



# Nate Herkelman

Scale Without Increasing Headcount | Founder & CEO @ Uppit AI

42k followers • [View Profile](#)

**3**

POSTS / WEEK

**3 PM, UTC**

MAIN POSTING TIME

**490**

AVG REACTIONS

55 comments

**90%**

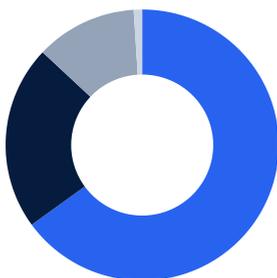
CALL-TO-ACTION

FREQUENCY

## #1 SUCCESS FACTOR

He keeps one simple promise ("scale without more headcount") and proves it over and over with practical builds people can copy. His best posts don't just teach—they ship a complete "do it now" package (clear result, real numbers, and usually a free resource plus a deeper walkthrough), so trust and engagement rise at the same time. That makes the next step feel natural: click to YouTube, grab the resource, then join the community when you're ready.

## FORMAT BREAKDOWN



Video	65%	520 avg reactions
Carousel	22%	390 avg reactions
Image	12%	390 avg reactions
Text Only	1%	640 avg reactions

## 7 Tactics You Can Steal

01 [Make one simple promise, then teach it again and again](#)

02 [Turn each post into a full kit: lesson, free file, and walkthrough](#)

03 Start with a hard number so people trust you fast

---

04 Make big skills feel small with a short "start here" roadmap

---

05 Explain your build like a simple machine with named parts

---

06 Use a popular argument, then settle it with a simple scorecard

---

07 Be excited, but also say what can go wrong

# 7 Tactics You Can Steal

## 1 Make one simple promise, then teach it again and again

His profile promise is plain and measurable (“scale without increasing headcount”), and most of his content is built to prove it with real, practical examples. This works because people remember one clear promise, and every useful tutorial feels like more proof that the promise is real. The bulk of his output supports this: technical “how-to” content is the majority (55 of 78 posts) and performs stronger on average (about 1.32x relative engagement) than business-only advice (about 0.96x). To copy it: write one sentence that names the outcome and the tradeoff you remove, then pick content topics that repeatedly show that outcome happening in the real world.

[Example 1 →](#)

[Example 2 →](#)

[Example 3 →](#)

[Profile →](#)

---

## 2 Turn each post into a full kit: lesson, free file, and walkthrough

Many of his highest-performing posts are not just explanations—they come with a free PDF/template/workflow plus a longer walkthrough people can follow. This works because readers get quick value in-feed, but also a clear path to actually doing it, which drives saves, shares, and “where’s the link?” comments. The numbers back it up: technical posts that include an explicit free asset average about 1.58x relative engagement (15 posts) versus about 1.23x for technical posts without a downloadable (40 posts). To copy it: write a short summary, show a preview list or screenshots, then offer the free asset and the longer walkthrough as the optional next step.

[Example 1 →](#)

[Example 2 →](#)

[Example 3 →](#)

---

## 3 Start with a hard number so people trust you fast

He often opens with numbers (count, time, cost, version, platforms) so the claim feels real and easy to judge at a glance. This works because numbers make your post feel testable, and they help the right people self-select (“this is exactly what I need”). It shows up repeatedly in his top performers, and it also shows up in his best business posts where specific dollars beat vague advice (money/case-study posts average about 1.14x relative engagement versus about 0.87x for generic frameworks). To copy it: put one specific number in the first line, then explain what it means and how someone can replicate it.

[Example 1 →](#)

[Example 2 →](#)

[Example 3 →](#)

---

4

## Make big skills feel small with a short “start here” roadmap

Some of his strongest posts take a huge topic and turn it into a simple order of steps (“do this before that”), with a checklist people can follow. This works because it removes overwhelm—readers don’t just learn a fact, they get a plan. A big reason it lands is that it’s framed as hard-earned hindsight (and often includes a free guide), which increases saves and shares. To copy it: pick one common beginner mistake, give the right sequence in 5–10 bullets, and end with one next action (template/guide/video) for people who want the full path.

[Example 1 →](#)[Example 2 →](#)[Example 3 →](#)

5

## Explain your build like a simple machine with named parts

When he shares a complex system, he breaks it into a few clear parts with names and jobs, so readers can picture how it works. This works because people don’t share “a pile of steps”—they share a simple model they can repeat to others. Several of his highest-performing build posts use this “parts + jobs” format, often adding how he tracks what’s happening so it feels safe to use. To copy it: name 3–5 parts, give each part one sentence (“it does X”), then show what the whole machine produces at the end.

[Example 1 →](#)[Example 2 →](#)[Example 3 →](#)

6

## Use a popular argument, then settle it with a simple scorecard

He jumps into debates people already care about, but instead of ranting, he tests and compares using a clear checklist (like a few categories anyone can understand). This works because it turns noise into clarity, and it feels fair—even if people disagree, they can talk about the same checklist in the comments. The comparison posts also naturally lead to a deeper walkthrough CTA, which fits his overall CTA pattern (YouTube is used in 67 of 78 posts). To copy it: pick two options people argue about, compare them across 4–7 simple categories, then end with the rule of thumb for choosing.

[Example 1 →](#)[Example 2 →](#)[Example 3 →](#)

7

## Be excited, but also say what can go wrong

When he shares something new, he often adds one honest limit (beta, rollout delays, where it struggles, what still matters). This works because it makes the post feel trustworthy, not like hype, and it prevents confusion-driven negativity in the comments. It also protects performance against his known weak spot: low-context quick-hit posts can underperform badly (some clusters average around 0.51x relative engagement, and a short teaser clip hit about 0.36x). To copy it: after the exciting claim, add a “watch out for this” section and one practical workaround so people can succeed on the first try.

[Example 1 →](#)[Example 2 →](#)[Example 3 →](#)

# Profile Breakdown

## HEADLINE FORMULA

Scale Without Increasing Headcount | Founder & CEO @ Uppit AI

[Big outcome] without [common tradeoff] | [Role + authority] @ [Company]. It works because the first half is plain-language business value (fast to scan, easy to remember), and the second half anchors the claim to real credibility (decision-maker title + where the offer lives).

## BANNER STRATEGY



Minimal gradient background (light cyan → deep blue), no text/logos. It works as a clean, modern “tech” signal that doesn’t compete with the headline, but it doesn’t carry positioning or a call-to-action—so the headline/About/Featured have to do the heavy lifting.

# Content Strategy

PILLAR	%	WHAT CRUSHES	EXAMPLES
<b>Practical automation builds &amp; tool teaching (templates, workflows, how-to)</b>	<b>71%</b>	This is the main growth engine: 55 of 78 posts are technical and average about 1.32x relative engagement versus business posts averaging about 0.96x. Within this pillar, the biggest performance lever is attaching a tangible free asset: "free asset" technical posts average about 1.58x relative engagement (15 posts) versus technical posts without a clear downloadable averaging about 1.23x (40 posts). What tends to lag is low-context quick tips/maintenance-style updates, which average about 0.51x relative engagement across 4 posts.	<a href="#">Post 1</a> <a href="#">Post 2</a> <a href="#">Post 3</a>
<b>Monetization, pricing, and getting clients (business results over theory)</b>	<b>21%</b>	Business content performs best when it's proof-based. Money/case-study posts average about 1.14x relative engagement (5 posts), while more generic positioning/pricing/client-acquisition frameworks average about 0.87x (11 posts). The weakest cluster is generic client-acquisition takes (especially cold email angles), averaging about 0.35x relative engagement across 3 posts.	<a href="#">Post 1</a> <a href="#">Post 2</a> <a href="#">Post 3</a>
<b>Personal wins, community, and identity-building stories</b>	<b>9%</b>	In this pillar, "real moments with receipts" beat generic encouragement. Credibility wins and milestone recaps average about 1.57x relative engagement (3 posts), while pure encouragement/seasonal community posts average about 0.55x (2 posts). These posts work best when they include a concrete moment (award, milestone, public event) and a clear belief it proves for the reader.	<a href="#">Post 1</a> <a href="#">Post 2</a> <a href="#">Post 3</a>

# Conversion Strategy

## Their Offer(s)

AI Automation Society / AI Automation Society Plus (Skool community)

YouTube tutorials (free long-form education)

B2B AI consulting/implementation (via his companies)

Partner/affiliate tool promotions (discounts/credits)

## How They Promote in Posts

CTAs are used very consistently: about 89.74% of posts in the 78-post sample include a CTA (70 of 78), and most of those are a low-friction “watch the full breakdown” push to YouTube (67 of 78). CTAs typically appear at the end, after the lesson is already complete, so they feel like help instead of a pitch. In this sample, posts with a CTA average about 719 total interactions (likes+comments+reposts) versus about 516 without a CTA, which fits the pattern that his CTAs are usually attached to higher-value tutorials and free resources. He adds higher-commitment asks (join community, discount code, event RSVP) less often and usually as an optional second step.

[Example 1 →](#)

[Example 2 →](#)

[Example 3 →](#)

## How They Promote in Profile

The profile sets an outcome-first promise in the headline (scale without adding headcount) and repeats that practical, accessible angle in the About section, backed by credibility markers (community leadership plus implementation work). The Featured section functions mainly as fast social proof (public leaderboard results for the paid community), which builds trust quickly, but it’s more credibility than a direct “click here to join” conversion asset. The banner is clean but doesn’t carry an offer or CTA, so conversion relies on the headline/About plus the consistent post-to-YouTube pathway.

[View Profile →](#)

# Top 10 Posts

1



## In the past year, I've built 200+ AI automations in n8n, and they all boil down to 17 core nodes.

He distills a year of building into a short list of core building blocks, then gives readers a free PDF that explains when to use each one. The post previews the full list in-feed and offers a longer walkthrough for anyone who wants to implement it.

♡ 2031    💬 148    ↻ 270

2



## I Built the Ultimate Army of Media Agents in n8n

He explains a full content-production system made of multiple "agents," each with a clear job (create, post, research, manage assets). He also shares free resources and a walkthrough so people can copy the setup.

♡ 1745    💬 131    ↻ 93

3



## n8n Just Released Native Data Tables, and It's a Game-Changer for Quick Builds

He breaks down a newly released feature and explains what it changes for building faster. He includes practical reasons it matters, plus a setup walkthrough and a simple comparison angle against a common alternative.

♡ 1683    💬 151    ↻ 115

4



## Base44 + n8n = Build Literally Anything (as a normal person)

He claims you can build a real web app quickly without coding by pairing a simple front end with an automation back end. He shows a business-useful example and links to a full build walkthrough.

♡ 1365    💬 96    ↻ 67

5



## I Tested OpenAI's AgentKit Against n8n: What You Need to Know

He addresses a popular narrative by saying he tested both tools and compared them across a clear set of categories. He ends with a simple rule of thumb: choose based on your problem, not hype.

♡ 1217    💬 116    ↻ 86

6



## I've built 200+ AI agents in n8n, if I had to start over in 2026, here's what I'd do differently...

He gives a "start over" learning plan based on experience, arguing for a specific learning order to avoid common mistakes. He supports it with business-focused justification and points to a longer guide and video.

♡ 1117    💬 119    ↻ 143

7



### I just built a Photoshop AI agent in n8n that uses Google's new state-of-the-art Nano-Banana image generation model 🐹

He demos a creative tool that can find, combine, and edit images, and he makes it feel practical with a clear cost and speed claim. He offers a free template and a walkthrough so readers can reproduce it.

♡ 1222    💬 84    ↻ 59

8



### I Won the First Ever AI Agent Game Show, and We Donated the \$20k to Charity

He shares a public win, the story around it, and a values-driven outcome through a charity donation with a matching contribution. He uses the moment to reinforce that non-traditional backgrounds can still succeed in this space.

♡ 1164    💬 143    ↻ 25

9



### This n8n Workflow Auto-Posts to 9 Different Socials

He shows a workflow that removes the hassle of posting across many platforms, then breaks it into a simple recipe using a few common tools. He adds an extra incentive for people who want to try it immediately.

♡ 1107    💬 82    ↻ 63

10



### n8n's Text-to-Workflow Somehow Made Building Agents Even Easier

He introduces a new feature that generates workflows from text, then teaches how to use it well with examples and workarounds. He sets expectations by explaining what it can and can't do yet and links to a deeper walkthrough.

♡ 983    💬 101    ↻ 63