arXiv:2601.03496v1 [cs.IR] 7 Jan 2026

STELLA：用于构建航空航天信息检索基准的术语感知自我反思框架

Bongmin Kim TelePIX
bmkim@telepix.net

摘要

航空航天业的任务高度依赖于对大量技术文档的搜索和再利用，但目前尚无能够体现该领域术语特点和查询意图特征的公开信息检索（IR）基准。为弥补这一空白，本文提出了**STELLA**（**S**elf-Reflective **TE**rminoLogy-Aware Framework for Bui**L**ding an **A**erospace Information Retrieval Benchmark，自我反思式术语感知航空航天信息检索基准构建框架）。基于该框架，我们推出了**STELLA基准**，这是一个专用于航空航天领域的信息检索评估集，利用由文档布局检测、段落分块、术语词典构建、合成查询生成和跨语言扩展组成的系统化流程，从 NASA 技术报告服务器（NTRS）文档中构建而成。该框架生成两类查询：术语一致查询 （TCQ），其中逐字包含术语，用于评估词汇匹配；以及术语无关查询 （TAQ），利用术语的描述来评估语义匹配。这使得能够分别评估嵌入模型的词汇匹配和语义匹配能力。此外，我们将密度链（CoD）和自我反思方法融入查询生成，以提升质量，并实现了一种能够反映真实用户查询习惯的混合式跨语言扩展。在 STELLA 基准上对七种嵌入模型进行的评估表明，大型解码器式嵌入模型展现出最强的语义理解能力，而 BM25 等词汇匹配方法在精确词汇匹配技术术语至关重要的领域仍具有很强的竞争力。STELLA基准为航空航天领域信息检索任务中的嵌入模型提供了可复现的可靠性能评估与改进基础。STELLA基准可通过 https://huggingface.co/datasets/telepix/STELLA 获取。

1 引言

在航空航天业，设计、制造和验证任务高度依赖对大量技术文档的搜索与再利用。需求、设计依据、测试条件、标准合规性和异常分析彼此紧密关联，快速定位相关证据并得出结论的能力直接关系到生产效率和安全。近年来，基于大语言模型（LLM）的检索增强生成系统展现出提升此类以文档为中心的工作流程效率的潜力 [Lewis等人，2020年，Fan等人，2024年，Zhao等人，2024a]。据报道，在检索增强生成系统中，检索增强通过将生成过程基于针对给定查询检索出的文档，能够减少幻觉并提高可靠性[Asai等人，2024年，Niu等人，2024年，Ayala和Bechard， 2024]。然而，目前尚无公开的信息检索基准能够同时反映航空航天领域实际所需的查询意图，以及领域专用术语的词汇和语义特征。因此，很难系统地衡量嵌入模型是否正确理解并表征航空航天术语——也就是说，它是否能够捕捉语义等价，而不仅仅是进行术语的词汇匹配。因此，系统地指导嵌入模型改进、比较系统性能并评估实际部署能力仍然颇具挑战。实践中，BEIR[Thakur等人， 2021] 已成为广泛比较嵌入模型在各种公共领域任务中的检索性能的事实标准，但此类基准难以充分捕捉并

评估专业术语的词汇和语义处理能力，以及航空航天专用检索的核心任务导向型信息需求。

航空航天公司通常依赖内部技术文档和设计报告，而这些资料不会对外公开。然而，安全和知识产权方面的限制使这类文档难以直接用作公开基准。NASA 技术报告服务器（NTRS）上的文档 [Nelson等人， 1995] 普遍遵循工程学文档的写作惯例，例如需求陈述、设计依据的阐述、测试与验证结果的报告、数值和单位标注，以及对标准的引用；在形式和实质内容上，它们都与内部技术文档相似。尽管 NTRS 大量归档并提供航空航天科学与工程成果，可作为内部运营文档的公开替代来源，但尚未以 NTRS 为数据源建立查询—段落相关性集合。在其他领域，如医学和法律，研究者利用大规模数据源，并结合用户行为或程序性知识，构建了领域专用信息检索基准 [Rekabsaz等人，2021年，Gao等人， 2024]。相比之下，尽管航空航天领域存在大规模数据源，但仍缺乏成熟的评估基准，无法准确衡量系统使用了哪份文档以及其中的哪些具体证据来回答查询。

为弥补这一空白，本文提出了 **STELLA** （**S**elf-Reflective**TE**rmino**L**ogy-Aware Framework for Bui**L**ding an **A**erospace Information Retrieval Benchmark，自我反思、术语感知的航空航天信息检索基准构建框架）。STELLA 框架包括：（1）从 NTRS 航空航天报告中系统地提取以文本为中心的文档；（2）将提取的文档拆分为语段，作为检索增强生成（RAG）中的检索单位；（3）从语段集合中精确提取航空航天术语（简称“术语”），以构建术语词典。该过程通过模式匹配、词性标注以及基于术语相对于通用语料稀疏程度的特异性过滤，构建领域专用术语词典。接下来，（4）利用构建的术语进行候选段落筛选，选出作为查询—段落对中段落一侧的候选段落。首先筛选出至少包含五个不同术语项的段落，然后将其划分为五种查询意图类别（例如定义／原理、比较／权衡）；这些类别是在与领域专家合作的基础上确定的。随后，在每种意图对应的段落池中应用 k-中心点聚类，为每种意图提取具有代表性的段落，并将其用作候选段落。之后，（5）根据这些代表性段落生成合成查询（合成查询生成）。这一阶段旨在构造能够衡量嵌入模型是否有效理解并表示航空航天术语的查询。生成的查询分为两种：术语一致查询（TCQ），其中逐字包含术语，用于评估词汇匹配；以及术语无关查询（TAQ），通过使用术语描述来评估语义匹配。在此过程中，将密度链（CoD） [Adams 等人， 2023] 以及自我反思 [Madaan 等人，2023年，Shinn 等人，2023年，Wang 和 Atanasova，2025] 方法应用于大语言模型，以提高查询质量。最后，（6）进行跨语言扩展，以体现航空航天业的全球协作环境。TAQ 会完整翻译为目标语言，而 TCQ 则采用混合翻译方案：保留英文术语，仅翻译其余部分，从而模拟真实用户的查询形式。

总而言之，STELLA 框架通过以下方式支持以可复现、面向实践的方式改进检索增强生成系统： (a) 使用领域一致的航空航天文档进行段落构建；(b) 通过术语提取和查询意图分类系统地筛选候选段落；(c) 采用可同时评估词汇匹配和语义匹配的双类型合成查询生成；以及 (d) 通过反映真实使用模式的跨语言扩展。本文的贡献如下。

1. **STELLA方案**：我们提出了一套特定领域的信息检索评估集构建流程，涵盖从 NTRS 中系统提取以文本为中心的语段，到合成查询生成；同时，我们构建并发布了由此得到的 STELLA 基准。

2. **双类型合成查询策略**：我们提出了一种新颖的合成查询生成策略，可分别测量嵌入模型的词汇匹配（TCQ）和语义匹配（TAQ）能力。在此过程中，我们应用 CoD 和自我反思来生成高质量查询。

3. **面向实践的跨语言扩展**：我们定义了翻译规则，在 TCQ 翻译中保留英语术语，并在 TAQ 翻译中进行完整翻译，以反映真实的查询实践；此外，我们提供了涵盖六种语言的跨语言评估集。

我们期待 STELLA 能成为一项基础设施，以严谨构建、面向实践的数据为依托，推动航空航天RAG系统实现可复现、可靠的改进。

2 相关工作

2.1 通用信息检索基准

用于客观评估检索性能的信息检索基准主要围绕开放域信息检索数据集构建。其中具有代表性的是 BEIR [Thakur等人， 2021] 该基准包含 18 项涵盖新闻、维基百科和科学文章等不同领域的信息检索任务，已成为广泛比较稠密与稀疏检索器零样本性能的事实标准基准。MTEB [Muennighoff等人， 2023] 和 KILT [Petroni等人， 2021] 等基准也为多任务学习和知识密集型任务提供了扩展的评估框架。然而，这些通用基准大多基于通用文本和文档，因此在捕捉专业领域复杂且特定领域的内容方面存在局限。

相比之下，STELLA基准使用直接从NTRS提供的航空航天领域来源中收集的材料构建语段。因此，STELLA提供的评估集在概念上与实际业务查询相契合，并以领域内一致的航空航天知识、术语和上下文为基础。这为更准确地揭示领域专用检索难度以及不同模型之间难以通过通用基准捕捉的性能差距提供了依据。

2.2 领域专用信息检索基准

医学和法律等领域的领域专用信息检索基准利用大规模数据源并反映文档特征，改进了查询—段落映射。Rekabsaz等人 [2021] 利用医学搜索引擎的点击日志，构建了一个大规模的基于点击的训练与评估数据集，用于信息检索。Gao等人 [2024]从官方判决网站收集源数据，并分析了对法律知识不熟悉的典型用户的查询风格，以构建法律案例检索数据集。
然而，在航空航天领域，尽管存在 NTRS 等大规模资源，但基本上没有标准化的查询—段落相关性数据集可用于衡量检索器质量。现有尝试主要聚焦于基于文档的摘要生成和问答等下游任务，或仅涵盖航空航天领域的部分子集 [Emmons等人，2024年，Oderinde等人， 2025]。换言之，尚未系统建立专门针对航空航天领域核心信息检索任务的标注方案。因此，本文提出了一个基于系统构建的指南、面向航空航天领域的专用信息检索基准，具有重要意义。

2.3 多语言信息检索基准

鉴于全球协作在航空航天业中的重要性，信息检索系统必须能够稳健地处理多语言查询和文档。近年来，MIRACL 等多语言信息检索基准（Zhang等 [2023] ）利用以维基百科为中心的大规模开放域资源，广泛评估跨语言检索性能，为模型的语言泛化能力提供了有意义的参考 [Bonifacio 等，2021年，Zhang等，2021]。然而，这些基准通常采用通用领域设定。
相比之下，航空航天领域文档主要采用形式化文体，其术语体系和推理线索在缩写、标准、部件标识符以及数值与单位换算等方面各不相同。此外，信息需求更偏向任务导向——关注原因与结果、流程和限制条件，而非简单的“事实检索”。因此，在跨语言开放领域环境中表现良好的模型，在特定领域条件下可能会因领域偏移和规范化失败而出现性能被高估或表现不一致的情况 [Voorhees 等, 2021年, Liu 等, 2023a]。

因此，尽管多语言通用基准仍是有用的起点，我们仍需要单独构建一个特定领域的跨语言信息检索评估集，以体现航空航天语境下的文档结构、术语体系和检索目标。我们通过直接从航空航天文档中构建语段，并结合面向任务的查询和术语词典，衡量检索模型的实际适用性。

2.4 信息检索合成查询生成

近年来，利用合成查询为信息检索模型构建训练和评估数据，作为一种借助大语言模型能力的高效数据构建方法，日益受到关注。Bonifacio 等 [2022] 提出从大语言模型生成大量查询，并将其映射到正例段落；Dai 等 [2023] 则证明，只需少量示例生成的查询，就能训练出强大的稠密检索器。Chaudhary 等 [2024] 研究了利用特定任务示例生成合成查询的方法，并通过实验证实了这些方法的有效性。他们通过使用与特定领域相关的示例重构提示词，而非使用通用领域的示例

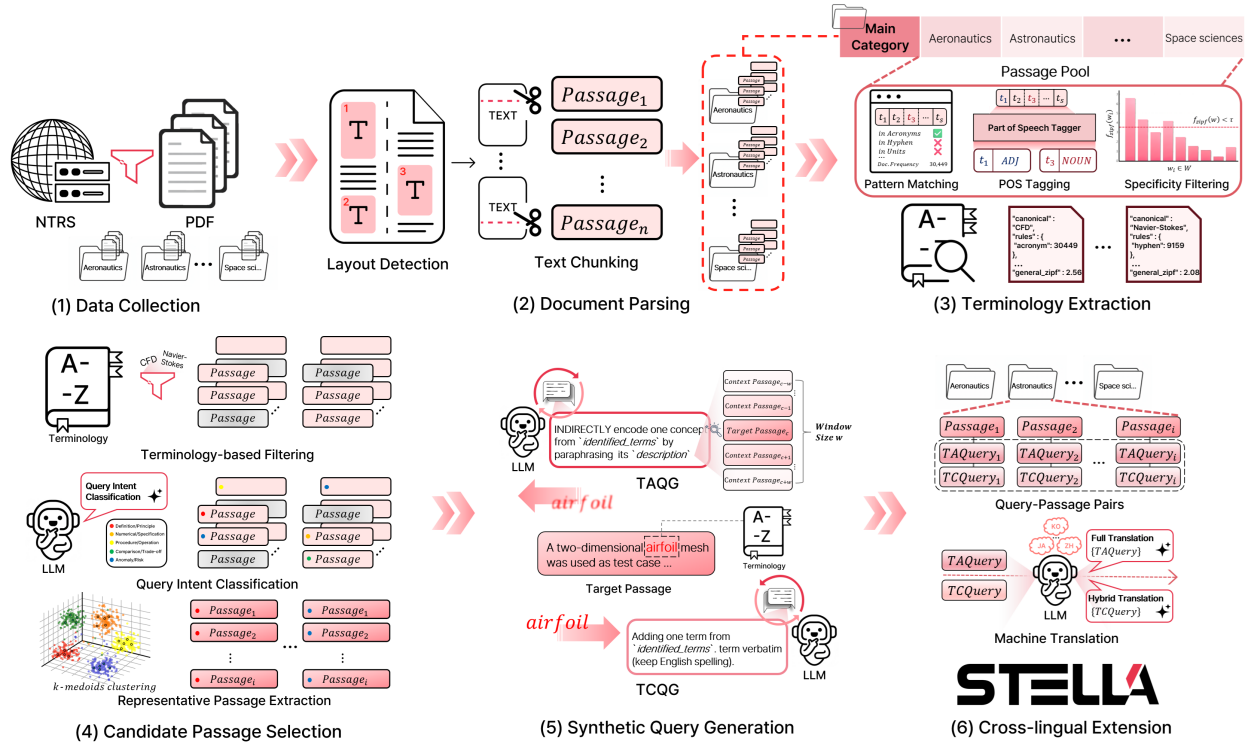

图 1：构建 STELLA 基准的总体流程。该过程包括六个系统化阶段：(1) 从 NASA 技术报告服务器（NTRS）收集数据，(2) 检测文档布局并进行段落分块，(3) 构建领域专用术语词典，(4) 根据查询意图筛选候选段落，(5) 使用密度链和自我反思生成两种类型的合成查询（TCQ 和 TAQ），以及 (6) 通过多语言查询翻译进行跨语言扩展。

（例如 MS MARCO [Campos 等中的示例），2016], 构建了数据集，并通过实验表明，使用这些数据训练检索器可进一步提升性能。这表明，领域特定示例有助于使合成查询的分布和风格与目标领域相契合。

综上，这些研究表明，基于大语言模型知识构建信息检索数据，无需耗费高昂成本进行人工标注，也能生成有效的数据集。我们将这一从段落到查询的通用合成生成范式应用于航空航天领域。然而，仅依赖示例也存在局限，例如：(i) 段落中的关键实体可能被遗漏或存在偏差；(ii) 违反禁止的格式要求；(iii) 意图分类错误且风格不稳定。为解决这些问题，我们将密度链（CoD）概念 [Adams 等人，2023], 应用于查询生成；该概念已在摘要生成中证明有效。我们还在每个步骤中执行自我改进式 [Madaan 等人，2023年，Shinn 等人，2023年，Wang 和 Atanasova，2025] 自我反思循环。密度链方法通过逐步识别目标文本中的关键实体并将其添加进来，以提高信息密度；自我反思则是一种方法，大语言模型会在生成过程中针对自身输出生成反馈，并据此进行改进，以提升最终质量。STELLA 将这两种方法应用于航空航天领域的查询生成，以系统地识别术语等关键实体，并生成能够体现复杂查询意图的精准合成查询。以往工作侧重于基于示例的风格对齐，而 STELLA 框架的特点在于其提示策略，旨在最大限度地提高航空航天领域专用查询的准确性。

3 STELLA 基准：构建流程

本节详细介绍 STELLA 基准的构建过程。STELLA 基准是一个面向航空航天领域的信息检索评估集。STELLA 框架由六个阶段组成，构成一套系统化流程：（1）筛选并收集大规模源文档；（2）进行文档布局检测和段落分块；（3）构建术语词典；（4）筛选候选段落；（5）生成合成查询；（6）扩展数据集，以支持跨语言评测。图 1 展示了该流程的顺序，以及 STELLA 基准由多个模块组合构建而成，而非单一的整体任务。

主要类别	总计	可用数量	不可用数量
航空学	4,372	2,860	1,512
航天学	2,622	1,068	1,554
化学与材料	4,706	2,935	1,771
工程学	5,935	3,564	2,371
地球科学	9,925	3,468	6,457
生命科学	9,185	2,331	6,854
数学与计算机科学	2,554	1,350	1,204
物理学	5,908	2,935	2,973
社会与信息科学	301	102	199
空间科学	18,405	5,864	12,541
总数	**63,913**	**26,477**	**37,436**

表1：按主要类别统计的已收集 NTRS 文档。“总数”表示发布日期为 2000 年或之后的 PDF 文档数量（时效性标准），“可用数量”表示除满足时效性标准外，还满足其余收集标准（文档格式、类别选择和版权）的文档数量。

主要类别	无下载链接	重复项	类型无效	版权无效	排除总数
航空学	235	211	830	236	1,512
航天学	552	139	746	117	1,554
化学与材料	623	162	857	129	1,771
工程学	715	259	1,190	207	2,371
地球科学	3,253	274	1,517	1,413	6,457
生命科学	3,752	160	2,726	216	6,854
数学与计算机科学	456	127	493	128	1,204
物理学	1,706	189	897	181	2,973
社会与信息科学	20	30	104	45	199
空间科学	7,150	385	3,185	1,821	12,541
总计	**18,462**	**1,936**	**12,545**	**4,493**	**37,436**

表2：按类别和第3.1节所述筛选标准划分的排除文档明细。各列对应具体的排除原因：技术不可用（无下载 URL ）、冗余处理（重复项）、格式限制（类型无效）和许可限制（版权无效）。排除总数与表 1 中的 “不可用数量”一致。

3.1 数据收集

该基准所依托的语料库收集自美国国家航空航天局技术报告服务器（NTRS）。NTRS 是航空航天领域可靠的公共技术资源。为确保领域一致性，我们选用了与真实运营文档在形式和内容上具有相似性的 NTRS 文档。为确保数据质量并使其适用于信息检索任务，我们依据以下标准筛选文档。

- **文档格式**：为兼顾文本提取的便利性和信息密度，我们仅收集 PDF 格式的英文文档，并排除以非文本内容为主的类型：“视频”“海报”“演示文稿”和“摘要”。
- **时效性**：为尽量减少较早期文档可能出现的解析错误和图像质量下降问题，我们仅纳入发布日期为 2000 年及之后的文档。
- **类别选择**：我们将收集范围设定为涵盖 NTRS 的全部十个主要类别（例如“航空学”“航天学”），并根据类别目录存储文档。当文档属于多个类别时，我们使用文档标识符（文档 ID）去除重复项，以确保数据一致性。
- **版权**：为确保所收集的材料可自由重复使用和分发，我们仅选择不受版权保护的文档，并排除未明确标注版权信息的文档。

对于每份收集到的文档，我们还提取了文档 ID、标题和作者等关键元数据，以便于管理。按类别统计收集到的文档结果汇总于表1和表2中。

类别	# 语段
航空学	306,505
航天学	97,773
化学与材料	225,136
工程学	292,337
地球科学	362,624
生命科学	197,600
数学与计算机科学	130,214
物理学	242,575
社会与信息科学	10,516
空间科学	539,764
总计	**2,405,044**

表3：按类别构建的语段统计信息。文本使用递归令牌分块器进行分块，大小为100个词元，重叠为20个词元。

3.2 文档解析

我们采用以下方法从收集到的 PDF 文档中提取文本，并将其切分为语段，作为检索单元。对于文档布局检测，我们使用了 DocLayout-YOLO [Zhao 等人，2024b]，该模型在各种文档布局中均表现出稳健的性能。DocLayout-YOLO 基于全局到局部架构，能够检测文档中的细粒度信息，并在包含 30 万份文档的合成数据集上进行预训练，因此在各种领域中均能实现稳健的布局检测性能。我们明确排除了可能给检索过程引入噪声的图形和表格，专注于纯文本内容，并在提取过程中移除了置信度低于 0.25 的边界框。最后，我们按照自然阅读顺序排列边界框，然后提取文本。

接下来，我们应用递归令牌分块器 [Chase, 2022] 对文本进行分块，将提取的文本划分为有意义的单位。正如 Amiri 和 Bocklitz [2025],所报告的那样，与定长分块方案相比，该方法能够更好地保留语义边界，从而提升语段连贯性。这对于帮助大语言模型清晰理解上下文并生成高质量查询至关重要。作为我们的分块策略，我们采用了 Amiri 和 Bocklitz [2025], 在多项实验中报告的最佳性能配置，即块大小为100个词元、重叠为20个词元（RT100-20）。该配置真实地模拟了实际检索增强生成系统中的分块环境，从而提高基准结果推广到实际应用的可能性。按类别构建的语段统计信息见表3。

3.3 术语提取

STELLA基准的一项关键目标是评估嵌入模型对航空航天术语在词汇和语义层面的掌握程度。为此，我们构建了一部高质量的航空航天术语词典，作为后续候选段落筛选（第 3.4 节）和合成查询生成（第 3.5 节）阶段的基础。

该词典首先从整个 NTRS 语料库中广泛提取候选技术术语，然后通过多阶段过滤进行精炼，仅保留高度特定于该领域的术语。总体过程包括 (1) 候选提取和 (2) 多阶段过滤。

3.3.1 候选提取

首先，我们通过基于正则表达式的模式匹配，从整个语料库中提取候选技术术语。此方法旨在捕捉航空航天与工程文档中技术术语的常见记法，包括：(a) 全部大写的术语（例如 CFD、MODIS 等缩写），(b) 带连字符的复合词（例如“Navier-Stokes”“XMM-Newton”），以及 (c) 包含单位、希腊字母（例如 α, β）或数学符号（例如“3-sigma”、H_2O）的术语。

3.3.2 多阶段过滤

为确保相关性和领域特异性，在第一步提取中获得的大量候选术语必须满足以下全部三项标准，才能纳入最终词典：

A. **文档频率**：为去除语料库中仅出现一次或因解析错误而产生的噪声，我们仅保留至少出现在十个不同段落中的术语。B. **词性标注**：鉴于大多数技术术语采用名词形式，我们使用 spaCy 库 [Honnibal 和 Montani, 2017] 进行词性标注，并仅将被识别为名词或专有名词的候选术语传递至下一阶段。C. **特异性过滤**：为有效去除“system”（系统）、“report”（报告）和“analysis”（分析）等常见于航空航天文档、但领域特异性较低的通用词汇，我们采用特异性过滤。我们使用 wordfreq [Speer, 2022], 估算术语的一般普及程度；该工具汇总了多个大型开放领域语料库（包括维基百科、Reddit 和 Common Crawl）中的词语使用统计数据。我们为每个术语计算一般频率得分 τ，并删除得分超过特定阈值（$\tau > 3.5$）的术语，即在通用领域用法中非常常见的术语。最终，仅将满足 $\tau \leq 3.5$ 的术语视为航空航天领域专用术语。该阈值是经过经验选择的数值，旨在平衡领域特异性与覆盖率。初步分析证实，该阈值实现了有效权衡：既能去除领域特异性较低的通用词汇（如“system”和“report”），又能保留“propellant”（推进剂）和“airfoil”（翼型）等核心技术术语，尽管这些术语在航空航天领域内也相对频繁出现。

通过这三阶段过滤过程的术语构成 STELLA 术语词典，并作为候选段落筛选和合成查询生成的关键资源。

3.4 候选段落筛选

本节旨在从第 3.2 节构建的完整 NTRS 段落池中，系统提取信息丰富且具有代表性的段落，作为合成查询生成 （第 3.5 节）的基础。该过程包含三个阶段：（1）基于术语的过滤；（2）查询意图分类；以及（3）代表性段落提取。

3.4.1 基于术语的过滤

第一阶段从整个 NTRS 语料库中粗略筛选领域特异性高、因而信息价值也高的段落。我们使用第 3.3 节中构建的术语词典，筛选出至少包含五个不同术语条目的段落。此步骤排除了技术密度较低的通用叙述性或行政性段落，得到一个聚焦于核心航空航天概念的段落池。

3.4.2 查询意图分类

第二阶段根据查询意图对经过一次筛选的语段进行分类，以反映真实用户的信息需求。首先，我们与航空航天领域专家合作，确认了实践中用户通常向检索增强生成系统提出的信息类型，可归纳为五种核心意图。为了将这些实际需求纳入基准，我们使用大语言模型对经过一次筛选的语段进行了意图分类。

这一过程旨在让大语言模型充当分类器：给定提示词后，判断每个语段最适合生成哪类查询。本文使用的所有大语言模型均为 GPT-5（Brown 等人） [2020], , 此步骤使用的提示词见附录 A.1。五种查询意图如下：

- **定义／原理（定义）**– 区分概念，解释机制／近似／变量依赖关系。
- **数值 / 规格（数值）**– 数值、范围、假设、单位、不确定性（避免仅查询单个数值）。
- **程序 / 操作（过程）**– 步骤、初始化/校准、计划安排、操作规则。
- **比较 / 权衡（比较）**– 对不同选项/配置的性能/质量/功率/裕度进行定量比较。
- **异常 / 风险（异常）**– 原因、复现条件、缓解措施/复发预防。

通过此步骤，筛选后的段落池被重新整理为五个互斥的意图专属段落池。

3.4.3 代表性段落提取

最后阶段从大型意图专属段落池中提取最能代表各意图的语段。为此，我们首先将每个意图专属段落池中的所有语段转换为向量。我们使用 embeddinggemma-300m [Vera 等人,2025] 作为嵌入模型。由于在 MTEB 上，该模型在同等规模的模型中展现出最佳的英语嵌入性能，因此我们认为它适合有效捕捉语段之间的语义关系。

随后，我们对每个意图的嵌入池进行 k-中心点聚类。 k-中心点聚类会选取实际数据点（中心点）作为簇中心，因此中心点对离群值具有鲁棒性，且易于解释。对于每个特定意图的池，我们将簇数设为 $k = 5$。在初步分析中，我们通过实证权衡确定了 $k = 5$ ，既能捕捉每种意图下的不同子主题，又能形成稳定的簇。

聚类后，我们从每个特定意图的池中，选取距离该意图对应的五个中心点最近的 20 个语段。这样，每种意图可得到 $i = 100$ 个代表性语段（5×20）。最终，五种意图共计 500 个语段被确定为候选段落，并作为第 3.5 节合成查询生成的段落来源。

3.5 合成查询生成

基于第 3.4 节提取的候选段落，我们使用大语言模型生成合成查询。STELLA 基准的根本目标是衡量嵌入模型对航空航天术语及其相关语境的理解深度。为此，我们生成双类型查询，以便分别评估嵌入模型的两项核心检索能力。

1. **术语一致查询（TCQ）**：段落中出现的术语会逐字包含在查询中，用于评估词汇匹配能力。2. **术语无关查询（TAQ）**：查询不包含术语本身，而是包含对该术语的描述或定义。此方法用于评估模型能否在不依赖术语表面形式的情况下进行概念匹配和语义匹配。

两种类型的查询　　均采用结合 CoD 与自我反思方法的提示策略生成　　s
以控制生成过程中的信息 密度和质量。

3.5.1 用于质量控制的生成框架：CoD 与自我反思

朴素的段落到查询生成往往会产生含义模糊、过于简单或遗漏段落中关键信息的查询。为克服这些局限并生成高质量查询，我们采用了 CoD 和自我反思。

- **密度链 (CoD)**：我们将最初经验证用于摘要生成的密度链方法改用于查询生成 Adams 等人 [2023]。该过程从一个初始种子查询开始，分三个步骤逐步增加信息密度，将段落中出现的关键实体补充进去。这种方法既能减少歧义，也能避免查询过于密集、包含不必要的信息。
- **自我反思**：在每个 CoD 步骤中，我们都会加入自我反思机制，让大语言模型评估并修订自身的查询 [Madaan 等人，2023年；Shinn 等人，2023年；Wang 和 Atanasova， 2025]。大语言模型会检查是否违反了本文定义的八项硬性约束（见附录 A.2）——例如：“是否仅凭给定段落就能回答此查询？”“是否包含单数字查找或列表请求等禁止的格式？”以及“是否在保留指定意图的同时遵守词元长度限制？”——并据此完善输出。即使在后续步骤中查询变得更加复杂，这一机制仍是维持其预期用途和格式的关键。

3.5.2 TCQ 和 TAQ 生成过程

这两类查询都遵循 CoD 和自我反思的三步框架，但在提高信息密度的方式上有所不同。

术语一致性查询生成（TCQG）从第一步生成的种子查询出发，TCQ 在第二步和第三步中通过明确加入段落中的术语（依据第 3.3 节构建的词典）进行优化。因此，这一过程生成的 TCQ 与源段落之间具有较高的词汇重叠。

非术语约束查询生成（TAQG）由于 TAQ 旨在衡量语义理解，因此严格禁止在查询中包含术语本身。相反，它通过间接将术语的描述加入第一步生成的种子查询中，来提高信息密度。此处使用的描述是在 TAQG 之外的一个初步步骤中构建的。

为了生成特定段落（段落索引 c）中术语的描述，我们参考该段落在原始文档中的上下文。具体而言，我们以目标段落为中心，采用大小为 $w = 2$ 的窗口，将从 $c - 2$ 到 $c + 2$ 的五个连续段落作为扩展上下文段落。随后，大语言模型根据这一扩展上下文生成简洁的术语描述。通过将这些描述用作 TAQG 的构建模块，我们鼓励模型理解诸如“燃烧后推动火箭的化学物质”之类的描述，从而检索出包含术语“推进剂”的段落，而不是依赖“推进剂”一词本身。合成查询生成和术语描述生成所用的具体提示词见附录 A。通过第 3.5 节中的整个过程，我们为 500 个候选段落中的每个段落生成一个 TCQ 和一个 TAQ，最终得到包含 1,000 个独特（查询，段落）对的评估集。

3.6 跨语言扩展

为体现航空航天业的全球协作特性，我们扩展了基准，以评估跨语言检索性能。基于第 3.5 节生成的 1,000 组英文（查询，段落）对，我们仅将查询部分翻译成多种语言。选择目标语言时考虑了类型学多样性，涵盖不同的语系和文字系统。这六种语言——韩语（ko）、印度尼西亚语（id）、泰语（th）、法语（fr）、中文（zh）和日语（ja）——分别代表不同的主要语系。
这种设计使 STELLA基准能够稳健地评估嵌入模型在多种语法结构、书写系统和分词方案下的跨语言检索性能。翻译通过使用大语言模型进行提示学习完成（见附录 A.3）。然而，重要的是不能止步于简单的机器翻译，还应反映真实的查询行为。航空航天领域的工程师和研究人员在用母语搜索文档时，往往仍会保留“RSRM”或“推进剂”等核心技术术语的英文原文。

为准确地将 这一现实做法纳入其中，我们针对 TAQ 和 TCQ 采用了不同的翻译规则 ：

- **TAQ 翻译**：将整个查询完整翻译成目标语言。由于 TAQ 由术语“描述”构成，完整翻译使我们能够衡量在该语言环境下纯粹的语义检索能力。
- **TCQ 翻译**：我们采用混合翻译，即保留查询中包含的术语（使用第 3.3 节中的词典识别）并维持其英文形式，仅将描述性部分翻译成目标语言。为此，我们精心设计了大语言模型提示词，以确保术语得到保留（见表12）。

最后，扩展后的 STELLA 基准涵盖六种语言，连同原始英文数据集在内，所有数据集均采用与 BEIR 完全兼容的标准化模式；BEIR 是一种广泛使用的信息检索评估框架。这使其他研究人员能够轻松使用 STELLA 基准，并复现模型性能、将其与现有系统进行比较。该基准旨在推动航空航天信息检索研究生态系统的发展，并促进可复现研究。

4 STELLA基准的验证

在本节中，我们将验证第 3 节提出的 STELLA 基准构建流程的各个阶段在统计和方法论上的可靠性。这些验证为第 5 节中的实验提供了关键证据。我们依次证明（1）查询意图分类的准确性、（2）合成查询的质量，以及（3）跨语言翻译的保真度，从而确立该基准总体上的可靠性。

4.1 意图分类的质量验证

我们在第 3.4.2 节中通过衡量大语言模型分类器重现专家意图标签的准确程度，验证意图分类步骤。一位领域专家从术语过滤池中随机抽取了 $N = 300$ 个语段，并为每个语段准确分配五种意图（定义、数值、过程、比较、异常）中的一种。随后，我们将同样的 300 个语段输入大语言模型分类器，并以专家标签为参照，使用 F1 分数评估预测结果。

面板 A：总体（5 类）		**N**	**F1**
微平均 F1		300	0.933
宏平均 F1		300	0.928
面板 B：按意图（以专家为参照）		**支持**	**F1**
Def	定义／原理	82	0.930
Num	数值 / 规格	65	0.920
Proc	程序 / 操作	73	0.940
Comp	比较 / 权衡	51	0.910
Anom	异常 / 风险	29	0.940

表 4：在 $N = 300$ 个语段上对意图分类器进行基于 F1 的验证。微平均 F1 汇总所有样本的结果，而宏平均 F1 对各意图的 F1 取平均，以减轻类别不平衡的影响。表中还报告了按意图划分的 F1 以及专家标注的支持样本数。

由于各意图的频次分布不均，我们同时报告**微平均 F1** 和 **宏平均 F1**。微平均 F1 反映所有样本汇总后的总体性能，而宏平均 F1 则对各意图的 F1 取平均，因此能够突出模型在少数类意图上的性能。我们还报告了按意图划分的 F1，以及专家支持样本数。

表 4 显示，分类器的总体性能良好（Micro-F1 = 0.933，Macro-F1 = 0.928）。按意图的结果表明，各意图的 F1 值始终较高，而 Comp 和 Num 的 F1 值相对较低，这说明在单标签分配下，这些意图更容易混淆。

局限性。 本次验证仅使用了一名专家标注员；因此，报告的分数衡量的是与专家标签的一致程度，而非专家间一致性。未来的工作将通过裁定扩展这项验证，采用多专家标注。

4.2 合成查询的质量验证

我们开展了实验，以验证第 3.5 节中介绍的自我反思在提升合成查询质量方面的效果。为此，我们使用了八种近期的大语言模型，并比较了采用和未采用自我反思时生成的查询质量。在评估中，我们从每个大语言模型生成的查询中随机抽取 100 个 TAQ 和 100 个 TCQ，并使用 G-Eval 框架 [Liu 等，2023b]对其评分。查询质量使用 G-Eval 框架进行评估，依据以下五项核心指标。

1. **可回答性**：查询是否仅凭给定段落即可回答。2. **无外部知识**：查询生成过程中是否排除了外部知识。3. **意图遵循**：是否遵循预定义的查询意图。4. **格式合规性**：是否避免了禁止的格式（例如，要求列出清单、回答单个词）。5. **风格与长度**：是否满足指定的风格（中性、技术性）和长度限制。

每项指标按 1–5 分评分，最终得分为五项指标的平均值。根据参数规模，将受评估的模型分为两类。

- **大规模模型（参数量超过 200B 或采用混合专家架构）**：GPT-5 Brown 等人 [2020], DeepSeek-V3.2-ExpDeepSeek-AI [2025a], Qwen3-235B-A22B-Instruct Yang 等人 [2025], 和 Llama-4-Maverick Meta AI [2025]。这些都是最先进的旗舰模型，拥有丰富的知识和先进的推理能力。
- **小规模模型（70亿–80亿参数，稠密）**：Qwen3-8B Yang 等人 [2025], Llama-3.1-8B-Instruct AI [2024], Mistral-7B-Instruct-v0.3 Jiang 等人 [2023], 和 DeepSeek-R1-0528-Qwen3-8B DeepSeek-AI [2025b]。这些模型在设计时注重效率，适用于资源受限环境。

大规模模型上的效果。如图2所示，与未采用自我反思的设置相比，采用自我反思设置的雷达图在所有维度上都向外扩展。尤其是，GPT-5 在可回答性轴上的变化最为显著，得分从 3.22 大幅提高到 3.99（+0.77）。这表明，针对生成的查询是否仅凭段落内容即可回答这一问题进行自我反思，非常有效。此外，格式合规性和无外部知识这两个维度接近



图2：通过 G-Eval 对合成查询进行质量验证。比较采用自我反思（红色实线）与未采用自我反思（灰色虚线）的合成查询生成在五项核心指标上的性能：可回答性、无外部知识、意图遵循、格式合规性以及风格与长度。上排展示大规模模型，在可回答性和约束遵循方面有显著提升。下排展示小规模模型，由于推理能力有限，自我反思带来的收益有限，甚至会产生负面影响。

所有模型的最高分 5.0，表明自我反思在落实硬性约束方面发挥着关键作用。

小规模模型的局限性。相比之下，在小规模模型组中，自我反思的效果甚微，甚至可能产生负面影响。对四种小规模模型取平均后发现，应用自我反思时，可回答性得分有所下降，总体平均得分也略有降低。我们认为，这是因为小规模模型的推理能力有限。自我反思要求模型批判性地评估并修改自身输出，这是一项复杂任务，小规模模型可能难以胜任。在某些情况下，不准确的自我反思似乎会降低原本合理的生成结果的质量。Qwen3-8B 是一个例外，其性能略有提升，不过与大规模模型相比，提升幅度仍然有限。

基于这些发现 ，本基准中的合成查询最终采用了 GPT-5 并结合 h
自我反思， 其生成的质量最高。

4.3 跨语言翻译的质量验证

我们进行了两项验证，以确保第3.6节所述基于大语言模型的翻译质量。首先，我们通过回译对语义保真度进行了定量评估。将所有查询翻译成六种语言，再回译成英语后，我们测量了原始查询与回译查询嵌入向量之间的余弦相似度。所有语言的平均得分均达到 0.93 或更高，证实翻译过程中很好地保留了原意。

其次，我们自动检查了所有翻译后的 TCQ，以验证其是否符合混合翻译规则。我们确认，所有查询都准确保留了英语术语，这表明该基准成功模拟了真实用户的查询形式。

模型	架构	参数	Ref.
词汇基线			
BM25	概率式	N/A	Robertson 等人 [1995]
仅编码器（第*1*组）			
Arctic-Embed-2.0-L	双编码器	0.6B	Yu 等人 [2024]
BGE-M3	双编码器	0.6B	Chen 等人 [2024]
mE5-instruct	双编码器	0.6B	Wang 等人 [2024]
mGTE	双编码器	0.3B	Zhang等人 [2024]
仅解码器（第 *2* 组）			
Llama-Embed-Nemotron	大语言模型解码器	8B	Babakhin 等人 [2025]
Qwen3-Embedding	大语言模型解码器	8B	Zhang等人 [2025]
SFR-Embedding-Mistral	大语言模型解码器	7B	Meng 等人 [2024]

表 5：评估模型汇总

5 实验

5.1 实验设置

为了验证 STELLA基准的实用性，并从多个角度分析航空航天领域的检索性能，我们评估了一个词汇基线和七种近期的神经嵌入模型。这些模型根据以下三个关键维度选取：

1. **架构：** 我们比较传统的仅编码器双编码器模型与近期的仅解码器大语言模型衍生模型。2. **模型规模：** 我们分析轻量级模型（参数量为 0.3B–0.6B ）与大型模型（参数量为 70 亿参数 –80 亿参数）在知识容量和效率之间的权衡。3. **多语言能力：** 我们考察各模型是否支持多语言环境，以便与第3.6节中的跨语言扩展保持一致。

表 5 全面总结了评估的模型。根据架构范式，评估的模型分为词汇基线和两组神经嵌入模型。

词汇基线。

• **BM25**：一种经典的概率信息检索排序函数，基于词汇匹配而非语义相似度进行运作。在我们的实验中，BM25 作为强基线，用于评估第 3.5 节所定义的 TCQ 和 TAQ 性能，并与神经嵌入模型的语义检索性能进行比较（Robertson 等人 [1995]）。

第1组：仅编码器架构。这些模型遵循传统的双编码器范式，特点是规模相对较小、推理效率较高。

• **Arctic-Embed-2.0-L（0.6B）**：基于 bge-m3-retromae 微调而成，旨在兼顾高检索性能与推理效率 Yu 等人 [2024]。
• **BGE-M3（0.6B）**：通过对 XLM-RoBERTa Conneau 等人 [2020]进行微调开发而成，其特点是具备多功能性 Chen 等人 [2024]。
• **mE5-instruct（0.6B）**：源自 E5 系列的多语言指令微调模型 Wang 等人 [2024]。
• **mGTE（0.3B）**：属于 GTE 模型系列，旨在多语言环境下实现高效且强大的检索性能 Zhang 等人 [2024]。

第 2 组：仅解码器架构。这些是近期模型，将参数量超过 70 亿参数的大语言模型应用于嵌入任务，其特点是凭借大量预训练获得了强大的语义理解能力。

• **Llama-Embed-Nemotron（80亿参数）**：基于 Llama-3.1-8B，并将解码器原有的单向注意力替换为双向自注意力，从而能够在完整的词元上下文中实现更丰富的语义理解 Babakhin 等人 [2025]。

模型	总体	Gap	术语一致查询（TCQ）						术语无关查询（TAQ）					
			Anom	Comp	Def	Num	Proc	Avg	Anom	Comp	Def	Num	Proc	Avg
词汇基线														
BM25	0.659	0.228	0.871	0.858	0.572	0.677	0.886	0.773	0.616	0.609	0.458	0.434	0.609	0.545
仅编码器（第*1*组）														
Arctic-Embed-2.0-L	0.672	0.227	0.870	0.839	0.598	0.765	0.854	0.785	0.639	0.611	0.438	0.496	0.608	0.558
BGE-M3	0.628	0.272	0.808	0.841	0.563	0.761	0.846	0.764	0.596	0.476	0.353	0.515	0.520	0.492
mE5-instruct	0.476	0.230	0.628	0.646	0.442	0.581	0.656	0.591	0.409	0.382	0.287	0.314	0.412	0.361
mGTE	0.572	0.220	0.739	0.751	0.490	0.664	0.768	0.682	0.551	0.517	0.310	0.447	0.487	0.462
仅解码器（第 *2* 组）														
Llama-Embed-Nemotron	**0.788**	**0.106**	**0.872**	**0.887**	**0.684**	**0.872**	**0.888**	**0.841**	**0.792**	**0.810**	**0.532**	**0.753**	**0.787**	**0.735**
Qwen3-Embedding	0.694	0.171	0.818	0.812	0.644	0.753	0.868	0.779	0.668	0.637	0.528	0.520	0.688	0.608
SFR-Embedding-Mistral	0.660	0.183	0.799	0.844	0.555	0.760	0.798	0.751	0.613	0.657	0.373	0.547	0.651	0.568

表 6：STELLA基准英语子集上的综合检索性能。本表展示了**总体** nDCG@10 分数，以及**术语一致查询（TCQ）与术语无关查询（TAQ）之间的性能差距**，差距越小表示词汇依赖性越低。表中详细列出了在 TCQ 和 TAQ 设置下五种特定查询意图的性能细分。最佳结果以**粗体**标示，次佳结果加下划线。

- **Qwen3-Embedding（80亿参数）**：来自 Qwen3 系列的嵌入模型，与 Llama-Embed-Nemotron 不同，它保留了核心解码器架构，同时优化了嵌入性能 Zhang等人 [2025]。
- **SFR-Embedding-Mistral（7B）**：基于 Mistral-7B-v0.1Jiang 等人的 [2023] 架构，并与 Qwen3-Embedding 一同进行评估，以衡量大语言模型解码器架构的嵌入性能 Meng 等人[2024]。

我们使用 nDCG@k（归一化折损累计增益，计算至 k）衡量模型的检索性能，这是信息检索领域的标准评估指标。nDCG@k 综合考量前-k 条检索结果的相关性和排序。在实验中，我们将 $k = 10$ 设为默认值，并重点关注前十条结果的质量（nDCG@10），这与典型用户场景一致。

5.2 总体性能比较

STELLA基准英语子集上的总体性能汇总于表 6。在分析跨语言能力之前，我们首先确定单语设置下领域特定检索性能的基线。这表明，近期神经模型并非总能在航空航天等专业领域占据优势。Llama-Embed-Nemotron 的表现大幅领先于其他模型，展现了其架构和规模的优势。其余模型之间的竞争则更为微妙。Arctic-Embed-2.0-L（0.6B）的表现令人瞩目，略胜 SFR-Embedding-Mistral（7B）和 BM25。

这表明，与单纯的模型规模相比，领域适用性和训练目标的有效性可能对性能有更大的决定作用。相比之下，mE5-instruct 和 mGTE 等旨在实现广泛通用性能的模型，表现不及 BM25 基线，揭示了它们在高度专业化领域中的局限性。

5.3 术语对TCQ与TAQ的影响

分析有无术语时的性能差异，可以清晰地揭示各模型的特征。词汇基线 BM25 在TCQ上的表现出色，因为其中明确出现了技术术语；但它在TAQ上的性能则急剧下降，导致约为0.228的词汇依赖差距（TCQ–TAQ）。

在神经嵌入模型中，Llama-Embed-Nemotron将这一差距缩小至约0.1053，展现出强大的概念理解能力，即使表层技术术语缺失，也能捕捉语境含义。相比之下，BGE-M3的差距甚至大于BM25（0.272），这表明它更像是一个关键词匹配器，而非能够捕捉专业领域深层语义联系的模型。

5.4 按查询意图分析

按查询意图来看，模型通常在比较／权衡和异常／风险方面表现良好。值得注意的是，在 TCQ 下，BM25 在程序／操作方面取得了 0.886 的极高分数。我们将其归因于程序性文档的性质：其中某些技术术语会被反复使用，因此词法检索尤其有效。

模型	Ref.	All	术语一致查询（TCQ）							术语无关查询（TAQ）						
	en	Avg	ko	id	th	fr	zh	ja	Avg	ko	id	th	fr	zh	ja	Avg
仅编码器（第*1*组）																
Arctic-Embed-2.0-L	0.672	0.524	0.661	0.717	0.643	0.722	0.677	0.698	0.686	0.345	0.422	0.290	0.402	0.317	0.393	0.362
BGE-M3	0.628	0.485	0.635	0.680	0.650	0.673	0.627	0.667	0.655	0.290	0.356	0.286	0.332	0.292	0.335	0.315
mE5-instruct	0.476	0.199	0.292	0.393	0.367	0.423	0.156	0.233	0.311	0.075	0.131	0.086	0.161	0.022	0.046	0.087
mGTE	0.572	0.383	0.475	0.546	0.461	0.554	0.584	0.523	0.524	0.196	0.288	0.165	0.310	0.271	0.219	0.242
仅解码器（第 *2* 组）																
Llama-Embed-Nemotron	**0.788**	**0.707**	**0.798**	**0.818**	**0.774**	**0.806**	**0.801**	**0.811**	**0.801**	**0.618**	**0.632**	**0.529**	**0.634**	**0.625**	**0.633**	**0.612**
Qwen3-Embedding	0.694	0.603	0.718	0.696	0.675	0.702	0.725	0.720	0.706	0.484	0.508	0.432	0.528	0.524	0.519	0.499
SFR-Embedding-Mistral	0.660	0.424	0.548	0.616	0.450	0.656	0.566	0.593	0.572	0.238	0.322	0.139	0.421	0.275	0.261	0.276

表 7：详细的跨语言检索性能。**Ref.** 表示表 6 中的单语性能。**All**表示所有跨语言设置下的总体平均值。对于 **TCQ** 和 **TAQ**，六种语言的详细分数之后，在最右侧列出各自的平均值（**Avg**）。最佳结果以**粗体**显示，次佳结果加下划线。

最具挑战性的意图是定义／原理，所有模型在此方面都表现不佳，尤其是在 TAQ 下。即使是 Llama-Embed-Nemotron，在这一设置下也只能达到 0.532，这表明，在缺乏术语匹配的情况下，即便是最先进的模型，检索抽象原理仍然十分困难。

5.5 架构见解

这些实验结果为设计领域专用嵌入模型提供了若干关键见解。

对大型基于解码器的模型的需求。 实验表明，词汇依赖差距较小、因而具备更好的概念理解能力的架构，是大型基于解码器的模型。尤其是，唯一明显超越强大 BM25 基线的模型是 Llama-Embed-Nemotron，该模型采用了双向注意力。与 Llama-Embed-Nemotron 相比，规模相近的单向模型（Qwen3-Embedding、SFR-Embedding-Mistral）性能较差，这表明领域专用检索同时需要模型规模和双向上下文理解。

编码器的效率。 Arctic-Embed-2.0-L（0.6B）的表现与规模大十倍以上的 SFR-Embedding-Mistral（7B）不相上下，这突显了编码器架构的效率。这一结果表明，模型规模和在通用基准测试上的表现都不一定能保证模型在真实世界或领域专用环境中表现出色，也凸显了开展适合领域的实验与评估的重要性。这也说明，像 Arctic-Embed-2.0-L 这样的模型在资源受限环境中可能是颇具吸引力的替代选择。

再次证明词汇基线的实力。 许多近期神经模型的表现不如 BM25，这表明在区分专业术语至关重要的领域中，词汇匹配仍极具竞争力。因此，在构建实用系统时，应将混合方法视为必不可少的方案。

5.6 跨语言性能

为体现航空航天业的全球协作环境，我们使用翻译成第 3.6 节所述六种语言的查询，评估了跨语言检索性能（表 7）。这项实验衡量了将多语言查询与英文段落进行匹配的能力。

总体性能。在单语设置中观察到的模型相对排名在跨语言设置中基本保持不变，但所有模型的性能均有所下降。Llama-Embed-Nemotron 在跨语言设置中仍以明显优势位居榜首。它的性能降幅最小，相较于单语基线仅下降约 8.1%，展现出强大的跨语言泛化能力。Qwen3-Embedding 在六种语言上的平均得分也达到 0.603，相较于单语设置下降了 9.1%。

相比之下，侧重于多语言支持的编码器模型性能显著下降。BGE-M3 相较于单语设置下降约 14.3%，Arctic-Embed-2.0-L 则下降约 14.8%。mE5-instruct 在六种语言上的平均得分为 0.199，较单语设置大幅下降 27.7%，这表明它在跨语言检索方面基本无效。相比之下，参数量约为其一半的 mGTE 模型在跨语言设置中的性能仅下降 18.9%，展现出稳健的检索性能。

语言层面的性能偏差与弱点。基准中的语言多样性揭示了各模型的明显弱点。对大多数模型而言，泰语（th）的性能降幅最大。泰语独特的文字系统和语法结构使其成为最具挑战性的语言；与其他语言相比，Llama-Embed-Nemotron 在泰语上的性能降幅也明显更大。mE5-instruct 尤其在亚洲语言上表现出严重弱点，在日语和中文上的性能

最低。有趣的是，SFR-Embedding-Mistral 在法语（fr）上表现出色，达到 0.539，但在泰语和韩语上的性能却大幅下滑，揭示了其语言偏差。

术语的影响。跨语言实验清楚地验证了第3.6节中设计的翻译规则的有效性。对于所有模型，保留英文术语的 TCQ 性能显著高于完全翻译后的 TAQ 性能。这是因为将技术术语保留为其原始英文形式，能够完美避开多语言模型尚未解决的领域专属知识跨语言对齐这一难题；我们的结果也提供了明确的经验证据，表明第3.6节中所假设的现实用户发出的“混合查询”是一种非常合理的策略。

对于 BGE-M3 等编码器模型，这一差距尤其明显。它在六种语言上的平均 TCQ 得分为 0.655，但 TAQ 得分降至 0.315，还不到前者的一半。与第5.3节的单语分析结果一样，这表明该模型仍然严重依赖表层形式术语匹配。即使是表现最佳的模型 Llama-Embed-Nemotron，其平均 TCQ 得分与 TAQ 得分之间也存在约 0.189 的较大差距。这个差距大于单语设置下观察到的差距，表明当前模型要同时克服概念理解和语言障碍，仍是一项极具挑战性的任务。

6 结论

本文提出 STELLA 框架，旨在解决航空航天领域缺乏对信息检索性能进行实际评估的问题，并介绍了由此构建的基准。STELLA 框架以双类型查询为核心，分别衡量词汇匹配（TCQ）和语义匹配（TAQ），通过自我反思进行质量控制，并采用反映真实使用模式的混合跨语言扩展。在验证该基准的可靠性并评估七种嵌入模型后，我们确认，TAQ 是当前神经嵌入模型面临的最具挑战性的设置。此外，BM25 基线的出色性能再次表明混合检索的必要性；即使在跨语言设置下，保留术语的 TCQ 也展现出实际效用。总之，STELLA基准为精准识别航空航天RAG系统所用嵌入模型的弱点提供了核心评估基础，而所提出的框架则有助于推动领域专属模型开发和混合检索策略的后续研究。

7 局限性与未来工作

STELLA 存在若干固有局限。首先，由于基准语料库仅依赖 NTRS 这一公共存储库，因此可能无法充分代表工业环境中的专有文档或其他机构的资料。其次，我们在构建过程中有意排除了图形和表格等非文本信息，因此该基准无法评估文本以外针对结构化或多模态数据的检索能力。第三，由于所有查询—段落对均由大语言模型通过段落到查询生成合成，因此难以充分模拟真实用户查询的多样性；此类查询可能含糊不清、需要整合多份文档（多跳），或无法作答。第四，“至少五个术语”筛选条件和五种意图类别等启发式假设贯穿整个构建流程，可能过度简化现实世界中信息需求的复杂性。最后，跨语言扩展依赖基于大语言模型的翻译；尽管在 TCQ 中保留术语的混合翻译规则反映了实际惯例，但也限制了对多样用户行为的覆盖，例如将术语翻译或音译为母语的行为。

尽管本研究是填补航空航天领域信息检索评估空白的第一步，但仍需进行多项扩展。首先，我们计划拓展基准的复杂度和范围。除了纯文本语料库，我们还希望将其扩展为多模态信息检索评估集，纳入真实技术文档中的图形、表格和图表。同时，我们将通过添加需要参考多个文档的多跳查询，以及不可回答查询类型，提升基准的真实性，而不再局限于单段合成查询。其次，我们希望提高语料库和语言的多样性。我们计划纳入来自 NTRS 以外机构的公开技术报告和专利文献，以减少数据偏差。此外，除了第3.6节中的 TCQ术语保留规则外，我们还将通过纳入真实用户行为来丰富跨语言评估场景，即用户会将术语完整地翻译或音译为其母语。

参考文献

Patrick Lewis、Ethan Perez、Aleksandra Piktus、Fabio Petroni、Vladimir Karpukhin、Naman Goyal、Heinrich Küttler、Mike Lewis、Wen-tau Yih、Tim Rocktäschel、Sebastian Riedel 和 Douwe Kiela。面向知识密集型自然语言处理任务的检索增强生成

发表于神经信息处理系统进展，第 33 卷，第 9459–9474 页，2020年。Wenqi Fan、Yujuan Ding、Liangbo Ning、Shijie Wang、Hengyun Li、Dawei Yin、Tat-Seng Chua 和 Qing Li。关于 RAG 与大语言模型结合的综述：迈向检索增强型大语言模型。发表于 第 *30* 届 *ACM SIGKDD*知识发现与数据挖掘会议论文集，2024年。doi:10.1145/3637528.3671470。Penghao Zhao、Hailin Zhang、Qinhan Yu、Zhengren Wang、Yunteng Geng、Fangcheng Fu、Ling Yang、Wentao Zhang、Jie Jiang 和 Bin Cui。面向人工智能生成内容的检索增强生成：综述。*arXiv* 预印本 *arXiv:2402.19473*，2024a。Akari Asai、Zeqiu Wu、Yizhong Wang、Avirup Sil 和 Hannaneh Hajishirzi。Self-RAG：通过自我反思学习检索、生成和批判。发表于国际学习表征会议，2024年。Cheng Niu、Yuanhao Wu、Juno Zhu、Siliang Xu、KaShun Shum、Randy Zhong、Juntong Song 和 Tong Zhang。RAGTruth：用于开发可信检索增强型语言模型的幻觉语料库。发表于计算语言学协会第 *62* 届年会论文集（第 *1* 卷：长篇论文），第 10862–10878 页。计算语言学协会，2024年。Orlando Ayala 和 Patrice Bechard。通过检索增强生成减少结构化输出中的幻觉。发表于 *2024*年计算语言学协会北美分会会议论文集：人类语言技术（第 *6* 卷：产业专题），第 228–238 页。计算语言学协会，2024年。Nandan Thakur、Nils Reimers、Andreas Rücklé、Abhishek Srivastava 和 Iryna Gurevych。BEIR：用于信息检索模型零样本评估的异构基准。发表于神经信息处理系统数据集与基准专题论文集，第 1 卷，2021年。Michael L. Nelson、Gretchen L. Gottlich、David J. Bianco、Sharon S. Paulson、Robert L. Binkley、Yvonne D. Kellogg、Chris J. Beaumont、Robert B. Schmunk、Michael J. Kurtz、Alberto Accomazzi 和 Omar Syed。美国国家航空航天局技术报告服务器。*互联网研究*，5(2):25–36，1995年。doi:10.1108/10662249510094768。Navid Rekabsaz、Oleg Lesota、Markus Schedl、Jon Brassey 和 Carsten Eickhoff。TripClick：大型医疗网络搜索引擎的日志文件。发表于第 *44* 届 *ACM SIGIR*信息检索研究与发展会议论文集，第 2507–2513 页，2021年。doi:10.1145/3404835.3463242。Cheng Gao、Chaojun Xiao、Zhenghao Liu、Huimin Chen、Zhiyuan Liu 和 Maosong Sun。通过扩大高质量合成查询-候选项对的规模来增强法律案例检索。发表于*2024*年自然语言处理实证方法会议论文集，第 7086–7100 页。计算语言学协会，2024年。Griffin Adams、Alexander R. Fabbri、Faisal Ladhak、Eric Lehman 和 Noémie Elhadad。从稀疏到稠密：使用密度链提示进行 GPT-4 摘要生成。发表于第 *4* 届摘要新前沿研讨会论文集，第 68–74 页。计算语言学协会，2023年。Aman Madaan、Niket Tandon、Prakhar Gupta、Skyler Hallinan、Luyu Gao、Sarah Wiegreffe、Uri Alon、Nouha Dziri、Shrimai Prabhumoye、Yiming Yang、Shashank Gupta、Bodhisattwa Prasad Majumder、Katherine Hermann、Sean Welleck、Amir Yazdanbakhsh 和 Peter Clark。自我改进：通过自我反馈进行迭代精炼。发表于第 *37* 届神经信息处理系统国际会议论文集，第 61 页，2023年。Noah Shinn、Federico Cassano、Ashwin Gopinath、Karthik Narasimhan 和 Shunyu Yao。Reflexion：采用语言强化学习的语言智能体。发表于第 *37* 届神经信息处理系统国际会议论文集，第 19 页，2023年。Yingming Wang 和 Pepa Atanasova。忠实自然语言解释的自我批判与精炼。发表于*2025*年自然语言处理实证方法会议论文集，第 8492–8518 页，2025年。Niklas Muennighoff、Nouamane Tazi、Loic Magne 和 Nils Reimers。MTEB：大规模文本嵌入基准。发表于计算语言学协会欧洲分会第 *17* 届会议论文集，第 2014–2037 页。计算语言学协会，2023年。Fabio Petroni、Aleksandra Piktus、Angela Fan、Patrick Lewis、Majid Yazdani、Nicola De Cao、James Thorne、Yacine Jernite、Vladimir Karpukhin、Jean Maillard、Vassilis Plachouras、Tim Rocktäschel 和 Sebastian Riedel。KILT：知识密集型语言任务基准。发表于*2021*年计算语言学协会北美分会会议论文集：人类语言技术，第 2523–2544 页，2021年。Jonathan Emmons、Taneesha Sharma、Bryan Matthews 和 Mariam Salloum。航空安全领域的文本摘要：一个

p比较研究 y 大型g语言guage模型。在 *AIAA* 航空论坛 *2024*，2024年。doi:10.2514/6.2024-4569。

Timilehin P. Oderinde、Chetan Chandra、Leslie Albertoli、Jirat Bhanpato、Mayank V. Bendarkar 和 Dimitri Mavris。用于从事故报告中提取知识的航空安全问答数据集。载于 *AIAA 航空论坛与 ASCEND 2025*，2025年。doi:10.2514/6.2025-3248。Xinyu Zhang、Nandan Thakur、Odunayo Ogundepo、Ehsan Kamalloo、David Alfonso-Hermelo、Xiaoguang Li、Qun Liu、Mehdi Rezagholizadeh 和 Jimmy Lin。MIRACL：涵盖 18 种不同语言的多语言检索数据集。计算语言学协会会刊，11:1114–1131，2023年。Luiz Bonifacio、Israel Campiotti、Roberto Lotufo 和 Rodrigo Nogueira。mMARCO：MS MARCO 段落排序数据集的多语言版本。*arXiv* 预印本 *arXiv:2108.13897*，2021年。Xinyu Zhang、Xueguang Ma、Peng Shi 和 Jimmy Lin。Mr. TyDi：用于稠密检索的多语言基准。载于 第一届多语言表征学习研讨会论文集，第 127–137 页。计算语言学协会，2021年。Ellen Voorhees、Tasmeer Alam、Steven Bedrick、Dina Demner-Fushman、William R. Hersh、Kyle Lo、Kirk Roberts、Ian Soboroff 和 Lucy Lu Wang。TREC-COVID：构建疫情信息检索测试集。*ACM SIGIR* 论坛，54(1):1–12，2021年。Yu-An Liu、Ruqing Zhang、Jiafeng Guo、Wei Chen 和 Xueqi Cheng。论生成式检索模型的鲁棒性：分布外视角。载于 *SIGIR 2023 Gen-IR* 研讨会论文集，2023a。Luiz Bonifacio、Vitor Jeronymo、Hugo Abonizio、Israel Campiotti、Roberto Lotufo 和 Rodrigo Nogueira。InPars：利用大语言模型进行信息检索的数据增强。*arXiv* 预印本 *arXiv:2202.05144*，2022年。Zhuyun Dai、Vincent Y. Zhao、Ji Ma、Yi Luan、Jianmo Ni、Jing Lu、Anton Bakalov、Kelvin Guu、Keith B. Hall 和 Ming-Wei Chang。Promptagator：仅凭 8 个示例实现少样本稠密检索。载于 国际学习表征会议，2023年。Aditi Chaudhary、Karthik Raman 和 Michael Bendersky。一切皆相对！——一种通过合成查询生成提升零样本相关性预测的方法。载于 计算语言学协会论文集：*NAACL 2024*，第 1645–1664 页。计算语言学协会，2024年。Daniel Fernando Campos、Tri Nguyen、Mir Rosenberg、Xia Song、Jianfeng Gao、Saurabh Tiwary、Rangan Majumder、Li Deng 和 Bhaskar Mitra。MS MARCO：一个由人工生成的机器阅读理解数据集。*arXiv* 预印本 *arXiv:1611.09268*，2016年。Zhiyuan Zhao、Hengrui Kang、Bin Wang 和 Conghui He。DocLayout-YOLO：通过多样化合成数据和全局到局部的自适应感知增强文档版面分析。*arXiv* 预印本 *arXiv:2410.12628*，2024b。Harrison Chase。Langchain：通过可组合性构建大语言模型应用，2022年。网址：https://github. com/langchain-ai/langchain。开源软件。Mahmoud Amiri 和 Thomas Bocklitz。切分两次，嵌入一次：化学感知检索增强生成中分段与表征权衡的系统研究。*arXiv* 预印本 *arXiv:2506.17277*，2025年。Matthew Honnibal 和 Ines Montani。spaCy 2：结合 Bloom 嵌入、卷积神经网络和增量解析的自然语言理解。即将发表，2017年。网址：https://github.com/explosion/spaCy。Robyn Speer。rspeer/wordfreq：v3.0，2022年9月。To m B. Brown、Benjamin Mann、Nick Ryder、Melanie Subbiah、Jared Kaplan、Prafulla Dhariwal、Arvind Neelakantan、Pranav Shyam、Girish Sastry、Amanda Askell、Sandhini Agarwal、Ariel Herbert-Voss、Gretchen Krueger、Tom Henighan、Rewon Child、Aditya Ramesh、Daniel M. Ziegler、Jeffrey Wu、Clemens Winter、Christopher Hesse、Mark Chen、Eric Sigler、Mateusz Litwin、Scott Gray、Benjamin Chess、Jack Clark、Christopher Berner、Sam McCandlish、Alec Radford、Ilya Sutskever 和 Dario Amodei。语言模型是少样本学习者。载于 神经信息处理系统进展，第 33 卷，第 1877–1901 页，2020年。Henrique Schechter Vera、Sahil Dua、Biao Zhang、Daniel Salz、Ryan Mullins、Sindhu Raghuram Panyam、Sara Smoot、Iftekhar Naim、Joe Zou、Feiyang Chen、Daniel Cer、Alice Lisak、Min Choi、Lucas Gonzalez、Omar Sanseviero、Glenn Cameron、Ian Ballantyne、Kat Black、Kaifeng Chen、Weiyi Wang、Zhe Li、Gus Martins、Jinhyuk Lee、Mark Sherwood、Juyeong Ji、Renjie Wu、Jingxiao Zheng、Jyotinder Singh、Abheesht Sharma、Divyashree Sreepathihalli、Aashi Jain、Adham Elarabawy、AJ Co、Andreas Doumanoglou、Babak Samari、Ben Hora、Brian Potetz、Dahun Kim、Enrique Alfonseca、Fedor Moiseev、Feng Han、Frank Palma Gomez、Gustavo Hernández Ábrego、Hesen Zhang、Hui Hui、Jay Han、Karan Gill、Ke Chen、Koert Chen、Madhuri Shanbhogue、Michael Boratko、Paul Suganthan、Sai Meher Karthik Duddu、Sandeep Mariserla、Setareh Ariafar、Shanfeng Zhang、Shijie Zhang、Simon Baumgartner、Sonam Goenka、Steve Qiu、Tanmaya Dabral、Trevor Walker、Vikram Rao、Waleed Khawaja、Wenlei Zhou、Xiaoqi Ren、Ye Xia、Yichang Chen、Yi-Ting Chen、Zhe Dong、Zhongli Ding、Francesco Visin、Gaël Liu、Jiageng Zhang、Kathleen Kenealy、Michelle Casbon、Ravin Kumar、Thomas Mesnard、Zach Gleicher、Cormac Brick、Olivier Lacombe、Adam

Roberts、Qin Yin、Yunhsuan Sung、Raphael Hoffmann、Tris Warkentin、Armand Joulin、Tom Duerig 和 Mojtaba Seyedhosseini。EmbeddingGemma：强大而轻量的文本表示。*arXiv* 预印本 *arXiv:2509.20354*，2025年。Yang Liu、Dan Iter、Yichong Xu、Shuohang Wang、Ruochen Xu 和 Chenguang Zhu。G-Eval：使用 GPT-4 进行 NLG 评估，实现更好的人类对齐。载于*2023*年自然语言处理实证方法会议论文集，2023b。DeepSeek-AI。DeepSeek-V3.2-Exp：借助 DeepSeek 稀疏注意力提升长上下文效率。https: //github.com/deepseek-ai/DeepSeek-V3.2-Exp，2025a。访问日期：2025年10月15日。An Yang、Anfeng Li、Baosong Yang、Beichen Zhang、Binyuan Hui、Bo Zheng、Bowen Yu、Chang Gao、Chengen Huang、Chenxu Lv、Chujie Zheng、Dayiheng Liu、Fan Zhou、Fei Huang、Feng Hu、Hao Ge、Haoran Wei、Huan Lin、Jialong Tang、Jian Yang、Jianhong Tu、Jianwei Zhang、Jianxin Yang、Jiaxi Yang、Jing Zhou、Jingren Zhou、Junyang Lin、Kai Dang、Keqin Bao、Kexin Yang、Le Yu、Lianghao Deng、Mei Li、Mingfeng Xue、Mingze Li、Pei Zhang、Peng Wang、Qin Zhu、Rui Men、Ruize Gao、Shixuan Liu、Shuang Luo、Tianhao Li、Tianyi Tang、Wenbiao Yin、Xingzhang Ren、Xinyu Wang、Xinyu Zhang、Xuancheng Ren、Yang Fan、Yang Su、Yichang Zhang、Yinger Zhang、Yu Wan、Yuqiong Liu、Zekun Wang、Zeyu Cui、Zhenru Zhang、Zhipeng Zhou 和 Zihan Qiu。Qwen3 技术报告。*arXiv* 预印本 *arXiv:2505.09388*，2025年。Meta AI。介绍 LLaMA 4：推进多模态智能。https://ai.meta.com/blog/ llama-4-multimodal-intelligence/，2025年。访问日期：2025年11月28日。Meta AI。Llama 3 模型家族。*arXiv* 预印本 *arXiv:2407.21783*，2024年。Llama 3.1 系列的权威技术报告，其中包括 8B Instruct 模型。Albert Q. Jiang 、 Alexandre Sablayrolles 、 Arthur Mensch 、 Chris Bamford 、 Devendra Singh Chaplot、Diego de las Casas、Florian Bressand、Gianna Lengyel、Guillaume Lample、Lucile Saulnier、Lélio Renard Lavaud、Marie-Anne Lachaux、Pierre Stock、Teven Le Scao、Thibaut Lavril、Thomas Wang、Timothée Lacroix 和 William El Sayed。Mistral 7B。*arXiv* 预印本 *arXiv:2310.06825*，2023年。DeepSeek-AI。DeepSeek-R1：通过强化学习激励大语言模型的推理能力，2025b。Stephen E. Robertson、Steve Walker、Susan Jones、Micheline Hancock-Beaulieu 和 Mike Gatford。TREC-3 上的 Okapi。载于第三届文本检索会议（*TREC-3*）概述，第 109–126 页。NIST 特别出版物 500-225，1995年。Puxuan Yu、Luke Merrick、Gaurav Nuti 和 Daniel Campos。Arctic-Embed 2.0：毫不妥协的多语言检索。*arXiv* 预印本 *arXiv:2412.04506*，2024年。Jianlyu Chen、Shitao Xiao、Peitian Zhang、Kun Luo、Defu Lian 和 Zheng Liu。M3-Embedding：通过自知识蒸馏实现多语言性、多功能性和多粒度文本嵌入。载于计算语言学协会论文集：*ACL 2024* 论文集，第 2318–2335 页。计算语言学协会，2024年。Liang Wang、Nan Yang、Xiaolong Huang、Linjun Yang、Rangan Majumder 和 Furu Wei。Multilingual E5 文本嵌入：技术报告。*arXiv* 预印本 *arXiv:2402.05672*，2024年。Xin Zhang、Yanzhao Zhang、Dingkun Long、Wen Xie、Ziqi Dai、Jialong Tang、Huan Lin、Baosong Yang、Pengjun Xie、Fei Huang、Meishan Zhang、Wenjie Li 和 Min Zhang。mGTE：用于多语言文本检索的通用长上下文文本表示与重排序模型。载于*2024*年自然语言处理实证方法会议论文集：产业赛道，第 1393–1412 页。计算语言学协会，2024年。Yauhen Babakhin、Radek Osmulski、Ronay Ak、Gabriel Moreira、Mengyao Xu、Benedikt Schifferer、Bo Liu 和 Even Oldridge。Llama-Embed-Nemotron-8B：适用于多语言和跨语言任务的通用文本嵌入模型。*arXiv* 预印本 *arXiv:2511.07025*，2025年。Yanzhao Zhang、Mingxin Li、Dingkun Long、Xin Zhang、Huan Lin、Baosong Yang、Pengjun Xie、An Yang、Dayiheng Liu、Junyang Lin、Fei Huang 和 Jingren Zhou。Qwen3 嵌入：通过基础模型推进文本嵌入与重排序。*arXiv* 预印本 *arXiv:2506.05176*，2025年。Rui Meng、Ye Liu、Shafiq Rayhan Joty、Caiming Xiong、Yingbo Zhou 和 Semih Yavuz。SFR-Embedding-Mistral：通过迁移学习增强文本检索。Salesforce AI Research 博客，2024年。网址：https://www.salesforce. com/blog/sfr-embedding/。Alexis Conneau、Kartikay Khandelwal、Naman Goyal、Vishrav Chaudhary、Guillaume Wenzek、Francisco Guzmán、Edouard Grave、Myle Ott、Luke Zettlemoyer 和 Veselin Stoyanov。大规模无监督跨语言表示学习。载于计算语言学协会第*58*届年会论文集，第 8440–8451 页。计算语言学协会，2020年。

A 提示工程详情

本节详细介绍构建 STELLA基准时所用的提示模板。所有提示均针对 GPT-5 设计，并在其上执行。在下方模板中，尖括号内的占位符（例如，<variable>）表示运行时插入动态数据的位置。

A.1 查询意图分类提示词

如第 3.4.2 节所述，我们使用 GPT-5 将候选段落分为五种不同的查询意图。表格 8 展示了确切的提示词从我们的流程中衍生出的提示模板。

查询意图分类提示模板

 # 角色
你是一位航空航天领域专家。请将该段落分类为最适合用于检索的单一查询意图。

以下是航空航天领域的 5 种查询意图：
- 定义／原理：区分概念，解释运行机制、近似或变量依赖关系。
- 数值 / 规格：检索数值/规格详情：数值、范围、假设、单位、不确定性（避免单一数值查询）。
- 程序 / 操作：步骤、初始化/校准、调度、操作规则。
- 比较 / 权衡：比较不同选项/配置的性能、质量、功率、裕度。
- 异常 / 风险：异常/故障的原因、复现条件、缓解措施/复发预防。

以下段落最适合用于生成哪一种意图的查询?
你只能回答 5 个意图名称之一。（例如：定义／原理）

段落：
<段落_文本>

最合适的意图：

表 8：用于查询意图分类的提示模板。运行时， <段落_文本> 占位符会替换为候选段落的实际文本。

A.2 合成查询生成提示词

本节详细介绍合成查询生成三个阶段所使用的提示模板：（1）术语描述生成，（2）术语一致查询生成（TCQG），以及（3）术语无关查询生成（TAQG）。

A.2.1 术语描述生成

表 9 展示了用于为技术术语生成上下文感知描述的提示词。这些描述对 TAQG 流程至关重要，因为它们可作为实际术语的语义替代项。

术语描述生成提示模板

 # 角色
你是一位航空航天领域专家。以下内容摘自一份特定文档。请根据上下文，
为技术术语“<term>”生成简短清晰的描述。如果无法根据上下文确定其含义，
请回答“难以在上下文中定义”。

上下文：
<上下文 -文本>

术语：
<术语>

简短描述：

表 9：用于生成上下文感知术语描述的提示模板。

A.2.2 术语一致的查询生成（TCQG）

表 10 展示了用于生成 TCQ 的提示词。该提示词实现了密度链（CoD）过程和自我反思，同时明确要求纳入段落中出现的术语。

TCQG 提示模板

 # 角色
你需要为航空航天信息检索基准生成合成查询。请应用三步密度链（CoD）过程，逐步增密段落中的实体。在每一步中，列出你识别或新增的实体，并提供一句英文自我反馈，以改进下一步。

 # 实体定义
“实体”必须是简短、具体的技术术语或名词短语（1-3 个词）。避免使用冗长的描述性短语或完整分句 es.

 # 输入（你将收到且仅收到一个 JSON 对象）<i
nput_json>
其中：
- 段落_文本：原始文本（单个段落）
- 已识别的_术语： [{ "term": "<技术术语（英文）>", "description": "<简短说明>" }...]
- 抽取的_意图：只能是以下其中一项：
- 定义／原理——区分概念，解释机制／近似／变量依赖关系 # - 数值／规格——数值、范围、假设、单位、不确定性（避免只查单个数值） # - 程序／操作——步骤、初始化／校准、计划安排、操作规则 # - 比较／权衡——对不同选项／配置的性能／质量／功率／裕度进行定量比较 # - 异常／风险——原因、复现条件、缓解措施／复发预防

 # CoD 步骤 -

步骤 1（种子）：有意省略一些核心实体，并写出一个 -
仅凭段落即可作答的句子式查询，同时严格避免使用已识别的_terms.term中的所有术语。-
“描述”字段仅供你理解，不要照抄。 . term 的术语来扩展步骤 1（保留英文拼写）
步骤 2（增密 通过添加一个来自 identified_terms
- 步骤描述“描述”字段仅供你理解，不要照抄。已识别_词语的术语来完善步骤 2 . 术语（保留英文拼写） .

 # 每个步骤都必须包含 -

查询：一个英文句子（15–25 个词元），采用中性／技术性语气 - 已识别_实体：简短文本实体（术语／名词短语）数组，反映在此步骤的查询中（最多 2 个实体） - 新增实体_：与上一步相比新增的实体数组（步骤 1 可为空） - 自我 _ 反馈：用一行简洁的英文描述下一步骤可采取的改进措施

 # 自我反馈指南（每个步骤）W
ri 用最具体的 3 词）。始终包括 1) 对“sampled_intention”的意图遵循；2) 约束检查：仅限篇章、避免禁用形式、15–25 个词元、不使用外部知识；3) 后续操作：指出要添加或完善的具体实体/条件（例如机制因素、操作条件、范围/假设/单位、比较指标）。
写一条简洁的，

 # 硬性约束（所有步骤）
1) 段落仅可回答（安全推断）。2）禁用形式：不得提出列表/引用请求；不得进行单一数值查询；不得使用是/否提示词；不得使用无所指的指示词（“this/that/these/those”）。3）风格/长度：使用英文，恰好一句；保持中性、技术性；15–25 个词元；仅使用段落中的术语或安全同义词。4）意图保留：不得更改“sampled_intention”。5）无外部知识：不得使用外部事实/来源/工具名称；不得编造符号/变量。6）技术术语使用：步骤 2 和步骤 3 必须逐字包含一个已识别的_terms.term（英文）。不得复制任何描述。7）实体粒度：“recognized_entities”和“entities_added”中的所有项目都必须遵循# 实体定义（即简短、具体的术语或名词短语，1–3 个词）。不得将长从句或完整句子列为实体。

8）“identified_terms”保留：在步骤 1 的“query”、“recognized_entities”或“entities_added”字段中，不得使用“identified_terms”列表中的任何术语。这些术语仅保留用于在步骤 2 和步骤 3 中引入。

 # 输出格式（仅返回一个 JSON 对象；不得包含额外文本/注释/Markdown）
{ "intention": "<复制采样的_意图>", "步骤_1": { "查询": "<一个英文句子>", "识别的_实体": ["<实体>", "..."],"已添加的_实体": [],"自我_反馈": "<一行英文>" }, "步骤_2": { "查询": "<一个包含一个已识别术语原文的英文句子>", "识别的_实体": ["<实体>", "..."], "已添加的_实体": ["<一个来自已识别_术语的实体。术语>"],"自我_反馈": "<一行英文>" }, "步骤_3": { "查询": "<一个包含一个已识别术语原文的英文句子>", "识别的_实体": ["<实体>", "..."], "已添加的_实体": ["<一个来自已识别_术语的实体。术语>"],"自我_反馈": "<一行英文>" } }

表 10：使用 CoD 和自我反思的 TCQG 指令提示词。

A.2.3 术语无关查询生成（TAQG）

表11展示了 TAQG 的提示词。该提示词的特点是采用**绝对术语禁用政策**，严格禁止使用特定术语，要求模型改为依靠间接描述。

A.3 跨语言翻译提示

为构建第3.6节所述的跨语言评估集，我们采用了一种根据查询类型进行调整的动态提示策略。翻译提示会在运行时组装，具体取决于目标查询是 TCQ 还是 TAQ。

对于 TCQ 翻译，提示词会注入一个**关键规则** 区块，明确列出从输入查询中提取的技术术语（使用第 3.3 节中的词典），并指示模型保留这些英文术语。对于 TAQ 翻译，则省略此规则，以便进行完整、自然的翻译。此外，我们还提供了特定语言的少样本示例，以确保模型遵循所需的输出格式和风格。表12详细说明了所用的提示模板和条件逻辑。

TAQG 提示模板

 # 角色
（注：与表 *10* 中所示的 *TCQG* 提示词相同。）

 # 实体定义A
n "entity" must b一个简短、具体的技术术语或名词短语（1–3 个词）。避免使用冗长、描述性的短语或完整分句。

 # 绝对术语禁用政策
任何步骤的查询中都绝对不能包含与任何已识别的_terms.term匹配的词元，无论大小写、是否带连字符、单复数形式或拼写变体如何。相反，应根据相应的“描述”进行简洁改述，间接表达该概念。不要逐字照抄“描述”；请改述，并避免使用引号。

 # 输入（你将恰好收到一个 JSON 对象）*(N*
ote: identical to the TCQG 提示词 *s hown in Table 10.)*

 # CoD 步骤 -
步骤 1（参见 -）：有意省略部分核心实体，并写出一个仅凭段落即可回答的单句查询，同时严格避开所有已识别的_terms.term。-
步骤 2（增密 -1）：通过转述其“描述”（不得逐字照抄，不得加引号），间接编码“identified_terms”中的一个概念，以扩展步骤 1 .
继续避免使用 identified_terms.term 中的所有术语。
- 步骤 3（增密 -2）：通过转述其“描述”（不得逐字照抄，不得加引号），间接编码“identified_terms”中的一个概念，以完善步骤 2 .
继续避免使用 identified_terms.term 中的所有术语。

 # 每个步骤必须包含 -
查询：一条英文句子（15–25 个词元），采用中性／技术性语气，其中不含任何与“identified_term”相同的词元 - recognized_实体：由简短的段落实体（术语／名词短语）组成的数组，这些实体应体现在本步骤的查询中（最多 2 个实体） - entities_added：与上一步相比新增的实体数组（步骤 1 可以为空） - self_feedback：一行简洁的英文内容，描述下一步骤的可执行改进 -（步骤 2、步骤 3）“descriptions_referenced”：列出你借助了哪些描述（简短的释义标签或简要标识符）的数组，以间接编码该概念（不得逐字复制）

 # 自我反馈指南（每个步骤）W
ri使用冒号串联 i2–3 项 iv。始终包含：1）对“sampled_intention”的意图遵循；2）约束检查：仅限篇章、避免禁用形式、15–25 个词元、零禁用术语重叠、不使用外部知识；3）后续行动：使用描述中的间接措辞，指明要添加／改进的具体实体／条件（例如，机制因素、操作条件、范围／假设／单位、比较指标）。

 # 硬性约束（所有步骤）1)
段落 -答案 ble only (saf推理）。
2) Forbidden f形式：无 list/quot请求 t s; 没有 s ingle-number lookup；不，是 /没有提示t s; 没有 bare deictics (“这 s/that/these/those”).
3) 风格/长度：恰好用英文写一个句子；中立、技术性；15–25 个词元；仅使用段落中的术语或安全的同义词；绝不使用任何已识别的术语。4) 意图保留：绝不更改 "sampled_intention"。
5) 无外部知识：不得使用外部事实、来源或工具名称；不得臆造符号或变量。
6) 间接编码要求：在步骤 2 和步骤 3 中，改写一条描述以传达该概念；将其记录在 "descriptions_referenced" 中（不得引用或复制原文）。

7) 实体粒度："recognized_entities" 和 "entities_added" 中的所有项目都必须遵循 # 实体定义（即简短、具体的术语或名词短语，1—3 个词）。绝不得将长篇从句或完整句子列为实体。8) "id
entified_terms " R eservation：在 Ste 的 "query"、"recognized_entities" 或 "entities_added" 字段中，不得使用 "identified_terms" 列表中的任何术语 p
1. 这些术语是 reser 仅供在步骤 2 和步骤 3 中引入。

 # 输出格式（仅返回一个 JSON 对象；不添加任何额外文本、注释或 Markdown）{

"意图": "<复制采样的_意图>", "步骤_1": { "查询": "<一个英文句子（不含已识别的_术语）>", "识别的_实体": ["<实体>", "..."],"新增的实体_": [], "自我_反馈": "<一行英文>" }, "步骤_2": { "查询": "<一个英文句子（不含已识别的_术语），其中至少有一个概念通过释义描述进行编码>", "识别的_实体": ["<实体>", "..."], "新增的实体_": ["<一个来自已识别_术语的实体.术语>"], "引用的描述_": ["<"术语"，即其概念被间接编码的确切已识别_术语.术语>", "<所采用描述的简短释义标签>"],"自我_反馈": "<一行英文>" }, "步骤_3": { "查询": "<一个英文句子（不含已识别的_术语），其中至少有一个概念通过释义描述进行编码>", "识别的_实体": ["<实体>", "..."], "新增的实体_": ["<一个来自已识别_术语的实体.术语>"], "引用的描述_": ["<"术语"，即其概念被间接编码的确切已识别_术语.术语>", "<所采用描述的简短释义标签>"],"自我_反馈": "<一行英文>" } }

表11：TAQG 的指令提示词。请注意新增的术语禁用政策和描述引用字段。

翻译提示模板

 # 角色
您是一位专门处理航空航天技术查询的专家译者。您的任务是将给定的英文文本翻译成 <目标_语言_名称>。请严格遵循以下规则和示例。

 # 示例
<少量_示例_>
（注：此处插入展示输入/输出对的特定语言示例。例如，韩语：输入："定义计算流体力学……"，输出："计算流体力学……"）

 # 规则。
1 将文本自然地翻译成 <target_language_name>。2. 3 <保留_术语_指令> 保留术语的指令。仅提供最终译文，不要附加解释或对话内容。

英文文本：<input_query>

— 动态指令逻辑（<保留_术语_指令>）—

案例 *A*：*TCQ*（混合翻译 - 术语保留）如果在输入文本中检测到特定技术术语：

本次请求的关键规则：

你不得翻译本输入中出现的以下特定技术术语，请保留其原始英文形式："term1"、"term2"……]。

情况 *B*：*TAQ*（完整翻译）如果没有特定的术语限制：

本次请求没有需要保留的特定术语。

表12：跨语言扩展的提示模板。 <保留_术语_指令> 会根据查询属于 TCQ 还是 TAQ 动态填充，以实现混合翻译策略。