



**Московский государственный университет  
имени М.В. Ломоносова**



ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И КИБЕРНЕТИКИ

---

## **Семантическая близость слов на основе коллекции текстов**

Зверев Дмитрий, 210-я группа ВМК МГУ

## План работы

Скачав файлы “WordSim 353 Goldstandard” и “wiki-news-300d-1M.vec” с векторами слов, насчитанными на Википедии, составить программу с использованием модели на основе fastText подсчета близости слов из групп “WordSim Similarity” и “WordSim Relatedness”. Оценить корреляцию Спирмена с человеческими оценками (или стандартом) для каждой группы слов. Сравнить полученные данные с результатами вычисления близости слов на основе WordNet.

### Составление программы (см. файл “test\_fasttext.py”)

Для начала подключим модуль io для импортирования векторов слов из файла “wiki-news-300d-1M.vec”.

```
1 import io #импорт данных
```

Рассмотрим основную функцию программы.

```
36 #ОСНОВНАЯ ФУНКЦИЯ
37 words = [] #создаем список для хранения пар слов
38 vectors = {} #создаем словарь для хранения слов и соответствующих им векторов
39
40 #ниже выбираем группу слов для анализа (Similarity/Relatedness), записываем их попарно в words и по отдельности в vectors
41 symbol = input("Введите символ 'S' для анализа группы Similarity; 'R' для анализа группы Relatedness: ")
42 if (symbol == 'S'):
43     reading('wordsim_similarity_goldstandard.txt', words, vectors) #Similarity
44 elif (symbol == 'R'):
45     reading('wordsim_relatedness_goldstandard.txt', words, vectors) #Relatedness
46 else:
47     print("Error: incorrect symbol!") #Error
48     exit()
49
50 load_vectors('wiki-news-300d-1M.vec', vectors) #загрузка необходимых векторов из спец.файла
51
52 calculating(words, vectors) #выдача результатов (числовое значение для каждой пары слов)
```

Создадим список words для хранения всех пар слов в следующем виде:

```
words = [ [word_1, word_2], [word_3, word_4], ... ]
```

Создадим словарь vectors для хранения слов и соответствующих им векторов.

Далее запросим у пользователя символ, который будет информировать систему о том, какую группу слов мы будем обрабатывать, и запишем его в переменную symbol:

- ‘S’ – если хотим обрабатывать группу слов Similarity;
- ‘R’ – если хотим обрабатывать группу слов Relatedness.

В противном случае выдаем на экран сообщение об ошибке и останавливаемся.

Итак, введя один из вышеперечисленных символов, переходим к выполнению функции `reading()`.

```
12 #ФУНКЦИЯ: чтение файла fname: попарная запись слов в список list_words и по отдельности в словарь dictionary_vectors,
13 # то есть создание ключей
14 def reading(fname, list_words, dictionary_vectors):
15     f = open(fname, 'r', encoding='utf-8', newline='\n', errors='ignore')
16     for each in f:
17         word_1, word_2, human_value = each.split()
18         list_words.append([word_1, word_2])
19         dictionary_vectors[word_1], dictionary_vectors[word_2] = 1, 1
20     f.close()
```

Данная функция принимает в качестве параметров имя файла `fname`, в котором хранятся пары слов, список `list_words` для хранения пар слов из этого файла и словарь `dictionary_vectors` для создания ключей из данных слов. В функции происходит поэлементное чтение данных из файла `fname` и их запись в список `list_words` и словарь `dictionary_vectors` в качестве ключей, при этом им в соответствие ставится значение 1, тем самым происходит информирование системы о существовании такого ключа.

Далее переходим к выполнению функции `load_vectors()`.

```
3 #ФУНКЦИЯ: загрузка необходимых векторов из спец.файла fname и их запись в словарь dictionary_vectors для соотв-их слов
4 def load_vectors(fname, dictionary_vectors):
5     fin = io.open(fname, 'r', encoding='utf-8', newline='\n', errors='ignore')
6     n, d = map(int, fin.readline().split())
7     for line in fin:
8         tokens = line.rstrip().split(' ')
9         if dictionary_vectors.get(tokens[0], None):
10             dictionary_vectors[tokens[0]] = list(map(float, tokens[1:]))


```

Параметры данной функции – имя файла `fname` и словарь `dictionary_vectors`, содержащий ключи в виде анализируемых слов и поставленные им в соответствие единичные значения. За основу функции взят код с официального сайта fastText: <https://fasttext.cc/docs/en/english-vectors.html>, при этом в нашем случае происходит копирование только тех векторов, для которых соответствующие им слова содержатся в словаре `dictionary_vectors` в качестве ключей, которым, в свою очередь, мы поставили в соответствие единичные значения.

Итак, используемые в основной функции программы список `words` и словарь `vectors` заполнены. Переходим к непосредственному вычислению близости между словами.

Рассмотрим функцию `calculating()`.

```
22 #ФУНКЦИЯ: последовательное вычисление косинусов углов между векторами, хранящихся в словаре dictionary_vectors,
23 # для каждой пары слов из списка list_words и выдача их значений на экран
24 def calculating(list_words, dictionary_vectors):
25     for each in list_words:
26         numerator, sum_1, sum_2 = 0, 0, 0
27         for i in range(len(dictionary_vectors[each[0]])):
28             numerator += (dictionary_vectors[each[0]])[i] * (dictionary_vectors[each[1]])[i]
29             sum_1 += (dictionary_vectors[each[0]])[i] ** 2
30             sum_2 += (dictionary_vectors[each[1]])[i] ** 2
31         denominator = (sum_1 ** 0.5) * (sum_2 ** 0.5)
32         print(numerator / denominator)
```

Параметрами этой функции являются список `list_words` с анализируемыми парами слов и словарь `dictionary_vectors`, содержащий все необходимые векторы слов. Данная функция для каждой пары слов из списка `list_words` вычисляет значение косинуса угла между векторами двух слов, тем самым происходит анализ их близости. При этом косинус угла между двумя векторами задается следующей формулой:

$$\cos \alpha = \frac{\sum_{i=1}^n u_i \cdot v_i}{\sqrt{\sum_i u_i^2} \cdot \sqrt{\sum_i v_i^2}} = \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{u}\|_2 \cdot \|\mathbf{v}\|_2}$$

где  $\mathbf{u}$ ,  $\mathbf{v}$  – векторы. Итак, получив необходимые значения, последовательно выводим их на экран. Полученные результаты копируем в таблицу, речь о которой пойдет ниже.

### Получение меры Спирмена

Рассмотрим сформированную таблицу (см. файл “Таблица, задание от 22 Mar 2024”). Будем располагать пары слов по мере снижения их близости с точки зрения человеческих оценок, или стандарта (данные о человеческих оценках содержатся в файле “WordSim 353 Goldstandart”).

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
SIMILARITY						РАСЧЁТ МЕР СПИРМЕНА				RELATEDNESS					
tiger	tiger	10.00	1.00	1.00	1.00					environment	ecology	8.81	0.62	1.00	9.00
fuck	sex	9.44	0.38	2.00	146.00					Maradona	football	8.62	0.41	2.00	150.00
journey	voyage	9.29	0.78	3.50	6.00					OPEC	oil	8.59	0.50	3.00	62.00
middy	noon	9.29	0.81	3.50	3.00					money	bank	8.50	0.54	4.50	39.50
dollar	buck	9.22	0.38	5.00	141.00					computer	software	8.50	0.70	4.50	2.00
money	cash	9.15	0.72	6.00	14.00	→	МЕРА СПИРМЕНА SIMILARITY	0,81		Jerusalem	Israel	8.46	0.70	6.00	1.00
coast	shore	9.10	0.69	7.00	20.00					law	lawyer	8.38	0.59	7.00	21.00
money	currency	9.04	0.53	8.00	62.00					weather	forecast	8.34	0.59	8.00	18.00
football	soccer	9.03	0.80	9.00	4.00					network	hardware	8.31	0.50	10.00	61.00
magician	wizard	9.02	0.60	10.00	44.00					nature	environment	8.31	0.60	10.00	15.00
type	kind	8.97	0.65	11.00	30.00					FBI	Investigation	8.31	0.52	10.00	46.00
gem	jewel	8.96	0.75	12.00	11.00					money	wealth	8.27	0.61	12.00	14.00
car	automobile	8.94	0.77	13.00	9.00		МЕРА СПИРМЕНА RELATEDNESS	0,68	←	psychology	Freud	8.21	0.55	13.00	32.00
street	avenue	8.88	0.59	14.00	49.00					news	report	8.16	0.50	14.00	58.00
asylum	madhouse	8.87	0.47	15.00	87.00					war	troops	8.13	0.53	15.00	45.00
boy	lad	8.83	0.77	16.00	8.00					physics	proton	8.12	0.48	16.50	84.00
furnace	stove	8.79	0.66	17.00	27.00					bank	money	8.12	0.54	16.50	39.50
seafood	lobster	8.70	0.62	18.00	41.00					planet	galaxy	8.11	0.69	18.00	3.00
mile	kilometer	8.66	0.74	19.00	12.00					stock	market	8.08	0.57	19.00	26.00
king	queen	8.58	0.76	20.00	10.00					planet	constellation	8.06	0.46	20.50	103.00
murder	manslaughter	8.53	0.63	21.00	34.00					credit	card	8.06	0.56	20.50	29.00
vodka	gin	8.46	0.67	22.00	25.00					hotel	reservation	8.03	0.51	22.00	56.00
planet	star	8.45	0.49	23.00	78.00					closet	clothes	8.00	0.48	23.00	79.00
calculation	computation	8.44	0.69	24.00	22.00					soap	opera	7.94	0.62	24.50	10.00
money	dollar	8.42	0.56	25.00	54.00					planet	astronomer	7.94	0.47	24.50	94.00
championship	tournament	8.36	0.69	26.00	21.00					planet	space	7.92	0.49	26.50	66.00
seafood	food	8.34	0.60	27.00	46.00					movie	theater	7.92	0.59	26.50	19.00
man	woman	8.30	0.82	28.00	2.00					treatment	recovery	7.91	0.55	28.00	31.00
vodka	brandy	8.13	0.69	29.60	23.00					baby	mother	7.85	0.68	29.00	4.00
Harvard	Yale	8.13	0.80	29.50	5.00					money	deposit	7.73	0.48	30.00	75.00
psychology	psychiatry	8.08	0.72	37.50	15.00					television	film	7.72	0.62	31.00	11.00
planet	moon	8.08	0.59	37.50	47.00					psychology	mind	7.69	0.48	33.00	74.00
planet	sun	8.02	0.56	33.00	53.00					game	team	7.69	0.63	33.00	6.00
tiger	jaguar	8.00	0.62	34.50	38.00					admission	ticket	7.69	0.48	33.00	80.00
tiger	feline	8.00	0.50	34.50	74.00					Jerusalem	Palestinian	7.65	0.59	35.50	23.00
liquid	water	7.89	0.65	36.00	29.00					Arafat	terror	7.65	0.38	35.50	184.00
words	standart	fastText	for standart	for fastText	for standart	fastText				stock	jaguar	0.92	0.29	244.50	240.00
monk	slave	0.92	0.34	195.50	171.00					monk	slave	0.92	0.34	244.50	219.00
lad	wizard	0.92	0.43	195.50	117.00					lad	wizard	0.92	0.43	244.50	137.00
sugar	approach	0.88	0.34	198.00	173.00					sugar	approach	0.88	0.34	247.00	221.00
rooster	voyage	0.62	0.24	199.00	200.00					rooster	voyage	0.62	0.24	248.00	248.00
noon	string	0.54	0.32	200.50	181.00					noon	string	0.54	0.32	249.50	229.00
chord	smile	0.54	0.29	200.50	191.00					chord	smile	0.54	0.29	249.50	239.00
professor	cucumber	0.31	0.35	202.00	165.00					professor	cucumber	0.31	0.35	251.00	214.00
king	cabbage	0.23	0.38	203.00	140.00					king	cabbage	0.23	0.38	252.00	179.00

- В столбцах А и В находятся соответствующие пары слов из группы Similarity, в столбцах К и L – пары слов из группы Relatedness;
- В столбце С для Similarity и столбце М для Relatedness находятся значения близости слов, определенные т.н. стандартом в файле “WordSim 363 Goldstandard” и с которыми будет производиться сравнение полученных результатов;
- В столбце D для Similarity и столбце N для Relatedness находятся значения близости слов, полученные с использованием модели на основе fastText составленной программой;
- Получим меру Спирмена (основа идеи взята с сайта <https://www.codecamp.ru/blog/spearman-rank-correlation-google-sheets/?ysclid=ltyzly076k177455617>). Рассмотрим группу слов Similarity (для группы слов Relatedness действия аналогичны). В столбцах Е, F находятся значения, полученные при вычислении спец.функции =РАНГ.СР() для стандарта и модели на основе fastText соответственно. Далее, в отдельных ячейках столбцов Н и I находится полученная мера Спирмена корреляции человеческих оценок близости и косинусов углов между векторами слов с использованием спец.функции =КОРРЕЛ();
- Стоит отметить, что коэффициент Спирмена может принимать значение от -1 до +1, т.е. от идеальной отрицательной связи между значениями до идеальной связи. Так, при использовании модели на основе

fastText для группы слов Similarity, как видно из таблицы, полученная мера приблизительно равняется **~0,81**, тогда как для группы слов Relatedness это значение близко к **~0.68**.

## Сравнение с результатами вычисления близости слов на основе WordNet

Рассмотрим результаты вычисления близости слов на основе WordNet с использованием пакета nltk методами lch, wup и jcn. Получив значения для каждого метода и составив аналогичную таблицу, рассмотрим соответствующие меры Спирмена.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
SIMILARITY										РАСЧЁТ МЕР СПИРМЕНА										RELATEDNESS				
tiger	tiger	0.00	3.64	1.00	1.000 000 000	2.00	1.00	5.00	5.00	5.00					environment	ecology	8.81	2.94	0.92	0.00	1.00	4.50	4.50	247.00
fuck	sex	8.44	2.54	0.88	0.25	40.00	49.00	44.00		6.00	6.00	6.00	6.00	Maradona	football	8.62	0.00	0.00	0.00	2.00	251.00	251.00	251.00	
journey	voyage	2.94	2.94	0.95	0.30	3.62	17.50	14.00	36.00	5.00	5.00	5.00	5.00	OPEC	oil	8.50	1.00	0.94	0.00	3.00	238.00	228.92	247.00	
country	state	5.29	3.64	1.00	1.000 000 000	2.00	1.00	5.00	5.00	5.00	5.00	5.00	5.00	money	bank	8.50	0.97	0.97	0.00	1.00	107.00	107.00	107.00	
dollar	buck	9.22	3.64	1.00	1.000 000 000	5.00	5.00	5.00	5.00	5.00	5.00	5.00	5.00	computer	software	8.50	1.07	0.14	0.06	4.50	236.00	243.50	174.00	
money	cash	0.88	0.88	0.80	0.30	6.00	40.00	41.00	35.00	5.00	5.00	5.00	5.00	Jerusalem	israel	8.46	1.69	0.70	0.07	6.00	107.00	54.50	165.00	
cost	price	8.15	2.94	0.91	0.30	7.00	17.50	14.00	31.00	5.00	5.00	5.00	5.00	laser	laser	8.24	1.17	0.90	0.00	7.00	197.00	213.00	168.00	
money	currency	9.04	2.94	0.93	0.51	8.00	17.50	21.00	18.00	5.00	5.00	5.00	5.00	weather	forecast	8.34	1.31	0.33	0.06	8.00	187.00	185.00	218.00	
football	soccer	8.03	0.96	0.43	0.43	8.00	17.50	12.00	21.00	5.00	5.00	5.00	5.00	network	internet	8.31	0.80	0.06	0.00	10.00	29.00	26.00	175.00	
monkey	wizard	3.40	1.00	1.000 000 000	10.00	5.00	5.00	5.00	5.00	5.00	5.00	5.00	5.00	nature	environment	8.31	0.90	0.22	0.06	10.00	76.50	120.50	57.00	
type	kind	9.77	2.94	0.99	0.67	11.00	17.50	14.00	31.00	5.00	5.00	5.00	5.00	reindeer	fbi	8.31	0.90	0.22	0.06	10.00	244.00	220.00	210.00	
gem	jewel	8.96	3.64	1.00	1.000 000 000	12.00	5.00	5.00	5.00	5.00	5.00	5.00	5.00	money	wealth	8.27	2.94	0.92	14.59	12.00	4.50	4.50	1.00	
car	automobile	8.54	2.94	0.99	0.67	14.00	17.50	14.00	31.00	5.00	5.00	5.00	5.00	psychology	Freud	8.21	0.64	0.10	0.05	13.00	249.00	251.00	239.00	
street	avenue	8.88	2.94	0.99	0.67	14.00	17.50	14.00	31.00	5.00	5.00	5.00	5.00	news	report	8.24	1.54	0.93	0.00	14.00	24.00	24.00	20.00	
asylum	madhouse	8.87	2.94	0.95	0.31	16.00	17.50	14.00	31.00	5.00	5.00	5.00	5.00	war	troops	8.13	1.34	0.31	0.06	16.00	177.50	185.00	172.00	
boy	lad	8.83	2.94	0.95	0.31	16.00	17.50	14.00	31.00	5.00	5.00	5.00	5.00	bank	bank	8.12	0.93	0.22	0.04	16.00	244.00	228.00	241.00	
face	skate	8.79	1.54	0.93	0.30	14.00	17.50	14.00	31.00	5.00	5.00	5.00	5.00	planet	galaxy	8.11	1.56	0.63	0.06	18.00	138.00	71.50	197.00	
seafood	lobster	8.70	2.94	0.98	0.23	18.00	40.00	49.00	48.00	5.00	5.00	5.00	5.00	stock	paper	8.08	1.69	0.56	0.13	19.00	107.00	116.00	45.00	
mite	spider	8.66	2.54	0.80	0.11	18.00	66.50	65.50	97.00	5.00	5.00	5.00	5.00	constellation	star	8.05	2.07	0.15	0.15	20.00	29.00	30.00	36.00	
king	queen	8.67	3.44	1.00	1.000 000 000	20.00	20.00	5.00	5.00	5.00	5.00	5.00	5.00	crisis	card	8.05	1.85	0.62	0.12	20.00	78.50	110.00	48.00	
parent	ancestor	8.53	2.94	0.92	0.21	21.00	38.00	38.00	52.00	5.00	5.00	5.00	5.00	hotel	reservation	8.03	2.14	0.38	0.06	22.00	197.50	175.00	173.00	
vodka	gin	8.46	2.54	0.91	0.00	22.00	40.00	30.50	205.00	5.00	5.00	5.00	5.00	closet	clothes	8.00	1.56	0.50	0.07	23.00	138.00	94.00	146.00	
planet	star	8.45	2.54	0.90	0.50	23.00	40.00	49.00	19.00	5.00	5.00	5.00	5.00	map	space	7.95	1.39	0.34	0.05	24.00	238.00	233.00	233.00	
calculator	computer	8.44	2.54	0.91	1.00	1.000 000 000	24.00	5.00	5.00	5.00	5.00	5.00	5.00	planet	astronomer	7.94	1.85	0.63	0.08	24.50	76.50	71.50	126.00	
competition	competition	8.44	2.54	0.90	0.21	25.00	69.00	72.00	53.00	5.00	5.00	5.00	5.00	computer	keyboard	7.95	1.85	0.63	0.00	35.50	76.50	71.50	247.00	
championship	tournament	8.42	2.03	0.78	0.21	25.00	69.00	72.00	53.00	5.00	5.00	5.00	5.00	bowling	rabbit	7.81	1.69	0.73	0.15	38.00	107.00	45.00	34.00	
team	club	8.41	2.54	0.90	0.13	26.00	40.00	49.00	82.00	5.00	5.00	5.00	5.00	computer	internet	7.98	1.56	0.63	0.08	39.00	138.00	71.50	124.00	
dog	pet	8.40	2.54	0.90	0.13	27.00	38.00	34.00	31.00	5.00	5.00	5.00	5.00	money	deposit	7.97	1.56	0.50	0.08	39.50	14.00	17.50	6.00	
man	woman	8.30	2.54	0.94	0.38	28.00	40.00	93.00	32.00	5.00	5.00	5.00	5.00	treasure	recovery	7.91	2.25	0.77	0.08	38.00	29.00	30.00	102.00	
dog	pet	8.29	2.54	0.94	0.31	28.00	40.00	28.50	32.00	5.00	5.00	5.00	5.00	baby	mother	7.85	2.03	0.60	0.17	38.00	50.00	71.50	27.00	
Harvest	Yale	8.17	2.54	0.94	0.31	28.00	40.00	28.50	32.00	5.00	5.00	5.00	5.00	television	film	7.73	2.03	0.78	0.18	39.00	19.00	19.00	21.00	
psychology	psychiatry	8.08	1.09	0.75	0.73	28.00	40.00	38.00	84.00	5.00	5.00	5.00	5.00	psychology	game	7.69	1.69	0.59	0.09	33.00	107.00	98.00	86.00	
planet	moon	8.08	2.25	0.80	0.20	31.00	66.50	65.50	43.00	5.00	5.00	5.00	5.00	administration	team	7.69	1.34	0.31	0.08	33.00	177.00	185.00	118.00	
dividend	payment	7.93	2.54	0.90	0.20	33.00	66.50	65.50	33.00	5.00	5.00	5.00	5.00	Jerusalem	Palestinian	7.66	0.86	0.29	0.00	35.00	247.00	204.50	247.00	
date	mate	7.80	2.54	0.90	0.20	34.00	40.00	24.00	39.00	5.00	5.00	5.00	5.00	Arabat	terror	7.65	1.85	0.63	0.00	35.50	76.50	71.50	247.00	
century	year	7.59	2.54	0.83	0.12	45.00	40.00	54.00	88.00	5.00	5.00	5.00	5.00	Asrat	terrorist	7.65	1.85	0.63	0.00	44.50	177.00	183.00	136.00	
rock	jazz	7.59	2.54	0.88	0.38	45.00	40.00	41.00	24.00	5.00	5.00	5.00	5.00	computer	keyboard	7.65	1.85	0.63	0.00	44.50	76.50	84.00	83.00	
announcement	news	7.59	2.54	0.90	0.13	45.00	40.00	41.00	30.00	5.00	5.00	5.00	5.00	book	paper	7.47	1.55	0.50	0.00	45.00	138.00	71.50	208.00	
food	snack	7.52	1.34	0.40	0.00	46.00	166.00	165.00	117.00	5.00	5.00	5.00	5.00	book	library	7.46	2.54	0.88	0.19	46.00	14.00	11.00	22.00	
marathon	sprint	7.47	1.34	0.50	0.05	48.00	179.50	146.00	194.00	5.00	5.00	5.00	5.00	psychology	depression	7.42	1.15	0.50	0.06	48.00	21.00	24.00	51.00	
Mexico	Brazil	7.44	1.34	0.50	0.05	50.00	69.00	65.00	129.00	5.00	5.00	5.00	5.00	cloud	string	7.44	2.18	0.60	0.11	49.00	38.00	33.00	134.00	
make	girls	7.42	2.25	0.85	0.35	54.00	66.50	67.00	27.00	5.00	5.00	5.00	5.00	professor	curriculum	7.31	0.51	0.21	0.04	49.00	129.00	124.00	120.00	
make	girls	7.42	2.04	0.87	0.37	51.00	17.50	10.00	26.00	5.00	5.00	5.00	5.00	words	standard	7.38	1.06	0.53	0.08	53.00	138.00	127.00	120.00	
professor	curriculum	0.21	1.07	0.50	0.04	202.00	184.50	149.00	198.00					standard	ich	1.19	1.69	0.83	0.06	242.00	107.00	76.50	214.00	
king	cabbage	0.23	1.34	0.57	0.07	203.00	168.00	128.50	158.00					for standard	for ich	0.82	1.82	0.82	0.09	244.50	76.50	84.00	83.00	
														for standard	for wup	0.82	1.82	0.82	0.09	244.50	220.00	228.92	247.00	
														for standard	for jcn	0.82	1.82	0.82	0.07	242.00	177.00	186.00	166.00	

основе fastText обучена на больших корпусах текстов и использует высокие размерности векторов, благодаря чему способна лучше определять контекст. Очевидным следствием этого является значительно более высокое значение меры Спирмена для группы слов Relatedness. Помимо этого, ресурс WordNet не всегда полон, например, одного из рассматриваемых нами слов (“Maradona”) в базе данных WordNet нет, из-за чего близость слов “Maradona” и “football” считается нулевой, что, конечно, некорректно.

## **Выводы**

Во время выполнения данного задания была изучена модель на основе fastText подсчета близости слов, которая была использована для анализа групп слов Similarity и Relatedness. В ходе работы была получена мера Спирмена корреляции человеческих оценок близости, содержащихся в файле “WordSim 353 Goldstandart”, и косинусов углов между векторами слов – для Similarity было получено значение, близкое к ~0.81, что говорит о наличии сильной связи между человеческими оценками близости и косинусами углов между векторами слов, для Relatedness – значения, близкие к ~0.68, что также говорит о наличии взаимосвязи. Также было проведено сравнение полученных результатов с результатами вычисления близости на основе WordNet, при этом выяснилось, что анализ близости слов с использованием модели на основе fastText работает лучше, чем на основе WordNet.