



# Alex Xu

Author of 4 Bestselling Books | Co-Founder of ByteByteGo

1M followers • [View Profile](#)

6

POSTS / WEEK

5 PM, UTC

MAIN POSTING TIME

1190

AVG REACTIONS

30 comments

98%

CALL-TO-ACTION

FREQUENCY

## #1 SUCCESS FACTOR

He turns confusing, high-stakes topics into simple “pick the right option” visuals people can use immediately, which makes the content highly saveable and shareable. That steady authority then makes his links feel like a natural next step, because the profile and the posts both point to the same promise: practical system design learning.

## FORMAT BREAKDOWN



■ Image 100% 1190 avg reactions

## 7 Tactics You Can Steal

- 01 [Use “X vs Y vs Z” so people instantly know what they’ll learn](#)
- 02 [Explain each choice by what it does and what it won’t do](#)
- 03 [Turn big topics into a simple list people can save](#)

04 Teach hard ideas as “first this, then this, then this”

---

05 Start with “you think it’s just X, but it’s bigger”

---

06 End with a question that tells people exactly what to share

---

07 Put your link in a separate footer so value stays first

# 7 Tactics You Can Steal

1

## Use “X vs Y vs Z” so people instantly know what they’ll learn

He often starts with a simple comparison title, so the reader knows in one second what decision the post will help with. This works because people are busy and love anything that reduces choice overload. In his data, comparison/selection-guide posts average about 2.67x relative engagement (6 posts) versus basic primers at about 0.65x (4 posts). To copy it, pick a common choice in your world (two or three options), promise a clear difference, and keep the visual to one page so it’s easy to save.

[Example 1 →](#)

[Example 2 →](#)

[Example 3 →](#)

---

2

## Explain each choice by what it does and what it won’t do

In the best-performing explainers, each option gets a short “it does this” line and a memorable “it does NOT do this” line. This works because clear boundaries stop confusion and give people quotable lines they can repeat (which drives reposts). You can see this style across several of his highest-performing comparison posts, including the single biggest outlier at 7.04x relative engagement. To copy it, write one sentence for what each option is for, then one sentence for the most common wrong assumption.

[Example 1 →](#)

[Example 2 →](#)

[Example 3 →](#)

---

3

## Turn big topics into a simple list people can save

He frequently publishes “types you should know” and “concepts you should know” checklists, with one-line explanations per item. This works because a good list doubles as a study guide, so people save it and share it with teammates. Multiple top performers use this exact packaging, with relative engagement around 3x–4x for these one-slide references. To copy it, pick a big topic, break it into 10–15 categories, keep each description to one short line, and put the whole thing into a clean visual.

[Example 1 →](#)

[Example 2 →](#)

[Example 3 →](#)

---

4

## Teach hard ideas as “first this, then this, then this”

His strongest networking posts read like a short story with clear steps, not a pile of facts. This works because people remember sequences, and they can reuse the same steps in interviews or real work. End-to-end explainers average about 2.62x relative engagement (4 posts), while troubleshooting/checklist posts average about 0.44x (4 posts). To copy it, outline the process as 4–6 steps, name what changes at each step, and put the flow into one diagram.

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

5

## Start with “you think it’s just X, but it’s bigger”

He opens some posts by challenging the “simple story” most people believe, then proves the bigger picture with a checklist or comparison chart. This works because curiosity spikes when you realize your current mental model is outdated. One of his top posts uses this exact move to reframe a well-known tool and then backs it up with a clear capability checklist, driving strong sharing at 3.45x relative engagement. To copy it, pick a common oversimplification in your niche, state it plainly, then show the fuller map in one visual.

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

6

## End with a question that tells people exactly what to share

He doesn’t ask vague questions; he asks for predictions, tool picks, or “what would you add?” so readers know how to answer. This works because it lowers the effort to comment and invites people to show their experience (which they enjoy). He uses engagement questions in 66 posts, and this prompt style shows up repeatedly in his highest-performing posts. To copy it, choose one clear question type (prediction, favorite tool, biggest mistake, what to add) and put it in a final “Over to you:” line.

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

7

## Put your link in a separate footer so value stays first

Nearly every post gives the lesson first, then uses a separated footer for the link or offer (usually at the end). This works because the post still feels helpful even if someone never clicks, and the “ask” doesn’t interrupt reading. CTAs appear in 98% of posts; pure promo posts are much weaker (0.55x on average), and discount-sale posts are the lowest performers (0.13x), which supports keeping selling secondary. To copy it, write the full lesson, add your discussion question, then add a short footer after a divider line with one clear link.

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

# Profile Breakdown

## HEADLINE FORMULA

Author of 4 Bestselling Books | Co-Founder of ByteByteGo

[Quantified authority people can't easily fake] | [Role tied to a recognizable asset]. It works because the number ("4 bestselling books") is instant proof, and the asset ("co-founder of ByteByteGo") tells people exactly what to expect and where the expertise lives.

## BANNER STRATEGY



Show your flagship product(s) as the main visual + one short promise + one clear URL. It works because the visuals (book covers) act like proof, the repeated niche message reduces confusion, and the single URL gives profile visitors an obvious next step without hunting.

# Content Strategy

PILLAR	%	WHAT CRUSHES	EXAMPLES
System design, backend, and "which tool should I use?" guides	48%	Decision and comparison content wins here: comparison/selection-guide posts average 2.67x relative engagement (6 posts) versus basic "what is X" primers at 0.65x (4 posts). The best-performing examples are scannable "types you should know" lists and simple trade-off charts that help readers choose quickly; generic intro diagrams tend to underperform.	<a href="#">Post 1</a> <a href="#">Post 2</a> <a href="#">Post 3</a>
Networking and web security explained as step-by-step journeys	21%	End-to-end "here's what happens from start to finish" explainers average 2.62x relative engagement (4 posts), while troubleshooting/checklist-style networking content averages 0.44x (4 posts). The winning pattern is a clear sequence (what happens first, second, third) that turns a fuzzy concept into a story you can retell in an interview or at work.	<a href="#">Post 1</a> <a href="#">Post 2</a> <a href="#">Post 3</a>
AI and AI-building concepts (with a strong focus on agents)	18%	Agent-focused AI content is the strongest AI sub-topic: agent posts average 2.01x relative engagement (6 posts) versus 1.35x for other AI topics (12 posts). The weakest AI posts are low-context "go watch/read this" link-forward posts at 0.30x (3 posts), which suggests the audience wants the explanation embedded in the post, not outsourced to a link.	<a href="#">Post 1</a> <a href="#">Post 2</a> <a href="#">Post 3</a>
Announcements and offers (products, cohorts, sales, hiring)	13%	Promotions work best when they feel like a helpful next step, not the whole post. Pure promo content averages 0.55x relative engagement (13 posts), and discount-sale posts are extreme low performers at 0.13x (3 posts); the better-performing announcement style is participation-based ("join/review/subscribe") at 0.98x (4 posts).	<a href="#">Post 1</a> <a href="#">Post 2</a> <a href="#">Post 3</a>

# Conversion Strategy

## Their Offer(s)

ByteByteGo all-in-one tech interview prep platform (coding + system design + OOD + ML), often sold with a discount

ByteByteAI "Become an AI Engineer" cohort-based course

Weekly email newsletter (often paired with a free System Design PDF lead magnet)

YouTube channels/content

## How They Promote in Posts

CTAs appear in 98% of posts and are typically placed at the end, after the educational content, often separated as a clean footer block. The most common CTAs are engagement questions (66 posts), newsletter/lead-magnet asks (41–43 posts depending on type), and product links (33 posts); these are frequently stacked together (value → "Over to you" question → link). Using total interactions, posts with CTAs average ~1804 interactions (98 posts) while posts without CTAs average 2793 (2 posts), but the no-CTA sample is too small to treat as a strong conclusion—still, it supports testing occasional value-only posts.

[Example 1 →](#)

[Example 2 →](#)

[Example 3 →](#)

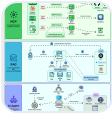
## How They Promote in Profile

The profile repeats one tight promise across headline, about, and banner (system design education backed by books + ByteByteGo) so visitors quickly trust what they'll get by following. The banner acts like a product shelf (book covers + bytebytego.com), while the Featured section highlights a free archive, making the first conversion step feel like a quick win before any paid offer.

[View Profile →](#)

# Top 10 Posts

1



## MCP vs RAG vs AI Agents

This post clears up three popular ideas that many people confuse by explaining what problem each one solves. It defines each concept in plain terms, shows how they fit at different "layers," and pairs the explanation with a one-slide visual reference. It ends with a learning-focused call to action for a free system design resource.

♡ 8372    💬 152    ↻ 1179

2



## Database Types You Should Know in 2025

This post turns the messy world of databases into a clear set of categories, with one-line guidance on what each type is good for. The visual maps each category to well-known examples so readers can connect the label to real tools. It ends with a forward-looking question that invites predictions and debate.

♡ 4727    💬 95    ↻ 559

3



## Evolution of HTTP

This post explains how a familiar web standard changed over time, version by version, and why each change mattered. It names the key improvements in simple terms and pairs them with a timeline-style visual. It closes with a question about real-world adoption of the newest version.

♡ 4525    💬 70    ↻ 468

4



## Cloudflare is much more than just a CDN and DDoS protection service.

This post challenges the common view of a well-known platform and argues it has expanded into much more. It backs that claim with a clear capability checklist and a visual comparison against other major platforms. It ends by asking readers whether they've tried the newer offerings.

♡ 4071    💬 70    ↻ 610

5



## Popular Backend Tech Stack.

This is a mostly-visual post that organizes the backend world into clear layers, showing common tools and where they fit. The caption is minimal, letting the pyramid-style infographic carry the value. It works as a one-page map someone can reference quickly.

♡ 4182    💬 69    ↻ 413

6



## RabbitMQ vs Kafka vs Pulsar

This post compares three messaging tools by explaining the different underlying models they are built on. It goes beyond surface features and explains how data moves, how long it stays, and how it can be read. It ends with a learning-focused call to action.

♡ 3876    💬 78    ↻ 594

7



## 12 Architectural Concepts Developers Should Know

This post lists core architecture concepts in a numbered format with short, plain-language definitions. The visual acts like a compact study sheet, making it easy to skim and revisit. It ends by inviting readers to add what they think is missing.

♡ 3807    💬 57    ↻ 664

---

8



## HTTP vs. HTTPS

This post explains the difference between two familiar web concepts by walking through what happens under the hood. It breaks the secure version into a clear sequence of steps and names the key pieces involved. It ends with a practical question about debugging tools.

♡ 3776    💬 55    ↻ 552

---

9



## Apache Kafka vs. RabbitMQ

This post compares two popular tools that people often treat as interchangeable, and explains the key differences in how they store and deliver data. It also points out a common misuse pattern to help readers avoid a bad design choice. It ends with a question about when one option is the wrong fit.

♡ 3691    💬 81    ↻ 514

---

10



## 12 Architectural Concepts Developers Should Know

This post presents the same evergreen set of core architecture concepts as a compact, visual study guide. Each item is short and skimmable, designed for quick review. It includes a free learning resource link as the call to action.

♡ 3429    💬 52    ↻ 535

---