（二）拟开展的研究工作（建议不超过4000字）

着重阐述拟开展的研究工作的科学意义和创新性，技术路线、研究方案等的可行性。

3.2 完成一稿

3.5 刷完二稿

**题目：面向智能化软件开发的超维需求建模与理解**

2.1 科学意义与创新性 - SYH

近年来，智能化软件工程领域呈现出迅猛的发展态势。在传统的软件开发模式中，程序员主要通过手动编写代码来实现软件功能。然而，在智能化软件工程时代，大模型的兴起为软件开发范式带来了颠覆性的变革。大模型依托深度学习与自然语言处理技术，能够理解程序员以自然语言描述的任务需求，并基于已有的软件工程知识库、代码模式和最佳实践，自动生成符合上下文语义的代码，并完成代码补全、代码生成等特定任务，极大地提高了软件开发的效率和质量。《2035年远景目标纲要》明确提出要“推动软件产业高质量发展，提升软件供给能力，培育壮大高水平软件企业和产业集群”，其中智能化软件开发作为软件产业的重要发展方向，受到了高度重视。随着代码大模型能力的不断提升以及其应用范围的逐渐普及，软件开发范式有望得到颠覆性的革新。

尽管大模型在已经在解决特定编程任务上表现出了令人满意的能力，但仍难以精准地理解实际开发任务的复杂需求，导致其生成的代码的一致性和可靠性较低。目前，主流的大模型，如GPT-4、GPT-3.5、DeepSeek Coder和CodeLLaMa等，在解决基于真实物理世界中实际开发任务而构建的测试时，表现均大幅下降。例如，GPT-4-turbo在 DevEval和ClassEval真实数据集上Pass@1平均得分仅有53.05%和37.6%，远低于在HumanEval上80%的平均得分。在当前的智能化软件开发范式中，大模型在执行代码智能任务时呈现出静态性与单维性的特征，即大模型仅能基于用户主动提供的自然语言描述与代码片段来对任务需求进行分析并生成代码。这种静态和单一维度的需求理解，给大模型在实际开发中落地带来了巨大阻碍。具体包括：

1. **缺乏对程序员编程环境的全面理解，导致智能化软件开发难以适配实际场景。**开发者的编程环境（例如集成开发环境 IDE）中既包含代码仓库上下文等静态信息，也涵盖程序员编程行为的数据记录等动态信息。实际的编程行为发生于一整套 “环境 - 行为” 时间序列之中，每一步操作与前后动作存在逻辑关联，通常与程序员当前的内在需求高度相关。然而，目前缺乏对程序员编程环境的全面理解，导致在实际智能化开发场景下大模型表现不佳，难以适配。
2. **缺少对程序员编程行为的序列化理解以及演化感知，难以突破智能化软件开发的高精准难题。**研究结果表明，若能在特定任务的代码推荐系统中，运用对开发者任务与活动的高层次抽象及预测，便能更为有效地把握需求，进而提升推荐任务的准确率。但在大模型驱动的实际软件开发过程中，由于开发活动的进展与程序员习惯的变化具有复杂性，从序列化行为的理解和对于任务的抽象与预测面临较大困难。
3. **缺乏对程序员自身特征的理解，难以实现智能化软件的个性化内容生成。**开发者自身的特征，例如技术能力、开发习惯、工作能力等，蕴含着隐式的个性化需求。针对同一种显式需求表达，应该根据不同的开发者自身特征给出不同的答案。然而用户维度的需求目前未能得到有效地利用。

综上，在实际开发任务中，开发者的真实需求是蕴含在一种更为复杂的“超维上下文”物理环境中的，这其中包含开发环境、执行任务和程序员特征等多种要素，具有整体性、多维性、互交性和动态性等特点。缺少“超维上下文”会导致大模型无法全面理解准确开发任务的真实需求，进而影响其生成代码的准确性和实用性。因此，迫切需要**构建一种新的超维需求建模方法，从多个维度刻画程序员开发的物理环境，解决现有单维需求理解所面临的信息瓶颈。**通过挖掘复杂编程任务的“超维上下文”信息，提升大模型对于实际开发任务真实需求和环境的理解，从而显著优化大模型在实际开发任务中生成答案的准确性和一致性，为我国的智能化软件开发技术的发展提供坚实的保障。

申请人拟基于智能需求分析的工作基础，扩展智能需求工程的外延，如图所示，聚焦面向智能化软件开发的超维需求建模与理解难题，突破：（1）**基于超维数据感知和压缩的环境维需求建模**，利用静态分析工具收集本地代码仓库环境和程序员个性化行为的静态信息动态变化，并利用大模型对超维数据上下文进行高效压缩；（2）**基于液态神经网络的任务维需求建模**，通过训练低开销、高实时性端侧模型系统，对开发者的行为序列进行理解和演化感知，实现对开发者未来行为的预测；（3）**基于启发式模型的用户维能力建模**，构建一个可解释的规则引擎与量化评估框架，将程序员的技术特征、能力边界与行为偏好等转化为可计算的维度化表征，在本地构建一个程序员数字画像。通过从环境维度、任务维度和用户维度这三个维度收集并充分利用丰富的背景信息，实现对开发者需求的全方位、深层次理解。最后依托软件开发集成环境，研发（4）**集成超维需求理解的智能IDE平台**，在跨行跨文件的代码补全、个性化代码生成、自动缺陷修复等代码智能任务上开展应用示范。

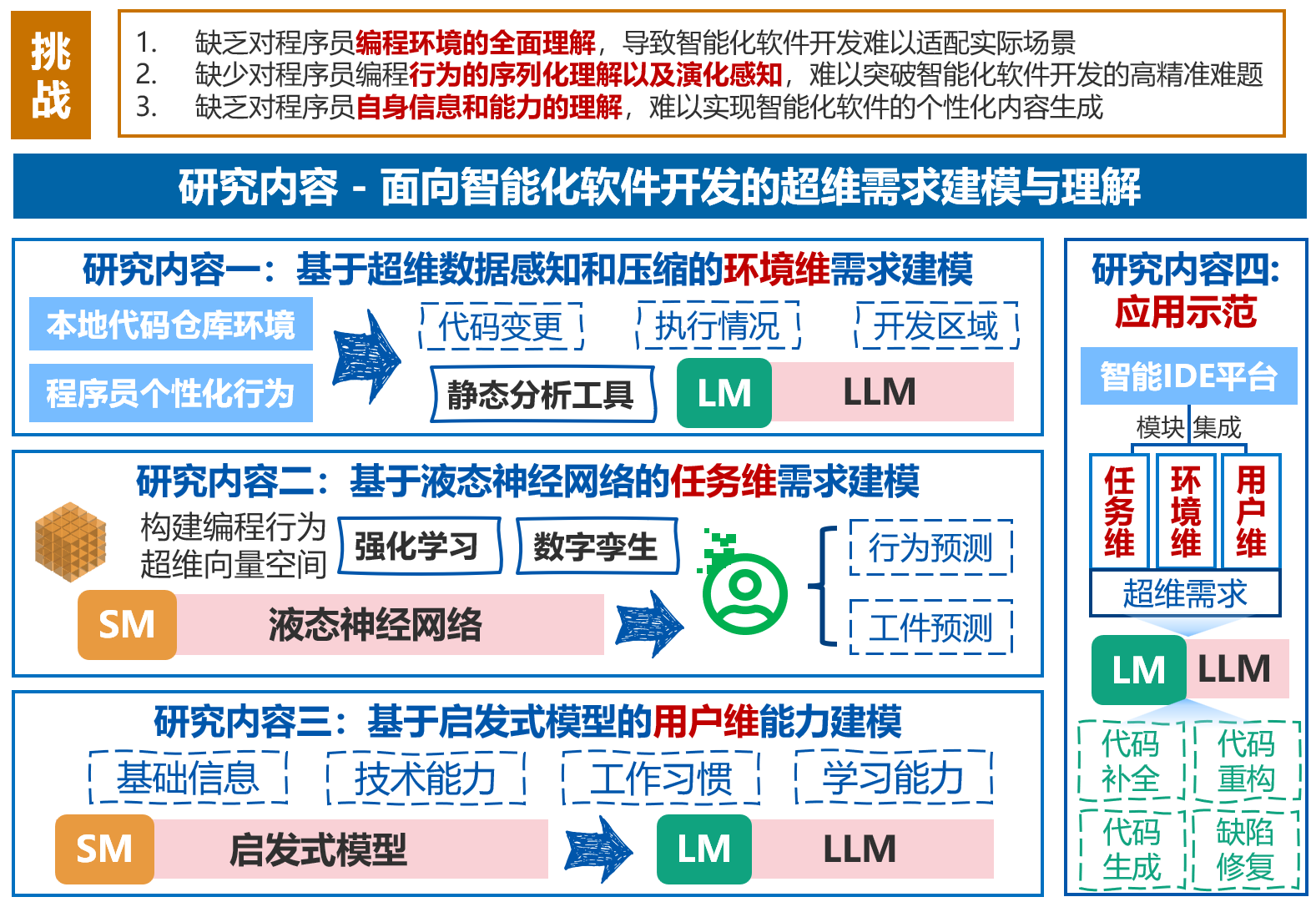


图2.1 拟开展的研究内容

参考文献： - ALL

2.2 研究内容

本项目具体拟研究基于超维数据感知和压缩的环境维需求建模，以期解决现有方法因上下文信息不足而无法有效利用IDE现场高维复杂数据的问题；研究基于液态神经网络的任务维需求建模，力图设计低开销、高实时性端侧模型系统，克服现有模型基于静态上下文、无法个性化适配及隐私担忧等问题；研究基于启发式模型用户维需求建模，旨在通过构建可解释规则引擎与量化评估框架，解决对开发者形象建模缺失的难题；开展面向智能化软件开发的应用示范，构建集成超维需求理解的智能IDE平台。最终形成一套支持智能化软件开发的超维需求建模与理解机制，针对数据理解、开发效率与应用适应性三个方面提升软件开发过程的智能化水平。

2.2.1 基于超维数据感知和压缩的环境维需求建模

随着代码智能体在自动化编程任务中的广泛应用，对程序员内在需求的准确理解和响应变得至关重要。IDE环境作为编程活动的第一现场，蕴含了大量与程序员当前需求紧密相关的高维复杂数据，但现有方法往往受限于上下文信息不足，无法有效利用这些潜在的数据资源来提升智能体的性能。研究期望通过整合IDE现场数据来全面描述程序员操作行为，为精准分析其内在需求提供坚实基础。研究拟开发从IDE环境感知用户需求的数据建模方式，包含三个关键部分：（1）**本地代码仓库环境建模**，涉及代码依赖检索、技术栈统计及IDE插件生态分析；（2）**程序员个性化行为建模**，通过监听细粒度事件并结合Git模块记录代码变更；（3）**环境上下文压缩**，采用大模型分部压缩策略，针对不同数据来源提取关键信息，为代码智能体提供编程现场上下文。

2.2.2 基于液态神经网络的任务维需求建模

随着AI编程工具如Cursor的兴起，通过结合微调的大模型与代码上下文分析技术，实现了对程序员下一步操作的预测。然而，现有方法多基于静态上下文信息，缺乏对具体动作在整体序列中的理解，且闭源的云端大模型解决方案引发了用户隐私和数据上云的担忧。此外，这些模型无法随个体用户的习惯和能力同步进化，限制了其个性化服务能力。为解决现有问题，研究计划设计一种低开销、高实时性的端侧模型系统，以同步理解程序员的开发任务。拟采用液态神经网络（Liquid Neural Networks），因其能高效处理非均匀时间序列数据，并适用于时序敏感任务。项目将开发一个实时编码器，将行为事件编码到高维向量空间，利用液态神经网络捕捉程序员的行为模式，并通过强化学习算法进行优化，使模型能个性化适配每位程序员的习惯，精准辅助代码任务，同时保护隐私并降低计算开销。

2.2.3 基于启发式模型用户维需求建模

随着IDE与AI编程助手的发展，对开发者形象建模的缺失成为限制其个性化解决用户问题能力的关键因素。开发者自身形象、技术类型和熟练度与个性化需求紧密相关，却未得到充分重视，这为智能体服务质量提升带来挑战。在此背景下，研究基于启发式模型的用户维需求建模方法意义重大。本项目拟通过构建可解释的规则引擎与量化评估框架，将程序员的技术特征、能力边界与行为偏好等转化为可计算的维度化表征，把开发者画像划分为开发者基本信息、技术能力、工作习惯、学习能力四个主要模块，通过多维度计算指标全面表征程序员画像，为下游任务的代码智能体提供用户维度的需求理解。

2.2.4 面向智能化软件开发的应用示范

随着智能化技术的发展，AI编程助手在软件开发中愈发重要，目前业界多数AI编程助手采用接受用户指令、相关代码和终端上下文作为一般上下文构建流程。但这种架构存在缺陷，当用户指令不够完善或相关上下文信息有限时，智能体难以准确把握用户真实意图，这阻碍了软件开发智能化进程与效率提升。在此背景下，基于前三部分研究内容的程序员超维需求的建模理解方案应运而生，方案拟构建面向智能化软件开发的应用示范，整合环境、任务、用户维需求建模，**研发集成超维需求理解的智能IDE平台**。智能体将能够通过该平台获取环境、任务、用户三个维度的需求和上下文信息，在代码补全、缺陷修复等任务中表现更佳，显著提升代码任务质量，推进软件开发全流程自动化，还为程序员提供自我认知与技术提升的平台。

2.3 研究目标 - PZP

（1）本项目旨在构建基于超维数据感知和压缩的环境维需求建模机制，通过整合IDE环境中的高维数据（代码仓库建模、行为建模、上下文压缩），全面描述程序员操作行为并精准分析其需求，为代码智能体提供丰富的编程上下文信息，提升任务执行质量。

（2）本项目研究基于液态神经网络的任务维需求建模方法，设计低开销、高实时性的端侧模型系统，利用液态神经网络处理非均匀时间序列数据的优势，同步理解程序员开发任务并预测个性化行为，为代码智能体提供精准辅助，同时保护隐私并降低计算开销。

（3）本项目构建基于启发式模型的用户维需求建模方法，通过可解释的规则引擎与量化评估框架，将程序员技术特征、能力边界与行为偏好转化为可计算的维度化表征，形成全面的开发者画像，为代码智能体提供用户维度需求理解，提升个性化服务能力。

（4）本项目开展智能化软件开发应用示范，整合环境、任务、用户三维需求建模，研发集成超维需求理解的智能IDE平台，为代码智能体提供多维上下文信息，提升代码补全、缺陷修复等任务质量，推动软件开发全流程自动化，为程序员提供自我认知与技术提升支持。

2.4 拟解决的关键科学问题 - PZP

**（1）面向超维数据利用的环境维需求建模**

当前代码智能体因上下文获取局限于用户主动提供部分，难以有效利用IDE环境中与程序员内在需求紧密相关的高维复杂数据，且IDE多种维度数据未被统一收集利用，限制了智能体性能。因此，需要研究基于超维数据感知和压缩的环境维需求建模方式，通过设计开发从IDE环境感知用户需求的数据建模方法，包括对本地代码仓库环境建模、程序员个性化行为建模以及环境上下文压缩，**为精准分析和理解程序员内在需求提供坚实数据基础**。

**（2）面向个性化、低开销、高实时的任务维需求建模**

当前基于云端大模型的AI编程工具存在代码上云隐私问题且模型无法随个体用户习惯和能力进化，难以精准预测程序员意图。因此，需要在端侧构建一种低开销、可学习、高实时的模型系统以满足任务维需求建模。拟采用液态神经网络模型，结合编码器将程序员行为事件编码为向量空间时序轨迹，采用基于GRPO的强化学习算法，设计多维度联合奖励函数，**使模型能以较低计算开销随程序员日常使用逐渐学习并个性化适配，实现与程序员同频理解开发任务，输出精准的行为模式理解、事件类型和编辑位置预测结果，**辅助下游智能体完成代码任务。

**（3）面向智能体个性化服务的用户维需求建模**

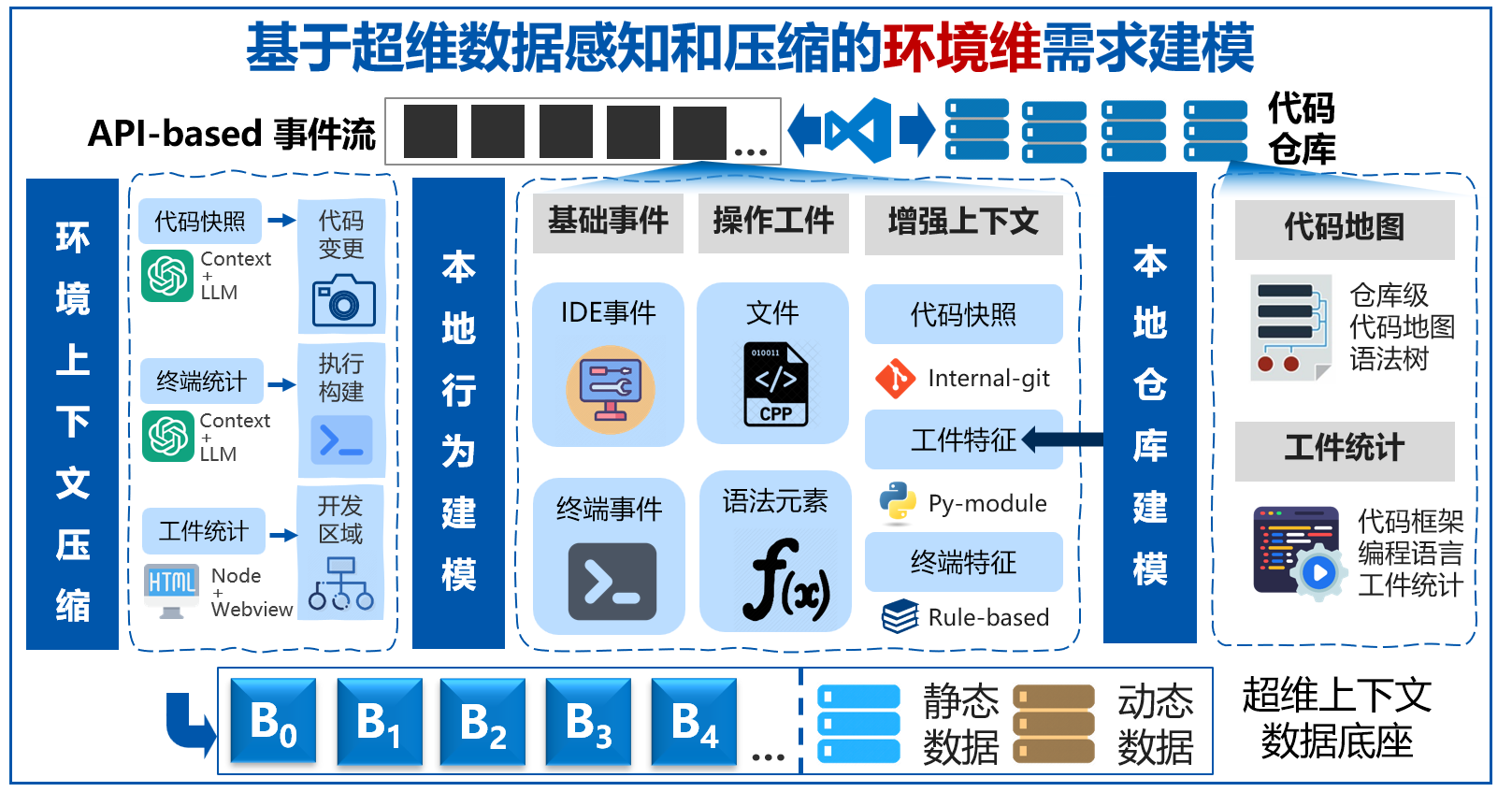
当前IDE与AI编程助手因缺乏对开发者形象的充分建模，导致无法有效利用开发者自身形象、技术类型和熟练度与个性化需求间的逻辑关联，限制了智能体个性化解决用户问题的能力。通过构建可解释的规则引擎与量化评估框架，将程序员的技术特征、能力边界及行为偏好转化为可计算的维度化表征。以前述超维上下文数据底座为输入，结合预定义规则库和动态权重分配机制，构建开发者画像，利用多维度计算指标全面表征程序员画像，从而**为大模型提供精准的用户认知上下文，提升智能体在下游任务中的个性化服务水平，满足用户维度的需求理解**。

2.5 研究方案 - LYH

2.5.1 基于超维数据感知和压缩的环境维需求建模

代码智能体高质量地完成执行程序员的指令，离不开对程序员内在需求的全面获取和准确分析。IDE作为编程的第一现场，其环境中含有大量高维复杂可供挖掘的编程行为数据记录，这些环境数据往往与程序员当前的内在需求高度相关。而当前的代码智能体能获取的上下文大多局限于用户主动提供的片段和输入记录等，而实际的编程行为发生在一整套“环境-行为”时间序列中，每步动作与前后动作存在逻辑关联，上下文不足限制了智能体的性能，IDE潜在的来自多种维度的数据未能得到统一收集和利用。因此，需要研究**一种基于超维数据感知和压缩的环境维需求建模方式，关键在于如何充分整合IDE现场数据、全面描述程序员在环境中的操作，从而为精准分析和理解程序员内在需求提供坚实的数据基础。**

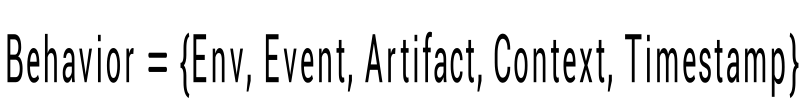
本项目拟研发一种从IDE环境感知用户需求的数据建模方式，形成描述程序员在IDE环境维需求的数据底座。该研究的技术路线如图所示，分为对仓库建模、程序员动作建模和环境上下文压缩三个部分：

1. **面向本地代码仓库环境建模**

首先对代码文件提取抽象语法树（AST），使用基于静态分析的算法构建代码地图，用于代码依赖关系的检索和计算。其次，统计仓库使用的各项编程语言和代码框架等，用于统计归类程序员的技术栈以及评估项目的复杂度。最后，还统计IDE的插件生态、本地跨仓库信息、识别单元测试、注释情况等，用于综合构建程序员个人偏好和能力。

1. **面向程序员个性化行为建模**

本项目拟在研发阶段使用开源的VSCode平台作为示范载体，在扩展中嵌入事件监听模块和相应的算法层，借助其丰富的开放式接口读取程序员开发过程中的所有细粒度操作事件。对于读取到的API-based事件流，拟将所有事件统一建模为如下形式：



其中，Behavior是程序员的每次动作；Env是动作发生的环境，按发生环境可以将事件类型分为IDE事件和终端事件；Event是事件类型，基于IDE可以收集到的事件类型进行总结，例如文本编辑、鼠标点击、符号跳转等；Artifact是事件操作对应的对象即操作工件，粒度由粗到细可以是代码文件、类符号、函数级符号，基于代码地图和AST进行识别；Context是事件对应的上下文信息，例如代码变更内容、终端命令与输出、操作工件之间的联系；Timestamp是时间戳。

此外，对于Context上下文字段，拟结合一个内部集成的Git模块，通过更细粒度的快照保存实现代码变更情况的近实时记录。通过规则匹配实现终端操作的类型和执行结果情况的细分类。通过前述的代码地图建模实现操作工件之间的联系计算，包含操作重叠度、依赖耦合度、内容相似度等指标综合表征工件之间超维意义上的联系强度。

1. **环境上下文压缩**

上下文压缩在代码大模型处理长代码序列时显得尤为重要，主要原因在于计算和实际应用的限制。Transformer架构的二次复杂度意味着随着上下文长度的增加，计算资源和时间的消耗呈指数级增长，直接处理超长代码序列变得不切实际。现有的代码上下文压缩技术大多基于代码片段单一维度，不能适配研究提出的超维环境上下文数据建模。而完整的环境数据往往包含原始的代码变更Diff、完整终端历史和多样的事件类型等。因此，为代码大模型提供编程现场的丰富上下文信息的关键在于有效压缩环境上下文，针对不同来源的数据设计对应的策略提取关键信息。

本项目拟结合大模型分部压缩和总结不同来源的上下文。对于终端历史，整合分析连续的多个简单指令例如cd、mkdir等，单独分析复杂指令，使用指向性提示引导大模型分析程序员终端命令的目的、执行成功的期望结果、执行失败的关键输出字段和初步错误定位。对于仓库中的代码变更，按变更行数限制时序读取Git Diff记录的数量，总结代码变更的摘要，推测代码编辑事件的目的和要实现的功能。对于开发区域，分析程序员的高频开发区域在仓库中所属的功能块，例如UI界面、单元测试、项目配置等。

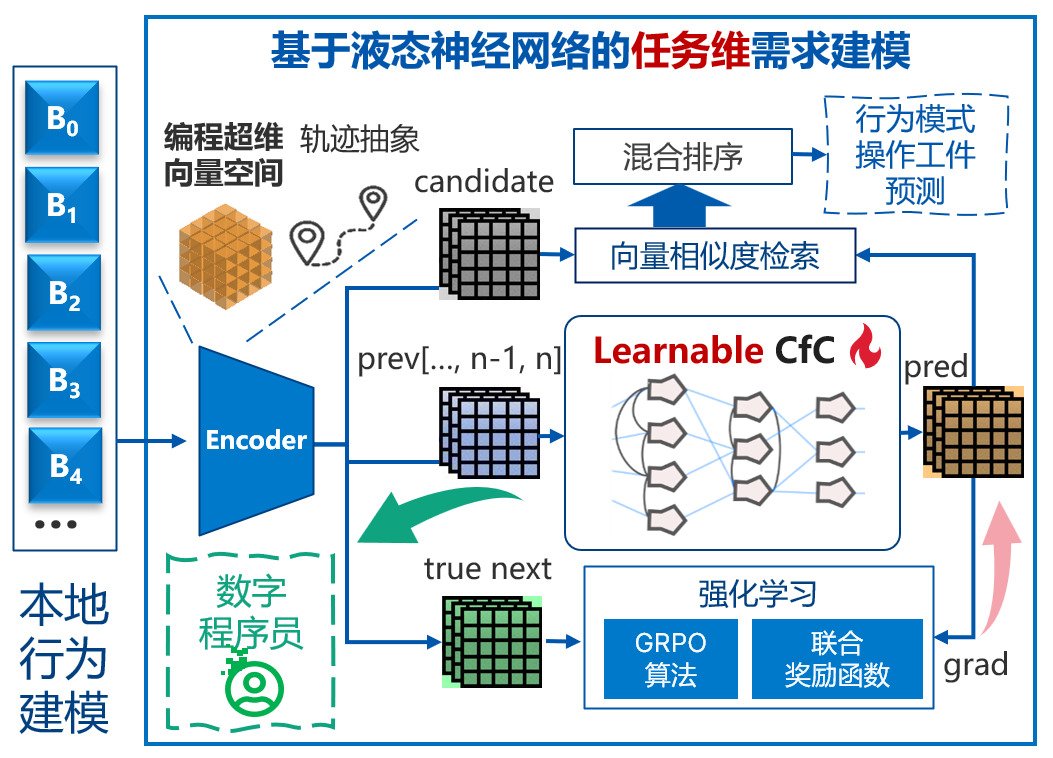
2.5.2 基于液态神经网络的任务维需求建模

近年来，Cursor等AI编程工具通过结合微调的大模型与代码上下文分析技术，实时分析用户当前编辑的代码文件及整个项目的上下文信息。模型能够理解当前光标位置的代码片段以及跨文件识别代码全局关系，实现了预测程序员下一步可能修改的位置。但是当前的预测仅基于静态的上下文，对具体动作缺少其在整体序列中的理解；基于云端大模型的闭源产品对于大多数企业用户存在代码上云等隐私问题，且Frozen的模型基于集体数据训练而来，不能随着个体用户的习惯和能力同步进化。该项目拟在开发任务维度进行程序员需求建模，预测程序员的个性化行为，关键在于**在端侧构建一种真正低开销、可学习、高实时的模型系统，实现与程序员同频理解开发任务。**

液态神经网络（Liquid Neural Networks，如LTC/CfC）是一类受生物神经元动态特性启发的模型，通过动态调整神经元的时间常数和连接强度，模拟生物神经系统的自适应特性。其核心机制包括连续时间处理、可变时间常数和稀疏连接，使其能够高效处理非均匀时间序列数据，并具备较强的噪声鲁棒性和低资源需求。液态神经网络擅长时序敏感任务和非均匀/稀疏数据场景，例如用户日志、实时预测等，尤其在低资源、低时延要求下表现出色，适合边缘计算或实时系统。

本项目拟使用液态神经网络模型来理解程序员行为模式、预测程序员的未来操作，其能够动态建模不规则的操作间隔例如代码键入突发与长时间思考等，并通过门控机制过滤噪声例如误触快捷键，从而聚焦关键事件。相比传统模型或连接到云端的大模型，使用端侧液态神经网络的方案在计算开销、去隐私化和可解释性方面表现更优。意图预测的结果可以提供行为模式和编辑位置预测，实现精准辅助下游智能体完成代码任务。

具体地，项目拟研究一种编码器，实时地接收流式Behavior事件，将其编码到高维向量空间中。该空间综合了多种维度的上下文信息，能够充分地将程序员动作序列表征为向量空间中的时序轨迹。程序员的行为模式隐式地包含在这些时序轨迹的Partten中。例如调查阅读一个仓库时一般会广泛地阅读的代码文件、查询代码依赖关系等，修复缺陷时往往会频繁执行构建命令、伴随少量的代码变更。液态神经网络模型负责实时捕捉和学习每种Partten下程序员动作之间的内在联系。其中拟采用基于GRPO的强化学习算法，设计多维度联合奖励函数，以较低的计算开销使得模型能随程序员的日常使用逐渐学习与个性化适配，实现与程序员始终保持近乎相同的编程习惯。

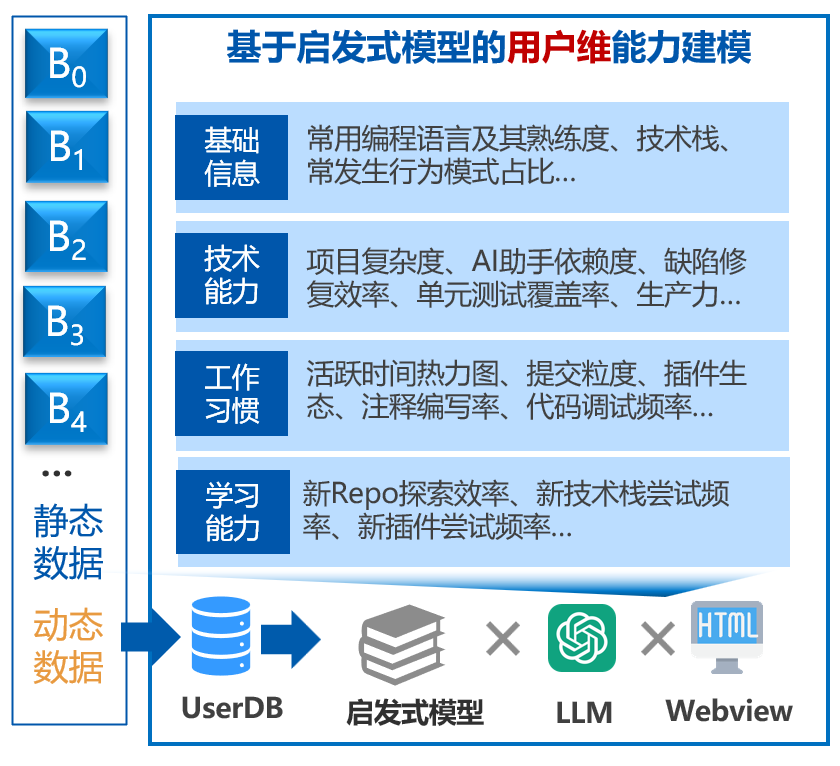
模型输出的预测值经过与数据底座提供的候选值进行相似度匹配并混合排序后，输出最终的行为模式理解、事件类型和编辑位置预测结果。

2.5.3 基于启发式模型用户维需求建模

当前IDE与AI编程助手普遍缺乏对开发者本身形象的建模，对开发者形象了解不够充分，而开发者的自身形象、技术类型和技术熟练度往往与个性化的需求存在逻辑关联，这限制了智能体在回答用户问题时的个性化解决能力。要解决这类问题，关键是**研究一种基于启发式模型的用户维需求建模方法，在本地构建和维护一个程序员的数字画像**。

本项目拟通过构建可解释的规则引擎与量化评估框架，将程序员的技术特征、能力边界与行为偏好等转化为可计算的维度化表征。该模型以前述的超维上下文数据底座为输入，结合预定义规则库与动态权重分配机制，既能为大模型提供精准的用户认知上下文，又可生成可视化的开发者能力演进图谱。

具体地，本项目拟将开发者画像划分为四个主要模块。一是开发者基本信息，例如常使用的代码框架、技术栈、常用编程语言占比与熟练度、代码阅读/编写/修复行为占比等。二是技术能力，包含AI代码补全依赖度、项目复杂度（仓库是否涉及多语言、多领域技术）、缺陷修复效率、代码生产力等。三是工作习惯，包含插件生态、工作活跃时间、提交粒度（偏向于持续集成或大块提交）、注释和单元测试编写覆盖率、以及通过继承的静态分析工具计算的代码质量等。四是学习能力，包含程序员探索新仓库时的理解效率、新插件和新技术栈的尝试频率等。通过以上多维度的计算指标，全面表征程序员画像。为下游任务的代码智能体提供用户维度的需求理解。

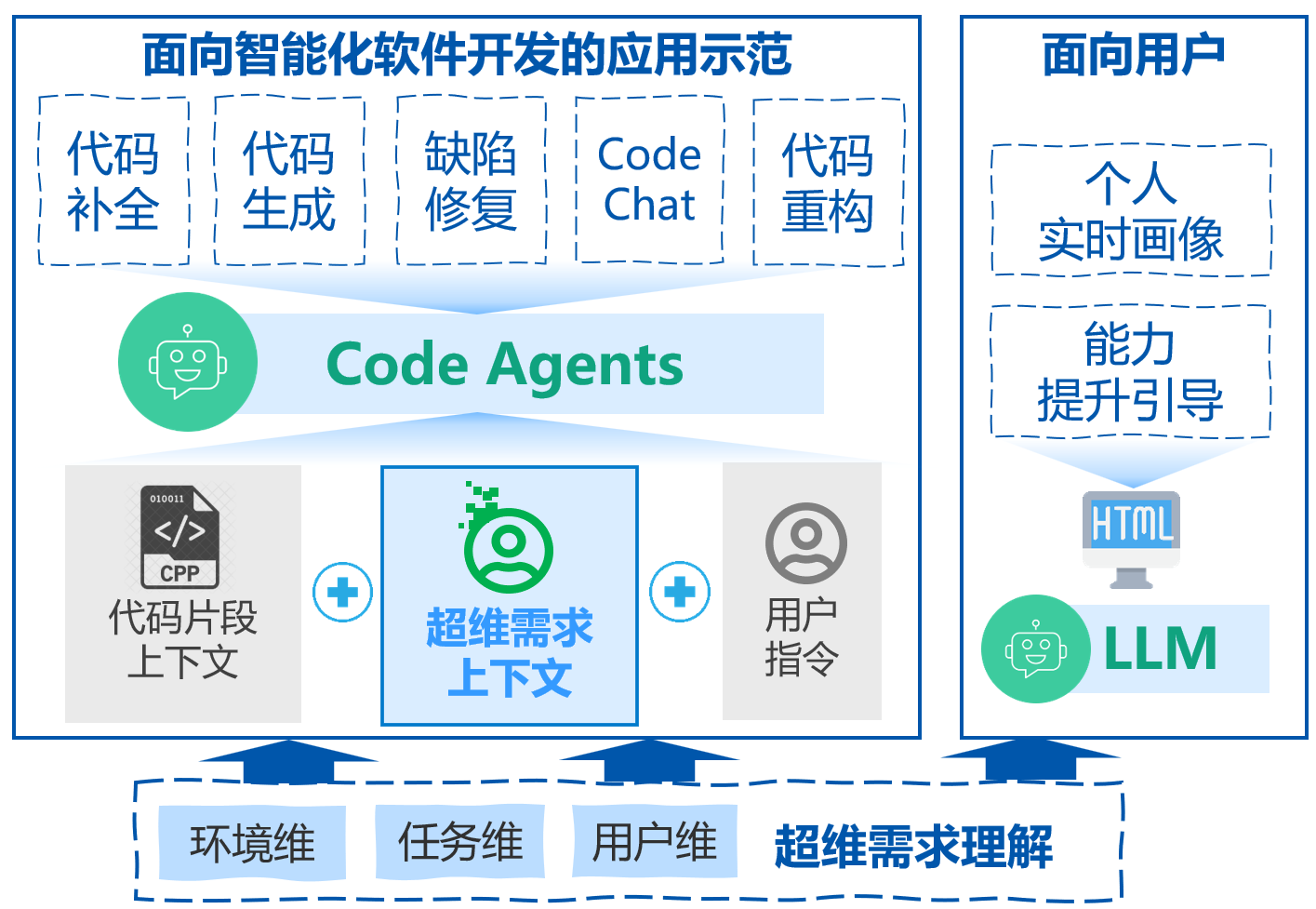


2.5.4 面向智能化软件开发的应用示范

业界大部分AI编程助手遵循的一般上下文构建流程，即执行一个任务时，接受用户指令和相关代码/终端上下文作为Prompt。但是在该架构中，用户只能通过优化指令和提供尽可能多的静态代码片段来表述自身的需求，当用户指令不够完善，或相关上下文信息有限时，智能体往往不能准确地理解到用户的真实意图。为解决这类问题，我们提出了基于研究内容1-3的程序员超维需求的建模理解方案，本项目**拟开展面向智能化软件开发的应用示范，整合环境、任务、用户维需求建模，研发集成超维需求理解的智能IDE平台。**

具体地，对于集成支持了超维需求建模预测的IDE，智能体可以增加了解到环境、任务和用户三个维度的上下文信息，这些信息直接或间接地表明了程序员的需求。例如在执行代码补全任务时，智能体不仅可以在程序员光标所在位置进行补全，还可以预测和推荐跨行跨文件的代码变更；执行缺陷修复时，能够根据终端报错历史定位到发生错误的代码位置，修复过程中，还可以自动执行或推荐程序员执行相关的终端命令辅助修复和验证。通过为智能体提供超维需求预测上下文，能够显著提升代码任务的完成质量，有效推动软件开发全流程的自动化。

此外，该数字程序员形象可以通过用户界面直接提供给开发者，并且结合相关大模型分析程序员自身的优势劣势，适合的技术类型，为程序员提供高效便捷了解自身的平台，以及寻求技术提升的有效途径。



2.6 可行性分析 -SYH

（1）研究基础坚实充分。项目申请人石琳具备总体负责该项目研究所需的科研基础和组织能力。申请人现为北京航空航天大学软件学院教授，长期从事代码智能大模型、智能软件工程、智能需求工程等与本项目密切相关领域的研究，在代码智能等方面取得了深厚的技术积累，在软件工程、人工智能领域的国际高水平期刊和会议等发表论文 60 余篇，多次获 ACM SIGSOFT 杰出论文奖和国际会议最佳论文奖（ASE'21，RE'21，RE'20）,入选 2022 年中国科学院软件研究所杰出青年人才。申请人承担了系列与研究内容相关的国家级研究项目，主持或参与了国家自然科学基金重点项目(“软件生命期数据组织、分析及应用研究”和“开源软件可信性分析与保障”)、面上项目（“面向社交媒体的开源软件漏洞智能感知”）、青年基金项目（“面向社交媒体的需求智能发现和分析方法”）。主持了两项 CCF-华为胡杨林基金（“大模型交互迭代式代码生成能力提升”与“基于大模型的代码精准重构技术”）。主持了两项北航-华为北航-华为关键软件联合实验室项目（“基于IDE开发流程和代码仓库的代码深度理解”与“AI代码生成能力优化”）。多次担任著名国际会议 ICSE、FSE、ASE 等的 PC，是 CCF 软件工程专委和系统软件专委会委员。

1. 研究方案科学合理。

本项目的研究内容和研究方案是在反复调研现有国内外研究进展，严格论证的基础上提出的，研究目标和研究方案切实可行。该方向是目前国际上大语言模型和智能需求工程领域的热点研究问题，受到来自北京大学、新加坡管理大学等研究者的关注。虽然当前的研究工作亟需进一步的突破，但这些工作为需求建模与理解研究提供了基础，为本项目提供了有意义的参考和借鉴。

**环境维需求建模研究。**该研究内容旨在解决当前大模型在代码智能任务中因缺乏对编程环境的认知而导致需求理解不足的问题。为此，研究将围绕仓库建模、程序员动作建模及环境上下文压缩三个核心方向，深入挖掘编程环境中所蕴含的需求信息，以提升大模型对需求的全面理解。在仓库建模和环境上下文压缩方面，已有研究者从多个视角开展了广泛的研究，为高效、全面地描述仓库环境奠定了基础。在程序员动作建模方面，申请人团队已在集成开发环境（IDE）中初步尝试实现了对程序员动作序列数据的全自动采集和结构化整理。这些基础研究成果为本项目的实施提供了坚实的理论与数据支撑，进一步验证了其可行性。

**任务维需求建模研究。**该研究内容聚焦于解决当前大模型在代码智能任务中因缺少对程序员正在执行任务的同频理解而导致需求理解不足的问题。为此，研究将通过构建一个低开销、可学习、高实时的模型系统，通过“任务抽象提取“和“开发行为预测“两个方面实现基于任务理解的开发者潜在需求挖掘。在“任务抽象提取”方面，已有研究者采用基于规则的方法对开发者的任务和活动进行初步提取，并将其应用于特定任务的代码推荐系统中。申请人也已经初步探索了开发任务的细粒度抽象与提取方法，拟发表于 FSE 2025。在“开发行为预测”方面，液态神经网络在时序序列预测中的应用已在多个领域取得了显著成果。例如在自动驾驶领域预测车辆行驶轨迹；在无人飞行领域预测飞行器的飞行状态等。虽然开发者行为预测具有特定领域的复杂性和特殊性，但这些跨领域的应用实践也为本项目提供了一定的参考，并证明了液态神经网络在行为预测中的潜力。这些前期研究与实践为本项目的成功实施奠定了基础。

**用户维需求建模研究。**该研究内容旨在解决当前大模型在代码智能任务中因缺少对程序员自身特征的认知而导致需求理解不足的问题。为此，研究将构建一个启发式模型，通过可解释的规则引擎与量化评估框架，来表征程序员的技术能力、工作习惯和学习能力，并通过构建一个程序员数字画像的方式帮助大模型更好地理解用户自身特征带来的潜在需求。在程序员特征分析方面，软件工程领域的众多专家已经对程序员特征分析展开了广泛的研究，并提出了多种成熟的程序员能力评估指标与方法，涵盖技能水平、编程风格、生产力分析等多个维度。在程序员数字画像构建方面，申请人已经基于人智与机智协同的角度完成了数字程序员以上这些为本项目研究提供了积极的借鉴和技术支持。

（3）研究环境国内先进。申请人的依托单位——北京航空航天大学软件学院，是国内开展软件工程方向研究的高水平科研机构，拥有软件开发环境国家重点实验室、大数据与脑机智能高精尖创新中心等国内一流的科研条件平台。围绕软件工程建设了一流的人才队伍和良好的软硬件环境、数据积累和长期稳定的国际交流合作。软件开发环境国家重点实验室拥有面积 4800 多平方米以及相对集中的科研环境，拥有浪潮 TS10000 集群计算系统 2 套、HP 刀片服务集群 1 套，各类高性能服务器 30 余台(套)，聚合计算能力达 5 万亿次，存储能力 40PB。大数据与脑机智能高精尖创新中心是首批北京高校高精尖创新中心之一。高精尖中心开展数据科学、智能科学、认知科学三大研究方向，高精尖中心拥有硬件基础设施和软件基础设施，包括：深度学习分布式集群 NAS 存储系统、GPU并行运算服务器、高速互联网络设备等，部署各类设备 207 台套，聚合高端 GPU处理器 300 余块，CPU 核心 1648 个，整体计算能力超过 2500 万亿次双精度运算（2.5PFlops），聚合内存 20TB，聚合存储能力 6PB。采购及自研了平台资源管理软件，实现了高效资源共享。平台整体性能达到世界超算排名TOP200，处于国内高校等科研机构领先水平。为项目的顺利实施提供了有力的技术依托和条件保障。

2.7 研究计划 - SYH

|  |  |  |
| --- | --- | --- |
| 时间 | 研究内容 | 预期目标 |
| 2025.01-2025.12 | 1. 跟踪和总结国际最新研究的程序员行为建模相关研究进展，制定更详细研究计划 2. 研究面向超维需求理解的行为建模机制 | 发表学术论文 3 篇以上,申请专利1项以上，参加国际学术会议 1 次，邀请境外专家来课题组指导 1次。 |
| 2026.01-2026.12 | 1. 跟踪和总结国际最新研究的数字程序员相关研究进展，制定更详细研究计划 2. 研究面向超维需求理解的数字程序员机制 | 发表学术论文 3 篇以上,申请专利1项以上，参加国际学术会议 1 次，邀请境外专家来课题组指导 1次。 |
| 2027.01-2027.12 | 1. 跟踪和总结国际最新研究的软件开发超维需求预测相关研究进展，制定更详细研究计划 2. 研究基于数字程序员的软件开发超维需求预测 3. 研发面向智能化软件开发的超维需求建模与理解系统 | 发表学术论文 3 篇以上,申请专利1项以上，参加国际学术会议 1 次，邀请境外专家来课题组指导 1次。 |

2.8 预期成果 - SYH

1. 通过本项目的实施，在面向智能化软件开发的超维需求建模与理解相关学术方向上提出和发展新的理论与方法，预期在程序员行为建模、虚拟程序员构建和需求预测方面取得重要进展和突破。
2. 研发面向智能化软件开发的超维需求建模与理解工具，将本项目的研究成果落地到IDE中，为实际的程序员软件开发工作提供智能化工具。
3. 在软件工程及相关领域的国内外高水平学术期刊和一流国际会议 (如ICSE、FSE、ASE、ISSTA、TOSEM、TSE、中国科学等) 上发表论文15篇以上，申请专利5项以上;
4. 培养博士生2~3 名、硕士生4~5名。同时，拟围绕面向智能化软件开发的超维需求建模与理解等主题组织 3~4 次研讨会，邀请国内外的优秀学者共同交流并寻求合作。
5. 通过本项目的开展，拟建设形成一支年龄和专业结构合理、理论研究与技术开发相结合的高水平研究团队，并与北京大学、约克大学、莫纳什大学等国内外相关机构的前沿研究团队保持良好的学术合作关系。

（三）其他需要说明的情况

1. 申请人同年申请不同类型的国家自然科学基金项目情况（列明同年申请的其他项目的项目类型、项目名称信息，并说明与本项目之间的区别与联系）。

无。

2. 具有高级专业技术职务（职称）的申请人是否存在同年申请或者参与申请国家自然科学基金项目的单位不一致的情况；如存在上述情况，列明所涉及人员的姓名，申请或参与申请的其他项目的项目类型、项目名称、单位名称、上述人员在该项目中是申请人还是参与者，并说明单位不一致原因。

不存在。

3. 具有高级专业技术职务（职称）的申请人是否存在与正在承担的国家自然科学基金项目的单位不一致的情况；如存在上述情况，列明所涉及人员的姓名，正在承担项目的批准号、项目类型、项目名称、单位名称、起止年月，并说明单位不一致原因。

不存在。

4. 其他。

无。