DARPA ASKE TA1 ANSWER: M6 Report

Andrew Crapo, Varish Mulwad, Nurali Virani
GE Research
April 30, 2019

Introduction

As we begin Phase 2 we are excited to meet with other program participants in the PI meeting, and to learn more about related projects and research in the Modeling the World's Systems conference. This will provide an opportunity to reflect upon what we have accomplished and the road forward as we refocus our Phase 2 plans and prepare the M7 Milestone Report.

Since the M5 Milestone we have finished up some code testing and integration and have released all our source code and documentation to Open Source under the BSD 3-Clause License. This is available through our github.com repository[1], which has been made public. Below we reiterate our next steps for Phase 2. For more information please see the full report.[2]

Next Steps

The ANSWER Knowledge Extraction platform is a proof-of-concept to demonstrate knowledge-driven extraction and curation of scientific models from code and texts with human-in-the-loop. With respect to the ideal vision of the DARPA ASKE program, the current ANSWER solution is as yet incomplete. Important next steps to move towards the vision include the following:

1. We have designed a mechanism to capture essential information about the semantic types and relationships of the inputs and outputs to a scientific model, and to allow collaborative interaction with the user to supply missing information and to receive user corrections. However, we have not completed the integration of mixed-initiative interactions with text extraction and code extraction necessary for a smooth development of a complete, grounded semantic representation of a model's inputs and constraints/assumptions.
2. More reasoning over extracted code models is needed to identify exactly what code is potentially interesting from a scientific knowledge perspective. The mixed initiative interaction will also be expanded to allow the user to select code snippets to be converted to computational graph models.
3. Currently, we can translate Java code snippets to Python code as well as equations identified in text to python code. This resulting python code needs to be made TensorFlow eager mode-compatible python code, which has been accomplished for equations from text (see demos in DARPA-ASKE-TA1/Notebooks), but is still in progress for codes from the java-to-python service. Note that computational graphs are language-neutral and hence, can use TensorFlow in Java, C++, and Python to execute the model for inference.
4. Integration of the external sources, e.g., DBpedia, Wikidata, etc., with the locally extracted information is demonstrated but not complete. The concept of locality in extraction from text has been identified and validated to some extent but not completely implemented. Additionally,

[1] Available at https://github.com/GEGlobalResearch/DARPA-ASKE-TA1.
[2] M5 report is available at https://github.com/GEGlobalResearch/DARPA-ASKE-TA1/wiki/Milestone-M5,-April-1,-2019.

we plan to augment the model extracted from text and code by identifying relations between scientific concepts (such as *depends*, *causes* etc.).

5. The computational graph construction from semantic models and execution for inference has been implemented to support both physics-based and data-driven models. However, graph operations of appending computational graphs in series or parallel as more knowledge is curated by the agent has not been addressed yet. This key capability to create large computational graphs which are consistent with semantic models will be addressed in Phase 2. One issue is that consistency of scientific units across models that are built is now tracked by semantic models, but if computational graphs can chain themselves together for inference, then the consistency of units and unit conversion will also have to be handled by the computational graph framework. We will explore this feature in Phase 2 using packages like pint and unyt.
6. Based on maturity of TensorFlow Probability (TFP) after stable release of TF 2.0 in Q2 2019, we will use TFP to update the computational graphs to represent and compute uncertainty in knowledge.
7. We wish to extend the current ANSWER agent to evolve over time by ingesting codes, more documentation and related publications, interaction with human users, and maintain provenance and confidence information of extracted knowledge from different sources.
8. In Phase 1 we created the Dialog grammar, an extension to the SADL grammar, to explore mixed-initiative interactions with the user. We did this in an Eclipse-based, Xtext editor window using an obscure Xtext extension to allow programmatic modification of editor content, thus allowing the ANSWER agent to put text into the editor. We chose to this approach so that all knowledge graph concept references in the dialog would be hyperlinked to their definitions and references, thus providing a natural, human-friendly way of providing drilldown capability. In Phase 2 we will need to make the mixed-initiative support much more robust while retaining the hyperlinking aspects and will be exploring ways of doing this, potentially moving the UI to a browser-based solution.

We will refine this list as we make sure that our work is aligned with what we learn in the May PI meeting.