

Report of Deep Learning for Natural Language Processing

JiaXiang Yu

Zy2403124@buaa.edu.cn

Abstract

本项目旨在探索如何借助 ChatGPT 实现一个基于 HTML5 Canvas 的网页小游戏——《合成大西瓜》自定义版的开发流程。通过与 AI 助手多轮交互，项目从最初的结构构建到逻辑实现、样式设计，再到功能完善与调试修复，完整实现了一个可交互、可定制、可扩展的游戏系统。游戏支持图片自定义、球体物理下落与碰撞、合成机制、得分系统、失败判定与重新开始等核心功能。开发过程中，ChatGPT 不仅提供了可运行的代码模板，还帮助分析报错、优化结构与提升用户体验，显著降低了开发门槛，提高了开发效率。最终成果展示了 AI 在前端游戏开发中的实用价值，并验证了其在代码生成、逻辑推理、交互设计等方面的综合能力。

Introduction

“合成大西瓜”是一款经典休闲类合成小游戏，玩家通过将相同等级的水果进行合并，不断获得更高等级的水果，最终合成为一个巨大的西瓜。其简单直观玩法、轻松的动画反馈和高度的休闲娱乐属性，使其在多个平台上获得了极高的用户参与度。

传统游戏开发过程中需要大量的时间投入于结构设计、逻辑实现、调试优化等环节，特别对于非专业程序员而言门槛较高。引入 ChatGPT 能够帮助快速理解复杂结构、生成可运行代码并实时优化开发过程。

本项目旨在通过网页技术实现一个可自定义图片资源的“合成大西瓜”类小游戏。在传统网页开发基础上，尝试引入人工智能（AI）辅助工具——ChatGPT，辅助完成整个开发过程，探索 AI 在代码生成、结构设计、逻辑推导和调试中的作用。

Methodology

本文有如下任务：

- 实现一个完整可运行的“合成大西瓜”网页小游戏；

- 支持用户自定义所有合成元素图像；
- 实现重力、碰撞、反弹、合成等物理机制；
- 支持点击生成、实时合并、得分系统；
- 加入失败判定（球体触顶）与重新开始功能；
- 显示下一个即将落下球体的预览图像。

Step1: 项目初始化与结构设计

我首先向 ChatGPT 提出“创建一个网页游戏的 HTML 文件，支持图片球体的下落”，AI 立刻生成了完整 HTML 框架，包括：

- `<canvas>` 区域用于渲染游戏画面；
- 嵌入式 CSS 控制画布样式和边框；
- 提供了 JavaScript 脚本框架用于绘制和更新游戏画面。

Step2: 图片加载与球体结构定义

我希望每一个球可以加载一张自定义图片并以圆形显示。ChatGPT 提出预加载图片数组，并根据等级关联不同半径与图像。

- 提供 `Ball` 类结构，封装图片、半径、速度、位置、绘图方法等；
- 使用 `ctx.arc()` 实现圆形裁剪，再用 `drawImage()` 实现图像填充。

Step3: 球体的生成与合并

我提到想从用户点击位置开始生成球体，ChatGPT 给出 `canvas.addEventListener("click",...)` 监听逻辑，并将点击坐标转为球体生成位置。此时，重力、反弹与边缘限制一并集成，生成的球自动下落并受物理限制。为了实现“两个相同球合并生成更大球”的功能，我请求 ChatGPT 实现合并逻辑。

- 它给出 `mergeBalls()` 函数：遍历所有球体，对每对球判断类型与距离是否满足合并条件；
- 合并后新球体继承前者位置、速度，并升级为更高等级；
- 原球体被移除，新球加入，分数随之更新。

Step4: 添加碰撞检测与物理响应

为避免球体穿透、堆叠问题，我提出“球与球之间应产生物理碰撞”，ChatGPT 提供了 `resolveCollisions()` 物理函数：

- 检测球与球之间的重叠距离；

- 计算碰撞方向角度，修正位置；
- 模拟动量守恒调整速度，提升真实感。

Step5: 球体预览与重新开始

我希望在点击前显示“下一个球体”图像，ChatGPT 迅速建议：

- 使用 `div#nextBallPreview` 预览区域；
- 每次生成球后更新预览图片；
- 使用 `appendChild(img)` 插入图片至 DOM。

为了增加挑战性，我要求“球体触顶游戏结束”，ChatGPT 在 `Ball.update()` 函数中加入顶部检测逻辑。

- 显示“游戏结束”提示语；
- 加入按钮控制 `resetGame()` 函数，重新初始化所有球体与得分。

Experimental Studies

最终版本游戏具备以下完整特性：

- 玩家点击任意位置生成球体；
- 球体自动下落，受重力与反弹影响；
- 不同等级球具备不同半径与图像。
- 相同等级球体碰撞时自动合并；
- 合并后球体等级+1，获得更高分；
- 球体之间具备碰撞反应，防止重叠。
- 实时显示得分，并根据合成等级更新；
- 提示下一个即将下落球体图像；
- 球体触碰顶部画布后游戏失败；
- 支持重新开始，重置所有状态。
- 所有图像均可替换，支持自定义主题（如水果、表情包、人物等）；
- 支持拓展更高等级图像与得分机制；
- 可改为键盘操作、增加道具、设定关卡等。。

Conclusion

在本项目中，AI 辅助开发让我深刻体会到智能工具对开发者的赋能作用。

通过 ChatGPT，我能够快速理解并实现诸如碰撞检测、物理反弹、图像裁剪等原

本需要深入学习的复杂算法，极大地降低了开发门槛。同时，AI 的高效响应显著提升了开发效率，无需频繁查阅文档或反复调试小错误，它能迅速定位 bug 并给出修改建议，大大缩短了开发周期。此外，在项目结构设计、命名规范和性能优化等方面，AI 也提供了专业建议，使项目逻辑更加清晰、代码更具可维护性。更重要的是，AI 激发了我的创造力，让我从原本可能止步于简单交互的想法不断拓展出新的功能设想，并能快速实现，使整个开发过程充满了探索与突破的乐趣。