Embodied Multimodal Multitask Learning

Devendra Singh Chaplot[1] , **Lisa Lee**[1] , **Ruslan Salakhutdinov**[1] , **Devi Parikh**[2,3] , **Dhruv Batra**[2,3]
[1]Carnegie Mellon University
[2]Facebook AI Research
[3]Georgia Institute of Technology
{chaplot, lslee, rsalakhu}@cs.cmu.edu, {parikh, dbatra}@gatech.edu

Abstract

Visually-grounded embodied language learning models have recently shown to be effective at learning multiple multimodal tasks such as following navigational instructions and answering questions. In this paper, we address two key limitations of these models, (a) the inability to transfer the grounded knowledge across different tasks and (b) the inability to transfer to new words and concepts not seen during training using only a few examples. We propose a multitask model which facilitates knowledge transfer across tasks by disentangling the knowledge of words and visual attributes in the intermediate representations. We create scenarios and datasets to quantify cross-task knowledge transfer and show that the proposed model outperforms a range of baselines in simulated 3D environments. We also show that this disentanglement of representations makes our model modular and interpretable which allows for transfer to instructions containing new concepts.*

1 Introduction

Humans learn language by interacting with a dynamic perceptual environment, grounding words into visual entities and motor actions [Smith and Gasser, 2005; Barsalou, 2008]. In recent years, there has been an increased focus on training embodied agents capable of visually-grounded language learning. These include multimodal tasks involving *one-way* communication, such as mapping navigational instructions to actions [MacMahon *et al.*, 2006; Chen and Mooney, 2011; Artzi and Zettlemoyer, 2013; Mei *et al.*, 2016; Misra *et al.*, 2018]; and tasks involving *two-way* communication such as embodied question answering [Gordon *et al.*, 2018; Das *et al.*, 2018] and embodied dialogue [de Vries *et al.*, 2018]. Other studies have shown that grounded semantic goal navigation agents can be effective at exploiting the compositionality of language to generalize to unseen instructions with an unseen composition of semantic attributes [Hermann *et al.*, 2017; Chaplot *et al.*, 2018], or an unseen composition of steps in a multi-step instruction [Oh *et al.*, 2017].

However, current grounded language learning models have certain limitations. Firstly, these models are typically trained only for a single multimodal task and lack the ability to transfer grounded knowledge of 'concepts'† across tasks. For example, if an agent learns to follow the instruction 'Go to the red torch' and answer the question 'What color is the pillar?', then ideally it should also understand 'Go to the red pillar' and 'What color is the torch?' without additional training. Training multitask grounded-language models can also improve training sample efficiency, as these multimodal tasks share many common learning challenges including perception, grounding, and navigation.

The second limitation is the inability of trained models to quickly transfer to tasks involving unseen concepts. For example, consider a household instruction-following robot trained on an existing set of objects. We would like the robot to follow instructions involving a new object 'lamp' that has been added to the house. Existing models would need to be trained with the new object, which typically requires thousands of samples and can also lead to catastrophic forgetting of known objects. Even if the models were given some labeled samples to detect the new objects, they would require additional training to learn to combine existing grounded knowledge with the new concept (e.g., 'blue lamp' if 'blue' is already known).

In this paper, we train a multimodal multitask learning model for two tasks: *Semantic Goal Navigation*, where the agent is given a language instruction to navigate to a goal location, and *Embodied Question Answering*, where the agent is asked a question and it can navigate in the environment to gather information to answer the question (see Figure 1). We make the following contributions in this paper:

First, we define a *cross-task knowledge transfer* evaluation criterion to test the ability of multimodal multi-task models to transfer knowledge of concepts across tasks. We show that several prior single-task models, when trained on both tasks, fail to achieve cross-task knowledge transfer. This is because the visual grounding of words is often implicitly learned as a by-product of end-to-end training of the underlying task, which leads to the entanglement of knowledge of concepts in the learnt representations. We propose a novel Dual-Attention

*Webpage: https://devendrachaplot.github.io/projects/EMML

†In this paper, we refer to the knowledge of a word and its grounding in the visual world as the knowledge of a concept (for example, concept 'torch' involves word 'torch' and how torch looks visually).

Figure 1: Examples of embodied multimodal tasks, following instructions and answering questions.

Task	Train Set	Test Set
SGN	Instructions *not* containing 'red' & 'pillar': 'Go to the **blue** object' 'Go to the **torch**'	Instructions containing 'red' or 'pillar': 'Go to the red pillar' 'Go to the tall red object'
EQA	Questions *not* containing 'blue' & 'torch': 'Which object is red in color?' 'What color is the tall pillar?'	Questions containing 'blue' or 'torch': 'Which object is **blue** in color?' 'What color is the **torch**?'

Table 1: Table showing training and test sets for both the tasks, Semantic Goal Navigation (SGN) and Embodied Question Answering (EQA). The test set consists of unseen instructions and questions. The dataset evaluates a model for cross-task knowledge transfer the embodied multimodal tasks.

model which learns task-invariant disentangled visual and textual representations and explicitly aligns them with each other. We create datasets and simulation scenarios for testing cross-task knowledge transfer and show an absolute improvement of 43-61% on instructions and 5-26% for questions over baselines (Section 5.1).

Second, the disentanglement and explicit alignment of representations makes our model modular and interpretable. We show that this allows us to transfer the model to handle instructions involving unseen concepts by incorporating the output of object detectors. We also show that our model is able to combine the knowledge of existing concepts with a new concept without any additional policy training (Section 5.4).

Finally, we show that the modularity and interpretability of our model also allow us to use trainable neural modules [Andreas *et al.*, 2016] to handle relational tasks involving negation and spatial relationships and also tackle relational instructions involving new concepts (Section 5.3).

2 Related Work

A lot of early work on visual instruction-following in the embodied space such as in robotics applications [Tellex *et al.*, 2011; Matuszek *et al.*, 2012; Hemachandra *et al.*, 2015; Misra *et al.*, 2016] and on mapping natural language instructions to actions [MacMahon *et al.*, 2006; Chen and Mooney, 2011; Artzi and Zettlemoyer, 2013; Mei *et al.*, 2016] required hand-designed symbolic representations. Recently, there have been efforts on learning to follow navigational instructions from raw visual observations [Anderson *et al.*, 2018; Misra *et al.*, 2018; Chen *et al.*, 2019; Blukis *et al.*, 2018]. Some previous works have studied the language learning aspect of instruction-following in a more controlled setting, and show that grounded language learning agents are able to learn spatial and logical reasoning and exploit the compositionality of language to generalize to new instructions [Oh *et al.*, 2017; Chaplot *et al.*, 2018; Hermann *et al.*, 2017]

Question Answering in the embodied space has been comparatively less-studied with recent work studying QA which requires exploration, navigation, and interaction with objects in the environment [Gordon *et al.*, 2018; Das *et al.*, 2018]. In contrast to the prior work which tackles a single grounding task, we tackle both instruction-following and question answering in the embodied space and study the ability to transfer the knowledge of concepts across the tasks and tackle instructions with new concepts.

3 Problem Formulation

Consider an autonomous agent interacting with an episodic environment as shown in Figure 1. At the beginning of each episode, the agent receives a textual input T specifying a task. T could be an instruction to navigate to a target object or a question querying some visual detail of objects in the environment. At each time step t, the agent observes a state $s_t = (I_t, T)$ where I_t is the first-person (egocentric) view of the environment, and takes an action a_t, which could be a navigational action or an answer action. The agent's objective is to learn a policy $\pi(a_t|s_t)$ which leads to successful completion of the task specified by T.

Environments. We adapt the ViZDoom-based [Kempka *et al.*, 2016] language grounding environment proposed by Chaplot *et al.* [2018] for embodied multitask learning. It consists of a single room with 5 objects. The objects are randomized in each episode based on the textual input. We use two difficulty settings: *Easy*: The candidate objects are in the field of view of the agent at the beginning of the episode. *Hard*: The candidate objects and the agent are dropped at random locations and the objects may or may not be in the agent's field of view in the initial configuration. The agent must explore the map to view all objects. The agent can take 4 actions: 3 navigational actions (forward, left, right) and 1 answer action. When the agent takes the answer action, the answer with the maximum probability in the output answer distribution is used.

Datasets. We use the set of objects and attributes from Chaplot *et al.* [2018] and create a dataset which includes instructions and questions about object types, colors, relative sizes (tall/short) and superlative sizes (smallest/largest). We create train-test splits for both instructions and questions datasets to explicitly test a multitask model's ability to transfer the knowledge of concepts across different tasks. Each instruction in the test set contains a word that is never seen in any instruction in the training set but is seen in some questions in the training set. Similarly, each question in the test set contains a word never seen in any training set question. Table 1 illustrates the train-test split of instructions and questions.

4 Proposed Method

In this section, we describe our proposed architecture (illustrated in Figure 2). At the start of each episode, the agent receives a textual input T (an instruction or a question) specifying the task. At each time step t, the agent observes an egocentric image I_t which is passed through a convolu-

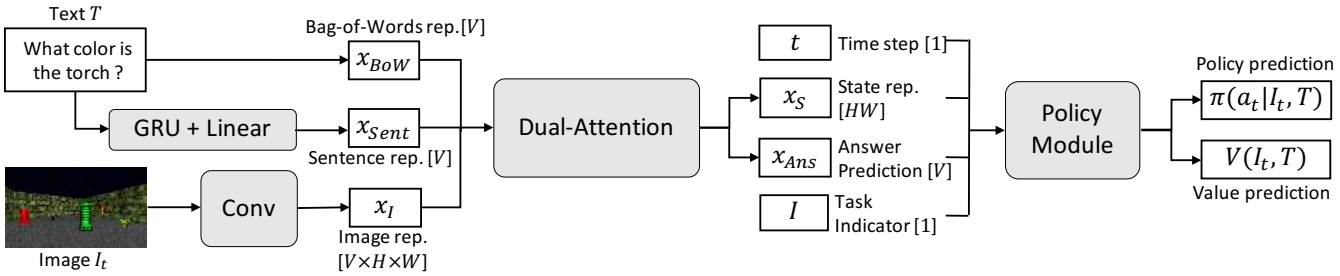

Figure 2: Overview of our proposed architecture, described in detail in Section 4.

tional neural network [LeCun *et al.*, 1995] with ReLU activations [Glorot *et al.*, 2011] to produce the image representation $x_I = f(I_t; \theta_{\text{conv}}) \in \mathbb{R}^{V \times H \times W}$, where θ_{conv} denotes the parameters of the convolutional network, V is the number of feature maps in the convolutional network output which is by design set equal to the vocabulary size (of the union of the instructions and questions training sets), and H and W are the height and width of each feature map. We use two representations for the textual input T: (1) the bag-of-words representation denoted by $x_{\text{BoW}} \in 0, 1^V$ and (2) a sentence representation $x_{\text{sent}} = f(T; \theta_{\text{sent}}) \in \mathbb{R}^V$, which is computed by passing the words in T through a Gated Recurrent Unit (GRU) [Cho *et al.*, 2014] network followed by a linear layer. Here, θ_{sent} denotes the parameters of the GRU network and the linear layer with ReLU activations. Next, the Dual-Attention unit f_{DA} combines the image representation with the text representations to get the complete state representation x_{S} and answer prediction x_{Ans}:

$$x_{\text{S}}, x_{\text{Ans}} = f_{\text{DA}}(x_I, x_{\text{BoW}}, x_{\text{sent}}) \tag{1}$$

Finally, x_S and x_{Ans}, along with a time step embedding and a task indicator variable (for whether the task is SGN or EQA), are passed to the policy module to produce an action.

4.1 Dual-Attention Unit

The Dual-Attention unit uses two types of attention mechanisms, Gated-Attention f_{GA} and Spatial-Attention f_{SA}, to align representations in different modalities and tasks.

Gated-Attention (GA). The GA unit [Chaplot *et al.*, 2018] attends to the different channels in the image representation based on the text representation. For example, if the textual input is the instruction 'Go to the red pillar', then the GA unit can learn to attend to channels which detect red things and pillars. Specifically, the GA unit takes as input a 3-dimensional tensor image representation $y_I \in \mathbb{R}^{d \times H \times W}$ and a text representation $y_T \in \mathbb{R}^d$, and outputs a 3-dimensional tensor $z \in \mathbb{R}^{d \times H \times W}$. Note that the dimension of y_T is equal to the number of feature maps and the size of the first dimension of y_I. In the GA unit, each element of y_T is expanded to a $H \times W$ matrix, resulting in a 3-dimensional tensor $M_{y_T} \in \mathbb{R}^{d \times H \times W}$, whose $(i, j, k)^{th}$ element is given by $M_{y_T}[i, j, k] = y_T[i]$. This matrix is multiplied element-wise with the image representation: $z = f_{\text{GA}}(y_I, y_T) = M_{y_T} \odot y_I$, where $\odot$ denotes the Hadamard product.

Spatial-Attention (SA). We propose an SA unit which is analogous to the Gated-Attention unit except that it attends to different *pixels* in the image representation rather than the channels. For example, if the textual input is the question 'Which object is blue in color?', then we would like to spatially attend to the parts of the image which contain a blue object in order to recognize the type of the blue object. The Spatial-Attention unit takes as input a 3-dimensional tensor image representation $y_I \in \mathbb{R}^{d \times H \times W}$ and a 2-dimensional spatial attention map $y_S \in \mathbb{R}^{H \times W}$, and outputs a tensor $z \in \mathbb{R}^{d \times H \times W}$. Note that the height and width of the spatial attention map are equal to the height and width of the image representation. In the spatial-attention unit, each element of the spatial attention map is expanded to a d dimensional vector. This again results in a 3-dimensional tensor $M_{y_S} \in \mathbb{R}^{d \times H \times W}$, whose $(i, j, k)^{th}$ element is given by: $M_{y_S}[i, j, k] = y_S[j, k]$. Just like in the Gated-Attention unit, this matrix is multiplied element-wise with the image representation: $z = f_{\text{SA}}(y_I, y_S) = M_{y_S} \odot y_I$.

Dual-Attention. We now describe the operations in the Dual-Attention unit shown in Figure 3, as well as motivate the intuitions behind each operation. Given x_I, x_{BoW}, and x_{sent}, the Dual-Attention unit first computes a Gated-Attention over x_I using x_{BoW}:

$$x_{\text{GA1}} = f_{\text{GA}}(x_I, x_{\text{BoW}}) \in \mathbb{R}^{V \times H \times W} \tag{2}$$

Intuitively, this GA unit grounds each word in the vocabulary with a feature map in the image representation. A particular feature map is activated if and only if the corresponding word occurs in the textual input. Thus, the feature maps in the convolutional output learn to detect different objects and attributes, and words in the textual input specify which objects and attributes are relevant to the current task. The Gated-Attention using BoW representation attends to feature maps detecting corresponding objects and attributes, and masks all other feature maps. We use the BoW representation for the first GA unit as it explicitly aligns the words in textual input irrespective of whether it is a question or an instruction.

Next, the output of the GA unit x_{GA1} is converted to a spatial attention map by summing over all channels followed by a softmax over $H \times W$ elements:

$$x_{\text{spat}} = \sigma\left(\sum_i^V x_{\text{GA1}}[i, :, :]\right) \in \mathbb{R}^{H \times W} \tag{3}$$

where the softmax $\sigma(z)_j = \exp(z_j) / \sum_k \exp(z_k)$ ensures that the attention map is spatially normalized. Summation of x_{GA1} along the depth dimension gives a spatial attention map which has high activations at spatial locations where

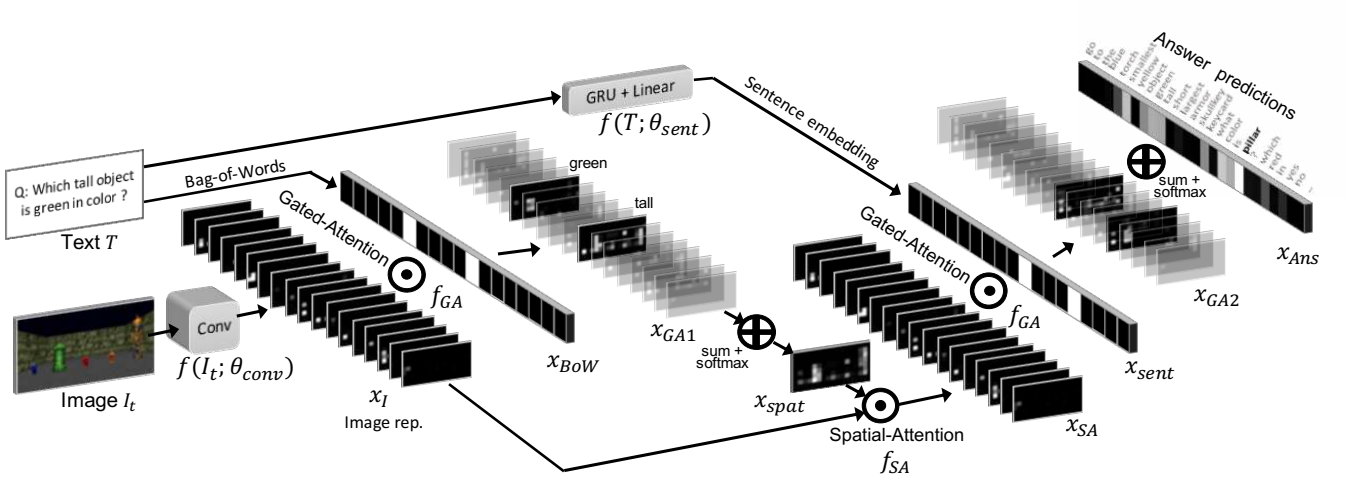

Figure 3: Architecture of the **Dual-Attention** unit with example intermediate representations and operations.

relevant objects or attributes are detected. ReLU activations in the convolutional feature maps makes all elements positive, ensuring that the summation aggregates the activations of relevant feature maps.

x_{spat} and x_I are then passed through a SA unit:

$$x_{\text{SA}} = f_{\text{SA}}(x_I, x_{\text{spat}}) \in \mathbb{R}^{V \times H \times W} \tag{4}$$

The SA unit outputs all attributes present at the locations where relevant objects and attributes are detected. This is especially helpful for question answering, where a single Gated-Attention may not be sufficient. For example, if the textual input is 'Which color is the pillar?', then the model needs to attend not only to feature maps detecting pillars (done by the Gated-Attention), but also to other attributes at the spatial locations where pillars are seen in order to predict their color.

x_{SA} is then passed through another GA unit with the sentence-level text representation:

$$x_{\text{GA2}} = f_{\text{GA}}(x_{\text{SA}}, x_{\text{sent}}) \in \mathbb{R}^{V \times H \times W} \tag{5}$$

This second GA unit enables the model to attend to different types of attributes based on the question. For instance, if the question is asking about the color ('Which color is the pillar?'), then the model needs to attend to the feature maps corresponding to colors; or if the question is asking about the object type ('Which object is green in color?'), then the model needs to attend to the feature maps corresponding to object types. The sentence embedding x_{sent} can learn to attend to multiple channels based on the textual input and mask the rest.

Next, the output is transformed to answer prediction by again doing a summation and softmax but this time summing over the height and width instead of the channels:

$$x_{\text{Ans}} = \sigma\left(\sum_{j,k}^{H,W} x_{\text{GA2}}[:, j, k]\right) \in \mathbb{R}^{V} \tag{6}$$

Summation of x_{GA2} along each feature map aggregates the activations for relevant attributes spatially. Again, ReLU activations for sentence embedding ensure aggregation of activations for each attribute or word. The answer space is identical to the textual input space $\mathbb{R}^V$.

Finally, the Dual-Attention unit f_{DA} outputs the answer prediction x_{Ans} and the flattened spatial attention map $x_{\text{S}} = \text{vec}(x_{\text{spat}})$, where $\text{vec}(\cdot)$ denotes the flattening operation.

Policy Module. The policy module takes as input the state representation x_{S} from the Dual-Attention unit, a time step embedding t, and a task indicator variable I (for whether the task is SGN or EQA). The inputs are concatenated then passed through a linear layer, then a recurrent GRU layer, then linear layers to estimate the policy function $\pi(a_t \mid I_t, T)$ and the value function $V(I_t, T)$.

All above operations are differentiable, making the entire architecture trainable end-to-end. Note that all attention mechanisms in the Dual-Attention unit only modulate the input image representation, i.e., mask or amplify specific feature maps or pixels. This ensures that there is an explicit alignment between the words in the textual input, the feature maps in the image representation, and the words in the answer space. This forces the convolutional network to encode all the information required with respect to a certain word in the corresponding output channel. For example, to predict 'red' as the answer, the model must detect red objects in the corresponding feature map. This explicit task-invariant alignment between convolutional feature maps and words in the input and answer space facilitates grounding and allows for cross-task knowledge transfer. As shown in the results later, this also makes our model modular and allows easy addition of objects and attributes to a trained model.

Optimization. The entire model is trained to predict both navigational actions and answers jointly. The policy is trained using Proximal Policy Optimization (PPO) [Schulman *et al.*, 2017]. For training the answer predictions, we use a supervised cross-entropy loss.

Auxiliary Task. As mentioned earlier, the feature maps in the convolutional output are expected to detect different objects and attributes. We add a spatial auxiliary task (trained with cross-entropy loss) to detect the object or attribute in the convolutional output channels corresponding to the word in the bag-of-words representation.

	Easy						Hard					
	No Aux			Aux			No Aux			Aux		
	Train	Test		Train	Test		Train	Test		Train	Test	
Model	MT	SGN	EQA	MT	SGN	EQA	MT	SGN	EQA	MT	SGN	EQA
Text only	0.33	0.20	0.33	0.31	0.20	0.33	0.36	0.20	0.33	0.36	0.20	0.33
Image only	0.41	0.20	0.09	0.40	0.21	0.08	0.36	0.16	0.08	0.36	0.15	0.08
Concat	0.97	0.33	0.21	**0.99**	0.31	0.19	0.57	0.20	0.26	0.71	0.39	0.22
GA	0.97	0.27	0.18	**0.99**	0.35	0.24	0.44	0.18	0.11	0.71	0.22	0.24
FiLM	0.97	0.24	0.11	**0.99**	0.34	0.12	0.52	0.12	0.03	0.55	0.25	0.15
PACMAN	0.66	0.26	0.12	0.79	0.33	0.10	0.56	0.29	0.33	0.54	0.11	0.27
Dual-Attention	**0.99**	**0.86**	**0.53**	**0.99**	**0.96**	**0.58**	**0.85**	**0.86**	**0.38**	**0.90**	**0.82**	**0.59**

Table 2: Accuracy of all models for both *Easy* & *Hard* difficulties. 'MT' stands for multi-task.

5 Experiments & Results

Jointly learning semantic goal navigation and embodied question answering essentially involves a fusion of textual and visual modalities. While prior methods are designed for a single task, we adapt several baselines for our environment and tasks by using their multimodal fusion techniques. We use two naive baselines, **Image only** and **Text only**; two baselines based on prior semantic goal navigation models, **Concat** (used by [Hermann *et al.*, 2017; Misra *et al.*, 2017]) and **Gated-Attention** (GA) [Chaplot *et al.*, 2018]; and two baselines based on Question Answering models, **FiLM** [Perez *et al.*, 2018] and **PACMAN** [Das *et al.*, 2018]. For fair comparison, we replace the proposed Dual-Attention unit with multimodal fusion techniques in the baselines and keep everything else identical to the proposed model. We will open-source the code for the training environment, datasets, and model implementation including all hyper-parameter details to support reproducbility and future work in this direction.

5.1 Results

We train all models for 10 million frames in the *Easy* setting and 50 million frames in the *Hard* setting. We use a +1 reward for reaching the correct object in SGN episodes and predicting the correct answer in EQA episodes. We use a small negative reward of -0.001 per time step to encourage shorter paths to the target and answering questions as soon as possible. We also use distance-based reward shaping for SGN episodes, where the agent receives a small reward proportional to the decrease in distance to the target. In the next subsection, we evaluate the performance of the proposed model without the reward shaping. SGN episodes end when the agent reaches any object, and EQA episodes end when the agent predicts any answer. All episodes have a maximum length of 210 time steps. We train all models with and without the auxiliary tasks using identical reward functions.

All models are trained jointly for both the tasks and tested on each task separately. In Table 2, we report the performance of all models for both *Easy* and *Hard* settings. The Dual-Attention (DA) model and many baselines achieve 99% accuracy during training in the Easy-Aux setting; however, the test performance of all the baselines is considerably lower than that of the DA model (see Table 2 (left)). Performance of all the baselines is worse than the 'Text only' model on the EQA test set, although the training accuracy is higher. This indicates that baselines tend to overfit on the training set and fail to generalize to questions which contain words never seen in training questions. As expected, using spatial auxiliary tasks improves performance of all models. Even without auxiliary tasks, the DA model achieves a test accuracy 86% (SGN) and 53% (EQA), compared to the best baseline performance of 33% (SGN & EQA).

	No Aux		Aux	
Model	SGN	EQA	SGN	EQA
w/o SA	0.20	0.16	0.20	0.15
w/o GA1	0.14	0.25	0.16	0.38
w/o GA2	0.80	0.33	0.97	0.15
w/o Task Indicator	0.79	0.47	0.96	0.56
w/o Reward Shaping	0.82	0.49	0.93	0.51
DA Single-Task	0.63	0.31	0.91	0.34
DA Multi-Task	0.86	0.53	0.96	0.58

Table 3: Accuracy of all the ablation models on SGN and EQA test sets for the Easy setting.

For the Hard setting, the DA model achieves a higher training (90% vs 71% with Aux) as well as test performance (82% vs. 39% for SGN, 59% vs. 33% for EQA with Aux) than the baselines (see Table 2 (right)). These results confirm the hypothesis that prior models, which are designed for a single task, lack the ability to align the words in both the tasks and transfer knowledge across tasks.

5.2 Ablation tests

We perform several ablation tests to analyze the contribution of each component in the Dual-Attention unit: without Spatial-Attention (**w/o SA**), without the first Gated-Attention with x_{BoW} (**w/o GA1**), and without the second Gated-Attention with x_{sent} (**w/o GA2**). We also try removing the task indicator variable (**w/o Indicator Variable**), removing reward shaping (**w/o Reward Shaping**), and training the proposed model on a single task, SGN or EQA (**DA Single-Task**).

In Table 3, we report the test performance of all ablation models. The results indicate that SA and GA1 contribute the most to the performance of the full Dual-Attention model. GA2 is critical for performance on EQA but not SGN (see Table 3). This is expected as GA2 is designed to attend to different objects and attributes based on the question and is used mainly for answer prediction. It is not critical for SGN

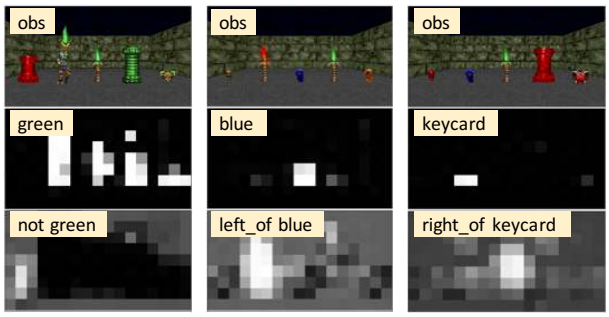

Figure 4: Outputs for relations 'not', 'left of', and 'right of' learned by the relational modules.

Instruction	Acc
Go to the **red** object	0.99
Go to the <color_name> **pillar**.	1.00
Go to the **red** <object_name>	1.00
Go to the largest/smallest **red** object	0.95
Go to the tall/short **red pillar**	0.99
Go to the **red pillar**	0.99
Go to the <color_name> object that is not a **pillar**	0.91
Go to the <object_name> that is left of the **red** object	0.96
Go to the **red** object that is right of the **pillar**	0.95

Table 4: The performance of a trained policy appended with object detectors on instructions containing unseen words ('red' and 'pillar').

as the spatial attention map consists of locations of relevant objects, which is sufficient for navigating to the correct object.

We observe that reward shaping and indicator variable help with learning speed, but have little effect on the final performance (see Table 3). DA models trained only on single tasks work well on SGN, especially with auxiliary tasks, because the auxiliary task for single task models includes object detection labels corresponding to the words in the test set.

5.3 Handling Relational Tasks

The instructions and questions considered so far contained a single target object. We propose a simple extension to our model to handle *relational tasks*, such as 'Which object is to the left of the torch?', where the agent is required to attend to the region *left of* the torch, not the torch itself.

We consider three relational operations: 'left of', 'right of' and 'not'. We add questions and instructions with all objects and attributes using these relational operations to the existing dataset and perform experiments in the Easy-Aux setting. We assume that the knowledge of relational words, and the words they modify, are given. We train a separate module corresponding to each relational operation, and apply it to the convolutional output of the words that are modified. For example, for the above question, we apply the module for relation 'left of' to the convolutional output channel corresponding to the word 'torch'. Each relational module is a trainable convolutional network which preserves the size of the input. The rest of the operations are identical to the Dual-Attention Unit. The relational modules are learned end-to-end without any additional supervision.

In Figure 4, we show convolutional outputs of the relational modules learned by our model. While the original DA model achieves test performance of 0.48 (SGN) and 0.44 (EQA), this simple extension achieves 0.97 (SGN) and 0.64 (EQA).

5.4 Transfer To New Concepts

Suppose that the user wants the agent to follow instructions about a new object such as 'pillar' or a new attribute such as 'red' which the agent has never seen during training. Prior SGN models [Chaplot *et al.*, 2018; Hermann *et al.*, 2017; Yu *et al.*, 2018] cannot handle instructions containing a new concept. In contrast, our model can be used for handling such instructions by using an object detector for each new concept. In order to test this, we train the DA model in the Easy setting on the training set for only instructions. We use auxiliary tasks but only for words in the vocabulary of the instructions training set. After training the policy, we test the agents on instructions containing test concept words 'red' and 'pillar', which the agent has never seen in textual input during training and never received any supervision about how this attribute or object looks visually.

For transferring the policy, we assume access to two object detectors for 'red' and 'pillar' separately. We append the object detections for the new concepts to the image representation x_I. We also append the words 'red' and 'pillar' to the bag-of-words representation in the same order such that they are aligned with the appended feature maps.

The results in Table 4 show that this policy generalizes well to different types of instructions with unseen concepts, including: combining knowledge of existing attributes with a new object, or knowledge of existing objects with a new attribute; and composing a new attribute with a new object. The results shown in the lower part of Table 4 indicate that the model also generalizes well to relational instructions containing new concepts. This means that given an object detector for a new object 'pillar', the model can (without any additional training) detect and differentiate between green and blue pillars, or between tall and short pillars; and understand left of/right of pillar. The model can also combine 'pillar' with another new attribute 'red' to detect red pillars and understand relational instructions involving both red objects and pillars. This suggests that a trained policy can be scaled to more objects provided the complexity of navigation remains consistent.

6 Conclusion

We proposed a Dual-Attention model for visually-grounded multitask learning which uses Gated- and Spatial-Attention to disentangle attributes in feature representations and align them with the answer space. We show that the proposed model is able to transfer the knowledge of concepts across tasks and outperforms the baselines on both Semantic Goal Navigation and Embodied Question Answering by a considerable margin. We showed that disentangled and interpretablew representations make our model modular and allow for easy addition of new objects or attributes to a trained model. For future work, the model can potentially be extended to transferring knowledge across different domains by using modular interpretable representations of objects which are domain-invariant.

References

[Anderson *et al.*, 2018] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3674–3683, 2018.

[Andreas *et al.*, 2016] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 39–48, 2016.

[Artzi and Zettlemoyer, 2013] Yoav Artzi and Luke Zettlemoyer. Weakly supervised learning of semantic parsers for mapping instructions to actions. *Transactions of the Association for Computational Linguistics*, 1:49–62, 2013.

[Barsalou, 2008] Lawrence W Barsalou. Grounded cognition. *Annu. Rev. Psychol.*, 59:617–645, 2008.

[Blukis *et al.*, 2018] Valts Blukis, Dipendra Misra, Ross A Knepper, and Yoav Artzi. Mapping navigation instructions to continuous control actions with position-visitation prediction. *Proceedings of The 2nd Conference on Robot Learning*, 2018.

[Chaplot *et al.*, 2018] Devendra Singh Chaplot, Kanthashree Mysore Sathyendra, Rama Kumar Pasumarthi, Dheeraj Rajagopal, and Ruslan Salakhutdinov. Gated-attention architectures for task-oriented language grounding. In *AAAI*, 2018.

[Chen and Mooney, 2011] David L Chen and Raymond J Mooney. Learning to interpret natural language navigation instructions from observations. In *Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.

[Chen *et al.*, 2019] Howard Chen, Alane Shur, Dipendra Misra, Noah Snavely, and Yoav Artzi. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *CVPR*, 2019.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[Das *et al.*, 2018] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *CVPR*, 2018.

[de Vries *et al.*, 2018] Harm de Vries, Kurt Shuster, Dhruv Batra, Devi Parikh, Jason Weston, and Douwe Kiela. Talk the walk: Navigating new york city through grounded dialogue. *arXiv preprint arXiv:1807.03367*, 2018.

[Glorot *et al.*, 2011] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323, 2011.

[Gordon *et al.*, 2018] Daniel Gordon, Aniruddha Kembhavi, Mohammad Rastegari, Joseph Redmon, Dieter Fox, and Ali Farhadi. Iqa: Visual question answering in interactive environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4089–4098, 2018.

[Hemachandra *et al.*, 2015] Sachithra Hemachandra, Felix Duvallet, Thomas M Howard, Nicholas Roy, Anthony Stentz, and Matthew R Walter. Learning models for following natural language directions in unknown environments. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5608–5615. IEEE, 2015.

[Hermann *et al.*, 2017] Karl Moritz Hermann, Felix Hill, Simon Green, Fumin Wang, Ryan Faulkner, Hubert Soyer, David Szepesvari, Wojtek Czarnecki, Max Jaderberg, Denis Teplyashin, et al. Grounded language learning in a simulated 3d world. *arXiv preprint arXiv:1706.06551*, 2017.

[Kempka *et al.*, 2016] Michał Kempka, Marek Wydmuch, Grzegorz Runc, Jakub Toczek, and Wojciech Jaśkowski. Vizdoom: A doom-based ai research platform for visual reinforcement learning. In *2016 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–8. IEEE, 2016.

[LeCun *et al.*, 1995] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[MacMahon *et al.*, 2006] Matt MacMahon, Brian Stankiewicz, and Benjamin Kuipers. Walk the talk: Connecting language, knowledge, and action in route instructions. *Def*, 2(6):4, 2006.

[Matuszek *et al.*, 2012] Cynthia Matuszek, Nicholas FitzGerald, Luke Zettlemoyer, Liefeng Bo, and Dieter Fox. A joint model of language and perception for grounded attribute learning. *arXiv preprint arXiv:1206.6423*, 2012.

[Mei *et al.*, 2016] Hongyuan Mei, Mohit Bansal, and Matthew R Walter. Listen, attend, and walk: Neural mapping of navigational instructions to action sequences. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[Misra *et al.*, 2016] Dipendra K Misra, Jaeyong Sung, Kevin Lee, and Ashutosh Saxena. Tell me dave: Context-sensitive grounding of natural language to manipulation instructions. *The International Journal of Robotics Research*, 35(1-3):281–300, 2016.

[Misra *et al.*, 2017] Dipendra Misra, John Langford, and Yoav Artzi. Mapping instructions and visual observations to actions with reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1004–1015, 2017.

[Misra *et al.*, 2018] Dipendra Misra, Andrew Bennett, Valts Blukis, Eyvind Niklasson, Max Shatkhin, and Yoav Artzi. Mapping instructions to actions in 3d environments with visual goal prediction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2667–2678, 2018.

[Oh *et al.*, 2017] Junhyuk Oh, Satinder Singh, Honglak Lee, and Pushmeet Kohli. Zero-shot task generalization with multi-task deep reinforcement learning. In *ICML*, 2017.

[Perez *et al.*, 2018] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

[Schulman *et al.*, 2017] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[Smith and Gasser, 2005] Linda Smith and Michael Gasser. The development of embodied cognition: Six lessons from babies. *Artificial life*, 11(1-2):13–29, 2005.

[Tellex *et al.*, 2011] Stefanie A Tellex, Thomas Fleming Kollar, Steven R Dickerson, Matthew R Walter, Ashis Banerjee, Seth Teller, and Nicholas Roy. Understanding natural language commands for robotic navigation and mobile manipulation. In *AAAI*, 2011.

[Yu *et al.*, 2018] Haonan Yu, Xiaochen Lian, Haichao Zhang, and Wei Xu. Guided feature transformation (gft): A neural language grounding module for embodied agents. In *Conference on Robot Learning*, pages 81–98, 2018.