

FREE RESOURCE · INSTALL & FIELD GUIDE

The LinkedIn Agent Skill

Eleven Claude skills that write your posts, comments, replies, DMs and weekly plan, then run every draft through a humanizer before you see it. This guide takes you from zero to your first post.

/li-post

/li-comment

/li-reply

/li-profile

/li-plan

/li-human

/li-carousel

/li-repurpose

/li-dm

/li-inbox

/li-audit

GET IT HERE

github.com/avi691/linkedin-agent-skill

Shared by **Avi Grondin** · Free, MIT licensed, no signup, no API key



SCAN FOR THE REPO

Contents

01	What this is, and the one rule	2 min read
02	Pick your install path	Which version of Claude you have
03	Install, step by step	Five ways, Mac and Windows
04	Check it works, and troubleshooting	The fixes for common errors
05	Set up your voice (do not skip)	10 minutes, biggest quality jump
06	The eleven skills	What to say to each one
07	The humanizer, in depth	Two scripts, five checks
08	Your weekly routine	20 minutes a day
A	Appendix: the 21 hook formulas	
B	Appendix: the 100-point profile rubric	
C	The fine print, credits and license	

IN A HURRY?

Open Claude, paste the block below, and let it install itself. Then jump to **Section 05** and set up your voice.

```
https://github.com/avi691/linkedin-agent-skill
```

```
Install this skill, then confirm /li-post works.
```

What this is, and the one rule

A **skill** is a folder of instructions (and sometimes small scripts) that Claude loads when the job calls for it. This pack has eleven, and together they cover the day-to-day work of running a LinkedIn account.

Writes
Posts from 21 hook formulas, carousels, comments on other people's posts, replies under yours, connection notes and DMs.

Plans
The week: what to post, when, and the ten people to engage with. Repurposes long content into a week of posts.

Checks
Scores your profile out of 100, audits what you already published, sorts your inbox, and humanizes every draft.

The one rule: it writes, you post

None of these skills post to LinkedIn, send a message or log into your account, and that's deliberate. Posting from a personal profile through automation or a browser bot breaks [LinkedIn's User Agreement](#) and gets accounts restricted. Every skill ends the same way: a copy-ready block of text that you read, approve and paste yourself.

How the pieces fit

Three small files live in a folder called `~/.claude/linkedin/` in your home folder, and they tie the pack together:

FILE	WHO WRITES IT	WHAT IT'S FOR
<code>voice.md</code>	You, once (Claude can draft it)	How you talk, who you're writing for, what you never say, and real numbers you're happy to use.
<code>plan.md</code>	<code>/li-plan</code>	This week's posting schedule and engagement list.
<code>log.md</code>	<code>/li-post</code> , <code>/li-comment</code>	Every post you approved and the hook it used, so <code>/li-audit</code> can learn from your history.

NOTHING IS MADE UP
The skills never invent metrics, clients or results under your name. If a draft needs a number you haven't given, it comes back with `{{your number}}` in it and a flag, so you fill in the real one.

Pick your install path

The right install depends on which Claude you use. Find your row, then follow that path in Section 03.

IF YOU USE...	USE PATH	WHAT YOU GET
Claude Code (terminal, VS Code, or the Code tab in the Claude desktop app)	A (easiest) or B / C	Everything, including the two humanizer scripts. Recommended.
Claude.ai in the browser or the Claude desktop/mobile chat	D	Skills upload. The humanizer scripts run if code execution is switched on in your settings.
Any chat assistant, no install allowed	E	Paste one skill at a time. Works, but you lose the humanizer scripts.

What you need

- **A Claude account.** Skills work on the paid plans; check Claude's current plan page if you're on the free tier.
- **For paths B and C:** [Git](#) (Macs usually have it; Windows users install "Git for Windows").
- **For the humanizer scripts:** Python 3. There's nothing else to install, since the scripts have no dependencies. Check with `python3 --version` (Mac) or `python --version` (Windows).
- **No API key, no LinkedIn login, and no third-party account.**

NEVER USED A TERMINAL?

Use **Path A**. You paste one message into Claude Code and it runs every command for you, asking before it does anything.

Install, step by step

PATH A · CLAUDE CODE · EASIEST

Let Claude install it for you

1 Open Claude Code

In a terminal type `claude`, or open the **Code** tab in the Claude desktop app.

2 Paste this message

```
https://github.com/avi691/linkedin-agent-skill  
  
Install this skill, then confirm /li-post works.
```

3 Approve the commands

Claude clones the repo and copies the eleven `li-*` folders into `~/.claude/skills/`. Approve each step when it asks.

4 Restart if needed

If `/li-post` isn't recognised straight away, start a new Claude Code session so it picks up the new skills.

Copy the skills yourself

Mac / Linux (Terminal)

```
# 1. download the repo
git clone https://github.com/avi691/linkedin-agent-skill.git

# 2. make sure the skills folder exists, then copy all eleven in
mkdir -p ~/.claude/skills
cp -r linkedin-agent-skill/skills/li-* ~/.claude/skills/

# 3. check: you should see eleven li-* folders
ls ~/.claude/skills
```

Windows (PowerShell)

```
git clone https://github.com/avi691/linkedin-agent-skill.git
New-Item -ItemType Directory -Force "$HOME\.claude\skills" | Out-Null
Copy-Item -Recurse linkedin-agent-skill\skills\li-* "$HOME\.claude\skills\"
Get-ChildItem "$HOME\.claude\skills"
```

Just for one project? Copy the same folders into that project's `.claude/skills/` instead of your home folder, so the skills only load inside that project.

Install as a plugin (easy updates)

The repo is also a Claude Code plugin marketplace. Inside Claude Code, run these two commands one at a time:

```
/plugin marketplace add avi691/linkedin-agent-skill
/plugin install linkedin-agent@linkedin-agent-skill
```

With a plugin install, Claude may show the commands with the plugin name in front, e.g. `/linkedin-agent:li-post`. They work the same way, and you can also just describe what you want in plain English. To get new versions later, update the marketplace from the `/plugin` menu.

PATH D · CLAUDE.AI (WEB, DESKTOP OR MOBILE CHAT)

Upload the skills to your Claude account

1 Download the repo

On [the GitHub page](#), click the green **Code** button → **Download ZIP**, then unzip it.

2 Zip each skill folder on its own

Open `skills/`. Each skill gets uploaded separately, so make a ZIP of each folder you want (right-click → *Compress* on Mac, *Send to* → *Compressed folder* on Windows). The ZIP must contain the folder, with `SKILL.md` inside it.

3 Turn on code execution

In Claude's **Settings**, find the capabilities section and switch on code execution / file creation. Skills need it, and it lets `/li-human` run its scripts.

4 Upload

In the same settings area, open **Skills** and upload each ZIP. Start with `li-human` and `li-post`, since the other skills hand drafts to `li-human`.

HEADS UP

Claude's settings menus get rearranged from time to time. If you can't find "Skills", search Claude's help centre for "upload a skill". Claude.ai can't see your computer's `~/.claude/linkedin/` folder, so paste your `voice.md` at the top of a chat or save it in a Claude **Project** as project knowledge.

PATH E · ANY CHAT · NO INSTALL

Paste a skill as a "mode"

Open any `SKILL.md` on GitHub (for example `skills/li-post/SKILL.md`), copy the whole thing, and paste it at the top of a new chat followed by your request. Paste `hooks.json` along with it for `li-post`, or `rubric.json` for `li-profile`.

The trade-off: without the two Python tools you lose the real humanizer, which is most of the point of `/li-human`. Everything else works.

Check it works

Three quick checks. If all three pass, you're installed.

1 Claude can see the skills

In Claude Code, type `/li`. The eleven commands should appear in the list. Or ask: *"Which LinkedIn skills do you have?"*

2 A real draft comes back

Run this:

```
/li-post we built an internal tool that cut proposal time from 5 hours to 20 min
```

You should get **three hook options** (each tagged with a formula number like `#17 Time Anchor`), one full draft, and a `POST READY` receipt with a human score.

3 The humanizer scripts run

From inside the `li-human` folder (in `~/.claude/skills/`):

```
cd ~/.claude/skills/li-human
python3 detect.py SKILL.md
```

You should see five bars (BURSTINESS, SPECIFICITY, SLOP DENSITY, FINGERPRINT, VOICE) and a HUMAN SCORE.

Troubleshooting

WHAT YOU SEE	FIX
<code>/li-post</code> isn't recognised	Start a new Claude Code session. Check the folders sit directly in <code>~/.claude/skills/li-post/SKILL.md</code> , not one level deeper (e.g. <code>skills/skills/li-post</code>).
<code>cp: ~/.claude/skills: No such file or directory</code>	Run <code>mkdir -p ~/.claude/skills</code> first, then copy again.
<code>git: command not found</code>	Install Git (Mac: run <code>xcode-select --install</code> ; Windows: Git for Windows). Or use Download ZIP and copy the folders by hand.
<code>python3: command not found</code>	Windows: use <code>python</code> instead of <code>python3</code> . Mac: run <code>xcode-select --install</code> , or install Python from python.org.

Drafts sound generic, like everyone else	You skipped <code>voice.md</code> . See Section 05. This fixes it more than anything else.
Commands show as <code>/linkedin-agent:li-post</code>	That's normal for a plugin install (Path C) and works the same way.
Claude.ai upload rejected	Each ZIP must hold one skill folder with <code>SKILL.md</code> inside. Don't zip the whole repo.
Hidden folder <code>.claude</code> not visible	Mac Finder: press <code>Cmd + Shift + .</code> to show hidden files. Windows: View → Show → Hidden items.

Updating later

Path B: run `git pull` inside your `linkedin-agent-skill` folder, then copy the `li-*` folders again. Path C: update the marketplace from the `/plugin` menu. Path D: re-upload the changed skill ZIPs.

The repo at a glance

```
linkedin-agent-skill/
├── skills/
│   ├── li-post/      SKILL.md + hooks.json      # 21 hook formulas
│   ├── li-human/     SKILL.md + humanize.py + detect.py + slop.json
│   ├── li-profile/   SKILL.md + rubric.json      # 100-point score
│   └── li-comment, li-reply, li-plan, li-carousel,
│       li-repurpose, li-dm, li-inbox, li-audit/  # one SKILL.md each
├── templates/voice.md # your voice profile
├── .claude-plugin/    # lets it install as a plugin
├── README.md
└── LICENSE            # MIT
```

BEFORE YOU UPDATE

If you've edited `slop.json` (your own banned and allowed words), keep a copy, because copying the folders again overwrites it.

Set up your voice

This is the ten minutes that decides whether you post the drafts or rewrite them. Every skill reads your voice profile. If you skip it, everything comes out sounding like everyone else on LinkedIn.

Option 1: let Claude write it (recommended)

Pick three of your own posts that sound most like you. Paste them into Claude and say:

```
Write my voice.md from these three posts, using the template in
templates/voice.md, and save it to ~/.claude/linkedin/voice.md.
```

```
[post 1]
[post 2]
[post 3]
```

Then read what it wrote and correct anything that's off. In particular, fill in **Proof I can use** with real numbers yourself. The skills never invent a number, so an empty Proof section means every draft comes back with `{{your number}}` placeholders. Five real figures (revenue, time saved, clients won, a mistake that cost you) unlock most of the strong hook formulas.

Option 2: fill in the template by hand

```
# Mac / Linux
mkdir -p ~/.claude/linkedin
cp linkedin-agent-skill/templates/voice.md ~/.claude/linkedin/voice.md

# Windows (PowerShell)
New-Item -ItemType Directory -Force "$HOME\.claude\linkedin" | Out-Null
Copy-Item linkedin-agent-skill\templates\voice.md "$HOME\.claude\linkedin\voice.md"
```

What the template asks for

- **Who I am:** what you do in one sentence, who you write for (as specific as "agency owners doing \$1–5M", not "professionals"), and what you sell.
- **What I sound like:** words you use and words you'd never use, sentence length, swearing, emoji, contractions (almost always yes).
- **My positions:** three to five things you believe that part of your audience doesn't. This is where the good posts come from.
- **Off limits & proof:** topics, clients and claims you can't use publicly, plus real numbers and stories you're happy to put your name on.

The eleven skills

You can type the slash command, or just say what you want in plain English. Claude picks the right skill from your request. The lines under "Say" are example prompts you can copy.

/li-post

One idea → a finished post

WHAT IT DOES

- Reads your voice and picks **three different hook formulas** from the 21, and says which one it would use.
- Writes the full post on the strongest one: hook alone on line 1, short paragraphs, one turn, one close.
- Humanizes it, then prints a copy-ready block and a **POST READY** receipt (hook, length, human score).
- When you reply "yes", it logs the post to **log.md**.

SAY

```
/li-post we lost a $40k deal because
our proposal took 9 days
Give me hook options for: I stopped
doing discovery calls
```

HOUSE RULES

- 900–1,300 characters
- No links in the body (link goes in the first comment)
- Max 3 hashtags, no "Thoughts?"

/li-comment

Comments on other people's posts

WHAT IT DOES

- Picks from **nine comment types** based on what the post actually says: add a datum, add the missing case, respectful disagree, extend one line, ask the real question, the receipt, the correction, the reframe, the one-liner.
- Gives two options of different types, plus which one to post and why.
- Batch mode: paste 5–10 posts for your daily engagement round.

SAY

```
/li-comment [paste the post + author's
name and role]
Here are 6 posts for my engagement
round: ...
```

HOUSE RULES

- 2–4 sentences, one idea
- Never "Great post!" and no emoji openers

/li-reply

The thread under your own post

WHAT IT DOES

- Sorts every comment into **LEAD, SUBSTANCE, PEER, SUPPORT, NOISE** and shows the counts.
- Writes replies in that order, so leads and real discussion come first.
- Handles critics (concede the true part, then hold your ground) and ignores pitches.

SAY

```
/li-reply [paste the comments, or a screenshot]
Someone disagreed with my post, help me reply: ...
```

TIP

- Run it within the first hour after you post. That's when replies do the most for reach.

/li-profile

Profile score out of 100 + rewrites

WHAT IT DOES

- Scores your profile on a **12-part, 100-point rubric** (Appendix B). Expect 30s–40s on the first pass, because it marks honestly.
- Rewrites in fix-first order: headline (3 options), about (first two lines), about body, featured, experience, banner.
- Re-scores at the end and tells you what a rewrite alone can't fix.

SAY

```
/li-profile [paste headline, about, current role, last two roles]
Score my LinkedIn from this screenshot
```

HEADLINE FORMULA

- {what you do for whom} | {proof} | {how to start}

/li-plan

The week, decided

WHAT IT DOES

- Plans **four posts a week**, mixing Proof, Opinion, Teach, Story and Offer, with a specific angle and hook number for each.
- Posting times built around when your audience is at a desk (B2B default: Tue–Thu, 7:30–9:30am in their timezone).
- An engagement list of 10 people: 5 reach, 3 peers, 2 buyers.
- Saves the plan to `p1an.md`. Say "write Tuesday" to draft a slot.

SAY

```
/li-plan this week I closed 2 clients, lost one to a cheaper agency, and rebuilt our onboarding
```

GIVE IT

- What actually happened this week. Posts come from real events, not topics.

/li-human

The humanizer (see Section 07)

WHAT IT DOES

- Removes invisible characters, em dashes and curly quotes, and swaps 113 stock "AI" words and phrases for plain ones.
- Flags sentence patterns that need a human rewrite, without changing them itself.
- Scores the draft on five checks and gives a verdict: PASS, REVIEW or FLAGGED.

SAY

```
/li-human [paste any text]
Does this sound like AI?
Remove the em dashes and de-slop this
```

NOTE

- The other skills run it automatically before showing you a draft.

/li-carousel

Document posts (PDF slides)

WHAT IT DOES

- 8–12 slides: cover (six words or fewer), the stakes, one idea per slide, a recap, a CTA.
- Writes the 2–3 lines of post text that go above the PDF.
- After you approve the copy, builds the 1080×1350 PDF for you to upload.

SAY

```
/li-carousel the 6 steps we use to
onboard a client in 48 hours
```

WHEN NOT TO

- If the idea is one claim and not a sequence, it'll send you to `/li-post`.

/li-repurpose

One long asset → a week of posts

WHAT IT DOES

- Pulls out claims, numbers, stories, how-it-works explanations, mistakes and quotable lines from a transcript, newsletter, podcast or call. It doesn't summarise.
- Builds a week with a different hook formula for each post, and each post stands on its own.
- Drafts one at a time when you ask, so they don't all sound the same.

SAY

```
/li-repurpose [paste transcript or
article]
Turn my latest newsletter into posts
```

/li-dm

Connection notes and follow-ups

WHAT IT DOES

- Asks who, why now, and what you want, and pushes back if there's no real reason to reach out.
- Writes the invite note (under 200 characters, with a character count), the first message, and **two** follow-ups (+4 days and +10 days).

SAY

```
/li-dm Jane Doe, Head of Growth at Acme, she posted about killing their SDR team
```

HOUSE RULES

- No calendar link in message one
- Max 20 invites/day, sent by hand

/li-inbox

Inbox triage

WHAT IT DOES

- Sorts pasted messages into **LEAD, RECRUITER, PEER, ASK, SPAM**, with counts first.
- Spots automated outreach sequences and tells you which giveaway it spotted.
- Drafts replies only for the messages worth answering, including polite, final declines.

SAY

```
/li-inbox [paste messages or screenshots]  
Should I reply to this?
```

/li-audit

What's actually working

WHAT IT DOES

- Ranks your posts by **engagement rate** and **reach multiple** (impressions ÷ followers), not raw impressions.
- Compares your top 5 and bottom 5 by hook, format, length and topic, and checks posting day last.
- Tells you what to stop and what to do more of, then passes that to `/li-plan`.

SAY

```
/li-audit [attach the LinkedIn analytics CSV]  
Why did this post flop?
```

GET THE DATA

- LinkedIn: Analytics → Content → Export

The humanizer, in depth

`/li-human` comes with two small Python scripts that need nothing else installed. They run on your own computer, on your own text, and nothing is uploaded anywhere.

```
# run from ~/.claude/skills/li-human
python3 humanize.py draft.txt --report          # clean it, list every change
python3 humanize.py draft.txt -o clean.txt       # save the cleaned version
python3 detect.py draft.txt                     # score it on five checks
python3 detect.py draft.txt clean.txt           # before vs after
```

1 · Invisible characters

Zero-width spaces and joiners, soft hyphens, byte-order marks, Unicode tag characters, odd spaces. Your keyboard doesn't type these, and they survive copy-paste.

2 · Typography

Em dash → comma, en dash → hyphen, curly quotes → straight, ellipsis → three dots, bullet character → hyphen.

3 · The slop lexicon

113 stock words and phrases (delve, leverage, robust, seamless, "let that sink in") swapped for plain ones, keeping capitals and leaving URLs alone.

Flagged, not fixed

These need judgement, so the script points them out and you rewrite them: *"It's not just X, it's Y"*, rule-of-three lists, one-word rhetorical questions ("The result?"), rocket/fire/sparkle emoji, hashtag walls, "Thoughts?"-style bait, and sentences or bullets that are all the same length.

The five checks (0–100, higher = more human)

CHECK	WHAT IT MEASURES	WHAT READS AS MACHINE-WRITTEN
BURSTINESS	How much sentence length varies	Every sentence about the same length
SPECIFICITY	Numbers, names and concrete details per 100 words	Abstract nouns, no figures
SLOP DENSITY	Lexicon hits per 100 words	Stock vocabulary
FINGERPRINT	Invisible characters, em dashes and curly quotes per 1,000 characters	Typographically perfect
VOICE	Contractions, first/second person, structural tells	No contractions, staged reveals

The final score is 60% the average and 40% the **weakest single check**, because a detector only needs one signal to fire. **PASS** needs 70+ overall with no check below 55.

The routine

1. `humanize.py draft.txt -o clean.txt --report`
2. Rewrite the flagged lines by hand, keeping the meaning.
3. `detect.py draft.txt clean.txt` to see the before and after.
4. Not PASS? Fix the weakest check it names and run it again. Two rounds is normal. If it takes five, the draft itself is the problem, so start a new draft rather than doing more passes.

MAKE IT YOURS

The lexicon lives in `skills/li-human/slop.json` and is meant to be edited. If it strips a word you genuinely use, delete that entry. If you have a pet phrase you want gone, add it.

AN HONEST LIMIT

These are local checks modelled on what public AI detectors look for. They're **not** GPTZero, Originality, Copyleaks, Winston or Turnitin, and they don't call those services. Fixing what they measure tends to improve those scores too, but nobody can honestly promise "undetectable".

Your weekly routine

The skills work best as a loop. Here's a simple week that takes about 20 minutes a day.

WHEN	RUN	WHY
Monday	<code>/li-plan</code>	Tell it what happened last week. You get four post slots, times and your ten people.
Every day, before posting	<code>/li-comment</code> (batch)	20 minutes commenting on your list. Comment early, before a post has 20 comments, or nobody sees yours.
Post days	"write Tuesday" → <code>/li-post</code>	Draft the planned slot, check the human score, paste it into LinkedIn yourself.
First hour after posting	<code>/li-reply</code>	Reply to leads and real discussion first. The first hour does most of the work for reach.
Friday	<code>/li-inbox</code> , <code>/li-dm</code>	Clear the inbox, then send a few targeted connection notes by hand.
Monthly	<code>/li-audit</code> , <code>/li-profile</code>	See what actually worked and feed it into next month's plans. Re-score your profile.
Whenever you make long content	<code>/li-repurpose</code> , <code>/li-carousel</code>	One video, newsletter or call becomes 4–6 posts. List-shaped ideas become carousels.

Your first 30 minutes

- 1 Install**
Path A, B, C or D from Section 03.
- 2 Write your voice.md**
Paste three of your posts and let Claude draft it. Add five real numbers to "Proof".
- 3 Score your profile**
`/li-profile` . Fix the headline and about first, since that's where most of the points are.
- 4 Plan the week**
`/li-plan` , then "write Tuesday".

The 21 hook formulas

Each one is in `skills/li-post/hooks.json` with a template, a filled-in example and the common way it goes wrong. `/li-plan` and `/li-post` refer to them by number.

#	FORMULA	BEST FOR
1	Contrarian Take	Establishing a point of view. Highest comment rate of any formula.
2	Number Reveal	Experiments, challenges, anything with a countable result.
3	Mistake Confession	Trust. People forward the posts where you look bad.
4	Before / After	Proof you've travelled the distance your reader wants to travel.
5	The List Promise	Saves. Highest save rate of the 21, which LinkedIn weights heavily.
6	Insider Secret	Credentialed takes. Works only if the years are real.
7	The Callout	Short posts with one instruction. Fastest to write.
8	Question Trap	Comment volume. People answer scenarios and ignore "thoughts?".
9	Story Cold Open	Long-form storytelling posts. Highest dwell time.
10	The Receipt	Anything you can prove. Pair it with an image.
11	Myth Bust	Reframing. Sets you up to deliver the real cause in line 3.
12	The Comparison	Tool and process posts. Very high share rate.
13	Permission Slip	Reaching people outside your niche. Travels further than tactical posts.
14	Pattern Interrupt	Stopping the scroll cold. The white space does the work.
15	The Warning	Readers who know the problem but haven't linked it to what it costs them.
16	Good vs Great	Quotable one-liners. People screenshot these.
17	Time Anchor	Tool, system and automation posts. Strongest shape for a how-to.
18	The Unpopular Rule	Positioning. Attracts the right clients by putting off the wrong ones.
19	Curiosity Gap	Openers that make people tap "see more".
20	The Walk-Away	Values posts with a business result attached.
21	The Direct Value	Lead magnets and comment-for-link posts. Highest conversion of the 21.

The 100-point profile rubric

This is what `/li-profile` scores you against (from `skills/li-profile/rubric.json`). Most profiles land in the 30s and 40s on the first pass.

ITEM	POINTS	WHAT FULL MARKS LOOKS LIKE
Headline	12	Says who you help and what changes for them, includes one piece of proof, and isn't just your job title. Uses the full 220 characters.
About, first 2 lines	10	Names the reader's problem before "see more". No "passionate about", no third-person bio, doesn't open with your name.
About body	10	Written to one person: their problem, what you do, one number as proof, what to do next. Under 1,400 characters.
Featured	8	Three items: your best post, a proof asset, and how to contact you.
Banner	6	One line of positioning and one way to reach you. Not the default gradient.
Photo	6	Face fills about 60% of the frame, eyes visible, taken in the last two years.
Current role	10	One line of scope, then 2–3 results with numbers. Duties aren't results.
Experience depth	8	Last two roles detailed; anything older than ten years cut to one line.
Skills	6	Your top three skills are the ones you want to be hired for, and they're endorsed.
Recommendations	8	At least three from the last two years, each naming a specific result.
Activity	10	Posted or commented in the last seven days.
Contact	6	Custom URL, a working email or booking link, and the same handle as your other platforms.
Total	100	

The fine print

These skills don't post for you, on purpose

There's no official way to post to a personal LinkedIn profile without an approved partner app, and automating the site with a browser or a third-party tool breaks LinkedIn's User Agreement and gets accounts restricted. Every skill ends with text you copy and paste. That's why you approve everything before it goes out.

The humanizer is honest about what it is

The five checks are heuristics that run on your machine. They don't call any detection service and can't promise a verdict from one. The invisible-character cleanup is real but limited: it removes stray formatting characters, and it doesn't claim to beat any cryptographic watermarking scheme.

Your data stays with you

The scripts read and write files on your own computer only. Your voice profile, plan and log live in `~/.claude/linkedin/`. Whatever you paste into Claude is handled under your own Claude account's terms.

Credits

This pack was originally created by **Jake Schincariol** (opusjake.ai) and released under the MIT License. It's shared here by **Avi Grondin** at github.com/avi691/linkedin-agent-skill.

License

MIT. You're free to use it, change it and share it. Keep the `LICENSE` file with any copy you pass on.

GET THE REPO

github.com/avi691/linkedin-agent-skill

If it helps you, star the repo and share this guide with someone who posts on LinkedIn.