به نام خدا

دانشگاه صنعتی امیرکبیر

Amirkabir University
of Technology

پروژه اول درس بازیابی اطلاعات

روش‌های سنتی بازیابی اطلاعات

استاد درس: دکتر ممتازی

نام: زهرا اخلاقی

شماره دانشجویی: ۴۰۱۱۳۱۰۶۴

پاییز ۱۴۰۲

فهرست مطالب

فصل اول - بازیابی با استفاده از مدل فضای برداری

۱-۱ مقدمه

در این بخش با استفاده از روش فضای برداری اسناد مرتبط با پرسش‌ها را مشخص میکنیم. برای این کارسندها را به صورت بردار TF-IDF نمایش میدهیم و با استفاده از معیار فاصله کسینوسی اسناد مرتبط به هر کوئری را مشخص میکنیم.

۱-۲ پیش پردازش

با توجه به بررسی داده‌های تمرین و و تنوع متون سوالات، تصمیم بر آن شد تا اولین پیش‌پردازش انجام شده تبدیل متن به حرف کوچک و سپس حذف تمامی کاراکتر علائم نگارشی و تگ های html از متن میباشد و سپس حذف تمامی کاراکترهای غیرحرفی از متن می‌باشد. بنابراین، تمامی علائم نگارشی و اعداد و حروف زبان‌های دیگر از سوالات حذف شده است.

در مرحله بعد پیش پردازش داده‌ها عبارات مخفف را به کامل أنها جایگزین کردم و با استفاده از کتابخانه nltk کلمات که شامل stop word ها هستند را از متن حذف کردم گه این‌کار به باقی‌ماندن کلمات تاثیرگذار کمک می‌کند. این مسئله باعث می‌شود کلماتی در شباهت دو جمله برسی شوند که به احتمال بالاتری تاثیر کلیدی در معنی و مفهوم جمله داشته باشند.

گام بعدی در پیش پردازش متن استخراج ریشه کلمات است که باعث میشود شکل‌های مختلف از یک کلمه به یک صورت مشاهده شوند و اگر در دو جمله تکرار شدند، این تکرار در شباهت دو جمله تاثیرگذار باشد. این عمل با استفاده از کتابخانه nltk انجام شده است.

پیش پردازش داده‌ها در تابع preprocess پیاده سازی شده و روی ستون document در فایل hw1_docs.csv و ستون query در فایل hw1_queries..csv انجام شده و نتیجه به ترتیب در ستون‌های processed_document و processed_query ذخیره شده است.

۱-۳ پیاده سازی الگوریتم

در تابع compute_idf در ابتدا تعداد داکیومنت هایی که یک کلمه را شامل میشود شمارش شده و سپس مقدار idf برای هر کلمه محاسبه می‌شود.

با توجه به اینکه درمتن تمرین گفته شده برای روش فضای برداری طول بردار حداقل برابر ۱۰۰۰ باشد، بنابراین باید حداقل ۱۰۰۰ کلمه برتر را استخراج کنیم و مشخص نشده برتر از نظر چه معیاری در نظر گرفته شده است، با توجه به امتحان کردن حالات مختلف تصمیم بر این شد تا کلماتی که بیشترین دفعات مشاهده در پیکره را دارند (کمترین میزان idf) انتخاب شوند که درون تابع get_top_words انجام میشود.

در تابع compute_tf تعداد تکرار هر کلمه در هر متن شمارش میشود، برای شمارش تعداد کلمات تنها کلمات برتر در نظر گرفته شده‌اند و در نهایت در تابع compute_tfidf بردار نهایی ساخته میشود.

با توجه به فرمول شباهت کسینوسی که برابر است با ضرب داخلی دو بردار تقسیم بر حاصلضرب اندازه آنهاست، میتوان دریافت که در محاسبه صورت فقط مواردی که از دو بردار در یک مکان خاص مقدار غیر صفر دارند مهم هستند. این مسئله در این سناریو به این معنای حضور کلمه مشخص شده در هر دو متن کوئری و داکیومنت می‌باشد.

بنابراین به جای چک کردن کلمات برتر، کلمات مشترک در کوئری و داکیومنت چک شده و tfidf آنها در یکدیگر ضرب شده و به حاصلضرب داخلی اضافه میشود، اما برای محاسبه میزان کل اندازه دو بردار علاوه بر کلمات مشترک هر دو بردار نیاز به کل کلمات دو بردار است، این بخش در توابع compute_cosine_tfidf و compute_full_cosine انجام میشوند.

۴-۱ نتیجه

با توجه به توابع mean_reciprocal_rank , mean_average_precision , precision_at_k تمامی روش‌های پیاده‌سازی شده با استفاده از این توابع ارزیابی خواهند شد. این توابع نشانگر معیارهای ارزیابی MRR, MAP, P@5 , P@10 هستند.

با در نظر گرفتن ۲۰۰۰ کلمه برتر، نتایج معیار های ارزیابی به صورت زیر می‌باشد:

MRR:0.879

MAP:0.7636926146384476

P@5:0.6600000000000003

P@10:0.57

فصل دوم - بازیابی با استفاده از مدل احتمالاتی BIM

۲-۱ مقدمه

در این بخش با استفاده از مدل احتمالاتی BIM اسناد مرتبط با پرسش‌ها را مشخص میکنیم. در این مدل وجود یا عدم وجود عبارت در یک متن یا پرسش در نظر گرفته میشود و ارتباطات میان عبارت در این مدل در نظر گرفته نشده است.

۲-۲ پیش پردازش

پیش پردازش در این الگوریتم مشابه روش بیان شده در فصل قبلی پیاده‌سازی شده است.

۲-۳ پیاده سازی الگوریتم

برای پیاده سازی این الگوریتم در ابتدا در دیکشنری index برای هر کلمه اسناد شامل آن کلمه ذخیره میشود و سپس در تابع RSV_weights مقدار RSV برای هر عبارت محاسبه می‌شود، سپس در تابع RSV_doc_query ارتباط هر پرسش را به داکیومنت مشخص می شود، برای انجام این کار کلمات مشترک در داکیومنت و پرسش RSV آنها به عنوان رتبه با یکدیگر جمع میشود.
در تابع compute_full_bim به ازای تمامی پرسش ها، ارتباط آن با هر سند با استفاده از تابع RSV_doc_query محاسبه می‌شود.

۲-۴ نتیجه

نتایج معیارهای ارزیابی MRR, MAP, P@5 , P@10 با در نظر گرفتن pt های متفاوت به صورت زیر است:

pt = 0.3 ->

MRR : 0.80

MAP : 0.6982617315444696

P@5: 0.56

P@10: 0.46

pt = 0.5 ->

MRR: 0.8589999999999999

MAP: 0.7632948696145125

P@5: 0.6480000000000001

P@10: 0.516

pt = 0.7 ->

MRR:0.7872142857142856

MAP:0.712228341521794

P@5:0.5880000000000003

P@10:0.486

به ازای pt=0.5 نتایج دقت بهتری دارند.

فصل سوم - بازیابی با استفاده از مدل احتمالاتی BM25

۳-۱ مقدمه

در این بخش با استفاده از مدل احتمالاتی BM25 اسناد مرتبط با پرسش‌ها را مشخص میکنیم. در این مدل تعداد تکرار هر کلمه در هر سند و طول سند را برای تعیین ارتباط یک سند با یک پرسش در نظر گرفته می‌شود و از چارچوب بازیابی احتمالی پیروی می کند، که فرض می کند اسناد مرتبط و غیر مرتبط از توزیع های آماری متفاوتی پیروی می کنند.

۳-۲ پیش پردازش

پیش پردازش در این الگوریتم مشابه روش بیان شده در فصل اول، پیاده‌سازی شده است.

۳-۳ پیاده سازی الگوریتم

در پیاده سازی این الگوریتم در توابع compute_tf و compute_idf به ترتیب مقادیر tf و idf محاسبه می‌شود و سپس متوسط طول داکیومنت ها محاسب می‌شود.
در تابع calculate_bm25_score ارتباط هر سند با هر پرسش مشخص می‌شود و در تابع compute_full_bm25 به ازای همه‌ی پرسش ها ارتباط آنها با هر سند مشخص می‌شود.

۳-۴ نتیجه

نتایج برای b= [0,0.5,1] و k = [0.5,1,2,10] به صورت زیراست:

k1=0.5 , b=0 => MRR:0.81 --- MAP:0.73 --- P@5:0.60 --- P@10:0.50
k1=0.5 , b=0.5 => MRR:0.87 --- MAP:0.81 --- P@5:0.69 --- P@10:0.59
k1=0.5 , b=1 => MRR:0.90 --- MAP:0.79 --- P@5:0.70 --- P@10:0.59
k1=1 , b=0 => MRR:0.82 --- MAP:0.76 --- P@5:0.64 --- P@10:0.53
k1=1 , b=0.5 => MRR:0.89 --- MAP:0.81 --- P@5:0.72 --- P@10:0.58
k1=1 , b=1 => MRR:0.87 --- MAP:0.79 --- P@5:0.68 --- P@10:0.58

k1=2 , b=0 => MRR:0.84 --- MAP:0.79 --- P@5:0.66 --- P@10:0.53

k1=2 , b=0.5 => MRR:0.88 --- MAP:0.80 --- P@5:0.70 --- P@10:0.60

k1=2 , b=1 => MRR:0.87 --- MAP:0.79 --- P@5:0.67 --- P@10:0.57

k1=20 , b=0 => MRR:0.87 --- MAP:0.76 --- P@5:0.63 --- P@10:0.50

k1=20 , b=0.5 => MRR:0.84 --- MAP:0.77 --- P@5:0.67 --- P@10:0.55

k1=20 , b=1 => MRR:0.79 --- MAP:0.74 --- P@5:0.66 --- P@10:0.54

نتایج اختلاف کمی با یکدیگر دارند بهترین نتیجه زمانی است که b=0.5 , k1=2 باشد , زمانی که k1=20 و یا b=0 است بدترین نتایج وجود دارد.

تاثیر پارامتر های مدل:

b: پارامتر b طول سند را کنترل می کند. مقدار بین 0 و 1 به معنای طول کمتر است، در حالی که مقدار بیشتر از 1 به معنای طول بیشتر است. مقدار این پارامتر بر میزان تأثیر طول یک سند بر رتبه بندی آن تاثیر می گذارد.

k1: این پارامتر تاثیر tf را کنترل می کند. مقدار بالاتر k1 مدل را بیشتر بر tf وابسته می کند و مقادیر معمول برای k1 بین 1.2 و 2.0 است