Eye of the Beholder: Improved Relation Generalization for Text-Based Reinforcement Learning Agents

Keerthiram Murugesan, Subhajit Chaudhury, Kartik Talamadupula

IBM Research

Abstract

Text-based games (TBGs) have become a popular proving ground for the demonstration of learning-based agents that make decisions in quasi real-world settings. The crux of the problem for a reinforcement learning agent in such TBGs is identifying the objects in the world, and those objects' relations with that world. While the recent use of text-based resources for increasing an agent's knowledge and improving its generalization have shown promise, we posit in this paper that there is much yet to be learned from visual representations of these same worlds. Specifically, we propose to retrieve images that represent specific instances of text observations from the world and train our agents on such images. This improves the agent's overall understanding of the game *scene* and objects' relationships to the world around them, and the variety of visual representations on offer allow the agent to generate a better generalization of a relationship. We show that incorporating such images improves the performance of agents in various TBG settings.

Introduction

Reinforcement Learning (RL) has seen a resurgence in recent years thanks to advances in representation, inference, and learning techniques – led by a massive scale-up and investment in deep neural network-based methods. Successful applications of RL have included domains such as Chess (Silver et al. 2018), Go (Silver et al. 2017), and Atari games (Mnih et al. 2016). However, with the emergence of natural language processing (NLP) as a key AI application area, research attention has turned towards text-based applications and domains. These domains offer their complexity challenges for RL algorithms, including large and intractable action spaces – the space of all possible words and combinations; partial observability of the world state; and under-specified goals and rewards.

Text-based games (TBGs) have emerged as prime exemplars of the above challenges. Inspired by games such as Dungeons & Dragons and Zork, researchers have worked on putting together challenging environments that offer the complexities of real-world interactions but in sandbox settings suitable for the training of RL agents. The foremost such example is *TextWorld* (Côté et al. 2018), an open-source text-based game engine that allows for the generation of text-based game instances and the evaluation of agents on those games. Much of the recent work on text-based RL (Ammanabrolu and Riedl 2019; Dambekodi et al. 2020; Murugesan et al. 2021) has focused on the *TextWorld* environment, and on imbuing agents with additional information to make them learn, scale, and act more efficiently.

However, much of the information that has been used in the prior art to improve the performance of AI agents in TBGs is still restricted to the medium of text. In contrast, when humans encounter games such as Zork and *TextWorld*, they do not restrict themselves to only textual information. Indeed, they are able to generalize to environments and the actions within them by considering not just the form of information provided by the environment; but also by *imagining* or visualizing various forms of that information. This imagination is key to generalizing beyond merely the information present in the instance currently under consideration. In this work, we posit that using images – either retrieved or imagined (generated) – that represent information from the game instance can help improve the performance of RL agents in TBGs.

Specifically, we consider RL agents in the *TextWorld* and *Jericho* TBG environments; and additional information that can be provided to such agents to improve their performance. Past work has focused on trying to use external knowledge to either limit (Chaudhury et al. 2020) or enhance (Murugesan et al. 2021) the space of actions: however, this has also been restricted to the text modality. At their crux, these efforts are all trying fundamentally to solve the problem of *relationships* within the environment – *how are different things in the world related to each other?* And how can the agent manipulate these relations to convert the initial state of the world – via a sequence of observations – into the desired goal state (or to maximize reward)? Purely text-based information is extremely sparse and is unable to sufficiently abstract the notion of relationships.

Consider for example the relationship `at` - a `patio chair` is `at` the `backyard`. What does this relation mean - what is the *at*-ness? Text cannot convey this information effectively on its own: as the size of the underlying vocabulary increases, the natural language space gets sparser and it

[*] Corresponding Author: keerthiram.murugesan@ibm.com. Code and data can be found at https://github.com/IBM/IFSceneIT.

becomes harder to extract signals to understand relationships between objects (in this case, 'patio chair' and 'backyard'). Images, on the other hand, go a bit further in conveying the meanings of relationships as understood by humans (Chen and Lawrence Zitnick 2015). Images also help generalize better: in text, a patio chair is always represented as `patio chair`; yet in a visual medium, there can exist different kinds of patio chairs, with different properties such as shape, size, color, texture, surroundings, etc.

In this paper, we introduce the `Scene Images for Text-based Games` (`SceneIT`) model (pronounced "*seen-it*") that integrates an external repository of images as additional knowledge for an RL agent in text-based game environments; and measure the performance of this model against the state-of-the-art text-only method. Our images come from two sources: pre-retrieved from prior existing images; and generated anew based on textual descriptions. We show that an agent with access to this additional visual information does significantly better, and examine some specific instances that show the reason for this improved performance.

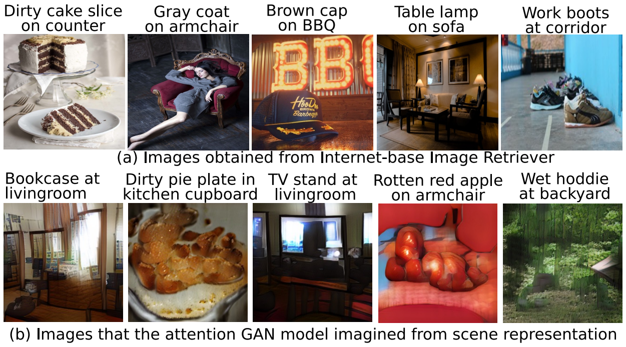
Methodology

Text-based reinforcement learning agents for TBGs interact with the environment only using the modality of text (Narasimhan, Kulkarni, and Barzilay 2015; He et al. 2016). TBGs convey the state of the game at every step as observations in natural language text, and the text-based RL agent learns to map the current state to one of the admissible actions (also in the text modality) available to it. Most current text-based RL agents (e.g. (Murugesan et al. 2021)) focus on integrating additional textual knowledge to learn and act in a complex environment. Such agents thus lack the ability for human-like imagination involved in solving TBGs efficiently.

In this section, we outline the methodology that we use to integrate the visual (image) representation of a game scene using our `SceneIT` approach for TBGs. In order to obtain the visual representation of the scene that the agent is currently situated in, as the first step, we extract noun phrases that represent objects and relational phrases between the objects in the scene from the text observation – for example, `kitchen of the white house`, `bottle on the table`, `desk chair at bedroom`, etc. These phrases portray the scene in terms of which object is located at what location, which we intend to use to create a "visual mind-map" of the scene for the agent. Since the key component and novelty of our system is the usage of images for the TBGs under consideration, we first outline the collection process for such images. Our technique relies on two main sources of images: *retrieval* from the internet, and *generation* from pre-existing models for imagining and generating visual scenes. We describe each of these methods in detail below.

Collecting Images

Retrieving Images from the Internet: In order to obtain images from the internet, we design an image retriever that obtains the best matching image from the list of query strings (noun phrases) that are used to represent the scene. This process also ties into one of the central motivations of our work, which is that images offer more signals to agents as they try to abstract, represent, and use the relationships between different objects in a scene.



(a) Images obtained from Internet-base Image Retriever

(b) Images that the attention GAN model imagined from scene representation

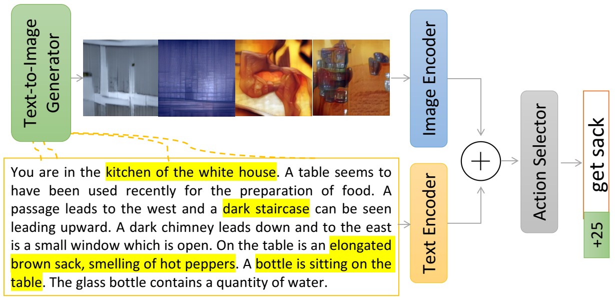
Figure 1: Examples of images obtained from (a) the web-based image retriever, and (b) imagination via AttnGAN (Xu et al. 2018a). The phrase used to retrieve or generate the picture is indicated above the respective picture.

To provide good generalization behavior, we design an image retriever that automatically searches the internet for a given query string without any human supervision [1]. In addition, we use image caching to improve the speed of retrieval such that the images corresponding to encountered queries are saved to disk and need not be downloaded from the web while training the agent. It is to be noted that the caching process is completely generic and does not involve saving specific situation-relevant images. Figure 1(a) provides some examples of images that are retrieved from the internet for specific phrases.

Imagining Images from Generative Models: The previous method of "visual mind-map" extraction uses pre-existing images from the internet for scene representation. Such a scene representation is useful for a human to visually parse the scene. However, we also explore the potential for representing visual scenes using images that are *imagined* by generative models. Our hypothesis is that such images can also provide useful visual features to improve generalization in tasks from *TextWorld* (and other text-based games).

We use the Attentional Generative Adversarial Network (AttnGAN) (Xu et al. 2018a) for attention-driven text-to-image generation. This generative model uses a multi-stage refinement for fine-grained generation of images from a given text snippet. AttnGAN gives attention to the relevant tokens in the natural language query in order to generate details at different sub-regions of the image. For our approach, we pre-train the AttnGAN model on the MS-COCO dataset (Lin et al. 2014). The queries used for image generation are the same as the ones used for the previous internet retrieval-based scene representation. We hypothesize that although such images may not always be interpretable by humans – see Figure 1(b) for a few examples – such images can provide some latent image features for neural models that might contribute to better generalization in *TextWorld* games.

[1] Based on Google Image Retriever: https://github.com/Joeclinton1/google-images-download



You are in the kitchen of the white house. A table seems to have been used recently for the preparation of food. A passage leads to the west and a dark staircase can be seen leading upward. A dark chimney leads down and to the east is a small window which is open. On the table is an elongated brown sack, smelling of hot peppers. A bottle is sitting on the table. The glass bottle contains a quantity of water.

Figure 2: Overview of our methodology of scene representation for a sample text observation taken from `Zork1` using text-to-image generative model. Highlighted text snippets show some of the phrases used by the agent to generate relevant images for scene representation.

Model Description

We now detail the models that we used to use and encode the images retrieved or generated in the previous step. Figure 2 shows the architecture overview of our proposed approach for scene representation using the AttnGAN (Xu et al. 2018a) based text-to-image generation. In order to capture the textual features from the text observation from the game, we use Stacked GRU as our text encoder: this keeps tracks of the state of the game across time steps. Once we have the images retrieved/generated from the text snippets, we extract the image features using image encoders which are combined with the features from textual inputs to obtain the action scores.

Specifically, we use Resnet-50 for encoding the retrieved images and for the images generated from the pre-trained AttnGAN. The text and image encoding features are then concatenated and passed to the action selector (as shown in Figure 2), which maps the encoding features to action scores using a multi-layer perceptron (MLP) to select the next action. Based on the reward from the game environment, we update text and image encoders and the action selector. Since the reward from the game can guide the text-to-image generator (AttnGAN) to generate meaningful images for the current context of the game, we finetune the pre-trained AttnGAN along with the encoders and the action selector to yield the best results. In this case, we use the inbuilt CNN-based image encoder (Inception v3 (Szegedy et al. 2016)) to map the generated images to the image features. We call this model `SceneIT` and use it by default for all our experiments in this paper.

Experimental Results

In this section, we present experimental results that demonstrate the advantage of our proposed `Scene Images for Text-based Games` approach – which makes use of images in addition to text – over existing state-of-the-art techniques that are text-only. We conduct our performance evaluation on three datasets: TextWorld Commonsense (*TWC*) [2], the First TextWorld Problems (*FTWP*) [3] and *Jericho*[4]. The *TWC* and *FTWP* datasets build on the Microsoft *TextWorld* Environment (Côté et al. 2018), and offer complementary tests: while *TWC* tasks require the retrieval and use of commonsense knowledge for more efficient solution, the *FTWP* problems test the agent's exploration capabilities. *Jericho* is a suite of 33 interactive fiction games that measures human performance on text-based games by offering stories from different domains – in our case, it helps evaluate the breadth and coverage of the image generation.

Distribution: In these datasets, a set of text-adventure games are provided for training reinforcement learning (RL) agents. In addition to these training games, the datasets contain two test sets of games: 1) Test games (IN) that are generated from the same distribution as the training games – these games contain similar sets of entities and relations as the train games; and 2) Test games (OUT), which contain games generated from a set of entities that have no overlap with the training games. This is a way of testing whether the RL agent can generalize its behavior to new and unseen games by leveraging the state observation from the TextWorld environment – and additionally in our case, the visual relationships between entities.

Agents: We compare three RL agents in our experiments: 1) *Random*, where the actions are selected randomly at each step; 2) *Text-Only*, where the actions are selected solely based on the textual observation available at the current step. We use three baseline text-only methods - DRRN (He et al. 2016), Template DQN (Hausknecht et al. 2019) and KG-A2C (Ammanabrolu and Hausknecht 2020); and 3) Our method – `SceneIT` – explained in the previous section, where the RL agent is allowed to imagine visual scenes and images using Attention GAN (Xu et al. 2018b), a Text-to-Image generator based on Generative Adversarial Networks (GAN) (Goodfellow et al. 2014).

Metrics: In our experiments, we measure the performance of various agents using two metrics: (1) *Average Normalized Score* – calculated as the total score achieved by an agent normalized by the maximum possible score for the game); and (2) *Average Steps Taken* – calculated as the total number of steps taken by the agent to complete the goals. A higher score is better, while a lower number of steps taken is better.

Quantitative Results

We first present the results of a quantitative evaluation of our proposed technique. In order to provide a well-rounded evaluation, we consider different text-based games: the *TWC* and *FTWP* problems, both based on the *TextWorld* (Côté et al. 2018) domain; and the *Jericho* (Hausknecht et al. 2019) domain, based on interactive fiction (IF) games. Detailed experimental setting are provided in the supplementary material.

Experiments on *TextWorld Commonsense* The first domain that we conduct our evaluation on is the *TextWorld Commonsense* (Murugesan et al. 2021) domain. This domain

[2] https://github.com/IBM/commonsense-rl

[3] https://competitions.codalab.org/competitions/21557

[4] https://github.com/microsoft/jericho

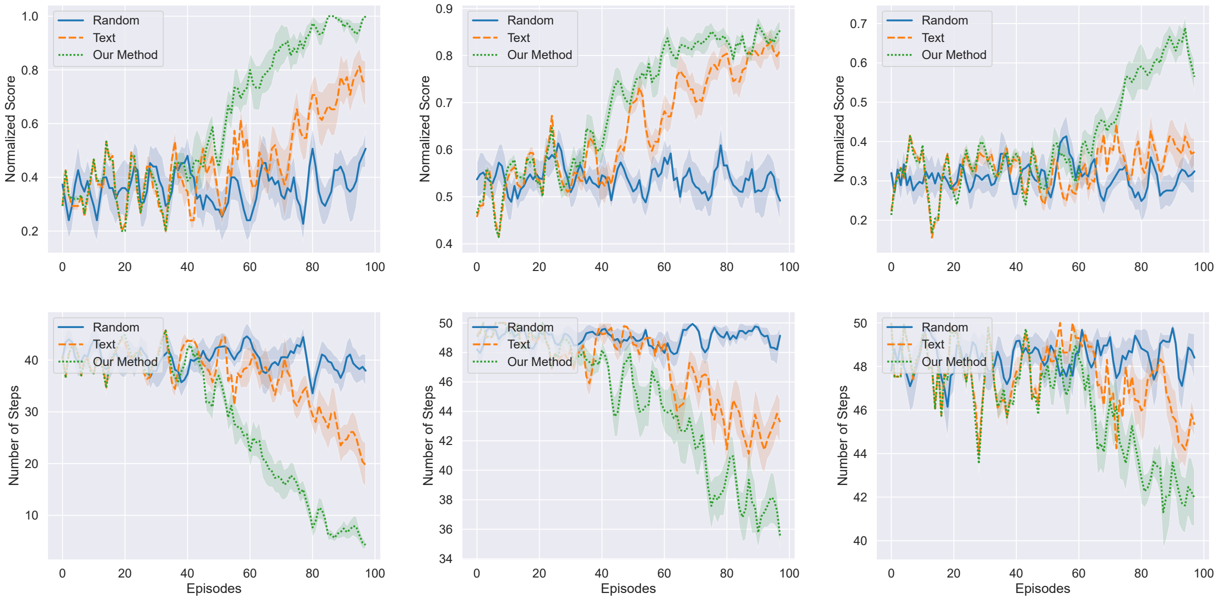


Figure 3: Training performance (showing mean and standard deviation averaged over 5 runs) for the three difficulty levels: Easy (left), Medium (middle), Hard (right). Higher normalized score is better, while lower number of steps is better. `Our Method` refers to our `SceneIT` technique.

`Norm. Score`
`(Num. Steps)`
	Test Games (IN)				Test Games (OUT)		

Level Model	Easy	Medium	Hard		Easy	Medium	Hard
Random	0.52 (38.52)	0.49 (49.66)	0.49 (46.21)		0.51 (38.92)	0.54 (48.94)	0.31 (48.95)
Text	0.82 (22.73)	**0.74** (46.36)	0.62 (39.54)		0.75 (30.18)	0.69 (46.29)	0.41 (46.90)
`SceneIT`	**0.96 (13.38)**	0.70 (**46.15**)	**0.77 (34.65)**		**0.88 (19.58)**	**0.78 (38.18)**	**0.59 (44.08)**

Table 1: Test performance (averaged over 5 runs) on the normalized score (higher is better) and number of steps (lower is better) metrics for the three difficulty levels.

is an extension of the *TextWorld* domain that adds scenarios where commonsense knowledge is required in order to arrive at efficient solutions.

Difficulty Levels: The *TWC* domain comes with difficulty levels for the problem instances associated with it, defined in terms of how hard it is for an agent (human or AI) to solve that specific instance. The difficulty of a level is set as a combination of the number of goals to be achieved, the number of actions (steps) required to achieve them, and the number of objects and rooms in the instance (which may be related to goal achievement, or may simply be distractors). In our evaluation for this work, we consider three distinct difficulty settings. In increasing order of hardness, these are: `easy`, `medium`, and `hard`. We follow Murugesan *et al.* (Murugesan et al. 2021) – who introduce the *TWC* domain, and are the current state-of-the-art on this domain – in choosing these difficulty levels.

Training Performance: Figure 3 shows the training performance of three different agents/models on the *TWC* problems for the three difficulty levels discussed above. For each level, the performance is reported via the normalized score (higher is better) as well as the average number of steps (lower is better). It is clear that `SceneIT` – with access to both the textual representation of the observations from the game, as well as the image/visual representation – does much better in all three settings. Furthermore, beyond the 60 episode mark, there is a clear divergence of our technique from the random and text-only baselines.

Test Performance: Table 1 shows the test results for 3 models - one random baseline, one text-only baseline, and `SceneIT` – which combines the text features with image features from the finetuned AttnGAN. We split our reporting across two conditions: Test games (IN) reports on test games that come from the same distribution as the training games; while Test games (OUT) reports on test games from outside the distribution of training games. It is clear that for both conditions, `SceneIT` is the state-of-the-art in 11 out of 12 instances – handily beating the existing text-only state-of-the-art (`Text`). In the one case where it is not the best (`medium` for `in` distribution), it is very close to the performance of the best performing model. This shows the added advantage of using visual features in addition to textual features when

	Human			Baselines		Ours
Game	**Max**	**Walkthrough-100**	**TDQN**	**DRRN**	**KG-A2C**	**SceneIT**
detective	360	350	169	197.8	207.9	**317.7**
enchanter	400	125	8.6	20	12.1	**21.6**
inhumane	90	70	0.7	0	3	**15.83**
karn	170	40	0.7	**2.1**	0	0.0
snacktime	50	50	9.7	0	0	**20**
spellbrkr	600	160	18.7	37.8	21.3	**40**
zork1	350	102	9.9	32.6	34	**43.58**
zork3	7	3	0	0.5	0.1	**2.67**

Table 2: Raw scores on a subset of Jericho games (selected randomly based on the difficulty level) achieved by the agents (proposed and baseline) averaged over 10 runs.

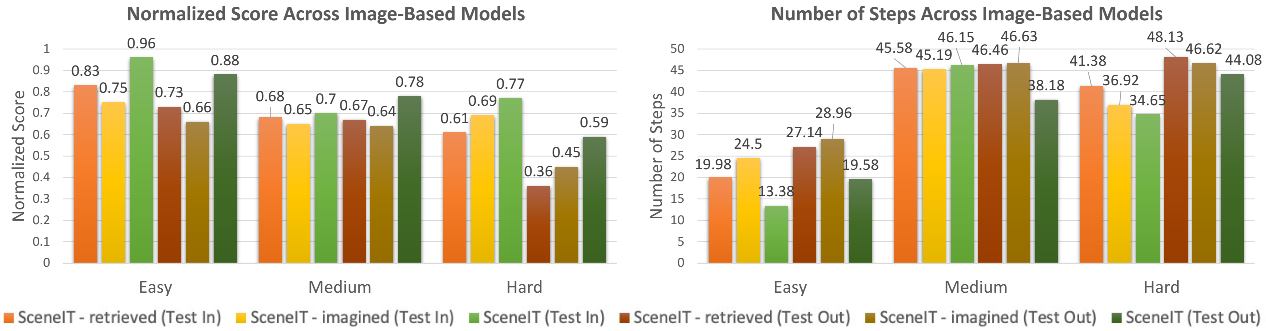


Figure 4: Results showing an improvement across both normalized score (higher is better) and number of steps (lower is better) by using images on the TWC dataset with different difficulty levels.

solving *TWC* games, thus validating the central hypothesis of our work.

Experiments on *First TextWorld Problems* In this section, we present the results of running the various agents/models on the *First TextWorld Problems* (*FTWP*) dataset. Figures 5 (*left* and *middle*) show the results across the `in` and `out` distributions, as introduced previously. Since the cooking task in FTWP focuses more on exploration rather than the meaningful relationship between the objects (as in *TWC*) to improve the performance, we can see that `SceneIT` shows results that are comparable to and even worse than the text-only model: this shows that merely adding images to a game does not always necessarily improve the metrics.

Experiments on *Jericho* Next, we consider *Jericho* (Hausknecht et al. 2019), a benchmark dataset in TBGs that consists of 33 popular interactive fiction (IF) games developed for humans a decade ago. We select a subset of games from different difficulty levels for our experiments. From Table 2, we can see that `SceneIT` outperforms the other state-of-the-art text-only baselines (Template DQN (Hausknecht et al. 2019), DRRN (Narasimhan, Kulkarni, and Barzilay 2015; He et al. 2016), and KG-A2C (Ammanabrolu and Hausknecht 2019)) by a significant margin. Our approach is currently able to achieve the best score (averaged over 10 runs) on 7/8 games from across the difficulty levels.

Images: Retrieval vs. Generation After establishing that the addition of the visual features from images that represent the scene described by the textual observations from the game does indeed help the performance of agents, we now explore further into the comparison between these different agents. Specifically, we compare the three models described in Section : `SceneIT` with retrieved images from the internet, `SceneIT` with generated/imagined images from the pretrained AttnGAN, and `SceneIT` with finetuned AttnGAN. This comparison is presented as a bar chart in Figure 4. As in the previous experiments, we plot the three difficulty levels across two conditions: `in` and `out` of distribution. We use a lighter shade of the corresponding color for the former, and a darker shade for the latter. It is clear that `SceneIT` – which combines text features with features from AttnGAN – outperforms the other two image baselines across different difficulty levels and conditions.

Finetuning image encoder It is worth noting that from the results in Figure 4, we see that the performance of SceneIT retrieved and imagined are not significantly better than the text-only agent. We strongly believe that one of the reasons lies in ImageNet pretrained Resnet-50 for image encoding. For these models, we didn't finetune the Resnet-50 model for image encoding as we did in the SceneIT with ModelGAN. Our assumption was using the Imagenet pre-trained Resnet-50, we will have a general image representation for common objects used in the games. To do so, the Imagenet classes should be diverse enough to cover all the objects seen in the TWC games.

To evaluate this, we performed a simple analysis on how

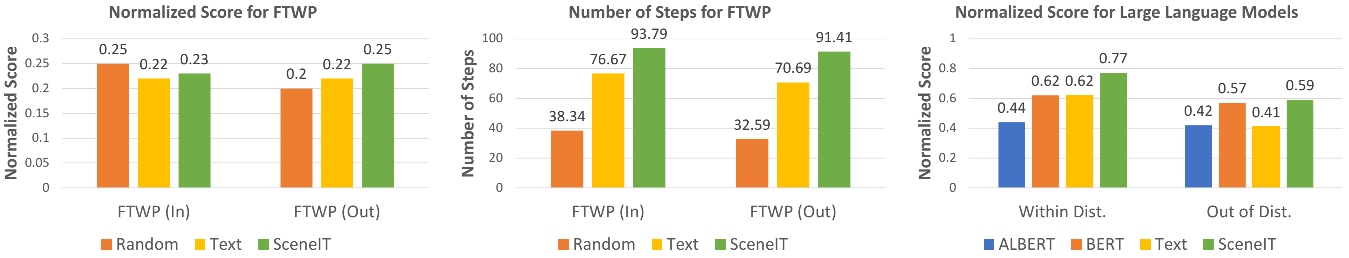


Figure 5: Test-set performance on FTWP Cooking Task (averaged over 5 runs) on the normalized score (*left:* - higher is better) and the number of steps (*middle:* lower is better) metrics. Figure *(right)* showing the normalized score of large language models compared to our proposed SceneIT model for TWC hard test level games.

well these classes represent or overlap with the objects that appeared in the game. We computed the average of the (GloVe embedding-based) similarity between objects in the games and the best matching class from the Imagenet's 1000 class labels. For TWC hard games, we obtained a similarity of 0.75, 0.73, and 0.76 for train, test (IN), and test (OUT) splits. Similarly, when comparing overlap between the objects seen in the TWC games and the entities in the caption text used in MS-COCO to train the AttnGAN, we get 0.93, 0.94, and 0.98 for train, test (IN) and test (OUT) splits in TWC hard games. We see that the SceneIt with finetuned AttnGAN has more advantage here. Based on this analysis, one simple solution to improve the performance of the SceneIT retrieved and imagined models is to finetune the Resnet-50 just like we finetuned the AttnGAN for SceneIT ModelGAN.

To verify this, we experiment with finetuning the Resnet-50 image encoder. As a result, for the internet-based retriever model, we found that the model with finetuned Resnet-50 gives metrics of 0.66/40.43 compared to the non-finetuned metrics of 0.61/41.38 for within distribution of hard test games. For out-of-distribution games, the model with finetuned Resnet-50 gives metrics of 0.45/46.88 compared to the non-finetuned metrics of 0.36/48.13. Our proposed SceneIT with finetuned AttnGAN still outperforms these reported results above.

Qualitative Results

In addition to the quantitative results described previously, we also present some qualitative examples of what the SceneIT agent focuses on as it uses images (retrieved or imagined) in order to solve specific problem instances. To illustrate this effectively, we use the notion of attention activation maps (Zhou et al. 2016; Selvaraju et al. 2017; Lu et al. 2012; Gupta, Dileep, and Thenkanidiyoor 2021), which can be used to demonstrate parts of an image that an agent/technique is attending to. We split our analysis into the two main ways in which we currently produce images for use by SceneIT: retrieval, and imagination (see Section).

Figure 6 shows examples of this for the imagined images. We present both the imagined images as well as the activation maps overlaid over those respective images for a given set of text phrases from the game observation. For example, the agent can focus on the right part of the image that is imagined for the phrase `wet brown dress on patio chair`; and can then choose the action `examine patio chair`. The other examples also illustrate a similar pattern.

Ablation Studies

We perform various ablation studies to better analyze the inner-workings of our proposed model as described below.

Performance with random images First, we fed randomly selected images to the image encoder to demonstrate whether the retrieved images were useful in generalizing objects' relationships to the world. In the Internet retriever-based model in Figure 4, we replaced the images retrieved for the extracted text phrases with random images. For TWC (Hard level) games, these are the results for the randomly retrieved images during training and test time. Test games (In: within the distribution) decreased to 0.57/42.6 (norm score/avg. steps), compared to the reported 0.61/41.38 for the proposed model. Test games (Out: out of distribution) got 0.25/49.51, compared to the reported result of 0.36/48.13. Since the SceneIT model imagines a text using AttnGAN, we replaced the extracted text phrases with random text phrases. For TWC (Hard level) games, we have: Test games within the distribution got 0.73/36.48 compared to the reported 0.77/34.65. Test games out of distribution got 0.45/46.83 compared to the reported result of 0.59/44.08.

For Jericho games, these are the results with random images: Detective got 246.68 (22.4% decrease from the reported raw score of 317.7), Enchanter got 20 (7.4% decrease from 21.6) and Zork1 got 38.3 (12.1% decrease from 43.58). Therefore, we have shown that the performance of the SceneIT agents with random images drops compared to the reported results in the paper, demonstrating that the image gives valuable knowledge to the agents.

Ablation with a large number of parameters Now we show that our proposed method improves relational generalization using generated images and not due to an increase in the number of model parameters. To show this, we replaced the text encoders (Stacked GRU) used in the text-only agents with large-scale language models such as BERT (Devlin et al. 2018) and ALBERT (Lan et al. 2019) for a fair comparison of text-only agents with the SceneIT model. We show the normalized score for various models in Figure 5 (*right*). The number of parameters for text-only Agent (Stacked GRU) is 0.20M, ALBERT based text agent is 11M, BERT based text agent is 109M and SceneIT is 10M. We observe that even adding additional parameters to the text-only agents (via

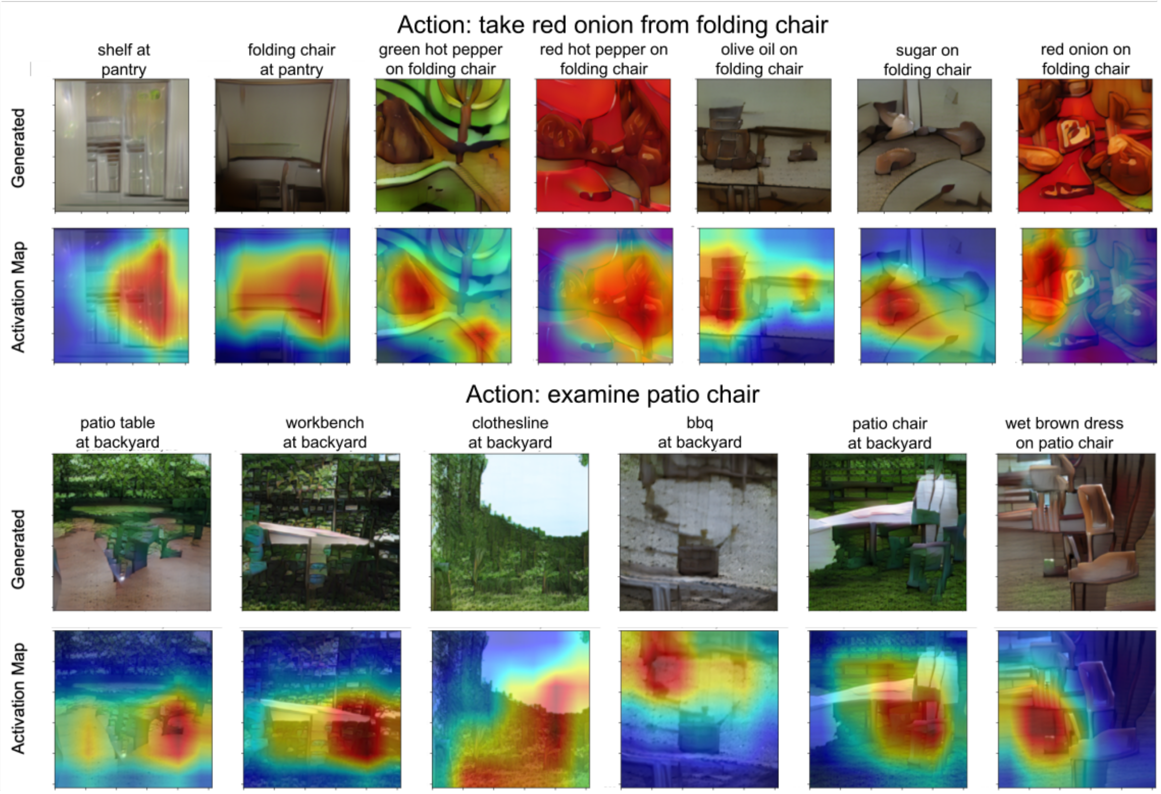

Action: take red onion from folding chair
shelf at pantry
folding chair at pantry
green hot pepper on folding chair
red hot pepper on folding chair
olive oil on folding chair
sugar on folding chair
red onion on folding chair
Generated
Activation Map
Action: examine patio chair
patio table at backyard
workbench at backyard
clothesline at backyard
bbq at backyard
patio chair at backyard
wet brown dress on patio chair
Generated
Activation Map

Figure 6: Activation maps showing the region of interest when producing the action command in each case, using the imagination based model for *TWC*. We include both the generated images and its attention plot for clarity. Images have a size of 500 × 500.

ALBERT or BERT for text encoders) and finetuning them doesn't improve the performance significantly compared to the proposed SceneIT model with images; this shows that the images are helpful over and above the textual observation. We have given additional results in the appendix.

Related Work

The field of text-based and interactive games has seen a lot of recent interest and work, thanks in large part to the creation and availability of pioneering environments such as *TextWorld* (Côté et al. 2018) and the *Jericho* (Hausknecht et al. 2019) collection. Based on these domains, several interesting approaches have been proposed that seek to improve the efficiency of agents in these environments (Ammanabrolu and Riedl 2019; Dambekodi et al. 2020; Chaudhury et al. 2020; Murugesan et al. 2021). We mention and discuss this prior work in context in the earlier parts of this paper.

Separate from this progress on TBGs, there has also been work on Inductive Logic Programming (ILP) methods – these methods have shown good relation generalization in symbolic domains using differentiable model learning on symbolic inputs (Evans and Grefenstette 2018; Richardson and Domingos 2006), even in noisy settings. Neural Logic Machines (Dong et al. 2019) have shown good generalization to out-of-sample games using dedicated MLP units for first-order rule learning by interacting with the environment. The work on Logical Neural Networks (Riegel et al. 2020) is a recent addition to the family of ILP methods that can learn differentiable logical connectives using constrained optimization over the differentiable neural network framework. Concurrently, there has been work in the (symbolic) automated planning community that has looked at learning and inferring the relations (predicates) that make up an underlying domain – like the eight-tile puzzle – by using variational auto-encoders (Asai 2019; Asai and Fukunaga 2018; Asai and Muise 2020; Asai and Tang 2020).

Conclusion

In this paper, we introduced `Scene Images for Text-based Games` (`SceneIT`), a model for RL agents executing in text-based games. `SceneIT` uses the text from observations provided by the game to either retrieve or generate images that correspond to the scene represented by the text; and then combines the features from the images along with features from the text in order to select the next best action for the RL agent. We show via an extensive experimental evaluation that `SceneIT` shows better performance – in terms of the normalized reward score achieved by agents, as well as the number of steps to complete a task – than existing state-of-the-art models that rely only on the observation text. We also presented qualitative results that showed that an agent guided by `SceneIT` focuses its attention on those parts of an image that we may expect a human to attend to as well.

References

Ammanabrolu, P.; and Hausknecht, M. 2019. Graph Constrained Reinforcement Learning for Natural Language Action Spaces. In *International Conference on Learning Representations*.

Ammanabrolu, P.; and Hausknecht, M. 2020. Graph constrained reinforcement learning for natural language action spaces. *arXiv preprint arXiv:2001.08837*.

Ammanabrolu, P.; and Riedl, M. 2019. Playing Text-Adventure Games with Graph-Based Deep Reinforcement Learning. 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)*, 3557–3565.

Asai, M. 2019. Unsupervised grounding of plannable first-order logic representation from images. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 29, 583–591.

Asai, M.; and Fukunaga, A. 2018. Classical planning in deep latent space: Bridging the subsymbolic-symbolic boundary. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Asai, M.; and Muise, C. 2020. Learning Neural-Symbolic Descriptive Planning Models via Cube-Space Priors: The Voyage Home (to STRIPS). In *International Joint Conference on AI (IJCAI)*.

Asai, M.; and Tang, Z. 2020. Discrete Word Embedding for Logical Natural Language Understanding. *arXiv preprint arXiv:2008.11649*.

Chaudhury, S.; Kimura, D.; Talamadupula, K.; Tatsubori, M.; Munawar, A.; and Tachibana, R. 2020. Bootstrapped Q-learning with Context Relevant Observation Pruning to Generalize in Text-based Games. In *The 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Chen, X.; and Lawrence Zitnick, C. 2015. Mind's eye: A recurrent visual representation for image caption generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2422–2431.

Côté, M.-A.; Kádár, Á.; Yuan, X.; Kybartas, B.; Barnes, T.; Fine, E.; Moore, J.; Hausknecht, M.; El Asri, L.; Adada, M.; et al. 2018. Textworld: A learning environment for text-based games. In *Workshop on Computer Games*, 41–75. Springer.

Dambekodi, S.; Frazier, S.; Ammanabrolu, P.; and Riedl, M. O. 2020. Playing Text-Based Games with Common Sense. arXiv:2012.02757.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dong, H.; Mao, J.; Lin, T.; Wang, C.; Li, L.; and Zhou, D. 2019. Neural logic machines. *arXiv preprint arXiv:1904.11694*.

Evans, R.; and Grefenstette, E. 2018. Learning explanatory rules from noisy data. *Journal of Artificial Intelligence Research*, 61: 1–64.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative Adversarial Nets. In Ghahramani, Z.; Welling, M.; Cortes, C.; Lawrence, N.; and Weinberger, K. Q., eds., *Advances in Neural Information Processing Systems*, volume 27, 2672–2680. Curran Associates, Inc.

Gupta, S.; Dileep, A.; and Thenkanidiyoor, V. 2021. Recognition of varying size scene images using semantic analysis of deep activation maps. *Machine Vision and Applications*, 32(2): 1–19.

Hausknecht, M.; Ammanabrolu, P.; Marc-Alexandre, C.; and Xingdi, Y. 2019. Interactive Fiction Games: A Colossal Adventure. *CoRR*, abs/1909.05398.

He, J.; Chen, J.; He, X.; Gao, J.; Li, L.; Deng, L.; and Ostendorf, M. 2016. Deep Reinforcement Learning with a Natural Language Action Space. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1621–1630.

Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; and Soricut, R. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

Lu, Y.; Zhang, W.; Jin, C.; and Xue, X. 2012. Learning attention map from images. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 1067–1074. IEEE.

Mnih, V.; Badia, A. P.; Mirza, M.; Graves, A.; Lillicrap, T.; Harley, T.; Silver, D.; and Kavukcuoglu, K. 2016. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, 1928–1937. PMLR.

Murugesan, K.; Atzeni, M.; Kapanipathi, P.; Shukla, P.; Kumaravel, S.; Tesauro, G.; Talamadupula, K.; Sachan, M.; and Campbell, M. 2021. Text-based RL Agents with Commonsense Knowledge: New Challenges, Environments and Baselines. In *The 35th AAAI Conference on Artificial Intelligence*.

Narasimhan, K.; Kulkarni, T.; and Barzilay, R. 2015. Language Understanding for Text-based Games using Deep Reinforcement Learning. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1–11.

Richardson, M.; and Domingos, P. 2006. Markov logic networks. *Machine learning*, 62(1-2): 107–136.

Riegel, R.; Gray, A.; Luus, F.; Khan, N.; Makondo, N.; Akhalwaya, I. Y.; Qian, H.; Fagin, R.; Barahona, F.; Sharma, U.; et al. 2020. Logical neural networks. *arXiv preprint arXiv:2006.13155*.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 618–626.

Silver, D.; Hubert, T.; Schrittwieser, J.; Antonoglou, I.; Lai, M.; Guez, A.; Lanctot, M.; Sifre, L.; Kumaran, D.; Graepel, T.; et al. 2018. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419): 1140–1144.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. 2017. Mastering the game of go without human knowledge. *nature*, 550(7676): 354–359.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2818–2826.

Xu, T.; Zhang, P.; Huang, Q.; Zhang, H.; Gan, Z.; Huang, X.; and He, X. 2018a. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1316–1324.

Xu, T.; Zhang, P.; Huang, Q.; Zhang, H.; Gan, Z.; Huang, X.; and He, X. 2018b. AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks. *CVPR*.

Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2921–2929.