
GenAIScript: Robust Scripting for Generative AI

Peli de Halleux, Markus Kuppe, Michal Moskal, Madan Musuvathi, Benjamin Zorn

April 2024

Abstract

Scripting languages, like Python and JavaScript, have enabled major advances in computing but were created prior to the advent of Large Language Models (LLMs). We introduce GenAIScript, a scripting language that makes LLMs a first-class part of the scripting process, allowing users to author, debug, and deploy LLM-based scripts that can perform tasks beyond the reach of conventional code. GenAIScript is intended to be used by users with different programming experience and applied to diverse tasks previously unable to be solved using traditional code. By extending the popular Visual Studio Code environment, GenAIScript simplifies the creation of robust and reusable scripts that can intelligently process and generate content, including code, documentation, and structured data. We explain what GenAIScript is, how it is implemented, and illustrate its use in several production deployments. GenAIScript is open-source and available to use now at: <https://aka.ms/genaiscript>.

1. Introduction

GenAIScript is a JavaScript-compatible¹ scripting language that allows users to write scripts to automate complex tasks by leveraging LLMs as a first-class script element.

The dramatic rise in the capabilities of artificial intelligence has revolutionized the way we interact with technology, offering unprecedented opportunities to automate and enhance many tasks. Incorporating calls to LLMs (or more generally any foundation model) at runtime into a software system greatly enhances what that software can do. The disruptive nature of this transition forces us to categorize software into Plain Ordinary Software (POSW), which does not use the

power of LLMs at runtime, and the more capable AI Software (AISW) that does leverage LLMs².

POSW represents all software written before the creation of LLMs and includes operating systems, applications, scripts, etc. While POSW is a key element of our existing computing infrastructure, AISW, with its enhanced capabilities, will dominate the development of future software systems. Our goal is to make GenAIScript the language for scripting AISW.

Historically, scripting languages like Perl, JavaScript, and Python, have become widely successful because they provide two key elements: (1) they allow diverse users to write small but useful programs, and (2) they give users access to new computing capabilities not previously available to programs (e.g., JavaScript gave its users access to the Web).

Given that context, the goal of GenAIScript is to empower a wide class of potential users to easily leverage the power of AISW. GenAIScript enables users, including those without extensive programming expertise, to author, debug, and deploy scripts that seamlessly incorporate calls to LLMs. GenAIScript is a JavaScript-compatible language with a dedicated VS Code extension to simplify the scripting process, making it accessible to a broad audience.

What are GenAIScripts used for? We believe that the incredible flexibility of LLMs allows our scripts to be used in many contexts for many purposes. Some of the scripts we have personally explored include: extracting information from images using the gpt-4-turbo-v model, extracting exact quotes from documents to determine if the content answers specific review questions, planning a vacation to a destination specified as a parameter leveraging information from search and web-hosted documents about the

¹ Currently GenAIScripts are JavaScript-compatible but in the future we anticipate bindings for other languages including Python.

² For a discussion of the implications of POSW versus AISW, see the SIGPLAN blog post: [AI Software Should be More Like Plain Old Software | SIGPLAN Blog](#)