Foundation Models for History Compression in Reinforcement Learning

Fabian Paischer [1], **Thomas Adler** [1], **Andreas Radler** [1], **Markus Hofmarcher** [2], **Sepp Hochreiter** [1 3]
[1] ELLIS Unit Linz and LIT AI Lab, Institute for Machine Learning,
[2] JKU LIT SAL eSPML Lab, Institute for Machine Learning,
Johannes Kepler University, Linz, Austria
[3] Institute of Advanced Research in Artificial Intelligence (IARAI), Vienna, Austria
paischer@ml.jku.at

Abstract

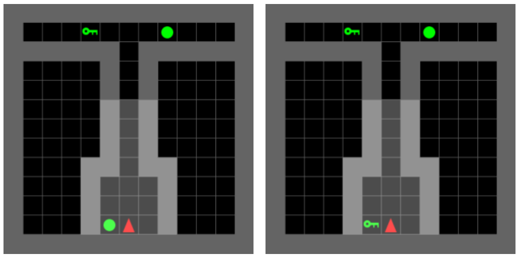
Agents interacting under partial observability require access to past observations via a memory mechanism in order to approximate the true state of the environment. Recent work suggests that leveraging language as abstraction provides benefits for creating a representation of past events. History Compression via Language Models (HELM) leverages a pretrained Language Model (LM) for representing the past. It relies on a randomized attention mechanism to translate environment observations to token embeddings. In this work, we show that the representations resulting from this attention mechanism can collapse under certain conditions. This causes blindness of the agent to subtle changes in the environment that may be crucial for solving a certain task. We propose a solution to this problem consisting of two parts. First, we improve upon HELM by substituting the attention mechanism with a feature-wise centering-and-scaling operation. Second, we take a step toward semantic history compression by leveraging foundation models, such as CLIP, to encode observations, which further improves performance. By combining foundation models, our agent is able to solve the challenging MiniGrid-Memory environment. Surprisingly, however, our experiments suggest that this is not due to the semantic enrichment of the representation presented to the LM, but rather due to the discriminative power provided by CLIP. We make our code publicly available at https://github.com/ml-jku/helm.

1 Introduction

In Reinforcement Learning (RL) an agent interacts with an environment and learns from feedback provided in the form of a reward function. RL agents that are deployed in the real world often have to cope with partial observability. Therefore, the capability to approximate the true state of their surrounding environment by virtue of an agent's perception is crucial (Åström, 1964; Kaelbling et al., 1998). To this end, many agents employ a memory mechanism to track events that occurred in the past. In this memory, it is much more efficient to store abstract representations of the past rather than every detail the agent encountered. Thus, memory mechanisms such as LSTM (Hochreiter & Schmidhuber, 1997) or Transformer (Vaswani et al., 2017) compress sequences of high-dimensional observations.

We propose to use foundation models (FM; Bommasani et al., 2021) as memory mechanism. Since FMs come pretrained, they allow for a much higher sample efficiency than memory mechanisms trained from scratch. Moreover, FMs have demonstrated remarkable few-shot capabilities (Brown et al., 2020) and can learn abstract symbolic rules and perform reasoning (Petroni et al., 2019; Talmor et al., 2020; Kassner et al., 2020). In this work, we aim to leverage these properties, which FMs

Deep Reinforcement Learning Workshop at Neural Information Processing Systems, 2022.



β	Hellinger distance δ
1	0.00036 ± 1.88e-9
10	0.0036 ± 1.84e-7
100	0.035 ± 2.09e-5
1000	0.091 ± 0.01
1e4	0.093 ± 0.06

Figure 1: **Left:** Instances of the MiniGrid-Memory environment. The agent only observes the white shaded region, and must navigate to the object seen in the starting room. **Right:** Hellinger distance (Hellinger, 1909) of softmax distributions over token embeddings for the two observations of the Memory environment depicted on the left. Mean over 1000 different initializations of $\boldsymbol{P}$ of FH is shown. For very high values of $\beta > 1\text{e}4$, the softmax converges to a one-hot encoding and δ directly corresponds to the probability of mapping to a single, but different token embedding. Therefore, with higher values of β, there is only a ~10% chance of avoiding collapse to the same token embedding.

learned on large-scale text corpora, for RL. We believe that language is well suited as domain for a memory mechanism, since it appears to be an essential component in forming meaningful abstractions. In humans, the ability to abstract is heavily influenced by the exposure to language in early childhood (Waxman & Markow, 1995). Recently, there has been a surge of complex models that combine multiple modalities, in particular images and text, as in CLIP (Radford et al., 2021). Since many RL environments rely on visual inputs, we use CLIP in our proposed memory mechanism to map visual inputs meaningfully into the language domain.

Prior work has illustrated that foundation models pretrained on language can efficiently compress sequences of observations and facilitate agent learning in partially observable RL environments (HELM, Paischer et al., 2022). The key challenge hereby is to map image-based observations to language representations. HELM tackles this problem using a randomly chosen mapping called FrozenHopfield (FH) that is — due to its randomness — inherently unable to form meaningful abstractions. In fact, we show that under certain conditions the representations are prone to collapse, rendering the agent unable to distinguish between different inputs (see Fig. 1). We build upon HELM by (i) substituting FH with a feature-wise centering-and-scaling operation and (ii) incorporating a CLIP image encoder (Radford et al., 2021), which is pretrained in a multimodal fashion on web data consisting of images and text. We term the resulting new method HELMv2. Using HELMv2, we are able to distinguish even minute differences in the input when necessary, which drastically enhances downstream performance.

We demonstrate the effectiveness of HELMv2 on a set of partially observable environments. Concretely, we train on 2D MiniGrid (Chevalier-Boisvert et al., 2018), and 3D MiniWorld environments (Chevalier-Boisvert, 2018). HELMv2 yields significant improvements over HELM in all environments. Further, we conduct ablation studies, which show that the improvements are not only due to the CLIP image encoder but also due to the replacement of the FH. Finally, we construct a mapping that successfully conveys the semantics extracted by CLIP to the LM. Surprisingly, however, using this mapping does not further improve the results in the selected environments.

2 Methods

HELM has demonstrated that pretrained LMs are well suited for compressing past observations that are randomly mapped to language tokens. In this regard, HELM performs two different forms of compression: (i) spatial compression, and (ii) temporal compression. The former is realized with the FH mechanism, while the latter is performed with a pretrained TransformerXL (TrXL, Dai et al., 2019). The spatial compression via FH consists of a random projection matrix $\boldsymbol{P}$ and an attention mechanism over pretrained token embeddings (VocabAttn). More formally, let $\boldsymbol{E} = (\boldsymbol{e}_1, \dots, \boldsymbol{e}_k)^\top \in \mathbb{R}^{k \times m}$ be the token embedding matrix of the pretrained LM consisting of k embeddings $\boldsymbol{e}_i \in \mathbb{R}^m$. At every timestep t, we obtain inputs $\boldsymbol{x}_t \in \mathbb{R}^m$ for the LM from observations $\boldsymbol{o}_t \in \mathbb{R}^n$ via the FH mechanism by

$$\boldsymbol{x}_t^\top = \sigma(\beta \boldsymbol{o}_t^\top \boldsymbol{P}^\top \boldsymbol{E}^\top)\boldsymbol{E}, \tag{1}$$

where σ is the softmax function and $\boldsymbol{P} \in \mathbb{R}^{m \times n}$ has entries sampled from $\mathcal{N}(0, n/m)$. The resulting $\boldsymbol{x}_t$ lies in the convex hull of the token embeddings of the LM. The parameter β is a scaling factor that controls the dispersion of $\boldsymbol{x}_t$ within that convex hull.

HELMcs Our aim is to avoid representation collapse caused by the FH mechanism. We illustrate in Fig. 1 that FH is prone to representation collapse even for higher values of β, especially if observations tend to be visually similar. To sidestep this issue, we substitute the attention mechanism in FH with a feature-wise centering-and-scaling operation. Let $\mathcal{B}$ be a batch of observations $\boldsymbol{o}_t$ and let $\boldsymbol{\mu}_{\mathcal{B}} \in \mathbb{R}^n$ be its mean feature vector and $\boldsymbol{\sigma}_{\mathcal{B}} \in \mathbb{R}^n$ the vector of standard deviations both taken over $\mathcal{B}$. Likewise, let $\boldsymbol{\mu}_E \in \mathbb{R}^m$ and $\boldsymbol{\sigma}_E \in \mathbb{R}^m$ be means and standard deviations of the token embeddings $\boldsymbol{E}$. Then we compute $\boldsymbol{x}_t$ as

$$\boldsymbol{x}_t = \operatorname{diag}(\boldsymbol{\sigma}_E)\boldsymbol{P}\operatorname{diag}(\boldsymbol{\sigma}_{\mathcal{B}})^{-1}(\boldsymbol{o}_t - \boldsymbol{\mu}_{\mathcal{B}}) + \boldsymbol{\mu}_E, \tag{2}$$

where $\operatorname{diag}(\cdot)$ takes a vector and constructs a diagonal matrix from it. Intuitively, the centering and scaling in the observation space pronounces differences between single observations. Subsequently, $\boldsymbol{P}$ approximately preserves the distances between the centered-and-scaled observations according to the Johnson-Lindenstrauss lemma (Johnson & Lindenstrauss, 1984). Finally, to obtain $\boldsymbol{x}_t$ we shift the observations to the language space by adding $\boldsymbol{\mu}_E$ and scaling by $\boldsymbol{\sigma}_E$. We refer to this setting as HELMcs.

HELMv2 We obtain another setting by encoding the observations with the ResNet-50 CLIP image encoder, which has the same output dimension as the embeddings $\boldsymbol{e}_i$ used by TrXL. Therefore we eliminate the need for the random mapping $\boldsymbol{P}$. Intuitively, performing the centering-and-scaling operation in the abstract CLIP space pronounces differences between different concepts encoded by CLIP. We believe this is more effective than looking at differences in the raw pixel space. That is, we let $\boldsymbol{z}_t = \text{CLIP}(\boldsymbol{o}_t) \in \mathbb{R}^m$ and construct $\mathcal{B}_\phi$ from $\boldsymbol{z}_t$ and, consequently, $\boldsymbol{\mu}_{\mathcal{B}_\phi} \in \mathbb{R}^m$ and $\boldsymbol{\sigma}_{\mathcal{B}_\phi} \in \mathbb{R}^m$. We compute the LM inputs as

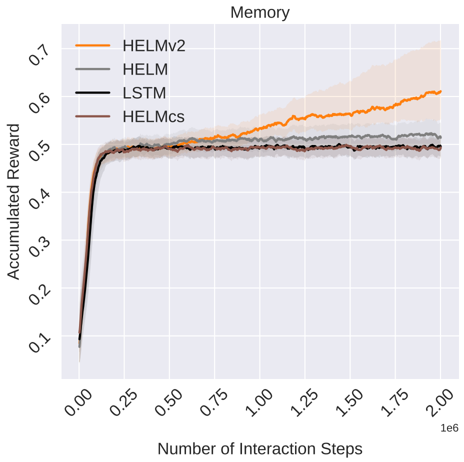
$$\boldsymbol{x}_t = \operatorname{diag}(\boldsymbol{\sigma}_E)\operatorname{diag}(\boldsymbol{\sigma}_{\mathcal{B}_\phi})^{-1}(\boldsymbol{z}_t - \boldsymbol{\mu}_{\mathcal{B}_\phi}) + \boldsymbol{\mu}_E. \tag{3}$$

The complexity imposed by the ResNet-50 CLIP encoder is negligible since it is kept frozen and only utilized during inference.

3 Experimental Results

We investigate the limitations of HELM on the MiniGrid-MemoryS11-v0 environment (Memory, Fig. 1, left). The task for the agent is to remember the object in the room it spawns in, after navigating through a corridor. The corridor ends at a T-junction showing two objects in each direction, one of which is equivalent to the object in the starting room. The agent then has to choose the direction towards the object in the starting room. If the agent chooses the wrong direction the episode ends yielding no reward. However, the objects only slightly differ in shape (green ball vs. green key). The agent receives a partially observable egocentric view of the environment in the form of RGB images. We demonstrate that FH collapses to the same representation for both objects by measuring the distances between the softmax distribution $\sigma(\cdot)$ over token embeddings (Fig. 1, right). Indeed, we observe that even for high values of β there is very little chance that HELM can discriminate between the two objects. Moreover, we take a closer look at the resulting representations for the two different observations and measure their distance in terms of cosine similaritiy. Compared to HELM and HELMcs, HELMv2 improves separability of the two observations by a large margin (see Fig. 2, right). Based on these findings we add higher values for β to the gridsearch for HELM. Also, we show the performance of a recurrent agent based on the LSTM architecture (Hochreiter & Schmidhuber, 1997). HELM, HELMcs and LSTM do not achieve better performance than randomly choosing a path at the end of the corridor (see Fig. 2, left) after 2M interaction steps. On the contrary, HELMv2 is able to distinguish and memorize the objects and solve the Memory task.

Additionally, we compare HELMv2 to HELMcs, HELM and an LSTM baseline on a set of six diverse partially observable gridworld environments. Particularly, we select the same MiniGrid environments as Paischer et al. (2022). Moreover, we add eight 3D environments from the MiniWorld benchmark suite (Chevalier-Boisvert, 2018). For more details about our selected MiniWorld environments see Appendix A.1. Results are shown in Fig. 3 for MiniGrid (left) and MiniWorld (right). HELMv2 significantly outperforms HELM on both benchmark suites ($p = 2.24\text{e-}10$, $p = 0.012$, for MiniGrid,



Method	Cosine Similarity (↓)
HELM, $\beta = 1$	0.99 ± 1.68e-7
HELM, $\beta = 10$	0.99 ± 5.65e-6
HELM, $\beta = 100$	0.99 ± 8.2e-4
HELM, $\beta = 1000$	0.97 ± 0.08
HELMcs (ours)	0.99 ± 3.65e-5
HELMv2 (ours)	**0.83 ± 0.04**

Figure 2: **Left:** IQM and 95% bootstrapped CIs across 15 seeds for MiniGrid-Memory. **Right:** Average cosine similarities between observation embeddings containing either key, or ball, for HELM, HELMcs, and HELMv2. HELMv2 can better discriminate between the two different objects. Average is computed over batches of environment observations collected by a random policy.

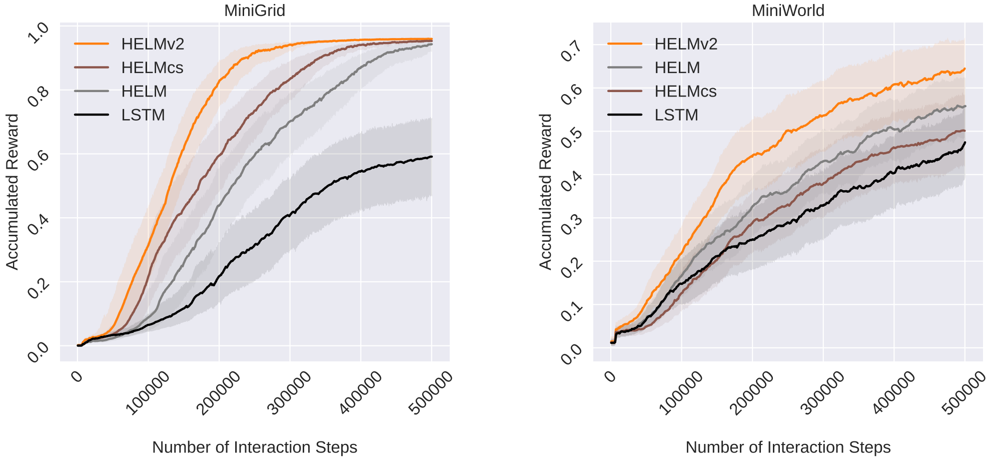


Figure 3: IQM and 95% bootstrapped CIs across 30 seeds for 6 MiniGrid environments (**left**), and 8 more 3D MiniWorld environments (**right**).

and MiniWorld, respectively). Also, HELMcs significantly outperforms HELM on the MiniGrid environemnts ($p = 8.7$e-4). There is no significant difference between the performance of HELM and HELMcs on MiniWorld environments. As in Paischer et al. (2022), the LSTM baseline is consistently outperformed by the HELM variants. In the following we analyse what components lead to the improved performance of HELMv2.

We conduct ablation studies to isolate the effect of the different components. In this regard, we add two additional settings: (i) HELMshift, and (ii) HELMclip. HELMshift adds the centering-and-scaling operation from HELMcs to the original HELM setting. HELMclip uses the same ResNet-50 CLIP image encoder with the centering-and-scaling operation followed by VocabAttn. We compare the performance of HELMclip and HELMshift to HELMv2 and HELM on all MiniGrid, and MiniWorld environments. Fig. 4 shows that adding the centering-and-scaling operation to HELM (HELMshift) does not lead to improved performance. In fact, on MiniWorld environments it even leads to worse performance than HELM. We suspect this is due to the frequent pixel changes in the observation space that do not correspond to significant changes in the environment state. While we also use image observations in MiniGrid the observation space is much more abstract since entire tiles change at once. Substituting $\boldsymbol{P}$ in HELMshift with the CLIP encoder (HELMclip) however,

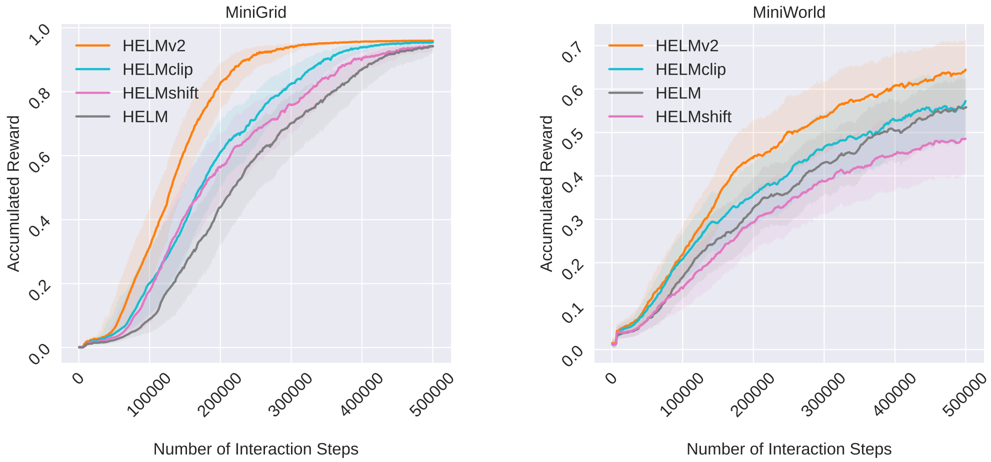


Figure 4: Mean IQM and 95% bootstrapped CIs across 30 seeds for RedBlueDoors (**left**) and TMaze (**right**) environments. HELMv2 consistently outperforms HELMclip on both environments.

results in an immediate improvement on MiniGrid environments. Since HELMclip still uses the VocabAttn, it is prone to representation collapse if changes in the CLIP space are subtle (see Fig. 1). Finally, discarding VocabAttn (HELMv2) drastically improves performance on both, MiniGrid and MiniWorld environment suites. While VocabAttn can help in some environments, its proneness to collapse hurts performance overall.

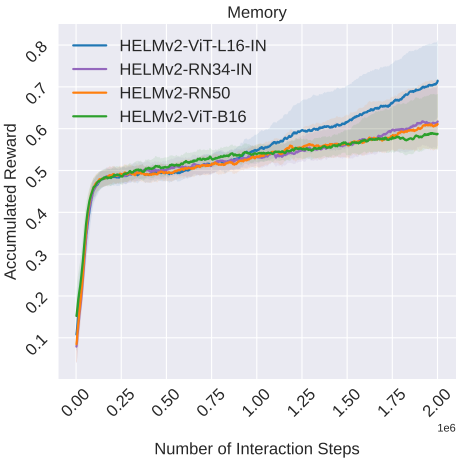
To show the effect of different image encoders, we perform an ablation where we substitute the CLIP image encoder with a ResNet (He et al., 2016, HELMv2-RN34-IN), and a Vision Transformer (Dosovitskiy et al., 2021, HELMv2-ViT-L16-IN), pretrained in a supervised manner on the popular Imagenet dataset (Deng et al., 2009). Additionally, we compare to a Vision Transformer version of CLIP (HELMv2-ViT-B/16). First, we measure cosine similarities to quantify the ability of the different vision encoders to separate the two observations (see Fig. 5, right). The HELMv2-ViT-B/16 variant exhibits the lowest cosine similarity between both observations. However, better separability does not correlate with improved downstream performance (see Fig. 5, left). In fact, the best performing agent uses the ViT-L16 pretrained on ImageNet. An explanation for this might be the difference in scale (Table 5) or the different pretraining paradigms (i.e. supervised vs. contrastive). We aim at answering this question in future work.

All our methods are trained with Proximal Policy Optimization (PPO, Schulman et al., 2017) on RGB observations. We evaluate all methods via the interquartile mean (IQM) and 95% bootstrapped confidence intervals (CIs) as suggested by Agarwal et al. (2021). To test for statistical significance, we perform a Wilcoxon test (Wilcoxon, 1945) at the end of training. We follow the architectural design of Paischer et al. (2022), but extend their hyperparameter search according to Appendix A.4.

4 Investigation on Semantic History Compression

We want to pave the way toward a semantic history compression for RL, that is, to preserve the semantic concepts of an observation when mapping it into the LM space. In this regard, we provide compelling evidence that it is indeed feasible to create such a semantic mapping between CLIP and the LM (see Fig. 6). When using this semantic mapping in the two selected minimalistic RL environments, however, we do not observe a significant improvement over HELMv2.

We can compute a lightweight mapping using the vocabularies of the CLIP language encoder $\mathcal{V}_{\text{CLIP}}$ and the LM encoder $\mathcal{V}_{\text{LM}}$ alone. First, we identify the overlap between the two vocabularies $\mathcal{V}_{OV} = \mathcal{V}_{\text{CLIP}} \cap \mathcal{V}_{\text{LM}}$ with size $l = |\mathcal{V}_{OV}|$. Next, we embed $\mathcal{V}_{OV}$ in the CLIP output space and the LM input space yielding embedding matrices $\boldsymbol{F} = (\boldsymbol{f}_1, \ldots, \boldsymbol{f}_l) \in \mathbb{R}^{n \times l}$ and $\boldsymbol{E}_l \in \mathbb{R}^{m \times l}$, respectively. In general, the mapping matrix $\boldsymbol{W}$ can be computed using any linear model. Algorithm 1 outlines a procedure for computing such a semantic mapping.



Method	Cosine Similarity (↓)
HELMv2-RN50	0.83 ± 0.04
HELMv2-ViT-B/16	**0.73** ± **0.06**
HELMv2-ViT-L16-IN	0.83 ± 0.04
HELMv2-RN34-IN	0.76 ± 0.04

Figure 5: **Left:** IQM and 95% bootstrapped CIs across 30 seeds for Memory environment. **Right:** Average cosine similarity between observations containing a key and a ball for different vision encoders in HELMv2. Average is computed over batches of environment observations collected by a random policy.

Algorithm 1 Associating CLIP output space with LM input space

```
Require: CLIP language encoder $\text{CLIP}_{LM}$, Vocabulary of CLIP $\mathcal{V}_{\text{CLIP}}$, Language Model Embedding
    Layer $\text{LM}(v_i)$, Language Model vocabulary $\mathcal{V}_{\text{LM}}$
    $\mathcal{V}_{\text{OV}} \leftarrow \mathcal{V}_{\text{CLIP}} \cap \mathcal{V}_{\text{LM}}$                ▷ Search for overlapping vocabulary
    $\boldsymbol{f}_i = \text{CLIP}_{LM}(v_i) \quad \text{for} \quad v_i \in \mathcal{V}_{\text{OV}}$    ▷ Embed tokens in CLIP output space
    $\boldsymbol{e}_i = \text{LM}(v_i) \quad \text{for} \quad v_i \in \mathcal{V}_{\text{OV}}$          ▷ Embed tokens in LM input space
    $\boldsymbol{W} \leftarrow \texttt{create\_mapping}(\boldsymbol{F}, \boldsymbol{E})$            ▷ Compute mapping between embeddings
```

A common choice for creating a mapping between monolingual embedding spaces is the Procrustes method (Artetxe et al., 2018; Hoshen & Wolf, 2018; Smith et al., 2017; Lample et al., 2018; Zhang et al., 2016; Xing et al., 2015; Minixhofer et al., 2022). In total, we compare four different linear mappings:

- **Linear:** Ordinary Least Squares, as in (Mikolov et al., 2013)
- **Ridge:** Least Squares with Thikonov regularization
- **Procrustes:** Least Squares with orthogonality constraint (Schönemann, 1966; Gower & Dijksterhuis, 2005))
- **RobProc:** The Robust Procrustes algorithm, that iteratively refines the Procrustes method based on its predicion error $\|\boldsymbol{F}\boldsymbol{W} - \boldsymbol{E}\|_F^2$[1].

We perform a 5-fold cross validation and measure the accuracy, considering each token as its own class. Table 2 in Appendix A.2 shows the average train and test accuracy for the different linear mapping methods for various CLIP backbones and TrXL. We observe a strong overfitting effect for Linear, while constrained optimization (Ridge, Procrustes, and RobProc) generalize better. The best method in terms of generalization is the Procrustes method. RobProc does not provide significant gains over Procrustes.

Due to the alignment of the CLIP image encoder and the CLIP language encoder, our mapping can be used to project images to the LM space while preserving its semantics. We demonstrate that by identifying the closest tokens in the LM space after applying our mapping to project natural images (see Fig. 6). Further, we conduct a quantitative analysis using publicly available image captioning datasets. In this regard, we draw a random subset of 1000 image-caption pairs of the popular MSCOCO dataset (Lin et al., 2014). The subset is filtered to contain only image-text pairs where the

[1] For more details, we refer the reader to Groenen et al. (2005)

captions contain at least 5 tokens of our computed vocabulary overlap above. For preprocessing of the captions we remove stop words and apply stemming. Next, we propagate the image through various CLIP backbones, to obtain an image embedding and map it to the LM space using our pre-computed mappings. Finally, we rank tokens in the LM space based on their cosine similarity to the mapped image. Based on the obtained ranking we compute the Mean Reciprocal Rank (MRR, Craswell, 2009) and the Normalized Discounted Cumulative Gain (NDCG, Järvelin & Kekäläinen, 2002). Table 1 shows the NDCG for various CLIP backbones mapping to the embedding space of TrXL relative to ranking in the CLIP space (rNDCG). We observe that the Procrustes mapping consistently outperforms the Linear and Ridge mapping. Furthermore, there is no improvement in iteratively refining the Procrustes method as in RobProc. Table 3 in Appendix A.2 shows the MRR for the various CLIP backbones.

	Linear	Ridge	Procrustes	RobProc
RN50	0.636±0.17	0.701±0.18	**0.774±0.171**	0.772±0.172
RN101	0.647±0.177	0.7±0.186	**0.819±0.178**	0.809±0.18
RN50x4	0.638±0.164	0.68±0.172	**0.824±0.178**	0.814±0.165
RN50x16	0.635±0.165	0.68±0.174	**0.814±0.178**	0.809±0.178
RN50x64	0.632±0.167	0.647±0.165	**0.797±0.179**	0.791±0.178
ViT-B/32	0.624±0.166	0.648±0.17	**0.808±0.169**	0.798±0.167
ViT-B/16	0.61±0.158	0.663±0.176	**0.822±0.186**	0.81±0.187
ViT-L/14	0.619±0.166	0.658±0.172	**0.815±0.186**	0.807±0.183
ViT-L/14*	0.61±0.164	0.646±0.172	**0.81±0.188**	0.802±0.186

Table 1: rNDCG for ranking of tokens in the LM space relative to ranking of tokens in the CLIP space. Image-caption pairs are drawn from the MSCOCO dataset. ViT-L/14* receives images resized to 336 pixels as input.

Fig. 6 shows promising results for natural images from the MSCOCO dataset and their corresponding token rankings. Following our analysis we propose a new setting, namely SHELM (for semantic HELM), that utilizes a CLIP vision backbone in combination with the semantic mapping. We select the best backbone-mapping combination by ranking all combinations according to the average absolute NDCG (aNDCG), and select the top 5 settings. Furthermore, we perform a Wilcoxon test for statistical significance between the combinations after bonferroni correction (Bonferroni, 1936). The two combinations, ViT-B/16+Procrustes, and ViT-L/14*+Procrustes, significantly outperform all competitors in terms of absolute NDCG. Due to the imposed complexity of ViT-L/14* (see Table 5), we choose ViT-B/16+Procrustes to instantiate SHELM.

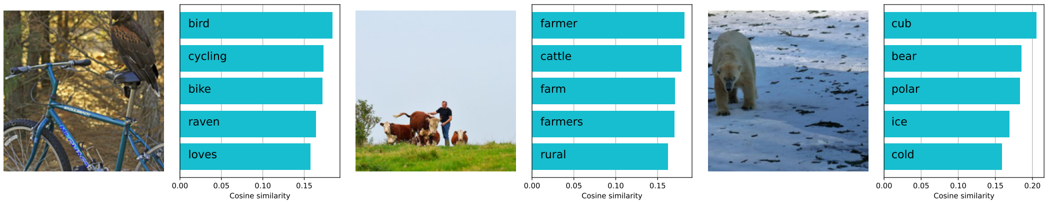


Figure 6: Top 5 closest tokens in the LM embedding space after applying the linear mapping to an image embedded by CLIP.

We show preliminary results of SHELM on the Memory environment, all MiniGrid, and all MiniWorld environments in Fig. 7. Additionally, we compare with HELMv2+RandOrtho, which samples random orthogonal matrices from the Haar distribution (Stewart, 1980). Surprisingly, we observe no statistically significant differences in performance between the different methods. We believe this is due to the fact that our selected environments are artificial 2D and 3D scenes for which the CLIP image encoder is unable to sufficiently extract semantically meaningful features. This is in line with findings of Fan et al. (2022) who minimally finetune CLIP for the Minedojo environment. We validate this finding by taking a closer look at the token rankings for the two observations of the Memory environment in Fig. 8 in the Appendix. Indeed, the tokens exhibiting the highest similarity in the CLIP space do not describe the semantics of the image, but rather similar concepts, i.e., the

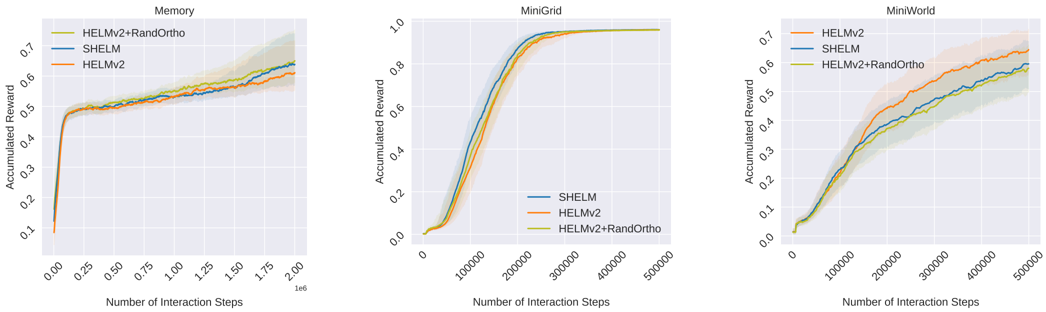


Figure 7: IQM and 95% bootstrapped CIs across 30 seeds on Memory (**left**), MiniGrid (**middle**), and MiniWorld (**right**) environments.

token *pong* is ranked higher when the ball is present in the observation. Still, it is worth mentioning that the ranked tokens differ, which aligns with our finding of CLIP's discriminative power. The ability of CLIP to extract semantics may improve in more realistic environments as shown in Tam et al. (2022). We aim to apply SHELM to similar environments in the future.

Finally, we conduct an additional analysis on how much the different methods preserve the semantics of natural images. In this regard, we perform the same quantitative analysis as above for the different methods presented in this work. We show the MRR and aNDCG for our MSCOCO subset of image-caption pairs in Table 4 in Appendix A.3. Methods such as HELMv2, HELMclip, or HELMv2+RandOrtho yield on-par or worse results compared to a random ranking. This is due to the fact, that for lower values of β the VocabAttn tends to output the mean over token embeddings. In this case, the resulting ranking is equal across images, and thus, results in lower MRR and NDCG than random rankings. Also, substituting the RobProc mapping with a random mapping drawn from the Haar distribution (HELMv2+RandOrtho) leads to random rankings. On the contrary, SHELM preserves more of the semantics resulting in much better rankings. Adding VocabAttn after the Procrustes mapping of SHELM again results in loss of information and a worse ranking.

5 Related Work

Reinforcement Learning with incomplete state information necessitates compression of past events. A variety of prior works has used History Compression to tackle Credit Assignment (Arjona-Medina et al., 2019; Patil et al., 2022; Widrich et al., 2021; Holzleitner et al., 2021), and partial observability (Hausknecht & Stone, 2015; Vinyals et al., 2019; Berner et al., 2019; Pleines et al., 2022). The question of what information to store given a stream of observations was investigated in prior works (Schmidhuber, 1992; Zenke et al., 2017; Kirkpatrick et al., 2016; Schwarz et al., 2018; Ruvolo & Eaton, 2013). Typical choices for network architectures that are capable of compressing a stream of data are LSTM (Hochreiter & Schmidhuber, 1997), and Transformer (Vaswani et al., 2017).

The Transformer architecture has blossomed in the realm of Natural Language Processing. Many works have focused on scaling pretraining of Transformers to create LMs that generate realistic human-like text (Devlin et al., 2019; Radford & Narasimhan, 2018; Radford et al., 2019; Brown et al., 2020; Thoppilan et al., 2022; Rae et al., 2021; Zhang et al., 2022; Hoffmann et al., 2022; Chowdhery et al., 2022; Dai et al., 2019). More recently, interest has sparked in leveraging the Transformer for decision making (Parisotto et al., 2020; Sukhbaatar et al., 2021; Chen et al., 2021; Janner et al., 2021; Zheng et al., 2022; Melo, 2022). Moreover, it has been found that pretrained LMs are very well suited for creating abstractions of a sequence of visual observations (Paischer et al., 2022).

Language provides useful abstractions for Reinforcement Learning. Hill et al. (2021) illustrate that language abstraction enables compositional generalization and intrinsic motivation in embodied environments. LMs have been used for exploration in text-based environments (Yao et al., 2020), or in visual environments that provide a language oracle (Mu et al., 2022). Multimodal models pretrained with language supervision provide abstract embedding spaces that can be used for visual exploration (Tam et al., 2022). Further, language pretraining has been leveraged to initialize policies in text-based environments (Li et al., 2022), and for sequence modeling in the offline RL setup (Reid et al., 2022). Moreover, language has been used for reward shaping (Wang et al., 2019; Bahdanau et al., 2019; Goyal et al., 2019). We leverage pretrained multimodal models and map them into a semantically meaningful region of the embedding space of a LM.

Foundation models (Bommasani et al., 2021), such as GPT-3 (Brown et al., 2020), demonstrated remarkable few-shot capabilities. Pretraining on large-scale text corpora enables such complex models to abstract relevant knowledge for solving math-word problems (Griffith & Kalita, 2019), symbolic math problems (Noorbakhsh et al., 2021), and even university-level math exercises (Drori et al., 2021). As shown by Petroni et al. (2019); Talmor et al. (2020); Kassner et al. (2020), pretrained LMs can learn abstract symbolic rules and perform reasoning. Recently, there has been a surge of complex models trained on large scale data that combines multiple modalities, such as images and text, as in CLIP (Radford et al., 2021). Moreover, vision FMs have been demonstrated to be well adaptable to foreign domains (Adler et al., 2020; Evci et al., 2022).

The computation of our mapping network is reminiscent of "model-stitching" (Lenc & Vedaldi, 2019; Bansal et al., 2021; Csiszárik et al., 2021; Scialom et al., 2020). In model-stitching an encoder is stitched via a sparse linear transformation to a compatible decoder. Moschella et al. (2022) use relative representation spaces to avoid the training of stitching layers, however requires decoder to be trained on these relative spaces. Merullo et al. (2022); Scialom et al. (2020) optimize a linear mapping from a vision encoder to a LM on the downstream task of image captioning. In contrast, our mapping can be computed with a closed form and does not require optimization of a vast amount of data, while preserving the symmetry between embedding spaces due to the orthogonality constraint.

A plethora of works have focused on conditioning LMs on the popular CLIP model. Alayrac et al. (2022) use cross-attention combined with the Perceiver architecture (Jaegle et al., 2021) trained on vast web data to combine a frozen pretrained CLIP (Radford et al., 2021) with a generative language model. Yu et al. (2022) adds a contrastive loss between CLIP and a LM encoder for image captioning. Similarly, (Mokady et al., 2021) trains a mapping model from the CLIP output space to a LM input space for image captioning. In contrast, (Su et al., 2022) proposes to condition the decoding step of a finetuned LM on image captioning on a frozen CLIP model. Finally, (Tewel et al., 2021) optimizes the context cache during inference to mazimize a similarity score computed with CLIP for zero-shot image captioning. In contrast to these methods, our mapping can be efficiently computed offline, and allows conditioning a LM on visual information without any additional optimization procedure. Zeng et al. (2022) use language as communication interface between various foundation models to solve certain tasks. Further, CLIP has demonstrated its robustness and versatility across various tasks in the RL setup (Ostapenko et al., 2022; Parisi et al., 2022). We aim to investigate extensions to the CLIP model, such as CLOOB (Fürst et al., 2021), in our framework as well.

6 Conclusion

Reinforcement Learning with incomplete state information requires an agent to remember past events. Recent work had illustrated that large pretrained LMs are suitable for creating abstractions of past events (Paischer et al., 2022). HELM randomly maps observations to language tokens before compressing the compounding information via a LM. We presented certain conditions under which HELM is incapable of remembering information that is essential to solve a given task. Further, we proposed an incremental improvement over HELM comprising a feature-wise centering-and-scaling operation and a pretrained vision encoder, which we call HELMv2. On the challenging MiniGrid-Memory environment HELMv2 overcomes the limitations of HELM and successfully solves the task. Furthermore, HELMv2 yields significant improvements over HELM on minimalistic 2D and 3D environments.

Additionally, we pave the way toward a semantic history compression mechanism, that does not require training in the RL context. By leveraging multi-modal pretrained foundation models we demonstrated how to obtain a semantic mapping between a vision encoder and a LM. First results on MiniGrid and MiniWorld environments, however, showed that a semantic alignment between CLIP and LM does not significantly improve performance. Inherently, the semantic mapping is limited by the ability of the vision encoder to extract semantically meaningful features of the corresponding simulations. We surmise that this restriction is the reason why we did not observe improvements. In the future, we aim at improving the semantic mapping between the foundation models and demonstrating that it can provide benefits for more realistic scenes.

Acknowledgements

The ELLIS Unit Linz, the LIT AI Lab, the Institute for Machine Learning, are supported by the Federal State Upper Austria. IARAI is supported by Here Technologies. We thank the projects AI-MOTION (LIT-2018-6-YOU-212), AI-SNN (LIT-2018-6-YOU-214), DeepFlood (LIT-2019-8-YOU-213), Medical Cognitive Computing Center (MC3), INCONTROL-RL (FFG-881064), PRIMAL (FFG-873979), S3AI (FFG-872172), DL for GranularFlow (FFG-871302), AIRI FG 9-N (FWF-36284, FWF-36235), ELISE (H2020-ICT-2019-3 ID: 951847). We thank Audi.JKU Deep Learning Center, TGW LOGISTICS GROUP GMBH, University SAL Labs initiative of Silicon Austria Labs (SAL), FILL Gesellschaft mbH, Anyline GmbH, Google, ZF Friedrichshafen AG, Robert Bosch GmbH, UCB Biopharma SRL, Merck Healthcare KGaA, Verbund AG, Software Competence Center Hagenberg GmbH, TÜV Austria, Frauscher Sensonic and the NVIDIA Corporation.

References

Adler, T., Brandstetter, J., Widrich, M., Mayr, A., Kreil, D. P., Kopp, M., Klambauer, G., and Hochreiter, S. Cross-Domain Few-Shot Learning by Representation Fusion. *CoRR*, abs/2010.06498, 2020. arXiv: 2010.06498.

Agarwal, R., Schwarzer, M., Castro, P. S., Courville, A. C., and Bellemare, M. G. Deep Reinforcement Learning at the Edge of the Statistical Precipice. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 29304–29320, 2021.

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J., Borgeaud, S., Brock, A., Nematzadeh, A., Sharifzadeh, S., Binkowski, M., Barreira, R., Vinyals, O., Zisserman, A., and Simonyan, K. Flamingo: a Visual Language Model for Few-Shot Learning. *CoRR*, abs/2204.14198, 2022. doi: 10.48550/arXiv.2204.14198. arXiv: 2204.14198.

Arjona-Medina, J. A., Gillhofer, M., Widrich, M., Unterthiner, T., Brandstetter, J., and Hochreiter, S. RUDDER: Return Decomposition for Delayed Rewards. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 13544–13555, 2019.

Artetxe, M., Labaka, G., and Agirre, E. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 789–798, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1073.

Bahdanau, D., Hill, F., Leike, J., Hughes, E., Hosseini, S. A., Kohli, P., and Grefenstette, E. Learning to Understand Goal Specifications by Modelling Reward. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Bansal, Y., Nakkiran, P., and Barak, B. Revisiting Model Stitching to Compare Neural Representations. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 225–236, 2021.

Berner, C., Brockman, G., Chan, B., Cheung, V., Debiak, P., Dennison, C., Farhi, D., Fischer, Q., Hashme, S., Hesse, C., Józefowicz, R., Gray, S., Olsson, C., Pachocki, J., Petrov, M., Pinto, H. P. d. O., Raiman, J., Salimans, T., Schlatter, J., Schneider, J., Sidor, S., Sutskever, I., Tang, J., Wolski, F., and Zhang, S. Dota 2 with Large Scale Deep Reinforcement Learning. *CoRR*, abs/1912.06680, 2019.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., Arx, S. v., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N. S., Chen, A. S., Creel, K., Davis, J. Q., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L., Goel, K., Goodman, N. D., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M. S., Krishna, R., Kuditipudi, R., and al, e. On the Opportunities and Risks of Foundation Models. *CoRR*, abs/2108.07258, 2021. arXiv: 2108.07258.

Bonferroni, C. *Teoria statistica delle classi e calcolo delle probabilità*. Pubblicazioni del R. Istituto superiore di scienze economiche e commerciali di Firenze. Seeber, 1936.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020.

Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., Abbeel, P., Srinivas, A., and Mordatch, I. Decision Transformer: Reinforcement Learning via Sequence Modeling. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 15084–15097, 2021.

Chevalier-Boisvert, M. MiniWorld: Minimalistic 3D Environment for RL & Robotics Research, 2018. Publication Title: GitHub repository.

Chevalier-Boisvert, M., Willems, L., and Pal, S. Minimalistic Gridworld Environment for OpenAI Gym, 2018. Publication Title: GitHub repository.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N., Prabhakaran, V., Reif, E., Du, N., Hutchinson, B., Pope, R., Bradbury, J., Austin, J., Isard, M., Gur-Ari, G., Yin, P., Duke, T., Levskaya, A., Ghemawat, S., Dev, S., Michalewski, H., Garcia, X., Misra, V., Robinson, K., Fedus, L., Zhou, D., Ippolito, D., Luan, D., Lim, H., Zoph, B., Spiridonov, A., Sepassi, R., Dohan, D., Agrawal, S., Omernick, M., Dai, A. M., Pillai, T. S., Pellat, M., Lewkowycz, A., Moreira, E., Child, R., Polozov, O., Lee, K., Zhou, Z., Wang, X., Saeta, B., Diaz, M., Firat, O., Catasta, M., Wei, J., Meier-Hellstern, K., Eck, D., Dean, J., Petrov, S., and Fiedel, N. PaLM: Scaling Language Modeling with Pathways. *CoRR*, abs/2204.02311, 2022. doi: 10.48550/arXiv.2204.02311. arXiv: 2204.02311.

Craswell, N. Mean Reciprocal Rank. In LIU, L. and ÖZSU, M. T. (eds.), *Encyclopedia of Database Systems*, pp. 1703–1703. Springer US, Boston, MA, 2009. ISBN 978-0-387-39940-9. doi: 10.1007/978-0-387-39940-9_488.

Csiszárik, A., Korösi-Szabó, P., Matszangosz, A. K., Papp, G., and Varga, D. Similarity and Matching of Neural Network Representations. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 5656–5668, 2021.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J. G., Le, Q. V., and Salakhutdinov, R. Transformer-XL: Attentive Language Models beyond a Fixed-Length Context. In Korhonen, A., Traum, D. R., and Màrquez, L. (eds.), *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 2978–2988. Association for Computational Linguistics, 2019. doi: 10.18653/v1/p19-1285.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

Drori, I., Tran, S., Wang, R., Cheng, N., Liu, K., Tang, L., Ke, E., Singh, N., Patti, T. L., Lynch, J., Shporer, A., Verma, N., Wu, E., and Strang, G. A Neural Network Solves and Generates Mathematics Problems by Program Synthesis: Calculus, Differential Equations, Linear Algebra, and More. *CoRR*, abs/2112.15594, 2021.

Evci, U., Dumoulin, V., Larochelle, H., and Mozer, M. C. Head2Toe: Utilizing Intermediate Representations for Better Transfer Learning. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 6009–6033. PMLR, 2022.

Fan, L., Wang, G., Jiang, Y., Mandlekar, A., Yang, Y., Zhu, H., Tang, A., Huang, D.-A., Zhu, Y., and Anandkumar, A. MineDojo: Building Open-Ended Embodied Agents with Internet-Scale Knowledge. *CoRR*, abs/2206.08853, 2022. doi: 10.48550/arXiv.2206.08853. arXiv: 2206.08853.

Fürst, A., Rumetshofer, E., Tran, V., Ramsauer, H., Tang, F., Lehner, J., Kreil, D. P., Kopp, M., Klambauer, G., Bitto-Nemling, A., and Hochreiter, S. CLOOB: Modern Hopfield Networks with InfoLOOB Outperform CLIP. *CoRR*, abs/2110.11316, 2021. arXiv: 2110.11316.

Gower, J. and Dijksterhuis, G. Procrustes Problems. *Procrustes Problems, Oxford Statistical Science Series*, Vol. 30, January 2005. doi: 10.1093/acprof:oso/9780198510581.001.0001. ISBN: 9780198510581.

Goyal, P., Niekum, S., and Mooney, R. J. Using Natural Language for Reward Shaping in Reinforcement Learning. In Kraus, S. (ed.), *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pp. 2385–2391. ijcai.org, 2019. doi: 10.24963/ijcai.2019/331.

Griffith, K. and Kalita, J. Solving Arithmetic Word Problems Automatically Using Transformer and Unambiguous Representations. *CoRR*, abs/1912.00871, 2019.

Groenen, P. J. F., Giaquinto, P., and Kiers, H. A. L. An improved majorization algorithm for robust procrustes analysis. *Springer US*, pp. 151–158, 2005.

Hausknecht, M. J. and Stone, P. Deep Recurrent Q-Learning for Partially Observable MDPs. In *2015 AAAI Fall Symposia, Arlington, Virginia, USA, November 12-14, 2015*, pp. 29–37. AAAI Press, 2015.

He, K., Zhang, X., Ren, S., and Sun, J. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 770–778. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.90.

Hellinger, E. Neue Begründung der Theorie quadratischer Formen von unendlichvielen Veränderlichen. *Journal für die reine und angewandte Mathematik*, 1909(136):210–271, 1909. doi: doi:10.1515/crll.1909.136.210.

Hill, F., Tieleman, O., Glehn, T. v., Wong, N., Merzic, H., and Clark, S. Grounded Language Learning Fast and Slow. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

Hochreiter, S. and Schmidhuber, J. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.

Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D. d. L., Hendricks, L. A., Welbl, J., Clark, A., Hennigan, T., Noland, E., Millican, K., Driessche, G. v. d., Damoc, B., Guy, A., Osindero, S., Simonyan, K., Elsen, E., Rae, J. W., Vinyals, O., and Sifre, L. Training Compute-Optimal Large Language Models. *CoRR*, abs/2203.15556, 2022. doi: 10.48550/arXiv.2203.15556. arXiv: 2203.15556.

Holzleitner, M., Gruber, L., Arjona-Medina, J. A., Brandstetter, J., and Hochreiter, S. Convergence Proof for Actor-Critic Methods Applied to PPO and RUDDER. *Trans. Large Scale Data Knowl. Centered Syst.*, 48:105–130, 2021. doi: 10.1007/978-3-662-63519-3_5.

Hoshen, Y. and Wolf, L. Non-Adversarial Unsupervised Word Translation. In Riloff, E., Chiang, D., Hockenmaier, J., and Tsujii, J. (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pp. 469–478. Association for Computational Linguistics, 2018. doi: 10.18653/v1/d18-1043.

Jaegle, A., Gimeno, F., Brock, A., Vinyals, O., Zisserman, A., and Carreira, J. Perceiver: General Perception with Iterative Attention. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4651–4664. PMLR, 2021.

Janner, M., Li, Q., and Levine, S. Offline Reinforcement Learning as One Big Sequence Modeling Problem. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 1273–1286, 2021.

Johnson, W. B. and Lindenstrauss, J. Extensions of Lipschitz mappings into a Hilbert space. *Contemporary mathematics*, 26, 1984.

Järvelin, K. and Kekäläinen, J. Cumulated Gain-Based Evaluation of IR Techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, October 2002. ISSN 1046-8188. doi: 10.1145/582415.582418. Place: New York, NY, USA Publisher: Association for Computing Machinery.

Kaelbling, L. P., Littman, M. L., and Cassandra, A. R. Planning and Acting in Partially Observable Stochastic Domains. *Artificial Intelligence*, 101:99–134, 1998.

Kassner, N., Krojer, B., and Schütze, H. Are Pretrained Language Models Symbolic Reasoners over Knowledge? In Fernández, R. and Linzen, T. (eds.), *Proceedings of the 24th Conference on Computational Natural Language Learning, CoNLL 2020, Online, November 19-20, 2020*, pp. 552–564. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.conll-1.45.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N. C., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., and Hadsell, R. Overcoming catastrophic forgetting in neural networks. *CoRR*, abs/1612.00796, 2016.

Lample, G., Conneau, A., Ranzato, M., Denoyer, L., and Jégou, H. Word translation without parallel data. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Lenc, K. and Vedaldi, A. Understanding Image Representations by Measuring Their Equivariance and Equivalence. *Int. J. Comput. Vis.*, 127(5):456–476, 2019. doi: 10.1007/s11263-018-1098-y.

Li, S., Puig, X., Paxton, C., Du, Y., Wang, C., Fan, L., Chen, T., Huang, D.-A., Akyürek, E., Anandkumar, A., Andreas, J., Mordatch, I., Torralba, A., and Zhu, Y. Pre-Trained Language Models for Interactive Decision-Making. *CoRR*, abs/2202.01771, 2022. arXiv: 2202.01771.

Lin, T.-Y., Maire, M., Belongie, S. J., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft COCO: Common Objects in Context. In Fleet, D. J., Pajdla, T., Schiele, B., and Tuytelaars, T. (eds.), *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pp. 740–755. Springer, 2014. doi: 10.1007/978-3-319-10602-1_48.

Melo, L. C. Transformers are Meta-Reinforcement Learners. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 15340–15359. PMLR, 2022.

Merullo, J., Castricato, L., Eickhoff, C., and Pavlick, E. Linearly Mapping from Image to Text Space, 2022.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. Distributed Representations of Words and Phrases and their Compositionality. In Burges, C. J., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

Minixhofer, B., Paischer, F., and Rekabsaz, N. WECHSEL: Effective initialization of subword embeddings for cross-lingual transfer of monolingual language models. In Carpuat, M., Marneffe, M.-C. d., and Ruíz, I. V. M. (eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pp. 3992–4006. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.naacl-main.293.

Mokady, R., Hertz, A., and Bermano, A. H. ClipCap: CLIP Prefix for Image Captioning. *CoRR*, abs/2111.09734, 2021. arXiv: 2111.09734.

Moschella, L., Maiorca, V., Fumero, M., Norelli, A., Locatello, F., and Rodolà, E. Relative representations enable zero-shot latent space communication. *CoRR*, abs/2209.15430, 2022. doi: 10.48550/arXiv.2209.15430. arXiv: 2209.15430.

Mu, J., Zhong, V., Raileanu, R., Jiang, M., Goodman, N. D., Rocktäschel, T., and Grefenstette, E. Improving Intrinsic Exploration with Language Abstractions. *CoRR*, abs/2202.08938, 2022. arXiv: 2202.08938.

Noorbakhsh, K., Sulaiman, M., Sharifi, M., Roy, K., and Jamshidi, P. Pretrained Language Models are Symbolic Mathematics Solvers too! *CoRR*, abs/2110.03501, 2021.

Ostapenko, O., Lesort, T., Rodríguez, P., Arefin, M. R., Douillard, A., Rish, I., and Charlin, L. Foundational Models for Continual Learning: An Empirical Study of Latent Replay. *arXiv preprint arXiv:2205.00329*, 2022.

Paischer, F., Adler, T., Patil, V. P., Bitto-Nemling, A., Holzleitner, M., Lehner, S., Eghbal-zadeh, H., and Hochreiter, S. History Compression via Language Models in Reinforcement Learning. *CoRR*, abs/2205.12258, 2022. doi: 10.48550/arXiv.2205.12258. arXiv: 2205.12258.

Parisi, S., Rajeswaran, A., Purushwalkam, S., and Gupta, A. The Unsurprising Effectiveness of Pre-Trained Vision Models for Control. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17359–17371. PMLR, 2022.

Parisotto, E., Song, F., Rae, J., Pascanu, R., Gulcehre, C., Jayakumar, S., Jaderberg, M., Kaufman, R. L., Clark, A., Noury, S., Botvinick, M., Heess, N., and Hadsell, R. Stabilizing Transformers for Reinforcement Learning. In *International Conference on Machine Learning*, pp. 7487–7498. PMLR, November 2020.

Patil, V. P., Hofmarcher, M., Dinu, M.-C., Dorfer, M., Blies, P. M., Brandstetter, J., Arjona-Medina, J. A., and Hochreiter, S. Align-RUDDER: Learning From Few Demonstrations by Reward Redistribution. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17531–17572. PMLR, 2022.

Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P. S. H., Bakhtin, A., Wu, Y., and Miller, A. H. Language Models as Knowledge Bases? In Inui, K., Jiang, J., Ng, V., and Wan, X. (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing* *and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pp. 2463–2473. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1250.

Pleines, M., Pallasch, M., Zimmer, F., and Preuss, M. Generalization, Mayhems and Limits in Recurrent Proximal Policy Optimization. *CoRR*, abs/2205.11104, 2022. doi: 10.48550/arXiv.2205.11104. arXiv: 2205.11104.

Radford, A. and Narasimhan, K. Improving Language Understanding by Generative Pre-Training. 2018.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning Transferable Visual Models From Natural Language Supervision. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021.

Rae, J. W., Borgeaud, S., Cai, T., Millican, K., Hoffmann, J., Song, H. F., Aslanides, J., Henderson, S., Ring, R., Young, S., Rutherford, E., Hennigan, T., Menick, J., Cassirer, A., Powell, R., Driessche, G. v. d., Hendricks, L. A., Rauh, M., Huang, P.-S., Glaese, A., Welbl, J., Dathathri, S., Huang, S., Uesato, J., Mellor, J., Higgins, I., Creswell, A., McAleese, N., Wu, A., Elsen, E., Jayakumar, S. M., Buchatskaya, E., Budden, D., Sutherland, E., Simonyan, K., Paganini, M., Sifre, L., Martens, L., Li, X. L., Kuncoro, A., Nematzadeh, A., Gribovskaya, E., Donato, D., Lazaridou, A., Mensch, A., Lespiau, J.-B., Tsimpoukelli, M., Grigorev, N., Fritz, D., Sottiaux, T., Pajarskas, M., Pohlen, T., Gong, Z., Toyama, D., d'Autume, C. d. M., Li, Y., Terzi, T., Mikulik, V., Babuschkin, I., Clark, A., Casas, D. d. L., Guy, A., Jones, C., Bradbury, J., Johnson, M., Hechtman, B. A., Weidinger, L., Gabriel, I., Isaac, W. S., Lockhart, E., Osindero, S., Rimell, L., Dyer, C., Vinyals, O., Ayoub, K., Stanway, J., Bennett, L., Hassabis, D., Kavukcuoglu, K., and Irving, G. Scaling Language Models: Methods, Analysis & Insights from Training Gopher. *CoRR*, abs/2112.11446, 2021. arXiv: 2112.11446.

Reid, M., Yamada, Y., and Gu, S. S. Can Wikipedia Help Offline Reinforcement Learning? *CoRR*, abs/2201.12122, 2022. arXiv: 2201.12122.

Ruvolo, P. and Eaton, E. ELLA: An Efficient Lifelong Learning Algorithm. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 507–515. JMLR.org, 2013.

Schmidhuber, J. Learning Complex, Extended Sequences Using the Principle of History Compression. *Neural Comput.*, 4(2):234–242, 1992. doi: 10.1162/neco.1992.4.2.234.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal Policy Optimization Algorithms. *CoRR*, abs/1707.06347, 2017. arXiv: 1707.06347.

Schwarz, J., Czarnecki, W., Luketina, J., Grabska-Barwinska, A., Teh, Y. W., Pascanu, R., and Hadsell, R. Progress & Compress: A scalable framework for continual learning. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4535–4544. PMLR, 2018.

Schönemann, P. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1): 1–10, 1966.

Scialom, T., Bordes, P., Dray, P.-A., Staiano, J., and Gallinari, P. What BERT Sees: Cross-Modal Transfer for Visual Question Generation. In Davis, B., Graham, Y., Kelleher, J. D., and Sripada, Y. (eds.), *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020, Dublin, Ireland, December 15-18, 2020*, pp. 327–337. Association for Computational Linguistics, 2020.

Smith, S. L., Turban, D. H. P., Hamblin, S., and Hammerla, N. Y. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

Stewart, G. W. The Efficient Generation of Random Orthogonal Matrices with an Application to Condition Estimators. *SIAM Journal on Numerical Analysis*, 17(3):403–409, 1980. doi: 10.1137/0717034. _eprint: https://doi.org/10.1137/0717034.

Su, Y., Lan, T., Liu, Y., Liu, F., Yogatama, D., Wang, Y., Kong, L., and Collier, N. Language Models Can See: Plugging Visual Controls in Text Generation. *CoRR*, abs/2205.02655, 2022. doi: 10.48550/arXiv.2205.02655. arXiv: 2205.02655.

Sukhbaatar, S., Ju, D., Poff, S., Roller, S., Szlam, A., Weston, J., and Fan, A. Not All Memories are Created Equal: Learning to Forget by Expiring. In *International Conference on Machine Learning*, pp. 9902–9912. PMLR, July 2021.

Talmor, A., Elazar, Y., Goldberg, Y., and Berant, J. oLMpics - On what Language Model Pre-training Captures. *Trans. Assoc. Comput. Linguistics*, 8:743–758, 2020.

Tam, A. C., Rabinowitz, N. C., Lampinen, A. K., Roy, N. A., Chan, S. C. Y., Strouse, D. J., Wang, J. X., Banino, A., and Hill, F. Semantic Exploration from Language Abstractions and Pretrained Representations. *CoRR*, abs/2204.05080, 2022. doi: 10.48550/arXiv.2204.05080. arXiv: 2204.05080.

Tewel, Y., Shalev, Y., Schwartz, I., and Wolf, L. Zero-Shot Image-to-Text Generation for Visual-Semantic Arithmetic. *arXiv preprint arXiv:2111.14447*, 2021.

Thoppilan, R., Freitas, D. D., Hall, J., Shazeer, N., Kulshreshtha, A., Cheng, H.-T., Jin, A., Bos, T., Baker, L., Du, Y., Li, Y., Lee, H., Zheng, H. S., Ghafouri, A., Menegali, M., Huang, Y., Krikun, M., Lepikhin, D., Qin, J., Chen, D., Xu, Y., Chen, Z., Roberts, A., Bosma, M., Zhou, Y., Chang, C.-C., Krivokon, I., Rusch, W., Pickett, M., Meier-Hellstern, K. S., Morris, M. R., Doshi, T., Santos, R. D., Duke, T., Soraker, J., Zevenbergen, B., Prabhakaran, V., Diaz, M., Hutchinson, B., Olson, K., Molina, A., Hoffman-John, E., Lee, J., Aroyo, L., Rajakumar, R., Butryna, A., Lamm, M., Kuzmina, V., Fenton, J., Cohen, A., Bernstein, R., Kurzweil, R., Aguera-Arcas, B., Cui, C., Croak, M., Chi, E. H., and Le, Q. LaMDA: Language Models for Dialog Applications. *CoRR*, abs/2201.08239, 2022. arXiv: 2201.08239.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., Choi, D. H., Powell, R., Ewalds, T., Georgiev, P., Oh, J., Horgan, D., Kroiss, M., Danihelka, I., Huang, A., Sifre, L., Cai, T., Agapiou, J. P., Jaderberg, M., Vezhnevets, A. S., Leblond, R., Pohlen, T., Dalibard, V., Budden, D., Sulsky, Y., Molloy, J., Paine, T. L., Gülçehre, C., Wang, Z., Pfaff, T., Wu, Y., Ring, R., Yogatama, D., Wünsch, D., McKinney, K., Smith, O., Schaul, T., Lillicrap, T. P., Kavukcuoglu, K., Hassabis, D., Apps, C., and Silver, D. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nat.*, 575(7782):350–354, 2019. doi: 10.1038/s41586-019-1724-z.

Wang, X., Huang, Q., Celikyilmaz, A., Gao, J., Shen, D., Wang, Y.-F., Wang, W. Y., and Zhang, L. Reinforced Cross-Modal Matching and Self-Supervised Imitation Learning for Vision-Language Navigation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 6629–6638. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.00679.

Waxman, S. R. and Markow, D. Words as Invitations to Form Categories: Evidence from 12- to 13-Month-Old Infants. *Cognitive Psychology*, 29:257–302, 1995.

Widrich, M., Hofmarcher, M., Patil, V. P., Bitto-Nemling, A., and Hochreiter, S. Modern Hopfield Networks for Return Decomposition for Delayed Rewards. In *Deep RL Workshop NeurIPS 2021*, 2021.

Wilcoxon, F. Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, 1(6):80–83, 1945. ISSN 00994987. Publisher: [International Biometric Society, Wiley].

Xing, C., Wang, D., Liu, C., and Lin, Y. Normalized Word Embedding and Orthogonal Transform for Bilingual Word Translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1006–1011, Denver, Colorado, May 2015. Association for Computational Linguistics. doi: 10.3115/v1/N15-1104.

Yao, S., Rao, R., Hausknecht, M. J., and Narasimhan, K. Keep CALM and Explore: Language Models for Action Generation in Text-based Games. In Webber, B., Cohn, T., He, Y., and Liu, Y. (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pp. 8736–8754. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.704.

Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., and Wu, Y. CoCa: Contrastive Captioners are Image-Text Foundation Models. *CoRR*, abs/2205.01917, 2022. doi: 10.48550/arXiv.2205.01917. arXiv: 2205.01917.

Zeng, A., Wong, A., Welker, S., Choromanski, K., Tombari, F., Purohit, A., Ryoo, M. S., Sindhwani, V., Lee, J., Vanhoucke, V., and Florence, P. Socratic Models: Composing Zero-Shot Multimodal Reasoning with Language. *CoRR*, abs/2204.00598, 2022. doi: 10.48550/arXiv.2204.00598. arXiv: 2204.00598.

Zenke, F., Poole, B., and Ganguli, S. Continual Learning Through Synaptic Intelligence. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3987–3995. PMLR, 2017.

Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., Mihaylov, T., Ott, M., Shleifer, S., Shuster, K., Simig, D., Koura, P. S., Sridhar, A., Wang, T., and Zettlemoyer, L. OPT: Open Pre-trained Transformer Language Models. *CoRR*, abs/2205.01068, 2022. doi: 10.48550/arXiv.2205.01068. arXiv: 2205.01068.

Zhang, Y., Gaddy, D., Barzilay, R., and Jaakkola, T. Ten Pairs to Tag – Multilingual POS Tagging via Coarse Mapping between Embeddings. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1307–1317, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1156.

Zheng, Q., Zhang, A., and Grover, A. Online Decision Transformer. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 27042–27059. PMLR, 2022.

Åström, K. J. Optimal control of Markov processes with incomplete state information. *Journal of Mathematical Analysis and Applications*, 10:174–205, 1964.

A Supplementary Material

A.1 Environments

We choose 8 diverse 3D environments of the MiniWorld benchmark suite:

- **CollectHealth:** The agent spawns in a room filled with acid and must collect medikits in order to survive as long as possible.
- **FourRooms:** The agent must reach a red box that is located in one of four interconnected rooms.
- **MazeS3Fast:** A procedurally generated maze in which the agent needs to find a goal object.
- **PickupObjs:** Several objects are placed in a large room and must be collected by the agent. Since the agent receives a reward of 1 for each collected object, the reward is unbounded.
- **PutNext:** Several boxes of various colors and sizes are placed in a big room. The agent must put a red box next to a yellow one.
- **Sign:** The agent spawns in a U-shaped maze containing various objects of different colors. One side of the maze contains a sign which displays a color in written form. The aim is to collect all objects in the corresponding color.
- **TMaze:** The agent must navigate towards an object that is randomly placed at either end of a T-junction.
- **YMaze:** Same as TMaze, but with a Y-junction.

We neglect the OneRoom and the Hallway environments, since those are easily solved by all our methods. Further, we neglect the Sidewalk environment since it is essentially the same task as Hallway with a different background. Since the reward of PickupObjs and CollectHealth are unbounded, we normalize them to be in the range of $(0, 1]$, which is the reward received in all other environments. For a more detailed description of the MiniGrid environments we refer the reader to Paischer et al. (2022).

A.2 Mapping between Language Embedding Spaces

We consider all publicly available CLIP backbone variants and align their output spaces with the TrXL used in HELM. According to Algorithm 1 we determine the overlap in the CLIP and TrXL vocabularies. In total, there are 5285 tokens that appear in both, $\mathcal{V}_{\text{CLIP}}$, and $\mathcal{V}_{\text{LM}}$. Prior work has found that as little as ten word correspondences are sufficient to train an orthogonal mapping between monolingual embedding spaces of closely related languages (Zhang et al., 2016). This assumes a certain degree of isomorphism between the embedding spaces. Since we train a mapping between embedding spaces of the same language, we expect this assumption to hold in our setting as well. The CLIP output space and the TrXL input space differ greatly in their statistics, therefore we perform centering and scaling as preprocessing. This way, during inference in the RL experiments we can simply re-center and re-scale by the statistics of LM space to obtain a suitable input for the TrXL.

Table 2 shows the average training and test accuracy across 5 folds for different linear projection methods. We find that generally there is a tendency that the mapping improves for larger backbones. The Procrustes variants consistently exhibit the highest average accuracy on the test set across 5 folds. We do not show variance estimates, because those are negligibly small.

A.3 Mapping Images to the Language Space

We complement our findings on the ranking experiment in Section 4 with results for additional vision backbones of CLIP. Table 1 shows the relative NDCG to ranking in CLIP space, while Table 3 shows the MRR for the different image encoders.

A.4 Hyperparameter Search

We adapt the hyperparameter search conducted in Paischer et al. (2022). Particularly, we search for learning rate in $\{5\text{e-}4, 3\text{e-}4, 1\text{e-}5, 5\text{e-}5\}$, entropy coefficient in $\{0.05, 0.01, 0.005, 0.001\}$, rollout

	Linear	Ridge	Procrustes	RobProc
RN50	0.733/0.172	0.529/0.246	0.637/**0.289**	0.658/0.285
RN101	0.523/0.243	0.522/0.243	0.618/**0.304**	0.65/0.303
RN50x4	0.581/0.239	0.579/0.238	0.675/**0.319**	0.701/0.309
RN50x16	0.647/0.233	0.645/0.233	0.718/**0.332**	0.742/0.33
RN50x64	0.75/0.241	0.74/0.235	0.737/**0.342**	0.752/0.34
ViT-B/32	0.531/0.258	0.529/0.26	0.592/**0.308**	0.616/0.3
ViT-B/16	0.541/0.268	0.538/0.268	0.613/**0.329**	0.638/0.327
ViT-L/14	0.656/0.272	0.632/0.28	0.664/**0.351**	0.683/0.346
ViT-L/14*	0.657/0.271	0.632/0.281	0.662/**0.353**	0.68/0.348

Table 2: Train/Test accuracy for different linear models optimized for mapping CLIP tokens to the TrXL embedding space. Average over 5-fold cross validation is shown. ViT-L/14* received images resized to 336 pixels as input during pretraining.

	Linear	Ridge	Procrustes	RobProc	CLIP
RN50	0.036±0.108	0.107±0.225	0.229±0.323	**0.232±0.336**	0.514±0.4
RN101	0.052±0.153	0.095±0.2	**0.3±0.365**	0.282±0.359	0.532±0.393
RN50x4	0.03±0.082	0.074±0.179	**0.308±0.37**	0.284±0.353	0.524±0.391
RN50x16	0.025±0.084	0.074±0.174	**0.289±0.367**	0.281±0.361	0.52±0.4
RN50x64	0.052±0.158	0.063±0.174	**0.285±0.362**	0.276±0.355	0.532±0.396
ViT-B/32	0.035±0.109	0.067±0.175	**0.306±0.363**	0.289±0.354	0.558±0.405
ViT-B/16	0.038±0.136	0.092±0.208	**0.349±0.378**	0.332±0.379	0.554±0.4
ViT-L/14	0.037±0.111	0.063±0.155	**0.309±0.372**	0.301±0.368	0.552±0.4
ViT-L/14*	0.039±0.11	0.065±0.155	**0.327±0.376**	0.318±0.371	0.575±0.393

Table 3: Mean Reciprocal Rank (MRR) for ranked tokens in the LM embedding space given an image as input and applying our mapping. Image-caption pairs are drawn from the MSCOCO dataset. MRR in the CLIP output space serves as upper bound. ViT-L/14* receives images resized to 336 pixels as input.

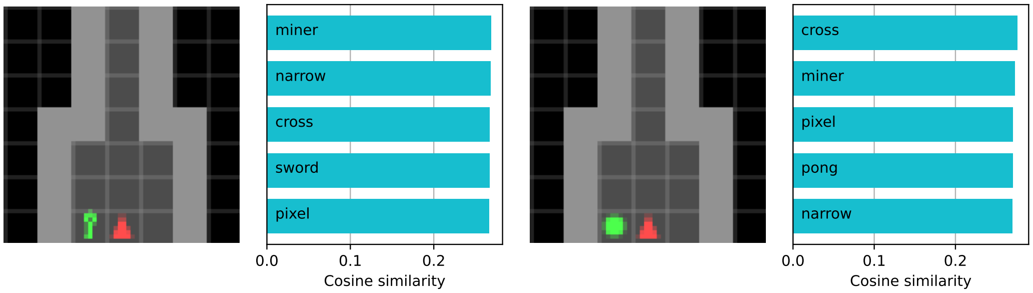


Figure 8: Top 5 closest tokens for observations of Memory environment in the CLIP space.

length in $\{32, 64, 128\}$ for HELMv2, and HELMcs. Since our analysis in Section 3 showed that for $\beta = 1, 10$ the spatial compression collapses to the mean over pretrained token embeddings, we alter the grid for β of HELM to $\{100, 500, 1000, 5000\}$. To decrease wall-clock time of HELM variants, we vary the size of the memory register of TrXL such that it can fit the maximum episode length (Table 6). We lower the number of interaction steps for the gridsearch if we observe convergence before the 500k interaction steps. If no convergence is observed within the 500K interaction steps, we tune for the entire duration. We apply the same scheme for tuning the LSTM baseline and tune the same hyperparameters as in Paischer et al. (2022).

	MRR (↑)	aNDCG (↑)
Random	0.016±0.058	0.243±0.029
HELMv2	0.018±0.057	0.244±0.031
HELMv2+RandOrtho	0.019±0.068	0.245±0.033
SHELM+VocabAttn, $\beta = 1$	0.002±0.002	0.23±0.023
SHELM+VocabAttn, $\beta = 10$	0.003±0.005	0.234±0.024
SHELM+VocabAttn, $\beta = 100$	0.308±0.39	0.323±0.091
SHELM+VocabAttn, $\beta = 1e3$	0.27±0.401	0.309±0.085
SHELM+VocabAttn, $\beta = 1e4$	0.267±0.402	0.307±0.085
SHELM	**0.349±0.378**	**0.35±0.088**

Table 4: MRR and aNDCG for tokens ranked in the LM embedding space for different ablation setups.

Vision Backbone	Approximate Parameter Count
CLIP-RN50	102M
CLIP-RN101	120M
CLIP-RN50x4	180M
CLIP-RN50x16	290M
CLIP-RN50x64	623M
CLIP-ViT-B/16	149M
CLIP-ViT-B/32	151M
CLIP-ViT-L/14	427M
CLIP-ViT-L/14*	427M
RN34-IN	21M
ViT-L/16-IN	325M

Table 5: Parameter count of different publicly available vision backbones used for HELMv2. ViT-L/14* receives images resized to 336 pixels as input.

Environment	Memory register of TrXL
DoorKey5x5	256
DoorKey6x6	256
DynamicObstacles	64
KeyCorridor	256
RedBlueDoors	512
Unlock	256
CollectHealth	64
FourRooms	256
MazeS3Fast	256
PickupObjs	512
PutNext	256
Sign	32
TMaze	256
YMaze	256

Table 6: Length of memory register of TransformerXL used in HELM, HELMcs, and HELMv2 for selected MiniGrid (top) and MiniWorld (bottom) environments.