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 2+子任务 1

目录

一、子任务 1

任务要求：思考需要使用大模型 API 的地方，拟定开发应用所针对的场景。（举一个例子，同学们可以不限于这个场景，大胆地发挥自己的想象：利用大模型 API 完成虚拟法律顾问的功能，开发一款法律咨询的应用）后端功能需要完整，不限开发语言，开发框架，但必须使用到大模型 API，并基于此为应用提供一些智能的服务。需要有一个能够展示的前端页面，能够演示同学们所开发的智能应用的主体功能，需要体现出自己使用到大模型 API 能力的地方。

项目演示视频：bilibili

1.1 项目设计

大模型是深度学习领域中的一种重要技术，近年来被广泛应用于处理文本数据和生成图像。这类模型通常包含数百亿甚至更多的参数，通过在大规模数据集上进行训练，旨在理解和生成自然语言及图像。它们能够基于大量的文本数据进行语言预测、文本生成等任务。像 OpenAI 的 GPT 系列和国内的文心一言等都是大模型的典型代表。

此外，元宇宙是一种新的技术概念，旨在以用户为中心，综合当前几乎所有软硬件技术的互联网应用。它代表了信息化发展的新阶段。在综合运用现有先进技术的同时，元宇宙还推动相关技术的迭代升级，甚至催生新的技术。

结合大语言模型和元宇宙两种最新技术，我构想在元宇宙框架下，利用大模型 API 和其余五种技术，构建一个生成式的元宇宙世界。对于基于 Metahuman 的 NPC，我利用科大讯飞作为 NPC 的“耳朵”，文心一言作为 NPC 的“大脑”，Vits 作为 NPC 的“嘴巴”，用户可以通过 prompt 构建不同的对话风格和专攻方向的 NPC，成功实现了一个逼真的 NPC 系统。前端将采用虚幻引擎进行用户界面、游戏世界和角色等的构建，包括 3D 模型、场景、动画和角色展示。用户可以通过键盘、鼠标等输入设备进行操作。后端利用虚幻引擎的蓝图进行编写，包含部分 C++封装的蓝图功能，负责与文心一言 API 进行通信，包括发送请求、接收响应和解析数据等。这样一个系统将提供一个高度互动和沉浸式的虚拟世界，结合先进的自然语言处理能力，实现丰富的用户体验和智能交互。项目的结构图如图 1 所示。

在完成上述功能后，我计划将其与任务 1 中的三维重建技术相结合，实现快速将现实场景虚拟化至元宇宙框架下，从而实现现实与虚拟的统一。这一具体概念将在任务 3 子任务 2 文档中未来展望部分详细阐述。

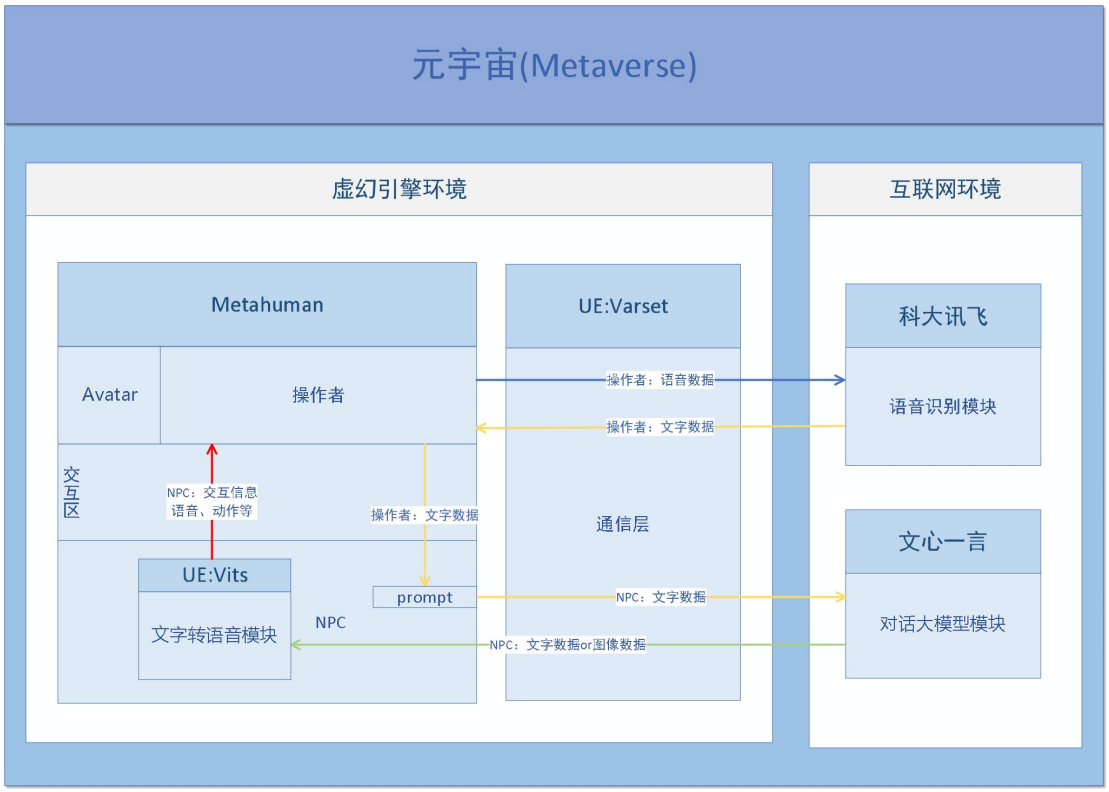

图 1 项目技术、功能结构图

1.1.1 Unreal Engine

2022 年 4 月，随着 UE5 的正式发布，一段官方演示视频点燃了我对虚拟现实技术和游戏开发的热情。虽然当时我还尚未了解元宇宙这个概念，我就已经拉着身边的同学组建了一个团队，开始了虚幻引擎技术的开发。如今，时间拉回到大三的这个暑假，我凭借本科期间发开的多个虚幻引擎项目，已经获得了**两项国家级一等奖**和**一项国家级二等奖**（此外还有机器人方向的**一项国家级一等奖**和**一项国家级三等奖**，这五项国家级奖项我**均为第一作者**）。我曾经参与的项目还被写进录取通知书，并向全体新生展示。我非常希望能够将我积累的技术带到浙江大学这一更广阔的平台，在未来元宇宙领域有所建树。

言归正题，虚幻引擎 5 将带来前所未有的自由度、保真度和灵活性，帮助游戏开发者和各行业创作者打造新一代实时 3D 内容和体验。新一代虚幻引擎不仅在游戏创作方面成熟，其光影效果、动画、渲染和物理系统等特性也广泛应用于其他领域，包括新兴的元宇宙等行业，越来越需要逼真、灵动的 3D 内容，以提升用户对产品和技术的体验，增强研发、生产和制作效率。

基于我已有的技术积累，考虑到虚幻引擎与项目题目的契合度，我决定依托大语言模型和虚幻引擎进行项目开发。

1.1.2 Metahuman

Avatar 在游戏和虚拟现实领域中扮演着至关重要的角色。它们是玩家或角色的可视化代表，为玩家提供沉浸式的体验，avatar 能沉淀你的个人行为、链上活动、资产信息、声誉和成就，使元宇宙从虚幻走向真实，也让玩家之间的互动更加丰富多彩。在某些观点中被认为是元宇宙等虚拟现实技术的入口。

传统的 3D 建模工具包括 3Dmax、Maya 和 Blender 等，雕刻软件则有 Zbrush 和 Blender 等，程序化建模则通常使用 Houdini。传统 3D 软件主要用于制作低模，雕刻软件则辅助制作高模。低模的特点是面数少，视觉效果一般，但计算资源占用少，运行速度快；高模则面数多，视觉效果好，但资源占用多，容易导致卡顿。

传统的建模方式不仅存在上述缺点，而且过于复杂，有一定的技术门槛，建模周期较长，对于快速定制化构建虚拟世界的需求有一定的劣势。而 Metahuman 填补了这一空缺。Metahuman 依托于 Unreal 引擎开发，是一种超写实数字人。MetaHuman Creator 是一款云端流送应用，设计目的是在不牺牲质量的前提下，将实时数字人的创作时间从数周乃至数月缩短到一小时以内，操作界面如图 2 所示。它的工作原理是基于一个不断增长的、丰富的人类外表与动作库进行绘制，并允许用户使用直观的工作流程雕刻和制作想要的结果，从而创作出新角色。

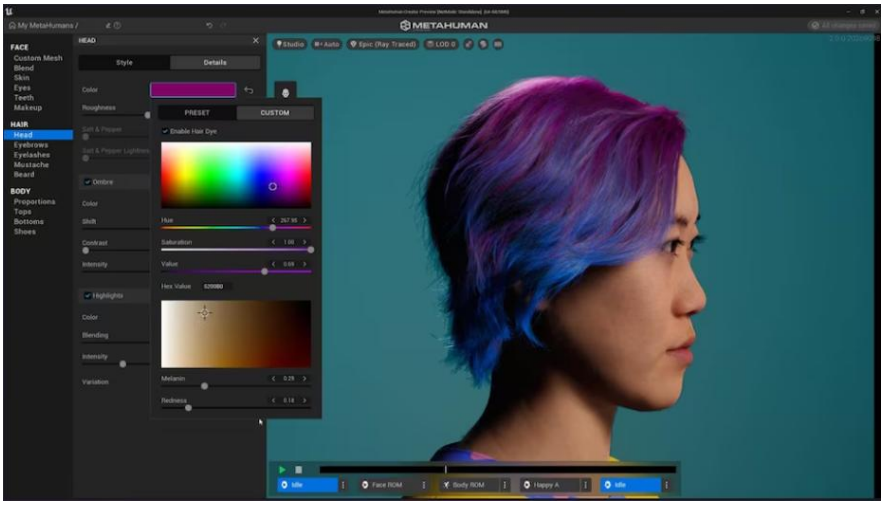

图 2 MetaHuman Creator 界面

Metahuman 除了可以用于 avatar，还可以用来构建游戏 NPC。实现一个逼真的 NPC 系统需要考虑多个方面，包括骨骼动画、面部表情捕捉、语音识别等。本项目的 NPC 基于 Metahuman 的数字人技术，其提供了完善的一键式的骨骼和动画系统。此外，我利用科大讯飞作为 NPC 的“耳朵”，文心一言作为 NPC 的“大脑”，Vits 作为 NPC 的“嘴巴”。在有限的时间内，成功实现了一个逼真的 NPC 基础功能。

1.1.3 Varest

Varest 是一个专门用于简化网络请求和数据交互处理的工具。它能够轻松地发送和接收 HTTP 和 HTTPS 请求，通过与远程服务器通信来支持各种数据格式，包括 JSON 和 XML。其采用事件驱动的设计，使开发者能够根据请求的状态执行相应的处理逻辑，有效地控制游戏流程和反馈。Varest 与引擎的其他功能和系统无缝集成，为开发者提供了更加灵活和高效的开发体验。基于这些功能，本项目选择采用 Varest 进行 HttpAPI 请求和 JSON 解析。

1.1.4 Vits

VITS 是一种结合了变分推理、标准化流和对抗训练的高性能语音合成模型。它与传统的语音合成方法不同，通过引入隐变量而非直接处理频谱信息，显著提升了合成语音的多样性和自然度。该模型的工作原理主要涉及两个关键组件：编码器和解码器。在 VITS 中，文本首先经过编码器转化为隐变量表示，然后解码器利用这些隐变量逐步生成语音波形。与传统方法不同的是，解码器不是一次性处理整个隐变量序列，而是逐步输入部分序列的隐变量，这一设计显著减少了计算负载和复杂度。

此外，VITS 还采用了对抗训练策略，通过生成器和判别器的竞争机制进一步优化生成的语音质量。在训练过程中，生成器致力于生成接近真实语音的合成结果，同时判别器则通过对比合成语音和真实语音来识别模型生成的内容。这种对抗训练方式提高了模型的生成能力和鲁棒性，使合成语音更加自然和易于理解。

由于本项目的重点不在于 VITS 模型的训练，而是在应用其功能方面，因此我选择采用了已经由他人训练好的模型数据。PyTorch 的 pth 模型资源地址为：(https://huggingface.co/spaces/sayashi/vits-uma-genshin-honkai)， pth 格式转换为 ONNX 格式的项目地址为(https://github.com/Winter-of-Cirno/MoeGoeONNX/)。驱动数字人说话的编程逻辑如图 3 所示。

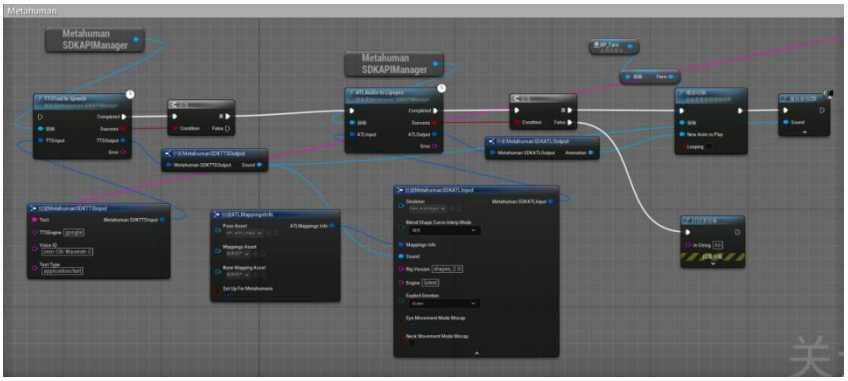

图 3 蓝图：驱动数字人说话（部分）

1.1.5 科大讯飞

语音识别是将人类的语音转换为计算机可识别的文本或命令的技术，在很多领域广泛应用。科大讯飞作为国内领先的语音技术公司，其语音识别库具有高准确性和稳定性。因为我之前有使用科大讯飞语音模块的开发经验，所以在项目选题时便考虑使用科大讯飞的语音转文字 API 来丰富项目的功能。

本项目使用科大讯飞的语音转换模块充当 NPC 的“耳朵”，科大讯飞语音识别模块 API 获取的操作过程如下。首先登录科大讯飞的开放平台官网图 4。

图 4 讯飞开放平台主页面

然后登录进入控制台或者我的应用创建一个应用实例，如图 5。

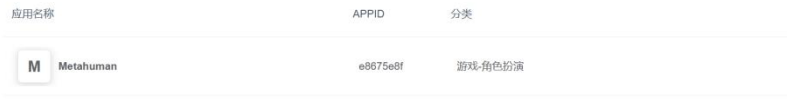

图 5 Metahuman 应用实例

下载科大讯飞提供的 SDK 添加依赖功，这里下载 Wondows MSC，如图 6。

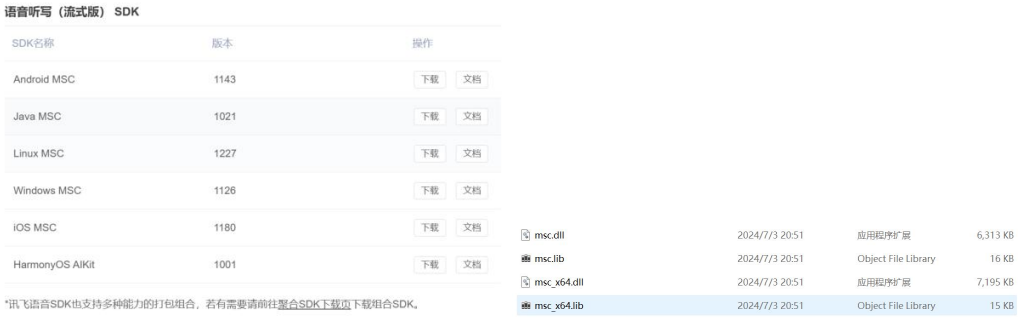

图 6 科大讯飞 SDK 下载展示

在虚幻引擎中，我利用 C++程序封装蓝图节点，以此蓝图节点调用科大讯飞的 API 实现了语音转文字的功能。

1.1.6 文心一言

日常中我通常使用 OpenAI 的 ChatGPT 及其 API 接口。在虚幻引擎中，关于 ChatGPT 的插件和使用教程比较完善，可以轻松地实现与 ChatGPT 的集成。然而，在这次的项目中，我打算采用国内一款大模型。首先，从零开始调用接口可以使得项目具有较高的可控性，这对于任务 3 的子任务 2 来说是非常重要的，能够为我留下足够的灵活空间。其次，选择不使用已有的插件可以更好地展示项目开发的难度和技术挑战，提升技术上的探索性。第三，我也希望通过这个项目支持国内大模型技术的发展，同时也借此机会深入了解国内大模型的现状和特点，为未来更深入的开发工作做好准备。

文心一言和天工大模型都是国内布局较早的大模型，本次开发中我选择了文心一言，其是由百度开发的人工智能大语言模型，具备跨模态和跨语言的深度语义理解与生成能力。它拥有五大核心能力：文学创作、商业文案创作、数理逻辑推算、中文理解和多模态生成。文心一言在搜索问答、内容创作生成、智能办公等领域展现出广泛的应用潜力，能够为用户提供更加智能和高效的服务体验。文心一言的企业服务由百度旗下的千帆大模型平台提供，包括推理服务和大模型微调等一系列开发和应用工具链，为企业提供了强大的技术支持和定制化解决方案。

其官方给出的调用流程如图 7 所示。

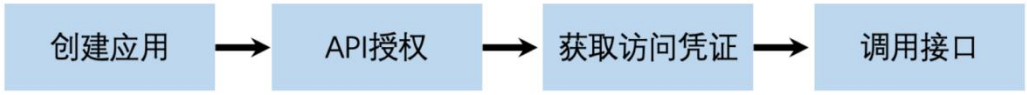

图 7 文心一言 API 调用流程

进入控制台创建千帆应用,创建应用后，获取 AppID、API Key、Secret Key。

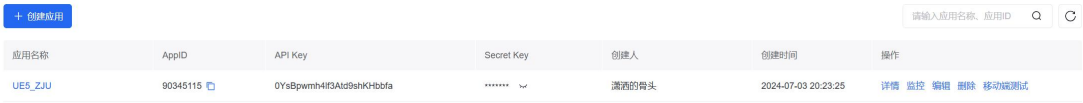

图 8 文心一言创建应用界面

应用创建成功后，千在千帆大模型平台-在线服务页面，点击开通付费。千帆平台默认为应用开通部分 PI 调用权限，这里我使用 ERNIE-4.0-8K 服务。

图 9 文心一言服务开通界面

在虚幻引擎中对文心一言 API 进行调用，这边只展示了最基础的的调用逻辑，

在子任务 2 中会对这部分进行更加详细的优化。如图 10 所示，展示了获取访问令牌的编程逻辑。

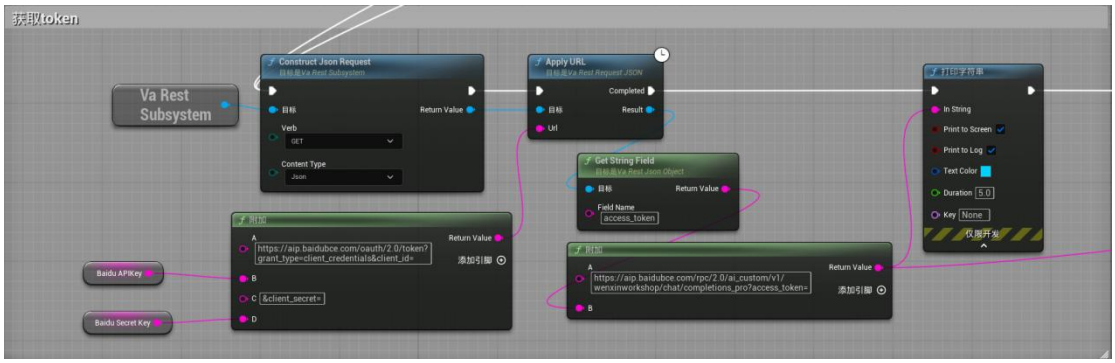

图 10 蓝图：获取文心一言 Token（部分）

图 11 展示了获取令牌后构建请求 URL，设置请求的有效负载和请求头，接利用 Varest 发送一个 POST 请求，并打印 API 返回的响应内容。

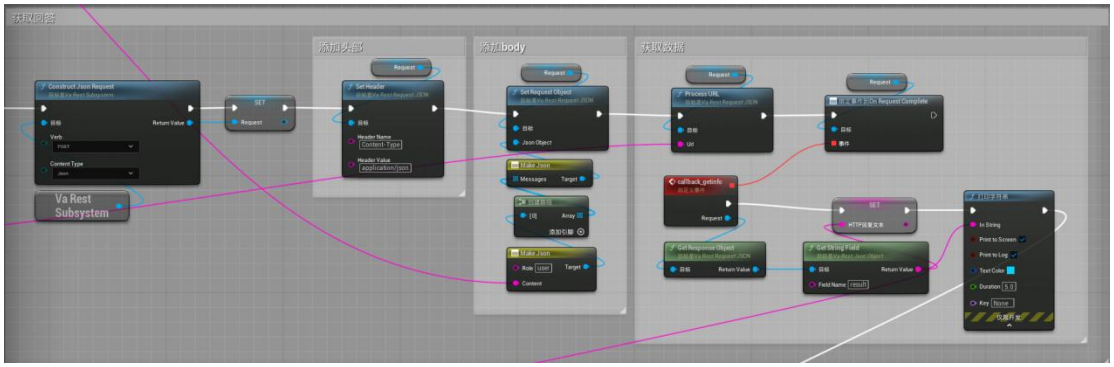

图 11 蓝图：获取请求数据（部分）

至此，项目所使用的 6 种技术架构都已经介绍完毕，下面就是项目展示部分，项目功能的完善和提高将在子任务 2 中详细的阐述。

1.2 项目说明与展示

这里只是对初版项目进行了一个最基础的说明与展示，更加优美和完善的项目将在任务 3 子任务 2 的项目报告中被介绍和展示。本部分的基础展示如所示。

首先按住空格建，开始读取语音，松开空格后会将语音数据传至科大讯飞平台，由科大讯飞平台将其转换为文字信息返回。在图 12 中，我将文字信息展示到了对话框中，其中左上角显示的是访问令牌信息。

图 12 科大讯飞

在通过科大讯飞获取文字信息后，将所获取的文字信息传输至文心一言的 API，并将返回的回复信息显示在对话框中。如图 13 所示。这里为了方便呈现，我通过 prompt 限制了返回的字数。

图 13 文心一言

项目中的 NPC 是基于 Metahuman 进行实现的，在基础的功能中，NPC 只需要实现发音和面部嘴型动画的实现，如图 15 所示，在虚幻引擎 5.1 版本中需要对 Metahuman 的面部动画蓝图进行修改才能正常运行，如图 14 所示，这里需要把默认插槽移动到图中的位置。

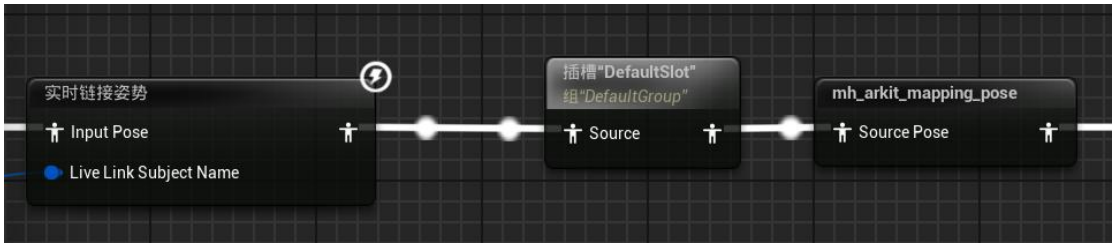

图 14 嘴部动画蓝图

这里 NPC 的发音使用了 Vits，由于本项目的重点不在于 Vits 模型的训练，而是侧重与大模型技术的应用，因此我这里使用了由他人训练好的模型数据。并且通过构建 Metahuman 的嘴部控制器来实现 NPC 的实时口型变换。其效果如图 105 所示。

图 15 NPC 发音展示