Summer Internship

PSci.AI

Procurement Sciences

ARTIFICIAL BIDDING INTELLIGENCE

Ray Poulton

About Procurement Sciences

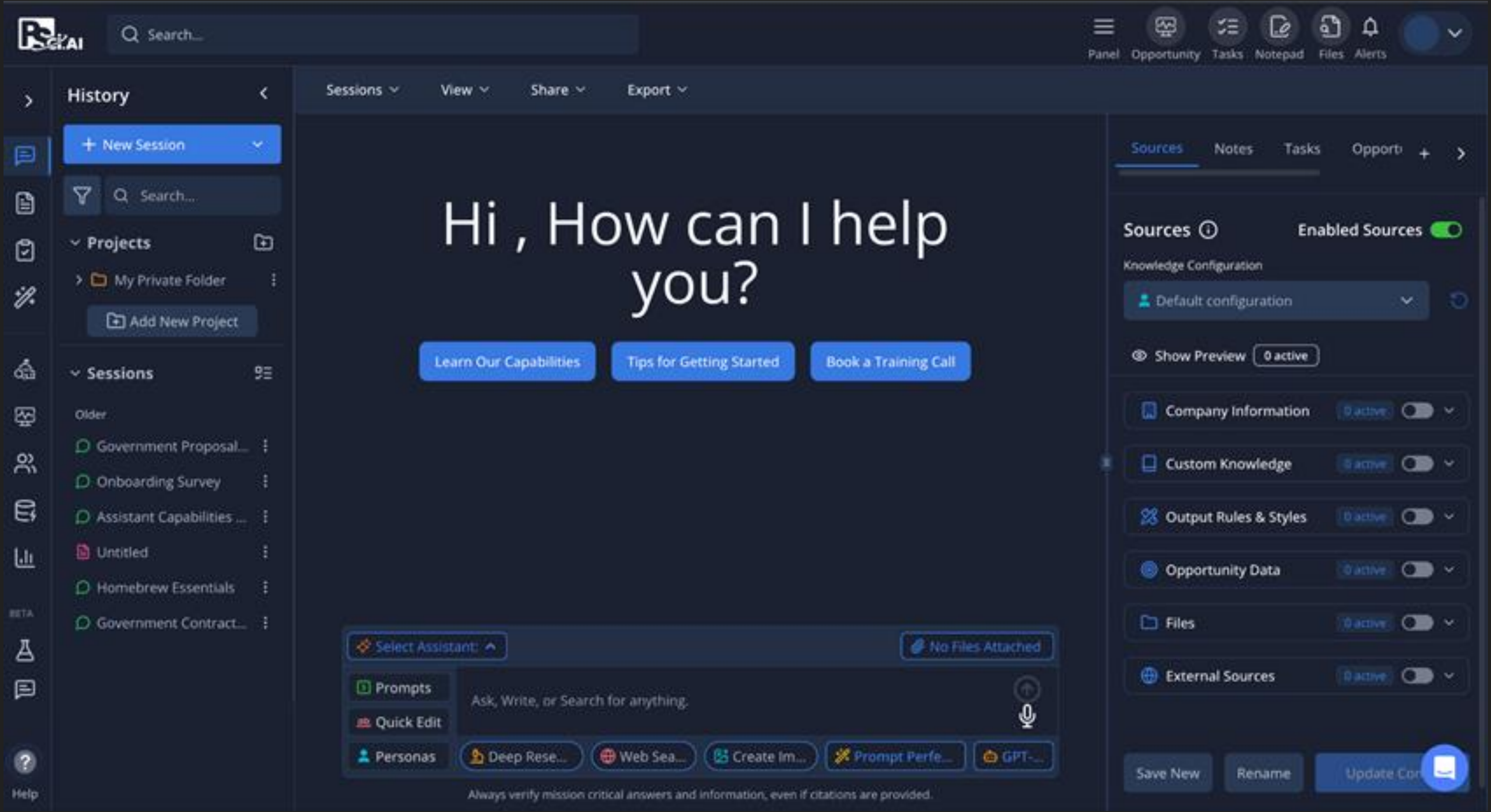
Procurement Sciences (PSci.AI) is a fast-**growing AI company** transforming government contracting through automation and **generative AI**. Their platform (**Awarded AI**) helps businesses discover opportunities, generate proposals, and manage compliance—**cutting bid time by over 90%**. Backed by **$10M in funding** and used by **top GovCon firms**, PSci.AI is redefining how companies win government work.

In other words:

Companies can use Awarded AI for…

- **Finding** the right contracts
- **Automatically** drafting proposals that meet government requirements
- **Checking** for compliance and risks
- **Learning** from past bids with AI-powered debrief tools
- **Uploading** company knowledge for focused AI knowledge

Awarded.AI - User Interface



Hi , How can I help you?

Ask, Write, or Search for anything.

Always verify mission critical answers and information, even if citations are provided.

Required Prerequisite Skills and Preparation

Programming Languages

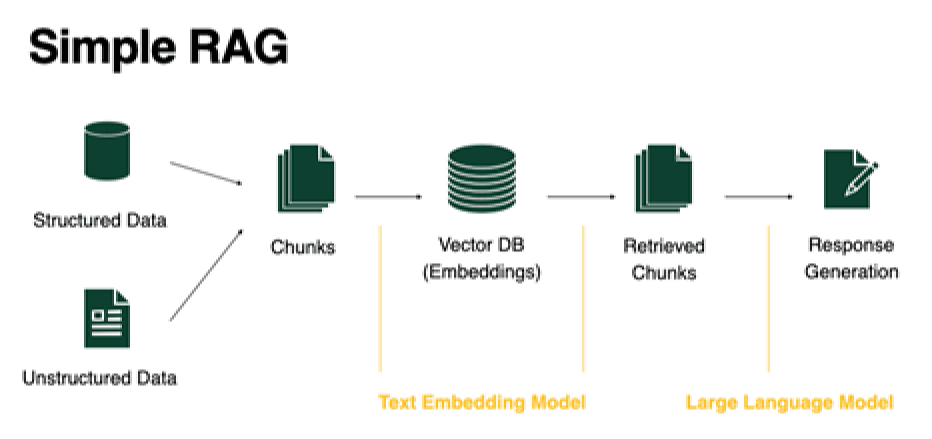
- Python
- JavaScript
- HTML
- CSS

AI Knowledge

- RAG (Retrievement-Augmented-Generation) Models
- LangChain
- Prompt Engineering

General Prior Knowledge

- Algorithms
- Backend/Frontend Dev
- Code Organization
- Cursor
- GitHub/git



My Tasks:

Project: Analyze ROI/Usage Trends

Data Migration: AWS and GovCloud

Frontend Development: Thinkific

Development Team Work: Agentic AI and Debugging

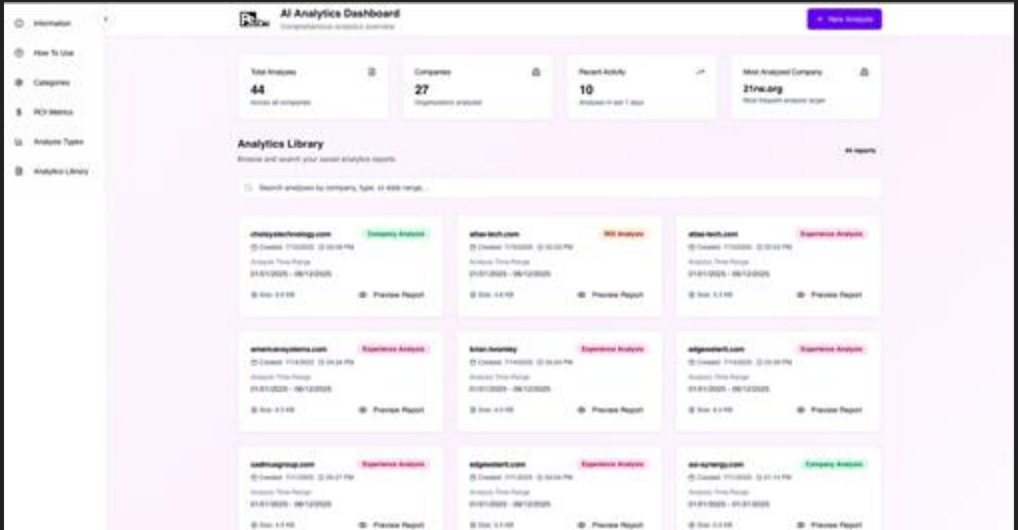
ROI/Usage AI Analytics Tool

Timeframe: March-Now

How it works: Utilizes generative AI to take in user inputs from Awarded.AI and map usage trends to GOVCON categories, time spent/time saved, dollars saved, user satisfaction, prompt patterns, and more via custom AI prompting.

Essentially a complete AI data analyst!

Powered by GPT-4o-mini from OpenAI



March - Proof of Concept

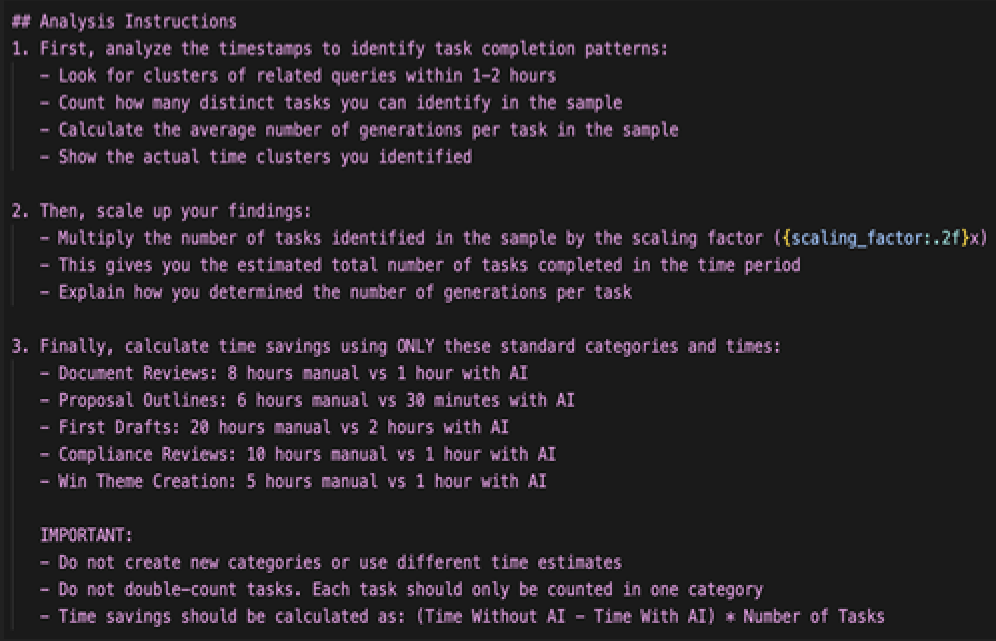
March: Got familiar with the current data collection libraries - PostHog, Langfuse, and Langsmith.

Langfuse/Langsmith = user prompt content

PostHog = clicked events/user activity

Created the first proof of concept script, which took in 1 day of user inputs via langfuse API, and fed it into OpenAI’s API key, then generated basic ROI estimates. Coded entirely in Python and output in terminal.

Techstack: Cursor, Python, OpenAI API, Langfuse API, Langchain.



```
## Analysis Instructions
1. First, analyze the timestamps to identify task completion patterns:
   - Look for clusters of related queries within 1-2 hours
   - Count how many distinct tasks you can identify in the sample
   - Calculate the average number of generations per task in the sample
   - Show the actual time clusters you identified

2. Then, scale up your findings:
   - Multiply the number of tasks identified in the sample by the scaling factor ({scaling_factor:.2f}x)
   - This gives you the estimated total number of tasks completed in the time period
   - Explain how you determined the number of generations per task

3. Finally, calculate time savings using ONLY these standard categories and times:
   - Document Reviews: 8 hours manual vs 1 hour with AI
   - Proposal Outlines: 6 hours manual vs 30 minutes with AI
   - First Drafts: 20 hours manual vs 2 hours with AI
   - Compliance Reviews: 10 hours manual vs 1 hour with AI
   - Win Theme Creation: 5 hours manual vs 1 hour with AI

   IMPORTANT:
   - Do not create new categories or use different time estimates
   - Do not double-count tasks. Each task should only be counted in one category
   - Time savings should be calculated as: (Time Without AI - Time With AI) * Number of Tasks
```

Snippet of first prompt instructions for AI analyzer

April - Established Parallel Workflow

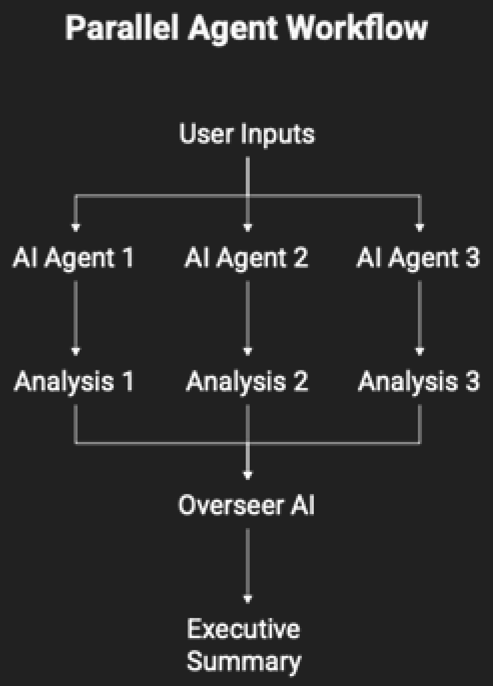
Created a multi-agent AI workflow which allows the tool to process data in parallel.

- DuckDB allows for temporary storing of user traces
- Single user’s prompts are ingested into multiple AI agents at the same time.
- Each agent has a sample of the user’s inputs and identifies usage trends within GOVCON specific categories, and creates an analysis.
- Each agent sends their analysis to an “overseer” AI that creates an executive summary on usage trends

Each agent is an instance of GPT-4o-mini - has 120k+ token limits

Proved effective for processing larger amounts of data (2 weeks at this point).

Techstack: Python, DuckDB, Marimo Notebooks, Cursor, OpenAI, Langchain, Azure Blob.



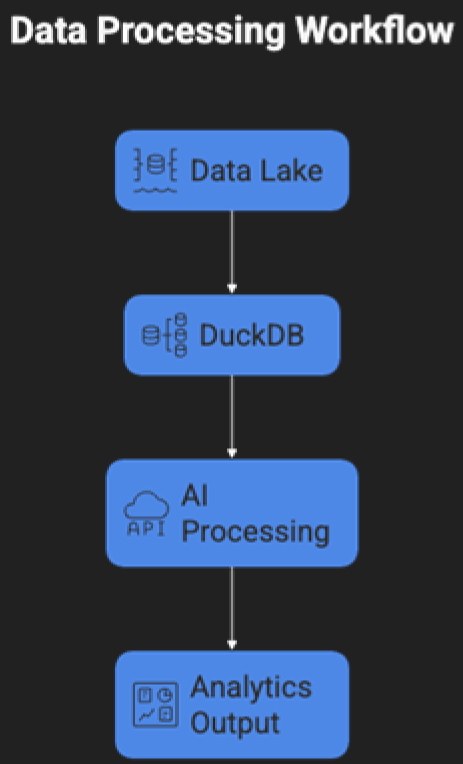
What is DuckDB?

It's a lightweight database you can run right inside your code, no server needed, and it's designed for analyzing large datasets fast, especially with tools like Python and Pandas.

It works great with files like CSV and Parquet, and lets you run SQL queries directly on your data, even without loading it into memory. It's like SQLite, but built for analytics.

Parquet format = extra speed

DuckDB allows me to pull in data from a database, and then organize it into tables locally, rather than needing to use the cloud for every query. A useful framework for feeding specific data into an LLM.



May - AWS S3 Data Lake and Company Analysis

Created the central data location for all of the Analytics Tool data. Major learning curve here!!! Needed to create a data lake workflow. Was supplied with sources and AWS access from Basit (CTO).

- Created a bronze-silver-gold workflow
- Made a glue job to process raw bronze traces, de-duplicate, convert from JSON to Parquet, sort by Year/Month/Day/, and place into Silver.
- Second glue job to process silver traces to be organizes by timestamp AND company via email domains

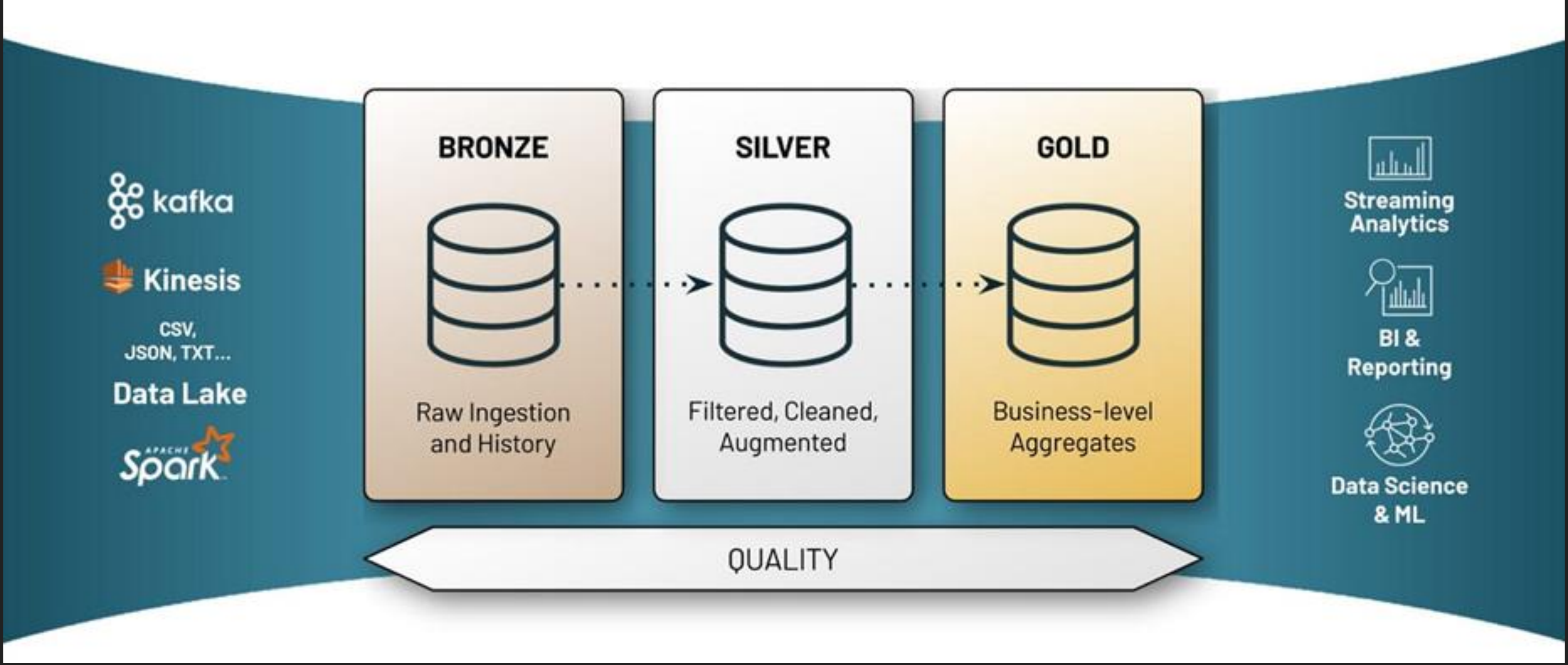
By the end of the month I was very comfortable using AWS and creating glue jobs. The current Marimo Notebook for the Analytics tool was processing months of data incredibly fast with the parallel workflow I created.

Additionally, I created AI analytics at the company level - utilize the same workflow but on a larger scale. Each user analysis is sent to a new overseer AI that generates a company level usage summary.

TechStack: AWS, Spark, Python, Boto3.

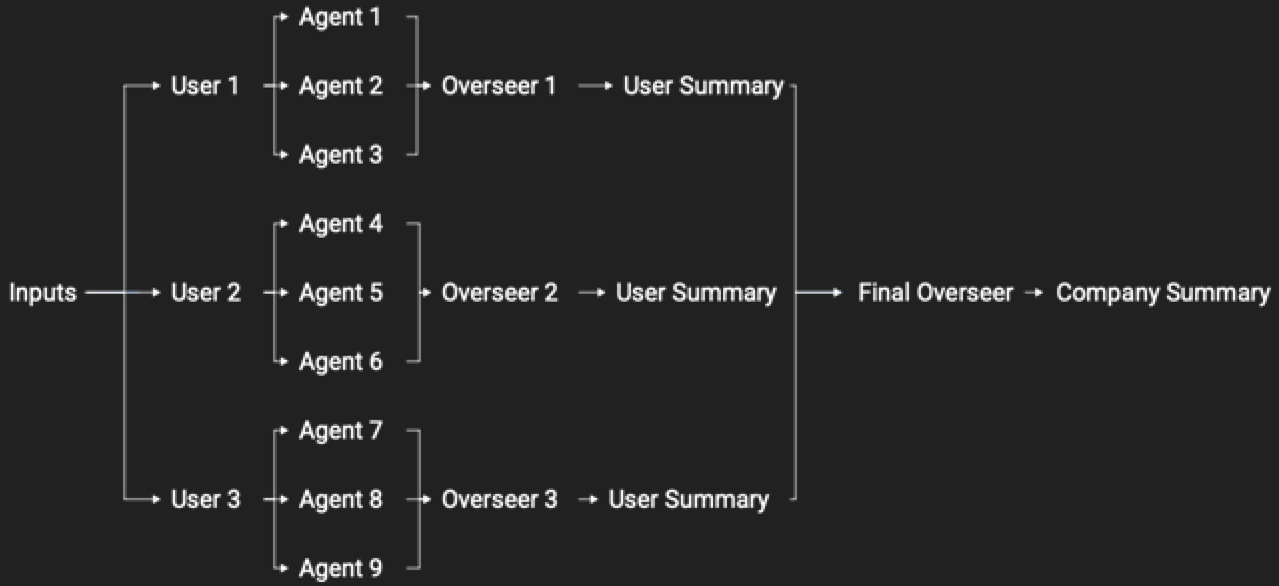
AI was a big teacher/reference for me in this new area.

S3 Data Lake Workflow



Company Analysis Workflow Graph

Parallel Workflow Graph



Inputs

User 1

Agent 1

Agent 2

Agent 3

Overseer 1

User Summary

User 2

Agent 4

Agent 5

Agent 6

Overseer 2

User Summary

User 3

Agent 7

Agent 8

Agent 9

Overseer 3

User Summary

Final Overseer

Company Summary

GOVCON Categories for AI to Map

AI categorizes user activities into these government contracting focus areas:

1: Proposal Outlines

2: Proposal 1st Drafts

3: Compliance Reviews

4: Color Team Reviews

5: Capture Support

6: Market Research

7: Customer Relationship Management

8: Solution Architecture Documentation

9: Oral Presentation Preparation

10: Presentation Creation

11: Win Theme Creation

12: Solution Design/Prototyping

13: System Design Documentation

14: Other

These are the primary govcon activities for the users of Awarded AI.

If the AI agents determine users were not engaged with these, it will categorize usage as “Other”

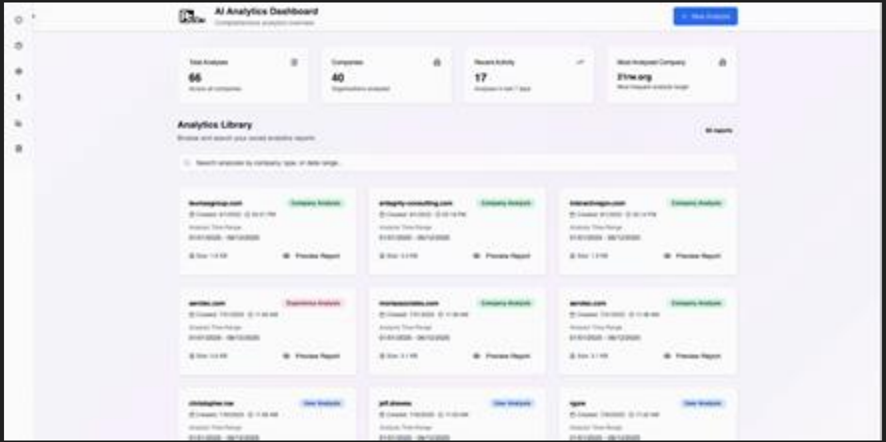
June - Putting the Pieces Together

Created the front end for the analytics tool and exported all of the user traces from langfuse

- Created the frontend using Replit Agent, then added the python backend
- Exported the rest of the traces into Bronze
- Processed the trace data into my silver and gold glue jobs
- Save analytics into the database, for re-accessible insights

The AI analytics tool could now process 6 months of data from January 2025 - June 2025 in less than 2 minutes - thanks to the efficiency of the parallel agent workflow and fast DuckDB processing.

Techstack: Python, Javascript, HTML, CSS, AWS, Replit Agent, DuckDB, PostgreSQL



Example AI Company Analysis Output

Company Usage Trends

Individual User Analysis

- **bob@company.com:** bob@company.com shows a high level of engagement with government contracting activities, focusing on **Proposal Outlines**, **Compliance Reviews**, and **Capture Support**. Their proficiency in cybersecurity and adherence to federal guidelines is notable, positioning them strategically in securing government contracts. They are encouraged to streamline documentation processes and foster stakeholder collaboration to enhance workflow effectiveness.
- **joe@company.com:** joe@company.com displays a focused commitment to **proposal development** within complex IT projects, particularly emphasizing **compliance** and **solution architecture**. Their methodical approach highlights a strategic mindset that prioritizes quality in their submissions. Developing standardized templates could enhance their efficiency, alongside further collaboration with cross-functional teams.

- bob@company.com: 805 prompts

- joe@company.com: 63 prompts

Note: This is just an example, the actual responses are much more thorough, but I am not allowed to share.

Security Measures Taken

In order to ensure the security of the customer's prompts, I created a system which…

- Maps the user's emails to anonymous ids. Ex: User_1, User_2, User_3, etc.
- Maps the actual user emails back to the final report, in place of their anonymous ids.

So, the AI never sees the user's real emails, but it can still associate different user's prompts.

The AI is given strict instructions to use the exact ids like User_1, to allow for proper mapping.

This feature was requested by the Head of Security.

We also use a secure OpenAI API which is NOT trained on, so the actual content going into the AI is secure.

June - More Analysis Options

Now that I could map usage trends to entire companies, I was able to add the following analysis options:

- ROI (using GOVCON salary averages and internal information on estimates ROI per task)
- Top 10 Users (Analyzes the top 10 most active users and finds prompt patterns with the AI usage analysis)
- User Experience (AI is prompted to handle user sentiment/satisfaction levels, rather than usage)

ROI Metrics:

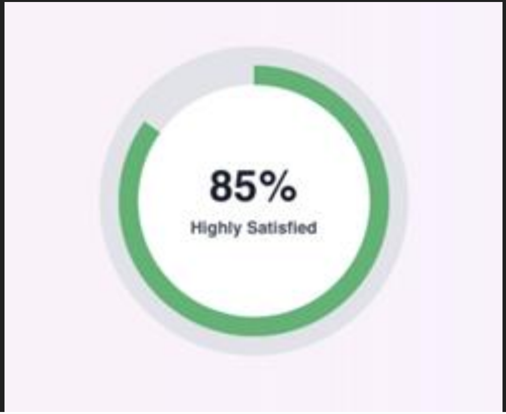
Task Time Savings (With AI vs Without AI)

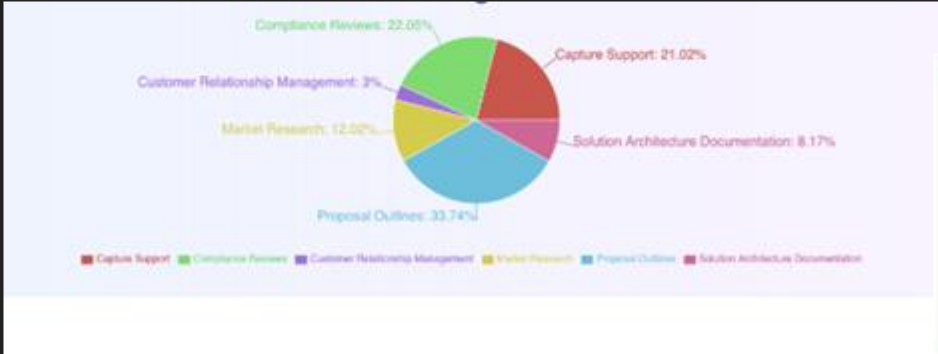
• Proposal Outlines: 30 min vs 6 hrs (5.5 hrs saved)
• Proposal 1st Drafts: 2 hrs vs 20 hrs (18 hrs saved)
• Compliance Reviews: 1 hr vs 10 hrs (9 hrs saved)
• Win Theme Creation: 1 hr vs 5 hrs (4 hrs saved)
• Market Analysis: 2 hrs vs 15 hrs (13 hrs saved)
• Color Team Reviews: 1 hr vs 8 hrs (7 hrs saved)
• Market Research: 2 hrs vs 15 hrs (13 hrs saved)
• Oral Presentation Prep: 2 hrs vs 8 hrs (6 hrs saved)
• Presentation Creation: 3 hrs vs 10 hrs (7 hrs saved)
• Solution Prototyping: 4 hrs vs 20 hrs (16 hrs saved)
• System Design Documentation: 2 hrs vs 12 hrs (10 hrs saved)

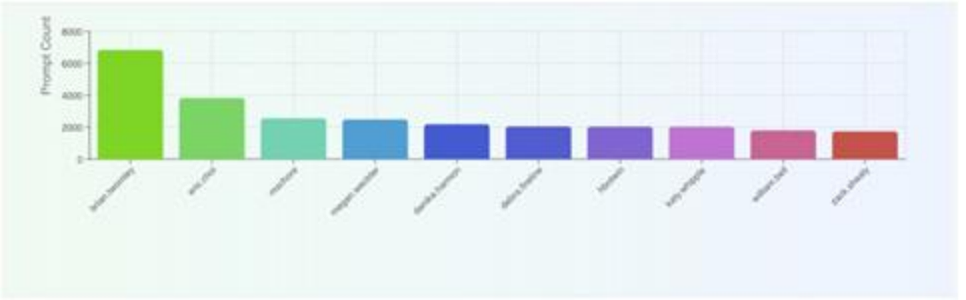
<u>Cost Calculations</u>

• Standard hourly rate: $75/hour
• Cost savings = Time saved × $75/hour
• Example: Proposal 1st Draft saves 18 hrs = $1,350

AI Generated Usage Graph







AI outputs its percentage data in JSON format, and my system processes it into an interactive chart

July - Migrating Data to GovCloud

Because I was processing user prompts to Awarded AI, there is loads of CUI in this database. So, I migrated to a secure AWS environment called GovCloud, which can hold sensitive government contracting information.

Why?

Many users deal with contracts involving:

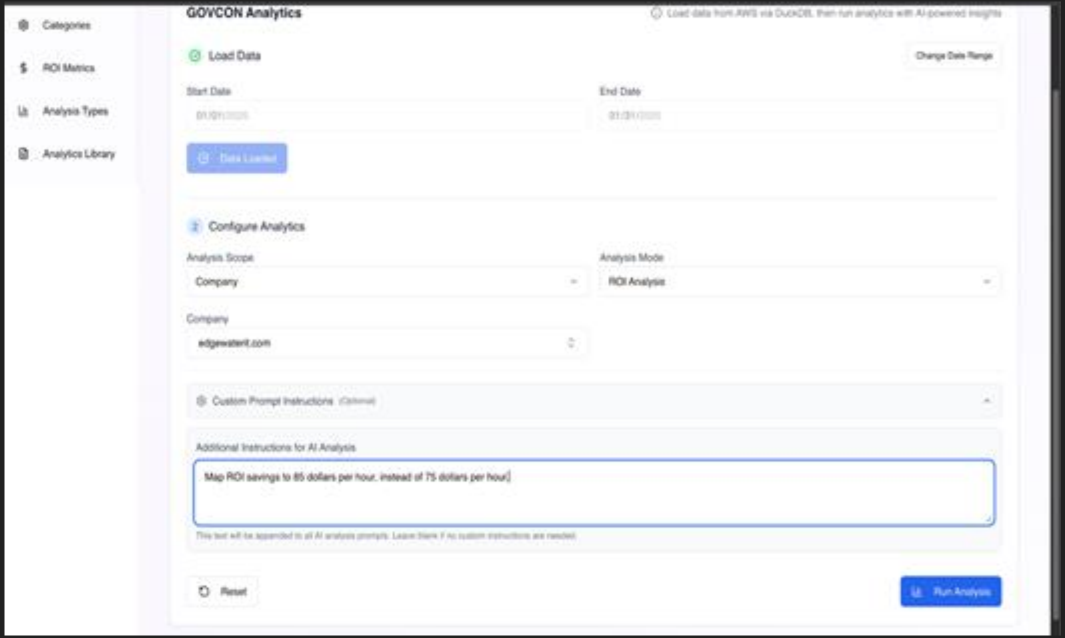
- Dept of Defense
- NASA
- Military Branches
- Other contract areas

This also made the tool processing with DuckDB MUCH faster, as I spent much of this month testing the tool as well.

Now - Getting Updated Data and Hosting

The next steps for the analytics tool…

- Hosting the tool for the Customer Success Managers (CSMs) and Product Managers (PMs) to use
- Getting updated prompt data using our new data collection tool, Langsmith



GOVCON Analytics

Load data from AWS via DuckDB, then run analytics with AI-powered insights

Load Data

Start Date: 01/01/2025

End Date: 01/31/2025

2 Configure Analytics

Analysis Scope: Company

Analysis Mode: ROI Analysis

Company: edgewaterit.com

Custom Prompt Instructions (Optional)

Additional Instructions for AI Analysis

Map ROI savings to 85 dollars per hour, instead of 75 dollars per hour

This text will be appended to all AI analysis prompts. Leave blank if no custom instructions are needed.

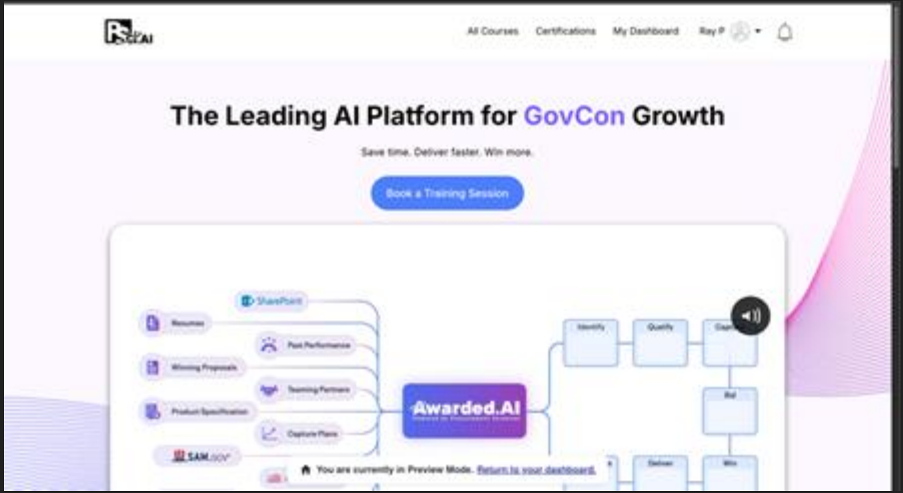
Other Tasks - Thinkific Site Frontend Development

I developed the user interface for the training site that teaches customers how to use Awarded AI, which just released 2 weeks ago with 300+ current users.

This was a challenging task…

- Search through a 100+ file codebase in Thinkific's native site builder (no AI tool help).
- Implement designs from Figma for the first time
- I lacked any real world frontend experience before this

https://academy.awardedai.com/



The Leading AI Platform for GovCon Growth

Save time. Deliver faster. Win more.

You are currently in Preview Mode. Return to your dashboard.

Other Tasks - Data Collection for CSMs

Implemented my bronze-silver-gold AWS workflow for organizing data from different services (Intercom, Salesforce, PostHog, Jira) into one location.

Worked with Kierin Flanagan (co-worker) for organizing this data. We worked as a team to combine his knowledge of all of the current data collections/tools, with my AWS/data workflows.

Placed all this data into a new tool called Planhat, which is an internal resource that the entire CSM/PM (Customer Success/Product Managers) team will use for analytics.

Kierin and I attended multiple data calls with the employees from Planhat to figure out implementation with our data.

I set the glue jobs on an automated timer, to collect data nightly.

Current/Future Tasks - Dev Team Work, **Agent Chat**

Past 3 weeks, and moving forward: working within a development team.

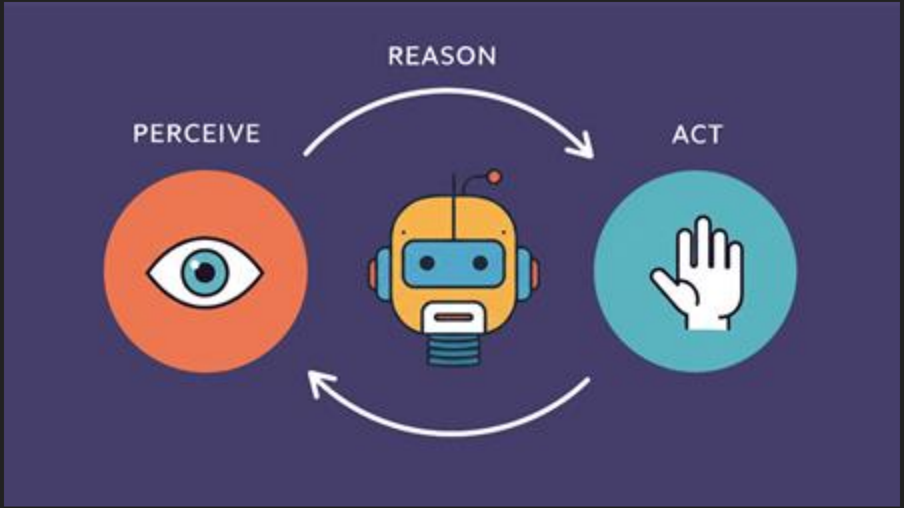
Tasks include:

- **Agent Chat** Development
- Debugging across the app (Awarded AI)

Agent Chat is Awarded AI's upcoming chat page that allows you to interact with a personal AI Copilot. This is a step forward to bringing in more agentic AI to carry out tasks, using the customer's data.

So far I've implemented:

1. Microphone chat input
2. AI Generated titles (Similar to ChatGPT's chat titles)
3. Stop generation button
4. Bid search bug fixes (AI analyzes/summarizes bid opportunities - which takes a person hours)



Takeaways

- AI will not replace us, it’s a “**force multiplier**”
- Learned how to create AI workflows
- Gained experience with full-stack development
- Learned AWS
- Picked up data engineering skills
- Improved prompt engineering skills
- Deployed AI applications to improve the job of CSMs/PMs
- Gained so many connections
- Gained real-world software development, AI, data science, and debugging experience
- **Get to CONTINUE part time!!**