Abstract

The goal of open-world compositional zero-shot learning (OW-CZSL) is to recognize compositions of state and objects in images, given only a subset of them during training and no prior on the unseen compositions. In this setting, models operate on a huge output space, containing all possible state-object compositions. While previous works tackle the problem by learning embeddings for the compositions jointly, here we revisit a simple CZSL baseline and predict the primitives, i.e., states and objects, independently. To ensure that the model develops primitive-specific features, we equip the state and object classifiers with separate, non-linear feature extractors. Moreover, we estimate the feasibility of each composition through external knowledge, using this prior to remove unfeasible compositions from the output space. Finally, we propose a new setting, i.e., CZSL under partial supervision (pCZSL), where either only objects or state labels are available during training, and we can use our prior to estimate the missing labels. Our model, Knowledge-Guided Simple Primitives (KG-SP), achieves state of the art in both OW-CZSL and pCZSL, surpassing most recent competitors even when coupled with semi-supervised learning techniques. Code available at: https://github.com/ExplainableML/KG-SP.

1. Introduction

As humans, we interact with objects depending on their state. For instance, we use ripe lemons rather than moldy ones to prepare a lemonade, and we clean dirty dishes after using them. Algorithms that can recognize objects together with their state are crucial for autonomous agents to show the same high-level interactions capabilities we have. In the literature, this problem is studied under the name of Compositional Zero-shot Learning (CZSL). In CZSL, we are given a training set with images of objects in a subset of their possible states and, at test time, the goal is to recognize compositions of the same set of objects and states, even unseen during training. Since an object has a different appearance depending on its state (e.g., dry dog vs wet dog) and a state modifies objects in different ways (e.g., wet dog vs wet car), the challenge of CZSL is modeling how states and objects interact with each other, extrapolating this knowledge from seen to unseen compositions. Under this perspective, multiple works modeled the interactions of objects and states, either through compositional classifiers [26, 31], or a shared embedding space [23, 28, 29].

Despite their effectiveness, [24] showed how the performance of CZSL methods degrade in the open-world setting (OW-CZSL). In OW-CZSL, there are no priors on the unseen compositions, and models must consider all possible compositions at test time. Due to the large cardinality of the output space, it is difficult to produce discriminative embeddings for the unseen compositions [24]. Inspired by the findings of [24], in this work we explore a completely different direction. Specifically, we design an architecture that disregards the compositional nature of the problem and produces the initial predictions independently for objects and states. The idea is that while discriminating between compositions is hard in OW-CZSL due to the large search space, recognizing primitives (i.e., objects and states) in isolation is easier since

Figure 1. We consider the problems of open-world CZSL (OW-CZSL), where we lack priors on unseen compositions at test time, and CZSL under partial supervision (pCZSL) where we also lack compositional labels during training (left). We tackle them by independently predicting object (red) and state (blue) labels and by using external knowledge (bottom) to estimate the feasibility of compositions, reducing the search space during inference and improving pseudo-labeling during training in pCZSL.
1) the cardinality of the two sets is much lower and 2) the sets are fixed at both training and test time.

Inspired from [26] and [19], we design a simple method that predicts objects and states with two independent classifiers. Since recognizing states requires different features w.r.t. recognizing objects, instead of having a shared feature representation, we train our model with two different non-linear feature extractors. Furthermore, since not all compositions are equally feasible in reality (e.g. ripe dog) we can refine the predictions of our model by eliminating less feasible compositions from the output space. With this goal, we use external knowledge (i.e. ConceptNet [39]) to estimate the compatibility between a state and an object, using these estimates to remove less feasible compositions at test time. We name our model Knowledge-Guided Simple Primitives (KG-SP). As our KG-SP method does not require compositional labels during training, we explore a new challenging setting, i.e. CZSL under partial supervision (pCZSL). In pCZSL, training samples have either only object or state annotation, but not both. Here we use our prior on feasible compositions to aid pseudo-labeling during training. Experiments show that KG-SP is either competitive or surpasses the current state of the art in OW-CZSL and outperforms recent CZSL approaches on pCZSL setting. Figure 1 provides an overview of KG-SP and the two tasks.

Contributions. To summarize, 1) inspired by [19, 26], our model predicts state and objects independently while at the same time removing less feasible compositions from the output space based on external contextual information about the feasibility of certain compositions; 2) we explore the problem of CZSL under partial supervision, where either object or state information is missing in the ground-truth; 3) we adapt recent baselines for pCZSL showing that KG-SP outperforms them even when coupled with semi-supervised learning techniques in both OW-CZSL and pCZSL settings.

2. Related works

Compositional zero-shot learning aims to recognize compositions of states and objects in images, even unseen during training. The main challenge of this setting is modeling how states modify objects, generalizing this capability to unseen compositions. Most of the previous works focused on how to model the interactions between states and objects either at the parameter level or in a given representation space. For instance, [26, 31] proposed to generate a classifier for a given state-object composition given two classifiers (or embeddings) for specific state and object primitives, using either a compositional module [26] or a gating network [31]. Differently, [23, 29] model each state as an operator transforming object embeddings, imposing properties on the state operators (e.g. commutativity, symmetry). In [29] the state operators are linear, while in [23] they are coupling and decoupling networks. Recently, [28, 36] used graph convolutional networks [17] to model the interactions between state, objects and their compositions. Differently, [2] tackles CZSL from a causality perspective, learning disentangled objects and states representations. In this work, we revisit VisProd [26], predicting objects and states in isolation, showing that this strategy is effective in OW-CZSL. As in [24], we estimate the feasibility of each composition to improve the model’s performance. However, we use ConceptNet to this aim, rather than compositional annotations, being the first to tackle CZSL without compositional labels during training.

Multi-task learning. Since we predict state and object independently, our work is related to Multi-Task [7,15,27,35,37] and Multi-Domain learning [4,22,32,33], where the goal is to learn a unique model able to address different visual tasks. Most of the approaches in this domain either learn task-specific parameters [4,22,32,33,37] and how to combine them [27,35], or focus on re-weighting different loss functions [15]. While we use multi-task learning to design primitive classifiers for CZSL, our final goal is different since we compose predictions from separate output spaces.

Learning from Partial Supervision. Our CZSL setting without compositional labels is related to semi-supervised learning and learning with missing labels. In semi-supervised learning, both labeled and unlabeled samples are available, and the goal is to effectively use the unlabeled samples. Popular ideas revolve around consistency regularization [5,6,38], and self-training [10,20,34]. Unlike semi-supervised learning, in pCZSL, all samples are labeled, but partially. Thus, we can also exploit the prior on how objects interact with states to estimate the missing labels.

For what concerns learning with missing labels, this is most prevalent in multi-label scenarios where it is unfeasible to annotate all labels that are present in a single image. Approaches in this field usually model the correlation among labels [8,9,12,18] to impose semantic objectives on missing ones. While we are also interested in learning from partial supervision, our labels lie on two separate spaces (i.e. objects and states), and the missing labels (e.g. state) influences the appearance of the positive one (e.g. object). In this setting, the main challenge is to model how the two spaces influence each other without any compositional supervision.

3. Knowledge Guided Simple Primitives

Problem formulation. CZSL [24] aims to recognize compositions of a set of objects $O$ and a set of states $S$. Formally, we are given a training set $T = \{(x, y)\}_{i=1}^{N}$, where $N$ is the size of the training set, $x \in X$ denotes an image in the input space $X$ and $y \in Y_s$ is its label in the set of seen compositions $Y_s$. The goal is to learn a model that can recognize a set of compositions $Y_t = Y_u \cup Y_d$, where $Y_u$ is a set of unseen compositions (i.e. $Y_u \cap Y_s = \emptyset$) and $Y_d \subseteq Y_s$ with $Y$ being the set of all possible compositions, i.e. $Y = S \times O$. OW-CZSL and pCZSL settings. In this work, we con-
sider two different CZSL settings. Open-World CZSL (OW-CZSL) \cite{24} assumes no prior on the set of unseen compositions at test time. This means that the model needs to operate on the full compositional space, \textit{i.e.} \(Y_t = Y\). Consequently, the number of unseen compositions is much larger than the number of seen ones \(i.e. Y_u = Y \setminus Y_s\), thus the main challenge is operating in a very large output space where most of the compositions are unseen and thus hard to discriminate.

In this work, we also consider a new challenging task, namely CZSL under partial supervision (pCZSL), where the training set does not contain any compositional label and all training images have either object or state label, but not both. This setting is more realistic than standard CZSL since most datasets are collected with single labels (\textit{e.g.} only object-level information) and collecting multiple labels is expensive and time-consuming. Formally, we consider the labels of our training set \(\mathcal{T}\) to be of the form \(y = (s, u) \lor y = (u, o), \forall (x, y) \in \mathcal{T}\), with \(s \in \mathcal{S}, o \in \mathcal{O}\) and \(u\) denoting an unknown label. Note that, as a consequence of this formulation, the set of training compositions \(Y_s\) is not known a priori anymore. This implies that, as in OW-CZSL, we need to consider the full compositional space at test time, \textit{i.e.} \(Y_t = Y\). Moreover, since no training image contains both object and state labels, we do not have explicit supervision on how states modify objects and vice-versa.

In the following, we describe the two components of our framework, Simple Primitives (SP) where we predict the primitives, \textit{e.g.} object and states, independently and Knowledge Guidance (KG) where we use external resources that guide our model on the feasibility of certain compositions.

### 3.1. Simple Primitives (SP) in KG-SP

Inspired by the early Visual Product (VisProd) baseline \cite{26}, our model completely disregards the compositional nature of the problem and predicts states and objects independently. This idea contrasts with recent approaches \textit{(i.e.} \cite{22,23,24,28,29,31}), explicitly modeling the interactions between objects and states within the model.

Formally, given an image \(x\), we extract its feature representation \(z = \omega(x)\) through a function \(\omega\), mapping images into a feature space \(Z\), \textit{i.e.} \(\omega : X \to Z\). We then have an object classifier \(\phi_o : Z \to \Delta_O\) that maps \(z\) to a vector in the probability simplex \(\Delta_O\), spanning all object categories. Similarly, we have another classifier that maps \(z\) to a probability over the states, \textit{i.e.} \(\phi_s : Z \to \Delta_S\). During training, we minimize the cross-entropy loss for both the object and state predictions. Specifically, we minimize:

\[
\mathcal{L}_{\text{visprod}} = \sum_{i=1}^{N} \left[ I_{s_i \neq u} \mathcal{L}_{\text{state}}(x_i, s_i) + I_{o_i \neq u} \mathcal{L}_{\text{obj}}(x_i, o_i) \right]
\]

\[
= - \sum_{i=1}^{N} \left[ I_{s_i \neq u} \log \phi_s(z_i, s_i) + I_{o_i \neq u} \log \phi_o(z_i, o_i) \right]
\]

where \(z_i = \omega(x_i), \phi_o(z, o)\) is the probability of the object \(o\) assigned by \(\phi_o\) to the input \(z\), and \(\phi_s(z, s)\) is the probability of the state \(s\) assigned by \(\phi_s\) to the input \(z\). In Eq. (1), \(I\) is an indicator function used to not compute the loss in pCZSL, in the absence of primitive labels. Our prediction function is:

\[
f = \arg\max_{(s, o) \in Y} \phi_o(w(x), o) \cdot \phi_s(w(x), s).
\]

Although learning simple primitives independently like this may not be effective in standard CZSL, we argue that the capability to separate state and objects predictions is crucial in OW-CZSL, where the search space is too large if predictions are made over the full compositional space.

In the original VisProd formulation, objects and state predictors are simple linear layers operating over the same feature vector. However, this choice is suboptimal and leads to low results in practice. In fact, by using separate linear layers, VisProd addresses CZSL as a multi-task learning problem \cite{7,15,27,32,33,35,37}, where there are two different tasks (\textit{i.e.} states and objects prediction) that share the same feature extractor while differing only for the classification head. However, multiple works in multi-task (MTL) and multi-domain learning (MDL) discussed how fully sharing the parameters to extract the feature representation for different tasks (\textit{i.e.} hard-sharing \cite{7}) is suboptimal when the tasks are not strictly related \cite{27,32,33} and may even lead to negative transfer \cite{21}.

In CZSL, recognizing objects is different than recognizing their states. Specifically, the former requires focusing on global features: for instance, distinguishing an animal from another requires focusing on their shapes and skins while distinguishing fruits requires detecting texture-based cues. On the contrary, recognizing states requires focusing on local patterns: for instance, the difference between dry and wet can be detected by the presence of drops in cars and apples while in animals it requires looking at the shape of the fur. With this premise, we need to overcome the limits imposed by hard-parameter sharing to ensure the objects and state classifiers have enough flexibility to learn primitive-specific features. While advanced MTL and MDL techniques can be used for this purpose, in this work we found that it is sufficient to implement the two classifiers as multi-layer perceptrons (MLP) with non-linear activations.

### 3.2. Knowledge Guidance (KG) in KG-SP

In the large output space of OW-CZSL and pCZSL, not all compositions are equally feasible (\textit{e.g.} ripe dog, hairy apple) and taking this prior into account can help in correcting incompatible state-object predictions of our model. In the following we describe how we estimate the feasibility scores and how we use them in our model.

#### Estimating feasibility scores

Formally, let us associate to each composition \((s, o)\) a compatibility score \(c^s_o \in [0, 1]\\).
Since there exist no database contain such information, previous works exploited the set of seen compositions $Y_s$ to estimate $c_o^s$ [24]. Here we explore an alternative direction by using external knowledge. In this way, our estimation is independent of the actual availability of the set $Y_s$ and can also be applied in pCZSL, where $Y_s$ is unknown. While we explored different strategies (see supplementary), we found ConceptNet [39] to give reliable feasibility estimates.

ConceptNet is a knowledge graph connecting words and phrases with labeled edges, extracted from various sources [39]. We can use ConceptNet in two ways. The first is querying for the existence of a composition and the second is querying for the relatedness between two entries (i.e. object and state). Since direct queries are very sparse, we follow the second approach, defining the scores as:

$$c_o^s = \rho_{KB}(s, o)$$

where $\rho_{KB}(s, o)$ returns the relational score between $s$ and $o$. In ConceptNet, these scores are computed from the cosine-similarity of ConceptNet Numberbatch embeddings [40]. The latter are built from ConceptNet adjacency matrix and existing word embeddings (e.g. word2vec [25], GloVe [30]).

**Using the feasibility scores during inference.** Similarly to [24], the most straightforward-way to use the feasibility sores is to remove from the output space less feasible compositions during inference. Thus, our prediction function becomes:

$$f_{KG} = \arg\max_{(s, o) \in Y, c_o^s > 0} \phi_o(z, o) \cdot \phi_s(z, s)$$

where we consider feasible all compositions with $c_o^s > 0$.

**Using the feasibility scores during training for pCZSL.** In pCZSL, we may obtain additional supervision by estimating missing labels. One straightforward way to achieve this is through pseudo-labeling [20], a semi-supervised learning technique that takes the model predictions as ground-truth for unlabeled samples. In pCZSL, this means that, when the state (object) label is missing, pseudo-labeling will impute as label the object (state) predicted with the highest score. To avoid that the pseudo-labels form unfeasible compositions, we can use our prior to aid the pseudo-labeling process.

Given either an object label $o$ or a state label $s$, we estimate their respective state and object pseudo-labels as:

$$\hat{s} \sim \text{Gumbel}(\phi_s(z) \odot c_s), \hat{o} \sim \text{Gumbel}(\phi_o(z) \odot c_o)$$

where $c_o$ is the vector containing the compatibility scores for all states given the object $o$, i.e. $c_o = [c_o^s]_{s \in S}$, and $c_s$ is its counterpart for all objects given the state $s$, i.e. $c_s = [c_s^o]_{o \in O}$. Note that in both equations we sample the pseudo-labels using Gumbel-softmax (Gumbel) [14]. We found this choice helpful to make the model more robust to noisy predictions and less biased toward the training set latent label distribution. Our objective function becomes:

$$L_{\text{visprod}}^{\text{pCZSL}} = \sum_{(x, s) \in T_s} L_{\text{state}}(x, s, \hat{s}) + L_{\text{obj}}(x, s, \hat{o})$$

$$+ \sum_{(x, o) \in T_o} L_{\text{obj}}(x, o) + L_{\text{state}}(x, o, \hat{o}).$$

We use this objective in pCZSL during training and Eq.(1) for OW-CZSL. In both cases we perform inference through Eq.(4). Since we couple independent primitive prediction with external knowledge to refine them, we name the method Knowledge-guided Simple Primitives (KG-SP). Figure 2 illustrates our approach during training and inference.

## 4. Experiments

**Datasets.** We use the three standard datasets for Compositional Zero-Shot Learning, namely UT-Zappos [43,44], MIT-States [13] and the recently proposed C-GQA [28] dataset.

---

1 We assume $S$ and $O$ to be alphabetically ordered.
UT-Zappos contains 12 object categories (shoe types) and 16 state categories (material types), with 83 seen compositions and a compositional space of 192 compositions. MIT States is a larger dataset that contains 245 object categories in 115 possible states. In total, it contains 1, 262 seen compositions and an output space of 28, 175 compositions in OW-CZSL. Finally, C-GQA is a recently proposed dataset with 674 object categories and 413 state categories. It contains 5, 592 training compositions and a full compositional space with 278, 362 compositions.

**Baselines.** In OW-CZSL, we compare KG-SP against standard CZSL approaches; namely Attributes as Operators (AoP) [29], Label Embed+ (LE+) [26], Task Modular Networks (TMN) [31], SymNet [23], Compositional Graph Embeddings (CGE) [28] and Compositional Cosine Logits (CompCos) [24]. In the tables, we refer to the closed world version of CompCos as CompCosCW and the variant of CGE with a frozen feature extractor as CGEff.

In pCZSL, we compare KG-SP with CGE [28] and CompCos [24], the state-of-the-art models in the closed and open-world settings respectively. These methods are adapted to pCZSL by marginalizing their predictions over states/objects, when the state/object information is available, minimizing the cross-entropy loss on the ground-truth annotation. We also experiment with popular semi-supervised learning techniques such as entropy minimization [10] and pseudo-labeling [20], adding them to both CompCos and CGE.

**Evaluation Protocol.** For the OW-CZSL setting, we follow the standard splits of [28, 31], evaluate all the methods on the generalized setting, where the model recognizes samples from both seen and unseen compositions. Following the protocol in [31], we apply a bias on the seen compositions at test time, measuring the performance as best seen (S) and best unseen (U) accuracy, best harmonic mean (HM) as well as the area under the curve (AUC) by varying the bias.

For the pCZSL setting, we propose a new split of the training split set, separating samples with object and state labels. This is done by keeping only the object label for half the samples, while for the remaining half, only the state label, ensuring that every object and state is seen in the training set. Furthermore, for pCZSL setting, the model has no access to seen compositions \( Y_s \). Thus, we evaluate the model in the full compositional space, without subtracting any bias on \( Y_s \). Therefore, we use as metric the seen (S) and unseen (U) accuracy, and their harmonic mean (HM), as it is standard in Zero-Shot Learning [42].

**Implementation Details.** We follow the standard practices in the CZSL literature [24, 28], by using a ResNet18 [11] feature extractor. For the state and object classifiers, we follow [28] and use Multi-Layer Perceptrons with three layers, comprising Layer Normalization [3] and Dropout [41]. The model is optimized using Adam [16] with the default hyperparameters, a learning rate and weight-decay of 5e-5.

### 4.1. Open-World CZSL

The results on the challenging OW-CZSL setting are reported in Table 1. In this setting KG-SP either outperforms or is competitive with the state of the art. Specifically, on UT-Zappos KG-SP outperforms the best competitor (CGE) in all metrics, with an improvement of 3.4% in AUC (26.5 vs 23.1), 3.3% in best HM (42.3 vs 39.0) and 4.4% best unseen (52.1 vs 47.7). Similarly, without end-to-end training, KG-SP surpasses the best baseline (CGE) by 1.3 in AUC (22.9 vs 21.5) and by 1.3 in best HM (39.1 vs 38.0).

These results are confirmed in the challenging C-GQA dataset. Despite an output space of almost 250k compositions, KG-SP obtains 0.78 AUC vs 0.47 of the best competitor (CGE), 4.7 HM (vs 3.3 of SymNet) and 2.9 on best unseen (vs 2.2 of SymNet). When non-finetuned, the model achieves competitive results w.r.t. SymNet (e.g. 3.4 HM) while being much easier to optimize, since KG-SP does not impose any constraint on the compositional space. Our

---

Table 1. **Open World CZSL results** on MIT-States, UT Zappos and C-GQA. We measure best seen (S) and unseen accuracy (U), best harmonic mean (HM), and area under the curve (AUC) on the compositions. KG-SPff refers to our proposed method with a frozen backbone.

<table>
<thead>
<tr>
<th>Method</th>
<th>MIT-States</th>
<th>UT Zappos</th>
<th>C-GQA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>S</td>
<td>U</td>
<td>HM</td>
</tr>
<tr>
<td>TMN [31]</td>
<td>12.6</td>
<td>0.9</td>
<td>1.2</td>
</tr>
<tr>
<td>AoP [29]</td>
<td>16.6</td>
<td>5.7</td>
<td>4.7</td>
</tr>
<tr>
<td>LE+ [26]</td>
<td>14.2</td>
<td>2.5</td>
<td>2.7</td>
</tr>
<tr>
<td>VisProd [26]</td>
<td>20.9</td>
<td>5.8</td>
<td>5.6</td>
</tr>
<tr>
<td>SymNet [23]</td>
<td>21.4</td>
<td>7.0</td>
<td>5.8</td>
</tr>
<tr>
<td>CompCosCW [24]</td>
<td>25.3</td>
<td>5.5</td>
<td>5.9</td>
</tr>
<tr>
<td>CGEff [28]</td>
<td>29.6</td>
<td>4.0</td>
<td>4.9</td>
</tr>
<tr>
<td>CompCos [24]</td>
<td>25.4</td>
<td>10.0</td>
<td>8.9</td>
</tr>
<tr>
<td>CGE [28]</td>
<td>32.4</td>
<td>5.1</td>
<td>6.0</td>
</tr>
<tr>
<td>KG-SP</td>
<td>23.4</td>
<td>7.0</td>
<td>6.7</td>
</tr>
<tr>
<td>KG-SPff</td>
<td>28.4</td>
<td>7.5</td>
<td>7.4</td>
</tr>
</tbody>
</table>

---

2We refer to the updated split in https://github.com/ExplainableML/czsl.
<table>
<thead>
<tr>
<th>Marginaliz.</th>
<th>Seen</th>
<th>Unseen</th>
<th>HM</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGE\textsubscript{eff}</td>
<td>✓</td>
<td>25.5</td>
<td>5.7</td>
<td>6.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>24.0</td>
<td>7.8</td>
<td>8.1</td>
</tr>
<tr>
<td>CGE</td>
<td>✓</td>
<td>27.2</td>
<td>6.6</td>
<td>7.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>25.1</td>
<td>8.1</td>
<td>8.1</td>
</tr>
</tbody>
</table>

Table 2. OW-CZSL results in the validation set of MIT-States when using marginalization. CGE\textsubscript{eff} is the approach of [28] with frozen backbone whereas CGE performs end-to-end training.

![Figure 3. Ablation study on the importance of the depth of the object and state-classifiers for KG-SP on UT-Zappos (red curve) and MIT-States (blue curve) validation set for the OW-CZSL setting. Performance is measured in AUC.](image)

Table 3. OW-CZSL results on MIT-states validation set while applying our feasibility-based masks ($f_{KG}$) on different models.

<table>
<thead>
<tr>
<th>Mask</th>
<th>Seen</th>
<th>Unseen</th>
<th>HM</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>VisProd</td>
<td>✓</td>
<td>24.8</td>
<td>6.8</td>
<td>7.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>24.7</td>
<td>7.2</td>
<td>7.6</td>
</tr>
<tr>
<td>KG-SP\textsubscript{eff}</td>
<td>✓</td>
<td>26.3</td>
<td>7.4</td>
<td>7.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td>26.5</td>
<td>7.7</td>
<td>8.2</td>
</tr>
</tbody>
</table>

The results indicate that modeling states and objects independently may be an effective approach to deal with the very large output space of OW-CZSL. This independence assumption ensures that each predictor learns a discriminative classifier over a few hundred classes rather than a single classifier over thousands of compositions, which does not scale well even with powerful architectures (e.g., graph-convolutional neural network of [28]) and initialization through side-information (e.g., word-embeddings [25]).

Finally, the table also highlights the gap between KG-SP and the VisProd baseline of [26]. In particular, our revised model (without fine-tuning) consistently surpasses VisProd in AUC (i.e., 1.0 vs 0.7 on MIT-States, 22.8 vs 19.7 in Zappos, 0.44 vs 0.33 in C-GQA) and best harmonic mean (6.7 vs 5.6 on MIT-States, 39.3 vs 36.9 on Zappos and 3.4 vs 2.8 on C-GQA). These results confirm the importance of our modifications to the original VisProd model, as we will ablate in the following.

### 4.1.1 Why KG-SP works in OW-CZSL?

#### Separately predicting objects and states.

We argue that an important reason for the success of KG-SP is the separate treatment of states and objects. As stated in previous discussion, predicting state and objects independently makes the OW-CZSL problem easier and more scalable w.r.t. predicting directly over thousands of compositions. To verify this hypothesis, we take the state-of-the-art method in standard CZSL, CGE [28] (with and without end-to-end training) and we modify its classifier in such a way that it can output separate objects and states. Specifically, we take the state predictions by marginalizing their scores across all possible objects and, similarly, we marginalize object predictions over the set of possible states. Results are reported in Table 2. We see a consistent increase in best unseen accuracy (5.7 vs 7.8 for CGE\textsubscript{eff}, 6.6 vs 8.1 for CGE) and in the best HM (6.5 vs 8.1 for CGE\textsubscript{eff}, 7.0 vs 8.1 for CGE) when we separate the two predictions. As a consequence, the methods also improve in AUC (1.1 to 1.2 CGE\textsubscript{eff}, 1.3 to 1.4 for CGE). This indicates how providing a separate ground for objects and states predictions is a useful strategy in the open-world setting. Operating in the primitives rather than the compositional space, provides a simplification of the problem that can improve the performance even of existing state-of-the-art CZSL models.

**Effect of depth of the classifier on KG-SP.** We ablate the impact of the depth in Fig 3 for KG-SP\textsubscript{eff} on both MIT-States and UT-Zappos validation sets. The validation AUC on both UT-Zappos (red curve) and MIT-States (blue curve) rapidly increases with the depth of the classifier. This shows the importance of taking into account the multi-task nature of the problem and instantiate objects and state classifiers that have enough capacity to extract primitive-specific features. While deeper predictors help, after 3 layers the performance degrades (i.e., going from 26.9 to 24.3 on UT-Zappos when depth is increased from 3 to 5 layers). The reason behind this drop is mainly linked to overfitting on seen compositions.

**Effect of the knowledge-based masking.** We ablate the impact of masking out unfeasible compositions from the output space in Table 3 for MIT-states validation set. We experiment with both VisProd and KG-SP\textsubscript{eff}. As the table shows, KG-SP\textsubscript{eff} benefits from removing unfeasible compositions, with the unseen accuracy going from 7.3 to 7.5 and the best HM from 7.6 to 8.1. Similarly, also for VisProd we see a consistent improvement for both the unseen accuracy (6.8 to 7.2) and the best HM (7.3 to 7.6) when its output space is filtered. These results show how removing unfeasible compositions from the output space benefit the performance of OW-CZSL models and that ConceptNet is a valuable source for estimating feasibility scores of the compositions.

### 4.2. CZSL under Partial supervision

The results on our proposed pCZSL setting are reported in Table 4. In addition to recognizing unseen compositions in an extremely large compositional space, in pCZSL the model has to cope with the lack of compositional labels. In this scenario KG-SP achieves state-of-the-art results on all the three datasets. On UT-Zappos, KG-SP achieves a HM of 13.1 vs 10.7 of the best competitor (CGE). Similarly, on
Table 5. Partial Open-World CZSL results on MIT-States, UT Zappos and C-GQA. We measure seen (S) and unseen accuracy (U) on the compositions and their harmonic mean (HM). KG-SP refers to our full model with our knowledge-guided pseudo-labeling and inference. CGE_ff and KG-SP_ff denotes the non-finetuned version of the methods. For each CZSL baseline, we show the results of the original methods and the same coupled with Entropy-Minimization (Entropy Min.) [10] or Pseudo-labeling (Pseudo-Lab.) [20].

<table>
<thead>
<tr>
<th>Method</th>
<th>MIT-States</th>
<th>UT Zappos</th>
<th>C-GQA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>S</td>
<td>U</td>
<td>HM</td>
</tr>
<tr>
<td>CGE_ff [28]</td>
<td>19.6</td>
<td>1.3</td>
<td>2.4</td>
</tr>
<tr>
<td>+Pseudo-Lab.</td>
<td>19.7</td>
<td>0.9</td>
<td>1.8</td>
</tr>
<tr>
<td>+Entropy Min.</td>
<td>15.1</td>
<td>1.7</td>
<td>3.1</td>
</tr>
<tr>
<td>CompCos [24]</td>
<td>10.8</td>
<td>2.0</td>
<td>3.6</td>
</tr>
<tr>
<td>+Pseudo-Lab.</td>
<td>9.2</td>
<td>1.9</td>
<td>3.2</td>
</tr>
<tr>
<td>+Entropy Min.</td>
<td>13.2</td>
<td>2.1</td>
<td>3.7</td>
</tr>
<tr>
<td>CGE [28]</td>
<td>17.9</td>
<td>1.6</td>
<td>3.0</td>
</tr>
<tr>
<td>+Pseudo-Lab.</td>
<td>10.6</td>
<td>2.3</td>
<td>3.8</td>
</tr>
<tr>
<td>+Entropy Min.</td>
<td>17.8</td>
<td>1.6</td>
<td>3.0</td>
</tr>
<tr>
<td>KG-SP_ff</td>
<td>13.5</td>
<td>2.6</td>
<td>4.4</td>
</tr>
<tr>
<td>KG-SP</td>
<td>18.4</td>
<td>2.2</td>
<td>4.0</td>
</tr>
</tbody>
</table>

Table 5. Partial Open-World CZSL results on MIT-States validation set for different methods in terms of Seen, and Unseen accuracy and their Harmonic Mean (HM). KG-SP refers to our full model with our knowledge-guided pseudo-labeling and inference. CGE_ff and KG-SP_ff denotes the non-finetuned version of the methods. KG-SP, sampling the pseudo-label using Gumbel-Softmax, and using ConceptNet to filter unfeasible labels are added one after another.

MIT-States and C-GQA, the results obtained by KG-SP exceed the performance of the competitors, with KG-SP achieving 4.4 HM vs 3.7 of CompCos with entropy minimization on MIT-States, and 2.3 (1.7 when non-end-to-end trained) HM on C-GQA vs 1.8 of the best competitor (CGE). It is interesting to note how, even incorporating semi-supervised learning techniques in CGE and CompCos does not bridge the gap between KG-SP and these methods. Interestingly, semi-supervised learning techniques do not bring consistent improvements across CZSL models and tasks. Entropy minimization, by pushing the method toward confident predictions, improves CGE (1.8 vs 1.4 AUC) and CompCos (1.1 vs 0.7) on C-GQA, and improves CGE_ff on MIT-States (3.1 vs 2.4 AUC). However, in the other settings either achieves the same performances of the baseline (e.g. CGE on MIT-States) or degrades them (e.g. CGE on UT-Zappos, 8.7 vs 10.7 AUC). On the other hand, pseudo-labeling degrades the performance of the CZSL models in all settings (e.g. from 3.6 to 3.2 AUC of CompCos on MIT-States), providing advantages only for CGE on MIT-States (3.8 vs 3.0 AUC). The low efficacy of standard semi-supervised learning techniques, is due to the fact that the top predicted object/state may form an unfeasible composition with the ground-truth state/object. Without modeling the feasibility of state-object compositions, pseudo-labeling may assign incorrect (and unfeasible) labels while entropy minimization may push the model toward incorrect predictions. In the following section, we discuss why this happens and why it is important to restore to our knowledge-aided pseudo-labeling in pCZSL.

4.2.1 Ablation Study

We ablate the two important components of our proposed approach: the pseudo-labeling strategy and the quality of the ConceptNet feasibility scores.

Effect of the pseudo-labeling strategy. In Table 5, we consider a few alternatives to our proposed pseudo-labeling strategy for KG-SP. The first option is to just introduce vanilla pseudo-labeling. This is done by replacing the missing labels with the top predicted class, when the confidence of the model is greater than a threshold. This strategy has a negative effect, reducing the accuracy of the model on both seen and unseen compositions (i.e. 16.6 vs 15.9 on the seen, 2.8 vs 2.7 on the unseen). This is because pseudo-labeling alone is prone to confirmation bias [1] since the model can become increasingly more confident about the predictions that, without modeling the feasibility of compositions, may be not only incorrect, but also unfeasible.

The second alternative is to use Gumbel-softmax [14], sampling a label instead of performing direct pseudo-labeling. This approach may allow the model to overcome the confirmation bias, especially in the initial training stages by sampling different labels w.r.t. the top prediction. However, this strategy alone does not achieve good results, degrading the performance of the base model (e.g. 4.5 vs 4.8 HM) and performing slightly worse than standard pseudo-labeling (4.5 vs 4.6 HM). This performance degradation is due in both cases by not exploiting the external knowledge to correct the pseudo-labeling process, avoiding using
When inspecting the most and least feasible compositions for random objects (Table 6, bottom), we find that the compositions ranked as most feasible such as bright tulip, caramelized sugar, and blunt sword are indeed feasible in the reality, while the compositions marked as least feasible are not. Interestingly, the unfeasible compositions merge objects and states from different categories. Examples are food vs tools (e.g. pureed chains, scratched sugar) and weather vs objects (e.g. cloudy laptop). This suggests that the ConceptNet-based feasibility scores encode a high-level notion of grouping, where objects and states can be separated depending on the contexts where they commonly occur.

### Table 6. Examples of ConceptNet feasibility scores. Top: Top-5 (left) and Least-5 (right) compositions per feasibility; Bottom: Top-3 highest and Bottom-3 lowest feasible states per random objects.

<table>
<thead>
<tr>
<th>Compositions</th>
<th>Most Feasible (Top-5)</th>
<th>Least Feasible (Bottom-5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>painted paint</td>
<td>grimy balloon</td>
<td></td>
</tr>
<tr>
<td>muddy mud</td>
<td>steaming bracelet</td>
<td></td>
</tr>
<tr>
<td>frozen ice</td>
<td>blunt clock</td>
<td></td>
</tr>
<tr>
<td>mossy moss</td>
<td>thin garage</td>
<td></td>
</tr>
<tr>
<td>cloudy cloud</td>
<td>unpainted belt</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Objects States</th>
<th>Most Feasible (Top-3)</th>
<th>Least Feasible (Bottom-3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>chains</td>
<td>frayed, broken, loose</td>
<td></td>
</tr>
<tr>
<td>sugar</td>
<td>melted, whipped, caramelized</td>
<td></td>
</tr>
<tr>
<td>sword</td>
<td>blunt, sharp, splintered</td>
<td></td>
</tr>
<tr>
<td>laptop</td>
<td>small, shattered, modern</td>
<td></td>
</tr>
<tr>
<td>tulip</td>
<td>bright, wilted, ruffled</td>
<td></td>
</tr>
</tbody>
</table>

In this work, we addressed the problem of Open-World Compositional Zero-Shot learning (OW-CZSL), where the goal is to recognize compositions of objects and states in images given only a subset of them during training and without any prior on unseen compositions at test time. We address the problem by revisiting a simple CZSL method, VisProd, that independently predicts state and object labels. The idea is to simplify the problem by exploiting the much smaller cardinality of object and state sets w.r.t. the compositional labels. Our model, KG-SP, uses two different feature extractors to account for the dissimilarity between the two tasks, and uses external information to remove less feasible compositions from the output space at test time. As KG-SP does not require compositional labels, we explore a new challenging setting, CZSL under partial supervision (pCZSL) where training images have either only object or state annotation. In pCZSL, we use the feasibility scores to aid the estimation of the missing labels. Experiments show that KG-SP achieves the state of the art in OW-CZSL and pCZSL, outperforming recent CZSL approaches coupled with standard semi-supervised learning techniques.

**Limitations and Broader Impact.** One weakness of our approach, shared with CZSL methods, is that the absolute performance on all OW-CZSL benchmarks is quite low (e.g. 2.9 unseen accuracy on C-GQA). This can significantly hamper the real-world deployment of these models. However, we believe that this research topic is crucial to bridge the gap between machine and human visual compositional recognition abilities: our work contributes to the field by making another step toward this direction. Another limitation of our work is that automatic retrieving feasible state-object compositions is a process without expert supervision, thus vulnerable to inaccuracies in the knowledge base. For instance, if a valid composition is marked as unfeasible, it will not be predicted during inference and removed from candidate pseudo-labels during pCZSL training. This may lead to the model producing erroneous outcomes, even reflecting potential biases in the knowledge base. Modeling issues in the knowledge base and/or merging multiple external sources is an important topic for future research in OW-CZSL and pCZSL.

**Acknowledgments** This work has been partially funded by the ERC (853489 - DEXIM), by the DFG (2064/1 – Project number 390727645), and as part of the Excellence Strategy of the German Federal and State Governments. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Shyamgopal Karthik.
References

[26] Ishan Misra, Abhinav Gupta, and Martial Hebert. From red wine to red tomato: Composition with context. In CVPR, 2017. 1, 2, 3, 5, 6


