MineObserver: A Deep Learning Framework for Assessing Natural Language Descriptions of Minecraft Imagery

Jay M. Mahajan, Samuel Hum, Jeff Ginger, H. Chad Lane
Department of Educational Psychology
Department of Computer Science
University of Illinois, Urbana-Champaign
1310 S. Sixth St.
Champaign, IL 61820 USA

Abstract

This paper introduces a novel approach for learning natural language descriptions of scenery in Minecraft. We apply techniques from Computer Vision and Natural Language Processing to create an AI framework called MineObserver for assessing the accuracy of learner-generated descriptions of science-related images. The ultimate purpose of the system is to automatically assess the accuracy of learner observations, written in natural language, made during science learning activities that take place in Minecraft. Eventually, MineObserver will be used as part of a pedagogical agent framework for providing in-game support for learning. Preliminary results are mixed, but promising with approximately 62% of images in our test set being properly classified by our image captioning approach. Broadly, our work suggests that computer vision techniques work as expected in Minecraft and can serve as a basis for assessing learner observations.

Introduction

Making scientific observations is one of the most important and difficult skills for children as they learn about science (Arias and Davis 2016). It is often difficult for learners since it is not always obvious what features or aspects of some phenomena are most relevant or what observations will ultimately be important during scientific inquiry. It is also challenging for learners to provide accurate descriptions that have an appropriate level of detail. Scaffolding is needed to help learners make the shift from producing more casual ("everyday") observations to those that are more scientific (Eberbach and Crowley 2009).

To provide such support automatically, data-driven approaches that leverage machine learning (ML) techniques for assessment have become common (Zhai et al. 2020). Assessing natural language input, in particular, presents unique challenges but also high potential to gain insights into how learners understand and develop important scientific skills, such as in formulating scientific explanations (Ariely, Nazaretsky, and Alexandron 2022).

In this paper, we introduce a framework for assessing learner observations in Minecraft worlds designed for science learning. Our framework, *MineObserver*, uses Computer Vision techniques and Natural Language Processing to make judgments on the quality of student descriptions of what they see. Here, we report on the design of our framework and provide results of preliminary tests of accuracy. Eventually, we will package MineObserver together in a pedagogical agent framework that will provide interactive dialogues (i.e., intelligent tutoring support) for helping learners grow in their ability to make better observations with practice.

Learning Context: Minecraft

Minecraft is an exceptionally popular game. Since its release in 2009, the user base has exploded with over 140M players and 241M logins per month and consistently ranks in the top 5 most popular games for children.[1] It spans many platforms and its players have a range of interests, ages and experience. It is referred to as a "sandbox" game because it can be used in several different modes and contexts and often participants come up with their own challenges and meanings when playing alone or with others. The Java Edition of the game has an enormous community following and is very modifiable, which makes it an ideal candidate to create more complex teaching and learning simulations like the one exhibited in this paper.

Minecraft for Science Learning

The research shared in this paper is part of the NSF-funded project WHIMC (What-If Hypothetical Implementations in Minecraft). WHIMC investigates the use of Minecraft as an educational tool for science learning, with an empahsis on Astronomy content that engages children and promotes interest in STEM. WHIMC is implemented as a Minecraft (Java Edition) server consisting of a space station hub and a collection of worlds to visit. On these worlds, learners interactively explore the scientific consequences of alternative versions of Earth via "what if?" questions (e.g., "What if the earth had no moon?") as well as feasible representations of several known exoplanets. It is hoped that such experiences will act as *triggers* of interest (Yi, Gadbury, and Lane 2021),

[1] https://news.xbox.com/en-us/wp-content/uploads/sites/2/2021/04/Minecraft-Franchise-Fact-Sheet_April-2021.pdf

which are required in order for interest to be sustained over time (Renninger and Hidi 2015).

A key component of the project is to analyze how learners interact with the system and assess their engagement with and understanding of the science content. Participants are invited to participate in quest challenges where they can learn about and measure pertinent science characteristics of simulated worlds (such as temperature and radiation). In addition to measurements, learners also make observations about things they think are noteworthy or important in some way. For example, without a moon and its gravitational pull, the Earth's rotation would be almost three times as fast as it is now. This would cause fierce winds on the surface of Earth. To withstand such winds, trees would need to be shorter, wider, and stronger (Comins 1993). An example of an observation of this type is shown in Figure 1. We note that the combination of a screenshot and a description forms the basis for our data set described in the next section.

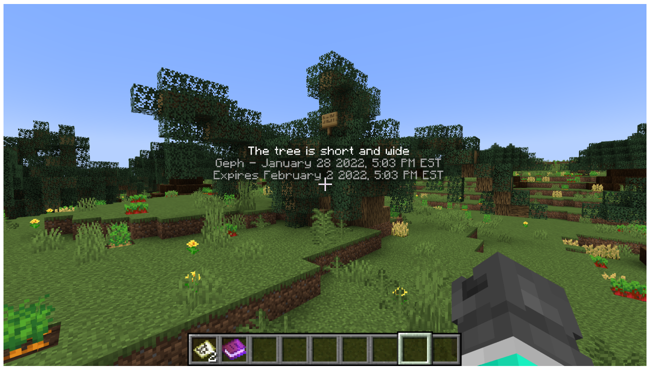

The tree is short and wide
Geph - January 28 2022, 5:03 PM EST
Expires February 2 2022, 5:03 PM EST

Figure 1: A sample observation of tree variation on a version of Earth with no moon.

WHIMC provides a framework for making observations like a scientist. In particular, based on our prior work to assess learner observations (Yi, Gadbury, and Lane 2020), we identified five key categories for observations:

1. *Factual*: observations are comprised of nouns without any elaboration.
2. *Descriptive*: observations related to color, temperature, quantity, and other physical attributes such as weight or size.
3. *Comparative*: observations comparing one natural phenomena to another.
4. *Analogies*: observations comparing natural phenomena with another similar structure or object (an advanced form of comparative).
5. *Inferences*: observations where a hypothesis or explanation is proposed (the most advanced form, rare for middle schoolers to do spontaneously).

Observations are also visible to all players on the server, so they might prompt other learners to take notice. In addition the WHIMC back end captures additional data, including coordinate and directional data to better understand what students were observing at the time.

AI support for learning in WHIMC

The methods explained in this paper will eventually provide assessments to a pedagogical agent (presented to players as a Minecraft NPC) that will support independent learners on our server. The agent would also enhance classroom-based instruction by directing attention and providing support when instructors are not available. Specifically, the agent will help learners stay on task and consider the underlying science learning goals. It would also provide evaluation data for teachers. In a sense it answers the questions of ”what are they looking at?” and ”are they paying attention to important STEM-related components?” We combine this method with Bayesian Knowledge Tracing learner modeling to provide feedback on the composition of observations (Hum et al. 2022), discover new aspects of their environments, and ask ”what if” questions. Their experience on the server is also guided by an AI-driven path-finding algorithm to help them transit between points of interest and stages of quest challenges. Combined, these approaches are a compelling foray into comprehensive learner support.

MineObserver AI Framework

Our framework can be split into major three stages (depicted in Figure 2). First, we employ an image captioning model that takes the student's current view (as an image). This is accomplished by recording the relevant coordinate and gaze data when an observation is made so that the system can process the exact view of the student. This image is sent to an image caption model that generates a caption of the possible accurate observation (serving as the “expert” representation). Second, we use RoBERTa (Robustly Optimized Bidirectional Encoder Representations from Transformers approach) (Liu et al. 2019) to contextually compare the image captioning model's results with the learner's observation using cosine similarity. And finally, we return feedback based on the cosine similarity.

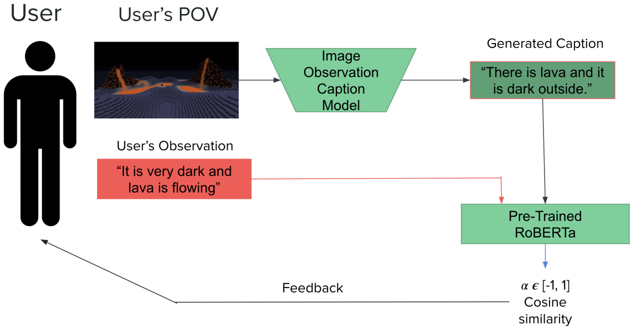

User
User's POV
Image Observation Caption Model
Generated Caption
“There is lava and it is dark outside.”
User's Observation
“It is very dark and lava is flowing”
Pre-Trained RoBERTa
Feedback
α ϵ [-1, 1] Cosine similarity

Figure 2: MineObserver AI Framework

Image Captioning

While many methods exist for image captioning, we follow a similar method as Google's *show and tell* caption generator (Vinyals et al. 2014). We use a convolutional neural network (CNN) to extract the features of our images and a recursive neural network, specifically, a long short-term memory unit (LSTM) (Hochreiter and Schmidhuber 1997), to provide possible captions for an image. While the approach mentioned above used a pre-trained CNN (trained on ImageNet dataset), this did not work well on our dataset thus we trained both the CNN and LSTM via backpropagation.

Our CNN architecture is a densely connected convolutional network (DenseNet) (Huang et al. 2018) with its final layer replaced with a linear layer. DenseNet has proven to outperform other convolutional neural networks (e.g AlexNet (Krizhevsky, Sutskever, and Hinton 2012) and ResNet (He et al. 2015)) on datasets such as CIFAR, Street View House Numbers (SVHM), and ImageNet. Thus, it is sufficiently suited to extract features that are needed to caption our images. The output of the DenseNet is then fed into the LSTM to generate words. The combination of extracting features from our CNN with our LSTM, creates a caption of the player's view.

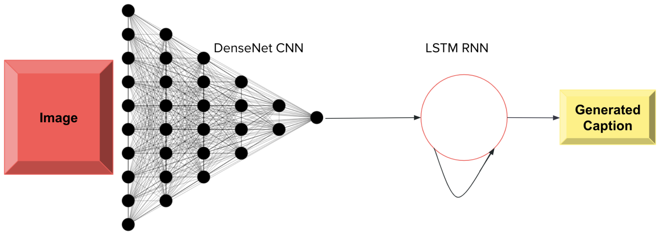


Figure 3: Image Caption Model Architecture

RoBERTa

As stated previously, students engaging with the WHIMC platform make observations while exploring different Minecraft maps. These observations are important: they simultaneously reflect how deeply engaged the learner is in the experience and reveal (to an extent) the level of understanding they have for the science concepts. Our agents must assess the content of these observations to guide their pedagogical actions.

To do this, We utilize the RoBERTa model fine-tuned on the Semantic Textual Similarity benchmark (STSb) and Natural Language Inference (NLI) dataset to encode the image caption generated by the CNN and the student's observation. Facebook's RoBERTa model has been shown to outperform BERT on the General Language Understanding Evaluation (GLUE) benchmark, Stanford Question Answer Dataset (SQuAD), and ReAding Comprehension from Examinations (RACE) dataset (Liu et al. 2019). In addition, fine-tuning the model with STSb and NLI has been shown to improve sentence encodings for common text similarity tasks (Reimers and Gurevych 2019). Thus, the model is well suited to compare the image captions and the student observations. The encodings are then compared using cosine similarity, which is used to generate appropriate responses from our agent to support the learner.

Feedback

The goal of MineObserver is to provide students with real-time feedback regarding the accuracy of their observation and relevancy to science goals on their current planet. Our current approach to feedback is dependent on the cosine similarity, α, from the RoBERTa model. Using the proximity of researcher identified points of interest on our maps and a threshold of $\lambda = 0.50$ for correctness, we provide researcher-designed context specific feedback to users based on their correctness. Although this form of feedback is not as sophisticated as other parts of the project, it provides some idea of context and correctness for the learner. We hope to expand this feature in the future to provide tailored feedback based on the exact details of their surroundings (See Future Work and Conclusion).

Training and Results

Dataset

Our dataset consists of 161 Minecraft image screenshots from the WHIMC project with different labels including the image name, the map or location the image is taken from in our Minecraft world, the type of observation (e.g Descriptive, Comparative, or Inferential) and finally the observation. Some images are repeated with different (correct) captions which allows the Image Caption model to consider different answers. Table 1 shows some examples from our dataset (images are in the Appendix).

Image	Type	Observation
Figure 4	Descriptive	The terrain is rocky, and there is a big planet in the background
Figure 5	Inferential	The high wind speeds will help generate electricity from the wind turbine
Figure 6	Comparative	It looks like Earth but more snowy and icy
Figure 5	Descriptive	I see a windmill on top of a hill

Table 1: Examples from the dataset used for training. Two images are repeated but have different captions. Observations can be a different type. See Appendix for images.

Training

The only part of our MineObserver AI framework that needs to be trained is our Image Caption model. Before any training, we center-crop and resize our images to be of size 256 x 256. This allowed our Image Caption model to focus on the center of the image which is where most of our intended observations are located.

We trained the entire model via backpropagation with an Adam optimizer (Kingma and Ba 2014), a learning rate of 0.0003, and used a cross entropy loss function for 150 iterations. Given the small dataset and GPU access, the training time was under 2 hours.

Results

Our results were mixed but promising. We split our dataset to leave 24 testing examples to run our analysis. Table 2 shows some of the best test examples using our framework.

Image	Generated Caption	True Observation	Cosine Similarity
Figure 7	There is a hot spring	Water coming from the ground a spring	0.6774
Figure 8	The trees and flowers are full and similar to rainforests	I am surrounded by flowers and trees	0.6863
Figure 9	There are many different plants here	Growing plants	0.6133

Table 2: These generated captions worked well and are close to the true observation. Moreover, the cosine similarity for all the examples are high, meaning that the captions match. See Appendix for images.

While Table 2 shows some good results on caption generation, there were some examples where the generated caption did not match the true observation (see Table 3).

Image	Generated Caption	True Observation	Cosine Similarity
Figure 10	The moon looks like Neptune	The two volcanoes are covered in lava	0.1294
Figure 11	The snow is everywhere and I see trees	If there is lava readily available on the surface of the planet, then the planet is most likely geologically active	-0.0395
Figure 12	The trees and flowers are full and similar to rainforest	There is no moon	0.0292

Table 3: These generated captions did not match the true observation. The cosine similarity is low for all of the examples, but ideally generated captions should be have a high cosine similarity. Example images can be found at the Appendix

All of the examples in Table 3 are not similar and the cosine similarity is what we expect. However, the objective of the image caption generator is to maximize the cosine similarity meaning the generated caption should be very close to the true observation. We also see a type of mode collapse on this generator seen by the 2nd example in Table 2 and the 3rd example in Table 3, however we believe this is due to a lack of data and not the model itself (See Future Work and Conclusion).

We summarize our results in Table 4 by separating our test examples into three different categories (unsatisfactory, fair and excellent) depending on the cosine similarity of the generated caption and the true observation. We also compare our image caption generator to Google's Show and Tell model trained on ImageNet and Flickr 8k dataset as a baseline. Based on manual inspections of the results our limited training set, we consider *Fair* and *Excellent* results to be generally accurate enough to act as a basis for assessing learner-generated observations. Therefore, over 62% of our current test-set images meet the threshold. Moreover, our generator clearly performs better than the baseline on both *Fair* and *Excellent* categories. While obviously the accuracy is not yet high enough for live use on our server, it does serve as initial evidence of the promise of our approach.

Model	< 0.25 (Unsatisfactory)	$0.25 - 0.49$ (Fair)	$0.5 >$ (Excellent)
Ours	37.5%	41.67%	20.8%
Show and Tell	100%	0%	0%

Table 4: Percent of test examples that were in the ranges measured by their cosine similarity from our generator and a baseline model.

Future Work and Conclusion

There are several ideas we wish to add or improve to our framework. This includes increasing and diversifying our dataset, re-structuring our image caption model, feedback generation, and real time Minecraft integration.

Dataset

As stated in our Training and Result section, we used 161 screenshot images from the WHIMC project with each of them having at least 1 caption. While suitable for our preliminary work to show feasibility, this is insufficient to train our image caption generator for real-time use on our server. The small training set caused the image caption generator suffer from over-fitting and mode collapse. We aim to expand our dataset to include over 1000 images and captions to train a more robust model.

Another potential limitation is the inherent potential bias in our dataset. Specifically, we plan to revisit the vocabulary used in our observations to ensure they align with the target age group (11-14 year olds). Our dataset was predominantly created by researchers on the team with science and Minecraft experience. Given the intended use of this framework, it will be critical to gather data from learners to capture different terms to describe ideas. Thus, we might have to add observations from that age group for the framework to work more effectively in the future.

Finally, the dataset used in this research is at risk of becoming obsolete as the project evolves. WHIMC is a rapidly changing and growing project, continually expanding by adding different worlds and locations players can go to. It is likely content will change for many of the science-related observations and the image caption model becomes ineffective. To counteract this, we hope to move our framework to be more dynamic and to continually learn as WHIMC grows in the future. This would most likely result in a supervised learning system so that learner generated observations could be checked by teachers or researchers before being integrated into the system.

Re-Structuring the Image Caption Model

Our dataset is comprised of descriptive, inferential, and comparative descriptions (the factual and analogy types are not represented). Currently our image caption model is fed the three different types and generates any type of caption. In the future, we would like to be able to generate a set of observations given an image and select the observation type we would like to compare against.

One possible way is to feed a one-hot vector of size 3 x 1 for each observation type into our CNN. This would allow the model to separate the data for each observation type and generate the correct type of observation but it would require more data to train an effective model.

Feedback Generation

Observation content is, however, only one facet of the observation writing domain. Thus as we stated previously, we are working on a machine learning approach tracking observation writing skills using Bayesian Knowledge Tracing and observation structure (Hum et al. 2022). We provide students feedback on their mastery of observations using an open-learner model in the form of progress bars.

We also want to provide better textual feedback to supplement the open-learner model. One possible direction could be using the keywords generated from our generated caption. Thus, future work may include measuring the impact of the open-learner model and textual feedback on student observation and engagement behaviors.

Real-Time Minecraft Integration

Currently, the model has not been deployed into Minecraft due to technical limitations and unacceptable response times. Of course, the longer a system takes to assess a learner observation the more frustrated that learner will likely become. We are investigating methods to capture and cache in game screenshots use them for dynamic assessments of what learners are observing. In other words, we are considering developing a server plugin or using a ”ghost” client that tracks the user in the Minecraft world to circumvent the need for the client to capture an image.

Acknowledgements

The authors of the paper would like to acknowledge Ateneo University and the developers of WHIMC for helping us create a working dataset for this paper. This material is based upon work supported by the National Science Foundation under Grants 1713609 and 1906873.

Appendix

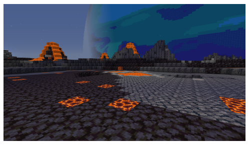


Figure 4: An image with a descriptive caption in our dataset.
Caption: The terrain is rocky, and there is a big planet in the background

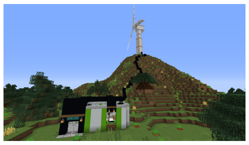


Figure 5: An image with two different captions.
Inferential Caption: The high wind speeds will help generate electricity from the wind turbine
Descriptive Caption: I see a windmill on top of a hill

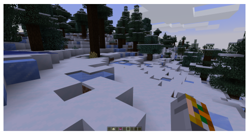


Figure 6: An image with a comparative caption in our dataset.
Caption: It looks like Earth but more snowy and icy

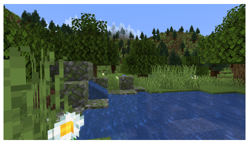
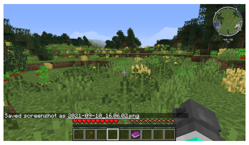


Figure 7: An excellent example in our test dataset.
Human Caption: There is a hot spring
Generated Caption: Water coming from the ground a spring
Cosine Similarity: 0.6774



Saved screenshot as 2021-09-10_16.06.02.png

Figure 8: An excellent example in our test dataset.
Human Caption: I am surrounded by flowers and trees
Generated Caption: The trees and flowers are full and similar to rainforests
Cosine Similarity: 0.6863

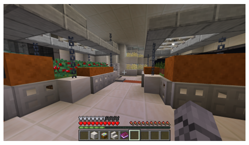


Figure 9: An excellent example in our test dataset.
Human Caption: There are many different plants here
Generated Caption: Growing plants
Cosine Similarity: 0.6133

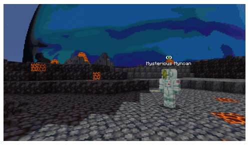


Figure 10: An unsatisfactory example in our test dataset.
Human Caption: The two volcanoes are covered in lava
Generated Caption: The moon looks like Neptune
Cosine Similarity: 0.1294

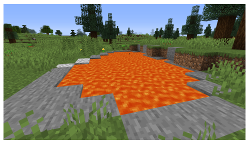


Figure 11: An unsatisfactory example in our test dataset.
Human Caption: If there is lava readily available on the surface of the planet, then the planet is most likely geologically active
Generated Caption: The snow is everywhere and I see trees
Cosine Similarity: -0.0395

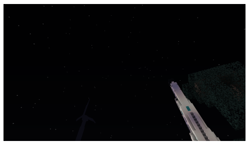


Figure 12: An unsatisfactory example in our test dataset.
Human Caption: There is no moon
Generated Caption: The trees and flowers are full and similar to rainforest
Cosine Similarity: 0.0292

References

Arias, A. M., and Davis, E. A. 2016. Making and recording observations. *Science and Children* 53(8):54–60.

Ariely, M.; Nazaretsky, T.; and Alexandron, G. 2022. Machine learning and hebrew nlp for automated assessment of open-ended questions in biology. *International Journal of Artificial Intelligence in Education* 1–34.

Comins, N. F. 1993. What if the moon didn't exist: voyages to earths that might have been. *New York*.

Eberbach, C., and Crowley, K. 2009. From everyday to scientific observation: How children learn to observe the biologist's world. *Review of Educational Research* 79(1):39–68.

He, K.; Zhang, X.; Ren, S.; and Sunr, J. 2015. Deep residual learning for image recognition.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory.

Huang, G.; Liu, Z.; van der Maaten, L.; and Weinberger, K. Q. 2018. Densely connected convolutional networks.

Hum, S.; Stinar, F.; Lee, H.; Ginger, J.; and Lane, H. C. 2022. Classification of Natural Language Descriptions for Bayesian Knowledge Tracing in Minecraft. Manuscript submitted for publication at AIED23.

Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach.

Reimers, N., and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks.

Renninger, K. A., and Hidi, S. E. 2015. *The power of interest for motivation and engagement*. Routledge.

Vinyals, O.; Toshev, A.; Bengi, S.; and Erhan, D. 2014. Show and tell: A neural image caption generator.

Yi, S.; Gadbury, M.; and Lane, H. C. 2020. Coding and analyzing scientific observations from middle school students in Minecraft.

Yi, S.; Gadbury, M.; and Lane, H. C. 2021. Identifying and Coding STEM Interest Triggers in a Summer Camp. In *Proceedings of the 15th International Conference of the Learning Sciences-ICLS 2021*. International Society of the Learning Sciences.

Zhai, X.; Yin, Y.; Pellegrino, J. W.; Haudek, K. C.; and Shi, L. 2020. Applying machine learning in science assessment: a systematic review. *Studies in Science Education* 56(1):111–151.