The Engagement Rate Playbook

A step-by-step system for spotting your best-performing posts and turning that pattern into a repeatable content system with Claude Code.

Why I built this

Most people check one number after posting: reach. Reach tells you how many people saw something. It doesn't tell you if it landed. I kept posting daily and only looking at views, until I pulled up two posts on the same account and saw one sitting at 7% engagement rate and another at 21%. Same account, same followers, same posting time. The only difference was what the post actually was.

That gap is the whole point of this guide. Once you can see which posts are genuine outliers instead of just lucky, you stop guessing at what to make next and start repeating what already works.

This isn't a cheat sheet of hashtags or posting times. It's the exact process I use to pull the pattern out of my own analytics and hand it to Claude Code as a standing instruction, so the next batch of content gets planned against data instead of a mood.

01

How it actually works

1. Pull your last 20 to 30 posts and line up each one's engagement rate (not just likes or views) side by side in a spreadsheet or export.
2. Sort by engagement rate and flag anything sitting well above the rest of your average. On most accounts that's 15% or higher, but the real test is relative to your own baseline, not a universal number.
3. Look at what the flagged posts share: format, topic, hook style, length, delivery. Usually it's a combination of two or three, and it repeats across more than one outlier.

---

02 The one thing almost everyone gets wrong

Most creators notice a post did well, feel good about it, and go straight back to posting on instinct the next day.

> **WATCH OUT**
>
> Reach and engagement rate are not the same thing, and a post can have huge reach with a low engagement rate at the same time. If you're only tracking views or follower growth, you're optimizing for the wrong number. The outlier posts, the ones people actually acted on, are the ones worth studying line by line, not the ones that just got seen the most.

---

03 The prompt template (full copy-paste)

SKILL.MD

```
---
name: engagement-playbook
description: Reads exported post analytics, finds engagement rate outliers, names the
shared pattern, and drafts new content ideas against it.
---

# Engagement Playbook

## Step 1: Calculate engagement rate
Read the attached CSV of post analytics (reach, likes, comments, shares, saves per
post).
For each post calculate engagement rate as (likes + comments + shares + saves) / reach.
Sort posts from highest to lowest engagement rate.
Flag any post more than 5 percentage points above the account average as an outlier.

## Step 2: Name the pattern
For the flagged outlier posts, compare format, topic, hook style, length, and delivery.
Output the one or two things they share in common, in one or two sentences.
If the outliers don't share anything obvious, say so instead of forcing a pattern.

## Step 3: Draft against the pattern
Using the pattern from Step 2, draft new content ideas that follow the same format,
topic angle, and hook style as the outlier posts.
For each idea, give a one-line hook and a two-sentence summary.
```

Set it up in Claude Code or Codex

1. If you're using Claude Code, create a folder at ~/.claude/skills/engagement-playbook/ and add the SKILL.md above inside it. The folder name has to match the name: field at the top of SKILL.md, or Claude Code won't find it.
2. If you're using Codex instead, put the same SKILL.md in your project folder, or under .codex/skills/engagement-playbook/, and add a line in AGENTS.md pointing to it so Codex knows to load it.
3. Export your analytics. Most platforms let you export reach, likes, comments, shares, and saves per post as a CSV. Drop that file into the same folder.
4. Open a new terminal in the project and ask Claude Code to load the engagement-playbook skill against your exported CSV.
5. A successful first run ends with two things: a short list of your outlier posts with their engagement rates, and a one or two sentence description of the pattern they share. If you only get new ideas back with no pattern description, ask it to name the pattern explicitly before drafting anything new.

> 💡 **QUICK TIPS**
>
> - Re-run this every time you export a fresh batch of analytics, weekly or every two weeks, so the pattern updates instead of going stale.
> - Keep the raw CSV exports in the same folder over time. More history makes the outlier detection more reliable.

04

The commands I actually run

FIRST RUN OF THE WEEK

```
Load the engagement-playbook skill against [FILENAME].csv and give me this week's outlier posts and the pattern they share.
```

DRAFT NEW IDEAS

```
Using the pattern you found, draft [NUMBER] new content ideas for [PLATFORM] that follow the same format and hook style as my outlier posts.
```

COMPARE TWO TIME PERIODS

```
Compare the outlier pattern from [DATE RANGE 1] against [DATE RANGE 2] and tell me if the pattern has shifted or stayed the same.
```

SANITY CHECK A NEW POST IDEA

```
Here is a new content idea: [IDEA DESCRIPTION]. Does it match the outlier pattern you found, or is it closer to my average-performing posts?
```

When it breaks

ERROR	FIX
Claude Code says it can't find the engagement-playbook skill	Check that the folder name under ~/.claude/skills/ matches the name: field at the top of SKILL.md exactly, including hyphens.
The pattern it names feels too vague to act on	Ask it to name the pattern using the exact words from your captions and hooks, not a general category like 'educational content.'
It flags almost every post as an outlier	Your average engagement rate might be low across the board. Lower the flag threshold from 5 points above average, or compare against your top 25% instead of the mean.
The CSV export doesn't include engagement rate directly	That's expected. Most platforms only export raw counts (reach, likes, comments, shares, saves). The skill calculates engagement rate from those, so raw counts are all you need.
New ideas it drafts feel repetitive after a few weeks	Re-run Step 2 on your most recent outliers only, not the full history, so the pattern reflects what's working now instead of an average of everything.
Codex can't find the skill even though it's in the folder	Confirm AGENTS.md actually points to the skill's path. Codex only loads skills it's told about in that file.