IMAGE TAG PREDICTION USING ZSL

A PREPRINT

Akash Gupta
Department of Computer Science and Engineering
Indian Institute of Technology, Roorkee
sky.gup7@gmail.com

Harsh Kumar Bansal
Department of Computer Science and Engineering
Indian Institute of Technology, Roorkee
hkbansal1997@gmail.com

Karan
Department of Computer Science and Engineering
Indian Institute of Technology, Roorkee
karan889295@gmail.com

Dr.Partha Pratim Roy
Department of Computer Science and Engineering
Indian Institute of Technology, Roorkee
proy.fcs@iitr.ac.in

April 22, 2019

ABSTRACT

Our work tries to show that there is a need for Zero-Shot based learning methods in image recognition problems. For this, we compare a Zero shot based learning algorithm with the existing state-of-the art supervised learning methods on a problem which can be formally quoted as 'Image Hashtag Prediction'. Semantically comparable results show that we don't need to train models with data for every problem, rather, existing models can be used with zero shot based learning methods.

1 Introduction

The fact that there are huge number of image classes, which is one of the major obstacles in the recognition of objects, makes data gathering and annotation too costly and infeasible. There is no major algorithm to deal with this issue. While we already have plenty of efficient and effective deep ConvNet architectures, which require supervised training, we need new methodologies that simulate how humans deal with this issue. Humans are capable of detecting new subjects the first time they see it. One way to do this is by training our model with the details about the new subject in a different form (such as text) and then try to identify the new subject. This approach is mainly used for the detection of unrecognized classes and is called Zero Shot Learning (ZSL).

As a part of our project, we decided to implement ZSL based methods, with a specific application in mind - predicting common hash-tags for images. The parsed tags (English words) are unseen classes that our ZSL tags map to. To compare this model to fully supervised methods, we collected the data, developed some data cleaning techniques and then applied finetuning on a popular deep ConvNet architecture ResNet18.

We have compared ZSL methods to fully supervised methods on a practical and application specific image classification task. This is justified as it serves all the causes and is an exciting application to build. The challenge was unavailability of publicly available "good" dataset that has annotated images with hash-tag labels, which is why we used HARRISON DATASET [1] in the later part of the project.

2 DATA COLLECTION AND CLEANING

One way is to collect data directly for individual hash-tags but this would result some serious issues. Semantically insensitive hash-tags are common in social media but there are no word vectors corresponding to those hash-tags. So, we considered an available novel hash-tag prediction benchmark, titled HARRISON. The HARRISON dataset is a practical dataset which brings real uploaded images with their related hashtags in the online social network Instagram. We found that HARRISON Dataset have noisy images that would hinder any classifier's accuracy for example consider images that have many logos ,text ,thumbnails or watermarks on top of image .We used the following procedure to remove such noisy images("Outliers"):

2.1 Feature Extraction

We extracted 4096 dimensional component vectors as outputs of the FC7 layer from the pre-trained AlexNet CNN architecture rather than going ahead with traditional features of SIFT or SURF. These features are considered good for outlier detection tasks and is destined to capture the semantic features present in the picture.

2.2 Dimensionality Reduction

We are using pipeline dimensionality reduction module, where we use PCA to decrease the size of the vector feature from 4096 to 128 to perform outer detection in reasonable amount of time and to eliminate similar or co-related features. For PCA, we have used SK-Learn module.

2.3 K-Means

For each class we employed K-Means[2] as a technique of clustering to create 8 clusters of image. Images too far from the centroids were supposed to be noisy. We verified the resultant noisy images where the images were not clear at all. An arbitrary threshold with two standard deviations from the average($\mu + 2\sigma$) were selected, and nearly all pictures above the centroid threshold turned out to be noisy. This outlier detection process has been able to eliminate approximately 40 noisy images per class, on average. For K-means and other calculations, we used SK-learn module of python. After K-means we found out images which have z score > 2 (Z-score being a distance metric). The images that were found as outliers in the dataset were removed from the dataset reducing the dataset size.

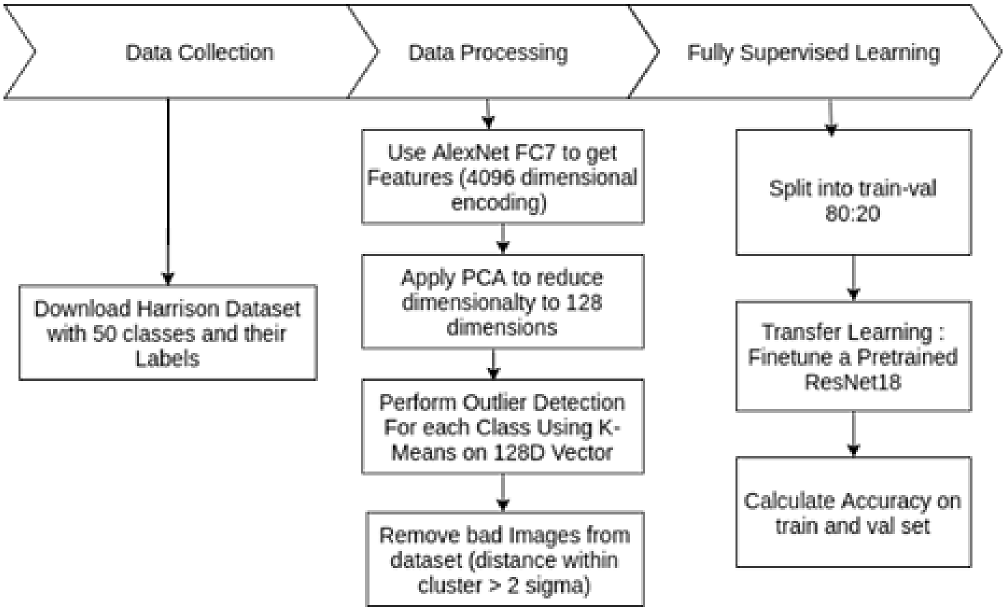

Data Collection
Data Processing
Fully Supervised Learning
Download Harrison Dataset with 50 classes and their Labels
Use AlexNet FC7 to get Features (4096 dimensional encoding)
Apply PCA to reduce dimensionalty to 128 dimensions
Perform Outlier Detection For each Class Using K-Means on 128D Vector
Remove bad Images from dataset (distance within cluster > 2 sigma)
Split into train-val 80:20
Transfer Learning : Finetune a Pretrained ResNet18
Calculate Accuracy on train and val set

Figure 1: Suprvised Hashtag Prediction Pipeline

3 SUPERVISED LEARNING

Hashtag prediction task can be understood as a multi-label classification problem. The underlying algorithm comprises of 2 important phases, the image features extractor and the multi-label classifier. In the image feature extractor model, image features are extracted and these features are given as the inputs for a pre-trained standard multi-label classifier. The score or probability for each class of label is produced by the model. From the sorted scores/ probability ,the hashtags with best scores are predicted for the input images. Details of each phase can be seen in the below figure.

Diverse and discriminatory picture information and high-level representation of features are required during the image extraction stage. We used the ResNet 18 architecture[3] as the extractor for the deep hierarchical features. The Residual

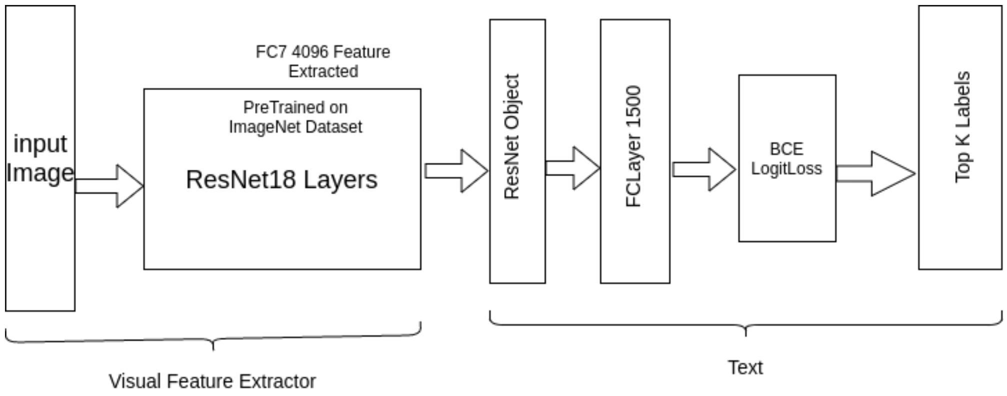


Figure 2: Overall Flow Diagram for Supervised method

network model is pre-trained on the ImageNet dataset which comprises of nearly 1.2 million images. Furthermore, we implemented a multi-label classifier by removing the output layer(Softmax layer) of ResNet and adding 2 Fully Connected layers with BCEWithLogitLoss[4] Loss function. The first added layer contains 1500 nodes and the second layer is outpur layer. Accuracy scores of the around 1000 hashtag classes is the output of multi-label classifier. We sort the scores and extract the top K classes in order to recommend the K number of hashtags.

We used Residual Network ConvNet architecture since it gives us state-of-art recognition results. Model parameters need to be adjusted to fit certain observations to specific scenarios or datasets through fine-tuning. Hence, we fine-tuned our pre-trained ResNet to prevent the network from overfitting the training data (since it is not large enough) and also to make sure that the network trains fast. This type of transfer learning is prevalent in Computer Vision and can be understood as learning how much weight is extracted from the initial CNN layers on highly informative features. As stated earlier, we used binary cross-entropy logistic loss which is usually used for multi-label classification problems because a threshold of 0.5 can be easily used to classify a label as positive and then there's no need of a threshold on the number of label to be shown in the output. The drop-out technique was also used as a regularizing method, where certain neuron connections have been allegedly dropped (0.7 in this case). This makes it hard for the network to overfit on the training data, especially if it's small (as ours).

4 ZSL Prediction

The recent survey paper titled, ”Zero-Shot Learning - The Good, the Bad and the Ugly”[5] was a great starting point for us as it provided a comprehensive overview of different techniques that have been tried in ZSL literature . The general approaches that recognize unseen classes in images consist of knowledge transfer between visual and semantic spaces. This is done by ensuring that there is compatibility (linear or non-linear) between the two spaces. Methods that learn non-linear compatibility between the two spaces outperform methods that learn linear compatibility. Hybrid models are the ones that express images and semantic class embeddings as a mixture of seen class proportions. Following the advice of this paper, we decided to implement CONSE models for our task, mostly because they are intuitive and simple to understand, and give decent results as well.

A simple approach and as first technique to the management of Zero-shot Learning is ConSE [6] or Convex Semantic Embedding Combination (ConSE). It utilizes an ImageNet[7] trained classifier to get semantic embedding vectors of images by a convex functional combination of class labels vectors embedding from training set. The insight is that weighted combination of most probable seen classes gives the semantic embedding of an unseen image.

$$f(x) = \frac{1}{Z}\sum_{t=1}^{T} p(\hat{y_0}(\mathbf{x}, t \mid \mathbf{x}).s(\hat{y_0}(\mathbf{x}, t)) \qquad \text{where } Z = \sum_{t=1}^{T} p(\hat{y_0}(\mathbf{x}, t \mid \mathbf{x})),$$

Here s(y) gives the semantic embedding of an image y and hyperparameter, $\hat{y_0}$ (x, t). Finally, the class closest to the obtained semantic embedding gives the prediction.

We used Gensim for 300 dimensional Word2Vec[8] features in PyTorch for implementation. We used the Word2Vec features for our experiments that were trained on Google News as it had the largest wording (3 M). We used the cosine similarity metric to perform the nearest search in the semantic word space. Moreover, since ImageNet classes consist mainly of several words per name of class, as recommended, we averaged the word vectors.

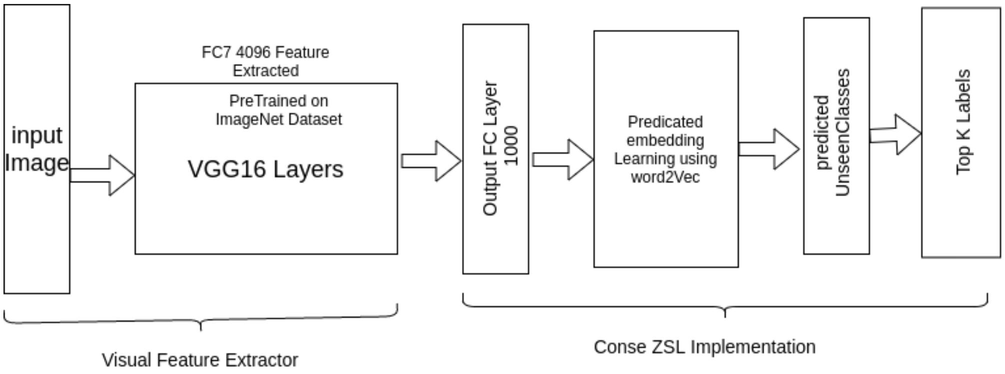


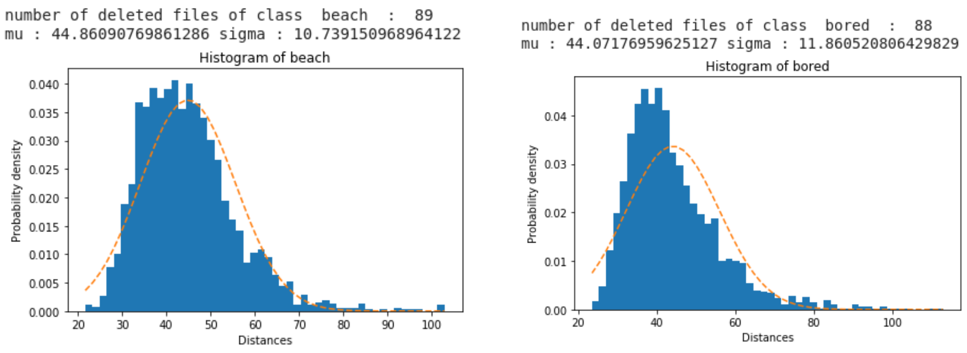
Figure 3: Overall Flow Diagram for ZSL method

As shown in the figure above, the prediction model is divided in two parts. First part is selection of top K most probable classes (among 1000 imagenet classes) for the image and second part is the prediction using those K most probable classes to find the prediction tags using a distance metric between the imagenet classes and the hash-tags. But, for using any distance metric, we need both the vectors to be in the same embedding space. Hence we use word2vec for both (imagenet classes and hashtags) to find their vector in the same semantic embedding space. Then we find the cosine similarity between the two. The hashtags corresponding to the closest vectors from the vectors of the K most probable imagenet classes, are then considered as the recommended hash-tags. We predicted tags for each images of 50 different classes of our dataset.

5 Result

5.1 Preprocessing Results

Below are the plots showing Z-scores(distance metric to measure seclusion from all the cluster in K-means). The images having z scores greater than 2 are removed from the dataset. Over 2000 images from 50 different classes were deleted from the dataset. Plots below show the relevant information about some of the classes.



```
number of deleted files of class  beach  :  89
mu : 44.86090769861286 sigma : 10.739150968964122
```

```
number of deleted files of class  bored  :  88
mu : 44.07176959625127 sigma : 11.860520806429829
```

Figure 4: Histogram of Classes after Preprocessing

5.2 Results From supervised Learning

Two performance measures were largely used in the prediction of hash-tags. One is precision, which demonstrates how many correct hashtags are being predicted, and the second is recall which depicts how many of the correct hashtags are being predicted. Most used measure is precision at 5 and recall at 10. The parameter details are described below. In this project, we choose three parameters to determine the hashtag prediction fairly, Precision and recall for primary use, and the correctness of the errors this system makes will be assessed by Accuracy.

$$Precision@K = \frac{|Result(K) \cap GT|}{|Result(K)|}$$

$$Recall@K = \frac{|Result(K) \cap GT|}{|GT|}$$

$$Accuracy@K = \begin{cases} 1 & \text{if} \quad Result(K) \cap GT \neq \emptyset \\ 0 & \text{if} \quad Result(K) \cap GT = \emptyset \end{cases}$$

where GT represents ground truth (Original) hashtags and Result(K) represents a group of the top K probable hashtags in predicted results. In our project, we set K to 5 for recall and k=1 for precision and Calculate accuracy taking into account the average number of HARRISON data set hashtags associated with each image.

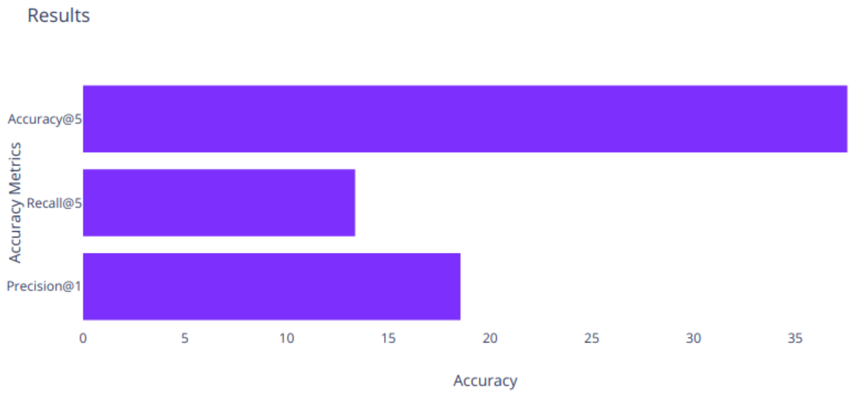


Figure 5: Result on Three Accuracy Metrics on our Dataset

By averaging all images in the test set, we assessed baseline results for Precision at 1, Recall at 5 and Accuracy at 5. The following graph shows the assessment outcomes for the HARRISON dataset baseline methods. With average precision@1 of 18.55%, the average recall@5 of 13.37% and average accuracy@5 of 37.56%, the ResNet18 model attains the highest performances. The outcomes of other models using a single visual feature have been slightly lower due to the visual information gap.

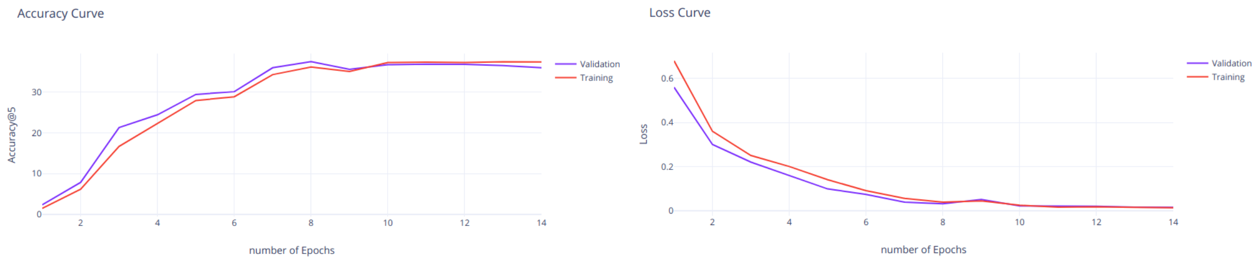


Figure 6: Accuracy and Loss Curve for ResNet18 CNN on our Dataset

Above figures show the training curves during fine-tuning. It can be seen that model, when started training on the harrison dataset, initially had too much loss and accuracy was very low. Over the course of 13 epochs, the model got fine-tuned to the extent that it could do multi label classification on around 1000 classes of labels with the top-5 accuracy of 37.56% and a whopping top-10 accuracy of 66.73%. Note that for each class of label, we only had an average of 50 images per label to train the model.

5.3 Result from ZSL

No training data was needed for the model used for this part of the project. That is why the model gives lower accuracy than supervised learning. However, the results found provide a critical drive that heavy research investment in zero shot learning is required and a large number of applications can be found in several areas, with the perspective that the model does not know about the images (on which we are testing) and their classes. Accuracy values can be seen below in the bar plot:

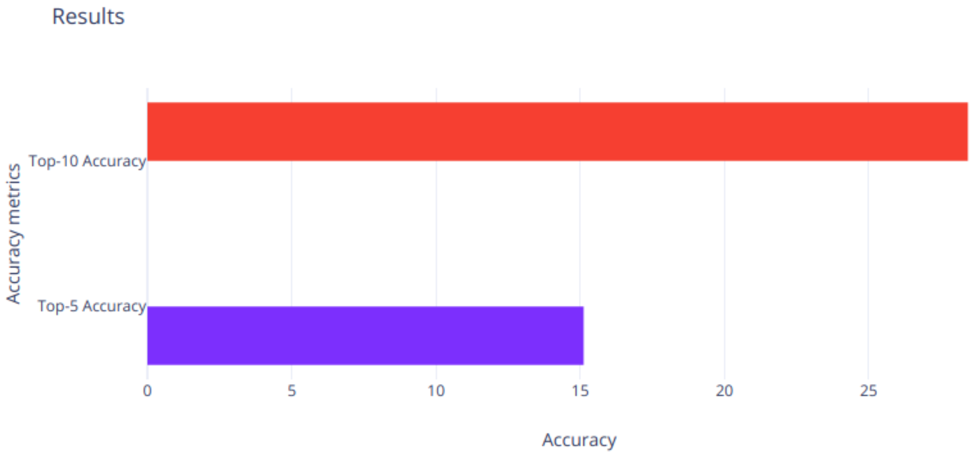


Figure 7: Result from ZSL method on our Dataset

As can be seen, the model produces a top-5 accuracy of 15.12% and a top-10 accuracy of 28.44% i.e. out of top5 or top10 predicted predictions, at least 1 matches the exact label that should be specified or the image was already annotated with. The big finding in our project was that the remaining tags that do not match the target label set of the image were not necessarily wrong. Most of the predicted label matched the target set of the image semantically if not exactly. The overall hash-tags recommended by the model were semantically correct and were relevant for the image.

5.4 Comparison of Result between ZSL and supervised Learning

We have mapped the images to the corresponding labels in 1 thousand labels specified in Harrison data set. We followed the following procedure to generate hashtags from labels, both for supervised and ZSL methods:

• ZSL methods are destined to have a low top-1 prediction accuracy, because the discrimination of semantically similar classes like "model" and "outfit" is especially difficult. To produce the hashtags, all the top 5 or top 10 predictions must be included. We randomly sample hashtags from the mapping to produce hashtags for our test images using cosine similarity in the word space as weights.

• Because the fully supervised method has top-1 accuracy of 18.37% and top-5 accuracy of 37.56%, all predictions after this can be safely ignored, particularly in order to avoid false alarms. As the goal of our project is to compare ZSL with data-driven methods by fairly calculating both the objective's correctness as numbers and subjective outcomes such as the quality of the hashtags achieved.

• Objective comparison: Accuracy metrics (Top-5 and Top-10) can be used for comparison between ZSL and supervised methods. Note that Top-k accuracy can be defined as number of times correct label is present in the top k predictions of the model. These metrics favour supervised methods, as certain tags (e.g.,' outfit' and' fashion') are semantically similar but are difficult for ZSL methods to discriminate between them. As the outcomes below show, ZSL falls well behind, causing a distinction of almost 22 percent in the accuracy of the top 5.

• Subjective comparison: Ground truth hash-tags aren't available for subjective comparison which would have made a lot more sense. Therefore, we thought of comparing the resultant hash-tags ourselves. Astonishingly, most of the hashtags recommended by ZSL were semantically suitable for the given images.

6 CONCLUSION AND FUTURE WORK

In this project, we used the HARRISON dataset, a dataset of real world images in social networks for hashtag prediction, We cleaned our dataset to remove noisy images then developed a basic structure of a ConvNet based feature extractor and a multilabel classifier to evaluate this dataset. We describes 3 assessment metrics and then analyzed our baseine models. We also described the difficulties faced with respect to our dataset as well as our problem statement of hashtag prediction system.

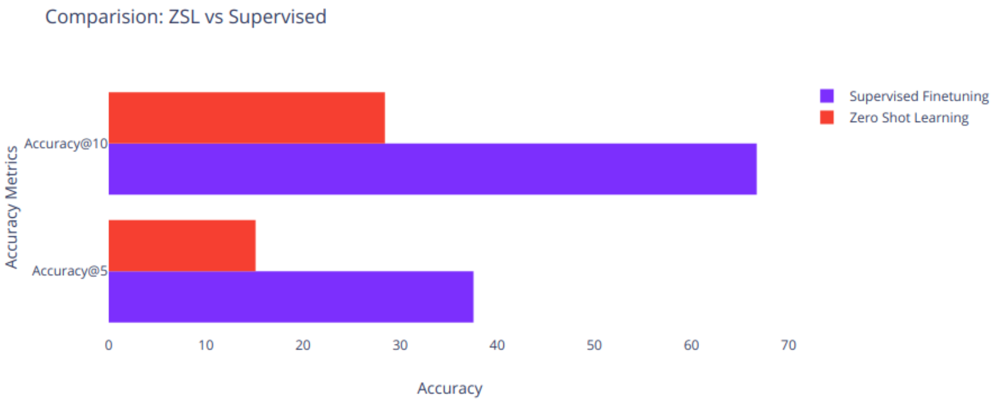


Figure 8: Result Comparison between ZSL vs Supervised Learning Method

As we said earlier, humans can detect new subject even when they see it for the first time. Our experiment was to build a model which is one step closer towards that aim. The astonishing results given by ZSL model without any training data give another path to explore in deep learning. This proves that we don't need new training data for every image classification problem, rather, we can use trained models with some zero-shot learning algorithms to tackle our problems. This way, lot of time can be saved and various applications can be produced extensively. Training deep ConvNets take a lot of time and data and hence makes the related tasks much more costly.

More future work is probably needed to improve the accuracy of the zero shot learning models before we can actually replace the traditional supervised learning algorithms. We conducted our experiment with a specific application (Image hash-tag prediction) in mind, but the dire need is to make a generalized model for zero shot learning to cover wide range of problems.

References

[1] Minseok Park, Hanxiang Li, and Junmo Kim. Harrison: A benchmark on hashtag recommendation for real-world images in social networks. *arXiv preprint arXiv:1605.05054*, 2016.

[2] Kiri Wagstaff, Claire Cardie, Seth Rogers, Stefan Schrödl, et al. Constrained k-means clustering with background knowledge. In *Icml*, volume 1, pages 577–584, 2001.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[4] François Fleuret. Ee-559–deep learning 11.3. word embeddings and translation.

[5] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[6] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[8] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.