به نام خدا

گزارش تمرین پنجم درس پردازش زبان طبیعی

استاد درس:

جناب دکتر برادران

نام و نام خانوادگی دانشجو:

امیررضا صدیقین

شماره دانشجویی:

۹۹۳۶۱۴۰۲۴

تمام کدها و خروجی‌های هر بخش در فایل جوپیتر مربوط به هر بخش در دسترس است. (هم خروجی ipynb و html)

بخش ۱

در این بخش فایل‌های sport.txt , tech.txt خوانده شده است.

بخش ۲ و ۳

در این بخش برای کلمات داده‌شده synset آن‌ها به دست آمد و name و definition(تعریف) و مثال‌هایی و hypernyms و part_of_speech آن مشخص شده است.

هر synset دارای فرمت name.pos.index است. که name خود کلمه و pos نوع کلمه که آیا اسم است یا فعل یا ... (part of speech) و index آن برای unique کردن آن است و صرفا برای تمایز بین synsetهایی است که name و pos یکسانی دارند.

بخش ۴

در این بخش تابعی به نام get_lemmas نوشته شده است که کلمه را در ورودی می‌گیرد و تمام lemma های تمام synset های کلمه را در قالب یک مجموعه (تکراری‌ها حذف می‌شود) برمی‌گرداند.

این تابع پارامتری به به نام print_lemmas است که در صورت true بود در هر مرحله به صورت جداگانه برای هر synset ، lemmaهای آن چاپ شده است.

سپس برای کلمات داده شده در صورت سوال این تابع اجرا شده است.

بخش ۵

در این بخش تابعی نوشته شده است که در آن هم‌معنا ها و متضادهای یک کلمه چاپ می‌شود. به این صورت که برای هم‌معنا ها lemma ها چاپ شده و برای متضادها از تابع antonyms روی هر lemma به دست می‌آید.

برای کلمات داده شده تابع اجرا شده است.

بخش ۶

در این بخش تابعی نوشته شده است که hypernyms و Hyponyms کلمه به دست می‌آید که در آن hypernyms و hyponyms برای تمام synset ها بدست می‌آید.

سپس برای کلمات داده شده این تابع اعمال شده است.

بخش ۷

بخش a:

در این بخش برای کلمات داده شده root_hypernyms برای synset اول (synsetهای گفته شده در صورت سوال) به دست آمده است.

محتویات جدول به صورت زیر پر می‌شود.

```
word = Dog      => ['entity.n.01']
word = Cat      => ['entity.n.01']
word = Car      => ['entity.n.01']
word = Bicycle => ['entity.n.01']
word = Tree     => ['entity.n.01']
word = Flower   => ['entity.n.01']
word = Water    => ['entity.n.01']
word = Rainbow => ['entity.n.01']
```

بخش b:

در این بخش برای زوج کلمات داده شده lowest_common_hypernyms به دست آمده است. (در این بخش از synsetهای گفته شده استفاده شده است.)

محتویات جدول به صورت زیر پر می‌شود.

```
for Dog & Cat       => ['carnivore.n.01']
for Car & Bicycle   => ['wheeled_vehicle.n.01']
for Tree & Flower   => ['vascular_plant.n.01']
for Water & Rainbow => ['abstraction.n.06']
```

بخش ۸

در این بخش path similarity و resnik similarity و jiang-conrath similarity برای دو به دو کلمات داده شده به دست می‌آید و در ماتریسی ذخیره می‌شود و به صورت جدولی نمایش داده شده است. همچنین تایع نوشته شده است که اتربیوتی که بیشترین شباهت داشته باشه در هر سطر را پیدا می‌کند و در قالب ستون most_similarity به دست آمده است.

که به صورت زیر جداول به دست آمده است.

path_similarities

In [56]:
```
df = pd.DataFrame(path_similarities , columns=words , index=words)
df["most_similarity"] = df.apply(lambda row:most_similary_word(row , df.columns) , axis=1)
df
```

Out[56]:

	Dog	Cat	Car	Bicycle	Tree	Flower	Water	Rainbow	most_similarity
Dog	1.000000	0.200000	0.076923	0.090909	0.125000	0.111111	0.083333	0.062500	Cat
Cat	0.200000	1.000000	0.055556	0.062500	0.076923	0.071429	0.058824	0.047619	Dog
Car	0.076923	0.055556	1.000000	0.200000	0.071429	0.066667	0.071429	0.055556	Bicycle
Bicycle	0.090909	0.062500	0.200000	1.000000	0.083333	0.076923	0.083333	0.062500	Car
Tree	0.125000	0.076923	0.071429	0.083333	1.000000	0.166667	0.076923	0.058824	Flower
Flower	0.111111	0.071429	0.066667	0.076923	0.166667	1.000000	0.071429	0.055556	Tree
Water	0.083333	0.058824	0.071429	0.083333	0.076923	0.071429	1.000000	0.076923	Bicycle
Rainbow	0.062500	0.047619	0.055556	0.062500	0.058824	0.055556	0.076923	1.000000	Water

resnik_similarities

In [57]:
```
df = pd.DataFrame(resnik_similarities , columns=words , index=words)
df["most_similarity"] = df.apply(lambda row:most_similary_word(row , df.columns) , axis=1)
df
```

Out[57]:

	Dog	Cat	Car	Bicycle	Tree	Flower	Water	Rainbow	most_similarity
Dog	9.006014	7.911667	1.531834	1.531834	2.224150	2.224150	0.801759	-0.000000	Cat
Cat	7.911667	9.040650	1.531834	1.531834	2.224150	2.224150	0.801759	-0.000000	Dog
Car	1.531834	1.531834	7.591401	6.452257	1.531834	1.531834	0.801759	-0.000000	Bicycle
Bicycle	1.531834	1.531834	6.452257	9.250664	1.531834	1.531834	0.801759	-0.000000	Car
Tree	2.224150	2.224150	1.531834	1.531834	7.764869	6.028316	0.801759	-0.000000	Flower
Flower	2.224150	2.224150	1.531834	1.531834	6.028316	8.295989	0.801759	-0.000000	Tree
Water	0.801759	0.801759	0.801759	0.801759	0.801759	0.801759	8.206018	0.596229	Flower
Rainbow	-0.000000	-0.000000	-0.000000	-0.000000	-0.000000	-0.000000	0.596229	12.856162	Water

jiang_conrath_similarities

In [61]:
```
df = pd.DataFrame(jiang_conrath_similarities , columns=words , index=words ,dtype="float32")
df["most_similarity"] = df.apply(lambda row:most_similary_word(row , df.columns) , axis=1)
df
```

Out[61]:

	Dog	Cat	Car	Bicycle	Tree	Flower	Water	Rainbow	most_similarity
Dog	inf	0.449776	0.073889	0.065820	0.081152	0.077799	0.064068	0.045741	Cat
Cat	0.449776	inf	0.073701	0.065670	0.080924	0.077590	0.063926	0.045669	Dog
Car	0.073889	0.073701	inf	0.253965	0.081350	0.077980	0.070453	0.048906	Bicycle
Bicycle	0.065820	0.065670	0.253965	inf	0.071675	0.069047	0.063079	0.045235	Car
Tree	0.081152	0.080924	0.081350	0.071675	inf	0.249736	0.069602	0.048494	Flower
Flower	0.077799	0.077590	0.077980	0.069047	0.249736	inf	0.067121	0.047277	Tree
Water	0.064068	0.063926	0.070453	0.063079	0.069602	0.067121	inf	0.050328	Car
Rainbow	0.045741	0.045669	0.048906	0.045235	0.048494	0.047277	0.050328	inf	Water

بخش ۹

بخش a:

در این بخش توکن‌های هر متن استخراج شده اند.

بخش b:

در این بخش کلمات stopwords و تک حرفی‌ها از توکن‌های استخراج شده حذف می‌شوند.

بخش c :

در این بخش typeهای توکن‌های هر متن به‌دست آمده است که در متن‌های sport ۱۱۰۱۰ تا type وجود دارد و در متن‌های tech ۱۳۴۰۴ تا type بدست آمده است.

بخش d:

در این بخش تعداد تکرار کلمات football و computer در هر متن بدست آمده است.

part 9-d:

In [24]:
```
for word in ['football','computer']:
    for text_name , tokens_list in [("sports",sport_tokens) , ("Technology" ,tech_tokens) ]:
        print(f"count of repeat {word} in {text_name} = {tokens_list.count(word)}")
```

```
count of repeat football in sports = 93
count of repeat football in Technology = 8
count of repeat computer in sports = 0
count of repeat computer in Technology = 299
```

بخش e:

در این بخش ماتریس term_document به دست آمده است که به ازای هر کلمه‌ی موجود در دو متن تعداد تکرار هر کدام در هر متن مشخص می‌شود.

In [27]:
```
term_doc_matrix
```

Out[27]:

	sport	tech
willingness	2	1
computer	0	299
recognising	0	1
Robotic	0	1
Defenders	1	1
...	...	...
firms	0	188
Partido	1	0
supremacy	0	3
debrief	1	0
vacuum	0	1

19378 rows × 2 columns

بخش f:

در این بخش شباهت کسینوسی برای کلمات زیر به دست آمده است. (دو به دو)

["football","sport","technology","computer","basketball","laptop","website"]

که شباهت کسینوسی به معنای 1- cos(alpha) است که منظور از alpha زاویه‌ی بین دو بردار است.

(ورودی تابع cosine دو تا بردار است.)

	football	sport	technology	computer	basketball	laptop	website
football	1.000000	0.988254	0.100045	0.085705	0.623970	0.085705	0.228332
sport	0.988254	1.000000	0.250923	0.236956	0.736062	0.236956	0.374434
technology	0.100045	0.250923	1.000000	0.999896	0.839953	0.999896	0.991542
computer	0.085705	0.236956	0.999896	1.000000	0.832050	1.000000	0.989570
basketball	0.623970	0.736062	0.839953	0.832050	1.000000	0.832050	0.903278
laptop	0.085705	0.236956	0.999896	1.000000	0.832050	1.000000	0.989570
website	0.228332	0.374434	0.991542	0.989570	0.903278	0.989570	1.000000

کلمه‌ی مرجع	کلمات	شباهت کسینوسی	بیشترین شباهت
Football	sport	0.988	Sport
	technology	0.1000	
	computer	0.085	
Sport	computer	0.236	basketball
	technology	0.250	
	basketball	0.736	
computer	basketball	0.832	laptop
	technology	0.999	
	laptop	1.000	
website	laptop	0.989	technology
	technology	0.991	
	football	0.228	