arXiv:2611.32374v1 [cs.SE] 24 Nov 2026

在需求工程中验证多模型人工智能生成的验收标准

Mateen Ahmed Abbasi[1] ✉、Petri Ihantola[1], Tommi Mikkonen[1], 和 Niko Mäkitalo[1]

于韦斯屈莱大学信息技术学院，芬兰于韦斯屈莱 {mateen.a.abbasi, petri.j.ihantola, tommi.j.mikkonen, niko.k.makitalo}@jyu.fi

摘要。 大语言模型（LLM）越来越多地应用于软件工程，用于生成需求、用户故事、验收标准、文档和代码等制品。在需求工程中使用这些模型，引发了有关正确性、可追溯性和幻觉的重要问题，尤其是因为有缺陷的规格说明可能影响后续的设计、实现和验证活动。近期许多关于人工智能辅助软件工程的研究都聚焦于代码生成和代码级验证，而人工智能生成的规格说明验证受到的关注较少。本文针对这一不足，提出一种面向人工智能生成的验收标准的验证方法。该方法结合了检索增强生成（RAG）、多模型生成以及明确结构化的人在回路工作流。我们介绍了一个工作原型，用户可以在其中准备或输入用户故事，使用多个大语言模型生成验收标准，可选择利用检索到的需求相关示例增强生成，逐项比较输出、检查检索上下文，并在明确提交经评审者最终确定的版本之前，选择、编辑或重新生成候选标准。系统还将模型输出、用户干预、检索设置和最终标准记录下来，作为记录化决策流程的一部分。本文的贡献并非声称生成准确性已经得到提升，而是将验证落实为一项明确、可检查且可问责的流程，用于验证人工智能生成的需求制品。

关键词： 需求工程 · 大语言模型 · 检索增强生成 · 人在回路验证 · 验收标准 · 多模型生成

1 引言

大语言模型（LLM）在软件工程领域的应用迅速扩展，涵盖从代码生成到分析和文档编写等方面 [9]。将大语言模型融入需求工程流程，为应对需求获取这一长期存在的挑战提供了颇具前景的路线图，

以及分析、规格说明和验证等方面 [4]。在需求工程领域，研究人员已将大语言模型作为工具，用于辅助创建需求、用户故事、验收标准及其他早期产出物 [6]。然而，这些能力也引发了人们对人工智能生成的规格说明的正确性和可靠性的疑问。

先前研究探索了面向需求生成的任务特定指令调优 [7]。然而，能够生成看似合理的需求相关文本，并不能证明其对于特定项目的正确性。验收标准描述了用于评估预期系统行为是否满足某项需求或用户故事的条件。因此，错误、不完整或缺乏依据的验收标准可能会影响后续实现和测试活动。

因此，核心挑战不仅在于人工智能能否生成验收标准，还在于如何以透明、负责且实用的方式验证这些标准。在本研究中，验证被视为一项首要活动，而不是生成后非正式的修正阶段。所提出的方法结合了可选的检索增强、多模型比较以及明确的人工批准，以支持在最终确定之前检查和修订人工智能生成的验收标准。通过展示检索到的上下文并记录人工决策，该工作流使生成和验证过程更加透明，也更便于后续分析。这种定位对于需求工程尤为重要，因为早期的规格说明产物往往需要解读、协商和结合情境进行判断，而非仅依赖形式化证明。

本研究延续了我们在需求工程中开展的人机协作研究。在早期工作中，我们考察了人工智能如何重塑需求工程实践，并指出了围绕歧义、可追溯性、透明性以及可信的人机协作等方面持续存在的问题 [1]。随后，基于对现有文献的分析，我们提出了人机需求工程协同模型（HARE-SM），将人工智能支持的需求工程界定为一种需要人工监督、可解释性和反馈循环的协作过程 [2]。本文从更为宏观的概念框架转向面向具体验证的原型。具体而言，本文聚焦于AI生成的验收标准，并探讨检索 grounding、多模型比较以及明确的人工批准确认如何在生成的标准成为最终版本之前支持验证。

本文有三项贡献：

– 提出了一种混合工作流，将检索增强生成（RAG）与明确的人在回路（HITL）验证相结合，从用户故事生成并验证验收标准。– 展示了一个可运行的多模型原型，支持可选检索、并排比较不同语言模型生成的验收标准，以及通过结构化用户操作来接受、编辑或重新生成输出。– 引入了一种日志设计，用于记录生成的输出、用户干预、检索设置和审核者最终确定的标准，从而将

验证转变为可分析流程，而非非正式的事后纠正步骤。

这些贡献使该原型成为研究如何以透明、负责的方式验证AI生成的需求制品的实验平台，而非一个已完成的准确性解决方案。

2 背景与动机

2.1 人工智能对需求工程的支持

自然语言处理（NLP）已广泛应用于需求工程，用于支持分类、可追溯性和质量分析等任务 [15]。大语言模型的最新进展推动了人工智能辅助的需求生成，近期综述也记录了人们对大语言模型支持的需求工程活动日益增长的兴趣 [14]。与此同时，近期研究指出，大语言模型支持的需求工程存在局限与风险，包括幻觉内容、可解释性有限，以及评估生成制品方面的挑战 [6]。不同于传统的基于自然语言处理的方法（此类方法倾向于对已有需求中的特征进行分类或检测），生成式模型会产生新的文本制品。这些制品可能读起来流畅、看似正确，但实际上缺乏依据，甚至可能与项目的其他方面相冲突。这一点对于验收标准尤为重要。验收标准旨在使满足某项需求或用户故事的条件更加明确且可验证。然而，生成的标准仍可能遗漏重要约束，或引入缺乏依据的假设。因此，需要进行人工验证，以评估这些标准是否符合预期需求及相关项目上下文。

因此，不应将大语言模型在需求工程中的应用仅理解为生成问题。这同样也是一个验证问题。生成的制品必须在正确性、依据性、可追溯性以及与利益相关者意图的一致性方面经过检查，之后才能将其视为可靠的需求制品。

2.2 从生成到验证

近期关于生成式人工智能用于需求工程的研究表明，大语言模型正越来越多地应用于语言密集型的需求工程任务，但同时也凸显了与可解释性、可复现性、可控性和评估相关的担忧 [6]。这些担忧十分重要，因为需求制品不仅是技术描述，也反映了利益相关者需求、领域假设和项目决策。生成的需求或验收标准可能在语法上结构良好，但从项目的角度来看仍然是错误的。

先前的研究也探讨了如何根据用户故事生成验收标准。例如，Li 等人提出了 SimAC，这是一种利用基于大语言模型的模拟敏捷协作来生成验收标准的方法 [11]。SimAC 等方法主要关注验收标准的生成，而我们的工作则着重于后续的人工验证过程。

具体而言，我们在一个集成化工作流中实现了多模型比较、检视检索上下文、审阅者干预以及明确的最终确定。在此语境下，验证意味着应依据用户故事、可用的项目上下文和人的判断来审查候选标准。这一立场与需求工程的更广泛观点一致：验证通常依赖利益相关者评估、解读和达成共识，而不仅仅是形式化检查 [12]。

2.3 微调与检索验证

微调可以使模型行为适应特定任务或领域。先前的研究探索了用于需求生成的任务特定指令调优 [7]。然而，微调本身并不会揭示哪些上下文来源支持某一条生成的标准。检索则提供了另一种机制，通过将外部信息作为生成上下文的一部分来实现这一点 [10]。这些方法并不互相排斥。

在当前原型中，选择检索是为了让生成过程中提供的上下文信息可供人工检查。这一设计使审阅者能够在评估候选验收标准时检查检索到的材料。然而，检索上下文的可用性本身并不能证明生成的标准是正确的，或得到特定项目需求的支持。

2.4 检索增强生成与可追溯性

检索增强生成（RAG）提供了一种实用方式，可以在不改变模型权重的情况下，将项目特定信息纳入大语言模型提示词 [10]。在需求工程中，这一点很有用，因为需求制品通常需要结合其他文档进行解读，例如项目描述、规格说明、利益相关者笔记或既有需求。通过检索并呈现相关上下文，RAG可以在验证过程中支持依据性和可追溯性。

近期与需求工程相关的研究已探索将检索增强生成用于多种任务，例如根据自然语言需求生成测试场景，以及恢复可追溯性链接 [5,8]。这些研究表明，基于检索的方法有助于将生成或分析的制品与作为支撑的文本证据联系起来。本研究沿着这一方向展开，但专门聚焦于验收标准，以及围绕人工智能生成标准的人工验证过程。

2.5 人在回路验证与可解释性

在需求工程中，人工审阅者必须继续负责批准需求内容。因此，人在回路工作流十分重要，因为它们使人工审阅者持续参与对生成的工件是否充分进行判断。在人工智能支持的工作中，可解释性、用户确认和迭代优化

是接受和负责任使用 [13]的重要因素。通用的人机交互指南还强调，应支持用户纠正问题，并对人工智能辅助活动进行适当控制 [3]。

在我们的语境中，HITL被设计为验证工作流的核心组成部分，而不是非正式的纠正阶段。选择多个模型时，用户可以比较输出结果、检查可用的检索上下文，并选择或修改候选验收标准。最终结果由审阅者明确提交，相应的交互事件也会被记录。我们早期的HARE-SM框架同样强调，人工智能支持的需求工程应将自动化与人工监督、可解释性和迭代反馈相结合 [2]。

3 所提出方法概述

该流程始于一个用户故事，或始于一个被转换为用户故事的非正式需求。该原型支持三种输入模式：用户可以粘贴现有的用户故事，通过引导式表单构建用户故事，或提供一项纯文本需求，由系统将其转换为候选用户故事。这种设计能够支持需求工程专业知识水平不同的参与者，同时保持下游验证工作流的一致性。

用户可以选择启用检索，使系统能够从需求相关示例集合中检索相关材料。检索到的文本使用TF–IDF进行表示，并按照余弦相似度进行排序。启用检索后，排名最高的文本块会被添加到所选模型的生成提示中，这遵循了检索增强生成的一般原则 [10]。

生成的验收标准随后会显示给用户。系统不会假定自动生成的输出是正确的。相反，用户会比较输出，在可用时检查检索上下文，选择或编辑相关标准，并明确决定接受、编辑还是重新生成生成的内容。所有用户决策都会被明确记录，在用户批准之前，任何产物都不会被视为最终版本。

图 1 展示了端到端架构，结合了两个方面：**（i）可选的检索**需求相关示例，以及**（ii）通过界面进行的人机协同验证**。

4 方法：面向需求工程的检索增强生成

4.1 语料准备与分块

检索语料库由跨领域的示例用户故事及相关验收标准材料组成。文本被划分为约 200 个 FLAN-T5 词元的文本块。这些文本块经过索引以支持稀疏检索，并在启用检索时用作补充上下文。该语料库提供的是与需求相关的示例，而非针对每位参与者软件项目的权威文档。

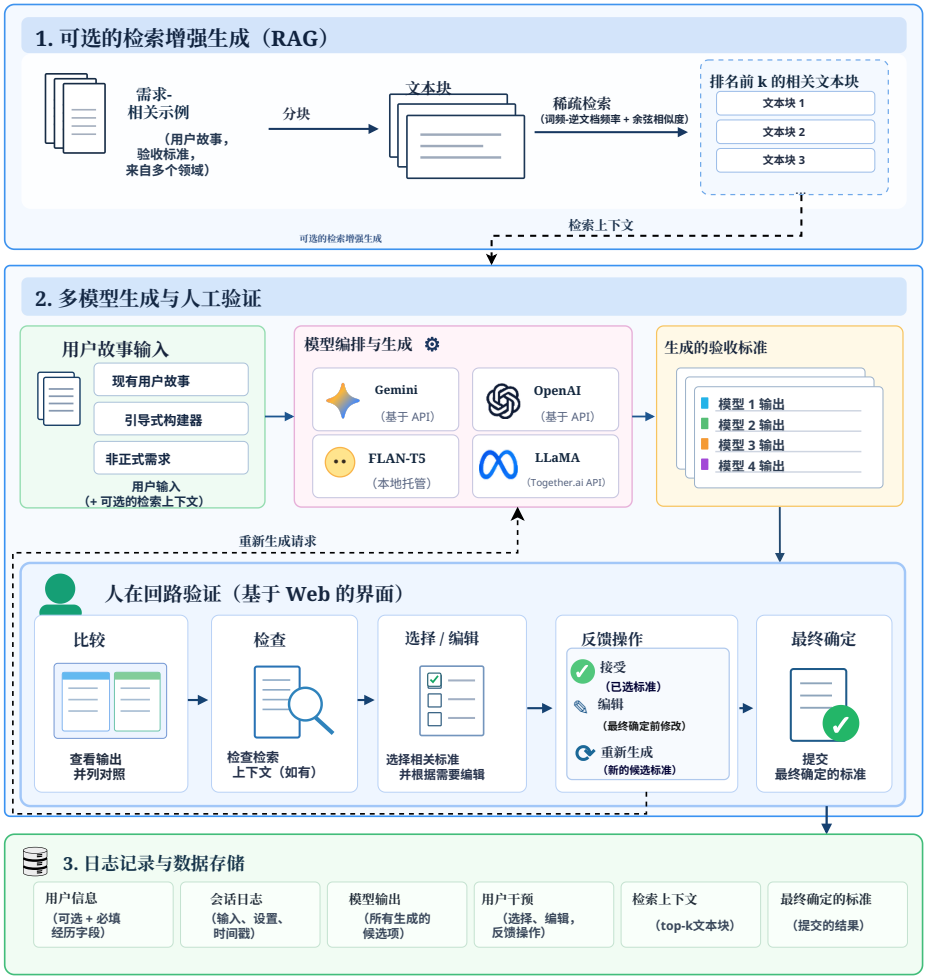


图1：验收标准助手的架构，展示了可选的检索增强、多模型生成、人工验证和交互日志记录。

4.2 使用词频-逆文档频率和余弦相似度的稀疏检索

在检索方面，我们采用轻量级的稀疏 TF–IDF 向量化方法，并通过余弦相似度对候选文本块进行排序。该模型是一个简单且易于解释的早期原型，可帮助理解检索行为，且无需任何嵌入基础设施支持。启用检索时，检索器会返回top-k文本块，用于扩充生成提示。

4.3 提示词增强与可追溯性

检索到的块会作为独立的上下文块插入生成提示中，同时保留用户故事作为主要输入。这样，在验证过程中，模型获得的上下文材料便可供检查。检索到的材料为生成提供了额外上下文，但其本身并不能证明生成的标准正确，也不能证明其符合特定项目的需求。

该检索机制提供了透明的基线，使审查者能够检查生成过程中提供的上下文示例。这种可见性有助于审查检索溯源和上下文相关性，但无法建立标准级别的可追溯性，将其关联到权威的项目需求，也无法独立验证生成的验收标准是否正确。稀疏检索方法可能无法捕捉使用不同术语表达、但语义相关的内容。定长词元块也可能将相关约束分隔在块边界两侧，从而难以检索到验证所需的完整上下文。因此，当前的检索配置应被视为原型基线，而非最终的检索方案。未来工作可以探索基于嵌入的检索方法，同时保留相同的验证工作流，并继续强调上下文的可检查性。

启用检索增强生成后，界面允许用户在验证过程中检查检索上下文。此功能旨在使依据性变得可见，而不是将检索视为隐藏的提示工程步骤。通过展示上下文材料，系统帮助用户判断生成的验收标准是否得到检索到的上下文示例支持。如果某个模型未启用检索增强生成，界面会明确指出未提供检索上下文。

4.4 多模型生成策略

为减少对单一模型的依赖并支持比较审查，该助手集成了四个模型：

– FLAN-T5-Large （本地推理）–
Gemini 2.5 Flash （API）– **OpenAI**
gpt-4o-mini（API）– **LLaMA-3.3** 通过
Together.ai（API）

该工具在各模型之间强制采用一致的提示词结构和输出格式化方式，以支持并排比较。

5 系统架构与实现

本文所述原型代表了为研究所提出的验证工作流而开发的实现；其组件和配置可能会在后续实证研究中进一步完善。系统架构

支持对多个大语言模型进行编排，包括本地托管模型和基于 API 的模型。通用提示生成器可为不同的大语言模型提供一致的指令，并且可以独立启用或禁用所选模型的检索功能。该系统允许审阅者比较所选大语言模型的输出，并在人工参与的验证过程中请求重新生成候选标准。

检索模块与模型无关，并采用轻量、可追溯的技术。作为验证过程的一部分，可以检查检索到的文档、提示和用户操作。反馈和最终决策与模型输出分开存储，以便对人工操作和模型生成的产物进行综合分析。

5.1 基于 Streamlit 的 Web 应用程序

该原型实现为基于 Streamlit 的 Web 应用程序。图 2 和图 3 展示了一个两阶段工作流界面。第一阶段支持用户故事准备、模型和检索配置，以及并排生成验收标准。第二阶段支持人工验证，包括标准选择、对检索上下文的可选检查、对生成标准的反馈，以及最终提交达成一致的版本。

用户故事准备阶段支持三种输入模式。用户可以粘贴现有的用户故事，通过引导式表单构建用户故事，或提供由系统转换为候选用户故事的非正式需求。该设计支持具备不同需求工程专业水平的参与者，同时保持下游验证工作流的一致性。在生成验收标准之前，系统会将准备好的用户故事展示给用户进行检查，并在支持的情况下进行修改。随后，用户选择一个或多个模型，并可选择是否启用检索增强。

图 2 展示了原型界面的第一阶段。此阶段涵盖用户故事准备、模型与检索配置，以及并排生成验收标准。此阶段旨在准备清晰的输入产物，并在验证开始前向用户展示不同的模型输出。

图 2 中标注的元素包括：(A) 会话元数据；(B) 用户故事输入模式，包括现有故事、引导式构建器和需求转换；(C) 用户故事输入；(D) 用于生成的最终故事；(E) 模型与 RAG 选择；以及 (F) 并排展示的模型输出。

生成完成后，工作流从模型比较转向人工验证。生成的验收标准不会被视为最终需求，而是作为暂定候选项呈现，用户必须先进行检查、选择、编辑或重新生成，然后才能提交达成一致的版本。

图 3 中的标注元素展示了（G）标准选择、（H）可选的 RAG上下文检查、（I）通过接受、编辑或重新生成进行反馈操作、（J）审核者最终确定的验收标准，以及（K）用于存储和后续分析的最终提交。

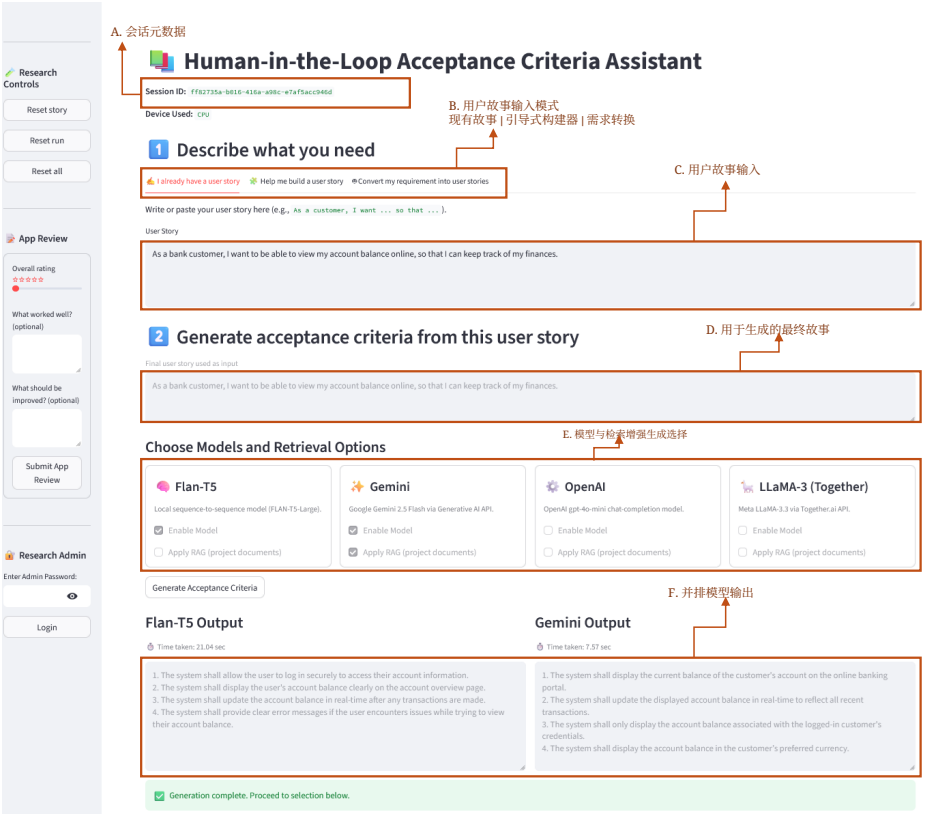


图 2：验收标准助手的生成与模型比较阶段。

图 2 和图 3 共同呈现了一个连续的验证工作流。用户首先准备或选择一个用户故事，配置模型和检索设置，并比较生成的输出。随后，用户通过选择相关项目、在可用时检查检索上下文、记录接受、编辑或重新生成操作，并提交最终一致标准，来验证生成的标准。这种设计使验证过程明确，而不是将人工智能生成的标准视为自动接受。

图 2 和图 3 中的示例展示了一个具有单一主要用户目标的银行业务用户故事。该示例用于说明生成、比较和人工验证工作流，而非界定助手所支持需求的复杂程度。该原型并不限于此示例，因为用户可以提供自己的用户故事或非正式需求，并使用多个大语言模型生成候选验收标准。然而，本研究并未系统地按复杂程度对用户故事进行分类，也未评估复杂程度如何影响生成标准的质量以及所需的

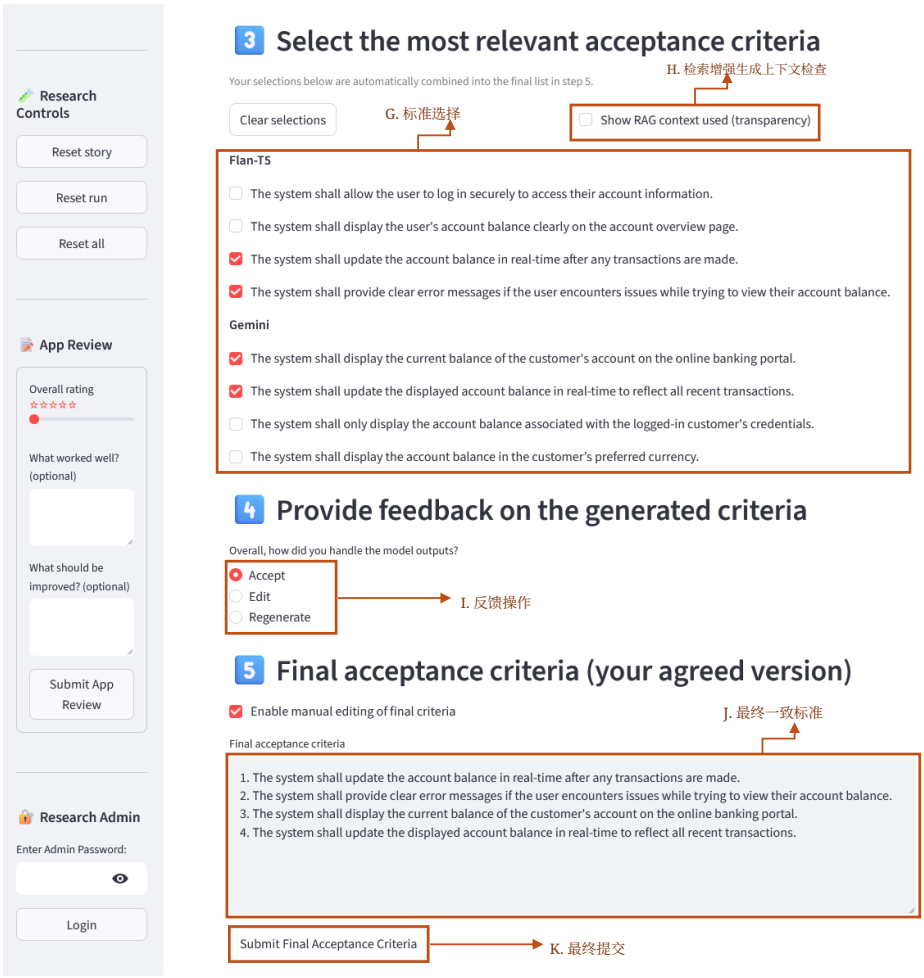


图 3：验收标准助手的人工验证与最终确定阶段。

人工验证工作量。未来的评估可以考察涉及多个条件、异常场景、依赖关系和领域特定约束的用户故事。

该界面旨在同时支持比较和验证。模型输出会并排显示，以便审阅者比较不同方案，而不是依赖单一的生成响应。随后，审阅者可以选择相关标准、根据需要进行编辑，或请求重新生成。该工作流会记录已选标准、反馈操作以及最终提交的结果，以供后续分析。

5.2 模型编排层

模型编排层会为每个已选模型构建一致的提示词，在启用时附加检索上下文，分发推理请求，收集响应，并统一输出格式，以便进行比较。研究配置会在生成过程开始时冻结，因此可以在后续分析中重新构建已选模型、检索设置、提示词版本和词元设置。

5.3 检索模块

检索模块包括文档存储、分块流水线、词频-逆文档频率向量化、余弦相似度排序、top-k检索以及提示词上下文构建。该模块与模型层分离，因此未来可以引入新的检索策略，而无需更改整体人工验证工作流。

5.4 日志记录与数据采集

该原型会记录生成的输出和经过检测的人工验证事件。除生成的提示和输出外，存储的数据还包括会话ID、时间戳数据、设备信息、冻结的研究配置、参与者提交的用户故事、所用模型、检索配置、每个模型生成的输出、所选标准、任何经过编辑的输出、最终确定的验收标准，以及包含交互事件信息的事件日志。事件日志数据可用于跟踪每次运行中的关键验证活动，其中记录了配置锁定、启动和完成生成、选择更新、编辑后输出的更改以及最终提交事件。最终提交内容会保存到私有研究仓库，以供后续分析。

日志设计区分了审阅者的中间交互与最终提交的成果。选择变更和文本修订描述了候选标准在审阅过程中如何被修改，而最终提交则标识审阅者定稿结果。这一区分使后续分析能够将验证过程与其最终结果分开考察。系统还包括一个可选的应用评审表，用于收集评分以及关于哪些方面运行良好、哪些方面需要改进的开放式反馈。这些应用层面的评审并非验收标准验证数据，但为改进研究原型提供了有价值的可用性反馈。

在使用助手之前，参与者填写了研究参与表，并为收集研究数据提供了肯定性同意。姓名、电子邮件地址和所属机构为可选项，而自我报告的 IT 和需求工程经验信息为必填项。

参与记录与验证日志分开存储，但仍可通过会话标识符进行关联。收集的研究数据存储在私有的访问受控仓库中。汇总报告交互模式时不包含身份识别信息。

这些日志记录有助于未来研究基于检索的验证的各个方面，包括接受率、编辑行为、最终决策所需的时间、不同模型之间的一致程度，以及参与者背景与验证操作之间的关系。

6 人在回路验证工作流

该工具将人在回路验证实施为结构化决策流程：

1. **准备：** 输入现有用户故事、使用引导式表单创建用户故事，或将非结构化需求转换为用户故事。2. **生成：** 基于一个或多个选定模型生成验收标准，并可选择是否为每个模型启用检索支持。3. **比较：**并排显示输出结果，以支持交叉核查。4. **检查：** 启用检索增强生成（RAG）时显示检索上下文，使用户能够检查提供给模型的证据。5. **决策：** 用户可以选择三种界面操作之一：**– 接受：**保留当前选定的验收标准，无需进行额外的手动文本编辑；**– 编辑：**在最终确定之前手动修改标准；**– 重新生成：** 请求新的候选标准以供进一步审查。6. **最终确定：**

p生成并存储已 g达成的验收 p收标准。

7. **日志：** 记录交互轨迹以供分析。

该工作流会记录已选标准、反馈状态和最终提交。这些记录使审阅者干预情况清晰可见，便于后续分析。最终的“接受”状态并不一定意味着所有生成的标准都被保留，因为审阅者可能只在提交前选择了其中一部分。本文所说的批准确认，指的是记录审阅者所选结果的最终提交。它使最终确定决策变得可观察，但不能据此独立确认标准的正确性，也不能证明所有相关利益相关者达成一致。

当前原型将不同模型面板中的标准分别呈现，以保留其来源溯源信息并支持比较。不过，不同模型可能生成描述相似预期行为的标准，从而增加审阅者需要检查的文本量。例如，图 3 展示了从两个模型面板中选出的、涉及交易后账户余额更新的相似标准。图 3 中的示例展示的是当前的选择与定稿流程，而非经过独立评估的一组高质量验收标准。审阅者定稿结果中保留相似标准，也表明最终提交不一定会消除冗余或歧义。这些质量方面需要单独评估，也促使我们进一步研究人工验证行为。

一种可能的扩展方案是，在保留每个变体的原始措辞和来源面板信息的同时，对模型面板中的潜在相似标准进行分组。界面可以为每个组显示一个代表性标准

同时允许审阅者检查并选择单个备选项。此类分组不应自动合并或丢弃标准，因为条件、约束或预期行为方面的细微差异可能十分重要。审阅者仍应负责选择和修订最终标准。分组能否在不掩盖有意义差异的情况下减少审查工作量，则需要通过实证评估加以验证。

7 初步观察与计划评估

该原型表明，所提出的验证工作流可以实现为可运行的交互系统，并能系统地收集后续实证分析所需的信息。该系统支持用户故事准备、可选的检索 grounding、多模型生成、人工选择和编辑验收标准、明确的反馈操作，以及最终提交商定的标准。因此，当前工作的贡献并非完成了对模型质量的评估，而是提供了一个带有观测功能的平台，用于研究人工智能生成的需求制品的验证过程。

该实现展现了验证工作流的若干设计特征。三种输入模式支持以不同方式准备用户故事。并排模型输出使审阅者能够在最终确定之前比较不同的标准。启用检索上下文后，审阅者可以检查这些内容，从而查看生成过程中提供的信息。按标准进行选择、编辑、反馈状态设置以及最终提交，均可形成可供后续分析的交互记录。

未来的实证评估可考察验证工作流程的四个主要维度。第一是干预频率，即在不同模型和检索设置下，用户接受、编辑或重新生成输出的频率。第二是人工编辑的性质，包括删除缺乏依据的标准、补充缺失的约束、澄清含糊表述，以及改写为更具可测试性的标准。第三是验证工作量，包括已选标准的数量、手动编辑的次数，以及达成最终一致版本所需的时间或交互步骤。第四是模型分歧，尤其是模型间的分歧是否预示着更高的人工修订率。

因此，我们并未声称该原型已经提高了正确性或消除了幻觉。相反，它提供了工作流程和数据收集基础设施，以便未来的研究能够系统地探究这些问题。

8 讨论与验证启示

所提出的方法体现了对需求工程中验证与验证的广义理解。验收标准并非可执行代码，但由于它们定义了

预期的系统行为，因此会强烈影响下游开发和测试。当此类标准由人工智能生成时，其语言流畅性并不能保证正确性、完整性或与利益相关者意图的一致性。因此，验证必须包括上下文 grounding、人工审查，以及明确决定生成的文本是否可接受。

检索通过在生成过程中提供额外的上下文材料，并使这些材料可供人类检查，从而为所提出的工作流作出贡献。审阅者在评估生成的标准时，可以检查检索上下文。然而，不应将检索到的示例视为证明项目特定正确性的权威证据。因此，当前实现支持对生成上下文进行检查，而不是自动验证生成的验收标准。

人在回路工作流构成了验证过程的第二个组成部分。模型生成的标准仅被视为候选标准，在最终确定之前，用户需要对其进行选择、编辑、重新生成或批准确认。这一批准确认阶段是系统的核心，因为验证并未被隐藏为非正式的纠正活动，而是被表示为一个可观察的决策流程。

多模型工作流同样有助于验证。不同模型可能会强调同一用户故事的不同方面，其输出还可能揭示歧义、上下文不足或过度具体化的可能性。因此，模型生成的标准之间的差异不仅是一个麻烦，也是一项有用的指标，表明可能需要进行更仔细的人为检查。

总体而言，所提出的系统表明，需求工程中值得信赖的人工智能支持，并不只取决于通过更强大的模型生成更好的标准。它还需要相应的接口和日志机制，使生成的工件可供检查、明确呈现人的决策，并能够分析验证行为。所提出的方法并非通过数学证明或自动化测试来验证生成的工件，而是通过检索 grounding 和人工评估来支持验证。这是合适的，因为需求工程也依赖评估、解释和利益相关者验证 [12]。

9 结论与未来工作

本文提出了一种面向验证的需求工程中人工智能生成的验收标准方法。本文的核心思想不仅是根据用户故事生成验收标准，还在于支持在将其接受为最终需求制品之前对其进行验证。为此，所提出的方法结合了基于检索的 grounding、多模型生成和人在回路验证。

该原型展示了这一理念如何在实践中实现。该系统允许用户准备或输入用户故事，选择一个或多个模型，可选启用检索增强，并排比较生成的验收标准、检查检索上下文，然后接受、编辑或重新生成生成的标准。审核者最终确定的验收标准将被存储

并与交互数据一同保存，从而可以分析人类如何与人工智能生成的需求制品进行交互。

本研究的贡献在于提供了一个可运行且经过工具化的验证工作流，而不是对模型准确性进行实证评估。由于流畅的 AI生成标准仍可能缺乏依据、不完整或属于幻觉生成，该方法使检索到的上下文可供检查，并要求进行人工审查和明确的最终确定。

未来的研究将重点关注所收集数据的实证分析。具体而言，我们计划考察检索配置、模型选择和用户故事复杂度如何影响人工验证决策。我们还计划分析用户所做编辑的类型、接受、编辑和重新生成操作的频率，以及检索增强是否会影响缺乏依据的验收标准的出现，并采用独立评估来衡量生成的需求质量。另一个研究方向是调查由不同模型生成的相似标准的保留溯源的分组。未来的研究应考察，这种分组能否在保留候选标准之间重要差异的同时，减少审查工作量。

另一个方向是将原型从验证生成的验收标准扩展到识别相关软件工件中的需求缺口。除了检查生成的标准是否有依据且可接受之外，未来的版本还可以支持面向缺口的分析，包括缺失需求、隐藏假设、不一致性、决策空白、可追溯性缺口和利益相关者沉默缺口。这将拓展系统的作用，使其从验收标准验证走向更一般的人工智能辅助需求质量分析。

致谢。 本研究得到了FAST（芬兰软件工程博士研究网络）的支持，该网络由芬兰教育与文化部资助。同时，谨此感谢Business Finland提供的支持（项目ANSE（1822/31/2025）和GENIUS（2545/31/2024））。

参考文献

1. Abbasi, M.A., Ihantola, P., Mikkonen, T., Mäkitalo, N.：重新审视需求工程：AI原生软件开发中的人机协作。载于：Euromicro软件工程与高级应用会议。第 164 180 2025 10 1007 978 3 032 04190 6_11 –页。2 Springer（ ）。https://doi.org/ . / - - - - 。Abbasi, M.A., Ihantola, P., Mikkonen, T., Mäkitalo, N.：迈向需求工程中的人机协同：一个框架与初步研究。载于：2025年第六届智能数据科学技术与应用国际会议（IDSTA）。第81–88页。IEEE（2025）。https://doi.org/10.1109/IDSTA66210.2025. 11202850 3. Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P.N., Inkpen, K., Teevan, J., Kikin-Gil, R., Horvitz, E.：人机交互指南。载于：2019年CHI计算机系统中的人为因素会议论文集。第1–13页。ACM（2019）。https://doi.org/10.1145/3290605.3300233

4. Arora, C., Grundy, J., Abdelrazek, M.：借助生成式人工智能推进需求工程：评估大语言模型的作用。载于：《生成式人工智能助力高效软件开发》，第129–148页。Springer（2024）。https://doi.org/10.1007/ 978-3-031-55642-5_65. Arora, C., Herda, T., Homm, V.：使用检索增强大语言模型从自然语言需求生成测试场景：一项工业研究。载于：2024年IEEE第32届国际需求工程会议（RE）。第240–251页。IEEE（2024）。https://doi.org/10 00031 6 1109 59067 2024 . /RE . . . Cheng, H., Husen, J.H., Lu, Y., Racharak, T., Yoshioka, N., Ubayashi, N., Washizaki, H.：需求工程中的生成式人工智能：系统文献综述。《软件：实践与经验》**56**(2)，141–170（2026）。https: //doi.org/10 70029 7 1002. /spe. . Habib, M.K., Graziotin, D., Wagner, S.：ReqBrain：面向AI辅助的需求生成的大语言模型任务特定指令调优。arXiv预印本 arXiv:2505.17632（2025）8. Hey, T., Fuchß, D., Keim, J., Koziolek, A.：通过检索增强生成实现需求可追溯性链接恢复。载于：需求工程国际工作会议：软件质量基础。第381–397页。Springer（2025 10 1007 978 3 031 88531 0_27）。https://doi.org/ . / -9 - - -。Hou, X., Zhao, Y., Liu, Y., Yang, Z., Wang, K., Li, L., Luo, X., Lo, D., Grundy, J., Wang, H.：面向软件工程的大语言模型：系统文献综述。ACM《软件工程与方法学汇刊》**33**(8)，1–79（2024 3695988 10 10 1145）。https://doi.org/ . / 。Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T.等：面向知识密集型自然语言处理任务的检索增强生成。《神经信息处理系统进展》**33**，9459–9474（2020）11. Li, Y., Keung, J., Yang, Z., Ma, X., Zhang, J., Liu, S.：SimAC：在用户故事细化中模拟敏捷协作以生成验收标准。《自动化软件工程》**31**(2)，55（2024）。https://doi.org/10.1007/ s10515-024-00448-7 12. Nuseibeh, B., Easterbrook, S.：需求工程：路线图。载于：软件工程未来会议论文集。第35–46页（2000）。https://doi.org/10 336523 13 1145 336512. / . 。Theis, S., Jentzsch, S., Deligiannaki, F., Berro, C., Raulf, A.P., Bruder, C.：协同工作中人工智能可解释性与接受的需求。载于：人机交互国际会议。第355–380页。Springer（2023 10 1007 978 3 031 35891 3_22）。14 https://doi.org/ . / - - - -。Zadenoori, M.A., Dąbrowski, J., Alhoshan, W., Zhao, L., Ferrari, A.：面向需求工程（RE）的大语言模型（LLMs）：系统文献综述。arXiv预印本 arXiv:2509.11446（2025）15. Zhao, L., Alhoshan, W., Ferrari, A., Letsholo, K.J., Ajagbe, M.A., Chioasca, E.V., Batista-Navarro, R.T.：面向需求工程的自然语言处理：系统映射研究。ACM计算综述（CSUR）**54**(3)，1–41（2021）。https://doi.org/10.1145/3444689