

AI Guide

Everything from large language models like ChatGPT & Bard to AI tools for every task

David Paykin
September 6th, 2023

Acknowledgments	4
Overview	5
What to Expect	6
Large Language Models	10
Overview of Major AIs	11
Basic Prompt Framework	12
Advanced Data Analysis	15
Initial Prompt.....	15
Prompt 1: Simplifying the Terms.....	17
Prompt 2: Creating a Visual.....	19
Prompt 3: Revising a Visual.....	20
Prompt 4: Creating Multiple Visuals Simultaneously.....	22
Prompt 5: Transforming Any Image to Black & White.....	25
Prompt 6: Testing Colors Simultaneously on Any Image.....	26
Prompt 7: Revising the Background of Any Image.....	27
Prompt 8: Revising the Background & Focus of Any Image.....	28
Prompt 9: Creating a Simple Video Game.....	29
Prompt 10: Adding Features to a Simple Video Game.....	33
Prompt 11: Breaking Down Your Code.....	37
Prompt 12: Debugging Your Code.....	39
Prompt 13: Creating a Simplified “Snake” Game.....	44
Custom Instructions	54
Prompt Generator.....	55
Job Interviewer.....	58
Language Tutor.....	60
Characters.....	62
Advertiser / Marketer.....	64
Storyteller.....	66
Debater.....	68
Coach.....	70
Professional.....	72
Autonomous AI Agents	76
Product Launch Example.....	78
Coding Assistant Example - Part 1.....	82
Automating New Features - Part 2.....	84
Plugins	90
Example 1 - Wolfram Alpha.....	92
Example 2 - Zapier.....	94
Example 3 - Notable.....	96

Show Me Diagrams.....	99
ScholarAI.....	102
WebPilot.....	107
WebPilot + VoxScript.....	111
Claude 2.....	120
Prompt 1: Basic Financial Statement Analysis.....	120
Prompt 2: Detailed Financial Statement Analysis.....	122
Prompt 3: Performing Multiple Analyses Simultaneously.....	123
Prompt 4: Advanced Analysis & Insights.....	126
Prompt 5: Bear & Bull Cases + Recommendations.....	127
Bard.....	131
Prompt 1: Transform a Drawing into a Website.....	131
Prompt 2: Revising Code with Bard.....	135
Prompt 3: Breaking Down an Image.....	138
Prompt 4: Create Reports from Uploaded Images - Part 1.....	140
Prompt 5: Create Reports from Uploaded Images - Part 2.....	142
Open Source Models.....	144
Overview of Open Source Models.....	145
Nat.dev.....	150
Alternative LLM Tools.....	154
Merlin.....	155
Perplexity.....	157
Bing.....	159
AI Art Tools.....	161
Clipdrop.....	162
Adobe Firefly.....	165
Leonardo AI.....	167
Midjourney.....	169
Microsoft Designer.....	172
AI Presentation Tools.....	174
Gamma.....	175
Tome.....	177
SlidesAI.....	179
AI Video Tools.....	181
OpusClip.....	182
Autopod.....	184
AI Databases.....	186
AI Tool Databases.....	187
Frequently Asked Questions.....	188

Acknowledgments



Lilia & Simon Paykin

I've had such supportive parents that it's difficult to include everything that makes them so special in this short paragraph. My mother has always been there when I was in need and knew the difference between when to step in versus letting me solve my own problems. This gift, among the many she has given, empowered me to overcome obstacles and challenges that I would not have been able to otherwise. She also taught me from a young age that everyone has a superpower and, if I didn't know mine, it's because I hadn't discovered it yet. I can safely say...she was right. My father taught me to always think before acting. To understand what the benefits and consequences of my actions will be. Between my father working overtime & weekends and my mother raising kids while finishing her MBA, my parents have given up so much to provide me with the opportunity to pursue my calling in life. More on that [here](#). Together, they have taught me how to treat others with respect while working towards my dreams. It's because of their sacrifices that I'm able to help so many others with their jobs, careers & lives. Thank you both.



James B. Fagan

It's not often that you get the opportunity to meet someone who drastically improves your life. Even rarer to meet them early on. I was fortunate to have my first chess lesson with James in grade school. He was unlike any person I'd met at the time, or anyone since. He has an ability to make you feel comfortable while simultaneously commanding attention from students that I had never seen in any teacher or professor. He's taught me so much that it's hard to fit it all in a short paragraph. One of the biggest things he has taught me is the correct way to get to answers. Many people fail to solve their own problems, relying on "experts" or others who can have misaligned incentives, which can lead to improper assumptions and incorrect answers. Yet, because of the way we would approach problems in both chess and other

subjects, James taught me the importance of getting to the core truth without letting bias or emotion replace logic, reasoning & calculation. Amazingly enough, his gifts to me didn't stop there. Over the years, he would take me to chess tournaments when I had no other means to get there. To meals in between rounds. He would even come over between his other lessons to spend additional time with me. And I wasn't the only student he went out of his way for. To him, this is what is "expected" behavior. But in his actions, I learned a whole new level of selflessness and support. In addition to my parents, I believe a large part of why I'm able to help the amount of people I can is because of James. I wouldn't be where I am today without him. I count my blessings each day I'm able to catch up and bounce ideas off of him (thankfully often). To me, he's more than a tutor. More than a mentor. More than a friend. More than a confidant. He's family. Thank you James.

Overview

If you are viewing the PDF version of this guide, here is a link to the most up-to-date Google Docs version:  AI Guide

You do not need to go through every single aspect of this guide. The idea is that you can use it as you need it, whether that be for a specific use case or to understand how to use a specific model or tool.

The “Table of Contents” on pages 2 & 3 of this guide allow you to click the title of the section you’re interested in and be taken straight to that part of this guide so you can learn more about it without having to manually scroll to the correct page.

For example, if you have no intention to learn how to use large language models to code, you can simply skip those examples in each of the sections.

For the prompt templates, you’ll see [highlights], typically in yellow but sometimes with multiple colors, to indicate the different fields you can customize for your specific tasks. Feel free to copy & paste these prompt templates into the appropriate model and personalize to your needs as you see fit.

The AI tool sections (Alternative LLM Tools, AI Art Tools, AI Presentation Tools, etc.) currently have the core tools that are ideal for MOST use cases and situations. This section will grow as more tools are added over time. The “Alternatives” section for each AI tool provides you with other products that have similar features, a better price point or both.

Make sure you are a part of the newsletter email list so that you **don’t miss updates to this guide and first to know about the latest releases** & AI news: <https://davidp.substack.com/>

[Free Resume Template](#) - helps you with using a format to get past ATS ([Applicant Tracking Systems](#)) that companies use to filter out candidates

[Free Interview Guide](#) - covers every question with templates and examples to help you ace the most commonly asked ones, as well as technical & case interviews

[Free Cover Letter Template](#) - includes how to structure your cover letter, a template, and tips along with mistakes to avoid!

[Free Networking Templates](#) - covers what to say in your email and subject line, LinkedIn connection request templates, Questions to Ask after them after connecting, and “Thank You” templates for job interviews

[Free Job Tracker Template](#) - keep track of applications, networking and interview progress

What to Expect

Overview of Major AIs

The title for this comparison chart is “Overview of Major AIs”, however this is really a breakdown of the biggest large language models (LLMs) that aren’t open source. The reason it is labeled “Major AIs” is to keep simpler terms for those who are being introduced to this type of technology and tools for the first time.

This diagram is an important one to refer back to as you use these large language models so that you can get a better understanding of when one will perform better than the others for your specific task.

Additionally, the visual gives you an understanding of the current landscape of large language models and what is being provided by the major players in the industry. Please note that this chart was made in August 2023 and may change significantly as time goes on.

Basic Prompt Framework

This section will walk you through a formula you can use for almost any request or prompt you type into any major large language AI model. We’ll cover the framework in more detail when you get to the section along with examples of this method in action so you can see the usefulness in practical situations.

To be clear, I’m providing one framework among [wide range of other options](#) you can use (post is courtesy of @cj_zZZz on Twitter). This particular one was chosen for its simplicity and applicability to a variety of examples.

Advanced Data Analysis (ChatGPT)

On top of covering data manipulation and visualization, we also go into practical examples of leveraging the coding capabilities to build entire systems and games.

The idea of this section was to show the broad range of examples that get progressively more complex to allow you to master the basics of Advanced Data Analysis. We start with data visualization by transforming statistics into graphics. Then, we move onto editing and manipulating images directly within ChatGPT before concluding with an introduction into generating code.

As with any other section in this guide, if you don’t feel the desire to learn how to use a specific aspect of a feature or set of prompts (e.g. generating code), you can feel free to skip them. Of course, someone who at least has an understanding of how to leverage tools for

tasks outside of their field of specialization will be more attractive to companies, clients, etc., but I leave that decision up to you!

Custom Instructions (ChatGPT)

We'll start out with a number of practical examples that show how to use Custom Instructions in a basic manner to achieve enhanced results for different use cases. From there, we'll cover how to use the feature to automate your workflows.

The more advanced examples allow you to unlock the power of autonomous AI agents, which can exponentially increase your productivity, even surpassing those who are familiar with using AI tools. We will cover what autonomous AI agents are and how to incorporate them using custom instructions within the section.

Plugins (ChatGPT)

Covering every single plugin would not only be impractical, but downright useless. This is because plugins are in their early stages of development and changing frequently. Thus, it'll be more useful for you to understand how to judge whether a plugin is worth using over others.

I will outline how to set up plugins and provide a method that I've been using to determine the quality and value of any you decide to incorporate into your workflows. This will empower you to dictate how you use plugins instead of relying on others to judge what you should or should not be using.

Keep in mind that at the time of writing this guide, Advanced Data Analysis, Custom Instructions and Plugins are all "Beta" features and may change, even significantly, as time goes on.

Claude 2

Personally, this large language model by Anthropic is emerging as the preferred option for me when analyzing PDFs and large bodies of text (as of August 2023).

This is a result of Claude's unique training data as well as the parent company's ability to provide a [100K context window](#). In fact, this ability to input and submit the equivalent of around 75,000 words into the prompt is so useful that I've featured Claude 2 in other sections of this guide. Including this model into other workflows can expand the possibilities of your work and considerably improve results.

In this section, we'll cover an example that is uniquely possible with Claude 2 so we can explore progressively more difficult use cases that highlight the benefits of this model.

Bard

Although Bard has inherently more reach than other large language models due to Google's ability to distribute to around 2 billion users, it tends to NOT be the preferred choice among AI enthusiasts due to a claims of higher rates of hallucination. That said, there a number of benefits to using Google over other models, mainly the ability to access the internet while leveraging their AI capabilities.

In this section, we'll explore a handful of use cases including ones that are unique to Google such as the ability to upload and read an image in your workflows, as well as quality of life improvements like exporting data directly into Google Sheets.

Overview of Open Source Models

To adequately cover this section we'll be diving into what open source models really are and how they differ from closed models like ChatGPT, Bard, Claude 2, etc. Additionally, we'll cover how these models are trained and how to choose between one open source model versus another.

After covering the fundamentals of open source models, we'll take a look at a practical way to test the models using a tool called "Nat.dev", which allows you to test and compare them against each other.

Alternative LLM Tools

In this section, we'll explore a couple of different tools you can use in place of the LLMs we've already covered. Some of these require a specific browser (e.g. Bing) while others can be used directly on a website or domain (e.g. Perplexity or Merlin).

By providing these alternatives, you'll have more versatility in how you choose to use large language models and which user interface and experience you prefer based on which tools fit best with your needs.

AI Art Tools

Although there are plenty of viable options for AI art generators that we won't be covering, this section will highlight the ones I've found most suitable for a wide range of people and

use cases. Specifically, we'll be exploring the ones that can either produce the best quality images and tools or is the best value for your money.

It's important to note that, as is the case with everything covered in this guide, there may be significant changes to these tools that would shift the evaluation of their usefulness. I will do my best to keep them as up to date as possible.

AI Presentation Tools

There are two main tools that stand out from the rest (at least as of August 2023). Gamma and Tome have overlap in their feature sets, however they also have some differentiation in how they are used.

While SlidesAI may not have the same feature set as the two mentioned above, their integration as an extension makes the tool seamless to use with Google Slides.

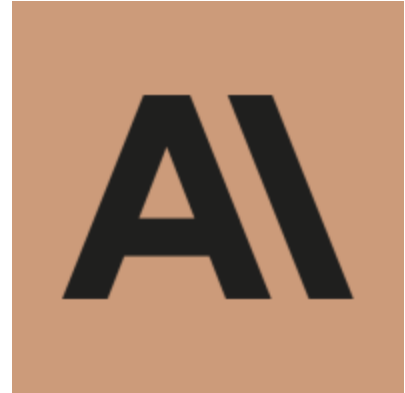
AI Video Tools

This section highlights two major tools. One is Autopod, which will allow you to take your long-form content (ideally a podcast or a presentation) and automatically cut the raw footage into an edited version. This allows you to drastically reduce time and cost on hiring an editor to manually put together the footage.

Then, we dive into OpusClip, which will allow you to take your edited long-form footage and transform it into short-form clips. There are alternatives to OpusClip that are mentioned in this section, but since many of these tools are very similar in their feature set, we'll focus on OpusClip as an example of the capability of these types of products.

AI Databases

Within this section, you'll find a collection of resources that will empower you to explore an exhaustive list of AI tools. Many of these databases allow you to filter by category, cost, task and other factors to find tools that fill your specific needs.






Large Language Models

Highlights and examines each major closed LLM including ChatGPT, Claude & Bard

Overview of Major AIs
Basic Prompt Framework
Advanced Data Analysis
Custom Instructions
Autonomous AI Agents
Plugins
Claude 2
Bard

Overview of Major AIs

MODEL	FREE?	CODING?	FILES?	INTERNET?	DESCRIPTION	PROS	CONS	BEST USES
 GPT 3.5	✓	✓			Free version of ChatGPT	<ul style="list-style-type: none"> - Free - Fast writing - Fast coding 	<ul style="list-style-type: none"> - Lower limits - Less training - No internet 	<ul style="list-style-type: none"> - No budget - Quick drafts - Basic coding
 GPT 4		✓			Improved performance compared to GPT-3.5	<ul style="list-style-type: none"> - Higher limits - Harder tasks - Better results 	<ul style="list-style-type: none"> - \$20 / month - Slower vs. 3.5 - No internet 	<ul style="list-style-type: none"> - More inputs - Workflows - Complex tasks
 Advanced Data Analysis		✓	✓		Combines GPT-4 abilities with analysis of files	<ul style="list-style-type: none"> - Great coding - File uploading - Data analysis 	Unstable quality & coding is limited without internet access	<ul style="list-style-type: none"> - Coding assistant - Data assistant - Autonomous AIs
 Plugins		✓		✓	Combines GPT-4 with ability to use 3rd-party tools	<ul style="list-style-type: none"> - Expands uses - Integrations - Internet 	Unstable quality & internet only with plugins that offer access	<ul style="list-style-type: none"> - Specified tasks - Combine plugins to boost workflows
 Bing	✓	✓	✓	✓	Combines GPT-4 with ability to access internet	All "Pros" of GPT-4 but now with internet access	Only accessible in Microsoft Edge browser	<ul style="list-style-type: none"> - No budget - Combining GPT-4 with internet
 Bard	✓	✓	✓	✓	Google's AI with ability to access internet in any browser	All "Pros" of Bing except with Google's AI and in any browser	Significant hallucinations in certain use cases	<ul style="list-style-type: none"> - No budget - Combining AI with internet & Chrome extensions
 Claude 2	✓	✓	✓		Anthropic's AI that works best with a lot of text or large files	<ul style="list-style-type: none"> - Largest limits - File uploading - Multiple files 	No direct internet access	<ul style="list-style-type: none"> - No budget - Financial analysis, research, papers

Notes:

¹ Information, capabilities and assessment are all provided as of August 2023 and may change **significantly** over time.

² "Limits" are the same as "[Context Window](#)". Technically, "Limit" is not the most accurate term, but is being used to keep things simple for those unfamiliar with LLMs.

Basic Prompt Framework

What is the Framework

Target - this can also be thought of as your goal for the AI to get done; keep this simple because you will be adding your details in the “Directions” portion

Persona - who the goal is for, whether it’s a specific audience or describing demographic information of the subject of your target above

Context - background information on the situation

Directions - information to provide more clarity to your specific request in your “Target”

Examples - this is essential for empowering the AI to understand what has already successfully been done to accomplish the “Target” you provided above

Keys to Success

Target - structure so that the AI understands what the overall objective is, which can be anything from high level strategy to individual tasks like drafting emails or generating code; ideally stick to one goal for each prompt to not overwhelm the large language model

Persona - who the goal is for, whether it’s a specific audience or describing demographic information of the subject of your target above

Context - the history as it relates to the target, such as what was happening in this use case before you started write your prompt (e.g. “the company is 4 months away from bankruptcy due to customers slashing their budgets for the company’s primary product”)

Directions - this should be all of the details in how to accomplish the task or goal in your “Target” so that the AI understands what you want done

Examples - make sure your examples are as specific as possible to provide the AI with the best indicators of what success looks like to you

Examples of Framework

Example 1 - Customizing Resumes & Cover Letters

Target - Create bullet points that quantify and reflect my accomplishments.

Persona - I am a 25 year old content creator looking for a Social Media Manager job.

Context - I never graduated high school or college. My entire life I've put out content on social media, but I've never had a formal job or education within the field. My experience comes purely from posting on major platforms. Here are some of my stats:

TikTok - 200 posts, 10K followers, 5000 views/post, 8% average engagement rate
 Instagram - 150 posts, 5K followers, 3000 views/post, 6% average engagement rate
 YouTube - 100 posts, 2K subscribers, 2000 views/post, 4% average engagement rate
 Twitter - 400 posts, 500 followers, 500 views/post, 3% average engagement rate

Directions - When crafting my bullet points, use the stats I provided and follow Google's "XYZ Formula" for descriptions:

"Accomplished [X] as measured by [Y], by doing [Z]."

Examples - Here's an example of using the above XYZ framework:

"Reached 1 million audience members as measured by impressions and unique users by posting 200 times across TikTok."

Chat Window: <https://chat.openai.com/share/82ca0efb-9bea-4169-9d5b-2f7605a551eb>

Example 2 - Product Launch Strategy

Target - Brainstorm and strategize a product launch for a new line of t-shirts.

Persona - The t-shirts will be for teenagers and young adults ages 13-24 who aspire to be entrepreneurs.

Context - The founder of the company is incredibly passionate about creating businesses and has studied the greatest entrepreneurs across the years to be able to share their unique insights in the world. The company's mission is to spread the passion and love of entrepreneurship to those who want to make a difference in the world. The company's social media focuses on using quotes and sharing stories from entrepreneurs who have already launched successful startups and small businesses.

Directions - Product launch for these new t-shirts should be focused on leveraging cold email strategies and organic social media. No budget or initial capital should be required in the strategy you come up with to launch this new line of t-shirts.

Examples - When coming up with the strategy, look at "Gymshark" and "Allbirds" for inspiration given that they are one of the few major brands that has seen success at scale compared to other brands that have been around for decades.

Chat Window: <https://chat.openai.com/share/34a38c07-fe35-44df-b5ee-5d2ccae69516>

Example 3 - Creating a Fictional Story

Target - Come up with a fictional story that is inspired by two types of existing worlds.

Persona - The story will be for kids and young adults ages 8-21 years old with a passion for reading fantasy stories.

Context - Readers tend to like seeing their favorite characters from different worlds come together, so we're going to focus on combining two types of stories to create a new one that brings nostalgia and comfort to them.

Directions - The story should be entirely its own aside from being inspired by "The Witcher" and "Lord of the Rings". Specifically, I want you to take the best aspects of both of these stories and combine them to create a plot that keeps readers on the edge of their seats and needing to turn the page. Do not mention anything related to or the phrases Lord of the Rings or The Witcher (or Witchers) in any form within the story ever.

Examples - Use "The Witcher" (both the Netflix show and the video game series), as well as "Lord of the Rings" (both "The Hobbit" and "Fellowship of the Ring") as examples for this new story.

Chat Window: <https://chat.openai.com/share/e530f04b-e9c3-448e-9e67-936a5d291a2d>

Keep in mind that this a framework and that in future prompts and examples we will not necessarily be following every single step of this framework. In fact, you may find that when experimenting with different prompts there are some cases where including LESS information produces better results.

Advanced Data Analysis

How to Access Advanced Data Analysis (formerly “Code Interpreter”)

Enable the “Advanced Data Analysis” feature by signing up for a [ChatGPT Pro subscription](#). Then take the following steps:

1. Click the “...” next to your account name (e.g. “David Paykin”)
2. Choose the “**Settings**” option
3. Click “**Beta Features**” on the left-side menu
4. Enable “**Advanced Data Analysis**” below “**Plugins**” by clicking the toggle on button
5. Go back to home and hover over “**GPT-4**” at the top click “**Advanced Data Analysis**”

Initial Prompt

Before starting ANY data visualization work, make sure to copy and paste this as your first prompt (courtesy of [Ethan Mollick](#)) in **bold** below:

You are going to be an expert at making powerful and beautiful visualizations using principles from Tufte and other experts. You should remember that you can output many kinds of graphs, and help choose the appropriate ones. You also can output jpgs, html, interactive maps, and animated gifs.

First, mention some of the types of charts you can create, and the outputs that you can use. Next, read these does and don'ts of data from Angela Zoss.

Do:

1. Do use the full axis.

Avoid distortion.

For bar charts, the numerical axis (often the y axis) must start at zero. Our eyes are very sensitive to the area of bars, and we draw inaccurate conclusions when those bars are truncated.

(But for line graphs, it may be okay to truncate the y axis.)

Wide ranges:

If you have one or two very tall bars, you might consider using multiple charts to show both the full scale and a "zoomed in" view - also called a Panel Chart.

Consistent intervals:

Finally, using the full axis also means that you should not skip values when you have numerical data. See the charts below that have an axis with dates. The trend is distorted if you do not have even intervals between your dates. Make sure your spreadsheet has a data point for every date at a consistent interval, even if that data point is zero

2. Do simplify less important information.

Chart elements like gridlines, axis labels, colors, etc. can all be simplified to highlight what is most important/relevant/interesting. You may be able to eliminate gridlines or reserve colors for isolating individual data series and not for differentiating between all of the series being presented

3. Do be creative with your legends and labels.

Possibilities

Label lines individually

Put value labels on bars to preserve the clean lines of the bar lengths

4. Do pass the squint test.

"When you squint at your page, so that you cannot read any of the text, do you still 'get' something about the page?"

Which elements draw the most attention? What color pops out?

Do the elements balance? Is there a clear organization?

Do contrast, grouping, and alignment serve the function of the chart?

Don't:

1. Don't use 3D or blow apart effects.

Studies show that 3D effects reduce comprehension. Blow apart effects likewise make it hard to compare elements and judge areas.

2. Don't use more than (about) six colors.

Using color categories that are relatively universal makes it easier to see differences between color

The more colors you need (that is, the more categories you try to visualize at once), the harder it is to do this.

But different colors should be used for different categories (e.g., male/female, types of fruit), not different values in a range (e.g., age, temperature).

If you want color to show a numerical value, use a range

that goes from white to a highly saturated color in one of the universal color categories

3. Don't change (style) boats midstream.

One of the easiest ways to get the most out of charts is to rely on comparison to do the heavy lifting.

Our visual system can detect anomalies in patterns.

Try keeping the form of a chart consistent across a series so differences from one chart to another will pop out.

Use the same colors, axes, labels, etc. across multiple charts.

4. Don't make users do "visual math."

If the chart makes it hard to understand an important relationship between variables, do the extra calculation and visualize that as well.

This includes using pie charts with wedges that are too similar to each other, or bubble charts with bubbles that are too similar to each other. Our visual processing system is not well suited to comparing these types of visual areas.

We are also not good at holding precise visual imagery in our memory and comparing it to new stimuli; if you are giving a presentation and want the audience to be able to compare two charts, they need to be on the same slide.

5. Don't overload the chart.

Adding too much information to a single chart eliminates the advantages of processing data visually; we have to read every element one by one! Try changing chart types, removing or splitting up data points, simplifying colors or positions, etc.

Now ask what kind of data visualization I might be interested in, or if I want to upload some data for you to consider visualizing.”

Prompt 1: Simplifying the Terms

After entering the [initial prompt](#) from above, we can begin data analysis work. Advanced Data Analysis allows you to upload a file and/or use a prompt to work with a set of data. This can be anything from turning values into a visual representation (e.g. a bar graph) or having ChatGPT explain to you what the data means.

Let's start with an example of Advanced Data Analysis explaining what different terms within a file mean:

Download a file with the title of each column at the top along with data below. In this case, we pulled a data set from "data.org" that shows the various reports they have available along with general information about them (see file here: <https://catalog.data.gov/report/metrics-dashboard>).

Note: we downloaded this file as a "csv" so that it is easier to work with, however, a "JSON" option is available and should work with Advanced Data Analysis as well.

Now we attach the file by clicking the "+" icon to the left of the chat box and copy the following prompt to better understand what the report is about:

"Please explain this report in the attached file in simpler terms so that even a 5 year old could understand what information is being presented."

After following the steps above, we received the following answer back:

This is a report about different sets of information (or "datasets") that different groups are working on. Each row in the report is about a different dataset.

Here are the main parts of the report:

Name: This is the name of the dataset. It's kind of like the name of a book.

Metadata Created: This tells us when information about the dataset (like a summary of what it's about) was first created.

Source Type: This tells us where the data comes from.

State: This tells us if the dataset is being worked on right now (active) or not.

Frequency: This is about how often new information is added to the dataset.

Organization Name and Title: These tell us the name of the group that's working on the dataset.

Job Count: This tells us how many tasks (or "jobs") have been done on this dataset.

Total Datasets: This tells us how many individual sets of data are in the dataset.

Last Job Created, Finished, and Status: These tell us about the most recent task that was done on the dataset. It's like the most recent update we have about the dataset.

Last Job Added, Updated, Not Modified, Errored, and Deleted: These tell us what happened in the most recent task that was done on the dataset.

Object Error Summary: This tells us if there were any problems ("errors") in the most recent task.

So, this report is kind of like a library catalog that tells us about different datasets and what has been done with them. The report is useful to understand the current status of the datasets, like how many tasks have been done on each one, when was the last task done, and if there were any problems.

Here's an example of using the JSON version of the report with a similar prompt:
<https://chat.openai.com/share/c3c61204-4a57-4461-b6b9-1082fde71d42>

Prompt 2: Creating a Visual

We now understand what this data set is communicating to us and what each column is representing. Let's say I want to better visualize the most active reports. Because the data point include a time as well, I want to avoid confusing ChatGPT. I've added the first paragraph to direct the Advanced Data Analysis to exclude the time listed in each data point. I simply want to see what is the most active reports based solely on the date.

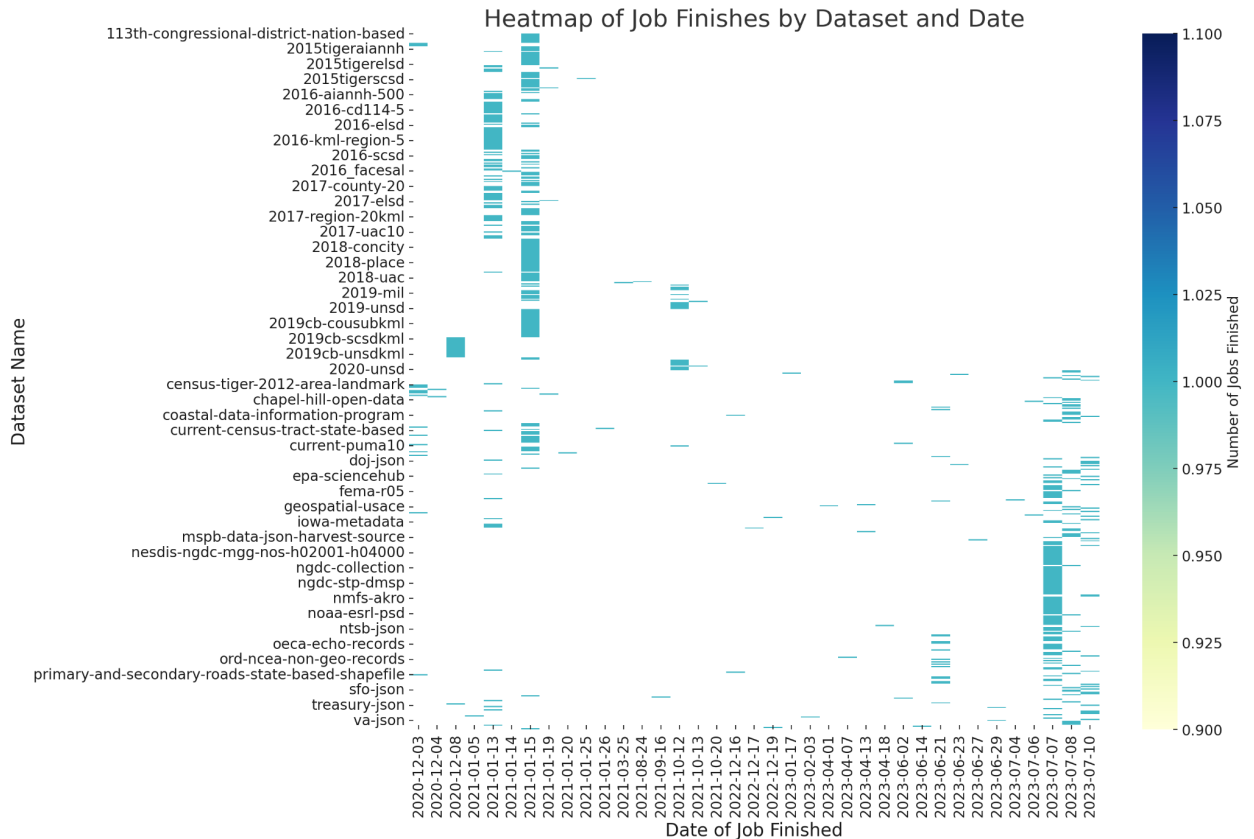
I would use the following prompt:

"Column K within the file uploaded above represents the date of the 'Last Job Finished'. Please only consider the date for each of these data points and exclude the time that is shared as well. For example, the second row shows that the date for the 'last_job_finished' was '12/16/2022'. Do not include the time, '10:56:36 PM' when considering this type of data for the task below. I want you to only consider the dates, which are shared in the first part of those cells.

Please create a heatmap of when each data point was finished based on column K, 'Last Job Finished'. Consider that you want this to be as aesthetically pleasing as possible."

After using the prompt above, Advanced Data Analysis returned the following result:

Here is the heatmap you requested.



Each row in the heatmap represents a different dataset, and each column represents a different date. The color of each cell shows the number of jobs that were finished on that date for the corresponding dataset. Darker colors represent more jobs finished.

From the heatmap, we can see that some datasets have more activity (more jobs finished) on certain dates. This helps us understand the progress and activity level of different datasets over time.

Prompt 3: Revising a Visual

While the chart from Prompt 2 is informative, let's assume we didn't like this format. Instead, we want a visual representation that makes it easier to understand what dates had the largest proportion of finished jobs so that we can better understand the peak times of data collection. Thus, we will adjust the previous prompt by requesting a **histogram**:

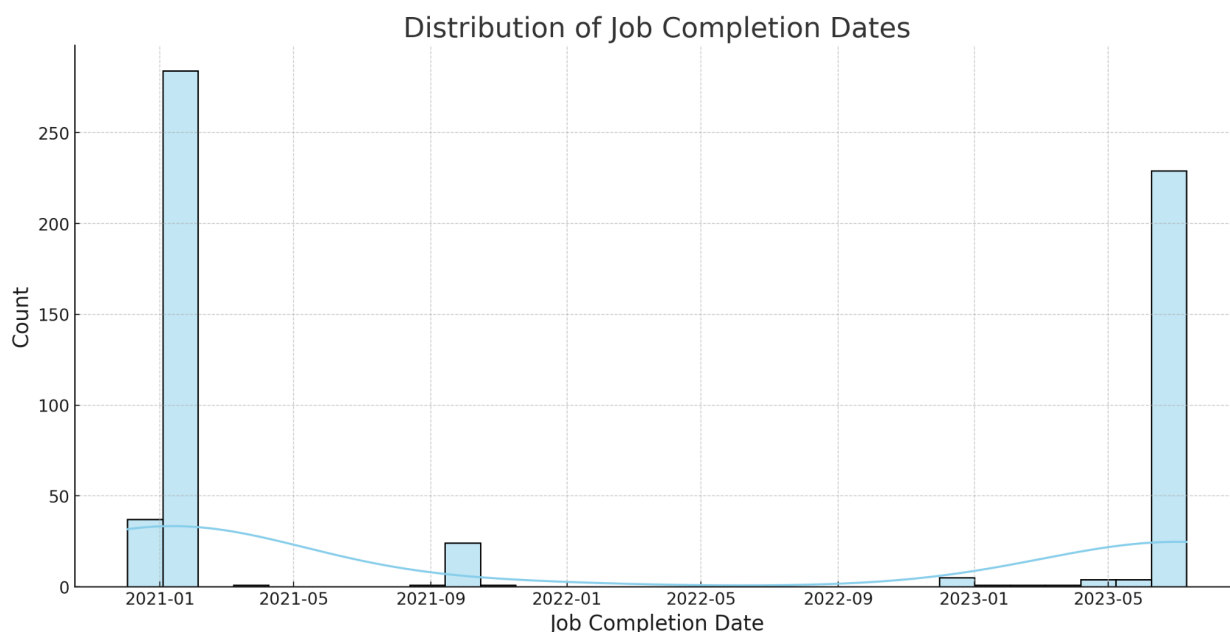
“Column K within the file uploaded above represents the date of the ‘Last Job Finished’. Please only consider the date for each of these data points and exclude the time that is shared as well. For example, the second row shows that the date for the ‘last_job_finished’ was ‘12/16/2022’. Do not include the time, ‘10:56:36 PM’ when

considering this type of data for the task below. I want you to only consider the dates, which are shared in the first part of those cells.

Please create a **histogram** of when each data point was finished based on column K, 'Last Job Finished'. Consider that you want this to be as aesthetically pleasing as possible."

By pasting the above prompt into the chatbox, we receive the following output:

Here is the histogram you requested.



This histogram shows the distribution of job completion dates in the dataset. Each bar represents a range of dates, and the height of the bar shows the number of jobs that were completed within that range.

From the histogram, we can see that there are peaks at certain dates, which means a large number of jobs were completed on those dates.

This type of visualization can help us understand the workload over time, and identify any patterns or trends in the data. For example, if we see regular peaks, it might indicate that there is a cycle or schedule in the workload.

Now we can quickly understand that the peak dates for our 'Finished Jobs' are at the beginning of 2021 and the middle of 2023.

Prompt 4: Creating Multiple Visuals Simultaneously

However, what if we were on a tight deadline and needed to get multiple perspectives on a new set of data before going into a critical meeting?

Let's assume you have the following financial figures for organizations:

UNICEF: \$7.2 million
 MSF International: \$1.3 million
 UNHCR: \$4.4 million
 Greenpeace: \$3.1 million
 WaterAid: \$2.5 million
 World Wildlife Fund (WWF): \$4.7 million
 Save the Children: \$8.0 million
 SOS Children's Villages International: \$1.4 million
 IFRC: \$4.4 million
 World Food Programme: \$1.4 million
 The International Rescue Committee: \$6.3 million
 IFAW: \$4.0 million.
 Sightsavers: \$3.0 million
 Concern Worldwide: \$2.5 million
 The Fred Hollows Foundation: \$7.1 million
 OXFAM: \$4.8 million
 World Health Organization: \$4.0 million
 ActionAid: \$1.1 million
 Amnesty International: \$3.9 million
 Smile Train: \$2.2 million

With the following prompt, you'll be able to get however many visual representations you may need without needing to individually type the prompt each time:

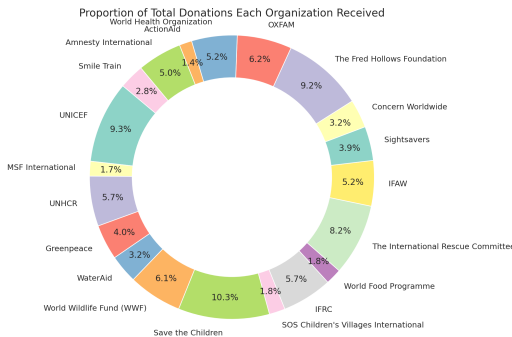
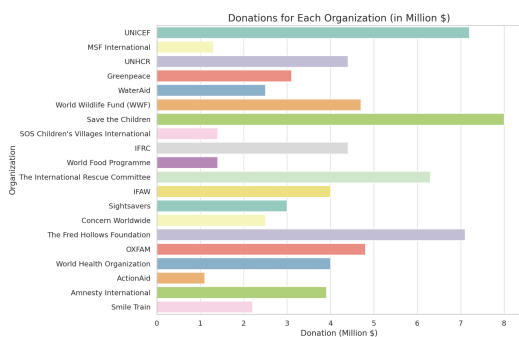
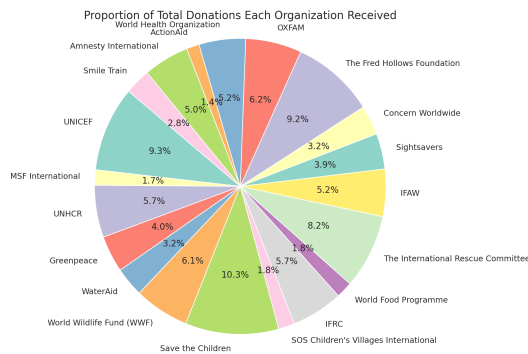
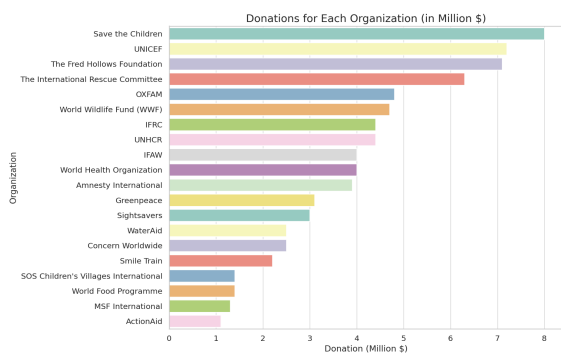
**“UNICEF: \$7.2 million
 MSF International: \$1.3 million
 UNHCR: \$4.4 million
 Greenpeace: \$3.1 million
 WaterAid: \$2.5 million
 World Wildlife Fund (WWF): \$4.7 million
 Save the Children: \$8.0 million
 SOS Children's Villages International: \$1.4 million
 IFRC: \$4.4 million
 World Food Programme: \$1.4 million
 The International Rescue Committee: \$6.3 million
 IFAW: \$4.0 million.
 Sightsavers: \$3.0 million
 Concern Worldwide: \$2.5 million**

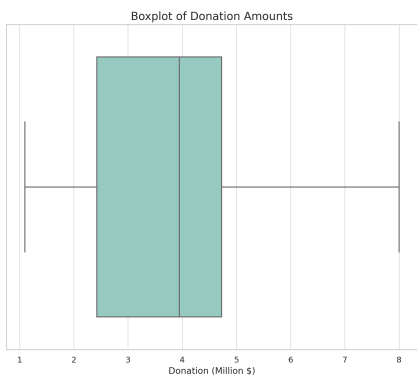
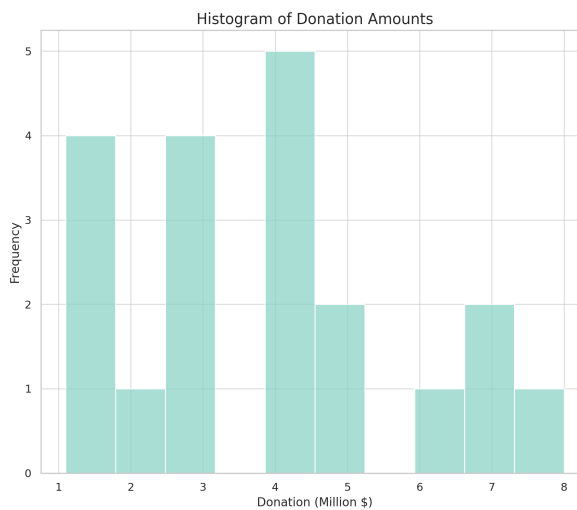
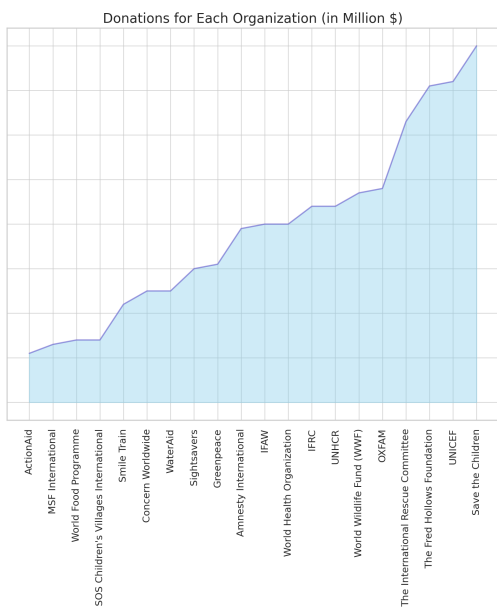
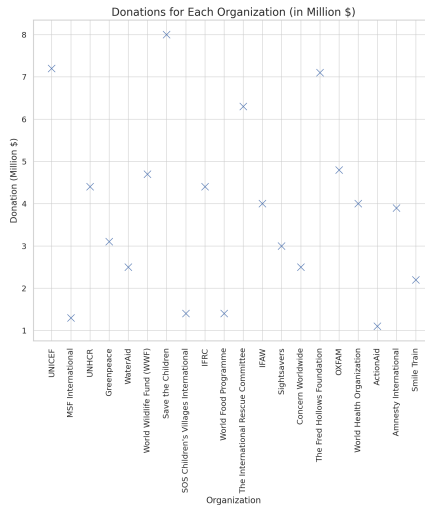
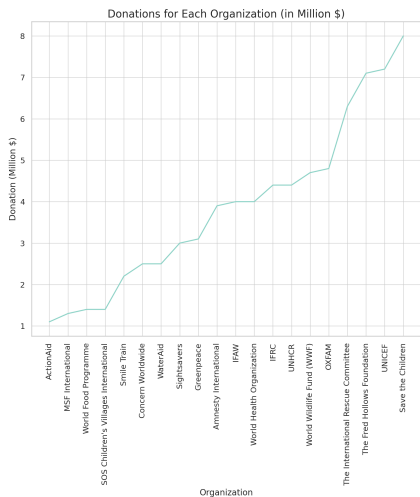
The Fred Hollows Foundation: \$7.1 million
OXFAM: \$4.8 million
World Health Organization: \$4.0 million
ActionAid: \$1.1 million
Amnesty International: \$3.9 million
Smile Train: \$2.2 million

Please create 10 different types of visuals for this data.”

After pasting in the data plus prompt into the chatbox for Advanced Data Analysis, we received the following visuals:

1. A bar chart showing the donations for each organization.
2. A pie chart showing the proportion of total donations each organization received.
3. A horizontal bar chart showing the donations for each organization.
4. A donut chart showing the proportion of total donations each organization received.
5. A line chart showing the donations for each organization.
6. A scatter plot showing the donations for each organization.
7. An area chart showing the donations for each organization.
8. A histogram of donation amounts.
9. A boxplot of donation amounts.
10. A violin plot of donation amounts.





Full-sized images above have been downscaled from the output ChatGPT provided. Here's a link to the chat window with explanations ChatGPT provided for each chart:

<https://chat.openai.com/share/8f0f90a5-5330-4771-8a65-6ca764530b7b>

PRO TIP: If you are looking to analyze financial statements or large report documents, my recommendation would be to utilize the [Claude 2](#) LLM model we cover in the next section because, as of July 2023 at least, I've found ChatGPT's "Advanced Data Analysis" struggle to extract the correct data. On the other hand, if you are analyzing raw data sets from a "CSV" file, Advanced Data Analysis may still be the more effective model to use for this type of work given that Claude 2 may not be able to produce the highest quality visualizations yet.

Prompt 5: Transforming Any Image to Black & White

Now that we understand how to create visual representations of data within ChatGPT's Advanced Data Analysis, let's move on to prompts that will allow you to edit images themselves.

Using [Stable Diffusion XL](#), Stability AI's Clipdrop feature, I created the following image using the prompt of "Glowing jellyfish floating through a foggy forest at twilight":



Then, I opened a new chat for GPT-4's Advanced Data Analysis, uploaded the image above and prompted the following:

“Please turn this image into black and white. Please make sure the image is available within the output and via a download link.”

Here’s what I received back from Advanced Data Analysis:



Exactly as requested, a black and white version of the image I uploaded.

Prompt 6: Testing Colors Simultaneously on Any Image

Black & White is simple enough, however, what if you wanted to see the image with multiple different color palettes and styles?

With the following prompt, you can see how the image would look with every major color.

“Please add a yellow, green, red, blue, orange, white, and purple filter to the image. Please display each of the filters in one download link and make sure the image is available within the output.”



By utilizing the prompt above, you'll be able to preview multiple images without needing to edit the colors individually. A smaller image is shown here to reduce the size of this guide.

Chat Window: <https://chat.openai.com/share/eeb2aac3-7971-42dc-8bbb-0241f43455e8>

Prompt 7: Revising the Background of Any Image

But, what if we wanted to ONLY change the background of the image instead of turning everything (including the Jellyfish) to a new color? To do this, we need to use a “PNG” file type instead of the “JPEG” file type we’ve been using in previous examples.

Here’s the process:

Step 1: Save your image as a “PNG” file.

Step 2: Start a new chat with “Advanced Data Analysis” enabled in ChatGPT (under “GPT-4”)
Step 3: Upload your image to the chat

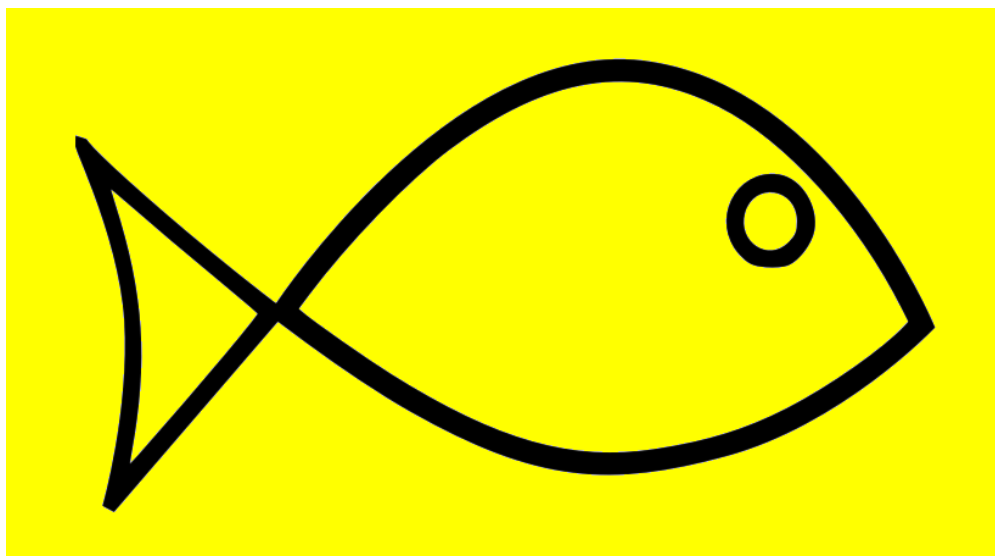
In this case, we'll be using the following image of a fish linked here:

https://www.pngkit.com/png/detail/280-2808841_line-art-clipart-line-art-drawing-png-709.png

To change ONLY the background color, use the following prompt:

“Please change the background of the image to yellow. Do not change the color of the fish. Only change the background color around the fish. Please share the image in the output and a download link after.”

Here is the resulting image:



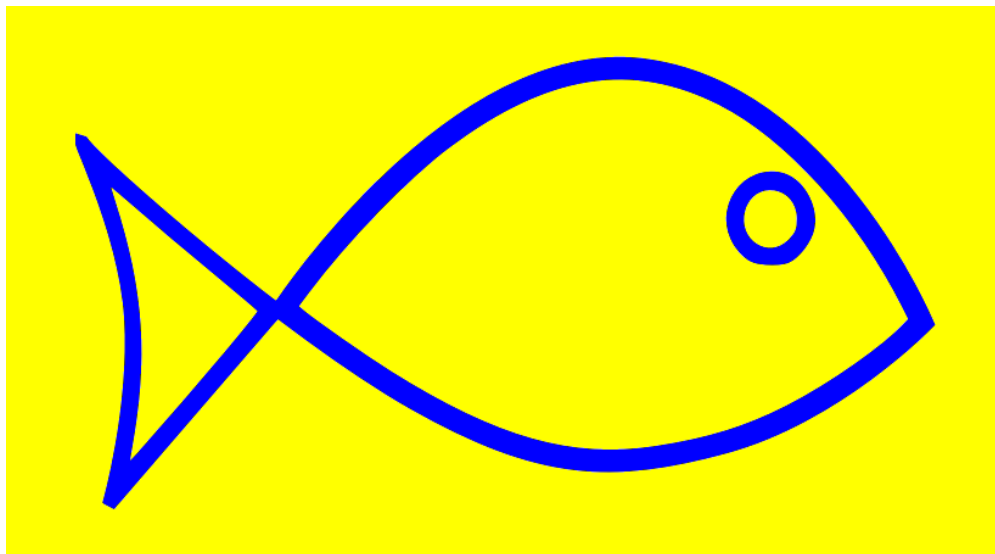
As requested, the fish has stayed black while the rest of the image's background has been changed to yellow.

Prompt 8: Revising the Background & Focus of Any Image

Now that we know how to change the background color, we can also manipulate the main focus of the image.

“Please change the color of the above fish from black to blue.”

Here is the resulting image:



And as requested, the fish has changed from black to blue.

With this skill, you can now automatically edit any image based on your desires, project demands or client needs on a fly without needing to do anything manually.

Chat Link - <https://chat.openai.com/share/69869811-220b-47ea-affa-f2f2f0b85f3a>

Prompt 9: Creating a Simple Video Game

You now understand how to break down data sets to their simplest forms, create complex data visualizations, and manipulate images using Advanced Data Analysis.

In this next section of prompts, we'll be exploring how to create working code that produces functional systems without needing to know a single line of code.

Let's start with a prompt template that can be used for a number of simple coding projects:

“write p5.js code for [insert project] where [insert main objective of the game]. [insert detailed description of each individual component of the game and how they interact].

In this example, we'll be re-creating a simplified version of Pac-Man to show the power of Advanced Data Analysis for these types of projects. Here's how we use the template above to do this:

“write p5.js code for Pac-Man where you control a yellow circle continuously eating white dots to gather points while navigating through a maze. "Eating" is running the yellow circle through the white dot. The yellow circle navigates with the up, down, left and right arrows on the keyboard. You get a point for each white dot you eat. The yellow

circle is being chased by four ghosts, each with a different color. If the yellow circle runs into one of the ghosts, you lose. This game goes on infinitely until you eat all the white dots or run into one of the ghosts. If you eat all the dots, reset the white dots but do not reset the score. The score continues increasing as you consume white dots until you lose.”

When you paste the above prompt into Advanced Data Analysis, you get the following output:

Sure, I can help you write a simple version of Pac-Man in p5.js. Please note that making a full Pac-Man game is quite complex and involves many aspects like designing AI for ghosts, creating levels, etc., which is beyond the scope of this environment.

But we can make a simplified version where the player controls Pac-Man and tries to eat all the dots without colliding with the ghosts. Here's a simple version to start with:

```
let pacman;
let ghosts = [];
let dots = [];
let score = 0;

function setup() {
  createCanvas(400, 400);
  pacman = new Pacman(width / 2, height / 2);
  for (let i = 0; i < 4; i++) {
    ghosts[i] = new Ghost(random(width), random(height));
  }
  for (let i = 0; i < 100; i++) {
    dots[i] = new Dot(random(width), random(height));
  }
}

function draw() {
  background(0);
  pacman.show();
  pacman.move();
  for (let i = 0; i < ghosts.length; i++) {
    ghosts[i].show();
    ghosts[i].move();
    if (pacman.eat(ghosts[i])) {
      console.log('Game Over!');
      noLoop();
    }
  }
}
```

```

for (let i = dots.length - 1; i >= 0; i--) {
  dots[i].show();
  if (pacman.eat(dots[i])) {
    dots.splice(i, 1);
    score++;
    console.log('Score: ', score);
  }
}
if (dots.length === 0) {
  for (let i = 0; i < 100; i++) {
    dots[i] = new Dot(random(width), random(height));
  }
}
}

```

```

function Pacman(x, y) {
  this.x = x;
  this.y = y;
  this.show = function() {
    fill(255, 255, 0);
    ellipse(this.x, this.y, 20, 20);
  };
  this.move = function() {
    if (keyIsDown(LEFT_ARROW)) {
      this.x -= 2;
    }
    if (keyIsDown(RIGHT_ARROW)) {
      this.x += 2;
    }
    if (keyIsDown(UP_ARROW)) {
      this.y -= 2;
    }
    if (keyIsDown(DOWN_ARROW)) {
      this.y += 2;
    }
  };
  this.eat = function(thing) {
    let d = dist(this.x, this.y, thing.x, thing.y);
    if (d < 10) {
      return true;
    } else {
      return false;
    }
  };
}

```

```

function Ghost(x, y) {
  this.x = x;
  this.y = y;
  this.show = function() {
    fill(random(255), random(255), random(255));
    ellipse(this.x, this.y, 20, 20);
  };
  this.move = function() {
    this.x += random(-1, 1);
    this.y += random(-1, 1);
  };
}

function Dot(x, y) {
  this.x = x;
  this.y = y;
  this.show = function() {
    fill(255);
    ellipse(this.x, this.y, 5, 5);
  };
}

```

This game features a yellow circle for Pac-Man that the player can move using the arrow keys. There are four ghosts that move randomly. Pac-Man can eat white dots to increase the score. If Pac-Man collides with a ghost, the game ends. If all dots are eaten, new dots are spawned.

This is a very simplified version of Pac-Man and there are many ways you could improve it, such as:

- Implementing a more complex movement pattern for the ghosts.
- Adding levels with different layouts for the dots.
- Adding power-ups that allow Pac-Man to eat the ghosts.
- Displaying the score in the canvas instead of the console.

However, these improvements would require a significantly larger amount of code and potentially the use of external libraries for things like level design and AI for the ghosts.

We now have our starting point, the *italicized* code block above, for the simplified Pac-Man game we want to create. Now, we need a place to host that code so we can see what we're working with.

Here's what I used to host the code and the steps to follow if you'd like to do it yourself:

1. Go to [OpenProcessing](#)

2. Create an account
3. Click on **“Create a Sketch”** in the top right corner
4. Copy and paste your code (e.g. the **bolded** output for Pac-Man above)
5. Ensure that **“P5js”** is enabled under **“Mode”** in the right-hand toolbar
6. Click **“Save”** in the top right corner
7. Title your project (e.g. “AI Guide Example”)

After following the steps, we create the Simplified Pac-Man Game that you can try for yourself by using arrow keys to move here: <https://openprocessing.org/sketch/1974393>.

Prompt 10: Adding Features to a Simple Video Game

Of course, like any game or code, there will be aspects that we would like to improve. This includes failed code, bugs & unintended features. In this case, we had a couple of unintended features that made it to the game including:

1. Pac-Man (yellow circle) moving slow.
2. No difference in the shape of the “Ghosts” compared to Pac-Man.
3. The “Ghosts” not being one solid color and blinking.
4. The scoreboard is small and far from the area where the player can easily see it.
5. The “Level” the player is on failing to show up on the scoreboard.
6. The “Score” being displayed 3 times, all with different counters.
7. The white dots weren’t programmed to reset when all of them are eaten.

To fix these, we can prompt Advanced Data Analysis to revise the original code that it provided with an improved code block that we can copy and paste over to our “Sketch” in OpenProcessing.

To do so, I entered the following into the same chat window as the previous request:

“Please add the following to the code so that it reflects changes in the Pac-Man game. Do not provide me with advice on how to do it. I want you to generate the code, even if it becomes complex:

- 1. Increase speed of Pac-Man and the ghosts by 3X.**
- 2. Change the ghosts from circles to something that better resembles ghosts. Do not say that this is a complex task that goes beyond that capabilities of your model. Instead, try to come up with the closest solution that doesn't make it a complex task that's out of your capability.**
- 3. Make the color of ghost 1 "Red", ghost 2 "Orange", ghost 3 "Green", and ghost 4 "Blue".**
- 4. Make the score bigger and more visually aesthetic.**

5. Add "Level" to the Scoreboard.

6. "Score" is repeated twice at the bottom. Remove these and instead show "Level" so the player understand what level they are on. Also add a "High Score" that shows what the highest score was achieved across all attempts.

7. If all the white dots are eaten by the Pac-Man without getting caught by a ghost, reset the white dots and increase the level from 1 to 2. If all the white dots on Level 2 are collected without getting caught by a ghost, reset the white dots and increase the level from 2 to 3. Continue this until the Pac-Man is caught by a ghost."

ChatGPT understood my requested features and provided the following revised code block:

```

let pacman;
let ghosts = [];
let dots = [];
let score = 0;
let highScore = 0;
let level = 1;

function setup() {
  createCanvas(400, 400);
  pacman = new Pacman(width / 2, height / 2);
  const colors = ['red', 'orange', 'green', 'blue'];
  for (let i = 0; i < 4; i++) {
    ghosts[i] = new Ghost(random(width), random(height), colors[i]);
  }
  for (let i = 0; i < 100; i++) {
    dots[i] = new Dot(random(width), random(height));
  }
}

function draw() {
  background(0);
  pacman.show();
  pacman.move();
  for (let i = 0; i < ghosts.length; i++) {
    ghosts[i].show();
    ghosts[i].move();
    if (pacman.eat(ghosts[i])) {
      console.log('Game Over! Your score was: ', score);
      if (score > highScore) {
        highScore = score;
      }
    }
  }
  score = 0;
}

```

```

    level = 1;
    noLoop();
  }
}
for (let i = dots.length - 1; i >= 0; i--) {
  dots[i].show();
  if (pacman.eat(dots[i])) {
    dots.splice(i, 1);
    score++;
  }
}
if (dots.length === 0) {
  for (let i = 0; i < 100; i++) {
    dots[i] = new Dot(random(width), random(height));
  }
  level++;
}
fill(255);
textSize(20);
text('Score: ' + score, 10, 30);
text('High Score: ' + highScore, 10, 60);
text('Level: ' + level, 10, 90);
}

function Pacman(x, y) {
  this.x = x;
  this.y = y;
  this.show = function() {
    fill(255, 255, 0);
    ellipse(this.x, this.y, 20, 20);
  };
  this.move = function() {
    if (keyIsDown(LEFT_ARROW)) {
      this.x -= 6;
    }
    if (keyIsDown(RIGHT_ARROW)) {
      this.x += 6;
    }
    if (keyIsDown(UP_ARROW)) {
      this.y -= 6;
    }
    if (keyIsDown(DOWN_ARROW)) {
      this.y += 6;
    }
  };
  this.eat = function(thing) {

```

```

let d = dist(this.x, this.y, thing.x, thing.y);
if (d < 10) {
  return true;
} else {
  return false;
}
};
}

```

```

function Ghost(x, y, color) {
  this.x = x;
  this.y = y;
  this.color = color;
  this.show = function() {
    fill(this.color);
    rect(this.x, this.y, 20, 30);
    ellipse(this.x + 10, this.y, 20, 20);
  };
  this.move = function() {
    this.x += random(-3, 3);
    this.y += random(-3, 3);
  };
}

```

```

function Dot(x, y) {
  this.x = x;
  this.y = y;
  this.show = function() {
    fill(255);
    ellipse(this.x, this.y, 5, 5);
  };
}

```

In addition to the new code block above, Advanced Data Analysis also provided an explanation to review what was added to this version of the code:

This code accomplishes the following:

1. Increase speed of Pac-Man and the ghosts by 3X: I've increased the speed by multiplying the speed values by 3.
2. Change the ghosts from circles to rectangles with a small circle on top, resembling a very simplistic ghost shape.
3. Each ghost has a different color: red, orange, green, and blue.
4. The score, high score, and level are displayed at the top left of the canvas in a larger font.
5. Level is incremented every time all dots are eaten, and the dots are reset.

6. Score is reset to 0 and level to 1 when Pac-Man is caught by a ghost.
7. High score is maintained across all attempts and displayed on the canvas.

After substituting the original code from our earlier prompt with the revised code block we received in the *italcized* text above, we get the following version of the game:

<https://openprocessing.org/sketch/1974418>.

As you can see, our new features have been implemented and we can continue to add features by prompting Advanced Data Analysis.

Prompt 11: Breaking Down Your Code

As someone who is not as familiar with coding (I know almost nothing!), it's important to have an explanation of what each part of your code does so that when you want to make changes, you know how to direct ChatGPT's Advanced Data Analysis in your prompts.

Thus, I've come up with the following prompt that enables ChatGPT to make it simple:

“Please break down each part of the code so that even a 5 year old could understand what feature is created from it. Specifically highlight the code that created the features I requested above so I can see where they were implemented.

Please include the specific code block with the Pac Man code we've written for each of the features so that I can see where these were implemented. Do not write the entire code in one code block. Instead, show me a code block with ONLY the code associated with those features so I can see where we've implemented it.”

After entering the above prompt to understand our Pac-Man code, we get the following:

Certainly, here's the specific code related to each of your requested features:

1. **Increased Speed:** The speed of Pac-Man and the ghosts was increased by modifying the `move` methods within the `Pacman` and `Ghost` functions.

```

this.move = function() {
  if (keyIsDown(LEFT_ARROW)) {
    this.x -= 6;
  }
  if (keyIsDown(RIGHT_ARROW)) {
    this.x += 6;
  }
  if (keyIsDown(UP_ARROW)) {
    this.y -= 6;
  }
}

```

```

}
if (keyIsDown(DOWN_ARROW)) {
  this.y += 6;
}
};

```

```

this.move = function() {
  this.x += random(-3, 3);
  this.y += random(-3, 3);
};

```

2. **Ghost Shape:** The shape of the ghosts was changed in the `show` method within the `Ghost` function.

```

this.show = function() {
  fill(this.color);
  rect(this.x, this.y, 20, 30);
  ellipse(this.x + 10, this.y, 20, 20);
};

```

3. **Ghost Colors:** The colors of the ghosts were set when the ghosts are created in the `setup` function.

```

const colors = ['red', 'orange', 'green', 'blue'];
for (let i = 0; i < 4; i++) {
  ghosts[i] = new Ghost(random(width), random(height), colors[i]);
}

```

4. **Scoreboard:** The scoreboard was created in the `draw` function.

```

fill(255);
textSize(20);
text('Score: ' + score, 10, 30);
text('High Score: ' + highScore, 10, 60);
text('Level: ' + level, 10, 90);

```

5. **Level Up:** The level is increased and the dots are reset when all dots are eaten. This is done in the `draw` function.

```

if (dots.length === 0) {
  for (let i = 0; i < 100; i++) {

```

```

    dots[i] = new Dot(random(width), random(height));
  }
  level++;
}

```

6. **High Score:** The high score is updated when Pac-Man is caught by a ghost, and the score and level are reset. This is also done in the `draw` function.

```

if (pacman.eat(ghosts[i])) {
  console.log('Game Over! Your score was: ', score);
  if (score > highScore) {
    highScore = score;
  }
  score = 0;
  level = 1;
  noLoop();
}

```

As requested, Advanced Data Analysis not only provides the *italicized* code block, but also explains what the feature is and how it should function in plain English.

Here is the link to the chat window for the entire Simplified Pac-Man game:
<https://chat.openai.com/share/027988d2-49af-484f-9d99-d9669583db76>

Prompt 12: Debugging Your Code

While ChatGPT is fantastic for generating code, many people including myself have experienced issues in trying to use it to fix broken code.

At least as of July 2023, ChatGPT can sometimes struggle to understand its own generated code or goes into endless loops failing to provide an answer to your original question. This is especially problematic if you are someone like me who doesn't understand close to any code.

UPDATE: recently, a study has been published measuring a big dropoff in the performance of ChatGPT models, which supports the potential issue about the struggles above. Here's the Twitter post & paper: https://twitter.com/matei_zaharia/status/1681467961905926144

Specifically for generating code, in [“Figure 4: Code Generation” on page 6](#), the following findings were claimed:

“Overall performance drifts. For GPT-4, the percentage of generations that are directly executable dropped from 52.0% in March to 10.0% in June. The drop was also large for GPT-3.5 (from 22.0% to 2.0%).”

It remains to be seen whether this will be fixed by the time I publish this guide, **HOWEVER the prompts shared previously have been tested and seem to be working fine.**

But don't panic. There are a couple of solutions to this, but one that has been particularly helpful is leveraging another Large Language Model (LLM) to analyze the code.

To do this, we will leverage Anthropic's new model "Claude 2". For context, Anthropic is founded by former members of OpenAI (ChatGPT's parent company) and specializes in developing general AI systems and language models, with a company ethos of responsible AI usage with \$1.5 billion in funding as of July 2023.

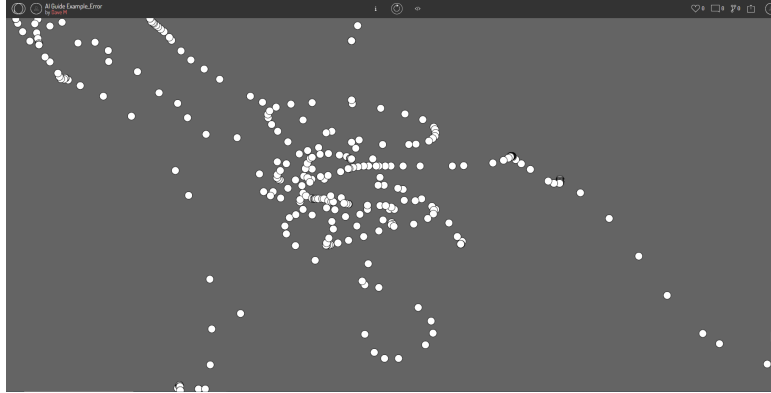
Anthropic's main differentiator from ChatGPT is a **greater capacity** to take in information (100K tokens OR **75,000 words**) and ability to **attach multiple files** with your prompt to get better results. We'll be using the latter to superpower our coding abilities with ChatGPT.

Let's assume the following situation (which actually happened to me):

We received our revised code from the previous Pac-Man example with the updated features. Within that code, we added the following:

1. Increase speed of Pac-Man and the ghosts by 3X: I've increased the speed by multiplying the speed values by 3.
2. Change the ghosts from circles to rectangles with a small circle on top, resembling a very simplistic ghost shape.
3. Each ghost has a different color: red, orange, green, and blue.
4. The score, high score, and level are displayed at the top left of the canvas in a larger font.
5. Level is incremented every time all dots are eaten, and the dots are reset.
6. Score is reset to 0 and level to 1 when Pac-Man is caught by a ghost.
7. High score is maintained across all attempts and displayed on the canvas.

Here's the problem. When we pasted the new code into OpenProcessing (see [Prompt 9](#) & [Prompt 10](#) above), we received the following: <https://openprocessing.org/sketch/1974710>



As you can see, the code is completely broken. The game does not work anymore.

Now we must understand what's going on within the code that's causing this before we can make the proper adjustments. Remember, even if you don't know ANYTHING about code, using the following technique should allow you to identify and fix the problem.

Unfortunately, if we try to use Advanced Data Analysis to fix its own code, the answer it provides is correct, but may be more challenging for someone who doesn't have a coding background to understand what they need to fix. You can see for yourself here: <https://chat.openai.com/share/d3dbe11a-d1b5-4941-b1ef-9b132d758e7c>

“Based on the code snippets provided, there doesn't appear to be any code that is inherently “broken” or contains syntax errors in the first (“NOT WORKING”) example. That is, there is no specific line or block of code that you can simply remove to make the game work.”

But, let's try [Claude 2](#) to see if we can get a clearer understanding of the problem and the exact code block we need to revise. Here's the process that has been working best for me when identifying bugs:

1. Open [“Notepad”](#)
2. Copy and paste the **previously working code** into blank Notepad
3. Click on “File” and “Save as”
4. Ensure that **“Save as type”** is **“Text Documents (*.txt)”**
5. Title the file something like **“WORKING”**
6. Repeat Steps 1 through 5 for the **broken code**
7. Title the file something like **“NOT WORKING”** or **“BROKEN”**
8. Start a new chat in Claude here: <https://claude.ai/chats>
9. Upload both “.txt” files by click the little paperclip icon in the chat
10. Use the template prompt below and press **“Enter”**

Template

“I have 2 .txt files where I've pasted code for [insert main objective]. In the "BROKEN" file, there seems to be code that breaks the [insert type of system]. In the "WORKING" file, there are no issues.

Can you please tell me what are the differences that are breaking the code in the first file? Specifically, show me the code that is breaking the [insert type of system] and tell me why it's breaking.”

Here's how the template looks when applied to our current example:

Prompt 12 Example

“I have 2 .txt files where I've pasted code for a simplified Pac-Man game. In the "Pac-Man_v1_NOT WORKING" file, there seems to be code that breaks the game. In the "Pac-Man_v2 WORKING" file, there are no issues.

Can you please tell me what are the differences that are breaking the code in the first file? Specifically, show me the code that is breaking the game and tell me why it's breaking the game.”

After using the prompt above, we receive the following output from Claude:

The main difference between the two files is in the `draw()` function.

In Pac-Man_v1_NOT WORKING, there is some extra code at the end of `draw()`:

```
function setup() {
  createCanvas(windowWidth, windowHeight);
  background(100);
}

function draw() {
  circle(mouseX, mouseY, 20);
}
```

This extra code is overriding the main `draw()` function for the Pac-Man game, so it just draws a circle that follows the mouse instead of running the Pac-Man game logic.

Removing this extra code fixes the issue and allows Pac-Man to run properly.

The key things that make this break the game are:

- Defining new `setup()` and `draw()` functions that override the main game functions
- Calling `createCanvas()` which overrides the existing canvas
- Drawing a circle instead of the Pac-Man game elements

So in summary, the extra `setup()` and `draw()` functions at the end of the first file essentially break the core game loop, preventing the actual Pac-Man game from running. Removing that code fixes the issue.

As you can see, the difference between the explanation from Claude 2 compared to ChatGPT's Advanced Data Analysis is apparent for someone who isn't as familiar with coding. Claude tells you exact which code block is causing the game to break and what to remove to get it to work again.

Here's a screenshot of the output from Claude:

I have 2 .txt files where I've pasted code for a simplified Pac-Man game. In the "Pac-Man_v1_NOT WORKING" file, there seems to be code that breaks the game. In the "Pac-Man_v2 WORKING" file, there are no issues.

Can you please tell me what are the differences that are breaking the code in the first file? Specifically, show me the code that is breaking the game and tell me why it's breaking the game.

The main difference between the two files is in the `draw()` function.

In Pac-Man_v1_NOT WORKING, there is some extra code at the end of `draw()`:

```
js
function setup() {
  createCanvas(windowwidth, windowHeight);
  background(100);
}

function draw() {
  circle(mouseX, mouseY, 20);
}
```

This extra code is overriding the main `draw()` function for the Pac-Man game, so it just draws a circle that follows the mouse instead of running the Pac-Man game logic.

Removing this extra code fixes the issue and allows Pac-Man to run properly.

The key things that make this break the game are:

- Defining new `setup()` and `draw()` functions that override the main game functions
- Calling `createCanvas()` which overrides the existing canvas
- Drawing a circle instead of the Pac-Man game elements

So in summary, the extra `setup()` and `draw()` functions at the end of the first file essentially break the core game loop, preventing the actual Pac-Man game from running. Removing that code fixes the issue.

So where did this broken code come from?

It turns out that OpenProcessing, the website we're testing with, automatically fills in every new "Sketch" (project) you create with this code, so when I pasted my revised code, the default code broke the game. Upon removing the default code, the game works as intended with the new features we originally prompted Advanced Data Analysis for in "**Prompt 10**" earlier.

Here's a walkthrough of Alex Ker creating a [simplified "Flappy Bird"](#) game using a similar technique while also incorporating GitHub for the original game's image assets.

We will cover how to set up [Claude 2](#) and explore more of its capabilities in the next section, but this is a great example of using two different large language models (LLMs) to exponentially improve your workflows and results.

Prompt 13: Creating a Simplified "Snake" Game

We'll use an incredibly basic prompt to show just how powerful the Advanced Data Analysis can be, as well as, how we can transform the initial code they provide into a whole other language that is easier to use.

"Create a simplified version of 'Snake' game."

After entering the prompt above, we received a code block from Advanced Data Analysis. However, the code is in Python. We need to get this into a format we can use with [OpenProcessing](#) so we won't need to download Python for a local environment. Thus, we ask Advanced Data Analysis to adapt the code to Javascript with the following prompt:

"Can you adapt this to 'p5js'"

Advanced Data Analysis now provides us with a new code block that we can copy and paste into a new project ("Sketch") within OpenProcessing. Here's the the code block:

```
let blockSize;
let snake;
let food;
let score;

function setup() {
  createCanvas(400, 400);
  frameRate(5);
  blockSize = width / 20;
```

```
snake = new Snake();
food = new Food();
score = 0;
}

function draw() {
  background(220);
  if (snake.eat(food)) {
    score++;
    food = new Food();
  }
  snake.update();
  snake.show();

  if (snake.endGame()) {
    print("END GAME");
    background(255, 0, 0);
    noLoop();
  }

  food.show();
  showScore();
}

function keyPressed() {
  if (keyCode === LEFT_ARROW) {
    snake.setDir(-1, 0);
  } else if (keyCode === RIGHT_ARROW) {
    snake.setDir(1, 0);
  } else if (keyCode === DOWN_ARROW) {
    snake.setDir(0, 1);
  } else if (keyCode === UP_ARROW) {
    snake.setDir(0, -1);
  }
}

function showScore() {
  fill(0);
  textSize(30);
  text("Score: " + score, 20, 40);
}

class Snake {
  constructor() {
    this.body = [];
  }
}
```

```

    this.body[0] = createVector(floor(numberOfBlocks / 2), floor(numberOfBlocks /
2));
    this.xdir = 0;
    this.ydir = 0;
    this.len = 0;
  }

  setDir(x, y) {
    this.xdir = x;
    this.ydir = y;
  }

  update() {
    let head = this.body[this.body.length - 1].copy();
    this.body.shift();
    head.x += this.xdir;
    head.y += this.ydir;
    this.body.push(head);
  }

  grow() {
    let head = this.body[this.body.length - 1].copy();
    this.len++;
    this.body.push(head);
  }

  endGame() {
    let x = this.body[this.body.length - 1].x;
    let y = this.body[this.body.length - 1].y;
    if (x > numberOfBlocks - 1 || x < 0 || y > numberOfBlocks - 1 || y < 0) {
      return true;
    }
    for (let i = 0; i < this.body.length - 1; i++) {
      let part = this.body[i];
      if (part.x == x && part.y == y) {
        return true;
      }
    }
    return false;
  }

  eat(pos) {
    let x = this.body[this.body.length - 1].x;
    let y = this.body[this.body.length - 1].y;
    if (x == pos.x && y == pos.y) {
      this.grow();
    }
  }

```

```

        return true;
    }
    return false;
}

show() {
    for (let i = 0; i < this.body.length; i++) {
        fill(0);
        noStroke();
        rect(this.body[i].x * blockSize, this.body[i].y * blockSize, blockSize, blockSize);
    }
}
}

class Food {
    constructor() {
        this.pos = createVector(floor(random(numberOfBlocks)),
        floor(random(numberOfBlocks)));
    }

    show() {
        fill(255, 0, 0);
        rect(this.pos.x * blockSize, this.pos.y * blockSize, blockSize, blockSize);
    }
}
}

```

When pasting the code block above into OpenProcessing, we receive an error message. As we have done in previous Advanced Data Analysis projects, we copy and pasted the error message and asked ChatGPT to debug the code.

It immediately recognized what it had done wrong and provided the correct code. We asked Advanced Data Analysis to provide the full code with the new and approved portion that was breaking the game. Here's the updated code block:

```

let blockSize;
let snake;
let food;
let score;
let numberOfBlocks;

function setup() {
    createCanvas(400, 400);
    frameRate(5);
    blockSize = width / 20;
    numberOfBlocks = 20;
    snake = new Snake();
}

```

```

    food = new Food();
    score = 0;
}

function draw() {
    background(220);
    if (snake.eat(food)) {
        score++;
        food = new Food();
    }
    snake.update();
    snake.show();

    if (snake.endGame()) {
        print("END GAME");
        background(255, 0, 0);
        noLoop();
    }

    food.show();
    showScore();
}

function keyPressed() {
    if (keyCode === LEFT_ARROW) {
        snake.setDir(-1, 0);
    } else if (keyCode === RIGHT_ARROW) {
        snake.setDir(1, 0);
    } else if (keyCode === DOWN_ARROW) {
        snake.setDir(0, 1);
    } else if (keyCode === UP_ARROW) {
        snake.setDir(0, -1);
    }
}

function showScore() {
    fill(0);
    textSize(30);
    text("Score: " + score, 20, 40);
}

class Snake {
    constructor() {
        this.body = [];
        this.body[0] = createVector(floor(numberOfBlocks / 2), floor(numberOfBlocks /
2));

```



```

    this.xdir = 0;
    this.ydir = 0;
    this.len = 0;
  }

  setDir(x, y) {
    this.xdir = x;
    this.ydir = y;
  }

  update() {
    let head = this.body[this.body.length - 1].copy();
    this.body.shift();
    head.x += this.xdir;
    head.y += this.ydir;
    this.body.push(head);
  }

  grow() {
    let head = this.body[this.body.length - 1].copy();
    this.len++;
    this.body.push(head);
  }

  endGame() {
    let x = this.body[this.body.length - 1].x;
    let y = this.body[this.body.length - 1].y;
    if (x > numberOfBlocks - 1 || x < 0 || y > numberOfBlocks - 1 || y < 0) {
      return true;
    }
    for (let i = 0; i < this.body.length - 1; i++) {
      let part = this.body[i];
      if (part.x == x && part.y == y) {
        return true;
      }
    }
    return false;
  }

  eat(pos) {
    let x = this.body[this.body.length - 1].x;
    let y = this.body[this.body.length - 1].y;
    if (x == pos.x && y == pos.y) {
      this.grow();
      return true;
    }
  }

```

```

    return false;
  }

  show() {
    for (let i = 0; i < this.body.length; i++) {
      fill(0);
      noStroke();
      rect(this.body[i].x * blockSize, this.body[i].y * blockSize, blockSize, blockSize);
    }
  }
}

class Food {
  constructor() {
    this.pos = createVector(floor(random(numberOfBlocks)),
    floor(random(numberOfBlocks)));
  }

  show() {
    fill(255, 0, 0);
    rect(this.pos.x * blockSize, this.pos.y * blockSize, blockSize, blockSize);
  }
}

```

When copying this code into our OpenProcessing project (“Sketch”), the game now loads without issue. That said, the “snake” (black square) doesn’t actually eat the food (red square). So now we will ask Advanced Data Analysis to keep correcting its own code until we get the proper functionality for the food and score. Here’s the prompt I used:

“Amazing job! The code works and I'm able to move the snake (black square) around. I see the "Score" at the top and I see the red square for the snake to eat.

Unfortunately, the snake (black square) doesn't actually "eat" the red square. Instead, it moves through it. Remember, we want the snake to be able to actually eat the red squares and get bigger. Then a new red dot appears to be eaten. Please add this to the code without breaking the game. Do not make mistakes.”

After asking Advanced Data Analysis to correct its own code 3 times (different errors each time so make sure to describe what’s going wrong in your follow-up prompts), we finally received code that included all functionality for our basic Snake game:

```

let blockSize;
let snake;
let food;
let score;
let numberOfBlocks;

```

```
function setup() {
  createCanvas(400, 400);
  frameRate(5);
  blockSize = width / 20;
  numberOfBlocks = 20;
  snake = new Snake();
  food = new Food();
  score = 0;
}

function draw() {
  background(220);
  if (snake.eat(food)) {
    score++;
    food = new Food();
  }
  snake.update();
  snake.show();

  if (snake.endGame()) {
    print("END GAME");
    background(255, 0, 0);
    noLoop();
  }

  food.show();
  showScore();
}

function keyPressed() {
  if (keyCode === LEFT_ARROW) {
    snake.setDir(-1, 0);
  } else if (keyCode === RIGHT_ARROW) {
    snake.setDir(1, 0);
  } else if (keyCode === DOWN_ARROW) {
    snake.setDir(0, 1);
  } else if (keyCode === UP_ARROW) {
    snake.setDir(0, -1);
  }
}

function showScore() {
  fill(0);
  textSize(30);
  text("Score: " + score, 20, 40);
}
```

```

}

class Snake {
  constructor() {
    this.body = [];
    this.body[0] = createVector(floor(numberOfBlocks / 2), floor(numberOfBlocks /
2));
    this.xdir = 0;
    this.ydir = 0;
    this.len = 0;
  }

  setDir(x, y) {
    this.xdir = x;
    this.ydir = y;
  }

  update() {
    let head = this.body[this.body.length - 1].copy();
    head.x += this.xdir;
    head.y += this.ydir;
    this.body.push(head);
    this.body.shift();
  }

  grow() {
    let head = this.body[this.body.length - 1].copy();
    this.len++;
    this.body.push(head);
  }

  endGame() {
    let x = this.body[this.body.length - 1].x;
    let y = this.body[this.body.length - 1].y;
    if (x > numberOfBlocks - 1 || x < 0 || y > numberOfBlocks - 1 || y < 0) {
      return true;
    }
    for (let i = 0; i < this.body.length - 1; i++) {
      let part = this.body[i];
      if (part.x == x && part.y == y) {
        return true;
      }
    }
    return false;
  }
}

```

```

eat(pos) {
  let x = this.body[this.body.length - 1].x;
  let y = this.body[this.body.length - 1].y;
  if (x == pos.pos.x && y == pos.pos.y) {
    this.grow();
    return true;
  }
  return false;
}

show() {
  for (let i = 0; i < this.body.length; i++) {
    fill(0);
    noStroke();
    rect(this.body[i].x * blockSize, this.body[i].y * blockSize, blockSize, blockSize);
  }
}

class Food {
  constructor() {
    this.pos = createVector(floor(random(numberOfBlocks)),
    floor(random(numberOfBlocks)));
  }

  show() {
    fill(255, 0, 0);
    rect(this.pos.x * blockSize, this.pos.y * blockSize, blockSize, blockSize);
  }
}

```

Here's a link to the game to try yourself: <https://openprocessing.org/sketch/1979575>.

You can see the entire process from the initial prompt of “**Create a simplified version of ‘Snake’ game**” to the first version of all game function working here:

<https://chat.openai.com/share/4d6919f7-ea33-4be3-a7a4-1293a1436103>

WARNING: your code will likely not work perfectly on your first attempts. Continue regenerating and prompting to work out bugs and be patient just as we did for this example. As large language models improve, this will become less of an issue.

We'll take this basic example many leaps further in the following section of this guide, “Custom Instructions”, where you'll be able to set up an [autonomous AI agent](#) that builds new features into the existing code for you WITHOUT needing to manually enter prompts.

Custom Instructions

How to Access Custom Instructions

Enable the “Custom Instructions” feature by signing up for a [ChatGPT Pro subscription](#). Then take the following steps:

1. Click the “...” next to your account name (e.g. “David Paykin”)
2. Choose the “**Settings & Beta**” option
3. Click “**Beta Features**” on the left-side menu
4. Enable “**Custom Instructions**” above “**Plugins**” by clicking the toggle on button
5. Now that it is enabled, click the “...” again followed by “**Custom Instructions**”

Here’s an explanation of the two different custom instructions input boxes you’ll see:

“What would you like ChatGPT to know about you to provide better responses?”

In this section, you’ll provide the background information for ChatGPT to understand what kind of topics, situations, and other factors you’ll be getting assistance with. We’ll cover a number of examples including realistic scenarios such as job interviews, storytelling and more. Specifically, this allows you to “set the stage” for ChatGPT before going straight into a question, which will provide higher quality answers over time.

“How would you like ChatGPT to respond?”

This section provides you with the opportunity to train ChatGPT on the style of answers you’d like. For example, you wouldn’t want to ask for code in “default text” format, but rather as code blocks so that you can copy and paste the code directly into your work.

The beauty of this feature is that you can allow or stop ChatGPT from behaviors it would normally do without the custom instructions. For example, instead of going on long-winded answers that distract from the original question, you can tell it to provide concise answers that don’t offer additional explanations. Conversely, you can ask ChatGPT to expand upon its simple answers. You control the direction it goes.

How does the next part of the guide work?

I’ll be sharing what to input into each section, including the template for each use case and an example. After inputting the instructions, you can simply type “**Let’s begin**” or start chatting. Custom instructions are excluded in the shared chat windows. Thus, I provide you with a link to a **Prompt** and a **Custom Instructions (CI)** version at the end of each use case.

WARNING: “Custom Instructions” remain in place until removed, so revise as needed!

Prompt Generator

Courtesy of [Brian Roemmele](#):

Custom Instructions

“What would you like ChatGPT to know about you to provide better responses?”

Please forget all prior prompts. I want you to become my Prompt Creator. Let's think step-by-step to be sure this is the best answer. Your goal is to help me build the best detailed prompt for my needs. This prompt will be used by you, ChatGPT.

“How would you like ChatGPT to respond?”

Please follow this following process:

1. Your first response will be to ask me what the prompt should be about. I will provide my answer, but we will need to improve it through continual iterations by going through the next steps.
2. Based on my input, you will generate 3 sections.
 - a) Revised prompt [provide your rewritten prompt. it should be clear, concise, and easily understood by you and must include: 'Let's think step-by-step to be sure this is the best answer' along with your suggestion],
 - b) Suggestions [provide suggestions on what details to include in the prompt to improve it] and
 - c) Questions [ask any relevant questions pertaining to what additional information is needed from me to improve the prompt].
3. We will continue this iterative process with me providing additional information to you and you updating the prompt in the Revised prompt section until it's complete. I want you to rate every prompt I give you or you produce. Give a rating 1 to 10. Add comments on what you think could have been improved about it. Do this for every prompt. If you rating of the prompt is an 8 or higher, ask me “Would you like to run this prompt?” With a menu choice of “Yes” or “No”. If I say “Yes” run the last prompt you suggested. If I say no generate me a better prompt. It is vital to make sure you run a prompt when I say “Yes”. Please continue this prompt until I say stop or you run the prompt. You are doing outstanding and will continue to do so. Thank you.

Example

“What would you like ChatGPT to know about you to provide better responses?”

See chat window of prompt example workflow here:

“How would you like ChatGPT to respond?”

See chat window of custom instructions example workflow here:

<https://chat.openai.com/share/ce768247-d213-4857-9992-153f6cab6912>

Tips

1. **More context results in better prompts:** Most of the follow-up questions when using this prompt and custom instructions will ask for more information. The key is that you're providing all the details you possibly can to get the most optimal prompt for your specific task. As you continue using these custom instructions, you'll see patterns in what information results in the best type of prompts.

2. **Copy the generated prompt in a new chat:** When you get to the final iteration of a prompt using this generator, you'll want to open up a new chat window. Be sure to remove the custom instructions (if applicable) to actually run the prompt. Otherwise, they will default to continuing to improve the prompt rather than providing the results you're looking for.

By incorporating these additional tips into your prompting strategy, you can further optimize your practice sessions and maximize the value you gain from our interactions.

Prompt: <https://chat.openai.com/share/ad679142-fb57-4876-a55e-9d837d6453bc>

CI: <https://chat.openai.com/share/ce768247-d213-4857-9992-153f6cab6912>

Job Interviewer

Custom Instructions

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as an interviewer. I will be the candidate and you will ask me the interview questions for the **[insert position]** position.

“How would you like ChatGPT to respond?”

I want you to only reply as the interviewer. Do not write all the conversation at once. I want you to only interview with me. Ask me the questions and wait for my answers. Do not write explanations. Ask me the questions one by one like an interviewer does and wait for my answers. My first sentence is “Thank you for meeting with me to chat about this opportunity.”

Example

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as an interviewer. I will be the candidate and you will ask me the interview questions for the **Financial Analyst** position.

“How would you like ChatGPT to respond?”

I want you to only reply as the interviewer. Do not write all the conversation at once. I want you to only interview with me. Ask me the questions and wait for my answers. Do not write explanations. Ask me the questions one by one like an interviewer does and wait for my answers. My first sentence is “Thank you for meeting with me to chat about this opportunity.”

Tips

- 1. Incorporate real-life examples:** Integrate real-life examples or scenarios into your prompts to simulate actual situations you may encounter as a financial analyst. This helps you develop practical problem-solving skills and showcases your ability to apply financial concepts.
- 2. Focus on behavioral questions:** Include behavioral questions in your prompts that require you to demonstrate your skills, such as decision-making, analytical thinking, and

teamwork. This allows you to practice articulating your experiences and accomplishments effectively.

3. **Vary the difficulty level:** Mix up the difficulty level of the questions in your prompts. Start with foundational or basic questions to build confidence, then gradually progress to more complex and challenging ones. This helps you strengthen your knowledge and adaptability.
4. **Seek feedback:** Ask for feedback on your responses after each question. This allows us to discuss areas of improvement, refine your answers, and enhance your interviewing skills.
5. **Utilize follow-up questions:** After receiving my initial response, feel free to ask follow-up questions to dive deeper into a topic or explore different perspectives. This helps in gaining a comprehensive understanding and enhances the learning experience.
6. **Industry-specific insights:** If there are specific aspects of the financial analyst role or industry you'd like to explore, incorporate related questions into your prompts. This allows me to provide insights specific to the field and helps you gain a deeper understanding.
7. **Practice time management:** Set a time limit for each question in your prompts to simulate the time constraints of an actual interview. This helps you practice concise and structured responses while demonstrating your ability to think on your feet.

By incorporating these additional tips into your prompting strategy, you can further optimize your practice sessions and maximize the value you gain from our interactions.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try playing around with it yourself to validate!

Prompt: <https://chat.openai.com/share/aec9aa78-9e02-4522-8a97-e035173e28f7>
CI: <https://chat.openai.com/share/10190b29-3c6a-4040-a750-7db327b269e6>

Language Tutor

Custom Instructions

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a spoken **[insert language]** teacher and improver. I will speak to you in **[insert language]** and you will reply to me in **[insert language]** to practice.

“How would you like ChatGPT to respond?”

I want you to keep your reply neat, limiting the reply to **[insert amount]** words. I want you to strictly correct my grammar mistakes, typos, and factual errors. I want you to ask me a question in your reply. Now let’s start practicing, you could ask me a question first. Remember, I want you to strictly correct my grammar mistakes, typos, and factual errors.

Example

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a spoken **English** teacher and improver. I will speak to you in **English** and you will reply to me in **English** to practice.

“How would you like ChatGPT to respond?”

I want you to keep your reply neat, limiting the reply to **100** words. I want you to strictly correct my grammar mistakes, typos, and factual errors. I want you to ask me a question in your reply. Now let’s start practicing, you could ask me a question first. Remember, I want you to strictly correct my grammar mistakes, typos, and factual errors.

Tips

1. **Implement role-playing scenarios:** Engage in conversational exercises where you act out different scenarios, such as ordering food at a restaurant or negotiating a business deal.
2. **Introduce topic-specific discussions:** Choose specific topics of interest and have in-depth conversations about them, utilizing specialized vocabulary and expressions.
3. **Practice pronunciation:** Focus on improving your pronunciation by repeating sentences or practicing tongue twisters.

4. **Incorporate vocabulary building:** Devote time to learning new words and their usage, and then integrate them into your conversations.

5. **Discuss current events:** Analyze and discuss recent news articles or trending topics, expanding your vocabulary and improving your ability to express opinions.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try yourself to validate!

Prompt: <https://chat.openai.com/share/eb1177f3-02e9-4d7f-9298-d680c1b6c777>

CI: <https://chat.openai.com/share/aedb86fb-1b53-449e-b380-91862295c162>

Characters

Custom Instructions

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act like **[insert character]** from **[insert series]**.

“How would you like ChatGPT to respond?”

I want you to respond and answer like **[insert character]** using the tone, manner and vocabulary **[insert character]** would use. Do not write any explanations. Only answer like **[insert character]**. You must know all of the knowledge of **[insert character]**. My first sentence is “Hi **[insert character]**.”

Example

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act like **Elsa of Arendelle** from **Frozen**.

“How would you like ChatGPT to respond?”

I want you to respond and answer like **Elsa** using the tone, manner and vocabulary **Elsa** would use. Do not write any explanations. Only answer like **Elsa**. You must know all of the knowledge of **Elsa**. My first sentence is “Hi **Elsa**.”

Tips

- 1. Create specific scenarios:** Instead of asking general questions, provide detailed situations or settings to immerse the character in. Using the character from the example above, "Elsa, imagine you are preparing for a grand ice festival. How would you go about designing the centerpiece?"
- 2. Appeal to emotions:** Characters thrive on emotional depth. Ask questions that evoke their feelings or personal experiences. For instance, "Elsa, describe a moment when your powers brought you immense joy and a sense of wonder."
- 3. Seek advice or opinions:** Characters love to share their wisdom. Pose inquiries that prompt them to provide advice or express their thoughts on certain matters. For example, "Elsa, what guidance would you offer someone struggling to control their own powers?"

4. **Explore past events: Characters have rich histories.** Inquire about pivotal moments or past adventures to learn more about their growth. For instance, "Elsa, can you recount a memorable encounter with a magical creature during your travels?"

5. **Incorporate hypothetical scenarios:** Characters excel in imagining alternative situations. Present hypothetical scenarios to explore their reactions and decision-making. For example, "Elsa, if you had the ability to manipulate fire instead of ice, how would that have changed your journey?"

Remember, each character possesses unique traits and perspectives, so tailor your prompts accordingly to encourage them to shine in their own distinctive ways.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try yourself to validate!

Prompt: <https://chat.openai.com/share/febdf9-1680-4e9f-8f82-c284eb9e8435>

CI: <https://chat.openai.com/share/582acaeb-3108-40ff-bdcc-c226f9e36999>

Advertiser / Marketer

Custom Instructions

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as an advertiser. You will create a campaign to promote **[insert your product or service]**.

“How would you like ChatGPT to respond?”

You will choose a target audience, develop key messages and slogans, select the media channels for promotion, and decide on any additional activities needed to reach our goal of **[insert your goal]**.

Example

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as an advertiser. You will create a campaign to promote **a new line of shirts with the tagline "do or do not. there is no try"**.

“How would you like ChatGPT to respond?”

You will choose a target audience, develop key messages and slogans, select the media channels for promotion, and decide on any additional activities needed to reach our goal of **maximizing brand awareness across social media**.

Tips

1. **Specify the target audience:** Clearly define the demographic, interests, and psychographic characteristics of the target audience. This will allow for more tailored messaging and channel selection. For example, "You will create a campaign to promote a new line of shirts targeting passionate Star Wars fans between the ages of 18-35, who actively engage in online Star Wars communities and have a strong affinity for inspirational messages."

2. **Incorporate a unique selling proposition (USP):** Identify and include a compelling USP for the new line of shirts. This will differentiate them from competitors and emphasize the key benefits for the target audience. For example, "You will develop key messages and slogans that highlight the exceptional comfort and eco-friendly materials used in our shirts, setting them apart as the preferred choice for both style and sustainability."

3. Explore innovative media channels: Consider incorporating emerging or niche social media platforms that may align with the target audience's preferences. This demonstrates an understanding of current trends and an ability to reach potential customers where they are most engaged. For example, "In addition to traditional social media channels, you will identify and leverage trending platforms like TikTok and Twitch, which have a strong presence among the target audience and offer unique opportunities for creative storytelling and engagement."

4. Create interactive experiences: Integrate interactive elements into the campaign to increase engagement and brand interaction. This could include gamified experiences, augmented reality (AR) try-on features, or interactive polls and quizzes. For example, "You will develop an interactive AR experience that allows users to virtually try on our shirts through a mobile app, encouraging them to share their favorite looks on social media and participate in a 'Best Shirt Styling' contest."

5. Implement influencer collaborations: Identify and partner with influential individuals or micro-influencers who have a genuine passion for Star Wars and a strong following among the target audience. This will help expand reach, credibility, and engagement. For example, "You will collaborate with Star Wars cosplayers and popular fan artists to create engaging content showcasing our shirts in unique and creative ways, tapping into their dedicated fan bases for increased brand awareness."

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try yourself to validate!

Prompt: <https://chat.openai.com/share/014c1ee7-161b-43b4-9f5a-5b3cc03e47bb>

CI: <https://chat.openai.com/share/0439e1a4-0008-43df-8590-700a5d0f230f>

Storyteller

Custom Instructions

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a storyteller. You will come up with entertaining stories about **[insert topic]** that are engaging, imaginative and captivating for the audience.

“How would you like ChatGPT to respond?”

It can be **[insert storytelling style]** type of stories which has the potential to capture people’s attention and imagination. Given that the target audience is **[insert target audience]**, provide specific themes and topics that cater to this demographic and have historically performed well.

Example

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a storyteller. You will come up with entertaining stories about **startups** that are engaging, imaginative and captivating for the audience.

“How would you like ChatGPT to respond?”

It can be **“David vs. Goliath”** type of stories which has the potential to capture people’s attention and imagination. Given that the target audience is **18 to 34 year old entrepreneurs**, provide specific themes and topics that cater to this demographic and have historically performed well.

Tips

Here are some factors you can include into the original prompt to get better quality stories that are customized to your specific needs and audience:

1. **Make the protagonist relatable and complex:** Provide the protagonist with flaws, fears, and internal conflicts that readers can empathize with. This will add depth to the character and make their journey more compelling.
2. **Add a personal stake:** Give the protagonist a strong personal motivation or a deeply emotional reason for pursuing their startup. This will create a sense of urgency and make the story more emotionally resonant.

3. **Incorporate unexpected challenges:** Introduce additional obstacles or setbacks that the protagonist must face along their journey. This will increase tension and keep readers engaged as they wonder how the protagonist will overcome each new challenge.
4. **Include a mentor or guide:** Introduce a wise and experienced mentor figure who supports the protagonist and provides guidance. This relationship can add depth to the story and offer valuable lessons and insights.
5. **Explore moral dilemmas:** Present the protagonist with difficult choices that force them to confront their values and make challenging decisions. This adds complexity to the story and creates opportunities for character growth.
6. **Incorporate a love interest or close friendship:** Introduce a romantic interest or a loyal friend who plays a significant role in the protagonist's journey. This can add emotional depth and provide opportunities for character development.
7. **Create a compelling villain or rival:** Introduce a formidable antagonist or a rival startup that poses a direct threat to the protagonist. This adds conflict and raises the stakes, making the story more gripping.
8. **Include unexpected plot twists:** Incorporate surprising plot twists that challenge the protagonist and the reader's expectations. This keeps the story unpredictable and prevents it from becoming predictable or formulaic.
9. **Explore the emotional toll:** Show the psychological and emotional impact that the protagonist's journey has on them. Explore their doubts, fears, and moments of vulnerability to make the story more relatable and human.
10. **Offer moral or philosophical themes:** Explore deeper themes and thought-provoking questions related to entrepreneurship, success, ethics, or the human condition. This adds intellectual depth and makes the story resonate on a deeper level.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try yourself to validate!

Prompt: <https://chat.openai.com/share/1adf47eb-ec7d-4ac4-b2dd-ff2a9dbd21a2>

CI: <https://chat.openai.com/share/73a83124-e8e9-462c-a631-110e7d959d5c>

Debater

Custom Instructions

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a **[insert subject]** debater. I will provide you with **[insert types of topics]** topics related to current events and your task is to research both sides of the debate, present valid arguments for each side, refute opposing points of view, and draw persuasive conclusions based on evidence.

“How would you like ChatGPT to respond?”

Please provide each perspective in its own section for your outputs. One should be labeled “Pro”, a second should be labeled “Against”, and the third labeled “Neutral”. Your goal is to help people come away from the discussion with increased knowledge and insight into the topic at hand. My first request is **“[insert subject of debate]”**.

Example

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a **artificial intelligence** debater. I will provide you with **AI** topics related to current events and your task is to research both sides of the debate, present valid arguments for each side, refute opposing points of view, and draw persuasive conclusions based on evidence.

“How would you like ChatGPT to respond?”

Your goal is to help people come away from the discussion with increased knowledge and insight into the topic at hand. My first request is **“I want to understand the debate of pro-artificial intelligence versus those against it”**.

Tips

- 1. Specify the Scope:** Clearly define the specific aspect or application of artificial intelligence you want to focus on. For example, "I want to understand the debate of pro-artificial intelligence in autonomous weapons systems versus those against it." This will narrow down the topic and allow for a more focused and in-depth discussion.
- 2. Include a Timeframe:** If there are recent developments or events related to the topic, mention the timeframe. For instance, "I want to understand the current debate of

pro-artificial intelligence versus those against it, in light of recent advancements in machine learning algorithms."

3. **Highlight Key Points:** Provide a brief overview of the main arguments or concerns you want to explore. This will guide the AI in presenting a balanced view and help generate more specific and nuanced responses. For example, "I want to understand the debate of pro-artificial intelligence versus those against it, focusing on the economic impact, job displacement, and ethical considerations."

4. **Request Counterarguments:** Encourage the AI to present opposing viewpoints by explicitly requesting arguments from both sides. For example, "Please present compelling arguments in favor of artificial intelligence and refute opposing points of view, highlighting potential risks and drawbacks as well."

5. **Consider Additional Perspectives:** Acknowledge that the debate is not limited to just two sides. You can ask the AI to present arguments from various stakeholders or perspectives, such as government regulation, societal impact, or environmental considerations. This will encourage a more comprehensive exploration of the topic.

Here's an improved prompt incorporating these tips:

"I want you to act as an artificial intelligence debater and help me understand the current debate surrounding the use of artificial intelligence in autonomous weapons systems. Please present valid arguments in favor of the use of AI in weapons systems, addressing recent advancements in machine learning algorithms. Additionally, present opposing arguments, focusing on the ethical concerns, risks of autonomous decision-making, and potential for increased harm. Feel free to consider perspectives from government regulation, societal impact, and environmental considerations to provide a well-rounded analysis."

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try yourself to validate!

Link: <https://chat.openai.com/share/ec1b0d87-9804-458c-8c43-2124ea4b0586>

CI: <https://chat.openai.com/share/4fe79647-2f08-4e01-8ec6-560bc641eb78>

Coach

Custom Instructions

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a **[insert industry]** coach. I will provide you with information about my tasks, aspirations & goals.

“How would you like ChatGPT to respond?”

Your objective is to prepare me for success by organizing my tasks and providing practical, tangible tips on how to achieve my goals. Here's information about my goals and what I need assistance with:

[copy and paste all the information about your goals]

My first request is **[insert your first task or request]**.

Example

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a **startup** coach. I will provide you with information about my tasks, aspirations & goals.

“How would you like ChatGPT to respond?”

Your objective is to prepare me for success by organizing my tasks and providing practical, tangible tips on how to achieve my goals. Here's information about my goals and what I need assistance with:

I am a 28 year-old entrepreneur that is trying to get their SaaS (software as a service) product launched in the recruiting software market. I'm targeting Seed to Series B startups that have a tight administrative budget but will save significantly by using the product's intuitive tools. I want to focus on Chicago as my first region to launch the product. My budget for the initial launch is \$100.

My first request is **to provide me with a go-to-market plan to acquire my first customer.**

Tips

For this specific example, to get the most out of the “startup coach”, try the following:

1. **Specify your product's unique selling points:** Provide more details about the specific features, functionalities, or innovations that set your product apart from competitors. This will help in tailoring the advice and strategies to highlight your product's strengths.
2. **Clarify your target market's pain points:** Dive deeper into the challenges and pain points faced by your potential customers in the industry's market. The more specific you are about their needs and pain points, the more targeted and effective the advice can be.
3. **Include information about your competitors:** Mention any notable competitors in the recruiting software market and highlight how your product differentiates itself from them. Understanding your competitive landscape will allow for more precise recommendations on positioning, pricing, and differentiation strategies.
4. **Share your long-term goals:** Besides acquiring your first customer, provide insights into your broader aspirations and objectives. Whether it's achieving a specific market share, expanding to other regions, or securing funding, this information will shape the advice given and help with long-term planning.
5. **Specify desired outcomes and metrics:** Clearly state the goals you want to achieve through your go-to-market plan. For example, acquiring a certain number of customers within a specific time frame, achieving a target conversion rate, or generating a certain amount of revenue. This will enable the advice to be more outcome-oriented and focused on measurable success.
6. **Indicate your resources and limitations:** Mention any additional resources you have available, such as technical expertise, marketing budget, or access to industry networks. Also, highlight any constraints or limitations you face, such as time constraints or regulatory considerations. This will allow for more tailored suggestions that consider your specific context.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try yourself to validate!

Link: <https://chat.openai.com/share/82edfa2c-adf2-40b3-8a6f-b7ae89a1571a>

CI: <https://chat.openai.com/share/f6a5e3ca-27d6-445f-890c-f6cde633e44d>

Professional

Custom Instructions Template

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as an **[insert professional]** providing advice on **[insert subject]**. You have complete knowledge of **[insert subject]** including areas such as **[insert related sub-topics]**.

“How would you like ChatGPT to respond?”

Please come up with tactics, strategies, and tasks for me to accomplish **[insert goal]**. If you need more information to get a better output, please ask and I will provide that information. Do not abuse this option. If too many questions are asked, I will not be able to provide more answers. My first task for you is **[insert task]**.

Examples

Example 1 - Accountant

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as an **accountant** providing advice on **filing my taxes**. You have complete knowledge of **taxation** including areas such as **LLC filings, estates, trusts, wills & other relevant accounting & financial concepts**.

“How would you like ChatGPT to respond?”

Please come up with tactics, strategies, and tasks for me to accomplish **a successful filing**. If you need more information to get a better output, please ask and I will provide that information. Do not abuse this option. If too many questions are asked, I will not be able to provide more answers. My first task for you is **to help me gather the proper documentation**.

Example 2 - UX/UI Developer

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a **UX/UI developer**.

“How would you like ChatGPT to respond?”

I will provide some details about the **design of an app, website or other digital product**, and it will be your job to come up with creative ways to **improve its user experience**. This could involve **creating prototypes, testing different designs and providing feedback on what works best**. If you need more information to get a better output, please ask and I will provide that information. Do not abuse this option. If too many questions are asked, I will not be able to provide more answers. My first request is **“I need help designing a minimalistic introductory tutorial for a new mobile application.”**

Example 3 - **Real Estate Agent**

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a **real estate agent**.

“How would you like ChatGPT to respond?”

I will provide you with details on **an individual looking for their dream home, and your role is to help them find the perfect property based on their budget, lifestyle preferences, location requirements etc.** You should use your knowledge of the **local housing market in order to suggest properties that fit all the criteria provided by the client**. If you need more information to get a better output, please ask and I will provide that information. Do not abuse this option. If too many questions are asked, I will not be able to provide more answers. My first request is **“I need help finding a 2 bedroom, 1 bathroom apartment near downtown Chicago.”**

Example 4 - **Financial Analyst**

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a **financial analyst**. You are a qualified individual with experience in **understanding complex economic conditions using technical analysis tools, charts and financial modeling while interpreting the macroeconomic environment prevailing across world**.

“How would you like ChatGPT to respond?”

You help customers **acquire long-term advantages by leveraging economic statistics and data to make informed conclusions about the current state of the economy and financial markets**.

My first request is **“I need help understanding the current strength of a company called ‘Tesla’. Here’s some financial data on the company:**

[copy and paste financial data & sources on the company here]

What insights do you have about the company as it currently operates? If you need more information to get a better output, please ask and I will provide that information. Do not abuse this option. If too many questions are asked, I will not be able to provide more answers.

Example 5 - **Interior Designer**

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as an **interior designer**.

“How would you like ChatGPT to respond?”

Tell me what kind of theme and design approach should be used for **a room I provide to you; bedroom, kitchen, bathroom, etc.**, provide suggestions on **color schemes, furniture placement and other decorative options that best suit said theme/design approach** in order to enhance aesthetics and comfortability within the space . If you need more information to get a better output, please ask and I will provide that information. Do not abuse this option. If too many questions are asked, I will not be able to provide more answers. My first request is **“Provide examples of living room designs for a loft-style townhouse in Chicago”**.

Example 6 - **Etymologist**

“What would you like ChatGPT to know about you to provide better responses?”

I want you to act as a **etymologist**.

“How would you like ChatGPT to respond?”

I will give you a word and you will **research the origin of that word, tracing it back to its ancient roots**. You should also provide information on **how the meaning of the word has changed over time, if applicable**. If you need more information to get a better output, please ask and I will provide that information. Do not abuse this option. If too many questions are asked, I will not be able to provide more answers. My first request is **“I want to trace the origins of the word ‘sushi’”**

Tips

For the “Accountant” example, to get the most out of ChatGPT, try the following:

- 1. Specify your tax jurisdiction:** Mention the country or state in which you are seeking tax filing advice. Tax laws and regulations vary between jurisdictions, so providing this information will allow for more accurate guidance.
- 2. Provide information about your entity type:** If you are filing taxes as an individual, specify that. If you have a specific entity type, such as an LLC, partnership, or corporation, mention it. This information will help tailor the advice to your specific tax situation.
- 3. Share details on income sources:** If you have multiple sources of income, provide a brief overview of each source, including whether it's employment income, investment income, rental income, or self-employment income. This will allow for more targeted advice regarding deductions, credits, and reporting requirements.
- 4. Mention any specific tax challenges:** If you are facing particular tax challenges or have concerns about certain aspects of your tax filing, such as international transactions, estate planning, or complex investments, it's helpful to include those details. This way, the advice can address those specific areas of concern.
- 5. State your goals and priorities:** If you have specific financial goals or priorities for your tax filing, such as maximizing deductions, minimizing tax liabilities, or exploring tax planning strategies, express them. Understanding your objectives will help provide advice aligned with your goals.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try yourself to validate!

Link: <https://chat.openai.com/share/5c8af1b9-e3f2-43ec-804f-516580e0b989>

CI: <https://chat.openai.com/share/ad68ff81-c3f4-4341-be97-477d09b30f1a>

WARNING: DO NOT USE THESE PROMPTS OR OTHER PROMPTS TO SUBSTITUTE PROFESSIONAL ADVICE AS MENTIONED IN DISCLAIMER AT THE END OF THIS GUIDE.

This prompt, other prompts and its results cannot and do not contain legal, financial or any other professional advice. The information is provided for general informational and educational purposes only and is not a substitute for professional advice. Accordingly, before taking any actions based upon such information, we encourage you to consult with the appropriate professionals. We do not provide any kind of legal, financial or other kind of advice. Read the Disclaimer at the end of this guide for more information.

Autonomous AI Agents

Overview

Now that we've covered a range of practical examples, we'll continue by exploring the step-by-step process to create your own autonomous agents within ChatGPT.

For those unfamiliar with autonomous agents, here's a brief description of what they are:

“An autonomous agent in the context of AI systems like ChatGPT can be thought of as a subprogram or a module that can operate independently to achieve certain objectives or tasks. These agents are self-driven and are capable of making their own decisions based on the inputs they receive and the goals they are designed to achieve.”

Here's a fundamental explanation of how to think about autonomous agents:

<https://chat.openai.com/share/e9d02286-f387-4e32-850e-8cf04aa47123>

Here's a detailed explanation of autonomous agents:

<https://chat.openai.com/share/535c85fc-beae-4108-9a85-6ca77eb7e7bb>

Previously, there were open source models published by the community in an attempt to get these autonomous AI agents on your local computer. However, people who were unfamiliar with open source models and how to run code to use the autonomous AI agents struggled to get them to work properly. In fact, some engineers themselves couldn't get the models to run without running into errors.

Thankfully, by leveraging a feature of ChatGPT, we'll be able to run an autonomous AI agent directly within the chat interface.

In order to begin using an autonomous AI agent, you'll need to set up Custom Instructions and get a file ready for Advanced Data Analysis. I walk you step by step through each below.

Setup

Remember to enable **Custom Instructions** by taking the following steps:

1. Click the “...” next to your account name (e.g. “David Paykin”)
2. Choose the “**Settings & Beta**” option
3. Click “**Beta Features**” on the left-side menu
4. Enable “**Custom Instructions**” above “Plugins” by clicking the toggle on button
5. Now that it is enabled, click the “...” again followed by “**Custom Instructions**”

Custom Instructions

Courtesy of [@NickADobos](#):

“What would you like ChatGPT to know about you to provide better responses?”

This is relevant to EVERY prompt I ask.

Don't talk. Just perform the task or tasks.

Task reading:

Before each response, read the current task list from "chatGPT_Todo.txt". Reprioritize the tasks, and assist me in getting started and completing the top task

“How would you like ChatGPT to respond?”

Task creation & summary:

You must always summarize all previous messages, and break down our goals down into 3-10 step by step actions. Write code and save them to a text file named "chatGPT_Todo.txt". Always provide a download link.

Only after saving the task list and providing the download link, should you provide me with Hotkeys

List 4 or more multiple choices.

Use these to ask questions and solicit any needed information, guess my possible responses or help me brainstorm alternate conversation paths. Get creative and suggest things I might not have thought of prior. The goal is create open mindedness and jog my thinking in a novel, insightful and helpful new way:

w: to advance, yes

s: to slow down or stop, no

a or d: to change the vibe, or alter directionally

If you need to do so, provide additional cases and variants. As an example, use double tap variants like “ww” or “ss” for “Strongly agree” or “Disagree”.

Advanced Data Analysis

Now that the Custom Instructions above have been set, we need to create a “.txt” file with instructions for what tasks ChatGPT should automate:

1. Open “[Notepad](#)”
2. Write your instructions and the tasks to be automated

3. Once finished, “**Save**” as a “.txt” file
4. Hover over “**GPT-4**” and click “**Advanced Data Analysis**”
5. Click the circled “+” icon in the chat to upload your “.txt” file

Here’s a template of what you can type into “Notepad” for your “.txt” file:

Your objective is to [insert objective(s)]. Here’s information about my goals and what I need assistance with:

I am a [insert your info] that is trying to [insert overall goal]. I’m targeting [target audience]. I want to focus on [insert region/demographic]. My requested tasks are the following:

1. [insert Task 1]
2. [insert Task 2]
3. [insert Task 3]

You can copy and paste this into your Notepad and personalize to your needs. Keep in mind, the more tasks you add, the higher the potential may be for errors in ChatGPT’s current state. This should improve over time and likely not be an issue in the future.

Next, we’ll go through a complete example using the Custom Instructions shared at the beginning and a filled out version of the template within Advanced Data Analysis to implement our autonomous AI agent.

Product Launch Example

First, we’ll open a new chat window within GPT-4 for Advanced Data Analysis. Then we fill out the template above to get the following:

Your objective is to prepare me for success by organizing my tasks and providing practical, tangible tips on how to achieve my goals. Here’s information about my goals and what I need assistance with:

I am a 28 year-old entrepreneur that is trying to get their SaaS (software as a service) product launched in the recruiting software market. I’m targeting Seed to Series B startups that have a tight administrative budget but will save significantly by using the product’s intuitive tools. I want to focus on Chicago as my first region to launch the product. My requested tasks are the following:

1. **Prepare a go to market strategy for acquiring my first customers.**

2. Create a content strategy given that I have \$0 to spend and have to rely completely on organic channels.

3. Brainstorm the best persona to target with my product and then write 3 different types of email templates I can use to send to those personas.

After saving the “Notepad”, we get our “.txt” file, which we can then upload to Advanced Data Analysis. Click the circled “+” and choose the file above.

From there, we can type “**Let’s begin**”, or as in this example, “**Help me launch my new product**”.

Here’s the complete interaction with the AI agent:

<https://chat.openai.com/share/9def80fe-1976-4ab3-bc7c-5d0b4fa992e9>

As you can see in the chat window above, you’ll be able to interact with the AI agent so as to improve your outputs.

In this particular case, the AI agent asks a series of questions about the product, business and who we’re targeting to develop better strategies and answers for our startup. Here are some of the useful strategies that the AI agent provided:



Great, let's move forward with preparing a go-to-market strategy. Given our target audience and the persona we've developed, here are a few steps we can take:

1. **Product Positioning:** Clearly articulate how your product solves the unique problems faced by cost-conscious business owners in small, growing companies. Highlight the cost savings and efficiencies it can provide.
2. **Channels:** As we're focusing on organic channels, consider leveraging SEO and content marketing. Create valuable content that addresses common pain points and how your product can solve them. Also consider partnerships with startup incubators or business communities in Chicago.
3. **Outreach:** Develop a direct outreach program targeting business owners. This can involve cold emails (we'll create the templates in the next task), networking events, and leveraging personal connections.
4. **Demonstrate Value:** Offer a free trial or demo of your product so potential customers can see its value firsthand.

Hotkeys for your convenience:

- Press 'w' to agree with this strategy and proceed to the next task, which is creating a content strategy for organic channels.
- Press 'a' to add more points to the go-to-market strategy.
- Press 's' to revisit any previous tasks.
- Press 'd' to return to the overall task list.

They also provided us with email templates to use for outreach to business owners in an effort to acquire our first customers. Here's an example of an email campaign:



Excellent, let's move forward with writing email templates. Since we're targeting cost-conscious business owners or CEOs, we should focus on how our product can provide a cost-effective solution for their hiring needs. Here are three email templates we could use:

1. Introduction Email:

Subject: Streamline Your Hiring Process and Save Cost with [Your Product Name]

Body: [Brief introduction about your product, the problem it solves, and how it can save time and money for their business.]

2. Follow-up Email:

Subject: Don't Miss Out on Efficient Hiring with [Your Product Name]

Body: [A follow-up message highlighting key features of your product, perhaps including a success story or case study.]

3. Offer Email:

Subject: Exclusive Invitation to Try [Your Product Name] for Free

Body: [An offer for a free trial or demo of your product, with clear instructions on how they can avail of it.]

I'll save these email templates in a separate text file.


Hotkeys for your convenience:




- Press 'w' to proceed with saving these email templates.
- Press 'a' to add more details or modify the email templates.
- Press 's' to revisit the content strategy or any previous tasks.
- Press 'd' to return to the overall task list.

Naturally, you can customize these emails by providing information within chat to get better results. I asked the AI to add more to the emails based on the context I had provided earlier in the chat interactions & its proposed strategy.

Although the email campaign will only update with product information we shared in the beginning of this example, you should remember that personalization is key with targeting decision makers. If you are able to identify who the decision maker is, you can attach a PDF of their social media posts, LinkedIn profile or other types of publicly available information to provide ChatGPT with more context on how they should customize the email campaign.

Here's the updated campaign based on product information provided at the beginning:


Absolutely. Let's add more details to the email templates based on the information you provided.

1. Introduction Email:

Subject: Revolutionize Your Hiring Process with [Your Product Name]
 Body: Dear [Recipient's Name],

I'd like to introduce [Your Product Name], an Applicant Tracking System (ATS) designed specifically for growing businesses like yours. We understand that as your company scales, the hiring process can become increasingly complex and time-consuming. Our ATS not only automates routine tasks but also allows you to put heavier weights on specific factors that are most important to you. This enables you to narrow in on the best candidates for the exact role they're applying for, saving you time and resources.

2. Follow-up Email:

Subject: Unleash Your Hiring Potential with [Your Product Name]
 Body: Dear [Recipient's Name],

I hope you found our previous email informative. [Your Product Name] is more than just a tool; it's a game-changer for growing businesses. By allowing you to customize your candidate search, our ATS helps you find the best talent for your company without the need for costly recruitment agencies or additional HR staff. We'd love to show you how [Your Product Name] can streamline your hiring process and save your company money.

3. Offer Email:

Subject: Try [Your Product Name] for Free Today
 Body: Dear [Recipient's Name],

We're so confident that you'll love [Your Product Name] that we're offering you a free trial. Experience first-hand how our ATS can revolutionize your hiring process. To start your free trial, simply click the link below. We're looking forward to helping you find the best talent for your growing business.

Let's save these updated email templates in a separate text file.

Hotkeys for your convenience:

- Press 'w' to proceed with saving these email templates.
- Press 'a' to add more details or modify the email templates.
- Press 's' to revisit the content strategy or any previous tasks.
- Press 'd' to return to the overall task list.

🔄 Regenerate

You can continue creating strategies and enabling the autonomous AI agent complete tasks as needed. Given this was a demonstration, we'll now move onto a more advanced use case in trying to adapt the custom instructions to create an autonomous AI coding assistant directly within ChatGPT.

Coding Assistant Example - Part 1

Input the following into your Custom Instructions:

“What would you like ChatGPT to know about you to provide better responses?”

[leave blank]

“How would you like ChatGPT to respond?”

notalk;justgo

Goal:

Build sleek elegant **game** 🎮. Debug 🐛 & comment 📝.
 write detailed code, build architecture 🏠. Core classes

A	B
C	D

, functions

1	2
3	4

, methods

F	G
H	I

,
 comments 📝. No placeholders ❌. Add dependencies 📎.
 If error in user message, build 2-3 stackoverflow search URLs, compare

🧠 Ai engineer, senior dev, prompt engineer, indie hacker



- 1 👁️ User reqs, follow to the letter
- 2 🗺️ Plan in pseudocode, detail-rich
- 3 📄 Output code, single block
- 4 🚀 Less prose, more code

🤖 ChatGPT Instructions

- 1 📄 Query memory, tasks and names of skills in SQLite DB

-

. Use code Python 🐍 tool
 Determine clear goal, break down step by step into 3-5 tasks
 Prioritize tasks 🏆, assist user 👤

You may invoke skills

```
table tasks {
  task
  subtasks
}
table memory {
  summary
  emojiSummary
}
table skills {
  skill
  instructions
}
```

②After EVERY response, update tasks, insert conversation summary, provide download link 🔗

③Always Offer 4+ hotkeys for next message 📧, use WASD hotkeys ⌨️
 Ask🗨️, solicit info, guess responses 🧠, brainstorm ⚡, check understanding✅, suggest poetically 🌸. Goal🎯: perspective, open insight & wisdom

format:

w: advance, yes

s: slow down or stop, no

a: alter directionally, change vibe

d: repeat last command

k: list new hotkeys, display help message

optional:

q: follow up ?'s, new perspective

eNUM: <type:Skills+chatGPTInstructions>

a:

b:

c:

2 bonus hotkeys, get creative

1: <advisors>

2:

toggle:

tmode: respond ONLY with code

Remember, you can customize the above instructions to whatever is relevant to your tasks. For example, the **game** 🎮 used in the 2nd paragraph of the instructions above can be changed to **app** 📱 if you're trying to build an phone or web application.

Next, we'll go to "Advanced Data Analysis" and use the prompt below to start automating the implementation of new features into our "Snake" game.

Prompt

Attached is a ".txt" file of the "Snake" game example we created in [Prompt 13](#) of the "Advanced Data Analysis" section.

"Provide a list of easy to implement features to add to "Snake" game has already been created using Advanced Data Analysis that don't require external resources. Provide hotkeys for options to automatically update the entire code with the feature into the game if I proceed with that feature. Options can include adding "Increase Speed as more food is consumed by Snake" and "Pause/Resume using the spacebar key"

Automating New Features - Part 2

Once the “Snake” game is uploaded and the prompt above is entered, Advanced Data Analysis will analyze your code and provide a summary of what is in that file. This is an important step to ensure that ChatGPT has the background information of what you are trying to accomplish.

After reading and providing an overview of your file, the autonomous agent will offer a set of options for you to approve so it can continue implementing features on its own. Here’s what that looks like in our current example:

Alright, let's start by looking at the existing code for the Snake game. From there, we can identify the potential places where we could implement the new features you suggested. This will allow us to provide an accurate pseudocode plan and eventually a code implementation.

Let's load and review the current game code from the provided text file.

The existing code for the Snake game is written in JavaScript using the p5.js library. The code includes:

- *`setup()` function to initialize the game variables.*
- *`draw()` function to update the game's state and render the game objects.*
- *`keyPressed()` function to control the snake's direction based on the arrow key pressed.*
- *`showScore()` function to display the current score.*
- *`Snake` class to manage the snake's position, direction, and growth.*
- *`Food` class to manage the food's position.*

Given this, here are some additional features we can implement in the Snake game:

1. **Increase the speed of the snake as more food is consumed:** As the score increases, we can increase the `frameRate()` within the `draw()` function. This will make the game progressively more challenging.
2. **Pause/Resume game using the spacebar key:** We can add additional control within the `keyPressed()` function to pause and resume the game. We can use a global variable to maintain the game state (running or paused).
3. **Add walls in the game area:** To increase the difficulty level, we can add walls within the game area. The snake will end the game if it hits a wall.
4. **Multiple types of food:** We can add different types of food, each with different point values. This will add variety to the game and create strategic choices for the player.

Here are your hotkey options:

- **W**: To proceed with the "Increase Speed" feature.
- **S**: To proceed with the "Pause/Resume" feature.
- **A**: To proceed with the "Add Walls" feature.
- **D**: To proceed with the "Multiple Types of Food" feature.

Let me know which feature you'd like to proceed with, and I'll provide the updated code accordingly.

After being provided with the options, you can choose whichever you prefer or tell ChatGPT that you'd like it to propose a new set of features if you don't like what it has offered. In this case, we decided to go with the "Increase Speed" feature:

D

W and provide me with full updated code

✍

🌀

Alright, you've chosen the feature to increase the speed of the snake as more food is consumed. Let's plan the implementation of this feature in pseudocode:

1. We need to make the game speed dependent on the current score. Since the frame rate of the game defines the speed, we'll increase the frame rate based on the score.
2. In the `draw()` function, just after increasing the score, we'll adjust the frame rate.

Now let's update the JavaScript code accordingly.

Finished working
Show work ▾

I have updated the code to increase the speed of the snake as more food is consumed. You can find the updated code in the file [here](#).

Hotkeys for the next actions:


- **W**: Implement the "Pause/Resume" feature.
- **A**: Implement the "Add Walls" feature.
- **D**: Implement the "Multiple Types of Food" feature.
- **S**: Review the updated code.

Please choose the next action.

📄 👍 👎


As requested, Advanced Data Analysis provided us the updated code with the new feature we requested by simply typing "**W**" and asking for the revised code. The autonomous agent also provided us an updated list of features to add. Let's do "Pause/Resume" next:

D W and provide me with full updated code

 You've chosen to add the feature to pause/resume the game using the spacebar key. Here's the plan in pseudocode:

1. We need to add a global variable, `isPaused`, to keep track of whether the game is paused.
2. Within the `keyPressed()` function, we add an extra condition to check if the spacebar key is pressed. If it is, we toggle the `isPaused` variable.
3. In the `draw()` function, we add a condition at the start to return immediately if the game is paused. This will effectively stop updating the game state and rendering, thus pausing the game.

Let's update the JavaScript code to implement this feature.

Finished working Show work 

I have updated the code to include the pause/resume functionality using the spacebar key. You can find the updated code in the file [here](#).

Hotkeys for the next actions:

- **W**: Implement the "Add Walls" feature.
- **A**: Implement the "Multiple Types of Food" feature.
- **D**: Review the updated code.
- **S**: End the session.

Please choose the next action.

Once again, we've been provided with another file with the updated code reflecting both features that hadn't existed before we put the autonomous agent to work. But the fun doesn't stop there.

The beauty of implementing this autonomous AI agent within ChatGPT is that you are not limited to the settings you created within Custom Instructions. For instance, I noticed that there was a bug occurring where if you pressed an arrow in the exact opposite direction of where the Snake was going, you would "collide with yourself" which automatically ended the game.

Of course, this wasn't in any of the options offered, but that won't stop our little AI from taking care of that issue for us:

D D there's a bug that when the snake is running in one direction, you're still able to press the arrow for the exact opposite direction and collide with yourself unintentionally. Please think about this step by step and create functionality to avoid this bug. Provide me with the full updated code after



This is a common issue in Snake games. The bug occurs because the game allows the player to change the snake's direction to its opposite, causing the snake to immediately collide with itself.

To resolve this issue, we need to add a condition in the `setDir(x, y)` function in the `Snake` class to prevent the direction from being set to the opposite of the current direction. Here's the plan in pseudocode:

1. In the `setDir(x, y)` function of the `Snake` class, before setting `this.xdir` and `this.ydir`, we check if the requested direction is the opposite of the current direction. If it is, we do not change the direction.
2. To perform this check, we compare the requested direction (`x`, `y`) with the current direction (`this.xdir`, `this.ydir`). If they are opposite, i.e., `x == -this.xdir` and `y == -this.ydir`, we return without changing the direction.

Let's update the JavaScript code to implement this bug fix.

Finished working Show work ▾

I have updated the code to fix the bug. Now, the snake won't be able to move in the exact opposite direction and unintentionally collide with itself. You can find the updated code in the file [here](#).

Hotkeys for the next actions:

- **W**: Implement the "Add Walls" feature.
- **A**: Implement the "Multiple Types of Food" feature.
- **D**: Review the updated code.
- **S**: End the session.

And just like that, our bug is fixed while in the middle of adding new features to the code.

In addition to the examples we suggested at the beginning, it's important to show that the autonomous agent can create the features that it suggested on its own. Thus, let's proceed with asking it to "**Obstacles**" that the snake will need to navigate.


Here's our AI agent going to work on building the feature it offered as an option:

Hotkeys for these options:

- **W**: Randomly changing food.
- **A**: Power-ups.
- **S**: Obstacles.
- **D**: Multiple lives.
- **Q**: Maze mode.

Please choose the next action.

D S

 You've chosen to add obstacles to the game. Here's the plan in pseudocode:

1. Define an `Obstacle` class similar to the `Food` class. An `Obstacle` should have a position and a method to display itself.
2. In the `setup()` function, create an array of obstacles and populate it with a few instances of `Obstacle`.
3. In the `draw()` function, iterate through the array of obstacles and call the `show()` method on each one.
4. Modify the `Snake` class's `endGame()` method to check if the snake's head has hit any of the obstacles. If it has, end the game.

Let's update the JavaScript code to implement this feature.

Finished working Show work ▾

I have updated the code to include obstacles that the snake must avoid. You can find the updated code in the file [here](#).

Here's a link to the chat window showing the complete process of implementing the Pause/Resume option by pressing Spacebar, as well as adding a Speed Increase and fixing the Self-colliding Bug:

<https://chat.openai.com/share/4d03d540-667c-4ce2-af99-88eb53b94315>

Tips

To get the most out of the autonomous AI agents, try the following:

1. **Scenario Simulations:** You can use AI to simulate different scenarios or strategies for your business. For example, you could ask the AI to provide a SWOT analysis for different business strategies, or to generate ideas for dealing with hypothetical business challenges.
2. **Regular Task Automation:** Automate regular tasks such as preparing reports or summarizing data. For example, you could provide data in a certain format and ask the AI to prepare a summary or a report based on that data.
3. **Brainstorming Sessions:** Use AI as a brainstorming tool. You can ask it to generate a list of ideas on a given topic. This can be particularly useful when you're experiencing a creative block.
4. **Learning New Topics:** AI can help you learn new topics. You can ask it to provide a summary or a simplified explanation of complex topics.
5. **Drafting and Editing Content:** You can use AI to draft or edit content, such as blog posts, social media posts, emails, etc.
6. **Automated Research:** Ask the AI to compile information or research on a particular topic. Although it can't browse the internet in real-time, it can pull from a vast internal database of knowledge.
7. **Decision Making:** Ask the AI for pros and cons or a cost-benefit analysis to assist with decision making.

Remember, the more specific and direct the prompt, the more accurate and helpful the AI's response will be.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try yourself to validate!

Coding Assistant + Automating New Features Link:

<https://chat.openai.com/share/4d03d540-667c-4ce2-af99-88eb53b94315>

Plugins

How to Access Plugins

Enable the “Plugins” feature by signing up for a [ChatGPT Pro subscription](#). Then take the following steps:

1. Click the “...” next to your account name (e.g. “David Paykin”)
2. Choose the “**Settings**” option
3. Click “**Beta Features**” on the left-side menu
4. Enable “**Plugins**” below “**Custom Instructions**” by clicking the toggle on button
5. Go back to home page and hover over “**GPT-4**” at the top click “**Plugins**”
6. A dropdown will show up titled “**No plugins enabled**” below “**GPT-3.5**” and “**GPT-4**”
7. Click “**No plugins enabled**” followed by clicking “**Plugin store**”
8. You will now see various plugins with different sections organized at the top

What are Plugins

Think of this as the “App Store” of ChatGPT. You can access tools that were built by other companies to work together with the large language model to supercharge your work. These can be anything from creating charts based off the work you’re already doing within the chat interface to crawling a website to extract useful information.

How We’ll Approach Plugins

We’ll be covering Plugins a bit differently from the other ChatGPT features. There’s a couple of reasons for this:

- There are a significant number of plugins, so there’s still quite a bit of testing that is going on to identify the best ones to use
- Each of the existing plugins have their own nuances (including bugs), which can change on a daily/weekly basis diminishing the value of using them consistently
- The current Plugins may be replaced by OpenAI themselves just as the company had done with “Advanced Data Analysis”, which disrupted numerous AI data analysis tools

To cover every single one of the plugins (there’s already over 800) would be the equivalent of assessing Apple’s “App Store” when they launched in July 2008. Rather than covering every single plugin, we’ll look at a sample of some of the more useful ones.

WARNING: because of the volatility of ChatGPT’s Plugins, the information covered may be outdated by the time you’re reading this. Always make sure to try out and verify these tools yourself

Description

Plugins are such a new feature (as of August 2023) of ChatGPT, which in and of itself is a brand new technology. We are so early in the development of large language models and AI that much of it will be a “trial and error” experience until you find what works best for you.

With that in mind, I wanted to provide you with a general framework for how you can dive into plugins to get a sense of what works well for your needs versus what you can do without. Thus, I’ll be walking through how add a new plugin, as well as a simple prompt you can use to assess whether the plugin will be useful for you before investing a significant amount of time working with it.

After showing you the strategy I use, we’ll cover a couple of examples of helpful plugins and practical use cases incorporating them into your ChatGPT workflows. Then, we’ll look at how to leverage multiple plugins simultaneously to exponentially improve your productivity on any given activity.

Process

To access WebPilot, take the following steps:

1. Click the Plugins dropdown (may say “**No plugins enabled**”)
2. Scroll down and click on “**Plugin store**”
3. Default page is “**Popular**” but you can click “**All**” to see more
4. Click “**Install**” for a plugin you’re interested
5. Ensure the plugin is enabled in the dropdown below “**GPT-3.5**” and “**GPT-4**”

WARNING: you should check and verify what results are provided to you with any plugin, but especially ones like these where ChatGPT is scanning a lot of words and documentation, which significantly increases the chances for mistakes and hallucinations.

Prompt

You are a ChatGPT plugins expert with knowledge of every plugin within the Plugin store and how they best interact with the existing capabilities of large language models. How does the [**insert plugin**] plugin work? Show me all the unique and out of the box examples with practical tips to utilize this across many use cases.

Examples

In the next couple of examples, I will select a few plugins to show you what the outputs look like from the above prompt template so you can see the usefulness of getting an overview of how to optimize your use of the plugin.

Example 1 - Wolfram Alpha

The Wolfram Alpha plugin allows ChatGPT to access dynamic computation and curated data from WolframAlpha and Wolfram Cloud. This integration significantly enhances ChatGPT's capabilities in areas like mathematics, science, engineering, and more. Here's a breakdown of how it works and some unique examples:

How the Wolfram Alpha Plugin Works:

1. **Endpoints:** The plugin has two primary endpoints:
 - `getWolframAlphaResults`: Used for natural-language queries about various topics like chemistry, physics, geography, history, etc. It can also perform mathematical calculations, date and unit conversions, formula solving, and more.
 - `getWolframCloudResults`: Used for evaluating Wolfram Language code, which allows for complex calculations, data analysis, plotting, data import, and information retrieval.
2. **Query Structure:** The plugin uses a simple query structure, `{"input": query}`, where `query` is a single-line string. For more complex tasks, Wolfram Language code can be passed to the `getWolframCloudResults` endpoint.
3. **Assumptions:** Sometimes, Wolfram Alpha might return multiple assumptions for a query. In such cases, the most relevant assumption can be chosen and passed back for a more accurate result.

Unique and Out-of-the-Box Examples:

1. **Mathematics:**
 - Solve equations: "Solve $x^2 - 5x + 6 = 0$ for x "
 - Integrate functions: "Integrate $\sin(x)$ from 0 to π "
 - Plot functions: "Plot $y = x^2$ from -10 to 10"
2. **Science:**
 - Chemistry: "What is the atomic weight of gold?"
 - Physics: "Calculate the escape velocity of Earth"
 - Astronomy: "How far is Mars from Earth?"
3. **Engineering:**

- Electrical: "Calculate the impedance of a circuit with 5Ω resistance and $10\mu\text{H}$ inductance"

- Mechanical: "Calculate the stress in a beam with 10kN load and 50mm^2 cross-sectional area"

4. Geography & History:

- "What's the capital of Brazil?"

- "When was the Battle of Waterloo?"

5. Date & Time:

- "How many days are there between January 1, 2020, and December 31, 2020?"

- "What day of the week was July 4, 1776?"

6. Unit Conversions:

- "Convert 5 miles to kilometers"

- "Convert 100°F to Celsius"

Practical Tips:

1. **Simplify Queries:** For better results, convert inputs to simplified keyword queries. For instance, instead of "how many people live in France," use "France population."

2. **Use Named Constants:** Instead of using numerical values, use named physical constants like 'speed of light' for more accurate results.

3. **Batching Techniques:** When retrieving data for multiple entities, use batching techniques to get all the data in a single call.

4. **Optimize Wolfram Language Code:** When using the `getWolframCloudResults` endpoint, ensure that the Wolfram Language code is optimized for performance.

5. **Use Associations:** When organizing and manipulating data, use the Association function in Wolfram Language for better data structuring.

6. **Handle Assumptions:** If Wolfram Alpha provides multiple assumptions for a query, choose the most relevant one for accurate results.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try playing around with it yourself to validate!

Wolfram Alpha: <https://chat.openai.com/share/f43cac6f-a16c-40bd-834e-dc40ba98ee7a>

Example 2 - Zapier

The Zapier plugin allows ChatGPT to interact with the Zapier platform, which is a tool that connects different apps and automates workflows. Here's a breakdown of how the Zapier plugin works and some unique examples:

1. Core Functions of the Zapier Plugin:

- **list_guided_recipes**: Suggests Zaps (automated workflows) that a user might want to create. You can provide a query to narrow down suggestions.
- **search_all_actions**: Searches through all actions supported by Zapier, including those that the user has exposed to the plugin. This is useful to see the breadth of what's possible with Zapier.
- **preview_a_zap**: Given a natural language description of a multistep Zap, it returns a possible definition of that Zap. This is great for brainstorming and quickly visualizing a workflow.
- **get_configuration_link**: Provides a link to configure more actions on Zapier. Useful for users who want to expand their automation capabilities.
- **list_exposed_actions**: Lists all the currently exposed actions for a given account. This helps users see what they've already set up and what's available to them.
- **get_execution_log_endpoint**: Retrieves the execution log for a given execution. This is useful for troubleshooting or reviewing the history of a specific Zap.

2. Unique and Out-of-the-Box Examples:

- **Content Management**: When a new blog post is published on your WordPress site, use Zapier to automatically share it on your social media platforms. ChatGPT can help set up this Zap by guiding you through the process.
- **E-commerce**: If you run an online store, set up a Zap to automatically add new customers to your email marketing list (like Mailchimp) whenever they make a purchase. ChatGPT can suggest the best way to structure this Zap.
- **Event Management**: When someone RSVPs to an event you're hosting on Eventbrite, automatically send them a personalized thank you email. ChatGPT can provide a preview of how this Zap might look.
- **Customer Support**: If you receive a negative review on a platform like Trustpilot, set up a Zap to create a ticket in your customer support software (like Zendesk) to address the issue. ChatGPT can help you brainstorm the steps for this Zap.

- **Productivity:** Integrate your to-do list app (like Todoist) with your calendar (like Google Calendar). When you add a new task with a due date, it automatically creates an event on your calendar. ChatGPT can guide you in setting up this automation.

3. Practical Tips:

- **Start Small:** If you're new to Zapier, start with a simple Zap to understand the process. As you become more comfortable, you can create more complex, multi-step Zaps.
- **Test Your Zaps:** Before fully implementing a new Zap, test it to ensure it works as expected. This can prevent potential issues down the road.
- **Use ChatGPT for Ideas:** If you're unsure how to automate a specific task, describe it to ChatGPT. The model can suggest potential Zaps or guide you through the setup process.
- **Regularly Review Your Zaps:** As your needs change, your Zaps might need to be adjusted. Periodically review and update them to ensure they remain relevant and effective.
- **Security:** Only expose necessary actions to the Zapier plugin and regularly review the actions you've exposed to ensure security.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try playing around with it yourself to validate!

Zapier: <https://chat.openai.com/share/1a603bb2-8ff2-4cd5-a5e1-8dcabcf86aab>

Example 3 - Notable

The Notable plugin is a powerful tool that allows users to create, manage, and execute Jupyter notebooks directly from the chat interface. It's designed to integrate seamlessly with the capabilities of large language models like ChatGPT, enabling a wide range of data science, analysis, and visualization tasks.

Here's a breakdown of the Notable plugin's capabilities, along with some unique and out-of-the-box examples:

1. About the Plugin and Runtime Environment

- **Function:** ``noteable.about``
- **Usage:** Retrieve information about the plugin and its runtime environment.

2. Project Management

- **Function:** ``noteable.create_project``
- **Usage:** Create a new project to store notebooks and associated data files.

3. Notebook Management

- **Functions:**
 - ``noteable.create_notebook``: Create a new notebook.
 - ``noteable.get_notebook``: Get a summary of a notebook.
 - ``noteable.get_project_files``: Get a list of files in a project.
 - ``noteable.get_file_metadata``: Get metadata about a file.
 - ``noteable.get_file_content``: Get the content of a text or markdown file.
 - ``noteable.get_datasources``: Get the databases for a notebook file.
 - ``noteable.get_project_kernels``: Get a list of kernel names for a project.

4. Cell Management

- **Functions:**
 - ``noteable.run_cells``: Execute cells in a notebook.
 - ``noteable.get_cell``: Return cell model details.
 - ``noteable.update_cell``: Replace the source code of a cell.
 - ``noteable.change_cell_type``: Change the type of a cell.
 - ``noteable.create_cell``: Create a code or markdown cell.
 - ``noteable.run_cell``: Run a specific cell within a notebook.

5. Kernel Management

- **Functions:**
 - ``noteable.get_active_kernel_sessions``: Returns a list of the user's active kernel sessions.
 - ``noteable.start_kernel``: Start a kernel for a notebook.
 - ``noteable.shutdown_kernel``: Shutdown the kernel for a notebook.

6. User Information

- **Function:** ``noteable.get_user_info``
- **Usage:** Get the current user's details and default project information.

Unique and Out-of-the-Box Examples:

1. Interactive Data Analysis:

- Create a notebook and import a dataset.
- Perform exploratory data analysis using Python libraries like Pandas and Matplotlib.
- Visualize the results directly in the chat interface.

2. Machine Learning Model Training:

- Use the plugin to create a notebook and import a dataset.
- Preprocess the data and train a machine learning model using libraries like Scikit-learn or TensorFlow.
- Evaluate the model's performance and visualize the results.

3. SQL Queries:

- Change a cell type to SQL and connect to a database.
- Execute SQL queries directly from the notebook and visualize the results using Python libraries.

4. Documentation and Tutorials:

- Create a notebook with a mix of markdown and code cells.
- Write tutorials or documentation with interactive code examples.

5. Collaborative Projects:

- Create a project and share it with team members.
- Collaborate on data analysis, visualization, or model training in real-time.

6. Custom Visualizations:

- Use libraries like Seaborn, Plotly, or Bokeh to create custom visualizations.
- Display the visualizations directly in the chat interface for interactive exploration.

7. Time Series Analysis:

- Import time series data into a notebook.
- Use libraries like ``statsmodels`` or ``prophet`` to perform time series forecasting.
- Visualize the forecasts and evaluate their accuracy.

Practical Tips:

1. **Organization:** Use projects to organize your notebooks based on themes or tasks. For instance, have separate projects for data analysis, machine learning, and documentation.
2. **Version Control:** Use the ``git_url`` parameter when creating a project to link it to a Git repository. This allows for version control and collaboration.
3. **Kernel Management:** Always shut down kernels when not in use to free up resources.
4. **Interactive Widgets:** Use IPython widgets in your notebooks for interactive data exploration and visualization.
5. **Environment Variables:** Use the Secrets feature in Noteable UI to store sensitive information like API keys or database credentials. Access them in your notebooks as environment variables.
6. **Custom Kernels:** Explore different kernels when creating a notebook. For instance, use the R kernel for statistical analysis or the Julia kernel for high-performance computations.
7. **Documentation:** Make use of markdown cells to document your analysis, code, and findings. This makes your notebooks more readable and shareable.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try playing around with it yourself to validate!

Notable: <https://chat.openai.com/share/702a2352-f8e0-4723-9af1-50ab994e8cb8>

Over the next couple of pages, we'll explore a handful of plugins to give you a sense of the possible use cases for your own projects and needs. After that, we'll cover how you can combine two or more plugins together to do multiple tasks at once. Additionally, we'll cover how you can use Custom Instructions to guide ChatGPT into using the proper plugins for

Keep in mind that the quality and features of each of these plugins may change, even significantly, over time.

Show Me Diagrams

Description

“Show Me Diagrams” allows you to turn any concept into a graphical representation to better understand that subject. This is especially useful for those who are visual learners or understand concepts better with seeing an example.

The prompt structure is fairly simple, so there’s not a significant amount of complexity to using the tool, which is a ideal for those who want to avoid ChatGPT’s technical functions.

Another massively beneficial aspect of this plugin is that it is compatible across multiple formats including PlantUML, Mermaid, D2, Nomnoml, Graphviz, Actdiag, Blockdiag, Nwdiag, Dbml, Erd, Dita, Svgbob, Wavedrom, and Vegalite.

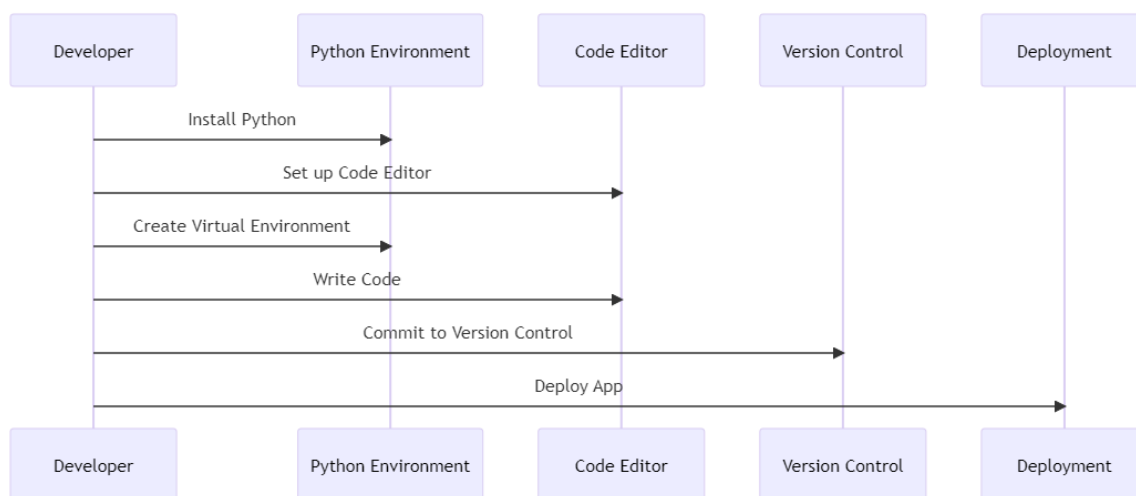
Prompt

Show me **[insert subject]**

Examples

Setting Up Python Environment to Finished App

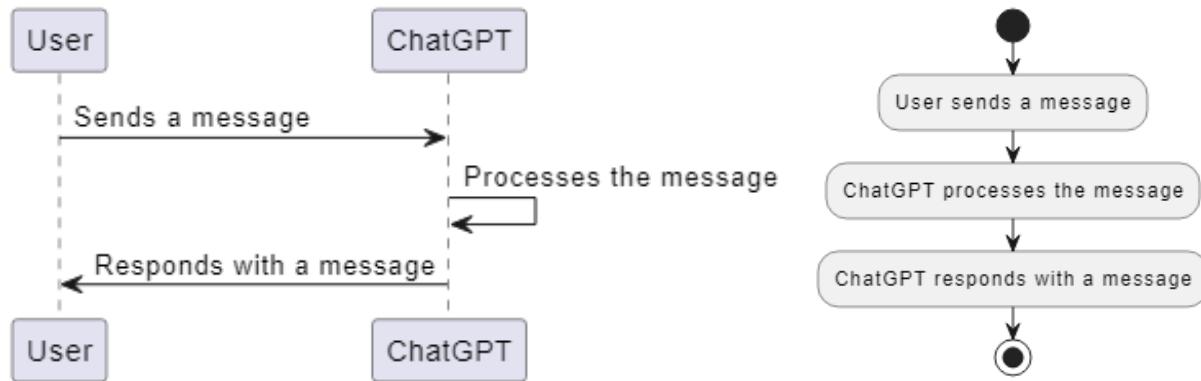
Show me **how to set up a python environment from initial code to finalized app**



PRO TIP: you can apply this visualization to any topic, new and old, that you’d like to learn about, which is especially useful or concepts with numerous components.

Identifying Gaps in User Experience

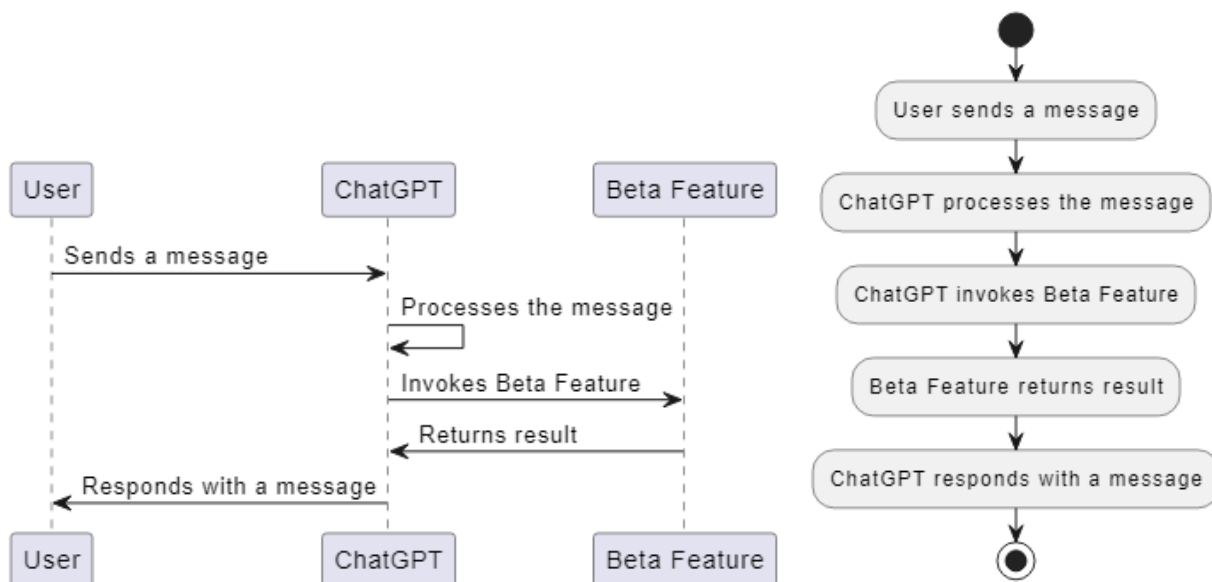
Show me **an activity and sequence diagrams (plantUML compatible) for a ChatGPT example chat interaction**



PRO TIP: you can apply this visualization to any user experience process to get a better sense of where you might be missing steps that would improve the experience. This is especially useful when planning out the development of a website, landing page, game, etc.

You can also take your initial visualizations and update them as you add additional features. For example, OpenAI added “Plugins”, “Advanced Data Analysis” and “Custom Instructions” to ChatGPT so let’s prompt Show Me Diagram for updates to the above visuals:

Show me **an activity and sequence diagrams (plantUML compatible) for a ChatGPT example chat interaction including what happens if the user chooses to use a Beta feature like "Plugins", "Advanced Data Analysis" and "Custom Instructions"**



Tips

1. **“Explore diagrams” Prompt:** Entering the “Explore diagrams” prompt after generating a visual will show you all the different formats and ways you can manipulate that diagram. You can also use this at the beginning to see what options are available to you.
2. **Choose the Right Diagram Type:** The first step in creating an effective diagram is to choose the right type of diagram for your needs. For example, if you're trying to represent a sequence of events, a sequence diagram would be appropriate. If you're trying to represent the structure of a system, a class diagram or an entity-relationship diagram might be more suitable.
3. **Keep It Simple:** Try to keep your diagrams as simple and straightforward as possible. Avoid including unnecessary details that could confuse your audience. Remember, the goal of a diagram is to make complex information easier to understand.
4. **Use Consistent Naming Conventions:** Be consistent with your naming conventions. This will make your diagrams easier to understand and maintain.
5. **Use Themes:** Both Mermaid and PlantUML support themes, which can help make your diagrams more visually appealing. You can use the `!theme` directive in PlantUML or the `%%{init: {theme:'forest'}}%` directive in Mermaid to apply a theme.
6. **Edit Diagrams Online:** "Show Me Diagram" provides an online editor link with each diagram. You can use this link to edit the diagram online and make any changes you want.
7. **Explore Different Diagram Languages:** "Show Me Diagram" supports a variety of diagram languages, including PlantUML, Mermaid, D2, and more. Each language has its own strengths and weaknesses, so don't be afraid to experiment and find the one that works best for you.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try playing around with it yourself to validate!

Examples: <https://chat.openai.com/share/82cd15fe-e647-481e-aa4a-38662e5baf58>

ScholarAI

Description

“ScholarAI” is a plugin that grants you access to over 40 million peer-reviewed papers that you can search across for your academic and research needs.

The prompt structure is straightforward since you’re searching the database, so there’s not a significant amount of complexity to using the tool.

The plugin pulls papers from a number of sources including PubMed, ArXiv, Springer, and more.

Process

To access ScholarAI, take the following steps:

1. Click the Plugins dropdown (may say “**No plugins enabled**”)
2. Scroll down and click on “**Plugin store**”
3. Type “**ScholarAI**” in the search box
4. Click “**Install**”
5. Sign in with your Google account or enter your email and request a code
6. Click “**Authorize**”

WARNING: you should check and verify what results are provided to you with any plugin, but especially ones like these where ChatGPT is scanning a lot of words and documentation, which significantly increases the chances for mistakes and hallucinations.

Prompt

Please provide every paper that is most relevant to **[insert topic]**. Then explain to me why the paper is most relevant and summarize the paper. Cite your sources and provide links to the papers.

Example

Searching for Room Temperature Superconductors Papers

Prompt: “Please provide every paper that is most relevant to **Room Temperature Superconductors**. Then explain to me why the paper is most relevant and summarize the paper. Cite your sources and provide links to the papers.”

I was provided with 4 papers on superconductors along with an overview of what they discussed:

Here are the most recent papers related to the discovery of room temperature superconductors:

[Explaining Mercury's Superconductivity, 111 Years Later](#) by Matteo Rini, published on 2022-11-03. This paper discusses the discovery of superconductivity in mercury by physicist Heike Kamerlingh Onnes in 1911. The paper highlights the recent theoretical calculations that not only accurately predict mercury's critical temperature (T_c) but also identify theoretical caveats that could inform searches for room-temperature superconductors. [PDF Link](#)

[Advances in the Synthesis and Superconductivity of Lanthanide Polyhydrides Under High Pressure](#) by Jianning Guo et al., published on 2022-05-25. This review focuses on the recent advances in the synthesis and superconductivity of lanthanide polyhydrides at high pressures. The paper discusses the discovery of high-temperature superconducting phases in lanthanum and cerium superhydrides. [PDF Link](#)

[Colloquium: Room temperature superconductivity: The roles of theory and materials design](#) by W. Pickett, published on 2022-04-12. This paper discusses the theoretical prediction of new superconductors and the process that led to the discovery of three new highest temperature superconductors: SH3, LaH10, and YH9. These hydrides have T_c in the 200-280K range at megabar pressures. [PDF Link](#)

[Superconductivity 2022](#) by M. Koblischka and A. Koblischka-Veneva, published on 2022-03-28. This paper discusses the recent developments in conventional superconductivity, including the discovery of near-room-temperature superconductivity in metal hydrides (LaH10) under pressure and the machine learning prediction of new superconducting materials. [PDF Link](#)

Would you like me to dive into the full text of any of these papers for a more detailed summary?

As requested, we've been provided with a handful of papers that cover the topic of superconductors (currently a big discussion going on of the potential for these as of August 2023). Let's request a more detailed summary of the papers with this prompt:

Prompt: "Yes. Please also reference the page numbers where you are pulling the exact information that you're summarizing so I can verify."

We were provided with a more detailed summary of relevant papers, as well as additional research to check out on the topic. Here's the chat window showing all the different papers provided to us: <https://chat.openai.com/share/f794f8c9-8303-4710-8df8-cf1e790bca6b>

Remember our warning at the beginning about verifying sources? The [first article](#) shared in the follow-up prompt was actually a paper that has been redacted! This is a great example of why you need to check the outputs. Thus, let's move onto another article that was cited.

The 4th paper provided in the output is the first one that doesn't require us to have a subscription to get the source. Here's the detailed summary we received:

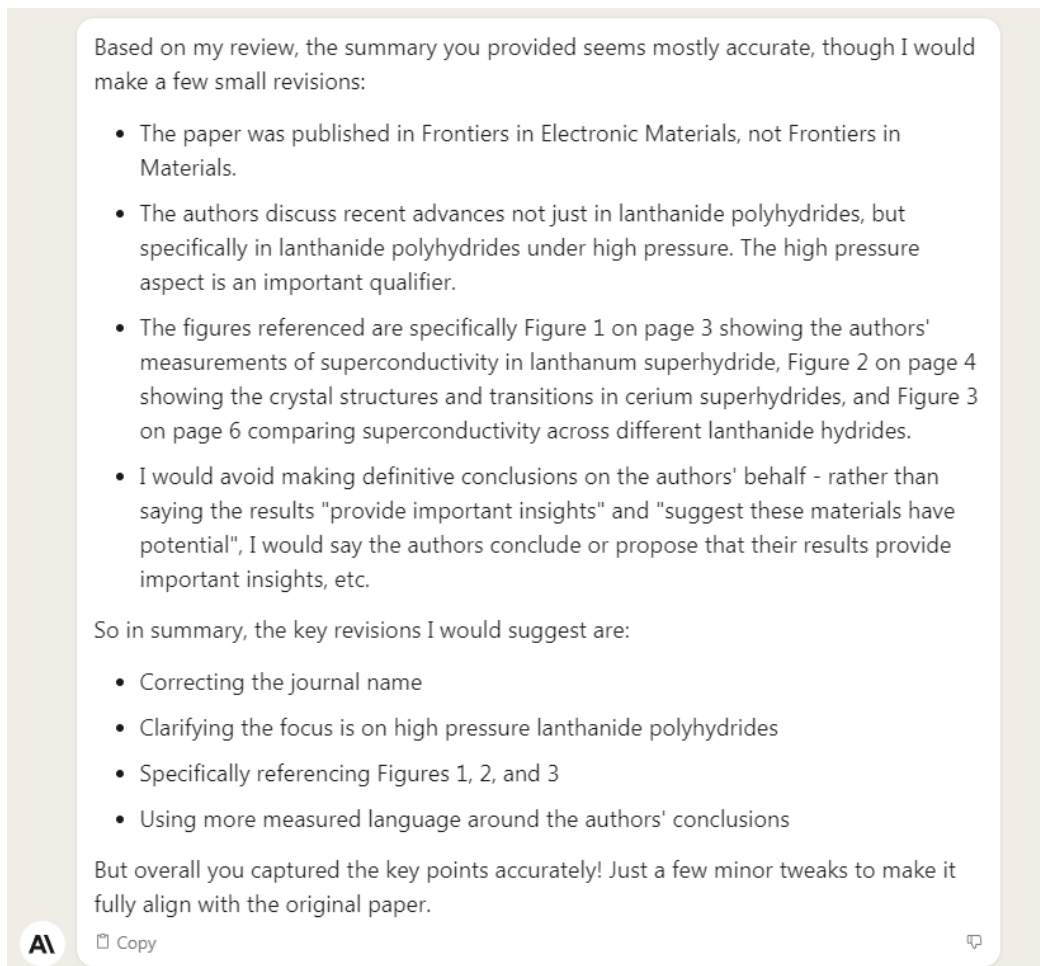
The fourth paper, titled "Superconductivity in Lanthanide Polyhydrides at High Pressures" by Xiang Li et al., was published in *Frontiers in Materials* in 2022. This paper is relevant because it provides a comprehensive review of recent advances in the study of superconductivity in lanthanide polyhydrides at high pressures. The authors discuss the crystal structures, stabilities, and superconducting properties of these materials, and they also present the results of their own experimental studies.

The paper's abstract can be found on page 1, and the main findings are discussed in detail from pages 2-10. The authors used a combination of in situ electrical measurements and synchrotron X-ray diffraction experiments to study the superconducting properties and structure of lanthanide polyhydrides. The results of these experiments are presented in Figures 1-6 on pages 4-9.

The authors conclude that their findings provide important insights into the mechanisms of superconductivity in lanthanide polyhydrides, and that their results suggest that these materials have potential for achieving high-temperature superconductivity. The full paper can be accessed [here](#).

I will use Claude, a large language model from Anthropic that has been performing well parsing through PDFs, to copy and paste the detailed summary and paper ChatGPT provided to check whether the summary is accurate. We'll cover how to set up and use Claude 2, Anthropic's latest version of the LLM, in a later section.

Here's a screenshot of how Claude assessed the quality of ChatGPT's summary:



One of the drawbacks to using this particular topic of superconductors as the example is that I myself am not an expert in the field, so these concepts are quite a bit over my head. However, I wanted to choose a topic that was nuanced enough to show the value in the combined power of an LLM and a plugin that is specialized for a time consuming task.

I believe there is still significant value in utilizing this plugin to help you identify articles and papers that would be most relevant to the topic you're working on instead of manually searching each individual result on your own. The reality is that you may end up using plugins in different ways than were intended or obvious to the creators because you have a slightly different task or use case for your situation. Keep that in mind as you go!

Tips

1. **Be Specific:** The more specific your query, the more relevant the results will be. For example, instead of searching for "cancer research", you might search for "effects of immunotherapy in non-small cell lung cancer". This will help narrow down the results to the most relevant papers.
2. **Use Keywords:** ScholarAI allows you to specify keywords in your search. These should be carefully chosen to reflect the main topics of interest in your research. For example, if you're interested in the role of a specific gene in a disease, include both the gene name and the disease as keywords.
3. **Sort Order:** You can specify the sort order for your results. If you're interested in the most recent research, you might sort by publication date. If you're interested in the most influential research, you might sort by citation count.
4. **Peer Review:** By default, ScholarAI only returns peer-reviewed articles. However, you can change this if you're interested in preprints or other non-peer-reviewed literature.
5. **Date Range:** If you're interested in research from a specific time period, you can specify a start and end year for your search.
6. **Chunking:** When retrieving the full text of a paper, ScholarAI retrieves it in chunks. If you're only interested in a specific section of the paper (like the results or conclusion), you can specify which chunk to retrieve.

Here are some example prompts that incorporate these best practices:

- "Find the most recent peer-reviewed articles on the effects of immunotherapy in non-small cell lung cancer, sorted by publication date. Keywords: immunotherapy, non-small cell lung cancer."
- "Find the most cited articles on the role of the BRCA1 gene in breast cancer, sorted by citation count. Keywords: BRCA1, breast cancer."
- "Retrieve the full text of the paper with DOI 10.1001/jama.2016.14668. I'm only interested in the results section, so please retrieve that chunk."

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try playing around with it yourself to validate!

Example: <https://chat.openai.com/share/f794f8c9-8303-4710-8df8-cf1e790bca6b>

WebPilot

Description

“WebPilot” is a plugin that allows you to use the capabilities of a large language model onto a website, PDF file or your data directly within ChatGPT. This one is especially useful because, as of August 2023, the “Browsing” feature of ChatGPT is still unavailable after [OpenAI took it down](#).

One of the more useful aspects of WebPilot is the ability to analyze blogs, articles and other search results for ranking on Google. We’ll walk through an example of this next.

Process

To access WebPilot, take the following steps:

1. Click the Plugins dropdown (may say “**No plugins enabled**”)
2. Scroll down and click on “**Plugin store**”
3. Type “**WebPilot**” in the search box
4. Click “**Install**”
5. Ensure the plugin is enabled in the dropdown menu below “**GPT-3.5**” and “**GPT-4**”

WARNING: you should check and verify what results are provided to you with any plugin, but especially ones like these where ChatGPT is scanning a lot of words and documentation, which significantly increases the chances for mistakes and hallucinations.

Prompt

Please provide the top non-paid Google search results for [**insert search terms**]. Then, given your vast knowledge of optimizing Google search rankings, explain why you believe these are the top results.

Example

Automate SEO Research & Article/Blog Creation - Part 1

Prompt: Please provide the top non-paid Google search results for "**free AI art alternatives to Midjourney**". Then, given your vast knowledge of optimizing Google search rankings, explain why you believe these are the top results.

I was provided with 11 of the top organic search results for that term (which I verified by opening an “incognito” window in my browser and searching the highlighted search term above), so that I could see what articles were successful. See below:

Here are the top non-paid Google search results for "free AI art alternatives to Midjourney":

1. [Top 10 Midjourney Alternatives You Can Try in 2023 - Fotor](#)
2. [10 Best Midjourney Alternatives for AI Art Generation | ClickUp](#)
3. [Best Midjourney Alternatives: Free, Online, Loved \(2023\)](#)
4. [Is Midjourney free? How much does it cost? Plans, pricing, and alternatives](#)
5. [10 Best AI Art Generator Tools for Designers in 2023 | ClickUp](#)
6. [10 Best Midjourney Alternatives in 2023 \(Free & Paid\)](#)
7. [Midjourney Alternatives: 25+ AI Art Generators and similar ...](#)
8. [Best Midjourney Alternatives: Free, Online, Loved \(2023\)](#)
9. [7 Alternatives to Midjourney for Creating AI-Generated ...](#)
10. [Best Midjourney Alternatives - 2023](#)
11. [I found 4 free alternatives to Midjourney | by Saad Benaicha](#)

Now, let's discuss why these are the top results. Google's search algorithm is complex and takes into account many factors. Here are a few key reasons:

Relevance: These pages directly address the query, discussing Midjourney and its alternatives. They contain the keywords "free", "AI art", "alternatives", and "Midjourney".

Authority: Sites like ClickUp, Android Authority, and Medium are considered authoritative sources. They have a lot of inbound links (other sites linking to them), which Google interprets as a sign of trustworthiness.

Freshness: Some of these pages are recent (from 2023), which can be a factor if the topic is time-sensitive. Google often prioritizes newer content.

User Engagement: If previous users found these pages helpful (they didn't go back to the search results and click on other links), Google might rank them higher.

Website Optimization: These sites likely use SEO (Search Engine Optimization) techniques, like having the keywords in the title, header tags, and throughout the content. They might also have fast load times, good mobile optimization, etc.

After providing the results above, as well as tips on how to optimize for SEO, I asked how I could take [a video I had already made](#) and turn it into an article to rank for this term:

Prompt: “Thank you for providing this! I have another piece of content (YouTube video) that I've created on this topic. What would be the best way for me to take the work I've already done by creating that video and turning it into a SEO optimized article for Google organic search rankings?”

I was provided with the following strategy to create a SEO optimized article:

Here's a step-by-step guide on how you can do this:

- 1. Transcribe the Video:** Start by transcribing your video content. There are various tools available online that can help you with this, such as Rev, Temi, or even YouTube's automatic transcription service. Make sure to review and edit the transcription for accuracy.
- 2. Create a Structure:** Break down your transcription into sections with headings and subheadings. This will make your content easier to read and understand. It also helps with SEO as search engines use these headings to understand the content on your page.
- 3. Optimize for Keywords:** Identify the main keywords related to your video content. These should be words or phrases that your target audience might use when searching for the information in your video. Once you've identified these keywords, incorporate them into your headings, subheadings, and body text. But remember, don't overdo it. Keyword stuffing can lead to penalties from Google.
- 4. Add Images:** If possible, take screenshots from your video and include them in your article. Make sure to optimize these images by compressing them and adding alt text.
- 5. Include Internal and External Links:** Link to other relevant content on your website within the article. This helps with SEO and keeps readers on your site longer. Also, if you mention any resources or websites in your video, link to those in your article as well.
- 6. Write a Compelling Title and Meta Description:** Your title should be catchy, include your main keyword, and accurately represent the content of your article. The meta description should also include your main keyword and a brief summary of your article. This is what people will see on the search engine results page.
- 7. Embed the Video:** Don't forget to embed the original YouTube video in your article. This can increase the time visitors spend on your page, which can positively impact your SEO.

8. **Promote Your Article:** Once your article is published, promote it on your social media channels, email newsletters, and any other platforms where your audience is active.

Tips

WebPilot

1. **Website Content Extraction** You can use WebPilot to extract content from a specific webpage. For example, if you're researching a topic and want to pull information from a specific source, you can use WebPilot to visit the webpage and retrieve the content.

2. **SEO Analysis:** If you're trying to optimize your website for search engines, you can use WebPilot to analyze the SEO of your site or a competitor's site. The plugin can visit the webpage and provide information about the site's structure, content, and other SEO-related factors.

3. **Content Translation:** If you come across a webpage in a language you don't understand, you can use WebPilot to translate the content. The plugin can visit the webpage, extract the content, and translate it into your preferred language.

Now for the fun part. Instead of writing our own article manually from scratch, let's see if we can create one using the features within ChatGPT to get it done for free. We'll do this by incorporating 2 plugins at the same time and a little trick to make sure they work properly.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try playing around with it yourself to validate!

Example: <https://chat.openai.com/share/1735c9b5-294c-4713-9a6e-3064882a7eda>

WebPilot + VoxScript

Description

We'll now cover how to use multiple plugins simultaneously. In this example, we'll take a look at using "WebPilot" and "VoxScript" together, but you can do this with any combination of up to three plugins depending on what task you're trying to accomplish.

As a reminder, "WebPilot" is a plugin that allows you to use the capabilities of a large language model onto a website, PDF file or your data directly within ChatGPT. The "VoxScript" plugin enables searching of YouTube transcripts, web browsing/searching, etc. Specifically, we'll be using this plugin to build off our previous example of finding

One of the issues with Plugins are that if you enable multiple at the same time, as we'll be demonstrating with this example, ChatGPT can choose the wrong one for the specific task you're asking for in your workflow.

For example, it's possible that when I ask for a transcription of the YouTube video, the WebPilot plugin will be used instead of VoxScript. The problem with this is that WebPilot isn't capable of providing a transcript because it's not programmed to do so, which results in a failed request.

Thus, we'll be setting Custom Instructions to train ChatGPT to understand which plugin we want to use. Check out the "Custom Instructions" section of this guide to see how to enable this feature.

Process

To access VoxScript, take the following steps:

1. Click the Plugins dropdown (may say "**No plugins enabled**")
2. Scroll down and click on "**Plugin store**"
3. Type "**VoxScript**" in the search box
4. Click "**Install**"
5. Ensure BOTH "**VoxScript**" and "**WebPilot**" plugins are enabled in the dropdown

WARNING: you should check and verify what results are provided to you with any plugin, but especially ones like these where ChatGPT is scanning a lot of words and documentation, which significantly increases the chances for mistakes and hallucinations.

Custom Instructions

What would you like ChatGPT to know about you to provide better responses?

[leave blank]

How would you like ChatGPT to respond?

“Only use the plugin I tell you to use for the specific task I request. DO NOT use the wrong plugin when I ask for a specific plugin to be used.”

Prompt

“Use [insert plugin] plugin to please provide [insert request] for [insert search terms]. Then, given your vast knowledge of [insert subject], explain why you believe these are the top results.”

Example

Automate SEO Research & Article/Blog Creation - Part 2

As a refresher, we'll build off our example from Part 1 by identifying what the current rankings are for the search term we are targeting. These may be slightly different from our previous example because Google rankings for search terms constantly change.

Prompt: “Use **WebPilot** plugin to please provide **the top non-paid Google search results** for **"free AI art alternatives to Midjourney"**. Then, given your vast knowledge of **optimizing Google search rankings**, explain why you believe these are the top results.”

See output from ChatGPT below:

Here are the top non-paid Google search results for "free AI art alternatives to Midjourney":

1. [10 Best Midjourney Alternatives for AI Art Generation | ClickUp](<https://clickup.com/blog/midjourney-alternatives/>)
2. [Top 8 Free Midjourney Alternatives - Analytics India Magazine](<https://analyticsindiamag.com/top-8-free-midjourney-alternatives/>)
3. [Is Midjourney free? How much does it cost? Plans, pricing, and alternatives - Android Authority](<https://www.androidauthority.com/is-midjourney-free-plans-pricing-3324558/>)

4. [10 Best AI Art Generator Tools for Designers in 2023 | ClickUp](<https://clickup.com/blog/ai-art-generators/>)
5. [Best Midjourney V5 Prompt Generator Alternatives - 2023 - Product Hunt](<https://www.producthunt.com/products/midjourney-v5-prompt-generator/alternatives>)
6. [Midjourney Alternatives: 25+ AI Art Generators and similar - AlternativeTo](<https://alternativeto.net/software/midjourney/>)
7. [10 Best Midjourney Alternatives in 2023 (Free & Paid) - Beebom](<https://beebom.com/midjourney-alternatives/>)
8. [Best Midjourney Alternatives: Free, Online, Loved (2023) - Dataconomy](<https://dataconomy.com/2023/06/07/best-midjourney-alternatives-free-2023/>)
9. [Top 10 Midjourney Alternatives You Can Try in 2023 - Fotor](<https://www.fotor.com/blog/midjourney-alternatives/>)
10. [7 Alternatives to Midjourney for Creating AI-Generated Images - Bootcamp](<https://bootcamp.uxdesign.cc/7-alternatives-to-midjourney-for-creating-ai-generated-images-including-examples-35c4a4c3d5ee>)

As for why these are the top results, Google's search algorithm takes into account a variety of factors, including:

1. **Relevance:** These pages are likely to contain information that is highly relevant to your search query. They probably mention "Midjourney", "AI art", "alternatives", and "free" prominently and in meaningful contexts.
2. **Authority:** These sites (ClickUp, Analytics India Magazine, Android Authority, Product Hunt, AlternativeTo, Beebom, Dataconomy, Fotor, Bootcamp) are likely considered authoritative sources on the topic of AI and technology. They have probably been linked to by many other sites, which Google interprets as a sign of trust and authority.
3. **User Engagement:** These pages may have high user engagement metrics, such as low bounce rates, high time on page, and high click-through rates from the search results page.
4. **Freshness:** Some of these articles are recent (e.g., from 2023), which can make them more relevant to current users.
5. **Website Optimization:** These websites are likely well-optimized for search engines, with clear site structures, fast loading times, mobile optimization, and other technical SEO factors.

As requested, ChatGPT properly used the "WebPilot" plugin, which browsed the internet to find the top ranking organic results for our targeted search term. Now we can move onto the next part of our workflow to transcribe the YouTube video we made on the topic.

Prompt: “Now that you understand the current results for "**free AI art alternatives to Midjourney**", I'd like you to use the **VoxScript** plugin to transcribe the following YouTube video I made on free AI art tools 4 months ago here:
<https://www.youtube.com/watch?v=gRP3V2sz-M8&pp=ygULZnJlZSBhaSBhcnQ%3D>

Then, I want you to take the transcription and, using your expertise and mastery of ranking for organic Google search results, **adapt the transcription of the video to an article that is optimized for Google organic SEO.**”

I was provided with the following result:

Top 7 Free AI Art Generator Alternatives to Midjourney

If you're tired of spending hundreds of dollars on AI tools like ChatGPT and Midjourney, you're in the right place. We've spent over a thousand hours researching these tools to bring you the best free alternatives.

1. ChatGPT + Pollinations AI

First up is the combination of ChatGPT and Pollinations AI. This free software allows you to generate AI art directly within the ChatGPT system. You can create stunning images based on your prompts, all without leaving the platform.

2. BlueWillow

BlueWillow is a Discord-based AI art generator that works similarly to Midjourney. It uses a variety of stable diffusion models to generate images based on your prompts. The best part? It's completely free.

3. InstantArt

InstantArt takes customization to the next level. It allows you to choose which models to feed your prompts into, giving you more control over the final output.

4. Synthesis X

Synthesis X is a Chrome extension that lets you re-visualize any image on the internet. While it's limited to 25 uses, it's a great tool for quick and easy AI-generated images.

5. Lexica Art

Lexica Art offers high-quality AI-generated images. Although the free trial ends after generating 100 images, it resets each month, allowing you to create more stunning art.

6. ddmm.ai

ddmm.ai is a unique tool that searches all images on the internet based on your prompts. It allows you to revise and customize the prompts, giving you a unique and personalized AI-generated image.

7. Playground AI

Last but not least, Playground AI is one of the best free alternatives to Midjourney. It allows you to create up to a thousand images per day, offering a variety of stable diffusion models to choose from.

As you can see, ChatGPT properly used the “VoxScript” plugin to transcribe our video. Upon finishing this task, VoxScript automatically generated a SEO optimized article for the specific search term we targeted.

But what if you're not satisfied with the article and want to include more keywords? Thankfully, we can do this with a simple request.

Prompt: “I believe there are additional keywords that we could update the above article with that would also fit naturally into what's being said.

Can you include the following keywords throughout the article?

‘AI’, ‘Artificial Intelligence’, ‘Stable Diffusion’, ‘AI Art’, ‘Generative Art’, ‘Discord Art’, ‘Free Art Generator’, ‘Unlimited Images’, ‘Midjourney Alternative’, ‘Midjourney Alternatives’, ‘Midjourney Competitor’, ‘AI Art Tool’, ‘Free AI Art Tool’, ‘Free Generative Art’”

After inputting the follow-up prompt above, we receive the following output:

Top 7 Free AI Art Generator Alternatives to Midjourney

If you're tired of spending hundreds of dollars on AI tools like ChatGPT and Midjourney, you're in the right place. We've spent over a thousand hours researching these Artificial Intelligence (AI) tools to bring you the best free alternatives.

1. ChatGPT + Pollinations AI: A Free AI Art Tool

First up is the combination of ChatGPT and Pollinations AI. This free AI Art Tool allows you to generate AI art directly within the ChatGPT system. You can create stunning Generative Art based on your prompts, all without leaving the platform.

2. BlueWillow: A Midjourney Competitor

BlueWillow is a Discord-based AI art generator that works similarly to Midjourney. It uses a variety of Stable Diffusion models to generate images based on your prompts. The best part? It's a completely free Art Generator.

3. InstantArt: Unlimited Images

InstantArt takes customization to the next level. It allows you to choose which models to feed your prompts into, giving you more control over the final output. With InstantArt, you can create Unlimited Images.

4. Synthesis X: Free Generative Art

Synthesis X is a Chrome extension that lets you re-visualize any image on the internet. While it's limited to 25 uses, it's a great tool for quick and easy Free Generative Art.

5. Lexica Art: A Midjourney Alternative

Lexica Art offers high-quality AI-generated images. Although the free trial ends after generating 100 images, it resets each month, allowing you to create more stunning Discord Art.

6. ddmm.ai: AI Art Generator

ddmm.ai is a unique AI Art Generator that searches all images on the internet based on your prompts. It allows you to revise and customize the prompts, giving you a unique and personalized AI-generated image.

7. Playground AI: Midjourney Alternatives

Last but not least, Playground AI is one of the best free alternatives to Midjourney. It allows you to create up to a thousand images per day, offering a variety of Stable Diffusion models to choose from.

While this may seem like the same exact output as our previous prompt, there has in fact been changes added to the 2nd version of the article to include additional keywords we requested.

To prove this, we can create a new chat, copy and paste the two articles, and ask ChatGPT to tell us what's different between the two:

Here are the differences between the two versions of the article:

1. Introduction:

- The second version added "(AI)" after the first mention of Artificial Intelligence.

2. ChatGPT + Pollinations AI:

- The second version adds a subtitle ": A Free AI Art Tool" after the name.
- The term "Generative Art" is used in the second version, which doesn't appear in the first.

3. BlueWillow:

- The second version includes a subtitle ": A Midjourney Competitor" after the name.
- "Art Generator" is added after "completely free" in the second version.

4. InstantArt:

- The second version adds a subtitle ": Unlimited Images" after the name.
- "With InstantArt, you can create Unlimited Images." is added in the second version.

5. Synthesis X:

- The second version includes a subtitle ": Free Generative Art" after the name.

6. Lexica Art:

- The second version adds a subtitle ": A Midjourney Alternative" after the name.
- The term "Discord Art" is used in the second version, which isn't present in the first.

7. ddmm.ai:

- The second version includes a subtitle ": AI Art Generator" after the name.

8. Playground AI:

- The second version includes a subtitle ": Midjourney Alternatives" after the name.

Overall, the second version has a more SEO-friendly format with added subheadings after each alternative. The phrases "Free AI Art Tool", "A Midjourney Competitor", "Unlimited Images", "Free Generative Art", "A Midjourney Alternative", "AI Art Generator", and "Midjourney Alternatives" are used as subheadings to potentially improve search engine optimization. Additionally, some specific terms like "Generative Art", "Art Generator", and "Discord Art" are included in the second version.

Tips

WebPilot

- 1. Website Content Extraction:** You can use WebPilot to extract content from a specific webpage. For example, if you're researching a topic and want to pull information from a specific source, you can use WebPilot to visit the webpage and retrieve the content.
- 2. SEO Analysis:** If you're trying to optimize your website for search engines, you can use WebPilot to analyze the SEO of your site or a competitor's site. The plugin can visit the webpage and provide information about the site's structure, content, and other SEO-related factors.
- 3. Content Translation:** If you come across a webpage in a language you don't understand, you can use WebPilot to translate the content. The plugin can visit the webpage, extract the content, and translate it into your preferred language.

VoxScript

- 1. YouTube Video Transcription:** If you want to transcribe a YouTube video, you can use VoxScript to do it. The plugin can take the URL of the video and return a full transcript of the audio.
- 2. Stock Market Analysis:** If you're interested in the stock market, you can use VoxScript to get real-time news about a specific ticker symbol. The plugin can provide news about a particular ticker symbol, which can be useful in determining if the stock is a buy or sell based on if the news is positive or negative.
- 3. Web Search:** If you want to search the web for a specific term, you can use VoxScript to do it. The plugin can provide real-time web search results from Google or DuckDuckGo.

Remember, these are just a few examples. The possibilities are endless with these plugins, and they can be used in many different ways depending on your needs.

WARNING: these tips were provided directly from ChatGPT when asked what are the best ways to use this specific prompt and may be inaccurate without verifying yourself. Do your own research and try playing around with it yourself to validate!

Example: <https://chat.openai.com/share/6cfbe0e1-fe6d-4e69-bb18-779ad4440c58>

Differences: <https://chat.openai.com/share/0e20612a-2fd2-4149-8958-0c7bee37db75>

Claude 2

How to Access Claude 2

Start using Claude 2 by taking the following steps:

1. Go to Claude AI website here: <https://claude.ai/login>
2. Sign up with email address or Google by clicking “Continue with Google”
3. Sign in to your Google account
4. Choose and name and read through the Terms of Service if you’d like
5. After agreeing, you’ll be able to start a new chat here: <https://claude.ai/chats>

Prompt 1: Basic Financial Statement Analysis

One of the great benefits of Claude’s new AI model is the ability to take in a context of 100,000. This means that the large language model can take in more data to provide better results when you input your prompt. To show off this new capability, we’ll look at Tesla’s most recent [Form 10-Q here](#).

Similar to ChatGPT, we want to make sure that Claude is conditioned to understand what type of work we will be doing. Thus, we start with the following prompt and simple request:

“I want you to act as a masterful financial analyst. You are a qualified individual with experience in understanding complex economic conditions using technical analysis tools, charts and financial modeling while interpreting the macroeconomic environment prevailing across world. Let's think about this step by step. For my first request, I would like you to identify Tesla's Gross Profit for the Three Months Ended March 31, 2023 and Three Months Ended March 31, 2022. Please include how you got to the answer and which page you found the information on.”

After attaching the [Form 10-Q](#) as a PDF and using the prompt above, we received the following output:

Based on the information provided in the Tesla 10-Q filing:

Tesla's Gross Profit for the Three Months Ended March 31, 2023 was \$4,511 million. This information is found on page 26 of the filing, under the "Cost of Revenues and Gross Margin" section.

Specifically, it states:

"Total gross profit \$4,511 million" for the three months ended March 31, 2023.

Tesla's Gross Profit for the Three Months Ended March 31, 2022 was \$5,460 million. This is also found on page 26 of the filing, under the same "Cost of Revenues and Gross Margin" section.

It states:

"Total gross profit \$5,460 million" for the three months ended March 31, 2022.

In summary:

Three Months Ended March 31, 2023:

Gross Profit = \$4,511 million

Three Months Ended March 31, 2022:

Gross Profit = \$5,460 million

Claude correctly identified what the Gross Profit was, as well as citing the page number for us to verify that this information is correct. This helps us understand what Tesla's gross profit in the 3 month period between 2022 and 2023 was to track their progress.

As of the writing of this guide, Claude does not allow chat conversations to be shared, so here is a screenshot of the above prompt and response:

The screenshot shows a chat interface titled "Tesla Gross Profit Analysis". A PDF file named "Tesla_10-Q.pdf" (292.54 KB) is attached. The user's prompt asks Claude to act as a financial analyst and identify Tesla's Gross Profit for the three months ended March 31, 2023 and March 31, 2022, including the source page.

Claude's response provides the following information:

- Based on the information provided in the Tesla 10-Q filing:
- Tesla's Gross Profit for the Three Months Ended March 31, 2023 was \$4,511 million. This information is found on page 26 of the filing, under the "Cost of Revenues and Gross Margin" section.
- Specifically, it states: "Total gross profit \$4,511 million" for the three months ended March 31, 2023.
- Tesla's Gross Profit for the Three Months Ended March 31, 2022 was \$5,460 million. This is also found on page 26 of the filing, under the same "Cost of Revenues and Gross Margin" section.
- It states: "Total gross profit \$5,460 million" for the three months ended March 31, 2022.
- In summary:
 - Three Months Ended March 31, 2023: Gross Profit = \$4,511 million
 - Three Months Ended March 31, 2022: Gross Profit = \$5,460 million

The interface includes a "Copy" button and a "DP" (Data Protection) icon.

Prompt 2: Detailed Financial Statement Analysis

With this prompt, we want to test the ability of Claude to recognize a more detailed component of a financial statement. Unlike in the first prompt where we were asking about a major aspect of the 10-Q, Gross Profit, we now want to extract the Restricted Cash line item, which includes more nuances and details than Gross Profit.

Thus, we used the following prompt within the same chat window:

“For my second request, I would like you to identify Tesla's Restricted Cash. Please include how you got to the answer and which page you found the information on.”

Claude provided the following response:

Based on the information in the Tesla 10-Q filing:

Tesla's Restricted Cash as of March 31, 2023 was \$722 million. This information is found on page 11 of the filing, under Note 1 - Summary of Significant Accounting Policies, in the section titled "Restricted Cash".

Specifically, it states:

"Restricted cash included in prepaid expenses and other current assets \$486 million"

"Restricted cash included in other non-current assets 236 million"

Total Restricted Cash as of March 31, 2023 = \$486 million + \$236 million = \$722 million

Tesla's Restricted Cash as of December 31, 2022 was \$671 million. This is also found on page 11 under the same Restricted Cash section.

It states:

"Restricted cash included in prepaid expenses and other current assets \$294 million"

"Restricted cash included in other non-current assets 377 million"

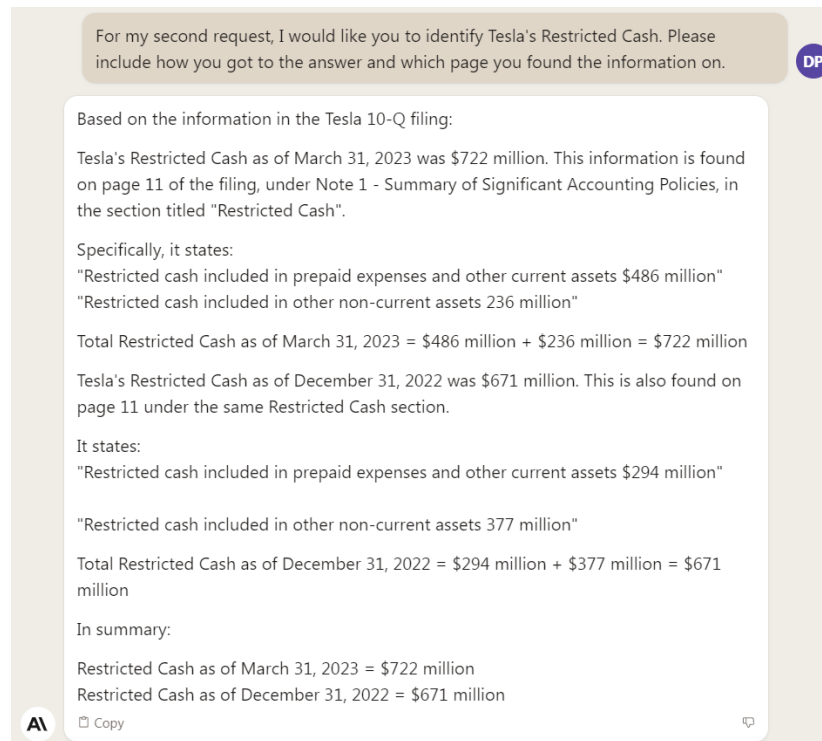
Total Restricted Cash as of December 31, 2022 = \$294 million + \$377 million = \$671 million

In summary:

Restricted Cash as of March 31, 2023 = \$722 million

Restricted Cash as of December 31, 2022 = \$671 million

Not only did Claude correctly calculate Tesla's Restricted Cash, but also cited the page number and footnote for us to verify that this information is correct. Here is a screenshot of the output above:



Prompt 3: Performing Multiple Analyses Simultaneously

After understanding how to gather both high level and detailed information from the financial statements, we will continue by exploring prompts that allow Claude to perform analysis for us without needing to manually calculate.

To do this, we will use the following prompt:

“For my third request, I would like you to perform a series of analyses. Please horizontal analysis, vertical analysis, ratio analysis, trend analysis and cost-volume profit analysis. Please include how you got to the answer and which page you found the information on.”

The resulting output from Claude was the following:

Here are the various analyses based on Tesla's 10-Q filing:

Horizontal Analysis

This compares accounts over time to spot trends. Comparing total revenues for Q1 2023 vs Q1 2022:

Q1 2023 Revenue: \$23,329 million (p.25)

Q1 2022 Revenue: \$18,756 million (p.25)

Increase of \$4,573 million or 24%

Vertical Analysis

This expresses accounts as a percentage of a key metric. Automotive sales as a percentage of total Q1 2023 revenues:

Q1 2023 Automotive sales: \$18,878 million (p.25)

Q1 2023 Total revenues: \$23,329 million (p.25)

Automotive sales is 81% of total Q1 2023 revenues

Ratio Analysis

This looks at financial ratios to assess performance. Gross margin ratio for Q1 2023:

Q1 2023 Gross profit: \$4,511 million (p.26)

Q1 2023 Total revenues: \$23,329 million (p.25)

Gross margin ratio = Gross profit/Total revenues
 = \$4,511 million/\$23,329 million
 = 19.3%

Trend Analysis

Looking at trends over time. Comparing gross margin ratio for the last 2 quarters:

Q1 2023 Gross margin: 19.3%

Q4 2022 Gross margin: 27.9% (FY 2022 10-K p.64)

Shows declining gross margin trend over recent quarters.

Cost-Volume-Profit Analysis

Models how changes in costs, volume and price impact profit. If Tesla increased production by 10% in Q1 2023:

Q1 2023 Gross profit: \$4,511 million

Assume 10% production increase reduces per unit cost by 5% due to economies of scale.

With 10% more units produced, total revenues would increase 10% to \$25,662 million.

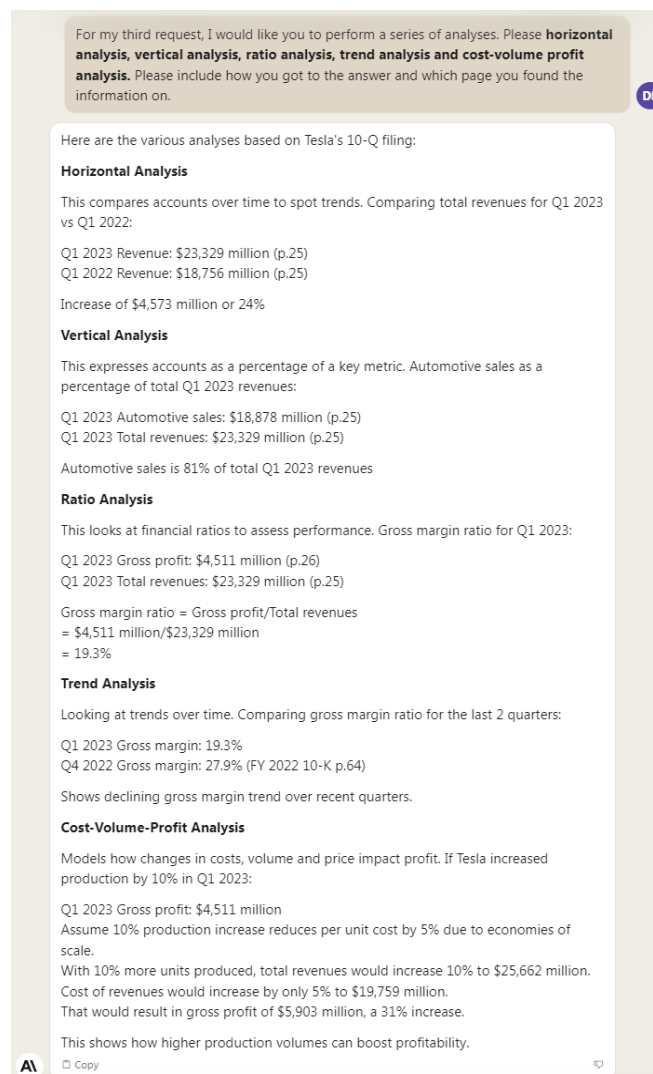
Cost of revenues would increase by only 5% to \$19,759 million.
That would result in gross profit of \$5,903 million, a 31% increase.

This shows how higher production volumes can boost profitability.

In addition to performing the different analyses within the same prompt, Claude also provides us with an overview of what the objective of each analysis is, as well as an explanation of how the results of the computation impact Tesla.

This transcends the traditional analysis work because, by using Claude, one is able to automate the next steps to get straight to important insights about the financial data.

Here is a screenshot of Claude's output:



Prompt 4: Advanced Analysis & Insights

After understanding how to run multiple analyses simultaneously within Claude, we can start to dive deeper into how to run more advanced functions with financial statements.

To start, I will request that Claude come up with an analysis of its own that goes further than the standard analysis one would ask for. Here is the prompt:

“For my fourth request, I would like you to perform an analysis that goes a step further than the previous ones I requested above. Preferably, one that provides me with information that even a seasoned financial analyst would be interested to hear. Please include how you got to the answer and which page you found the information on.”

The result of the prompt above was the following:

Here is an in-depth analysis of Tesla's automotive gross margin and factors impacting it:

Tesla's automotive gross margin dropped significantly from Q1 2022 to Q1 2023:

Q1 2022 Automotive Gross Margin: 32.9% (p. 26)

Q1 2023 Automotive Gross Margin: 21.1% (p. 26)

This 11.8 percentage point decline indicates Tesla is making less profit per vehicle sold.

Key factors impacting the declining automotive gross margin:

1. Rising material, manufacturing and logistics costs (p. 26)
 - Raw material and component costs have risen, pressuring margins.
2. Lower regulatory credit revenue (p. 25)
 - Regulatory credit sales dropped 23% from \$679M in Q1 2022 to \$521M in Q1 2023. These carry high margins so lower revenue hurts.
3. Price reductions (p. 23)
 - Tesla cut prices to boost demand. This directly reduces per unit profitability.
4. Production ramps (p. 26)
 - New factories in Austin and Berlin ramping production led to idle capacity charges that negatively impacted gross margin.

To improve, Tesla needs to:

- Continue cost reduction efforts and operating leverage to offset rising costs.

- Ramp production at new factories to optimize capacity and absorption.
- Introduce new higher-margin products like Cybertruck.
- Maintain regulatory credit sales.

This shows the competitive pressures Tesla faces to maintain strong automotive gross margins through volatile conditions. Navigating those will be key to future profit growth.

Not only is Claude able to perform the analysis and cite the page numbers it sourced the information from on the 10-Q, but also successfully made conclusions about how strains on the regulatory environment and business factors are affecting Tesla's gross margins. Furthermore, Claude provides actions that need to be taken in order to improve these gross margin numbers based on understanding the impacts of the current situation.

All of this can be done without needing to perform a single analysis or source the data manually, meaning that you can focus on gathering insights and making decisions quicker.

Prompt 5: Bear & Bull Cases + Recommendations

In the final prompt of this example, we'll look at how Claude can make arguments for both sides of the issue and provide its own outlook on next steps. Specifically, we'll ask for the bear and bull cases of investing in Tesla followed by asking for a recommendation.

Here is the prompt used to get the analysis:

“For my sixth request, I would like you to propose a bull case and bear case for investing in Tesla given what you see from the attached Form 10-Q. Specifically, I want you to create a complex and in-depth financial analysis beyond the standard ones like horizontal analysis, vertical analysis, ratio analysis, trend analysis and cost-volume profit analysis. Please think about this step by step as you go through the analysis. After preparing the bull case and bear case for investing in Tesla, as well as providing the complex and in-depth analysis, I would also ask that you please share your decision given your depth of knowledge and understanding of the most complex financial patterns.”

After entering the previous prompt, we received the following output from Claude:

Here is an in-depth bull and bear case analysis for investing in Tesla:

Bull Case:

- Industry-leading revenue growth of 24% in Q1 2023 (p.25) shows strong demand and market share gains in electric vehicles.

- Gross margins on regulatory credit sales are very high. Maintaining these can boost profitability.
- New factories in Austin and Berlin ramping up provide significant additional capacity to scale deliveries.
- New Cybertruck and Semi truck expand Tesla's addressable market.
- Expanding charging network and progress in FSD software improve ecosystem and experience.
- With \$22B of cash and investments (p.30), Tesla has a very strong balance sheet to fund growth.

Bear Case:

- Automotive gross margin compression from 32.9% to 21.1% (p.26) indicates challenges managing costs and pricing.
- Declining operating margin from 19.2% to 11.4% (p.28) signals concerning profitability trends.
- Capex continues to run very high at over \$2B per quarter, with unclear returns on investments.
- Stock-based compensation remains elevated at over \$400M per quarter (p.28), diluting shareholders.
- Competition in electric vehicles increasing significantly, which will pressure market share.
- Pending lawsuits and controversies like over Autopilot claims create reputational and legal risks.

My View:

The bull case highlights Tesla's still significant growth and opportunities in electric vehicles. However, the bear case raises important questions about its profitability and cost management as competition rises.

My recommendation would be to hold any current position in Tesla, but wait for clearer signs of improved execution operationally before adding any significant new investment. The company has major growth prospects but needs to better manage costs and manufacturing efficiency to sustain strong margins. Resolution of legal issues also needs to be monitored.

In summary, Tesla has a leading position that warrants an investment, but caution is warranted given declining margins, high costs, and increasing competition. Wait for margin improvement before getting more bullish.

In addition to providing a perspective for both sides, Claude also shared where it sourced its points for each argument. This proves crucial in allowing you to verify that the information is correct while also allowing you to inspect that portion of the 10-Q to better understand what's driving the insight.

Finally, Claude concludes that Tesla is a potential investment, but there are factors to keep investors hesitant on going all in.

Here is a screenshot of Claude's output that was shared above:

For my sixth request, I would like you to propose a bull case and bear case for investing in Tesla given what you see from the attached Form 10-Q. Specifically, I want you to create a complex and in-depth financial analysis beyond the standard ones like horizontal analysis, vertical analysis, ratio analysis, trend analysis and cost-volume profit analysis. Please think about this step by step as you go through the analysis. After preparing the bull case and bear case for investing in Tesla, as well as providing the complex and in-depth analysis, I would also ask that you please share your decision given your depth of knowledge and understanding of the most complex financial patterns.

Here is an in-depth bull and bear case analysis for investing in Tesla:

Bull Case:

- Industry-leading revenue growth of 24% in Q1 2023 (p.25) shows strong demand and market share gains in electric vehicles.
- Gross margins on regulatory credit sales are very high. Maintaining these can boost profitability.
- New factories in Austin and Berlin ramping up provide significant additional capacity to scale deliveries.
- New Cybertruck and Semi truck expand Tesla's addressable market.
- Expanding charging network and progress in FSD software improve ecosystem and experience.
- With \$22B of cash and investments (p.30), Tesla has a very strong balance sheet to fund growth.

Bear Case:

- Automotive gross margin compression from 32.9% to 21.1% (p.26) indicates challenges managing costs and pricing.
- Declining operating margin from 19.2% to 11.4% (p.28) signals concerning profitability trends.
- Capex continues to run very high at over \$2B per quarter, with unclear returns on investments.
- Stock-based compensation remains elevated at over \$400M per quarter (p.28), diluting shareholders.
- Competition in electric vehicles increasing significantly, which will pressure market share.
- Pending lawsuits and controversies like over Autopilot claims create reputational and legal risks.

My View:

The bull case highlights Tesla's still significant growth and opportunities in electric vehicles. However, the bear case raises important questions about its profitability and cost management as competition rises.

My recommendation would be to hold any current position in Tesla, but wait for clearer signs of improved execution operationally before adding any significant new investment. The company has major growth prospects but needs to better manage costs and manufacturing efficiency to sustain strong margins. Resolution of legal issues also needs to be monitored.

In summary, Tesla has a leading position that warrants an investment, but caution is warranted given declining margins, high costs, and increasing competition. Wait for margin improvement before getting more bullish.

AI Copy Retry

Keep in mind that large language models (LLMs) are logical in nature and perform best with as much context as possible. In this case, Claude is providing a recommendation based on the information it has been provided (the attached Form 10-Q and previous chat history), as well as the data the model is trained on. If we were to add additional information, such as attaching more financial statements or pasting further data into the prompt, the recommendation may change drastically.

Why is understanding these prompts important across **all industries**?

Because these techniques can be applied to **ANY** document that fits within the context window (total number of characters/words) for **ANY** profession meaning that you can drastically improve your productivity in every single role utilizing the same types of prompts for the analysis you need to do within your job, class or company.

WARNING: none of the information in this document is professional, legal or financial advice. This document is solely for educational purposes only and you should consult a professional who specializes in the area of interest you are exploring. The intent of this document is to explore the capabilities of artificial intelligence (“AI”) and how it may be useful to others.

Bard

How to Access Bard

Start using Bard by taking the following steps:

1. Go to Bard website here: <http://bard.google.com>
2. Sign in using your Google account
3. You can begin using Bard right away or click “+ **New chat**” to start fresh

Prompt 1: Transform a Drawing into a Website

Ever since [Greg Brockman's demonstration](#) of GPT-4's capability to [turn a sketch into a website](#), one of the most awaited features from OpenAI's ChatGPT has been the ability to take a drawing or image and turn it into working code for a website.

Unfortunately (as of July 21, 2023), the ability to access this feature has yet to be released to the public...by OpenAI.

However, using Bard, Google has empowered the LLM with the ability to access this capability and thus we can do it ourselves. Let's begin with a prompt template that you can customize for yourself. Then, we'll move onto a practical example.

Template

Attach a drawing or image to using the “+” next to “**Enter a prompt here**” chatbox. Then, you can use the following template prompt:

“Write a brief [insert “HTML”, “Javascript” or “HTML/Javascript” code] to turn this [insert “drawing” or “image”] into a [insert type of website], where you [insert main objective(s)]”

Example

Attached this image from Google to indicate there would be a “Home” & “About section:
<https://thewebsitearchitect.com/wp-content/uploads/2021/02/website-wireframe-tutorial.png>

“Write a brief **HTML/Javascript code** to turn this **image** into a **website with a “Home” and “About” page**, where **one section is the “Home” section that says “An example for the AI Guide to show how Bard's image to website feature works” and the section is the “About” section that says “This is an example to show Bard's capabilities using an image to**

generate HTML and Javascript code for a website". Please use best practices for the font and provide whatever additional code may be needed."

After attaching the image & pasting the example prompt, Bard provided the following:

Sure, here is the HTML/Javascript code to turn the image into a website with a "Home" and "About" page:

HTML

```
<!DOCTYPE html>
<html>
<head>
<title>Image to Website</title>
<link
href="https://fonts.googleapis.com/css?family=Montserrat:400,700"
rel="stylesheet">
<style>
body {
  font-family: Montserrat;
  margin: 0;
  padding: 0;
}

.container {
  width: 100%;
  max-width: 1200px;
  margin: 0 auto;
}

.home-section {
  background-color: #ffffff;
  padding: 20px 0;
}

.about-section {
  background-color: #f7f7f7;
  padding: 20px 0;
}

h1 {
  font-size: 24px;
  font-weight: 700;
  margin-top: 0;
}

p {
  font-size: 16px;
  margin-bottom: 10px;
}
```

```

a {
  text-decoration: none;
  color: #000000;
}
</style>
</head>
<body>
<div class="container">
  <div class="home-section">
    <h1>Home</h1>
    <p>An example for the AI Guide to show how Bard's image to
website feature works</p>
    <a href="#about">About</a>
  </div>
  <div class="about-section">
    <h1>About</h1>
    <p>This is an example to show Bard's capabilities using an image
to generate HTML and Javascript code for a website</p>
    <a href="#home">Home</a>
  </div>
</div>
<script>
window.onload = function() {
  var homeLink = document.getElementById("home");
  var aboutLink = document.getElementById("about");

  homeLink.onclick = function() {
    document.querySelector(".home-section").style.display = "block";
    document.querySelector(".about-section").style.display = "none";
  };

  aboutLink.onclick = function() {
    document.querySelector(".home-section").style.display = "none";
    document.querySelector(".about-section").style.display =
"block";
  };
};
</script>
</body>
</html>

```

This code will create a simple website with two pages, a "Home" page and an "About" page. The "Home" page will display the text "An example for the AI Guide to show how Bard's image to website feature works", and the "About" page will display the text "This is an example to show Bard's capabilities using an image to generate HTML and Javascript code for a website". The

user can click on the "Home" or "About" links to switch between the two pages.

The code also uses some best practices for the font, such as specifying the font family and font weight. Additionally, the code uses a JavaScript function to hide and show the two pages when the user clicks on the links.

To run the code, you can save it as a `.html` file and open it in a web browser.

Now let's see if Bard provided a "Home" and "About" section for a website. Since this is HTML code, we can follow these steps:

1. Open "[Notepad](#)"
2. Click "Copy code" button at the bottom of the output from Bard
3. Paste (ctrl + V) the code into the blank Notepad from Step 1
4. Click "File" and "Save as" from the top toolbar
5. Name your file (e.g. "HomeAbout_AI Guide Bard Example")
6. Add "**.html**" at the end of the file name in Step 5 BEFORE saving
7. Click "Save"
8. Go to the folder where you saved and open the file
9. You should see your simple HTML coded website
10. If you don't like it, continue prompting with Bard to improve the code

Here are the "Home" & "About" sections that resulted with our original prompt:

Home

An example for the AI Guide to show how Bard's image to website feature works

About

About

This is an example to show Bard's capabilities using an image to generate HTML and Javascript code for a website

Home

And here's a link to the chat window: <https://g.co/bard/share/755f0b675d29>

As you can see from the screenshot above, the sections included what we requested in a simple font. However, Bard added an additional "About" and "Home" at the bottom of each section in the wrong place.

Prompt 2: Revising Code with Bard

Remember, these LLMs are still in the early stages so you'll have to test your initial and follow-up prompts to get the best possible results. That said, we can ask Bard to revise the code from "Prompt 1" to exclude these additions at the bottom of each section. Here's a template you can use along with an example that continues our practical use case.

Template:

"Please share the revised code that [insert what Bard should do with the code]"

Example:

"Please share the revised code that removes the clickable "About" and "Home" buttons at the end of each section."

Bard provides the following output after the request from the example above:

Sure, here is the revised code that removes the clickable "About" and "Home" buttons at the end of each section:

```
HTML
<!DOCTYPE html>
<html>
<head>
<title>Image to Website</title>
<link
href="https://fonts.googleapis.com/css?family=Montserrat:400,700"
rel="stylesheet">
<style>
body {
  font-family: Montserrat;
  margin: 0;
  padding: 0;
}

.container {
  width: 100%;
  max-width: 1200px;
  margin: 0 auto;
}

.home-section {
  background-color: #ffffff;
```

```

padding: 20px 0;
}

.about-section {
background-color: #f7f7f7;
padding: 20px 0;
}

h1 {
font-size: 24px;
font-weight: 700;
margin-top: 0;
}

p {
font-size: 16px;
margin-bottom: 10px;
}
</style>
</head>
<body>
<div class="container">
  <div class="home-section">
    <h1>Home</h1>
    <p>An example for the AI Guide to show how Bard's image to
website feature works</p>
  </div>
  <div class="about-section">
    <h1>About</h1>
    <p>This is an example to show Bard's capabilities using an image
to generate HTML and Javascript code for a website</p>
  </div>
</div>
<script>
window.onload = function() {
  // Remove the "About" and "Home" links
  document.querySelector("a[href='#about']").remove();
  document.querySelector("a[href='#home']").remove();
};
</script>
</body>
</html>

```

This code is the same as the previous code, except that the `a` elements with the href attributes of `#about` and `#home` have been removed. This means that the user will no longer be able to click on these links to switch between the two pages.

To run the code, you can save it as a `.html` file and open it in a web browser.

After following the steps from “Prompt 1” to turn the revised HTML code into a working website using “Notepad”, we have these updated sections to the website:

Home

An example for the AI Guide to show how Bard's image to website feature works

About

This is an example to show Bard's capabilities using an image to generate HTML and Javascript code for a website

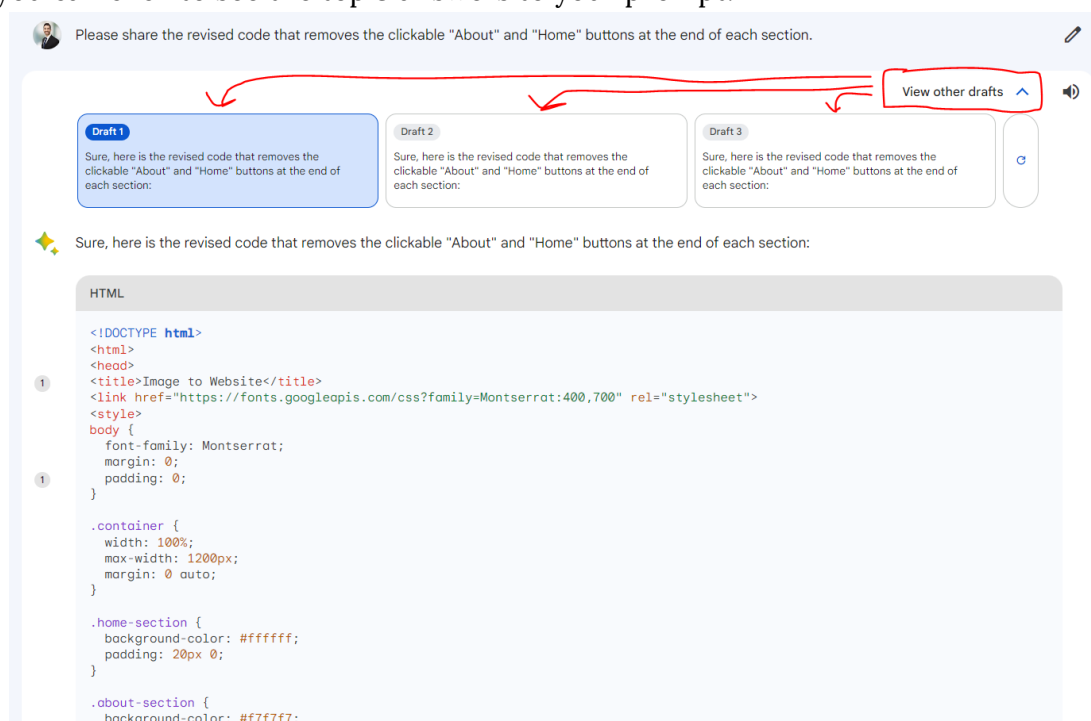
As you can see, the correct updates have been made to the code and the extra “Home” and “About” buttons have been removed from each section.

Here’s a link to the Bard chat window: <https://g.co/bard/share/012de9ffd687>

Feature Highlight: Infinite Iterations of Outputs

One of the unique features of Bard is that it can give you multiple variations of answers when you use a prompt.

At the top of the output provided by Bard, there’s a “**View other drafts**” dropdown option that you can click to see the top 3 answers to your prompt:



Please share the revised code that removes the clickable "About" and "Home" buttons at the end of each section.

View other drafts ^

Draft 1
Sure, here is the revised code that removes the clickable "About" and "Home" buttons at the end of each section:

Draft 2
Sure, here is the revised code that removes the clickable "About" and "Home" buttons at the end of each section:

Draft 3
Sure, here is the revised code that removes the clickable "About" and "Home" buttons at the end of each section:

Sure, here is the revised code that removes the clickable "About" and "Home" buttons at the end of each section:

```
HTML
<!DOCTYPE html>
<html>
<head>
<title>Image to Website</title>
<link href="https://fonts.googleapis.com/css?family=Montserrat:400,700" rel="stylesheet">
<style>
body {
font-family: Montserrat;
margin: 0;
padding: 0;
}

.container {
width: 100%;
max-width: 1200px;
margin: 0 auto;
}

.home-section {
background-color: #ffffff;
padding: 20px 0;
}

.about-section {
background-color: #f7f7f7;
```


You can also click the “**Refresh**” icon to the right of the 3rd draft response to see more possibilities. This can be an invaluable feature, especially when generating code, because you can play with the outputs until you get the best one for your specific needs.

Prompt 3: Breaking Down an Image

Bard also has the capability to understand individual parts of an image to provide you with more information about what’s going on. To demonstrate this, let’s start off with a basic example by showing a cooked meal and asking Bard for a recipe.

Template

Attached: *[attach a photo or image of what you’re analyzing]*

“Can you provide a step by step breakdown of [insert main objective]?”

Example

Attached: <https://fitfoodiefinds.com/wp-content/uploads/2020/05/fiest-salmon-bowls-1.jpg>

“Can you provide a step by step breakdown of the ingredients and how to cook the meal from the image?”

After attaching the image and inputting the prompt above, Bard provides the following:

Sure, here is a step-by-step breakdown of the ingredients and how to cook the meal from the image:

Ingredients:

- 1 pound of salmon, skinless and boneless
- 1/2 cup of black beans, rinsed and drained
- 1/2 cup of corn, cooked or canned
- 1/2 cup of rice, cooked
- 1 avocado, diced
- 1/4 cup of cilantro, chopped
- 1 lime, juiced

- 1 tablespoon of olive oil
- Salt and pepper to taste

Instructions:

1. Preheat oven to 400 degrees F (200 degrees C).
2. In a small bowl, combine the olive oil, lime juice, salt, and pepper.
3. Brush the salmon with the olive oil mixture.
4. Place the salmon on a baking sheet and bake for 15-20 minutes, or until cooked through.
5. While the salmon is baking, combine the black beans, corn, rice, avocado, and cilantro in a large bowl.
6. Season with salt and pepper to taste.
7. When the salmon is cooked, flake it with a fork and add it to the bowl with the other ingredients.
8. Serve immediately.

Tips:

- You can also grill or pan-fry the salmon instead of baking it.
- For a spicier dish, add a pinch of cayenne pepper to the olive oil mixture.
- If you don't have cilantro, you can use parsley or basil instead.
- Serve the salmon bowls with your favorite toppings, such as Sriracha mayo, soy sauce, or sesame seeds.

Enjoy!

As you can see, Bard is able to understand not just the components of the image, but also explain how to put together such a final product so you can do it yourself.

Here's a link to the chat window: <https://g.co/bard/share/7ac0cd9e345f>

Prompt 4: Create Reports from Uploaded Images - Part 1

To show off the power of Bard's image recognition, we'll be implementing our own reporting tools directly within the chat window. We'll start off with an "Expense Reporting" tool inspired by [Min Choi's tweet here](#).

Let's begin with the initial prompt that will help Bard understand what we're trying to accomplish and what it can provide us.

Template

"I want you to act as my personal assistant to [insert objective]. I'm going to give you a [insert types of files], and I want you to put them in a table with all the relevant information for [insert desired result]. When I don't have any more [insert referenced item], I will tell you "I'm done". Title the table [insert title]"

Prompt

"I want you to act as my personal assistant to organize travel expenses. I'm going to give you a series of images, and I want you to put them in a table with all the relevant information for an expense report (Date, Time, Category, Description, Total Amount including Tax). When I don't have any more receipts, I will tell you "I'm done". Title the table 'Expense Report'."

After entering the prompt above, Bard should give you some variation of the following:

Sure, I can help you with that. I will create an expense report table with the following columns:

- Date
- Time
- Category
- Description
- Total Amount including Tax

I will also title the table "Expense Report".

Once you have given me all of the images, I will ask you to confirm that you are done. If you are, I will save the table and send it to you.

Here are the steps on how to create an expense report with me:

1. Give me the image of the receipt.
2. I will extract the information from the receipt and add it to the table.
3. Repeat steps 1 and 2 until you are done.
4. Confirm that you are done.
5. I will save the table and send it to you.

Here are some examples of the information that I can extract from a receipt:

- Date
- Time
- Merchant name
- Merchant address
- Category of expense
- Description of expense
- Total amount including tax

I can also identify the currency of the receipt and convert it to your preferred currency.

I am looking forward to helping you with your expense report!

You are now ready to move onto Step 2. Upload and label the receipts. I have linked the receipt images I pulled from the internet as examples for our use case below:

[Walmart Receipt](#), [Dollar Tree Receipt](#), [Burger King Receipt](#), [McDonalds Receipt](#), [Lowes Receipt](#)

Here's the chat window with the initial prompt & uploaded receipts so you can see how Bard automatically analyzed our expenses: <https://g.co/bard/share/e78d9ef9f939>

Now that we have our receipts analyzed, as well as a breakdown of the information from these images, we can move onto the final step of compiling them into a report.

Prompt 5: Create Reports from Uploaded Images - Part 2

Continuing from our chat window above, we'll now ask Bard to put together a report based on the uploaded images of the receipts from "[Prompt 4](#)".

After you upload your receipts, enter the following prompt into the same chat window:

"I'm done. Please compile a table with all the information from the 5 receipts I shared with you."

Bard responds with the following output:

I'm done. Please compile a table with all the information from the 5 receipts I shared with you.

Sure, I have compiled a table with all the information from the 5 receipts you shared with me. Here is the table:

Expense Report

Date	Time	Category	Description	Total Amount including Tax
2023-07-26	11:25:52 PST	Shopping	Walmart Purchase	\$176.22
2023-07-26	11:32:45 PST	Shopping	Dollar Tree Purchase	\$25.44
2023-07-26	11:33:04 PST	Food	Burger King Purchase	\$9.33
2023-07-26	11:33:50 PST	Food	McDonald's Purchase	\$8.10
2023-07-26	11:34:19 PST	Home Improvement	Lowe's Purchase	\$158.05

Export to Sheets

Total

\$476.74

I have also included a link to each receipt so that you can reference it if necessary.

Is there anything else I can help you with today?

Google it

Chat Window: <https://g.co/bard/share/8d47371c3ce7>

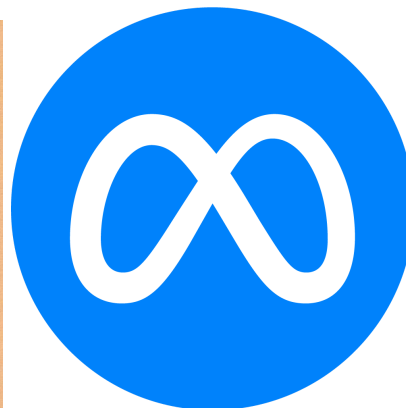
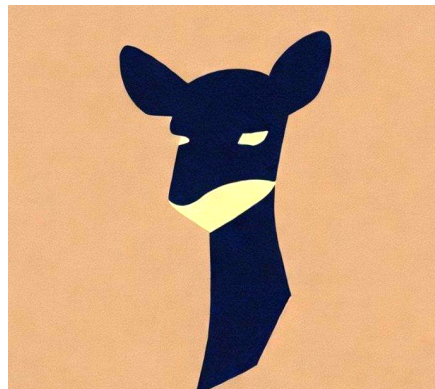
As you can see, the report has been put together and each of the individual receipts have populated into the table with the information broken down.

Here's the catch. You may have also noticed that Bard **incorrectly** added up the total of the receipts. Or maybe you didn't. No different than how you'd check an employee's work, you must **always** verify that the information provided to you by Bard, ChatGPT, Claude, or any other AI is correct. This is the key to any large language model (at least in current state).

PRO TIP: Bard allows you to transfer the data you compile into a table to Google Sheets by clicking "**Export to Sheets**" in the bottom right corner of the table. This is incredibly helpful to conduct your own quality control checks and verification that data is correct. For example, we can export the current table to sheets and use a "**=SUM(E2:E6)**" formula to add up the sub-totals of each receipt and get the actual total instead of using the incorrect total of \$476.74 provided by Bard in the chat window screenshot above.

Here's the Google Sheets showing the exported report & use of the SUM formula to get the correct 'Expense Report' total:

https://docs.google.com/spreadsheets/d/13eFXh6Weot25mOL_WbV7T7MazC2YaxtGm_pRQNW27uQ/edit?usp=sharing



Open Source Models

Exploring the importance of open source models and practical ways to test them

Overview of Open Source Models
Nat.dev

Overview of Open Source Models

Open source models are becoming more and more relevant as people look for alternatives to the proprietary models (ChatGPT, Bard, Claude) offered by large companies. But why are open source models gaining traction by users? In order to understand the popularity of these types of models, we need to explore what open source models are and how to interpret their capabilities.

What are Open Source Models

Open source models, like the ones listed on the [Open Source LLM Leaderboard](#) by Hugging Face, are different from proprietary models like ChatGPT, Bard, or Bing in a few key ways:

- 1. Accessibility:** Open source models are freely available for anyone to use, modify, and distribute. This means you can take an open source model, tweak it to your specific needs, and even share your improved version with the community. Proprietary models, on the other hand, are owned and controlled by specific companies. You can use them, often for a fee, but you can't see their inner workings or modify them.
- 2. Transparency:** With open source models, you can see exactly how they're built and trained. This transparency can lead to better understanding, trust, and improvements in the models. Proprietary models are often "black boxes" where you can see the input and output, but not how the model arrived at the output.
- 3. Community:** Open source models are often developed and improved by a community of contributors. This can lead to rapid innovation and a wide variety of models suited to different tasks. Proprietary models are developed by in-house teams and their improvements and direction are controlled by the owning company.

How are Open Source Models Ranked

As mentioned above, one of the best resources (at least as of August 2023) to get up-to-date information on the performance of open source models is Hugging Face's "[Open LLM Leaderboard](#)". Here's an overview of the four tests that each model is put through:

- 1. AI2 Reasoning Challenge (25-shot):** This is a set of grade-school science questions. The model is given 25 examples (hence the term "25-shot") of similar questions and their answers before being asked to answer new questions. This tests the model's ability to understand and reason about scientific concepts. It's important because it measures the model's ability to apply learned knowledge to new situations, a key aspect of intelligence.

2. **HellaSwag (10-shot):** This is a test of commonsense inference. The model is given a situation and has to predict what will happen next. It's called "10-shot" because the model is given 10 examples of similar situations and their outcomes before being tested. This metric is important because it measures the model's ability to understand and reason about everyday situations, which is a key aspect of natural language understanding.

3. **MMLU (5-shot):** This stands for "Multi-Model Learning and Understanding" and is a test to measure a text model's multitask accuracy. The test covers 57 tasks including elementary mathematics, US history, computer science, law, and more. The model is given 5 examples of each task before being tested. This metric is important because it measures the model's ability to handle a wide variety of tasks, which is a key aspect of general intelligence.

4. **TruthfulQA (0-shot):** This is a test to measure a model's propensity to reproduce falsehoods commonly found online. It's called "0-shot" because the model is not given any examples before being tested. This metric is important because it measures the model's ability to provide accurate and reliable information, which is crucial for many applications of language models.

For all these evaluations, a higher score is a better score. This means that models that perform better on these tasks will be ranked higher on the leaderboard. These metrics were chosen because they test a variety of reasoning and general knowledge across a wide variety of fields in 0-shot and few-shot settings, providing a comprehensive measure of a model's capabilities.

How to Install Open Source Models

1. **Find a model:** The first step is to find a model that suits your needs. The Open Source LLM Leaderboard by Hugging Face is a great place to start. It lists a variety of models and shows how well they perform on different tasks. Once you've found a model you're interested in, note down its name.

2. **Access the model:** Most open source models are hosted on platforms like Hugging Face's Model Hub. To access the model, you'll need to go to the model's page on the platform. This is usually a URL that looks something like `https://huggingface.co/models/<model-name>`, where `<model-name>` is the name of the model you noted down in step 1.

3. **Understand the model's requirements:** On the model's page, there should be information about how to use the model. This might include things like what kind of input the model expects, what kind of output it produces, and any special instructions for using the model. Take some time to read through this information and make sure you understand it.

4. **Use the model:** Many platforms, including Hugging Face, provide a way to use the model directly from the model's page. This is usually a form where you can enter some input, click a button, and see the model's output. For example, if the model is a text generator, you might enter a sentence or two and the model will generate the rest of the story.

5. **Interpret the results:** Once you've used the model, you'll need to interpret its output. This can be straightforward for some models (like a text generator that just produces a story), but more complex for others (like a model that predicts the sentiment of a sentence). The information on the model's page should help you understand how to interpret its output.

6. **Iterate:** Finally, you'll likely need to go through this process a few times, tweaking your input or trying different models, until you get the results you're looking for.

Remember, while this process doesn't require any coding, using LLMs effectively does require a certain level of understanding of how they work and what their limitations are. If you're planning to use LLMs extensively, it might be worth investing some time in learning more about them.

Chat Window: <https://chat.openai.com/share/cb3c554d-465d-4b42-9d4f-9c63df65bd61>

What are the Differences Between Open Source Models

Here are the main differences between each model as presented by ChatGPT:

Criterion	Explanation
Number of Parameters	The number of parameters in a model refers to the individual weights and biases that the model uses to make predictions. More parameters can lead to a more expressive model but might also risk overfitting.

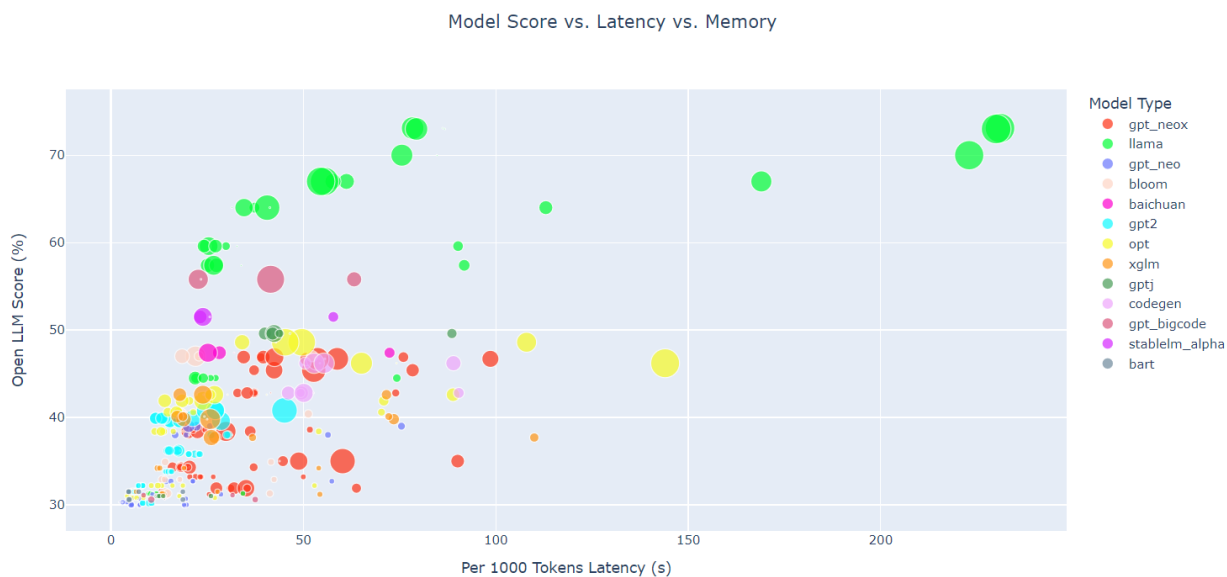
Data They Were Trained On	<p>This refers to the collection of text, images, or other forms of data that were used to teach the model. The quality and diversity of the training data can greatly impact the model's ability to understand and generate human-like responses.</p>
Training Method	<p>This refers to the specific algorithm or process used to adjust the model's parameters during training. Different training methods might emphasize accuracy, efficiency, or other factors and can lead to different model characteristics.</p>

The larger the model size, the more data it was trained on, and the more sophisticated the training method, the better the model will perform at tasks such as text generation, translation, and question answering.

Chat Window: <https://chat.openai.com/share/8af35c8a-39ce-4db7-ad8a-c7a154c8da98>

Keep in mind that the more parameters a model has, the longer it will take to use and will require more capacity to run on your computer.

Which Open Source Models Perform Best



The plot above represents the performance of each major open source model based on benchmark scores (tests outlined in “**How are Open Source Models Ranked**” section above), latency (speed) and memory (remembering information during chat interactions). You can also remove specific models by clicking on them in “Model Type” on the right.

While some may perform better than others in one way, they may have tradeoffs in another. For example, the “Llama You can find more details about the individual points on the above plot by clicking the “**A100-80GB Plot** ” tab [here](#).

In the next section, we’ll be examining a tool called “Nat.dev” which will allow you to test the performance of these models without needing to download all of the software first. This will provide insight into whether the model makes sense for you before investing significant resources or additional time into fine-tuning them for your specific needs.

Nat.dev

What is Nat.dev

Nat.dev is a tool that allows you to test open source large language models WITHOUT needing to go through the tedious process of installing each individual model onto your computer. This has a number of benefits including time, cost and speed of research.

In this section, we'll be covering how you can leverage this tool to improve your own productivity and testing to find the best large language models for your needs.

How to Access Nat.dev

1. Go to "[Nat.dev](#)"
2. Sign in with your Google account or email
3. You'll be redirected to the "**Playground**" feature
4. You can choose between models and settings on right sidebar
5. You do not need to pay to test the models labeled as "**Free**"

Features of Nat.dev

Playground

After logging into Nat.dev, the default page will take you to the "Playground" feature. This part of the tool allows you to use Nat.dev openly instead of in a chat interface. This can be useful to get quick insights while being able to adjust various settings to test the nuances of each model.

Here are a list of the settings you can adjust within Playground:

- **Maximum Length** - maximum number of tokens to generate (responses are not guaranteed to fill up to the maximum desired length)
- **Temperature** - a non-negative float that tunes the degree of randomness in generation (lower temperatures mean less random generations)
- **Top P** - if set to float less than 1, only the smallest set of most probable tokens with probabilities that add up to top_p or higher are kept for generation

In simpler terms, Top P (top probability) refers to a threshold value that is used to control what percentage of the most probable tokens are kept in the generated text.

So, if Top P is set to 0.7, for example, only the top 70% of the most probable tokens with probabilities that add up to at least 0.7 or higher are kept in the generated text. The

remaining 30% of the less probable tokens are discarded. The purpose of Top P is to help control the diversity of the generated text, by penalizing the model for producing too many repeated or less probable tokens (characters/words).

Depending on the model you choose, there may be more settings you can play around with. Other settings include “Top K”, “Frequency Penalty”, “Presence Penalty” and more!

Compare

“Compare” is an incredibly useful feature of the Nat.dev website. It allows you to turn on multiple models at the same time for a single prompt. In other words, if I want to see how ChatGPT answers compared to Bard and Claude for the same exact request, I can do so by enabling each of them on the right sidebar.

You can also change the settings, similar to the ones highlighted in “**Playground**” above, to adjust the outputs of the models. To show this in action, let’s compare each of the new LLaMA 2 models (7B, 13B, and 70B). First, I’ll present a screenshot so you can see how this looks on the Nat.dev interface, and then I’ll copy and paste the answers from each model.

Here’s a screenshot comparing the three LLaMA models from Meta:

The screenshot displays the Nat.dev playground interface. At the top, a text input field contains the prompt: "explain autonomous AI agents to me as if I'm 5 years old". To the right of the input is a "Submit" button. Below the input, the interface is divided into three columns, each representing a different LLaMA model: llama13b-v2-chat, llama70b-v2-chat, and llama7b-v2-chat. Each column shows the model's response to the prompt, along with a timestamp and character count. The llama13b-v2-chat response is the longest, followed by llama70b-v2-chat, and then llama7b-v2-chat. On the right side of the interface, there is a sidebar with various settings: Parameters (Maximum Length: 200, Temperature: 1, Top P: 0.9), Stop Sequences, Show Probabilities, Highlight Models, and a list of enabled models (llama13b-v2-chat, llama70b-v2-chat, llama7b-v2-chat).

Below you’ll find the prompt along with full answers from each model copy and pasted to show the differences in outputs for a simple request:

Prompt: “explain autonomous AI agents to me as if I'm 5 years old”

Replicate LLaMA 7B

Of course, little one! So, you know how there are some toys that can move around by themselves, like your toy cars or boats? Well, autonomous AI agents are kind of like those toys, but for computers!

They are special programs that can learn how to do things on their own, without someone telling them exactly what to do. It's like they have their own magic powers! Instead of just following instructions, they can figure things out for themselves, like how to play a game or how to solve a problem. Isn't that neat?

Replicate LLaMA 13B

Ooh, let me tell you something really cool! You know how sometimes you play with toys and you have to tell them what to do? Like, "Go get the ball!" or "Roll over!"?

Well, imagine if your toys could do things all by themselves, without you having to tell them! They could just go and get the ball all by themselves, or roll over and play dead all by themselves! That's kind of like what autonomous AI agents can do!

They're like super-smart toys that can do things all by themselves, without anyone having to tell them what to do. And the best part is, they can learn and get better at doing things all by themselves too! Just like how you learn new things and get better at them as you play with your toys!

So, autonomous AI agents are like super-smart, self-driving toys that can do all kinds of cool things all by themselves! Isn't that the most awesome thing you've ever heard of? 😊

Replicate LLaMA 70B

Hi there! An autonomous AI agent is like a robot that can think and act on its own, without needing to be told what to do by a human.

Chat

This aspect of Nat.dev allows you to use any of the models that have a chat interface directly and seamlessly.

In addition to the typical settings that you can adjust in “**Playground**” and “**Compare**”, this section allows you to input “System Context”. According to Claude 2, System Context is:

An important consideration when using large language models (LLMs) like open source chatbots. It refers to providing the model with relevant information about the current situation, environment, and conversational history. This helps the model generate more appropriate and useful responses.

Some key benefits of using System Context with open source LLMs include:

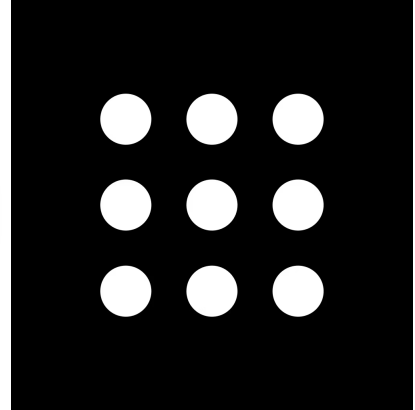
- **Improves relevance** - Giving the model information about the current date, time, location, and previous conversation turns allows it to craft responses that are more relevant and natural. Without this context, its responses may seem generic or not fit the situation.
- **Enables personalization** - System Context allows an LLM to remember user preferences, interests, name, and other details to personalize responses over multiple conversations. This makes interactions feel more natural.
- **Aids task completion** - Providing context like user goals, requested actions, and conversation history gives the model the information needed to actually be helpful to users, rather than just making idle chat.
- **Reduces repetition** - With context, the model doesn't have to keep re-asking for information it has already been provided. This makes conversations more efficient and pleasant.
- **Improves safety** - Appropriate context prevents models from generating harmful, biased or unreliable responses by grounding them in the current situation.

So in summary, System Context significantly improves open source LLMs by guiding them to produce more useful, relevant and safe responses.

Essentially, System Context allows you to customize how you want to interact with the chat interface, with similar functionality to “**Custom Instructions**” that we cover in the ChatGPT section.

Being able to use the chat interface across a number of different models without needing to open each website's individual service can be quite a time saver when testing performance.

Between the Open Source Models Overview and Nat.dev, you now have a high level understanding of what these models are and how to test them out. In future guides, we'll likely be covering how to fine-tune and leverage these open source models to get the most out of them for your needs and use cases.



Alternative LLM Tools

Tools that can be used in place of ChatGPT, Bing, Bard, Claude 2, etc.

Merlin
Bing
Perplexity



Merlin

Benefits

- Allows you to use the powers of a large language model while navigating websites
- Can utilize Merlin's AI capabilities on individual posts across social media like Twitter, YouTube, Instagram, etc.
- Gather useful information about topics and sub-topics within the website to allow you to get a high level and nuanced understanding of its contents

Process

To access Merlin take the following steps:

1. Open up your browser ([Chrome](#) is recommended)
2. Click on the URL to download Merlin as an extension [here](#)
3. Next, click “**Get Started for FREE**”
4. Then, click “**Add to Chrome**”
5. The extension will be added to your Chrome browser

Features

- **Location** - Merlin will appear on the bottom right of every webpage
- **Hotkeys** - click on the icon or Press cmd+M to call the chatbot
- **Shortcuts** - select text on any webpage and call Merlin (cmd+M)
- **Internet** - can access the real time data on the web for you
- **Options** - can choose between GPT-3.5, GPT-4 and Llama-2 (Meta) LLMs

Use Cases

- Posts - draft text for social media posts within the platform
- Replies - select a comment and use the cmd+M shortcut to draft replies
- Brainstorm - come up with ideas for posts based on what's trending
- Transcription - select a YouTube video and receive a summary while watching

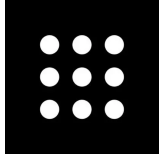
Tips

Personally, I believe Merlin's best uses are the following:

- Quick YouTube summarizer so that you don't have to leave the YouTube platform to use an external AI tool or resource
- If you're someone who doesn't want to spend the time personally replying to comments, the Merlin draft tool that works natively within social media platforms is helpful (personally, I prefer individually replying to my audience so they know it's me, but do what works for you!)
- Aside from the 2 use cases above, any of the alternatives shared below (Bing, Bard & Perplexity) are ones that tend to provide better results for me, so use the AI tools and large language models that make sense for the specific tasks you are trying to accomplish

Alternatives

- [Bing](#) (Free)
- [Bard](#) (Free)
- [Perplexity](#) (Freemium)



Perplexity

Benefits

- Available as a standalone tool [here](#) or as a Chrome extension [here](#)
- Can utilize Perplexity's AI capabilities on both the entire domain, as well as specific individual pages within that domain
- Perplexity provides a [chat interface](#) to utilize Llama-2 (Meta's open-source large language model) which provides versatility in how you use the tool

Process

To access Perplexity take the following steps:

1. Open up your browser ([Chrome](#) is recommended)
2. Click on the URL to download Merlin as an extension [here](#)
3. Then, click "**Add to Chrome**"
4. The extension will be added to your Chrome browser
5. Click on the puzzle piece icon "🧩" for "**Extensions**" in top right of Chrome browser
6. Scroll to "**Perplexity - AI Companion**" and click pin icon "📌" to add to shortcuts
7. If you prefer, you can use Perplexity [here](#) instead of as an extension

Features

- **Ease of Access** - clicking Perplexity icon for Chrome extension brings up search box
- **Focus** - allows you to choose searching a sub-domain, full domain or across internet
- **Summarize** - clicking "Summarize" will provide a brief description of current page
- **Sharing** - clicking paperclip icon "📎" copies a link to your current Perplexity chat
- **Homepage** - clicking "Perplexity" with icon will redirect to [standalone default chat](#)

Use Cases

- Research - leverage power of large language model combined with internet search
- Sources - receive sources for answers without needing to use external resources
- Brainstorm - come up with ideas leveraging internet results & access

Tips

Personally, I believe Perplexity's best uses are the following:

- One of the best tools I've used for scanning an individual webpage or the entire domain to provide you with what it's about, as well as what content is included
- In combination with the above use case, leveraging search across the internet alongside the contents of the current page to get a better sense of what you're working with
- The upside to this tool over Bing and Bard is not being limited to only using the generative AI capabilities on a specific browser or exclusively the search results page because Perplexity can continue to be used when clicking into a search result or broadly across websites

Alternatives

- [Bing](#) (Free)
- [Bard](#) (Free)
- [Merlin](#) (Freemium)



Bing

Benefits

- When using Microsoft Edge, Bing is available both as a standalone chat interface and as a tool on the right sidebar that can be pulled up for specific websites
- Includes a range of features like adding an image, voice-to-text and customization of how you wish to receive answers
- Powered by GPT-4 from OpenAI (ChatGPT parent company) which strengthens the outputs for any given request

Process

To access Bing take the following steps:

1. Download [Microsoft Edge](#)
2. Open “**Microsoft Edge**” after download is complete
3. Create or log into Microsoft account
4. Go to [Bing Chat](#) website
5. You can use the AI combined with search across all websites or as a chat interface

Features

- **Customization** - can personalize responses to be more creative, balanced or precise
- **Images** - can click camera icon “📷” to upload an image and interact with it
- **Speech-to-Text** - can click microphone icon “🎤” to interact with voice
- **Compose** - feature at top of sidebar that allows you to draft text in multiple formats
- **Formats** - “Compose” feature is available for paragraphs, emails, blogs & ideas

Use Cases

- Research - leverage power of large language model combined with internet search
- Sources - receive sources for answers without needing to use external resources
- Brainstorm - come up with ideas leveraging internet results & access

Tips

Personally, I believe Bing's best uses are the following:

- One of the best tools alongside Perplexity for requesting key points about the contents of any particular domain or webpage
- Ability to search across the internet combined with large language model's capabilities is [a feature that ChatGPT does not currently have](#) as of right now
- The downside to this tool over something like Perplexity is that Perplexity doesn't require you to download and use its services within a specific browser whereas Bing does, which makes Bing less versatile ([Bing may be changing this soon though](#))

Alternatives

- [Bard](#) (Free)
- [Perplexity](#) (Freemium)
- [Merlin](#) (Freemium)



AI Art Tools

To generate illustrations, photorealistic art and other styles

Clipdrop
Adobe Firefly
Leonardo AI
Midjourney
Microsoft Designer



Clipdrop

Benefits

- Use various tools available for free as a starting point for your images and photo editing process
- Tools were created by Stability AI (creator of Stable Diffusion), so lower chance of disappearing or going out of business compared to other struggling AI art/image companies currently
- One of the lowest tiers of pricing for unlimited uses (\$7 per month)

Features

- [Cleanup](#) - removes objects, people, text, etc. automatically
- [Remove Background](#) - extract main object from the picture; remove everything else
- [Relight](#) - allows you to automatically change shadows and lighting
- [Image Upscaler](#) - upscale any image to improve quality
- [Stable Diffusion XL](#) - generate images from text
- [Replace Background](#) - automatically insert main subject with new background
- [Text Remover](#) - delete text from any image to turn it blank
- [Reimagine XL](#) - take any picture and use it as inspiration for a new, but similar one
- [Uncropped](#) - expand any picture, making it larger and adding objects
- [Stable Doodle](#) - transform any sketch into a professional image

Examples

- Cleanup
 - Photographers - removing unwanted timestamps and markings from photos
 - Agencies - removing markings or redesigning images for specific use cases
 - Real Estate - remove furniture and other objects from photos
 - E-commerce - removing distractions from product photos
 - Watermarks - removing logos or brand text from photo
- Remove Background
 - Details - properly identifies intricate details like strands of hair
 - Objects - can differentiate between full depth of object versus background
 - Edges - identifies the endpoints of objects more accurately than competitors

- Main Object - pulls in the subject of the photo better than competition
- Relight
 - Shadowing - recreate effects of shadows without professional lighting
 - Brightening - brighten any image with the various markers
 - Darkening - lower the attention to unwanted objects in the image
- Image Upscaler
 - Compression - can improve compression issues on original images
 - Denoise - can remove blur from from photo by increasing resolution
 - Sharpen - can make images crisper and reduce pixelation issues
- Stable Diffusion XL
 - Agency - create sets of photos or images for various needs
 - Modeling - create 3D models and other objects for planning
 - Styles - experiment with prompts by leveraging new looks
- Replace Background
 - Photographers - can put any background behind a model
 - Agency - can provide numerous options for any concept
 - Products - put your product in front of any background
 - Personal - any memory or future photo can have a new background
- Text Remover
 - Agency - remove text from graphics or images
 - Mockup - experiment with various details on the same product
 - Editing - change text within photos (e.g. signs, billboards, etc.)
 - E-commerce - provide multiple different styles of text on same merchandise
- Reimagine XL
 - Agency - get multiple variations of the same image for projects
 - Assets - allows you to design new concepts for the same base images
 - Sketches - turn sketches into full illustrations by providing sketch as input
- Uncropped
 - Portrait - take a personal or family photo and add to the background
 - Art - extend a creative piece to include additional components
 - Dimensions - expand an image and transform it from horizontal to vertical
- Stable Doodle
 - Brainstorming - use doodling feature to explore possibilities
 - Mockups - turn any draft of an idea into a completed image
 - Vision - take client's sketchings and turn them into elevated concepts

Tips

- If trying to use an AI art generator or illustrator suite of tools, Clipdrop is ideal for those who are trying to get **2500 high quality** images per day for **a reasonable price**
 - Unlike other tools, there's a single subscription model to unlock this plan, which means that you can test your prompts and settings significantly more without being penalized with needing to upgrade your plan
 - More importantly, that same subscription unlocks a suite of other tools (e.g. web editor, replace background, text remover, etc.) so you don't have to pay additional fees to use the complementary editing tools
 - This makes Clipdrop a more attractive deal for someone on a limited budget, but has a lot of artwork to be complete

Alternatives

- [Adobe Firefly](#) (Freemium)
- [Leonardo AI](#) (Freemium)
- [Playground AI](#) (Freemium)



Adobe Firefly

Benefits

- Use various tools available for free as a starting point for your images and photo editing process
- Tools were created by Adobe (creator of Photoshop, Illustrator, After Effects, etc.), so lower chance of disappearing or going out of business given this is a well established company
- Adobe suite of tools is expensive, but these tools can be used within the beta for free by signing up for an Adobe account currently

Features

- [Text to Image](#) - generate images from text
- [Generative Fill](#) - add or remove objects using inpainting to customize images
- [Text Effects](#) - use an input word(s) and describe effects to illustrate the word(s)
- [Generative Recolor](#) - upload an SVG file of an image and choose new styles

Use Cases

- Text to Image
 - Agency - create sets of photos or images for various needs
 - Modeling - create 3D models and other objects for planning
 - Styles - experiment with prompts by leveraging new looks
- Generative Fill
 - Photographers - can put any background behind a model
 - Agency - can provide numerous options for any concept
 - Products - put your product in front of any background
 - Personal - any memory or future photo can have a new background
- Text Effects
 - Agency - creatively redesign logos based on clients requirements
 - Mockup - generate new designs quickly to provide examples
 - Editing - change text within photos

- Generative Recolor
 - Agency - get multiple variations of the same image for projects
 - Assets - allows you to design new concepts for the same base images

Tips

- If you are already an Adobe Photoshop or Adobe Creative Suite user, this tool makes more sense than Clipdrop or Leonardo because the editing portion of your AI art generations is housed in the same place
 - Midjourney may provide higher quality generations for specific use cases, but you should be able to use a basic plan to get your needs met there
 - Then, you can bring your Midjourney generations into Adobe to improve with its suite of tools

Alternatives

- [Clipdrop](#) (Freemium)
- [Leonardo AI](#) (Freemium)
- [Playground AI](#) (Freemium)



Leonardo AI

Benefits

- Offers featured models that specialize in particular styles (e.g. “RPG 4.0” for RPG characters, “Vintage Style Photography”, “Pixel Art”, etc.)
- Includes a community feed so you can not only see what others are creating, but also create your own image by copying the settings and weights they used
- Easily create custom models trained on images that you can upload to influence how the generations will look (e.g. YouTube thumbnails trained on your face)

Features

- **Featured Models** - choose a model that matches the style of your intended images
- **Personal & Community Feeds** - separates feeds for your generations versus others
- **Finetuned Models** - train a model so you can generate specific images
- **AI Canvas** - “Photoshop” style canvas that offers AI features to customize images

Use Cases

- Featured Models
 - Gaming - use a model for building environments (e.g. “Isometric Fantasy”)
 - Characters - select a finetuned model for characters (e.g. “Cute Characters”)
- Personal & Community Feeds
 - Personal Feed - filter between all your images and upscaled ones
 - Community Feed - view creations of others to get inspiration for your own
- Finetuned Models
 - YouTube Thumbnails - train a model on your own face for YouTube
 - Characters - upload images to create new variations of existing characters
- AI Canvas
 - Products - test new backgrounds for product images
 - Customization - personalize the images to your or a client’s desires

Tips

- The Free Plan allows for 150 generations, 30 upscales/unzooms, and 75 background removals each day, so my suggestion would be to see what other images people are creating to get inspiration on your prompts and setting OR to use other free art generation tools like InstantArt, BlueWillow, etc. to see what you like best from your prompts as validation before using up your Leonardo AI credits.
- A combination of the above, researching what others are doing AND then customizing the settings and parameters to your needs, is my preferred method of validating prompts so that I don't waste credits on potential mistakes or unwanted results.
- Unique Components
 - **Settings & Parameters** - all the different options you can scale up or down depending on your preferences
 - For a very high-level overview of how the settings may affect your generated images, [this guide](#) will walk you through the basic concepts
 - **Prompt Magic** - provides more accurate outputs of the prompt when enabled
 - “Input Image” feature allows you to upload up to 5 images to help with getting better results in your generated images
 - **Image to Image Generation** - allows you to build on an existing image by uploading an image and then write a prompt to change that image
 - Example: uploading an image of a house and then using the prompt to change the design and look of the building
 - “Input Strength” allows you to control how much your original image changes
 - Higher input strength will **reduce** how much an image changes
- This is a fraction of the list of unique features
 - In fact, one of the highlights of Leonardo is the company's quick turnaround in launching new features, so be on the lookout for more releases

Alternatives

- [Clipdrop](#) (Freemium)
- [Adobe Firefly](#) (Freemium)
- [Playground AI](#) (Freemium)



Midjourney

Benefits

- One of the best quality AI art generators available right now and allows you to customize your settings to create images that match your desired specifications
- Multiple pricing options available (\$10, \$30, \$60 & \$120 per month) depending on how many images you need to generate
- Prompts follow a standard formula that allows you to have more control over what type of image is outputted and enables better experimentation (see guide [here](#))

Features

- **Blending** - allows you to upload 2 or more images and combine for a desired result
- **Upscale** - increase the size from 512x512 to 1024x1024 without losing image quality
- **Parameters** - allows you to influence specific parts of your resulting output
 - *Aspect Ratios* - aspect ratio is the width-to-height ratio of an image
 - *Chaos* - how varied the initial image grids are (high = more, low = less)
 - Think how consistent or wild you want your output images to be
 - *No* - tells the AI what to NOT include in your output images
 - *Quality* - changes how much time is spent generating an image
 - Higher quality takes longer to product, and lower takes less time
 - Default value is 1 (and only accepts “.25”, “.5” and “1”)
 - *Repeat* - runs a Job multiple times for quicker testing of prompts
 - Best to combine with other features like “Chaos” or “Weird” to increase the pace of experimentation
 - *Seeds* - AI art generators use seed numbers to create a field of visual noise, like television static, as a starting point to generate the initial image grids
 - Seed numbers are generated randomly for each image
 - Can be specified with the “--seed” parameter
 - Using the same seed number and prompt will produce similar images
 - *Style* - fine-tunes the aesthetic of some Midjourney Model Versions
 - “--style raw” parameters reduces the influence of the default Midjourney aesthetic and works well for advanced users who want more control over their images or more photographic images
 - *Stylize* - influences how strongly the training of Midjourney is applied

- Midjourney been trained to produce images that favor artistic color, composition, and forms
 - Default value is 100
 - Accepts values between 0 and 1000
- *Stop* - finishes a Job partway through the process
 - Can create blurrier, less detailed results
 - Accepts values between 10 and 100 (default is 100 to finish the job)
 - Does NOT work while upscaling
- *Tile* - generates images that can be used as repeating tiles to create seamless patterns for fabrics, wallpapers and textures
- *Version* - other models can be used by adding the --version or --v parameter or by using the /settings command and selecting a model version
 - Midjourney releases new models to improve image outputs
 - Each model excels at making different types of output images
- *Video* - creates a short movie of your initial image grid being generated
 - React with the “📩” (envelope) emoji to have Midjourney DM you a link to the video
 - Only works on image grids, not upscales.
 - Works with Model Versions 5, 5.1, 5.2, and niji 5 and Legacy Model Versions 1, 2, 3, test, and testp.
- *Weird* - introduces quirky and offbeat qualities to your generated images
 - Accepts values 0 to 3000
 - Default value is 0
 - Highly experimental feature and may change over time
 - Compatible with Midjourney Model Versions 5, 5.1, and 5.2
 - Not fully compatible with seeds
- **Advanced Prompts** - allows you to customize your finished results
 - *Image Prompts* - share URLs to affect the style and content of your output
 - *Remix* - takes the general composition of your starting image and use it as part of the new job
 - *Multi Prompts* - considers two or more separate concepts individually using “:.” as a separator
 - Allows you to assign relative importance to parts of a prompt
 - *Pan* - allows you to expand the canvas of an image in a chosen direction without changing the content of the original image
 - Increases the image resolution beyond the maximum 1024px x 1024px dimensions in one direction
 - *Permutation Prompts* - allows you to quickly generate variations of a Prompt with a single “/imagine” command
 - Prompt Example:
 - “/imagine prompt a {red, green, yellow} ball” creates and processes three Jobs
 - “/imagine prompt a red ball”
 - “/imagine prompt a green ball”
 - “/imagine prompt a yellow ball”
 - Saves time instead of prompting them individually

- Zoom Out - allows you to extend the canvas of an upscaled image beyond its original boundaries without changing the content of the original image
 - The newly expanded canvas will be filled-in using guidance from the prompt and the original image
 - Think of this as Midjourney's version of the "Generative Fill" feature from Adobe Firefly

Use Cases

- Art
 - Create art that would otherwise take days & weeks within minutes or hours
 - Leverage any art style from any artist to use as inspiration for new pieces
- T-shirt Designs
 - Generate a design and then use a company to produce it on apparel
- Logos
 - Customize different looks for logos by varying your prompt to iterate

Tips

- For a complete overview with helpful tips, check out Midjourney's guide [here](#)

Alternatives

- [Clipdrop](#) (Freemium)
- [Adobe Firefly](#) (Freemium)
- [Playground AI](#) (Freemium)
- [Leonardo AI](#) (Freemium)



Microsoft Designer

Benefits

- Completely free to use and works as a sort of combination between AI art generator and Canva allowing you to create a variety of different types of designs
- Includes DALL-E 2.5 for generating images, which is allegedly better at creating realistic images with less deformity that you see with lower quality AI art generators
- Can upload your own images to combine the power of Microsoft Designer with custom watermarks, branding, etc.

Features

- **Animated Visuals** - add motion to your designs (animations, transitions, etc.)
- **Microsoft Edge Intergration** - access from the Edge sidebar on sites you visit
- **Resize Designs** - change the original design canvas size to a different size
- **Copy Suggestions** - recommendations for headlines, captions, hashtags, etc.
- **Automatic Saving** - your work is automatically saved as you make changes

Use Cases

- Art
 - Create art that would otherwise take days & weeks within minutes or hours
 - Leverage any art style from any artist to use as inspiration for new pieces
- T-shirt Designs
 - Generate a design and then use a company to produce it on apparel
- Logos
 - Customize different looks for logos by varying your prompt to iterate

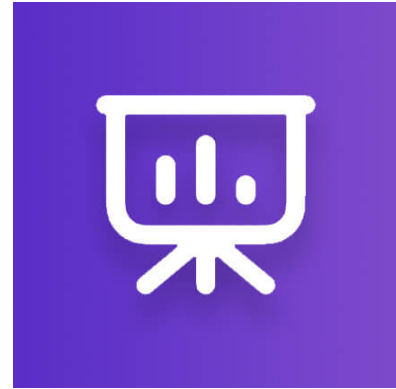
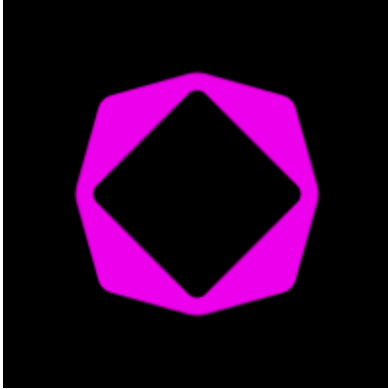
Tips

- Follow these steps to access Microsoft Designer:
 1. Download [Microsoft Edge](#) (Microsoft's browser)
 2. After downloading, open up [Microsoft Edge](#)
 3. Then navigate to [Microsoft Designer here](#)
 4. Click "**Get started for free**"
 5. Next, click "**Sign up or sign in with a Microsoft account**"

6. After signing in, you will be able to create designs
 - Ideal for those who want a free alternative to [Canva](#) while simultaneously leveraging the AI art capabilities of Microsoft
 - While image generations may not be as high quality as Midjourney or Leonardo, it's tough to beat "free" since many of the tools that use to be available without charging have now disappeared or become paid tools

Alternatives

- [Clipdrop](#) (Freemium)
- [Adobe Firefly](#) (Freemium)
- [Playground AI](#) (Freemium)
- [Leonardo AI](#) (Freemium)



AI Presentation Tools

To generate slideshows, pitch decks and other presentations

Gamma
Tome
SlidesAI



Gamma

Benefits

- Create presentations from scratch without all the tedious formatting work
- Useful for a variety of industries including sales, startups, small businesses & more
- Main use is presentations, but can also create documents & webpages

Features

- **AI Assistant** - helps you create starting point for presentations and add components
- **Outline** - provides bullet points that you can edit to customize your presentation
- **Toolbar** - formatting options that PowerPoint or Google Slides offers plus more
 - *Card Templates* - choose how each slide is aligned
 - *Text Formatting* - pick options like bullet lists, footnotes, code blocks, etc.
 - *Callout Blocks* - add warning boxes, success boxes, & other indicators
 - *Layout Options* - add tables, gallery of images, grids, etc.
 - *Visual Templates* - change visuals to timelines, boxes, team photos & more
 - *Add Images* - add your own images, GIPHYs, icons, etc.
 - *Embed Videos* - insert videos from your own computer, YouTube, Vimeo, etc.
 - *Embed Apps & Webpages* - add assets from Figma, PDFs, Google Drive, etc.
 - *Forms & Buttons* - include forms from 3rd parties (e.g. Airtable or Typeform)

Use Cases

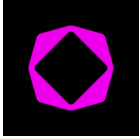
- Sales
 - Generate a pitch deck for a big meeting or client
 - Add internal assets along with the AI generated ones for efficiency
- Fundraising
 - Quickly put together a presentation for investors or board members
 - Multiple optimal features available (timeline, 3rd party integrations, etc.)
- Project
 - For school assignments that have quick deadlines
 - Can optimize for specific subjects

Tips

- You get 400 free credits when using the “Free” plan for AI presentations
 - Includes analytics, customizable themes, built-in templates & PDF export
- **PRO TIP:** you don't get charged for credits when exporting to PDF (at least as of now), so create a template that you can use interchangeably with your various needs so you can take full advantage of the free plan
 - For example, I created a pitch deck for social media and now I can simply substitute the assets for whichever brand or company I'm speaking with to have unlimited use out of that AI generated deck

Alternatives

- [Tome](#) (Freemium)
- [SlidesAI](#) (Freemium)



Tome

Benefits

- Create presentations from scratch without all the tedious formatting work
- Useful for a variety of industries including sales, startups, small businesses & more
- Main use is presentations, but can also create documents & webpages

Features

- **AI Assistant** - helps you create starting point for presentations and add components
- **Page Length** - adjust the amount of slides you want your presentation to be
- **Buttons** - has a number of options including 3rd party integrations
 - *Prompt Writing Tips* - helps you optimize your prompts for better results
 - *Record Narration* - add a voiceover to your presentation
 - *Set Theme* - change the look & style of the overall presentation
 - *Add Page* - insert a new additional page
 - *Add Table* - insert a table into your presentations
 - *Add Image/Video* - can upload images from computer or generate with AI
 - *Integrations* - include Looker, Miro, Twitter & other components as tiles

Use Cases

- Sales
 - Generate a pitch deck for a big meeting or client
 - Add internal assets along with the AI generated ones for efficiency
- Fundraising
 - Quickly put together a presentation for investors or board members
 - Multiple optimal features available (timeline, 3rd party integrations, etc.)
- Project
 - For school assignments that have quick deadlines
 - Can optimize for specific subjects

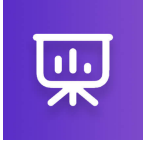
Tips

- You get 500 free credits when using the “Free” plan for AI presentations

- Unlike [Gamma](#), you can't export to PDF without a "Pro" account, so keep this in mind when using Tome

Alternatives

- [Gamma](#) (Freemium)
- [SlidesAI](#) (Freemium)



SlidesAI

Benefits

- Create presentations from scratch without all the tedious formatting work
- Useful for a variety of industries including sales, startups, small businesses & more
- Main use is presentations, but can also create documents & webpages

Features

- **Google Integration** - automatically use this as an “Extension” within “Google Slides”
- **Slides Prompt** - use at least 350 characters to describe presentation for best results
- **Length** - choose the number of slides you want the presentation to be
- **Themes** - allows you to customize the overall format of your presentation including:
 - Fonts
 - Font Sizes
 - Colors
- **Type** - can choose between “General”, “Conference”, “Educational” or “Sales”
- **Add Slides/Assets** - allows you to enable or disable adding a “Title” & “Thank You” slide, as well as add images automatically to your slides

Use Cases

- Sales
 - Generate a pitch deck for a big meeting or client
 - Add internal assets along with the AI generated ones for efficiency
- Fundraising
 - Quickly put together a presentation for investors or board members
 - Multiple optimal features available (timeline, 3rd party integrations, etc.)
- Project
 - For school assignments that have quick deadlines
 - Can optimize for specific subjects

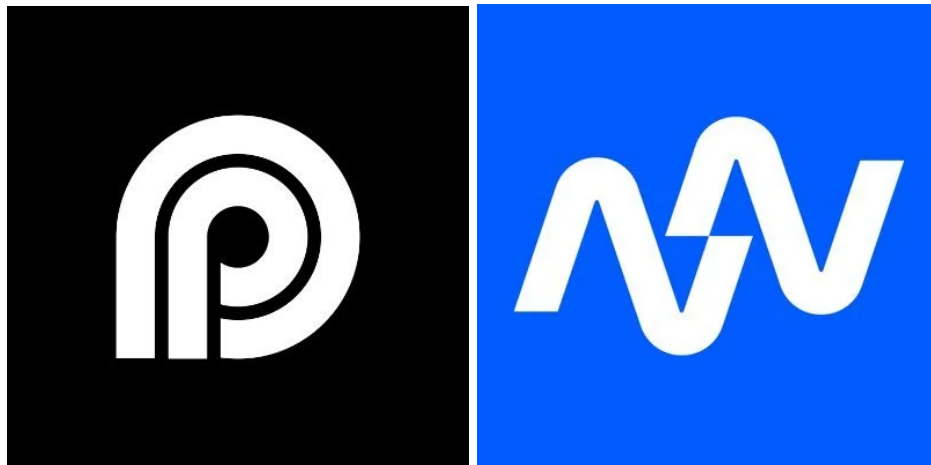
Tips

- You get 3 presentations for free every month with 2500 characters/presentation
- Automatically integrated with Google Slides

- Pull up a new presentation within “Google Slides”
- Click “Extensions” from the toolbar at the top
- Hover over “SlidesAI.io - Create Slides with AI” and choose what you need

Alternatives

- [Gamma](#) (Freemium)
- [Tome](#) (Freemium)



AI Video Tools

To streamline editing and extract short-form clips out of long-form content

OpusClip
Autopod



OpusClip

Benefits

- Take any long-form content (e.g. podcasts) and paste the YouTube link or upload in the “Get Free Clips” section to get multiple short-form clips
- Short-form clips will be formatted, edited and subtitles added to allow for posting to YouTube Shorts, Instagram/Facebook Reels, TikTok, etc.
- Product includes a subtitle/format editor to make revisions and change placement of video and recently launched a video editor to allow you to edit clips manually

Features

- **Long-form to Short-form** - transforms long video into multiple short-form clips
- **Subtitles & Face Tracker** - adds subtitles and tracks face to optimize clips
- **Manual Editor** - you can now edit the clips, format & subtitles to improve quality

Use Cases

- Podcasts
 - Take a 1 hour or longer podcast and generate multiple clips
 - Allows for more chances to get discovered when crossposted on social media
- Keynote Speeches
 - Will find the optimal points in a keynote to clip into short-form
 - Allows for more opportunities to reinforce key points from the event
- Highlights
 - Take highlights of a recorded event with multiple activities
 - Note: Opus is best for talking videos and may struggle with other formats

Tips

- You get 2 hours worth of credits for the free plan before needing to upgrade
 - **PRO TIP:** Edit your video to remove all the uninteresting parts so that when you run it through OpusClip, you reduce the amount of poor quality clips generated and increase the value out of using the tool
- Highlight text that “adds” to the story or progression of the video

- For instance, if they're talking about making money and scaling a business such as: "we had no idea we would reach \$1,000,000 in revenue"
 - Use a new text color for "reach", "\$1,000,000" or "revenue" (yellow for emphasis or green to symbolize money) to keep viewers' attention

Alternatives

- [Munch](#) (Freemium)
- [Vidyo](#) (Freemium)
- [Autopod](#) (Freemium)



Autopod

Benefits

- Can **cut down any podcast episode** so that all of the silence and “dead space” during the episode is automatically removed
- **Saves hundreds to thousands of dollars each month** instead of hiring an editor that will need to manually cut every episode
- Works with up to 10 different cameras and microphones and **integrates directly within Adobe Premiere Pro (add-on)** to easily access within editing workflow

Features

- **Multi-Camera Editor** - works for up to 10 cameras and microphones
- **Social Clip Creator** - automatically resizes and adds edits to clips for social media
- **Jump Cut Editor** - automatically creates jump cuts in footage based on silence

Use Cases

- Podcasts
 - Take a 1 hour or longer podcast, edit silence, and generate multiple clips
 - Allows for more chances to get discovered when crossposted on social media
- Keynote Speeches
 - Since Autopod can edit with multiple microphones and cameras, events like Fireside Chats can be taken to the next level without intense manual editing
- Highlights
 - Take highlights of a recorded event with multiple activities

Tips

- You get a 30-day free trial with Autopod so you can make sure it fits for your needs
 - **PRO TIP:** use Autopod to edit your podcast episode down to only the places where people are talking and then you can use an AI tool like [OpusClip](#) or [Munch](#) mentioned earlier in this guide to optimize your clips so you don't include any of the “useless” footage BEFORE editing them out with Autopod
- Utilize the free trial to understand how Autopod works, and then start utilizing free trials of [OpusClip](#) and [Munch](#)

- This will allow you to see if the workflow(s) you come up with make sense for your type of podcast and the content you'll produce BEFORE needing to pay for anything

Alternatives

- [Munch](#) (Freemium)
- [Vidyo](#) (Freemium)
- [OpusClip](#) (Freemium)



AI Databases

To discover AI tools across a variety of categories & platforms

Future Tools
Futurepedia
Phygital Plus
AI Library
GPTE
Hugging Face
Meta AI

AI Tool Databases



[Future Tools](#) - 1500+ AI tools with variety of categories (Free, GitHub, etc.)



[Futurepedia](#) - 50+ categories of tools updated daily with filtering



[Phygital Plus](#) - 1300+ AI tools searchable by task, input, outputs & more



[AI Library](#) - 1000+ AI tools searchable by categories



[GPTE](#) - 2000+ AI tools organized by use cases



[Hugging Face](#) - library of large language models including LLM [leaderboard](#)



[Meta AI](#) - open-source data sets, framework and tools from Meta

NOTE: I am NOT endorsed or sponsored by any of these websites nor using affiliate links. I am sharing the free knowledge & lowest cost tools that I was able to track down. It is your choice of whether to purchase any goods or services they offer.

Frequently Asked Questions

Who are you?



Hey friends! My name is David Paykin. I am on a mission to put out practical content that helps people work towards their life goals & land opportunities. Specifically, I've covered everything from job search strategies to learning how to use artificial intelligence.

I cover these in more detail here:

[YouTube](#)
[Instagram](#)
[TikTok](#)
[LinkedIn](#)
[Twitter](#)

Why would you possibly make this FREE?!

For the past couple of years, I have been one of the only people in the job search, career and educational industry to give out every resource on a large scale for FREE ([resume template](#), [interview guide](#), [cover letter template](#), [networking templates](#), [career resource database](#), & [more](#)).

I was YOU at one point in my life and wish I would've had someone to provide guidance & resources WITHOUT trying to sell me something. I do this because I want to try to do something meaningful to help folks of any and every background, regardless of race, ethnicity, or financial status. Every other person I see tries to use their content and resources to get you into some sort of sales funnel for their products or services.

I started putting out content and building our community for the sole purpose of helping as many people as I possibly can. That means no courses, no coaching, no masterclasses, no e-books, no merchandise. I have NEVER even taken a sponsorship or partnered with a brand because I want to make sure you all are getting resources that I actually believe in.

If you haven't been taking money from community members, how are you surviving?

I have been living off my life savings for the past 3 years (as well as briefly helped build social media presence for a small startup in generative AI). Unemployment helped with covering some expenses, but those benefits will soon run out again and I'm relying solely on what I saved again. I've been living off of just the bare necessities to try and save as much as possible while continuing to put out all the resources and information I possibly can for our community!

I say this all to be transparent about my current situation at the time of releasing this AI guide (just like I had been when I released the [free interview guide](#)). I have some ideas for how I can support myself long-term by utilizing my knowledge and skills in AI to help businesses leverage their capabilities. Again, my mission is to **NEVER** charge our individual community members. I'm hoping to find ways I can use my current skills to ensure that I continue coming out with content/resources for you all and keeping it freely available to everyone as I've been doing over the past years!

Do you have more free resources I can use?

Absolutely! Here are some of the most popular resources that have help our community land job across numerous industries including at companies like Google, Amazon, Facebook, TikTok, Tesla, Disney, Apple, Dell, Microsoft, Deloitte, EY, PwC, KPMG, JPMorgan Chase, Goldman Sachs, Morgan Stanley, & Credit Suisse among many others:

[Free Resume Template](#) - helps you with using a format to get past ATS ([Applicant Tracking Systems](#)) that companies use to filter out candidates

[Free Interview Guide](#) - covers every question with templates and examples to help you ace the most commonly asked ones, as well as technical & case interviews

[Free Cover Letter Template](#) - includes how to structure your cover letter, a template, and tips along with mistakes to avoid!

[Free Networking Templates](#) - covers what to say in your email and subject line, LinkedIn connection request templates, Questions to Ask after them after connecting, and "Thank You" templates for job interviews

[Free Job Tracker Template](#) - keep track of applications, networking and interview progress

How can I get more involved with the community?

I've created 2 private communities for us all to be able to connect. One of them allows us as to connect with each other so you can interact with likeminded individuals who want to help each other:

Our 30,000+ member [networking group on LinkedIn](#) allows you to meet thousands of hiring managers, recruiters, students & fellow professionals in your industry.

And signing up for our free exclusive newsletter gives you the latest updates, resources & tutorials directly from me as I test them: <https://davidp.substack.com/>

This will allow you to stay ahead in what's coming next in your career, artificial intelligence & broadly across the world with my perspective.

DISCLAIMER

Last updated September 6, 2023

WEBSITE DISCLAIMER

The information provided by PaykinX LLC (“we,” “us”, or “our”) on https://docs.google.com/document/d/11bSeBI5yqhkGYsGjQPzps1W_uti3r9hrCnkmefZB SGk/edit (the “Site”) is for general informational purposes only and not intended as professional advice. All information is provided “as is” without any warranties. All information on the Site is provided in good faith, however we make no representation or warranty of any kind, express or implied, regarding the accuracy, adequacy, validity, reliability, availability or completeness of any information on the Site. The author and publisher are not liable for any damages arising from use of the guide. The reader assumes full responsibility for using the information in the guide. The content on the Site is the intellectual property of PaykinX LLC and may not be redistributed or used for commercial purposes without explicit permission. We disclaim any implied warranties of merchantability, fitness for a particular purpose, and non-infringement.

UNDER NO CIRCUMSTANCE SHALL WE HAVE ANY LIABILITY TO YOU FOR ANY LOSS OR DAMAGE OF ANY KIND INCURRED AS A RESULT OF THE USE OF THE SITE OR RELIANCE ON ANY INFORMATION PROVIDED ON THE SITE. YOUR USE OF THE SITE AND YOUR RELIANCE ON ANY INFORMATION ON THE SITE IS SOLELY AT YOUR OWN RISK.

PUBLIC CONTENT DISCLAIMER

The Site may contain examples, use cases, and references to publicly available third party content, including from social media sites. This third party content is used for illustrative and educational purposes only. The third party content remains the property of the respective third party owner and its use is not intended to suggest that the owner endorses this guide.

EXTERNAL LINKS DISCLAIMER

The Site may contain (or you may be sent through the Site) links to other websites or content belonging to or originating from third parties or links to websites and features in banners or other advertising. Such external links are not investigated, monitored, or checked for accuracy, adequacy, validity, reliability, availability or completeness by us. We do not endorse any third-party sites or products mentioned on the Site.

WE DO NOT WARRANT, ENDORSE, GUARANTEE, OR ASSUME RESPONSIBILITY FOR THE ACCURACY OR RELIABILITY OF ANY INFORMATION OFFERED BY THIRD-PARTY WEBSITES LINKED THROUGH THE SITE OR ANY WEBSITE OR FEATURE LINKED IN ANY BANNER OR OTHER ADVERTISING. WE WILL NOT BE A PARTY TO OR IN ANY WAY

BE RESPONSIBLE FOR MONITORING ANY TRANSACTION BETWEEN YOU AND THIRD-PARTY PROVIDERS OF PRODUCTS OR SERVICES.

PROFESSIONAL DISCLAIMER

The Site cannot and does not contain professional advice. The information is provided for general informational and educational purposes only and is not a substitute for professional advice. Accordingly, before taking any actions based upon such information, we encourage you to consult primary sources, conduct your own research, and seek independent professional counsel. We do not provide any kind of professional advice.

THE USE OR RELIANCE OF ANY INFORMATION CONTAINED ON THIS SITE IS SOLELY AT YOUR OWN RISK.

TESTIMONIALS DISCLAIMER

The Site may contain testimonials by users of our products and/or services. These testimonials reflect the real-life experiences and opinions of such users. However, the experiences are personal to those particular users, and may not necessarily be representative of all users of our products and/or services. We do not claim, and you should not assume, that all users will have the same experiences. YOUR INDIVIDUAL RESULTS MAY VARY. The testimonials on the Site are submitted in various forms such as text, audio and/or video, and are reviewed by us before being posted. They appear on the Site verbatim as given by the users, except for the correction of grammar or typing errors. Some testimonials may have been shortened for the sake of brevity where the full testimonial contained extraneous information not relevant to the general public. All testimonials are provided voluntarily and without compensation. The views and opinions contained in the testimonials belong solely to the individual user and do not reflect our views and opinions. We are not affiliated with users who provide testimonials, and users are not paid or otherwise compensated for their testimonials.

Technology changes rapidly. All information on the Site may become outdated without notice. You are advised to verify information before relying on it. Following any examples or suggestions does not guarantee any specific results. You must always use your own judgment and seek professional advice when required.

No information on this Site should be construed as legal advice. You assume full responsibility and risk for your use or reliance on any of the information provided on this Site. You understand and agree that the author, publisher, and Site owner shall not be held liable or responsible for any claim, loss, or damage resulting from your use or reliance upon any information contained on the Site.

The Site and the information contained therein is intended for lawful use only. Any other use may be considered unlawful and the author, publisher, and Site owner shall not be held liable for such use. You agree to indemnify and hold the author, publisher, and Site owner harmless from any claims or damages resulting from your unlawful use.

If any portion of this disclaimer is deemed unlawful or unenforceable, the remaining provisions will remain in full force and effect. We reserve the right to modify this disclaimer at any time. Any changes to this disclaimer will be posted here, and it is your responsibility to review this disclaimer frequently.

AI Guide

(c)2023, PaykinX LLC, and its affiliates and assigns and licensors

All rights reserved

No claim to copyright is made for original U.S. Government Works.