

DeRAGEC: 使用合成理由的 ASR 错误纠正来降噪命名实体候选

Solee Im^{*1}, Wonjun Lee^{*2}, Jinmyeong An^{1,3},
Yunsu Kim⁴, Jungseul Ok^{1,2}, Gary Geunbae Lee^{1,2}

¹Graduate School of Artificial Intelligence, POSTECH, Republic of Korea

²Department of Computer Science and Engineering, POSTECH, Republic of Korea

³Mobile eXperience Business, Samsung Electronics, Republic of Korea

⁴aiXplain Inc., Los Gatos, CA, USA

{ solee0022, lee1jun, jungseul.ok, gblee } @postech.ac.kr, jinmyeong.an@samsung.com, yunsu.kim@aixplain.com

Abstract

我们提出了 DeRAGEC，一种用于改进自动语音识别 (ASR) 系统中命名实体 (NE) 纠正的方法。通过扩展检索增强生成式错误纠正 (RAGEC) 框架，DeRAGEC 使用合成降噪理据在纠正之前过滤掉噪声的 NE 候选。在不需额外训练的情况下，它通过上下文学习利用语音相似性和增强定义来改进噪声的检索 NE。在 CommonVoice 和 STOP 数据集上的实验结果显示，DeRAGEC 在词错误率 (WER) 和 NE 命中率上显著优于基线 ASR 和 RAGEC 方法。具体来说，我们的 WER 相较于未经过后处理的 ASR 减少了 28 %。我们的源代码公开于：<https://github.com/solee0022/deragec>。

1 介绍

最近的研究 (Ma et al., 2023; Li et al., 2024; Chen et al., 2023) 表明，使用大型语言模型 (LLMs) 对语音识别转录进行后处理可以显著提高自动语音识别 (ASR) 的准确性。特别地，人们提出了生成性错误校正 (GEC) 来优化 ASR 输出 (Yang et al., 2023; Chen et al., 2023)。此外，最近的工作已将 GEC 发展为一个多模态范式 (Radhakrishnan et al., 2023; Hu et al., 2024b)，该范式同时使用文本转录和对应音频以进一步提高校正质量。在 GEC 框架下，ASR 模型首先处理用户的语音，并通过束搜索解码输出多个假设 (前 k 个)。然后 LLM 对这些假设进行优化，或者通过不同假设组合重新排列单词，或者预测上下文合适的替换词。这种方法利用了预训练 LLMs 的广泛语言知识和强大生成能力，后者是在大型文本库上进行训练的 (Brown et al., 2020)。

尽管在提高 ASR 准确性方面非常有效，但 GEC 面临一个关键限制：它无法可靠地引入在初始假设中缺失的单词，尤其是命名实体 (NEs) (Wang et al., 2024; Pusateri et al., 2024)。这一挑战源于大语言模型 (LLMs) 的固有偏见，由于它们接触到了大量多样化的语料，倾

向于偏好高频词汇和表达 (Gong et al., 2024; Gallegos et al., 2024)。因此，罕见或不在词汇中的 NEs——特别是那些缺失在 ASR 假设中的——很难以高准确性恢复 (Ghosh et al., 2024; Pusateri et al., 2024)。对此，检索增强的方法 (Wang et al., 2024; Ghosh et al., 2024) 最近获得了关注，作为一种通过引入外部知识来实现更精确 NE 校正的方法。一个显著的例子是检索增强生成性错误校正 (RAGEC)，它通过从外部知识库中检索相关的 NEs 并将其整合到 LLM 的输入上下文中来增强 GEC (Pusateri et al., 2024)。然而，无关或弱相关的 NEs 的存在往往会引入噪声，从而削弱校正性能 (Wang et al., 2024; Ghosh et al., 2024)。目前的解决方案通常隐式处理这个噪声，严重依赖于 LLM 的内部 GEC 能力。

在这项工作中，我们提出了一种显式的方法，用于去噪命名实体 (NE) 候选项，该方法在 GEC 处理之前过滤掉不相关的命名实体。我们的方法结合了语音分数、增强的命名实体定义和合成理由，在上下文学习 (ICL) 框架内实施。通过直接减少检索到的命名实体候选项中的噪声——无需额外的训练——我们降低了系统对 ASR 输出和 LLM 推理的依赖。我们在 CommonVoice 和 STOP 语音数据集上测试了我们的方法，实现了 % 28 的相对字错误率 (WER) 降低。

2 相关工作

2.1 ASR 中的生成式错误校正

GEC (Chen et al., 2023; Yang et al., 2023; Ma et al., 2023) 已成为 ASR 系统的一种有效后处理方法。然而，GEC 模型在处理新颖或特定领域的命名实体 (Wang et al., 2024; Pusateri et al., 2024) 时遇到了困难。虽然采用检索增强的 GEC (RAGEC) (Lei et al., 2024; Pusateri et al., 2024) 方法通过检索语音上相似的命名实体候选项来改善命名实体纠正，但在确定检索到的最佳命名实体数量时面临挑战，导致语音混淆 (Wang et al., 2024) 或命名实体不足。为了

^{*}Equally contributed

解决这一问题，我们提出了一种具有显式去噪机制的创新、无需训练的 RAGEC 系统，该机制生成理由以筛选和选择检索列表中相关的命名实体。

2.2 检索候选项过滤

RAG (Lewis et al., 2020; Izacard et al., 2023; Guu et al., 2020) 改善了语言模型 (LMs) 在知识密集型任务中的准确性，但通常会检索到噪音数据 (Li et al., 2022; Yoran et al., 2023)。在标准 RAG 系统中，包括 RAGEC 任务 (Pusateri et al., 2024; Ghosh et al., 2024)，检索数据中的噪音去除 (例如命名实体列表) 通常通过训练语言模型去预测正确答案 (例如，转录) 来隐式处理，这样会面临潜在噪音输入。然而，这种方法容易受到高噪音比率的影响，缺乏透明性，并且严重依赖于语言模型 (Cuconasu et al., 2024; Wu et al., 2024)。最近的研究 (Wei et al., 2024) 引入通过生成理由进行显式去噪来筛选噪音，而无需额外训练。然而，在 RAGEC 中去噪检索到的语音信息 (例如命名实体列表) 仍然没有充分探索。为了解决这一空缺，我们提出了 DeRAGEC，一种无需训练、以理由为驱动的方法，通过显式去噪语音候选信息来完善检索到的命名实体列表。

3 DeRAGEC

3.1 初步

我们首先使用预训练的 ASR 模型，通过束搜索生成 5-best 假设集 $H = \{h_1, h_2, h_3, h_4, h_5\}$ 。从顶层假设 h_1 中，一个命名实体识别 (NER) 模型提取 NE 候选，这些候选作为语音查询 q_p ，用于从外部 NE 数据库中检索出前 k 个语音相似的 NE 候选 $N = \{n_1, \dots, n_k\}$ 。

在 ASR 基线中，最高的假设直接用作转录 $\hat{a} = h_1$ (方程 1)。

在 GEC 框架中，语言模型 $\mathcal{M}_\theta$ 使用从数据集 $\mathcal{D}$ 中抽样的 T^{fs} 个少样本例子 $\mathcal{E}^{gec} = \{(H_i, a_i)\}$ 对假设集 H 进行后处理 (公式 2)。其中 $\mathcal{D} = \{(H_i, a_i, N_i), 1 \leq i \leq T\}$ 包含 T 个假设 (H)、真实转录 (a) 和检索到的 NE 候选项 (N) 三元组。

RAGEC 框架通过在生成过程中结合检索到的 NE 候选项 N 来增强这一点，演示 $\mathcal{E}^{ragec} = \{(H_i, a_i, N_i), 1 \leq i \leq T^{fs}\} \in \mathcal{D}$ (方程 3)，其中 T^{fs} 是 ICL 示例的数量。

$$ASR : \hat{a} = h_1 \quad (1)$$

$$GEC : \hat{a} = \mathcal{M}_\theta(a|H, \mathcal{E}^{gec}) \quad (2)$$

$$RAGEC : \hat{a} = \mathcal{M}_\theta(a|H, N, \mathcal{E}^{ragec}) \quad (3)$$

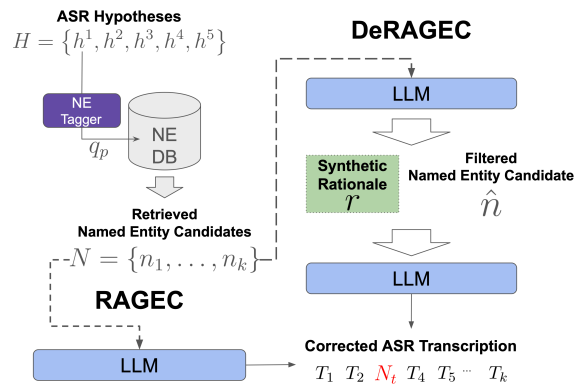


Figure 1: 比较 RAGEC 和 DeRAGEC 在处理检索到的 NE 候选项方面的性能。为了清楚起见，输入特征如 PS 和 Def 被省略，以突出这两种方法之间的关键差异。

3.2 DeRAGEC 框架

DeRAGEC 在 RAGEC 的基础上扩展了一个显式的、无训练的去噪门。对于每个检索到的命名实体 (NE) 候选，我们附加 (i) 一个语音相似度得分 (PS) 和 (ii) 一个一行的维基百科定义 (Def) 到每个检索到的 NE 候选。去噪门会修剪掉不相关的候选，返回所选实体 $\hat{n}$ 及其理由 r 。然后将元组 $(H, \hat{n}, r)$ 传递给 GEC 模型，该模型输出最终的转录 $\hat{a}$ 。图 1 突出了 DeRAGEC 对原始 RAGEC 的扩展。

对于每个候选者 $n_i \in N$ ，我们附加其语音相似度得分 $PS_i = \text{sim}(q_p, n_i)$ 和一个句子的实体定义 Def_i 。结果三元组被序列化为：

$\langle n_i \mid \text{phonetic-score: } PS_i \mid \text{def: } Def_i \rangle$ 并用于去噪门。

合成推理生成： 受到先前工作的启发 (Wei et al., 2024)，我们使用一个单独的理由生成模型 $\mathcal{M}_r$ 来合成去噪理由 r^{syn} ，以指导相关 NE 候选项的选择。如算法 1 所示，对于每个训练三元组 $(h_1, a, N) \in \mathcal{D}$ ，以及 PS 和 Def ，模型生成一个 MCQ 风格的理由 r^{syn} ，受启发于 (Hu et al., 2024a)，解释哪些 NE 候选项对恢复真实值 a 有意义地贡献。通过与从 H 提取的 NE 连接，增强 NE 列表 N ，确保完整性。因此，这允许我们有效地扩充当前数据集 $\mathcal{D} \rightarrow \mathcal{D}^+ = \{(H_i, a_i, N_i, PS_i, Def_i, r_i), 1 \leq i \leq T\}$ 。

学习去噪理由： DeRAGEC 使用 ICL 从这些合成的推理中学习，而不需要进一步训练。给定一个实例 (h_1, N, PS, Def) ，模型采样 T^{fs} 理由增强的例子 $\mathcal{E}^{deragec} \in \mathcal{D}^+$ 并生成：

1. 一个经过过滤的 NE $\hat{n}$ 和理由 r ，描述了选择 $\hat{n}$ 的推理过程

2. 最终的修正抄本 $\hat{a}$, 以 H 、 $\hat{n}$ 和 r 为条件
DeRAGEC :

$$\begin{aligned}\hat{n}, r &\leftarrow \mathcal{M}_\theta(n|(h^1, N, PS, Def), \mathcal{E}^{deragec}) \\ \hat{a} &= \mathcal{M}_\theta(a|H, \hat{n}, r)\end{aligned}\quad (4)$$

有关推理合成和 GEC 过程的详细提示, 请参见附录 G 中的表格 6–9。

Algorithm 1 合成理由

Input: Phonetic Retrieval $\mathcal{R}$, Rationale Generator $\mathcal{M}_r$, Data $\mathcal{D}$, phonetic similarity set PS and textual definitions Def for NE candidates N .

```

1: 合成推理
2: for each  $\{(H, a, N)\} \in \mathcal{D}$  do
3:   Extract named entities (NE) from  $H$  :
4:    $N^{hyp} = \{n_1^{hyp}, \dots, n_5^{hyp}\}$ 
5:   Concatenate  $N^{hyp}$  and  $N$  :
6:    $N \leftarrow N^{hyp} \oplus N$ 
7:   Synthesize denoising rationales:
8:    $r^{syn} \leftarrow \mathcal{M}_r(h_1, a, N, PS, Def)$ 
9: end for

```

4 实验结果

对于我们的评估, 我们使用了 CommonVoice (CV) 数据集的一个子集, 包含来自 (Chen et al., 2023) 的 2000 个样本, 以及一个抽样的 STOP 测试集, 包含 5000 个样本。选择这些数据集是为了评估我们的方法在自由形式语音 (CV) 和语音命令的口语语言理解 (STOP) 上的有效性。我们检索了用于 RAGEC 和 DeRAGEC 的前 10 个在音素上相似的 NE。此外, 从 CV 和 STOP 训练集中随机选择最多 5 个 ($T^{fs} = 5$) 样本作为 ICL 少量样本例子 ($\mathcal{E}$)。

我们使用 Whisper-large-v3-turbo (0.8B) (Radford et al., 2022), 并设置了 5 的束搜索大小, 作为我们的 ASR 模型。对于 GEC, 我们使用 Llama-3.1 (70B) (MetaAI, 2024) 和 GPT-4o-mini (gpt-4o-mini-2024-07-18) (OpenAI, 2023)。此外, 我们使用 o1 (o1-2024-12-17) (Jaech et al., 2024) 进行理由合成。Epitrans (Mortensen et al., 2018) 和 Panphon (Mortensen et al., 2016) 用于发音命名实体 (NEs) 并基于发音特征计算语音相似性 (PS)。具体来说, 语音相似性是在假设中标注的 NE (表示为语音查询 q_p) 与从 NE 数据库检索到的 NEs ($N = \{n_1, \dots, n_k\}$) 之间进行测量的。

从 CV 训练集、一个开源媒体实体数据集 (Van Gysel et al., 2022) 和维基百科中, 总共收集了 3,003,462 个命名实体用于构建命名实体

数据库。在命名实体收集和 GEC 过程中, 我们使用 GliNER-large-v2* 模型作为命名实体标注器。

4.1 DeRAGEC 与基线

为了展示每个特征的效果, 我们对特征 MCQ 的 NE 过滤格式提示、 PS 、 Def 和理由 (Rat) 进行了消融实验, 如表 1 所示, 而 NE 候选项 (N) 则用于 RAGEC 和 DeRAGEC。

与我们的基线进行比较, 包括纯 ASR、GEC 和 RAGEC, DeRAGEC 在 WER 和 NE 命中率方面持续获得最佳性能。NE 命中率是衡量方法在纠正 NEs 方面效果的指标。它是通过将正确识别的 NEs 数量除以最终转录中的 NEs 总数计算得出的。更高的 NE 命中率表示在 NE 纠正方面的更好性能。

从表格 1 中我们观察到, DeRAGEC 比其他方法都有明显的优势, 相对于纯 ASR 基线平均相对 WER 减少了 28%, 相对于最佳 RAGEC 配置减少了 5.9%。即使与 ORACLE 设置进行比较, 在这种设置下提供了真实值的 NE 给 GEC 过程, DeRAGEC 在 WER 和 NE 命中率上显示出一个小的差距, 这表明即使与上界设置相比它仍然有效。

4.2 去噪效果

为了评估去噪的影响, 我们考虑了除了 WER 之外的额外指标来评估过滤效果。图 2 表明我们的方法在提高精度的同时保持了去噪步骤的候选命中率 (召回率), 这表明去噪步骤成功地选择了正确的 NE 并淘汰了不相关的候选项。精度的上界为 0.166 ($1/6$), 假设正确的 NE 总是包含在选定候选项之一 n 中, 以及 ASR 假设中的 5 个现有 NE。虽然我们的方法没有达到这个上界, 但与其他基线相比, 它达到了最高的精度 0.139。此外, 在不应用任何去噪的 N 仅设置中召回上界设为 0.841, 我们的方法保持了 0.839 的召回率, 显示出仅有很小的差距。

5 结论

DeRAGEC 是一种新颖的、无需训练的方法, 通过使用音素相似性、增强的定义和合成的理由来过滤噪声的命名实体 (NE) 候选, 从而增强 ASR 系统中的命名实体校正。在 CommonVoice 和 STOP 数据集上的实验表明, DeRAGEC 的表现优于基线 ASR 和 RAGEC 模型, 取得了相对 28% 的 WER (词错误率) 减少, 并提高了命名实体命中率。这些结果突出了 DeRAGEC 在命名实体校正中的有效性, 即便与 ORACLE

*https://huggingface.co/urchade/gliner_large-v2

	Denoising feature					WER (%) ↓				NE Hit Ratio ↑			
						Llama-3.1 (70B)		gpt-4o-mini		Llama-3.1 (70B)		gpt-4o-mini	
Model	<i>N</i>	<i>MCQ</i>	<i>PS</i>	<i>Def</i>	<i>Rat</i>	CV	STOP	CV	STOP	CV	STOP	CV	STOP
ASR only	-	-	-	-	-	7.7	8.9	7.7	8.9	0.751	0.787	0.751	0.787
GEC	-	-	-	-	-	6.8	7.8	6.9	7.4	0.782	0.805	0.784	0.804
RAGEC (+ Denoising)	✓	-	-	-	-	6.5	6.5	7.1	6.6	0.804	0.807	0.788	0.814
	✓	✓	-	-	-	6.5	6.5	7.1	6.5	0.804	0.807	0.788	0.816
	✓	✓	✓	-	-	6.6	6.0	7.1	6.0	0.796	0.828	0.785	0.827
	✓	✓	-	✓	-	6.5	7.2	7.0	6.9	0.807	0.697	0.795	0.752
	✓	✓	✓	✓	-	6.5	6.2	6.7	6.0	0.807	0.815	0.802	0.827
DeRAGEC (ours)	✓	✓	✓	✓	✓	6.0	5.9	6.2	5.8	0.831	0.838	0.813	0.842
ORACLE	A ground truth of NE					5.8	5.7	6.0	5.7	0.837	0.857	0.828	0.847

Table 1: 在不同的模型和去噪设置下，对 CV 和 STOP 数据集的 WER (%) 和 NE 命中率的比较。对于 RAGEC，仅使用 *N* 表示没有应用额外的去噪，而其他设置则结合特定特征来去噪 NE 候选项。

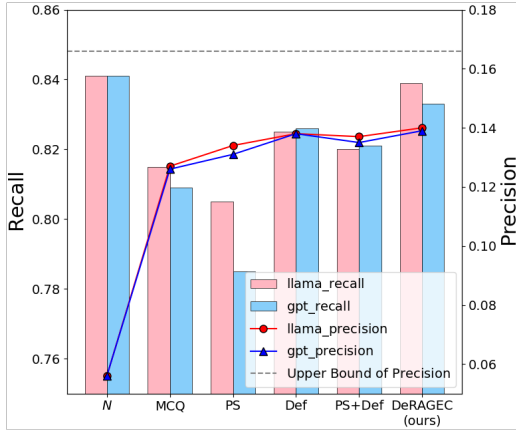


Figure 2: 候选去噪过程的命中率的召回率和精确率。不同的过滤方法在 CV 数据集上进行了评估。

设置相比，DeRAGEC 也表现出色，并且具有更广泛的 ASR 应用潜力。

我们的研究探索了在 RAGEC 任务中使用额外的知识（语音和文本知识）和理由来去噪检索到的命名实体 (NEs)，这是一个去噪仍未得到充分探索的领域。然而，我们的方法存在一些局限性。可能有必要探讨利用我们的合成理由的训练方法。虽然这个过程可能成本高昂并依赖于 ASR 或后处理模型，但需要进一步的研究来评估模型在训练过程中如何适当地吸收和应用提供的去噪指令。此外，可能有必要探索我们的方法是否可以在更广泛的 ASR 和后处理模型中进行泛化，以确保其适应更广泛的应用。展望未来，我们计划探索替代的去噪方法，并提升去噪技术的稳健性，以进一步提高上述 ASR 后处理的效果。

6

致谢

本研究部分得到了韩国政府 (MSIT) 资助的信息通信技术规划与评估研究所 (IITP) 的资助 (No.RS-2019-II191906, 人工智能研究生院项目 (POSTECH)) (5 %), 并得到了韩国政府 (科学与 ICT 部) 资助的信息通信技术规划与评估研究所 (IITP)-信息技术研究中心 (ITRC) 资助 (IITP-2025-RS-2024-00437866) (47.5 %), 以及由韩国国家警察厅 (KNPA) 资助的智能健康护理项目的支持 (No. RS-2022-PT000186) (47.5 %)。

References

- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Chen Chen, Yuchen Hu, Chao-Han Huck Yang, Sabato Marco Siniscalchi, Pin-Yu Chen, and Eng-Siong Chng. 2023. Hyporadise: An open baseline for generative speech recognition with large language models. *Advances in Neural Information Processing Systems*, 36:31665–31688.
- Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. 2024. The power of noise: Redefining retrieval for rag systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 719–729.
- Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. 2024. Bias and fairness in large language models: A survey. *Computational Linguistics*, pages 1–79.
- Sreyan Ghosh, Mohammad Sadegh Rasooli, Michael Levit, Peidong Wang, Jian Xue, Dinesh Manocha,

- and Jinyu Li. 2024. Failing forward: Improving generative error correction for asr with synthetic data and retrieval augmentation. *arXiv preprint arXiv:2410.13198*.
- Xun Gong, Anqi Lv, Zhiming Wang, and Yanmin Qian. 2024. Contextual biasing speech recognition in speech-enhanced large language model. *Proc. Interspeech. ISCA*, pages 257–261.
- Kelvin Guu, Kenton Lee, Zora Tung, Panupong Paspapat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.
- Yuchen Hu, Chen Chen, Chengwei Qin, Qiushi Zhu, Eng Siong Chng, and Ruizhe Li. 2024a. Listen again and choose the right answer: A new paradigm for automatic speech recognition with large language models. *arXiv preprint arXiv:2405.10025*.
- Yuchen Hu, CHEN CHEN, Chao-Han Huck Yang, Ruizhe Li, Chao Zhang, Pin-Yu Chen, and EngSiong Chng. 2024b. Large language models are efficient learners of noise-robust speech recognition. In *The Twelfth International Conference on Learning Representations*.
- Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2023. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251):1–43.
- Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.
- Zhihong Lei, Xingyu Na, Mingbin Xu, Ernest Pusateri, Christophe Van Gysel, Yuanyuan Zhang, Shiyi Han, and Zhen Huang. 2024. Contextualization of asr with llm using phonetic retrieval-based augmentation. *arXiv preprint arXiv:2409.15353*.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.
- Daliang Li, Ankit Singh Rawat, Manzil Zaheer, Xin Wang, Michal Lukasik, Andreas Veit, Felix Yu, and Sanjiv Kumar. 2022. Large language models with controllable working memory. *arXiv preprint arXiv:2211.05110*.
- Sheng Li, Chen Chen, Chin Yuen Kwok, Chenhui Chu, Eng Siong Chng, and Hisashi Kawai. 2024. Investigating asr error correction with large language model and multilingual 1-best hypotheses. In *Proc. Interspeech*, pages 1315–1319.
- Rao Ma, Mengjie Qian, Potsawee Manakul, Mark Gales, and Kate Knill. 2023. Can generative large language models perform asr error correction? *arXiv preprint arXiv:2307.04172*.
- MetaAI. 2024. *The llama 3 herd of models*. *arXiv preprint arXiv:2407.21783*.
- David R. Mortensen, Siddharth Dalmia, and Patrick Littell. 2018. *Epitran: Precision G2P for many languages*. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- David R. Mortensen, Patrick Littell, Akash Bharadwaj, Kartik Goyal, Chris Dyer, and Lori Levin. 2016. *PanPhon: A resource for mapping IPA segments to articulatory feature vectors*. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3475–3484, Osaka, Japan. The COLING 2016 Organizing Committee.
- OpenAI. 2023. *Gpt-4 technical report*. *arXiv preprint arXiv:2303.08774*.
- Ernest Pusateri, Anmol Walia, Anirudh Kashi, Bortik Bandyopadhyay, Nadia Hyder, Sayantan Mahinder, Raviteja Anantha, Daben Liu, and Sashank Gondala. 2024. Retrieval augmented correction of named entity speech recognition errors. *arXiv preprint arXiv:2409.06062*.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. *Robust speech recognition via large-scale weak supervision*. *arXiv preprint*.
- Srijith Radhakrishnan, Chao-Han Yang, Sumeer Khan, Rohit Kumar, Narsis Kiani, David Gomez-Cabrero, and Jesper Tegnér. 2023. Whispering llama: A cross-modal generative error correction framework for speech recognition. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10007–10016.
- Christophe Van Gysel, Mirko Hannemann, Ernest Pusateri, Youssef Oualil, and Ilya Oparin. 2022. Space-efficient representation of entity-centric query language models. In *Proc. Interspeech 2022*, pages 679–683.
- Yi-Cheng Wang, Hsin-Wei Wang, Bi-Cheng Yan, Chi-Han Lin, and Berlin Chen. 2024. Dancer: Entity description augmented named entity corrector for automatic speech recognition. *arXiv preprint arXiv:2403.17645*.
- Zhepei Wei, Wei-Lin Chen, and Yu Meng. 2024. Instructrag: Instructing retrieval-augmented generation with explicit denoising. *arXiv preprint arXiv:2406.13629*.

Kevin Wu, Eric Wu, and James Zou. 2024. How faithful are rag models? quantifying the tug-of-war between rag and llms' internal prior. *arXiv e-prints*, pages arXiv-2404.

Chao-Han Huck Yang, Yile Gu, Yi-Chieh Liu, Shalini Ghosh, Ivan Bulyko, and Andreas Stolcke. 2023. Generative speech recognition error correction with large language models and task-activating prompting. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8. IEEE.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2023. Making retrieval-augmented language models robust to irrelevant context. *arXiv preprint arXiv:2310.01558*.

A

附录

B 静态阈值过滤的效果。

表格 2 和 3 展示了三种现有的方法来减少候选集：在检索过程中减少 K 值（检索到的 NEs 数量， $N = \{n_1, \dots, n_k\}$ ），设置静态距离阈值（ θ_d ），以及应用基于标准差（ STD ）的统计阈值。然而，静态阈值方法，包括 K 和 θ_d ，存在一个局限：即使是正确的候选也可能被排除，从而导致召回率的下降。

我们没有为所有情况使用固定的常量阈值（ K, θ_d ），而是进行了一项实验，使用 STD 设置更具样本特异性的阈值。统计过滤的工作原理如下：当候选者的距离 STD 较大时（表示一个倾斜的分布），经过过滤后剩下的候选者较少，具体来说，只保留那些位于平均值的 $\sigma * STD$ 以内的候选者。相反，当 STD 较小时（表明更均匀的分布），则保留更多的候选者。然而，统计阈值方法在召回率低的情况下表现不佳。

在 4 表中，我们展示了在真实检索情况下和一个上界检索场景下的去噪效果，其中正确答案 NE 被无条件检索到。在这个实验中，我们使用了 CommonVoice 数据集和 Llama-v3.1。在这个设置中，ORACLE-retr 表示一个情境，其中实际检索到的 NE（召回率 @10 为 0.841）被输入到 GEC 中，并去除了噪声 NE。ORACLE 表示一个场景，其中仅向 GEC 提供正确答案 NE，而 ORACLE+noise 则包括语音上相似的噪声 NE。这个设置使我们能够观察到 NE 噪声对 GEC 造成的性能下降。这些结果表明，将我们的方法应用于各种检索场景可以带来性能改进。

C 不使用 MCQ 的 DeRAGEC 性能。

我们试图将所有过程、推理、选择 NE 以及进行错误纠正进行串联。然而，如表 5 所示，缺少 MCQ 的 DeRAGEC 表现出性能下降。当 DeRAGEC 作为单一步骤执行时，生成的推理平均有 583 个词，这太长了，使得过程过于复杂。相反，MCQ 步骤生成的推理平均为 211 个词，并且表现更好。因此，我们的实验表明，将 NE 选择和错误纠正的步骤分开，比将它们合并成一个推理步骤表现得更好。

D 案例研究

图 3 和 4 对比了在 RAGEC 和 DeRAGEC 的 MCQ 步骤中生成的答案。RAGEC 只生成 NE，而 DeRAGEC 首先生成一个理由来选择 NE，

Cloze sentence: The party is led by [Blank]

Options:

A: Samraj Singh (1.00 | Maharaja (also spelled Maharajah or Maharaj; lit. 'great ruler'; feminine: Maharani) is a South Asian royal title of Sanskrit origin.)

B: Amrit Singh (0.70 | Amrit Singh may refer to: Amrit Singh (cyclist) (born 1991), Indian track cyclist Amrit Singh, daughter of former Indian Prime)

C: Samurai Gun (0.64 | Samurai Gun (Japanese: サムライガン, Hepburn: Samurai Gan) is a Japanese manga series written and illustrated by Kazuhiro Kumagai.)

D: Samresh Singh (0.64 | Samresh Singh (1940/1941 – 1 December 2022) was an Indian politician from the state of Jharkhand.)

E: Yuvraj Singh (0.64 | Yuvraj Singh (born 12 December 1981) is a former Indian international cricketer who played in all formats of the game.)

F: samurai prince (0.62 | Samurai Jack is an American science fantasy action-adventure animated television series created by Genndy Tartakovsky for Cartoon Network and Adult)

G: Cambyses (0.60 | Cambyses II (Old Persian: romanized: Kam̥būjiya) was the second King of Kings of the Achaemenid Empire from 530)

H: Sam Rice (0.60 | Edgar Charles "Sam" Rice (February 20, 1890 – October 13, 1974) was an American pitcher and outfielder in Major League)

I: Preissing (0.60 | Thomas Joseph Preissing (born December 3, 1978) is an American former professional ice hockey player who played six seasons)

J: Samboy Lim (0.60 | Avelino "Samboy" Borromeo Lim Jr. (April 1, 1962 – December 23, 2023), nicknamed "The Skywalker", was a Filipino professional basketball)

K: Sandeep Singh (0.60 | Sandeep Singh (born 27 February 1986) is an Indian professional field hockey player from Haryana and an ex-captain)

RAGEC

Answer: A: samraj singh ❌

DeRAGEC

<think>

The sentence is "The party is led by [BLANK]." We need to identify the most suitable named entity to fill the blank.

- Option A (Samraj Singh) has the highest phonetic similarity score of 1.00. However, the definition provided does not directly relate to a person but rather to a royal title.
- Options B (Amrit Singh), C (Samurai Gun), D (Samresh Singh), and E (Yuvraj Singh) all have lower phonetic similarity scores but are actual names of individuals, which could fit the context of leading a party.
- Among these, D (Samresh Singh) stands out because the definition mentions that Samresh Singh was a politician from the state of Jharkhand, indicating that he could be associated with leading a political party.

Hence, D: Samresh Singh is the most appropriate choice given the context of the sentence and the definitions provided. </think>

<answer> D: Samresh Singh </answer> ✅

Figure 3: 在 CommonVoice 数据集中对 REGEC 和 DeRAGEC 的案例研究。去除不相关 NE 候选项的过程用粉色突出显示，而正确 NE 的选择则用蓝色表示。

Cloze sentence: How many miles from [Blank] to Brinkley, Arkansas

Options:

A: Kentucky Tennessee (1.00 | The Kentucky–Tennessee football rivalry is an American college football rivalry between the Kentucky Wildcats and Tennessee Volunteers.)

B: Kenton Tennessee (0.91 | Kenton is a town in Gibson and Obion counties, Tennessee.)

C: Memphis Tennessee (0.69 | Memphis is a city in the U.S. state of Tennessee. It is the county seat of Shelby County)

D: Phantom Menace (0.69 | I Want to Die in New Orleans is the debut studio album by American hip hop duo Suicideboys)

E: Captain Kennedy (0.69 | Captain Kennedy may refer to: Archibald Kennedy, 11th Earl of Cassilis (bef. 1736–died 1794), Scottish peer who lived in the)

F: Austin Hennessey (0.69 | Hennessey Performance Engineering (HPE) is an American hypercar manufacturer and high-performance vehicle creator. In addition to building the Venom F5)

G: Kendra Hennessy (0.69 | Linnie Greene (born May 21, 1954) is an actress, writer, director and producer in the television industry.)

H: Launch Tennessee (0.69 | Tennessee (, locally), is a landlocked state in the Southeastern region of the United States.)

I: Kieran Hennessey (0.69 | The 1957 Croke Cup was the sixth staging of the Croke Cup since its establishment by the Gaelic Athletic Association)

J: Kenton Kelley (0.69 | Stan Kenton Presents Gabe Baltazar is an album by Gabe Baltazar. It was the last recording by Stan Kenton's Creative)

K: Twenty-Ten Tennessee (0.65 | Tennessee (, locally), is a landlocked state in the Southeastern region of the United States.)

RAGEC

Answer: **A: Kentucky Tennessee**



DeRAGEC

<think> The sentence "how many miles from [Blank] to Brinkley, Arkansas" is asking for a location that can be measured in distance to Brinkley. Among the options provided, "Kenton, Tennessee" (Option B) is the only recognized geographical location that fits this context. \n\n1. **Phonetic Match**: "Kenton Tennessee" has a high phonetic score of 0.90. \n\n2. **Geographical Relevance**: Kenton, Tennessee is a real place, making it a suitable candidate for the blank, as it can be measured in miles to Brinkley, Arkansas. \n\n3. **Other Options**: The other options either do not refer to geographical locations or are not relevant to the context of measuring distance. \n\nGiven this reasoning, the optimal choice to fill in the blank is "Kenton, Tennessee."

<answer> **B: Kenton Tennessee** </answer>



Figure 4: 在 STOP 数据集上进行 REGEC 和 DeRAGEC 的案例研究。选择正确 NE 的过程以蓝色突出显示。

		Recall	Precision	WER
TopK	1	0.823	0.137	7.2
	5	0.835	0.084	6.6
	10	0.841	0.056	6.5
	15	0.843	0.032	6.7
PS (θ_d)	0.6	0.697	0.078	6.5
	0.7	0.457	0.117	6.6
	0.8	0.251	0.147	6.9
	0.9	0.126	0.163	7.2
STD (σ)	0.5	0.830	0.124	7.0
	1	0.827	0.131	7.3
	1.5	0.822	0.137	7.4
	2	0.815	0.138	7.4

Table 2: 静态语音过滤对 CommonVoice 中使用 Llama-v3.1 的 WER 的影响。

		Recall	Precision	WER
TopK	1	0.887	0.148	6.7
	5	0.895	0.089	6.7
	10	0.896	0.060	6.5
	15	0.897	0.043	6.6
PS (θ_d)	0.6	0.857	0.065	6.5
	0.7	0.549	0.105	6.7
	0.8	0.244	0.143	6.7
	0.9	0.035	0.165	7.2
STD (σ)	0.5	0.841	0.107	6.8
	1	0.836	0.124	7.5
	1.5	0.822	0.134	7.6
	2	0.821	0.146	7.6

Table 3: 在 STOP 时，Llama-v3.1 中静态语音过滤对 WER 的影响。

	Recall	WER	NE hit ratio
RAGEC	0.841	6.5	0.804
DeRAGEC		6.0	0.831
ORACLE-retr		5.8	0.837
ORACLE	1.000	4.1	0.922
ORACLE+noise		5.4	0.887

Table 4: 在 CommonVoice 上使用 Llama-v3.1 进行不同检索设置的 ORACLE 比较。

	RAGEC	DeRAGEC w/o MCQ	DeRAGEC
WER	6.5	6.8	6.0
NE hit ratio	0.804	0.780	0.831

Table 5: 在 CommonVoice 上使用 Llama-v3.1 的 DeRAGEC 不包含 MCQ 步骤。

然后生成 NE 本身。这项研究表明，我们的模型有效地利用 *PS*, *Def* 和其推理技能来消除无关的 NEs，这包括识别相关信息、去除无关的 NEs，并选择最佳候选 NE。

E 少样本数量的影响

图 5 显示，GEC 和 RAGEC 在增加少样本数量时没有提升。相反，随着少样本数量的增加，DeRAGEC 在生成理由和选择正确的 NE 方面表现出更好的性能。这些结果表明，DeRAGEC 有效地利用了额外的少样本来优化其推理过程并改善任务特定的结果。

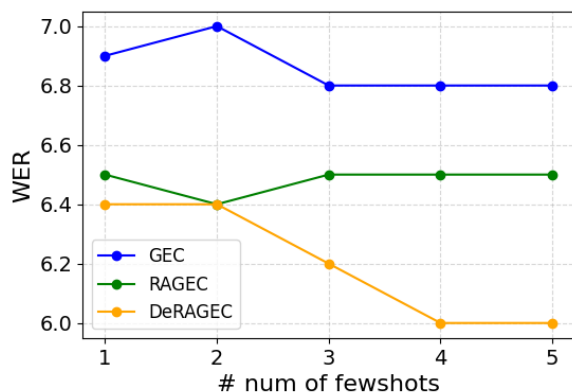


Figure 5: 少样例数量对 CommonVoice 中 Llama-v3.1 的 WER 影响。

F ASR 假设上的命名实体识别性能。

图 6 显示了在 ASR 1-best 假设上的 NER 性能，使用来自真实转录本的 NER 结果作为伪标签。随着 ASR 假设的 WER 增加，NER 性能下降，凸显了从 h_1 中提取语音查询并检索潜在命名实体候选的瓶颈。

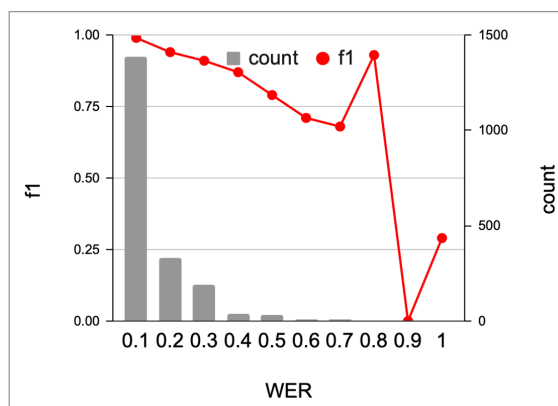


Figure 6: 在不同 WER 上的 NER 性能 (F1 值)。

G 提示模板

我们将框架模块化，以简化整体过程，并为每个模块提供专门的提示模板。模块 1 执行合成原理生成，产生中间解释，并用作少量示例以指导模型的推理。模块 2 通过将输入重新格式化为 MCQ 格式来执行命名实体过滤。模块 3 进行 GEC。模块 1、2 和 3 的相应提示模板在表格 6、7、8 和 9 中详细说明。

Module 1: Synthesize rationales for fewshots

You are given a cloze sentence, candidate named entities, their phonetic similarity scores, and their definitions. Each named entity is provided with several options indicated by ID letters A, B, C, etc. Each option follows the format: [ID letter]: [Named-Entity] ([Phonetic similarity score] | [Definition])

Your task is to identify the most appropriate Named-Entity for [BLANK].

Answer with the "letter: Named-Entity" corresponding to your choice A, B, C, etc.

{ 输入 }

Explain in detail how the input results in the answer: { 答案 }

Answer should not be said at first. The rationale and answer are enclosed within <think> </think> and <answer> </answer> tags, respectively, i.e., <think> rationale here </think> <answer> answer here </answer>.

Table 6: 综合少量样本的理由。

Module 2 -(1): Filtering NEs without reasoning process

[Task Description]

You are given a cloze sentence, candidate named entities, their phonetic similarity scores, and their definitions. The sentence are formatted as a cloze test, where the blanks to fill are indicated by [Blank]. Each named entity is provided with several options indicated by ID letters A, B, C, etc. Each option follows the format: [ID letter]: [Named-Entity] ([Phonetic similarity score] | [Definition])

Your task is to identify the most appropriate Named-Entity for [BLANK]. Please provide the answer with [letter]: [Named-Entity] format. REMEMBER you should return only the answer, not return any explanation.

I will give you few-shot examples.

{ 少量示例 }

[Test Case]

<input>

Cloze sentence: { 填空句 }

Options: { 选项 }

<output>

Answer:

Table 7: 过滤不经过推理过程的 NEs。

Module 2 -(2): Filtering NEs with reasoning process

[Task Description]

You are given a cloze sentence, candidate named entities, their phonetic similarity scores, and their definitions. The sentence are formatted as a cloze test, where the blanks to fill are indicated by [Blank]. Each named entity is provided with several options indicated by ID letters A, B, C, etc. Each option follows the format: [ID letter]: [Named-Entity] ([Phonetic similarity score] | [Definition])

Your task is to identify the most appropriate Named-Entity for [BLANK]. Please generate a brief explanation how the given input lead to your answer. You should not generate the answer before reasoning process. The rationale and answer are enclosed within <think> </think> and <answer> </answer> tags, respectively, i.e., <think> rationale here </think> <answer> answer here </answer>.

I will give you few-shot examples.

{ 少样本示例 }

[Test Case]

<input>

Cloze sentence: { 完形填空句子 }

Options: { 选项 }

<output>

Table 8: 利用推理过程过滤命名实体。

Module 3: GEC

[Task Description]

You are given the 5-best hypotheses from a speech recognition system and a list of named entity candidates. Your task is to generate a corrected transcription using the 5-best hypotheses and named entities.

I will give you few-shot examples.

{ 少样本示例 }

REMEMBER you should return only the corrected transcription, not return any explanation.

[Test Case]

<input>

5-best: { 假设 (H) }

Named-Entities: { filterd_named_entities_with_generated_rationale ($\hat{n}, r$) }

<output>

Corrected:

Table 9: GEC 的提示。