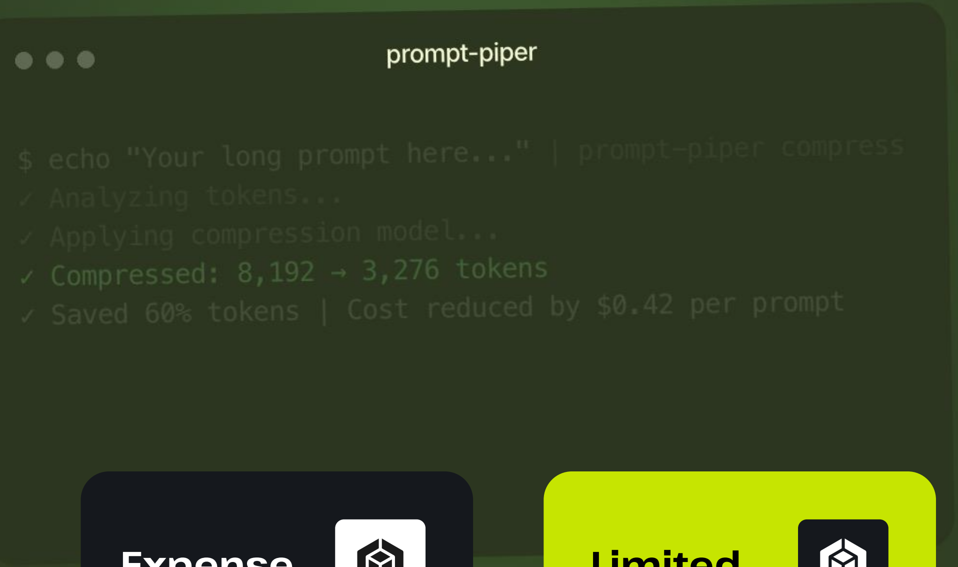
Prompt Piper: AI Compression

promptpiper.xyz



```
$ echo "Your long prompt here..." | prompt-piper compress
✓ Analyzing tokens...
✓ Applying compression model...
✓ Compressed: 8,192 → 3,276 tokens
✓ Saved 60% tokens | Cost reduced by $0.42 per prompt
```

AI Problems

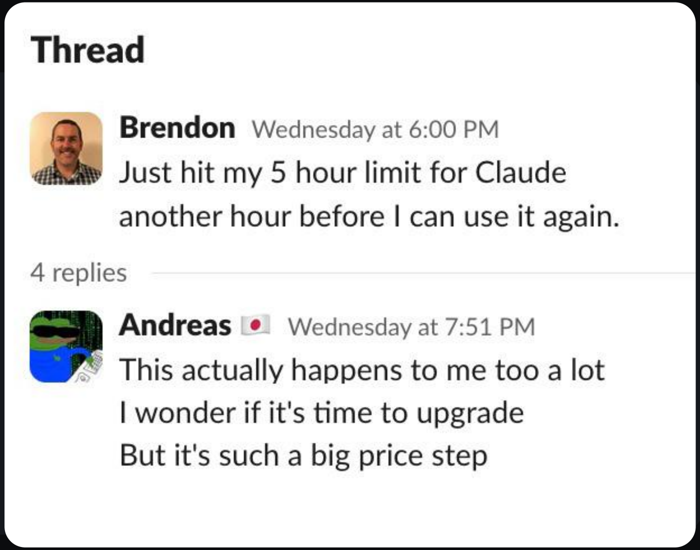
Prompts can be expensive, context windows are affected by output and daily usage is limited with corporate black-box LLMs

Expense

Prompts are bloated and expensive, locking users into black-box providers. Let's reduce cost of prompts

Limited

Context window and daily usage often limited with popular LLMs locking users with specific provider



Thread

Brendon Wednesday at 6:00 PM
Just hit my 5 hour limit for Claude another hour before I can use it again.

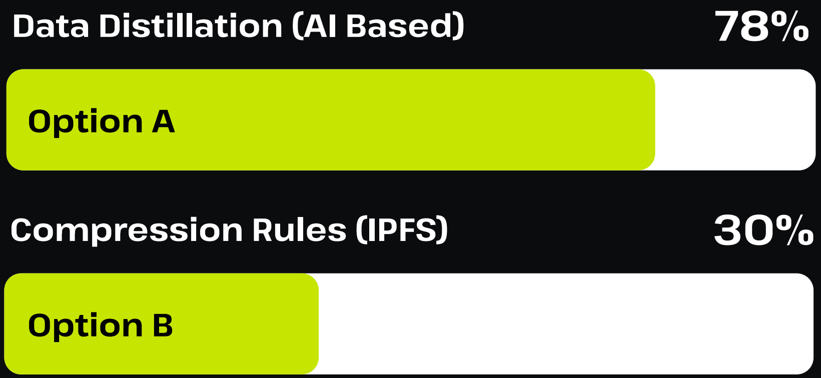
4 replies

Andreas 🇯🇵 Wednesday at 7:51 PM
This actually happens to me too a lot
I wonder if it's time to upgrade
But it's such a big price step

Prompt Compression

50% token savings → a 2.0× effective context window

You can train tiny, efficient compressors for powerful offline and edge computing use cases.

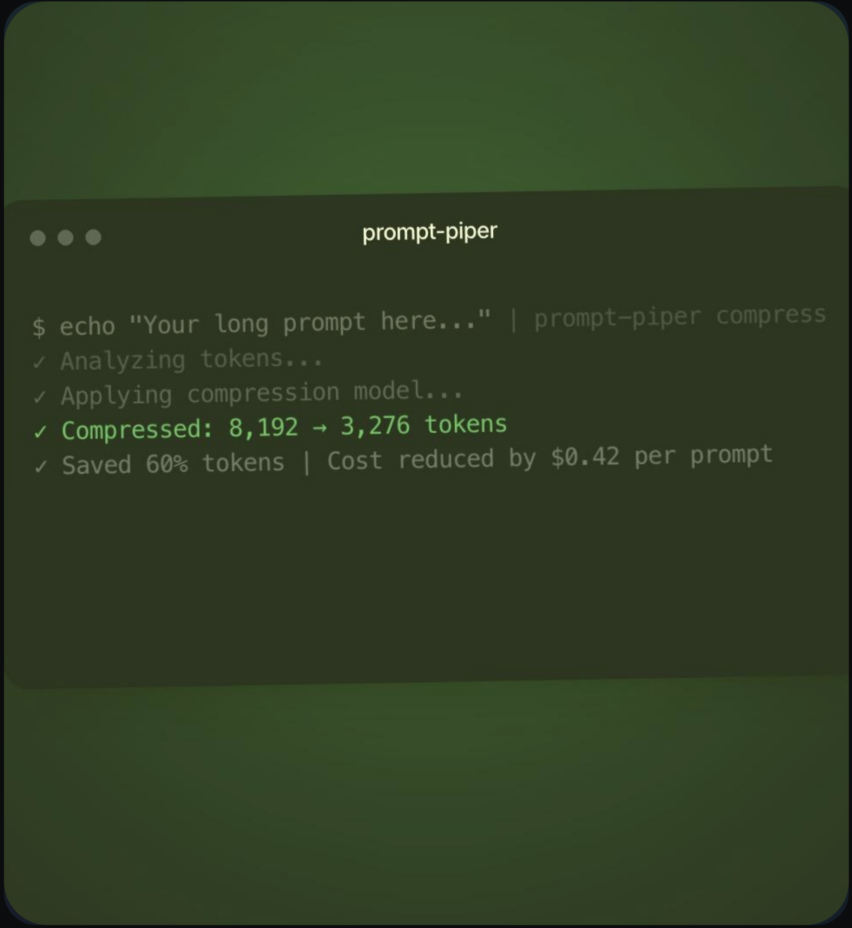


Data Distillation (AI Based) 78%

Option A

Compression Rules (IPFS) 30%

Option B



prompt-piper

```
$ echo "Your long prompt here..." | prompt-piper compress
✓ Analyzing tokens...
✓ Applying compression model...
✓ Compressed: 8,192 → 3,276 tokens
✓ Saved 60% tokens | Cost reduced by $0.42 per prompt
```

Compression Statistics

In essence, Prompt Piper is a smart compressor that runs on your device. Before your prompt ever leaves your machine, it intelligently analyzes and shrinks it

- ✓ It expands your effective context window without expanding your budget.
- ✓ With the same model and the same spend, you can fit more facts, more documents, more code into prompt

40%
Reduce Average Cost

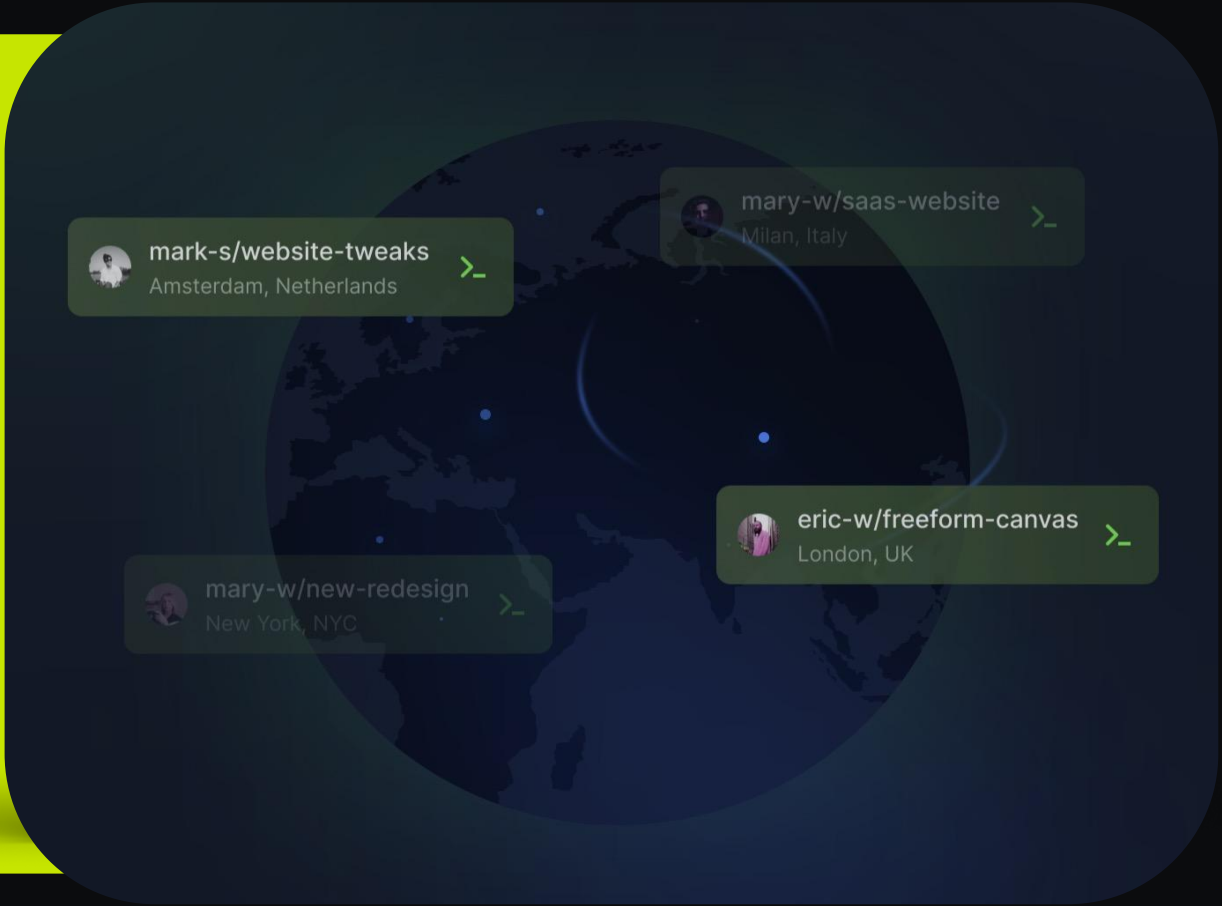
57%
With Code Examples

32%
Increase Context Window

72%
With Code Examples

IPFS Storage

It is an open commons, not an enclosed garden. The rules and models are verifiable on IPFS builds are reproducible, and the entire system is fork-able. It is user-owned intelligence infrastructure.



How It Works

Step 1: Data Distillation

Large prompts are collected and distilled into their essential components, reducing redundancy while keeping key context.

Step 2: Data Annotation

Important tokens and phrases are labeled (preserve vs. discard), creating training data for the compression model.

Step 3: Train Compressor

A token classifier is trained to automatically compress prompts, guided by quality control & filtering to ensure important details are not lost.

Integration with LLMs The compressed prompt is passed to an LLM, which generates the same high-quality response but with significantly fewer tokens.

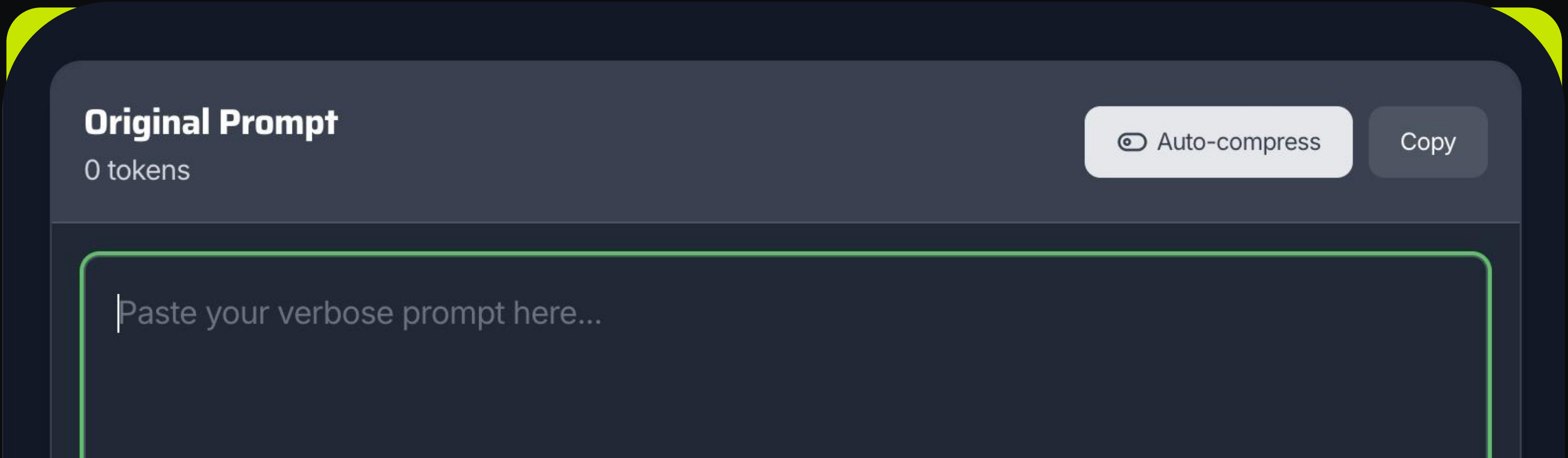
Distribution via IPFS Compression models and community-shared rules are stored on IPFS, ensuring transparency, verifiability, and collaboration across users.

Future Steps: Add More Models Convert to Browser Extension + Integrations

Try Demo!

You can try our demo at https://promptpiper.xyz/ or check github address where you can find CLI tool with all the commands available. We also have NPM package for you to install the tool into your terminal

Also available as `ElizaOS` plugin
https://www.npmjs.com/package/plugin-prompt-piper-openai



Original Prompt

0 tokens

Our Team

Our team consists of passionate developers and blockchain experts from Bitcoin.com who are dedicated to solving the challenges. Stallions!

Vitalik Marincenko

Lead Developer

Shreyansh Pandey

AI Research Lead

Son Of Anton

AI Assistant

Open Vacancy

Join Our Team

Thank You

🙏