AI in digital marketing -- what works and what does not

Eight use cases, what to verify before anything ships, and the one task with no working version.

> *Written from buying and testing a large number of AI products. The verify column is the important one: the output reads identically whether it is right or wrong, which is the whole problem.*

Five rules

- Give it your data. A model working from your export is useful; one working from memory invents.
- Anything checkable, check. The confident tone is identical whether it is right or wrong.
- Never let it produce a statistic, a price or a citation that reaches a client unverified.
- Use it to go from nothing to a draft, not from a draft to published.
- If you cannot tell whether the output is right, you are not qualified to use it for that task yet.

Where it works, and where it does not

Task	Works for	Fails at	Saved
First-draft ad copy and subject lines	Producing twenty variants to test, from a brief you wrote.	Publishing any of them unedited, or letting it invent a product claim.	High
Clustering keyword exports	Grouping a 3,000-row export into intent clusters far faster than by hand.	Trusting its volume or difficulty figures -- it does not have them.	High
Summarising long reports or transcripts	Turning a 40-minute call into decisions and owners.	Treating the summary as the record. It drops the caveats first.	High
Writing formulas and regex	Spreadsheet formulas, GA4 regex, tracking rules -- where the output is testable.	Deploying a regex to production filters without testing it on real data.	High
Competitor and SERP research	Reading twenty pages and telling you what claim they all make.	Asking it who ranks for a term. It cannot see live rankings.	Medium
Alt text and metadata at volume	Draft titles and descriptions across hundreds of pages, from real page content.	Publishing without a human pass. It will write near-duplicates.	Medium
Analysing your own performance data	Spotting a pattern in an export you would have missed.	Asking why something happened. It will produce a plausible cause with no evidence.	Medium
Publishing whole articles unedited	Nothing. There is no version of this that works.	All of it. It is the fastest way to fill a site with content nobody cites.	None -- a net cost

What to verify, task by task

The single habit that separates useful AI work from expensive AI work: decide, before you start, how you will check the output. If you cannot answer that, do not use it for that task yet.

Task	How to check it
First-draft ad copy and subject lines	Check every factual claim against the product page. Models invent features confidently.
Clustering keyword exports	Spot-check 20 rows against the export. Re-read the clusters for merged intents.
Summarising long reports or transcripts	Keep the source. Check any number that made it into the summary.
Writing formulas and regex	Run it against a sample where you already know the answer.
Competitor and SERP research	Give it the pages. Never let it recall them from memory.
Alt text and metadata at volume	Sort the output and scan for repeated phrasing before publishing.
Analysing your own performance data	Treat every explanation as a hypothesis to check, never a finding.

The one with no working version

Publishing whole articles unedited. There is no configuration of this that produces something worth citing -- it produces pages that read as filler, earn nothing, and dilute the pages on your site that are genuinely good. The time saved is a net cost.

Five worked prompts

Each one gives the model your data and asks for something you can check. None asks it to recall a fact, which is where these tools fail.

Cluster a keyword export

The strongest use on the list. You supply the data, and the output is checkable row by row.

> Here is a CSV export of 800 keywords with volume and difficulty.
>
> Group them into topic clusters where a single page could satisfy every keyword in the cluster without compromise. For each cluster give me:
> - a cluster name
> - the keyword I would target as the primary
> - the total volume across the cluster
> - the modifiers present (template, example, checklist, pdf, software, meaning)
>
> Rules: do not invent keywords that are not in my data. Do not estimate volume -- sum only the numbers I gave you. If a keyword does not fit a cluster, put it in "unclustered" rather than forcing it.

What you get: Clusters that are roughly 80% right and save several hours of sorting.

Check before it ships: Spot-check 20 rows against the export, and re-read each cluster for two intents that got merged -- that is the failure mode, and it is invisible unless you look.

Draft ad copy variants

Volume of options is genuinely useful. Judgement about which to run is not delegated.

> Product: [one paragraph, pasted from the product page -- not from memory]
> Audience: [who, and the trigger that makes them search]
> The one thing we want to communicate: [single proposition]
> Banned claims: anything about speed, "leading", or numbers I have not given you.
>
> Write 15 headlines of 30 characters or fewer and 4 descriptions of 90 characters or fewer. After each headline, note which part of the product paragraph it is based on.

What you get: Fifteen usable starting points, of which perhaps four are worth testing.

Check before it ships: The "which part it is based on" line is the whole trick -- it makes an invented feature obvious instead of plausible. Check every claim against the product page anyway.

Turn a call into decisions

Removes the worst half-hour of the week. The source is still there to check against.

> Here is the transcript of a 40-minute client call.
>
> Produce:

1. Decisions made, with who made each one
2. Actions, with an owner and a date where one was stated
3. Anything raised and NOT resolved
4. Any number, date or commitment mentioned -- quoted exactly

Do not summarise the discussion. If an owner or date was not stated, write "not stated" rather than inferring one.

What you get: A usable set of notes in under a minute.

Check before it ships: Section 3 is the one that earns its place -- unresolved items are what a summary drops first. Verify every figure in section 4 against the transcript.

Write a formula or a regex

Testable output. You find out immediately whether it is right, which is rare.

In Google Sheets, column B is sessions and column C is conversions.

Write a formula for conversion rate that returns an empty cell rather than an error when sessions is zero or blank, formatted as a percentage. Then give me three test rows -- including the zero case -- with the answer you expect for each, so I can check it.

What you get: A correct formula more often than not, and a way to tell when it is not.

Check before it ships: Run the test rows it gave you. If the expected answers are wrong, the formula usually is too -- and asking for the tests costs nothing.

Compare competitor pages you supply

Reading twenty pages is slow. Recalling them is impossible, so never ask it to.

I am pasting the full text of 6 competitor pages that rank for the same query.

For each page: the promise in its first 100 words, whether it offers a downloadable asset, and the specific questions it answers.

Then: which questions are answered by fewer than two of the six pages?

Use only the text I pasted. Do not comment on rankings, domain authority, or anything you cannot see in the text.

What you get: A gap list that is genuinely useful for deciding what to write.

Check before it ships: The last instruction matters most: without it, the answer drifts into invented ranking commentary. It cannot see live rankings.

Weak prompt vs strong prompt

Weak: Write a blog post about digital marketing reporting.

Nothing to check against, so the model fills the gap from memory. The output is generic, unverifiable, and reads as filler -- the failure mode with no working version.

Strong: Here is our report template and three client reports. Which questions do clients ask that these reports do not answer? Use only these documents.

Answerable from data you supplied, checkable against it, and the output is a decision about what to build rather than prose to publish.

Score your own workflow

Priority = hours a month x how repetitive x how checkable the output is. Verifiability belongs in the formula: an unverifiable task is where AI does damage, however repetitive it is.

Task	Hours a month	Repetitive (1-5)	Checkable (1-5)	Decision