

# Compositional Zero Shot Learning with pre-trained Vision Language Model

Ans Munir

Information Technology University  
Lahore

msds20033@itu.edu.pk

Dr. Mohsen Ali (Supervisor)

Information Technology University  
Lahore

mohsen.ali@itu.edu.pk

## Abstract

*Predicting the unseen composition “Yellow Parrot” from the seen compositions “Green Parrot” and “Yellow Sparrow” is the aim of Composition zero-shot learning. It is challenging because same state “Sliced” appears differently for different objects like “Sliced Bread” will appear differently from “Sliced tomato”. Current state-of-the-art approaches rely on pre-trained models to obtain representations that have been trained particularly for that domain. Word2vec and ResNet, for example, have been trained exclusively for text and images, respectively. Recent Vision-Language Models, on the other hand, have been trained on both images and text, and so both domains are aligned in them. Obtaining picture and text representations from such pre-trained VLM models will result in improved representations.*

## 1. Introduction

If we have green parrots at home and have never seen any other parrots. However, when we encounter yellow parrots, we immediately recognize them as parrots because the human brain has the ability to generalize concepts to related objects, which computers lack. In the preceding case, we need to retrain the model in order to recognize yellow parrots.

If our model is capable of generalizing the concepts, then it can address this particular problem. For example, if our trained model has seen a green parrot and a yellow sparrow, it may transfer the properties of the parrot to the sparrow and vice versa because these two objects are closely connected and their properties can be transferred. As a result, our model will be able to identify yellow parrots by applying the states and objects that it has encountered during training. The task of predicting the compositions of objects that a model has never seen before is referred to as compositional zero-shot learning.

We frequently generalize knowledge to new objects in real life, which makes this a fascinating topic. Analogously,

we want our model to be able to apply the knowledge it learned during training to objects in various states when it comes time for testing. But because the model is biased toward the examples it saw during training, it is challenging to achieve.

The pre-trained models are used by all state-of-the-art CZSL systems to obtain textual and visual representations. These pre-trained models have been tailored specifically for that domain. For instance, Word2vec was trained primarily for text data, whereas ResNet was created especially for visual data. Therefore, distinct visual and textual representations of the same concept, such as “car,” may exist. This impairs downstream tasks, such as composition recognition in the case of CZSL. In this paper, we argue that using the Foundation Vision-Language model can help us get equivalent text and image representations. The foundation model exhibits alignment between the visual and text domains due to its training on both of these domains.

The current study has the constraint that they have considered CZSL tasks in a closed world scenario. Nevertheless, the new setting for the CZSL task has been introduced by recent work [4] and [5]. Nonetheless, there is potential to improve the accuracy because it is now quite low for this new configuration. For further information on closed and open world settings, see section 3.

## 2. Related Work

### 2.1. Compositionality

We can define compositionality as the ability to decompose an observation into its sub-parts. Then these sub-parts can be used for complex reasoning. For example compositionality of word “Yellow Parrot” would be “Yellow” and “Parrot”.

### 2.2. Zero Shot Learning

Zero shot learning is defined as leaning the model that predicts the novel classes of objects that hasn’t been observed during the training. Usually novel classes can be described using text descriptions [11] or word embeddings [9].

### 2.3. Compositional Zero-Shot Learning (CZSL)

The characteristics of compositionality and zero-shot learning are combined in compositional zero-shot learning. The goal of compositional zero shot learning is for the model to learn the object and state compositions during training and then predict the object and state compositions that it has never seen before. Here, the object and the state have been combined to make the composition. In this case, our composition or label would be “Wet Cat” because the object “Cat” is in the condition of “wet”.

Previous works include modelling state as linear transformation of objects [7], learning a hierarchical decomposition and composition of states and objects [3], and learning a transformation upon individual states and object classifiers [2]. A few other works [10] [13] discover the joint compatibility function with respect to image, state, and object. In [13], visual transformation is learned via a causal graph, and objects and states are represented latently and independently of one another.

Graph Convolution Network is utilised to exploit the dependency between the states, objects, and their compositions in more recent works like CGE [6], where states, objects, and their compositions are formulated through graphs. Prior approaches take into account the compositional zero-shot learning task in a closed environment. But the CZSL task is taken into account in the open world via the CompCos model [4]. Similar to this, Co-CGE [5] integrates the techniques of [4] and [6]. Like in [6], it describes the dependence structure between states, objects, and their compositions, and like in [4], it takes the CZSL task into account in an open world environment.

### 2.4. Vision-Language Models

Vision-Language models have gained a lot of popularity because of their superiority on both vision and language tasks. Flava [12] is such a foundation vision-language model. FLAVA is a language vision alignment model that learns robust representations from both multimodal (image-text pairs) and unimodal (unpaired images and text) data.

## 3. Problem Definition

We shall define the problem of compositional zero shot learning (CZSL) in this section. We have provided the photos and their compositional labels in CZSL. Using those photos and their corresponding labels, we train our model. We can only have a limited number of states of the objects given to us, and we can utilise those states to train our model, but at test time, we will also have novel compositions that the model has not seen during training. However, those compositions contain states and objects encountered by the model during training, but with different combinations.

Dataset	s	o	Training		Validation			Test		
			sp	i	sp	up	i	sp	up	i
MIT-States	115	245	1262	30k	300	300	10k	400	400	13k
UT-Zappos	16	12	83	23k	15	15	3k	18	18	3k
C-GQA	453	870	6963	26k	1173	1368	7k	1022	1047	5k

Table 1. Three Datasets used in CZSL along with the statistics. (s: # states, o: # objects, sp: # seen compositions, up: # unseen compositions, i: # images)

There are two settings in compositional zero-shot learning task:

#### 3.1. Closed world setting

In a close world setting, we assume that we know test compositions a priori. So at test time, we take the test compositions given in the test set and predict the composition of given image. [6] uses a Generalized setting where test set consists of both seen as well as unseen compositions.

#### 3.2. Open world setting

Now let’s consider the open world setting. In open world setting we consider all the combinations of training states and objects. As a result if we have 5 states and 4 objects, our test space will consists of 20 compositions.

However, there is a problem in open world setting. As there can be any combination of training compositions at test time therefore there would be many test compositions that are unfeasible. Consider the example where we have two compositions during training i.e. “Hairy Cat” and “Red Tomato”. Now our test time composition can be “Hairy Cat”, “Hairy Tomato”, “Red Tomato” and “Red Cat”. Now we can observe that “Hairy Tomato” is a composition that we probably haven’t seen in our lives. [4] called these unusual terms as distractors. [4] and [5] has used cosine similarities to either eliminate or separate the distractors from the usual compositions.

## 4. Dataset

There are three datasets used for CZSL task. One of them i.e. CGQA [6] have been recently proposed and it has more objects and states. Other two are MIT-States [8] and UT-Zappos [14]. Table 1 shows the statistics of the three datasets used in CZSL. s and o means states and objects respectively. Similarly sp represents number of seen compositions and up represents number of unseen compositions and i stands for number of images. The seen compositions, unseen compositions and images present in the Test section of Table 1 forms the test set in closed world setting. However open world setting has all the combinations of training states and objects at the tes time.

Table 2 shows the samples from the C-GQA dataset along with its compositional labels below the images.

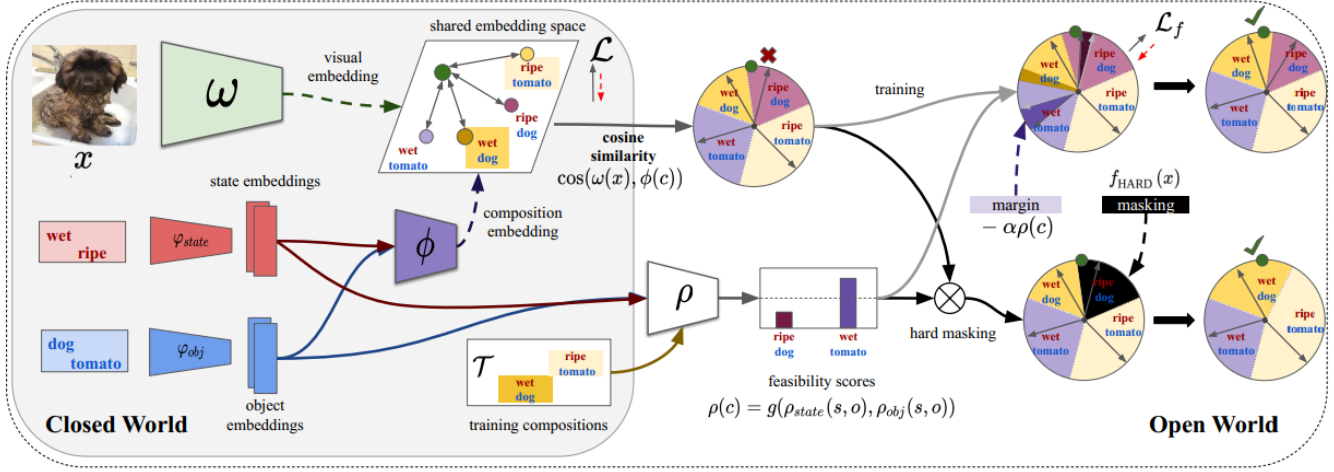


Figure 1. Model Diagram of Compositional Cosine Logits (CompCos). Image Source: [4]




		
White Swan	Pointy Fence	Marble Bathtub

Table 2. Table contains the images from the C-GQA dataset and their compositional labels below the images.

## 5. Methodology

### 5.1. Close world model

Our model is the same as the one used by CompCos [4], as seen in 1. We first extract characteristics from the images, in a manner similar to [4]. However, this is where our model and [4] diverge. We extracted features of the image from Flava [12]. CompCos, on the other hand, employed a ResNet-18 [1] model pre-trained on Imagenet to extract image features.  $\omega$  converts the extracted image features to 300 dimensional vector.  $\omega$  is a two layer MLP that has dropout, LayerNorm and a Relu non-linearity after first layer. During training this 2-layer MLP has been trained.

Training labels are provided in compositions such as *Black Cat*. These labels must be converted to embedding space. We first break these labels down into their component parts, attribute and object, before converting them to embedding space. Composition *Black Cat*, for instance, will break down into attribute *Black* and object *Cat*. Next, we'll transform the object and attribute into the embedding space. This is another area where our model and Comcos vary. Flava [12] has been utilised by us to obtain the ob-

ject and state embeddings. CompCos, on the other hand, employed pre-trained word2vec embeddings.

Then we use a simple linear projection for a compositional label:

$$\phi(c) = [\varphi(s)\varphi(o)]^T W$$

This will take two vectors i.e. state and object and convert them to one vector of 300 dimensional. It is of same size as image features.

### Cosine Similarity

Now we compute cosine similarity between image features and compositional label as follows:

$$\cos(y, z) = \frac{y^T z}{\|y\| \|z\|}$$

We want the image and compositional label that has maximum cosine similarity:

$$f(x) = \arg \max_{c \in C^+} \cos(\omega(x), \phi(c))$$

where  $\omega : X \rightarrow Z$  is the mapping from the image space to the shared embedding space  $Z \in R^{300}$  and  $\phi : C \rightarrow Z$  embeds a composition to the same space and cos is the cosine similarity image and composition embedding.

### Objective Function

If we follow the closed world setting then following is the loss function that we use:

$$L = -\frac{1}{|T|} \sum_{(x,c) \in T} \log \frac{e^{\frac{1}{T} \cdot p(x,c)}}{\sum_{y \in C^s} e^{\frac{1}{T} \cdot p(x,y)}}$$

where  $T$  is the temperature value that balances the probabilities for cross-entropy loss and  $p(x, c) = \cos(\omega(x), \phi(c))$

Methods	MIT-States				UT-Zappos				C-GQA			
	AUC	HM	Seen	Unseen	AUC	HM	Seen	Unseen	AUC	HM	Seen	Unseen
Compcos [4]	4.5	16.4	25.3	24.6	28.1	43.1	59.8	62.5	2.6	12.4	28.1	11.2
CGE [6]	6.5	21.4	32.8	28.3	33.5	60.5	64.5	71.5	4.2	16.0	33.5	15.5
Co-CGE [5]	6.6	20.0	32.1	28.3	33.9	48.1	<b>62.3</b>	66.3	4.1	15.5	33.3	14.9
<b>Ours</b>	<b>9.4</b>	<b>24.0</b>	<b>36.9</b>	<b>33.2</b>	<b>34.7</b>	<b>48.1</b>	62.0	<b>69.0</b>	<b>8.7</b>	<b>24.3</b>	<b>40.1</b>	<b>25.3</b>

Table 3. Results in close-world setting on all the three datasets. Last row represents our model results.

## 5.2. Open world model

In open setting we have all the combinations of compositional labels. Therefore we need to select the correct label from all the labels. Some of the labels can be distractors and we need to eliminate those extractors.

### Estimating Compositional Feasibility

If there are some attributes of tomato like *Red Tomato* or *Yellow Tomato* then same attributes can also be possible for Apple like there can be *Red Apple* and *Yellow Apple*. So we can transfer the properties that are related to each other. Given a composition  $c = (s, o)$ , we define its feasibility score with respect to the object  $o$  as:

$$\rho_{obj}(s, o) = \max_{\hat{o} \in O^s} \cos(\varphi(o), \varphi(\hat{o}))$$

with  $O^s$  being the set of objects associated with state  $s$  in the training set. Similarly, we define the score with respect to the state  $s$  as:

$$\rho_{state}(s, o) = \max_{\hat{s} \in S^o} \cos(\varphi(s), \varphi(\hat{s}))$$

with  $S^o$  being the set of states associated with the object  $o$  in the training set  $C^s$ . The feasibility score for a composition  $c = (s, o)$  is then:

$$\rho(c) = \rho(s, o) = g(\rho_{state}(s, o), \rho_{obj}(s, o))$$

where  $g$  is a mixing function, e.g. max operation ( $g(x, y) = \max(x, y)$ ) or the average ( $g(x, y) = (x + y)/2$ ).

### Loss function for open world setting

We will inject the feasibility scores  $\rho(c)$  directly within the objective function:

$$L_f = -\frac{1}{|T|} \sum_{(x,c) \in T} \log \frac{e^{\frac{1}{T} \cdot p^f(x,c)}}{\sum_{y \in C^s} e^{\frac{1}{T} \cdot p^f(x,y)}}$$

with:

$$p^f(x, c) = \begin{cases} \cos(\omega(x), \phi(c)) & \text{if } c \in C^s \\ \cos(\omega(x), \phi(c)) - \alpha \rho(c) & \text{otherwise} \end{cases}$$

where  $\rho(c)$  are used as margins for the cosine similarities, and  $\alpha > 0$  is a scalar factor.

Methods	AUC	HM	Seen	Unseen
Compcos [4]	21.3	36.9	59.3	46.8
CGE [6]	23.1	39.0	<b>61.7</b>	47.7
Co-CGE [5]	23.3	40.8	61.2	45.8
<b>Ours</b>	<b>24.3</b>	<b>41.5</b>	57.7	<b>55.3</b>

Table 4. Results in open-world setting on UT-Zappos datasets. Last row represents our model results.

## 6. Results

Table 3 shows the results on all the datasets in the close-world setting. Table 4 shows the results of all the models on UT-Zappos dataset in the open-world setting. The abbreviations AUC, HM, seen, and unseen denote area under the curve, seen accuracy, and unseen accuracy, respectively. Our model performs better than all current state-of-the-art models, as shown in Table 3. This demonstrates that obtaining representations from the foundation model, i.e. Flava, is advantageous because it aligns both the visual and text domains. We will specifically compare our results to those of the Compcos because we employ similar models. Table reftab:close-results shows that we have obtained more than double the AUC of Compcos in the MIT-States dataset. The largest and most complex dataset, CGQA, has a higher number of states and objects. Our AUC on CGQA was 8.7, whereas Compcos' was 2.6.

## 7. Conclusion and Future Work

By using Compositional Zero-Shot Learning, a model may identify the states of items that are not seen during training and generalise the states to related objects. Current state-of-the-art works use the pre-trained models to obtain picture and label representations that have been trained separately. In this paper, we demonstrated how employing the Foundation Vision-Language model, i.e. Flava, can help us generate better picture and label representations that are more aligned with each other.

## References

- [1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceed-*

*ings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

- [2] A. Gupta I. Misra and M. Hebert. From red wine to red tomato: Composition with context. In *CVPR*, 2017. 2
- [3] J. Yan X. Liu M. Yang, C. Deng and D. Tao. Learning unseen concepts via hierarchical decomposition and composition. In *CVPR*, 2020. 2
- [4] Yongqin Xian Zeynep Akata Massimiliano Mancini, Muhammad Ferjad Naeem. Open world compositional zero-shot learning. In *CVPR*, 2020. 1, 2, 3, 4
- [5] Yongqin Xian Zeynep Akata Massimiliano Mancini, Muhammad Ferjad Naeem. Open world compositional zero-shot learning. 2021. 1, 2, 4
- [6] Federico Tombari Zeynep Akata Muhammad Ferjad Naeem, Yongqin Xian. Learning graph embeddings for compositional zero-shot learning. In *CVPR*, 2020. 2, 4
- [7] T. Nagarajan and K. Grauman. Attributes as operators: factorizing unseen attribute-object compositions. In *ECCV*, 2018. 2
- [8] Joseph J Lim Phillip Isola and Edward H Adelson. Discovering states and transformations in image collections. In *CVPR*, 2015. 2
- [9] C. D. Manning R. Socher, M. Ganjoo and A. Ng. Zero-shot learning through cross-modal transfer. In *NeurIPS*, 2013. 1
- [10] A. Gupta S. Purushwalkam, M. Nickel and M. Ranzato. Taskdriven modular networks for zero-shot compositional learning. In *ICCV*, 2019. 2
- [11] H. Lee S. Reed, Z. Akata and B. Schiele. Learning deep representations of fine-grained visual descriptions. In *CVPR*, 2016. 1
- [12] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15638–15650, 2022. 2, 3
- [13] U. Shalit Y. Atzmon, F. Kreuk and G. Chechik. A causal view of compositional zero-shot recognition. 2020. 2
- [14] Aron Yu and Kristen Grauman. Fine-grained visual comparisons with local learning. In *CVPR*, 2014. 2