arXiv:2611.30778v1 [cs.CE] 23 Nov 2026

质量工程师需要了解的生成式人工智能：结构化数据提取

Fadel M. Megahed[1], Ying-Ju Chen[2], Yamin Dahwich[3], Zhao-Wei Chen[4], Lucas K. Hung[5], Arthur Carvalho[1],L. Allison Jones-Farmer[1], Ibrahim Yousif[6], 和 Inez M. Zwetsloot[7],*

[1]Farmer商学院，迈阿密大学，牛津，OH 45056，美国[2]文理学院，代顿大学，代顿，OH 45469，美国[3]三一学院，剑桥大学，剑桥 CB2 1TQ，英国[4]工商管理学院，代顿大学，OH 45469，美国[5]工程学院，代顿大学，代顿，OH 45469，美国

[6]应用计算与工程技术系，迈阿密大学，Hamilton，OH 45011，美国

[7]荷兰阿姆斯特丹阿姆斯特丹大学阿姆斯特丹商学院*通讯作者。电子邮件：i.m.zwetsloot@uva.nl

摘要

在这篇说明性论文中，我们将介绍质量工程师如何使用大语言模型（LLM）和视觉语言模型（VLM）进行结构化数据提取。我们将介绍相关概念，从测量系统的角度讨论结构化提取，并演示如何使用 R语言和 Python 实现。随后，我们将介绍两个免费提供的基于网页的应用程序（应用）：（a）从汽车召回通知中提取信息；（b）监测 arXiv 上的质量工程研究。为确保人工智能驱动的结构化提取结果准确无误，我们请人工评估者对结果进行验证。在第一个应用中，四位审阅者使用两个模型评估了从 30 份召回通知中提取的信息；在 720 项字段级评分中，有 98.6% 被判定为完全正确。在第二个应用中，21 位作者评估了从其 45 篇 arXiv 论文中提取的信息。所有论文摘要的评分均为基本正确或正确，

不过，评估也发现部分分类字段和数学表达式存在问题。这些结果展现了结构化数据提取的实际应用潜力，同时强调了验证提取信息的必要性。我们公开提供这两个应用及其源代码，供读者调整这一方法，以用于自己的应用场景。

关键词：数据质量；生成式人工智能；JSON 模式；大语言模型；结构化数据提取；非结构化数据；视觉语言模型

1 引言

保修索赔和客户投诉通常通过自由文本描述产品故障，而不是将其表示为可直接分析的变量。这是质量工程中日益庞大且复杂的数据源所带来的实际挑战之一（Hong、Zhang 和 Meeker 2018年；Megahed 和 Jones-Farmer 2015年；Weese 等人 2016年）。例如，一项保修索赔可能会指出受影响的部件、描述故障，并记录所采取的服务措施。要识别最常见的失效模式，我们需要从每项索赔中提取并分类这些信息，将结果汇总为结构化数据集，再使用帕累托图进行总结。对数千项索赔逐一手动提取信息会非常耗时。此外，要确保编码一致，还需要明确的指导原则，尤其是在不同索赔中对同一故障的描述不尽相同的情况下。

以往研究曾使用文本挖掘和自然语言处理（NLP），从消费者投诉和汽车缺陷记录中提取并分类信息（Francis, Cates, and Caldiera2023年；Ghazizadeh, McDonald, and Lee 2014年；Uglanov 等 2024）。虽然这些方法可以减少人工编码，但需要制定提取规则，和/或针对特定任务训练自然语言处理模型。大语言模型（LLM）和视觉语言模型（VLM）提供了另一种进行结构化数据提取的方式。具体而言，用户可以指定感兴趣的字段，并用简明英语描述想要提取的内容

（或其选择的其他语言）。大语言模型/视觉语言模型会利用这些信息处理输入的记录，并以结构化格式提取定义的字段（OpenAI 2025）。

在这篇说明性论文中，我们展示质量工程师如何使用大语言模型和视觉语言模型进行结构化数据提取。我们首先介绍相关概念，并讨论这种方法可能适用的场景。随后，我们提供 R语言和 Python 中的实现示例，并介绍两个应用：(a) 从汽车召回通知中提取信息；(b) 监测 arXiv 上最新的质量工程研究。这两个应用均可在线免费使用，我们还报告了人工评估其提取输出的结果。

2 背景

2.1 结 构化数据提取作为测量系 m

使用人工智能进行结构化数据提取的过程依赖于大语言模型/视觉语言模型和输出模式，二者结合起来可以被概念化为用于提取变量的测量系统。在这一类比中，该测量系统可以接收以下输入：

（1）包含我们希望提取的数据的输入信息（例如，投诉叙述、扫描表格、PDF 或照片）；

(2) 一段简短的提示词，用于描述提取任务；

(3) 一个模式，其中列出要提取的字段、设定每个字段的数据类型，并用通俗语言定义每个字段；以及

(4) 还可以输入几个示例演练，为模型提供更好的上下文。随后，人工智能模型会生成一条结构化记录，其中包含按预先指定的数据类型提取的变量。对于单条记录，生成的输出现在是一个结构化数据集，可以附加到数据框行中。第3节对此流程进行了详细说明。

抽取系统可以从三个维度进行评估：(a) 正确性，即评估抽取的数据是否准确反映源文档；(b) 重复性，即衡量从同一文档中重复抽取是否会得到一致的结果；以及 (c)

可复现性，即确定使用不同的大语言模型/视觉语言模型时，结果是否一致。在我们的两个应用中，我们关注正确性，请人工审阅者根据源文档评估抽取的信息。尽管我们在应用 1 中比较了两个模型，但并未对重复性或可复现性开展正式研究。感兴趣的读者可参阅 Megahed等人（2026年b），了解基于大语言模型的文本分类应用中这些问题的详细论述。

2.2 对大语言模型和视觉语言模型的通俗介绍 s

大语言模型通常基于 Transformer 架构（Vaswani 等，2017）。它们在大量文本集合上进行预训练（Brown 等，2020），其中许多模型随后会经过微调，以遵循用户指令（Ouyang 等，2022）。对我们而言，一项重要能力是上下文学习：模型根据指令以及用户（可选）提供的少量示例来执行任务，而无需额外训练（Brown 等，2020）。这样，我们就可以用简明英语定义提取任务，而不必为每种应用单独开发模型。

质量工程记录通常以视觉形式包含信息，包括照片、检验图像和扫描文档。视觉语言模型（VLM）扩展了大语言模型的能力，使图像也能作为输入。例如，视觉语言模型可以从扫描的检验表单或基于图像的 PDF 文件中提取信息，也可以解读照片中的视觉信息。除了文本提取之外，相关的视觉语言方法也已用于制造业图像检验，通过产品的视觉特征而非图像中的文本进行检验（Megahed等人，2026年a）。关于大语言模型及其在统计学中的应用，读者可参阅 Ji 等人（2026年）。

2.3 结构化数据提取中质量工程师需要了解的术语

我们认为，读者的人工智能相关背景可能各不相同。因此，表1列出了本文中使用的人工智能术语。

表1：质量工程师需要了解的生成式人工智能术语。

Term	通俗易懂的含义	对数据提取的重要性
LLM	一种处理并生成文本的模型 以响应指令	能够从投诉叙述中提取信息 或报告
VLM	能够处理图像和 text	能够从扫描表单中提取信息， 照片或基于图像的 PDF
提示词	提供给模型的指令	规定应提取哪些信息
模式	字段名称、数据类型和描述 定义所需的输出	类似于包含操作性定义的编码表- 每个变量的定义
令牌/上下文窗口	词元是文本的单位；上下文窗 口限制了模型在一次请求中可以处理的词元数量 在一次请求中处理	影响一次可处理的信息量 以及相关成本
结构化 JSON 输出	包含以下内容的机器可读记录 模式中指定的字段	可转换为数据框中的一行
幻觉	由模型生成的不正确或不受支持的输出 由模型生成的不正确或不受支持的输出	提取的 值 may 需要验证 是否与源文档一致

大语言模型和视觉语言模型的一个局限性是，它们可能会生成源文档不支持的信息，这通常被称为幻觉 （Huang 等，2025年）。对于结构化数据提取，我们尤其关注提取出的值是否有源文档的支持（忠实性）（Huang 等，2025年）。这与认识论准确性密切相关，后者强调，生成正确结果取决于能否获取必要的证据并恰当地使用这些证据（Hamilton 等，2026年）。例如，如果一份维护报告指出电动机过热，但未确定原因，模型应将根本原因报告为未知，而不是假定是冷却系统故障导致的。

2.4 与质量工程的相关性

对结构化数据提取的关注，是工程和统计学领域更广泛转型的一部分。近期调查记录了生成式人工智能在设计、制造业和

质量控制（Nguyen和Kittur 2025年）中的快速普及；应用统计学领域的综述描述了统计方法与人工智能工具之间日益互补的关系（Min等人 2026年）；一组探讨统计工程未来的产业界—学术界—政府联合小组认为，该学科应将数据科学和人工智能作为核心能力加以拥抱（Anderson-Cook等人 2022年）。具体到质量工程领域，近期研究已开始将大语言模型和视觉语言模型与质量控制相结合（Megahed等人 2026a；Megahed等人 2024a；Megahed等人 2024b）。本文介绍这些模型的一种不同用例：从文档/图像中提取结构化信息，以支持后续分析。

当记录数量众多且文本内容繁杂、感兴趣的字段可以预先定义，并且可以将部分提取记录与源文档进行核对时，这种方法最为有用。当任务规模足够小、可以通过人工可靠地编写代码完成，简单的确定性规则即可满足需求，所需信息不在源文档中，或者错误可能造成严重后果且未计划审核流程时，这种方法的实用性较低。当文档包含专有信息或敏感信息时，组织还应考虑是否可以将其发送给外部模型提供商（参见第 6.2节）。

3 使用大语言模型/视觉语言模型进行结构化数据提取

图 1 概述了数据提取工作流。我们首先定义模式，其中规定要从文档中提取的变量。对于每个变量，我们提供字段名、其数据类型（例如字符串、整数或布尔值），以及以简明英语撰写的描述。然后，我们提供源文档，并要求模型将提取的信息作为 JSON 记录返回（OpenAI 2025年）。随后可以将这些记录合并为数据框，以便进行分析。请注意，这种方法可用于不同的输入格式，包括文本、PDF文件和图像（后者需要使用视觉语言模型）。

字段 `description` 让我们能够明确指定所需的信息。例如，从召回通知中提取制造商时，我们可能需要完整的

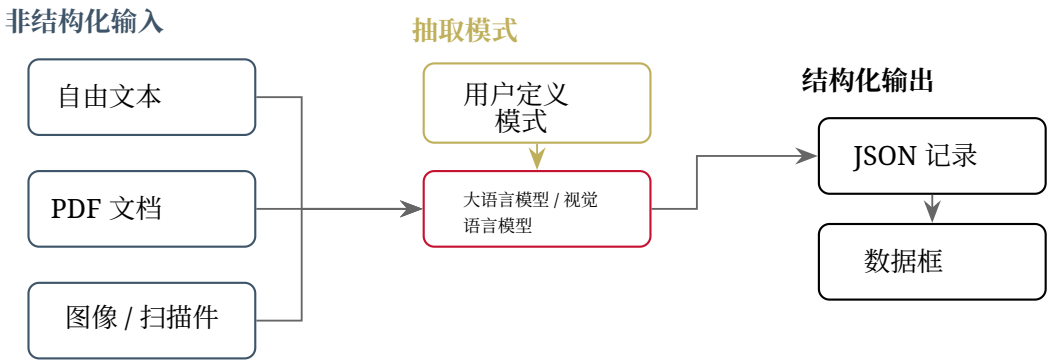


图 1：结构化数据提取工作流。用户指定描述待提取字段的类型化模式。大语言模型（对于图像和扫描文档输入，则使用视觉语言模型）返回符合模式的 JSON 记录，可直接映射到数据框中的一行。

公司名称、其缩写，或两者兼有。我们可以在描述中指定这一点，并在不更改提取代码的情况下进行修改。不过，格式正确的 JSON 记录并不一定意味着提取结果正确。模型可能会返回所有必需字段及相应的数据类型，但提取的信息仍然可能不正确。因此，我们在两个演示应用程序中都使用人工审阅者，将提取的值与其源文档进行核对。

代码清单 1 和 2分别展示了如何在 R 和 Python 中执行结构化提取。这两个示例都从一份召回通知中提取两个字段： `manufacturer` 和 `defect summary`。在 R 中，我们使用 `ellmer` 包（Wickham 等，2025），其中 `type object()` 用于定义模式，而 `chat structured()` 用于执行提取。在 Python 中，我们使用 `chatlas`（PositDevelopment Team 2026），它使用 Pydantic `BaseModel` 定义字段，并使用 `extract data()`执行提取。所列模型仅用于说明。选择模型时应考虑准确性、延迟、成本和隐私需求（Singh 等，2026年）。

请注意，这两个软件包都支持多个大语言模型提供商。例如，在 `ellmer` 中，我们可以将 `chat openai()` 替换为用于本地部署模型的 `chat ollama()`、用于 Claude 的 `chat anthropic()`，或用于 Gemini 的 `chat google gemini()`。切换提供商时，模式和提取调用保持不变。在下一节中，我们将利用这一功能，在相同的召回提取任务上比较两个模型。

列表 1：R语言中的结构化提取（`ellmer`）。

```
# --- 加载软件包 ---
library ( ellmer )


# --- 初始化提供商 ---
聊天   <- 聊天_ OpenAI (
   模型   =" gpt -6 - luna "
)

# --- 定义模式 ---
模式   <- 类型_ 对象 （
   制造商   =类型_ 字符串 （
      描述   =
      “ 公司          发布召回。”
   ) ,
   缺陷 _ 摘要   =类型_ 字符串 (
      描述   =
      " 一句话缺陷摘要。"
   )
)


# --- 提取结构化数据 ---
结果   <- 聊天 $chat _ 结构化 (
   召回     _text ,
   类型  =  模式
)
# 结果$ 制造商
# 结果$ 缺陷_ 摘要
```

代码清单 2：Python 中的结构化提取（`chatlas`）。

```
# --- 加载软件包 ---
from chatlas import ChatOpenAI
from pydantic import BaseModel ,
      Field
# --- 初始化提供方 ---
chat   =  ChatOpenAI (
      模型 =" gpt -6 - luna "
)

# --- 定义模式 ---
class RecallInfo ( BaseModel ):
      制造商 : str   =Field(
            描述 =
            " 公司          发布
                召回。"
      )
      缺陷_摘要 : str   =Field(
            描述 =
            " 一句话                缺陷
                摘要 . "
      )

# --- 提取结构化数据 ---
结果   =聊天 . 提取_数据 (
      召回     _text ,
      数据_模型   =  召回信息,
)
# 结果. 制造商
# 结果. 缺陷_摘要
```

4 应用 1：召回通知数据提取

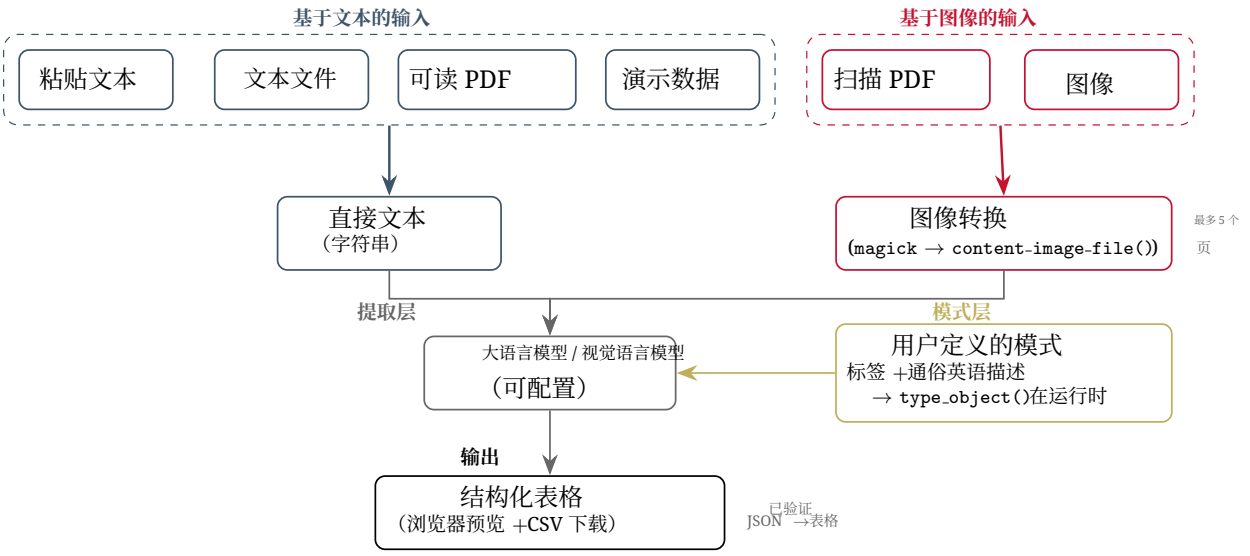
动机与实际需求。 联邦机构发布的召回通知，是制造企业所关注的非结构化数据的一个例子。在美国，产品安全召回十分普遍。例如，2025年，NHTSA发布了891项召回活动，影响超过2,900万辆车辆（美国国家公路交通安全管理局 2026年）。每次召回都包含对潜在缺陷的描述，而这些缺陷必须进行编码、分类和跟踪。在美国各行业中，*Sedgwick*产品安全与召回指数 记录了2026年1月至6月影响9.412亿件产品的1,634次召回（Sedgwick 2026年）。先前的研究表明

研究表明，专门的自然语言处理（NLP）技术能够从召回通知及相关记录中提取并分类缺陷（Francis、Cates 和 Caldiera 2023年；Uglanov 等 2024年）。然而，在实践中应用此类技术，需要许多中小型制造企业（例如二级供应商和三级供应商汽车供应商）可能并不具备的计算资源和专业知识。尽管如此，对于这些企业而言，召回通知能够提供有关与其所制造部件相似的部件故障的经验教训，从而为审查自身产品和流程中的潜在脆弱性提供参考。将这些通知转换为结构化数据，有助于企业识别反复出现的失效模式，优先开展预防性质量改进工作，同时减少手动审查和编码报告所耗费的时间。在此，我们演示如何利用大语言模型和简明英语解决这一问题。我们提供了一个免费的在线应用程序，可用于从非结构化文本中提取任意用户定义的字段。

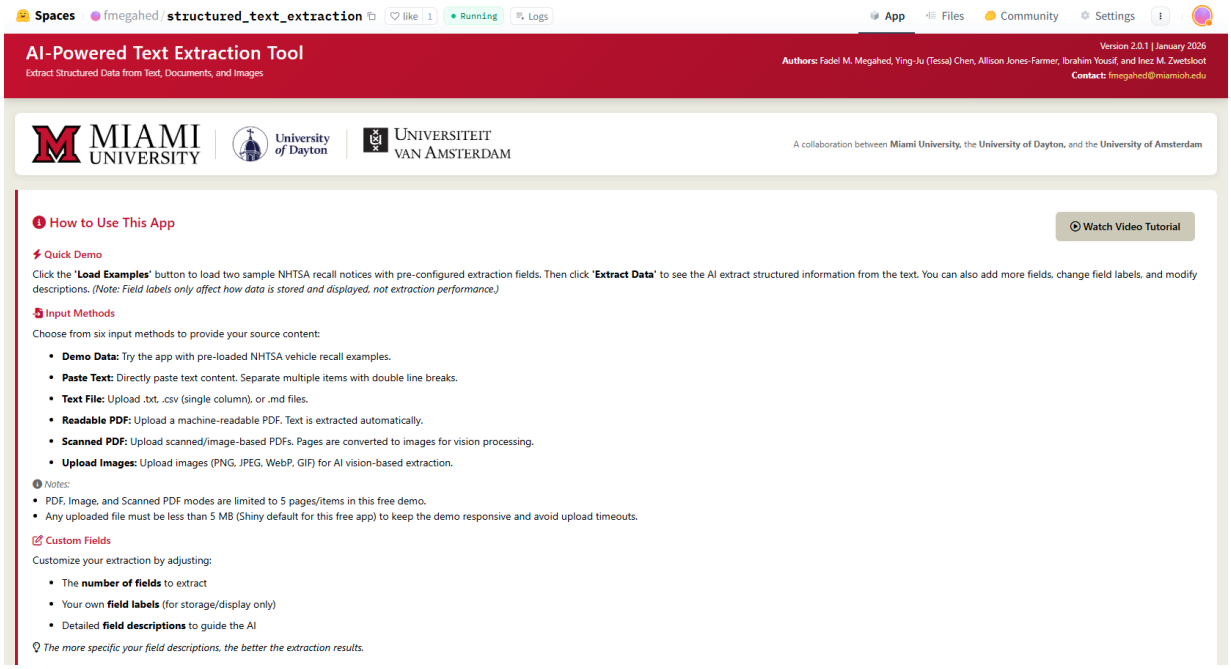
应用程序架构。 为了开发面向从业者的应用程序，我们使用 Shiny `R` 软件包进行开发，并将其托管在 https://huggingface.co/spaces/fmegahed/structured text extraction。在图2中，我们展示了应用程序的架构以及截至 2.0.1版本的应用界面；在我们开始撰写论文时，该版本已公开提供。后续变更总结于第 6.2节。

这个输入层同时支持基于文本的输入和基于图像的输入。对于基于文本的输入，用户可以使用我们的 NHTSA 演示示例数据，了解如何使用大语言模型进行文本提取。我们还支持用户通过以下方式使用 Shiny 应用来处理自己的应用场景：(a) 将文本直接粘贴到应用的文本框中；(b) 上传纯文本文件（即 `.txt`文件）；或 (c) 上传机器可读PDF。另一方面，对于基于图像的输入，用户可以上传扫描版（基于图像的）PDF 进行提取。此外，用户还可以上传图像进行文本提取（例如，读取条形图中的标签和值）。

预处理层 会为人工智能模型处理输入内容做好准备。文本输入和机器可读PDF文件会作为字符串直接传递。另一方面



(a) 结构化文本提取应用程序的系统架构，包括输入处理、预处理、运行时模式构建以及基于大语言模型/视觉语言模型的提取。



（b）托管式 Shiny 界面（2.0.1版本），展示了基于浏览器的结构化文本提取的使用指南和支持的输入模式。

图 2 ：面向质量工程工作流程的结构化文本提取应用程序 .

手动处理扫描 PDF 和图像时需要额外步骤；首先使用 `magick` R 软件包（Ooms2026）将每一页转换为图像，然后将其封装到 `ellmer`'s`content image file()` 函数中，以便由视觉语言模型处理。请注意，我们限制

扫描 PDF 的提取页数不超过五页，以降低成本和处理时间。

用户在 模式层 中定义提取字段。每个字段都包含一个标签（例如，`manufacturer`）和一段简明英语描述（例如，“发布召回通知的公司”）。用户可以通过浏览器界面按需添加任意数量的字段。这些定义会在运行时组合成一个 `ellmer type object()` 模式。由于字段描述相当于给模型的自然语言指令，其具体程度会直接影响提取质量。例如，如果用户对 `manufacturer's`提取效果不满意，可以：(a) 明确说明不需要同时提供公司名称和简称，并指出需要其中哪一个；和/或 (b) 在提示词中提供一个或少样本学习示例，以便人工智能模型更好地理解要求。

提取层 通过 API 调用将预处理内容和模式发送给大语言模型/视觉语言模型。模型会为每份输入文档返回一条 JSON 记录，我们再将其转换为结构化表格，其中每一列对应一个提取字段，每一行对应一个缺陷/页面/等。为方便用户使用，我们的 Shiny 应用支持用户在网页浏览器中查看结果，和/或将结果下载为 CSV 文件。

需要注意的是，我们的 Shiny 应用的用途并不局限于这个 NHTSA 示例。简而言之，用户可以更改字段标签及其描述，以便将提取功能用于其他应用场景。用户可以通过应用的界面用简明英语（或自己选择的任何其他语言）完成这些操作；也就是说，用户无需编写任何 R语言代码。

提取示例。为了具体说明工作流程，我们来看内置演示中包含的两则 NHTSA 召回通知。第一则通知部分内容如下：“Ford Motor Company（Ford）正在召回部分 2021–2022 年 Bronco 车辆。车辆挂出倒挡后，后视摄像头图像可能仍显示在中控台屏幕上。”第二则通知描述了与燃油喷射 ECU 软件错误有关的本田召回。假设用户定义了两个提取字段：

- `manufacturer` （描述：“召回车辆的公司名称”）

- `defect_summary` （描述：“主要缺陷摘要”）

应用将这两份通知及其 Field 定义一并发送给模型，并返回了以下表格：

表 2：使用内置演示功能从两份 NHTSA 召回通知中提取信息的示例。用户定义了两个字段，模型为每份通知填充了一行。

制造商	缺陷摘要
Ford 电动机公司（福特）	倒车操作结束后，后视摄像头图像可能仍会显示，从而不符合联邦机动车安全标准（FMVSS）第111号《后方视野》标准。
本田（美国汽车公司。） 本田	燃油喷射电子控制单元（FI-ECU）中的软件错误可能导致发动机熄火或动力损失。

部署和模型选择。托管演示运行在 Hugging Face Spaces 提供的免费 CPU 层级上。由于免费层级不提供足以在本应用中可靠运行本地视觉语言模型/大语言模型的硬件，因此演示在上线时使用了 OpenAI 的 `gpt-5-mini`作为后端。对于以下人工评估，我们只需修改代码中的一个输入参数，就改用了更新且成本更低的 `gpt-5.4-nano`。随着 OpenAI 弃用旧模型，我们将继续更新演示所用的模型。我们选择的模型在结构化提取任务中实现了实用的成本与性能平衡，因为我们的应用免费向公众开放。处理敏感数据或专有数据的用户可以按照第3节所述，仅更改提供商，即可使用本地模型在自己的基础设施上运行该应用。代码已公开发布在 Hugging Face代码仓库中，并为新用户提供了视频教程（使用 2.0.1 版本录制）。

对提取质量进行人工评估。为评估结构化输出是否准确呈现了源文档，我们从 NHTSA召回数据集中抽取了一份样本，并进行了盲法人工评估。我们使用两个模型分别独立处理了 30 条召回描述：(a) `gemma4:e2b` （一种功能较强且体积较小的开放权重模型，无需 GPU 即可部署），以及 (b) `gpt-5.4-nano` （评估时 OpenAI 最便宜的模型之一）。对于每条召回，我们提取了三个字段：制造

商、车型和车型年份。我们选择这些字段，是因为无需主观解读，就能直接对照召回描述进行核查。

提取结果以随机顺序呈现，且不标明生成结果的模型。为开展评估，我们构建了一个基于网络的评审应用程序，并为每位评审人员提供个性化链接。我们的应用程序通过以下方式简化了评审流程：(a) 每个界面显示一条召回描述，以及一个大语言模型针对三个字段的提取结果；(b) 在每个界面提供评分标准定义。四名评审人员（由学生和研究助理组成）使用五点正确性量表，针对召回描述对每个字段进行评估，评分范围为 1（不正确）至 5（正确）；其中间类别用于区分部分正确的提取结果，以及包含轻微错误或遗漏的提取结果。该评分标准已在第 5 节中完整呈现并作进一步讨论。评审人员获指示，仅根据所提供的召回文本判断每项提取结果。该应用程序的源代码和一段简短的概览视频可在“可复现性材料”部分提供的 GitHub代码仓库中获取。

在总计 720 项字段级评分中，有 710 项（98.6%）获得了 5 分（正确）。表 3显示，两个模型在三个字段上的表现均始终良好。

表 3：对从 30 条 NHTSA 召回描述中提取的结构化字段进行人工评估。表中数据表示人工评分等于 5（正确）的百分比。

模型	制造商	车型	车型年份	总体
gemma4:e2b	100.0%	97.5%	98.3%	98.6%
gpt-5.4-nano	100.0%	96.7%	99.2%	98.6%
总体	100.0%	97.1%	98.8%	98.6%

四位人工审阅者在评估中也表现出高度一致性。他们对180个模型–文档–字段组合中的174个给出了完全相同的评分（96.7%）。在存在分歧的六个组合中，有五个涉及评分为4分与5分的情况。剩余的一例涉及从召回描述中提取车型。来源指出， `PACCAR` 正在召回某些 `2021-2025 Peterbilt` 和 `Kenworth` 车辆，但让读者查阅召回报告以获取完整的车型列表。 `gpt-5.4-nano` 模型没有

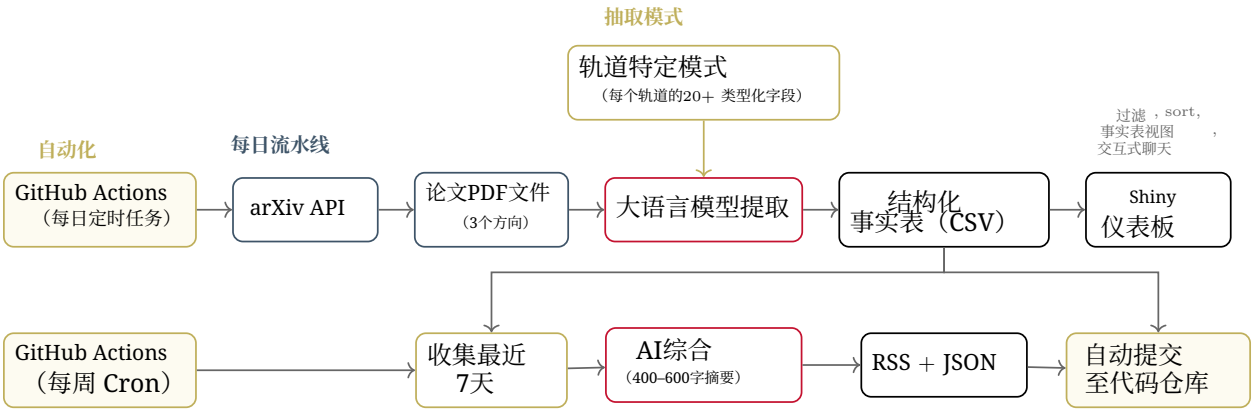
提取任何车型；三位审阅者将其评为不正确（1分），而一位将其评为基本正确（4分）。这种分歧似乎源于审阅者对所评估字段的理解不同。尽管来源中明确提到了 `Peterbilt` 和 `Kenworth`，但它们是汽车品牌，而非具体车型。因此，鉴于所提供的信息，模型决定不提取这些内容是合理的。这一案例表明，即使评估的是可以直接根据来源核实的字段，对字段的定义和理解方式不同，也可能导致人工审阅者之间产生分歧。

5 应用 2：监测 arXiv 上的质量工程研究

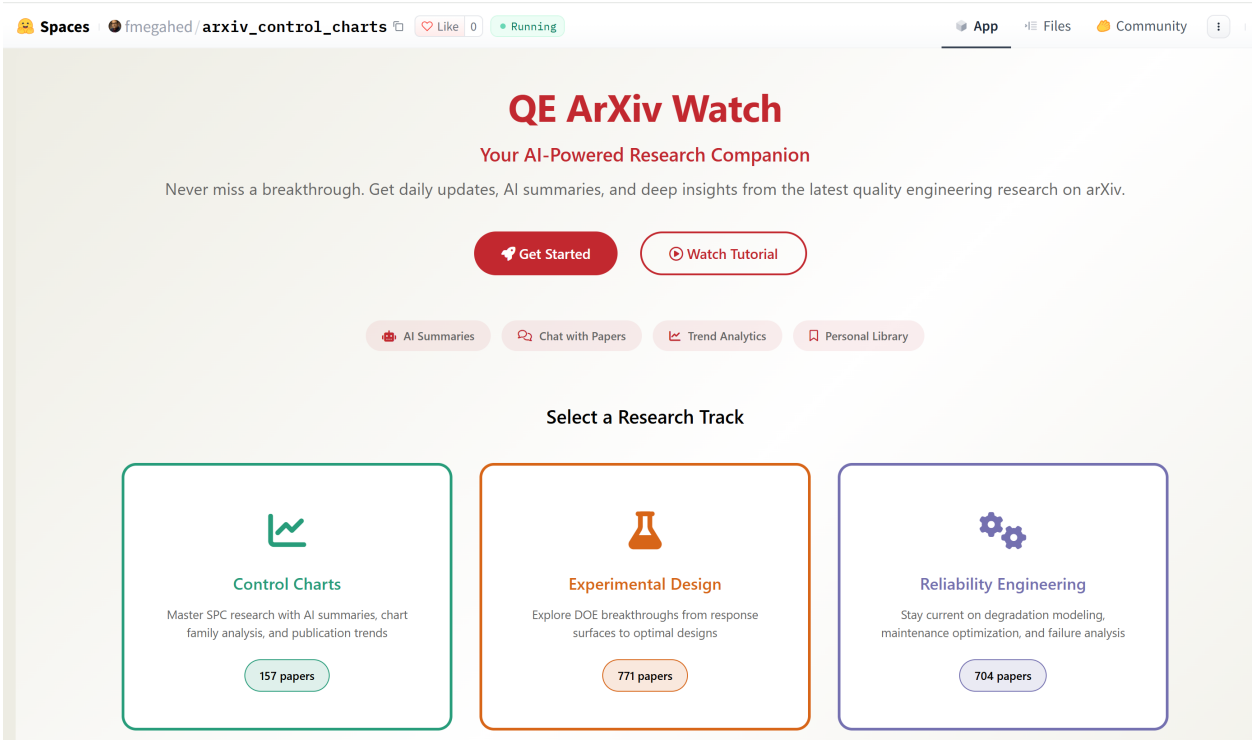
动机与实际需求。 我们推测，读者可能使用过 ChatGPT（或类似工具）来总结 PDF，或查找质量工程主题的相关文献。虽然生成的（技术）摘要通常很有用，但默认情况下往往相当冗长，分布在多个段落中；如果没有明确提示大语言模型提取读者所关注的内容，摘要还可能遗漏重要方面。此外，搜索（自 ChatGPT 早期以来已有显著改进）往往是一次性任务，用户需要定期重复搜索，以发现新的论文。需要注意的是，由于不同会话/论文的回复没有遵循统一格式，因此跨会话/论文进行比较也很困难。为应对这些局限性，我们开发了 `QE ArXiv Watch`。它每天检查 arXiv 中是否有新的质量工程论文，读取每篇论文的全文，并从每篇论文中提取 20 多个预定义字段。随后，它通过多种方式在网页界面中呈现结果：(a) 可筛选表格；(b) 可视化图表；(c) 每篇论文的专属页面；以及 (d) 按学科/作者进行的汇总。QE ArXiv Watch 可免费访问：
https://huggingface.co/spaces/fmegahed/arxiv controlcharts。

应用架构。 该应用包含三个部件：(a) 每日提取流程，(b) 每周摘要流程，以及 (c) 交互式仪表板。这两个流程通过 `GitHub Actions`按计划运行，从而确保 `QE ArXiv Watch`更新自动

进行。在图 3a–3b中，我们总结了 `QE ArXiv's` 架构，并展示了截至2026年10月8日已处理论文数量的着陆页。着陆页的截图基于3.3.0版本，该版本的事实表曾由一组arXiv论文作者进行评估。



(a) 采用 GitHub Actions 自动化的系统架构，用于每日从 arXiv 获取论文、基于大语言模型进行结构化抽取、生成每周摘要并更新代码仓库。



（b）着陆页展示了三个研究方向、按方向统计的论文数量以及用于访问仪表板结构化事实表和相关工具的摘要指标。

图 3：QE ArXiv Watch 的架构和着陆页。

该每日提取流程是一个 R 脚本，每天早晨由 GitHub Actions 的 cron（即定时）任务触发。脚本会通过 arXiv API 查询三个研究方向的新论文：(a) 统计过程控制（SPC；当前版本中称为统计过程监控），(b) 实验设计（DOE），以及 (c) 可靠性工程。在评估版本中，SPC 赛道会搜索标题或摘要中包含“控制图”等词语的论文。DOE 赛道会搜索包含“实验设计”“响应面”和“超饱和设计”等词语的论文。最后，可靠性赛道会搜索包含“可靠性”“退化”“维护”和“剩余使用寿命”等词语的论文，并将范围限定在相关统计学类别中。之后我们修订了这些搜索条件（第6.2节）。发现新论文后，脚本会直接从 arXiv 读取每篇论文的 PDF（不在本地保存副本），使用 `ellmer's content pdf url()` 函数，并将 PDF 连同针对相应研究方向的抽取模式发送给大语言模型。生成的结构化事实表会追加到 CSV 文件中，并同时存储在我们的 GitHub 和 HuggingFace 代码仓库中。

每周摘要流程是每周一运行的第二个 GitHub Actions 作业。它会收集前七天内创建的所有事实表，并将其连同指令发送给大语言模型，请其准备一份 400–600 字的最新研究摘要（发表于过去七天内）。其目标是在无需研究人员阅读每份事实表的情况下，为他们简要概述近期论文。该应用会同时将摘要发布为 RSS订阅源（读者可以订阅）和 JSON文件（可在 R/Python 中轻松读取）。

这个交互式仪表板 是一个托管在 Hugging Face Spaces 上的 Shiny 应用程序。它读取每日流水线生成的 CSV文件，并提供三个针对不同方向的页面，以及一个展示汇总统计量的着陆页。在每个方向的页面上，用户可以根据抽取的字段过滤论文（例如，基于模拟的EWMA论文），按日期或其他抽取维度排序，并在弹出面板中查看论文的完整事实表。仪表板还包含交互式聊天功能，用户可以就论文提出后续问题。

抽取模式。 由于必须捕捉研究人员用于对论文进行分类的维度，本应用中的抽取模式比面向从业者的示例更为详细。我们选用了能力更强的模型（`gpt-5.2` 在评估版本中），因为这项任务需要阅读完整的 PDF 格式研究论文，并以一致的结构填写许多字段。此外，若干字段采用多标签枚举（例如，单篇论文可能涉及多个图表族，或报告多项性能指标），这增加了抽取的复杂性。仅 SPC 模式就包含二十多个类型化字段，分为五组（表 4）。DOE 和可靠性方向也采用类似的模式，并根据领域设置相应字段（例如，DOE 的设计类型、设计目标、最优性准则和因子数量；可靠性的可靠性主题、建模方法、数据类型和维护策略）。此后，我们对模式和模型都进行了更改（第 6.2节）。

Ta表 4：评估版本的统计过程控制抽取模式中的字段组 .

组	字段
分类	图表族（例如 Shewhart、累积和控制图、指数加权移动平均）；制图统计量；第一阶段与II期指定；应用领域
数据假设	论文是否假设正态性、是否处理自相关，或是否处理缺失数据
评估	评估类型（模拟、真实数据、理论）；性能指标（例如，平均运行长度或运行长度标准差）；样本量要求
软件	是否使用代码；所用平台（R语言、Python、MATLAB、SAS）；代码是否公开提供；托管位置
内容	简要摘要；关键方程；关键结果；明确说明的局限性；模型识别出的未明示局限性；明确说明的未来工作；模型建议的未来工作
	模型

提取示例。为说明输出结果，来看 arXiv 上由 `2512.23602v1`Burger 撰写的论文（2025年），题为《基于保形预测的无分布假设过程监控》（提交于2025年12月29日）。每日流水线下载了 PDF，将其发送给 `gpt-5.2`并使用 SPC 模式，生成了表 5所示的事实表。

`assumes normality` Field 为 `FALSE`，这与论文强调通过保形预测实现无分布假设监控的观点一致。提取后，这份事实表会——

表 5：使用 SPC 模式（评估版本）的 arXiv 论文中提取字段的子集 `2512.23602v1` 。竖线分隔的值表示多标签字段。这些值`Both` 和 `Other` 是本表中所示的作者报告的问题示例，相关解决方法详见第6.2节。

字段	提取值
图表族	单变量 | 多变量 | 非参数 | 高维 | 其他
图表统计量	Shewhart | 基于机器学习 | 其他
阶段	Both
假设正态性	FALSE
评估类型	仿真研究 | 其他
性能指标	误报率 | 检测概率 | 其他
使用的代码	TRUE
代码可用性	公共代码仓库（GitHub）
代码网址	https://github.com/christopherburger/ConformalSPC
摘要	该论文提出了一种混合式统计过程控制（SPC）框架 该框架集成保形预测（CP），从而在可交换性而非正态性假设下，获得与分布无关且与模型无关的监控保证。 在可交换性而非正态性条件下，与分布无关的监控保证。 论文介绍了：（i）保形增强控制图，用于替代参数化 基于校准非一致性分数得出的保形阈值的控制限， 并可选择绘制随时间变化的保形预测区间，以 直观呈现不确定性并触发“不确定性峰值”信号。此外，它还提出 （ii）保形增强过程监控，它将多变量统计过程控制重新定义为 在线异常检测，并通过监控随时间变化的单个保形 p 值 来实现，使用 p 值图，并在显著性水平为 α 处设置控制限。非一致性 分数采用稳健中心度量（例如，基于中位数的绝对 偏差）或基于模型的残差（还可能按估计的局部 尺度进行归一化），因此可用于任意预测模型或异常检测模型（例如， 支持向量机、自编码器）。论文提供了针对正态数据和指数数据的模拟图表比较， 并指出该方法提高了稳健性和灵敏度 相较于传统的 Shewhart 风格图表，同时保持可解释性。

成为仪表板 SPC表中的一行。研究人员随后可以过滤仪表板，找出所有不假设正态性的论文，或所有提供公开代码的论文，从而将文献综述过程的一部分转化为结构化查询任务。

当前规模与总体洞见。 截至2026年10月8日，评估版本包含1,632篇不同论文的结构化事实表：其中157篇属于SPC赛道（涵盖2008年至2026年10月的投稿）、771篇属于DOE，704篇涉及可靠性。

通过结构化事实表和仪表板过滤器，可以直接获得那些手动生成会十分困难的汇总分析。例如，在157篇SPC论文中，最常见的图表族是单变量（93篇）和多变量（49篇），其次是非参数（36篇）、轮廓监测（17篇）和高维方法（16篇）。其中仅有58篇

157篇论文被编码为假设数据服从正态分布，这表明无分布假设方法和稳健性导向方法仍然保持着持续活跃。最常被识别的软件平台是R语言（55篇），其次是Python（12篇）和MATLAB（9篇）。代码可用性模式同样值得注意：仅有12篇论文在公共代码仓库中提供代码，另有10篇通过包注册表（CRAN或PyPI）提供代码。相比之下，有111篇论文被编码为使用软件，但未提供可公开访问的代码。

质量检查与作者评估。 为了评估人工智能生成的事实表质量，我们邀请作者评估从其论文中提取的信息。我们从每个研究方向中随机选取30篇论文，排除由我们团队成员合著的论文，并邀请其余每篇论文的一位作者参与评估（共发出85份邀请，其中81份成功送达）。作者可以评估仪表板中收录的任何一篇自己的论文。共有21位作者评估了45份事实表：26份SPC、14份可靠性和5份DOE。

对于每份事实表，作者对八个字段进行评分：四个适用于所有研究方向的通用字段（AI摘要、关键结果、关键方程和未陈述的未来工作），以及四个特定于研究方向的分类字段。我们采用了首次应用中介绍的相同五点正确性量表，这与此前关于评分量表和大语言模型输出专家评估的研究一致（Allen和Seaman2007；Megahed等人2024b）。对于未陈述的未来工作，我们采用了相应的五点合理性量表，因为该字段代表人工智能生成的建议，而不是直接从论文中提取的信息。表6 给出了这两种量表的定义。作者也可以提供意见，我们据此识别信息提取问题，并改进提示词和模式。

图4 总结了作者评估结果。结果表明，事实表提供了有用的初步提取结果，但不同字段和方向的准确度各不相同。

在四个叙述性字段中，AI摘要的评分最高，所有评估结果均为4分或5分，其次是关键结果（93.3%）、关键方程（77.8%）和未陈述的未来工作（66.7%）。关键方程的一些较低评分与

表6：用于评估结构化提取输出和未来工作合理性的量表。

分数	正确性锚点	正确性定义	合理性锚点	合理性定义
1	不正确	该回答是错误的，没有论文依据，或明显歪曲了论文的内容。	不合理	建议的未来工作是不可信的，与论文脱节，或没有切实地受到其内容的启发。
2	大多不正确	回答中包含少量正确信息，但仍存在大量错误，误分类，或重要的遗漏。	大多不合理	该建议与论文的关联有限，并且仅得到其方法、研究结果或讨论的微弱支持。
3	部分正确	回答既包含正确信息也包含不正确的信息，或者只涵盖了论文所支持内容的一部分。	有些合理	该建议部分合理可信，且与论文有一定关联，但论证不完整或仅属中等程度令人信服。
4	大部分正确	回答基本正确，且有论文支持，仅有轻微错误、遗漏或不够精确。	大部分合理	该建议看起来合理，与论文相关，并且有其依据，且依据较为充分内容，仅存在轻微论证薄弱。
5	正确	该回答正确，并有论文作为依据，且不包含实质性错误或遗漏。	合理	建议的未来工作切实可行，与论文高度契合，并且有充分的依据支撑。

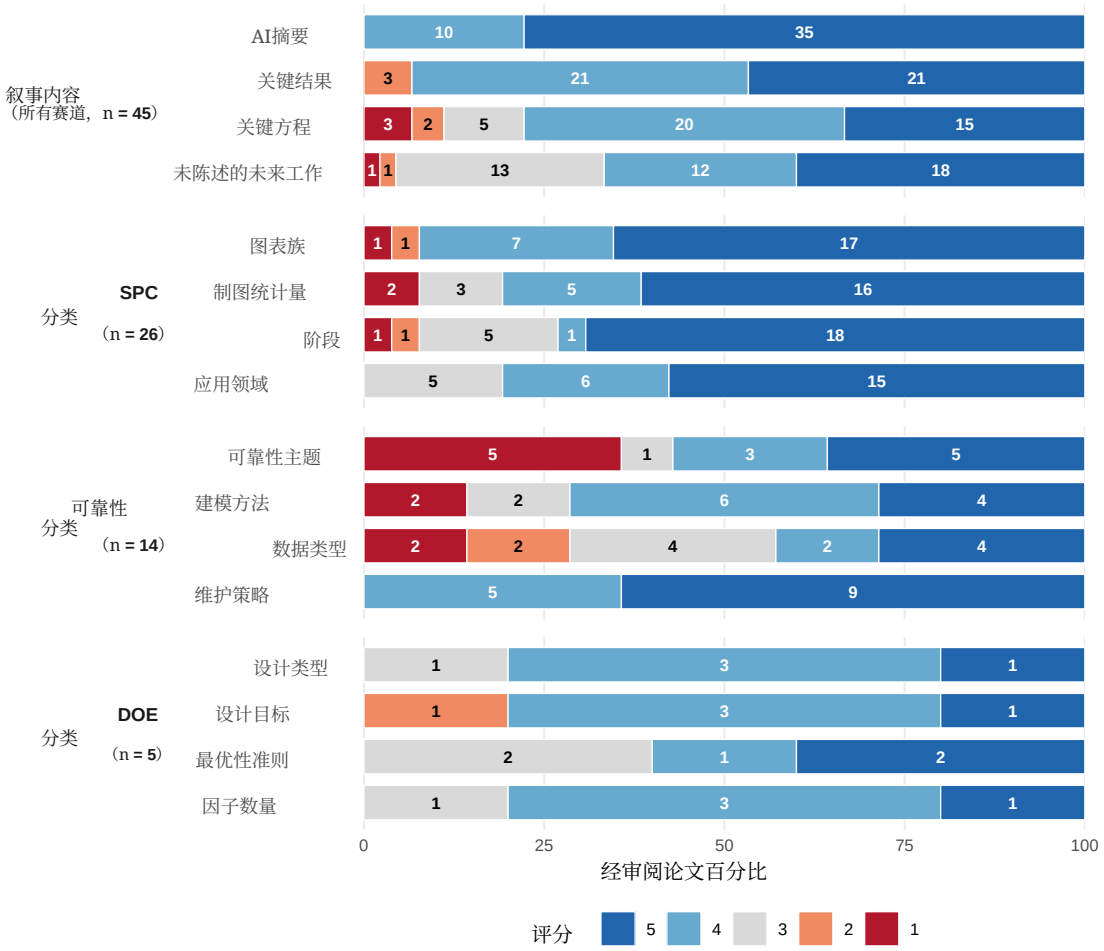


图4：各人工智能提取的事实表字段的作者评分分布。叙述性字段适用于所有赛道；分类字段则因赛道而异。条形图中的数字表示论文数量；评分范围为1（不正确/不合理）至5（正确/合理）。

未能正确呈现的数学符号（45份事实表中有12份收到了关于呈现问题的评论，其中包括该字段评分为1的全部3个案例）。对于未陈述的未来工作，几位作者认为这些建议合理，但属于渐进式改进。

分类结果因赛道而异。对于 SPC 赛道，图表族、制图统计量和应用领域的评分中，有 80.8%–92.3% 为 4 分或 5 分。在 SPC分类中，阶段分类标注最有改进空间（73.1% 的评分处于 4–5 分区间）。评审者指出，他们的论文仅涉及第二阶段监测，或使用了不适用第一阶段/第二阶段区分的自启动图表。可靠性的评分最低，尤其是在可靠性主题方面

（57.1%）和数据类型（42.9%）。其中一些评分对应于通过可靠性检索获取的论文，而这些论文的主要主题（例如可解释性或主动学习）并不符合模式中的分类。另有 4 个评分为 1 的情况与显示为nan的缺失值有关。由于仅评估了 5 篇论文，样本量过小，无法得出有意义的结论，因此我们不对 DOE 结果发表评论。我们利用这些评分和评论对应用进行了修订，具体说明见第 6.2节。

6 讨论

6.1 应用中的经验教训

我们从这两项应用中得出的一个观察结果是，提取质量取决于任务以及所提取的字段。在召回应用中，两个相对较小的模型在提取制造商、车型和车型年份等事实信息时表现良好。在 QE ArXiv Watch中，人工智能生成的摘要通常获得了较高的评分，而一些分类字段的评分则明显较低。尽管这两项评估涉及不同的任务，不能直接进行比较，但它们表明，使用单一的总体准确率指标来评估提取应用可能会产生误导。相反，我们建议考察单个字段的质量，尤其是那些将用于后续分析的字段。

另一个教训涉及提取错误的来源。在这两个应用中，人工复核人员都发现了一些与字段定义/解释方式有关的问题。在 QE ArXiv Watch中，部分较低的评分还与数学方程的显示问题以及缺失值的处理问题有关。需要注意的是，切换到更强大的模型未必能解决这些问题。相反，这些问题可能需要修改模式、提取提示或应用本身。这也是我们鼓励用户复核部分提取记录，并利用研究结果改进提取工作流的原因之一。我们将在下一节介绍针对所发现问题所做的修改

小节。

6.2 改　　进和作者审阅引发的变更

对 QE ArXiv Watch的作者评估帮助我们发现了由不同原因引起的若干问题。例如，以原始 LaTeX 形式呈现的方程属于显示问题，需要修改应用存储和呈现数学表达式的方式。其他问题则与搜索术语和抽取模式有关。最初的可靠性检索范围过于宽泛，检索出了包含“reliable”但与可靠性工程无关的论文。为了改进检索，我们考察了发表于八本“质量与可靠性”期刊的 13,780 篇文章中所使用的术语。随后，我们使用 arXiv 论文测试搜索术语，并保留主要能够检索到相关论文的术语。我们还增加了一个筛选步骤，对每篇论文的标题和摘要进行评估，以确定其是否属于相应的研究方向。对于抽取模式，我们修改了分类字段，要求只能指定一个主要类别，提供了更清晰的定义，并在需要时增加了“不适用”选项。这解决了诸如为一篇论文分配过多类别，或将自启动控制图错误分类为第一阶段或 II期等问题。我们还修正了召回应用中的若干问题，包括一个失败的文档可能导致其他文档的结果被丢弃的问题。

另一项改进是使用多个模型来提取分类字段。修订后的方法使用一个小型大语言模型（目前为 gpt-6-luna）阅读每篇论文并填写其事实表。另一个来自不同提供商的模型（Jev）会独立评估主要分类字段（TypeSafe AI 2026）。与大语言模型不同，这个模型会从预定义选项中进行选择，并返回所选项的概率，而不会生成文本。当两个模型意见一致时，我们就采用它们的分类结果；否则，我们会让一个更大的大语言模型（目前为 gpt-6.1-sol）阅读论文并解决分歧。这类似于让第二位检查员检查同一个部件，再由第三位检查员解决分歧。我们还会在每份事实表中注明分类结果是否经过一致确认

或是否需要使用更大的模型。在对 125 篇论文进行评估时，两个模型在 82% 的单值分类上意见一致；对于 34% 的范围内论文，则需要使用更大的模型。由此，每篇范围内论文的成本约为对所有提取任务都使用更大模型时的五分之一。

我们还采取了几项预防措施，以确保应用程序更新不会引入新的问题。首先，我们归档了作者评估过的原始事实表，以保留本文报告的结果。其次，我们将作者提出的具体更正转化为 17 项自动化检查。例如，其中一项检查会验证某篇论文是否被正确归类为 II期。在部署之前，我们要求每个新配置都通过这些检查。最终配置通过了全部 17 项检查，而单独使用的每个大语言模型都未通过其中一项。第三，在处理完整集合之前，我们在包含通过修订后的搜索检索到的论文在内的 125 篇和 150 篇论文样本上测试了完整的提取工作流。我们还将新的事实表单独存储，以便与原始结果进行比较。最后，我们将模型名称、搜索术语和字段定义移入一个统一的配置文件中。提取提示、应用程序筛选器和帮助文档均由该文件生成，以确保在进行更改时保持一致。

更新后，修订后的检索共找到 2,982 篇论文，其中 2,316 篇通过了筛选步骤并获得了事实表。这些论文包括 244 篇关于过程监控的论文、1,540 篇关于实验设计的论文，以及 532 篇关于可靠性工程的论文。37% 的论文需要使用更大的模型，21% 的论文因此改变了主要分类。两个应用现在都会告知用户，自研究团队和 arXiv 作者样本进行首轮评估以来，应用已进行了更新。出于对作者时间的尊重，我们没有请他们再次评估修订后的事实表。因此，尽管我们解决了 arXiv 论文作者提出的问题，并通过了自动化检查，但仍无法断定这些改动会提高评估评分。

6.3 实际局限性与权衡

我们只对更换底层大语言模型的影响进行了部分考察。例如，我们计划定期更新 `QE ArXiv Watch` 所使用的模型，以便采用更新的模型。然而，即使抽取模式保持不变，不同模型也可能对同一篇论文给出不同的分类（Chen、Zaharia 和 Zou，2023年）。在 arXiv 作者评审之后，我们评估了 OpenAI 的两个替代模型：一个较小的模型（`gpt-6-luna`）和一个较大的模型（`gpt-6.1-sol`）。在指令相同的情况下，视领域而定，这些模型对抽样的 125 篇论文中 74%–98% 给出了相同的主要标签。对于会随时间不断积累记录的应用，我们建议在更换模型时抽查此前处理过的文档，并记录用于抽取的模型。当前版本的每份事实表都记录了模型、模式版本和提取日期。

模型的选择还取决于具体应用。在我们的召回示例中，在提取三个事实字段时，本地部署的小型模型表现与商业模型相当。对于更复杂的提取任务，例如 `QE ArXiv Watch` 中所使用的任务，我们尚未评估是否能够取得相同的结果。成本是另一项需要考虑的因素，尤其是在处理大量文档，或使用视觉语言模型从扫描 PDF 和图像中提取信息时。当文档本身非常庞大（例如，$100+$ 页的报告）时，可能需要使用预处理管道，在提交文档给大语言模型或视觉语言模型之前筛选出相关页面，以确保相关成本处于合理范围内。最后，对于处理专有信息或敏感信息的组织而言，如果不适合将文档发送给外部服务提供商，可能会更倾向于使用本地部署的模型。

7 结论

总体而言，本文旨在展示质量工程师如何利用大语言模型和视觉语言模型，将非结构化文档转换为适合进一步分

析的数据。通过代码示例和两个公开可用的应用程序，我们展示了如何在无需开发专用自然语言处理管道的情况下实现这一目标。我们相信，这些工具可以减少为质量工程应用准备数据所需的工作量。尽管`QE ArXiv Watch`监测研究文献，但同样的定时检索、基于模式的提取和可筛选仪表板架构，也可以应用于任何需要使用特定领域类别进行分类的定期文档流，例如供应商纠正措施报告、客户投诉或监管更新。但是，定义变量、评估提取信息的质量以及解读所得数据的责任，仍由质量工程师承担。

可复现性材料

本文的代码及配套材料可在此处获取：https://github.com/fmegahed/qegenai extraction。代码仓库包含 NHTSA 召回样本数据、用于模型比较的 R脚本及其结果，以及人工评估材料，包括评审应用程序和一段简短的演示视频。两个应用程序的部署版本可通过各自章节中提供的链接访问。应用程序 1 和 2 的评估版本已分别在 Hugging Face 和 GitHub 上的相应代码仓库中标记。此外，作者评分的事实表未经修改，归档在第二个代码仓库中。

生成式人工智能工具的使用

我们使用 Claude Code，通过人工智能辅助编程（通常称为“氛围编程”）开发了这两个应用及其相关评估工具，具体过程如我们的 GitHub代码仓库所述。在准备论文期间，我们使用 ChatGPT、Claude、Grammarly 和 Writefull 改进语法、编辑和精简文本，并减少各部分之间的重复内容。我们还使用 Claude Code 和 ChatGPT 改进 LaTeX 图表的格式和呈现效果。

致谢

作者感谢 21 位接受我们事实表审阅邀请、提供反馈并完成调查的 arXiv 论文作者。我们还感谢 Maria Weese 和 Ozge S¨urer 对 `QE ArXiv`早期版本提出的反馈。我们感谢迈阿密大学的*Raymond E. Glos* 商学教授席位、*Dinesh & Ila Paliwal* 创新讲席和*Van Andel* 商业分析学捐赠教授席位 对人工智能使用成本的支持。

参考资料

Allen, I. Elaine 和 Christopher A. Seaman。2007 年。《李克特量表与数据分析》。*Quality Progress* （7 月）：64–65。

Anderson-Cook, ChristineM, Lu Lu, WilliamBrenneman, Jeroen DeMast, Frederick Faltin, Laura Freeman, William Guthrie, Roger Hoerl, Willis Jensen, Allison Jones-Farmer 等。2022 年。《统计工程——第 2 部分：未来》。质量工程34（4）：446–467。https://doi.org/10.1080/08982112.2022.2106440。

Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell 等。2020 年。《语言模型是少样本学习器》。 神经信息处理系统进展 33：1877–1901。https://proceedings.neurips.cc/paper files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf。

Burger, Christopher. 2025年。“基于保形预测的无分布假设过程监控。”*arXiv 2512.23602*，https://doi.org/10.48550/arXiv.2512.23602。

Chen、Lingjiao、Matei Zaharia 和 James Zou。2023年。《ChatGPT 的行为如何随时间变化？》*arXiv* 预印本 *arXiv:2307.09009v3*，https://doi.org/10.48550/arXiv.2307.09009。

Francis、Jack、Kim Cates 和 Gianluigi Caldiera。2023年。《SmarTxT：一种用于高效车辆缺陷调查的自然语言处理方法》*Transportation ResearchRecord* 2677 (3)：1579–1592。https://doi.org/10.1177/03611981221125744。

Ghazizadeh, Mahtab, Anthony D. McDonald 和 John D. Lee。2014年。《文本挖掘解读自由回答式消费者投诉：来自 NHTSA 车主投诉数据库的洞见》。*Human Factors* 56 (6)：1189–1203。https://doi.org/10.1177/0018720813519473。

Hamilton, Austin, Ryan Singh, Michael Wise, Ibrahim Yousif, Arthur Carvalho, Zhe Shan, Mohammad Mayyas, Lora A. Cavuoto 和 Fadel M. Megahed。2026年。《认知准确性的 Token 成本：比较面向文档的检索增强生成与长上下文架构在生成式人工智能应用中的表现》。*arXiv 2606.20898*，https://doi.org/10.48550/arXiv.2606.20898。

Hong, Yili, Man Zhang 和 William Q. Meeker。2018年。《大数据与可靠性应用：复杂性维度》。*Journal of Quality Technology* 50 (2)：135–149。https://doi.org/10.1080/00224065.2018.1438007。

Huang, Lei, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qian- glong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin 等。2025年。“大语言模型中的幻觉综述：原理、分类体系、挑战与未解问题。”*ACM Transactions on InformationSystems*43 (2)：1–55。https://doi.org/10.1145/3703155。

Ji, Wenlong, Weizhe Yuan, Emily Getzen, Kyunghyun Cho, Michael I Jordan, S ong Mei, Jason Weston, Weijie J Su, Jing Xu 和 Linjun Zhang。2026年。“面向统计学家的大语言模型概述。” *The American Statistician*，第 just-accepted 期，1–106。https://doi.org/10.1080/00031305.2026.2657480。

Megahed, Fadel M, Ying-Ju Chen, Bianca Maria Colosimo, Marco Grasso, L Allison Jones- Farmer, Sven Knoth, Hongyue Sun 和 InezM Zwetsloot。2026a。“将 OpenAI 的 CLIP 模型应用于制造业质量控制中的少样本图像检测：包含多个应用示例的说明性案例研究。” *Journal of Quality Technology*，1–27。https://doi.org/10.1080/00224065.2026.2640171。

Megahed, Fadel M、Ying-JuChen、Joshua A Ferris、Sven Knoth 和 L Allison Jones-Farmer。2024年a。“生成式人工智能模型（如 ChatGPT）如何在 SPC 实践、教育和研究中被（误）用？一项探索性研究。”质量工程 36（2）：287–315。https://doi.org/10.1080/08982112.2023.2206479。

Megahed, Fadel M、Ying-Ju Chen、L. Allison Jones-Farmer、Gabe Lee、Jiawei Brooke Wang 和 Inez M. Zwetsloot。2026年b。“借助大语言模型实现可靠的决策支持：评估二元文本分类应用一致性的框架。”《商业分析期刊》，1–19。https://doi.org/10.1080/2573234X.2026.2652281。

Megahed, Fadel M、Ying-Ju Chen、Inez M Zwetsloot、Sven Knoth、Douglas C Montgomery 和 L Allison Jones-Farmer。2024年b。“推出 ChatSQC：利用增强人工智能提升统计质量控制。” *Journal of Quality Technology* 56（5）：474–497。https://doi.org/10.1080/00224065.2024.2372328。

Megahed, Fadel M, 和 L Allison Jones-Farmer。2015。“关于‘大数据’的统计学观点。”载于 《统计质量控制前沿》*11*，由 S. Knoth 和 W. Schmid 编，29–47。施普林格。https://doi.org/10.1007/978-3-319-12355-4 3。

Min, Jie、Xinyi Song、Simin Zheng、Caleb B King、Xinwei Deng 和 Yili Hong。2026年。“人工智能时代的应用统计学：综述与展望。”*Applied Stochas-tic Models in Business and Industry* 42 (2)：e70075。https://doi.org/10.1002/asmb.70075。

美国国家公路交通安全管理局。2026年。*2025*年度报告：安全召回。https://www.nhtsa.gov/sites/nhtsa.gov/files/2026-03/2025-annual-recalls-report.pdf。访问日期：2026年10月8日。美国交通部。

Nguyen, San S 和 Javeed Kittur。2025年。“大型语言模型在工程领域当前及潜在用途综述。”*Frontiers in Education* 10:1649650。https://doi.org/10.3389/feduc.2025.1649650。

Ooms, Jeroen。2026年。 *magick*：*R*中的高级图形与图像处理。 R软件包版本 2.9.1。https://doi.org/10.32614/CRAN.package.magick。 https://CRAN.R-project.org/package=magick。

OpenAI。2025年。 结构化模型输出。访问日期：2026年2月24日。https://developers.openai.com/api/docs/guides/structured-outputs/。

Ouyang, Long、Jeffrey Wu、Xu Jiang、Diogo Almeida、Carroll Wainwright、Pamela Mishkin、ChongZhang 等。2022年。“训练语言模型以遵循带有人类反馈的指令。”载于神经信息处理系统进展，由 S. Koyejo 编，

S. Mohamed、A. Agarwal、D. Belgrave、K. Cho 和 A. Oh，35:27730–27744。Curran Associates, Inc.https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf。

Posit开发团队。2026年。*chatlas*。GitHub代码仓库发布版本 v0.15.1，https://github.com/posit-dev/chatlas/releases/tag/v0.15.1。访问日期：2026年2月24日。

Sedgwick。2026年。*2026*年美国召回量超过十亿件。新闻稿。访问日期：2026年10月8日。Sedgwick。https://www.sedgwick.com/press-release/u-s-recall-volume-exceeds-one-billion-units-in-2026/。

Singh、Ryan、Austin Hamilton、Amanda White、Michael Wise、I brahimYousif、Arthur Carvalho、Zhe Shan 等。2026年。“多模态制造安全聊天机器人：知识库设计、基准开发以及多种检索增强生成方法的评估。”《制造快报》49：30–41。https://doi.org/j.mfglet.2026.07.012。

TypeSafe AI。2026年。介绍 *System One* 模型与 *Jev* [inen]。https://typesafe.ai/blog/introducing-system-one-models-and-jev。最后访问日期：2026年10月9日。

Uglanov, Alexey、Felician Campean、Amr Abdullatiff、Daniel Neagu、Aleksandr Doikin、David Delaux 和 Pascal Bonnaud。2024年。“一种基于自然语言处理的框架，用于从异构汽车生命周期数据中及早识别设计可靠性问题。”第34届CIRP设计大会，*Procedia CIRP* 128：728–733。ISSN：2212-8271。https://doi.org/10.1016/j.procir.2024.05.098。

Vaswani, Ashish、Noam Shazeer、NikiParmar、JakobUszkoreit、LlionJones、AidanNGomez、Lukasz Kaiser 和 Illia Polosukhin。2017年。《Attention Is All You Need》。载于《神经信息处理系统进展》，I. Guyon、U. Von Luxburg、S. Bengio、H. Wallach、R. Fergus、S. Vishwanathan 和 R. Garnett 编，第30卷。Curran Associates, Inc. https://proceedings.neurips.cc/paperfiles/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf。

Weese、Maria、Waldyn Martinez、FadelM Megahed 和L Allison Jones-Farmer。2025年。“应用于过程监控的统计学习方法：概述与展望。”*Journal of Quality Technology* 48（1）：4–24。https://doi.org/10.1080/00224065.2016.11918148。

Wickham、Hadley、Joe Cheng、Aaron Jacobs、Garrick Aden-Buie 和 Barret Schloerke。2025年。*ellmer*：与大语言模型对话。R软件包版本 0.4.0。https://doi.org/10.32614/CRAN.package.ellmer。