Responsible AI Impact Assessment: Research Project

Section 1: Project Information

Project profile

__1.1__ Complete the project information below.

Project Name	GenAIScript
Group	Microsoft Research Redmond / RiSE
Point of contact	Ben Zorn (zorn@microsoft.com)

Track revision history below.

Authors	Peli de Halleux, Ben Zorn, Michal Moskal
Last updated	1/18/2024

Identify the individuals and institutions (including academic) who are partners, contributors, or reviewers of this project.

Contributors with affiliations	Russell Conard, DevDiv; Markus Kuppe, MSR RiSE; Pantazis Deligiannis, MSR India; Tom Ball, MSR RiSE

Project components and timeline

__1.2__ Please describe the components of your project, including what you've done so far, what you plan to do, conferences you plan to attend, current or future partners, and your desired timeline.

Project description
GPTools is a framework that empowers teams, including non-developers, to create and use AI-enhanced scripts to support their workflows. GPTools provides support for authoring and debugging JavaScript scripts that incorporate calls to foundation models in their execution. GPTools is a programming framework that allows its users to author AI scripts (which we call a GPTool), author and debug those scripts in a development environment that is an extension of VS Code, and package those scripts as command-line scripts that can be deployed in many contexts. Our VS Code extension supports easy authoring of a GPTools by writing natural language in markdown syntax plus a small amount of stylized JavaScript programming. Our dev environment allows users to leverage multiple LLM models, parameterize the calls to the models, execute scripts, trace the construction of the LLM prompts and provide a full trace of execution from prompt construction to LLM generation. Our framework also supports extracting multiple forms of output from LLM generations, including output in files of different types, outputs intended as edits to existing files and outputs in structured formats, such as JSON. Note that GPTools does not in any way require the execution of an LLM in its operation. It allows users to write tools that use LLMs but the framework we provide is an enabler for writing the tools, much as VS Code is an enabler for writing programs in different programming languages but does

not itself use those languages when it executes. When being used, GPTools enables users to specify the LLM that will power their tool and provide necessary credentials to invoke it.

Supporting material

1.3 If you have links to any supplementary information about the project such as slide decks, demos, or particularly relevant prior work / preprints, please include links below. Please also link to any related R&CT records (consulting, dataset onboarding, user studies, release or other projects, departing interns) or ERP records.

Description of supplementary information	Link
GitHub private repo (MS internal)	microsoft/gptools: AI Scripting for Teams (github.com) (this requires you have a GitHub login that is linked to your MS login. More info about that here: Account linking and join organizations | Docs - Microsoft Open Source)
Azure hosted repo (clone of GitHub, doesn't require linking accounts)	Summary - Overview (azure.com)
White paper describing GPTools	gptools/packages/whitepaper/gptools-overview.md at main · microsoft/gptools (github.com)
Presentation describing GPTools	gptools Overview January 2024.pptx
Readme file from repo with links to various demonstrations of uses of GPTools	microsoft/gptools: GenAI Scripting (github.com)

Project goal and overview

1.4 Briefly explain, in plain language, what the goal of the project is. ***Do not copy your paper abstract.*** *Help your peers outside your area of expertise understand your work. Readers of this document can drill down by visiting your supplementary links above.*

Project goal and overview
We believe that scripts that leverage the power of LLMs will become increasingly important and widely used by both developers and non-developers. Languages like JavaScript and Python have had huge impact because they provided a wide audience with capabilities that were not present in previous languages. We believe the same thing will happen with AI and GPTools is an effort to create a language and framework to enable individuals to create, use, and share AI-powered scripts.

Relationship to products

1.5 Briefly describe how this project relates to any Microsoft systems or products (if applicable). Is it likely for this project to be integrated into future products? Do you have an existing collaboration or plans for a collaboration?

Relation to other products/features

GPTools can be used by product groups to create general scripts for various purposes, just as product groups write scripts in PowerShell, Python, etc. currently. GPTools might be used to define extensibility mechanisms in existing Copilot products but we have not explored that possibility in any depth.

Section 2: Sharing

What?

***2.1** What assets are you planning to share at the point of the milestone we're reviewing now? (Check all that apply)*

☒ demo
☒ source code
☐ model(s)
☐ data
☒ project webpage
☒ paper
☐ other (Explain)

Who?

***2.2** Who are you sharing with? (Check all that apply)*

☒ Open Source/Public
☒ paper reviewers
☐ customer(s) (please list)
☐ partners (please list)
☐ product teams (please list)
☐ other (please describe)

Why?

***2.3** What are your goals for sharing these assets now? (Check all that apply)*

☒ Share with research community for replication to support a publication
☐ Share with partner to support collaboration
☒ Share with public to generate excitement for this work/MS
☒ Other reason (please describe) To help Microsoft demonstrate technical leadership around empowering individuals to leverage the capabilities of LLMs and foundation models.

Section 3: Data information and considerations

Data documentation

***3.1 REMINDER:** If you have any dataset onboarding records, user study records in which you gathered data, or additional data documentation (git hub docs, readmes, description in your paper), please link to these under section 1.3 above.*

Data reflections

3.2 *Brainstorm the top 1-5 impacts that the makeup of your data could have on your work. Find seed questions here: aka.ms/tnrdataquestions*

Impacts of data makeup
Our release does not leverage or contain any data.

Section 4: Project impacts and limitations

Impacts of sharing

4.1 *Please describe what impact successful sharing will have on your work, your field of research, and Microsoft.*

Impacts of sharing the AI technology
We anticipate that by sharing GPTools, other researchers and individuals will consider using our framework to author and share new AI-powered scripts. By having a common, open source framework in which to author, debug, and deploy AI-powered scripts, we hope that collections of such scripts will be packaged and shared much as libraries in Python or JavaScript are shared.

Known limitations

4.2 *Reflecting on your data (see 3.2 above) and your techniques, what are the current limitations of the system? This could include scenarios where the system will not perform well (e.g., a language system being use in a natural language that is not supported, or an image generation tool what works better for scenery then for images of humans), environmental factors to consider (lighting for images, or background noise for audio recordings), or other factors to be aware of.*

Known limitations
We continue to expand the capabilities of GPTools as we work with individuals on specific use-cases. We anticipate that we will continue to improve the framework as new use-cases arise. For example, we recently added the ability to use open source small language models (SLMs) such as Phi2 in GPTools. Because the user has the ability to choose what foundation model a particular GPTool uses, the capabilities of the specific GPTool they create are their responsibility and not ours.

Risks of sharing

4.3 *If a malicious entity were to use the assets you share, what could happen? What are the most obvious abuse scenarios? What could go wrong? Will anyone be hurt? In what ways?*

Abuse risks
Just as any bad actor can write malicious code in Python or JavaScript, an adversary could write a malicious GPTool. Similarly, an adversary could use VS Code to write malicious software. Our project is a framework that makes writing AI-powered scripts easier but anything a user could create using GPTools could also be created without it, albeit with more time and effort. We provide guidance in the GPTools documentation to provide users with knowledge of Responsible AI best practices when building GPTools.

***4.4** Is it possible for your assets to be misused unintentionally? For example, by a non-familiar or novice user or used in an operational environment that the system was not designed for? What could go wrong? Will anyone be hurt? In what ways?*

Unintended use risks
GPTools encourage users to write, debug, and deploy AI-powered scripts. Existing frameworks that integrate foundation models and software, such as Sematic Kernel, already allow this possibility. GPTools does not enable any new unintended use scenarios that are not already present with existing software frameworks.

Mitigations for immediate risks

***4.5** Check all mitigations you intend to use for risks identified and add any additional mitigations in the box provided.*

Limiting open release:

☐ Not releasing code/models/data

☐ Providing a mitigated demonstration web site in lieu of releasing high risk models/code/data publicly

☐ Leveraging a process similar to the ACM artifact evaluation process – i.e., providing code/models/data specifically to reviewers without public release

☐ Gating the public release through an email/form to the team

Monitor use for abuse:

☐ Releasing in such a way that usage can be tracked

☐ Actively setting up alerts and escalations when abusive patterns are discovered

Clear and complete transparency documentation:

☒Ethics and Responsible AI statements in paper

☐Model card

☐ Transparency note

☒RAI section in product documentation

☒RAI information in blogs and other messaging (please describe): We will highlight the importance of responsible use of GPTools in the messaging announcing the project and also highlight that GPTools themselves can be used in ensure other AI-based tools are created responsibly.

☐ Any limitation of release should clearly state the RAI reasoning for not sharing the technology openly

☐ Clearly communicates the intention of use only in research settings

☐ Content moderation (please describe): ____________________

☐ Prompt engineering

☐ Code of conduct

☐ Research only license

☐ Other:

Section 5: Thinking into the future

Possible real-world uses

***5.1** List the most likely real-world uses of the research project. What might happen when this research leaves the lab and moves forward? Are there characteristics of your technique that make it particularly well suited for particular use cases?*

Real-world use	Description
Writing scripts to detect issues with software deployments (checking for incorrect usage, security, etc.)	Existing tools like lint help developers find issues in software. Writing a GPTool that works similarly can be relatively easy and more effective.
Scripts that help users manage collections of documents (e.g., code, specifications, etc.) for correctness, consistency, style, content, etc.	GPTools can be written that represented different roles (such as an architect, quality assurance, etc.) and check related documents for consistency, etc.
Writing tools that apply existing software tools to foundation model outputs to ensure correctness and/or other properties	GPTools can be written that both use a foundation model to generate outputs and then check the resulting generations for correctness using existing correctness tools.

Possible real-world misuses

***5.2** List the most likely real-world misuses of the research project. What might happen when this research leaves the lab and moves forward? Where is it likely to be misused or potentially abused. Think like a novice and an attacker?*

Real-world misuse/abuse	Description
GPTools might make building adversarial scripts easier.	Currently, an adversary would need to write Python code to use autogen, or Sematic Kernel, for example. GPTools might make it easier to write such scripts.
Overreliance on the GPTools scripts that users write	Because a user can use GPTools to create a script and then use that script in some other automation process, the user might not realize that because the GPTool uses AI, the outcome from the script is not as reliable as it might be if it were purely written in software.

Stakeholders

***5.3** instructions…*

Stakeholders
Writers of GPTools
Users of GPTools
Foundation model creators

Intended Uses

5.4 What are the intended use cases for this work?

Intended use cases
Building general purpose scripts that automate processes using AI.

5.5 Who are the possible stakeholders that would be most impacted by both your research and the future real world uses?

Stakeholders
Non-programmers using and modifying tools: GPTools can help automate processes that were not possible to automate before. For example, a GPTool can be written that checks the consistency of instructions written in different human languages (like English and French) and flags cases where the instructions are inconsistent. **Developers creating and using new tools**: GPTools can be written, debugged, and made general by individuals with programming skill and experience but used by individuals without that knowledge and experience (just as non AI-scripts are written). **Foundation model developers:** GPTools enable the user to define what foundation models are used for a particular script. Foundation model developers might want to tune the performance of their models to perform well for a particular widely used GPTool if one existed.

5.6 Which use cases and stakeholders is this work best suited for? Why?

Best use case/stakeholder scenarios
The best early users and developers of GPTools are existing software developers who will benefit from the new power the AI element provides. They will appreciate the limitations of the foundation models they use in a tool and build checking mechanisms to prevent or avoid bad generations.

Harms and mitigations

5.7 After reflecting you your real-world use cases, stakeholders and data, what are the anticipated harms to stakeholders and how can the identified harms be mitigated? Some mitigations may serve for more than one stakeholder and use case.

Stakeholders/Use Case	Potential harms	Potential mitigation
Non-developer GPTool users using a tool	Overreliance on a GPTool	Messaging when a GPTool runs that the results produced are generated by a foundation model
GPTool developer writing a tool	Building a tool that does not follow RAI best practices	Messaging in repository about proper use of GPTools, RAI risks, and possible ways to use GPTools themselves to help check for improper RAI practices in other tools
Non-developer modifying an existing tool	A non-developer may modify an existing tool and cause it to break	Guidance for non-developers to be careful when modifying GPTools taking into account the potential to reduce robustness