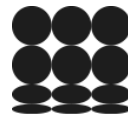
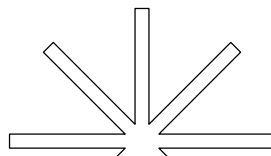


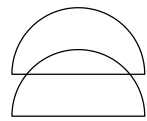
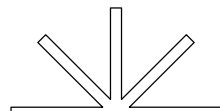
ИИ ДЛЯ АНАЛИЗА ПОЛЬЗОВАТЕЛЬСКИХ ОТВЕТОВ

Решение команды I.-.-.I



ОПИСАНИЕ ЗАДАЧИ

Разработать систему на основе ИИ,
которая анализирует список
пользовательских ответов возвращает
понятное и интерпретируемое облако
слов.

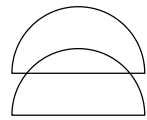
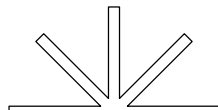


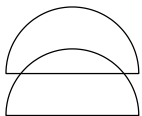
ТРЕБОВАНИЯ

Макс. время выполнения: 5 с.

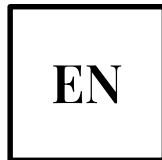
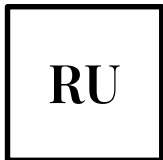
Кол-во ответов: 1000 строк

Языки: русский и английский



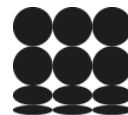
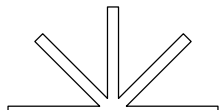


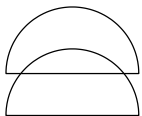
КОД РЕШЕНИЯ. ПУНКТУАЦИЯ



Наиболее полный набор знаков препинания

```
pattern = re.compile  
(r'[0-9!"#$%&\'()*+,./  
:;<=>?@[\\]\ ]^_`{|}~"\'` ,...- ]') )
```





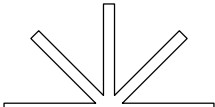
КОД РЕШЕНИЯ. ОПЕЧАТКИ



bakwc / JamSpell

```
jsp_eng = jamspell.TSpellCorrector()  
assert jsp_eng.LoadLangModel('en.bin')  
jsp = jamspell.TSpellCorrector()  
assert jsp.LoadLangModel('ru_small.bin')
```

	Errors	Top 7 Errors	Fix Rate	Top 7 Fix Rate	Broken	Speed (words/second)
JamSpell	3.25%	1.27%	79.53%	84.10%	0.64%	4854
Norvig	7.62%	5.00%	46.58%	66.51%	0.69%	395
Hunspell	13.10%	10.33%	47.52%	68.56%	7.14%	163
Dummy	13.14%	13.14%	0.00%	0.00%	0.00%	-





КОД РЕШЕНИЯ. ТОКЕНИЗАЦИЯ

RU



nlk / nlk

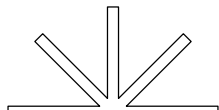
```
words = word_tokenize(jsp.FixFragment(text))
```

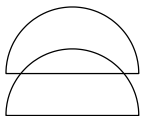
EN



explosion / spaCy

```
nlp = spacy.load('en_core_web_sm')  
doc = nlp(jsp_eng.FixFragment(text))
```





КОД РЕШЕНИЯ. СТОП-СЛОВА

Google Code Archive



badwordslint



explosion / spaCy



bars38 / Russian_ban_words



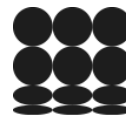
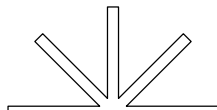
nlTK / nlTK

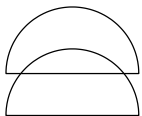
RU

```
word not in bad_lines and word not in  
stopwords.words("russian")
```

EN

```
token.lemma_ not in STOPWORDS and  
token.lemma_ not in bad_w
```





КОД РЕШЕНИЯ. ЛЕММАТИЗАЦИЯ

RU



no-plagiarism / pymorphy3

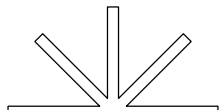
```
words[k]=morph.parse(word)[0].normalized.word
```

EN



explosion / spaCy

```
doc = nlp(jsp_eng.FixFragment(text))
```





КОД РЕШЕНИЯ. СИНОНИМИЗАЦИЯ

```
s11 = model.get_word_vector(current_word)
s12=navec.get(current_word, navec['<unk>'])
s01 = model.get_word_vector(word)
s02=navec.get(word, navec['<unk>'])
```

```
if (1 - spatial.distance.cosine(s01, s11))>0.7 or
(1 - spatial.distance.cosine(s02, s12))>0.421:
    current_freq += freq
    words_and_freqs.remove((word, freq))
```



RU

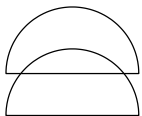


DeepPavlov




navec





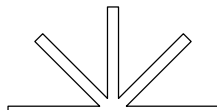
КОД РЕШЕНИЯ. СИНОНИМИЗАЦИЯ

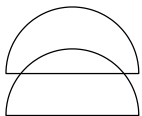


 fse/word2vec-google-news-300

```
try:
    if glove_vectors.similarity(current_word, word)>0.4 and
(current_word != "good" and word != "bad") :
        current_freq += freq
        words_and_freqs.remove((word, freq))
except KeyError:
    continue
```

EN





КОД РЕШЕНИЯ. API

Что заставляет людей выбирать одно и то же каждый раз?

честность

комфорт,

привычка,

СТРАХ,

отсутствие ВыборА

Люди хотят простоты.

****зона комфорта****

голодная НЕволя ...

фатализм

банальная ЛЕНЬ?!...

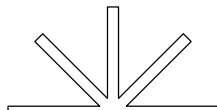
отсутствие ЭНЕРГИИ

СУЩЕСТВУЮЩИЕ, проверенные схемы,

проверенные связи,



Flask





2.48 секунд

Среднее время выполнения

КОМАНДА



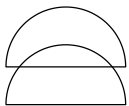
Ахрамович Иван
ya.aif2002@yandex.ru



Новокрещенов Пётр
npa002@campus.mephi.ru



Сиваченко Наталья
nsforst@mail.ru





СПАСИБО ЗА ВНИМАНИЕ!

CREDITS: This presentation template was created by [Slidesgo](#), and includes icons by [Flaticon](#), and infographics & images by [Freepik](#)

