O'REILLY®

What Are AI Agents?

When and How to Use LLM Agents

Benjamin Labaschin

REPORT

What Are AI Agents?

When and How to Use LLM Agents

Benjamin Labaschin

O'REILLY®

Beijing · Boston · Farnham · Sebastopol · Tokyo

What Are AI Agents?

by Benjamin Labaschin

Printed in the United States of America.

Published by O’Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O’Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://oreilly.com*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

- Acquisitions Editor: Nicole Butterfield
- Development Editor: Virginia Wilson
- Production Editor: Katherine Tozer
- Copyeditor: nSight, Inc.
- Interior Designer: David Futato
- Cover Designer: Karen Montgomery
- Illustrator: Kate Dullea

- November 2023: First Edition

Revision History for the First Edition

- 2023-11-15: First Release

The views expressed in this work are those of the author and do not represent the publisher's views. While the publisher and the author have used good faith efforts to ensure that the information and instructions contained in this work are accurate, the publisher and the author disclaim all responsibility for errors or omissions, including without limitation responsibility for damages resulting from the use of or reliance on this work. Use of the information and instructions contained in this work is at your own risk. If any code samples or other technology this work contains or describes is subject to open source licenses or the intellectual property rights of others, it is your responsibility to ensure that your use thereof complies with such licenses and/or rights.

978-1-098-15971-9

[LSI]

What Are AI Agents?

The last century in the history of computing has had many notable milestones: the invention of the computer; the development of the personal computer as we know it; the internet; the smart phone; machine learning; and cloud computing. Every few decades it seems societies are thrust forward, riding the wave of some new computational innovation. If you are reading this, then congratulations (or I'm sorry?), you are living during another one of these milestones in computational advancement.

Artificial intelligence (AI) agents, as powered by large language models (LLMs) and the user data they are provided, have emerged in recent years as powerful new tools in humanity's computational repertoire. But what are AI agents? And how can we be so certain AI agents will be so impactful? Readers of this report will not only learn the answers to these questions but will also be exposed to such insights as *when* to use AI agents and, critically, *how* to get started!

What Are AI Agents?

So, before we get ahead of ourselves, what exactly are AI agents, and why should you want to learn about them in the first place? AI agents are tools designed to allow users to interact with LLMs to achieve a more productive or creative workflow as seamlessly as possible. Before AI agents, users would be forced to build their own statistical language models—a time-consuming, technical, and expensive endeavor! Now, with AI agents, users who want to interact with AI simply get to log in to an interface and conduct business ranging from asking questions of their documents to getting help with their homework.

At a more granular level, you might think of AI agents as UI "wrappers" around the models that power them. That is to say, AI agents are often user-friendly "frontends" that make using the models that fuel them easier, often by focusing and limiting just *how* users interact with the model. Take ChatGPT, for instance. The models fueling ChatGPT (GPT-3.5 Turbo or GPT-4) are massively complex, powerful, and difficult to use and operate on their own. As an AI agent, ChatGPT abstracts away these models' technical features and allows users to interact with them simply via text.

How Do AI Agents Relate to LLMs?

All right, but what are these complicated "LLMs" that we are talking about? LLMs are the brains of the operation—the "AI" in AI agents. These models are trained to extrapolate and interpret natural language text in the context that you provide them. Some are better at math, others are better at jokes or speaking Spanish, and some are general purpose. In this way, you can think of AI agents as cars, while the models that fuel them can be thought of as their engines. How so? Let's break it down.

We all know that cars can't move forward without an engine, and we also know that some engines are designed for specific use cases—such as ATV engines versus school bus engines. Well, like cars, AI agents cannot function without their underlying LLM engines. And like cars, some AI agents often work better with particular "engines" that suit their purposes better. For example, maybe an AI agent is assisting you with writing code, such as GitHub Copilot. So it would be nice if the agent you're using—the means through which you're interacting with the model to become more productive—were fueled by an LLM built for the purpose of code assistance. And indeed that's exactly what the machine learning engineers who build these models do: they build engines fit for purpose.

Ultimately, this is what you need to know about LLMs and their relationships to AI agents: like driving a car, users of AI agents do not need to understand *how* the car's engine works to drive it. Drivers simply need to know how to use the car itself. Similarly, the best AI agents are intuitive, effective for their use case (such as assisting with writing code), and have appropriate LLMs operating as the engine for their users.

In fact, that is exactly why there has been such a push to create AI agents recently. Business people, machine learning engineers, and AI researchers understand that in order for their LLMs to become ubiquitous and beneficial to humanity, AI agents have to be developed for users to be able to use them.

When to Use AI Agents

All right, so you now have a general understanding of what AI agents are, but you're still likely wondering when you should use them and what tasks AI agents are useful for. While there are new AI agents and use cases emerging almost daily, here are a few common situations in which you may want to begin using AI agents.

Document Question and Answer Agents

Document Q&A agents are AI-enabled tools designed for the problem of information glut. With a document Q&A agent, users are able to simply ask their document questions and receive answers about exactly what they contain, without engaging in the potentially laborious effort of searching through them manually. If this sounds too abstract, that's OK! Let's work through a use case together.

Imagine you've just been hired as a new employee at Very Fake Company, Inc. It's your first day. As you sit down at your desk to start your day, your manager walks up to you, drops a giant stack of documents on your desk, and says, "HR is out and I'm too busy to onboard you. Here are all the onboarding documents you'll need to understand how to work here at Very Fake Company, Inc. Enjoy!" You *could* start reading all those documents, but honestly, it sounds like a time-consuming task. Not to mention that this isn't your first rodeo: you know that a lot of that information (such as "what email is") won't be useful to you.

In such a situation, a document Q&A agent could make your life far easier. Document Q&A agents ingest documents, such as the papers you were given on your first day, and store them in memory. But they do more than *store* information; they also are able to consolidate and regurgitate relevant information back to their users through a natural language interface. All you have to do is ask your document Q&A agent a question, and it will return the requested information, as you can see in this example interaction:

User: Hello, can you help me consolidate the onboarding information I received on my first day at Very Fake Company, Inc.?

Document Q&A Agent: Of course! Just provide me with the contents of the documents, and I'll assist you.

<uploads the stack of onboarding documents>

Document Q&A Agent: Documents received. Please give me a moment to process them... Done! You may ask a question about these documents now.

User: Please summarize for me all critical information I need to start my first day at Very Fake Company, Inc.

Document Q&A Agent: Absolutely! Here's a condensed summary of the essential details:

Company Values: At VFC Inc., we prioritize integrity, teamwork, innovation, and customer satisfaction.

Working Hours: Monday to Friday, 9 am - 5 pm. You're expected to log in to our time tracking system each day.

Dress Code: Casual, but presentable. Fridays are company-branded attire days.

IT and Login Details: Your company email is john.doe@vfcinc.com. In order to access the Very Fake Company, Inc code editor needed to do your job, you will need to reach out to Janice Smith in IT.

HR Details: For any HR-related queries, reach out to Emma at extension 123. Remember to fill out your health insurance and emergency contact forms by the end of the week.

Parking: Spot #45 has been assigned to you. Remember to display your parking pass.

Company Events: There's a welcome mixer this Friday at 6 pm in the lounge. It's a great

> *opportunity to meet your coworkers.*
>
> *First Tasks: In your first week, your manager would like you to translate legacy code written in the python programming language into the Java programming language.*
>
> *Would you like more detailed information on any of these topics?*
>
> *User: That's perfect, thank you! I'll reach out if I have more questions.*
>
> *Document Q&A Agent: You're welcome! Have a great first day at Very Fake Company, Inc. Let me know if you need anything else.*[1]

If you are ever pressed for time and have a lot of information to digest, document Q&A agents may be the tool for you.

Chatbot Agents

Like their document Q&A counterparts, chatbot agents are capable of ingesting natural language inputs and responding with relevant output. Unlike document Q&A agents, however, chatbot agents are far more dynamic in their capabilities. Whereas the focal point of document Q&A agents is question and answering from source material that *you* provide them, chatbot agents are fed "prompts" ahead of time, before their users ever begin interacting with them. Prompts are just what they sound like: instructions that agents are "prompted" to follow at all times as they assist users.

Let's continue with the Very Fake Company scenario to look at a chatbot agent use case. Imagine that you've finished getting all the first-day information you need from the document Q&A agent. As part of the critical information relayed to you by the Q&A agent, you're required to reach out to IT to activate Very Fake Company's proprietary code editor software. But when you walk up to IT, they say they're *also* swamped and can't help you either. Instead, they've asked you to contact their IT chatbot agent BERTA: Bother Energetic Robot To Activate. It turns out IT has gotten into the AI agent game and built BERTA as an AI to automatically activate the company's software for users. IT wrote the following prompt for BERTA:

> *User: You are a chatbot agent. Your name is BERTA: Bother Energetic Robot To Activate. Your only objective is to assist users at Very Fake Company, Inc. Your primary role is to take in user email information, verify they are employees at the company, and then check whether they are authorized to use our proprietary code editor software by checking the internal database you have been given access to. If the user is permitted to access the coding software, you should activate their permission in the database and send them the key they need to get started. If they are not permitted, tell them to email Janice Smith at janice.smith@vfcinc.com.*
>
> *BERTA: Great! I will be sure to follow these instructions.*

BERTA has been instructed to follow the guidelines before ever interacting with users. So, as an employee at Very Fake Company, Inc., when you log on to the BERTA software and provide your email, BERTA will already know exactly how to interact with you—by following the parameters initially set by IT.

Chatbot agents are particularly compelling in cases where connecting to third-party systems such as databases and the internet is useful. This makes chatbot agents potentially far more powerful than Q&A agents due to the wider-range of product possibilities they enable. Unlike brittle, nonagent chatbots of the past, chatbot agents are also more capable of interacting with users in a more dynamic and potentially more inclusive way. Chatbot agents can be prompted to communicate in multiple languages, cope with ornery users, and answer precisely the questions users ask of them. They are also capable of being accessed 24/7, allowing for an improvement in user feedback.

Ultimately, chatbot agents are appropriate for any situation in which users seek information that may not be whittled down to source material like onboarding documents. Chatbot agents are also a great choice when companies anticipate being contacted more frequently and in greater numbers than they might otherwise be able to handle.

Code Assistant Agents

Code assistant agents are AI agents that are fueled by models specifically designed to help users like you write code more productively and efficiently. Popular code assistant agents include GitHub Copilot, Amazon CodeWhisperer, and Hugging Face’s StarCoder. Code assistant agents are tools designed to edit error-ridden code, autocomplete simple functions to common coding problems, or design templates for more difficult coding problems. Whereas in the very recent past, software engineers and developers might resort to search engines, chatrooms, or colleagues to solve their problems, code assistant agents reduce the need for such costly context switching, allowing for large improvements in productivity.

Imagine your boss at Very Fake Company, Inc., asks you to translate legacy code from Python into Java. The problem is, while you know Python, you’re not that familiar with Java. Luckily, as you log in to Very Fake Company, Inc.’s code editor, you see that it comes equipped with a code assistant agent. As you log in to the repository that contains the legacy Python code you’ve been instructed to change, you see a Python function that converts marketing copy from lowercase, to all uppercase:

```
def convert_marketing_copy(text: str) -> str:
    """This function takes marketing text and converts it all to
    uppercase."""
    return text.upper()

>>> convert_marketing_copy("I love Very Fake Company, Inc.")
"I LOVE VERY FAKE COMPANY, INC."
```

Now, you *could* search the internet for Java resources, read up on Java, and slowly write out a function yourself, or you could instruct a code assistant agent to write the function for you. You opt for the latter. In your editor, you connect to the code assistant agent and write out the beginning of the following code that appears in black. The code assistant agent, which is embedded in your text editor, picks up anything you write and processes input in real time, like a spell checker. It offers you the autocompleted function in blue. By pressing Tab, your code assistant agent provides you with the fully completed function of Java code, and all you had to do was simply set up the problem and describe what you wanted:[2]

```
public class MarketingCopyConverter {

  // Function completed with code completion software
  /**
   * This function takes marketing text and converts it all to
   * uppercase.
   * For example, the text "I love Very Fake Company, Inc."
   * is converted in the function convertMarketingCopy().
   * @param text The marketing text to be converted.
   * @return The converted uppercase string.
   */

      public static String convertMarketingCopy(String text) {
          return text.toUpperCase();
      }
      public static void main(String[] args) {
          // Main method completed with code completion software
          String result = convertMarketingCopy(
              "I love Very Fake Company, Inc.");

          System.out.println(result);
          // Output: I LOVE VERY FAKE COMPANY, INC.
      }
}
```

Code assistant agents allow programmers to complete their work more quickly and efficiently. But they are not without their costs. Like all AI agents, code assistant agents are tools, and as with any tool, you must know when and how to wield them. Code assistant tools can easily generate wrong or misguided output that can cause problems to snowball. It is important when using code assistant agents, or any agent for that matter, to verify their output with human oversight and ultimately to ensure they are tools that work for you rather than the other way around. The best way to achieve this is to understand *how* to use these tools.

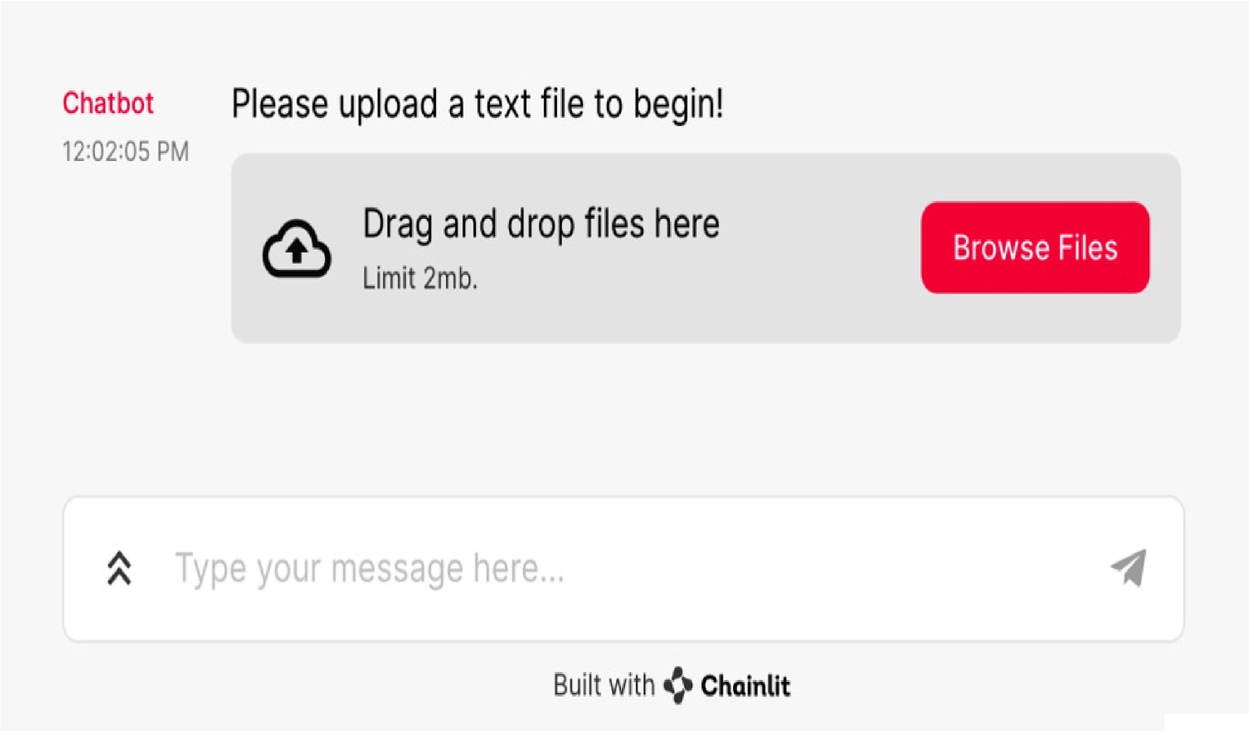
AI Agent Tools and How to Use Them

Thus far, you have read about using AI agents in realistic but imagined scenarios. While useful as a teaching device, this may lead you to believe that AI agents are abstract tools for the future and not of today. But nothing could be further from the truth. AI agents are *actual* tools that are really being used every single day by millions of people. And while it would take too much time to demonstrate all the AI agents available today, the following real-world AI agent software solutions correspond to the tools in the scenarios previously presented.

Document Q&A Agents with Chainlit and PandasAI

As we've discussed, document Q&A agents are particularly useful when you would like to query your own documents in order to summarize or extract pertinent information. One popular agent tool to enable this functionality is called Chainlit. The concept of Chainlit is simple: users provide information through a Chainlit-enabled UI. After the data is processed by an underlying LLM, the agent will then be prepared to answer real questions from users like you about the data it was provided. In this case, I've leveraged a tool called PandasAI to handle the data manipulation, which is fueled by an LLM under the hood, but you could have easily switched it out for another software solution.

This process is demonstrated in Figure 1. When you log in to Chainlit, your agent prompts you to provide it with data. I've provided the agent with a file containing information on popular computer science books.



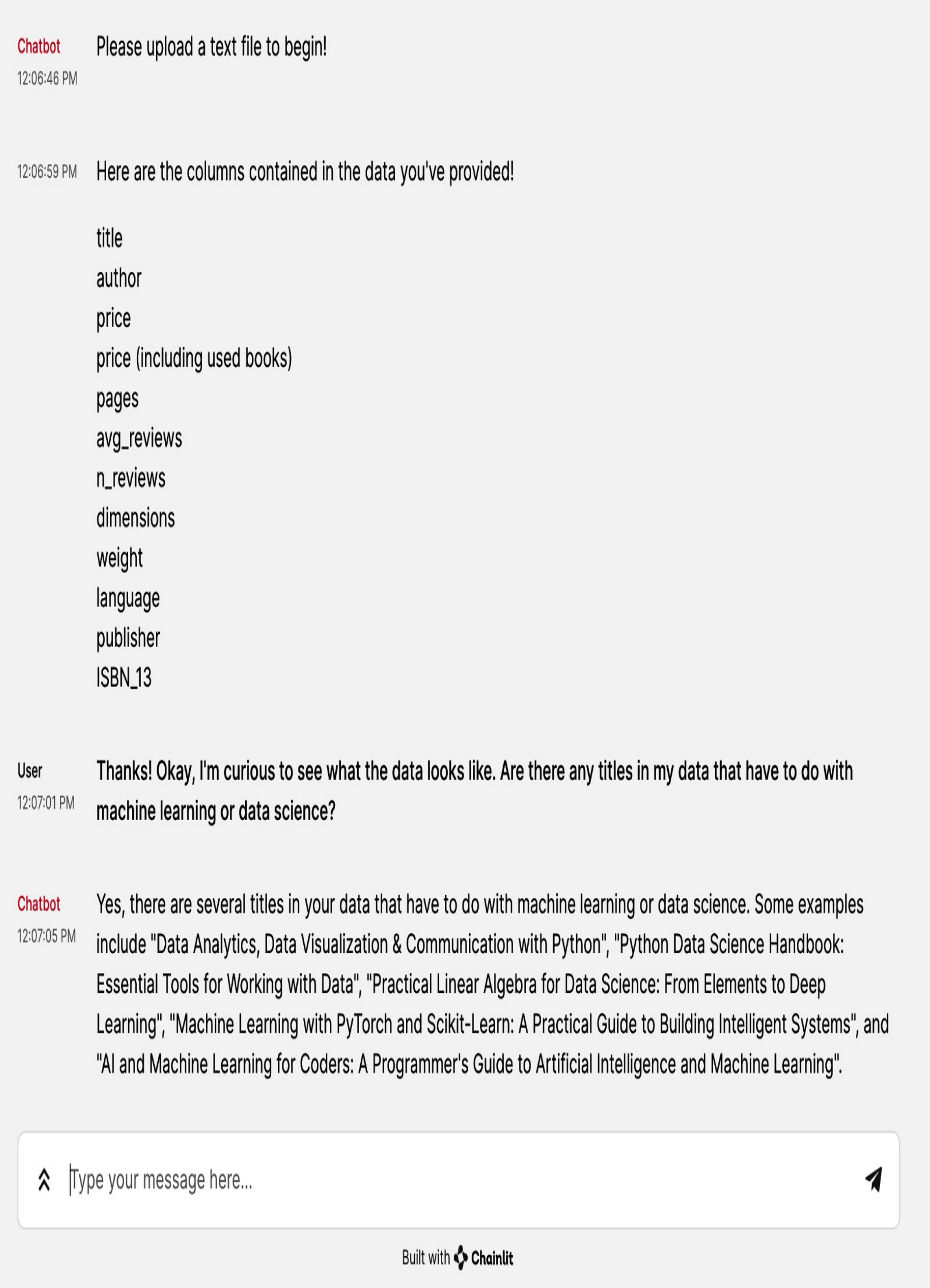
Chatbot

12:02:05 PM

Please upload a text file to begin!

Figure 1. Initial prompt asking for documents[3]

As Figure 2 shows, the agent easily ingests the information and returns the names of the columns in the file.



Chatbot
12:06:46 PM
Please upload a text file to begin!

12:06:59 PM
Here are the columns contained in the data you've provided!

title
author
price
price (including used books)
pages
avg_reviews
n_reviews
dimensions
weight
language
publisher
ISBN_13

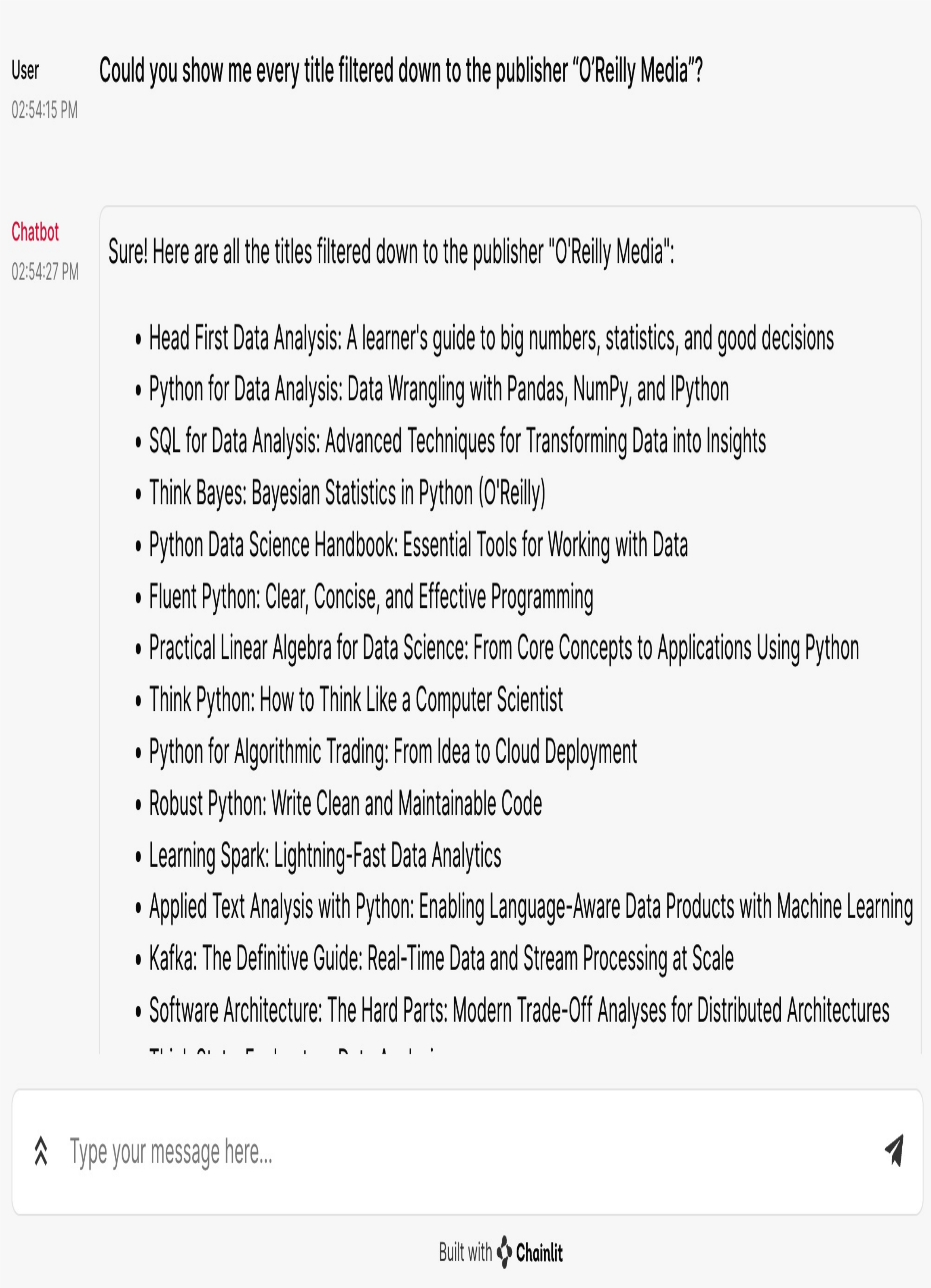
User
12:07:01 PM
Thanks! Okay, I'm curious to see what the data looks like. Are there any titles in my data that have to do with machine learning or data science?

Chatbot
12:07:05 PM
Yes, there are several titles in your data that have to do with machine learning or data science. Some examples include "Data Analytics, Data Visualization & Communication with Python", "Python Data Science Handbook: Essential Tools for Working with Data", "Practical Linear Algebra for Data Science: From Elements to Deep Learning", "Machine Learning with PyTorch and Scikit-Learn: A Practical Guide to Building Intelligent Systems", and "AI and Machine Learning for Coders: A Programmer's Guide to Artificial Intelligence and Machine Learning".

Figure 2. Provided documents, agent answers question[4]

You can then follow up with the agent by asking it for information on titles having to do with data science and machine learning. The agent passes the text to PandasAI, which leverages an LLM to respond with the correct manipulations needed of our data. The resulting response is correct: you are provided a list of machine learning and data science books.

And if that’s not enough, you are free to continue pressing the agent for more relevant information, as seen in Figure 3, in which you are able to restrict data to a single publisher, O’Reilly Media.



User
02:54:15 PM

Could you show me every title filtered down to the publisher “O’Reilly Media”?

Chatbot
02:54:27 PM

Sure! Here are all the titles filtered down to the publisher "O'Reilly Media":

- Head First Data Analysis: A learner's guide to big numbers, statistics, and good decisions
- Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython
- SQL for Data Analysis: Advanced Techniques for Transforming Data into Insights
- Think Bayes: Bayesian Statistics in Python (O'Reilly)
- Python Data Science Handbook: Essential Tools for Working with Data
- Fluent Python: Clear, Concise, and Effective Programming
- Practical Linear Algebra for Data Science: From Core Concepts to Applications Using Python
- Think Python: How to Think Like a Computer Scientist
- Python for Algorithmic Trading: From Idea to Cloud Deployment
- Robust Python: Write Clean and Maintainable Code
- Learning Spark: Lightning-Fast Data Analytics
- Applied Text Analysis with Python: Enabling Language-Aware Data Products with Machine Learning
- Kafka: The Definitive Guide: Real-Time Data and Stream Processing at Scale
- Software Architecture: The Hard Parts: Modern Trade-Off Analyses for Distributed Architectures
- Think Stats: Exploratory Data Analysis

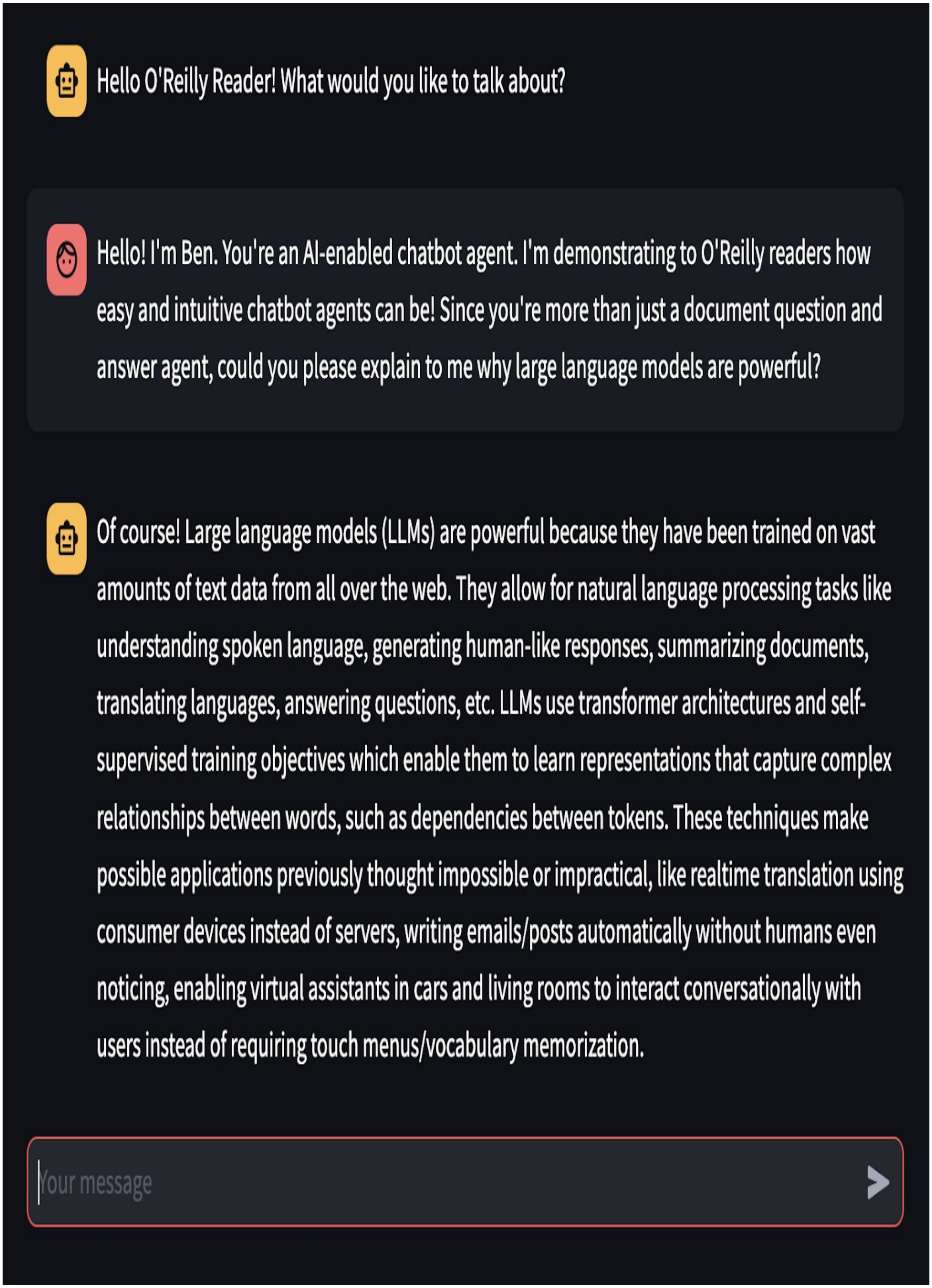
Figure 3. Exploring data by filtering information to O’Reilly[5]

If you’re interested in learning how to implement this code yourself, you can check out the code for this example here or check out Hands-On Large Language Models by Jay Alammar and Maarten Grootendorst (O’Reilly).

Dynamic Chatbot Agents with HuggingFace Chat, Llama 2, and Streamlit

There are many general purpose chatbot agents on the market today, including OpenAI’s ChatGPT and Anthropic’s Claude. But there are also chatbot agents powered by open source LLMs like Meta’s LLaMa 2, which is used in Figure 4.

Chatbot agents tend to have a common user interface. First is the “assistant” persona, represented by the yellow icon in Figure 4. This persona is played by the chatbot agent. Assistants typically start all new interactions with their users by asking how they can help them. In the case of the chatbot in Figure 4, the assistant persona asks you, the O’Reilly reader, what it is you’d like to talk about. The second persona is the “user” persona, represented by the red icon. Users can follow up by inputting just about any question or comment they want to the assistant—the possibilities are about as unlimited as any conversation you could have with a human.



Hello O'Reilly Reader! What would you like to talk about?

Hello! I'm Ben. You're an AI-enabled chatbot agent. I'm demonstrating to O'Reilly readers how easy and intuitive chatbot agents can be! Since you're more than just a document question and answer agent, could you please explain to me why large language models are powerful?

Of course! Large language models (LLMs) are powerful because they have been trained on vast amounts of text data from all over the web. They allow for natural language processing tasks like understanding spoken language, generating human-like responses, summarizing documents, translating languages, answering questions, etc. LLMs use transformer architectures and self-supervised training objectives which enable them to learn representations that capture complex relationships between words, such as dependencies between tokens. These techniques make possible applications previously thought impossible or impractical, like realtime translation using consumer devices instead of servers, writing emails/posts automatically without humans even noticing, enabling virtual assistants in cars and living rooms to interact conversationally with users instead of requiring touch menus/vocabulary memorization.

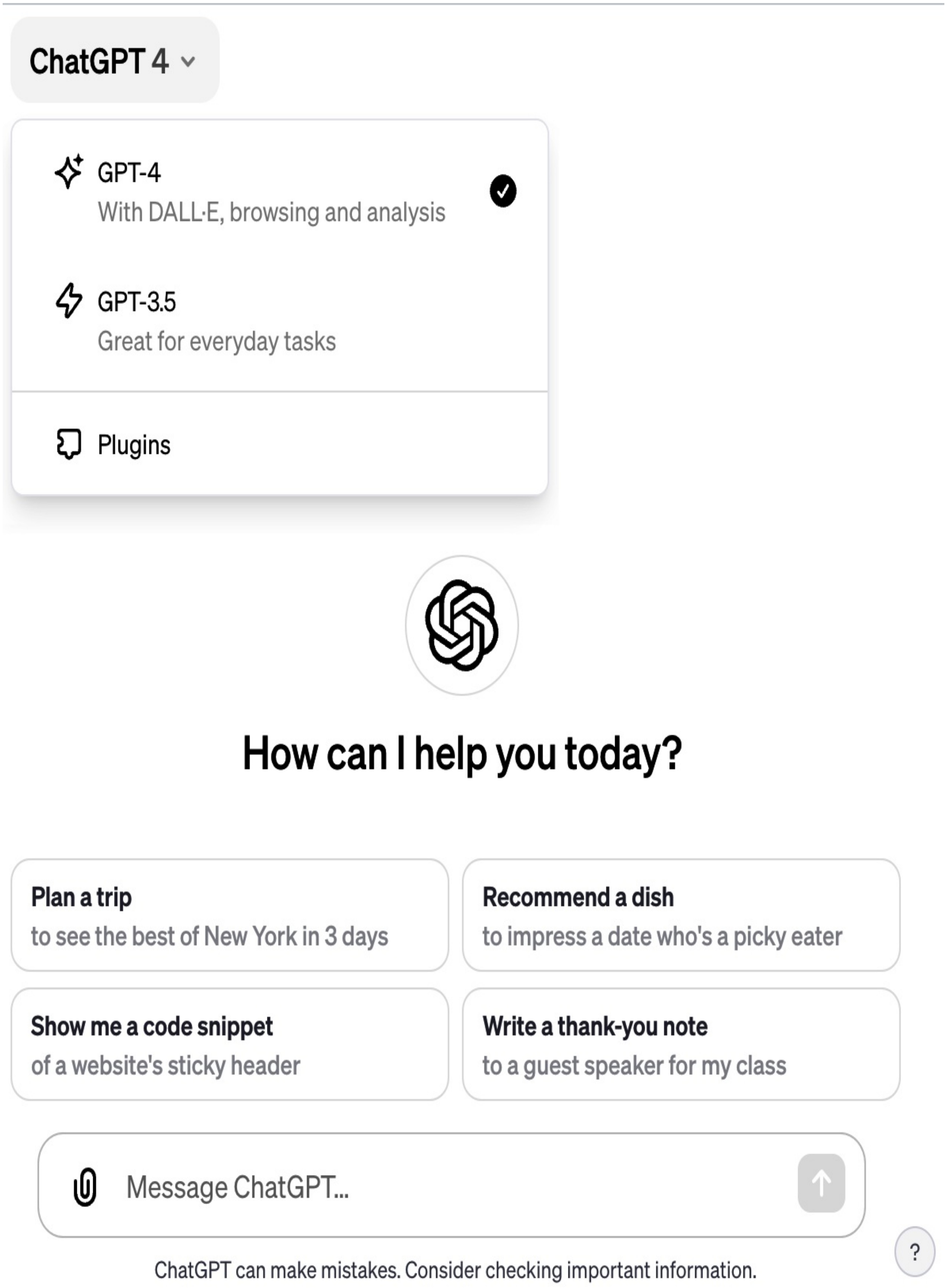
Figure 4. Chatbot Agent, HuggingFace, and LLaMa 2 example[6]

But if you're looking for the most useful responses possible from a chatbot agent, then it's best to be as explicit as possible with your words. For instance, in the Figure 4 chatbot demonstration, the agent is told exactly the role it should play—an "AI-enabled chatbot agent"—and that it should demonstrate its capabilities by having it explain why LLMs are so powerful. By being explicit with the instructions it's been given, the agent is able to structure its response into more useful output. If you would like to learn how this agent was created, see more on GitHub.

Coding Assistant Agents with OpenAI ChatGPT

Instead of creating your own coding assistant agent, this time let's use one out of the box with OpenAI's ChatGPT. In order to use this agent, you will have to sign up for ChatGPT Plus, which, as of this writing, costs about $20 US per month. Now, you may recall from earlier that a coding assistant agent is an AI agent that is specifically designed to help users with technical problems, such as translating code from Python to Java. But code assistant agents can do more than just interpret code; they can even take data you've provided with them and create interpretable charts for users. Let's see how.

In Figure 5, you start your interaction with an agent in what has likely become a familiar view. At the bottom of your screen is a text box in which you can interact with the ChatGPT agent, while the rest of the screen is empty—reserved for the assistant-user chat history. At the top of the screen, you are able to select between OpenAI's most advanced user-facing LLM (GPT-4) and its cheaper but faster LLM (GPT-3.5). After you've selected your preferred LLM, you are now able to submit data by either dragging and dropping files into the text box or pressing on the paperclip sign that will appear in the text box on your screen.

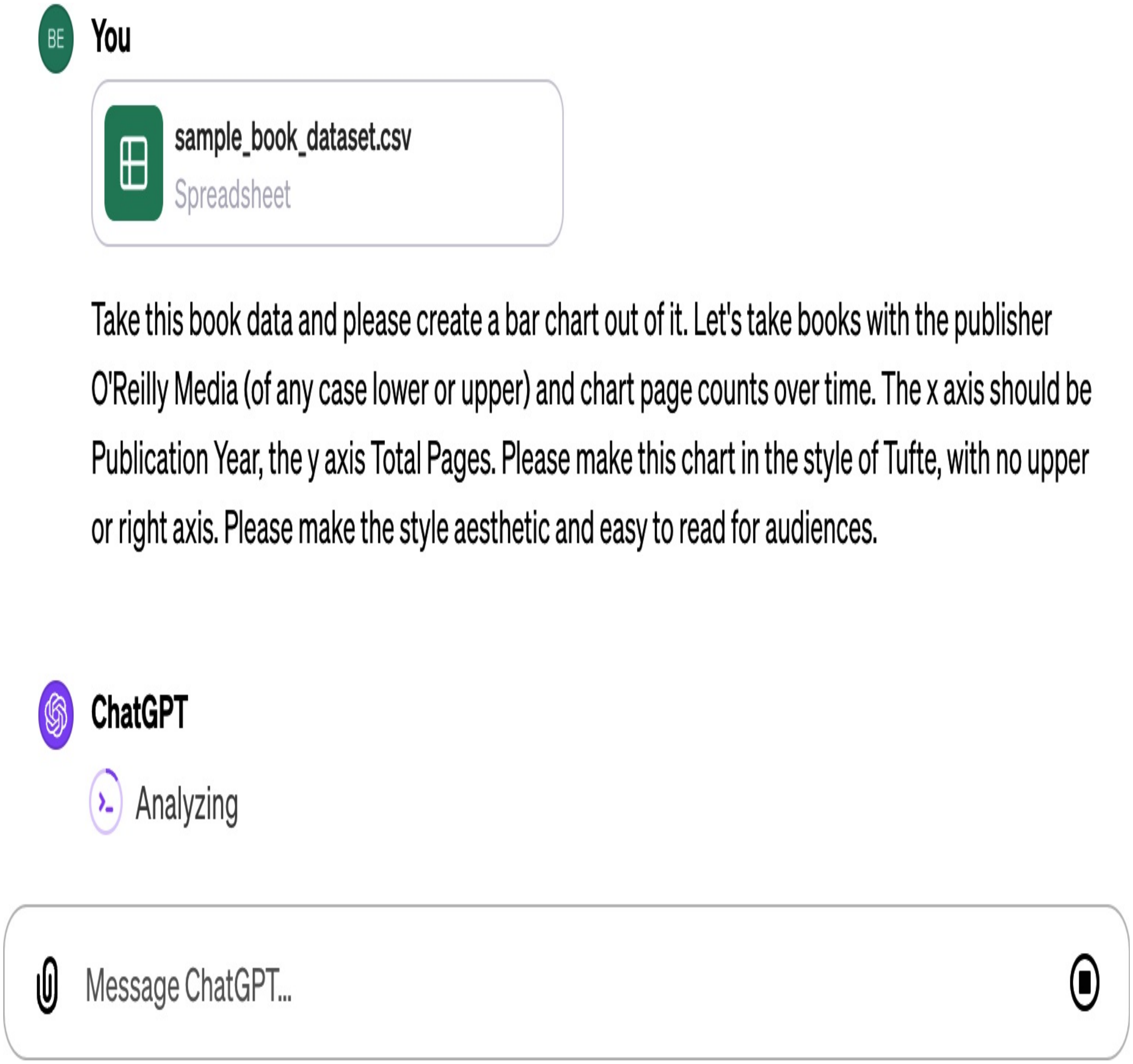

ChatGPT 4
GPT-4
With DALL·E, browsing and analysis
GPT-3.5
Great for everyday tasks
Plugins
Plan a trip
to see the best of New York in 3 days
Recommend a dish
to impress a date who's a picky eater
Show me a code snippet
of a website's sticky header
Write a thank-you note
to a guest speaker for my class
Message ChatGPT…
?

How can I help you today?

ChatGPT can make mistakes. Consider checking important information.

Figure 5. Accessing ChatGPT for coding assistance[7]

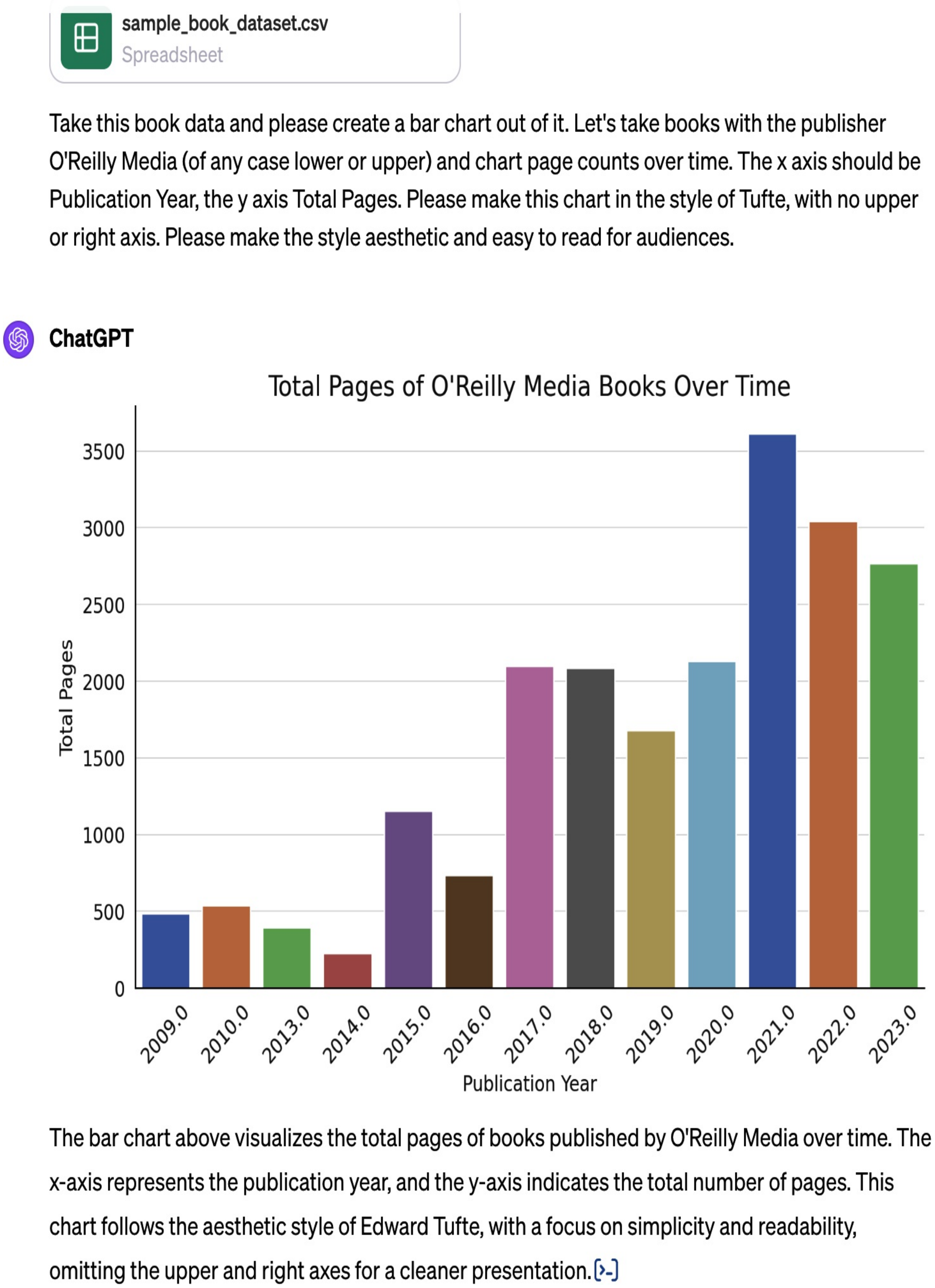
In Figure 6, the same book data from the Q&A agent example is used. This time, though, the code assistant agent is asked to interpret the data it’s provided and output a chart that summarizes how many pages per year O’Reilly books contain in the sample data. You can even go so far as to tell it to style the chart with the aesthetic advice as given by the American statistician Edward Tufte. The agent is familiar with Tufte and has no trouble taking that instruction and rolling with it.



Take this book data and please create a bar chart out of it. Let's take books with the publisher O'Reilly Media (of any case lower or upper) and chart page counts over time. The x axis should be Publication Year, the y axis Total Pages. Please make this chart in the style of Tufte, with no upper or right axis. Please make the style aesthetic and easy to read for audiences.

Figure 6. Providing a data file with chart instructions[8]

You can see the results of this interaction in Figure 7. The ChatGPT code assistant agent has very little trouble reading an entire file of data and generating an entirely new, accurate bar chart, following its instructions even down to aesthetic preferences. And this is only a sliver of the power that code assistant agents like ChatGPT can provide their users. It’s no wonder then why so many people, from venture capitalists and AI researchers to executive assistants and students, are so excited about AI agents.



sample_book_dataset.csv
Spreadsheet

Take this book data and please create a bar chart out of it. Let's take books with the publisher O'Reilly Media (of any case lower or upper) and chart page counts over time. The x axis should be Publication Year, the y axis Total Pages. Please make this chart in the style of Tufte, with no upper or right axis. Please make the style aesthetic and easy to read for audiences.

ChatGPT

The bar chart above visualizes the total pages of books published by O'Reilly Media over time. The x-axis represents the publication year, and the y-axis indicates the total number of pages. This chart follows the aesthetic style of Edward Tufte, with a focus on simplicity and readability, omitting the upper and right axes for a cleaner presentation. [>-]

Figure 7. Chart output by agent using provided CSV[9]

Moving Forward with AI Agents

By now you have seen a glimpse of what AI agents are capable of. But be assured, this is just the start. It’s been just less than a year since AI agents like ChatGPT were released to the public. Since then, large corporations such as Alphabet, Microsoft, Meta, Amazon, and Apple have all felt the pressure to integrate AI agents into their workflows, and to invest heavily into building LLMs of their own to power these agents. Meanwhile, upstart companies have been forming overnight to fill in the gaps that AI agents produce. AI agents aren’t always easy to build after all and can be computationally expensive to maintain. So, while the future looks bright for AI agents, they are not without their flaws or opportunities for growth. Here are a few agent-related issues and opportunities to keep an eye out for in the coming months.

Augmented Search

One of the most powerful aspects of AI agents is their ability to consolidate and reproduce relevant information. Search engines are a prime example of products that will be augmented by AI agents or even replaced altogether. In response to Microsoft releasing Bing AI, an AI-augmented search agent, Google released its own agent called Bard. But the question remains whether these agents are cost-effective software products. After all, search engines are typically funded by ad revenue.

Painting with a broad brush, in the search market, businesses bid in auctions to associate their goods or services alongside search keywords. For every click on an ad by a user, the search engine host gets paid (a pay-per-click model). Traditionally, in exchange for their free-to-use model, users implicitly accept being exposed to ads placed by businesses corresponding to keywords, time of day, etc. But what happens to the market for search when you can search exactly for the content you want, summarized, without having to click an advertisement at all? Certainly, there are market mechanisms that can ameliorate these issues: perhaps the authors of the articles that are summarized by search get paid fractions of a penny? The fact of the matter is, whatever mechanisms emerge, they will be different in the face of a new, more efficient means of search powered by AI agents.

Product Integration and Isolation

Search won't be the only area affected by AI agents. Many companies, including Notion, Salesforce, and Adobe, are integrating agents into their products, making the bet that these tools will garner them greater market share and better customer experience. Other companies are sticking to their guns and refusing to adopt AI agents into their products. So skewed is the divide between companies that invest in AI-based technology and those who eschew it, that recent estimates suggest only a quarter of companies with more than 250 employees have invested in AI of any kind. By comparison, that's three times the rate of firms with fewer than 10 employees —7.7%.

This imbalance in AI investment has left researchers to speculate that, if this trend continues, there will only be further separation between small and medium-sized firms and "superstar" firms such as Apple, Google, and Amazon. According to Erik Brynjolfsson, professor of human-centered AI at Stanford University and a contributing author to this research, at the current rates of investment into AI, "we are flying blind into what has been called the fourth industrial revolution."[10]

Over the coming months and years, it will be interesting to see whether investment into AI surges as the use of agents becomes normalized.

Infrastructure

Maintaining AI agents can be a costly affair. This is in large part due to the intensive computational nature of running the LLM engines that power these agents. But this is sure not to last. Over the past year, as interest in AI agents has risen exponentially, so too has the movement to build more robust, user-friendly infrastructure for them. Whereas once running AI agents required cloud hosting, today LLMs like Llama 2 can be run on your computer locally. Indeed, while many investors and researchers are focused on what AI agents can do, many other developers and software engineers are furiously building new types of infrastructure, such as streaming LLMs, to allow users to power agents at a fraction of their current cost. In the coming months and years, it will be important to keep an eye on the infrastructure that is developed for these tools. Individuals who become experts in these tools, known as *LLMOps*, are likely to be well employed and remunerated.

Security, Legislation, and Ethics

Finally, it’s important to note that while AI agents are sure to become ever more powerful and impactful in the economy, they aren’t without their flaws or ethical gray zones. This is because, for the very same reason that AI agents can be useful, they can also be dangerous. Unsuspecting users can and have provided agents confidential company and personal information that can compromise critical company intellectual property or infrastructure. Others note that even such choices as the default tone and voice of an AI agent can be ethically fraught. Do the engineers who build agents intentionally or unintentionally gender the agents they build? Could this contribute to unintended societal harm?

This is to say nothing of the LLM engines behind agents. Some LLMs that agents use have been trained on materials that their intellectual property owners did not give permission to use. And many LLMs have the potential to disseminate disinformation or clear lies. For these reasons and more, legislators in many countries are already considering legislation to limit or prevent the use of AI agents. These issues are sure to be front and center in the months and years to come, and they are worth paying attention to.

Conclusion

While the future of AI agents is sure to be different even six months from now, the genie is out of the bottle on AI agents. Too many individuals have benefited from AI agents for their usefulness to be ignored—and too many developers know how to build these tools for there to be any hope of stopping their progress. The question moving forward with AI agents isn’t whether they will be used but how they will be used. If you are at all interested after having read this report, there is no better time than now to get started.

[1] Written with text generated by OpenAI’s Chat GPT-4, August 5, 2023.

[2] Text generated by OpenAI’s Chat GPT-4, August 5, 2023.

[3] “Document Question-Answering,” GitHub, accessed October 29, 2023.

[4] Text generated by Chainlit and PandasAI, August 5, 2023.

[5] Text generated by Chainlit and PandasAI, August 5, 2023.

[6] Text generated by HuggingFace Chat and LlaMa 2I, August 6, 2023.

[7] Text generated by OpenAI’s ChatGPT, August 6, 2023.

[8] Text generated by OpenAI’s ChatGPT, August 6, 2023.

[9] Text generated by OpenAI’s ChatGPT, August 6, 2023.

[10] Nikolas Zolas et al., “Advanced Technologies Adoption and Use by U.S. Firms: Evidence from the Annual Business Survey,” US Census Bureau, July 2, 2020.

About the Author

Ben Labaschin is principal machine learning engineer at Workhelix, where he develops generative AI and large language models and applies them to economic problems. Previously Ben led machine learning initiatives at companies such as Hopper, XPO Logistics, and Blackstone. When Ben isn’t working, he likes to spend his time with his dog Atlas or his niece Shoshana.