Implement Code:

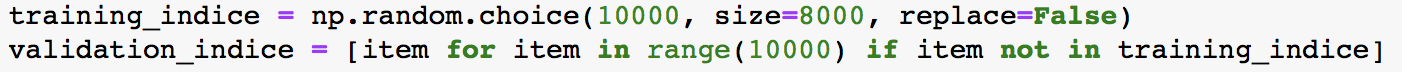
The files we have:

Training files: train\_description, train\_tags, train\_fc1000, train\_pool5, train\_image

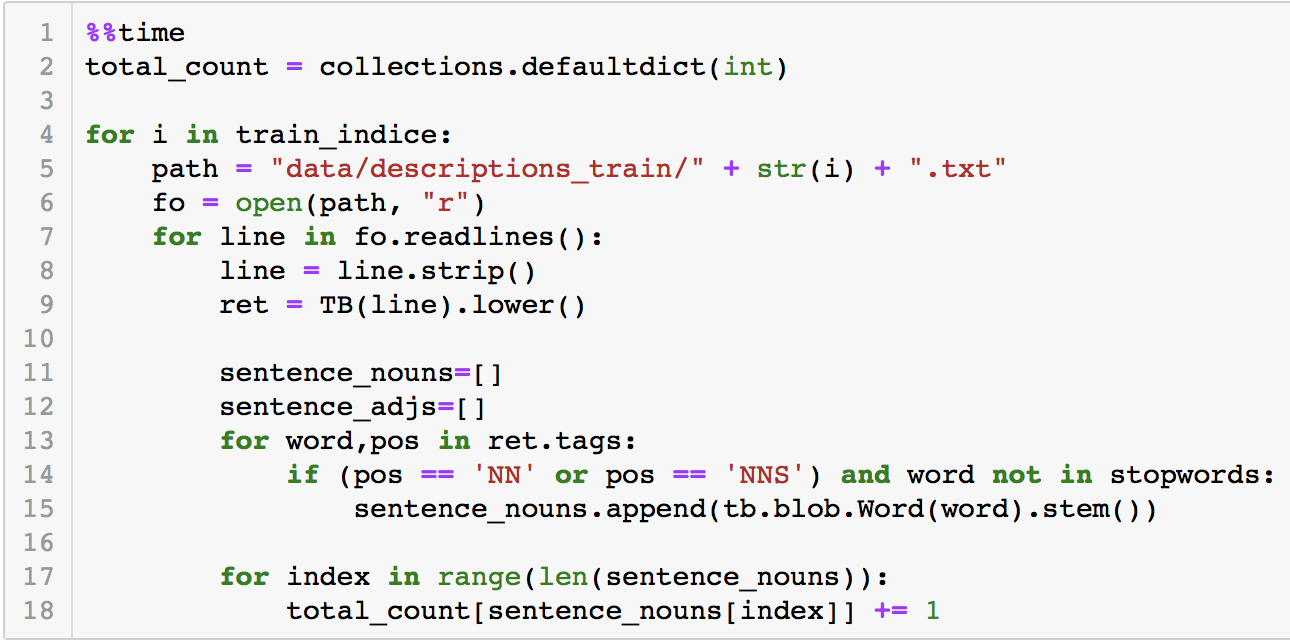
Testing files: test\_description, test\_tags, test\_fc1000, test\_pool5, test\_image

Note: test\_description is not in the same order of test\_image. Test\_fc1000 and test\_pool5 need is not sorted by image index (need to be sorted).

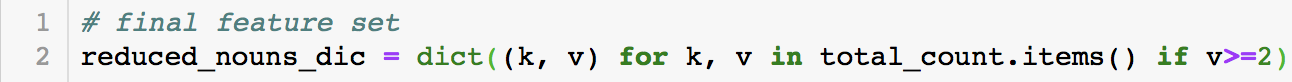
1. Mapping BoW to tags.
2. Splitting the 10000 training data to 8000 for training and 2000 for testing.



1. Read tag files into a dictionary\_list, where each element is a dictionary for that tag file, and each dictionary record the tag categories appeared in this tag\_file, and the tags appeared under those categories in this tag file.
2. Extract the categories and it’s tags from all tag files, and find out that there are totally 12 tag categories: “vehicle, furniture, outdoor, indoor, food, accessory, kitchen, electronic, sports, appliance, person, animal.” Each categories have several tags belongs to it.
3. Building labels based on the tag files.
4. Forming BoW space:
   1. Use TextBlob to extract nouns from all description files:



* 1. The obtained BoW space is too large, we decided to filter out some nouns which are not very important. After several experiments, nouns that only appear once in the whole description files are deleted.

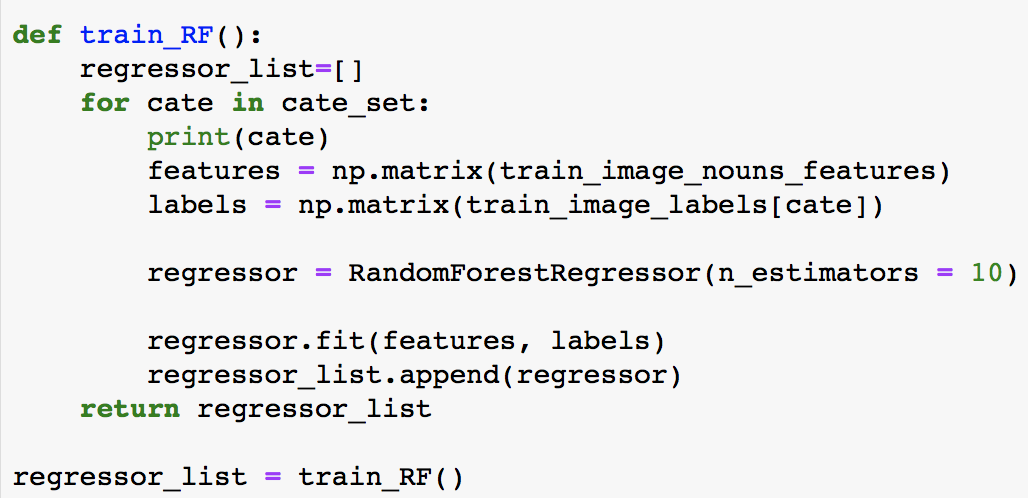


* 1. Our final BoW space has 2982 dimension.

1. Generate input matrix:

Finally, we form the input matrix by extract the nouns from each description file, based on the BoW space.

1. Training the Model:

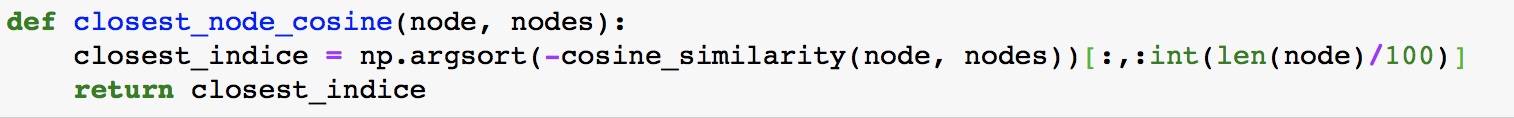


1. Computing the distance:

We tried two ways for computing the distance between the predicted output and the ground truth output.

1. Euclidean distance: with 0.23 score
2. Cosine similarity distance: with 0.255 score

So the cosine similarity distance is finally applied:

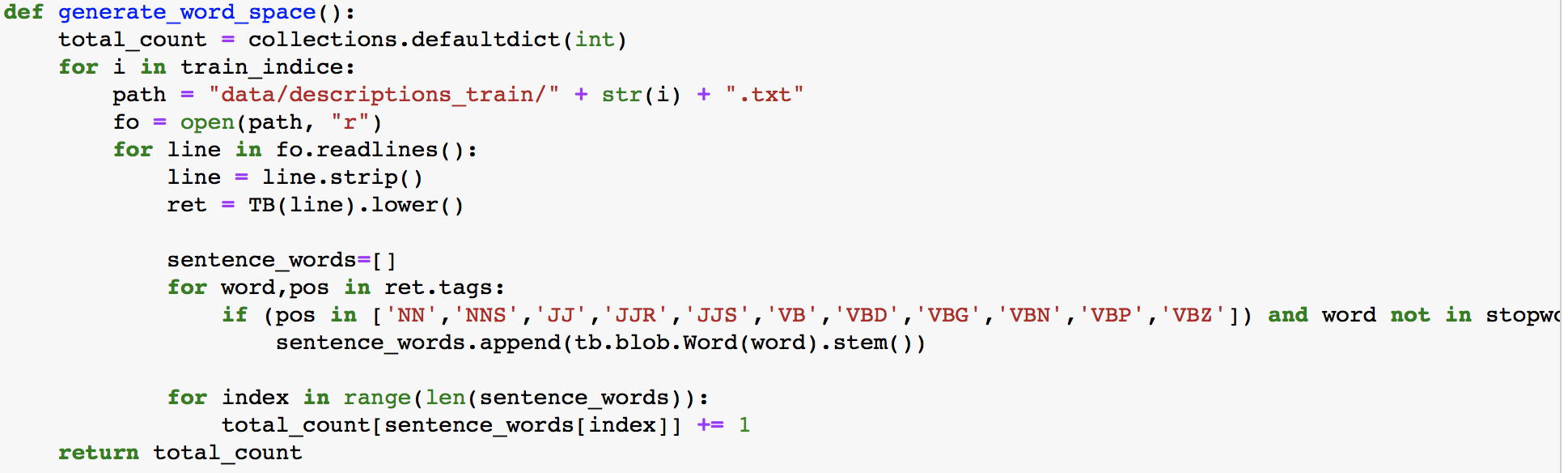


We also picked the first 20 images with the smallest distance to the predicted data and finally forming the submission file.

1. Mapping pool5 to BoW:
2. Since the pool5 file doesn’t order the image correctly, we re-ordered the records in the pool5 file into the correct order, where the first line match image\_0, second line match image\_1, etc.
3. Extract the BoW from description file:

Since the pool5 might have some information of the relationship among objects. Only extracting nouns should not be enough. So, besides extracting nouns, adjective and verb are also extracted to form the BoW space.

Also, se count the word frequency at the same times and finally we filter out the words that only appear less than three times.



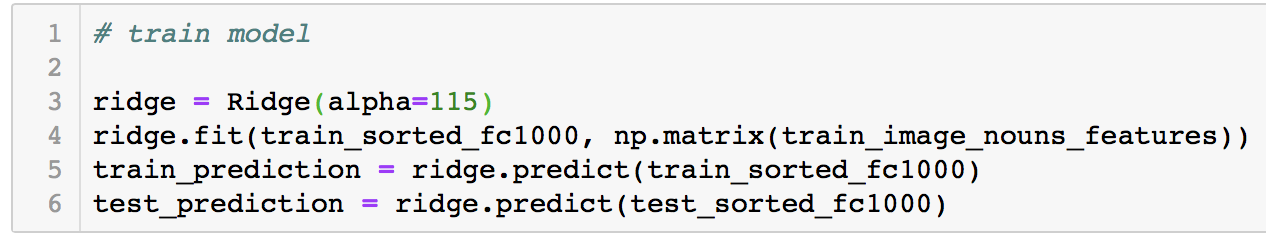
The obtained BoW space has 3138 dimensions.

1. Generate the output matrix (BoW matrix) from all description files based on the BoW space formed in the last step.



1. Preparing the input matrix: the input matrix should be exactly the pool5 matrix extracted from the pool5 file (that is the fc1000intermedian file)
2. Training the model:

We apply the Ridge Regression model since it trained fast and perform well. After testing, we select the parameter “alpha = 115”, which perform the best.



1. Matching the nearest 20 images:

After applying this model on all 2000 test images, we got 2000 piece of BoW vector. Comparing the distance from the predicted BoW vector to the ground truth BoW vector (extracted from the test input). And we got the first 20 images based on the Euclidean distance. (We tried cosine similarity distance but the it performs not as good as Euclidean distance this time)

1. Forming the submission file!