Iterated Learning Improves Compositionality in Large Vision-Language Models

Chenhao Zheng[2], Jieyu Zhang[1], Aniruddha Kembhavi[3], Ranjay Krishna[1,3]
[1]University of Washington, [2]University of Michigan, [3]Allen Institute for Artificial Intelligence
neymar@umich.edu, {jieyuz2,ranjay}@cs.washington.edu, anik@allenai.org

Abstract

A fundamental characteristic common to both human vision and natural language is their compositional nature. Yet, despite the performance gains contributed by large vision and language pretraining, recent investigations find that most—if not all—our state-of-the-art vision-language models struggle at compositionality. They are unable to distinguish between images of "a girl in white facing a man in black" and "a girl in black facing a man in white". Moreover, prior work suggests that compositionality doesn't arise with scale: larger model sizes or training data don't help. This paper develops a new iterated training algorithm that incentivizes compositionality. We draw on decades of cognitive science research that identifies cultural transmission—the need to teach a new generation—as a necessary inductive prior that incentivizes humans to develop compositional languages. Specifically, we reframe vision-language contrastive learning as the Lewis Signaling Game between a vision agent and a language agent, and operationalize cultural transmission by iteratively resetting one of the agent's weights during training. After every iteration, this training paradigm induces representations that become "easier to learn", a property of compositional languages: e.g. our model trained on CC3M and CC12M improves standard CLIP by 4.7%, 4.0% respectfully in the SugarCrepe benchmark.

1. Introduction

Scholars across disciples herald *compositionality* as a fundamental presupposition characterizing both human perception and linguistic processing [12, 17]. Through compositional reasoning, humans can comprehend the photos they take and describe those images by composing words together [2, 8, 26, 27]. For instance, compositionality allows people to differentiate between a photo of "a gold colored dog facing a person wearing black" and "a black colored dog facing a person wearing gold". Given its importance, research in both computer vision and natural language processing has sought to develop models that can similarly comprehend scenes and express them through compositional language [20, 28, 34, 41].

Yet, a series of recent evaluation benchmarks conclude that state-of-the-art vision-language models exhibit little to no compositionality [24, 42, 50, 58, 66, 69]. In fact, in many specific evaluation conditions, models perform almost close to random chance. Even models such as CLIP [47], which has become the backbone for many vision tasks, exhibit little compositionality. More striking are the experiments that suggest that compositionality doesn't emerge with scale, *i.e*. vision models do not become more compositional with increasing model size or training data [24, 42]. Similar experiments in natural language processing find that large language models also struggle with compositionality [1, 16].

Meanwhile, Cognitive Scientists have spent the last two decades studying the emergence of compositionality in human language. The results seem to indicate that the primary inductive prior that leads to language compositionality is *cultural transmission*: a phenomenon where an older generation transmits their language to a new generation [3, 5, 59, 61]. They hypothesize that this need to teach our offsprings our language creates a natural preference towards languages that are easier to learn. A compositional language, which necessitates learning only a limited number of symbols to express infinite concepts, is therefore preferred to ones with unique symbol-to-concept bindings.

To demonstrate this hypothesis, scientists study the language that emerged from the "Lewis Signaling Game". Lewis Signaling Game [37] is a theoretical framework where two people communicate with one another to solve the "object reference" problem (Fig. 1(1a)). Their communication channel is restricted to symbols, which do not represent any known language, forcing participants to develop a new shared language to communicate. They simulate cultural transmission by replacing human participants across "generations", and observe how new combinations of participants modify their language (Fig. 1(1b)). They verify that over multiple generations, the emergent language becomes more compositional [10, 11, 21, 30, 38, 61].

In this paper, we operationalize cultural transmission as an iterated learning (IL) algorithm for vision-language models. Consider the popular CLIP model; it is trained to learn

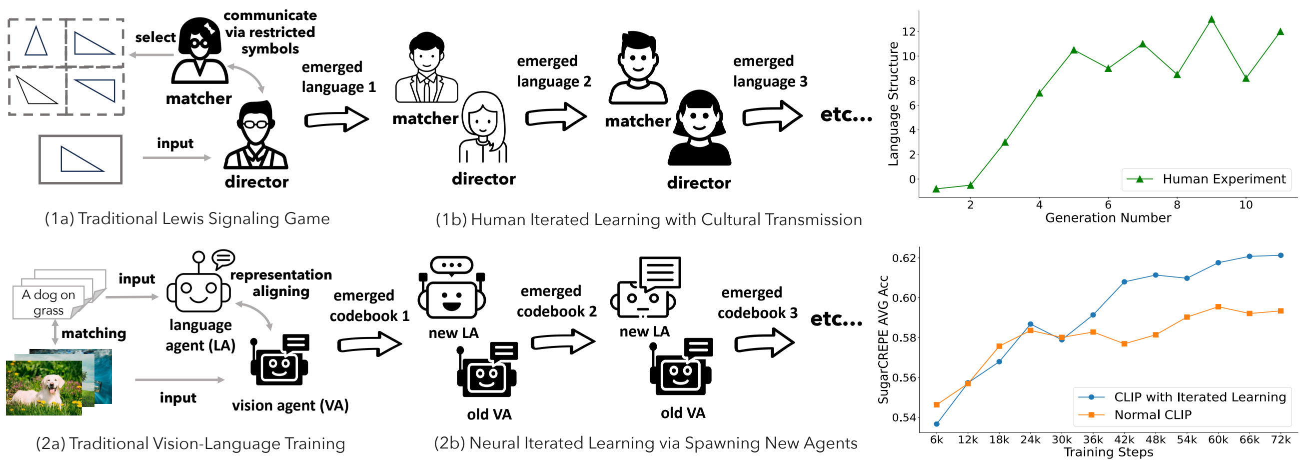

Figure 1. (1). From studying the language that emerged from Lewis Signaling Game, evolutionary linguistics found that iterated learning with cultural transmission leads to language compositionality. (2). We interpret vision-language model training as Lewis Signaling Game between neural agents, and discovered iterated learning can also improve the compositionality of vision-language model's representation

representations through an interplay between vision and language representations [47]. At a high level, its contrastive learning objective trains image representations that can retrieve their corresponding textual description from a set of distractors, and vice versa. We reframe this objective through the lens of the Lewis Signaling Game (Fig. 1(2a)). Similar to the cognitive science studies that involve two human participants, vision-language training can be viewed as a game between two model participants: a vision agent and a language agent attempting to learn a shared representation. With this framing in mind, we apply cultural transmission by periodically spawning a new language agent to replace the old one (Fig. 1(2b)). Intuitively, the need to re-train a new language agent is akin to "teaching a new generation" and should similarly encourage the vision agent to produce representations that are easier to learn. We also create the notion of "shared and limited communication symbols" by learning a shared codebook as the basis for representations that both agents can use.

Our experiments demonstrate that our algorithm does in fact induce easy-to-learn representation, improving compositionality in vision-language models. For example, our model trained in CC3M improves standard CLIP by 4.7% in SugarCrepe [24] and by 3.8% in CREPE [42], both benchmarks are specially designed for testing compositionality for vision-language models. Notably, our model exhibits better compositionality than existing compositional methods, such as NegCLIP [65]. Our model does not require extra training time despite periodically resetting model weights, and does not harm the CLIP's recognition performance. We also demonstrate the easy-to-learn property in our representation in experiments and find that the emerged codebook contains interpretable concepts.

2. Related Works

Our work is inspired by cognitive science literature and the related works span various areas, including large vision-language models, the emergence of language, and interacting neural agents.

Compositionality in vision-language models. With the popularity of the CLIP model [47], contrastive learning has become the de-facto way of aligning representations for different modalities [18, 25, 39, 53, 63, 64]. However, despite their remarkable ability in zero-shot recognition [47], their features exhibit little compositionality [24, 42, 50, 58, 66, 69]. For example, all models struggle to identify the captions with correct word order [65], compose concepts together to express compositional concepts [42], and compose attributes and relations [22, 50, 65, 69]. Attempts have been made to enhance CLIP's compositionality, including hard-negative mining [65], cleaning the data [39], and using novel representation formats [7, 40, 70]. However, the recently proposed SugarCrepe benchmark [24] finds that their improvements are overestimated, calling for a more effective method.

Iterated learning and cultural transmission. Human language is, for the most part, compositional. Evolutionary linguists have spent decades studying the origin of compositionality of human language [45, 46]. One important factor appears to be the need to transmit the language across multiple generations [32], formulated by Kirby [30–32] as a framework called iterated learning. Extensive simulations [4, 9, 57] and human experiments [11, 31, 32, 56] demonstrate its ability to incentivize the emergence of compositional structure in their language, in small-scale and quantized environments. Newer experiments in open and continuous environments also conclude similar findings [5, 55, 62], although they observe a large amount of randomness across experiments [5].

Emergence of linguistic structure in neural agents. Collaborative AI agent systems have been the subject of much research, in which neural agents communicate to learn a language while accomplishing goals [13, 33, 35, 36, 51]. Most approaches learn a discrete communication protocol while playing the Lewis Signaling Game [10, 33, 35, 38, 51]. Researchers find the language developed by agents, if compositional, shows enhanced systematic generalization capabilities [6, 33, 51]. However, compositionality does not occur naturally [6, 33] and is not tied to generalization pressure [29]. Some works introduced neural iterated learning frameworks [10, 38, 49, 51, 60]. Using topographic similarity as a measurement, they found that emergent language is more compositional [10, 38, 51]. Some works also show the resulting compositional language is easier to learn [38, 52], corresponding to the finding in cognitive science [32]. However, the experiments are limited to small domains with easily-categorizable inputs like simple cubics or balls [33, 35, 38, 51]. The message structures and network architecture are also simple, raising the question of scalability. Our work is similar in the idea of using iterated learning to boost compositionality. However, our model observes large-scale real-world data that are not easily categorizable and uses contrastive learning as opposed to reinforcement learning used by most methods.

3. Method

We design an iterated learning algorithm to improve the compositionality of vision-language models. To do so, we draw an analogy between the process of vision-language contrastive learning and the procedure of Lewis Signal Game [36], and build our method upon CLIP's training objective [47]. We first reframe CLIP as Lewis Signaling Game (Sec. 3.1); then we introduce the shared codebook module that bottlenecks each modalities' representations (Sec. 3.2); finally, we describe our iterated learning algorithm (Sec. 3.3).

3.1. Reframing vision-language contrastive learning as a Lewis Signaling Game

In the traditional Lewis Signaling Game, two people communicate through restricted symbols to solve a referential task. In the task, one person called the "director" observes an input stimuli (usually a picture of abstract shapes) and needs to choose a sequence of symbols from a limited vocabulary to send over to the second person, the "matcher". The matcher sees only the symbols and a set of observations, from which they must identify the one seen by the director. The evolving conversation patterns across time are treated as emergent language. This game setup is similar to the contrastive learning procedure popular today in vision-language training, where a vision agent and a language agent observe modality-specific inputs and need to communicate together to identify the matching image-text pairs from distractors.

More formally, during the training process, two agents observe their distinctive inputs (images $\{u_i\}_{i=1}^N$ for the vision agent and texts $\{v_i\}_{i=1}^N$ for the language agent). They encode the inputs to representations $(f_\theta(u_i), g_\phi(v_i))$, which serves as the cross-agent communication messages. The communication objective is that, given N images and N pieces of text, the corresponding image-text pairs should be successfully matched, implemented using the contrastive objective:

$$\mathcal{L} = -\sum_{i=1}^{N} \log \frac{\exp\left(f_\theta(u_i) \cdot g_\phi(v_i)/\tau\right)}{\sum_{j=1}^{N} \exp(f_\theta(u_i) \cdot g_\phi(v_j)/\tau)}, \quad (1)$$

where τ is a small constant. The final aligned representation $(f_\theta(u_i), g_\phi(v_i))$ can be viewed as *the shared language* emergent between the two agents.

3.2. Shared codebook for a regulated representation

One of the key designs of the Lewis Signaling Game is the limited vocabulary that participants can use, while in vision-language contrastive learning the agents don't have any regulation on the representations they use to communicate. Therefore, to follow the Lewis Signaling Game, we employ a learnable codebook as the basis of representations shared by agents to regulate their representation space. In particular, the codebook is composed of a finite number of codes, representing shared and limited "vocabularies" in the learned language. The final layer of vision and language encoders sparsely combines the codebook to produce the final representation, representing the learnable "vocabulary composition rule"

Let $\{c_i\}_{i=1}^C$ be a codebook, where C is the predefined number of codes. We use the Transformer architecture for both agents. Thus, given an input image u, the vision agent f extracts patch embeddings p_j for each patch j from the transformer's last-layer activations. We define the similarity score between code c_i and the image u as the maximum cosine value between the code and patch features:

$$r_i^u = \max_j < f_{p_j}, c_i > \quad (2)$$

This codebook architecture is derived from recent work using codebooks for vision-language training [7]. Following [7], we normalize r_i^u using Sparsemax function [43], which generates a *sparse* similarity score w_i^v for each code. The output representation for the input image u is the linear combination of codes c_i, with w_i^v being multiplied as weights:

$$f(u) = \sum_{i}^{C} w_i^v \cdot c_i \quad (3)$$

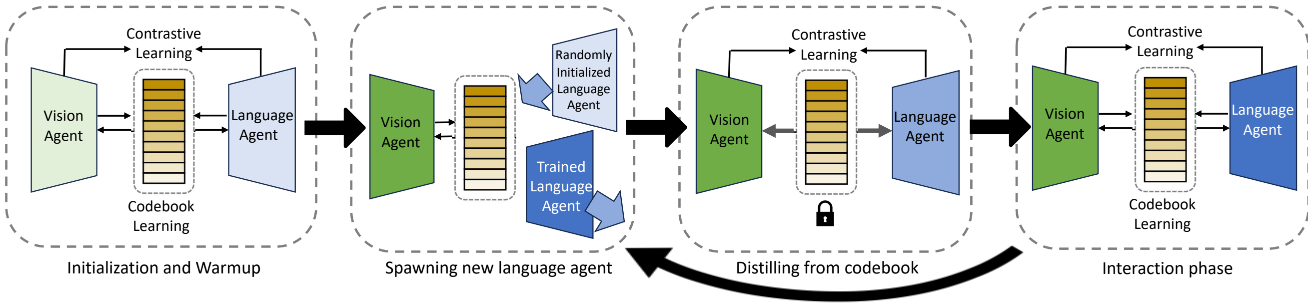

Figure 2. Our iterated learning algorithm is built on CLIP augmented with a shared codebook. The algorithm consists of a warmup stage and three iterated phases that cycle until the end of training. In each cycle, we 1) spawn a new language agent to replace the old one. 2) frozen codebook weight for a certain number of steps. 3) let agents interact under standard vision-language contrastive learning.

The procedure for obtaining text representation $g(v)$ using the language agent g is defined analogously. Instead of patch embeddings for the vision agent, here, we use the language input's token embeddings.

3.3. Iterated learning in training

Our iterated learning algorithm consists of a warmup stage, followed by K training cycles; each mirrors the concept of 'generations' in cultural transmission theory and consists of three phases: spawning a new language agent, distillation from the codebook, and an interaction phase. We visualize the algorithm in Fig. 2.

Initialization and warmup stage. The beginning of our training algorithm is similar to CLIP's algorithm. We randomly initialize the parameters of both the agents and let them train for T_{warm} number of iterations.

Spawning a new language agent. This stage simulates introducing a new participant that replaces an older one, representing a new generation in cultural transmission. While studies in cognitive science [3, 5] replace both participants over multiple generations, our ablation study indicates that replacing both is unnecessary; it even increases the training time required to achieve the same level of compositionality. By contrast, we replace only the language agent between generations by reinitializing it with random parameters. Although outside the scope of this paper, we hypothesize that resetting just the language agent works better empirically because the vision agent needs to simultaneously learn lower-level visual features and also associate them with high-level concepts while the language agent only needs to learn to extract high-level concepts from text.

Distilling from the codebook. Serving as the basis for both agent's representations, the quality of the learned codebook is essential. We find that introducing a new agent, with its randomly initialized and under-trained weights, leads to large changes to the codebook gradients, causing instability in training. We, therefore, add a distillation stage to ensure that the codebook evolves smoothly across generations. The older language agent is distilled into the new language agent for $T_{distill}$ iterations [23]. We temporarily freeze the codebook's weights during this phase. This allows the new agent to adapt to the existing codebook structure without introducing disruptive changes. Unlike traditional distillation, this phase does not train till convergence. After $T_{distill}$ steps, we switch to the interaction phase.

Interaction phase. After distillation, we unfreeze the codebook and train the model normally following the standard vision-language contrastive learning paradigm [47]. By letting agents interact freely, we expect their representations to begin aligning with one another again. We also limit the duration of this interacting phase to be $T_{interact}$ step, ensuring a learning bottleneck such that the education process from the old vision agent to the new language agent is incomplete and biased.

After the interacting stage, the current generation of agents is considered finished and the next generation begins. We repeat the above three phases until the end of training. During the last phase, we extend the interaction phase to allow training till convergence.

Understanding our algorithm. *From a cognitive science perspective*, the "knowledge gap" between old and new agents creates an implicit "teaching" scenario, where the vision agent interacts with the newly initialized language agent to realign both their cross-modal representations. This pressure to teach, as posited in cultural transmission theory, encourages the developed representations to be easier for subsequent agents to learn, potentially leading to better compositionally. We empirically demonstrate this "easy-to-learn" property at Sec. 4.4.

From a machine learning perspective, theory and results suggest that self-distillation performs label smoothening [68] and smoothness regularization in the function space [44]. It reinforces the optimization bias of neural networks for smooth solutions [48]. In other words, distillation with early stopping—like the one we are doing—makes the new generation a smoother low-frequency approximation of the older

Dataset	Method	CREPE-systematicity		CREPE-productivity			SugarCrepe			Cola	Winoground	Mean
		atom	compound	replace	swap	negate	add	replace	swap	Txt2Img	Txt2Img	
CC3M	CLIP [47]	28.1	38.4	9.8	18.1	4.0	61.9	64.3	52.9	17.6	8.1	28.3
	Codebook-CLIP [7]	28.8	40.3	10.9	19.2	3.5	65.9	64.8	54.9	15.7	8.8	31.2
	NegCLIP* [65]	29.5	41.8	11.6	**33.3**	**5.8**	59.3	59.2	**60.1**	16.5	11.8	32.8
	IL-CLIP (Ours)	**33.2**	**47.7**	**14.6**	22.3	5.3	**66.1**	**67.0**	54.5	**20.0**	**13.3**	**34.4**
CC12M	CLIP [47]	35.0	42.7	12.3	19.5	14.6	67.5	70.0	60.2	21.5	7.2	34.9
	Codebook-CLIP [7]	35.6	43.9	14.4	22.0	12.8	71.3	71.1	59.5	20.8	9.5	36.1
	NegCLIP* [65]	**36.6**	45.2	14.9	**35.8**	**15.2**	65.0	70.2	**67.2**	**22.7**	7.3	**38.0**
	IL-CLIP (Ours)	**36.6**	**47.5**	**17.9**	23.9	14.8	**73.8**	**73.0**	62.9	20.2	**10.1**	**38.0**

Table 1. **Evaluation on compositionality benchmarks.** We do image-to-text retrieval on CREPE systematicity-CC12M split, CREPE productivity split, and SugarCrepe [24, 42]. We do text-to-image retrieval on Cola and Winoground [14, 50]. We report the retrieval R@1 scores. IL-CLIP notably improves CLIP's compositionality, and exhibits better performance than NegCLIP in most datasets. (*) Note that NegCLIP directly trains on the text negatives close to "swap" objectives, and therefore obtains unusually high scores for that split.

Dataset	Method	ImageNet1k	CIFAR-100	CIFAR-10	STL-10	VOC2007	Caltech101	SUN397	Pets	Flowers102	Food101	ObjectNet	CLEVR	Smallnorb	Resisc45	DMLAB	ImageNet-A	ImageNet-R	IN-sketch	Mean
CC3M	CLIP [47]	13.7	18.6	43.5	80.7	44.3	60.1	28.6	8.9	9.1	8.3	8.0	**19.5**	5.2	12.6	11.7	3.0	17.7	7.2	21.7
	Codebook-CLIP [7]	**14.8**	**22.0**	**49.8**	85.4	**48.3**	60.8	30.4	8.8	8.5	**10.5**	**9.1**	16.7	4.8	**19.8**	17.5	**3.7**	**20.1**	**8.2**	**24.4**
	NegCLIP [65]	11.8	19.6	44.0	78.2	44.6	52.1	25.8	9.1	8.6	6.6	6.9	15.1	6.2	13.8	11.9	2.4	15.8	5.1	21.0
	IL-CLIP (Ours)	14.2	20.9	48.6	**87.7**	**48.3**	**61.1**	**32.8**	**10.0**	**9.2**	9.1	8.4	15.8	**5.5**	15.6	**18.7**	2.9	18.8	6.5	24.2
CC12M	CLIP [47]	31.4	30.9	60.1	89.3	53.3	72.5	41.0	49.6	21.1	31.5	17.8	20.0	11.7	26.5	13.6	4.4	44.2	24.0	35.7
	Codebook-CLIP [7]	**34.2**	**39.6**	**68.1**	90.3	55.5	75.4	45.8	**53.9**	**24.8**	**32.3**	20.4	**24.0**	**15.5**	27.6	11.7	5.2	48.8	**26.9**	**38.8**
	NegCLIP [65]	28.9	27.1	55.4	89.7	54.1	72.8	42.6	44.6	22.3	30.2	17.8	17.5	10.5	26.2	**15.9**	4.1	39.6	22.0	34.5
	IL-CLIP (Ours)	32.8	32.5	61.6	**94.1**	**60.0**	**76.9**	**49.7**	51.6	21.4	31.8	**22.7**	20.6	12.9	**27.7**	15.3	**7.2**	**49.0**	25.6	38.5

Table 2. **Evaluation of zero-shot image classification on 18 commonly used public datasets.** Scores are reported in terms of top-1 accuracy. Using a shared codebook (Codebook-CLIP) boosts standard CLIP's classification performance, and adding our iterated learning paradigm on top of Codebook-CLIP (IL-CLIP) does not sacrifice the overall performance.

generation. During the interaction phase, the vision agent adjusts its parameters to align better with this newer, smoother language agent. Since smoother functions are characterized by a smaller Lipschitz constant, they are easier to learn; therefore, every iteration should lead to easier-to-learn functions. Since compositional languages are easier to learn, every cycle possibly makes the representations more compositional. We observe this phenomenon empirically in our experiments. At Sec. 4.4, we show through experiment that the upper bound of Lipschitz constant indeed decreases over time.

4. Experiment

Our experiments evaluate both the compositionality (Sec. 4.2) and recognition capability (Sec. 4.3) of the trained representation. In Sec. 4.4, we provide a detailed analysis of iterated learning, followed by model ablations in Sec. 4.5. We start by describing implementation details.

4.1. Experiment Setup

Training. We utilize controlled experimental settings to ensure fair comparisons across models. We train our model and all the baselines on both CC3M and CC12M datasets [54]. For the vision agent, we adopt the default Vision Transformer (ViT-B/32) architecture [15], while the language agent is the same basic transformer architecture as the text encoder in CLIP [47]. Following [7], the codebook contains $16,384$ codes, each a 512-dimensional vector. In CC3M, we set $T_{warm}, T_{distill}, T_{interact}$ to be 6000, 1000, and 5000 steps respectively. We extended the training of the model with the final generation's parameters for additional 12k steps to ensure better convergence. We use a batch size of 1024. Detailed hyperparameter settings are available in the appendix.

Baseline models. We compare our method with standard CLIP [47], CLIP augmented with codebook (codebook-CLIP) [7], and CLIP enhanced through negative mining for improving compositionality (NegCLIP) [65]. Hard negative mining assumes an underlying compositional structure and produces hard negatives given that structure. As such, NegCLIP serves as an unfair baseline that has additional information about how the compositionality evaluation sets were constructed. We follow the NegCLIP design in [65], with the difference that we are training from scratch. We create text negatives by swapping linguistic elements. We generate image negatives by maintaining a running pool of image representations, from which we extract the nearest

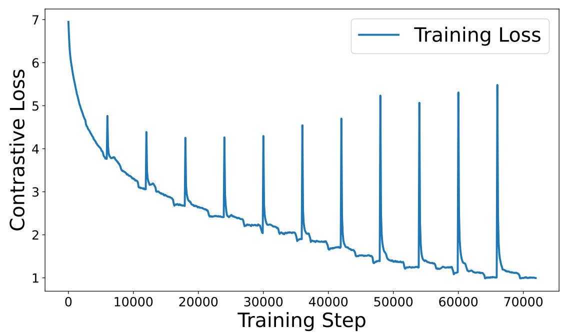

Figure 3. **Iterated learning loss curve.** Cross-modality alignment steadily improves across generations.

neighbors for each batch. For a fair comparison, all models (except for NegCLIP) are trained using identical dataset and training protocols. NegCLIP sees twice the amount of text data because of the hard negatives, and takes ~1.5x more steps to train.

4.2. Iterated learning improves compositionality

We evaluate compositionality using SugarCrepe [24], CREPE [42], Cola [50], and Winoground [14] (Tab. 1). These benchmarks contain image-text retrieval tasks with compositional hard-negative distractions. CREPE and SugarCrepe generate hard negative captions by swapping, replacing, or adding linguistic elements, whereas Cola and Winoground feature hand-curated hard negative images with similar visual elements but differing semantic meanings. We show examples of these data in Tab. **??**. Tab. 1 shows image-to-text retrieval accuracy for CREPE and SugarCrepe, alongside text-to-image retrieval accuracy for Cola and Winoground.

Our model outperforms all baselines in most datasets and shows notable improvement over standard CLIP. In particular, our model improves CLIP more significantly than NegCLIP, which sees text negatives close to the data in benchmarks in training time. NegCLIP achieves high scores in subsets that are close to its training objective (e.g. word swapping and negating), but fails to generalize to other hard negative types. Codebook-CLIP also gains performance improvement over CLIP, perhaps because the sparse codebook weight cleans supervision when facing part-of-image matches part-of-text scenarios. So the improvement of our IL-CLIP is contributed both by the iterated learning paradigm and by the shared codebook.

4.3. Iterated learning doesn't harm recognition

We evaluate how iterated learning affects image recognition, following the common practice of evaluating zero-shot image classification. We report the zero-shot image-text retrieval and linear probing performance in the appendix. We conduct the zero-shot image classification on 18 widely-used datasets (Tab. 2), including both standard recognition datasets and datasets from the VTab benchmark [67] that measure the model's robustness.

In line with findings from [7], we also observe improvements for Codebook-CLIP over the standard CLIP model. Benefiting from the shared codebook, IL-CLIP also improves standard CLIP performance. We observe that using iterated learning on top of CLIP-codebook downgrades its performance slightly, but the difference is minimal, and IL-CLIP ranks the best in several datasets. NegCLIP, however, performs notably worse than standard CLIP. This is perhaps because compositionality is often viewed to be in opposition to tasks that improve with *context*. Intuitively, if a model uses *context* to predict the existence of the "road" when it sees a "car", it will increase performance on recognition benchmarks but is not compositional. Such contextual biases are commonplace in vision benchmarks, causing compositionality to be at odds with recognition Surprisingly, iterated learning renders on-par performance compared with its normal training counterparts. Thus, we conclude that the iterated learning paradigm does not harm recognition.

4.4. Analysis on iterated learning

We provide a detailed analysis of iterated learning here, including evidence that IL produces easy-to-learn representations, improvement of cross-modality alignment across generations, and interpretability in the codebook.

Iterated learning produces easy-to-learn visual representation. As shown in cognitive science studies [5, 32, 45], compositional languages are easy-to-learn. While it is difficult to explicitly prove that the learned visual representations are compositional, we design an experiment to demonstrate they are easy-to-learn by new language agents. In particular, given a visual agent and the codebook from a certain generation, we fix their weights and use them to train a new language agent via contrastive loss. We target to observe how well a language agent can "learn" to align its representation from different well-trained visual agent "teachers". We evaluate both our IL-CLIP (with iterated learning) and codebook-CLIP (without iterated learning). The spawned language agents in all runs are initialized using the same random weights. The results are shown in Fig. 5. We find the language agents paired with vision representations developed through iterated learning achieved significantly higher matching accuracy, implying enhanced ease of learning. This is further underscored by the steeper initial slope of accuracy curves of IL-CLIP, indicating the faster learning speed for the new language agent. Thus, we conclude that IL-trained visual representation is significantly easier to learn and therefore has more chance to be compositional. Additionally, we observe from the curves of IL-CLIP that the top-1 accuracy is much higher if visual representations from later genera-

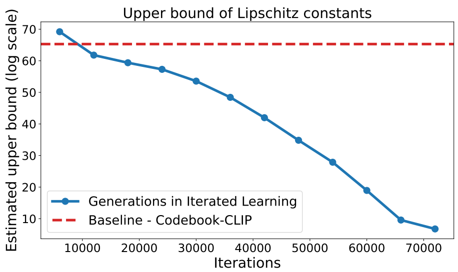

Figure 4. **Estimated Upper bound of Lipschitz Constant for Codebook-CLIP and different generations of IL-CLIP (log scale).**

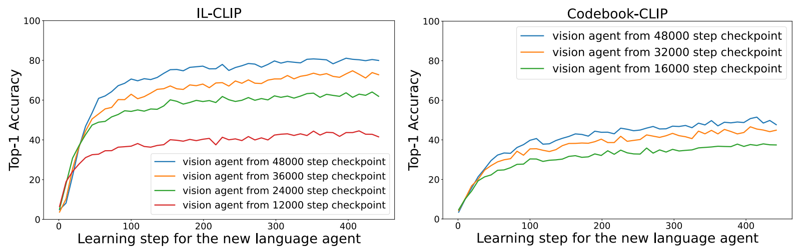

Figure 5. **Plot of in-batch image text accuracy vs. training step when a new language encoder is trained to align with fixed visual representation.** We compare between visual representation produced with iterated learning (left) and without iterated learning (right).

tions are used, suggesting that the property of being easy to learn progressively evolves across generations.

Iterated learning performs smoothness regularization We find that iterated learning can be seen a smoothness regularization by comparing the Lipschitz constant between our models and codebook-CLIP. While the exact Lipschitz constant for a complex model is intractable, we can estimate the upper bound of Lipschitz constant [19]. As shown in Fig. 4, the estimated upper bound of Lipschitz constant decreases as generation increases in iterated learning setting and is much smaller than the model trained with the standard scheme.

Cross-modality alignment steadily improves across generations. The contrastive loss measures the cross-modality alignment between image-text pairs. We plot the training loss for one of our IL-CLIP models (Fig. 3). Despite the big increase in loss when a new language agent is spawned, the loss still decays smoothly across generations. We attribute this to the representations becoming easier to learn, so the new language agents need fewer iterations to reach the alignment of the last generation and start to improve further.

The evolved codebook is (mostly) interpretable. We visualize the learned codebook by retrieving the top 5 most relevant images for each code (using Eq 2). We find that the codes correspond to different (somewhat) interpretable semantic concepts In Fig. 6(a), we show three examples of codes that happen to align with human vocabulary, while we show the foremost codes (sorted by index) in the appendix to ensure unbiased evidence. After mapping the codes manually, we can reverse the process and interpret which codes are selected when viewing a new image (Fig. 6(b)). For example, both the “horse” and “tent” codes are assigned a higher weight when viewing an image that contains both, indicating the model’s compositional understanding. We find that such interpretations are harder to find in codebook-CLIP (e.g. Fig. 7), which is shown via a user study in the appendix.

4.5. Ablation Study

We ablate the training duration for each generation, which agent to reset, and the choice to freeze the codebook during distillation. All models are trained on CC3M dataset.

Generation cycle. We train three models using different numbers of steps for each generation while ensuring the same total number of training steps. Tab. 3 shows that both too few and too many steps will result in a decrease in compositionality performance, while the recognition performance is positively related to the number of steps. We hypothesize that, on one hand, the interacting agents may not be able to produce reasonably aligned representation in a very short generation cycle, and the resulting low recognition performance can negatively influence compositionality, demonstrated in [24, 42]. On the other hand, a long generation cycle enables agents to converge better in one generation, potentially leading to better recognition. However, the reduction of generational transition frequency possibly decreases the chance to evolve more compositional representation.

Which agent to spawn? We experiment with resetting only language/vision agents and resetting both alternatively. Resetting only language agents renders the best performance. The alternative reset setting significantly downgraded the performance, suggesting ensuring the continuity of at least one side of agent weight is necessary for preventing the loss of recognition capacity. Interestingly, spawning language agents exhibit better performance than resetting vision agents, although the training paradigm is entirely symmetric. This is perhaps because vision agents need to learn low-level feature extractors before obtaining high-level concepts while the text is naturally abstracted by humans, therefore resetting vision agents would require more re-training efforts.

Frozen codebook. We study the necessity of enforcing the continuous evolution of the codebook. We train another model without fixing the codebook weight at the start of each generation. According to Table 3, this downgrades both the compositionality and recognition performance, since the randomly initialized weight of the newly initialized agent may contaminate the codebook.

IL w/wo codebook. Finally, we compare our method with-/without the codebook. The results demonstrate the efficacy of using a codebook for iterated learning, since our method without the codebook underperforms its counterpart with the codebook under both compositionality and image classifica-

(a) Top-5 most relevant images for three codes

(b) Ranks of all codes in descending order based on their respective weights when linearly combining into image representation

Figure 6. **Visualization of the codebook.** Most of the codes in the evolved codebook are well-grounded to specific semantic meanings, and we found some of them align with human vocabulary. We can also visualize the model's compositional reasoning by measuring how much each code contributes to the image representation.

tion evaluations.

IL vs. Lipschitz Regularization. In Sec. 4.4, we show that iterated learning performs smoothness regularization and reduces Lipschitz constant. A natural question is if Lipschitz regularization can achieve the same effect as iterated learning. We therefore trained a variant of Lipschitz-regularized CLIP that applies spectral normalization after each linear layer. As shown in Tab. 3, the model trained with only Lipschitz regularization barely improves performance.

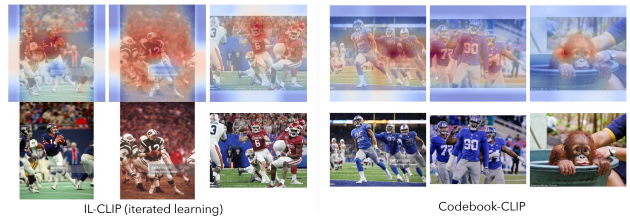

Figure 7. **Comparison of codebook interpretability.** As an example, we retrieve Top-3 most relevant images for the "football player" code, and find IL-CLIP produces more consistent images.

Study Objectives	Variation	COMP	CLS
Generation Cycle	3k step	32.1	23.8
	6k steps	**34.4**	24.2
	12k steps	32.9	**24.3**
Spawn Target	Language Agent	**34.4**	**24.2**
	Vision Agent	33.9	24.0
	Alternatively	30.7	21.4
Codebook Continuity	w weight fixed	**34.4**	**24.2**
	w/o weight fixed	31.9	24.0
IL w/wo Codebook	w codebook	**34.4**	**24.2**
	w/o codebook	28.0	21.5
Lipschitz Regularization	Iterated learning	**34.4**	**24.2**
	L-Regularized	27.8	21.0

Table 3. **Ablation study**: "COMP" represents average scores of compositional benchmarks in Sec. 4.2. "CLS" represents average scores of image classification (same datasets as in Sec. 4.3)

5. Discussion

Conclusions. In this paper, we design an iterated learning algorithm that improves the compositionality in large vision-language models, inspired by cultural transmission theory in cognitive science. To achieve this, we treat vision-language contrastive learning as two agents playing the Lewis Signaling Game, and iteratively spawning new language agents by resetting weights. Our model demonstrates improvements in compositional understanding over the standard CLIP across various benchmarks, while maintaining comparable recognition capabilities. This work paves the way for future advancements in other areas requiring compositional understanding, suggesting the potential applicability of iterated learning in a broader range of tasks.

Limitations. Similar to the findings in cognitive science [5], we observe that the learning process of IL-CLIP could be unstable due to the randomness in spawning new agents. More work is needed to stabilize the learning process.

References

[1] Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The reversal curse: Llms trained on" a is b" fail to learn" b is a". *arXiv preprint arXiv:2309.12288*, 2023. 1

[2] Léon Bottou. From machine learning to machine reasoning. *Machine learning*, 94(2):133–149, 2014. 1

[3] Henry Brighton and Simon Kirby. Understanding linguistic evolution by visualizing the emergence of topographic mappings. *Artificial life*, 12(2):229–242, 2006. 1, 4

[4] Angelo Cangelosi and Domenico Parisi. *Simulating the evolution of language*. Springer Science & Business Media, 2012. 2

[5] Jon W Carr, Kenny Smith, Hannah Cornish, and Simon Kirby. The cultural evolution of structured languages in an open-ended, continuous world. *Cognitive science*, 41(4):892–923, 2017. 1, 3, 4, 6, 8

[6] Rahma Chaabouni, Eugene Kharitonov, Diane Bouchacourt, Emmanuel Dupoux, and Marco Baroni. Compositionality and generalization in emergent languages. *arXiv preprint arXiv:2004.09124*, 2020. 3

[7] Yuxiao Chen, Jianbo Yuan, Yu Tian, Shijie Geng, Xinyu Li, Ding Zhou, Dimitris N Metaxas, and Hongxia Yang. Revisiting multimodal representation in contrastive learning: from patch and token embeddings to finite discrete tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15095–15104, 2023. 2, 3, 5, 6

[8] Noam Chomsky and Morris Halle. Some controversial questions in phonological theory. *Journal of linguistics*, 1(2): 97–138, 1965. 1

[9] Morten H Christiansen and Simon Kirby. Language evolution: Consensus and controversies. *Trends in cognitive sciences*, 7 (7):300–307, 2003. 2

[10] Michael Cogswell, Jiasen Lu, Stefan Lee, Devi Parikh, and Dhruv Batra. Emergence of compositional language with deep generational transmission. *arXiv preprint arXiv:1904.09067*, 2019. 1, 3

[11] Hannah Cornish, Rick Dale, Simon Kirby, and Morten H Christiansen. Sequence memory constraints give rise to language-like structure through iterated learning. *PloS one*, 12(1):e0168532, 2017. 1, 2

[12] MJ Cresswell. Logics and languages. 1973. 1

[13] Roberto Dessì, Eugene Kharitonov, and Baroni Marco. Interpretable agent communication from scratch (with a generic visual processor emerging on the side). *Advances in Neural Information Processing Systems*, 34:26937–26949, 2021. 3

[14] Anuj Diwan, Layne Berry, Eunsol Choi, David Harwath, and Kyle Mahowald. Why is winoground hard? investigating failures in visuolinguistic compositionality. *arXiv preprint arXiv:2211.00768*, 2022. 5, 6

[15] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 5

[16] Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jian, Bill Yuchen Lin, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D Hwang, et al. Faith and fate: Limits of transformers on compositionality. *arXiv preprint arXiv:2305.18654*, 2023. 1

[17] Jerry A Fodor and Zenon W Pylyshyn. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2): 3–71, 1988. 1

[18] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15180–15190, 2023. 2

[19] Henry Gouk, Eibe Frank, Bernhard Pfahringer, and Michael J Cree. Regularisation of neural networks by enforcing lipschitz continuity. *Machine Learning*, 110:393–416, 2021. 7

[20] Madeleine Grunde-McLaughlin, Ranjay Krishna, and Maneesh Agrawala. Agqa: A benchmark for compositional spatio-temporal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1

[21] Shangmin Guo, Yi Ren, Serhii Havrylov, Stella Frank, Ivan Titov, and Kenny Smith. The emergence of compositional languages for numeric concepts through iterated learning in neural agents. *arXiv preprint arXiv:1910.05291*, 2019. 1

[22] Lisa Anne Hendricks and Aida Nematzadeh. Probing image-language transformers for verb understanding. *arXiv preprint arXiv:2106.09141*, 2021. 2

[23] Geoffrey Hinton and et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 4

[24] Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. *Advances in neural information processing systems*, 2023. 1, 2, 5, 6, 7

[25] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*, 2023. 2

[26] Dieuwke Hupkes, Verna Dankers, Mathijs Mul, and Elia Bruni. Compositionality decomposed: How do neural networks generalise? *Journal of Artificial Intelligence Research*, 67:757–795, 2020. 1

[27] Theo MV Janssen and Barbara H Partee. Compositionality. In *Handbook of logic and language*, pages 417–473. Elsevier, 1997. 1

[28] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as compositions of spatio-temporal scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10236–10247, 2020. 1

[29] Eugene Kharitonov and Marco Baroni. Emergent language generalization and acquisition speed are not tied to compositionality. *arXiv preprint arXiv:2004.03420*, 2020. 3

[30] Simon Kirby. Spontaneous evolution of linguistic structure-an iterated learning model of the emergence of regularity and irregularity. *IEEE Transactions on Evolutionary Computation*, 5(2):102–110, 2001. 1, 2

[31] Simon Kirby, Hannah Cornish, and Kenny Smith. Cumulative cultural evolution in the laboratory: An experimental approach to the origins of structure in human language. *Proceedings of the National Academy of Sciences*, 105(31): 10681–10686, 2008. 2

[32] Simon Kirby, Tom Griffiths, and Kenny Smith. Iterated learning and the evolution of language. *Current opinion in neurobiology*, 28:108–114, 2014. 2, 3, 6

[33] Satwik Kottur, José MF Moura, Stefan Lee, and Dhruv Batra. Natural language does not emerge'naturally'in multi-agent dialog. *arXiv preprint arXiv:1706.08502*, 2017. 3

[34] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017. 1

[35] Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. Multi-agent cooperation and the emergence of (natural) language. *arXiv preprint arXiv:1612.07182*, 2016. 3

[36] Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark. Emergence of linguistic communication from referential games with symbolic and pixel input. *arXiv preprint arXiv:1804.03984*, 2018. 3

[37] David Lewis. *Convention: A philosophical study*. John Wiley & Sons, 2008. 1

[38] Fushan Li and Michael Bowling. Ease-of-teaching and language structure from emergent communication. *Advances in neural information processing systems*, 32, 2019. 1, 3

[39] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022. 2

[40] Alexander H Liu, SouYoung Jin, Cheng-I Jeff Lai, Andrew Rouditchenko, Aude Oliva, and James Glass. Cross-modal discrete representation learning. *arXiv preprint arXiv:2106.05438*, 2021. 2

[41] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 852–869. Springer, 2016. 1

[42] Zixian Ma, Jerry Hong, Mustafa Omer Gul, Mona Gandhi, Irena Gao, and Ranjay Krishna. Crepe: Can vision-language foundation models reason compositionally? *arXiv preprint arXiv:2212.07796*, 2022. 1, 2, 5, 6, 7

[43] Andre Martins and Ramon Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *International conference on machine learning*, pages 1614–1623. PMLR, 2016. 3

[44] Hossein Mobahi, Mehrdad Farajtabar, and Peter Bartlett. Self-distillation amplifies regularization in hilbert space. *Advances in Neural Information Processing Systems*, 33:3351–3361, 2020. 4

[45] Amy Perfors. Simulated evolution of language: a review of the field. *Journal of artificial societies and social simulation*, 5(2), 2002. 2, 6

[46] Steven Pinker and Paul Bloom. Natural language and natural selection. *Behavioral and brain sciences*, 13(4):707–727, 1990. 2

[47] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1, 2, 3, 4, 5

[48] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019. 4

[49] Sai Rajeswar, Pau Rodriguez, Soumye Singhal, David Vazquez, and Aaron Courville. Multi-label iterated learning for image classification with label ambiguity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4783–4793, 2022. 3

[50] Arijit Ray, Filip Radenovic, Abhimanyu Dubey, Bryan A Plummer, Ranjay Krishna, and Kate Saenko. Cola: How to adapt vision-language models to compose objects localized with attributes? *Advances in Neural Information Processing Systems*, 2023. 1, 2, 5, 6

[51] Yi Ren, Shangmin Guo, Matthieu Labeau, Shay B Cohen, and Simon Kirby. Compositional languages emerge in a neural iterated learning model. *arXiv preprint arXiv:2002.01365*, 2020. 3

[52] Mathieu Rita, Florian Strub, Jean-Bastien Grill, Olivier Pietquin, and Emmanuel Dupoux. On the role of population heterogeneity in emergent communication. *arXiv preprint arXiv:2204.12982*, 2022. 3

[53] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2

[54] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 5

[55] Catriona Silvey, Simon Kirbey, and Kenny Smith. Communication leads to the emergence of sub-optimal category structures. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, 2013. 3

[56] Catriona Silvey, Simon Kirby, and Kenny Smith. Word meanings evolve to selectively preserve distinctions on salient dimensions. *Cognitive science*, 39(1):212–226, 2015. 2

[57] Kenny Smith, Simon Kirby, and Henry Brighton. Iterated learning: A framework for the emergence of language. *Artificial life*, 9(4):371–386, 2003. 2

[58] Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pages 5238–5248, 2022. 1, 2

[59] Simon W Townsend, Sabrina Engesser, Sabine Stoll, Klaus Zuberbühler, and Balthasar Bickel. Compositionality in animals and humans. *PLoS Biology*, 16(8):e2006425, 2018. 1

[60] Ankit Vani, Max Schwarzer, Yuchen Lu, Eeshan Dhekane, and Aaron Courville. Iterated learning for emergent systematicity in vqa. *arXiv preprint arXiv:2105.01119*, 2021. 3

[61] Tessa Verhoef, Simon Kirby, and Bart De Boer. Iconicity and the emergence of combinatorial structure in language. *Cognitive science*, 40(8):1969–1994, 2016. 1

[62] Jing Xu, Mike Dowman, and Thomas L Griffiths. Cultural transmission results in convergence towards colour term universals. *Proceedings of the Royal Society B: Biological Sciences*, 280(1758):20123073, 2013. 3

[63] Fengyu Yang, Chao Feng, Ziyang Chen, Hyoungseob Park, Daniel Wang, Yiming Dou, Ziyao Zeng, Xien Chen, Rit Gangopadhyay, Andrew Owens, and Alex Wong. Binding touch to everything: Learning unified multimodal tactile representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 2

[64] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 2

[65] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *The Eleventh International Conference on Learning Representations*, 2022. 2, 5

[66] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *International Conference on Learning Representations*, 2023. 1, 2

[67] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019. 6

[68] Zhilu Zhang and Mert Sabuncu. Self-distillation as instance-specific label smoothing. *Advances in Neural Information Processing Systems*, 33:2184–2195, 2020. 4

[69] Tiancheng Zhao, Tianqi Zhang, Mingwei Zhu, Haozhan Shen, Kyusong Lee, Xiaopeng Lu, and Jianwei Yin. Vl-checklist: Evaluating pre-trained vision-language models with objects, attributes and relations. *arXiv preprint arXiv:2207.00221*, 2022. 1, 2

[70] Yiwu Zhong, Jing Shi, Jianwei Yang, Chenliang Xu, and Yin Li. Learning to generate scene graph from natural language supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1823–1834, 2021. 2