj
ve - VERB - aux
reported - VERB - ROOT
on - ADP - prep
numerous - ADJ - amod
cases - NOUN - pobj
like - SCONJ - prep
this - DET - det
one - NOUN - pobj
where - ADV - advmod
the - DET - det
muslim - ADJ - amod
refugeesimmigrants - NOUN - nsubj
commit - VERB - relcl
fraud - NOUN - dobj
by - ADP - prep
scamming - VERB - pcomp
our - DET - poss
systemits - NOUN - dobj
way - NOUN - npadvmod
out - SCONJ - prep
of - ADP - prep
control - NOUN - pobj
more - ADV - advmod
related - ADJ - amod
```

лемматизация

```
1 for token in spacy_text1:
2       print(token, token.lemma, token.lemma_)
```

```
 962983613142996970

here 411390626470654571 here
we 561228191312463089 -PRON-
go 8004577259940138793 go
again 4502205900248518970 again
another 7270490914741406701 another
group 16767868930224892138 group
stealing 11134437368562332972 steal
from 7831658034963690409 from
the 7425985699627899538 the
government 3625794390087546215 government
and 2283656566040971221 and
taxpayers 14995217432718161090 taxpayer
a 11901859001352538922 a
group 16767868930224892138 group
of 886050111519832510 of
somalis 4433042178246960311 somalis
stole 11134437368562332972 steal
over 5456543204961066030 over
four 13283271314760746512 four
million 17365054503653917826 million
in 3002984154512732771 in
government 3625794390087546215 government
benefits 12488923932015381607 benefit
over 5456543204961066030 over
just 7148522813498185515 just
  8532415787641010193
months 14920206370424861916 month

 962983613142996970

we 561228191312463089 -PRON-
ve 14692702688101715474 have
reported 2729752284408055516 report
on 5640369432778651323 on
numerous 9257680907642490936 numerous
cases 8110129090154140942 case
like 18194338103975822726 like
this 1995909169258310477 this
one 17454115351911680600 one
where 16318918034475841628 where
the 7425985699627899538 the
muslim 123001378226201854 muslim
refugeesimmigrants 14044176667317729169 refugeesimmigrant
commit 14584963062971571048 commit
fraud 15577962042453312152 fraud
by 16764210730586636600 by
scamming 3255926512416114527 scamme
our 561228191312463089 -PRON-
systemits 15356233622089482303 systemit
way 6878210874361030284 way
out 1696981056005371314 out
of 886050111519832510 of
control 572204754761179701 control
```

```
    more 2160362229054775535 more
    related 9648789318421011716 related
```

выделение именованных сущностей

```
1 for ent in spacy_text1.ents:
2     print(ent.text, ent.label_)
```

```
    two years DATE
    somalis NORP
    over four million CARDINAL
    just  months DATE
    muslim NORP
```

```
1 from spacy import displacy
2 displacy.render(spacy_text1, style='ent', jupyter=True)
```

print they should pay all the back all the money plus interest the entire family and everyone who came in

with them need to be deported asap why did it take two years **DATE** to bust them

here we go again another group stealing from the government and taxpayers a group of somalis

```
1 print(spacy.explain("NORP"))
```

```
    Nationalities or religious or political groups
```

```
1 print(spacy.explain("DATE"))
```

```
    Absolute or relative dates or periods
```

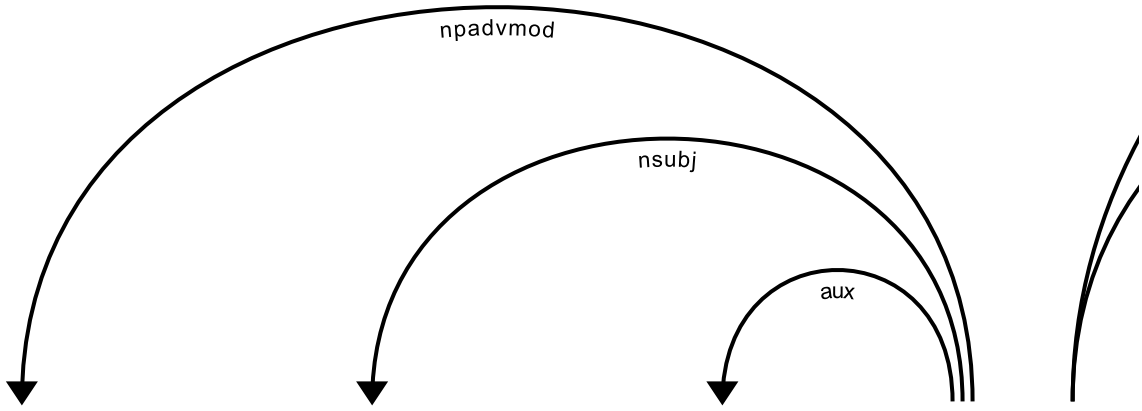
```
1 print(spacy.explain("CARDINAL"))
```

```
    Numerals that do not fall under another type
```

разбор предложения

```
1 from spacy import displacy
```

```
1 displacy.render(spacy_text1, style='dep', jupyter=True)
```



Классификация

```
1 import sklearn
2 from sklearn.neighbors import KNeighborsClassifier
3 from sklearn.model_selection import train_test_split
4 from sklearn.metrics import classification_report
5 from sklearn.linear_model import LogisticRegression
6 from sklearn.feature_extraction.text import TfidfVectorizer
7 from sklearn.pipeline import Pipeline
8 from sklearn.model_selection import cross_val_score
```

```
1 data = data.dropna()
```

```
1 tfidfv = TfidfVectorizer(ngram_range=(1,3))
2 tfidf_ngram_features = tfidfv.fit_transform(data['text'])
3 tfidf_ngram_features
```

```
    <1972x1161155 sparse matrix of type '<class 'numpy.float64'>'
            with 2234986 stored elements in Compressed Sparse Row format>
```

```
1 y = data['label'].values
```

```
1 cross_val_score(LogisticRegression(C=3.0), tfidf_ngram_features, y, scoring='accuracy',
```

```
    0.582660584555076
```

```
1 cross_val_score(KNeighborsClassifier(n_neighbors=5), tfidf_ngram_features, y, scoring='
```

```
    0.45793411765431585
```

```
1 !pip install fasttext
```

```
    Collecting fasttext
      Downloading fasttext-0.9.2.tar.gz (68 kB)
         |████████████████████████████████| 68 kB 3.4 MB/s
    Collecting pybind11>=2.2
      Using cached pybind11-2.9.2-py2.py3-none-any.whl (213 kB)
    Requirement already satisfied: setuptools>=0.7.0 in /usr/local/lib/python3.7/dist-pa
    Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages (from
    Building wheels for collected packages: fasttext
      Building wheel for fasttext (setup.py) ... done
      Created wheel for fasttext: filename=fasttext-0.9.2-cp37-cp37m-linux_x86_64.whl si
      Stored in directory: /root/.cache/pip/wheels/4e/ca/bf/b020d2be95f7641801a6597a29c8
    Successfully built fasttext
    Installing collected packages: pybind11, fasttext
    Successfully installed fasttext-0.9.2 pybind11-2.9.2
```

```
1 !gunzip /content/drive/MyDrive/Colab_data/MMO/cc.en.300.bin.gz
```

```
1 import fasttext
2 ft = fasttext.load_model('/content/drive/MyDrive/Colab_data/MMO/cc.en.300.bin')
```

```
    Warning : `load_model` does not return WordVectorModel or SupervisedModel any more,
```

```
1 matrix_ft = []
2 for text in data['text'].values:
3  matrix_ft.append(ft[text])
4 matrix_ft = np.array(matrix_ft)
```

```
1 matrix_ft
```

```
array([[-0.00098612, -0.00476133, -0.0075634 , ...,  0.02601686,
         0.00984338, -0.0103699 ],
       [ 0.00010597, -0.0002962 , -0.00247709, ...,  0.02275638,
         0.00959554, -0.00702063],
       [ 0.00171071, -0.00028555, -0.00728132, ...,  0.02532279,
         0.01022426, -0.00673363],
       ...,
       [ 0.00287685, -0.00164437, -0.01062832, ...,  0.02614961,
         0.00533141, -0.00310368],
       [ 0.00188689, -0.0037404 , -0.00514202, ...,  0.02265813,
         0.00613548, -0.00226545],
       [-0.0131774 , -0.0109064 , -0.00970652, ...,  0.02438283,
         0.01350168, -0.00817819]], dtype=float32)
```

```
1 cross_val_score(LogisticRegression(C=3.0), matrix_ft, y, scoring='accuracy', cv=3).mean
```

```
0.6186613186030265
```

```
1 cross_val_score(KNeighborsClassifier(n_neighbors=5), matrix_ft, y, scoring='accuracy',
```

```
0.5086219791844974
```