

BREWING DISTILLING + AI

The Brewing + AI Guidebook

A practical, plain-language guide for brewers and distillers on using AI thoughtfully: clear goals, clean data, a realistic mental model, real verification, and a hand on the control.

Six chapters · Built by Ankur Napa · ankurnapa.github.io/brewing-distilling-ai

Brewing Goals + Defining Quality

It is tempting to start with the technology: which app, which model, which dashboard. That is the wrong end to start from. AI is only useful once you can say, in plain words, what you are trying to achieve and how you will know if you got there.

Brewers already do this instinctively at the kettle. You have a picture of the finished beer before the mill ever turns. This chapter is about making that picture explicit enough that a tool (or a new brewer, or your future self) could act on it.

Start with the beer, not the tool

A model does not know your house style. It has never tasted your water, your yeast, or the pale ale your regulars come back for. If you ask it to "improve my recipe" without a goal, it will give you a confident, generic answer that may have nothing to do with the beer you actually want.

So the first job is not technical at all. It is to write down what success looks like, in terms you could measure or taste.

What "good" can mean

Quality is never a single number. For most breweries it is a balance of four things, and naming which ones matter most for a given beer is half the work:

- **Flavor target:** the sensory profile you are aiming for: balance, key descriptors, what must be present and what must be absent.
- **Consistency:** how close batch ten must taste to batch one. A flagship lager lives or dies on this; a one-off special does not.
- **Cost:** ingredient, energy, and labor cost per finished litre or hectolitre.
- **Time:** tank turns, time to package, how long a beer ties up capacity.

A goal like "make it better" hides which of these you are willing to trade. "Hold the same flavor but shave two days off tank time" is something a tool, and a brewer, can actually work on.

KEY IDEA

A useful goal names a **target**, a **measure**, and a **trade-off you accept**. "Hit 4.8% ABV, clean fermentation, no diacetyl, without extending the lager by more than a day" is far more workable than "make a great lager."

Write a quality definition you can check

For each beer, a short, written quality definition turns vague intent into something you can hold an answer up against. It does not need to be long. A practical one covers:

- **Numbers and ranges:** OG, FG, ABV, IBU, colour, pH, with acceptable spreads (for example FG 1.010 to 1.012, not a single point).
- **Sensory descriptors:** the three or four notes that must be there, and the off-flavors that are an automatic fail (diacetyl, DMS, acetaldehyde, oxidation).
- **Pass / fail line:** what sends a batch to sale, what sends it to blending, and what sends it down the drain.

Once this exists, "is this AI suggestion any good?" has an answer. You check the suggestion against the definition instead of against a gut feeling you cannot share with your team.

Vague goal vs. defined goal

VAGUE

"Help me make a better IPA."

DEFINED

"Keep my IPA's hop character and 6.2% ABV, but reduce haze and lower my dry-hop cost by 15%. FG must stay 1.012 or below."

The second version gives any tool (and any colleague) something to aim at, and gives you a clear way to judge what comes back.

Where AI fits, once the goal is clear

With a written goal and a quality definition in hand, AI stops being a magic box and becomes an assistant pointed at a specific job: suggest grain bills that hit a target gravity, flag which past batches drifted from spec, draft a tasting note, or talk through why a fermentation stalled. Every one of those is easier to judge because you already decided what good looks like.

TRY THIS IN YOUR BREWERY

Pick one flagship beer. In ten minutes, write its quality definition: target numbers with ranges, the must-have and must-not-have sensory notes, and the pass/fail line. Keep it to half a page. That single document is what makes every later chapter (data, trust, feedback) actually usable.

Takeaways

- Start with the beer and the goal, never with the tool.

- "Good" is a balance of flavor, consistency, cost, and time, name which matter most for each beer.
- A useful goal states a target, a measure, and a trade-off you accept.
- A short written quality definition lets you judge any AI answer against something real.

Brew Data + Record Keeping

Your brew log is a dataset. Treat it like one and almost every useful thing AI can do for your brewery becomes possible. Treat it casually, and even the best tool will quietly mislead you.

This is the least glamorous chapter and the most important. Nothing downstream (prediction, consistency checks, searching your own knowledge) works without records that are complete, consistent, and honest.

What data does brewing AI actually need?

You do not need a sensor on every valve. You need the handful of fields that describe each batch well enough to compare it to the next. For most breweries that is:

- **Recipe inputs:** grain bill, hops, yeast strain and pitch rate, water profile, adjuncts, with quantities.
- **Process:** mash temperatures and times, boil length, fermentation temperature curve, dry-hop timing.
- **Measurements:** OG, FG, ABV, pH, IBU or bitterness where you have it, dates and tank IDs.
- **Outcome:** sensory notes, pass/fail against your quality definition from Chapter 1, and anything unusual that happened.

KEY IDEA

The most valuable column in your log is the **outcome**. Inputs and process tell a tool what you did; the outcome tells it what that did to the beer. Without honest outcomes, there is nothing to learn from.

Capture it cleanly

Messy records are not just untidy, they actively break tools. A model cannot tell that "Maris", "Maris Otter", and "MO" are the same malt, or that one batch logged gravity in Plato and the next in specific gravity. A few habits prevent most of the damage:

- **One unit per field, always.** Pick Plato or SG, Celsius or Fahrenheit, and never mix them in the same column.

- **Consistent names.** Spell ingredients and yeast strains the same way every time. A short pick-list beats free text.
- **Dates and IDs on everything.** Every batch needs a date and a tank or batch number so events can be put in order.
- **Record the misses.** Stuck fermentations, infections, and dumped batches are some of the most useful data you have. Do not quietly delete them.

Evaluate whether your data is any good

Before trusting any answer built on your records, ask whether the records deserve that trust. Three quick checks catch most problems:

Gaps

How many batches are missing a final gravity, a fermentation temperature, or a sensory note? A tool will happily ignore the gaps and give you a confident answer based on the half of the data that is complete, which may not be representative.

Errors

Scan for impossible values: an FG above the OG, a 9% ABV on a session recipe, a mash at 9 degrees. These are usually typos, but a tool reads them as fact.

Bias

If you only kept good records when a batch went well, your data quietly tells the tool that everything works. The failures that would teach it the most are missing.

TRY THIS IN YOUR BREWERY

Export your last 20 batches into one sheet, one row each. Highlight every empty cell and every value that looks impossible. The pattern of what is missing (and which beers it is missing for) tells you exactly where your records are not yet ready for a tool to lean on.

Start small

You do not need three years of perfect data to begin. A clean, consistent log of one beer over a dozen batches is more useful than a sprawling, contradictory archive of everything. Get the habit right on one flagship, prove it helps, then widen it.

Takeaways

- Your brew log is a dataset; its quality sets the ceiling on what any tool can do.
- Capture inputs, process, measurements, and (above all) honest outcomes.

- One unit per field, consistent names, dates and IDs on everything.
- Check for gaps, impossible values, and the missing failures before you trust a result.

Understanding the Tool

Most bad experiences with AI come from a wrong mental model. People expect a calculator and get a confident colleague; or expect an expert and get a parrot. The truth sits in between, and naming it makes the tool far easier to use.

A very well-read apprentice

The most useful picture of a general AI tool is an apprentice who has read almost everything ever written about brewing (every textbook, forum, and brewing journal) but who has never set foot in your brewhouse. They are fast, tireless, and widely read. They have also never tasted a beer, never smelled a stuck fermentation, and never run your kit.

That single image predicts most of the tool's behavior. It is brilliant at recalling and explaining general knowledge, and weak at anything that depends on your specific equipment, water, and palate.

KEY IDEA

AI is strong on **general, written-down knowledge** and weak on **your specific, tasted reality**. You supply the brewhouse; it supplies the reading. Neither replaces the other.

What it knows well

- Explaining concepts: what diacetyl is, why a rest helps, how attenuation works.
- Drafting and summarizing: tasting notes, SOPs, batch reports, supplier emails.
- Doing the textbook arithmetic: rough gravity, ABV, strike-water, and dilution maths.
- Talking through possibilities when a fermentation behaves oddly.

What it has never tasted

- How *your* beer actually tastes, or what your regulars expect.
- The quirks of your specific kit, that one tank that runs warm, your real mash efficiency.
- Anything that happened after its training, including this season's hop crop.
- Facts it does not have, which it may fill in with confident invention rather than admit.

Why the same question gives different answers

Ask a model the same thing twice and you may get two different replies. This is not a fault; these tools are built to vary their wording, and they do not "remember" past brews unless you tell them in the conversation. Two consequences follow for brewers:

- **Do not treat an answer as a fixed fact.** It is one plausible response, not the single correct readout of an instrument.
- **Context lives in the conversation.** If you do not give it your numbers and house style, it is working from generic knowledge every time.

TRY THIS IN YOUR BREWERY

Ask the same brewing question twice in two fresh chats, once with no detail, once with your recipe, gravities, and equipment. Compare the four answers. You will see clearly where the tool is genuinely helpful and where it is just filling a vacuum with generic text.

Setting expectations with your team

If others in the brewery will use the tool, agree on the mental model out loud: it is a knowledgeable assistant for thinking and drafting, not an authority that decides what goes in the tank. That one sentence prevents both the fear ("it will replace us") and the over-trust ("the computer said so") that cause most problems.

Takeaways

- Picture a very well-read apprentice who has never tasted your beer.
- Strong on general written knowledge; weak on your specific, tasted reality.
- Answers vary and the tool has no memory of past brews unless you provide it.
- Agree the mental model with your team to avoid both fear and over-trust.

Trust + Verification

The single most important habit in using AI well is separating how confident an answer *sounds* from how likely it is to be *true*. These tools are fluent by design, and fluency reads as authority even when the facts underneath are invented.

Confident is not the same as correct

A model will give you a precise strike-water temperature, a named enzyme, or a specific hop alpha figure in the same calm, assured tone whether it is right or guessing. It does not flag its own uncertainty the way an honest colleague would. So you cannot use tone as a signal. You have to bring the doubt yourself.

KEY IDEA

Treat every AI answer as a **well-informed hypothesis from an apprentice**, not a reading from an instrument. Hypotheses are useful, but you confirm them before you act, especially when a batch is on the line.

Ask for the reasoning, not just the answer

A bare number is hard to check. The reasoning behind it is much easier. Asking "walk me through how you got that" does two things: it lets you spot a wrong assumption, and it often makes the tool catch its own mistake. If the reasoning does not hold up against your brewing knowledge, the answer does not either, however confident it sounds.

Watch for invented sources

If an answer cites a study, a book, or a number, be especially careful. Models sometimes produce citations that look real but are not. Treat any specific reference as unverified until you have seen the source yourself.

Verify against the brewhouse

You have something the tool does not: the actual beer. Use it. The strongest verification is always physical:

- **Check the numbers** against your own measurements and your quality definition from Chapter 1. Does the suggested FG match what your yeast actually does?

- **Sanity-check with experience.** If a suggestion contradicts what a hundred batches have taught you, your batches win.
- **Test small before you scale.** A trial batch, a bench addition, or a single tank is a cheap way to confirm an idea before it touches your flagship.

Match your scrutiny to the stakes

Not everything needs the same caution. A draft tasting note or a brainstorm of style ideas is low risk, a wrong word costs nothing. A change to a fermentation schedule, a dilution calculation, or anything that affects safety, duty, or a full tank is high risk and deserves full verification every time.

TRY THIS IN YOUR BREWERY

Take one AI suggestion you would actually consider using. Before acting, write down: the number or claim, how you would check it, and what it would cost if it were wrong. If you cannot answer the second question, you are not ready to act on it yet.

Takeaways

- Confidence in the wording tells you nothing about correctness.
- Ask for the reasoning; check it against your brewing knowledge.
- Treat any citation as unverified until you have seen the source.
- Verify against real measurements and small tests; scale your scrutiny to the stakes.

Feedback + The Brewer's Hand

Good use of AI looks like a conversation, not a vending machine. You ask, you judge, you correct, you ask again, and you stay the one who decides what actually happens to the beer. This chapter is about keeping that loop healthy.

You are in the loop, on purpose

The most reliable way to use these tools in a brewery is with a human in the loop at every decision point: the tool proposes, the brewer disposes. That is not a limitation to grow out of. For anything that touches the tank, your hand on the final call is the safety mechanism.

KEY IDEA

AI should **widen your options and speed up your thinking**, then step back. The moment a tool is quietly making a brewing decision no human reviewed, you have given away the thing that makes you a brewer.

Steering: the first answer is a draft

Treat the first reply as a starting point, never the finished article. The skill is in the correction:

- **Add what it missed.** "You assumed 75% efficiency; mine runs 68%. Redo it." Specific corrections get specific improvements.
- **Push back.** "That would over-bitter it for my house style. Give me a gentler option." The tool will gladly revise.
- **Narrow the scope.** If an answer is generic, it usually means the question was. Add your numbers and constraints and ask again.

Build a control point into your workflow

Decide in advance where the brewer's sign-off sits. A simple rule works: anything that changes a recipe, a schedule, a dilution, or anything affecting safety or duty gets a human review before it is acted on, no exceptions, no matter how confident the tool sounded. Low-stakes drafting can flow more freely.

Feed your own knowledge back in

The tool gets far more useful when you give it your context: your house numbers, your past batches, your quality definition. Over time, keeping a short reference you paste into

conversations (your efficiencies, your yeast behavior, your standard processes) turns a generic apprentice into one that answers in your brewery's terms. You are not training the model; you are briefing it well, every time.

TRY THIS IN YOUR BREWERY

Write a short "house facts" note: your real mash efficiency, typical attenuation by yeast strain, water profile, and the three rules you never break. Paste it at the start of a brewing conversation and watch how much more specific and useful the answers become.

Takeaways

- Keep a human in the loop at every decision that touches the tank.
- The first answer is a draft; correct, push back, and narrow it.
- Define a clear control point where the brewer signs off before acting.
- Brief the tool with your house facts so it answers in your brewery's terms.

Mistakes + Red Flags

Every tool fails in its own characteristic ways. Once you know how AI tends to get brewing wrong, the mistakes become predictable, and predictable mistakes are easy to catch before they cost you a batch.

How AI gets brewing wrong

- **Invented facts.** It can state a hop alpha, an enzyme's rest temperature, or a yeast's attenuation with total confidence and be simply wrong. It would rather give a fluent answer than admit it does not know.
- **Out-of-date knowledge.** It does not know this season's harvest, a supplier's reformulation, or anything newer than its training. Brewing moves; the model's knowledge is frozen.
- **Unit and scale slips.** Mixing Plato and SG, Celsius and Fahrenheit, or grams and ounces; or giving homebrew-scale advice for a commercial batch.
- **Fake citations.** References and figures that look authoritative but do not exist.
- **Losing the thread.** In a long conversation it can forget a constraint you set earlier and contradict itself.

THE RED-FLAG CHECKLIST

- A suspiciously precise number with no working shown.
- A confident claim about something recent or local.
- A citation you cannot independently find.
- Advice that ignores your scale, kit, or house style.
- An answer that contradicts what your own batches have taught you.
- Units that do not match the rest of your records.

Diagnose: the tool or the question?

When an answer is wrong, it pays to know why before you give up or blame the tool. Usually it is one of two things:

- **The question was thin.** A vague prompt with no brewing context gets a generic, often wrong answer. The fix is on your side: add your numbers, scale, and constraints and ask

again. (Chapter 1 of the prompts guide covers this in detail.)

- **The tool genuinely does not know.** If the question is specific to your kit, your local ingredients, or very recent events, no amount of prompting will help. That is a job for your own measurement, not the model.

The safe way forward

Catching a mistake is only half the job; handling it well is the other half:

- **Stop at the tank.** A wrong answer that never gets acted on costs nothing. Your verification step (Chapter 4) and your control point (Chapter 5) are what keep it harmless.
- **Correct and re-ask** when the fault was a thin question.
- **Fall back to your own knowledge and measurement** when the tool is out of its depth. The beer is the final authority, always.
- **Note the failure mode.** Once you have seen the tool make a particular kind of mistake, you will spot it instantly next time.

TRY THIS IN YOUR BREWERY

Deliberately ask the tool something it is likely to get wrong, a very recent fact, or a number specific to your kit. Watch how it answers anyway, with full confidence. That felt experience of a confident wrong answer is the best training your instincts can get.

The throughline

Across all six chapters the message is one thing: AI is a capable assistant and a poor authority. It will not taste your beer, pull your tanks, or replace an experienced brewer. The brewers and distillers who learn to use it thoughtfully and critically (clear goals, clean data, a realistic mental model, real verification, a hand on the control, and an eye for its mistakes) will get the most from it, while staying firmly in charge of the craft.

Takeaways

- AI fails predictably: invented facts, stale knowledge, unit slips, fake citations, lost context.
- Run the red-flag checklist on any answer before acting.
- Diagnose whether the fault is a thin question or a tool out of its depth.
- Stop wrong answers at the tank; fall back to measurement; the beer is the final authority.