

AI in Coding: The Era of Humans Writing Syntax is Over



Jacob Adams
jdadams@utah.gov

Scott Davis
stdavis@utah.gov



Resource Sheet
<https://bit.ly/4cM4hkX>



Scott
Ryan Dahl - NodeJS

Jake - Set a stopwatch

Outline



Core Concepts

- AI & Agentic Coding
- GitHub's Copilot



Best Practices

- Slop Cannon Suppression
- Micro Tasks & Tools
- Quality Checks
- Refactoring



The Future

Where is this going?

Story/Demo

Create data classes for holding all the information about wordpress media items from a Wordpress REST endpoint



What Is AI?

AI is just a Large Language Model (LLM)

An LLM is just a mathematical construct that guesses what comes next based on probabilities



Jake

Explain LLMs
Explain models

What Likely Comes Next?

Input: Four score and

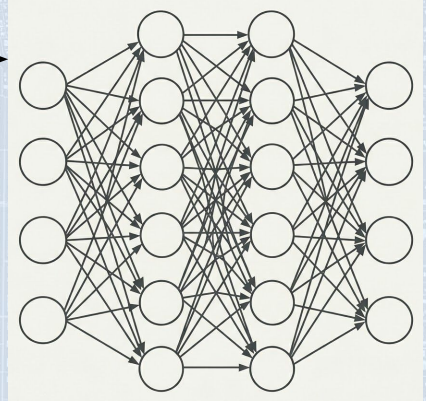


Jake

Explain LLMs
Explain models

What Likely Comes Next?

Input: Four score and

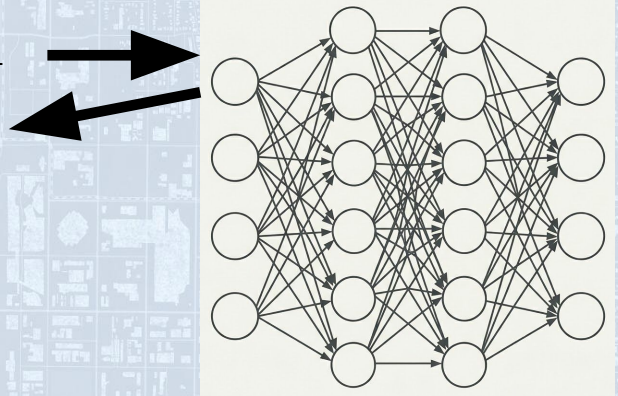


Jake

Explain LLMs
Explain models

What Likely Comes Next?

Input: Four score and
Four score and seven

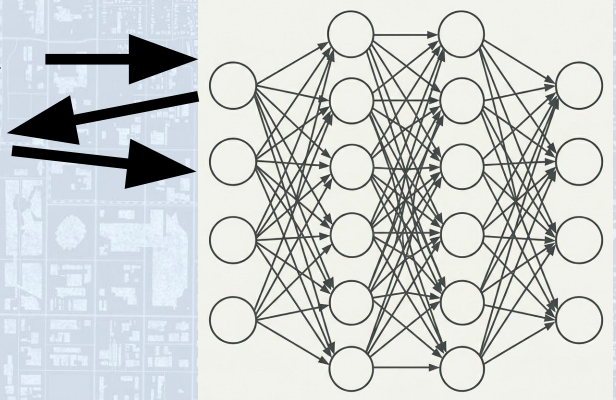


Jake

Explain LLMs
Explain models

What Likely Comes Next?

Input: Four score and
Four score and seven



Jake

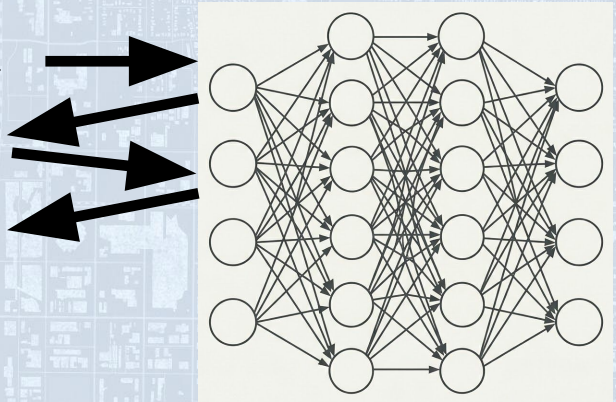
Explain LLMs
Explain models

What Likely Comes Next?

Input: Four score and

Four score and seven

Four score and seven
years



Jake

Explain LLMs
Explain models

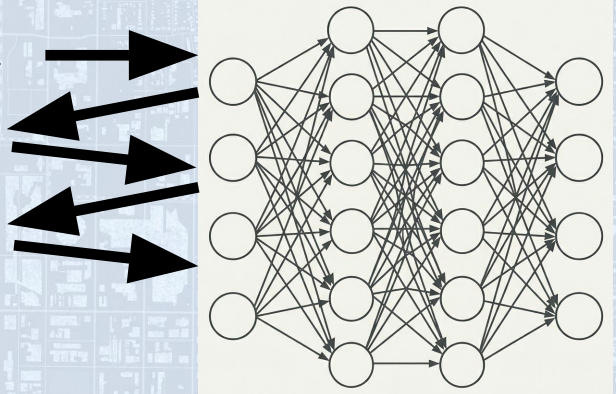
What Likely Comes Next?

Input: Four score and

Four score and seven

Four score and seven
years

...



Jake

Explain LLMs
Explain models

Tokenization

LLMs operate on tokens: words (or common bits of words), punctuation, etc that are assigned numeric values

AI providers charge you based on the number of input and output tokens.



Jake

Explain LLMs
Explain models

Tokenization

Four Score and seven years ago

[10051, 6115, 323, 5015, 984, 4636]

10051: [-0.0151 -0.0212 0.0084...]

6115: [-0.0042 0.0118 -0.0254, ...]

323: [0.0015 -0.0183 -0.0076 ...]



Jake

Explain LLMs
Explain models

Tokenization

Each token is assigned a long (1.5-3k) list, or vector, of numbers from -1 to 1 that embeds it's meaning and relationship to other tokens

```
10051: [-0.0151 -0.0212 0.0084...]  
6115: [-0.0042 0.0118 -0.0254, ...]  
323: [0.0015 -0.0183 -0.0076 ...]
```



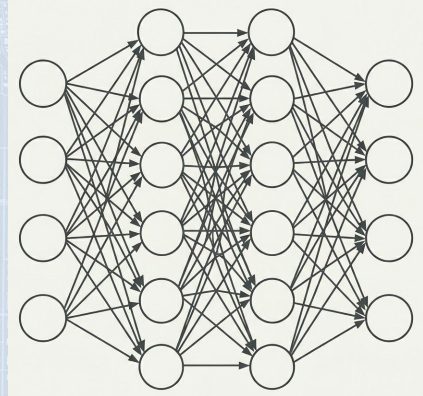
Jake

Explain LLMs
Explain models

Weights

The vectors are passed into the model, which consists of many layers of "transformers"

These transformers perform lots of additions and multiplications on the vectors

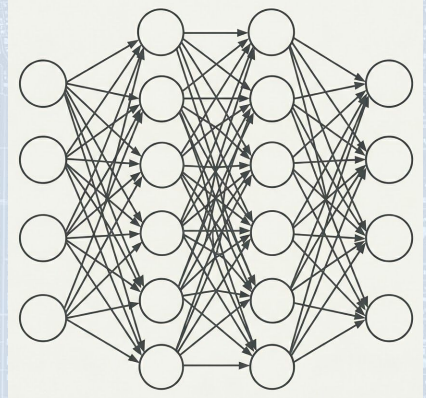


Jake

Explain LLMs
Explain models

Weights

The final layer of transformers returns the probability of every token in its vocabulary to be the next token



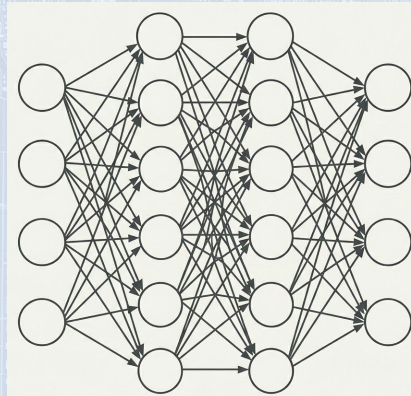
Jake

Explain LLMs
Explain models

Weights

The LLM black box multiplies these arrays of numbers by billions of other numbers (which are the result of the model training process) multiple times

The final iteration returns the probability of every token in its vocabulary to be the next response



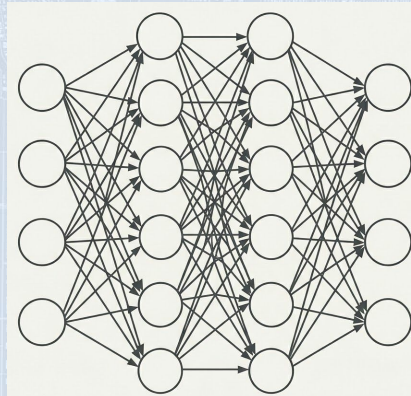
Jake

Explain LLMs
Explain models

Weights

The more weights in a model, the more info it has about relationships between the tokens in its training dataset

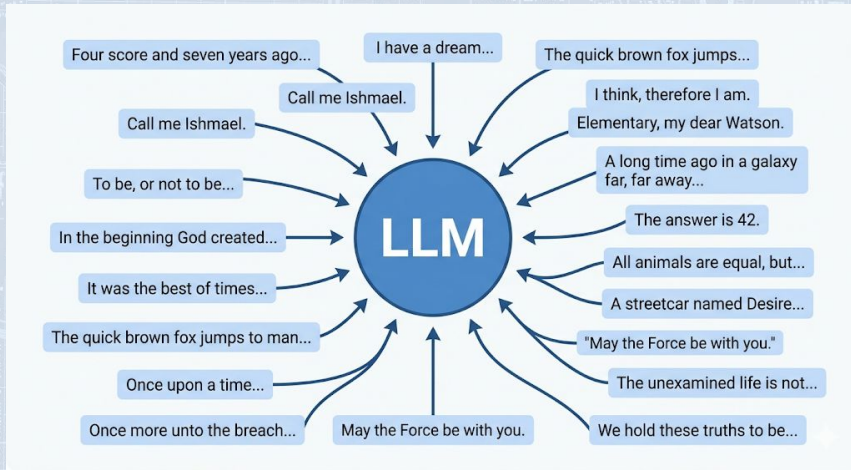
However, the more weights, the more computational power required to infer the next token



Jake

Explain LLMs
Explain models

Training



Jake

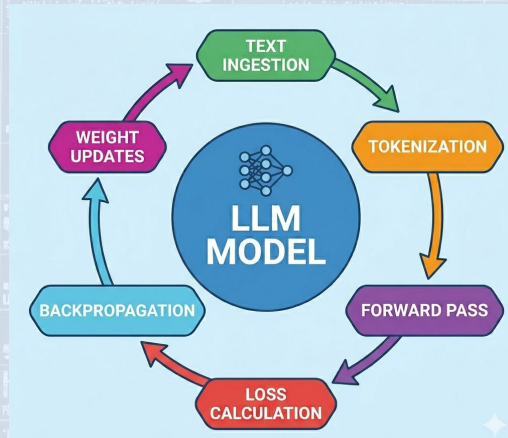
Explain LLMs
Explain models

Training

Start with random values for each weight

Feed in a training text one token at a time, tweaking the weights until it correctly predicts the next

Shampoo, rinse, repeat



Jake

Explain LLMs
Explain models

Models

An AI model is the combination of transformer architecture plus weights

Each model is unique based on its training data and methods



Claude



Gemini



Jake

Explain LLMs
Explain models

Context

Each model has its own limit for the number of tokens it includes in each prompt, called its context

It includes instructions, inputs, discussion history, and any external context (files, MCP connections, etc)

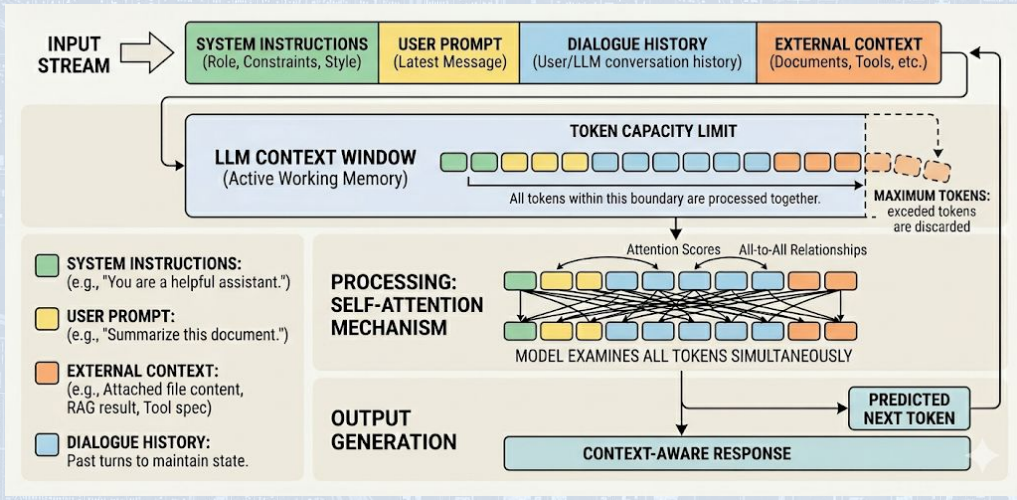
More importance is put on tokens in the beginning and end of the context window, less on the middle



Jake

Explain LLMs
Explain models

Context



Jake

Explain LLMs
Explain models

Hallucinations

Traditional programming is deterministic: we set the value of variable `name` to `Jake`, and everytime we refer to `name` it returns `Jake`

LLMs are predictive: it guesses the correct output based on the patterns it discovered in its training data as encoded in its weights combined with the current running context



Jake

Explain LLMs
Explain models

What Is AI?

- Predictive guess of what comes next based on terabytes of training data
- Non deterministic
- Purely based on patterns, not hard facts
- Entirely dependent upon its training data
- Subject to disagreements with reality (hallucinations)
- Delivered as individual models with different patterns derived from different training methods and datasets



Jake

Explain LLMs
Explain models



GitHub Copilot

!=

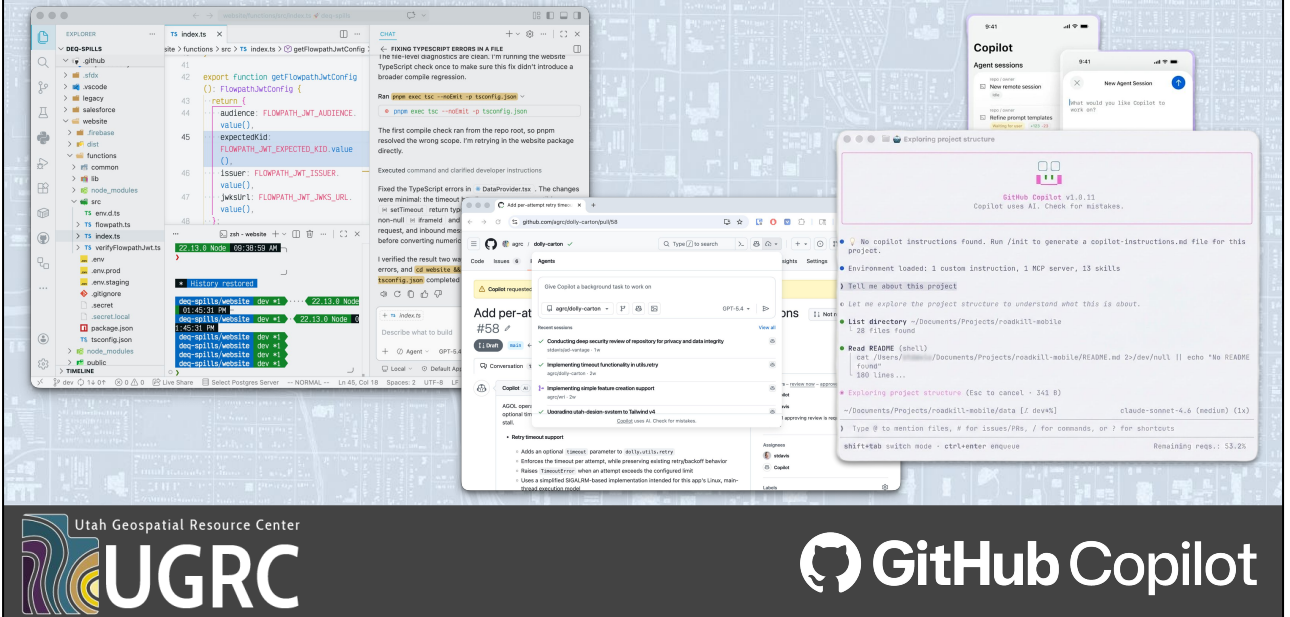


Microsoft Copilot



Scott

Write code faster and with less effort



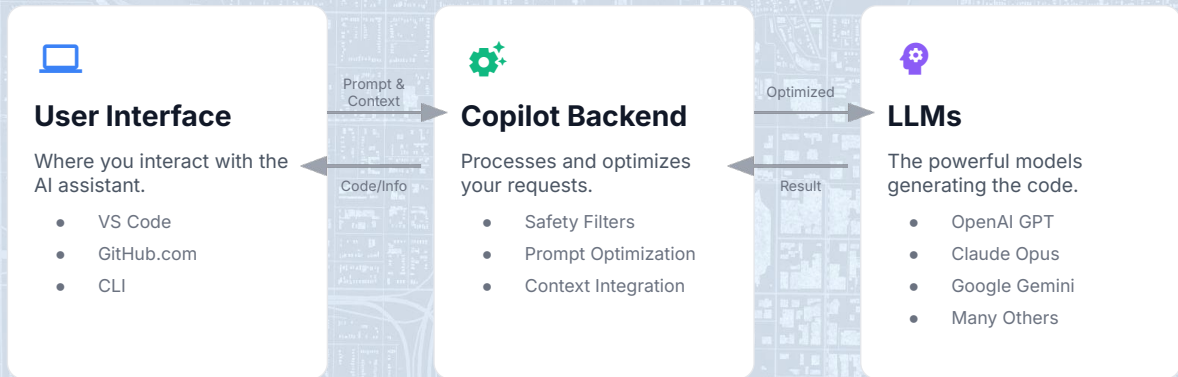
Scott
VSCode (and other IDEs)
[GitHub.com](https://github.com)
CLI
Mobile

Many other options:
Anthropics Claude Code
OpenIA Codex
Cursor

The principles that we will cover are universal.



AI Harness Data Flow

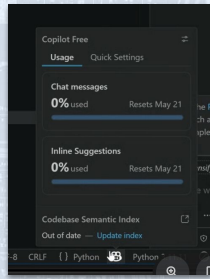
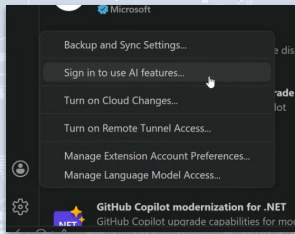


Scott

Copilot is not an LLM. It is a set of tools that you use to interact with an LLM to help code.

Getting Started

1. Create GitHub account
2. Install Visual Studio Code
3. Sign in with GitHub account

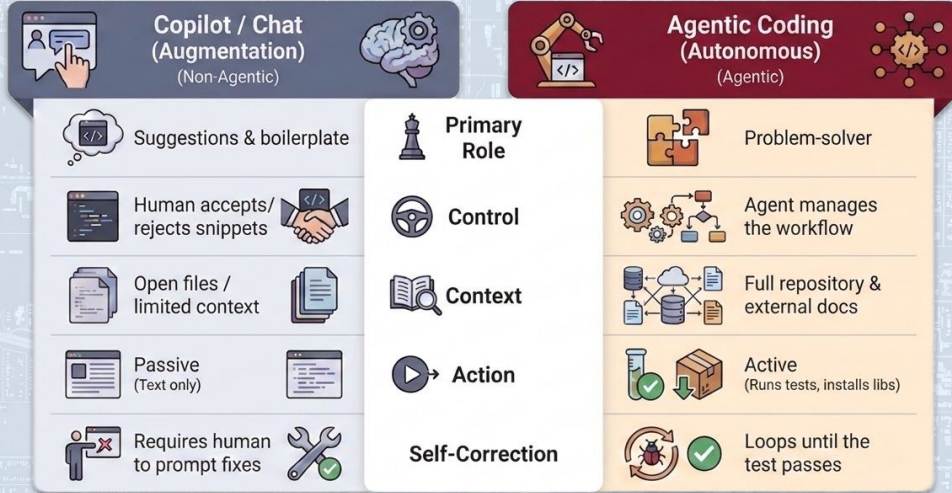


Free	Pro <small>Most popular</small>	Pro+
A fast way to get started with GitHub Copilot.	Accelerate workflows with GitHub Copilot.	Scale with agents and more models.
\$0 USD	\$10 USD per user / month	\$39 USD per user / month
Get started	Get started	Get started
Open in VS Code		
What's included: <ul style="list-style-type: none">✓ 50 agent mode or chat requests per month✓ 2,000 completions per month✓ Access to Haiku 4.5, GPT-5 mini, and more✓ Copilot CLI	Everything in Free and: <ul style="list-style-type: none">✓ Copilot cloud agent✓ Copilot code review✓ Claude and Codex on GitHub and VS Code✓ 300 premium requests, with the option to buy more!✓ Unlimited agent mode and chats with GPT-5 mini!✓ Unlimited inline suggestions✓ Access to models from Anthropic, Google, OpenAI, and more	Everything in Pro and: <ul style="list-style-type: none">✓ Access to all models, including Claude Opus 4.7 and more✓ 5x as many premium requests as Pro to use the latest models, with the option to buy more!◆ Access to GitHub Spark



Scott

Non-agentic vs. Agentic



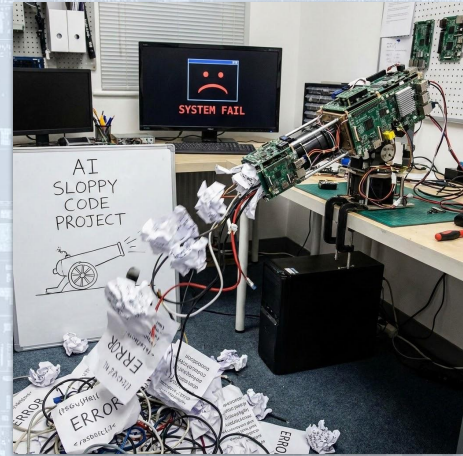
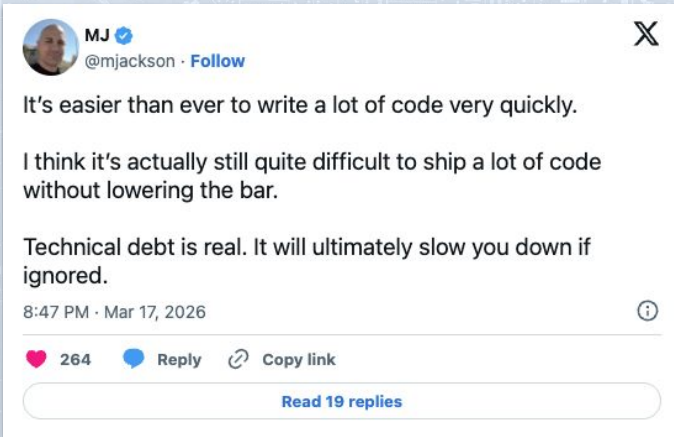
Scott
Demo - VSCode
New Python File

```
def multiply(
```

Hit tab to show auto-complete

```
Write tests for these functions. Make sure that they all pass.
```

Slop Cannon Suppression



Jake?

Michael Jackson - Remix/React Router

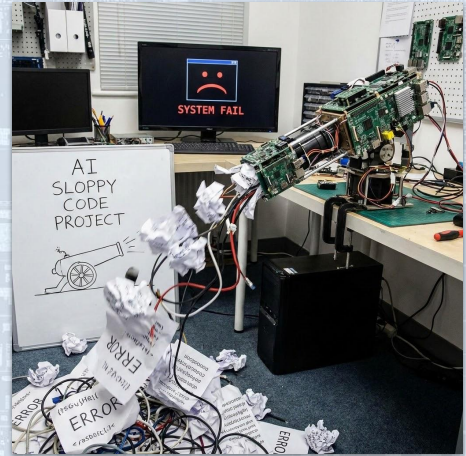
We are going to talk about some best practices to avoid sloppy code that can sometimes be generated by these models.

Slop Cannon Suppression

If you can push a button and get content, then that content is a commodity. Its value is next to zero.

Counterintuitively, that's my biggest reason to be optimistic about AI and creativity. When hard parts become easy, the differentiator becomes love.

Andy Conen, AI Dev at Google
cannoneyed.com/projects/isometric-nyc



Jake?

Michael Jackson - Remix/React Router

We are going to talk about some best practices to avoid sloppy code that can sometimes be generated by these models.

Slop Cannon Suppression

Don't think of AI as a magic button that solves all your problems

Treat AI like a power tool that makes the repetitive and boring parts of your job easier so you can get more done



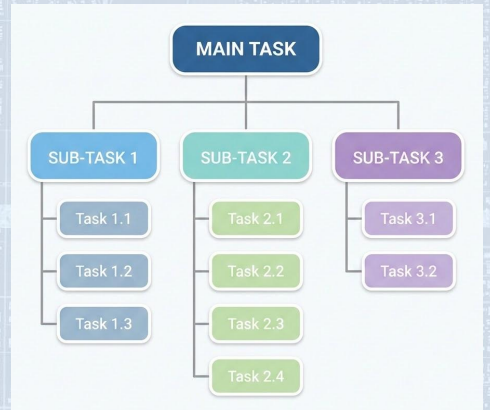
Jake

Micro Tasks

Don't ask AI to code the entire thing all at once

Break it down into micro tasks and prompt them one at a time

Read and test the results so you know how your code works



Jake

Talk about plan mode?

Plan Mode

Plan mode asks the model to do the task decomposition for you

Helpful, but you can lose sight of the program's structure and pieces



Jake

Talk about plan mode?

Code Quality Tools

Formatters
Linters
Tests



Scott

Formatters

Python

[Ruff](#)



JavaScript/TypeScript



Learn to love the defaults



Scott

For example:

- Always use 4 spaces, not 2 or not tabs
- Always use double-quotes

Best Practice:

Learn to like the default settings rather than come up with your own. Most people these days use the default settings. Your code will be easier for others including AI agents to read. Other people's code will be easier for you to read.

Only change if you have a very good reason.

Ruff

```
# Input
def _make_ssl_transport(
    rawsock, protocol, sslcontext, waiter=None,
    *, server_side=False, server_hostname=None,
    extra=None, server=None,
    ssl_handshake_timeout=None,
    call_connection_made=True):
    '''Make an SSL transport.'''
    if waiter is None:
        waiter = Future(loop=loop)

    if extra is None:
        extra = {}
    ...
```



```
# Ruff
def _make_ssl_transport(
    rawsock,
    protocol,
    sslcontext,
    waiter=None,
    *,
    server_side=False,
    server_hostname=None,
    extra=None,
    server=None,
    ssl_handshake_timeout=None,
    call_connection_made=True,
):
    """Make an SSL transport."""
    if waiter is None:
        waiter = Future(loop=loop)

    if extra is None:
        extra = {}
    ...
```

Scott

From the ruff docs website

“Code is read more than it’s written”

Differences:

- Newlines after function parameters
- Single -> double quotes

Linters

Python

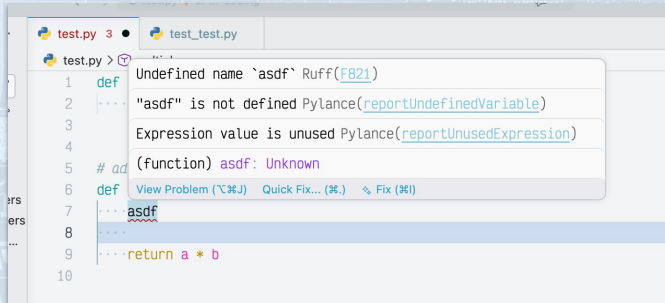


JavaScript/TypeScript



ESLint

TypeScript



```
test.py 3 • test_test.py
test.py >
1 def
2 "asdf" is not defined Pylance(reportUndefinedVariable)
3 Expression value is unused Pylance(reportUnusedExpression)
4 # ad
5 (function) asdf: Unknown
6 def
7 ... asdf
8
9 ... return a * b
10
```

Learn to love the defaults



Scott

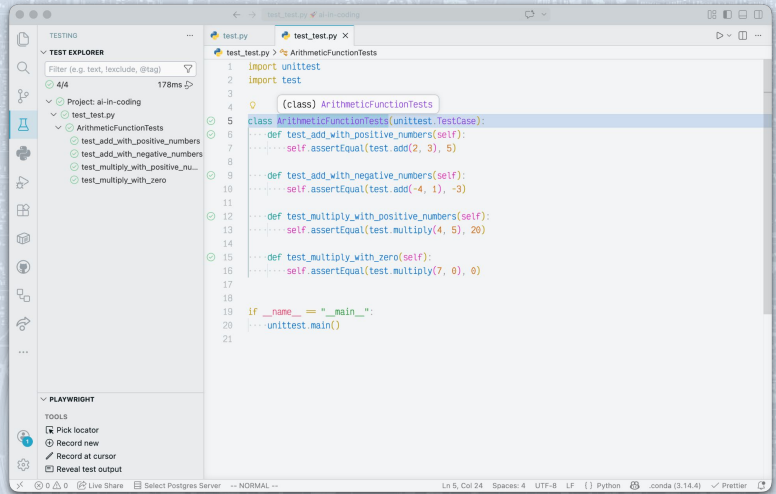
Catch common errors and enforce best practices.

Agents can run these tools and iterate until all checks are passing

Wire them up in GitHub Actions

Tests

- Unit
- Integration
- End-to-end



```
1 import unittest
2 import test
3
4
5 class ArithmeticFunctionTests(unittest.TestCase):
6     def test_add_with_positive_numbers(self):
7         self.assertEqual(test.add(2, 3), 5)
8
9     def test_add_with_negative_numbers(self):
10        self.assertEqual(test.add(-4, 1), -3)
11
12    def test_multiply_with_positive_numbers(self):
13        self.assertEqual(test.multiply(4, 5), 20)
14
15    def test_multiply_with_zero(self):
16        self.assertEqual(test.multiply(7, 0), 0)
17
18
19 if __name__ == "__main__":
20     unittest.main()
21
```

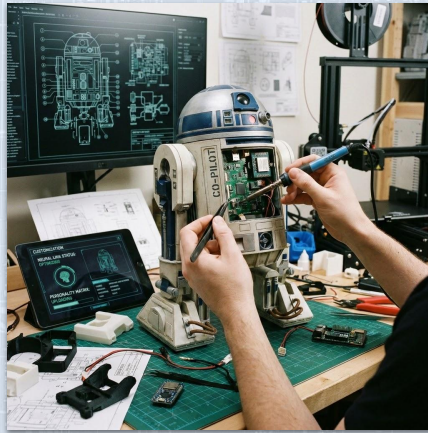


Jake

Refactoring: Don't just stop when it's working



Customizing Copilot



Scott
Repetitive coding task

Instructions Files

- Always-on
- General preferences
- Included in every request

```
copilot-instructions.md X
github > copilot-instructions.md > # Electrofishing Copilot Instructions > ## Testing > ### E2E Tests (Playwr
1 # Electrofishing Copilot Instructions
2
3 ## Project Overview
4
5 + React application for recording electrofishing surveys for Utan DWR. Uses Firebase
+ backend for geospatial data.
6
7 ## Architecture
8
9 ### Data Flow
10 1. **Frontend** → Firebase Cloud Functions ('functions/index.js') → ArcGIS Server
11 2. The proxy in 'functions/index.js' uses '@ugrc/firebase-auth-arcgis-server-proxy'
12 3. All ArcGIS endpoints are defined in 'src/config.js' under 'urls'
13
14 ## State Management
15 - **App-level state**: 'AppContext' in 'src/App.jsx' using 'useImmerReducer' - han
16 - **Event data**: 'SamplingEventContext' in 'src/hooks/samplingEventContext.jsx' -
fish, habitat, etc.)
17 - Both contexts use Immer for immutable state updates
18
19 ## Key Configuration
20 - 'src/config.js': Central config with field names, table names, URLs, and constan
21 - Field names map to ArcGIS feature service fields (e.g., 'config.fieldNames.fish
22 - Table names match ArcGIS tables: 'SamplingEvents', 'Fish', 'Equipment', 'Habitat
23
```



Scott

Always make sure that formatting, linting and tests pass

Preference for certain libraries

Prompt Files

- Reusable tasks
- Invoked via slash command

```
PROMPT.md 2 X
prompts > ugrc-accessibility > PROMPT.md > # UGRC Accessibility Checklist
1 ---
2 name: ugrc-accessibility
3 description: common checks and fixes for accessibility issues in ugrc projects
4 ---
5
6 # UGRC Accessibility Checklist
7
8 ## Steps
9
10 Check for the following issues:
11
12 ### Broken skip links
13
14 Our default header uses a skip link similar to this:
15
16 ```html
17 <a href="#main-content">skip to main content</a>
18 ```
19
20 Search for `id="main-content"` in the codebase. If not found, add it to the main content container, for example:
21
22 ```tsx
23 <main id="main-content" tabIndex="-1">
24 | ← main content here →
25 </main>
26 ```
27
28 If there is already a valid skip link and target in place, don't make any changes
```



Scott
Repetative coding task
Migration
Upgrade

Agent Skills

- Package multi-step workflow with scripts
- Invoked with the task matches the skill description

```
SKILL.md x
p1ss-review > .github > skills > app-relationships > SKILL.md > ** # App Relationships Skill > ** ## Definitions > ** ### review app
1  ---
2  name: app-relationships
3  description: Use this skill when working with the plss and plss-review apps.
4  ---
5
6  # App Relationships Skill
7
8  This skill is designed to assist in understanding the relationship between the submission app
  (plss) and the review app (plss-review).
9
10 ## Definitions
11
12 This skill is intended to be used in a workspace with two folders:
13
14 ### submission app
15
16 This is the application contained in the 'plss' folder. It is the app that users use to browse and
  submit PLSS corner submissions.
17
18 ### review app
19
20 This is the application contained in the 'plss-review' folder. It is the app that users use to
  review and approve PLSS corner submissions.
```

Next Level

- Custom Agents
- MCP Servers
- Agent Hooks



Scott
Repetative coding task

Bonus

handy

[download](#) [about](#) [docs](#)   [donate](#)



Speak into any text field

the free and open source app for
speech to text



Scott
Global hotkey

Will this replace us?

When hard parts become easy, the differentiator becomes love.

*Andy Conen,
cannoneyed.com/projects/isometric-nyc*

Resource Sheet
<https://bit.ly/4cM4hkX>



Scott

Abundance

A tool for making your programming more productive

Superpower

Lower's the bar for effort to do something.

Specialized markets like GIS make our jobs harder to replace