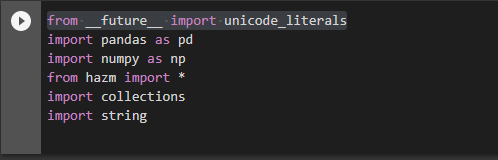
به نام **خدا**

""گزارش فاز اول پروژه بازیابی اطلاعات""

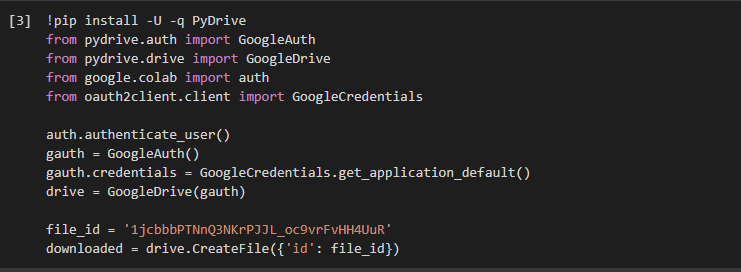
محمدرضا قادری 9627057

در این پروژه از کتابخانه‌های numpy (برای برخی محاسبات احتمالی مورد نیاز ماتریسی) و pandas ( برای ارتباط با داده‌ها و خواندن و نوشتن در فایل‌ها) و همچنین از کتابخانه hazm برای برخی از قسمت‌های پیش پردازشی ابتدایی برای اسناد استفاده شده است.



* 1. **پیش پردازش اسناد**

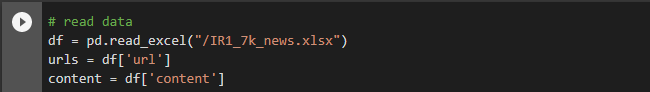
در این قسمت ابتدا بایستی از فایل دیتاهای مورد نظر را بخونیم. برای این کار روش‌های زیادی مانند openpyxl و pandas و تعدادی دیگر از کتابخانه‌ها موجود هست که برای این پروژه از pandas استفاده کردیم. با توجه به این که از google colab استفاده کردم برای اینکه بتونم از فایل بخونم از google drive استفاده کردم تا به فایل دسترسی داشته باشم بایستی یک دسترسی به colab با آیدی فایل بدهم تا استفاده کند از آن.



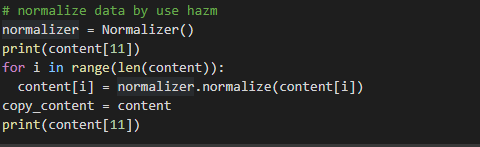
سپس با دانلود فایل می‌تونیم ازش استفاده کنیم.



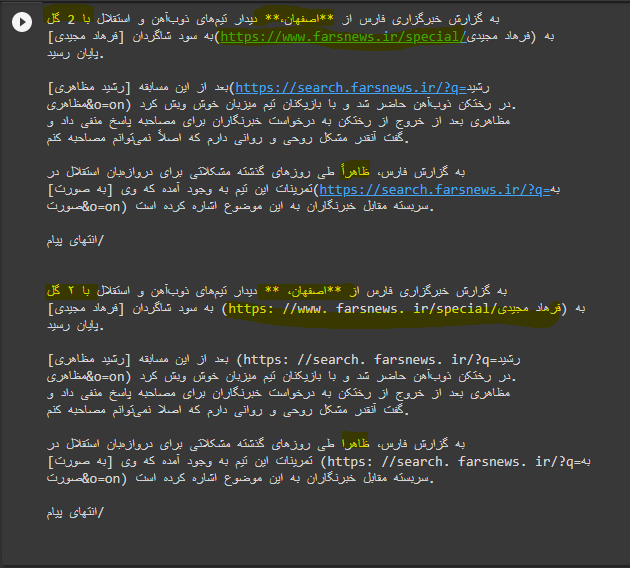
با استفاده از کتابخانه pandas شروع به خواندن فایل می‌کنیم چون در ادامه بیشتر با ستون content کار داریم فقط فعلا از این ستون و urls اون‌ها استفاده می‌کنیم.



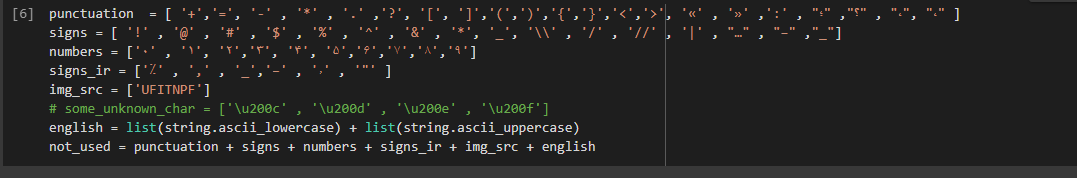
در ادامه ابتدا از normalizer کتابخانه hazm استفاده می‌کنیم که در ابتدا کمی content ها رو نرمال کنیم چرا که برخی نگارش‌های اشتباه مانند شماره‌ها که به زبان فارسی نیستند و برخی نیم فاصله وبرخی از تنوین‌های زبان عربی و تعدادی دیگر از اشتباهات رایج را با استفاده از این نرمالایزر تغییر می‌دهد.



نمونه‌ایی از تغییر در content را برای خبر 9 به صورت زیر است.

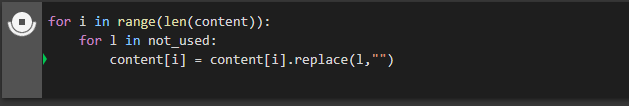


ولی این نگارش‌ها برای ما کافی نیست هنوز وجود کلمات اینگلیسی و نشانه گذاری‌هایی مانند پرانتز و کروشه و غیره و حتی اعداد و تمامی علائم نگارشی (که با توجه به اینکه از normalizer استفاده کردیم خیلی از علائم به فرسی تبدیل شده اند مانند علامت سوال و ویرگول) پس این‌ها نیز بایستی از متن ما خارج شوند.

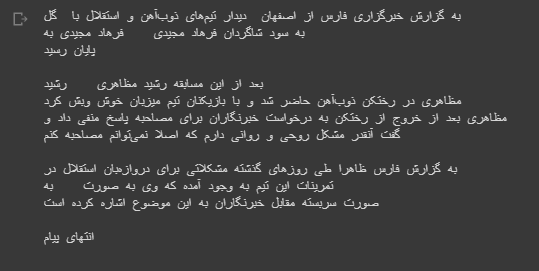


برای کلمات و حروف خارجی از کتابخانه string استفاده کردیم و تمامی حروف را به صورت بزرک و کوچک چک کردیم و تنها حالت متفاوت UFITNPF بود که normalizer در تبدیل آدرس عکس به جا می‌گذاشت.

حالا رو تمامی content ها حرکت می‌کنیم و به جا موارد بالا “” جاگذاری می‌کنیم. نکته‌ای که جا موند انتهای پیام بود که در شکل بالا نیومده است ولی در کد اضلی استافده شده چرا که بنظر می‍‌رسید که فقط برای اطلاع از پایان متن بوده.

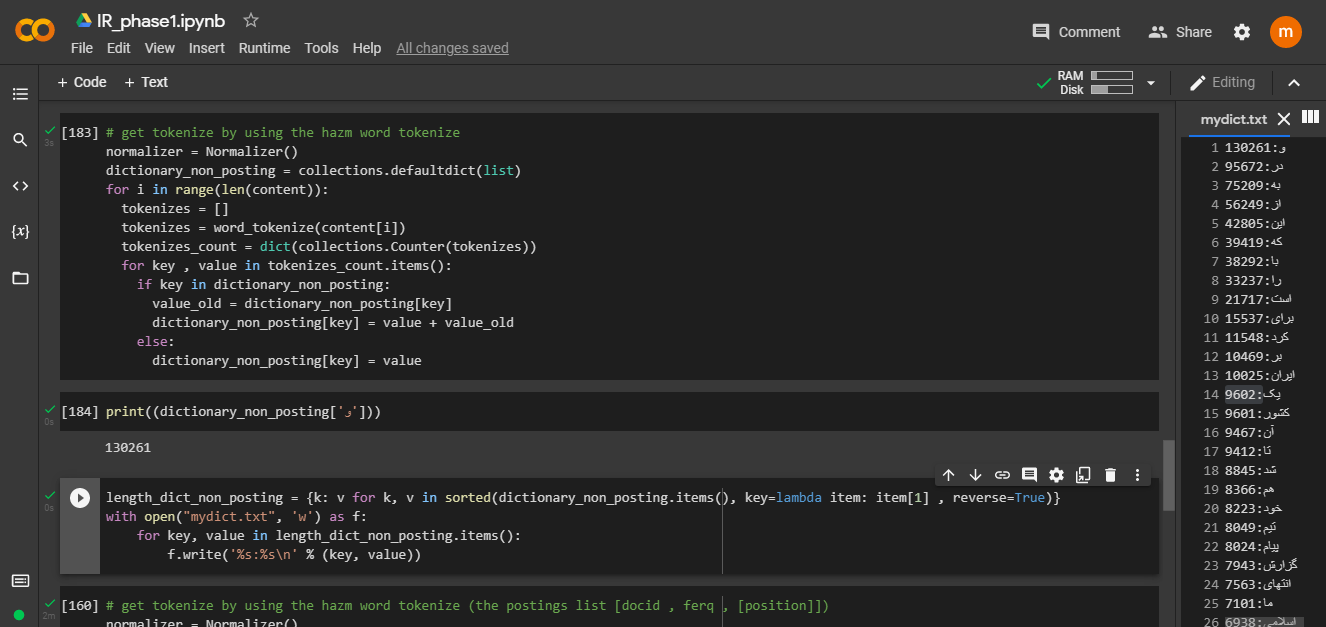


حالا با استفاده از نگارش بالا متن قبلی به صورت زیر در آمد.(که تمامی موارد مورد نظر حذف شد)



پس تمامی متن‌ها برای تحلیل بیشتر آمده شده است.

برای جذف کلمات پرتکرار (stop word) ها ابتدا بایستی تو کل متن‌ها فرکانس هر کلمه را بدست آوریم تا بتوانیم در مورد کلماتی پرتکرار مثل و، در، به، از، این و ... تصمیم بگیریم. برای این منظور از کتابخانه collections برای شمردن تکرار در هر مت به این صورت که وقتی word\_tokenize انجام می‌دهیم یک لیست از کلمات به ما می‌دهد میتوانیم تکرار را با کمک تابع counter از کتابخانه بالا بدست آوریم سپس در همه متن‌ها میتوانیم تعداد تکرار را به همین صورت بدست آورد.



برای مثال و به تعداد 130261 بار تکرار شده در این حالت برای اطمینان 100 تا کلمه اول را چک می‌کنیم و کلماتی Stop word هستند را حذف می‌کنیم.(فایل mydict تکرارها را دارد) حالا یک آرایه تهیه می‌کنیم از stop wordها تا برای لیست کردن از آن‌ها استفاده نکنیم.

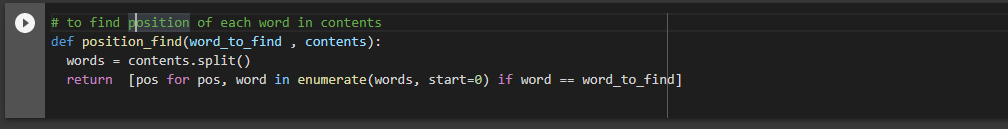
برای اینکه کلمات بالا فقط در متون به کثرت استفاده می‌شوند میتوانیم آنها را از درون dictionary خود کنار بگذاریم (چرا که در کوئری های سرچ خیلی استفاده نمی‌شوند و چون در اکثر متون هستند فرقی برای ما نمی‌کند چرا که مرج می‍شوند)



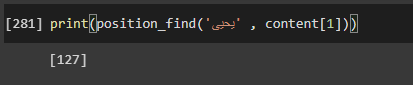
این یک روش برای stop word هاست روش دیگر استفاده از کتابخانه hazm هست که خوش یک لیستی از stop word ها تهیه کرده.که ما با توجه با اینها و اینکه ریشه یابی کلمات در کل کانتنت رو تغیر میدهیم و از همین طریق dictionary مورد نظر خود را می‌سازیم . برای ریشه یابی از stem که در کتابخانه hazm موجود است استفاده می‌کنیم.

بعد از در نظر نگرفتن این کلمات می‌توانیم با یافتن مکان هر کلمه در متن که موجود است نسبت به کلمه ابتدایی آنها رو به عنوان لیستی برای هر کلمه ذخیره کرد. از انجایی که برای ترتیب اهمیت اینکه یک کلمه در یک متن چندبار تکرار شده از یک لیست سه تایی برای هر متن استفاده می‌شود، پس هر key که کلمه مورد نظر ماست دارای value لیست سه تایی به ازای هر متنی که در آن موجود باشد دارد که اولین عضو لیست شماره آن متن و دمین تعداد تکرار آن کلمه در متن و سومین عضو لیست position‌های آن کلمه در آن متن نسبت به ابتدای آن متن هست(کلمه به کلمه).

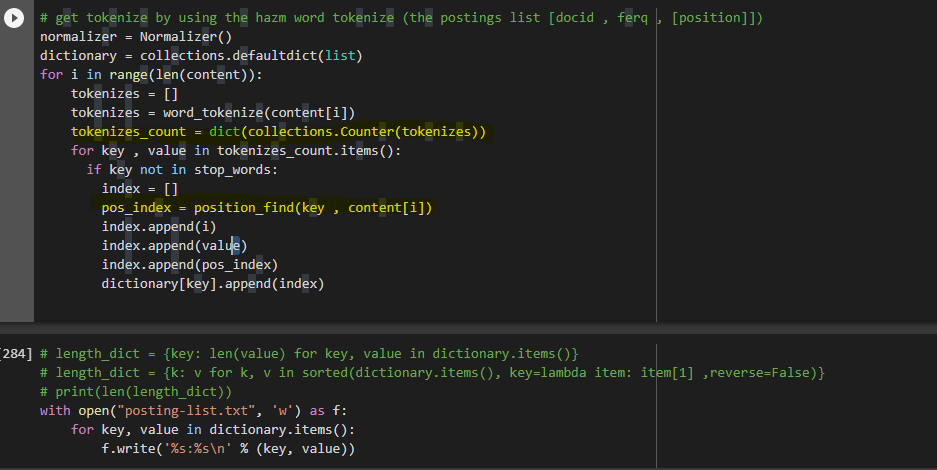
برای بدست آوردن جایگاه یک کلمه در متن از قطعه کد زیر استفاده شده



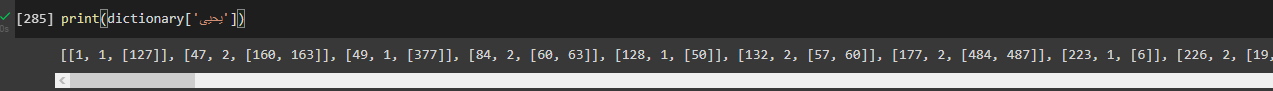
که متن را می‌گیرد و به نسبت به آن کلمه میشمارد در کدام موقعیت‌ها استفاده شده ولیستی برمی‌گرداند. برای مثال کلمه یحیی در متن 1(که بهتر هست متن رو به صورت تکه تکه شده کلمات بفرستیم) (که در اصل دومین متن اطلاعات ماهست) در یک جا و position 127 از ابتدا قرار گرفته است.



در این حالت پس برای ساخت شاخص مکانی اقدام می‌کنیم با این تفاسیر این شاخص را در دیکشنری نگه می‌داریم تا نتوانیم به راحتی و باسرعت به آن دسترسی داشته باشیم در ابتدا یکی یکی متن‌ها رو می‌خوانیم و کلمات آن را بدست می‌آوریم و کلمه به کلمه جلو می‌رویم و با تابع بالا لیست مکان‌های آن را میابیم . الان هر سه فاکتور مورد نظر را داریم پس می‌توانیم شاخص مکانی بسازیم.



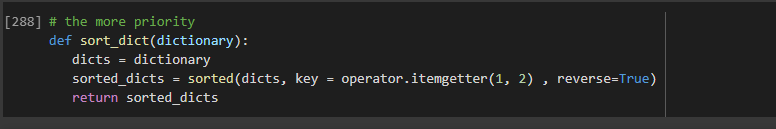
برای مثال کلمه یحیی رو شاخص مکانی برایش بدست آوردیم.



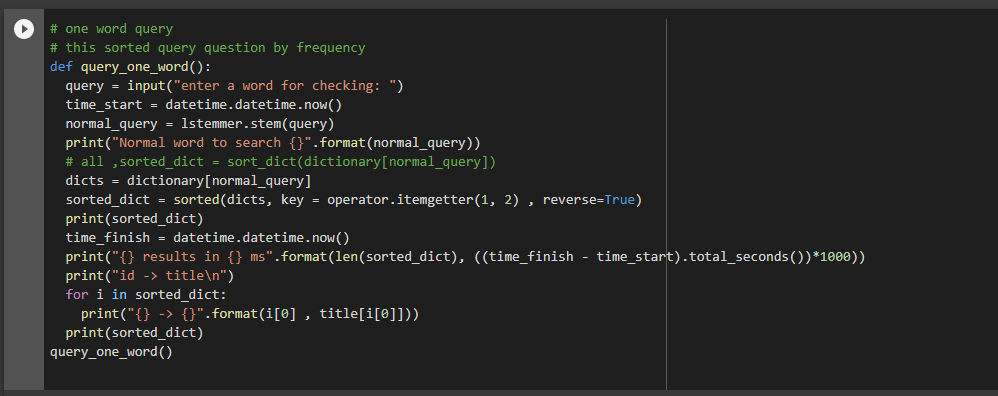
که میبینم در متن اول در مکان 127 و فقط یک بار و در متن 47 در 2 جا آمده است.

برای اینکه بتوانیم کمی منطقی‌تر عمل بهتر است نسبت به تعداد تکرار کلمه در متن آن کلمه رو با ترتیب برای کاربر نشان دهد چرا که درصد بیشتری برای اینکه درست باشد دارد.

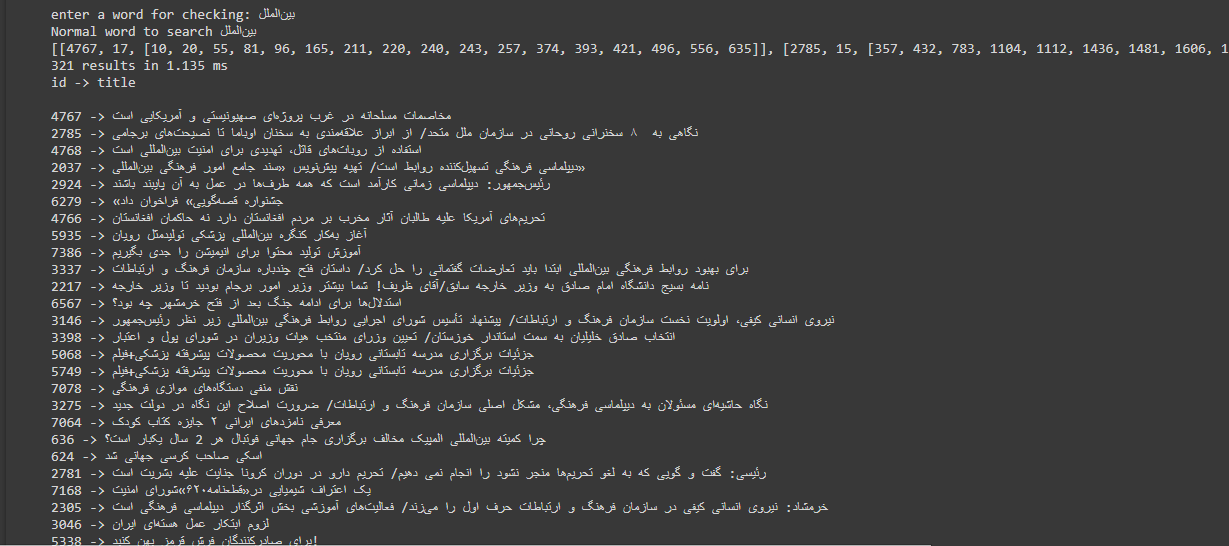
برای این کار یک تابع به صورت زیر داریم که دیکشنری ما رو بر اساس مقدار دو که فرکانس در هر متن هست مرتب می‌‌کند.



حالا اماده برای بدست آوردن جواب‌های پرسمان هستیم برای پرسمان تک کلمه‌ای تنها کافی هست ما از کاربر یک ورودی میگیریم ابتدا بایستی این کلمه رو نرمال کنیم وسپس در داشته‌های خودمان دنبال کلمه مشابه آن باشیم. سپس در دیکشنری مربوطه به دنبال کلمه نرمال شده می‌گردیم تا اینکه مکان‌های مربوط به آن را بیابیم.

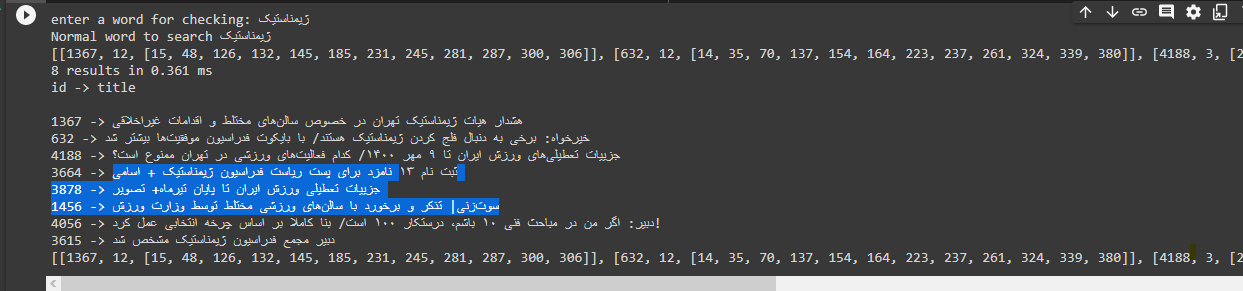


در حالت بالا سه تایی که در دیکشنری داشتیم را بر حسب دومی که تعداد تکرار در اون متن هست مرتب میکنیم برای خروجی بین الملل این کوئری به صورت زیر است

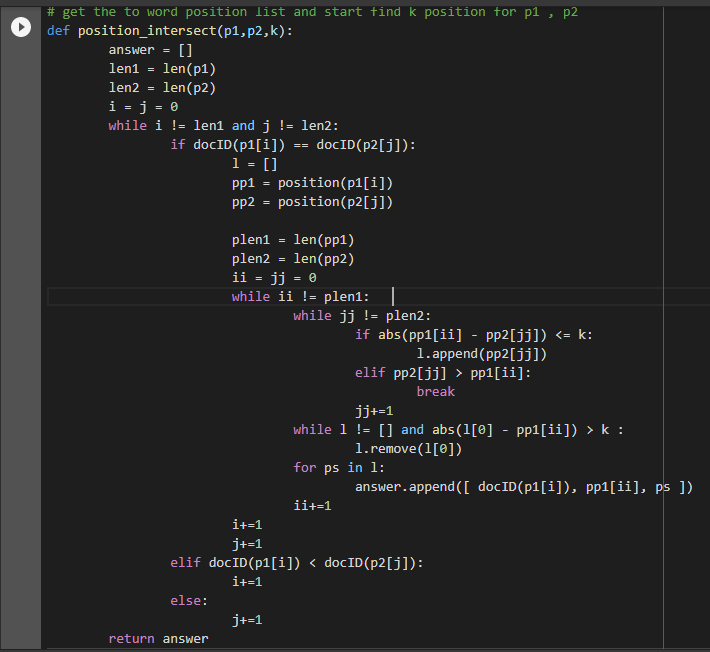


همونطور هم که در شکل مشخص هست شکل نرمال همان بین‌الملل هست و در متن 4767 (در اصل 4769) 17 تکرار برایش ثبت شده و همینطور به پایین پس جواب به صورت مرتب شده است در 7 تا متن اول جواب‌ها حدودا شبیه به مقادیری که در متن بود هست.

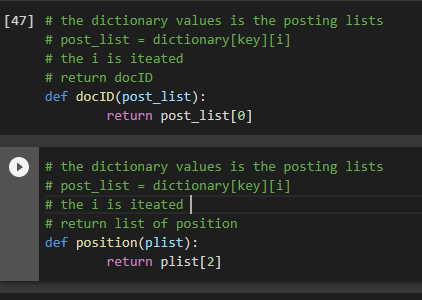
برای ژیمناستیک تنها 8 پاسخ پیدا شد که همانطور که گفته شد مرتب شده است.



در ادامه میبایستی برای کوئری های چندتایی بایستی از position intersection استفاده کنیم که به صورت زیر است و موقعیت مکانی رو مرج میکند.

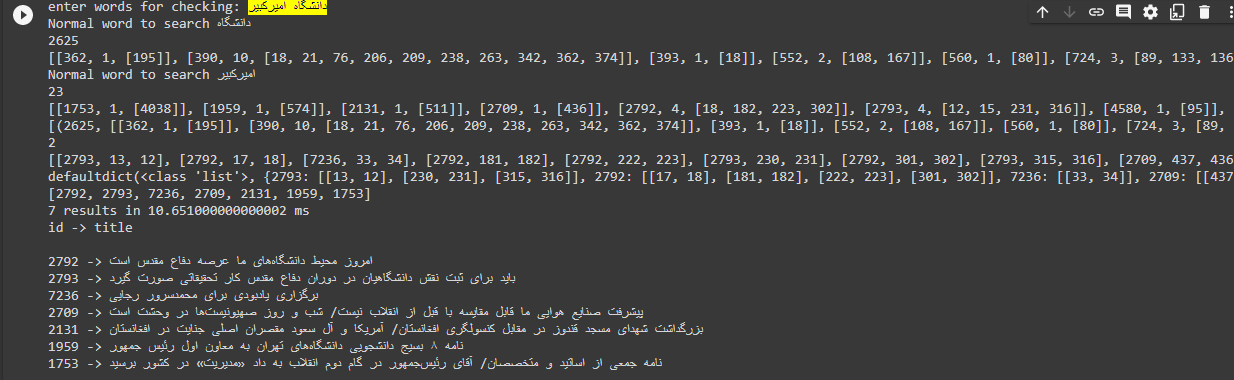


و برای گرفتن doc id و لیست موقعیت‌ها از ای دو استفاده شده

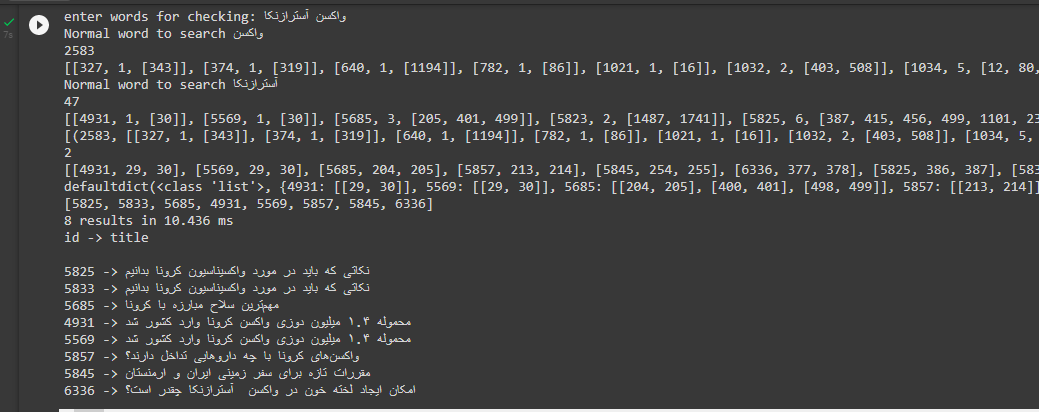


در کوئری‌های دوتایی کنار هم بودن دو کلمه برای ما ارزش دارد پس از position intersection استفاده می‌کنیم تا با k = 1 وجود دوتایی آن‌ها را بیابیم

برای دانشگاه امیرکبیر پاسخ به صورت زیر بوده است که با توجه به اینکه نیاز به مرج و مقایسه بین دو intersection position زمان بر تر خواهد بود و تعداد کمتری جواب خواهد داشت اینجا هم تعداد پاسخ‌ها کمتر از قبل و تعداد تکی این کلمات هست.



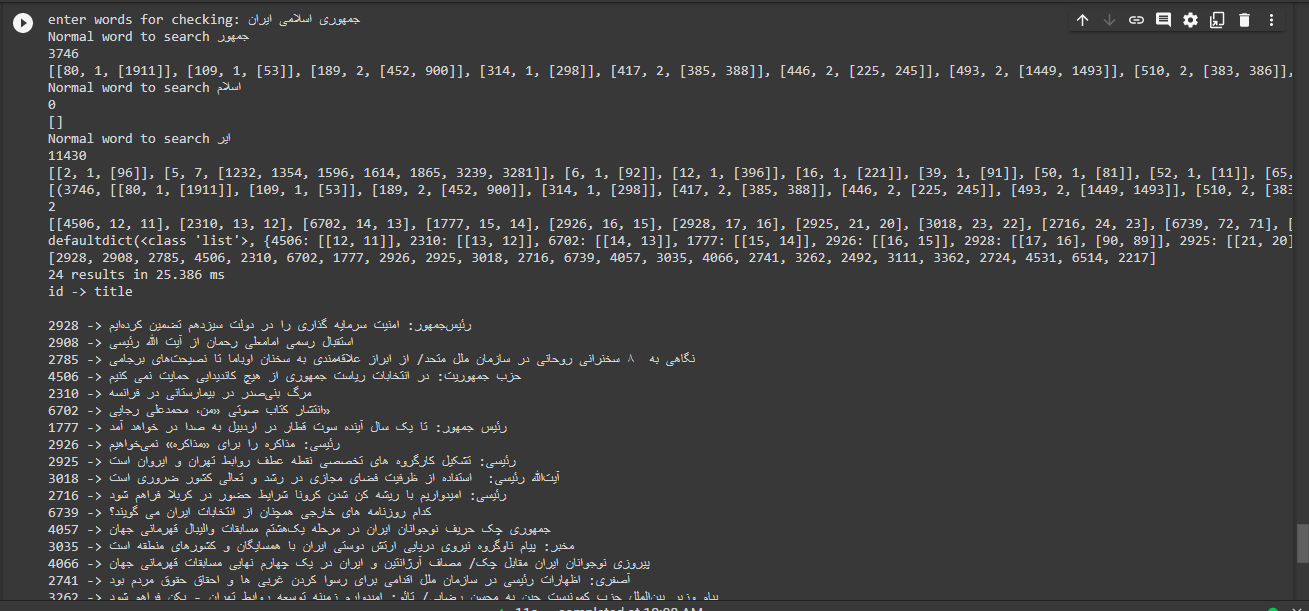
برای کلمه واکسن آسترازنکا هم پاسخ به شکل زیر بوده



در ابتدا با توجه به لینک ها به صورت سورت شده که در این لینک‌ها به صورت سورت شده بنابر تعداد آمدن این دو کنار هم در لینک‌هاست. به مانند قبلی برای مرج کردن زمان بیشتری برده است.

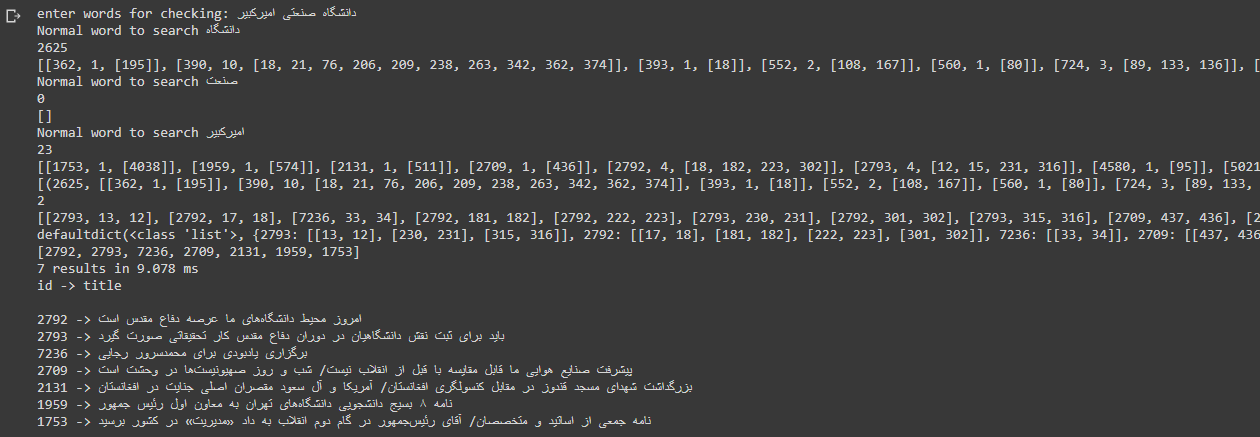
برای سه تایی ها و بیشتر دوتایی دوتایی کنار هم رو می‌توانیم بگیریم و با position intersection موقعیت مکانی‌های اون دوتا را با هم بدست بیاریم و سپس باید در متن‌های یکی دنبال اون مکان‌های بگردیم که کلمه مشترکشون یک جا باشد تا پاسخ دهد(این کار رو میتوان به صورت بازگشتی انجام داد)

برای مثال جمهوری اسلامی ایران

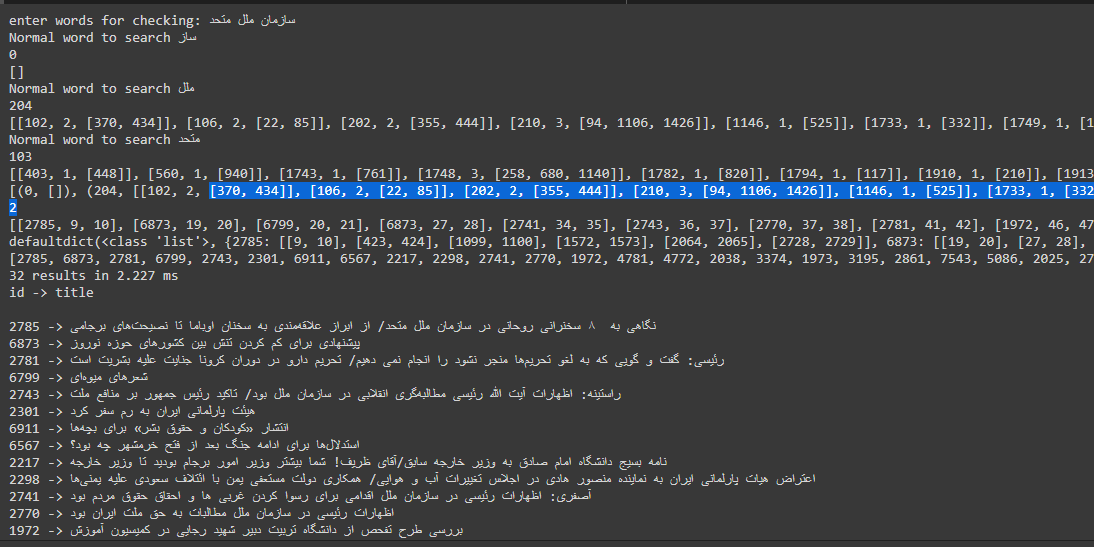


در اینجا جمهوری به جمهور و اسلامی به اسلام و ایران به ایر نرمال شده حالا در کل متون اسلام وجود ندارد به صورت خالی پس فقط جمهور و ایر رو مرج می‌کنیم و در نهایت لیست پایانی را سورت می‌کنیم.

در ادامه دانشگاه صنعتی امیرکبیر هم به مانند بالا اتفاق می‌افتد با توجه به متن‌هایی که بازگردانده شده مشخص شده که اطلاعات بازگردانده شده درست است.



در اخرین کوئری سازمان به ساز نرمال شد و بقیه به همان فرم ماندند



با توجه به پاسخ‌ها وجود سازمان ملل متحد در تییتر اولین لینک (و صد البته حضور نام آقای روحانی ^\_^) نشان دهند درست بودن جوابها دارد.

سوال 2) با توجه به اینکه ما رابطه log از Cfi ها رو میخواهیم و این رابطه به صورتی هست که بایستی به صورت خطی باشد با توجه به این که قبل از اینکه بخواهیم stop word را حذف کنیم حدودا شبیه و متمایل به یک رابطه خطی هست میتوان گفت صدق میکند(با توجه به اینکه فقط چندین stop wordکمی وجود داشت که فاصله زیادی داشت انتظار همچین چیزی می‌رفت)

سوال3) در قانون heap رابطه خطی با log آن دارد مقادیر k , b دارد که حدودا هر دو ثابت هستند البته که امکان تغییر برای k رو داریم که می‌توانیم فاصله برای کلمات در متن از هم استفاده کرد و تعداد کلمات می‌شود.

سوال4) توی ریشه‌یابی یکی از مسائل مهم فعل بودن یا کلمه بودن بود که مثال بارز آن است بود که اگر ابتدا به عنوان کلمه در نظر گرفته می‌شد به اس تبدیل می‌شد و فعل (Lemmatizer) تاثیری بر آن نداشت.به طور کلی ت در انتهای کلمات که حذف می‌شد میتوانست برای فعل ها مشکل ساز باشد البته ها و ... هم حذف می‌شد.کلمه اعلام که به وفور در خبرها استفاده می‌شود به اعلا با نرمال کردن تغیر میکند که کلمه‌ای با معنی جداست. یکی از مشکلات دیگه کلمه سازمان بود که در نرمال کردن آن به ساز تبدیل می‌شد و باعث می‌شد به فعل تبدیل شود.فعل‌های که به ان ختم میشوند مثل نمی‌توان هم به نمی‌تو تبدیل میشد که با stop ها به مشکل میخورد.

سوال5) و سوال 1) هم در متنهای بالا در موردشان صحبت شد.

با تشکر از توجه شما \*\_\*