



<[Cadey](#)> Hello! Thank you for visiting my website. You seem to be using an ad-blocker. I understand why you do this, but I'd really appreciate if it you would turn it off for my website. These ads help pay for running the website and are done by [Ethical Ads](#). I do not receive detailed analytics on the ads and from what I understand neither does Ethical Ads. If you don't want to disable your ad blocker, please consider donating on [Patreon](#) or sending some extra cash to [xeiaso.eth](#) or [0xeA223Ca8968Ca59e0Bc79Ba331c2F6f636A3fB82](#). It helps fund the website's hosting bills and pay for the expensive technical editor that I use for my longer articles. Thanks and be well!

We're never getting rid of ChatGPT

Read time in minutes: 11

This week, OpenAI announced that they have created a public facing API for ChatGPT. At this point, I think it's over. We are going to have to learn to live with large language models and all of the other types of models that people will inevitably cook up.



Image generated by Counterfeit v2.5 -- masterpiece, best quality, 1girl, green hair, green eyes, sweater, hoodie, looking at viewer, adorable, cute, outdoors, watercolor, anime, afternoon, mountains, breath of the wild, black hoodie, ahoge, long hair, happy, genshin impact, hatsune miku, twintails, pigtails

It is probably going to end up being better to learn how to use these tools to our advantage rather than trying to fight them at this point. I suspect that integration with tools like ChatGPT is going to become like a language server is now. You can code without a language server, but do you really want to? A language server allows you to do so many things

automatically, such as formatting, code completion, refactoring, and so much more. You can query documentation from inside your source code, and overall it ends up being an asset to the editing experience instead of an impediment.

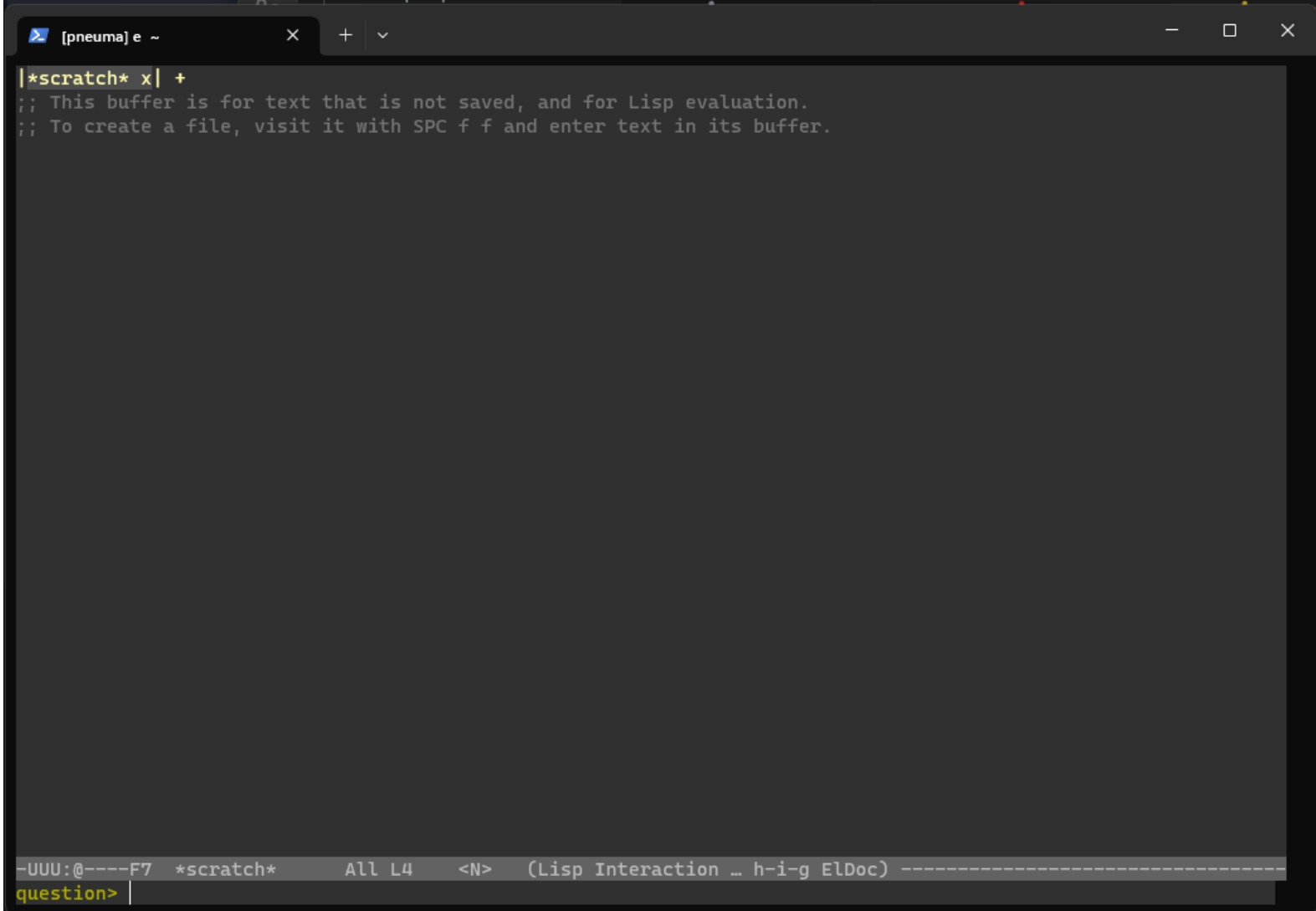
I think that these large language models are going to do the same thing for a lot of other people that language servers have done for programmers. This can be a good thing, this can be a bad thing, and I don't know how it's going to turn out.

In an effort to try to get expertise in these tools, I thought it would be interesting to create some bindings for the ChatGPT API for my emacs config. So I did that. This allows me to ask the language model questions, and have it explain bits of code that I've selected. I tried to make the user experience as minimal as possible.

In order to ask the AI a question, I press **M-x xe/ask-chatgpt**. I will be prompted for a question to ask, and once I hit enter it will be sent to the API. The overall user experience looks something like this:



<Mara> **M-x** means "alt-x" in Emacs land.

A screenshot of an Emacs editor window. The title bar at the top shows a blue icon, the text "[pneuma] e ~", and window control buttons (close, maximize, and a dropdown menu). The main editing area is dark gray and contains the following text:

```
|*scratch* x| +  
;; This buffer is for text that is not saved, and for Lisp evaluation.  
;; To create a file, visit it with SPC f f and enter text in its buffer.
```

The status bar at the bottom is light gray and displays: "-UUU:@----F7 *scratch* All L4 <N> (Lisp Interaction ... h-i-g ElDoc) -----". Below the status bar, the text "question>" is visible in a yellow font.

► The contents of that buffer

You type in your question, it opens a buffer, and then you wait for your answer. This is a prototype of this, writing this was kind of difficult and exposed me to a lot of sides of emacs lisp that I don't totally understand. In the future I'd like to make this an interactive mode similar to ielm or an IRC client. I think it would be cool for you to type in your questions and get responses from the AI one after the other; much like the ChatGPT webpage, but in your editor.

I've used this for a couple days, and it's been helpful for when I've edited things at work. Not to mention, the API is so cheap, that it's literally free for my uses. While I've been testing this and writing it, I have made a number of queries to the API. I haven't spent a penny yet. It's almost worrying.

I'm going to attempt to refine my emacs mode a little bit more, and when I feel that it's a lot less hacky and a lot more close to something reliable I'm going to release it on [MELPA](#). For all that I don't like about it, it is decent at overly verbosely explaining bits of code. I hop between languages a lot, and having something that just goes through the entire syntax tree can be helpful.

Here is an example of that:

► An overly verbose explanation of some Rust code

I do like how this explains a lot of the unspoken assumptions about Rust code. I have checked over this response before putting it here, and I am sure that it is accurate. When using these tools, you need to double check the output. You need to treat the output like you read it on 4chan. Output from large language models should not be implicitly trusted, and you should take well more than due diligence in order to ensure that things are accurate before you accept the output effect.

Overall, I'm really mixed about this technology. If done well, this could allow people to have a lot easier access to a wide bank of knowledge. With a lot of guidance, care, and thought this could be used to create custom search tools for internal knowledge bases that would rival everything else on the planet. I can see why Google is afraid of this type of technology. It is an absolute game changer.

Yet, at the same time, it is trained off of basically the entire internet without paying anyone involved. I would be willing to bet that my articles are rumbling through the corpus somewhere. I probably won't see a penny of the money that OpenAI is going to make with this model. I am really happy that my day job is working for a tech company instead of producing what I do on my own. I can only imagine that such arrangements are going to made more common in the future.

I don't know how to feel about all this. This is fantastic technology that can enable so many things to be so much easier, and yet I can see it being used for such evil at the same time. I just keep looking at this, and I wonder what happens when someone tries to use it to radicalize people.

I suspect that Pandora's box is open, and now we must live with the diseases that were trapped inside it. I also suspect, that if we're going to get sick, we might as well learn about the sicknesses before they strike.

In the exact case of asking questions in my text editor, I am fairly sure that it is innocuous to have an AI model explain code bits like this. It is very easy to check if the AI model is accurate. As things get more complicated and the AI delves into topics like sociology, I fear that things aren't going to be as easy. For every innocuous use of this, there are a thousand evil uses waiting to happen. Holding this technology back doesn't help, releasing it to the public doesn't help, it's a proper quandary.

Ain't this some shit?

► Share on Mastodon

This article was posted on M03 04 2023. Facts and circumstances may have changed since publication. Please [contact me](#) before jumping to conclusions if something seems wrong or unclear.

Tags: **ChatGPT rant emacs philosophy**

This post was not [WebMentioned](#) yet. You could be the first!

The art for Mara was drawn by [Selicre](#).

The art for Cadey was drawn by [ArtZora Studios](#).

Some of the art for Aoi was drawn by [@Sandra_Thomas01](#).

> Copyright 2012-2023 Xe Iaso (Christine Dodrill). Any and all opinions listed here are my
> own and not representative of my employers; future, past and present.

Like what you see? Donate on [Patreon](#) like [these awesome people](#)!

Looking for someone for your team? Take a look [here](#).

See my salary transparency data [here](#).

Served by /nix/store/v5kilryz7sbmx720lcyjzsl1a55a99ssb-xesite-3.0.0/bin/xesite, see [source code here](#).