AFRL-AFOSR-VA-TR-2023-0099

Architectures for Continuously Learning Agents

Mitchell, Tom
CARNEGIE MELLON UNIVERSITY
5000 FORBES AVE
PITTSBURGH, PA, 15213
USA

10/19/2022
Final Technical Report

DISTRIBUTION A: Distribution approved for public release.

Air Force Research Laboratory
Air Force Office of Scientific Research
Arlington, Virginia 22203
Air Force Materiel Command

REPORT DOCUMENTATION PAGE

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE	**2. REPORT TYPE**	**3. DATES COVERED**	
20221019	Final	**START DATE** 20190401	**END DATE** 20200831

4. TITLE AND SUBTITLE
Architectures for Continuously Learning Agents

5a. CONTRACT NUMBER	**5b. GRANT NUMBER**	**5c. PROGRAM ELEMENT NUMBER**
	FA9550-17-1-0218	61102F
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**

6. AUTHOR(S)
Tom Mitchell

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	**8. PERFORMING ORGANIZATION REPORT NUMBER**
CARNEGIE MELLON UNIVERSITY 5000 FORBES AVE PITTSBURGH, PA 15213 USA	

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	**10. SPONSOR/MONITOR'S ACRONYM(S)**	**11. SPONSOR/MONITOR'S REPORT NUMBER(S)**
Air Force Office of Scientific Research 875 N. Randolph St. Room 3112 Arlington, VA 22203	AFRL/AFOSR RTA2	AFRL-AFOSR-VA-TR-2023-0099

12. DISTRIBUTION/AVAILABILITY STATEMENT
A Distribution Unlimited: PB Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
In recent years, progress in artificial intelligence has led to increasingly ambitious efforts to build integrated intelligent agents, from robotic agents that perceive and affect the physical world, to software agents that perceive and affect their cyber-world. Despite significant progress in the component parts of these agents (e.g., progress in computer vision, language processing, etc.), a key open question remains: what kind of software architecture is needed to integrate these components into a continuously self-improving intelligent agent? This is an increasingly important question, as embedded intelligent systems in continuous operation are becoming increasingly widespread in commercial and military systems. Such continuously operating systems, with sensors and effectors that perceive and act on their environment, are exposed to a continuous stream of data that in many cases could be used for automatic self improvement, if we understood how to architect these systems appropriately. Beyond its practical importance, the question of how to architect continuously learning agents is also at the core of the scientific understanding of intelligence. We propose here an experimental and theoretical research program to study this question. If successful, this research will provide new guidance for design of continuously learning sensor-effector agents across many domains, and many embedded systems. Our goal is to produce both theoretically justified design principles and experimental demonstrations of successful continuous learning systems.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	UU	4
U	U	U		

19a. NAME OF RESPONSIBLE PERSON	**19b. PHONE NUMBER** *(Include area code)*
RICHARD RIECKEN	696-9736

Attn: Dr. Doug Riecken, AFOSR/RTA2-7

Project Title: Architectures for Continuously Learning Agents

Final Report Covering: April 1, 2019 through August 31, 2020

AFOSR Award # FA9550-17-1-0218

Principal Investigator: Tom M. Mitchell
tom.mitchell@cs.cmu.edu
412 268-2611
Machine Learning Department
Carnegie Mellon University

Recipient Organization: Carnegie Mellon University
Business office email: rdo@andrew.cmu.edu
Report Due Date: 2020-11-29
Report Period Start Date: 2019-04-01
Report Period End Date: 2020-08-31
Current Program Officer: Dr. Doug Riecken
Distribution statement: this report is cleared for public release

Report Abstract

In recent years, progress in artificial intelligence has led to increasingly ambitious efforts to build integrated intelligent agents, from robotic agents that perceive and affect the physical world, to software agents that perceive and affect their cyber-world. Despite significant progress in the component parts of these agents (e.g., progress in computer vision, language processing, etc.), a key open question remains: what kind of software architecture is needed to integrate these components into a continuously self-improving intelligent agent? This is an increasingly important question, as embedded intelligent systems in continuous operation are becoming increasingly widespread in commercial and military systems. Such continuously operating systems, with sensors and effectors that perceive and act on their environment, are exposed to a continuous stream of data that in many cases could be used for automatic self improvement, if we understood how to architect these systems appropriately. Beyond its practical importance, the question of how to architect continuously learning agents is also at the core of the scientific understanding of intelligence. We propose here an experimental and theoretical research program to study this question. If successful, this research will provide new guidance for design of continuously learning sensor-effector agents across many domains, and many embedded systems. Our goal is to produce both theoretically justified design principles and experimental demonstrations of successful continuous learning systems.

Report Document

During this period we pursued multiple research thrusts as follows:

1. Instructable Intelligent Agents

In this research we developed a prototype AI agent based in a mobile phone, which can be instructed to perform new tasks, through a combination of speech and demonstration. For example, one migh instruct the agent that "Whenever it snows at night, wake me 30 minutes earlier." Initially, our agent will respond to this request by stating "I don't understand -- do you want to teach me?". At this point the user may say yes, and proceed to explain and demonstrate, e.g., that to determine whether it is snowing the agent should use its weather app, and to wake me 30 minutes earlier it should open the alarm app, and substract 30 minutes from the alarm time.

In our publication [3] PUMICE: A Multi-Modal Agent that Learns Concepts and Conditionals from Natural Language and Demonstrations we began with the observation that natural language programming is a promising approach to enable end users to instruct new tasks for intelligent agents. However, our formative study found that end users would often use unclear, ambiguous or vague concepts when naturally instructing tasks in natural language, especially when specifying conditionals. Existing systems have limited support for letting the user teach agents new concepts or explaining unclear concepts. In this paper, we described a new multimodal domain-independent approach that combines natural language programming and programming-by-demonstration to allow users to first naturally describe tasks and associated conditions at a high level, and then collaborate with the agent to recursively resolve any ambiguities or vagueness through conversations and demonstrations. Users can also define new procedures and concepts by demonstrating and referring to contents within GUIs of existing mobile apps. We demonstrate this approach in PUMICE, an end-user programmable agent that implements this approach. A lab study with 10 users showed its usability.

In our publication [1] Interactive Task Learning from GUI-Grounded Natural Language Instructions and Demonstrations we further extended this line of research, introducing the SUGILITE system, an intelligent task automation agent that can learn new tasks and relevant associated concepts interactively from the user’s natural language instructions and demonstrations, using the graphical user interfaces (GUIs) of third-party mobile apps. This system provides several interesting features: (1) it allows users to teach new task procedures and concepts through verbal instructions together with demonstration of the steps of a script using GUIs; (2) it supports users in clarifying their intents for demonstrated actions using GUI-grounded verbal instructions; (3) it infers parameters of tasks and their possible values in utterances using the hierarchical structures of the underlying app GUIs; and (4) it generalizes taught concepts to different contexts and task domains. We describe the architecture of the SUGILITE system, explain the design and implementation of its key features, and show a prototype in the form of a conversational assistant on Android.

2. Contextual Parameter Generation

In this line of research we explored a novel approach to training deep neural networks to perform multiple distinct tasks. The general idea is to use our earlier Contextual Parameter Generation, applying it to a new task, and showing experimentally that it performs better than the previous state of the art. In particular, in our publication [2] Contextual Parameter Generation for Knowledge Graph Link Prediction, we consider the task of knowledge graph link prediction.

Given a question consisting of a source entity and a relation (e.g., Shakespeare and BornIn), the objective is to predict the most likely answer entity (e.g., England). Recent approaches tackle this problem by learning entity and relation embeddings. However, they often constrain the relationship between these embeddings to be additive (i.e., the embeddings are concatenated and then processed by a sequence of linear functions and element-wise non-linearities). We show that this type of interaction significantly limits representational power. For example, such models cannot handle cases where a different projection of the source entity is used for each relation. We propose to use *contextual parameter generation* to address this limitation. More specifically, we treat relations as the *context* in which source entities are processed to produce predictions, by using relation embeddings to generate the parameters of a model operating over source entity embeddings. This allows models to represent more complex interactions between entities and relations. We apply our method on two existing link prediction methods, including the current state-of-the-art, resulting in significant performance gains and establishing a *new state-of-the-art* for this task. These gains are achieved while also *reducing convergence time by up to 28 times*.

3. Relating Natural Language Processing in AI Systems to Language Processing in the Human Brain

In this research thrust we collected brain image data while human subjects research simple sentences such as "The dog at the bone." We then gave these same sentences to a number of state of the art AI natural language processing systems, and examined the correspondence between observed human brain activity, and activations in hidden layers of the AI neural network systems. The results are summarized in our publication [4] Relating Simple Sentence Representations in Deep Neural Networks and the Brain, and in the paragraphs below.

What is the relationship between sentence representations learned by deep recurrent models, and sentence representations in the human brain? Is there any correspondence between hidden layers of these recurrent models and brain regions when processing sentences? Can these deep models be used to synthesize brain data which can then be utilized in other extrinsic tasks? We investigate these questions using sentences with simple syntax and semantics (e.g., The bone was eaten by the dog.). We consider multiple neural network architectures, including recently proposed ELMo and BERT. We use magnetoencephalography (MEG) brain recording data collected from human subjects when they were reading the same sentences that we gave to the artificial neural network.

Overall, we find that BERT's hidden layer activations correlate the best with MEG brain data. We also find that the deep network representation can be used to generate synthetic brain data from new sentences to augment existing brain data. To the best of our knowledge, this is the first work showing that the MEG brain recording when reading a word in a sentence can be used to distinguish earlier words in the sentence. Our exploration is also the first to use deep neural network representations to generate synthetic brain data and to show that it helps in improving subsequent stimuli decoding task accuracy.

References

[1] Interactive Task Learning from GUI-Grounded Natural Language Instructions and Demonstrations, T. Li, T. Mitchell, and B. Myers, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations (ACL 2020)*, DOI: 10.18653/v1/2020.acl-demos.25, July 2020.

[2] Contextual Parameter Generation for Knowledge Graph Link Prediction, G. Stoica, O. Stretcu, E. A. Platanios, T. Mitchell, and B. Poczos, *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI 2020)*, 2020.

[3] PUMICE: A Multi-Modal Agent that Learns Concepts and Conditionals from Natural Language and Demonstrations , T Li, M. Radensky, J. Jia, K. Singarajah, T. Mitchell, and B. Myers, *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST 2019)*, DOI: 10.18653/v1/2020.acl-demos.25. (3 minute demo video). October 2019.

[4] Relating Simple Sentence Representations in Deep Neural Networks and the Brain, S. Jat, P. Talukdar, and T. Mitchell, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pp. 5137--5154, 2019.