CURI: A Benchmark for Productive Concept Learning under Uncertainty

Ramakrishna Vedantam [1] **Arthur Szlam** [1] **Maximilian Nickel** [1] **Ari Morcos** [1] **Brenden Lake** [1 2]

Abstract

Humans can learn and reason under substantial uncertainty in a space of infinitely many compositional, productive concepts. For example, if a scene with two blue spheres qualifies as "daxy," one can reason that the underlying concept may require scenes to have "only blue spheres" or "only spheres" or "only two objects." In contrast, standard benchmarks for compositional reasoning do not explicitly capture a notion of reasoning under uncertainty or evaluate compositional concept acquisition. We introduce a new benchmark, Compositional Reasoning Under Uncertainty (CURI) that instantiates a series of few-shot, meta-learning tasks in a productive concept space to evaluate different aspects of systematic generalization under uncertainty, including splits that test abstract understandings of disentangling, productive generalization, learning boolean operations, variable binding, etc. Importantly, we also contribute a model-independent "compositionality gap" to evaluate the difficulty of generalizing out-of-distribution along each of these axes, allowing objective comparison of the difficulty of each compositional split. Evaluations across a range of modeling choices and splits reveal substantial room for improvement on the proposed benchmark.

1. Introduction

Human concept learning is more flexible than today's AI systems. Human conceptual knowledge is *productive*: people can understand and generate novel concepts via compositions of existing concepts ("an apartment dog") (Murphy, 2002), unlike standard machine classifiers that are limited to a fixed set of classes ("dog", "cat", etc.). Further, humans can induce goal-based, "ad hoc" categories such as "things to take from one's apartment in a fire" (children, dogs, keepsakes, etc.) (Barsalou, 1983). Thus, unlike AI systems, humans reason seamlessly in large, essentially "unbounded" concept spaces.

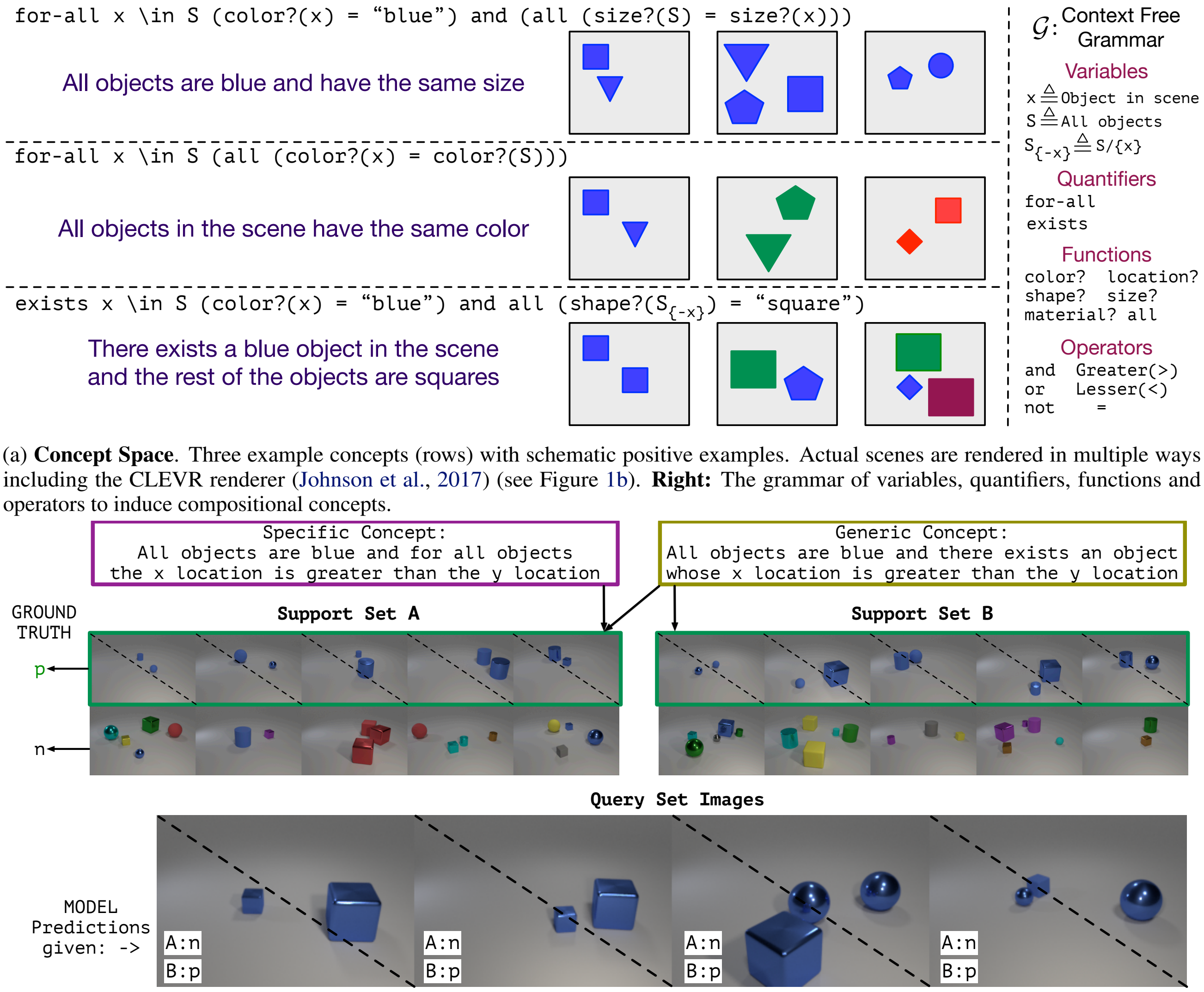
While popular compositional reasoning benchmarks such as CLEVR (Johnson et al., 2017) for visual question answering and Ravens Progressive Matrices (PGM) (Barrett et al., 2018) appear to have large, unbounded concept spaces, the tasks they instantiate miss a key feature of human concept learning, namely our ability to deal with *uncertainty*. For example, consider the set of images labelled "A" (green border) in Figure 1b. Given this set, the underlying concept is ambiguous, and could either be "All objects are blue and the x-coordinate of all objects is greater than the y-coordinate" or "All objects are blue and there exists an object whose x-coordinate is greater than the y-coordinate". As humans, we are able to conceive of these alternatives and make predictions that place more weight on the former, more specific concept (Figure 1b, bottom) (Tenenbaum & Griffiths, 2001; Xu & Tenenbaum, 2007; Goodman et al., 2008; Piantadosi et al., 2016). In contrast, neither CLEVR nor PGM requires reasoning about alternative, equally-consistent concepts and their relative generality when making predictions.

We address this gap in the literature by proposing the Compositional Reasoning Under Uncertainty (CURI) benchmark to study how modern machine learning systems can learn concepts spanning a large, productively defined space (Figure 1a) while dealing with the entailed uncertainty. In pursuit of this goal, we instantiate a meta-learning task where a model must reason in a compositional space about alternative concepts, and make predictions taking such uncertainty into account. A signature of productivity in human thought is our ability to handle novel combinations of known, atomic components. Thus, in CURI we instantiate a variety of different train-test splits that involve novel combinations of intrinsic properties (*e.g.* color, shape) with boolean operators, counting, disentangling (Higgins et al., 2018) of extrinsic (*e.g.* object location) and intrinsic (*e.g.* object material) object properties, and a novel test of variable binding in the context of compositional learning.

Compositional Reasoning Under Uncertainty (CURI) Task. Concretely, the CURI task tests few-shot learning of relational concepts in a large compositional conceptual

[1] Facebook AI Research (FAIR), USA [2] New York University (NYU), USA. Correspondence to: Ramakrishna Vedantam <ramav@fb.com>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).


for-all x \in S (color?(x) = “blue”) and (all (size?(S) = size?(x)))
All objects are blue and have the same size
for-all x \in S (all (color?(x) = color?(S)))
All objects in the scene have the same color
exists x \in S (color?(x) = “blue”) and all (shape?($S_{\{-x\}}$) = “square”)
There exists a blue object in the scene and the rest of the objects are squares
$\mathcal{G}$: Context Free Grammar
Variables
x ≜ Object in scene
S ≜ All objects
$S_{\{-x\}}$ ≜ S/{x}
Quantifiers
for-all
exists
Functions
color? location?
shape? size?
material? all
Operators
and Greater(>)
or Lesser(<)
not =
Specific Concept: All objects are blue and for all objects the x location is greater than the y location
Generic Concept: All objects are blue and there exists an object whose x location is greater than the y location
GROUND TRUTH
Support Set A
Support Set B
p
n
Query Set Images
MODEL Predictions given: ->
A:n
B:p
A:n
B:p
A:n
B:p
A:n
B:p

(a) **Concept Space**. Three example concepts (rows) with schematic positive examples. Actual scenes are rendered in multiple ways including the CLEVR renderer (Johnson et al., 2017) (see Figure 1b). **Right:** The grammar of variables, quantifiers, functions and operators to induce compositional concepts.

(b) **Illustration of reasoning under uncertainty.** Given a support set D_{supp} corresponding to a more specific concept (purple, left), such that D_{supp} is also consistent with a more generic concept (gold), a fully-successful CURI model would make predictions on held out images (bottom) that place more weight on the specific concept (producing a negative label n). In contrast, given the support set for the more generic concept (gold, right), the model would make the opposite predictions (positive p). Actual predictions shown are from a relation network trained with symbolic, schema-based scene representations.

Types

`P, Q`	BOOL
`A, B`	STR
`M, N`	INT or FLOAT
`SET[•]`	A set of objects of any Type

Variables

`x`	Denotes an object in a scene
`S`	Denotes the SET of all objects in scene
$S_{\{-x\}}$	Denotes the SET of all objects except x

Constants (Illustrated for Images)

Counts	*1, 2, 3*
Materials	*Rubber, Metal*
Shapes	*Cube, Sphere, Cylinder*
Sizes	*0.35, 0.70*
X or Y location	*1, 2, 3, 4, 5, 6, 7, 8* (numbering starts from top-left of image)

Functions on Objects or Sets of Objects*

These compute property? (x). Illustration of properties in image domain:

`size?(x)`	Returns M, size of object x
`material?(x)`	Returns A, material of object x|
`shape?(x)`	Returns A, shape of object x
`locationX? (x)`	Returns INT M, x-coordinate of center of object x
`locationY? (x)`	Returns INT M, y-coordinate of center of object x
`color? (x)`	Returns A, color of object x

*All these operations also apply when the argument is a SET.
Example `color?(S)` Returns `{color?(x): x in S}`

Quantifiers

`for-all x in S`	Returns TRUE if condition holds for all x
`exists x in S`	Returns TRUE if condition holds for any x

Logical Functions and Functions on BOOL Sets

`P and Q`	Returns TRUE if P and Q are both true
`P or Q`	Returns TRUE if either P or Q is true
`not P`	Returns TRUE iff P is False
`all SET[P]`	Returns TRUE if all elements of P are TRUE
`any SET[P]`	Returns TRUE if all elements of P are TRUE
`count SET[P]`	Returns INT, number of elements of P which are TRUE

Comparison Operations*

`M = N`	Returns TRUE if M and N are equal
`A = B`	Returns TRUE if A and B are equal
`M > N`	Returns TRUE if M is greater than N
`M < N`	Returns TRUE if M is lesser than M

*All these operations also apply when one of the arguments is a `SET`.
Example `SET[M] = N` Returns `{M = N: M in SET[M]}`

(c) **Language of thought.** All valid (type-consistent) compositions of functions are potential complex concepts in our dataset. Note that the functions are illustrated for the case of images and schemas. Location, size, shape *etc.* correspond to different properties for sounds.

Figure 1. Elements of the Compositional Reasoning Under Uncertainty (CURI) task.

space, with design inspiration from studies in cognitive modeling using a language of thought (LOT) approach (Fodor, 1975; Piantadosi, 2011; Kemp et al., 2005). CURI includes scene-based concepts such as "All objects have the same color" and "There exists a blue object while the rest are triangles" (Figure 1a) but unlike CLEVR (Johnson et al., 2017) there are too few examples to deduce answers with certainty. Our benchmark is defined through a series of meta-learning episodes (see example in Figure 1b): given positive and negative examples of a new concept D_{supp} (known as the "support set"), the goal of an episode is to classify new examples D_{query} (the "query set"). While typical benchmarks for meta-learning and few-shot classification (Vinyals et al., 2016; Lake et al., 2019) evaluate new atomic concepts at test time, we consider compositionally novel concepts and explicitly study reasoning under uncertainty—an ideal learner must marginalize over many hypotheses when making predictions (Gelman et al., 2004; Xu & Tenenbaum, 2007; Piantadosi et al., 2016). Consequently, we focus on the structure in the concept space and compare different ways to reason in it, rather than on the mechanics of meta-learning (comparing and benchmarking different meta-learning algorithms in terms of the number of shots, lengths of episodes etc.) which in our opinion are better addressed by benchmarks such as the meta-dataset (Triantafillou et al., 2020).

We also vary the modality in which scenes are presented—rendering them as images, symbolic schemas, and sounds—enabling future research on modality-specific representational choices for compositional reasoning under uncertainty. Finally, we vary the concepts learned by the model during meta-training and meta-testing to test different aspects of systematic generalization.

Compositionality Gap. While systematic splits are increasingly common in the literature (Barrett et al., 2018; Hill et al., 2019; Agrawal et al., 2018; Vedantam et al., 2018; Higgins et al., 2018; Bakhtin et al., 2019; Lake & Baroni, 2018; Ruis et al., 2020), previous work has often lacked an objective way to assess how difficult each split is, or to provide a baseline for how well a model needs to perform for it to be "compositional" in nature. We address this issue by introducing the notion of a model-independent "compositionality gap"—defined as the difference in test performance between an ideal Bayesian learner with access to the full hypothesis space, and a Bayesian learner with access to only a (potentially large) list of the hypotheses examined during meta-training. A large gap for a given split indicates that any learner must extrapolate compositionally from the training hypotheses to solve the task; additionally, models can be compared to ideal learners that either do or do not engage in such extrapolation which provides a baseline for compositionality. We anticipate this tool to be useful for analyzing other benchmarks with compositional splits.

Table 1. Comparison of CURI to other compositional and meta-learning benchmarks: CLEVR (Johnson et al., 2017), PGM (Barrett et al., 2018), Meta-Dataset (Triantafillou et al., 2020), SCAN (Lake & Baroni, 2018).

	Compositional	Uncertainty	Few-Shot
CLEVR	✓	✗	✗
PGM	✓	✗	✗
Meta-Dataset	✗	✗	✓
SCAN	✓	✗	✗
CURI (Ours)	✓	✓	✓

Models. We evaluate models around various dimensions which concern the difficulty of learning productive concepts under uncertainty, including: 1) the modality in which the input is rendered (image, schemas, sounds), 2) method used for reasoning across objects in a scene (transformer, relation-network, global average pooling, concatenation), 3) whether or not training provides ground-truth symbolic descriptions of concepts, and 4) how negative examples are sampled. Overall, our evaluations suggest that there is substantial room for improvement in compositional reasoning under uncertainty, w.r.t the compositionality gap, representing a novel challenge for compositional learning.

Summary of contributions:

1. We introduce the Compositional Reasoning Under Uncertainty (CURI) benchmark for evaluating compositional, relational learning under uncertainty from observational data;
2. We introduce a 'compositionality gap' metric for measuring the difficulty of systematic generalization from train to test;
3. We provide various baseline models for benchmarking progess on the proposed task.

2. Related Work

Compositional Reasoning and Meta-Learning Benchmarks. Table 1 compares CURI to other compositional reasoning and meta-learning benchmarks. In contrast to CLEVR and PGM, CURI requires reasoning about uncertainty (see Section 1 for more details). Different from benchmarks for few-shot meta-learning, the concepts we use are compositional, and explicitly evaluate reasoning under uncertainty. Also, while there exist sequence to sequence compositionality benchmarks like SCAN (Lake & Baroni, 2018) and meta-learning based approaches for the task such as Lake (2019) these do not tackle few-shot meta-learning or explicitly reason about uncertainty in the concept space.

In other related work, Keysers et al. (2020) proposed a benchmark and a method to create "difficult" systematic splits based on the principle that they should share atoms but have maximally different compositions. This is complementary to our splits, which provide interpretable notions of what each split tests such as disentangling, complexity, and variable binding. Finally, Andreas et al. (2018) studied few-shot learning with concepts defined by language-like structured descriptions. Their models were trained to induce novel classifiers via the space of linguistic expressions, while we don't necessarily commit to such a choice (although we compare with language-augmented models as well).

Language of Thought (LOT). Our choice of compositional concepts was most closely inspired by Piantadosi et al. (2016) along with other studies of human concept learning in the Language of Thought (LOT) framework (Fodor, 1975; Goodman et al., 2008; Kemp & Jern, 2009; Piantadosi et al., 2012; Goodman et al., 2015; Overlan et al., 2017; Lake & Piantadosi, 2019). In typical LOT studies of human learning, the conceptual space $\mathcal{H}$ is defined through a probabilistic context-free grammar $\mathcal{G}$, which specifies a set of conceptual primitives and their rules of combination. Our work is similar, in that we use an LOT-inspired grammar $\mathcal{G}$ to generate an unbounded set of concepts $\mathcal{H}$, but our goals are different – we want to evaluate if machine learning models can "acquire" a language of thought when trained without access to the underlying LOT, while traditional studies in cognitive science utilize it as a tool for modeling human learning.

3. Compositional Reasoning Under Uncertainty (CURI) Dataset

Concept space. The compositional concepts in CURI were inspired by the empirical and cognitive modeling work of Piantadosi et al. (2016). The space of concepts (LOT) is defined by a context free grammar ($\mathcal{G}$). Figure 1c shows the LOT and specifies how primitives and functions compose to produce a large unbounded concept space (see Appendix 2.1 for a full description of the underlying LOT/grammar.). The LOT has three variables: $\mathbf{x}$, representing an object in a scene, $S = \{\mathbf{x}\}_{i=1}^{N}$ representing the set of all objects in the scene, and $S_{-\mathbf{x}} = S/\{\mathbf{x}\}$, representing the set of all objects in the scene *except* $\mathbf{x}$. Each concept describes a rule composed of object and scene properties, logical operators, and/or comparison operators, and can be evaluated on a given scene S to determine whether the scene satisfies the rule.

Object and scene properties are defined by functions which can be applied to objects or scenes: for example, $\texttt{size?}(\mathbf{x})$ yields the size of an object $\mathbf{x}$, while $\texttt{size?}(S)$ returns a set with the sizes of all the objects ($\{\texttt{size?}(\mathbf{x}) : \mathbf{x} \in S\}$). Comparison and logical operators can be used to compare and relate various properties of objects in scenes. In contrast to Piantadosi et al. (2016), we include a `count` operator, which determines how many times a condition is satisfied by a set, which allows us to check how well deep learning models are able to count (Chattopadhyay et al., 2017; Agrawal et al., 2018). Finally, quantifiers such as `exists` and `for-all` enrich the LOT by allowing concepts to specify how many objects should satisfy a given condition.

Consider the following example concept (Figure 1a bottom): "There exists a blue object in the scene and the rest of the objects are squares." To access the color of a given object, we use $\texttt{color?}(\mathbf{x})$ and to access the shape of a given object, we use $\texttt{shape?}(\mathbf{x})$. To determine whether an object matches a specific property, we can combine this with equality: $\texttt{shape?}(\mathbf{x})$ = ``square''. Finally, we can use `exists` to specify that at least one object must be blue, $S_{-\mathbf{x}}$ to specify all the objects except for that blue object, and `all` to specify that all the objects in $S_{-\mathbf{x}}$ must be squares. Putting it all together: `exists` $\mathbf{x} \in S$ ($\texttt{color?}(\mathbf{x})$ = ``blue'') `and` `all` ($\texttt{shape?}(S_{-\mathbf{x}})$ = ``square'').

Structured Generalization Splits. A signature of productivity is the ability to handle novel combinations of known components (Fodor, 1975; Fodor & Pylyshyn, 1988). Thus, in CURI, we consider splits that require generalizing to novel combinations of known elements from our LOT (Figure 1c), including combinations of constants, variables, and functions. We achieve this by creating disjoint splits of concepts $\mathcal{H}_{train}$ and $\mathcal{H}_{test}$ for training and evaluating models. By varying the held out elements and their combinations, we obtain splits that evaluate different axes of generalization. In practice, we use our grammar $\mathcal{G}$ to sample and filter a large set of concepts (see appendix for more details), which yields a set of 14,929 concepts $\mathcal{H}$ for training and evaluation (see Appendix 2.2 for more details). We next describe how each split divides $\mathcal{H}$ into $\mathcal{H}_{train}$ and $\mathcal{H}_{test}$, to test productive, out of distribution generalization (Appendix 3 contains more details):

- **Instance IID**: Evaluates generalization to novel episodes from the same concept set. This is the standard setup in machine learning (Murphy, 2013), in which $\mathcal{H}_{train} = \mathcal{H}_{test}$. This is the only split where train and test concepts overlap.
- **Concept IID**: Evaluates generalization to novel concepts based on an arbitrary random split of the concepts into $\mathcal{H}_{train}$ and $\mathcal{H}_{test}$.[1]
- **Counting**: Evaluates the ability to learn a new concept h with novel property-count combinations, e.g, the training concepts never filter for exactly '3 squares'.

[1] While some strings h might be different in surface form, they may yeild the same results when applied to images. In this split we account for such synonomy, and ensure that no two concepts which are synonyms are in different splits. See Appendix 2.6 for more details.

- **Extrinsic properties**: Evaluates the ability to learn a new concept h, with novel combinations of extrinsic (e.g. location) and intrinsic (e.g. color) object properties.
- **Intrinsic properties**: Evaluates the ability to learn a new concept h with novel combinations of intrinsic properties, e.g., the training concepts never reference both 'red' and 'rubber'.
- **Boolean operations**: Evaluates the ability to learn concepts which require application of a familiar boolean operation to a property to which the operation has never been applied previously.
- **Complexity split**: Evaluates generalization from simple concepts (those which have less than or equal to 10 symbols) to more complex concepts (longer than 10 symbols). This is indicative of the productivity (Fodor, 1975) exhibited by models, in generalizing from simpler concepts to more complex concepts.
- **Variable binding**: Evaluates learning of entirely novel intrinsic properties, e.g. the training concepts involve only "red", "blue", and "green" but test concepts involve "yellow" (although 'yellow' objects can still appear in training scenes). This is indicative of inferential coherence (Fodor, 1975) in models, in generalizing rules of inference to novel atoms.

A model that infers the underlying LOT during meta-training would be expected to perform well on any such systematic split. By comparing the performance of current models to such ideal learners, this benchmark will allow us to evaluate progress on the systematic out-of-distribution generalization capabilities of our current models.

From Concepts to Meta-learning Episodes. A single episode comprises a support set (D_{supp}) and a query set (D_{query}), each of which is generated from a given concept, h. Formally, a support or query set D has input data $\mathbf{u}$ and corresponding label y, *i.e.* $D = \{\{y_i\}_{i=1}^N, \{\mathbf{u}_i\}_{i=1}^N\}$. Each support and query set generally contains 5 positive and 20 negative examples — negative examples are oversampled since the space of negatives is generally much larger than that for positives. The set of positive examples are sampled uniformly from a categorical distribution over all positives. However, we consider two types of negatives: 1) easy negatives, in which the negatives are also sampled at random, and 2) hard negatives, in which negatives are generated from a closely related concept which also evaluates true on the positive examples in D_{supp}, such that these negatives are maximally confusing. Regardless of the method, any inputs we sample to be negatives that incidentally satisfy the target concept h are relabelled as positives. Altogether, for each split, our train, validation, and test sets contain 500000, 5000, and 20000 episodes respectively. See Appendix 4 for a more formal description of the procedure for creating episodes.

Compositionality Gap. A key goal of our work is to define the difficulty in learning that arises from the compositional structure of the concept space. Most of the splits above are structured in a way such that $\mathcal{H}_{\text{test}} \cap \mathcal{H}_{\text{train}} = \emptyset$, which forces a learner to use the compositional structure of the concept space to generalize to $\mathcal{H}_{\text{test}}$. We conceptualize the difficulty of this task through the notion of its *compositionality gap*.

Intuitively, the compositionality gap captures the difference between the generalization performance of two "ideal" or oracle models, which are both are perfect in labelling an input $\mathbf{u}$ with the correct label y for a concept h. However, the key difference is that one is an ideal compositional learner (*strong oracle*) demonstrating perfect ability to extrapolate to test concepts $\mathcal{H}_{\text{test}}$, while the other is an ideal non-compositional learner which is unable to perform any extrapolation (*weak oracle*) beyond $\mathcal{H}_{\text{train}}$.

More formally, let $\Omega \in \{\text{strong}, \text{weak}\}$ denote an oracle over a concept space $\mathcal{H}_\Omega$. The posterior predictive distribution of an oracle for query scene $\mathbf{u}$ and query label $y \in \{p, n\}$ is then given as:

$$p_\Omega(y|\mathbf{u}, D_{\text{supp}}) = \sum_{h \in \mathcal{H}_\Omega} p_\Omega(y|h, \mathbf{u}) p_\Omega(h|D_{\text{supp}}) \qquad (1)$$

where given a prior $p_\Omega(h)$,[2] the posterior $p_\Omega(h|D_{\text{supp}})$ satisfies (applying Bayes rule):

$$p_\Omega(h|D_{\text{supp}}) \propto p_\Omega(h)\, p(\{y_i\}_{i=1}^N | h; \{\mathbf{u}_i\}_{i=1}^N) \qquad (2)$$

We assume that both the strong and weak oracles are able to label inputs $\mathbf{u}$ based on a given hypothesis h perfectly, that is $p_{\text{strong}}(y|h, \mathbf{u}) = p_{\text{weak}}(y|h, \mathbf{u})$. Thus, the only difference between the oracles is in their priors over the hypothesis space. The strong oracle has access to the union of train and test concepts—that is $\mathcal{H}_{\text{strong}} = \mathcal{H}_{\text{train}} \cup \mathcal{H}_{\text{test}}$. In contrast, the weak oracle only has access to $\mathcal{H}_{\text{weak}} = \mathcal{H}_{\text{train}}$, which means it is unable to consider any hypothesis outside what has been seen in training and assigns it zero probability mass. Given a support set D_{supp} this difference in priors leads then to different posterior predictive distributions and allows us to quantify how compositionally novel a learning task is relative to these ideal learners.

Given a metric of interest M (e.g., mean average precision or accuracy), the compositionality gap of a learning task is then simply defined as the difference in performance of the posterior predictive from the strong and weak oracles when evaluating on concepts from $\mathcal{H}_{\text{test}}$, i.e., $M_{\text{strong}} - M_{\text{weak}}$.

[2] See Appendix 2.3 for detailed definition of the prior, which penalizes hypotheses that longer description lengths (Feldman, 2000)

4. Metrics and Baselines

During meta-test, given D_{supp} models are evaluated on their ability to learn novel concepts. We use two metrics for quantifying this: 1) **Accuracy:** evaluates the accuracy of model predictions across the query set D_{query}, as is standard practice in meta-learning (Lake et al., 2019; Snell et al., 2017). Since there are more negative than positive labels, we report class balanced accuracy for better interpretability, averaging accuracies for the positive and negative query examples; and 2) **mean Average Precision (mAP):** evaluates models on a much larger number of test scenes $\mathcal{T}$ for each episode (comprising 44,787 scenes, 3 for each concept in $\mathcal{H}$). This resolves an issue that with a small query set, a strong model could achieve perfect accuracy without grasping the concept. Since episodes typically have many more negative than positive examples, Average Precision sweeps over different thresholds of a model's score and reports the average of the precision values at different recall rates, e.g., (Everingham et al., 2010). mAP is then the mean across all of the meta-test episodes.

4.1. Training Loss

Let $\mathbf{u} \in \mathbb{R}^M$ be the input to the model, which can take the form of an image, sound or schema. We work in a binary classification setting where labels y live in the space $\mathcal{Y} \in \{p, n\}$, (where p stands for positive, and n for negative). Then, given a support set $D_{\text{supp}} = \{\mathbf{u}_i, y_i\}_{i=1}^{T}$ and a query set $D_{\text{query}} = \{\mathbf{u}_i, y_i\}_{i=1}^{T}$, sampled in accordance with a productive concept h, our training objective for a single training instance can be written as:

$$\mathcal{L}_{\text{query}} + \alpha \mathcal{L}_{\text{concept}} \tag{3}$$

Here $\mathcal{L}_{\text{query}} = \sum_{\mathbf{u}, y \in D_{\text{query}}} \log p(Y = y | \mathbf{u}, D_{\text{supp}})$ is a standard maximum likelihood meta-learning loss (Ravi & Larochelle, 2017; Snell et al., 2017; Finn et al., 2017), and $\mathcal{L}_{\text{concept}} = \log p(H = h | D_{\text{supp}})$ is an additional (optional) term that provides the underlying ground-truth hypothesis in string form as a means of strong supervision.

4.2. Baseline Models and Losses

Query Loss ($\mathcal{L}_{\text{query}}$): Our baseline models (shown in Figure 2) parameterize the probability in the $\mathcal{L}_{\text{query}}$ term above using prototypical networks (Snell et al., 2017). The prototypical network consists of an embedding function $f = f_\theta$ and uses it to compute prototypes $\mathbf{c}_p$ and $\mathbf{c}_n$ for positive and negative examples by averaging $f(\mathbf{u})$ for positive and negative examples in the support set respectively. In equations, given a query datapoint $\mathbf{u}'$, we maximize:

$$\log p_\theta(Y = y | \mathbf{u}'; D_{\text{supp}}) = \\ \log \left(\frac{exp(-||f_\theta(\mathbf{u}') - \mathbf{c}_y||^2}{exp(-||f_\theta(\mathbf{u}') - \mathbf{c}_p||^2) + exp(-||f_\theta(\mathbf{u}') - \mathbf{c}_n||^2)} \right) \tag{4}$$

In this formalism, the models we study in this paper span different choices for f. Roughly, in each modality, we start with an encoder that converts the raw input into a set of vectors, and then a pooling operation that converts that set of vectors into a single vector. In the case of images and sound (input as spectrograms), the encoder is a ResNet-18; and the set of vectors is a subsampling of spatial locations; and for schemas we vectorize components with a lookup table and combine them into a set via feed-forward networks. In the case of images and sounds, the output of the encoder is enriched with position vectors. For the pooling operation, we study global averaging, concatenation, relation networks (Santoro et al., 2017) and transformers (Vaswani et al., 2017) equipped with different pooling operations (max, mean, sum, min) for reasoning inspired by Wang et al. (2020) (Figure 2 middle panel, also see Appendix 6).

Concept Loss ($\mathcal{L}_{\text{concept}}$): For the probability in $\mathcal{L}_{\text{concept}}$, we represent the concept h as a sequence $s^h = \{s^h_0, s^h_1, \cdots, ^h_T\}$ by prefix serialization, and D_{supp} with $[\mathbf{c}_p, \mathbf{c}_n]$ (where $[\cdot]$ represents concatenation) and then use an LSTM (Hochreiter & Schmidhuber, 1997) to parameterize $p(s^h | D_{\text{supp}}) = p(s^h_0 | [\mathbf{c}_p, \mathbf{c}_n]) \Pi_{t=1}^{T} p(s^h_t | s^h_0, \cdots s^h_{t-1}; [\mathbf{c}_p, \mathbf{c}_n])$, where at each step of the LSTM we concatenate $[\mathbf{c}_p, \mathbf{c}_n]$ to the input.

5. Experimental Results

We first discuss the compositionality gap induced by the different generalization splits and then delve into the impact of modeling choices on performance on the generalization splits. All models are trained for 1 million steps, and are run with with 3 independent training runs to report standard deviations. We sweep over 3 modalities (image, schema, sound), 4 pooling schemes (avg-pool, concat, relation-net, transformer), 2 choices of negatives (hard negatives, random negatives) and choice of language ($\alpha = 0.0, 1.0$). Unless mentioned otherwise in the main paper we focus on results with hard negatives and $\alpha = 0.0$. When instantiated for a given modality, we note that the encoders $f(\mathbf{u})$ (Figure 2) all have a similar number of parameters (see appendix for more details).

5.1. Dataset Design and Compositionality

How compositional are the structured splits? Our main results are shown in Figure 3. Using our model-independent measure of the compositionality gap (Section 3), different splits present varying challenges for generalizing from train to test. The most difficult splits, with the largest compositionality gaps, are the Binding (color) and Binding (shape),

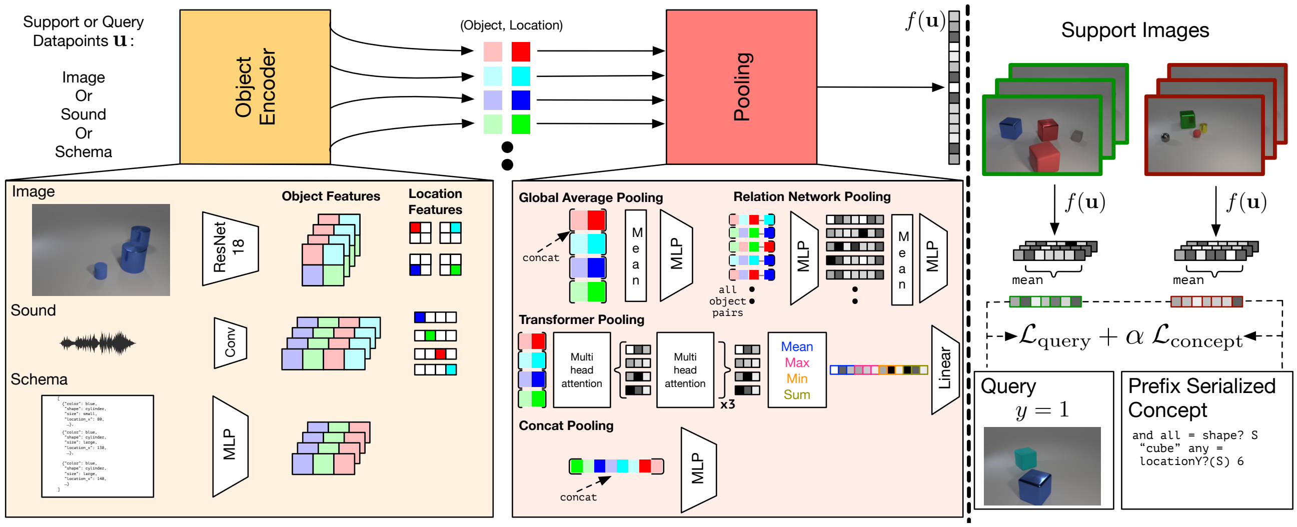


Figure 2. **Baseline models (left).** Different choices for the encoder $f(\mathbf{u})$ parameterization explored in the paper. We consider three modalities each of which is processed with a modality specific encoder, followed by four kinds of pooling architecture which take as input objects and their corresponding locations to provide an encoding for the datapoint. **Training (right).** The model is trained by processing the support images D_{supp} with positive (green) and negative (red) images, using $f(\mathbf{x})$ to compute $\mathcal{L}_{\text{query}}$ which computes generalization error on queries and $\mathcal{L}_{\text{concept}}$ which learns to decode the true concept as an auxiliary task. Losses are weighted by $\alpha \geq 0$.

which is reasonable since they require learning concepts with entirely new property-values. In contrast, the easiest split with the smallest compositionality gaps is the Instance IID split since it does not require compositionality. Finally, while the mAP metric (Section 4) exposes a larger value of the compositionality gap, the ordering of splits in terms of compositionality gap is same for both metrics – suggesting similar coarse-grained notions of compositionality.

Results for the best overall architecture, a relation network (relation-net), are shown in Figure 3. Network performance on the easiest data format (schema; yellow bars) is generally better than the weak oracle, but substantially worse than the strong oracle. Counting is a particularly challenging split where the models underperform even the weak oracle. Broadly, this suggests that the models capture some notion of compositionality—especially for images and schemas—relative to a weak oracle that rigidly considers only training hypotheses, but there is substantial room to improve (especially with respect to the more stringent mAP metric). These results demonstrate that CURI provides a challenging yet tractable setting for evaluating the compositional capabilities of models.

Finally, we found that the performance on the Instance IID split is not equal to the weak (and strong) oracle—which are both equal in this case—indicating that the best model does not make ideal posterior predictions even when compositionality is not an issue. Ideal predictions in this case would require the network to behave as if marginalizing over the training hypotheses, as the strong oracle does.

Influence of Negatives. Previous work (Hill et al., 2019) has shown that the choice of random *vs.* hard negatives for training and evaluation impacts compositional generalization substantially in the case of a particular set of analogical reasoning models. However, we argue that such decisions on dataset design can be made more objectively if one can evaluate the model-independent compositionality gap. In our context, we find that the compositionality gap with mAP when using random negatives decreases on average by 5.5 ± 1.4% compared to when we use hard negatives. This indicates that it is not only the choice of $\mathcal{H}_{train}$ and $\mathcal{H}_{test}$, which are identical for a given compositional split (say Counting), but also the choice of the negatives which “makes” the task compositionally novel. More generally, this indicates that the compositionality gap has utility as a more general diagnostic tool for making principled design decisions in compositional learning settings *without* the confound of specific model decisions.

How much uncertainty is entailed by each of the structured splits? We next compute the expected entropy of the posterior predictive distribution of the strong oracle, $p_{\text{strong}}(y|\mathbf{u}, D_{\text{supp}})$ computed over the support sets D_{supp} in the test set of each compositional split (see Figure 4). We find that overall, all the splits have a similar amount of associated uncertainty over concepts, meaning that any difference in model performances across the splits would come from compositionality and extrapolation, as opposed to higher uncertainty over the “correct” concept. We next discuss the best models on the CURI benchmark.

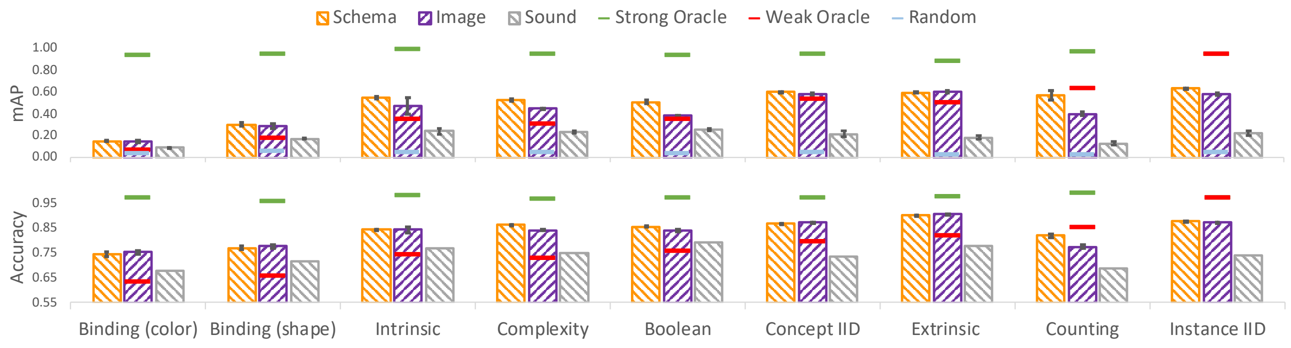


Figure 3. **Compositionality Gap**. Different splits (x-axis) plotted w.r.t performance of the strong oracle (green line) and weak oracle (red line) on the mAP (top) and Accuracy (bottom) evaluated on respective test splits (using hard negatives in support and query sets). Difference between the two is the compositionality gap (compositionality gap). **Yellow:** shows the (best) relation-net model on schema inputs, **purple:** shows the model on image inputs, and **gray:** shows the model on sound inputs. Error bars are `std` across 3 independent model runs.

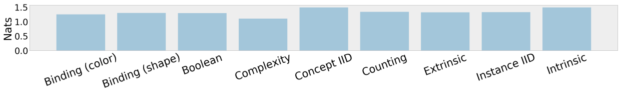


Figure 4. Average test entropy (nats) (y-axis) of the strong oracle, that extrapolates perfectly over the concept space *vs.* compositional split (x-axis).

5.2. Differences Between Models

Best Models. A summary of all results is shown in Table 2. In general, the best performing model is the relation-net applied to schema inputs, outperforming other combinations of models and input modalities on the Boolean, Concept IID, Complexity, and Instance IID splits on both the mAP as well as accuracy metrics (Figure 3); although as mentioned above, none of the models are close to the strong oracle. It is closely followed by the transformer model on schema inputs, which performs the best on Binding (color), Binding (shape), and Intrinsic splits. Utilizing schema inputs proves easier for abstraction except for the Extrinsic setting, where the task requires generalization to novel locations for objects in images, which is well supported by the inductive bias of the CNN encoder (Figure 2). In this case, the image-transformer gets an mAP of $62.1 \pm 0.7\%$, compared to the next best schema-transformer model at 60.9 ± 0.7. Further, relational learning proves more crucial in the schema case than for images, with all image models (regardless of pooling) performing better than $59.4 \pm 1.3\%$ mAP (achieved for image-avg-pool) while schema-avg-pool models get only get to $53.4 \pm 1.5\%$.

When to use a transformer? Transformer models appear to outperform relation networks in splits concerning disentangling. For instance, for the Intrinsic split with schema-relation-net is at $55.1 \pm 0.8\%$ mAP *vs.* $57.9 \pm 0.6\%$ for schema-transformer. Similarly, for the Extrinsic split the image-transformer is at $62.1 \pm 0.7\%$ mAP compared to the image-relation-net at $60.8 \pm 1.1\%$. We hypothesize that this is because the iterative message passing via. attention in transformers improves object representations for disentangling compared to relation networks that lack such a mechanism.

What is the relative difficulty of abstraction from different modalities? One of the key contributions of our work is in providing multiple modalities (image, schema, sound) for productive concept learning. We next characterize the difficulty of abstraction based on modality for the various generalization settings. In the Intrinsic setting, we find that the schema models, which have access to a "perfect" disentangled representation significantly outperform image models—a schema-avg-pool model gets to $52.7 \pm 3.1\%$ mAP while an image-avg-pool model gets to $34.4 \pm 0.0\%$ mAP.

Similarly, for the Counting split where the total number of objects are exactly specified in the schema (Figure 2), schemas are substantially better than images. For example, schema-relation-nets get to $56.25 \pm 5.32\%$ mAP while image-avg-pool is at $48.4 \pm 1.2\%$ mAP. Interestingly, the next best model—image-relation-net—is substantially worse, at $39.45 \pm 1.6\%$. Curiously, while transformer models perform well at disentangling, they seem to be quite poor for Counting, with image-transformer models getting to only $32.4 \pm 1.4\%$ mAP, suggesting a potential weakness for transformers. Overall, there appears to be an intimate link between the generalization setting and the input modality, suggesting avenues where representation learning could be improved for a given modality (e.g. images), relative to the kind of reasoning one is interested in (e.g. counting).

When does language help? On average, training models

Table 2. Results for various models with different object encoders and pooling schemes (Figure 2) on the mAP metric (top half) and the accuracy metric (bottom half) on the compositional splits of the CURI dataset. The best model (Overall) is a relation-net trained with schema inputs according to both mAP and accuracy metrics. Results with easy negatives can be found in the appendix.

Models		Performance on Splits with Hard Negatives (mAP metric)									
Encoder	Pooling	Binding (color)	Binding (shape)	Intrinsic	Complexity	Boolean	Concept IID	Extrinsic	Counting	Instance IID	Overall
Image	Average Pool	15.1 (±0.4)	28.7 (±1.0)	34.5 (±0.4)	45.8 (±0.4)	36.3 (±2.3)	56.7 (±1.0)	59.4 (±1.3)	48.5 (±1.3)	58.1 (±0.7)	42.57
Image	Relation Net	14.8 (±0.7)	29.1 (±1.8)	47.6 (±7.5)	45.4 (±0.4)	38.4 (±1.0)	58.5 (±0.7)	60.8 (±1.1)	40.3 (±2.0)	58.6 (±1.3)	43.72
Image	Transformer	14.2 (±0.2)	28.7 (±0.1)	54.1 (±3.0)	45.6 (±0.2)	44.0 (±2.2)	60.8 (±0.4)	62.1 (±0.8)	32.4 (±1.4)	61.5 (±0.9)	44.82
Image	Concat	13.5 (±0.4)	27.0 (±0.8)	34.0 (±1.0)	42.6 (±3.4)	29.6 (±2.4)	53.0 (±0.6)	60.8 (±0.7)	38.5 (±2.8)	57.1 (±0.9)	39.57
Schema	Average Pool	15.1 (±1.2)	28.3 (±1.9)	53.3 (±3.0)	48.4 (±1.0)	48.0 (±2.1)	54.6 (±1.3)	53.4 (±1.5)	55.0 (±6.4)	57.4 (±0.9)	45.94
Schema	Relation Net	15.3 (±0.9)	30.4 (±1.5)	55.1 (±0.9)	52.8 (±0.7)	51.1 (±1.4)	60.7 (±0.3)	59.8 (±0.4)	57.4 (±4.8)	63.9 (±0.4)	**49.61**
Schema	Transformer	15.9 (±0.9)	30.9 (±0.6)	57.9 (±0.6)	49.7 (±0.6)	46.9 (±1.1)	59.8 (±0.2)	60.9 (±0.6)	40.7 (±0.4)	59.8 (±0.5)	46.94
Schema	Concat	14.0 (±0.7)	27.0 (±0.8)	52.2 (±3.5)	48.0 (±1.2)	47.1 (±2.1)	53.3 (±0.9)	54.3 (±2.3)	50.0 (±3.2)	57.1 (±1.4)	44.78
Sound	Average Pool	9.4 (±0.6)	16.2 (±1.6)	26.5 (±2.3)	23.0 (±1.7)	24.1 (±1.6)	22.7 (±2.3)	25.7 (±-)	13.4 (±1.0)	23.5 (±1.4)	20.50
Sound	Relation Net	9.2 (±0.3)	17.5 (±0.4)	24.3 (±2.4)	23.3 (±1.0)	25.5 (±0.8)	21.8 (±-)	18.3 (±1.7)	13.0 (±1.7)	22.3 (±2.3)	19.47
Sound	Concat	8.0 (±0.6)	17.4 (±1.3)	29.4 (±1.6)	26.0 (±-)	22.3 (±1.1)	23.6 (±2.7)	23.2 (±1.4)	13.8 (±2.4)	21.7 (±0.5)	20.60
		Performance on Splits with Hard Negatives (accuracy metric)									
Image	Average Pool	76.3 (±0.6)	78.1 (±0.3)	82.3 (±0.2)	84.9 (±0.1)	84.4 (±0.5)	87.4 (±0.1)	90.8 (±0.2)	80.7 (±0.3)	87.6 (±0.1)	83.61
Image	Relation Net	75.6 (±0.9)	77.9 (±0.6)	84.7 (±1.2)	84.5 (±0.1)	84.5 (±0.5)	87.6 (±0.2)	90.9 (±0.1)	77.7 (±0.8)	87.5 (±0.3)	83.43
Image	Transformer	75.8 (±0.2)	78.2 (±0.0)	86.2 (±0.4)	84.9 (±0.2)	85.5 (±0.5)	87.8 (±0.0)	91.2 (±0.2)	75.6 (±0.4)	88.2 (±0.0)	83.71
Image	Concat	73.8 (±0.5)	76.9 (±0.3)	82.0 (±0.5)	84.4 (±0.3)	83.2 (±1.0)	86.6 (±0.2)	90.9 (±0.1)	77.5 (±0.7)	87.1 (±0.2)	82.49
Schema	Average Pool	74.6 (±1.1)	76.4 (±0.7)	85.0 (±1.0)	85.1 (±0.4)	85.3 (±0.9)	86.0 (±0.5)	89.2 (±0.5)	82.5 (±1.4)	86.4 (±0.6)	83.39
Schema	Relation Net	74.7 (±0.8)	77.2 (±0.6)	84.8 (±0.2)	86.5 (±0.1)	86.0 (±0.3)	87.3 (±0.1)	90.3 (±0.2)	82.2 (±0.8)	87.9 (±0.2)	**84.10**
Schema	Transformer	76.3 (±0.6)	77.9 (±0.2)	86.0 (±0.3)	85.0 (±0.1)	85.3 (±0.3)	87.3 (±0.1)	90.7 (±0.1)	76.4 (±1.0)	87.4 (±0.2)	83.59
Sound	Average Pool	68.4 (±1.2)	71.7 (±0.1)	77.9 (±1.5)	75.0 (±0.6)	79.0 (±0.7)	75.5 (±0.6)	80.8 (±-)	68.4 (±0.9)	74.7 (±0.5)	74.60
Sound	Relation Net	68.0 (±0.9)	71.8 (±0.4)	77.0 (±1.2)	75.3 (±0.7)	79.4 (±0.4)	73.9 (±-)	78.2 (±1.5)	68.7 (±1.0)	74.3 (±0.7)	74.07
Sound	Concat	65.5 (±0.7)	72.7 (±0.1)	79.3 (±0.9)	76.5 (±-)	77.4 (±0.7)	74.4 (±2.2)	80.1 (±0.8)	68.9 (±2.6)	73.8 (±0.3)	74.29

with explicit concept supervision using the concept loss (Section 4.1) improves performance by 2.8 ± 0.6% mAP (SEM error). This is a small boost relative to the gap between the original model and the strong oracle, suggesting that this simple auxiliary loss is not sufficient to internalize the LOT in a neural network. Overall, image models benefit more from language than schema models which natively utilize symbols.

6. Conclusion

We introduced the compositional reasoning under uncertainty (CURI) benchmark for evaluating few-shot concept learning in a large compositional space, capturing the kinds of productivity, unboundness and underdetermination that characterize human conceptual reasoning. We instantiate a series of meta-learning tasks, and evaluate numerous baseline models on various aspects of compositional reasoning under uncertainty, including inferential coherence, boolean operation learning, counting, and disentangling. Further, we introduce the notion of a compositionality gap to quantify the difficultly of each generalization type, and to estimate the degree of compositionality in current deep learning models. We hope our contributions of dataset, compositionality gaps, evaluation metrics and baseline models help spur progress in the important research direction of productive concept learning under uncertainty.

Acknowledgments

We would like to thank Laurens Van Der Maaten, Rob Fergus, Larry Zitnick, Edward Grefenstette, Devi Parikh and numerous other colleagues at Facebook AI Research for their feedback and discussions in helping shape this project. Specifically, we thank Larry Zitnick and Edward Grefenstette for comments on this draft. Finally, we would like to thank the developers of Hydra and PyTorch for providing amazing frameworks for running large scale deep learning experiments.

References

Agrawal, A., Batra, D., Parikh, D., and Kembhavi, A. Don't just assume; look and answer: Overcoming priors for visual question answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Andreas, J., Klein, D., and Levine, S. Learning with latent language. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2018.

Bakhtin, A., van der Maaten, L., Johnson, J., Gustafson, L., and Girshick, R. PHYRE: A new benchmark for physical reasoning. In *Neural Information Processing Systems (NeurIPS)*, 2019.

Barrett, D. G. T., Hill, F., Santoro, A., Morcos, A. S., and Lillicrap, T. Measuring abstract reasoning in neural networks. In *International Conference on Machine Learning (ICML)*, 2018.

Barsalou, L. W. Ad hoc categories. *Mem. Cognit.*, 11(3): 211–227, May 1983.

Chattopadhyay, P., Vedantam, R., Selvaraju, R. R., Batra, D., and Parikh, D. Counting everyday objects in everyday scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision (IJCV)*, 88(2):303–338, June 2010.

Feldman, J. Minimization of boolean complexity in human concept learning. *Nature*, 407(6804):630–633, October 2000.

Finn, C., Abbeel, P., and Levine, S. Model-Agnostic Meta-Learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, 2017.

Fodor, J. *The Language of Thought*. Harvard University Press, Cambrida, MA, 1975.

Fodor, J. A. and Pylyshyn, Z. W. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28:3–71, 1988.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. *Bayesian Data Analysis*. Chapman and Hall/CRC, 2004.

Goodman, N. D., Tenenbaum, J. B., Feldman, J., and Griffiths, T. L. A rational analysis of rule-based concept learning. *Cognitive Science*, 32:108–54, 2008.

Goodman, N. D., Tenenbaum, J. B., and Gerstenberg, T. Concepts in a probabilistic language of thought. In Margolis, E. and Laurence, S. (eds.), *Concepts: New Directions*. MIT Press, Cambridge, MA, 2015.

Higgins, I., Sonnerat, N., Matthey, L., Pal, A., Burgess, C. P., Botvinick, M., Hassabis, D., and Lerchner, A. SCAN: Learning abstract hierarchical compositional visual concepts. In *International Conference on Learning Representations (ICLR)*, 2018.

Hill, F., Santoro, A., Barrett, D. G. T., Morcos, A. S., and Lillicrap, T. Learning to make analogies by contrasting abstract relational structure. In *International Conference on Learning Representations (ICLR)*, 2019.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Lawrence Zitnick, C., and Girshick, R. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Kemp, C. and Jern, A. Abstraction and relational learning. *Neural Information Processing Systems (NeurIPS)*, 2009.

Kemp, C., Bernstein, A., and Tenenbaum, J. B. A Generative Theory of Similarity. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society*, 2005.

Keysers, D., Schärli, N., Scales, N., Buisman, H., Furrer, D., Kashubin, S., Momchev, N., Sinopalnikov, D., Stafiniak, L., Tihon, T., Tsarkov, D., Wang, X., van Zee, M., and Bousquet, O. Measuring compositional generalization: A comprehensive method on realistic data. In *International Conference on Learning Representations (ICLR)*, 2020.

Lake, B. M. Compositional generalization through meta sequence-to-sequence learning. In *Neural Information Processing Systems (NeurIPS)*, 2019.

Lake, B. M. and Baroni, M. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International Conference on Machine Learning (ICML)*, 2018.

Lake, B. M. and Piantadosi, S. T. People Infer Recursive Visual Concepts from Just a Few Examples. *Computational Brain & Behavior*, 2019.

Lake, B. M., Salakhutdinov, R., and Tenenbaum, J. B. The Omniglot Challenge: A 3-Year Progress Report. *Current Opinion in Behavioral Sciences*, 29:97–104, 2019.

Murphy, G. L. *The Big Book of Concepts*. MIT Press, Cambridge, MA, 2002.

Murphy, K. P. *Machine learning : a probabilistic perspective*. MIT Press, Cambridge, Mass. [u.a.], 2013. ISBN 9780262018029 0262018020.

Overlan, M. C., Jacobs, R. A., and Piantadosi, S. T. Learning abstract visual concepts via probabilistic program induction in a Language of Thought. *Cognition*, 168: 320–334, 2017.

Piantadosi, S. T. *Learning and the language of thought*. PhD thesis, Massachusetts Institute of Technology, 2011.

Piantadosi, S. T., Tenenbaum, J. B., and Goodman, N. D. Bootstrapping in a language of thought: A formal model of numerical concept learning. *Cognition*, 123:199–217, 2012.

Piantadosi, S. T., Tenenbaum, J. B., and Goodman, N. D. The logical primitives of thought: Empirical foundations for compositional cognitive models. *Psychol. Rev.*, 123 (4):392–424, July 2016.

Ravi, S. and Larochelle, H. Optimization as a model for few-shot learning. In *International Conference on Learning Representations (ICLR)*, 2017.

Ruis, L., Andreas, J., Baroni, M., Bouchacourt, D., and Lake, B. M. A Benchmark for Systematic Generalization in Grounded Language Understanding. In *Neural Information Processing Systems (NeurIPS)*, 2020.

Santoro, A., Raposo, D., Barrett, D. G., Malinowski, M., Pascanu, R., Battaglia, P., and Lillicrap, T. A simple neural network module for relational reasoning. In *Neural Information Processing Systems (NeurIPS)*. 2017.

Snell, J., Swersky, K., and Zemel, R. Prototypical networks for few-shot learning. In *Neural Information Processing Systems (NeurIPS)*. 2017.

Tenenbaum, J. B. and Griffiths, T. L. Generalization, similarity, and Bayesian inference. *Behavioral and Brain Sciences*, 24:629–40, 2001.

Triantafillou, E., Zhu, T., Dumoulin, V., Lamblin, P., Evci, U., Xu, K., Goroshin, R., Gelada, C., Swersky, K., Manzagol, P.-A., and Larochelle, H. Meta-Dataset: A dataset of datasets for learning to learn from few examples. In *International Conference on Learning Representations (ICLR)*, 2020.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In *Neural Information Processing Systems (NeurIPS)*, 2017.

Vedantam, R., Fischer, I., Huang, J., and Murphy, K. Generative models of visually grounded imagination. In *International Conference on Learning Representations (ICLR)*, 2018.

Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., and Wierstra, D. Matching networks for one shot learning. In *Neural Information Processing Systems (NeurIPS)*, 2016.

Wang, D., Jamnik, M., and Lio, P. Abstract diagrammatic reasoning with multiplex graph networks. In *International Conference on Learning Representations (ICLR)*, 2020.

Xu, F. and Tenenbaum, J. B. Word learning as Bayesian inference. *Psychological Review*, 114:245–272, 2007.