CS229 COURSE PROJECT MILESTONE

Predicting Instagram tags using Zero Shot Learning

Shreyash Pandey - shreyash@stanford.edu
Abhijeet Phatak - aphatak@stanford.edu

Abstract

There are around 30,000 human-distinguishable basic object classes and many more fine grained ones. A major barrier to progress in computer based visual recognition is thus collecting training data for many classes. To counter this problem, a technique known as Zero Shot Learning (ZSL) has recently been introduced through which one is able to detect classes which were not part of the training set. In this project, we have analyzed variety of techniques within this area, describing the algorithms, strengths and weaknesses.

1 Introduction

One of the major bottlenecks in recognizing objects from images is that the number of different classes that the image could comprise of are huge in number. Data collection and annotation process for all those classes can be too inefficient and unfeasible. The other way to recognize those objects is to design algorithms that simulate how humans overcome this issue. A human being can detect the object in question even though it may be the first time they are seeing it. We are able to perform this inference by drawing information about that object from a different source (like text) and then using that to attempt to identify the object. This method is essentially what is used in practice to detect unseen classes and is referred to as Zero Shot Learning (ZSL).

A ZSL model typically utilizes information from text corpora, images and their labels and maps them to a common semantic space. Such a semantic space could either be a word space or an attribute space. Attribute space is defined using attributes(usually binary) such as *'hasFur'*, *'hasTail'*, *'isBrown'* etc. and is usually not preferred since manually tagging images with such attributes is not scalable and is inefficient. In case of a word space, where the labels are already mapped to that space, a mapping is learnt from the data to project images into that space. During test time, the input image is mapped in the semantic space and then a nearest neighbour search or some other similarity metric is used to select the closest unseen class.

As part of our project, we have decided to implement and compare two baseline methods, and the current state-of-the-art method that performs ZSL, with a specific application in mind - predicting common Instagram tags for images. The parsed tags are unseen classes that our ZSL tags map to.

We start this report by covering the basic aspects of all the common ZSL techniques, followed by a comparison between two baseline methods and finally concluding by enlisting the next steps in our project. We also include parallel work on Flask server that acts as a front-end for our hash-tag generation module.

2 Methods Studied

The recent survey paper titled, "Zero-Shot Learning - The Good, the Bad and the Ugly" was a great starting point for us as it provided a comprehensive overview of different techniques that have been tried in ZSL literature [1]. 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 two hybrid models as our baselines, mostly because they are intuitive and simple to understand, and give decent results as well.

3 Preliminary experiments - Baselines Implemented

3.1 ConSE

The first technique, known as ConSE or “Zero-Shot Learning by Convex Combination of Semantic Embeddings” (ConSE), employs a straightforward approach to dealing with ZSL tasks [2]. It uses a classifier to obtain semantic embedding of images by a convex combination of class label embedding vectors from the training set. The intuition is that the semantic embedding of an unseen image would be close to a weighted combination of the most likely seen classes:

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

T here is a hyperparameter, $\hat{y_0}(\mathbf{x},t)$ gives the t^{th} most probable label and $s(y)$ gives the semantic embedding of an image y. Finally, the prediction is obtained by finding the class nearest to the obtained semantic embedding.

We implemented this in PyTorch, and we used gensim for 500 dimensional Word2Vec features. For our initial experiments, we trained our Word2Vec features on free and available text8 corpus.

3.2 HierSE

The next paper we studied was an extension of ConSE : “Zero-shot Image Tagging by Hierarchical Semantic Embedding” (HierSE) [3]. It improves upon ConSE and obtains better semantic embedding by extracting hierarchical structure defined in the WordNet. This is to ensure that labels with low/no occurrence in the vocabulary, which are of particular interest in ZSL, get reliable embedding vectors. It also creates its semantic space from Flickr tags as opposed to Wikipedia in ConSE. This is based on the motivation that Flickr might be a better source since their tags better capture the label’s visual context. The embedding vectors now are obtained by :

$$\begin{gathered} f(x) = \frac{1}{Z}\sum_{t=1}^{T} p(\hat{y_0}(\mathbf{x},t \mid \mathbf{x})s_{hi}(\hat{y_0}(\mathbf{x},t)) \\ \text{where } s_{hi}(y) = \frac{1}{Z_{hi}}\sum_{y' \in y \cup super(y)} w(y' \mid y)s(y) \\ Z_{hi} = \sum_{y' \in y \cup super(y)} w(y' \mid y), \end{gathered} \tag{2}$$

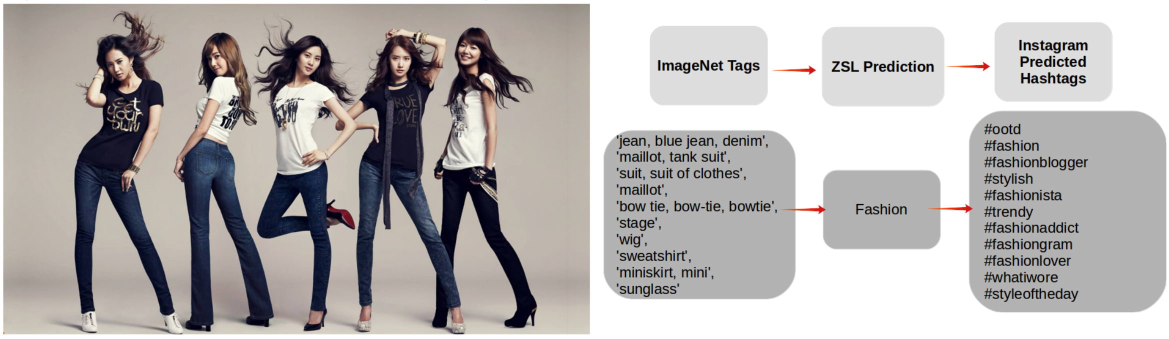
Here $super(y)$ refers to the ancestors of a label obtained using WordNet and $w(y' \mid y)$ is a weight subject to exponential delay with respect to the minimal path length from y to y'. Prediction is performed in a similar manner as described above. The code [4] for HierSE is available online in Caffe and TensorFlow. We used the TensorFlow code for testing purposes.

For comparison between ConSE and HierSE, we used a VGGNet CNN trained on ImageNet as our classifier, and the test label set consisted of labels within 2 tree hops of the 1000 training labels, namely their direct parent and child nodes, resulting in 1,548 novel labels in total. The ground-truth test images are from ImageNet, with the number of relevant images per label ranges from 1 to 2,330, with an average number of 846. The total number of test images was 1.3m, but we tested on a random subset of 10000 images from these 1.3m images due to resource constraints. The results are compiled below. Clearly, HierSE does much better than ConSE.

Method	hit@1	hit@10
ConSE	10.5	31.5
HierSE	17.8	50.9

Table 1: Comparison between ConSE and HierSE

Figure 1: Results of our PyTorch ConSE implementation for Hash-tag generation using ZSL. For initial experiments, we took a set of 20 unseen (instagram relevant) tags. We then map 20 tags to popular hashtags as shown below. Future work on generalized ZSL will help us improve the accuracy.



3.3 Flask Server for Front End

We have deployed our code as a web-service using Flask. Flask is a powerful web-development micro-framework in python that has support for debugging, templating, RESTful-API-like model and thus facilitates smooth deployment. The idea was to solve a real-world problem and to be able to reach many people. We expect that our improved model in the future can be actually used by many people on social-media. If given an opportunity, we would also like to deploy our project on an actual server provided by Stanford and also explore what kinds of problems can be solved by our project.

4 Next Steps

The ideal next step is to improve the accuracy of our Zero Shot Learning classifier. We could then look into the more general problem of Generalized Zero Shot Learning, wherein, the test classes include training labels as well.

4.1 Synthesized Classifiers for Zero-Shot Learning

To improve the accuracy of our classifier, we plan to implement the state-of-the-art method that performs ZSL. The paper titled, ”Synthesized Classifiers for Zero-Shot Learning” proposes to tackle the problem from the perspective of manifold learning [5]. The main idea is to align the semantic space that is derived from external information to the model space that concerns itself with recognizing visual features. To this end, they introduce a set of “phantom” object classes whose coordinates live in both the semantic space and the model space. Serving as bases in a dictionary, they can be optimized from labeled data such that the synthesized real object classifiers achieve optimal discriminative performance. We plan to demonstrate superior accuracy of this approach over the baselines that we have implemented until now. This will hopefully be our final ZSL model that predicts instagram tags for images.

4.2 Location Tagging

To improve the generated hash-tags, we plan to include an optional location field that can be entered along with the image input. We also plan to use the Exif information in the JPEG images to check for GPS coordinates, that will help us with the location specific hash-tags.

5 Contributions

Both the team members contributed equally to the project. Till the milestone, work was divided between the two of us. Shreyash did most of the bench-marking, chose the dataset and programming framework to be used. Abhijeet worked on setting up the Flask application and supported the base-lining. Now that the basic model is ready, both team members will contribute equally towards the next steps of this project - which are implementing the state-of-the-art method, and improving the hash-tag generation module. We plan to use unsupervised techniques like PCA (t-SNE) or k-means for generating more accurate and relevant tags.

References

[1] Yongqin Xian et al. "Zero-shot learning-A comprehensive evaluation of the good, the bad and the ugly". In: *arXiv preprint arXiv:1707.00600* (2017).

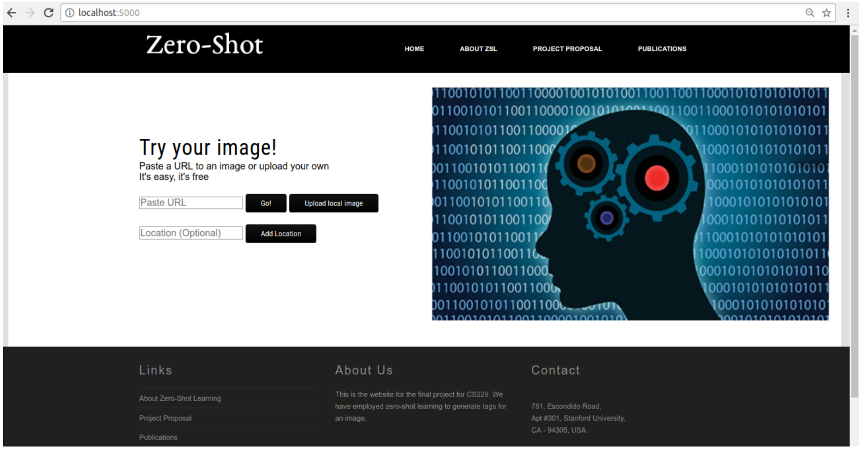
[2] Mohammad Norouzi et al. "Zero-shot learning by convex combination of semantic embeddings". In: *arXiv preprint arXiv:1312.5650* (2013).

[3] Xirong Li et al. "Zero-shot image tagging by hierarchical semantic embedding". In: *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM. 2015, pp. 879–882.

[4] *Hierarchial Semantic Embedding*. URL: `https://github.com/li-xirong/hierse`.

[5] Soravit Changpinyo et al. "Synthesized classifiers for zero-shot learning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 5327–5336.

Figure 2: A screen-shot of the Flask based web-service



Try your image!

Paste a URL to an image or upload your own
It's easy, it's free

This is the website for the final project for CS229. We have employed zero-shot learning to generate tags for an image.

781, Escondido Road,
Apt #301, Stanford University,
CA - 94305, USA.