Learning Attention Propagation for Compositional Zero-Shot Learning

Muhammad Gul Zain Ali Khan[1,3,4] Muhammad Ferjad Naeem[2] Luc Van Gool[2]
A. Pagani[1,3] Didier Stricker[1,3,4] Muhammad Zeshan Afzal[1,3,4]
[1]DFKI, [2]ETH Zürich, [3]TU Kaiserslautern, [4]MindGarage

Abstract

Compositional zero-shot learning aims to recognize unseen compositions of seen visual primitives of object classes and their states. While all primitives (states and objects) are observable during training in some combination, their complex interaction makes this task especially hard. For example, wet changes the visual appearance of a dog very differently from a bicycle. Furthermore, we argue that relationships between compositions go beyond shared states or objects. A cluttered office can contain a busy table; even though these compositions don't share a state or object, the presence of a busy table can guide the presence of a cluttered office. We propose a novel method called Compositional Attention Propagated Embedding (CAPE) as a solution. The key intuition to our method is that a rich dependency structure exists between compositions arising from complex interactions of primitives in addition to other dependencies between compositions. CAPE learns to identify this structure and propagates knowledge between them to learn class embedding for all seen and unseen compositions. In the challenging generalized compositional zero-shot setting, we show that our method outperforms previous baselines to set a new state-of-the-art on three publicly available benchmarks.

1. Introduction

Dog species differ considerably from each other. However, when presented with an unseen dog specie, we humans can recognize its states without hesitation. A child that has seen a wet car can recognize a wet dog regardless of the vast difference in appearance. Humans excel at recognizing previously unseen compositions of states and objects. This remarkable ability arises from our ability to reason about various aspects of objects and then generalize them over previously unseen objects. In zero-shot learning, the goal is to predict unseen classes, having seen a set of seen classes and a description of all the classes. A vector of attributes for all classes is provided in the most common configuration of zero-shot learning. The task is to learn the mapping between class description vectors and images such that it can be generalized over unseen classes [34, 33, 25]. Although deep neural networks have been modelled after the human mind [34, 33, 25], they struggle to perform well in zero-shot learning. In this paper, we study a special setting of zero-shot learning called Compositional Zero-Shot Learning.

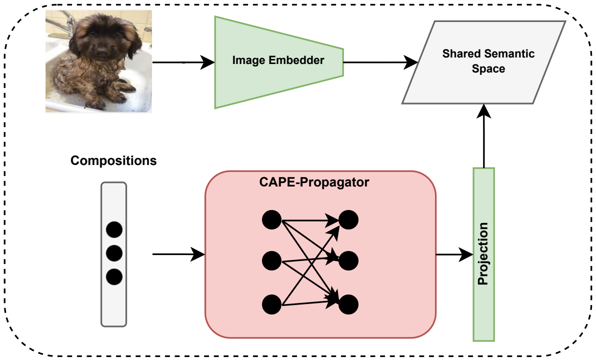

Figure 1. Shows overview of our approach. CAPE-Propagator exploits self-attention mechanism to learn interdependency structure between compositions by identifying critical propagation routes. We project output of self attention into a shared semantic space along with image embedding. Compositions that are similar to each other are placed near and far away from other compositions in shared semantic space.

"Compositional" in Compositional Zero-Shot Learning derives from the composition of primitives of objects and their states. At training time, all primitives (state, object) are provided in some combination but not all compositions. The goal is to predict novel compositions of primitives during test time. This poses several challenges because of the complex interaction between objects and their possible states. For example, a wet car is very different from a wet dog in visual appearance. Furthermore, a composition can be abstract as well, i.e., Old city, Ancient Tower. In real-world settings, multiple valid compositions can be found in one image, i.e. A clean desk in an image of a wet dog. A successful methodology should be able to learn all aspects of the complex interaction of objects and their states.

One simple solution to the above mentioned challenges is to disentangle states from objects. If states can be completely disentangled from objects, new combinations of states and objects can be predicted easily. This approache have been studied in some recent works [2, 30] on synthetic or simple datasets like UT-Zappos [40, 41]. In recent work, contrastive loss function-based methodology is introduced [14] for real-world datasets. The approach in [14] uses a generative network to generate novel samples in order to bridge the domain gap between seen and unseen samples. In the real world, states can be imagined as transformation functions of objects. A dry car changes the visual appearance after the application of state wet. Approaches in [24, 15] have studied the function of states as a transformation function. Other methodologies have tried to exploit all available information to learn useful correlations [28, 22]. Recent works have explored learning dependency structures between word embeddings and propagating them to unseen classes [23, 18].

In the real world, compositions do not occur in isolation. They are intricately entangled with each other and full of noise. No approach has considered the holistic view of complex interactions between compositions or their primitives. While the approach in [23] exploits the shared information between compositions that share primitives, it still ignores the interaction of compositions that do not share primitives, such as a coiled plate can be found in a cluttered kitchen or a narrow road can be seen in an ancient city. We study a more holistic view of interactions between compositions. We argue that there is a hidden interdependency structure between compositions. An overview of our approach is shown in Figure. 1. We exploit self-attention mechanism to explore hidden interdependency structure between primitives of compositions and propagate knowledge between them. Our contributions are as follows:

- We propose a multi-modal approach that learns to embed related compositions closer and unrelated far away.
- We propose a methodology that learns the hidden interdependency structure between compositions by learning critical propagation routes and propagates knowledge between compositions through these routes.
- Unlike [23], our approach does not require prior knowledge of how the compositions are related.

2. Related Work

Recent works have exploited the fundamental nature of states and objects to build novel algorithms. This includes reasoning over the effect of states over objects [15, 24]. The approach introduced in [24] considers states as a linear function that transforms objects into compositions. These linear functions can add a state to a composition or remove a state from the composition by inverting the linear function. Approach in [15] also considers the symmetry of states. Both approaches [24, 15] exploit group theory principles like closure, associativity, commutativity and invertibility. Both approaches [24, 15] use triplet loss [9] as the objective function. In contrast with [24], [15] uses a coupling and decoupling network to add or remove a state from a composition. Other approaches have tried to exploit the relationship between states and objects instead of assuming states as transformative functions [2, 30, 12, 18, 22, 28].

The approach in [22] argues that context is essential to model, i.e. red in red wine is different from red in red tomato. The approach in [22] argues that compositional classifiers lie in a smooth plane where they can be modelled and propose to model compositional classifiers of primitives using SVMs. These classifiers are pre-trained using SVMs and then fed into a transformation network that translates them into compositional space. The transformation network is three layered non-linear Multi-Layer Perceptron (MLP). Final predictions are retrieved by a simple dot product between the output of the transformation network and image embedding. One recent work uses word embeddings of primitives, and a simple MLP projects embeddings into a shared semantic space [18] (Compcos). Compcos [18] proposes to replace logits with cosine similarity between image embedding and projected word embeddings. Another recent work proposes to replace multi-layer perceptron with a Graph Convolutional Network [12] (GCN) to model the interaction between compositions (CGE). CGE [23, 19] argues that compositions are entangled by their shared primitives and proposes to use GCN that propagates knowledge through entangled compositions. CGE [23] utilizes a dot product based compatibility function between compositional nodes and image embeddings to calculate the scores of compositions. While CGE [23] uses average of state and object embeddings as compositional embeddings, Compcos [18] uses concatenation of state and object embeddings to represent a composition.

Another view for solving CZSL problem is to disentangle states from objects [2, 30, 14]. Approach in [2] proposes to exploit causality [32, 43, 39, 4, 7, 27, 26] to disentangle states and objects. The causal view in [2] assumes that compositions are not the effect of images but rather the cause of images, and do-intervention on primitives generates a distribution from which a given image can be sampled. Another recent work proposes to learn independent prototypes of states and objects by enforcing independence between the representation of states and objects [30]. This approach [30] further exploits a GCN to propagate prototypes of states, objects and their calculated composition. Like CGE [23], the approach in [30] calculates scores of compositions by a dot product between compositional nodes of GCN and image embeddings. This leads to further knowledge sharing between completely independent prototypes. Approaches in [2, 30] focuses on synthetic dataset like Ao-Clevr [2] or UT-Zappos [40, 41]. A recent approach has explored the disentanglement of states and objects on real-world datasets (SCEN) [14]. SCEN [14] uses contrastive loss and proposes to use compositions with shared primitives as positive samples and others as negative samples. SCEN [14] further proposes using a generative network to create novel compositions to bridge the gap between seen and unseen compositions. Our methodology is closer to CGE [23] that proposes to model the interdependency between compositions. However, CGE [23] only considers a dependency based on shared primitives. CGE [23] does not model more complex interdependencies that do not share primitives. Such as, the presence of a cluttered desk can guide the presence of cluttered office or the presence of a coiled plate can guide the presence of cluttered kitchen. The major limitation of CGE [23] is the usage of GCN to model interdependency structure because GCN relies on a fixed adjacency matrix to hardcode the interdependency structure. We propose to exploit self-attention mechanism like the one proposed in Transformer network [37] to learn this interdependency structure instead.

Transformer networks were first introduced for natural language processing to solve the problem of forgetting past hidden states for long sentences and vanishing gradient posed by RNNs [37]. Transformer networks [37] use a stack of encoder and decoder blocks that contain Multi-Head attention and multi-layer perceptrons. Multi-Head Attention (MHA) calculates attention on slices of features from query, key and value pairs. Number of slices are determined by the number of heads in MHAs. The usage of transformers [37] in the image classification task was explored in [6] that proposed to divide an image into 16 × 16 tokens. An encoder network extracts features from 16 × 16 tokens that are used for classification. This has lead to a number of methodologies that utilize transformer networks for image and video processing [29, 36, 42, 16, 13, 38, 17]. Large networks like proposed in [29] also have zero shot capabilities because they are trained on large datasets.

Our proposed approach builds on the findings of several prior works [22, 23, 18]. These works have explored the interdependency between compositions [23, 22, 18] in a simplistic manner. We propose a more holistic view of the interdependency structure between compositions. We argue that compositions are not simply related based on explicitly shared primitives. There is also implicitly hidden interdependency between compositions. We further propose a self-attention-based methodology to explore and exploit this interdependency structure between primitives of compositions and propagate the knowledge between them. Our approach learns this interdependency structure during training in an end-to-end manner and can find more various dependencies between compositions than simply shared primitives.

3. Approach

3.1. Problem Formulation

Firstly, we formally define Compositional Zero Shot Learning. Let $x \in \mathcal{X}$ denote an RGB image and $y \in \mathcal{Y}$ denote compositional label $y = (s, o)$ where $s \in \mathcal{S}$ is state and $o \in \mathcal{O}$ is object. We can then define training set τ as $\tau = \{(x, y) | x \in \mathcal{X}, y \in \mathcal{Y}_s\}$ where $\mathcal{Y}_s \in \mathcal{Y}$ represents seen compositions. The task of CZSL is to predict novel compositions $\mathcal{Y}_n$ during test time having seen set of seen labels $\mathcal{Y}_s$. Novel compositions $\mathcal{Y}_n$ does not include any compositions from $\mathcal{Y}_s$ i.e, $\mathcal{Y}_n \cap \mathcal{Y}_s = \emptyset$. We consider a specialized case of this problem called generalized compositional zero shot learning that includes seen compositions as well during test time $\mathcal{Y} = \mathcal{Y}_n \cup \mathcal{Y}_s$.

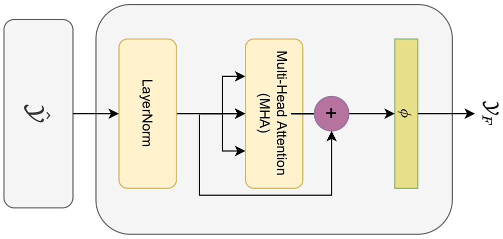

Figure 2. The architecture of CAPE-Propagator module that finds propagation routes and projects embeddings into share semantic space. Embeddings are first passed through a LayerNorm [3] followed by a Multi-Head Attention block. The output of Multi-Head Attention is residually added to the input and fed into a Multi-Layer Perceptron ϕ that projects it into a shared semantic space.

3.2. Compositional Attention Propagated Embeddings (CAPE)

CZSL is the image classification task, where each image is associated with a composition of state (s) and object (o). In the most straightforward setting, the compositional primitives, i.e. states and objects, which are all observed during training, should provide an avenue for generalization. However, the interaction of states and objects is complex. For example, a visual transformation of dry to wet carpet is very different from a dry to the wet car. This means that our model needs to learn how each state and object interact with each other. Moreover, as noted in [23], compositions of states and objects are inherently a multi-label problem, e.g.

while wet dog and dry car represent states with respect to liquid interaction, these objects simultaneously have other valid states representing colour, size etc. While not represented in the label set, these valid states, if discovered by the model, can present another avenue for knowledge transfer.

Furthermore, multiple compositions are a mixture of other compositions, e.g. An image of cluttered desk might also contain a blue mug. Learning a dependency structure between cluttered desk and blue mug will help propagate knowledge about blue mug to cluttered desk. Our novel model **Compositional Attention Propagated Embedding (CAPE)** aims to learn this knowledge propagation from training data as shown in Figure 3 leading to state-of-the-art performance. Unlike prior methods [23], we do not limit the exploration of dependency structures by constraining our methodology by assumed priors (i.e. compositions are only related by shared primitives). We rely on our approach to discover all possible dependency structures during training.

Learning the Propagation routes. We introduce a module (CAPE-Propagator) for discovering interdependency structure between compositions. An overview of the CAPE-Propagator is given in Figure. 2. Given a list of compositional pairs, CAPE-Propagator is tasked with finding a propagation route to transfer knowledge between them before outputting the final embedding. We frame this as a self-attention problem. Given the set $\mathcal{Y}_s$, we find the compatibility between each composition as a query search problem. Each composition $\mathcal{Y}_{s_i}$ is defined in the feature space by a concatenation of the word embedding of its represented state s and object o as $\hat{\mathcal{Y}_{s_i}} \in \mathbb{R}^{|Y| \times D}$ where D is the feature dimension. Let $\mathcal{T}_Q$, $\mathcal{T}_K$ and $\mathcal{T}_V$ be linear transformations that maps from D to D as a transformation of input pre-trained word embedding. These transformations map the input to a new linear space suitable for propagation. We pass $\hat{\mathcal{Y}}_{\int}$ through a LayerNorm layer followed by each of these transformation layers to get the Query $\mathcal{Q}$, Key $\mathcal{K}$ and Value $\mathcal{V}$.

For a given Compositional pair y_i, we define a propagation route as a compatibility between its query Q_i and all Keys K as:

$$\mathcal{P}_i = Softmax(\mathcal{Q}_i \cdot \mathcal{K}_j \text{ for } j \in |K|) \qquad (1)$$

P_i is the propagation coefficient and defines the contribution of each composition to the output embedding of a given composition. In essence, we expect the model to learn that a wet dog is related to other wet animals and other properties of the contained state and object. At test time, the propagation coefficients for all Compositions $\mathcal{Y}$ are computed to form $\mathcal{P} \in \mathbb{R}^{|\mathcal{Y}| \times |\mathcal{Y}|}$.

Propagating the Compositional Knowledge. CAPE utilizes the Propagation coefficients $\mathcal{P}$ to propagate knowledge between compositions at test time. The propagated knowledge results in computing a new representation of each pair $\mathcal{Y}_P \in \mathbb{R}^{|\mathcal{Y}| \times D}$ as:

$$\mathcal{Y}_P = \mathcal{P} \times \mathcal{V} \qquad (2)$$

This operation aggregates knowledge across all compositions defined in our dataset and makes the model aware of all the properties of each state and object. Since we learn the propagation coefficients from data, these propagation routes are improved during training on $\mathcal{Y}_s$ instead of having to hardcode them from our label set similar to previous methods[23]. The propagated embeddings are added as a residual to the initial embedding to get the output Compositional Embedding $\mathcal{Y}_A = \hat{\mathcal{Y}} + \mathcal{Y}_P$. This propagation is done for multiple heads where each head can learn to identify separate important properties between compositions. We set the number of heads to six for CAPE. In the end, a three-layer non-linear Multi-Layer Perceptron (MLP) projects concatenation of all the heads into a shared semantic space to get $\mathcal{Y}_F = \phi(\mathcal{Y}_A)$ where ϕ represents three layer non-linear MLP. For projection, ϕ expands the input embeddings $\mathcal{Y}_A$ into 4096 dimensions, then projects it into the original dimension D. Last layer of ϕ projects embeddings into shared semantic space and is defined as $\mathcal{W}_l \in \mathbb{R}^{|D|}$ where $D = |f(x)|$ represents dimensionality of image features, W_l represents weights of layer. Each of the first two layers are followed by LayerNorm [3], ReLu [1] and Dropout [35] where dropout rate is set at $p = 0.5$. The last layer is followed by the activation function ReLu [1].

Measuring Compatibility of an Image to Composition. Given an image x, we pass it through a learnable feature extractor f to get feature respresentation $f(x)$. The compatibility of an Image to each composition is measured to get the score s:

$$s(x, Y_i) = \frac{f(x) \cdot \mathcal{Y}_{F_i}}{|f(x)||\mathcal{Y}_{F_i}|} \qquad (3)$$

Objective function. We define a cross-entropy on top of our scoring function to learn the feature extractor f and CAPE-Propagator in an end-to-end manner.

$$L = -\log(\frac{\exp s(x, Y_i)}{\sum_{j \in \mathcal{Y}_s} \exp s(x, Y_j)}) \qquad (4)$$

By optimizing the full model end to end, CAPE learns to identify critical propagation paths between compositions leading to more generalizable embeddings.

4. Comparison with State of the Art

Datasets. We evaluate our methodology on three standard benchmark datasets MIT-States [10], CGQA [23] and UT-Zappos [40, 41]. MIT-States [10] dataset was collected

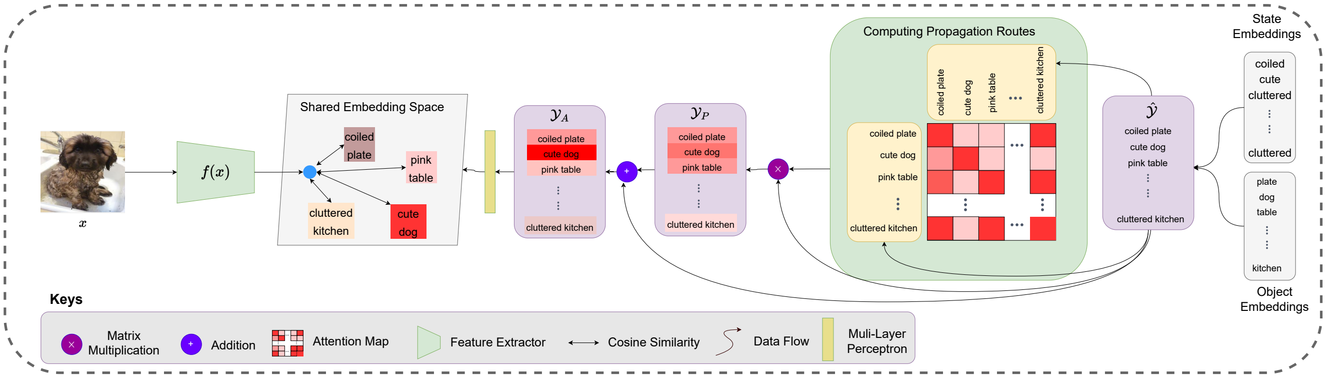

Figure 3. Compositional Attention Propagated Embedding (CAPE) learns interdependency between compositions. We concatenate word embeddings of states and objects to form embeddings of compositions $\mathcal{Y}$. During training, critical propagation routes are learnt by exploiting self-attention mechanism. These propagation routes are used to calculate an updated representation of embeddings in a residual manner. In the end, we learn to project these embeddings into shared semantic space along with image embeddings.

using an older search engine with limited human annotations and significant label noise. This dataset is more abstract than other datasets and includes intangible objects and states such as ancient city, old town. MIT-States [10] consists of 53k images with 245 objects, 115 states and 1252 compositions. Out of all compositions, 300 compositions are in the validation set, and 400 compositions are in the test set.

The second dataset that we use is CGQA [23]. CGQA [23] is a relatively newer dataset with the largest composition space among all three datasets. It consists of 38k images, 453 states, and 870 objects. There are 6963 seen compositions, 1368 unseen compositions in the validation set and 1047 unseen compositions in the test set.

UT-Zappos [40, 41] is relatively simpler dataset and contains 16 states and 12 objects. It has 29k images, 83 seen compositions, 15 unseen compositions in the validation set and 18 unseen compositions in the test set.

Implementation Details. We use Resnet-18 [8] to extract 512 dimensional feature vector for each image. Resnet-18 [8] is pre-trained on ImageNet [31] dataset. We utilize word embeddings for states and objects. For UT-Zappos [40, 41] and MIT-States [10], we use concatenation of FastText [5] and Word2Vec [20, 21]. For CGQA [23], we use only word2vec [20, 21] embeddings to represent states and objects. We use PyTorch to implement our methodology. We use Adam Optimizer [11] with initial learning rate of 5.0×10^{-05} and batch size 30. We train all our models for 120 epochs.

Metrics. We follow the evaluation setting proposed in [28]. We evaluate our methodology on Area Under Curve (AUC), Harmonic Mean (HM), Seen Accuracy (S) and Unseen Accuracy (U). Seen accuracy is calculated on seen compositions, and unseen accuracy is calculated on unseen compositions. Harmonic Mean (HM) is calculated on Seen and Unseen accuracy. AUC is calculated based on the variation of calibration bias between seen and unseen compositions and represents performance at different operating points [14].

4.1. Results

Our approach outperforms all methodologies in AUC with especially significant improvement in CGQA [23], the most challenging dataset. We achieve SOTA AUC of 4.6% in CGQA dataset as compared to the last best result of CGE [23] of 4.2%. We outperform unseen accuracy (U) and Harmonic Mean (HM). We set a new state-of-the-art of 16.3% in Harmonic Mean and 16% in unseen accuracy. In seen accuracy, we are comparable with the previous state-of-the-art CGE [23] by achieving 33.0%. This impressive performance demonstrates our method's scalability to a large compositional space.

In the MIT-States dataset, we set a new state of the art in AUC by achieving 6.7%. In Seen and Unseen accuracy, state of art results is set by [23] by achieving 32.8% and 28.3% respectively. We are comparable to the previous state-of-the-art in seen and unseen accuracy by achieving 32.1% and 28.0%, respectively. MIT-States [10] dataset has considerable label noise that can affect the interdependency structure discovery. However, our comparable results with the previous state-of-the-art show robustness to this label noise.

UT-Zappos is the smallest dataset and contains compositions that can not be differentiated visually [23]. We set a new state of the art by achieving 35.2% AUC. We achieve 68.5% unseen accuracy and 62.3% seen accuracy, that is comparable to the previous state of the art.

We observe that we outperform all baseline methods consistently on large search spaces like CGQA [23]. This is because while previous algorithms only use a basic notion of compositional interdependency structure, we focus on a more complex view by integrating the discovery of the interdependency structure in our approach. This leads to the dis-

	MIT-States				UT-Zappos				C-GQA			
Method	**S**	**U**	**HM**	**AUC**	**S**	**U**	**HM**	**AUC**	**S**	**U**	**HM**	**AUC**
AoP [24]	14.3	17.4	9.9	1.6	59.8	54.2	40.8	25.9	17.0	5.6	5.9	0.7
LE+ [22]	15.0	20.1	10.7	2.0	53.0	61.9	41.0	25.7	18.1	5.6	6.1	0.8
TMN [28]	20.2	20.1	13.0	2.9	58.7	60.0	45.0	29.3	23.1	6.5	7.5	1.1
SymNet [15]	24.2	25.2	16.1	3.0	49.8	57.4	40.4	23.4	26.8	10.3	11.0	2.1
Compcos [18]	25.3	24.6	16.4	4.5	59.8	62.5	43.1	28.1	28.1	10.1	12.4	2.3
CGE$_{ff}$ [23]	28.7	25.3	17.2	5.1	56.8	63.6	41.2	26.4	28.1	10.1	11.4	2.3
Co-CGE$_{ff}$ [19]	27.8	25.2	17.5	5.1	58.2	63.3	44.1	29.1	29.3	11.9	12.7	2.8
CGE [23]	**32.8**	28.0	**21.4**	6.5	**64.5**	**71.5**	**60.5**	33.5	**33.5**	15.5	16.0	4.2
Co-CGE [23]	32.1	**28.3**	20.0	6.6	62.3	66.3	48.1	33.9	33.3	14.9	15.5	4.1
SCEN [14]	29.9	25.2	18.4	5.3	63.5	66.3	47.8	32.0	28.9	12.1	12.4	2.9
CAPE$_{ff}$ (Ours)	30.5	26.2	19.1	5.8	60.4	67.4	45.5	31.3	32.9	15.6	16.3	4.2
CAPE (Ours)	32.1	28.0	20.4	**6.7**	62.3	68.5	49.5	**35.2**	33.0	**16.4**	**16.3**	**4.6**

Table 1. Results on MIT-States, UT-Zappos and C-GQA. We report best seen (S) accuracy, best unseen (U) accuracy, best harmonic mean (HM), and area under the curve (AUC) on the compositions. ff denotes frozen feature extractor. Our model outperforms prior methodologies on AUC in all the datasets.

covery of beneficial connections between different related compositions.

Visualizing Propagation routes. Table 2 shows the top 5 and bottom 5 activations in the attention map from Multi-Head attention. The listed compositions have the top five and bottom five actions for a given query. We conduct this analysis on CGQA [23] and MIT-States [10] dataset. We report interesting compositions from all heads of Multi-Head attention. Results in Table 2 confirm our hypothesis that a complex interdependency structure exists between compositions that do not share primitives. “Cracked Mud” is related with “Cracked Window” and “Shattered Window”. Shattered Window composition may visually contain a broken mirror with cracks visually similar to the state “Cracked”. Learning how “Shattered” looks also updates how “Cracked” looks. We observe the same with the composition “Sliced Salmon” related to “Pureed Seafood”. While “Sliced Salmon” and “Pureed Seafood” do not share any primitive, they contain visual information as they come from a similar family of dishes containing seafood. Furthermore, “Blue Mug” is related to “Cluttered Kitchen” and “Cluttered Desk”. An image of “Cluttered Kitchen” might also contain a blue mug. Learning how “Blue Mug” looks will also help determine a “Cluttered Kitchen” or “Cluttered Desk” due to the propagation of knowledge. We also observe that the bottom 5 activation always contain completely irrelevant compositions. Such as, “Red Floor” has the least activation for “Green Salad”, a food category. The same is observed with “Winter Picture”, which has least activation for “Yellow Desk”. On the other hand, “Winter Picture” has high activations for “Leafless”, “Barren”, “Forested”, and “Tree”, and these objects can be found in an image of a “Winter Picture”. Furthermore, object “Redwood” in “Weathered Redwood” have high activations for object “Log” that are both representation of wood. Similarly, states “Weathered”, “Broken”, “Splintered” are related with each other such that all of them represent a damaged or worn out state of an object. We also observe that our approach is able to find propagation routes for compositions with shared primitives such as, “Yellow Wall” have highest activation for “Yellow Chair”, “Dry Pond” have high activations for “Dry Bush” and “Dry Forest” and “Weathered Redwood” have high activation for “Weathered Log”. This shows that our approach can find simple propagation routes that share primitives and also complex propagation routes share some property of state or object and not expressed in primitives.

Impact of Feature Representations. Consistent with previous works, we experimented with the frozen backbone in our network. CAPE$_{ff}$ represents our approach with frozen backbone in Table 1. We outperform all previous methods with a frozen backbone, including a recent approach SCEN [14] that is specially developed for only frozen features. In CGQA [23], we match state of the art from CGE [23] in AUC and outperform in harmonic mean and unseen accuracy. In UT-Zappos [40, 41], CAPE$_{ff}$ outperforms in unseen accuracy over previous approaches with frozen backbone. While in the MIT-States dataset, CAPE$_{ff}$ outperforms previous approaches with the frozen backbone in all metrics.

4.2. Qualitative Results

In this section, we will discuss our qualitative results in detail. Figure 4 shows qualitative results of our approach on MIT-States [10] and CGQA [23] dataset. We observe that our approach can predict compositions of images containing noise and other valid compositions. Composition “Crumpled Jacket” contains objects clouds and grass. Like-

	MIT-States	
Query	**Top 5**	**Bottom 5**
Cracked Mud	Cracked Window, Shattered Window, Cracked Door, Broken Window, Cracked Mirror	Fresh Bread, Fresh Butter, Fresh Cheese, Fresh Meat, Fresh Orange
Weathered Redwood	Broken Log, Splintered Log, Peeled Log, Burnt Log, Weathered Log	Old Bus, Old City, Old Car, Old Truck, Old Street
Rusty Bridge	Old Log, Old Library, Old Boat, Old Computer, Old Bear	Broken Column, Broken City, Broken Lightbulb, Broken Jewelry, Broken Necklace
Dry Pond	Dry Bush, Wet Bush, Damp Bush, Barren Bush, Dry Forest	Eroded Shore, Broken Shore, Weathered Shore, Verdant Shore, Mossy Shore
Sliced Salmon	Pureed Seafood, Pureed Fish, Pureed Salmon, Diced Seafood, Cooked Seafood	Ruffled Bed, Wide Blade, Draped Bed, Ruffled Shower, Ruffled Leaf
	CGQA	
Query	**Top 5**	**Bottom 5**
Red Floor	Carpeted Floor, Textured Floor, Cracked Floor, Painted Floor, Sripped Floor	Green Salad, Green Brocoli, Green Cabbage, Green Asparagus, Green Apple
Winter Picture	Overgrown Tree, Forested Tree, Leafless Tree, Overgrown Weeds, Barren Tree	Yellow Desk, Comfortable Chair, Yellow Chair, Red Desk, Plaid Chair
Large Cooler	Large Snow, Large Crust, Large Mountain, Large Omelette, Large Barier	Full Hangar, Full Floor, Full Ground, Open Door, Transparent Door
Blue Mug	Cluttered Kitchen, Cluttered Desk, Cluttered Shelf, Cluttered Office, Cluttered Counter	Leafless Bush, Leaflless Tree, Leafless Branch, Bushy Bush, Bamboo Bush
Yellow Wall	Yellow Chair, Colorful Chair, Red Chair, Purple Chair, Striped Chair	Bare Tree, Huge Tree, Overgrown Tree, High Tree, Tall Tree

Table 2. The table shows the Top 5 and bottom 5 pairs for the query pairs given in the first column taken from MIT-States and CGQA datasets. Top 5 and bottom 5 pairs are selected from the attention matrix before softmax. We observe that query compositions have highest activations for similar compositions shown in column "Top 5" and least activations for different compositions shown in column "Bottom 5". We also observe that our approach can find diverse propagation routes.

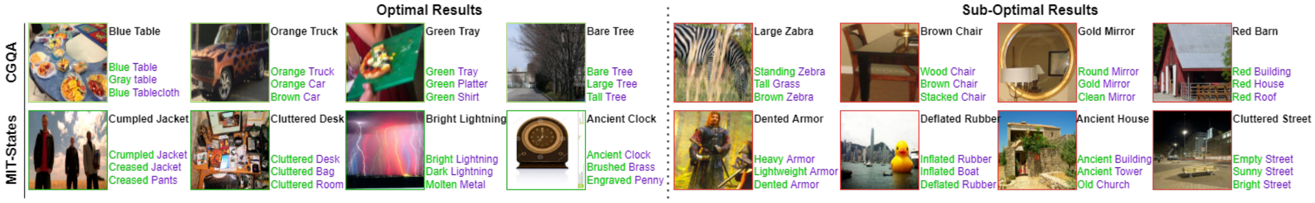

Figure 4. Shows qualitative results of our approach on MIT-States and CGQA dataset. First four images in each row from the left show positive results where our approach predicted correct composition. Last four image show sub optimal results where our approach did not predict correct composition. First row contains results from CGQA dataset and second row contains results from MIT-States dataset.

wise, the composition "Bare Tree" also contain objects road and a car. The composition "Cluttered Desk" contains objects table and frame. The composition "Green Tray" contains object "Pizza" as well.

We also show sub-optimal results where our approach did not predict compositions correctly. We observe that in multiple sub-optimal results, the correct composition is present in the top 3 predictions. Sub-optimal results that contain correct predictions in the top 3 results are "Dented Armor", "Deflated Rubber", "Brown Chair", and "Gold Mirror". We also observe label noise in MIT-States results, such as "Deflated Rubber" is a duck floating because it is "Inflated", as predicted by our approach. Furthermore, "Dented Armor" does not show any dents in it, and our approach predicts it as "Heavy Armor" which is more close to the visual representation of the image. We also observe that our sub-optimal predictions are not necessarily incorrect. For example, composition "Gold Mirror" is a circular mirror, and our approach predicts it as "Round Mirror". Composition "Brown Chair" predicted as "Wooden Chair" is a chair made of wood. Composition "Large Zebra" predicted as "Standing Zebra" shows an image of "Zebra" while standing. There are multiple valid compositions for a given image. We come to the same conclusion as [23] that problem of CZSL should be considered a multi-label problem.

5. Ablation

In this section, we ablate over different configurations of the attention mechanism of CAPE. Results of ablation for harmonic mean and AUC are shown in Table 3. CAPE in Table 3 represents our original approach proposed in section 3. We conduct our ablation on MIT-States [10] and CGQA [23] datasets. Results are reported on validation sets from both datasets. Our original approach as proposed in section 3 achieves 8.2% AUC and 23.2% Harmonic mean on MIT-States [10] dataset. It achieves 6.1% AUC and 19.5% harmonic mean on CGQA [23] dataset. The architecture of the CAPE-Propagator does not change during these experiments. In all experiments, number of heads in Multi-Head Attention are kept constant at 6. We keep all hyperparameters constant across all experiments.

Self Attention on States, Objects and Compositions (CAPE$_{self}$). This configuration is represented by CAPE$_{self}$ in Table 3. We create one tensor $\hat{\mathcal{Y}}_{self}$ where $|\hat{\mathcal{Y}}_{self}| = |\mathcal{S}| + |\mathcal{O}| + |\mathcal{Y}_s|$ and containins word embeddings of states, objects and compositions. During testing, we append unseen compositions to $\hat{\mathcal{Y}}_{self}$ to get $|\hat{\mathcal{Y}}_{self}| = |\mathcal{S}| + |\mathcal{O}| + |\mathcal{Y}_s| + |\mathcal{Y}_n|$. Compositions are calculated as mean of their state and object word embeddings. CAPE-Propagator projects $\hat{\mathcal{Y}}_{self}$ to shared semantic space to get $\mathcal{Y}_F$. Final composition scores are calculated between image embedding and compositional nodes in $\mathcal{Y}_F$ by using compatibility function shown in Eq. 3. During training CAPE-Propagator learns propagation routes by exploiting self-attention mechanism as explained in section 3. CAPE$_{self}$ configuration is very similar to the configuration used in [23] that also applies supervision on only compositional nodes. CAPE$_{self}$ achieves 8.1% AUC and 23.0% harmonic mean on the MIT-States dataset. It achieves 6.0% AUC and 19.2% harmonic mean on CGQA [23] dataset. We observe a performance loss as CAPE$_{self}$ lags in both datasets in HM and AUC.

Cross Attention on Primitives and Self Attention on Compositions (CAPE$_{dual}$). This configuration is represented by CAPE$_{dual}$ in Table 3. CAPE$_{dual}$ applies cross attention on states and objects and self-attention to their compositions. Firstly, we apply cross attention on word embeddings of state and objects. We use one Multi-Head attention to apply cross attention between states and objects to get $\hat{\mathcal{Y}}_{states}$. We use second Multi-Head attention to apply cross attention between objects and states to get $\hat{\mathcal{Y}}_{objects}$. Afterwards, we concatenate $\hat{\mathcal{Y}}_{states}$ and $\hat{\mathcal{Y}}_{objects}$ to get compositions $\hat{\mathcal{Y}}_{dual}$ where $|\hat{\mathcal{Y}}_{dual}| = |\mathcal{Y}_s|$ during training and $|\hat{\mathcal{Y}}_{dual}| = |\mathcal{Y}_s| + |\mathcal{Y}_n|$ during testing. We input $\hat{\mathcal{Y}}_{dual}$ into CAPE-Propagator to get $\mathcal{Y}_F$. CAPE-Propagator discovers propagation routes on $\hat{\mathcal{Y}}_{dual}$ by exploiting self-attention mechanism as explained in section 3. Final composition scores are calculated between image embeddings and $\mathcal{Y}_F$ by using compatibility function shown in Eq. 3. CAPE$_{dual}$ achieves 8.2% AUC and 23.2% harmonic mean on MIT-States dataset [10]. It achieves 6.0% AUC and 19.5% harmonic mean on CGQA [23] dataset. This configuration matches the performance of CAPE in MIT-States [10] dataset but lags behind in AUC in CGQA [23] dataset.

Multi-Layer Perceptron (MLP) as a replacement for CAPE-Propagator. In this experiment, we used Multi-Layer perceptron instead of CAPE-Propagator or Multi-Head attention. The results are represented by heading "MLP" in Table 3. MLP was configured to have same amount of parameters as CAPE-Propagator. MLP achieves 7.5% AUC and 22.1% harmonic mean on MIT-States [10] dataset. It achieves 5.4% AUC and 18.3% harmonic mean on CGQA [23] dataset. Since MLP does not model interdependency structure, it lags in all datasets.

We observe that our original configuration, as proposed in section 3 outperforms all configurations. Introducing additional embeddings or MHAs leads to poorer performance. CAPE is an effective approach that exploits self-attention to learn hidden interdependency structures between compositions caused by the primitives. Introducing new networks, like in the case of CAPE$_{dual}$ leads to an increase in the number of learnable parameters and a redundant cross-attention mechanism. On the other hand, primitives do not get supervision in CAPE$_{self}$ leading to poorer performance.

	MIT-States		**CGQA**	
Method	**AUC**	**HM**	**AUC**	**HM**
CAPE$_{dual}$	**8.2**	**23.2**	6.0	**19.5**
CAPE$_{self}$	8.1	23.0	6.0	19.2
MLP	7.5	22.1	5.4	18.3
CAPE	**8.2**	**23.2**	**6.1**	**19.5**

Table 3. *Ablation* over different configurations of CAPE. We report highest achieved AUCs on MIT-States [10] and CGQA [23] validation dataset.

6. Conclusion

We propose a novel approach to the task of Compositional Zero-Shot Learning. We evaluated our approach on three benchmark datasets (CGQA [23], MIT-States [10], UT-Zappos [40, 41]) extensively. We argued that there is a complex interdependency structure between compositions that do not share any primitives. We exploited the attention mechanism to discover this interdependency structure and propagate it to unseen classes. Our qualitative analysis reaffirm our original hypothesis that there exists a complex interdependency structure between compositions. Our approach outperforms prior methodologies and shows improvement in all benchmark datasets. Our qualitative results demonstrate that there can be multiple valid predictions of one image, and the problem of CZSL should be considered a multi-label problem. We encourage future works to consider this aspect while building their methodologies.

Acknowledgement. This work was partially funded by the European project FLUENTLY (101058680) and partially by the European project INFINITY (883293).

References

[1] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

[2] Yuval Atzmon, Felix Kreuk, Uri Shalit, and Gal Chechik. A causal view of compositional zero-shot recognition. 2020.

[3] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[4] Yoshua Bengio, Tristan Deleu, Nasim Rahaman, Rosemary Ke, Sébastien Lachapelle, Olexa Bilaniuk, Anirudh Goyal, and Christopher Pal. A meta-transfer objective for learning to disentangle causal mechanisms. *arXiv preprint arXiv:1901.10912*, 2019.

[5] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146, 2017.

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[7] Olivier Goudet, Diviyan Kalainathan, Philippe Caillou, Isabelle Guyon, David Lopez-Paz, and Michele Sebag. Learning functional causal models with generative neural networks. In *Explainable and interpretable models in computer vision and machine learning*, pages 39–80. Springer, 2018.

[8] 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.

[9] Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. In *International workshop on similarity-based pattern recognition*, pages 84–92. Springer, 2015.

[10] Phillip Isola, Joseph J Lim, and Edward H Adelson. Discovering states and transformations in image collections. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1383–1391, 2015.

[11] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[12] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[13] Kwonjoon Lee, Huiwen Chang, Lu Jiang, Han Zhang, Zhuowen Tu, and Ce Liu. Vitgan: Training gans with vision transformers. *arXiv preprint arXiv:2107.04589*, 2021.

[14] Xiangyu Li, Xu Yang, Kun Wei, Cheng Deng, and Muli Yang. Siamese contrastive embedding network for compositional zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9326–9335, 2022.

[15] Yong-Lu Li, Yue Xu, Xiaohan Mao, and Cewu Lu. Symmetry and group in attribute-object compositions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11316–11325, 2020.

[16] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1833–1844, 2021.

[17] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[18] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. Open world compositional zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5222–5230, 2021.

[19] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. Learning graph embeddings for open world compositional zero-shot learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

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

[21] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

[22] Ishan Misra, Abhinav Gupta, and Martial Hebert. From red wine to red tomato: Composition with context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1792–1801, 2017.

[23] Muhammad Ferjad Naeem, Yongqin Xian, Federico Tombari, and Zeynep Akata. Learning graph embeddings for compositional zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 953–962, 2021.

[24] Tushar Nagarajan and Kristen Grauman. Attributes as operators: factorizing unseen attribute-object compositions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 169–185, 2018.

[25] Leonardo Noriega. Multilayer perceptron tutorial. *School of Computing. Staffordshire University*, 2005.

[26] Giambattista Parascandolo, Niki Kilbertus, Mateo Rojas-Carulla, and Bernhard Schölkopf. Learning independent causal mechanisms. In *International Conference on Machine Learning*, pages 4036–4044. PMLR, 2018.

[27] Nick Pawlowski, Daniel Coelho de Castro, and Ben Glocker. Deep structural causal models for tractable counterfactual inference. *Advances in Neural Information Processing Systems*, 33:857–869, 2020.

[28] Senthil Purushwalkam, Maximilian Nickel, Abhinav Gupta, and Marc'Aurelio Ranzato. Task-driven modular networks for zero-shot compositional learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3593–3602, 2019.

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learn-

ing transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[30] Frank Ruis, Gertjan Burghouts, and Doina Bucur. Independent prototype propagation for zero-shot compositionality. *Advances in Neural Information Processing Systems*, 34, 2021.

[31] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[32] Bernhard Schölkopf. Causality for machine learning. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, pages 765–804. 2022.

[33] Subana Shanmuganathan. Artificial neural network modelling: An introduction. In *Artificial neural network modelling*, pages 1–14. Springer, 2016.

[34] F Sorbello, A Tarantino, M Tarantino, and G Vassallo. Artificial neural networks and real world problems: Character recognition and chemical compounds classification. *WIT Transactions on Information and Communication Technologies*, 11, 1970.

[35] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[36] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxvit: Multi-axis vision transformer. *arXiv preprint arXiv:2204.01697*, 2022.

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[38] Jun Wang, Mingfei Gao, Yuqian Hu, Ramprasaath R Selvaraju, Chetan Ramaiah, Ran Xu, Joseph F JaJa, and Larry S Davis. Tag: Boosting text-vqa via text-aware visual question-answer generation. *arXiv preprint arXiv:2208.01813*, 2022.

[39] Mengyue Yang, Furui Liu, Zhitang Chen, Xinwei Shen, Jianye Hao, and Jun Wang. Causalvae: Disentangled representation learning via neural structural causal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9593–9602, 2021.

[40] Aron Yu and Kristen Grauman. Fine-grained visual comparisons with local learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 192–199, 2014.

[41] Aron Yu and Kristen Grauman. Semantic jitter: Dense supervision for visual comparisons via synthetic images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5570–5579, 2017.

[42] Bowen Zhang, Shuyang Gu, Bo Zhang, Jianmin Bao, Dong Chen, Fang Wen, Yong Wang, and Baining Guo. Styleswin: Transformer-based gan for high-resolution image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11304–11314, 2022.

[43] Cheng Zhang, Kun Zhang, and Yingzhen Li. A causal view on robustness of neural networks. *Advances in Neural Information Processing Systems*, 33:289–301, 2020.