Learn Python Game Development with ChatGPT

Techniques for creating engaging games with generative AI

Micheal Lanham

www.bpbonline.com

First Edition 2024

ISBN: 978-93-55516-435

LIMITS OF LIABILITY AND DISCLAIMER OF WARRANTY

The information contained in this book is true to correct and the best of author's and publisher's knowledge. The author has made every effort to ensure the accuracy of these publications, but publisher cannot be held responsible for any loss or damage arising from any information in this book.

All trademarks referred to in the book are acknowledged as properties of their respective owners but BPB Publications cannot guarantee the accuracy of this information.

Dedicated to

For those who dream of a future with AI, may your imagination be boundless, your innovations endless, and your journey as remarkable as mine has been. I dedicate this book to all us dreamers

About the Author

Micheal Lanham is a recognized software and technology innovator with over 25 years of experience. He has significantly contributed to game development, graphics, web and desktop applications, engineering, AI, GIS, and ML. Micheal is mainly known for pioneering the integration of evolutionary computation with deep learning models. Starting his career at the turn of the millennium, he focused on neural networks and evolutionary algorithms in game development. His expertise has established him as a developer, consultant, and author of numerous projects. Currently, Micheal resides in Calgary, Alberta, with his large family, who enjoys cooking for him.

Acknowledgement

I always thank my family, friends, and coworkers for supporting and acknowledging my writing. Their support drives me to write more innovative and exciting titles like this one. But my genuine support comes from my wife, Rhonda, for always being by my side. Also, a special acknowledgment to my kids, Ava and Aaron Thiessen, who assisted me in reviewing, playing games and just fun to pester.

I especially want to thank BPB Publications for their assistance and nudging to complete this book. Building a book driven by AI was a unique journey for everyone, and the editors, reviewers, and technical experts all successfully took to the challenge.

Of course, I also need to acknowledge the tireless work of those in the field of AI. Even being as close as I was to the AI field just a few years ago, I would have never thought I could collaborate with AI to write a book. Thanks, ChatGPT and the AI revolution.

Preface

I have worked full-time as a game and graphics developer several times throughout my career and even tried my hand at founding a studio. The game studio fell flat, not because I couldn't write games but because I couldn't find the artists and coders who could help me. But that has all changed with the advent of AI.

A big part of what I wanted to showcase in this book is how a game developer can create games independently or with a small team. I still genuinely believe that gaming needs to return to the small, lean, agile development team—teams that can push the edge of game development and how we think of games.

I also wanted to encourage people developing games to be adventurous. AI now gives us the ability to push ourselves beyond our capabilities. If you don't know how to make a game, or build a feature, or create an animation, ask ChatGPT or another AI service.

To me, AI can empower all of us to be the best we can be. I learned this through my journey writing this book. AI is a tool. In competent hands, it is powerful, but it is still a tool. That may change, but for now, be better with AI.

General introduction to the book followed by,

Chapter 1: ChatGPT and the Magic of Prompt Engineering – This chapter introduces readers to ChatGPT and the concept of prompt engineering. It explains how to use ChatGPT effectively to generate Python code, particularly in game development. The chapter covers constructing effective prompts, understanding ChatGPT's decision-making process, and leveraging system prompting. Readers will engage in practical exercises to create simple Python games, gaining hands-on experience using ChatGPT to automate coding tasks.

Chapter 2: Text Adventure: Entering the Enchanted Realm – In this chapter, readers dive into the world of text-based games by creating a game like the classic Zork. The focus is on fundamental game development concepts such as managing player input, building game worlds, and handling items and non-player characters (NPCs). By the end of the chapter, readers will have created a functional text adventure game and learned about game loops, data management, and narrative design.

Chapter 3: The AI Chronicles: Text Game Evolution – Building on the previous chapter, this chapter introduces visual elements and more complex game mechanics to the text adventure game. Readers will learn to use tools like Streamlit for creating web-based

interfaces and Stable Diffusion for generating images. Additionally, the chapter covers integrating SQLite databases to manage game state and progress. By the end of the chapter, readers will have enhanced their text game with visuals and a persistent game world.

Chapter 4: 2D Platformer: Leap into Pixelated Fun! – This chapter transitions readers from text-based games to 2D platformers. It covers the basics of building a platformer, including player controls, movement mechanics, and level design. Readers will also learn about sprites and animation to bring their game characters and environments to life. The chapter introduces AI tools for generating game assets, helping readers create visually appealing games. By the end, readers will have a functional 2D platformer game.

Chapter 5: Bot Brawls: AI Opponents Enter the Arena – This chapter teaches readers how to create a fighting game with AI-controlled opponents. It covers the development of a Street Fighter clone, focusing on integrating behavioral AI for dynamic and challenging gameplay. Topics include behavior-driven AI, managing AI difficulty levels, and restyling the game with AI-generated assets. By the end of the chapter, readers will have created a game where players can battle against intelligent AI opponents.

Chapter 6: Revving up: Cars, Ramps, and Pymunk – This chapter explores game physics through the creation of a car jump challenge game. Readers will learn to use the Pymunk physics library to simulate realistic motion and rotation. The chapter covers building a game engine that incorporates physics-based interactions, including adding particle systems for enhanced visual effects. By the end, readers will have a deeper understanding of game physics and a completed car physics game.

Chapter 7: Building Isometric Worlds – In this chapter, readers explore creating 2.5D isometric games. They also explain the complexities of isometric perspectives, including world-to-screen space conversion and designing isometric tile maps. Readers will learn to add detail and variation to their maps with overlays and integrate user interfaces and menus within the isometric world. By the end of the chapter, readers will be able to create intricate and visually engaging isometric game worlds.

Chapter 8: Leveling up with GPT Agents and AutoGen – This chapter focuses on integrating intelligent GPT powered agents into games using AutoGen. Readers will learn how to set up an OpenAI account and configure AutoGen for game development. The chapter covers the fundamentals of GPT agents and demonstrates how to use them to enhance game functionality, such as creating assistant agents for various game tasks. By the end, readers will have a deeper understanding of AI integration in games and practical experience with AutoGen.

Chapter 9: Building a 3D First-Person Shooter – This chapter introduces readers to the development of 3D first-person shooter (FPS) games. It covers implementing 3D player controls, camera movement, and understanding 3D coordinate systems. The chapter also discusses using materials and textures to create realistic environments to enhance gameplay. By the end, readers will have a foundational understanding of 3D game development and a basic FPS game.

Chapter 10: Games That Respond to Your Voice – This chapter explores the creation of voice-controlled games. Readers will learn to implement speech recognition systems and text-to-speech functionalities using tools like ElevenLabs and ChatGPT. The chapter covers building a voice-driven game where players interact with the game world through spoken commands. By the end, readers will have created a unique game that leverages voice input for an immersive experience.

Chapter 11: The Future Beckons: Developing GPT Games – The final chapter looks to the future of AI in game development. It covers creating games with OpenAI GPT assistants, focusing on designing engaging and fun GPT-based games. The chapter discusses the potential future trends in AI-driven game development, including generative AI for creating game content. By the end, readers will understand the cutting-edge possibilities in game development and ideas for future projects.

Code Bundle and Coloured Images

Please follow the link to download the
Code Bundle and the ***Coloured Images*** of the book:

https://rebrand.ly/88ts118

The code bundle for the book is also hosted on GitHub at
https://github.com/bpbpublications/Learn-Python-Game-Development-with-ChatGPT.
In case there's an update to the code, it will be updated on the existing GitHub repository.

We have code bundles from our rich catalogue of books and videos available at **https://github.com/bpbpublications**. Check them out!

Errata

We take immense pride in our work at BPB Publications and follow best practices to ensure the accuracy of our content to provide with an indulging reading experience to our subscribers. Our readers are our mirrors, and we use their inputs to reflect and improve upon human errors, if any, that may have occurred during the publishing processes involved. To let us maintain the quality and help us reach out to any readers who might be having difficulties due to any unforeseen errors, please write to us at :

errata@bpbonline.com

Your support, suggestions and feedbacks are highly appreciated by the BPB Publications' Family.

Piracy

If you come across any illegal copies of our works in any form on the internet, we would be grateful if you would provide us with the location address or website name. Please contact us at **business@bpbonline.com** with a link to the material.

If you are interested in becoming an author

If there is a topic that you have expertise in, and you are interested in either writing or contributing to a book, please visit **www.bpbonline.com**. We have worked with thousands of developers and tech professionals, just like you, to help them share their insights with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Reviews

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions. We at BPB can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about BPB, please visit **www.bpbonline.com**.

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Table of Contents

Chapter 1
ChatGPT and the Magic of Prompt Engineering

Introduction

This book will take you on a journey to learn game programming-starting, from text games to 2D platformers, 3D games, and even the future of speech-based games. There will be a lot of information to cover, but thankfully we will employ an AI tool called **ChatGPT**.

In this chapter, we will explore how ChatGPT can help us learn to build games or anything beyond the horizon. We will explore how to effectively use ChatGPT, from constructing effective prompts with prompt engineering to exploring the reasoning behind the AI's decisions.

Structure

This chapter explores the following topics:

- Entering the realm of ChatGPT
- Exploring ideas to interactions
- Unlocking potential with system prompting
- Engaging retrieval augmented generation with EasyCode

Objectives

Our objective for this chapter is to learn how to use ChatGPT effectively. While our book focuses on learning to make games, the skills you will learn may be applied to numerous problems. Once you start using ChatGPT, you most likely will want to keep using it.

Entering the realm of ChatGPT

In late 2022, the world was changed forever with the release of **ChatGPT 3.5**, a chat-based AI tool that allowed users to ask questions and receive some insightful answers. Since these answers were shockingly high in quality, OpenAI, the industry that made ChatGPT, reflected upon its capabilities of what it could build.

Trivia time: ChatGPT was the first application to break 100 million users in a week. Do you know what the next app to break that record was?

Before we get into using ChatGPT let us go over some technical specifications. ChatGPT is a deep learning model comprised of billions of parameters using a transformer architecture. It is trained using an updated loss mechanism called **Reinforcement Learning from Human Feedback** (**RLHF**). Initially, the GPT model was trained on billions of documents, then fine-tuned on millions, and finally optimized with RLHF on thousands.

The core GPT model ChatGPT is built on is a chat completion model. That means it only tries to complete the content/text it is given. A GPT model is designed to consume text as tokens. It uses an internal probability engine to reply with the next most likely token. This type of GPT model, designed to respond with text, is called a completion model.

ChatGPT, however, is trained a few steps above a base GPT model. *Figure 1.1* shows the progression of GPT model training and what makes ChatGPT special:

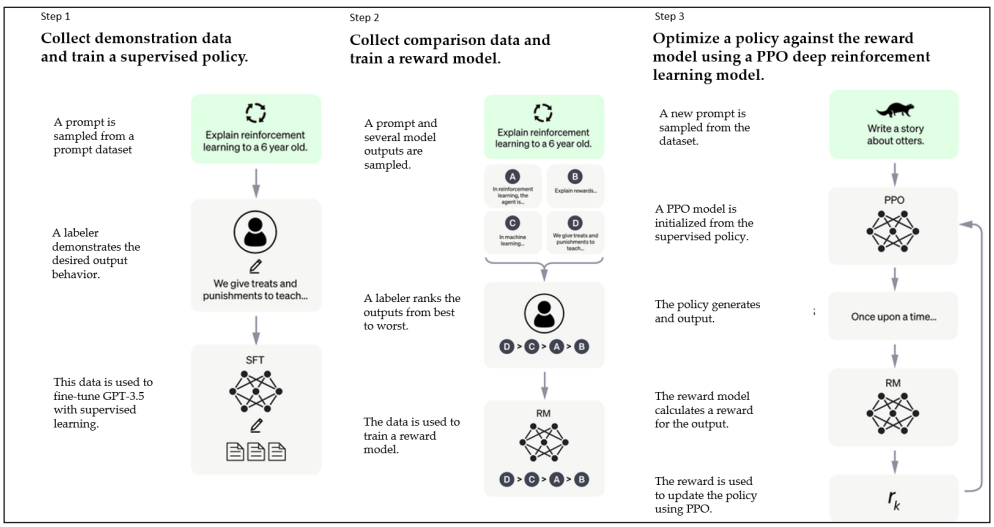

***Figure 1.1**: Training ChatGPT*

Figure 1.1 illustrates how ChatGPT is trained from the initial supervision of human feedback for building a reward model. It finally uses reinforcement learning to optimize the model. We only need to fully understand the process to identify how this form of training makes ChatGPT so unique and powerful. The power of ChatGPT comes from fine-tuned labeling provided by human feedback augmented with a reward model.

This extra training transforms the GPT model from a generalist to a model that can fine-tune its responses through continued prompting or chatting. But what makes this transformation interesting and so special is that it can extend this ability to identify the steps or process of a conversation chain. And, if we extend this further, ChatGPT can follow a chain of thoughts or reasoning processes.

We will take advantage of ChatGPT's ability to reason throughout this book. For now, let us get some hands-on experience using ChatGPT, partaking in a couple of quick experiments in the exercise below:

Exercise 1.1: **Conversing with ChatGPT**

1. Open your web browser to: **https://chat.openai.com/**
2. Go ahead and either log in or sign up and log in. *Figure 1.2* shows the ChatGPT login screen:

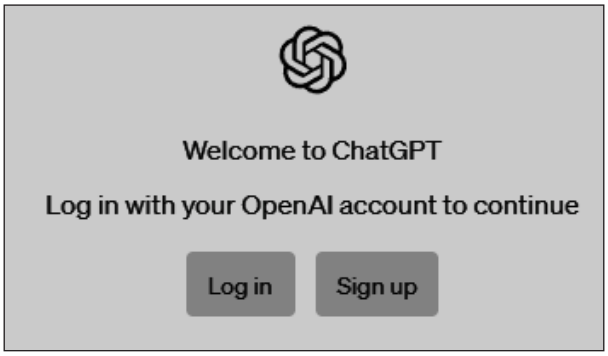

***Figure 1.2**: ChatGPT login screen*

3. Our first prompt will be the basic *write me a game* and assess its response.
4. In the prompt input, enter the following text and hit enter, as shown in *Figure 1.3*.
   Prompt: `Please write me a snake game in Python using PyGame.`

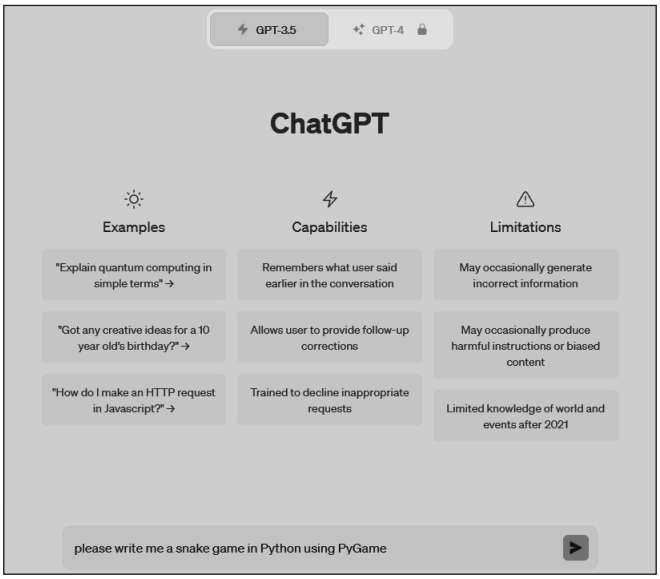

***Figure 1.3**: ChatGPT interface*

5. After some time, depending on your version, ChatGPT will respond with some Python code explaining how to use it, as shown in *Figure 1.4*:

Default (GPT-3.5)

```
GRID_SIZE = 20
GRID_WIDTH, GRID_HEIGHT = WIDTH // GRID_SIZE, HEIGHT // GRID_SIZE
FPS = 10
WHITE = (255, 255, 255)
RED = (255, 0, 0)
GREEN = (0, 255, 0)

# Initialize pygame
pygame.init()
win = pygame.display.set_mode((WIDTH, HEIGHT))
pygame.display.set_caption("Snake Game")

# Snake class
class Snake:
    def __init__(self):
        self.body = [(GRID_WIDTH // 2, GRID_HEIGHT // 2)]
        self.direction = (1, 0)

    def move(self):
        head = self.body[0]
        x, y = head[0] + self.direction[0], head[1] + self.direction[1]
        self.body.insert(0, (x, y))
```

***Figure 1.4**: ChatGPT outputs a snake game*

6. Next, we will take the produced code and run it on your desktop. This book assumes you have some experience writing Python code, so if you are unsure how to set up your environment, consult *Appendix A*.

7. First, install the dependencies into your virtual Python environment:

```
1. pip install pygame
```

8. Then, copy the code from ChatGPT using the **copy code** button on the top right and paste it into a new file called **`snake_game.py`**. Make sure to save the file after pasting.

9. Run the file from the console using Python:

```
1. python snake_game.py
```

10. You should see a PyGame window open, as shown in *Figure 1.5* and the snake game running. Try and use the *up/down/left/right arrows* to control the snake.

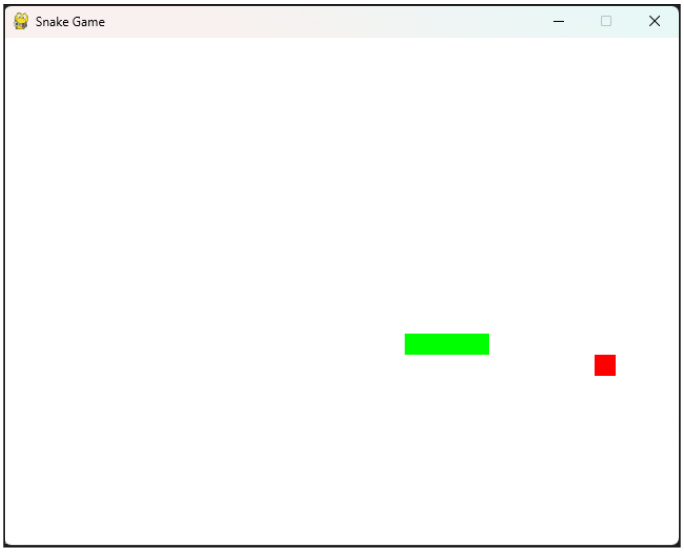

***Figure 1.5**: Snake game running*

What makes ChatGPT great is that this exercise likely worked for most readers but not all. If your code does not work the first time, go back and try again by opening a new chat prompt and entering the prompt we used earlier.

Not all the responses you receive will match what you see in the book. This is because, by its nature, ChatGPT produces variable output. As this book uses ChatGPT for most exercises, you may see wildly different results if you use a different version or a completely different LLM.

Indeed, you can go back and perform this same exercise multiple times and it will likely produce very different versions of that snake game. Is this something we need to worry about? How will we fix or address this concern?

Well, it all comes down to the prompt and what we ask and want from ChatGPT. As we will see in the next section, it is important to plan how and what we ask of ChatGPT.