

The Master Skills Setup Guide

Install the four Claude master skills (Edit, Virality, Build, Publish) and run your first cut, script and post.

What this is

Four Claude agent skills that cover the whole loop: find the idea, cut the video, build the page, post it with the keyword DM wired up. They live in one public repo and install with one command.

I spent weeks testing every Claude skills video I could find. Most of them were one clever trick with nothing on either side of it. A skill that writes a script and then leaves you to edit by hand. A skill that cuts video and has no opinion about what you should have filmed. The gap between them is where the whole week goes.

So I kept the parts that moved my numbers and packed them into four skills that hand work to each other. Virality writes the script. You shoot it. Edit cuts it and writes the captions. Publish posts it with the caption Virality already wrote and sets the comment-keyword DM. Build sits slightly off to the side, for the times the idea needs a page instead of a video. Over the test period my following doubled and my accounts crossed a million views combined. Nobody can promise you that. What the pack does is take the structure out of videos already working in your niche and put your voice on top of it.

This edition builds on Jake Schincariol's original master-skills pack. What I added: research that fetches itself instead of you reading view counts off your phone, a publisher that actually posts, a longer humanizer list, and a 58-second fit pass. Everything is MIT licensed and free.

How it actually works

1. Virality pulls real videos from your niche and ranks them by views as a multiple of each creator's own median, not by raw views. A 60k video from an account that averages 5k is the outlier. A 400k video from an account that averages 2 million is a flop. Raw counts put the biggest account on top every time and teach you nothing. It then breaks the winner down into hook formula, format, structure and CTA, and rewrites it in your voice from templates/voice.md.
2. Edit works off the transcript, not the waveform. That is why it strips the ums and the dead space without clipping the first consonant off the next word. It writes one to three word captions and places them in the safe zone instead of across your face, plans a visual for the moment you say the thing rather than four seconds later, and can turn frames or a live clip into stop motion in clay, paper, bricks, sticky notes or pixel. Audio finishes at -14 LUFS with no compression on the voice.
3. Publish sends the rendered file through Blotato to Instagram, TikTok and YouTube, plus Facebook, LinkedIn, X, Threads and Bluesky if you want them, using the caption, hashtags and YouTube title the script already carries. It waits until every post is live or scheduled and logs the links. With --dm-url it sets up the comment-keyword DM on Instagram and Facebook. Nothing posts until you type publish, and YouTube uploads land private until you say public.

The one thing almost everyone gets wrong

They skip templates/voice.md, run /virality once, and get back a script that reads like every other AI caption on the platform. The voice file is the only input that makes the rewrite yours, and every skill in the pack reads it.

It takes about five minutes. Open templates/voice.md and fill it in, or paste three captions you have already written and say "write my voice file". Claude will pull your sentence lengths, your openers, the words you actually use and the ones you never would, and write the file for you. Do it once and every script, caption and comment after that comes out sounding like you.

WATCH OUT

The second mistake is running /virality-publish before you have checked which accounts the ids point at. Run the --check command below first. It prints the account name behind every Blotato id, so you find out you had the wrong Instagram account before the video goes out, not after.

The files

SOURCE CODE

The full repo, MIT licensed, four skills:

<https://github.com/nessalazne/master-skills>

INSTALL BY HAND (CLAUDE CODE TERMINAL)

```
git clone https://github.com/nessalazne/master-skills.git
cp -r master-skills/skills/* ~/.claude/skills/
```

INSTALL AS A PLUGIN (INSIDE CLAUDE CODE)

```
/plugin marketplace add nessalazne/master-skills
/plugin install master-skills
```

~/.CLAUDE/VIRALITY/.ENV

```
YOUTUBE_API_KEY=...          # free: console.cloud.google.com, enable YouTube Data
API v3
APIFY_TOKEN=...              # Instagram fetch, about $0.16 per 30 reels on Apify's
free plan
BLOTATO_API_KEY=...          # posting and the keyword DM
BLOTATO_ACCOUNT_INSTAGRAM=1234
BLOTATO_ACCOUNT_TIKTOK=1234
BLOTATO_ACCOUNT_YOUTUBE=1234
BLOTATO_ACCOUNT_FACEBOOK=1234 # plus BLOTATO_FACEBOOK_PAGE_ID=... for Facebook
```

CHECK WHICH ACCOUNT EACH ID POINTS AT

```
python3 ~/.claude/skills/virality-publish/publish.py --check
```

CONFIRM THE PACK IS WIRED UP

```
python3 tests/test_ness.py
```

Every skill works by paste with no keys at all. The keys only turn on the fetching and the posting. The YouTube one is free. Apify runs about \$0.16 per 30 reels on its free plan. Blotato is a paid subscription. Keys go in ~/.claude/virality/.env and never in the repo.

SITES SHOWN IN THE VIDEO

- master-skills (the four skills covered here)
<https://github.com/nessalazne/master-skills>
- linkedin-agent-skill (15 LinkedIn skills, where the humanizer came from)
<https://github.com/nessalazne/linkedin-agent-skill>
- youtube-agent-skill (12 YouTube skills, where the fetch tools came from)
<https://github.com/nessalazne/youtube-agent-skill>

Set it up in Claude Code or Codex

1. Open Claude Code, paste `https://github.com/nessalazne/master-skills` and type install skills. That is the whole install for most people and it takes a couple of minutes. The rest of these steps are the manual route and the optional extras.
2. To do it by hand instead, run `git clone https://github.com/nessalazne/master-skills.git` in a folder you can find again, then `cp -r master-skills/skills/* ~/.claude/skills/`. You should end up with four folders: `~/.claude/skills/edit/`, `~/.claude/skills/virality/`, `~/.claude/skills/build/` and `~/.claude/skills/virality-publish/`.
3. Check the folder names. Each folder name has to match the name: field inside that folder's SKILL.md exactly. A skill in a renamed folder will not load and Claude will not tell you why.
4. For Codex, put the same four folders in your project, or in `.codex/skills/`, and add a line in AGENTS.md pointing at them so the agent knows to look. The folder-name rule is identical.
5. Install ffmpeg if you do not have it. `/edit` will not run without it on your PATH. If you also install faster-whisper, Edit transcribes for you; without it, you paste a transcript you already have.
6. Keys are optional. If you want them, ask Claude to add YOUTUBE_API_KEY, APIFY_TOKEN and BLOTATO_API_KEY to `~/.claude/virality/.env`, then open a new terminal window so they load. The YouTube key is free from console.cloud.google.com once you enable YouTube Data API v3.
7. Fill in templates/voice.md, or paste three of your own captions and say write my voice file. Skipping this is the mistake in the section above.
8. Test it: run `python3 tests/test_ess.py`. A clean pass means the pack is wired up. Then try the smallest real thing, `/virality` on your own niche. A successful first run hands you a ranked list of outliers with each creator's median beside the view count, not a generic list of trending topics. If you see raw view counts with no median, the fetch did not run and you are looking at something Claude guessed.

QUICK TIPS

- Start with `/virality` before `/edit`. Cutting a video you should not have filmed is the expensive mistake.
- Run `publish.py --check` once before your first post so you know which account every id belongs to.
- The skills also respond to plain English. You do not have to type the slash commands: "edit this video", "make this go viral", "build me a site that...", "publish it" all route correctly.

04

The commands I actually run

FIND THE OUTLIERS IN MY NICHE

`/virality`

Pull the last 90 days of Shorts from [CHANNEL HANDLE OR URL] and the reels under [#HASHTAG]. Rank them by views as a multiple of each creator's own median, not raw views. Give me the top 5 outliers with the hook, the format, the structure and the CTA for each, and tell me which one is closest to what [MY AUDIENCE] already responds to.

REWRITE THE WINNER IN MY VOICE

/virality

Take outlier [NUMBER] and rewrite it for [MY NICHE] using templates/voice.md. I need a shoot-ready script: on-screen text per beat, the visual for each beat, a caption, hashtags, and a comment-a-keyword CTA using the keyword [KEYWORD]. Score the hook before you hand it over and tell me what is weak about it.

CUT THE RAW TAKE

/edit

Here is the raw take: [PATH TO FILE]. Cut the dead space and the filler without clipping words, fit it under 58 seconds, and write 1 to 3 word captions in the safe zone. Plan dynamic visuals that land on the word, not after it. Finish the audio at -14 LUMFS, no compression on the voice. Do not trim the ending, the CTA is there.

BUILD THE PAGE THE VIDEO POINTS AT

/build

I want a page that [WHAT IT DOES] for [WHO IT IS FOR]. Write the one-page spec in plain English first and show me, then pick the simplest stack that works and commit to a design direction before you write any code. QA it in a real browser at phone, tablet and desktop widths, then run tools/sitecheck.py and fix what it finds.

POST IT AND WIRE THE DM

/virality-publish

Post [PATH TO RENDERED FILE] to Instagram, TikTok and YouTube using the caption and hashtags from the script. YouTube stays private for now. Set up the comment-keyword DM on Instagram and Facebook for the keyword [KEYWORD] with --dm-url [MY LINK]. Run --check first and show me the account names before anything goes out.

When it breaks

ERROR	FIX
command not found: ffmpeg	/edit needs ffmpeg on your PATH. Install it, then open a new terminal window so the path refreshes before you run the skill again.
Claude does not see the skill at all	The folder name has to match the name: field in that folder's SKILL.md. Check all four. In Codex, also confirm the pointer line in AGENTS.md.
The YouTube fetch returns nothing, or a 403	Either YOUTUBE_API_KEY is missing from ~/.claude/virality/.env, or the key exists but YouTube Data API v3 is not enabled on that project at console.cloud.google.com. Enabling it is a separate click from creating the key.
The Instagram fetch comes back empty	APIFY_TOKEN is missing or out of credit. The Apify fetch runs about \$0.16 per 30 reels on the free plan.
It posted to the wrong account	Run python3 ~/.claude/skills/virality-publish/publish.py --check. It prints the account name behind every Blotato id. Do this before the first post, not after.
The comment-keyword DM never fired	DMs run on Instagram and Facebook only, Blotato has no DM automation for TikTok. If that keyword already has a DM attached from an earlier video, the skill leaves it alone on purpose. Pass --dm-replace to overwrite it.
The cut is still over 58 seconds	fit.py speeds picture and sound together up to 1.1x and no further. Past that it stops and names the phrases to cut rather than guessing. It will never trim the ending, so the cut has to come from the middle.
The script sounds like every other AI caption	templates/voice.md is empty or generic. Paste three captions you wrote yourself and say write my voice file, then rerun.

Ready to go deeper?

This playbook is the short version. Inside the hub you get the full build walkthroughs, the prompt library, and the systems I use to ship this in a weekend.

Join the Build →

<https://hub.digicuratoragency.com/join>