



浙江大学  
ZHEJIANG UNIVERSITY

浙江大学软件学院优秀大学生夏令营  
项目报告

姓 名 \_\_\_\_\_ 张宏远 \_\_\_\_\_

选 题 \_\_\_\_\_ 任务 3+子任务 2 \_\_\_\_\_

## 目录

一、子任务 2.....	3
2.1 前端优化.....	3
2.1.1 项目场景优化.....	3
2.1.2 项目人物优化.....	4
2.1.3 项目 UI 优化.....	5
2.2 大模型优化.....	6
2.3 后端优化.....	9
2.3.1 上下文相关优化.....	9
2.3.2 模型类型切换.....	10
2.3.3 提示词模板.....	11
2.3.4 图像生成功能.....	12
二、未来展望.....	13

# 一、子任务 2

任务要求：往大模型开发技术方向深入，可以考虑利用大模型开发框架（如 langchain）进行大模型 API 的开发，同时可以考虑使用上 RAG, Agent 这些技术激发大模型的潜能，使得应用更加智能化。往后端开发深入，设计更加复杂的后端架构，可以考虑采取微服务，DDD（Domain-Driven Design）等开发方法，并针对实际场景需求增添一些中间件，丰富后端功能。往前端开发深入，可以考虑为应用开发跨平台的前端（支持 Win, IOS, Android, Mac 等），也可考虑优化前端页面，使得前端更加炫酷，优美。

项目演示视频：[bilibili](#)

## 2.1 前端优化

### 2.1.1 项目场景优化

在原始场景中，我使用的是游戏自带的默认场景，但是我希望整个项目可以处在一个充满科技感的场景中，因此我选择了 Epic 商城中的 Modular SciFi Season 1 Starter Bundle 环境资源，其资源地址为([Modular SciFi Season 1 Starter Bundle](#))。这个项目共有 4 个环境区。这里我选择了其中一个作为主场景，其环境展示如图 1 所示。



图 1 项目场景优化

## 2.1.2 项目人物优化

在项目初版中 NPC 是以固定的姿态出现的，这样会显得不自然。因此，我计划对人物的站姿和动作进行优化。因此首先需要替换人物的初始骨骼结构。初始骨骼结构为了减少资源消耗，只包含手脚的骨骼。替换后的骨骼如图 2 所示。



图 2 替换骨骼

替换后的骨骼并不兼容其他骨骼模型，因此无法使用部分骨骼动画。这里需要找到当前骨骼所用骨架的资产详情，并在兼容骨架中配置如下两种骨架，如图 3 所示。做完这些步骤就可以对模型替换不同的动画和姿态，这里我选择了其中一种姿态，人物站立效果如图 4 所示。这里也可以选择其他的动作姿态。

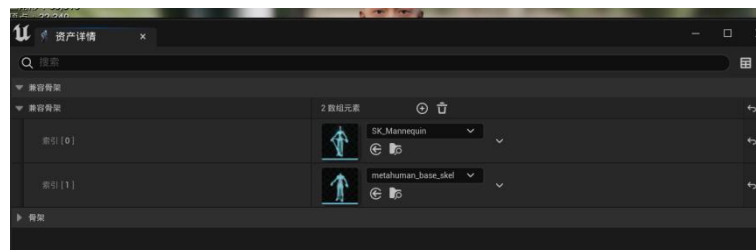


图 3 兼容骨架

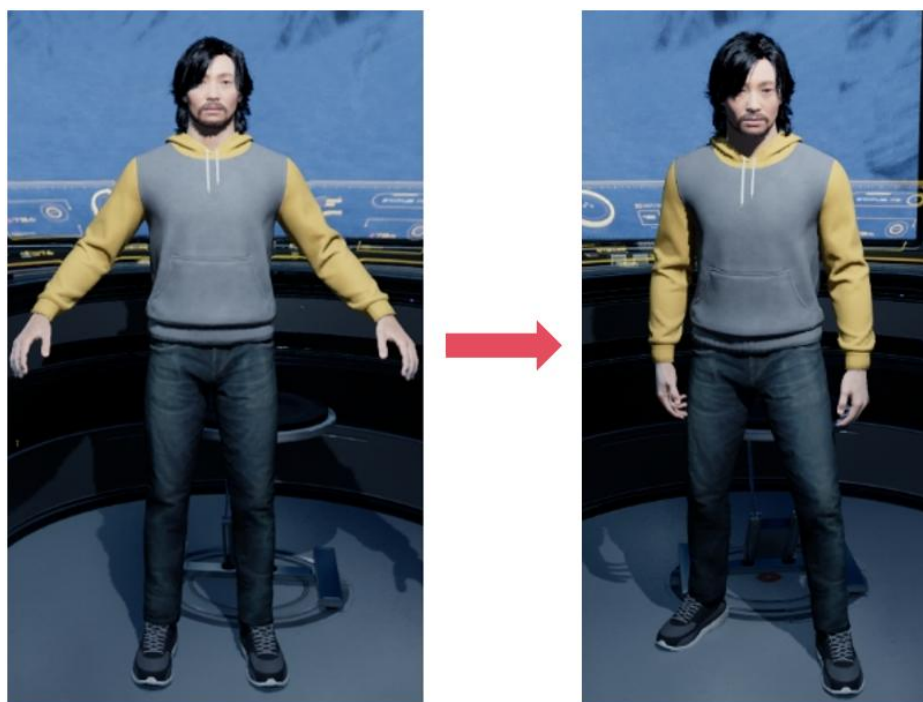


图 4 姿态对比

2.1.3 项目 UI 优化

项目原始版本中的对话方式过于简陋。为了贴合项目的科技风格需求，我重新设计了项目的 UI。首先，我们通过编写一个相机转场视角，当特定事件触发后，相机会移动到与数字人对话的视角，蓝图如图 5 所示。

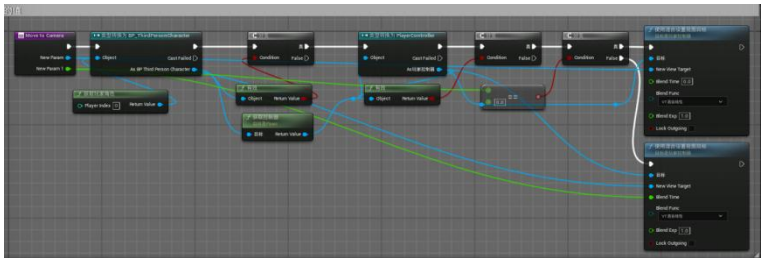


图 5 相机移动视角（部分）

然后，我们需要设计对话框的蓝图和 UI，以便更好地呈现对话内容。对话框的蓝图如图 6 所示，UI 优化后呈现的效果如图 7 所示。其中具体每个按钮的功能和作用将在后端优化中介绍。

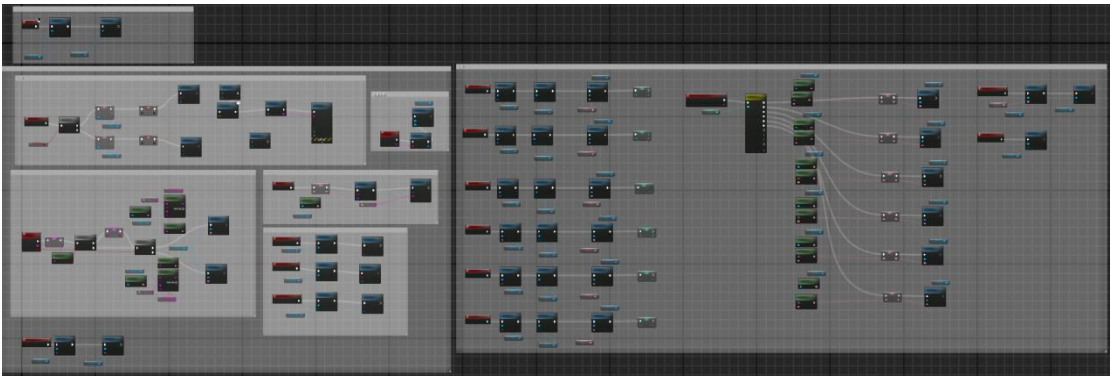


图 6 对话框蓝图（部分）

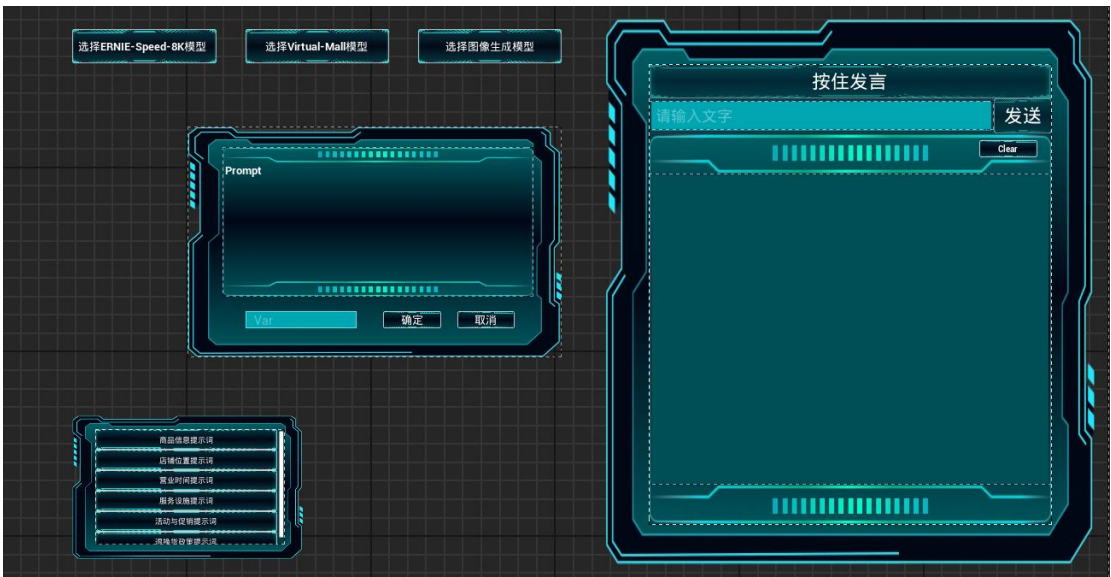


图 7 UI 优化效果

## 2.2 大模型优化

对于大模型模块，可以训练一个专门的大模型或对现有基础模型进行精调。这里我考虑了一个**虚拟数字商城**的场景，并采用对大模型进行专项精调的方法来实现这个应用场景。首先要确定模型精调的方式，这里我采用监督微调，即 **SFT**。通过监督微调，大模型可以根据特定任务的监督数据进行调整，以更好地满足特定任务的性能要求。然后需要选择微调的基础模型并确定训练的相关参数，具体参数在图 8、图 9 中给出。

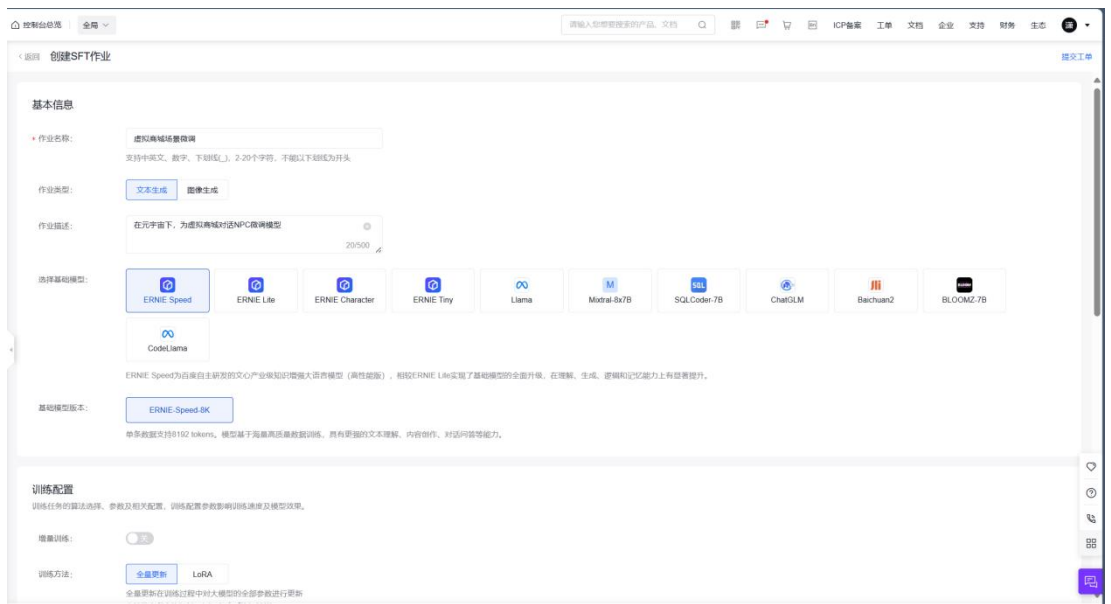


图 8 SFT 参数 1

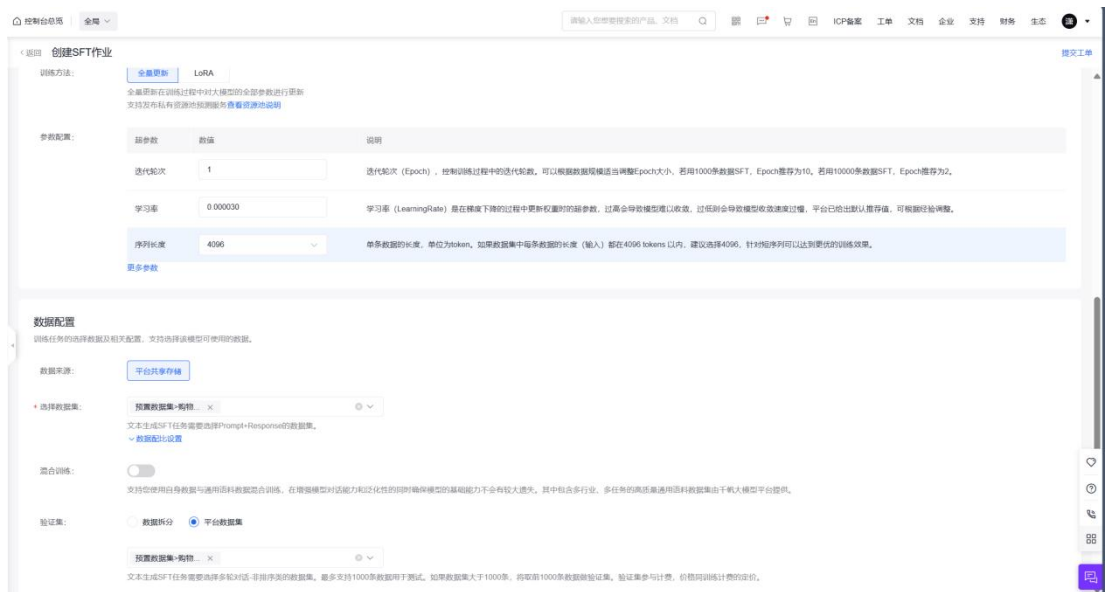


图 9 SFT 参数 2



然后就开始对模型进行训练，训练的过程在图 10 中给出。

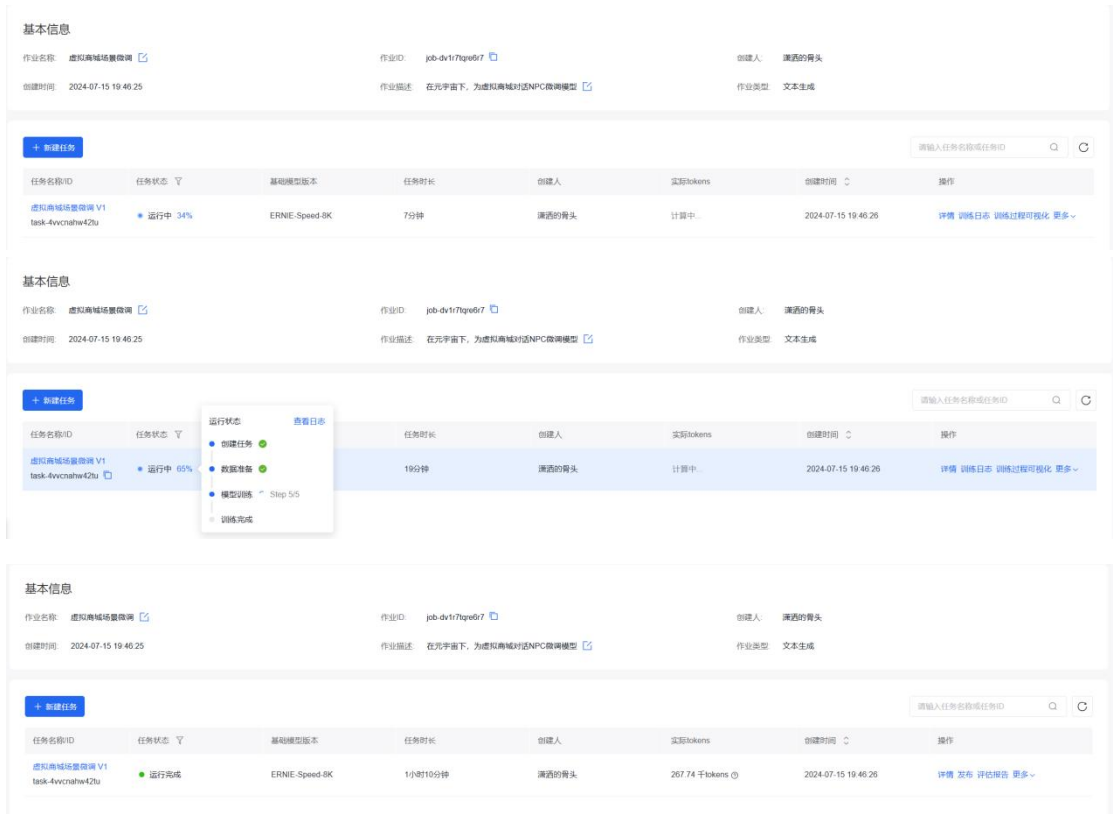


图 10 训练过程

在训练过程中，可以通过可视化展示训练中的 **loss** 和 **PPL** 参数。其中 **loss** 表示模型预测与实际结果之间的差异，数值越低表明模型性能越好。对于 **PPL**，其全称为 **perplexity** 是语言模型的评估指标，表示模型对数据的预测能力，数值越低表明模型的预测效果越好。具体参数可视化结果在图 11 中展示。



图 11 训练过程可视化

在模型运行完成后，这里可以查看模型的训练评估报告，其中包含了更加完善的模型评估指标，具体内容如图 12 中所示。

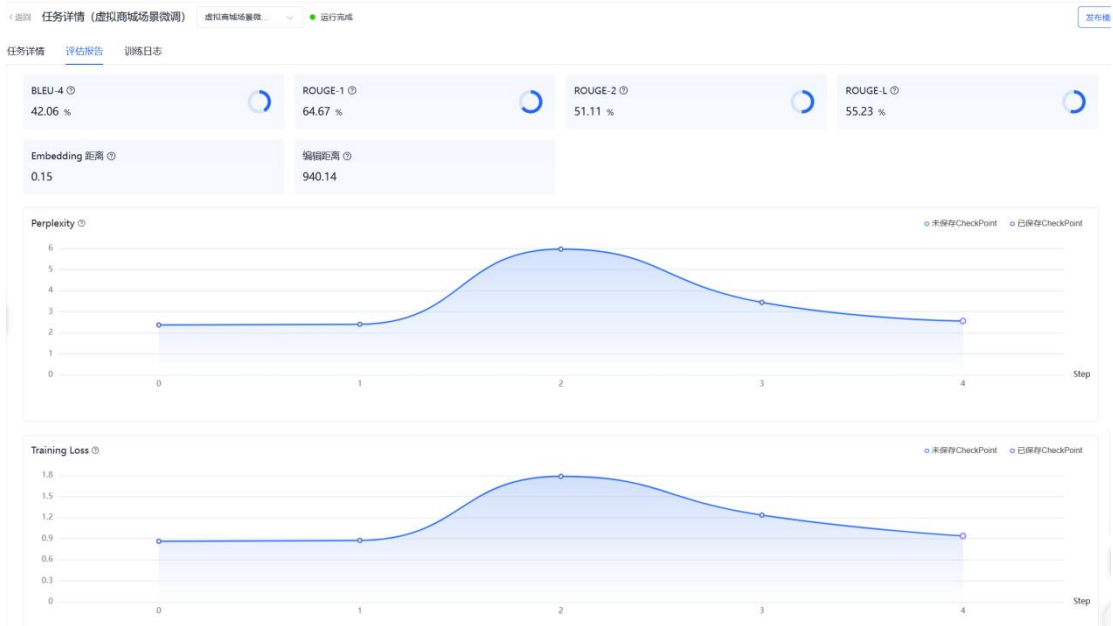


图 12 模型评估报告

在完成上述步骤后，即可进行模型的部署。首先需要发布训练好的模型。然后，在之前创建的应用实例中启用该模型。然后就可以在获取令牌后访问指定的 API 以获取回答数据。模型的信息在图 13 中给出。

+ 创建服务									
请输入服务名称、服务ID									
服务名称	服务ID	服务状态	模型名称及版本	基础模型版本	付费方式	创建人	创建时间	更新时间	操作
元宇宙虚拟商城服务	svco-j3cyd2f...	已发布	Metastar_zju v1	ERNIE-Speed-8K	Tokens (后付费)	潇洒的骨头	2024-07-15 21:20:19	2024-07-15 21:27:40	详情 下线 查看日志

服务详情 (元宇宙虚拟商城服务) 已发布

服务地址: [https://aip.baidubce.com/ipc/v2/ai\\_custom/v1/wenxinworkshop/chat/g6baer6\\_metastar\\_zju](https://aip.baidubce.com/ipc/v2/ai_custom/v1/wenxinworkshop/chat/g6baer6_metastar_zju)

使用前提: 请先完成应用接入权限配置, 详情请查看使用文档

服务信息

版本列表

基本信息

服务名称	元宇宙虚拟商城服务	服务ID	svco-j3cyd2f...	服务状态	已发布
模型名称	Metastar_zju	模型版本	v1	创建人	潇洒的骨头
更新时间	2024-07-15 21:27:40	数据回流	未开启	服务描述	-

付费信息

付费类型	Tokens	付费方式	后付费	输入单价	0.004元/千tokens
输出单价	0.008元/千tokens				

服务配置

QPS	1	购买时间	2024-07-15 21:22	开始时间	2024-07-15 21:27
-----	---	------	------------------	------	------------------

图 13 模型信息



## 2.3 后端优化

### 2.3.1 上下文相关优化

原版本的对话是上下文无关的。通过保存每次询问和返回的数据，确保每次保存的数据为偶数个，再在下次询问时加入新数据，从而保证每次询问的总条数为奇数，使对话实现上下文关联。相关修改的蓝图如图 14 所示，最终的效果展示如图 15 所示。

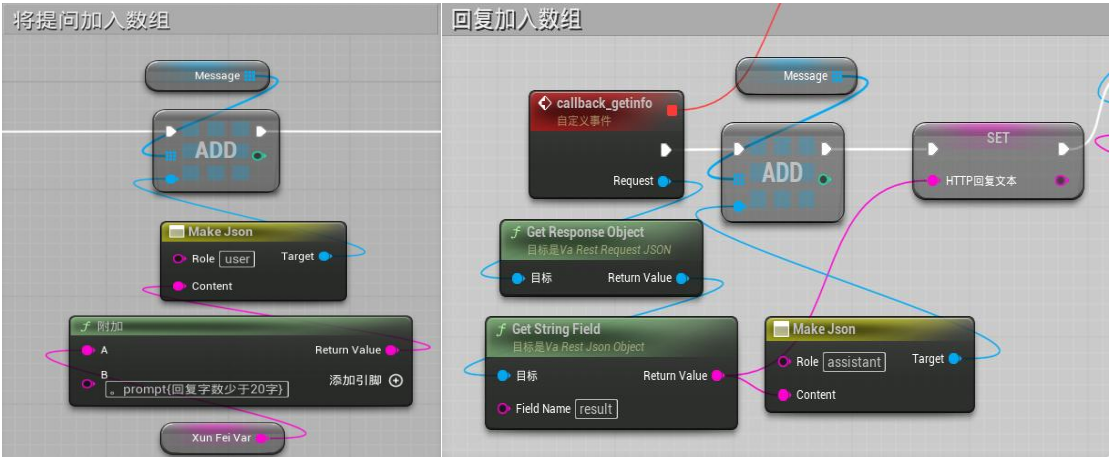


图 14 数据保存蓝图（部分）



图 15 上下文相关对话效果展示

### 2.3.2 模型类型切换

在对大模型进行优化之后，就可以在项目中调用它。为了方便不同模型的切换，可以添加一些按钮来实现模型之间的切换。如图 16 所示。



图 16 模型切换按钮

为了对比微调后的模型与基础模型在虚拟数字商城场景中对话的智能和专业程度，我使用相同的语句对两个模型进行测试。结果显示，微调后的模型明显比基础模型的回答更加专业。测试对比在图 17 图 18 中给出。



图 17 使用 ERNIE-4.0-8K 模型



图 18 使用微调后的模型

### 2.3.3 提示词模板

在虚拟数字商城的使用场景中，用户可能会有许多相似的询问。为了方便用户操作，我根据 6 种常用询问的问题，提供了 6 种 prompt 模板，用户只需替换模板中的变量即可直接使用。

优化后可见左下角出现了提供提示词模板的选项框，如图 19 所示。



图 19 用户界面

点击提示词模板按钮，会出现如下界面，如图 20 所示。

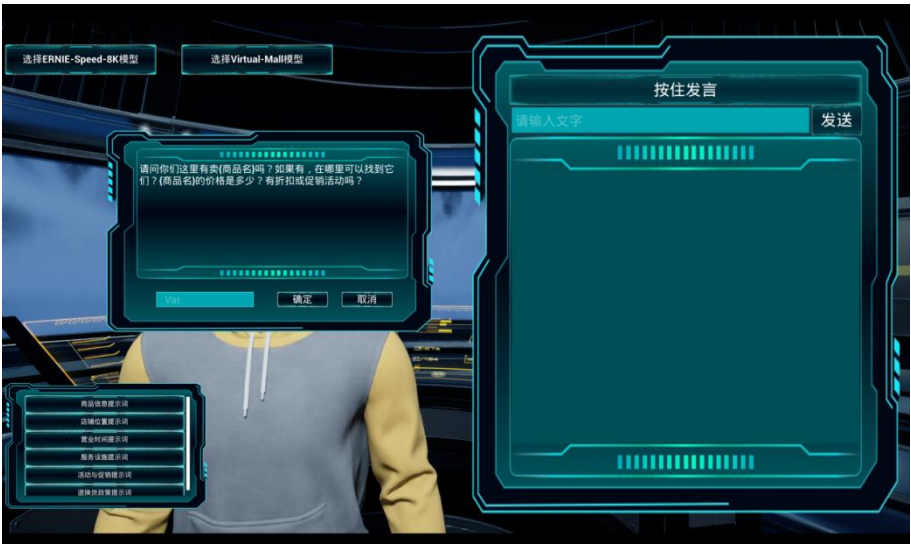


图 20 点击提示按钮界面

提示词模板界面具体细节在图 21 中给出，上方文字是提示词模板的具体内容，{} 中的部分是可以替换的变量。用户可以在 Var 输入框中输入想要询问的变量，上方文字会根据输入内容实时更新。点击确认后，模板内容会转入发送框，用户点击发送即可进行询问。如图 22 所示。





图 21 提示词修改框



图 22 提示词演示界面

### 2.3.4 图像生成功能

对于图像生成部分，在获取 Token 时与上述文本处理模型类似，但在处理返回值时需要特殊处理。返回的数据是 base64 编码的图像信息，需要先解析 JSON，再对数据进行进一步处理，将其转换为 Texture 2D 格式的图像，并在 UI 界面的图像单元上显示。具体处理如图 23 所示。

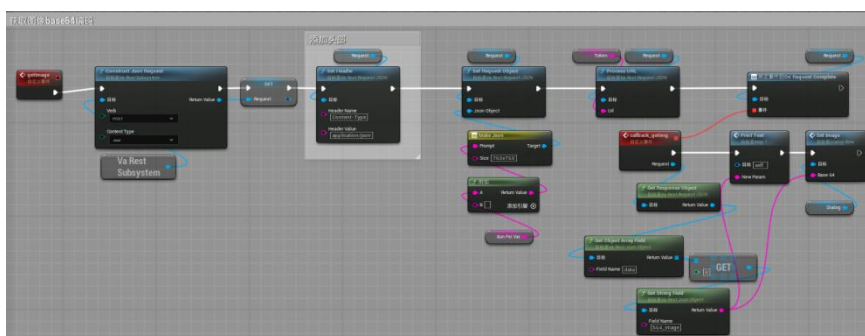


图 23 图像解析蓝图（部分）

点击“选择图像生成模型”按钮会弹出如图 24 所示的默认图片界面，在右端输入需要生成图像的 prompt 即可显示生成图像信息，如图 25 所示。



图 24 默认图像界面



图 25 生成图像界面

## 二、未来展望

在完成任务一和任务三的过程中，我构想了以下场景：

基于现有的快速建模技术和游戏引擎，我们可以在需要虚拟化的现实环境中安装摄像机和雷达等设备，利用 3D Gaussian 等快速建模技术，将实时生成的模型呈现在虚拟环境中，实现即时的现实虚拟化。人们只需站在家中的特定移动设备上，就能体验到穿越到世界任何地方的效果。

在这些虚拟场景中，我们可以添加仿真的 NPC，这些 NPC 可以通过大模型技术实现智能化的互动，并接入当前环境的数据，提供定制化的回答策略。在上述内容构建的元宇宙环境中，我们可以在这些虚拟场景内设置虚拟商城，玩家可以咨询虚拟 NPC 关于货物的信息，并且在虚拟世界中购物，通过现实中的网购平台完成实际的发货。同时，也可以将世界上著名的景点实时虚拟化，玩家可以不用考虑经济和拥挤等因素，实现远程旅游。

图 26 中展示了该构想的一个雏形，其场景是由 3D Gaussian 进行建模生成，NPC 利用 Metahuman 实现，并接入的大模型对话功能。



图 26 3D Gaussian 技术与大模型技术在元宇宙框架下的结合

但是想要真正实现上述内容还需要考虑如下问题：

- 1、现有技术生成的点云包含过多细节，可能会导致虚拟环境较为卡顿，这里需要平衡实时性和精度，在保证建模精度的同时，需要实现实时建模，以保证用户体验的流畅性。这需要对 3D Gaussian 的建模过程进行修改，并且对引擎中模型的材质、光照、纹理特别是 LOD 进行优化。
- 2、NPC 需要加强语言处理和理解能力，理解并回应不同用户的多样化需求。在任务 3 子任务 2 的大模型优化的步骤中我已经对虚拟数字商城这个场景下的 NPC 调用的大模型进行了初步的模型精调，但是想要真正满足这种场景的应用需要对大模型进行更加复杂的优化，或者直接开发针对性的大模型框架。
- 3、想要完成上述构想还需要对游戏引擎进行开发，首先要开发支持实时更新和渲染点云模型的自研插件。还需考虑服务器的搭建、VR 设备的支持等技术问题。
- 4、此外，隐私安全、虚拟行为规范和市场潜力等因素也需要考虑。