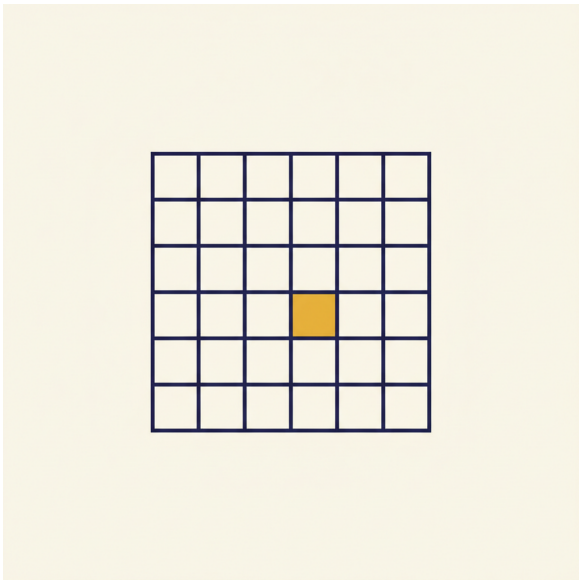
TIER 3 - EMPLOYABLE * V1.0 -- MAY 2026

DSHEETS -- THE UNDER-THE-RADAR P

Google Sheets + Excel AI integrations. Apps Script + AI for the next level. Cell-level formulas that pull AI inline. The 100 vs 10,000-row threshold where this approach hits its ceiling -- and what comes after.



BY Alex Jahn / Agent Logic

v1.0 -- May 2026 **EDITION**

Anyone who lives in spreadsheets -- analysts, ops folks, small-business owners, finance, anyone with a Google Sheet open right now -- who hasn't figured out that AI inside the Sheet is the highest-leverage AI move available **AUDIENCE**

15-20 minutes **READ TIME**

Free. Forever. **COST**

Prepared by Agent Logic / alexanderjahn79@icloud.com / theaiguywi.com

CONTENTS

What's in here

SECTION 1

The most-overlooked AI productivity hack

"I use AI" but "I don't use it in Excel" is the gap

Most working professionals' AI use happens in chat windows -- ChatGPT, Claude, the standalone apps. Meanwhile, the same people spend half their workday in spreadsheets. AI in the chat window helps them think; AI in the SHEET would do their actual work.

The gap is real. AI inside a spreadsheet -- operating on the data in the cells, producing outputs in adjacent cells, scaling across hundreds of rows -- is one of the highest-leverage moves you can make. Most people haven't done it because they didn't know it was possible without writing code, or they tried it once with a janky add-on and gave up.

This module is the practical version. What works, what doesn't, the threshold where this approach maxes out, and what comes after.

What you'll have by page 12

By the end of this primer:

- The **three app**
- The **cell-level**
- The **bulk-proc**
- The **100 vs 10**
- **Three worked sessions** -- review sentiment, lead enrichment, content categorization.
- The **honest li**

> ***AI in the chat window helps you think. AI in the spreadsheet does your actual work. The gap between those two is where most professionals lose hours per week.***

SECTION 2

The three approaches

Approach 1 -- Built-in AI features (lowest effort)

Google Sheets has built-in Gemini AI features (rolling out steadily) -- type `@Gemini` in a cell or use the side-panel Gemini assistant. **Excel** has ($30/mo on top of M365).

What it does: generates content per cell or per range, summarizes data, suggests formulas, classifies entries.

Best for: quick ad-hoc tasks, classification, content generation against a small number of rows.

Limits: doesn't scale well past 100ish rows; per-cell latency adds up; output quality varies; not always customizable.

Approach 2 -- Add-ons / extensions (medium effort)

Google Sheets: GPT for Sheets, SheetAI, ChatGPT extension. **Excel**: GP add-in.

What they do: add custom functions like `=GPT("prompt", A2)` you can drop into cells. You control the prompt; the function calls OpenAI's or Claude's API behind the scenes (with your API key or the add-on's bundled credits).

Best for: repeated AI operations across many rows. The cell-formula pattern.

Limits: add-ons cost money or use credits. Some are buggy. Privacy depends on the vendor's data handling. Performance scales poorly past ~1000 rows in a single sheet.

Approach 3 -- Apps Script + API (light-code, most powerful)

Google Sheets only. Write a small Apps Script function that calls the Anthropic, OpenAI, or Gemini API. Use it as a custom function in cells, or run it on a button click.

What it does: full control. Your prompt, your model, your API key, your error handling. Scales to thousands of rows.

Best for: production workflows where you'll run the same operation regularly across larger datasets, or where the data is sensitive enough that you want direct API control.

Lift: ~30 lines of Apps Script for the first version. Built once, runs forever.

One-line picker:

- **Ad-hoc, 1-50 rows** -> Built-in (Gemini in Sheets, Copilot in Excel)
- **Repeated, 50-500 rows** -> Add-on (GPT for Sheets etc.)
- **Production, 500+ rows or sensitive data** -> Apps Script + API

You can use all three at once. They're not mutually exclusive.

SECTION 3

The cell-level formula pattern

The pattern that does the work

Once you have an AI function in your spreadsheet (built-in, add-on, or Apps Script), the basic pattern is:

``` A column: input data (review text, lead description, raw text, whatever) B column: =AI("Classify this review as positive, negative, or mixed: " & A2) ```

Drag B2 down across 500 rows. The AI function runs for each row. 10 minutes later, every row in column B has its AI-generated output.

This trivially-simple pattern is what most people miss. They keep thinking AI work happens in the chat window. The shift is to realize the spreadsheet is the chat window -- once you have the right function.

Common cell-formula patterns

- `=AI("Summarize in one sentence: " & A2)` -- summarization
- `=AI("Extract the company name from this email: " & A2)` -- extraction
- `=AI("Classify into one of these categories [list]: " & A2)` -- classification
- `=AI("Rewrite this to be professional but warm: " & A2)` -- rewriting
- `=AI("Score this review 1-5 for clarity: " & A2)` -- scoring
- `=AI("Translate to Spanish: " & A2)` -- translation

Each one solves a real workflow you used to do manually. Each one runs against 100 rows in minutes instead of hours.
```

SECTION 4

The bulk-processing workflow

Beyond the single formula -- running a pipeline

For larger jobs, the workflow gets a bit more structured:

> **The bulk-processing workflow:**

1. **Stage data** in a clean sheet. Column A = input. Leave columns B, C, D etc. for AI outputs.
2. **Write the prompt** in a separate cell (say A1) where you can edit it once and reference it from every row.
3. **Cell formula references the prompt:** `=AI($A$1 & A2)` -- `$A$1` is the prompt template, `A2` is the row's input.
4. **Drag down** across all rows. Watch progress.
5. **For very large jobs** (1000+ rows), break into batches. Run 200 rows at a time. Saves you from rate-limit errors and gives you checkpoints.
6. **Review a sample** of outputs after each batch. If quality is off, edit the prompt in A1, re-run the batch.
7. **When done**, copy values (paste-special) so the outputs become static text. Otherwise re-running formulas re-runs the API calls (and bills).

This pattern handles most production-spreadsheet AI work cleanly. The "copy values when done" step is critical -- without it, every spreadsheet open re-fires the API calls.

SECTION 5

Three worked sessions

Worked session 1 -- Customer review sentiment

You've got 300 customer reviews in column A. You want sentiment classification + theme extraction.

Prompt in A1: *"Classify t*
Mixed. Then identify the single most important theme the reviewer raised (e.g., 'response time', 'product quality', 'pricing'). Reply in this exact format: SENTIMENT | THEME."

Cell formula in B2: `=AI($A$1 & A2)`

Drag down 300 rows. Total cost: a few dollars depending on review length. Total time: ~15 minutes including review.

You now have sentiment + theme on every review, can pivot the sheet to count themes, surface the negative-review themes for action. Was this previously a half-day manual coding task? Probably.

Worked session 2 -- Lead enrichment

You've got 200 inbound leads. Each has a name, company name, email. You want to enrich with: industry, company size estimate, role inference, fit score.

Prompt in A1: *"Given thi*
size category (1-10, 11-50, 51-200, 201+), (3) likely role of the contact based on email pattern, (4) fit score 1-5 for a small-business AI consulting service. Reply in JSON: industry | size | role | score."

Cell formula in B2: `=AI($A$1 & "Company: " & B2 & " Email: " & C2)`

Drag down. 200 leads enriched in 15 minutes. Cost: a few dollars.

Honesty caveat: the inferences from AI on small companies are often wrong (it'll guess "tech" for a name that could be anything). Verify the fit-score column before acting on it; treat AI's enrichment as a starting point, not gospel.

Worked session 3 -- Content categorization

You're moving 500 old blog posts into a new CMS and need each one categorized. Manual categorization would take a full day.

Prompt: *"Below is*
categories]. If none fit, reply 'Uncategorized'. Reply with just the category name."

Formula: `=AI($A$1 & "Title: " & A2 & " Excerpt: " & B2)`

Drag down 500 rows. 25 minutes of processing. Review the "Uncategorized" outputs manually (probably 30-50 of them). Done.

> ***The work that used to be a full-day manual task becomes a 30-minute spreadsheet job. Compound across your year of these tasks and the time savings are obvious.***

SECTION 6

The 100 vs 10,000-row threshold

When spreadsheet AI hits its ceiling

Spreadsheet AI works dramatically well up to ~1000 rows. It works tolerably from 1000-5000. Past 5000, you start hitting walls:

- **Rate limits** -- APIs throttle when you fire 5000 calls in a few minutes
- **Performance** -- Google Sheets gets sluggish with 5000+ formula cells doing API calls
- **Error handling** -- one failed cell breaks the chain; debugging across 5000 rows is painful
- **Cost** -- at 10,000 rows × a few cents per call, you're at real-money territory
- **State management** -- re-running becomes risky (resetting all 10,000 cells fires all 10,000 calls again)

At 10,000+ rows, the right tool is no longer a spreadsheet. It's a Python script or a real data pipeline. The structure is the same (input -> prompt -> output) but the platform changes.

> **The threshold rule:**

- **Under 1000 rows** -> spreadsheet AI is the right tool. Period.
- **1000-5000 rows** -> spreadsheet AI works, but you'll fight rate limits + performance. Worth it if it's a one-time job; not worth it if it's recurring.
- **5000+ rows** -> graduate to a Python script. Same API; better tooling.

Most "I need to migrate from spreadsheet to script" decisions get made too late. Recognize the threshold; move when you hit it.

SECTION 7

The honest limit

Three categories where spreadsheet AI is the wrong tool:

- **Real-time / streaming.** Spreadsheets are batch. If you need AI processing as data arrives, you need a script or service, not a sheet.
- **Anything with privacy red lines.** Sending customer data through a free add-on whose privacy policy you haven't read is dangerous. For sensitive data, use Apps Script + direct API call (so YOU control the data flow) or skip the spreadsheet approach.
- **Anything where output quality matters more than speed.** Spreadsheet AI is fast and "good enough." For high-stakes outputs (legal documents, contracts, regulatory filings), the speed isn't worth the quality variance. Use careful AI workflows with eval steps, not bulk processing.

Within those limits, spreadsheet AI is one of the most-underused productivity tools available to working professionals. Most people who use AI heavily in chat haven't tried it. The unlock is one cell formula.

SECTION 8

Where to go from here

Eight more Tier 3 modules ahead:

- **Internal AI tools without being a developer** -- when you outgrow spreadsheets, what comes next.

Get the next module the day it drops: theaiguywi.com/training

If you want this spreadsheet-AI pattern installed in your actual workflows -- Apps Script written for your specific use case, the right model picked, the right prompt structure -- that's the consulting offer.

Reach out: alexanderjahn79@icloud.com

Closing -- the lock-in line

The chat window helps you think. The spreadsheet does the work. The bridge is one cell formula. Most professionals have been doing AI work the wrong way for two years because they never made the jump. The jump is shorter than it looks. The leverage is bigger than you'd guess.

1000

One thousand rows

is roughly the spreadsheet-AI sweet spot. Below that, this is the highest-leverage AI move you can make. Above that, you're graduating to scripts -- which is the next module's territory.

-- Alex

Agent Logic -- *theaiguyw*

Lac, WI. This is module 10 of 18 in Tier 3 (Employable).