Задача тематическое моделирование

продолжаем исследование датасета с твитами

Скачиваем датасет (источник): положительные, отрицательные.

или можно через ноутбук

!wget https://www.dropbox.com/s/fnpq3z4bcnoktiv/positive.csv (https://www.dropbox.com/s/fnpq3z4bcnoktiv/positi
!wget https://www.dropbox.com/s/r6u59ljhhjdg6j0/negative.csv (https://www.dropbox.com/s/r6u59ljhhjdg6j0/negati

как альтернатива можно скачать данные из Роспотребнадзора
https://zpp.rospotrebnadzor.ru/Forum/Appeals (https://zpp.rospotrebnadzor.ru/Forum/Appeals)
для этого берём ноутбук parse_rospotrebnadzor.ipynb
устанавливаем количество скачанных страниц больше не 50-сят хотябы 500 и для анализа берём только вс

что надо сделать

1. объединить в одну выборку (это только для твитов), для роспотребнадзора сформировать датасет из в
2. провести исследование и выявить тематики о которых говорят в твитах (для твитов), а для роспотребна
3. сделать визуализацию кластеров тематик
4. проинтерпритировать получившиеся тематики

In [474]:

```
import pandas as pd
import numpy as np
import re
import os

from sklearn.metrics import *
from sklearn.model_selection import train_test_split
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LogisticRegression
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer

import matplotlib.pyplot as plt
import seaborn as sns

import nltk
from nltk import ngrams
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords

from collections import Counter

import string
from pymorphy2 import MorphAnalyzer
from stop_words import get_stop_words

import annoy
from gensim.models import Word2Vec, FastText
import pickle

from tqdm import tqdm_notebook, tqdm

import warnings

warnings.filterwarnings("ignore", category=DeprecationWarning)
```

In [475]:

```
df_lenta = pd.read_csv("lenta-ru-news.csv")
```

```
c:\program files\python37\lib\site-packages\IPython\core\interactiveshell.py:3444: DtypeWarn
y=False.
  exec(code_obj, self.user_global_ns, self.user_ns)
```

In [476]:

```
df_lenta.head()
```

Out[476]:

	url	title	
0	https://lenta.ru/news/1914/09/16/hungarnn/	1914. Русские войска вступили в пределы Венгрии	Бои у Сопоцкина и Д
1	https://lenta.ru/news/1914/09/16/lermontov/	1914. Празднование столетия М.Ю. Лермонтова от...	Министерство народно
2	https://lenta.ru/news/1914/09/17/nesteroff/	1914. Das ist Nesteroff!	Штабс-капитан П. Н.
3	https://lenta.ru/news/1914/09/17/bulldogn/	1914. Бульдог-гонец под Льежем	Фотограф-корреспон
4	https://lenta.ru/news/1914/09/18/zver/	1914. Под Люблином пойман швабский зверь	Лица, приехавшие в Ва

In [477]:

```
df_lenta.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 800975 entries, 0 to 800974
Data columns (total 6 columns):
 #   Column  Non-Null Count   Dtype 
---  ------  --------------   ----- 
 0   url     800975 non-null  object
 1   title   800975 non-null  object
 2   text    800970 non-null  object
 3   topic   738973 non-null  object
 4   tags    773756 non-null  object
 5   date    800975 non-null  object
dtypes: object(6)
memory usage: 36.7+ MB
```

In [478]:

```
df_lenta = df_lenta.dropna(axis=0, how='any')
```

In [479]:

```
df_lenta.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 712654 entries, 0 to 739175
Data columns (total 6 columns):
 #   Column  Non-Null Count   Dtype 
---  ------  --------------   ----- 
 0   url     712654 non-null  object
 1   title   712654 non-null  object
 2   text    712654 non-null  object
 3   topic   712654 non-null  object
 4   tags    712654 non-null  object
 5   date    712654 non-null  object
dtypes: object(6)
memory usage: 38.1+ MB
```

In [480]:

```
data = pd.DataFrame( columns=['title','topic','text','tags'])
```

In [481]:

```
data['title'] = df_lenta.title
```

In [482]:

```
data['topic'] = df_lenta.topic
```

In [483]:

```
data['text'] = df_lenta.text
```

In [484]:

```
data['tags'] = df_lenta.tags
```

In [485]:

```
data.sample(3)
```

Out[485]:

	title	topic	
125889	Старьевщик-любитель устроил взрыв в Каире	Мир	В воскресенье в Каире произошел в
123453	В Китае утвержден девиз Олимпиады-2008	Спорт	Организационный комитет по проведени
599064	Звезды YouTube сравнили «Евровидение» с Олимпи...	Интернет и СМИ	Популярные американские YouTube-бло

In [486]:

```
data.topic.value_counts().head(15)
```

Out[486]:

```
Россия                155004
Мир                   136620
Экономика              76423
Спорт                  57894
Культура               53530
Наука и техника        53136
Бывший СССР            51370
Интернет и СМИ         44421
Из жизни               27513
Дом                    21734
Силовые структуры      11223
Ценности                7581
Бизнес                  7375
Путешествия             6370
69-я параллель          1268
Name: topic, dtype: int64
```

Для ускорения вычислений оставим тексты из 9 тем:

In [487]:

```
data.topic.unique()
```

Out[487]:

```
array(['Библиотека', 'Россия', 'Мир', 'Экономика', 'Интернет и СМИ',
       'Спорт', 'Культура', 'Из жизни', 'Наука и техника', 'Бывший СССР',
       'Дом', 'Сочи', 'ЧМ-2014', 'Путешествия', 'Силовые структуры',
       'Ценности', 'Легпром', 'Бизнес', 'МедНовости', 'Оружие',
       '69-я параллель', 'Культпросвет ', 'Крым'], dtype=object)
```

In [488]:

```
topics = ['Россия', 'Мир','Экономика', 'Спорт', 'Культура', 'Путешествия',
          'Наука и техника','Дом', 'Силовые структуры' ]

data = data[data.topic.isin(topics)]
data.dropna(inplace=True)
len(data)
```

Out[488]:

```
571934
```

Предобработка

Все модели, с которыми мы будем работать далее, работают с предобработанными текстами, поэтому спер

In [489]:

```
import re
import numpy as np
from nltk.corpus import stopwords
from tqdm.notebook import tqdm
from multiprocessing import Pool
from pymystem3 import Mystem
```

In [490]:

```
import pymorphy2
morph = pymorphy2.MorphAnalyzer()
```

In [491]:

```
words_regex = re.compile('\w+')

def find_words(text, regex = words_regex):
    tokens =  regex.findall(text.lower())
    return [w for w in tokens if w.isalpha() and len(w) >= 3]


stopwords_list = stopwords.words('russian')

# mystem = Mystem()
# def lemmatize(words, lemmer = mystem, stopwords = stopwords_list):
#     lemmas = lemmer.lemmatize(' '.join(words))
#     return [w for w in lemmas if not w in stopwords
#             and w.isalpha()]

def lemmatize(words, lemmer = morph, stopwords = stopwords_list):
    lemmas = [lemmer.parse(w)[0].normal_form for w in words]
    return [w for w in lemmas if not w in stopwords
            and w.isalpha()]

def preprocess(text):
    return (lemmatize(find_words(text)))
```

In [492]:

```
data.text.iloc[1]
```

Out[492]:

```
'В зале игровых автоматов в третьем ярусе подземного комплекса "Охотный ряд" на Манежной пло
е менее четырех человек, 20 человек ранены. Однако уточненная оценка числа пострадавших в ре
нии от 21.15 со ссылкой на источники в ГУВД Москвы, говорит только о 30 раненых, в том числе
этого взрыва может составить до ста человек. По данным РИА "Новости", Боткинская больница, И
ько на прием пострадавших. С места происшествия запросили 20 машин скорой помощи. Из торгово
ристический акт, связанный с последними событиями в Дагестане, однако, по сообщению ОРТ, не
причнах взрыва и количестве жертв представители УФСБ называть не торопятся.'
```

In [493]:

```
print(preprocess(data.text.iloc[1]))
```

```
['зал', 'игровой', 'автомат', 'третий', 'ярус', 'подземный', 'комплекс', 'охотный', 'ряд', '
в', 'данные', 'фсб', 'погибнуть', 'менее', 'четыре', 'человек', 'человек', 'ранить', 'однако
ь', 'достигнуть', 'сто', 'человек', 'агентство', 'итар', 'тасс', 'сообщение', 'ссылка', 'ист
й', 'состояние', 'однако', 'число', 'пострадать', 'результат', 'это', 'взрыв', 'мочь', 'сост
'институт', 'склифосовский', 'градский', 'горбольница', 'работать', 'приём', 'пострадать', '
ентр', 'эвакуировать', 'всё', 'посетитететь', 'среди', 'причина', 'произойти', 'называть', '
о', 'сообщение', 'орт', 'исключаться', 'версия', 'авария', 'взорваться', 'игровой', 'автомат
ь', 'уфсб', 'называть', 'торопиться']
```

Ниже мы оставим только случайные 10,000 текстов из нашей коллекции, чтобы ускорить вычисления:

In [494]:

```
data = data.sample(10000)
data.topic.value_counts()
```

Out[494]:

```
Россия               2750
Мир                  2316
Экономика            1380
Спорт                1043
Наука и техника       908
Культура              895
Дом                   420
Силовые структуры     174
Путешествия           114
Name: topic, dtype: int64
```

In [495]:

```
preprocessed_text = list(tqdm(map(preprocess, data['text']), total=len(data)))
```

100% 10000/10000 [03:29<00:00, 53.22it/s]

In [496]:

```
data['text'] = preprocessed_text
data.sample(3)
```

Out[496]:

	title	topic	
325576	Российский теннисист дисквалифицирован за допинг	Спорт	[международный, теннисный, федерация, itf,
678927	Следователи попросили поместить Серебренникова...	Культура	[следственный, комитет, россия, скр, попрос
508571	SMS-рассылки сотовых операторов предложили при...	Экономика	[оператор, сотовый, связь, смочь, рассылат

In [497]:

```
data.sample(3)
```

Out[497]:

	title	topic	
402252	Куба построит электростанцию на сахарном трост...	Экономика	[куба, открыть, первый, электростанция, ка
603687	ЦБ начал проверку данных об иностранных счетах...	Экономика	[банк, россия, начать, проверка, информаци
173052	Замначальника Дальневосточной таможни отпустил...	Россия	[первый, заместитель, начальник, дальневост

Модель LDA

Первая модель, которую мы рассмотрим, LDA - латентное размещение Дирихле. Воспользуемся реализаци

In [498]:

```
from gensim.models import *
from gensim import corpora
```

In [499]:

```
dictionary = corpora.Dictionary(data['text'])

dictionary.filter_extremes(no_below = 10, no_above = 0.9, keep_n=None) # игнорируем слова, к
dictionary.save('lenta.dict')
```

Векторизуем документы:

In [500]:

```
corpus = [dictionary.doc2bow(text) for text in data['text']]
corpora.MmCorpus.serialize('lenta.model', corpus)
```

Теперь можем обучать модель:

In [501]:

```
%time lda = ldamodel.LdaModel(corpus, id2word=dictionary, num_topics=9, chunksize=50, update
```

```
Wall time: 8.57 s
```

Посмотрим на получившиеся темы:

In [502]:

```
topics
```

Out[502]:

```
['Россия',
 'Мир',
 'Экономика',
 'Спорт',
 'Культура',
 'Путешествия',
 'Наука и техника',
 'Дом',
 'Силовые структуры']
```

In [503]:

```
lda.show_topics(num_topics=9, num_words=20, formatted=True)
```

Out[503]:

```
[(0,
  '0.032*"год" + 0.023*"тысяча" + 0.015*"метр" + 0.015*"проект" + 0.013*"москва" + 0.012*"ст
0.009*"квадратный" + 0.008*"хороший" + 0.007*"новый" + 0.007*"квартира" + 0.007*"первый" + 0
жимость"'),
 (1,
  '0.018*"человек" + 0.012*"сообщать" + 0.010*"находиться" + 0.009*"произойти" + 0.009*"резу
0.007*"город" + 0.007*"полиция" + 0.007*"задержать" + 0.007*"сотрудник" + 0.006*"сообщить" +
*"агентство" + 0.005*"около"'),
 (2,
  '0.041*"компания" + 0.024*"миллион" + 0.023*"доллар" + 0.022*"год" + 0.014*"фильм" + 0.012
+ 0.008*"проект" + 0.008*"картина" + 0.007*"акция" + 0.007*"финансовый" + 0.007*"предприятие
0.006*"также"'),
 (3,
  '0.018*"страна" + 0.015*"президент" + 0.013*"россия" + 0.012*"сша" + 0.011*"который" + 0.0
+ 0.007*"власть" + 0.007*"сообщать" + 0.006*"год" + 0.006*"партия" + 0.006*"свой" + 0.005*"г
нять"'),
 (4,
  '0.022*"игра" + 0.022*"матч" + 0.016*"команда" + 0.015*"клуб" + 0.012*"сборная" + 0.011*"ч
ссия" + 0.009*"который" + 0.008*"стать" + 0.008*"сезон" + 0.008*"первый" + 0.008*"российский
 (5,
  '0.046*"процент" + 0.034*"год" + 0.030*"рубль" + 0.016*"россия" + 0.014*"это" + 0.011*"дол
8*"тысяча" + 0.007*"правительство" + 0.007*"российский" + 0.007*"уровень" + 0.006*"нефть" +
с"'),
 (6,
  '0.024*"год" + 0.019*"который" + 0.012*"the" + 0.010*"это" + 0.008*"новый" + 0.008*"британ
тать" + 0.006*"также" + 0.005*"выйти" + 0.005*"первый" + 0.005*"получить" + 0.005*"весь" + 0
 (7,
  '0.023*"год" + 0.016*"это" + 0.013*"который" + 0.010*"суд" + 0.008*"дело" + 0.008*"свой" +
я" + 0.006*"бывший" + 0.006*"мочь" + 0.005*"глава" + 0.004*"однако" + 0.004*"действие" + 0.0
 (8,
  '0.011*"военный" + 0.011*"боевик" + 0.010*"заявить" + 0.010*"депутат" + 0.008*"представите
+ 0.007*"израиль" + 0.006*"украина" + 0.006*"операция" + 0.006*"сирия" + 0.006*"аль" + 0.006
*"глава" + 0.005*"выборы"')]
```

In [504]:

```
print(lda.log_perplexity(corpus))
```

```
-7.858298369096038
```

In [505]:

```


print('Персплексия: ', np.exp(lda.log_perplexity(corpus)))

coherence_model_lda = CoherenceModel(model=lda, texts=data['text'], dictionary=dictionary, c
%time coherence_lda = coherence_model_lda.get_coherence()
print('Средняя когерентность: ', coherence_lda)
```

```
Персплексия:  0.0003865308827564778
Wall time: 305 ms
Средняя когерентность:  -1.8680302400181672
```

In [ ]:

```
```

In [ ]:

```
```

In [ ]:

```
```

объединяю в одну выборку

In [506]:

```
# считываем данные и заполняем общий датасет
positive = pd.read_csv('positive.csv', sep=';', usecols=[3], names=['text'])
positive['label'] = ['positive'] * len(positive)
negative = pd.read_csv('negative.csv', sep=';', usecols=[3], names=['text'])
negative['label'] = ['negative'] * len(negative)
df = positive.append(negative)
```

In [507]:

```
def funk_del(input_txt):
    pattern = "@[\w]*"
    if re.findall(pattern, input_txt):
        return re.sub(pattern, ' ', input_txt)
    else:
        return re.sub(pattern, ' ', input_txt)
```

In [508]:

```
df.head()
```

Out[508]:

	text	label
0	@first_timee хоть я и школота, но поверь, у на...	positive
1	Да, все-таки он немного похож на него. Но мой ...	positive
2	RT @KatiaCheh: Ну ты идиотка) я испугалась за ...	positive
3	RT @digger2912: "Кто то в углу сидит и погибае...	positive
4	@irina_dyshkant Вот что значит страшилка :D\nH...	positive

In [509]:

```
stopwords_list = stopwords.words('russian')
```

In [510]:

```
print(stopwords_list)
```

```
['и', 'в', 'во', 'не', 'что', 'он', 'на', 'я', 'с', 'со', 'как', 'а', 'то', 'все', 'она', 'т
'ее', 'мне', 'было', 'вот', 'от', 'меня', 'еще', 'нет', 'о', 'из', 'ему', 'теперь', 'когда',
'до', 'вас', 'нибудь', 'опять', 'уж', 'вам', 'ведь', 'там', 'потом', 'себя', 'ничего', 'ей',
'чем', 'была', 'сам', 'чтоб', 'без', 'будто', 'чего', 'раз', 'тоже', 'себе', 'под', 'будет',
м', 'здесь', 'этом', 'один', 'почти', 'мой', 'тем', 'чтобы', 'нее', 'сейчас', 'были', 'куда'
'хоть', 'после', 'над', 'больше', 'тот', 'через', 'эти', 'нас', 'про', 'всего', 'них', 'кака
'перед', 'иногда', 'лучше', 'чуть', 'том', 'нельзя', 'такой', 'им', 'более', 'всегда', 'коне
```

In [511]:

```
df['text'] = df.text.apply(funk_del)
```

In [512]:

```
df = df.sample(10000)
```

In [513]:

```
df['token_text'] = list(tqdm(map(preprocess, df['text']), total=len(df)))
```

```
100%                                                10000/10000 [00:15<00:00, 851.84it/s]
```

In [514]:

```
df[:5]
```

Out[514]:

	text	label	token_text
105076	Да так,так называемые мысли вслух:(	negative	[называть, мысль, вслух]
50242	дожить бы до них(((завтра первый зачёт, а е...	negative	[дожить, завтра, первый, зачёт, ещё, готовиться]
23582	RT : Скоро и до замера черепов доберутся.....	negative	[скоро, замер, черепов, добраться]
31809	RT : Спасибо... В аварии я попадал и ранее,...	negative	[спасибо, авария, попадать, ранее, переворачив...
106992	музыку надо хранить у себя на носителях :) а...	positive	[музыка, хранить, носитель, сеть]

In [515]:

```
topics_dict = {0:'Экономика',
 1:'Спорт',
 2:'Культура',
 3:'Путешествия',
 4:'Наука и техника',
 5:'Дом',
 6:'Силовые структуры',
 7:'Россия',
 8: 'Мир'}
```

In [516]:

```
def topic_team(line):
    other_corpus = dictionary.doc2bow(line)
    pr = lda[other_corpus]
    topic_ = ['topic']
    max_prob = 0
    res_dict = {i:j for i,j in pr}
    max_prob = max(res_dict.values())
    for i, j in res_dict.items():
        if j == max_prob:
            topik_team = topics_dict[i]
#     print(topik_team, max_prob)
    return topik_team, max_prob
```

In [517]:

```
top = []
top = [{i:v} for i, v in df.token_text.apply(lambda x: topic_team(x))]
```

In [518]:

```
df['topik'] = [i.keys() for i in top]
```

In [519]:

```
df['topik_prob'] = [i.values() for i in top]
```

In [520]:

```
df.head(15)
```

Out[520]:

	text	label	token_tex
105076	Да так,так называемые мысли вслух:(	negative	[называть, мысль, вслух
50242	дожить бы до них(((завтра первый зачёт, а е...	negative	[дожить, завтра, первый, зачёт, ещё, готовиться
23582	RT : Скоро и до замера черепов доберутся.....	negative	[скоро, замер, черепов, добраться
31809	RT : Спасибо... В аварии я попадал и ранее,...	negative	[спасибо, авария, попадать, ранее, переворачив.
106992	музыку надо хранить у себя на носителях :) а...	positive	[музыка, хранить, носитель, сеть
37688	RT : Девочка из Индонезии просит, чтоб я ей н...	positive	[девочка, индонезия, просить, научить, русский.
4193	В среду или четверг приедет мой аппарат)))) жд...	positive	[среда, четверг, приехать, аппарат, ждать, дож.
20370	По реклама "мы за мир, за Украину без насили...	negative	[реклама, мир, украина, насилие, реклама, част.
56245	Вышла в школу первый день после болезни.Русски...	positive	[выйти, школа, первый, день, болезнь, русский,.
92936	жалко,могут замерзнуть и погибнуть..((	negative	[жалко, мочь, замёрзнуть, погибнуть
54422	Нужно идти в больницу,оттягиваю время до после...	negative	[нужно, идти, больница, оттягивать, время, пос.
21554	Мы с Настей смешные такие: только мы можем пер...	positive	[настя, смешной, мочь, перепутать, дом, заблуд.
96493	это же про нас во время файналов)) http://t.co...	positive	[это, время, файналов, http
93211	Ну это на самом деле жутко.\nВ темноте такое у...	positive	[это, дело, жутко, темнота, увидеть, http
48198	Я на радио победителем стала !!! Хаха блин оде...	positive	[радио, победитель, стать, хах, блин, оделать,.

In [521]:

```
corpus_twit = [dictionary.doc2bow(text) for text in df['token_text']]
```

In [522]:

```
from gensim.models import *
from gensim import corpora
```

In [523]:

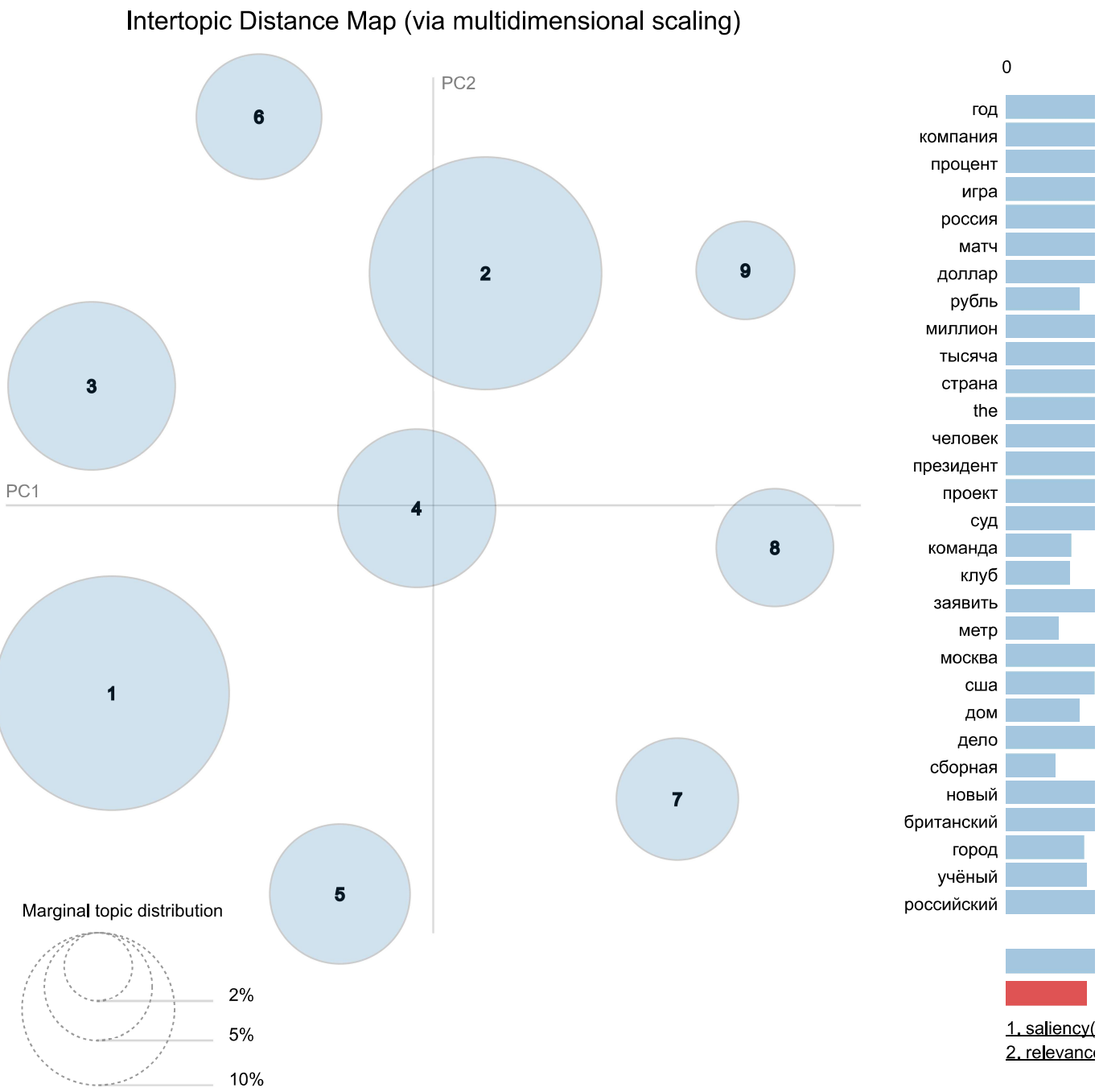
```
import pyLDAvis
import pyLDAvis.gensim_models as gensimvis

%time topicData = gensimvis.prepare(lda, corpus_twit, dictionary, mds='mmds')
pyLDAvis.display(topicData)
```

```
c:\program files\python37\lib\site-packages\pyLDAvis\_prepare.py:247: FutureWarning: In a fu
abels' will be keyword-only
  by='saliency', ascending=False).head(R).drop('saliency', 1)

Wall time: 3.83 s
```

Out[523]:



Роспотребнадзор

In [524]:

```
df = pd.read_pickle("df_rospotreb_question.pkl")
```

In [525]:

```
df[:5]
```

Out[525]:

	question
0	Купил монитор в одном из магазинов днс. Монито...
1	Здравствуйте. Купил телефон на алиэкспресс. На...
2	Добрый день. Планирую открыть магазин или лав...
3	Добрый день! Меня зовут Евгения. 25.12.2021г. ...
4	Добрый день! В ходе поиска проведения отпуска ...

In [526]:

```
stopwords_list = stopwords.words('russian')
```

In [527]:

```
pattern = r'[^а-яА-Я0-9]'
df['question'] = df['question'].apply(lambda x: re.sub(pattern, ' ', x))
```

In [528]:

```
df['token_question'] = list(tqdm(map(preprocess, df['question']), total=len(df)))
```

100% 245/245 [00:02<00:00, 103.82it/s]

In [529]:

```
top = []
top = [{i:v} for i, v in df.token_question.apply(lambda x: topic_team(x))]
```

In [530]:

```
df['topik'] = [i.keys() for i in top]
```

In [531]:

```
df['topik_prob'] = [i.values() for i in top]
```

In [543]:

```
df.sample(15)
```

Out[543]:

	question	token_question	
223	Региональный оператор в лице ООО Чистая стан...	[региональный, оператор, лицо, ооо, чистый, ст...	
228	Добрый день Можно ли через границу Абхазия ...	[добрый, день, граница, абхазия, россия, перев...	(Путе
143	Добрый день подскажите пожалуйста существуе...	[добрый, день, подсказать, пожалуйста, существ...	
147	Добрый день Оформил заказ по акции в магазине...	[добрый, день, оформить, заказ, акция, магазин...	
85	Добрый день Подскажите пожалуйста купил смес...	[добрый, день, подсказать, пожалуйста, купить,...	
210	Здравствуйте такая ситуация даже не знаю как ...	[здравствуйте, ситуация, знать, написать, авто...	
111	Добрый день В посудацентре купила саше от ко...	[добрый, день, посудацентр, купить, саша, кото...	(
0	Купил монитор в одном из магазинов днс Монито...	[купить, монитор, магазин, днс, монитор, витри...	
18	Я приобрела чехол для углового дивана в интерн...	[приобрести, чехол, угловой, диван, интернет, ...	
139	Не прошло и месяца из которого я использовал з...	[пройти, месяц, который, использовать, зарядны...	(Силовые с
124	не вывозятся длительный период ТБО так в пери...	[вывозиться, длительный, период, тбо, период, ...	
154	Ч отправила заявку диспетчеру в ДЕЗ Калининско...	[отправить, заявка, диспетчер, деза, калининск...	
230	Мы жильцы дома по улице Н Быстрых 3 г Пер...	[жилец, дом, улица, быстрый, пермь, довольный,...	
212	Здравствуйте Сегодня зашла в магазин Ветапте...	[здравствуйте, сегодня, зайти, магазин, ветапт...	
208	В квартире в ванной комнате и туалете нет цир...	[квартира, ванная, комната, туалет, циркуляция...	

In [544]:

```
print(df.question.iloc[212])
print(df.topik.iloc[212])
```

```
Здравствуйте  Сегодня зашла в магазин  Ветаптека и зоолавка  по адресу Московская область  г
ми для лотка кошкам и увидела следующую картину  на вс м стеллаже  где стоял наполнитель  бы
опрос  где ценники  продавец сказала  Ценники постоянно переоформляются  а мы пробиваем по
возможность узнать цену товара до того  как обратится за покупкой  узнает цену только на кас
dict_keys(['Дом'])
```

In [534]:

```
corpus_question = [dictionary.doc2bow(text) for text in df['token_question']]
```

In [535]:

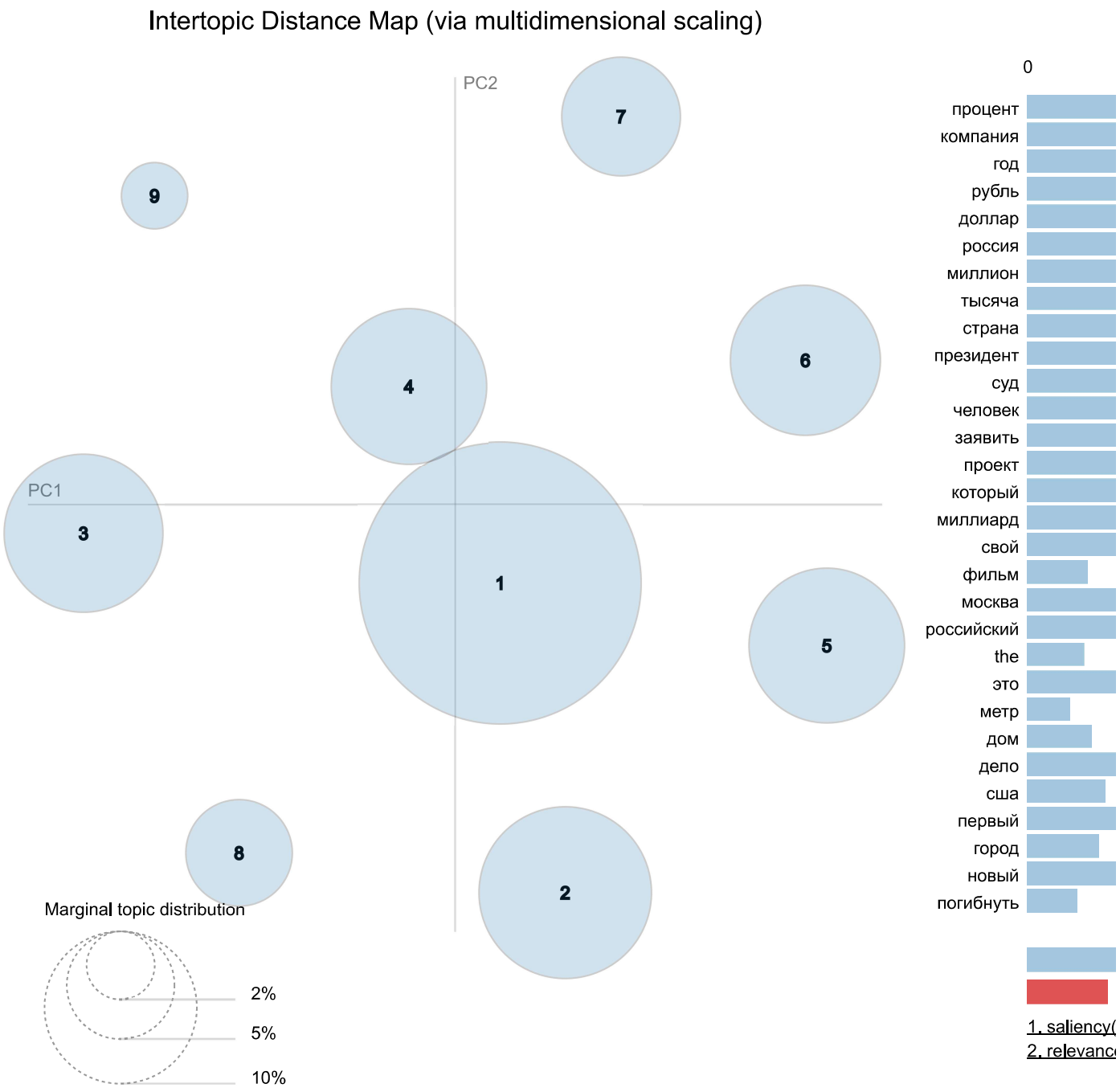
```
import pyLDAvis
import pyLDAvis.gensim_models as gensimvis

%time topicData = gensimvis.prepare(lda, corpus_question, dictionary, mds='mmds')
pyLDAvis.display(topicData)
```

```
c:\program files\python37\lib\site-packages\pyLDAvis\_prepare.py:247: FutureWarning: In a fu
abels' will be keyword-only
  by='saliency', ascending=False).head(R).drop('saliency', 1)

Wall time: 1.6 s
```

Out[535]:


Intertopic Distance Map (via multidimensional scaling)
PC2
PC1
1
2
3
4
5
6
7
8
9
Marginal topic distribution
2%
5%
10%
0
процент
компания
год
рубль
доллар
россия
миллион
тысяча
страна
президент
суд
человек
заявить
проект
который
миллиард
свой
фильм
москва
российский
the
это
метр
дом
дело
сша
первый
город
новый
погибнуть
1. saliency(
2. relevanc

In [547]:

```
topics
```

Out[547]:

```
['Россия',
 'Мир',
 'Экономика',
 'Спорт',
 'Культура',
 'Путешествия',
 'Наука и техника',
 'Дом',
 'Силовые структуры']
```

Вывод

некоторые темы содержат одни и теже токены, из-за того,
что взял для обучения названия текстов где большое содержание строк, такие как 'Россия',
'Мир', можно сказатьБ что на них модель переобучилась, в общем где-то видно правильное определение те
твитах тяжело определить тему, т.к.
большенство токенов не несут никакого смысла

In [ ]:

```

```