亚马逊云科技

亚马逊云科技

生成式 AI 时代的智能翻译创新与实践

前述特定亚马逊云科技生成式人工智能相关的服务目前在亚马逊云科技海外区域可用。亚马逊云科技中国区域相关云服务由西云数据和光环新网运营，具体信息以中国区域官网为准。

目录

CATALOGUE

01

翻译的技术趋势

一 翻译的技术趋势

翻译是大模型应用的热门场景，大模型在翻译场景相对传统机翻具备很强的优势，能够捕捉语言深层次含义，适配文化背景，保持原文的情感和风格，更自然流畅的语言 (没有机翻中常见的生硬问题)。成本方面，随着大模型推理成本的降低，相对传统机翻服务也非常具备竞争优势。速度方面，一般情况下传统机翻会有优势，但随着推理技术的发展，以及各种尺寸的大模型可供选择，这方面的劣势正在逐步被拉平。

此外传统机翻更多是一个单纯的黑盒 SAAS 服务，技术人员无法做太多调整。 大模型翻译可供技术人员调整的空间更大，通过 Prompt 调优，RAG 召回上下文和背景知识等诸多手段，可以进一步明显提升翻译质量，满足业务上的特殊要求。

大模型翻译已经逐步替代人工翻译，成为本地化团队的一种重要选择，但在实际的不同业务场景中，仍存在诸多问题。本文主要针对遇到的问题和这些场景中的技术实践进行总结。

02

不同业务场景下的特点

二 不同业务场景下的特点

在过去一年多的大模型翻译落地实践中，我们观察到了以下多个行业下多个不同的场景的业务特点。

行业	场景	要求		
		实效性	翻译质量	成本敏感性
电商	商品标题翻译	低	要求用词准确，比如服装款式，一字肩和漏肩要能准确区分	中
	广告词	低	很高的信雅达水准，不能直译，需要体现文化特点	低
游戏	游戏社区翻译	低	要求用词准确，游戏人物之类的专用词汇要准确	高
	游戏公告	低	属于官方发布，要求用词准确更高	低
	游戏剧情	低	很高的信雅达水准，需要风格 / 游戏的世界观保持一致	低
	游戏内即时聊天	高	不高，但要支持一些黑话	高
	APP UI 界面翻译	低	用词严谨准确，符合当地文化和语言规则	高
在线旅游代理商 (Online Travel Agents/ OTAs)	机票政策翻译	低	严格遵循国际航协的术语体系，以及多语言条件下，在涉及数字计算的公式需要保持算法统一	高
	酒店内容	低	用词严格准确，涉及到酒店名称 / POI 等需要精确翻译	高
网络社交	IM 对话翻译	中	较低要求，需要拼写 / 语法正确	低

总的来说，翻译的效果 / 成本 / 速度是我们需要关注的通用维度。在不同业务场景下，翻译侧重点又各有不同，例如电商场景中更加关注翻译的关键词，比如品牌、款式是否准确，能够准确传达关键信息，帮助高效的进行流量分发，对 GMV 的影响相对直接；游戏场景中的社区内容翻译，更加关注的是否能增加社区活跃度和内容丰富性，从提升社区生态和游戏话题度活跃度等。在线旅游代理商的业务场景中，无论是酒店名称、旅游景点、货币等都会涉及到多语言国家翻译的问题，往往会涉及文化适配，术语直译等问题，处理不当会引发 OTA 平台结算，法律合规等纠纷。这些业务考量是推动大模型在翻译场景的落地重要的衡量维度。

03

大模型翻译的问题分析

三 大模型翻译的问题分析

大型语言模型 (LLM) 在翻译领域虽然已经广泛应用，但由于其通用性设计而非专门针对翻译场景的优化，在实际应用中仍存在多方面的局限性。本章将系统分析大模型翻译面临的主要问题，涵盖游戏、电商、旅游等多个行业场景。

1. 专业领域术语精准度问题

1.1 游戏行业专业术语不精准

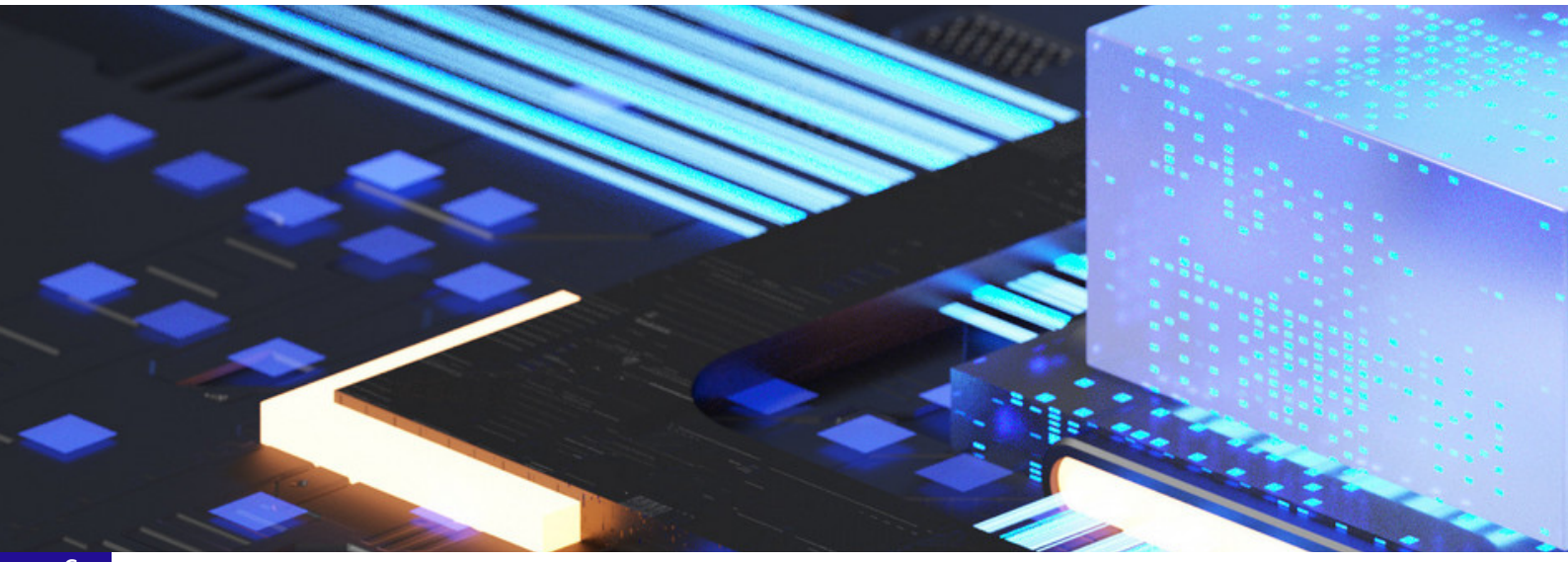
在游戏本地化过程中，专业术语的翻译精准度直接影响玩家的理解和体验。当前问题主要体现在：

技能名称、装备名称、角色名称等专业术语翻译不准确：

如某 MMORPG 游戏中，"Crowd Control" 被不同团队翻译为 " 群体控制 "、" 人群控制 " 和 " 控场 " 等多种表述，导致玩家混淆。

游戏机制相关术语使用不统一：

例如《魔兽世界》中的 "Taunt" 技能在早期被翻译为 " 嘲讽 "，而在某些界面又被翻译为 " 激怒 "，缺乏术语一致性。



1.2 电商领域专业词汇翻译失准

大模型在电商翻译中如果缺乏专业术语库和品牌认知，容易导致关键信息失真：

品牌名错误处理：

如将护肤品牌「SK-II」错误拆解为「SK-2」，破坏品牌资产完整性。

服装品类术语错位：

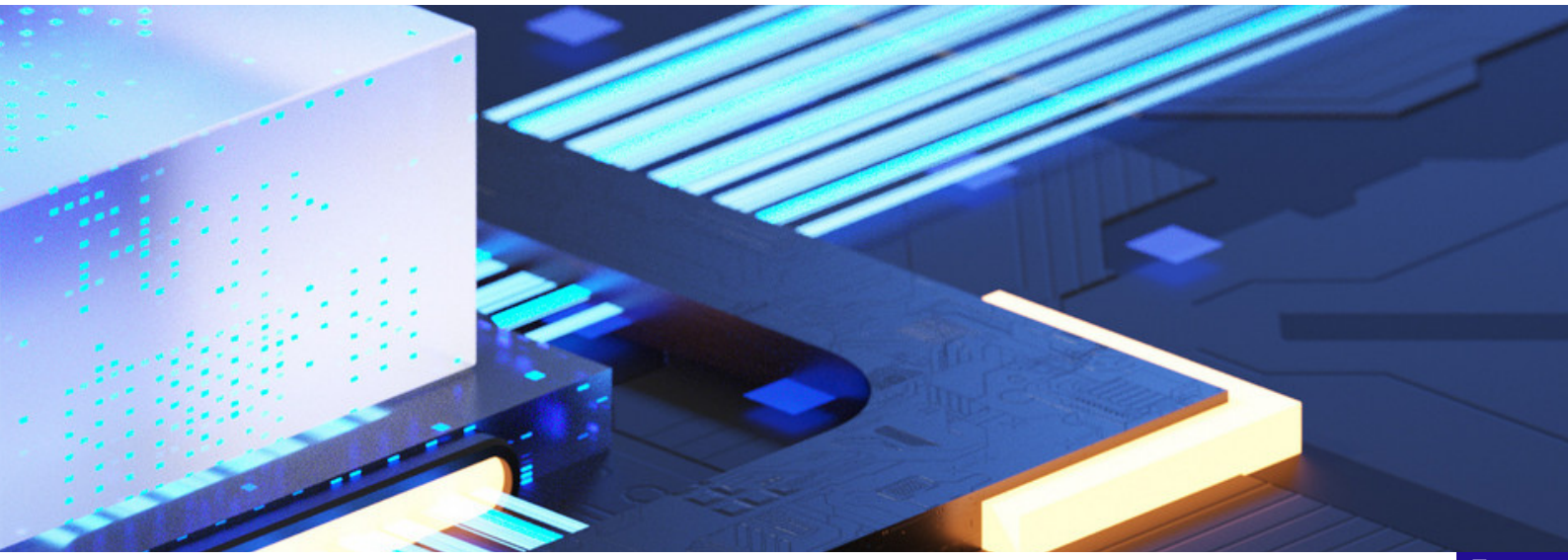
如将「女式黑色睡衣」直译为「Ladies' black pajamas」，而非更符合欧美消费认知的「Women's Black Sleepwear」，既模糊了产品形态（睡衣套装 vs 睡裙），又降低了搜索可见性。

1.3 旅游酒店业 (OTA) 术语问题

在旅游和酒店预订平台上，术语翻译问题尤为突出：

酒店品牌名称本地化不足：

酒店名称往往采用直接翻译而非本土化名称，如 Yitel Premium（和颐至尊）被翻译成 " 亿铂·尚品 "，导致用户识别困难。



2. 行业黑话与文化适配问题

2.1 游戏行业黑话翻译缺失专业性

游戏社区发展出独特的语言体系，这些行业黑话的准确翻译对维持游戏体验至关重要：

游戏社区常用术语翻译不当：

"Nerf"（削弱）、"Buff"（增强）、"Farming"（刷资源）等游戏黑话在翻译过程中容易被字面理解，失去原有的游戏语境含义。

官方公告翻译误导：

某竞技类游戏的官方公告中，"We're buffing the drop rate of legendary items" 被直译为 " 我们正在缓冲传奇物品的掉落率 "，而非正确的 " 我们正在提高传奇物品的掉落率 "，造成了玩家对游戏更新方向的误解。

2.2 文化适配不足

不同语境下的文化元素转换不当会导致翻译失真：

促销词汇文化差异：

如中文促销词「福袋」直译为日式概念「Lucky Bag」，而面向欧美市场时需本地化为「Mystery Gift Box with Discounts」，否则可能引发用户对商品内容的错误期待。

游戏世界观元素丢失：

以《巫师 3》为例，游戏中充满了斯拉夫神话元素，若翻译者不了解相关文化背景，"Leshen" 这类神话生物的名称和特性就无法准确传达，可能被简单翻译为 " 森林怪物 "，丧失了原作的文化深度和氛围营造。

3. 上下文理解与语境识别问题

3.1 同词多义问题

同一个中文词在不同场景下可能表达完全不同的含义，通过精准匹配引入术语映射，可能会误导大模型出现翻译错误：

美妆领域 vs 物流领域：

" 直发 " 在美妆产品 " 日本进口直发梳 负离子陶瓷电热梳 " 中指 " 使头发变直 "(Hair Straightening)；而在物流描述 " 海外仓直发 3-5 天送达 " 中则表示 " 直接发货 "(Direct Shipping)。

3.2 游戏场景识别不足

游戏翻译需要对游戏世界观和剧情的深入理解，仅靠语言转换是不够的：

无法理解游戏剧情上下文：

角色对话中包含的历史事件或前文铺垫在缺乏整体理解的情况下容易被错误翻译。

游戏对话语气和风格把握不准：

不同类型游戏 (如休闲游戏 vs 硬核游戏)、不同角色 (如反派 vs 主角) 的语言风格需要差异化处理。

4. 风格与一致性问题

4.1 翻译风格不一致

大模型翻译往往缺乏统一的风格标准：

表达习惯差异：

" 啊，这 ..." 等语气词，大模型可能直接翻译为 "Oh. Well"，而人工译员会根据语境选择更自然的表达如 "Ah, this"。

4.2 术语一致性缺失

翻译结果前后不一致是游戏及其他领域本地化中的常见问题：

同一术语在不同场景下翻译不一致：

在《最终幻想 XIV》中，同一职业技能 "Benediction" 在技能描述中被翻译为 " 天赐祝福 "，而在任务对话中又被称为 " 圣光祝福 "。

角色名称在不同章节翻译不同：

如《赛博朋克 2077》中的某些角色绰号，在主线任务和支线任务中出现不同翻译版本，导致玩家难以辨认是否为同一角色。

5. 格式与 UI 适配问题

5.1 UI 文本适配不当

翻译后文本长度控制不当会导致严重的用户界面问题：

UI 元素翻译后超出显示范围：

英文版 "Settings" 按钮被翻译为 " 系统设置与个人偏好 " 后，导致按钮文本严重溢出。

提示信息翻译后排版混乱：

《刺客信条》系列中的简短提示 "Press X to interact" 在某些语言版本中被翻译成较长的指令，导致提示框无法完整显示所有信息。

5.2 格式规范差异

不同语言和领域的格式规范差异未被正确识别和应用：

日期、货币格式不适配：

日期格式 (MM/DD/YYYY 与 DD/MM/YYYY)、货币单位 (美元、欧元、日元) 若未本地化，可能造成用户混淆甚至预订错误。如日元 (￥) 与人民币 (￥) 符号相似，若未明确标注 JPY 或 CNY，用户可能误判金额。

6. 大模型固有技术限制

大模型在处理翻译需求时，受限于 LLM 自身的一些技术特性，难以满足多种实际需求：

6.1 处理能力局限

长文本翻译受限：

由于输出长度限制 (通常为 8k Tokens)，无法直接完成长文本翻译，而传统机翻的支持长度通常更大。

文件翻译不完善：

虽然支持文件输入，但不能输出文件，文件格式保持也是一个问题。

特定片段保持原语言难度大：

实际业务中要求某些片段不翻译的需求，仅通过 Prompt 描述，其可靠性难以保证。

6.2 稳定性与一致性问题

大模型的推理过程基于概率采样，导致输出不稳定，加之训练语料庞杂且语种分布不平衡，在生产实践中经常出现：

拒绝翻译：

通常由大模型的无害化对齐策略导致，如将成人用品描述 "Clitoral Sucking Vibrators..." 翻译为泰语时仅返回 " ม่มีการแปลเนื้อหาที่มีนัยทางเพศ（不翻译涉及成人内容）"。

语种夹杂：

由训练语料中不同语言借词导致，如将英文 "Camping" 在泰语翻译中保留原词而非使用泰语词汇。

冗余重复：

特定语种中容易出现词语重复，如阿拉伯语翻译中将 "T-shirt" 重复为 "تي شيرت تي شيرت"。

符号 / 语法 / 拼写错误：

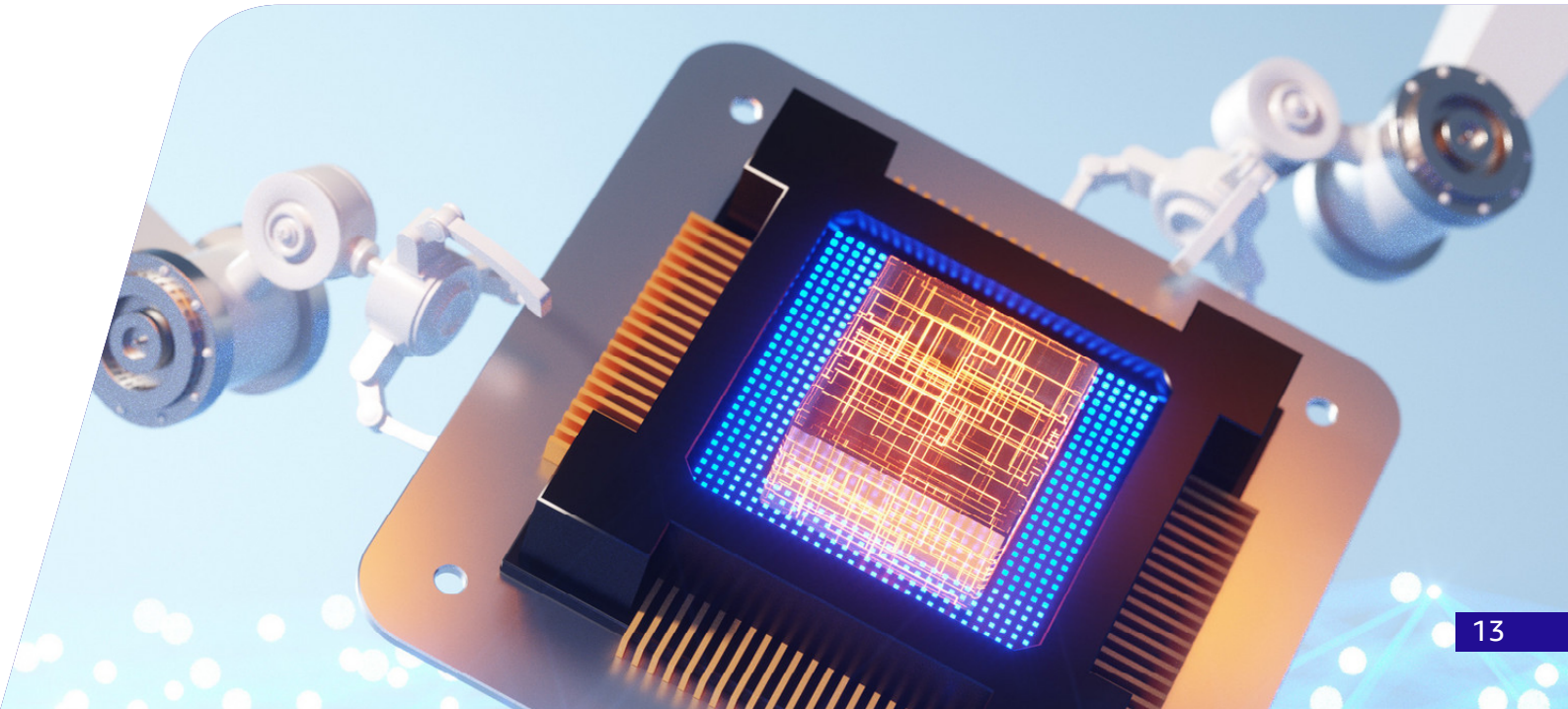
不同语种间符号使用差异，如 Apostrophe 使用错误 (Men`s 而非 Men's)。

格式变化：

指令对齐阶段过度强化 Markdown 格式输出，将普通文本转化为带编号和分段的格式。

source	translation	error_type
Clitoral Sucking Vibrators 2 IN 1 G Spot Nipple Sucker Tongue Licking Dual Stimulation Massager Mastrubation Sex Toys For Women,Sex Toys,Sex,Sexy	ไม่มีการแปลเนื้อหาที่มีนัยทางเพศ（不翻译涉及成人内容）	拒绝翻译
1pc Cartoon Print Flannel Blanket, Warm, Soft And Comfortable, Suitable For Living Room, Bedroom, Camping, Travel, Versatile Gift, Square Blanket	1ชิ้น ผ้าห่มฟลานเนลพิมพ์การ์ตูน อุ่น นุ่มสบาย เหมาะสำหรับห้องนั่งเล่น ห้องนอน การcamping การเดินทาง ของขวัญที่ใช้งานได้หลากหลาย ผ้าห่มสี่เหลี่ยมรเดินทาง ของขวัญที่ใช้งานได้หลากหลาย ผ้าห่มสี่เหลี่ยม	语种夹杂
Plus Size Heart Pattern Print Casual Elegant T-Shirt, Autumn	تي شيرت تي شيرت أنيق كاجوال بطباعة شكل القلب مقاس كبير، خريف	出现冗余重复
男士休闲简约 PU Leather 翻领拉链开襟夹克	Men`s Casual Minimalist PU Leather Turndown Collar Zip-up Jacket	拼写 / 语法 / 符号错误
儿童 轻质软底时尚袜子靴。 儿童 轻底一脚蹬加绒毛毛里雪地靴	1. Children's Light-weight Soft Bottom Fashion Sock Boots 2. Children's Light-weight One-touch Add Fleece Lining Snow Boots	格式变化

除了这些要求以外，业务人员对大模型翻译往往提出更高要求，如广告词创意翻译、信雅达表达、跨语言谐音梗等，这些要求超出了当前大模型的基本翻译能力范围，需要更复杂的定制优化。



04

LLM 翻译质量的优化思路

四 LLM 翻译质量的优化思路

1. 基于 Prompt/Agent 的优化思路

1.1 通过设定语言的语法规则、固定的样例来帮助 LLM 提高翻译质量

针对目标语言国家、翻译风格来设定语法规则，以及通过样例描述可以帮助 LLM 来更好理解翻译意图，以便于朝着用户的场景进行翻译。下面是以泰语为例，给出的一个来自翻译专家总结的翻译规则：

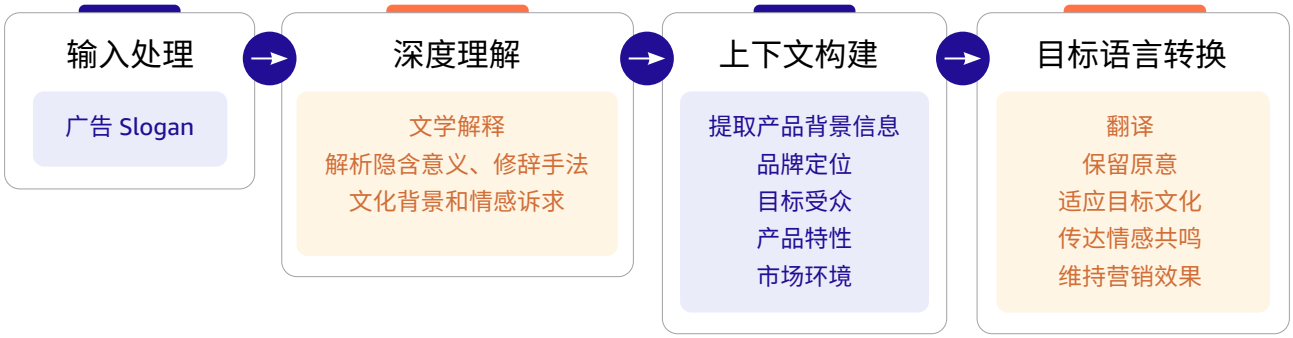
代码块

```
##  Thailand Language Rules:
1. Word Order
* Definition: Thai follows Subject-Verb-Object (SVO) structure.
* Examples:
    * ฉันกินข้าว (I eat rice)
    * เขาอ่านหนังสือ (He reads a book)
2. Classifiers
* Definition: Numeric classifiers are mandatory when counting nouns.
* Examples:
    * คนหนึ่งคน (one person) → คน (person) + หนึ่ง (one) + คน (classifier)
    * แมวสองตัว (two cats) → ตัว (animal classifier)
    * น้ำสามแก้ว (three glasses of water) → แก้ว (glass classifier)
3. Honorifics
* Definition: Social hierarchy markers in pronouns and verbs.
* Examples:
    * Casual: เธอไปไหน (Where are you going?)
    * Formal: ท่านจะไปที่ใดคะ (Where is your honor going?)
    * Honorific verb: ทาน (eat) instead of
4. Particles
* Definition: Sentence-final particles indicating mood/politeness.
* Examples:
    * Question: คุณชื่ออะไร คะ (What's your name?)
    * Emphasis: ร้อน จัง (It's so hot!)
    * Softening: ช่วยเปิดประตู หน่อย (Please open the door)
5. Tense Markers
* Definition: Adverbs indicate tense without verb conjugation.
* Examples:
    * Present: ฉัน กำลัง อ่านหนังสือ (I am reading)
    * Past: เขา แล้ว กิน (He already ate)
    * Future: พรุ่งนี้ฉัน จะ ไป (I will go tomorrow)
6. Ellipsis
* Definition: Subject omission in context-aware situations.
* Examples:
    * (ฉัน) ไปเที่ยวพรุ่งนี้ ((I) will travel tomorrow)
    * (เธอ) กินข้าวหรือยัง (Have (you) eaten?)
7. Cultural Adaptation
* Definition: Buddhist lexicon integration.
* Examples:
    * Greeting: สวัสดี (Hello)
    * Farewell: ไปล่ะ นะ (I'm leaving)
8. Reduplication
* Definition: Word repetition for emphasis or lexical creation.
* Examples:
    * เร็วๆ (very fast)
    * ดีดี (properly)
9. Writing System
* Definition: Script without spaces between words.
* Examples:
    * Correct: ฉันรักเธอ (I love you)
    * Incorrect: ฉัน รัก เธอ (spacing breaks meaning)
```

1.2 多步骤的翻译 Workflow

对于信雅达要求比较高的翻译场景，比如广告词 **Slogan** 翻译，文学作品翻译等，补充更多的背景信息有助于 LLM 更好的进行意译，从而达到更高的信雅达水准。

以下面**广告词 Solgan 翻**译为例：



代码块

```
## 输入
广告词： 遇见人间蜜桃肌
```

代码块

```
## 第一轮 Prompt - 理解 & 解释
请帮我用白话文解释产品 {prod_catalog} 的广告词 {source_text}
Please enclose the explanation in <explanation></explanation> XML tags
```

代码块

```
## 第二轮 Prompt - 提供背景 & 意译
The slogan to be translated will be give in the xml tags: <paragraph></paragraph>
The translate result should be wrapped in xml tags: <translation></translation>

Instructions:
- 参考explanation里面的解释，不要丢失 explanation 的内容
- 最大可能的保证信雅达
- 使用地道的目标语言
- 不要改变原文的语序
- 不要改变原文的目的
- 保留原文的修辞手法，例如夸张、比喻、拟人和双关

<example>
source_text: 易如反掌
destination_lang: English
translation: a piece of cake
</example>

The paragraph explanation is <explanation>
{explanation}
</explanation>

This is a slogan of {prod_catalog}.
请遵循上面的Instructions, 把下面的广告词改写成地道的{destination_lang}, 你可以参考<example></example>中的例子
<paragraph>{source_text}</paragraph>
```

2. 基于 Retrieval 增强翻译的优化思路

基于 Retrieval 增强的翻译，相对于固定 Prompt 的调优，可以引入更相关且更有价值的上下文。实现中可以分为两类，Reference Retrieval 和 Glossary Retrieval。 前者一般利用之前累计的人工翻译数据做 LLM 翻译的参考样例，LLM 可以模仿参考样例的翻译风格和用词。后者需要提前构建好跨语言的术语库，在翻译时引入术语映射供 LLM 参考。

由于样例和术语的量一般都比较大，不适合全部置入到 Prompt 中。对于样例召回，一般是通过搜索（语义或关键词）取 Topk；对于术语召回，一般通过精确词匹配，召回原文中出现过的术语以及对应多语言映射。从效果和实现技术方案上看，两者各有优劣。

2.1 Reference Retrieval 方法

这种方法是比较简单，可以复用 RAG 的经典方法，把样例直接注入向量数据库即可实现样例的搜索召回。在搜索召回时，一般可以采用 Full Text Search 和 Semantic Search。

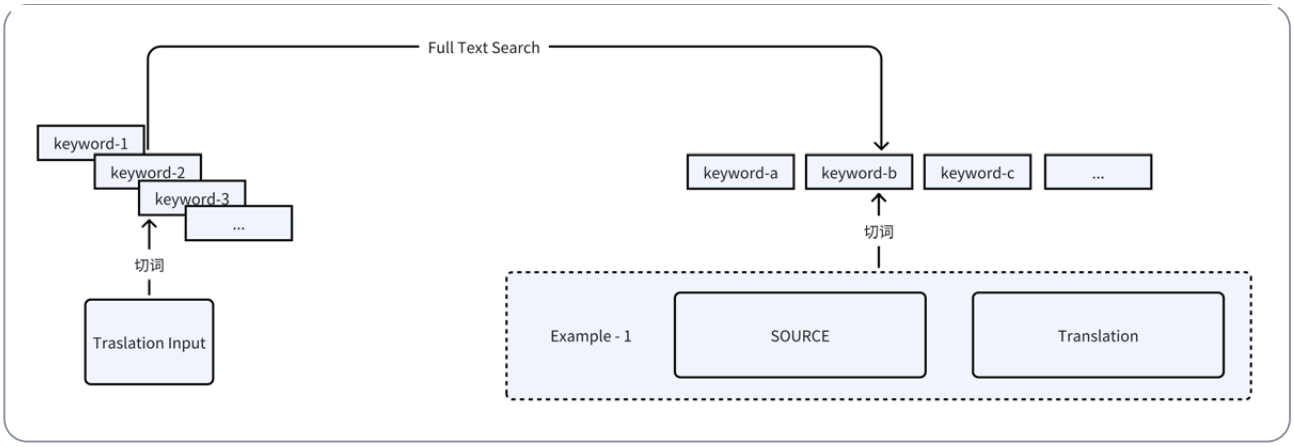


图1: Full Text Search

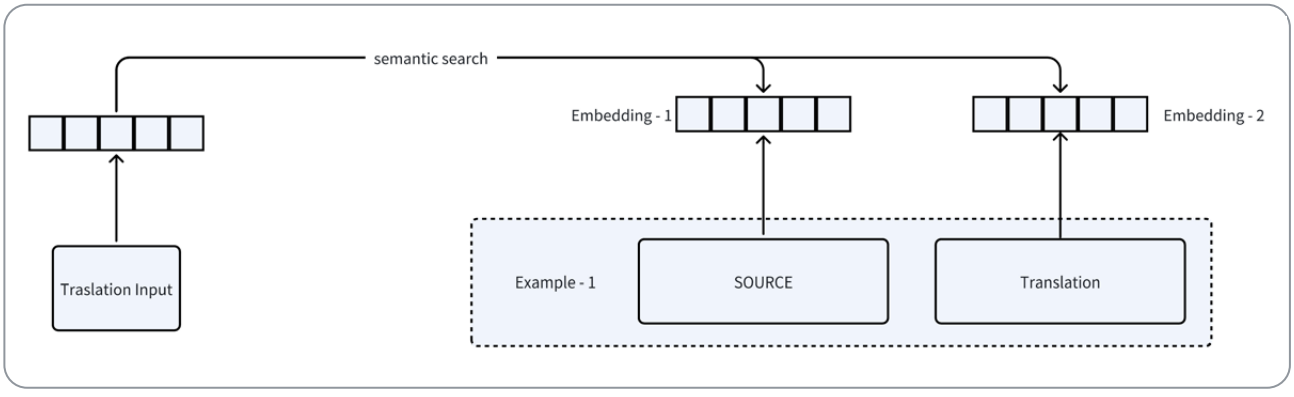


图2: Semantic Search

前者无需依赖向量模型，主要依靠关键词来进行召回，不会召回没有关键词命中的翻译样例，但可能会出现搜索结果为空的情况以至于没有提供翻译样例，大多数情况下基于 Semantic Search 样例召回效果要超过 Full Text Search(具体对比可以参见第五章实验结果对比)。我们建议在大多数情况下，采用基于 Semantic Search 的样例召回。

以下两个样例展现了添加 Reference 前后带来的翻译效果变化。通过 Reference Retrieval 增强后的翻译指标得分提升明显，翻译结果也更加符合商品标题的人工翻译质量要求，如其中的“原装”和“4 件装”等关键词。

方法对比	中文翻译结果	sacrebleu	meteor	nist
英文原文：GENUINE Whirlpool 3394083 Retainer Clip				
无任何 Retrieval 增强	**真正的**惠而浦 3394083 固定夹	32.47	0.57	0.93
Reference Retrieval 增强	惠而浦**原装** 3394083 固定夹	100.00	0.83	1.29
英文原文：4P Refrigerator Water Drip Tray Catcher,Water Drip Splash Guard Catcher Absorbent Mat Pads for Ge,Whirlpool,Samsung Refrigerator Water & Ice Dispenser,Kitchen Gadgets Accessories,White Grey,Big…				
无任何 Retrieval 增强	**4P** 冰箱漏水盘接水托盘 , 适用于 Ge、Whirlpool、Samsung 冰箱水和冰制冰机 , 厨房用品配件 , 白色灰色 , 大尺寸…	3.62	0.22	1.56
Reference Retrieval 增强	**4 件装**冰箱接水盘 , 接水防溅垫吸水垫 , 适用于 GE、惠而浦、三星冰箱饮水机和制冰机 , 厨房小工具配件 , 白色灰色 , 大尺寸 ...	31.88	0.54	2.80

通过翻译案例观察和具体的实践，对于这种方法的优劣势有如下总结：

优势方面：

上下文中给出翻译样例，不仅仅可以提供了之前翻译词汇的参考，也给出翻译风格的参考，能够让大模型了解当前翻译场景的业务 / 文化背景，实现翻译的信雅达。

劣势方面：

- **翻译样例的粒度问题和召回精确性问题**

  这两个问题本质上与传统 RAG 所面临的问题是一样的。如果已有的翻译样例是句子粒度的，那么可以按照当前粒度直接入库，如果之前的翻译样例太长，那么会召回的精确性问题就会增加，同时引入的额外 Token 成本也会大幅增加，假设待翻译的文本和翻译样例均是 300 Token 长度的段落，召回 4 个翻译样例，那么翻译成本的额外开销就是 1200 Token。所以段落级别的翻译样例，最好的办法是做句子级别的拆分，一般实践中，也可以采用大模型来做。

- **术语一致性问题**

  由于供候选的翻译样例比较多，样例之间的术语可能存在差别，而且通过翻译原文的语义进行召回，可能导致获取到的术语参考是不一致的。无法保证不同语言之间术语对应的一致性。

- **长期的维护问题**

  如果翻译服务上线以后，如果出现一些 Bad Case 不好解决，可能有以下原因：1. 缺乏某类翻译样例，人工添加某些翻译样例后，不一定能稳定精准召回。2. 召回了错误的翻译样例，误导了翻译大模型。

  在维护时，只能针对性的去更新翻译样例，这些更新对于整体翻译样例的质量是否为正向则很难评估。

2.2 Glossary Retrieval 方法

这种方法相对会复杂一些，它要求具备完善的专词术语数据，并且在翻译前能够精准匹配的召回这些专词映射。这就带来了一些技术的复杂性，首先很多翻译场景，专词术语数据并不是已经具备的，需要构建这些数据，人工构建成本高，需要更好的构建方法。其次，精准匹配召回，需要对多语言下的翻译原文按照专词表进行切词。最后，术语数据的存储维护更新，质量检查过滤是工程实践中需要考虑的。接下来阐述了这些环节的一些经验细节：

离线专词构建 / 评估

如果具备多语言翻译数据，可以通过大模型构建工作流来一次性自动化提取各个语言的术语映射，在工作流中共进行 2 次大模型调用，第一次调用先提取初筛的专词映射，第二次调用对术语映射的专业度 / 准确性 / 跨语言一致性进行评分，最后按照评分评估阈值进行筛选。通过实验中数据验证，利用大模型抽取专词是一种非常高效稳定的手段，对于专词可以达到很高的召回率，完全可以满足生产需要。

图3: 术语抽取流程图

如果不具备多语言的翻译数据，则可以根据单语种原文进行提取术语，再翻译到多个语种。这种方案的精准率不足，特别垂直的领域更低，需要配合人工进行审核校对，但也能够提升整个术语库构建的效率。

专词分词召回

为了召回专词的多语言映射，首先要从原文的句子或者段落中将专词切出来。需要有一个分词器来实现这个任务。分词器的核心目标不是为了实现完美分词，而是尽可能在文本中快速高效的找出所有出现的专词。分词器的多语言支持，时间复杂度，内存占用都会影响整个专词翻译的服务性能 / 成本。

为了实现这个目标，尝试了多种方法，得到如下总结：

分词方法	方法特点	实现方式
开源分词库	1. 多语言支持不好，不能支持所有语言 2. 对于不同语言需要采用不同的切词工具	直接导入包，比如 import jieba，NLTK
基于 KV 数据库的最大匹配	1. 无需安装依赖，内存占用小，无需索引加载时间 2. 长本文慢，需要调用侧拆分保证文本长度超长，切词时间复杂度随着文本长度指数级上升，秒级延迟	无需安装外部依赖
基于前缀树的最大匹配	1. 冷启时需要构建前缀索引 2. 切词时间复杂度和文本长度为线性关系，一般为毫秒级延迟	选择基于 C++ 内核的 marisa_trie 包，内存占用优秀，速度快

方案实现中采用第三种思路，采用前缀树的数据结构 (如下图所示) 来构建前缀索引，实际生产中，专词的量级可能达到几十万以上，对比了各种前缀树的包，最终采用了 Marisa_trie (基于 C++ 构建，性能优异、内存占用小) 来构建 Trie。

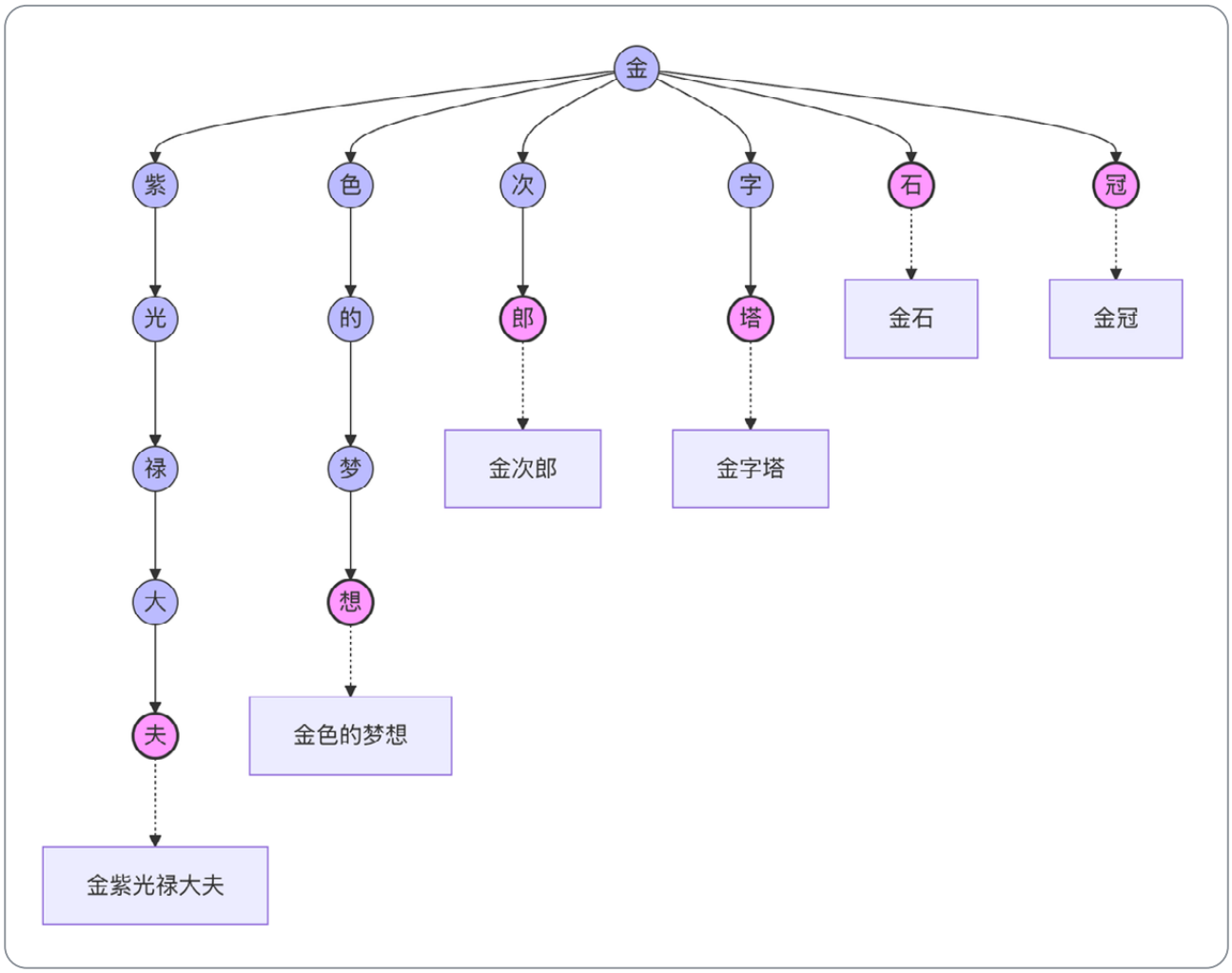


图4: 专词前缀树

其他

除了核心思路以外，在工程落地实践中，帮助用户更好的维护专词术语表，进行翻译测试，调查专词相关的问题也是非常重要的。所以方案中专门构建了 Portal (如下图)，提供专词上传 / 查询 / 检查的能力，以及测试翻译的界面，可以切换不同的专词版本进行翻译测试。

图5: 专词翻译界面

工程架构层面如下图所示，专词管理部分，利用 KV 数据库 (Dynamodb) 来存储专词映射，通过参数中心 (ParameterStore) 来记录各种专词表与 KV 数据表的关系。翻译部分，调用了 Bedrock 的 LLM，整个后台服务逻辑部署在 Lamda 中，使得整个方案都是轻量的 Severless 架构。

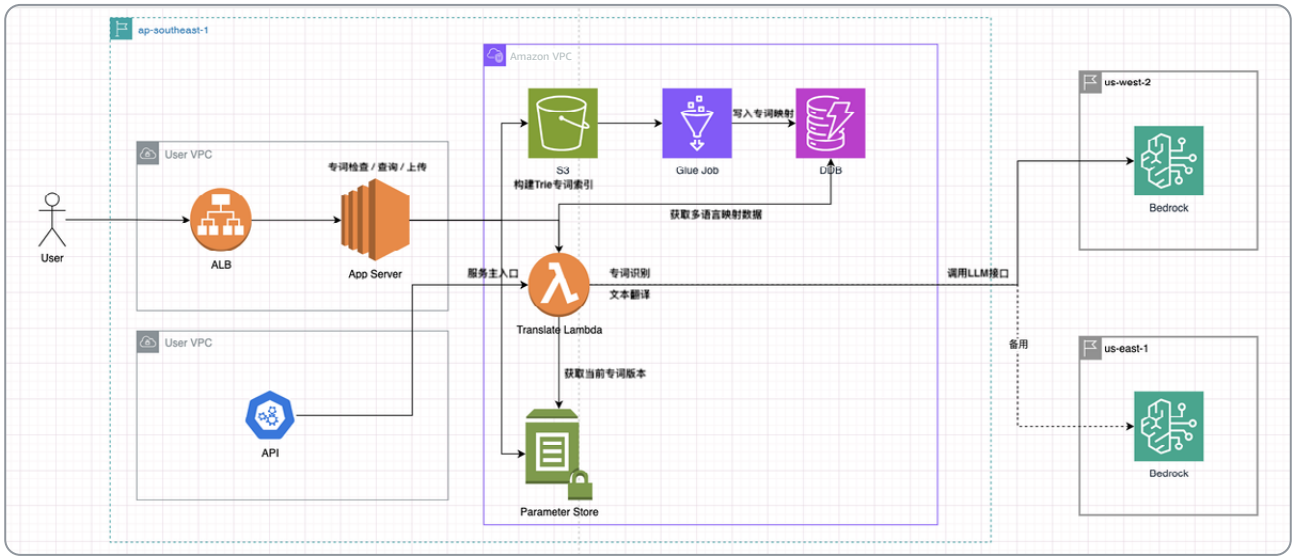


图6:专词翻译方案架构图

通过具体的实践，对于这种方法的优劣势有如下总结：

优势方面：

- 可以保证术语召回的精准性及一致性，实现全局的术语一致性
- 额外的 Token 消耗少
- 便于长期维护和对翻译问题的 Trouble Shooting

劣势方面：

- 实现相对复杂
- 无法提供翻译风格的参考信息

3. 基于 MTQE 翻译质量评估的优化思路

前面提到的机制都是通过提供更全面和准确的上下文来优化模型的翻译效果，但是有些问题通过这种思路并不是很合适。比如之前提到的翻译拒答问题，单位量词问题，很难引入参考信息来提示大模型。这些问题的本质都属于模型本身的能力问题和稳定性问题。

这些问题产生也往往和模型的训练机制有关，比如语种夹杂的问题，这种情况就是和训练数据中大量出现的语言间借词有关，并不影响理解，但某些业务不接受这种情况。比如格式变化问题，是因为大模型一般有较强的使用 Markdown Bullets 格式进行输出的倾向。又比如，敏感拒绝翻译，这和大模型的安全性对齐机制有关，由于模型概率输出的特性，这些问题通用大模型无法完全避免。通用大模型不可能够能完全契合某特定业务的要求。

通过微调可以让模型以业务诉求的目标，但大模型发展日新月异，为某个特定业务专门微调一个以翻译为目的的大模型，和其他头部大模型在翻译上效果上相比很难保证其领先性，而且其巨大的成本和难度并不适合作为一个广泛有效的技术手段。

为了在业务诉求，成本和技术难度等多种因素取得平衡。建议构建服务于专属业务的翻译质量检测流程。它可以和任意翻译模型配合使用。

3.1 基于 LLM-as-a-judge 的评估方法

机器翻译质量评估（Machine Translation Quality Evaluation），用于在没有参考译文的情况下预测机器翻译输出的质量。如下是一些使用 LLM 进行翻译评估的的思路。

直接进行评估：

定义 LQA (Linguistic Quality Assurance) 的要求，即语言学家或者译员对对翻译要求制定出一套标准。并为每种分类类型分配等级（如中性，轻微，重大，关键）、每种等级进行分值设定。最后由 LLM 理解标准后，对翻译内容评估和打分。

以下运营人员提供的业务评估标准示例：

1. 产品描述准确性 (Product Description Accuracy)	S 级 (卓越) - 5 分	• 完美传达产品特性、功能和卖点 • 技术规格和参数翻译精准无误 • 产品使用场景和优势描述具有说服力和吸引力

	A 级 (优秀) - 4 分	• 产品信息不准确
		  ◦ 产品特性和功能描述准确，偶有小瑕疵
		  ◦ 技术规格和参数基本准确，极少错误

	B 级 (合格) - 3 分	• 规格错误
		  ◦ 核心产品信息正确，但部分次要信息不够精准
		  ◦ 约 10% 的产品特性或技术规格描述有误

	C 级 (需改进) - 2 分	• 严重产品描述错误
		  ◦ 多处产品信息错误，可能误导消费者
		  ◦ 技术规格和参数翻译有明显问题

	D 级 (不合格) - 1 分	• 产品描述严重失实，完全误导消费者 • 技术规格和参数翻译错误导致产品信息完全不可靠
2. 营销说服力 (Marketing Persuasiveness)	S 级 (卓越) - 5 分	• 翻译具有极强的营销感染力，完美传达品牌调性 • 文案富有创意，能有效触发目标消费者的购买欲望 • 完美适应目标市场的消费文化和购物心理

	A 级 (优秀) - 4 分	• 营销效果不足
		  ◦ 翻译具有良好的营销感染力，基本传达品牌调性
		  ◦ 文案有吸引力，能够引起消费者兴趣

	B 级 (合格) - 3 分	• 品牌调性问题
		  ◦ 翻译基本传达了营销意图，但感染力不足
		  ◦ 文案平淡，缺乏吸引力和说服力

	C 级 (需改进) - 2 分	• 严重营销问题
		  ◦ 翻译未能有效传达营销意图，缺乏说服力
		  ◦ 文案生硬，无法引起消费者共鸣

	D 级 (不合格) - 1 分	• 翻译完全丧失营销效果，可能产生负面影响 • 文案不专业，损害品牌形象

3. 用户体验一致性 (User Experience Consistency)	S 级 (卓越) - 5 分	• 所有界面元素翻译完美统一，增强用户体验 • 购物流程、结账和支付相关术语精准无误 • 错误提示和帮助信息清晰易懂，有效指导用户
	A 级 (优秀) - 4 分	• 界面元素翻译基本统一，用户体验良好 • 购物流程术语准确，偶有不一致 • 评论区域：界面不一致处
	B 级 (合格) - 3 分	• 流程术语问题 ◦ 界面元素翻译有明显不一致，但不影响核心功能 ◦ 购物流程中约 10% 的术语使用不统一
	C 级 (需改进) - 2 分	• 严重用户体验问题 ◦ 界面元素翻译混乱，影响用户操作 ◦ 购物流程术语不一致，导致用户困惑
	D 级 (不合格) - 1 分	• 界面元素翻译严重不一致，导致用户无法顺利完成购物 • 购物流程术语错误，使平台几乎不可用
4. 合规与安全性 (Compliance & Safety)	S 级 (卓越) - 5 分	• 所有法律条款、隐私政策和用户协议翻译精准无误 • 完美处理地区特定的法规要求和合规信息 • 产品安全警告和使用注意事项翻译清晰完整
	A 级 (优秀) - 4 分	• 合规问题 ◦ 法律条款和政策翻译准确，偶有小瑕疵但不影响法律效力 ◦ 地区法规要求基本满足，极少疏漏
	B 级 (合格) - 3 分	• 法律术语问题 ◦ 法律条款和政策翻译有明显不足，但不影响核心法律保护 ◦ 部分地区特定法规要求未完全满足
	C 级 (需改进) - 2 分	• 严重合规风险 ◦ 法律条款和政策翻译有严重问题，可能影响法律效力 ◦ 多处地区法规要求未满足，存在合规风险
	D 级 (不合格) - 1 分	• 法律条款和政策翻译错误严重，完全丧失法律保护作用 • 忽视地区法规要求，存在重大法律风险

下面是针对上面业务要求的部分提示词示例：

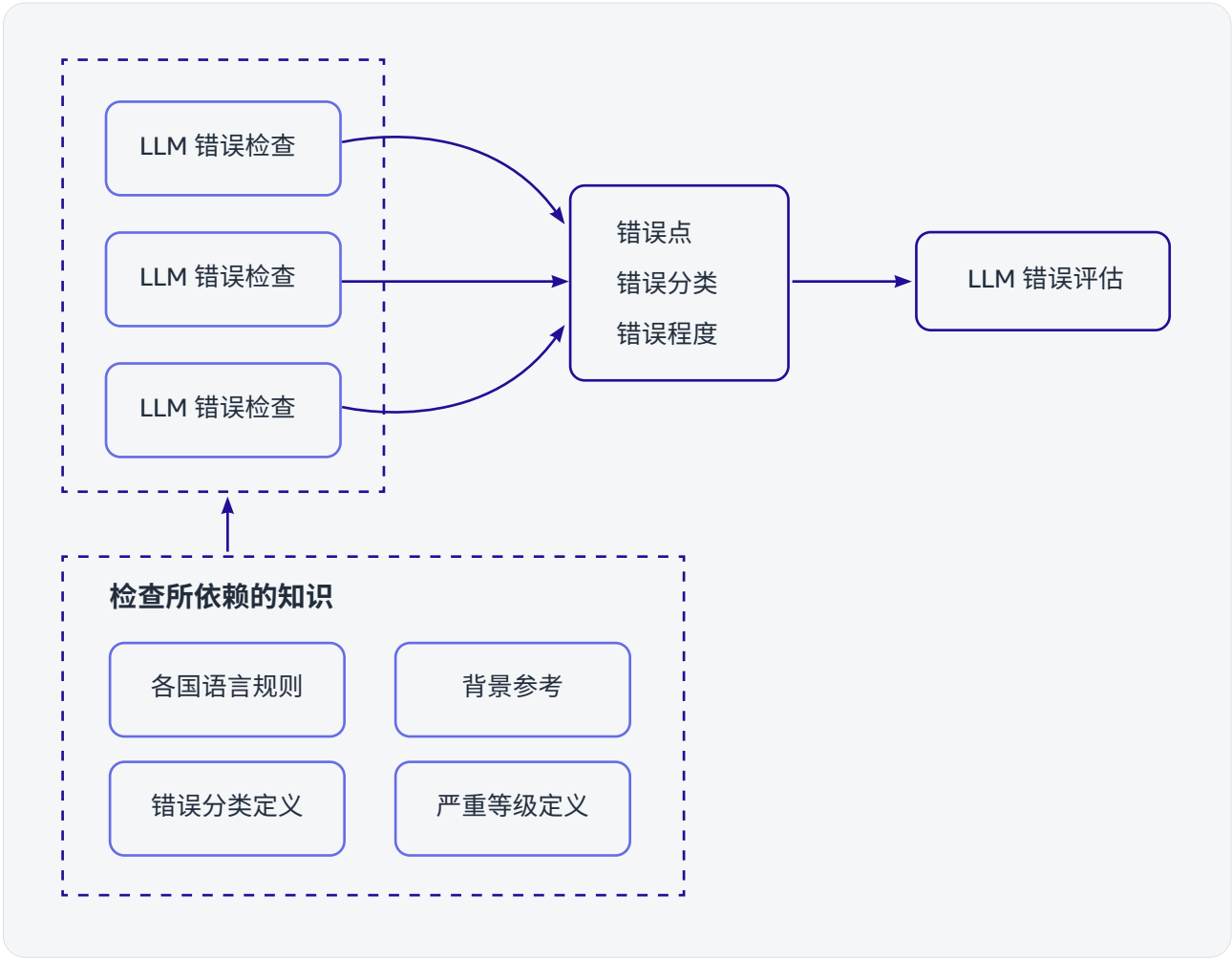
代码块

1. **产品描述准确性评分标准：**
   分数5：完美翻译，产品特性、功能、规格和参数全部准确无误，卖点表达清晰有力。
   分数4：高质量翻译，产品核心信息准确，错误或遗漏不超过5%，不影响消费者理解。
   分数3：可接受翻译，产品关键信息正确，错误或遗漏不超过10%，消费者仍能获取基本信息。
   分数2：需改进翻译，产品信息有明显错误或遗漏，约占30%，可能导致消费者误解。
   分数1：质量差翻译，产品信息错误或遗漏达50%，严重影响消费者决策。
   分数0：不可接受翻译，产品信息完全错误或具有误导性，可能导致投诉或退货。
2. **营销说服力评分标准：**
   分数5：卓越营销翻译，完美传达品牌调性，文案极具创意和说服力，完全符合目标市场消费文化。
   分数4：优质营销翻译，品牌调性表达清晰，文案有吸引力，仅有少量表达不够地道，整体营销效果良好。
   分数3：标准营销翻译，基本传达营销意图，文案平实但可接受，营销感染力有限但不影响主要卖点传达。
   分数2：弱效营销翻译，品牌调性表达不清，文案缺乏吸引力，约30%的营销元素未能有效传达。
   分数1：失效营销翻译，无法引起消费者共鸣，文案生硬，超过50%的营销元素丧失效果。
   分数0：负面营销翻译，可能损害品牌形象，文案完全不专业或包含不当内容。用户体验一致性评分标准：
   分数5：完美一致性，所有界面元素、购物流程术语和提示信息翻译统一且精准，显著提升用户体验。
   分数4：良好一致性，界面元素翻译基本统一，购物流程清晰，仅有少量不一致但不影响操作。
   分数3：基本一致性，界面和流程术语有约10%的不一致，用户可能注意到但不会影响完成购物。
   分数2：明显不一致，界面和流程术语有约30%的不一致，影响用户体验但大多数功能仍可使用。
   分数1：严重不一致，界面和流程术语有约50%的不一致，用户操作困难，可能导致放弃购物。
   分数0：完全混乱，界面和流程术语完全不一致或错误，平台基本不可用。
3. **合规与安全性评分标准：**
   分数5：完全合规，所有法律条款、隐私政策、安全警告和地区特定法规要求翻译精准无误，法律保护完整。
   分数4：高度合规，法律和安全信息翻译准确，仅有极少量表述不够严谨但不影响法律效力。
   分数3：基本合规，核心法律保护内容正确，约10%的法律术语或地区特定要求有不精确之处。
   分数2：合规风险，约30%的法律条款或安全信息翻译有问题，可能影响部分法律效力。
   分数1：严重合规缺陷，约50%的法律条款或安全信息翻译错误，法律保护作用大幅减弱。
   分数0：不合规，法律条款和安全信息翻译完全错误或缺失，存在重大法律风险和责任隐患。
4. **评分指导补充说明**
   评分时请考虑目标市场特性：不同地区的消费者有不同的购物习惯和期望，评分应考虑翻译是否符合特定市场的语言习惯和文化背景。
   专业术语一致性：电商平台中的专业术语（如SKU、配送方式、支付选项等）应保持一致，任何不一致都应在相应类别中扣分。
   跨类别影响：某些严重错误可能同时影响多个类别，例如产品尺寸单位错误既是准确性问题也可能是安全问题，应在相关类别中都予以考虑。
   优先级考量：在最终评估中，"合规与安全性"和"产品描述准确性"应被赋予更高权重，因为这些直接关系到法律风险和消费者权益。
   评分证据：所有低于满分(5分)的评分必须提供具体例子和建议，以帮助翻译人员理解问题并改进。

建设评估工作流：

在“直接评估”的方法基础之上。为了保证 LLM 输出的错误是真实可靠的，可以设定“错误识别”和“错误可信度评估”2 个环节进行设计，不同的模型做不同的任务（避免单模型自问自答导致幻觉）。同时在“错误识别”环节可以引入多个 LLM，让每个 LLM 基于规则都对译文进行一次检查后输出错误点 / 错误分类 / 错误等级。然后由负责评估的模型对各个专家 LLM 输出的总体结果进行置信度评估，然后再依次对每个专家 LLM 输出的错误点进行投票记数 (认同就加分，不认同不给分)。

最后对每个错误的投票记数和专家排名记分加权求和，得出每个错误的分数。如下为工作流评估示意图：



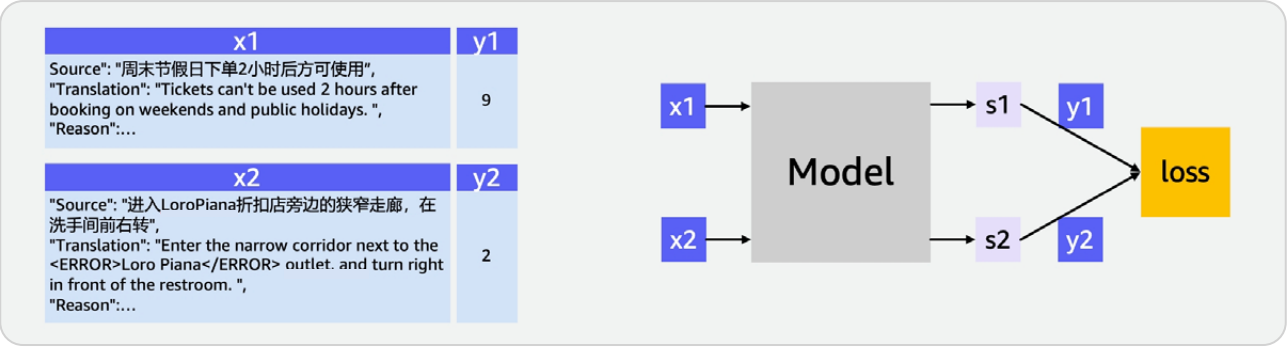
3.2 基于模型微调的评估方法

基于 LLM-as-a-judge 构建的 MTQE 流程，在实践中发现，仍然可能会存在一些问题。比如召回的问题太多，打分过于集中难以筛选，与人工打分的尺度始终存在一些差别。为了解决这些问题，引入了翻译错误打分模型。

3.2.1 翻译错误可信度打分模型

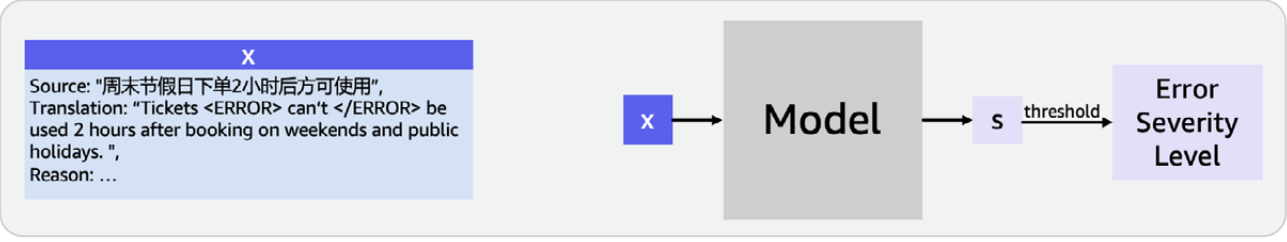
通过人工标注的数据（原文 / 译文 / 错误点 / 错误原因和分类）去训练“评估”模型，可以帮助评估模型更加客观的对翻译错误进行评估，提升整体评估的准确率，使其更接近人工的评判标准。在上述 " 评估工作流 " 的基础之上，训练开源模型 (Bert or Mbart) 去构建打分模型（最后的 Hidden Layer 层加线性层输出分数，用于对“错误”进行打分），从而基于分值来评估错误 (错误等级) 的严重程度。

1. 训练阶段



原始语料包含原文 / 译文 / 错误点 / 原因，以及错误等级所对应的分值。组装成两两配对的样本示例 (正确｜错误，严重错误｜重大错误等)，输出错误得分。通过 Pairwise Loss 计算实际得分和样本分数的差距，不断优化模型对错误严重程度的判别能力。

2. 推理阶段



“推理阶段”输入原文 / 译文 / 错误点 / 错误原因。模型输出错误点的最终得分，通过得分匹配出“错误等级”。

3.2.2 翻译错误识别模型

除了仅仅对翻译错误进行打分，还可以微调大模型直接进行错误识别，且同时输出错误等级。一般 7B 级别的模型也可达到不错的效果，从而避免在 MTQE 中反复调用昂贵的商业大模型，这样 MTQE 也可以应用在一些费用敏感的场景中。

模型业务目标

在真实的业务中，比较关注是避免影响业务的明显错误，需要足够高的检出率，需要能对错误进行分级。按照这个要求，我们拟定了下面的错误类型，需要模型能识别这些类型的错误，给出错误等级分方便分级处理。

错误编号	错误类型
1	拒绝翻译敏感内容
2	出现非目标语种词汇
3	出现不相关或者无意义的重复词汇
4	拼写 / 语法 / 符号错误
5	数量 / 单位 / 量词错误
6	格式变化，如添加编号
7	关键词遗漏或者不恰当

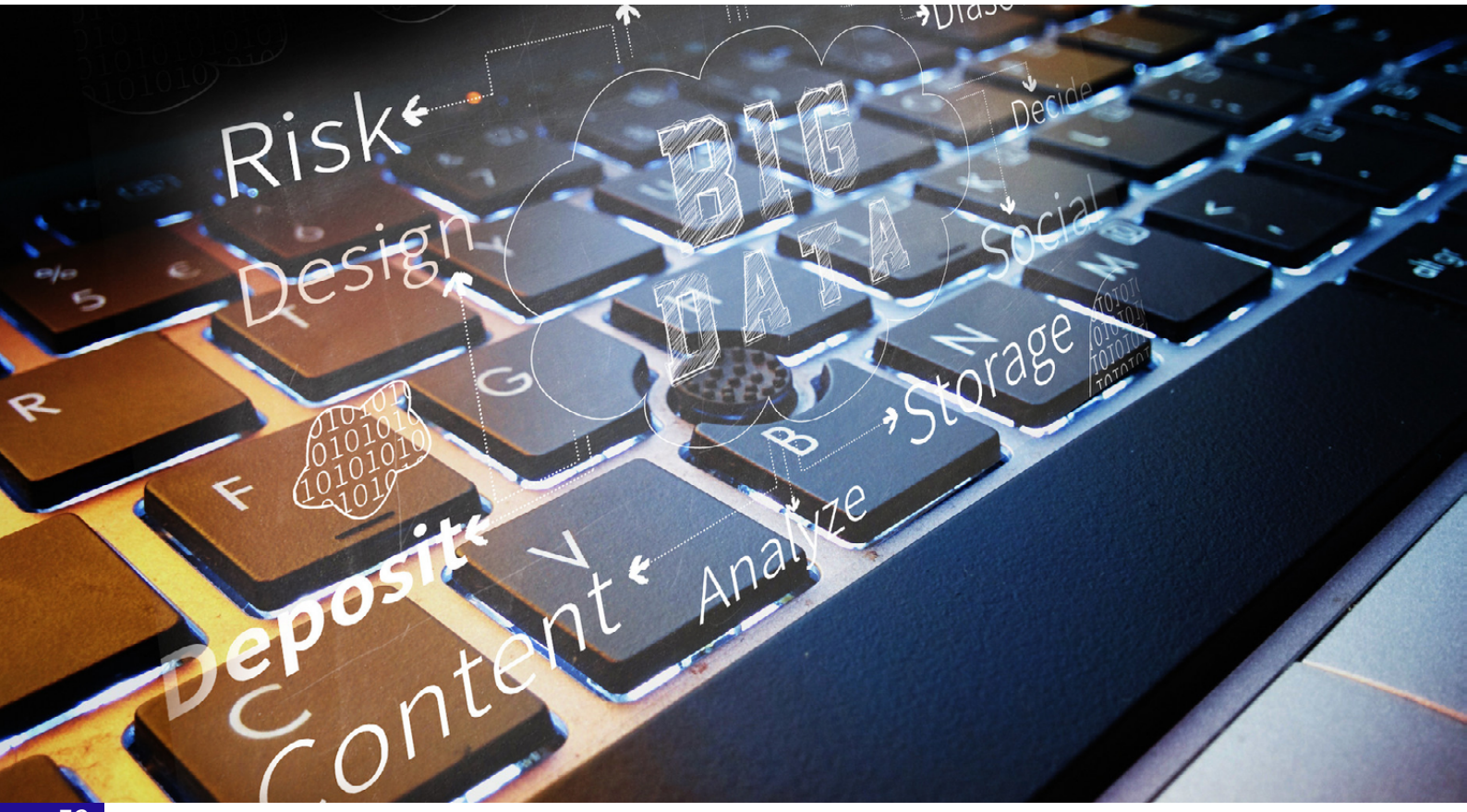
训练数据构造

根据业务目标和对数据的分析探索，通过不断训练观察迭代，主要发现两方面的有效优化：

1. 划分简单问题和复杂问题

编号 1-6 的错误，都属于容易总结的模式类错误，具备非常强的特征，容易被模型所掌握。而编号 7 的错误，缺乏显著的模式，需要大量的词汇知识。前者属于简单问题，后者属于复杂问题。由于在真实场景中，编号 1-6 错误发生频次低，能采集到的数据量和编号 7 的错误不在一个数量级，存在数据不平衡问题，训练单一模型对这 7 种问题进行识别会引起编号 1-6 的识别准确率大大下降，如果对编号 7 错误数据采取下采样，那么会导致编号 7 问题的效果无法保证，对于复杂问题，数据量越丰富越好。

为了平衡成本和效果，编号 1-6 错误训练一个模型，编号 7 错误则单独训练一个模型。编号 1-6 对数据量的要求没那么高，数据总量达到千级别，然后在类别进行一定的数据平衡即可。编号 7 错误则需要尽可能采集更多数据，数据越多越好，微调模型见过的案例越多越好。



2. 利用 COT 技巧提升分数输出的可靠性

由于大模型 Decoder 的自回归特性，前面生成的 Token 对于后面的 Token 影响巨大，如果直接输出得分，缺乏基于事实分析的过程，其随机性非常强，在模型训练的时候，由于参数尺寸 (<7B) 和数据规模 (千条) 的影响，模型并不能发现翻译结果和分数之间内在的联系，需要利用模型本身的 Reasoning 能力，先对翻译情况进行推理，得到了前序的这些 Token 以后，模型在后续输出时会根据这些前序 Token 中的观察和思考，提升正确 Rating 出现的概率。

优化后的训练数据样例如下：

代码块

```
{
  "messages":[
    {
      "role":"system",
      "content":"You are an expert linguist specializing in translation quality assessment. Your task is to evaluate the translation produced by translation model. Assess the translation based on the following criteria:
        1. Sensitive content should not be refused to translate
        2. No non-target language word appears
        3. No irrelevant or useless repetitive words.
        4. No Spelling, abnormal symbols and grammar errors detected
        5. Quantity, Quantifiers and Units are translated accurately
        6. Format maintained between source and translation. No added numbering/bullet
        The target language is zh-cn, please evaluate translation quality, and give your rating(0.0-5.0). "
    },
    {
      "role":"user",
      "content":"Here is the source text in <src> tag and also its translation from an translator in <translation> tag. <src> Reformed Soteriology </src> <translation> 改革宗 Soteriology </translation>"
    },
    {
      "role":"assistant",
      "content":"<think>The translation contains untranslated English word 'Soteriology' which should be fully translated into Chinese. '改革宗' alone does not fully communicate the doctrine of salvation term.</think>, my ratings is [5, 2, 5, 5, 5, 5, 5]"
    }
  ]
}
```

其中 <think> 和 </think> 之间为大模型的分析内容，这部分内容需要在“My ratings is ”最终得分的输出之前。通过这种形式的模型输出，我们即可以得到错误类别和错误等级，也可以得到错误的具体信息，可以提供给后续做翻译优化的模型做参考。

训练方法选择

Lora Finetune vs Full Finetune

一般来说 Full Finetune 会比 Lora Finetune 拟合能力更强， 容易达到更好效果，但 Lora Finetune 对于资源的要求要低不少。在该场景下，通过实践对比，发现 Lora Finetune 无论是调整模型尺寸和 Lora 的训练参数 (Lora_rank, Lora_alpha)，模型效果均不稳定。而 Full Finetune 很容易取得稳定效果。所以建议采用 Full Finetune 方式进行 SFT 训练。

DPO vs GRPO

通过 Supervised Full Fintune 已经可以实现比较好的输出效果，无论是格式还是思考过程，但唯一有不足的地方是 Supervised Full Fintune 在计算 loss 时，计算范围为整体 Assistant 的输出 Token，Rating 的部分占比较小，特别是个别维度的分数值差别更小，比如下面这个例子：

代码块

```
{
    "src": "MCLVN",
    "translation": "无法翻译\"MCLVN\"，这似乎是一个不明确的缩写或产品代码。需要更多信息才能进行正确翻译。",
    "label": "<think>The translation avoids translating the term by claiming it's unclear, violating the first criterion that requires an attempt at translation. The explanation is reasonable but doesn't fulfill the requirement to attempt translation.</think>, so my ratings is [2, 5, 5, 5, 5, 5, 5]",
    "predition": "<think>The translator has declined to translate the source text by explaining that it's an unclear abbreviation. While the explanation is logical, the task requires attempting a translation rather than refusing it.</think>, my ratings is [3, 5, 5, 5, 5, 5, 5]"
}
```

在 Supervised Full Fintune 训练时，分数一个是 2，一个是 3，在整个输出文本中占比太小。很难优化到这里，但是在业务上，我们规定 [0,1,2] 分需要进行人工干预，[3,4,5] 是轻微错误不需要引入人工。

为了把这部分在模型上进行优化，尝试对比了 DPO 和 Rule-based Reward 的 GPRO 的强化训练。DPO 没有取得效果提升，反而出现了一些复读现象。然而 GRPO 由于可以通过 Rule-based Reward，把业务诉求和 Reward 精确对应起来，取得了一定提升。具体的实现方法可以参见实验部分。

实践经验总结

数据质量是关键，根据机器学习“Garbge in Garbge out”的定律，数据质量决定了模型的效果。训练数据的构造非常重要，其中 <think> 部分是通过大模型进行合成的，在实践过程中总结了二点经验：

1. 数据的多样性很重要

训练数据不能有任何显而易见规律或者文字模式。其中 COT 要循序渐进，不要一开始就暴露结论，而是应该先列举事实，然后再给对对应分析。合成时，这种输出顺序会保证合成内容的正确性，在微调模型时这种方式也有利于利用模型的基础能力，而不是死记硬背微调数据。

下面是一个负面例子，一开始还没摆事实就亮出了观点。

代码块

```
<think>
The translation performs well in several aspects:
1. No sensitive content is refused to translate
...
4. No spelling or grammar errors are present.
...
</think>
```

下面是另一个负面例子，按照顺序一条一条打分规则逐步分析，由于大多数 Bad Case 仅仅是个别规则出问题，正面的分析占绝大多数，这会导致训练总是倾向于给出正面的分析结果，引起问题召回率大大下降。

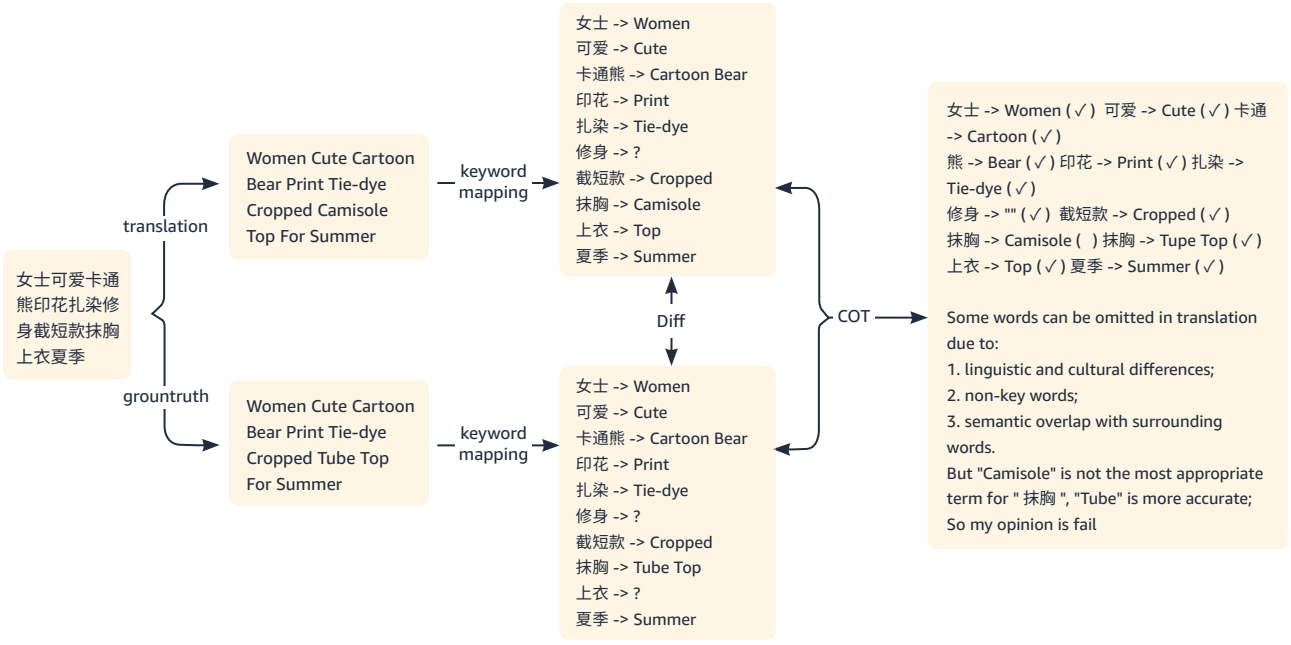
代码块

```
Let me check the content as thoroughly as possible.

1. No sensitive content issues
Yes, there are no sensitive content issues in this product description.

2. No non-target language appears
No, \"T-shirt\" appears in English within the Arabic text.
...
```

为了尽可能遵循 COT 过程先列举事实再分析总结的范式，在合成中也可以采用一些辅助手段定向优化。如下图所示，先对比错误和正确的样本差异，通过 LLM 抽取出词级别的对应关系，通过代码把差别精确找出来，然后再根据这些差别作为事实正确，再合成一些 COT 过程，这种方式会提升 COT 的质量。



■ 图7: 优化后的 COT 合成思路

此外，COT 的合成数据不宜太长，会引起无必要的输入延迟和计算成本增加。可以在合成的 Prompt 中强调，要求大模型给出简要精准的分析。

3.2.3 模型微调提效工具

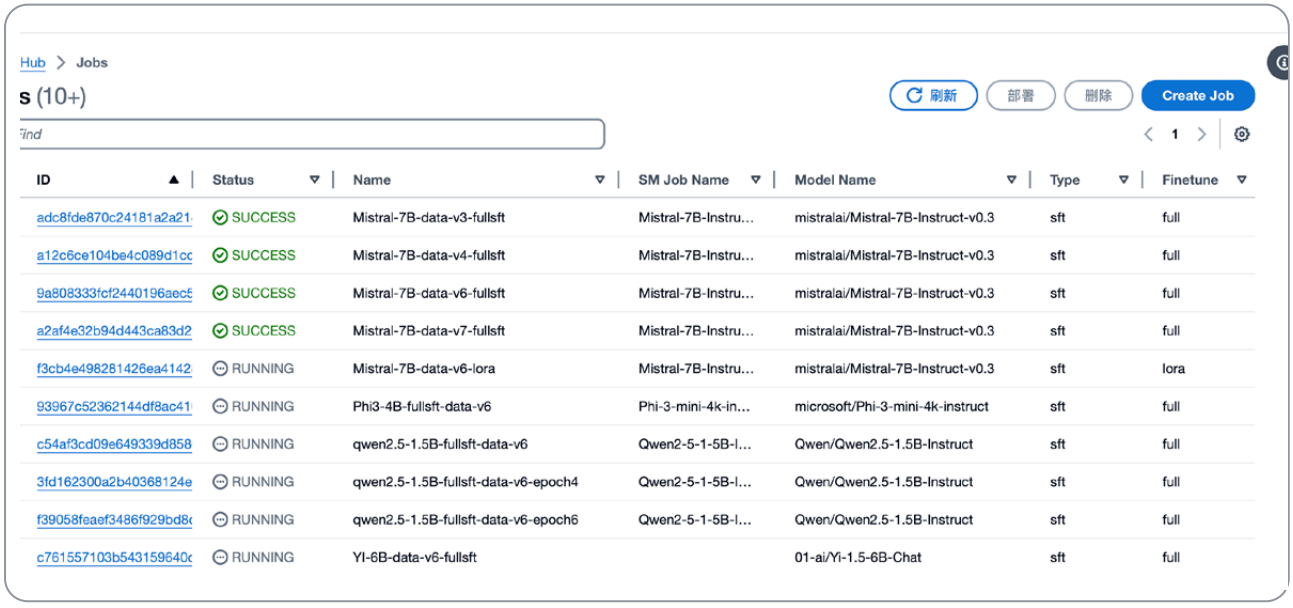
模型微调是需要反复实验分析的工作，相对于基于 Prompt 和 Retrieval-based 的优化，难度偏高工作量也更大。工欲善其事，必先利其器。在实践中，我们采用了 Model Hub 和 Dify 来进行模型训练和数据合成。

Model Hub

Model Hub 是亚马逊云科技在 Github 上的一个开源项目，基于 LLaMA-Factory 和 Amazon SageMaker 提供了一站式的模型微调、部署、调试的零代码可视化平台，可以帮助用户快速微调各类开源模型，高效同时地进行大量的实验和验证，提升模型微调的工程效率。特别适合希望用到亚马逊云上的训练资源，能够进行大量并行微调实验，但又不希望花太多精力关注 Amazon SageMaker 的使用方式和代码开发的用户。

它具备 LLaMA-Factory 的所有优势，能够支持 Pre-training,Supervised-Finetune, DPO, KTO, GRPO 的训练方法，可以直接部署微调 HuggingFace 上的几乎所有 LLM，训练中可以通过选项的形式引入任何公开的数据集， 同时它还可以使用亚马逊云上的 Spot 实例，降低训练成本。

在本文的实践中，在各个维度进行了对比实验，选择尝试了多种训练方法和模型，数据层面也进行了多次迭代，微调实验的次数超过 100 次，微调模型的部署和测试也进行了几十次。通过 Model Hub 大大的缩短了实验过程，让整个实验清晰可控的逐步推进。



Model Hub > Endpoints

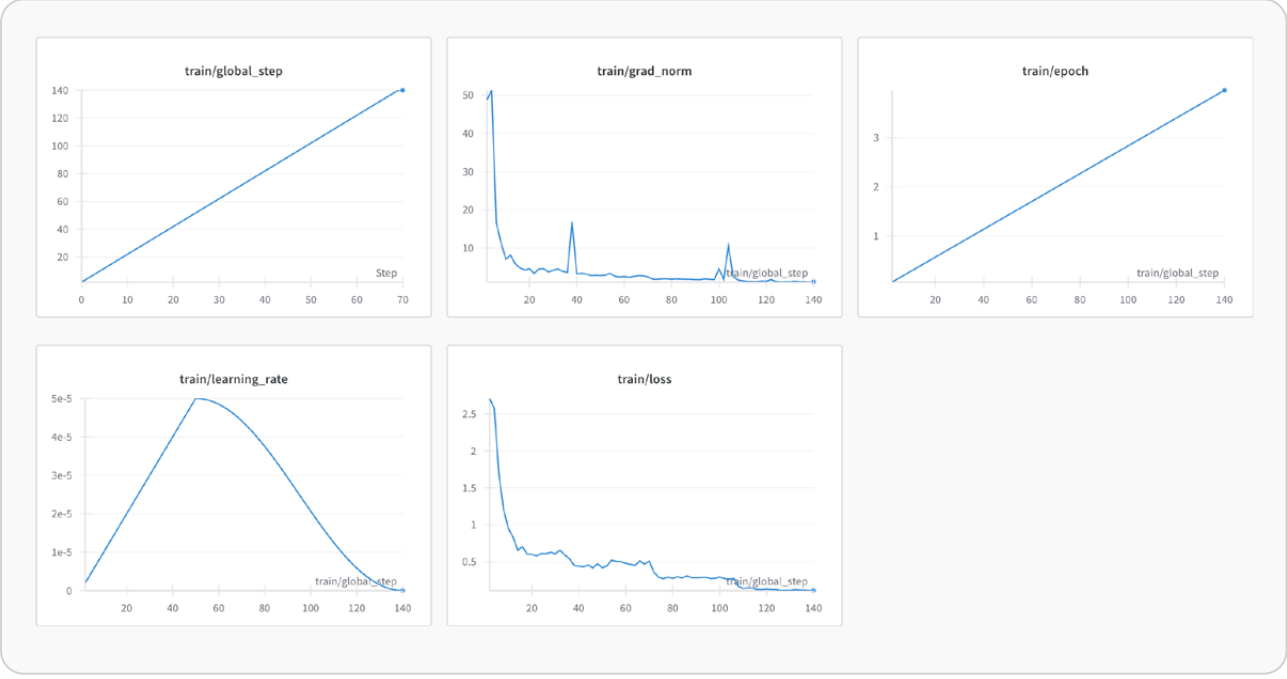
Endpoints (7+)

Refresh

Find

ID	Status	Endpoint Name	Model	Engine	Instance
6916c3a648c74e218121878579643eb0	INSERVICE..	Mistral-7B-Instruct-v0-3-2024-09-22-05-50-04-403	mistralai/Mistral-7B-Instruct-v0.3	auto	ml.g5.2xlarge
f6bc8eeb3e644941b60f93884407c393	INSERVICE..	Yi-1-5-6B-Chat-2024-09-23-06-34-34-403	01-ai/Yi-1.5-6B-Chat	auto	ml.g5.2xlarge
734796bb8d7d4cae8bfc5b04055ce6f9	INSERVICE..	Qwen2-1-5B-Instruct-2024-09-24-14-16-03-388	Qwen/Qwen2-1.5B-Instruct	auto	ml.g5.2xlarge
76857594e0fd4a29844aeaa8909ced60	INSERVICE..	Qwen2-1-5B-Instruct-2024-09-25-02-57-18-237	Qwen/Qwen2-1.5B-Instruct	auto	ml.g5.2xlarge
6ff18379c90d44f4b08df8a57c58cedd	INSERVICE..	Mistral-7B-Instruct-v0-3-2024-09-26-08-53-28-007	mistralai/Mistral-7B-Instruct-v0.3	auto	ml.g5.2xlarge
57a5609ef86f48e282817b06a671e48c	FAILED	Phi-3-mini-4k-instruct-2024-09-26-08-53-41-307	microsoft/Phi-3-mini-4k-instruct	auto	ml.g5.2xlarge
1fd8ad3195884ddb9b7701d252c787cd	FAILED	Qwen2-1-5B-Instruct-2024-09-26-09-05-16-853	Qwen/Qwen2-1.5B-Instruct	auto	ml.g5.2xlarge

Model Hub 支持接入 Wandb(一个用于机器学习实验跟踪、可视化和协作的工具)，可以方便的查验训练中的各种指标细节，帮助你观察了解 Loss， Learning_rate 等多种指标，从而更好的发现训练中的一些问题线索。



■ 图10: Wandb 指标图

Dify

Dify 是一个开源的大语言模型 (LLM) 应用开发平台，它融合了后端即服务 (BaaS) 和 LLMOps 理念，可快速搭建生产级的生成式 AI 应用。目前在 Github 上已经超 100k star 数，非常的流行。为了借助 Dify 的力量，作者之前的工作 - amazon-samples/dify-amazon-tool 这个仓库提供了众多的 Amazon 与 Dify 集成的模型连接器和工具。Dify 主分支已经整合 amazon-samples/dify-amazon-tool 的全部工作，使得 Dify 和 Model Hub 完成了接口层面的对齐，所有通过 Model Hub 部署的 LLM 模型 (On SageMaker) 都可以在 Dify 直接使用。

在训练数据合成阶段，作者利用 Dify 的 Workflow 编排功能，以低代码的形式，高效的编排了较为复杂的数据合成的工作流。其演进过程是从单次 LLM 调用，到多次 LLM 调用，再到并发多个 LLM 进行投票，再到结合代码进行一些逻辑处理。COT 合成数据的过程迭代了至少 10 轮，通过 Dify 工作流，让作者能够快速的通过可视化的方式进行数据校验和调优。

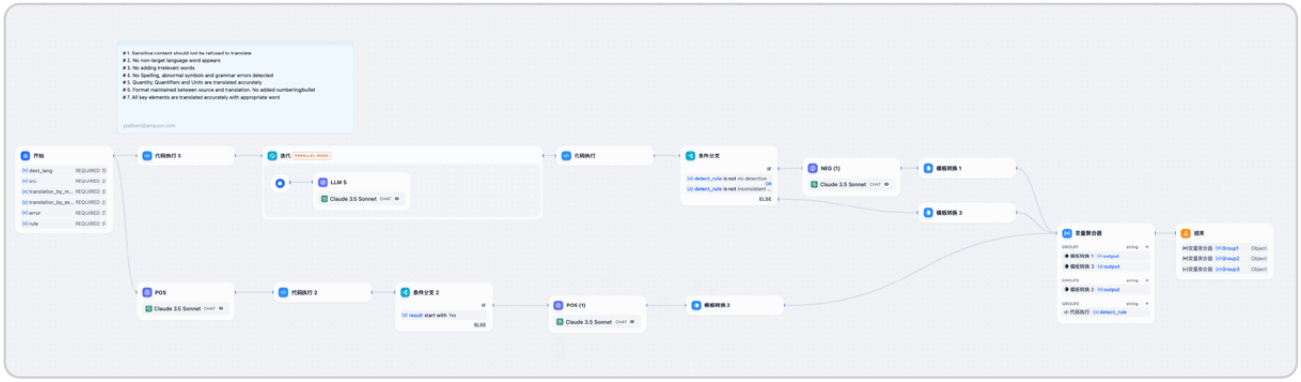
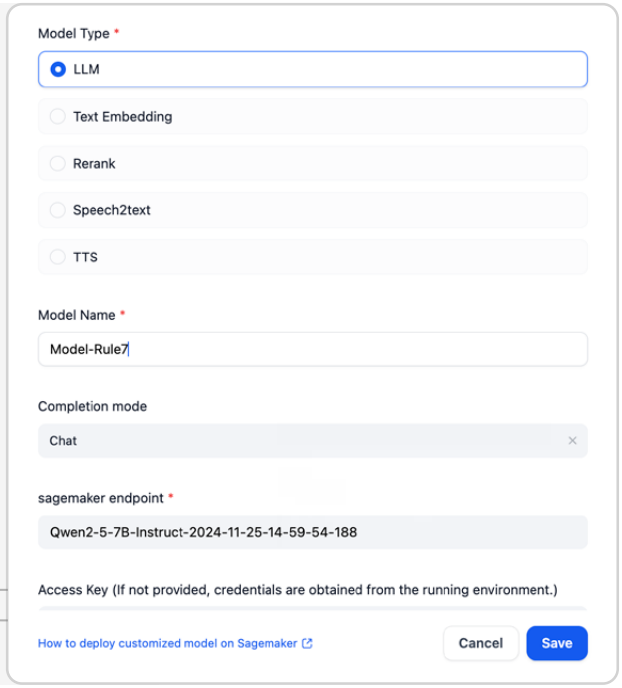
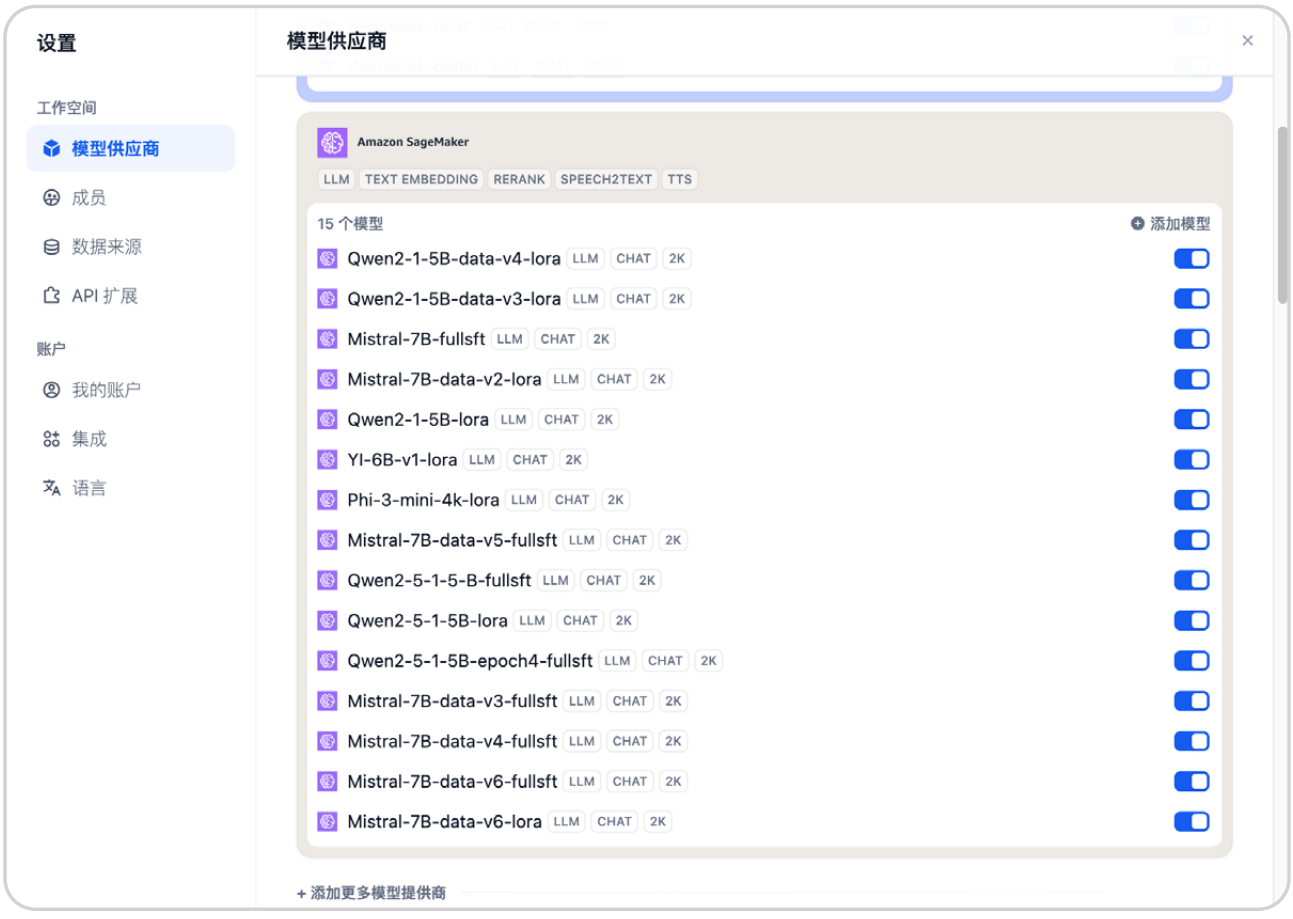


图11: Dify 样本合成工作流

模型训练完毕以后，在模型的测试阶段，作者利用 Dify 对接 SageMaker 中部署的微调模型，仅仅需要填写 SageMaker 中的 Endpoint 即可。如下图所示，左图为配置 SageMaker 的界面，右图为作者配置的所有用于测试的模型。

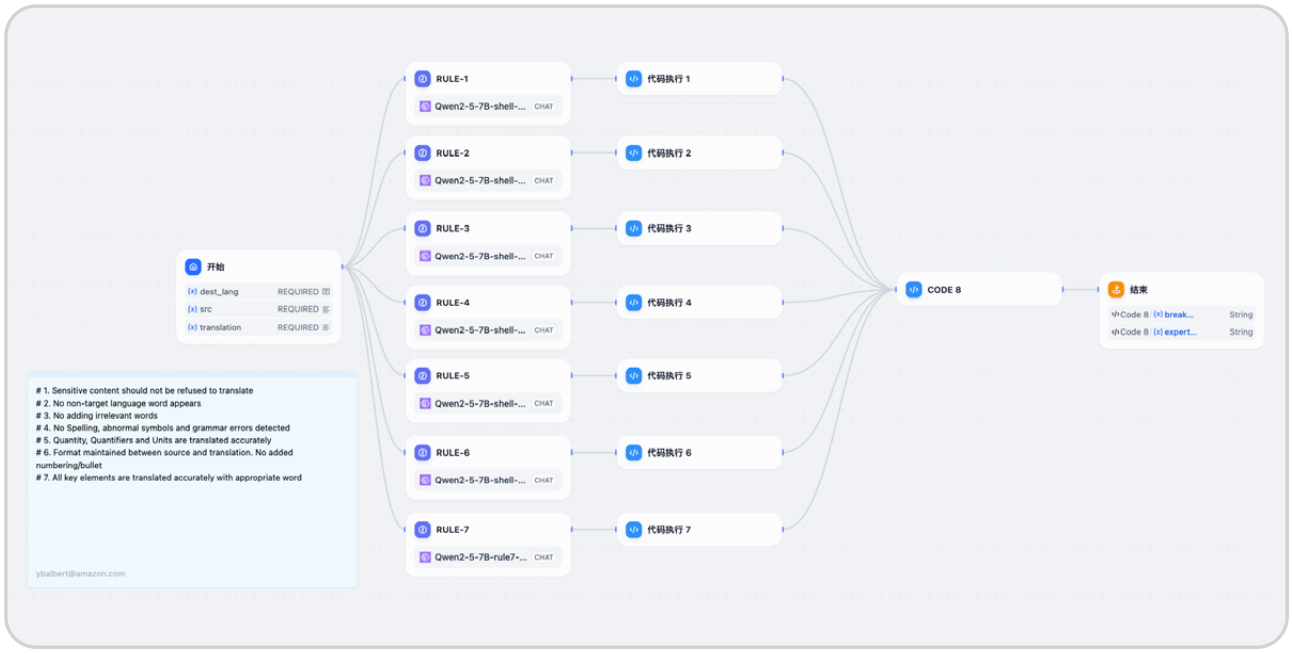


■ 图12: Dify 的 SageMaker 模型配置

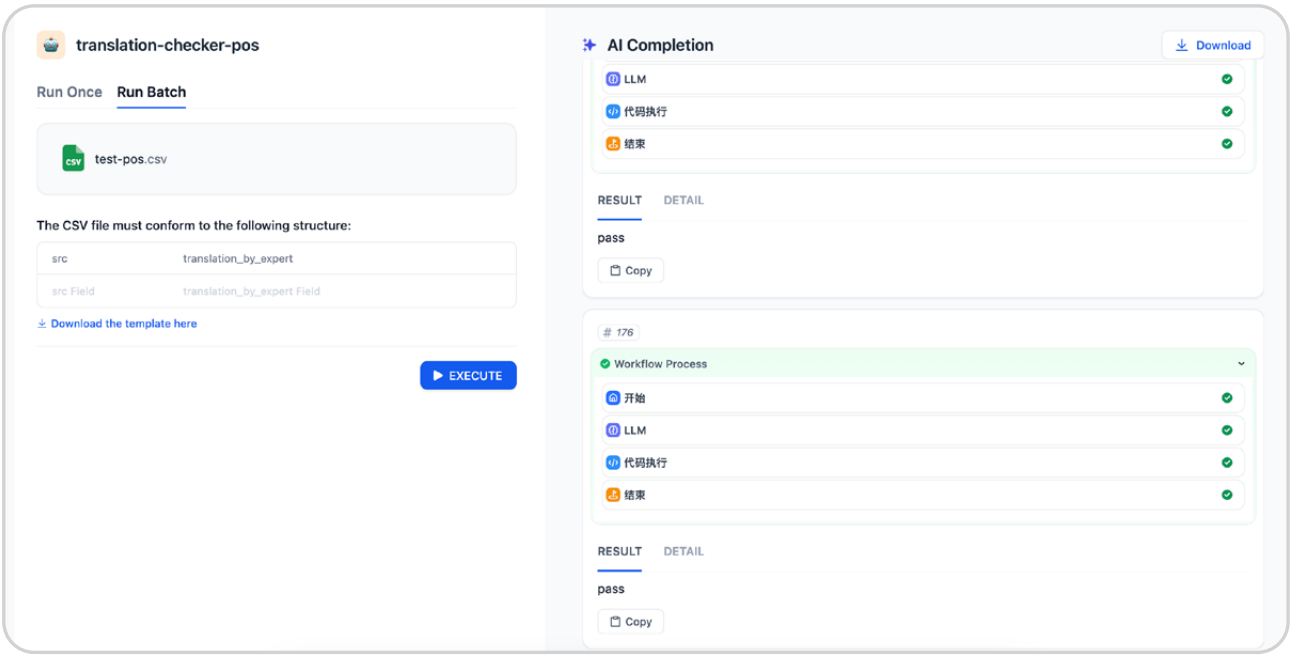


■ 图13: Dify 的 SageMaker 模型列表

模型配置好以后，可以很容易的编排到测试验证工作流中，比如下面的工作流可以并发的检测 7 条规则，在发布该工作流以后，就得到了一个公开可以访问的测试 API 供外部集成测试，并且可以获得一个测试前端页面，可以上传 CSV 文件，进行批量的测试，非常有利于分享给业务方，用于后续的验证评估。



■ 图14: Dify 的测试工作流



■ 图15: Dify 的工作流测试页面

05

LLM 翻译实验部分

五 LLM 翻译实验部分

1. 实验数据

数据领域	原始来源	数据详情
游戏	genshin-impact-ja-zh	原神的公开数据，存在中文 / 英文 / 日文的 ground truth
游戏	genshin-dictionary	原神的词典数据
电商	McAuley-Lab/Amazon-Reviews-2023	亚马逊商品评论数据中的商品标题，不存在 ground truth, 以 Claude Sonnet 3.5 翻译结果作为 Ground truth
混合	Korean - English Parallel Corpus	英韩的平行语料

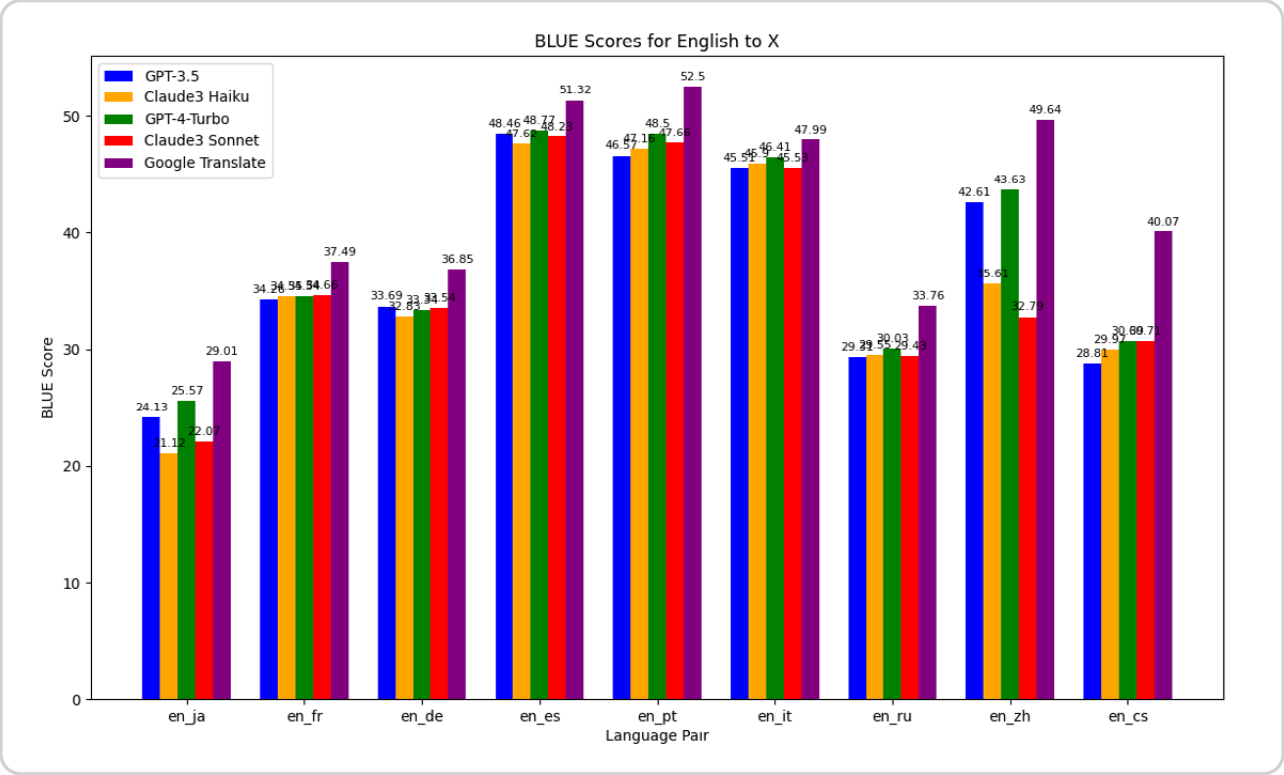
2. 实验评估方法

2.1 翻译质量评估所面临的问题

传统机器翻译的评估指标，是根据固定数学公式计算出来的，以学术领域最常用的 BLEU 为代表，它通过计算 N-gram（连续 N 个词）的匹配程度，来评估机器翻译的精确率。它广泛用于学术领域的效果对比，一般论文提出新的翻译方法 (模型) 时都必须提供 BLEU 值方便与之前的工作进行对比。

但是在 WMT2024 的论文中提出，指标自动评分最优的翻译系统 (Unbabel-Tower70B) 和人工评估存在不一致，反而认为总体表现最佳的系统是 Claude-3.5-Sonnet（赢得 9 个语言对），明确指出不认为自动评分的指标适合作为最终评判标准，而是认为人工评估应该作为翻译质量的最终评判标准。

这与我们的实践经验相符，我们也发现通过人工评估，LLM 翻译的效果要明显超过传统机器翻译 (NMT)，但是往往在 BLEU 值的比较中所体现出的情况并不一致。下图中 Google Translate 作为传统机器翻译方法的 BLEU 要远超 LLM 翻译。单独的 BLEU 值不能客观的体现翻译质量。



特别是在一些情况下失真的情况特别明显，比如文本长度短，或仅仅只有一份标准答案 (标准的评估集要提供三份答案，取多份标准答案 BLEU 的最高值作为分数)，下面例子很好的展现了这一点：

Source	The avalanche came amid the season's first major snowfall.	
Reference	雪崩发生时正值这个季节的第一场大雪。	
Translation	Translation A	Translation B
	雪崩在本季首次大雪中发生。	雪崩发生在季节的第一场大雪中。
Blue Score	7.83	54.09

2.2 翻译质量评估的优化路径

从评估指标角度，除了 BLEU 以外，学术界也提出了更多 的一些评估指标，如 SacreBLEU、TER、Meteor、Nist 等。

SacreBLEU

传统 BLEU 可能在分词 / 大小写等处理上的差异，导致评估标准没有完全对齐，SacreBLEU 是对传统 BLEU 的**标准化改进**，核心解决了评测中的可复现性问题。

Meteor

最接近人工评价的指标，结合精确度和召回率，还能支持同义词 / 词干匹配等。

Nist

BLEU 的改进版，侧重信息量，会加权罕见 N-gram；更关注信息丰富的词汇（如术语、低频词），对流畅性和语义覆盖不足，强调关键信息传递的任务。

以上三种评估指标，从他们的特点来看，具备一定的互补性，为了得到更可靠的评估，综合多种评估指标是一种可行的方式。

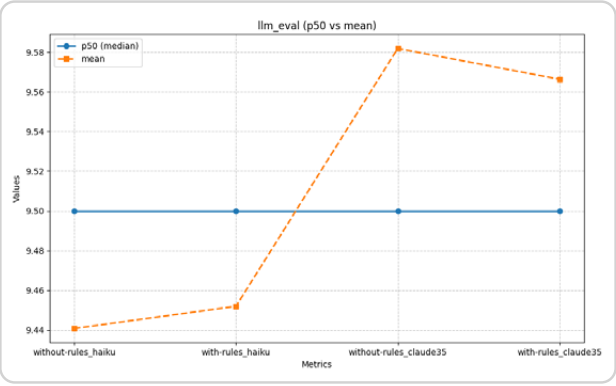
此外另外还有一种利用大模型评估方法。这种方法取决评估大模型的能力，评估分数多次之间不稳定。但相对于传统的评估指标而言，对评估集的数据质量要求更低（很多时候业务场景中的数据很难构建出包含多份参考答案的评估集 ），失真等问题的影响更小。

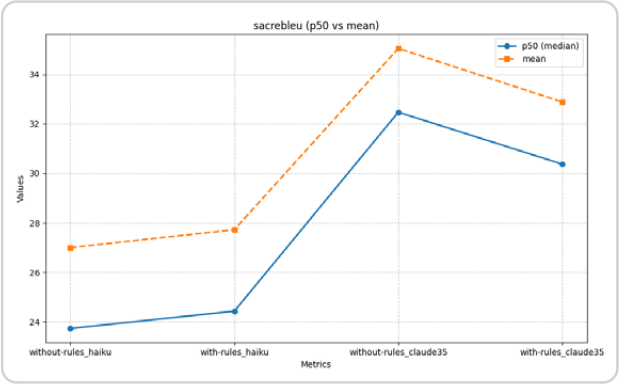
在下面试验的评估中，本文综合考虑了这些指标，尝试更客观的对各种翻译优化方法进行综合评估。

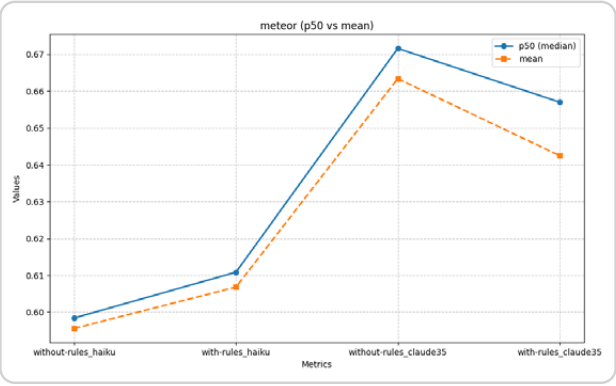
3. 基于 Prompt/Agent 优化思路的相关实验

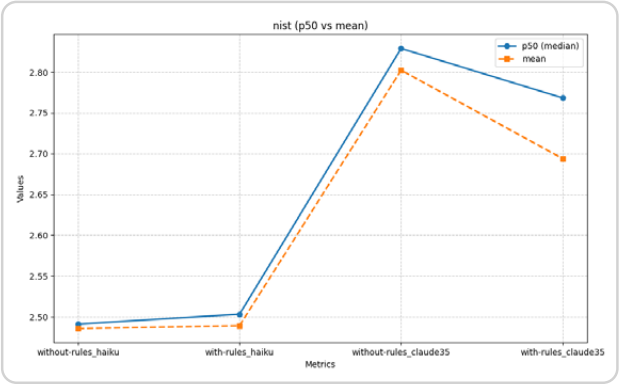
以下是第 4 章的 1.1 部分实验结果 , 采用的数据集为 Korean - English Parallel Corpus(数据集在第五章的第 1 部分), 实验通过“语法规则”提示词放入或者不放入来观察翻译的质量是否有提升。

实验数据共计 4800 条样例，分别使用使用 Haiku，Sonnet3.5 对样本数据进行测试，2 个 LLM 分别使用语法规则、不使用语法规则进行测试。在 Llm_eval、Sacrebleu、Meteor 和 Nist 多个指标上进行对比。









从测试结果来看有如下结论

1. 模型基础能力越强，翻译准确度越高，Sonnet 3.5 准确度明显优于 Haiku 这种小参数规模的模型。

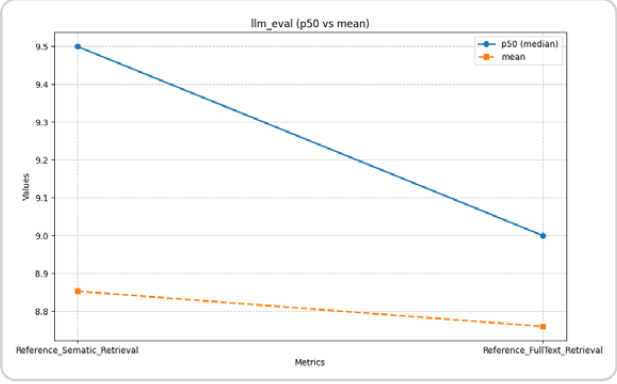
2. 加上语法规则后，Haiku 的翻译质量比不带语法规则效果有提升。但模型基础能力更强的模型 - Claude Sonnet3.5， 不使用语法模版也可以达到比较好的效果，加上 Prompt 语法模版反而可能会降低其翻译质量。

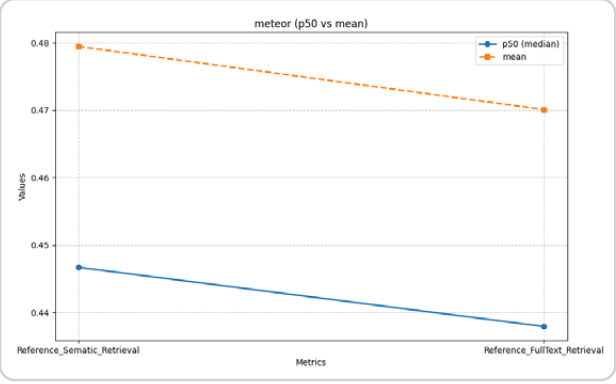
根据以上结论，如果对翻译质量有要求，优先提升基础模型能力，如果对于实时性和成本更关注，可以使用小模型，并叠加一些语法规则来改善其翻译效果。

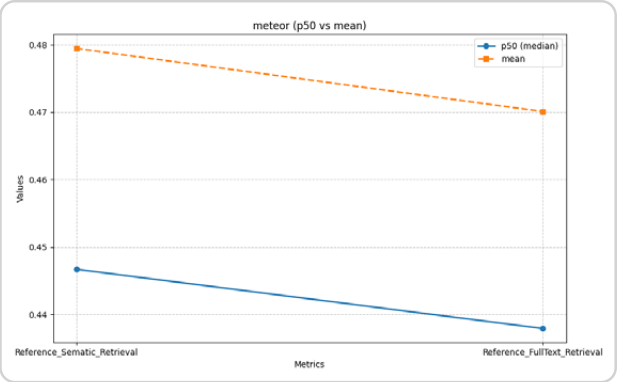
4. 基于 RAG 的优化思路的翻译方法相关实验

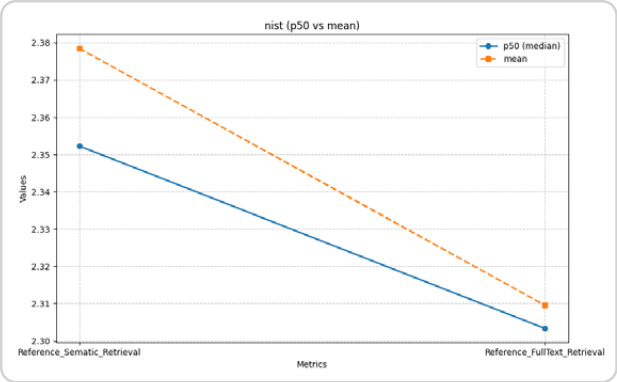
4.1 Reference_Retrieval 不同搜索方式的效果对比

本实验基于游戏场景的数据 genshin-impact-ja-zh，主要目的对 Reference_Retrieval 的不同搜索方式进行定量的对比。 样例库总数为 8323 条，共 2k+ 测试数据，分别结合通过 FullText Search 方式与 Semantic Search 方式召回的翻译样例进行翻译并评分，在 Llm_eval、Sacrebleu、Meteor 和 Nist 多个指标上进行对比。









根据以上的实验数据，可以得到简单结论，Semantic Search 各个指标均好于 FullText Search 方式。

从下面这个具体例子可以看出，FullText Search 召回的时候只考虑关键词，Semantic Search 更多考虑语义，召回的样例和要翻译内容的之间相关性更强，参考性也更强。

语种	原文	召回样例 (FullText Search)	召回样例 (Semantic Search)
英文	The Knight and his knights also fought for their land.	The knights, on expedition in a foreign land, would face the horror head on, their formation steely as the northern glaciers.	Many years later, when the cataclysm came, he led the knights to fight for his homeland.

4.2 大模型构建专词的效果评估

本实验基于原神的词典数据作为 Ground Truth 进行验证，输入的翻译对与原神词典重叠词汇共 453 个，利用最新的 Claude Sonnet 3.7 进行抽取。按照抽取的专词的专业性和准确性得分进行评估，得到如下的召回率指标。

过滤条件 (专业性与准确性)	召回专词量	召回率
specialty_score>1 & accuracy_scores >1	9807	438/453 = 96.68%
specialty_score>2 & accuracy_scores >2	8600	436/453 = 96.24%
specialty_score>3 & accuracy_scores >3	6222	429/453 = 94.70%
specialty_score>4 & accuracy_scores >4	4368	417/453 = 92.05%

当得分标准放宽时，召回率可以达到接近 97%，个别无法召回的专词一般是标准术语库中的词太短或者专业度不高，又或者是术语为复合词，如下面这些具体的例子所示。

代码块

```
## 短词 & 专业度不高
{"en": "Wish", "zh": "祈愿"}
{"en": "Makoto", "zh": "真"}
{"en": "Nameless","zh": "无名"}

## 复合词（大模型识别出了“大蛇远吕羽氏尊” & “堆高高大赛”）
{"en": "Orobashi no Mikoto", "zh": "远吕羽氏尊"}
{"en": "Pile 'Em Up", "zh": "堆高高"}
```

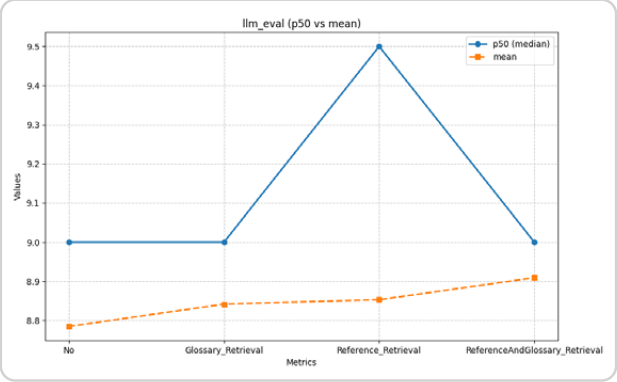
在实验中，抽取专词的精准率暂时不做评估，主要两个原因：1. Ground Truth 数据中的词汇并一定是所有专词，即使不在 Ground Truth 中，有不少抽取的专词也是正确。2. 多余抽取出的专词，并不会影响翻译质量。

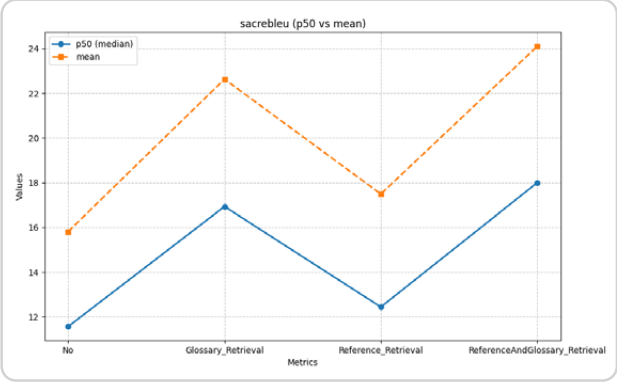
4.3 Reference_Retrieval 与 Glossary_Retrieval 方法的效果对比

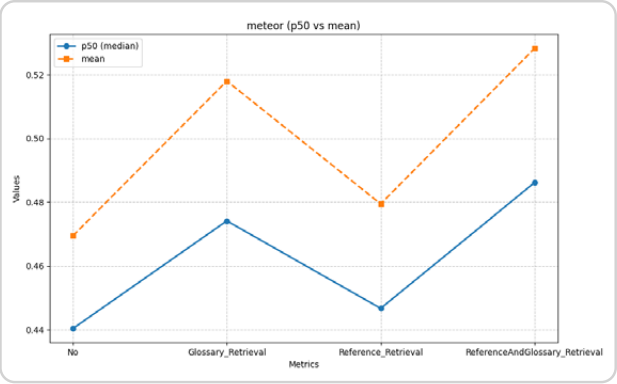
为了定量评估基于 Retrieval 优化思路的翻译方法，针对两个场景中数据对比了下面四种方法

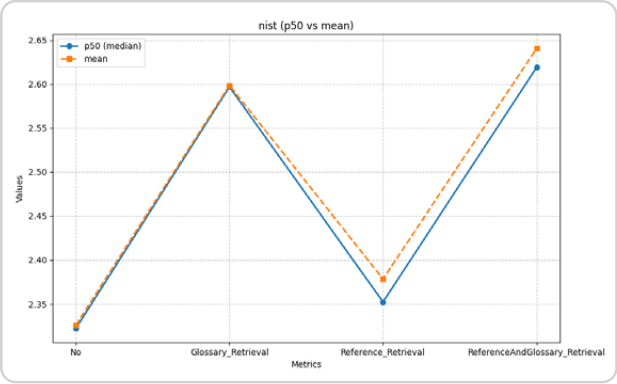
- No：无任何召回
- Glossary_Retrieval: 术语召回
- Reference_Retrieval: 历史样例召回
- ReferenceAndGlossary_Retrieval: 术语 / 样例叠加召回

- 游戏场景 - genshin 数据集 - haiku3

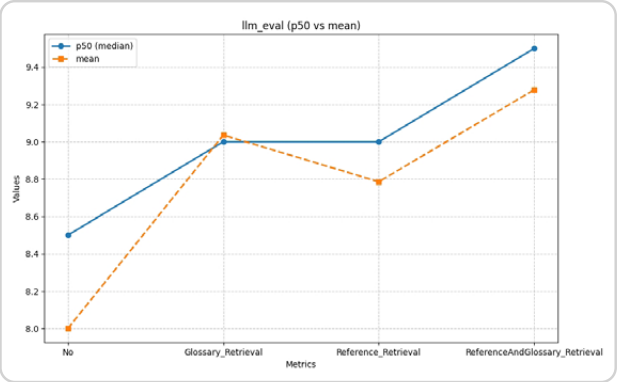


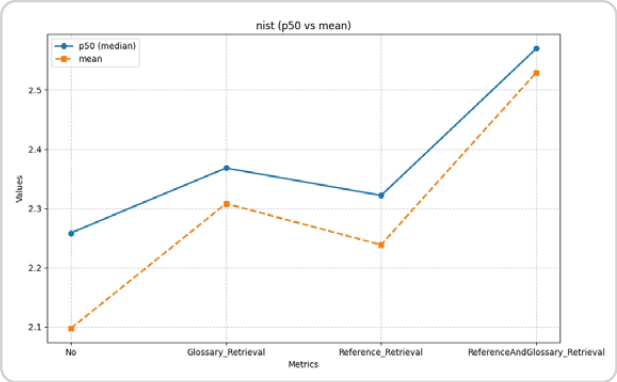


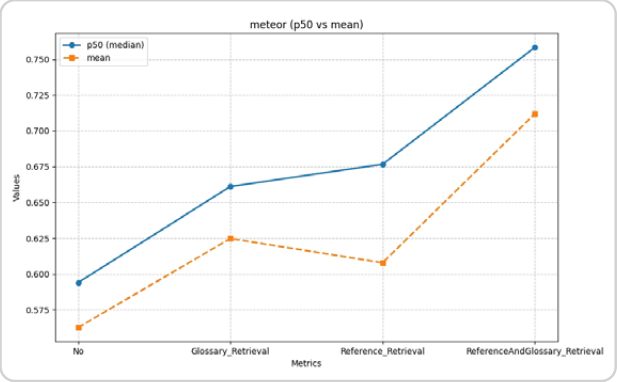


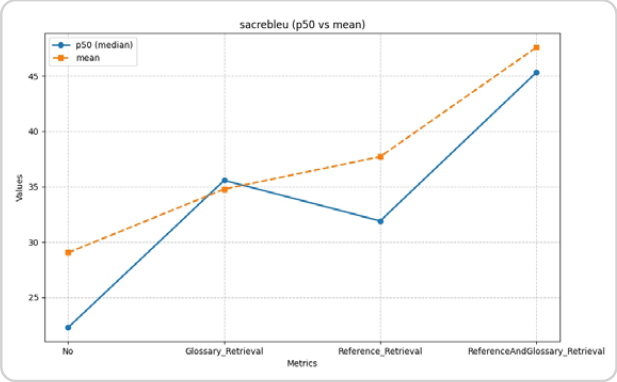


- 电商场景 - Amazon Product Title 数据集 - haiku3









根据上面的实验数据得到 2 个核心观察：

1. ReferenceAndGlossary_Retrieval 这种叠加的方式在四个指标上均取得最高分。

2. Sacrebleu、Meteor 和 Nist 几个指标上，分数上均是 Glossary_Retrieval > Reference_Retrieval

可以做如下实验解读：

Llm_eval 是大模型的评分，更关注翻译风格和流畅性。Sacrebleu、Meteor 和 Nist 是传统 NLP 评价翻译质量的指标，原理上更关注词汇的准确性。其表现说明 Glossary_Retrieval 更能改善词汇的正确性，Reference_Retrieval 更多是改善非统计的一些指标，如翻译风格和流畅性等。两者叠加的方式 ReferenceAndGlossary_Retrieval 综合来看，能结合两者优点取得最佳效果。

所以如果仅仅考虑翻译质量，ReferenceAndGlossary_Retrieval 是最优解，如果成本敏感，Reference_Retrieval 引入的额外 Token 成本需要综合考虑，也可以考虑仅仅使用 Glossary_Retrieval。

5. 基于模型微调的 MTQE 优化实验

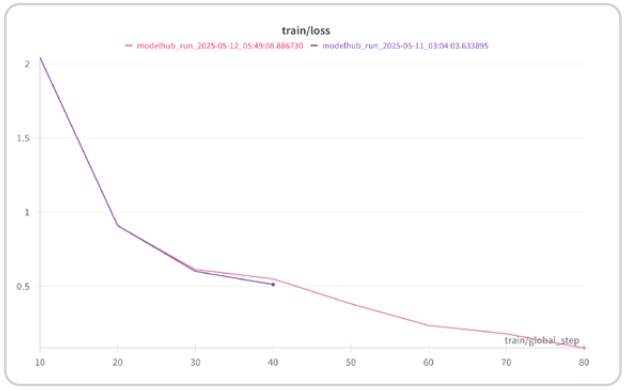
以下是第 4 章的 3.2 部分实验结果 , 采用的数据集为 McAuley-Lab/Amazon-Reviews-2023(亚马逊商品评论数据) 中的商品标题，以 Claude3.5 Sonnet 的翻译结果为 Ground Truth, 以 Amazon Nova Lite 的翻译结果作为待评估数据。

训练的 Base 模型采用了 Qwen-8B-Instruct，以第四章 3.2.2 小节中的错误 1-6 作为识别目标。配合数据合成，得到如下的数据，其中每条错误编号的分数分布为 [0, 5]，业务上以 [0,1,2] 为负例，[3,4,5] 为正例，正负例作了均衡化处理。

错误编号	错误类型	训练集条数	测试集条数
1	拒绝翻译敏感内容	320	80
2	出现非目标语种词汇	319	80
3	出现不相关或者无意义的重复词汇	320	80
4	拼写 / 语法 / 符号错误	104	26
5	数量 / 单位 / 量词错误	274	70
6	格式变化，如添加编号	136	34

5.1 基于 Supervised Full Finetune 的翻译错误识别模型

基于以上数据进行 Full Finetune 训练，训练了 2 个 Epoch 和 4 个 Epoch。



通过 Train/Loss 和 Eval/Loss 得到结论，后续的 2 个 Epoch 导致了过拟合。主要因为问题难度不高，且问题较为低频，导致采集不到足够丰富和高多样性的训练数据。

进行定量的评估后得到如下准召指标：

Metric	错误 1	错误 2	错误 3	错误 4	错误 5	错误 6	总计
精准率	0.683	0.861	0.886	0.733	0.863	0.923	0.802
召回率	0.7	0.775	0.975	0.846	0.828	0.705	0.811

其中错误 1 的精准率和召回率都明显偏低，拉低了整体水平。通过查验 Bad Case 发现，第四章 3.2.2 中的例子是典型情况，分析无误，但是打分在 2 和 3 之间随机波动。

5.2 基于 GRPO 的模型优化

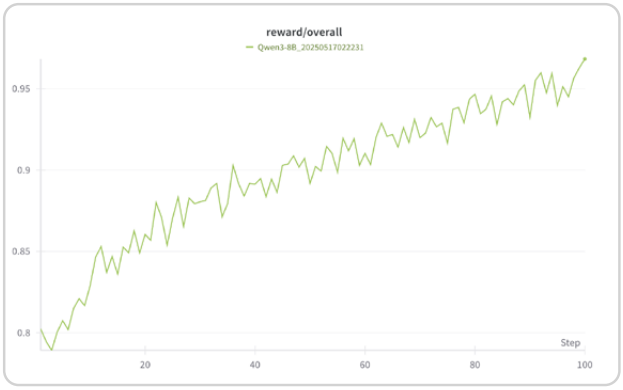
针对上一小节中的问题，除了进一步提高数据规模和质量，还可以通过 GRPO 的 Rule-Based Reward 来针对这个问题进行建模。

本文构建了下面的 Reward_Function, 从模型输出中抽取的得分数组，直接根据分数在业务上的正负来为模型训练提供 reward。

代码块

```
def accuracy_reward(predict: str, ground_truth: str) -> float:
    reward = 0.0
    try:
        predict_scores = json.loads(predict.split("my ratings is ")[-1])
        ground_truth_scores = json.loads(ground_truth.split("my ratings is ")[-1])
        gt_score = min(ground_truth_scores)
        gt_cate_id = ground_truth_scores.index(min(ground_truth_scores))
        pred_score = predict_scores[gt_cate_id]
        if (gt_score <=2.0 and pred_score <=2.0) or (gt_score >= 3.0 and pred_score >= 3.0):
            reward = 1.0
        else:
            reward = 0.0
    except Exception as e:
        print(f"Exception of accuracy_reward: {e}")
    return reward
```

下面两图为 GRPO 训练时准确率的 Reward 曲线，左图训练集上经过 100 轮训练，从 0.8 左右提高到了 0.95 以上； 右图测试集也呈单调上升，从 0.8 左右提升到了 0.89 左右。



对第四章 3.2.2 中的 Bad Case 再次进行推理，得到优化后的答案，满足了业务上的诉求。

代码块

```
{
    "src": "MCLVN",
    "translation": "无法翻译\"MCLVN\"，这似乎是一个不明确的缩写或产品代码。需要更多信息才能进行正确翻译。",
    "label": "<think>The translation avoids translating the term by claiming it's unclear, violating the first criterion that requires an attempt at translation. The explanation is reasonable but doesn't fulfill the requirement to attempt translation.</think>, so my ratings is [2, 5, 5, 5, 5, 5]",
    "predition": "<think>The translator has declined to translate the source text by explaining it lacks clarity, which fails to meet the requirement to attempt a translation. The response provides analysis but doesn't attempt to translate the given text.</think>, my ratings is [1, 5, 5, 5, 5, 5]"
}
```

根据实验得到结论，如果想对微调大模型生成内容的提高其业务判别能力，通过 GRPO 的 Rule-based Reward 是一种有效方案。

06

其他工程实践经验

六 其他工程实践经验

1. 翻译费用优化

1.1 极简 Prompt + Prefill 技巧

对于短文本翻译，过长的翻译 Prompt 会带来额外费用消耗，在成本特别敏感的场景中，比如 IM 即时聊天场景，待翻译的文字一般比较短，我们可以通过极简的 Prompt 和 Prefill 技巧来尽可能的优化成本。参考下面提示词模版，仅仅引入了 40 个额外 Tokens。

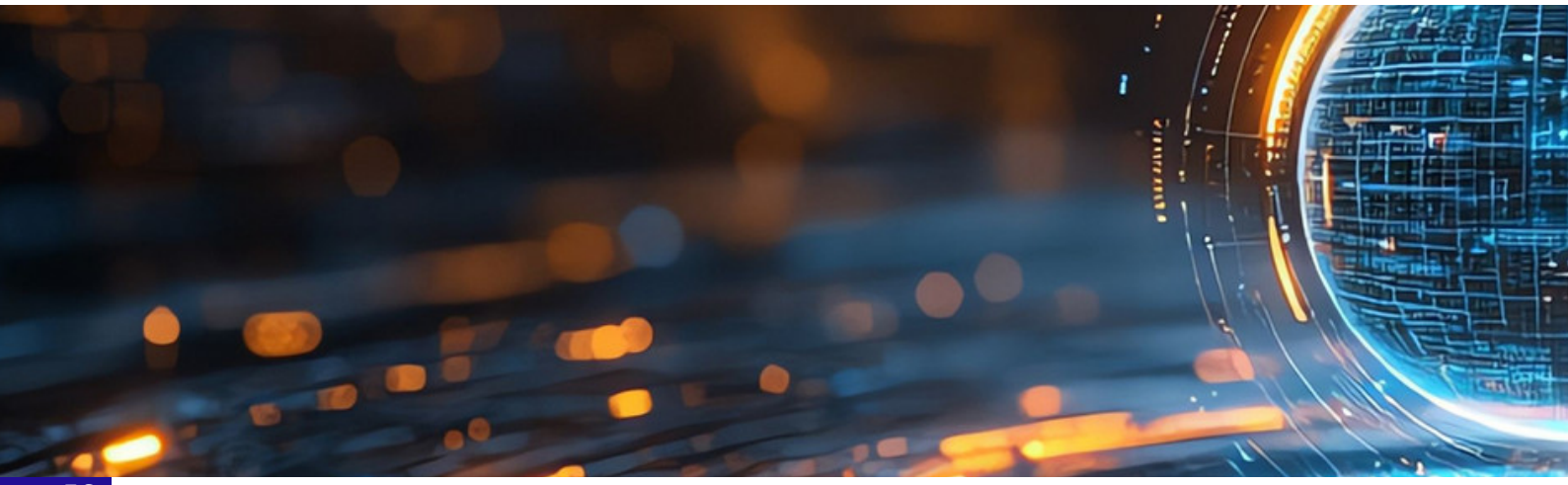
代码块

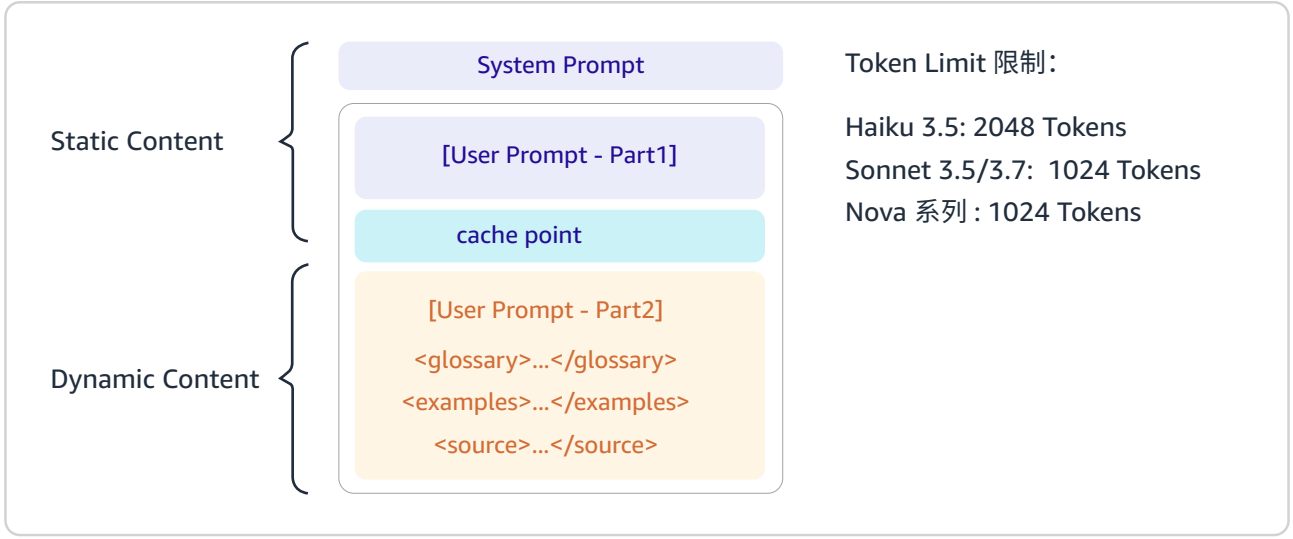
```
1  Human: {source_lang}: `{content}` to same meaning conversational {dist_lang}. Translate within
   <translate></translate>, no explanation.
2  Assistant: Sure, <translate>
```

1.2 Prompt Caching

它是连续多个请求（相同 Input Prompt 的前缀) 之间的缓存复用。缓存键是通过对 Prompt 进行加密哈希生成的，直到缓存控制点。这意味着只有具有相同提示符的请求才能访问特定的缓存。Bedrock 上支持 Explicit Prompt Cache(EPC)，这里指需要用户手动设置缓存的位置 (Cache Point)。

对于本文提到的翻译优化方法，Prompt 中除了原文，其他 Retrieval 的一些上下文也都是动态变化的内容，不适用于 Prompt Cache， 在实现时需要把 Prompt 中的静态内容全部置于前半部分，Cache Point 前达到了一定量的 token 后，Prompt Cache 才能生效。如下图所示，



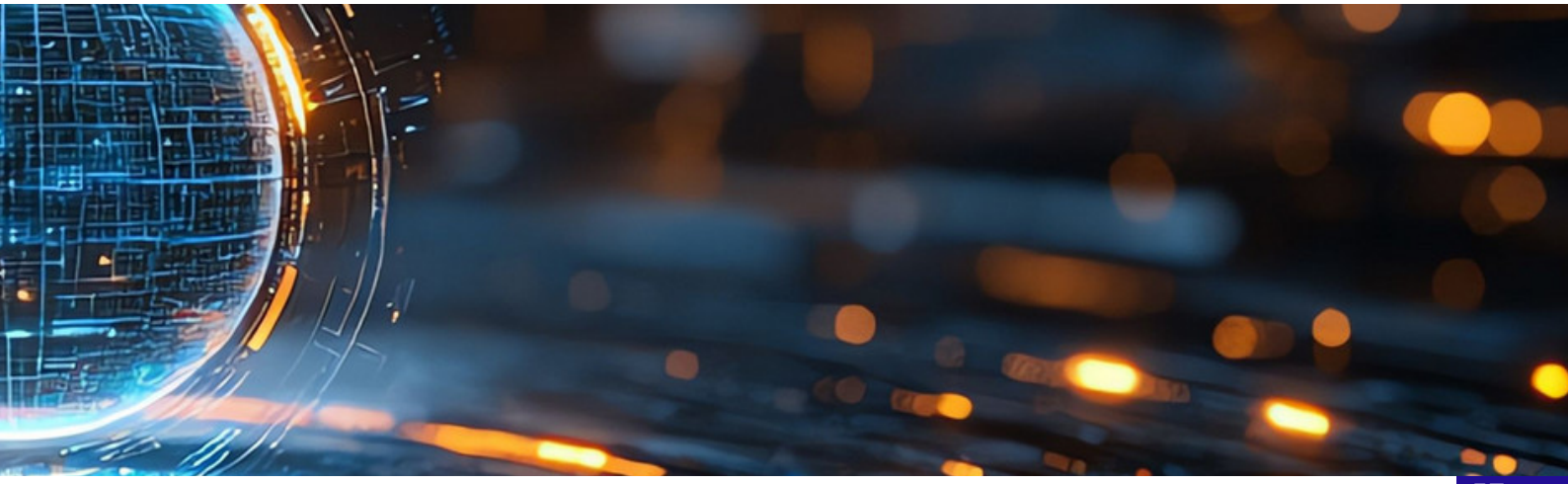


激活了 Prompt Caching 以后，对于翻译中的长提示词会有比较明显的费用节省，以 Claude 3.7 Sonnet 为例，这部分费用下降为之前的 1/10。更详细信息参考官方文档。

1.3 Batch Inference

对于离线的翻译任务，Batch Inference 也是一种有效降低成本的方式，它利用的 Bedrock 背后资源池的闲时资源，价格是 OnDemand 调用方式的 50%。

Batch Inference 以离线 Job 的方式进行管理和追踪，输入和输出均为 S3 路径。一般来说，Batch Inference 的 Job 大部分情况会在 24h 内结束，但是由于有些热门模型导致的资源紧张，会放大任务完成的不确定性。建议通过程序监控 Job 的状态，可以等待 N 个小时，如果 N 小时内 Job 没有启动，再回退到 OnDemand 方式完成翻译。



2. No Translation Tag 支持

在真实的业务场景中，翻译内容可能包含一些无需翻译需要保留原语种的内容，这些内容往往被一些富文本标签所包裹，或者以类似的方式标记出来。如下面这个例子， 其中 Span 中 Class 为 "Notranslate" 中的内容，需要保留原文，这个是某个游戏社区活动的主题：

代码块

```
1    ··· <span class="notranslate">Two Worlds Aflame, the Crimson Night Fades</span> ···
```

为了避免翻译这类文字，可以通过 Prompt 提示大模型，注意这类 No Translation Tag，不要翻译其内容，但 LLM 不能 100% 完全遵从该指令。特别是对于能力欠佳的模型，对这类指令的遵从能力往往不足，可能会翻译，也可能把这段奇怪的文本直接移除。

经过多次试验，依照下面的处理步骤取得比较稳定的优化效果，且可以最大程度的保留上下文完整。

- 把 No Translation Tag 转换成 Markdown 的图片形式

代码块

```
… <span class="notranslate">Two Worlds Aflame, the Crimson Night Fades</span> …

替换为

![Two Worlds Aflame, the Crimson Night Fades](icon.png)
```

- Prompt 中强调 LLM 不要翻译图片的名称
- 翻译完成后，再按照顺序关系把 Markdown 图片替换为原始的内容

3. 文件翻译支持

文件翻译的主要挑战在于文件的格式的解析和保持。对于不同的文件类型，其处理难度也不一样。以 Word 文档举例：

1. 研究方案

表 1　TEST-COMP 在 MV-4-11 皮下瘤模型中抗肿瘤药效实验方案总结

组别	给药剂量 (mg/kg)	溶剂	给药方案
溶媒	0	0.5% CMC-Na (pH=2.1)	PO, QD×28
SNDX-5613	25		PO, QD×28
	50		PO, QD×28
TEST-COMP	5		PO, QD×28
	10		PO, QD×22
	25		PO, QD×22

溶媒组动物给予溶剂CMC-Na (pH=2.1) 作为对照；PO: 经口灌胃；QD：每天一次。

2. 研究方法与材料

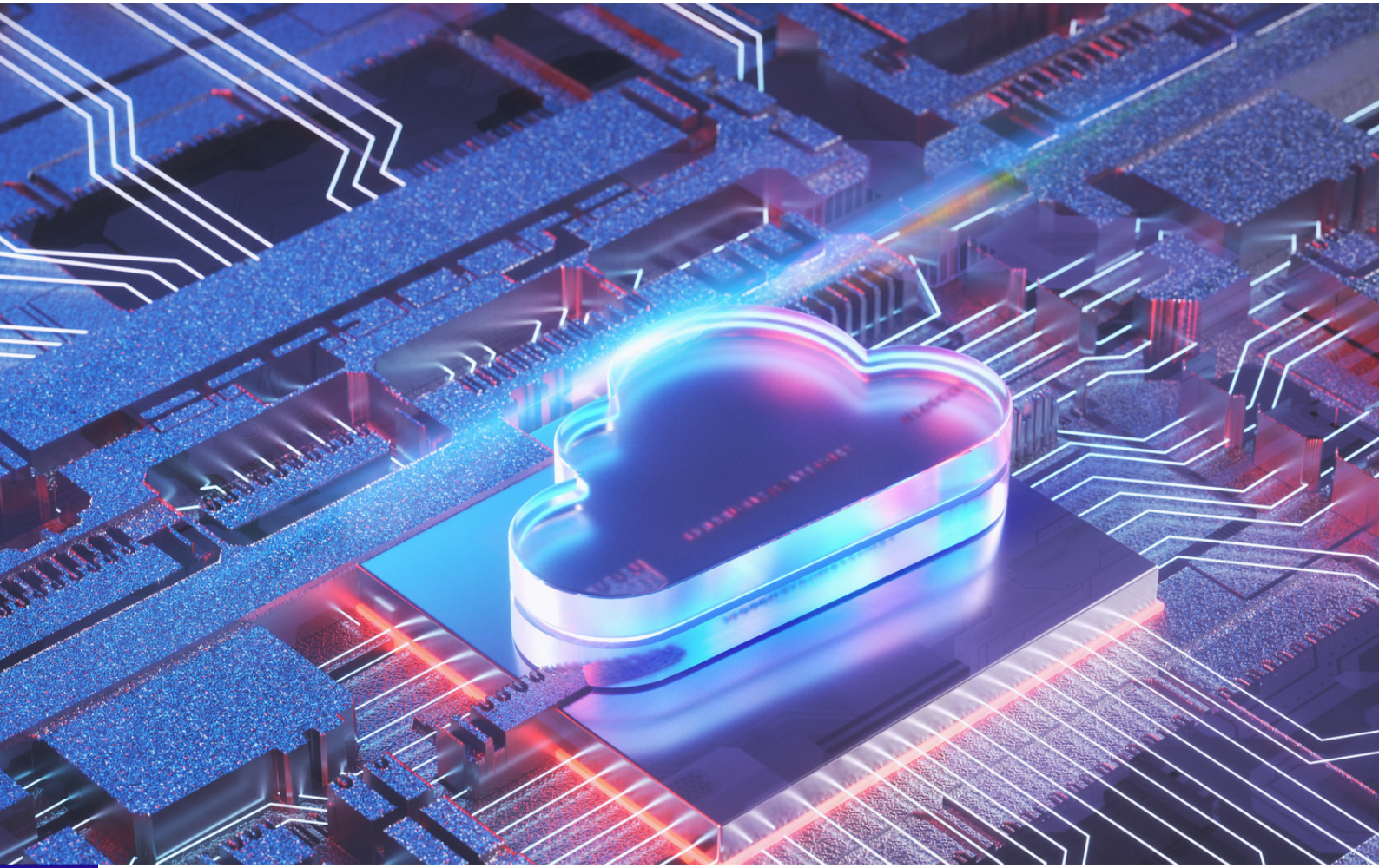
2.1.受试样品及配制方法

TEST-COMP游离碱（批号为S04013-067-01，其纯度为100%）由和记黄埔医药（上海）有限公司化学部提供。给药制剂配制方法为：称取适量TEST-COMP置于无菌离心管中，加入一定量的0.5%酸化羧甲基纤维素钠溶剂（0.5%CMC-Na，pH=2.1），涡旋1~3分钟，在59 kHz下超声30分钟得到浓度为0.5、1以及2.5 mg/mL的均一溶液，用作实验动物给药，给药体积为10 mL/kg。

保留原文的格式我们主要是通过识别 Word 里面不同文字的 Element，并按顺序 Append 在一个列表里，然后进行翻译，最后翻译后的文字逐个 Replace 文档对应位置中，实现格式保留。Word 文档相对容易达到比较好的效果，不太需要考虑太多额外排版的问题。

但 PDF 文档的难度会大很多，因为 PDF 的文字是基于空间位置的，需要通过文档内的坐标来保持文件的格式， PDF 的元素更为细碎，一段文本位于多行，表现为多个元素，需要合并起来再翻译，无法按照元素级别进行翻译。不同语言的长度不同，导致可能存在格式上的变形，或者文本出现重叠。需要自行计算文字渲染后的宽度，然后结合坐标轴定位来进行换行操作。

鉴于上面阐述的文件翻译的复杂度，建议尽可能把文件转化到 Word 格式进行翻译，或者采用一些方案 (Medical-Insight-Hub 方案中提供文件翻译的功能) 而不是自行从零进行实现。



07

总结

七 总结

综合前六个章节所论述的各种经验，从业务角度出发，可以提供以下总结和建议：

质量优先型业务场景（如游戏剧情、品牌广告词、法律合规文件）：

- 优先选择更强大的基础模型能力，如 Claude 3.7 Sonnet 等高性能模型
- 采用 ReferenceAndGlossary_Retrieval 的组合方案，同时获取术语准确性和风格一致性
- 引入 MTQE 机制，以人工审校作为最后兜底，特别是对关键内容（如游戏世界观元素、品牌核心信息）

成本敏感型业务场景（如 UGC 内容、社区评论）：

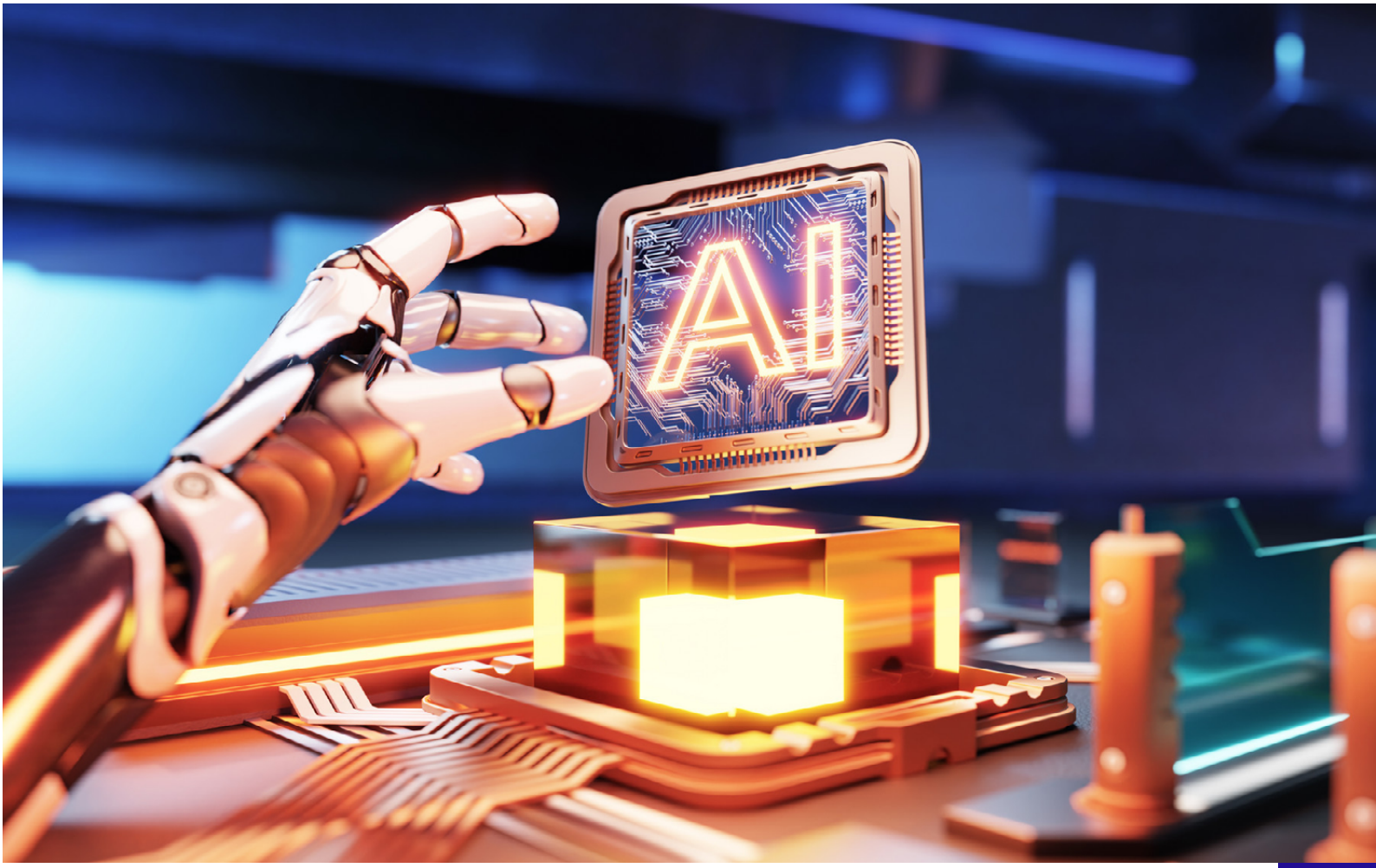
- 使用小型模型配合 Glossary_Retrieval 术语召回，在保证关键词准确的同时控制成本
- 优化 Prompt 设计（极简 Prompt + Prefill 技巧）减少无效 token 消耗
- 充分利用 Prompt Caching 和 Batch Inference 降低运营成本
- 设置简化版的 MTQE 评估机制，重点关注拒翻、语种夹杂等高频错误类型

实时性要求高的业务场景（如游戏内即时聊天、IM 交流）：

- 采用小模型 + 极简 Prompt 的方案，确保响应速度
- 重点优化工程端的并发和稳定性，而非追求极致翻译质量
- 对于业务属性强的领域，比如游戏术语多的场景，仍建议采用 Glossary_Retrieval 提升术语召回

下面是优化方法的业务价值对比：

优化方法	翻译质量提升	适用业务场景	实施难度
更强基础模型	高	质量优先型	低
Prompt 优化	有限	成本敏感型	低
Reference 召回	中 - 高 (风格)	追求风格一致性	中
Glossary 召回	中 - 高 (术语)	术语精准度要求高	偏高
MTQE 机制	提供保证	特定领域长期应用	高



八 参与人

编撰作者（首字母排序）：

韩医徽　李元博　王泽耀　杨俊

审阅指导（首字母排序）：

高寅敬　郭韧　黄卓斌　李艺明　刘力力　王晓野　薛军

特别鸣谢：

所有经验贡献者，本文关于翻译的研究总结，源引了于多个内部团队的工作经验：

夏宁 — 关于利用翻译专家的不同语言语言规则的 Prompt 调优的实践经验；

胡益恺 — 关于广告词翻译场景中的一些实践技巧；

韩医徽，王泽耀，李元博 — 在关于使用样例召回和专词召回翻译的实践经验；

杨俊，朱子琦 — 关于利用 Llm-as-a-judege 构建翻译 MTQE 的实践经验；

王泽耀，李元博 — 关于采用模型微调识别翻译错误的实践经验；

董孝群 — 关于翻译评估指标评价能力的调研经验；

谢川 — 提供关于微调平台 Model Hub 技术支撑；

陈逸斐 — 在 IM 翻译中，关于在翻译费用优化方面的一些总结；

Medical-Insight-Hub 方案团队 — 在文件翻译方面的经验总结。

九 附录

1. 参考文献

a. *Scientific Credibility of Machine Translation Research: A Meta-Evaluation of 769 Papers (98.9)*

b. *Findings of the WMT24 General Machine Translation Shared Task: The LLM Era is Here but MT is Not Solved Yet*

c. Amazon Blog - LLM 微调实践 – 利用大语言模型微调进行翻译质量检测（上）

d. Amazon Blog - LLM 微调实践 – 利用大语言模型微调进行翻译质量检测（下）

e. Amazon Blog - 基于亚马逊云科技服务实现具备专词映射能力的大语言模型翻译

f. Amazon Blog - 集成 Dify 和亚马逊云科技实现更具灵活性的翻译工作流

2. 翻译实验数据

a. 游戏 - 原神英中日译文（引用 Kaggle 公开数据） - genshin-impact-ja-zh

b. 游戏 - 原神词典（引用 Kaggle 公开数据） - genshin-dictionary

c. 电商 - Amazon 商品标题 - McAuley-Lab/Amazon-Reviews-2023

d. 其他 - 英韩对照翻译数据 - Korean - English Parallel Corpus

3. 相关技术资产

a. 专词翻译方案 - https://github.com/aws-samples/rag-based-translation-with-dynamodb-and-bedrock

b. Dify On Amazon 插件 - https://github.com/aws-samples/dify-aws-tool/

c. LLM 微调方案 - Model Hub(Llamafactory on SageMaker) - https://github.com/aws-samples/llm_model_hub

d. 智能医学内容生成中心解决方案指南 - https://www.amazonaws.cn/en/solutions/industry/health/medical-insights-hub/

e. 实验参考代码 - https://github.com/ybalbert001/llm_translation_research/tree/main

亚马逊云科技

前述特定亚马逊云科技生成式人工智能相关的服务目前在亚马逊云科技海外区域可用。亚马逊云科技中国区域相关云服务由西云数据和光环新网运营，具体信息以中国区域官网为准。