The Workshops of the Thirty-Second
AAAI Conference on Artificial Intelligence

Threat, Explore, Barter, Puzzle: A Semantically-Informed Algorithm for Extracting Interaction Modes

Nancy Fulda, Daniel Ricks, Ben Murdoch, David Wingate
Brigham Young University
nfulda@byu.edu, danielricks@byu.edu, murdoch@byu.edu, wingated@cs.byu.edu

Abstract

In the world of online gaming, not all actions are created equal. For example, when a player's character is confronted with a closed door, it would not make much sense to brandish a weapon, apply a healing potion, or attempt to barter. A more reasonable response would be to either open or unlock the door. The term *interaction mode* embodies the idea that many potential actions are neither useful nor applicable in a given situation. This paper presents a AEGIM, an algorithm for the automated extraction of game interaction modes via a semantic embedding space. AEGIM uses an image captioning system in conjunction with a semantic vector space model to create a gestalt representation of in-game screenshots, thus enabling it to detect the interaction mode evoked by the game.

1 Introduction and Related Work

Video and computer games are a valuable resource for AI researchers. They serve as test domains for novel algorithms (Mnih et al. 2015) (Kaplan, Sauer, and Sosa 2017), provide sensory-rich virtual learning environments (Vinyals et al. 2017) and encourage innovation as researchers strive to improve in-game characters (Laird and van Lent 2001). In addition, the emerging field of automated game design learning (Osborn, Summerville, and Mateas 2017) uses the structure of the games themselves to extract properties that can be used to (a) improve human play, (b) facilitate virtual character development, (c) create novel tools for developers, and (d) verify that human-specified design properties hold on the model.

In the spirit of this emerging field, we present AEGIM, an algorithm that uses language-based common sense reasoning to distinguish between interaction modes. Game output in the form of pixels or text (or both) is encoded within a 4800-dimensional semantic embedding space trained based on local context[1]. A set of linear classifiers is then used to determine which of several possible interaction modes is evoked by the current situation.

'Interaction modes', in the context of this paper, refer to the set of player actions that would be reasonable to execute in the current situation. For example, when confronted with a locked door, it would be reasonable for the player to attempt to unlock it with a key or lockpick. It would be less reasonable to brandish a sword, apply a healing potion, or attempt to barter with the closed door. Similarly, when confronted with an aggressive enemy, one would expect the player to either attack or flee. Under those circumstances, it would not make sense to engage in casual conversation or examine an ornate rug on the floor.

In other words, not all actions are equally preferable in every situation.

Given that the relative value of an action is context-dependent, we wish to identify a set of contexts (i.e. 'interaction modes') in which certain actions best apply. In theory, a system capable of identifying such modes could map a game based on behavioral contexts rather than on world geography, a potentially useful diagnostic tool.

Our research utilizes recent work in computer vision (Tran et al. 2016) (Clarifai 2017) to convert pixels into text descriptions of a scene, but goes beyond simple object recognition (Krizhevsky, Sutskever, and Hinton 2012) or semantic segmentation (Long, Shelhamer, and Darrell 2015) in order to acquire a common-sense representation of the observed items. To do this, we use the skip-thought embedding space (Kiros et al. 2015). Related semantic embedding spaces include word vectors (Mikolov et al. 2013) (Pennington, Socher, and Manning 2014), sentence embeddings using a simple or weighted average of word vectors (Faruqui et al. 2014) (De Boom et al. 2016), fixed-length and LSTM-based sentence embeddings (Saha et al. 2016) (dos Santos and Gatti 2014), and document-level embeddings (Le and Mikolov 2014).

This work touches tangentially on the field of affordance detection[2] (Zhu, Fathi, and Fei-Fei 2014) (Song et al. 2015)

Copyright © 2018, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

[1] In other words, the vector representation of each sentence is influenced by the sentences that tend to appear before or after it in the input corpus. Word-level semantic spaces trained in this way have been shown to encode common-sense knowledge about the physical and sociological properties of our world (Mikolov, Yih, and Zweig 2013) (Nematzadeh, Meylan, and Griffiths 2017).

[2] *Affordances* (Gibson 1977) refer to the set of actions that are made possible by an object or situation. For example, a ladder affords the possibility of climbing, while level terrain affords the possibility of running. Gibson asserts that an affordance is neither a property of the environment nor of the actor, but of the complementarity of the two.

(Navarro et al. 2012), particularly as explored by (Fulda et al. 2017) in their work with text-based adventure games. Fulda et al. used analogical operations performed in vector space to detect the behaviors afforded by a specific game item. Our work takes this idea one step farther by detecting the current interaction mode of the game (and by extension the subset of actions afforded by the situation) rather than the affordances of a single object.

2 Methodology

Our algorithm for the Automated Extraction of Game Interaction Modes (AEGIM) begins with screenshots extracted during game play, which are then converted to a plain text representation using an online caption generation tool. We do not require the caption generator to provide a coherent sentence; a simple string of objects or adjectives is sufficient for our purposes.

Once we have a plain text description of an image, we encode the description as a geometric point in a semantic embedding space. For this step, we rely on the skip-thought embedding model (Kiros et al. 2015), which is trained by requiring a neural network to predict both the sentence that directly precedes and the sentence that directly follows each sentence in the training corpus. This is, of course, a mostly futile task: The number of possible antecedents and successors for any given sentence is enormous. However, during the training process the network creates an internal representation that roughly corresponds to each sentence's semantic meaning. It is this internal representation - the semantic meaning of a sentence - that interests us. We use it to encode each input string as a 4800-dimensional vector that represents the location of the text in skip-thought space.

Conversion into a semantic embedding space is essential in order to create a gestalt representation of the items in a scene: A hammer alone does not indicate an imminent physical threat, nor does an angry man. But an angry man holding a hammer is cause for immediate concern.

Next, the AEGIM algorithm determines which of several possible interaction modes are indicated by the original screenshot. This is done by comparing the skip-thought vector that represents the image description with the vector embeddings of a set of hand-coded example texts that exemplify each mode. To determine whether an image evokes the interaction mode of 'Threat', for example, we would enter sentences like those shown in Fig. 1.

We note with interest that these example sentences do not necessarily have to be hand-coded. It is not difficult to imagine scenarios where example text is extracted algorithmically by observing which subset of actions produces reasonable results under specific game conditions.

Once the example texts have been provided, each text is encoded as a skip-thought embedding and an average representative vector is calculated for each example set. New sentences are then classified based on their proximity to these cluster centers. More formally, let $M = m_1, ..., m_k$ be the set of interaction modes to be detected and let $S_i = \{s_i^1, ...s_i^n\}$ be the example strings associated with the ith interaction mode (see Fig. 1). Then $V_i = \{v_i^1, ...v_i^n\}$ is the set of 4800-dimensional vector encodings of S in the skip-thought embedding space, where k is the number of interaction modes to be classified and n is the number of examples used to define each interaction mode. For each incoming text t to be classified, AEGIM determines which interaction modes apply according to the following algorithm:

Interaction Mode: Threat
'You see a soldier holding a sword'
'You are badly wounded'
'A massive troll bars the path'
'The bull paws the ground, then charges toward you'
'The poisonous spider advances, ready with its deadly bite'
'You are in danger'
'If you fall from this height, you will die'
'The battle rages around you'
'The angry man begins to attack you'
'You are plummeting to your death, with only a few seconds before you hit the ground'

Figure 1: Example texts used to define the 'Threat' mode, meaning that an immediate physical danger is present.

1: $v_t =$ the vector encoding of text t
2: **for** i in $1...k$ **do**
3: $\quad d_i = v_t - 1/n \sum_0^n v_i^n$
4: $\quad d_q = v_t - 1/(k-1) \sum_q 1/n \sum_0^n v_q^n, q \neq i$
5: $\quad$ **if** $|d_i| < |d_q|$ **then**
6: $\quad\quad$ return *True*
7: $\quad$ **else**
8: $\quad\quad$ return *False*
9: $\quad$ **end if**
10: **end for**

In other words, AEGIM returns *True* for the ith interaction mode if and only if v_t is closer to the ith cluster center than it is to the average of the cluster centers of all other interaction modes. Note that it is possible for AEGIM to output more than one interaction mode for a given image, or no interaction modes at all.

Because skip-thought representations encode the *meaning* of a sentence rather than the *text* or *syntax* of the sentence, this simple linear classifier is sufficient to detect the correct interaction mode in many instances. Accordingly, one could imagine that a more sophisticated classifier might improve performance beyond that reported in this paper.

3 Results

As a proof of concept for this idea, we collected images from the popular Bethesda game Skyrim (Bethesda Softworks LLC 2013). The dataset consists of 65 images collected during game play, including peaceful encounters with shopkeepers, wandering animals, and local architecture as well as high-risk encounters with monsters and magic-wielders[3]. Images were then passed through one of three text-generators:

[3]No player-controlled characters were harmed during the collection of this dataset.

	AEGIM	keyword baseline	constant
Clarifai	72.3%	8.6%	74.2%
CaptionBot	73.3%	13.3%	74.2%

	AEGIM	keyword baseline	constant
Clarifai	21.9%	14%	43.8%
CaptionBot	31.3%	18.8%	43.8%

Figure 2: Left: Classification accuracy, counting each image/category pair individually. Right: Exact matches, meaning the percentage of images where all four categories were classified correctly. Human captions were not available for the full dataset.

1. **Clarifai** (Clarifai 2017), an online object recognition system that returns a list of identified elements within a scene
2. **CaptionBot** (Tran et al. 2016), an online caption generator that creates a complete sentence describing the contents of an image
3. **Human-generated** captions

The human-generated captions were provided by an 11-year-old girl who has never played Skyrim, is not familiar with our algorithm, and was unaware of how the captions would be used. She was instructed only to 'give a one-sentence description of what's in the picture'. Her description was used exactly as given, without prompting, hints, or post-processing. The automatically-generated captions were post-processed as follows: Clarifai returns a list of 20 identified elements for each scene, along with their estimated likelihood. In our experiments, we concatenated the 10 elements with the highest likelihoods into a single text string which was then encoded as a skip-thought vector. CaptionBot prefaces each sentence with a qualifier indicating its certainty about the given description, and sometimes appends a description of whether people in the image appear happy or sad. This supplementary information was omitted; only the core image description was passed through the skip-thought encoder.

The purpose of including human-generated captions was to test the soundness of our algorithm when provided with high-quality text descriptions. This was motivated by the observation that online caption generation systems, which are optimized for photorealistic images of real world objects, perform poorly when presented with game images.

We focused on four interaction modes for this experiment: **Threat** (as depicted in Fig. 1); **Explore**, indicating an opportunity to traverse the landscape and discover items of interest; **Barter**, indicating an opportunity for exchange of goods; and **Puzzle**, indicating that a manual manipulation task is required or available. The values of k and n were set to 4 and 10 respectively.

To create reasonable baseline comparisons, we created two naive classifiers. The keyword method compiles a list of keywords extracted from the AEGIM example texts provded for each interaction mode. During naive classification, an interaction mode was marked as True if any of the keywords appeared within the text to be classified. The constant baseline simply returned the most common classification ('explore') in all cases. Results are shown in Figure 2. Remarkably, the AEGIM algorithm was able to match or exceed both baselines with only 10 example texts per category and no online training.

Figures 3-6 show screenshots from the game along with interaction modes identified by AEGIM. Correct classifications are highlighted using boldface text. Overall (and unsurprisingly), the human-generated captions produce far better results than automated captions. This suggests that it would be worthwhile to either train a caption-generation system on images that more directly align with the task, or to use in-game annotations as an alternative to caption generation.

Generated Text		AEGIM Output
Clarifai:	'no person', 'travel', 'landscape', 'outdoors', 'snow', 'sky', 'mountain', 'daylight', 'winter', 'water'	**Explore**
CaptionBot:	'A tower with a mountain in the background'	**Explore**
Human text:	'A windmill near rocky mountains'	**Explore**

Figure 3: Overall AEGIM correctly identifies exploration scenes regardless of captioning method, perhaps because Clarifai and CaptionBot both detect landscape elements like hills, clouds, and buildings.

4 Conclusions and Future Work

AEGIM is a novel and potentially powerful tool for identifying the interaction modes evoked by an image. Preliminary experiments show that it is able to produce correct classifications in a variety of situations.

AEGIM offers the following key advantages over methods that predict interaction modes directly from images: (1) AEGIM's semantic embedding space can be customized via the selection of input corpus: Skip-thought embeddings trained using game manuals, genre-relevant articles, or in-game dialogue may prove to be particularly effective. (2) By encoding images into the skip-thought embedding space, AEGIM is able to interpret the gestalt meaning of items in a scene rather than evaluating each element independently.

Generated Text		AEGIM Output
Clarifai:	‘people’, ‘adult’, ‘smoke’, ‘flame’, ‘vehicle’, ‘military’, ‘one’, ‘man’, ‘weapon’, ‘war’	Barter
CaptionBot:	‘A man jumping over a fire’	Explore
Human text:	‘An archer ready to fight against the enemy’	**Threat**, Explore

Figure 4: This is one of the few combat images that was even moderately well-captioned by the automated systems. (Scenes with similar elements were described as ‘A group of people jumping’ or ‘a young man practicing his tricks on his skateboard’)

(3) AEGIM offers the possibility of dynamically-generated example sets extracted from game-internal images. (4) With AEGIM, image descriptions may be augmented using character dialogue, in-game annotations or other raw text produced by the game engine.

AEGIM’s greatest current weakness lies in the poor quality of the automatically generated captions. More research is required to determine which (if any) of the currently available open-source vision systems is able to produce acceptable descriptions of fictional scenarios. For many games, this limitation may be circumvented by utilizing character dialogue and/or in-game annotations instead of a vision system.

Future work in this area should include the use of neural networks or K-nearest-neighbor classifiers in lieu of linear classification. AEGIM’s performance thus far indicates that the skip-thought space is well-structured for the task of distinguishing between interaction modes; however, the current system is easily foiled by modifying clauses. ‘A dragon flying’ is consistently classified as a threat, but ‘a dragon flying through a cloudy sky’ is not. We anticipate that the use of more sophisticated classifiers will rectify this problem.

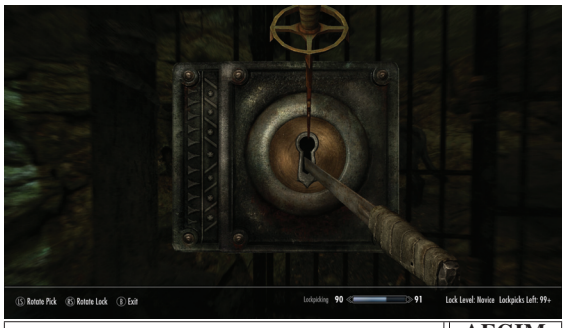
Lastly, attention should be given to the task of training high-quality image recognition systems for computer-generated images extracted during gameplay. Such a system would not only be useful for AEGIM, but also for many other potential applications.

5 Acknowledgements

We thank Alex, Aubrey, and Janika F. for data generation.We thank Nvidia, the Center for Unmanned Aircraft Systems, and Analog Devices, Inc. for their generous support.

Generated Text		AEGIM Output
Clarifai:	‘people’, ‘adult’, ‘religion’, ‘indoors’, ‘man’, ‘group’, ‘home’, ‘one’, ‘no person’, ‘travel’	**Barter**
CaptionBot:	‘A kitchen with wooden cabinets and a fireplace’	-
Human text:	‘A blacksmith that is sitting in his shop, but he also looks very buff’	-

Figure 5: In this case the human captioner’s commentary on the shopkeeper’s physique obscured the correct classification. AEGIM correctly returns the interaction mode ‘Barter’ when given the input text ‘A blacksmith sitting in his shop’.



Generated Text		AEGIM Output
Clarifai:	‘rusty’, ‘old’, ‘iron’, ‘security’, ‘dirty’, ‘safety’, ‘no person’, ‘steel’, ‘lock’, ‘door’	**Puzzle**
CaptionBot:	‘A clock that is looking at the camera’	-
Human text:	‘A door lock which is trying to be opened with a floating knife and a sharp thingie’	**Puzzle**

Figure 6: It is not difficult to see why CaptionBot mistook the image for a clock. Less obvious is why AEGIM does not consider the clock a puzzle item to be interacted with.

References

Clarifai. 2017. Clarifai image captioning demo. https://www.clarifai.com/demo.

De Boom, C.; Van Canneyt, S.; Demeester, T.; and Dhoedt,

B. 2016. Representation learning for very short texts using weighted word embedding aggregation. *Pattern Recognition Letters* 80:150–156.

dos Santos, C. N., and Gatti, M. 2014. Deep convolutional neural networks for sentiment analysis of short texts. In *COLING*, 69–78. ACL.

Faruqui, M.; Dodge, J.; Jauhar, S. K.; Dyer, C.; Hovy, E.; and Smith, N. A. 2014. Retrofitting word vectors to semantic lexicons. *arXiv preprint arXiv:1411.4166*.

Fulda, N.; Ricks, D.; Murdoch, B.; and Wingate, D. 2017. What can you do with a rock? affordance extraction via word embeddings. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, 1039–1045.

Gibson, J. J. 1977. The theory of affordances. In Shaw, R., and Bransford, J., eds., *Perceiving, Acting, and Knowing*.

Kaplan, R.; Sauer, C.; and Sosa, A. 2017. Beating atari with natural language guided reinforcement learning. *CoRR* abs/1704.05539.

Kiros, R.; Zhu, Y.; Salakhutdinov, R.; Zemel, R. S.; Torralba, A.; Urtasun, R.; and Fidler, S. 2015. Skip-thought vectors. 3294–3302.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States.*, 1106–1114.

Laird, J. E., and van Lent, M. 2001. Human-level AI's killer application: Interactive computer games. *AI Magazine* 22(2):15–26.

Le, Q., and Mikolov, T. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 1188–1196.

Long, J.; Shelhamer, E.; and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. 3431–3440.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *CoRR* abs/1301.3781.

Mikolov, T.; Yih, W.; and Zweig, G. 2013. Linguistic regularities in continuous space word representations. Association for Computational Linguistics.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; Petersen, S.; Beattie, C.; Sadik, A.; Antonoglou, I.; King, H.; Kumaran, D.; Wierstra, D.; Legg, S.; and Hassabis, D. 2015. Human-level control through deep reinforcement learning. *Nature* 518(7540):529–533.

Navarro, S. E.; Gorges, N.; Wörn, H.; Schill, J.; Asfour, T.; and Dillmann, R. 2012. Haptic object recognition for multi-fingered robot hands. In *2012 IEEE Haptics Symposium (HAPTICS)*, 497–502. IEEE.

Nematzadeh, A.; Meylan, S. C.; and Griffiths, T. L. 2017. Evaluating vector-space models of word representation, or, the unreasonable effectiveness of counting words near other words.

Osborn, J. C.; Summerville, A.; and Mateas, M. 2017. Automated game design learning. *Computational Intelligence in Games*.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, 1532–1543.

Saha, T. K.; Joty, S. R.; Hassan, N.; and Hasan, M. A. 2016. Dis-s2v: Discourse informed sen2vec. *CoRR* abs/1610.08078.

Song, H. O.; Fritz, M.; Goehring, D.; and Darrell, T. 2015. Learning to detect visual grasp affordance. In *IEEE Transactions on Automation Science and Engineering (TASE)*.

Tran, K.; He, X.; Zhang, L.; Sun, J.; Carapcea, C.; Thrasher, C.; Buehler, C.; and Sienkiewicz, C. 2016. Rich image captioning in the wild.

Vinyals, O.; Ewalds, T.; Bartunov, S.; Georgiev, P.; Vezhnevets, A. S.; Yeo, M.; Makhzani, A.; Küttler, H.; Agapiou, J.; Schrittwieser, J.; Quan, J.; Gaffney, S.; Petersen, S.; Simonyan, K.; Schaul, T.; van Hasselt, H.; Silver, D.; Lillicrap, T. P.; Calderone, K.; Keet, P.; Brunasso, A.; Lawrence, D.; Ekermo, A.; Repp, J.; and Tsing, R. 2017. Starcraft II: A new challenge for reinforcement learning. *CoRR* abs/1708.04782.

Zhu, Y.; Fathi, A.; and Fei-Fei, L. 2014. Reasoning about object affordances in a knowledge base representation. In *ECCV*.