

امیرحسین احمدی ۹۷۵۲۲۲۹۲ - تمرین یازدهم

۱- برای قسمت کانولوشنی:

لایه اول Conv2D خروجی : $64 * 24 * 24$ پارامترها : $64 * (7 * 7 + 1)$

لایه اول MaxPool2D خروجی : $64 * 12 * 12$ پارامترها : 0

لایه دوم Conv2D خروجی : $128 * 8 * 8$ پارامترها : $128 * (5 * 5 + 1)$

لایه دوم MaxPool2D خروجی : $128 * 4 * 4$ پارامترها : 0

لایه سوم Conv2D خروجی : $256 * 2 * 2$ پارامترها : $256 * (3 * 3 + 1)$

لایه سوم MaxPool2D خروجی : $256 * 1 * 1$ پارامترها : 0

لایه Flatten خروجی : 256 پارامترها : 0

لایه Dense بعد از Flatten خروجی : 128 پارامترها : $128 * (256 + 1)$

برای قسمت RNN خروجی برابر با تعداد یونیت ها و تعداد پارامترها برابر با تعداد ورودی به علاوه

یک(بایاس) ضرب در تعداد یونیت ها و به علاوه تعداد یونیت به توان ۲.

بنابراین تعداد پارامترهای لایه های RNN از سمت چپ به ترتیب برابر با 49280، 6324352، ...

است. خروجی لایه های dense باقی مانده نیز برابر با تعداد یونیت ها و تعداد پارامترها برابر با تعداد

ورودی به علاوه یک در تعداد یونیت است. لایه concat نیز پارامتری ندارد و فقط خروجی دو dense

لا کنار هم میگذارد که خروجی اش برابر میشود با ۲۵۶.

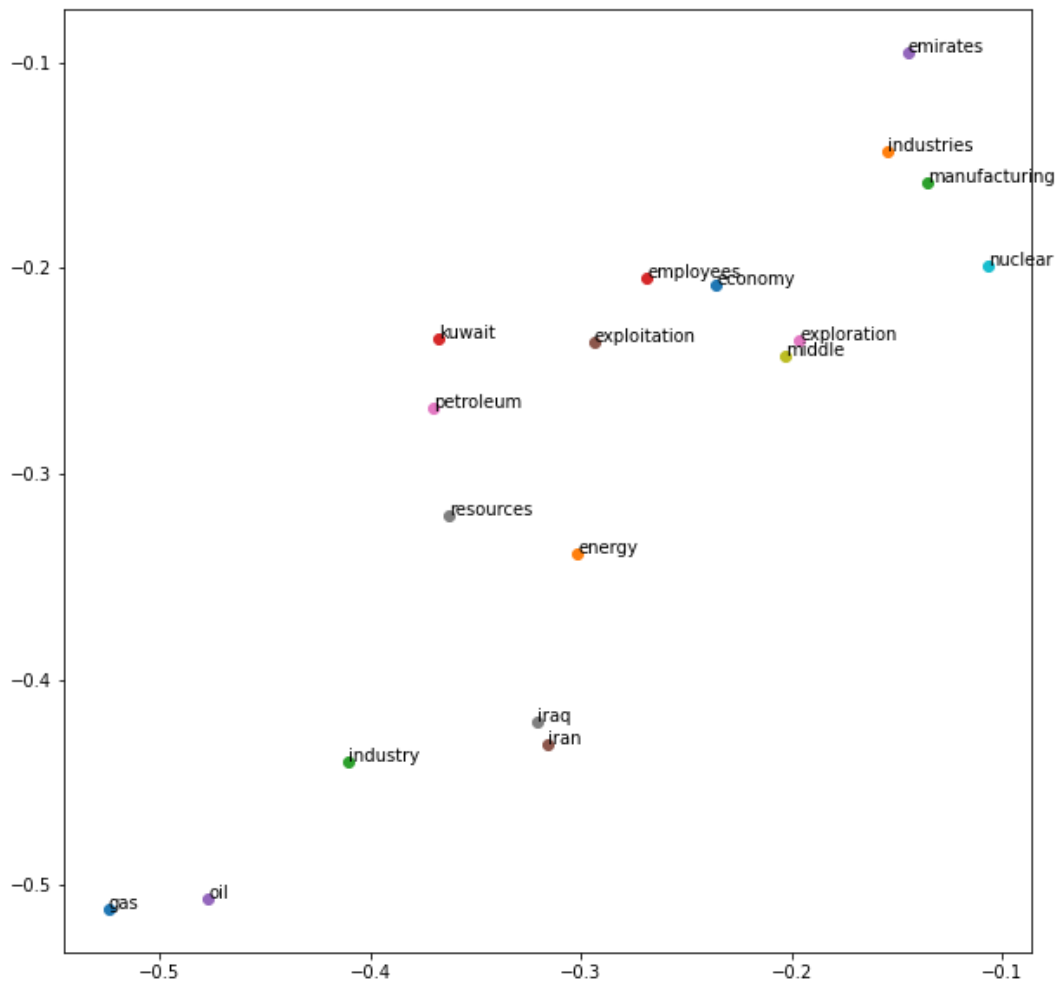
۲- در این سوال ابتدا دیتابیس را دانلود کرده و سپس طبق لینک اول داده شده آن را خوانده و ۱۰۰

دیتای اول را به عنوان دیتای train نگه میداریم. سپس مانند کد موجود در لینک دوم داده شده عمل

کرده و از این داده ها ۱۰۰ خبر را میسازیم و X و Y را برای train بدست میآوریم. سپس مدل را

ساخته و آن را ۱۰ epoch آموزش میدهیم. در نهایت وزن ها را گرفته و embedding ها را بدست

میاوریم. نتایج روی نمودار بردن کلمات خواسته شده به شکل زیر در آمد.



همانطور که میبینید کلمات *iraq* و *iran* نزدیک به هم قرار گرفته اند، ولی *kuwait* و *emirates* در فاصله نسبتاً زیادی از این دو قرار دارند که میتوان حدس زد اخبار ایران و عراق به دلیل مشکلاتی که دارند نزدیک به هم است و کشورهای پول دارتر مانند *kuwait* و *emirates* اخبار متفاوتی دارند. سوالات تئوری:

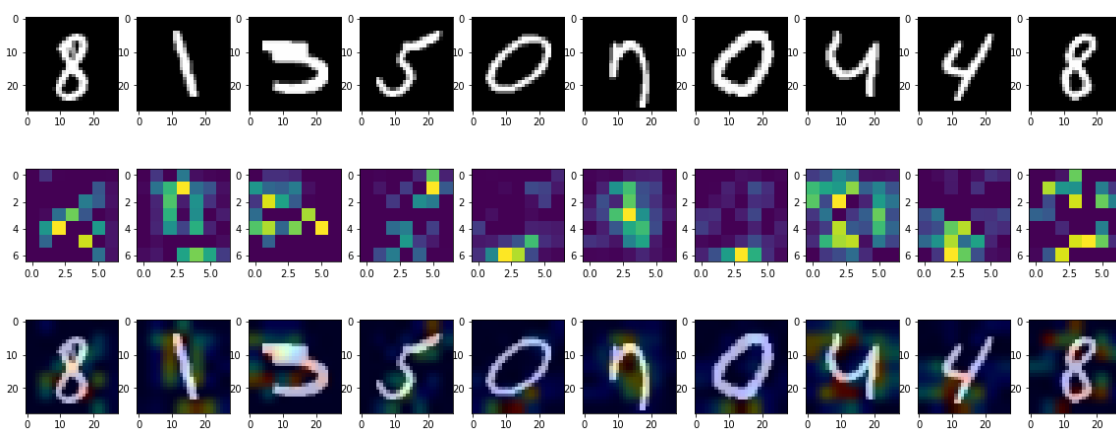
- بیشتر پیش پردازش های روی متن ها برای حذف **stop words** ها مانند *the, a* ... است زیرا **stop words** ها معمولاً تکرار زیادی دارند و باعث کم اهمیت شدن کلمات کلیدی میشوند.
- داده آموزشی مدل در قالب فولدرهای مختلف برای کتگوری های مختلف دسته بندی شده است. همچنین میتوان در صورت نیاز داده های موجود در چند دسته را استفاده کرد.
- اگر **window size** را کاهش دهیم باعث میشود که ارتباط کلمه را با کلمات کمتری ببینیم و روابط بین کلمات با دقت کمتری در میآید ولی این در حالیست که خود کلمه اهمیت زیادی پیدا

می‌کند و مثلاً تکرار آن با اهمیت می‌شود. همچنین با افزایش window size ارتباط کلمات با هم بهتر درمی‌آید ولی اهمیت خود کلمه کمتر می‌شود و کلمات به هم نزدیک تر می‌شوند.

- بردار های word embedding بعد از train شدن و از وزن های مول بدست می‌آید.

۳- برای سوال سه ابتدا دیتای mnist را گرفته و بعد از shuffle همانطور که خواسته شده ۱۰ تا را به عنوان نمونه نمایش می‌دهیم و ابعاد دیتاهای train و test را نمایش داده ایم. در ادامه برای استفاده از لایه های کانولوشنی لازم بود دیتای ورودی را reshape کنیم، همچنین اعداد تصاویر را نرمالایز کرده و از ۰ تا ۲۵۵ به بین ۰ تا ۱ می‌آوریم و y، train و test را به صورت کتگوریکال در می‌آوریم. در تابع build_model مدل را همان طور که در سوال خواسته شده ساخته و summary آن را نمایش می‌دهیم. در نهایت هم مدل را train کرده و دقت تست آن را بدست می‌آوریم که برابر با ۹۹ درصد می‌باشد که عدد خوبیست و میتوان دید مدل به خوبی ترین شده.

قسمت اصلی این تمرین اجرای الگوریتم grad cam بر روی لایه آخر کانولوشنی است. این الگوریتم در واقع برای نمایش عملکرد لایه ها انجام می‌شد که بتوان دید که چگونه فیچر ها را از تصاویر استخراج میکنند. برای این کار ۱۰ تا داده ی اول دیتای train را در نظر می‌گیریم، hitmap مربوط به لایه مورد نظر را بدست می‌آوریم و سپس آن را برای بررسی بهتر در کنار عکس قرار می‌دهیم. نتیجه بدست آمده را میتوانید در زیر ببینید.



همانطور که می‌بینید برای عدد ۳ که دو دندان بالایش به هم چسبیده مدل به خوبی تشخیص داده و تاثیر فیچرهای بالا بیشتر است و همچنین برای صفر های درون تصویر بیشتر فیچرهای اطراف تصویر را تاثیر داده است.