



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



ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И КИБЕРНЕТИКИ

Анализ близости слов автоматическими методами

Зверев Дмитрий, 210-я группа ВМК МГУ

План работы

Скачав файл “WordSim 353 Goldstandard” и воспользовавшись пакетом nltk доступа к WordNet, составить программы подсчета близости группы слов “Wordsim Similarity” и “Wordsim Relatedness” методами lch, wup и jcn. Оценить меру Спирмена для каждой группы слов и используемого метода относительно человеческих оценок, приведенных в файле “WordSim 353 Goldstandard”. Провести анализ, почему некоторые близкие слова по мнению человека оказались далекими по смыслу при их автоматической обработке (и наоборот).

Метод lch

```
synset1.lch_similarity(synset2): Leacock-Chodorow Similarity: Return a score denoting how similar two word senses are, based on the shortest path that connects the senses (as above) and the maximum depth of the taxonomy in which the senses occur. The relationship is given as -log(p/2d) where p is the shortest path length and d the taxonomy depth.
```

Программа с подробными комментариями для анализа близости слов методом lch приведена ниже (см. файл “test_lch.py”).

В бесконечном цикле вводится пара слов, после чего заполняются synset-ы для каждого слова – synsets1_full и synsets2_full соответственно. Так как данный метод применим только для слов, части речи которых одинаковы, то будем составлять 2 списка-подмножества полученных ранее synset-ов для каждого слова соответственно – synsets1 и synsets2, в которых будут храниться слова одной и той же части речи. Таким образом, происходит анализ близости двух слов последовательно для каждой части речи (**noun**, **verb**, **adjective**). Далее создадим и заполним список my_list, содержащего результаты применения метода lch между словами из synset-ов synsets1 и synsets2. Отыскав максимальное число в этом списке (для одной части речи), находим значение близости исходных двух слов и записываем его в переменную similarity_buf. Пройдясь по всем частям речи, печатаем результат – значение близости исходных двух слов similarity.

```

1  from nltk.corpus import wordnet as wn           #импортируем данные
2
3  #бесконечный цикл прохода по всем словам (чтобы завершить цикл и, соотв-но, программу, следует нажать Ctrl + C)
4  while 1:
5      word1, word2 = input("").split()           #вводим пару слов, разделяя их через пробел
6      synsets1_full = wn.synsets(word1)          #здесь храним synset-ы для 1-го слова
7      synsets2_full = wn.synsets(word2)          #здесь храним synset-ы для 2-го слова
8      similarity = 0                            #инициализация переменной подсчета близости слов
9      for part_of_speech in ('n', 'v', 'a'):      #цикл прохода по всем частям речи (noun, verb, adjective)
10         synsets1, synsets2 = [], []             #иниц-ия списков для хранения synset-ов с соответ-й частью речи для обоих слов
11         for x in synsets1_full:
12             if x.pos() == part_of_speech:
13                 synsets1.append(x)
14         for x in synsets2_full:                  #циклически "достаем" из synsets2_full слова нужной части речи
15             if x.pos() == part_of_speech:
16                 synsets2.append(x)
17     #ниже - непосредственное применение метода для подсчета близости слов и запись результатов в список my_list
18     my_list = [synset1.lch_similarity(synset2) for synset1 in synsets1 for synset2 in synsets2]
19     if len(my_list) != 0:                      #проверка на пустоту списка my_list (т.е. на возможность применения метода)
20         similarity_buf = max(my_list)          #вычисляем близость двух слов для соответствующей части речи
21     else:
22         similarity_buf = 0
23     similarity = max(similarity, similarity_buf) #подсчет близости двух слов
24 print(similarity)                          #печать результата для каждой пары

```

Итак, последовательно подав на вход программе все пары слов из файла “WordSim 353 Goldstandard” из каждой группы слов – Similarity и Relatedness, получаем соответствующие числа, которые мы построчно запишем в таблицу, речь о которой пойдет ниже.

Метод wup

`synset1.wup_similarity(synset2)`: Wu-Palmer Similarity: Return a score denoting how similar two word senses are, based on the depth of the two senses in the taxonomy and that of their Least Common Subsumer (most specific ancestor node). Note that at this time the scores given do **not** always agree with those given by Pedersen's Perl implementation of Wordnet Similarity.

The LCS does not necessarily feature in the shortest path connecting the two senses, as it is by definition the common ancestor deepest in the taxonomy, not closest to the two senses. Typically, however, it will so feature. Where multiple candidates for the LCS exist, that whose shortest path to the root node is the longest will be selected. Where the LCS has multiple paths to the root, the longer path is used for the purposes of the calculation.

Программа для анализа близости слов методом wup приведена ниже (см. файл “`test_wup.py`”).

Суть этой программы практически полностью совпадает с программой для метода `lch`, с тем лишь отличием, что метод `wup` применим одновременно ко всем частям речи, что немного сокращает код.

```

1  from nltk.corpus import wordnet as wn           #импортируем данные
2
3  #бесконечный цикл прохода по всем словам (чтобы завершить цикл и, соответственно, программу, следует нажать Ctrl + C)
4  while 1:
5      word1, word2 = input("").split()             #вводим пару слов, разделяя их через пробел
6      #ниже получаем synset-ы для каждого слова в паре двух слов - здесь можно рассматривать все части речи сразу
7      synsets1 = wn.synsets(word1)
8      synsets2 = wn.synsets(word2)
9      #ниже - непосредственное применение метода для подсчета близости слов и запись результатов в список my_list
10     my_list = [synset1.wup_similarity(synset2) for synset1 in synsets1 for synset2 in synsets2]
11     #вычисляем максимальное число близости этих двух слов
12     if len(my_list) != 0:
13         similarity = max(my_list)
14     else:
15         similarity = 0
16     #печатать результата для каждой пары слов
17     print(similarity)

```

Аналогично, подав на вход программе все необходимые пары слов, получаем соответствующие значения и записываем их в таблицу.

Метод jcn

```

synset1.jcn_similarity(synset2, ic): Jiang-Conrath Similarity Return a score denoting how similar two word senses are, based on the Information Content (IC) of the Least Common Subsumer (most specific ancestor node) and that of the two input Synsets. The relationship is given by the equation 1 / (IC(s1) + IC(s2) - 2 * IC(lcs)).

```

Программа для анализа близости слов методом jcn приведена ниже (см. файл “test_jcn.py”).

Суть этой программы также практически полностью совпадает с программой для метода lch, с тем отличием, что здесь нет необходимости рассматривать слова с частью речи adjective, так как в файле с информационным содержимым brown_ic содержатся только 2 ключа – noun и verb.

```

1  #импортируем данные
2  from nltk.corpus import wordnet as wn
3  from nltk.corpus import wordnet_ic
4  brown_ic = wordnet_ic.ic('ic-brown.dat')
5
6  #бесконечный цикл прохода по всем словам (чтобы завершить цикл и, соответственно, программу, следует нажать Ctrl + C)
7  while 1:
8      word1, word2 = input("").split()             #вводим пару слов, разделяя их через пробел
9      synsets1_full = wn.synsets(word1)           #здесь храним synset-ы для 1-го слова
10     synsets2_full = wn.synsets(word2)           #здесь храним synset-ы для 2-го слова
11     similarity = 0                             #инициализация переменной подсчета близости слов
12     for part_of_speech in ('n', 'v'):
13         synsets1, synsets2 = [], []
14         for x in synsets1_full:                  #цикл прохода по всем частям речи (noun, verb)
15             if x.pos() == part_of_speech:          #инициализация списков для хранения synset-ов с соответствующей частью речи для обоих слов
16                 synsets1.append(x)                #циклически "достаем" из synsets1_full слова нужной части речи
17             for x in synsets2_full:              #циклически "достаем" из synsets2_full слова нужной части речи
18                 if x.pos() == part_of_speech:
19                     synsets2.append(x)
20     #ниже - непосредственное применение метода для подсчета близости слов и запись результатов в список my_list
21     my_list = [synset1.jcn_similarity(synset2, brown_ic) for synset1 in synsets1 for synset2 in synsets2]
22     if len(my_list) != 0:                      #проверка на пустоту списка my_list (т.е. на возможность применения метода)
23         similarity_buf = max(my_list)          #вычисляем близость двух слов для соответствующей части речи
24     else:
25         similarity_buf = 0
26     similarity = max(similarity, similarity_buf) #подсчет близости двух слов
27     print(similarity)                         #печатать результата для каждой пары

```

Аналогично, подав на вход программе все пары слов, получаем соответствующие значения и записываем их в таблицу.

Подсчет меры Спирмена

Рассмотрим сформированную таблицу (см. файл “Таблица, задание №1”). Будем располагать пары слов по мере снижения их близости с точки зрения человеческих оценок, или стандарта (данные о человеческих оценках содержатся в файле “WordSim 363 Goldstandart”).

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																											
tiger	tiger	10.00	3.54	1.00	1 000 000 000	0.25	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	9.44	3.54	0.86	2.00	0.25	2.00	5.00	5.00	5.00					Maradona	football	8.50	1.00	0.90	0.00	2.00	251.50	251.50	251.50			
journey	voyage	8.95	3.54	0.86	0.00	0.50	2.50	17.50	15.00	14.00	36.00				OPEC	oil	8.50	1.00	0.84	0.00	3.00	238.00	228.50	207.00			
money	cash	8.23	3.54	1.00	1 000 000 000	0.50	1.00	5.00	5.00	5.00					monkey	ape	8.50	1.49	0.87	0.13	4.50	107.00	107.00	85.50			
dollar	buck	8.22	3.54	1.00	1 000 000 000	0.50	5.00	5.00	5.00	5.00					computer	software	8.46	1.07	0.14	0.06	4.50	235.50	245.50	174.00			
money	cash	8.15	3.54	0.86	0.30	6.00	40.00	41.00	35.00	35.00					Jerusalem	israel	8.46	0.99	0.76	0.07	6.00	107.00	54.50	165.00			
case	case	8.04	3.54	0.86	0.11	7.50	15.00	20.00	21.00	21.00					lawyer	law	8.24	1.24	0.17	0.09	7.00	197.00	213.00	198.00			
money	currency	8.04	3.54	0.51	8.00	17.50	21.50	18.00	18.00	18.00					weather	forecast	8.34	1.31	0.33	0.06	8.00	187.50	185.50	218.00			
football	soccer	8.03	3.54	0.86	0.43	9.00	12.00	12.00	21.00	21.00					network	hardware	8.31	2.25	0.80	0.06	10.00	29.50	29.50	175.00			
monument	monument	8.02	3.54	1.00	1 000 000 000	0.50	5.00	5.00	5.00	5.00					nature	environment	8.34	0.93	0.50	0.06	7.00	125.00	125.00	125.00			
type	kind	8.97	3.54	0.95	0.67	11.00	17.50	17.00	14.00	14.00					FBI	investigation	8.31	0.93	0.22	0.06	10.00	244.50	220.00	210.00			
gem	jewel	8.96	3.54	1.00	1 000 000 000	12.00	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	8.89	3.54	0.86	0.24	14.00	17.50	16.00	17.00	16.00					psychology	psychiatry	8.21	0.54	0.15	0.05	13.00	249.00	251.00	239.00			
street	avenue	8.88	3.54	0.86	0.80	14.50	17.00	16.50	17.50	17.50					news	report	8.24	0.54	0.54	0.06	4.50	145.00	145.00	20.00			
asylum	madhouse	8.87	3.54	0.95	0.31	16.00	17.50	14.00	31.00	31.00					war	troops	8.13	1.34	0.31	0.06	16.00	177.50	195.00	172.00			
boy	lad	8.83	3.54	0.86	0.23	16.00	17.50	17.00	17.00	17.00					physics	proton	8.12	0.22	0.04	0.04	16.50	244.00	228.00	241.00			
face	face	8.79	3.54	0.86	0.23	16.00	17.50	17.00	17.00	17.00					bank	bank	8.50	0.93	0.27	0.04	16.00	107.00	107.00	85.50			
seafood	lobster	8.75	3.54	0.86	0.23	18.00	40.00	49.00	48.00	48.00					planet	galaxy	8.11	1.96	0.83	0.06	18.00	138.00	71.50	197.00			
mite	kilometer	8.66	3.54	0.86	0.21	18.00	40.00	49.00	48.00	48.00					stock	market	8.08	1.69	0.56	0.13	19.00	107.00	116.00	45.00			
king	queen	8.65	3.54	1.00	1 000 000 000	0.50	20.00	20.00	6.00	6.00					planet	constellation	8.25	1.77	0.45	0.06	20.00	39.00	39.00	46.00			
owner	maneater	8.53	3.54	0.86	0.21	21.00	40.00	49.00	52.00	52.00					credit	card	9.06	1.85	0.62	0.12	20.00	76.50	81.00	49.00			
vodika	gin	8.45	3.54	0.86	0.00	22.00	40.00	30.50	20.50	20.50					hotel	reservation	8.03	1.24	0.38	0.06	22.00	197.50	175.50	173.00			
planet	star	8.41	3.54	0.86	0.50	23.00	40.00	49.00	19.00	19.00					closet	clothes	8.00	1.56	0.50	0.07	23.00	138.00	98.00	146.00			
calculator	computer	8.41	3.54	1.00	1 000 000 000	0.50	24.00	5.00	5.00	5.00					map	area	7.93	1.24	0.24	0.06	24.00	238.00	237.00	232.00			
money	dollar	8.42	3.54	0.20	0.21	25.00	89.00	72.00	53.00	53.00					planet	astronomer	7.94	1.85	0.63	0.08	24.50	76.50	71.50	126.00			
championship	tournament	8.38	3.54	0.86	0.13	26.00	40.00	49.00	82.00	82.00					space	space	7.92	1.69	0.53	0.09	26.50	107.00	127.50	85.50			
student	student	8.34	3.54	0.86	0.21	27.00	34.00	24.00	24.00	24.00					tree	wood	7.93	1.29	0.29	0.06	27.00	107.00	107.00	86.50			
man	woman	8.30	3.54	0.86	0.23	28.00	40.00	49.00	93.00	93.00					treatment	recovery	7.91	2.25	0.77	0.08	28.00	29.00	30.00	102.00			
dog	dog	8.29	3.54	0.86	0.21	29.00	40.00	49.00	93.00	93.00					baby	mother	7.85	2.03	0.83	0.17	29.00	50.00	71.50	27.00			
don	don	8.27	3.54	0.86	0.21	30.00	40.00	49.00	93.00	93.00					money	deposit	7.72	0.82	0.27	0.06	30.00	14.00	14.00	41.00			
yen	yen	8.25	3.54	0.86	0.21	31.00	40.00	49.00	93.00	93.00					television	film	7.72	2.03	0.78	0.18	31.00	90.00	29.00	23.00			
wood	forest	7.73	3.54	0.86	0.21	32.00	40.00	49.00	5.00	5.00					psychology	team	7.69	1.69	0.59	0.09	33.00	107.00	98.00	86.00			
discreet	payment	7.65	3.54	0.86	0.20	33.00	40.00	49.00	65.00	65.00					administration	team	7.69	1.56	0.37	0.07	33.00	177.00	180.50	118.00			
rope	rope	7.63	3.54	0.86	0.21	34.00	40.00	49.00	65.00	65.00					Jerusalem	Palestinian	7.65	0.86	0.29	0.06	34.00	257.00	204.50	247.00			
century	year	7.59	3.54	0.86	0.21	35.00	40.00	49.00	65.00	65.00					Arabat	terror	7.65	1.85	0.63	0.00	35.50	76.50	71.50	124.00			
rock	rock	7.59	3.54	0.86	0.21	36.00	40.00	21.50	21.50	21.50					computer	internet	7.65	1.23	0.63	0.07	37.00	23.00	13.00	130.00			
announcer	jazz	7.56	3.54	0.86	0.21	37.00	40.00	49.00	70.00	70.00					boxing	round	7.61	1.69	0.73	0.15	38.00	107.00	43.00	34.00			
food	fruit	7.51	3.54	0.86	0.21	38.00	40.00	49.00	168.00	168.00	168.00					computer	internet	7.58	1.56	0.83	0.08	39.00	138.00	71.50	124.00		
marathon	sprint	7.47	3.54	0.86	0.05	39.00	48.00	178.00	148.00	148.00	194.00				money	property	7.58	1.23	0.63	0.05	40.00	14.00	14.00	6.50			
Mexico	Brazil	7.44	3.54	0.86	0.05	40.00	48.00	60.00	66.00	66.00	63.00				tones	rocket	7.56	1.44	0.60	0.06	41.00	159.00	94.00	215.00			
rice	rice	7.42	3.54	0.86	0.21	41.00	40.00	49.00	57.00	57.00	27.00				canary	landscape	7.53	1.15	0.35	0.05	42.50	213.00	175.50	234.00			
professor	cucumber	0.31	1.07	0.50	0.04	202.00	194.50	149.00	149.00	149.00	186.00				telephone	communications	7.53	0.82	0.23	0.04	43.50	181.00	181.00	181.00			
king	cabbage	0.23	1.34	0.57	0.07	203.00	188.00	168.00	126.50	126.50	158.00				currency	market	7.50	1.34	0.31	0.07	44.50	177.00	193.00	136.00			
words	words					standard	lch	wup	jcn	for standard	for lch	for wup	for jcn					psychology	cognition	7.48	1.85	0.62	0.14	44.50	76.50	84.00	38.00

- В столбцах А и В находятся соответствующие пары слов из группы Similarity, в столбцах Р и Q – пары слов из группы Relatedness;
- В столбце С для Similarity и столбце R для Relatedness находятся значения близости слов, определенные т.н. стандартом в файле “WordSim 363 Goldstandard” и с которыми будет производиться сравнение для автоматических методов;
- В столбцах D, E, F (S, T, U) находятся значения близости слов из группы Similarity (Relatedness), полученные методами lch, wup, jcn соответственно;
- Получим меру Спирмена (основа идеи взята с сайта <https://www.codecamp.ru/blog/spearman-rank-correlation-google-sheets/?ysclid=ltyz1y076k177455617>). Рассмотрим группу слов Similarity (для группы

слов Relatedness действия аналогичны). В столбцах G, H, I, J находятся значения, полученные при вычислении спец.функции =РАНГ.СР() для стандарта, методов lch, wup, jcn соответственно. Далее, в отдельных ячейках столбцов L, M, N находится полученная мера Спирмена для методов lch, wup, jcn соответственно с использованием спец.функции =КОРРЕЛ() относительно стандарта (т.е. человеческих оценок);

- Стоит отметить, что коэффициент Спирмена может принимать значение от -1 до +1, т.е. от идеальной отрицательной связи между значениями до идеальной связи. Так, для методов lch, wup и jcn и группы слов Similarity, как видно из таблицы, полученная мера примерно равняется ~0,60, тогда как для группы слов Relatedness это значение близко к ~0.00.

Анализ полученных данных

Требуется исследовать проблему семантического анализа слов и их схожести с точки зрения человека и автоматизированных методов, таких как lch, wup и jcn. Перечисленные методы основываются на структуре таксономии, такой как WordNet, для определения степени близости между словами. Итак, рассмотрим причины, почему некоторые близкие по значению слова для человека оказались далекими по смыслу при их автоматической обработке (и наоборот).

Одно слово может иметь несколько значений (в зависимости от контекста), и автоматические методы не всегда могут учитывать этот фактор. Это можно наблюдать, например, при обработке пары слов “jaguar” и “car” группы Similarity:

jaguar	car	7,27	0,80	0,33	0,06	56,00	202,00	173,00	180,00
--------	-----	------	------	------	------	-------	--------	--------	--------

Как можно видеть, сходство этих двух слов с точки зрения человека достаточно высокое (3-й столбец слева: оценка в стандарте идет по шкале от 0.00 до 10.00 по мере увеличения смысловой близости между словами), ведь “jaguar” – это не только животное, но и марка автомобиля. Однако в базе данных WordNet последняя информация отсутствует, поэтому оценка близости этих слов от автоматизированных методов невелики (столбцы голубого цвета).

WordNet Search - 3.1
[- WordNet home page](#) - [Glossary](#) - [Help](#)

Word to search for:

Display Options:
 Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
 Display options for sense: <lexical filename> (gloss)

Noun

- <noun.animal>S: (n) [jaguar](#), [panther](#), [Panthera onca](#), [Felis onca](#) (a large spotted feline of tropical America similar to the leopard; in some classifications considered a member of the genus *Felis*)

Также можно наблюдать ситуацию, когда слова очень близки по смыслу, но в системе WordNet эти слова считаются идентичными, например, пара слов “calculation” и “computation” (группа Similarity):

calculation	computation	8,44	3,64	1,00	1 000 000 000	24,00	5,00	5,00	5,00
-----------------------------	-----------------------------	------	------	------	---------------	-------	------	------	------

WordNet Search - 3.1
[- WordNet home page](#) - [Glossary](#) - [Help](#)

Word to search for:

Display Options:
 Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
 Display options for sense: <lexical filename> (gloss)

Noun

- <noun.act>S: (n) [calculation](#), [computation](#), [computing](#) (the procedure of calculating; determining something by mathematical or logical methods)
- <noun.cognition>S: (n) [calculation](#), [computation](#), [figuring](#), [reckoning](#) (problem solving that involves numbers or quantities)
- <noun.cognition>S: (n) [calculation](#), [deliberation](#) (planning something carefully and intentionally)

WordNet Search - 3.1
[- WordNet home page](#) - [Glossary](#) - [Help](#)

Word to search for:

Display Options:
 Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
 Display options for sense: <lexical filename> (gloss)

Noun

- <noun.act>S: (n) [calculation](#), [computation](#), [computing](#) (the procedure of calculating; determining something by mathematical or logical methods)
- <noun.cognition>S: (n) [calculation](#), [computation](#), [figuring](#), [reckoning](#) (problem solving that involves numbers or quantities)

При этом следует сделать замечание, что даже в базе данных WordNet слово “calculation” имеет на одно значение больше, чем слово “computation”, но, тем не менее, без контекста система определяет их как идентичные.

Аналогичную ситуацию можно наблюдать со словами “wood” и “forest” (группа Similarity), некоторые смысловые значения которых пересекаются.

wood	forest	7,73	3,64	1,00	1 000 000 000	42,00	5,00	5,00	5,00
----------------------	------------------------	------	------	------	---------------	-------	------	------	------

WordNet Search - 3.1

- WordNet home page - Glossary - Help

Word to search for: wood

Display Options: (Select option to change) Change

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: <lexical filename> (gloss)

Noun

- <noun.substance> S: (n) **wood** (the hard fibrous lignified substance under the bark of trees)
- <noun.group> S: (n) **forest, wood, woods** (the trees and other plants in a large densely wooded area)
- <noun.person> S: (n) **Wood, Natalie Wood** (United States film actress (1938-1981))
- <noun.person> S: (n) **Wood, Sir Henry Wood, Sir Henry Joseph Wood** (English conductor (1869-1944))
- <noun.person> S: (n) **Wood, Mrs. Henry Wood, Ellen Price Wood** (English writer of novels about murders and thefts and forgeries (1814-1887))
- <noun.person> S: (n) **Wood, Grant Wood** (United States painter noted for works based on life in the Midwest (1892-1942))
- <noun.artifact> S: (n) **woodwind, woodwind instrument, wood** (any wind instrument other than the brass instruments)
- <noun.artifact> S: (n) **wood** (a golf club with a long shaft used to hit long shots; originally made with a wooden head)

WordNet Search - 3.1

- WordNet home page - Glossary - Help

Word to search for: forest

Display Options: (Select option to change) Change

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: <lexical filename> (gloss)

Noun

- <noun.group> S: (n) **forest, wood, woods** (the trees and other plants in a large densely wooded area)
- <noun.object> S: (n) **forest, woodland, timberland, timber** (land that is covered with trees and shrubs)

Verb

- <verb.contact> S: (v) **afforest, forest** (establish a forest on previously unforested land)

Имеют место ситуации, когда слова, с точки зрения человека, имеют мало общего, однако автоматическими методами определяются как достаточно близкие. Например, пары слов “school” и “center”, “monk” и “slave” (принадлежат как группе слов Similarity, так и Relatedness).

school	center	3,44	2,54	0,88	0,18	143,00	40,00	41,00	65,00
monk	slave	0,92	2,03	0,67	0,07	195,50	89,00	103,50	141,00

Такие ситуации могут возникать вследствие маленького пути от одного слова до другого через общий гипероним:

WordNet Search - 3.1

- WordNet home page - Glossary - Help

Word to search for: school

Display Options: (Select option to change) Change

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: <lexical filename> (gloss)

Noun

- <noun.group> S: (n) **school** (an educational institution)
- <noun.artifact> S: (n) **school, schoolhouse** (a building where young people receive education)
 - *direct hyponym / full hyponym*
 - *part meronym*
 - *direct hypernym / inherited hypernym / sister term*
- <noun.artifact> S: (n) **building, edifice** (a structure that has a roof and walls and stands more or less permanently in one place)
 - *direct hypernym / full hyponym*
 - *part meronym*
 - *direct hypernym / inherited hypernym / sister term*
- <noun.artifact> S: (n) **structure, construction** (a thing constructed; a complex entity constructed of many parts)
 - <noun.Tops> S: (n) **artifact, artefact** (a man-made object taken as a whole)
 - <noun.Tops> S: (n) **whole, unit** (an assemblage of parts that is regarded as a single entity)
 - <noun.Tops> S: (n) **object, physical object** (a tangible and visible entity; an entity that can cast a shadow)
 - <noun.Tops> S: (n) **physical entity** (an entity that has physical existence)
 - <noun.Tops> S: (n) **entity** (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

WordNet Search - 3.1

- WordNet home page - Glossary - Help

Word to search for: center

Display Options: (Select option to change) Change

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: <lexical filename> (gloss)

Noun

- <noun.location> S: (n) **center, centre, middle, heart, eye** (an area that is approximately central within some larger region)
- <noun.artifact> S: (n) **center field, centerfield, center** (the piece of ground in the outfield directly ahead of the catcher)
- <noun.artifact> S: (n) **center, centre** (a building dedicated to a particular activity)
 - *direct hyponym / full hyponym*
 - *direct hypernym / inherited hypernym / sister term*
- <noun.artifact> S: (n) **building, edifice** (a structure that has a roof and walls and stands more or less permanently in one place)
 - <noun.artifact> S: (n) **structure, construction** (a thing constructed; a complex entity constructed of many parts)
 - <noun.Tops> S: (n) **artifact, artefact** (a man-made object taken as a whole)
 - <noun.Tops> S: (n) **whole, unit** (an assemblage of parts that is regarded as a single entity)
 - <noun.Tops> S: (n) **object, physical object** (a tangible and visible entity; an entity that can cast a shadow)
 - <noun.Tops> S: (n) **physical entity** (an entity that has physical existence)
 - <noun.Tops> S: (n) **entity** (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

WordNet Search - 3.1

- [WordNet home page](#) - [Glossary](#) - [Help](#)

Word to search for: monk

Display Options:

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: <lexical filename> (gloss)

Noun

- <noun.person>S: (n) **monk**, monastic (a male religious living in a cloister and devoting himself to contemplation and prayer and work)
 - direct hyponym / full hyponym
 - has instance
 - direct hypernym / inherited hypernym / sister term
 - <noun.person>S: (n) **religious** (a member of a religious order who is bound by vows of poverty and chastity and obedience)
 - <noun.person>S: (n) **religious person** (a person who manifests devotion to a deity)
 - <noun.Tops>S: (n) **person, individual, someone, somebody, mortal, soul** (a human being)
 - <noun.Tops>S: (n) **organism, being** (a living thing that has (or can develop) the ability to act or function independently)
 - <noun.Tops>S: (n) **whole, unit** (an assemblage of parts that is regarded as a single entity)
 - <noun.Tops>S: (n) **object, physical object** (a tangible and visible entity; an entity that can cast a shadow)
 - <noun.Tops>S: (n) **physical entity** (an entity that has physical existence)
 - <noun.Tops>S: (n) **entity** (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

WordNet Search - 3.1

- [WordNet home page](#) - [Glossary](#) - [Help](#)

Word to search for: slave

Display Options:

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: <lexical filename> (gloss)

Noun

- <noun.person>S: (n) **slave** (a person who is owned by someone)
- <noun.person>S: (n) **slave, striver, hard worker** (someone who works as hard as a slave)
- <noun.person>S: (n) **slave** (someone entirely dominated by some influence or person)
 - direct hypernym / inherited hypernym / sister term
 - <noun.Tops>S: (n) **person, individual, someone, somebody, mortal, soul** (a human being)
 - <noun.Tops>S: (n) **organism, being** (a living thing that has (or can develop) the ability to act or function independently)
 - <noun.Tops>S: (n) **living thing, animate thing** (a living (or once living) entity)
 - <noun.Tops>S: (n) **whole, unit** (an assemblage of parts that is regarded as a single entity)
 - <noun.Tops>S: (n) **object, physical object** (a tangible and visible entity; an entity that can cast a shadow)
 - <noun.Tops>S: (n) **physical entity** (an entity that has physical existence)
 - <noun.Tops>S: (n) **entity** (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))
- <noun.Tops>S: (n) **causal agent, cause, causal agency** (any entity that produces an effect or is responsible for events or results)
- <noun.Tops>S: (n) **physical entity** (an entity that has physical existence)
- <noun.Tops>S: (n) **entity** (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

В некоторых случаях в базе данных недостаточно информации для определения близости двух слов, например, в случае отсутствия определения для одного из них. Для иллюстрации рассмотрим пару слов “Maradona” и “football” (группа Relatedness): их близость считается равной 0, так как для слова “Maradona” отсутствует какое-либо определение в базе данных WordNet.

Word to search for: Maradona

Display Options:

Your search did not return any results.

Группа слов Relatedness значительно шире, чем группа Similarity, так как она включает в себя пары слов, которые по отдельности могут иметь независимые значения, но в паре друг с другом быть связанными по смыслу, т.е. иметь некоторое “смысловое” родство. Вследствие этого мера Спирмена для группы слов Relatedness значительно ниже, чем для группы Similarity.

Рассмотрим, к примеру, пару слов “psychology” и “Freud”:

psychology	Freud	8,21	0,64	0,10	0,05	13,00	249,00	251,00	239,00
-------------------	--------------	------	------	------	------	-------	--------	--------	--------

С точки зрения человека, эти слова взаимосвязаны, так как “Freud” (имя), или **Фрейд**, является известным психологом, психоаналитиком и психиатром, т.е. имеет непосредственное отношение к понятию **психология** (англ. “psychology”). Однако система определяет их как совершенно разные сущности.

Word to search for:

Display Options:

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
 Display options for sense: <lexical filename> (gloss)

Noun

- <noun.cognition>S: (n) **psychology**, **psychological science** (the science of mental life)
 - [direct hyponym](#) / [full hyponym](#)
 - [domain term category](#)
 - [direct hyponym](#) / [inherited hyponym](#) / [sister term](#)
 - <noun.cognition>S: (n) [science](#), [scientific discipline](#) (a particular branch of scientific knowledge)
 - <noun.cognition>S: (n) [discipline](#), [subject](#), [subject area](#), [subject field](#), [field](#), [field of study](#), [study](#), [bailiwick](#) (a branch of knowledge)
 - <noun.cognition>S: (n) [knowledge domain](#), [knowledge base](#), [domain](#) (the content of a particular field of knowledge)
 - <noun.cognition>S: (n) [content](#), [cognitive content](#), [mental object](#) (the sum or range of what has been perceived, discovered, or learned)
 - <noun.Tops>S: (n) [cognition](#), [knowledge](#), [noesis](#) (the psychological result of perception and learning and reasoning)
 - <noun.Tops>S: (n) [psychological feature](#) (a feature of the mental life of a living organism)
 - <noun.Tops>S: (n) [abstraction](#), [abstract entity](#) (a general concept formed by extracting common features from specific examples)
 - <noun.Tops>S: (n) [entity](#) (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

Выводы

При выполнении данного задания были изучены принципы работы с системами nltk и WordNet, а также автоматические методы определения близости слов, такие как lch, wup и jcn, которые были применены к группам слов Similarity и Relatedness. В ходе работы была получена мера Спирмена относительно “WordSim 353 Goldstandard” (т.е. относительно человеческих оценок, или стандарта) для каждого метода и каждой группы слов – для Similarity были получены значения, близкие к ~0.60, что говорит о наличии осмыслинности результатов автоматизированных методов для этой группы слов, для Relatedness – значения, близкие к ~0.00, что означает неприменимость этих методов для данной группы слов. Также были подробно изучены причины тех или иных отклонений от человеческих оценок.

Так, для более точного определения близости слов следует использовать более сложные подходы, такие как нейронные сети и глубокое обучение, которые могут учитывать контекст и все нюансы значений слов.