Cross-Modal Multimedia Retrieval for Image Search Engines

Akshay Jha, Disheng Zheng, David Hachuel
Cornell Tech
Cornell University
New York, NY 10044
{aj545, dz336, dh649}@cornell.edu

Abstract

In this paper, we investigated different approaches to build a large-scale image search engine. Given a natural language query, the task involves searching for relevant images. The fundamental problem in this challenge boils down to bridging the textual and image representation of the same entity or scene. To this end, we combined image features extracted from intermediate layers in a pre-trained Residual Network (**ResNet**) and text descriptions of the images to train our models. We experimented with multilayer perceptrons and partial least square regressions and proposed a meta model that combines the output of several models. This approach proved successful when compared to the rest of submissions in the Kaggle leaderboard.

1 Introduction

1.1 Problem Definition

Our challenge was to develop a search engine. The input would be a set of natural language query(ies) and the output ought to be a shortlisted list of images, rank ordered by likelihood of relevance. Thus, our output was expected to perform well on two relevance metrics, namely, shortlisting images and rank ordering.

Fundamentally, our problem consisted of building a text pre-processing function to process the query and extract relevant, usable features. This would be fed into a set of models that ingest these features and output the most relevant images for the query. Subsequently, these models would be ensembled to combine the shortlisted images from all the models. Finally, there would be a ranking mechanism to ensure that the final set of images is ordered by likelihood of relevance.

Our training set consisted of 10K images of size 224x224 and their corresponding fc1000 and pool5, which are features extracted from a CNN. These are compressed (down from 50,176 dimensions to 1,000 for fc1000 and 2,048 for pool5), noise free and therefore ideal for model development. Secondly, we also have a set of 5 sentence descriptions corresponding to each image, expounding on the content of the image. Thirdly, we have tags that have been provided in the format of category: subcategory. These human-labelled tags correspond to the objects or items in the image.

In the test set we have been provided 2K images with the same features as the training set, except that there was no correspondence between an image, its features and descriptions, tags. The relevance we desire in the output is informed by training set: therefore, our goal for model building was to capture

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

the mapping between the images (their features) and their corresponding descriptions and tags given in the training set. In this paper, we present an ensemble of models that combine to give a boosted performance over each of the individual models, thereby validating our proposed ensemble as well as ranking technique.

1.2 Related Work

We reviewed papers on cross-modal retrieval i.e. techniques studying mapping query data to image data. A comprehensive survey on cross-modal retrieval by Kaiye Wang et al. [1] suggests using PLS as an unsupervised method for real-valued representation learning.

We have also explored techniques and work in Latent Dirichlet allocation, for the purpose of compressing our bag of words of description. Latent Dirichlet Allocation by David Blei et al. [2] explains how to model a collection as a finite mixture of over an underlying set of topic probabilities. We also use the nltk library: corpus to remove stop-words in pre-processing our queries to create bag of words. However, there were limitations of directly using this technique, such as the assumption of topic distribution to be of a Dirichlet prior, which means that there is sparsity in terms of the number of topics. This means that the documents are assumed to cover only a small set of topics and that topics use only a small set of words frequently- however, we have a wide variety of topics in our set of images.

2 Data Flow and Process Architecture

We began by pre-processing the raw description documents to form a training and test set of bag-of-words (see Figure 1 for reference). This involves cleaning the documents, labeling parts of speech, removing stop words, stemming and counting frequencies. Given the high dimensionality of the bag-of-words representation, we reduced dimensionality through Principal Components Analysis. We then proceed to tested several modeling approaches trying to find the best bridge between textual representation and image representation. For each approach, we generated a new sample and found the top 20 closest neighbors in the reference data using K Nearest Neighbors with a cosine measure of distance. The latter was suggested in Wang, et al. [1] and outperformed the default Euclidean distance measure.

Moreover, tags are great intermediate resources used to map descriptions to relevant pictures. They capture essential objects that could be found in a picture. Also, tags are highly likely to appear in the description. Therefore, we created a bag-of-words representation of the tags data which include 91 unique tags. All tags are formatted as category:subcategory (ex. animal:cow).

Category only captures the high-level class of the object which, as the above example “animal”, may not appear in the description. Therefore, when creating the bag of words, subcategory tags are given a higher weight than category tags. Tag space needs to be mapped to description space. As before, PCA with 2048 components was applied to the description bag of words to reduce noise and computation expense. Tag space only has 91 columns, it does not need PCA processing.

3 Modeling Approaches

Cross-media retrieval aims to find a common representation in the entities during training (see Figure 2 for reference). Wang, et al. [1] proposed a series of supervised and unsupervised models. Given that we only have co-occurrence information to learn the common representation across different media, we focus our analysis on unsupervised methods. In particular, we tested partial least squares regression (PLS) and multilayer perceptron (MLP).

3.1 Partial Least Squares Regression

Partial least squares regression (PLS regression) is an extension of the multiple linear regression model and bears some similarity with principal components analysis. Given two sets of observations, X and Y, PLS finds a projection into a common subspace with some specified dimensionality so that

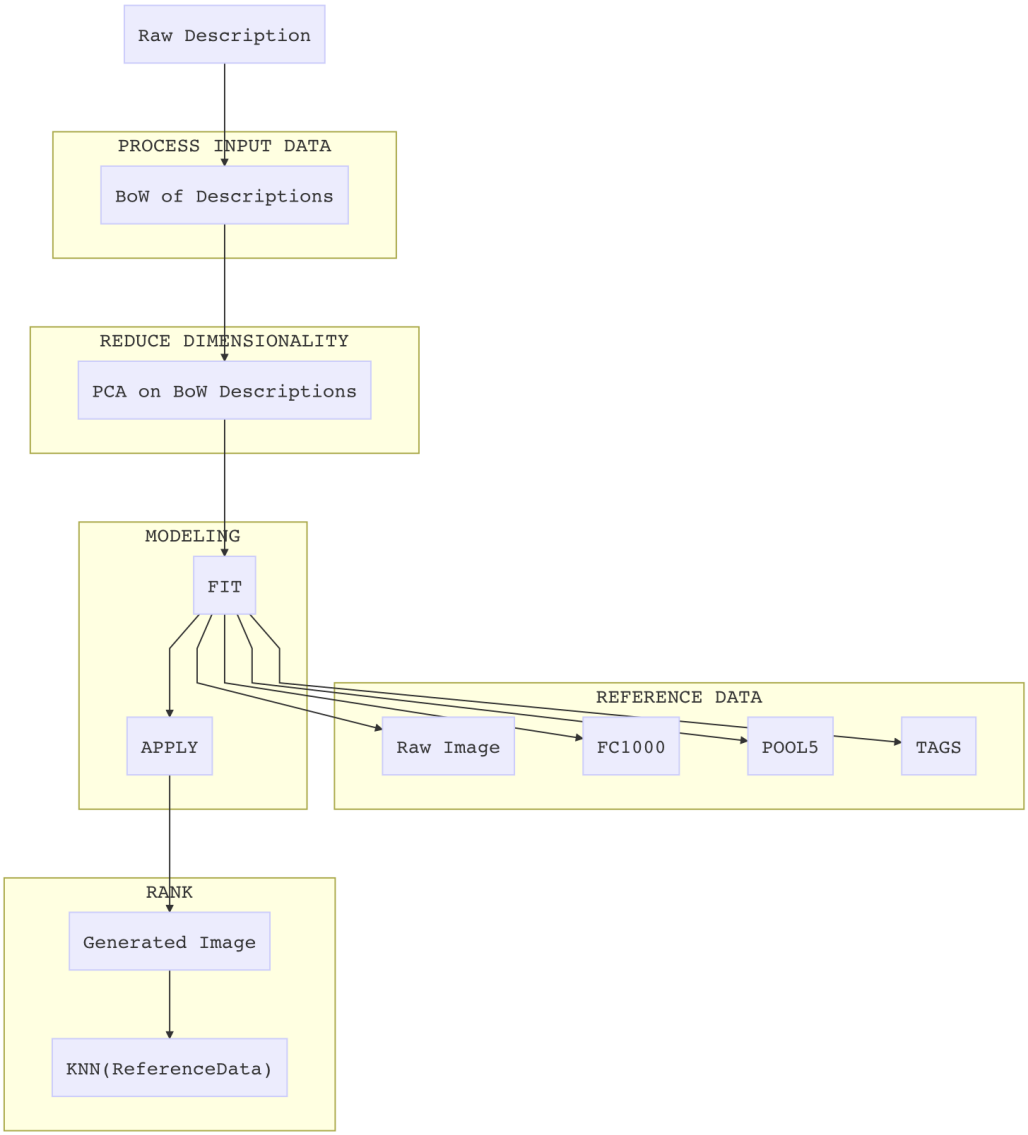

Figure 1: Data flow diagram.

the relationship between each pair of observations or co-occurrence is as strong as possible. It seeks the direction in the subspace that maximizes the variance and finds a linear regression model.

We attempted this approach with the following setups:

- PLSR(PCA(BoW description nouns, 1000), FC1000) with 1000 components
- PLSR(PCA(BoW description, 2048), POOL5) with 2048 components
- PLSR(PCA(BoW description, 2048), BoW tags) with 91 components

For each approach, we observe a boost in performance and accuracy score when reducing the dimensionality of the bag-of-words description through PCA. In addition, we observed that increasing the number of components in the PLS contributes to an increase in our accuracy score.

In addition, we subset the bag-of-words description to only nouns for both FC1000 and the tags since they both only relate to nouns as far as parts of speech. In the case of POOL5, we assumed more richness in the representation and allowed all parts of speech.

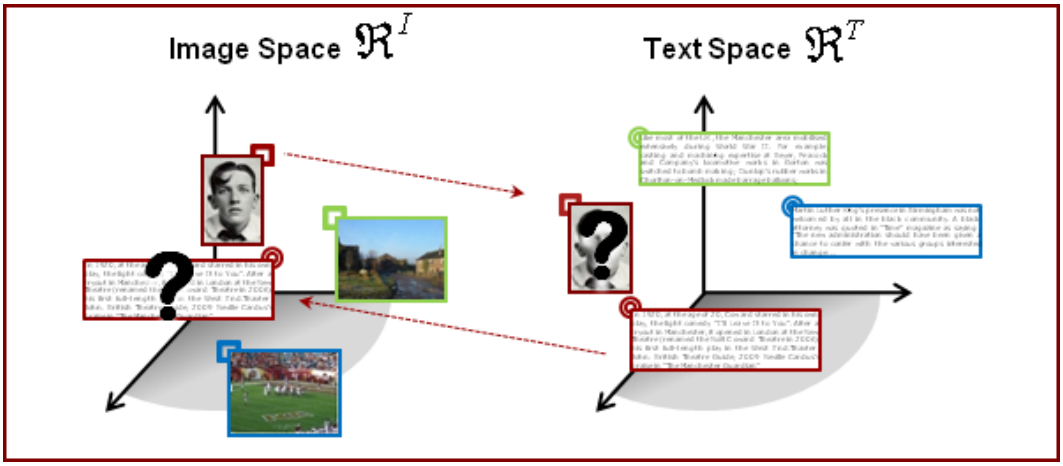

Figure 2: Cross-Modal Multimedia Retrieval.

In terms of implementation, we used Python's scikitlearn's library **PLSRegression** class.

3.2 Multilayer Perceptron

Multilayer Perceptron (MLP) is an extension of linear classifier(perceptron). It is a neural network. Each layer assigns each feature a weight w and a constant bias w0 then thresholds it with a non-linear activation function for binary classification. MLP utilizes backpropagation supervised learning technique. It classifies data that are not linearly separable.

This approach is attempted with the following setups:

- MLP(PCA(BoW description nouns, 1000), FC1000)
- MLP(PCA(BoW description, 2048), POOL5)
- MLP(PCA(BoW description, 2048), BoW tags)

We think PLS outperformed MLP because it better describes relations between two subspaces. Therefore, it was chosen to be included in the final model.

3.3 Reverse Approach

In the reverse procedure, we leveraged our test and training set to construct a new set of features for the test set. This is explained further here. For each image in the testing set, we did the following:

1. Pick up its POOL5 feature. Perform a 20-Nearest Neighbours with the training set, using their POOL5 features. This gives us the closest 20 training images to this test image.
   The intention here is to leverage the stronger mapping in data structures of a similar type to bridge the heterogeneity map. In this case specifically, the mapping between the POOL5 features in expected to be stronger, thus a simple k-Nearest Neighbours selects relevant images from the training set, for our test image.
2. For these 20 training images, we pick their bag of words of description, and create a new bag of words vector by averaging all of these 20 bag of word vectors.
3. This new averaged bag-of-words vector is assigned to the test image. The idea is that this averaged bag-of-words description will be an 'average' description found from the closest images in the training set.
4. Given a natural language query, our pre-processing converted it to a bag of words description. We then performed a k-Nearest Neighbour between the input BoW and the new BoW description that we created for the test images. Again, our intention was to leverage the stronger mapping within the bag of words data structures.

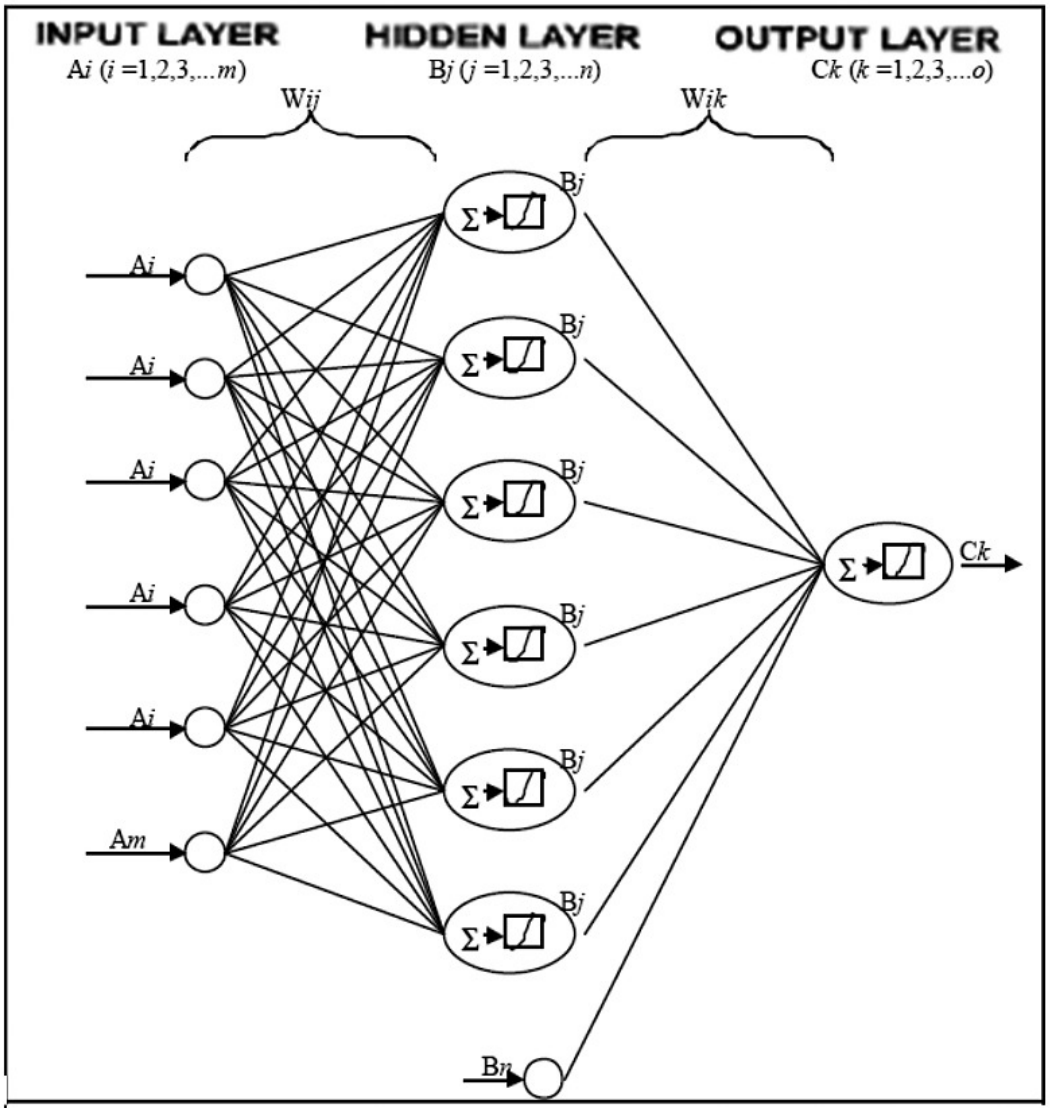

Figure 3: MLP Model illustration.

We performed this process with the following permutation of parameters and evaluated as follows, changing one parameter ceteris paribus:

- POOL5 > FC1000
- BoW nouns < BoW all
- We have also attempt this with the following number of neighbours from the training set in the k-NN: Top 10, Top 30 < Top 20

Therefore, Reverse (POOL5, BoW all, Top 20) does the best job.

4 Ensemble Modeling

4.1 Method 1: Plain average

The first try of the ensemble model is by weighting each model equally. Since it is not clear why one model is performing better than the other. Plain average of every model is a good start. It is later proven to be outperformed by weighted average.

4.2 Method 2: Weighted average

When having a few working predicting models, an ensemble method boosts performance by aggregating the results from each model. Each model has a knn ranked 80 most relevant pictures, so no relevant pictures would be loft out from the aggregation. The number 80 is a chosen after validation. Accuracy score would drop if it's above or below 80. The priorities of the picture ranking are: If a picture appears in multiple models, it is more relevant. If the picture is ranked higher in a model, it is more relevant since models use knn to rank pictures. If a picture is in a model with a higher accuracy, it is more relevant. Here is the proposed ensemble model that incorporates the above three priorities: Final ranking = Sum over all models: rank*weight

Notation: Final ranking is the ranking of the pictures; Rank is the rank of the picture in a single model; Weight is the accuracy score of the model.

Also, a few combinations of weights were employed to validate that the accuracy based weighted ensemble model indeed performs the best.

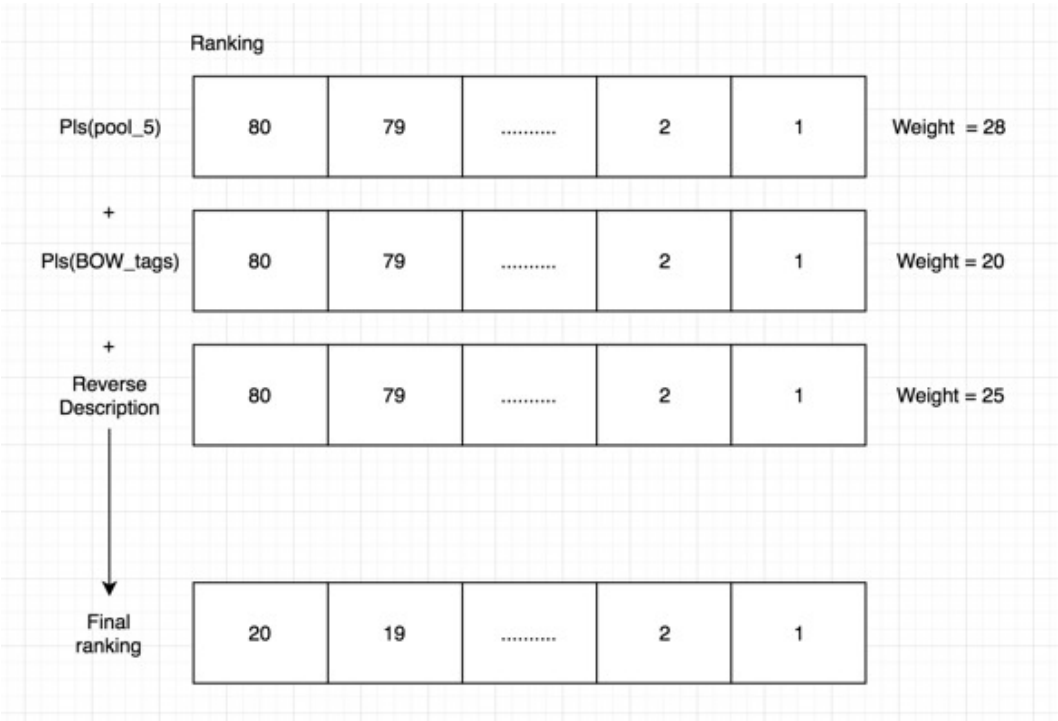

Figure 4: Weighing approach example.

4.3 Method 3: Page rank

Despite that the result is ranked, other ways to optimize ranking was employed. Pictures should be ranked by their relevance to the description. Page rank is employed to rank pictures by relevance. A network of all pictures as nodes was created. Pictures(nodes) are linked to each other if they share a common tag. Then for the final 20 result, the rankings are ranked by page rank with epsilon = 0.07 on the subnetwork of the 20 nodes.

However, the result was not satisfactory. The accuracy dropped from 0.33 to 0.14. There could be 2 reasons to explain the drop. First, the rankings were optimized by K Nearest Neighbors and the weighted average ensemble. Secondly, since the tags only describe pictures in high level, the network wasn't a good representation of the picture description relations.

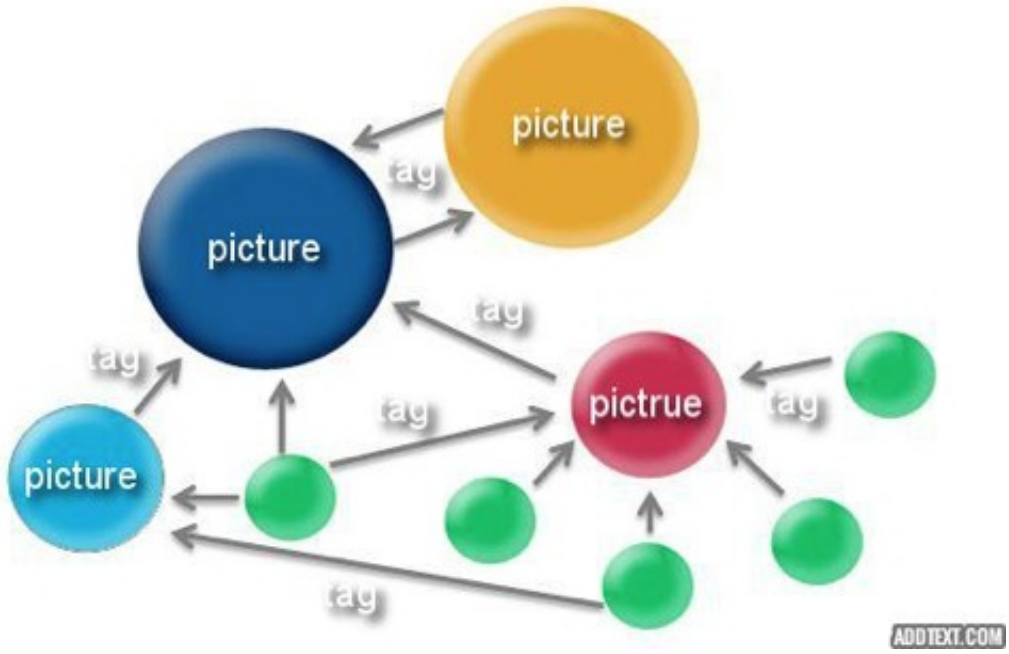

Figure 5: Page Rank illustration.

4.4 Final Version:

The final model is a combination of three models. 80 nearest neighbor pictures with cosine distance was selected for each model to higher the chance of including the right picture. For each picture, the scores are computed as 28*PLS(Pool 5)rank + 20*PLS(BOW tags)rank + 25*Reverse rank description.

The higher the total score, the higher the final ranking. The 20 pictures come from the top 20 pictures in the aggregated score.

The accuracy for this final model is **0.3674**.

5 Experiment and Evaluation

5.1 Evaluation

The evaluation metric is Mean Average Precision at 20 (MAP@20) on the test set. Here is how it works.

$$score = \frac{20 + 1 - i}{20}$$

For every test case, there is only one true image associated with the description. Given that the model/models output 20 choices, the score is a function the rank of the expected image in the top 20 choices.

5.2 Optimization

For every modeling approach and set of parameters, we ran the training process, produced top choices for each test case, submitted to the Kaggle competition and evaluated our accuracy. Before each

submission and as a sanity check, we visually checked a few test cases outputting all the selected choices as images. Here are our top performing models and ensembles:

5.3 Selected Model

Model	MAP@20 score
PLS(BOW description nouns, FC1000)	0.18744
MLP(BOW description nouns, FC1000)	0.12617
PLS(BOW description, POOL5)	0.28027
MLP(BOW description, POOL5)	0.22942
PLS(BOW description, BOW tags)	0.20374
Reverse Approach	0.25222
28*PLS(BOW desc, POOL5) + 20*PLS(BOW desc, BOW tags) + 25*ReverseApproach	0.3674

6 Conclusion

We see that for this problem, our input would essentially be a bag of words which has to be mapped to abstract image features captured by the CNN.

On PLS: Since these are 2 disparate spaces, PLS does a marvelous job because, at the very essence, it is formulated to handle 2 matrices of different width i.e. heterogeneity gap, by projecting them to new spaces that are of equal dimensions.

On using PCA in conjunction with PLS: Since PCA also accomplishes the goal of unsupervised compression while preserving variance of the data-set, we have attempted to apply it to the BoW matrix before using the PLS. We find that this considerably improves the running time, since PLS involves far greater number of operations to accomplish the same task.

On using image tags vs. image descriptions within PLS: We find that while both are important features, image tags are of inherently different spaces, even in the BoW representation, since there are far fewer total number of categories and sub-categories than there are words within bag of words. This is why we built a separate model for handling BoW of tags as opposed to concatenating it with BoW of descriptions.

Assigning greater weight to sub-categories as opposed to categories was an effective approach to creating features, since they are more specific when it comes to identifying images.

We suggest exploring techniques to use tags for shortlisting or eliminating images in sequence with another regression model on BoW description as potential avenues to explore for future.

On the ensemble approach: We found that assigning weights to both the frequency and ranking of images given by our models is an effective strategy for both shortlisting and ranking images.

On ranking images: We have attempted to apply a weighted page-rank to improve the ranking of our images, and this was not an effective approach. This could be because page-rank finds the degree of influence of images, while this is not necessarily the best degree of centrality when it comes to mapping BoW description to image features.

7 Acknowledgements

We would like to thank Prof. Serge Belongie, for teaching us the fundamentals and the TA team including Xun, Trishala, Micheal who helped us with assignments and our doubts throughout the semester. Also, grateful to our classmates who shared their experiences with us in the duration of the course.

8 References

[1] K. Wang, Q. Yin, W. Wang, S. Wu, and L. Wang,"A comprehensive survey on cross-modal retrieval," arXiv preprint arXiv:1607.06215, 2016.

[2] Latent Dirichlet Allocation by David M. Blei, Andrew Y.Ng, Micheal I. Jordan

[3] M.W Gardnera, S.R Dorlinga, Artificial neural networks (the multilayer perceptron), a review of applications in the atmospheric sciences, volume 32, issues 14-15, August 1998, page 2627-2623.