

CCL25-Eval 任务 10 的系统报告: SRAG-MAV 用于细粒度的中文仇恨言论识别

Jiahao Wang, Ramen Liu, Longhui Zhang, Jing Li*
Harbin Institute of Technology, Shenzhen, China
wjh123king@gmail.com, jingli.phd@hotmail.com

Abstract

本文介绍了我们为 CCL25-Eval 任务 10 设计的系统, 该任务涉及细粒度中文仇恨言论识别 (FGCHSR)。我们提出了一种新颖的 SRAG-MAV 框架, 该框架协同整合了任务重新构建 (TR)、自检索增强生成 (SRAG) 和多轮累积投票 (MAV)。我们的方法将四元组提取任务重新构建为三元组提取, 使用从训练集中动态检索的方式创建上下文提示, 并通过多轮推理结合投票来提高输出的稳定性和性能。基于 Qwen2.5-7B 模型, 我们的系统在 STATE ToxiCN 数据集上的硬得分为 26.66, 软得分为 48.35, 平均得分为 37.505, 显著优于基线模型如 GPT-4o (平均得分 15.63) 和微调后的 Qwen2.5-7B (平均得分 35.365)。代码可在 <https://github.com/king-wang123/CCL25-SRAG-MAV> 获得。

1 引言

社交媒体的发展显著放大了仇恨言论的传播 (Fortuna and Nunes, 2018), 恶意内容针对种族、地区和性别等属性, 对个人和社会造成了巨大伤害。有效的仇恨言论检测已成为自然语言处理 (NLP) 的重要关注点, 旨在缓解这些负面影响 (Davidson et al., 2017; Waseem and Hovy, 2016)。此外, 确保检测模型的公平性以避免潜在偏见对于其实际部署至关重要 (Sap et al., 2019)。传统方法通常依赖于二元分类来识别仇恨内容 (Fortuna and Nunes, 2018), 但这些方法缺乏探测仇恨言论内部结构的细粒度能力, 限制了它们的可解释性和对下游应用的效用 (Yin and Zubiaga, 2021)。因此, 细粒度中国仇恨言论识别 (FGCHSR), 它提取具体目标或仇恨类型等结构化信息, 正受到越来越多的关注 (Basile et al., 2019; Mathew et al., 2021; Ren et al., 2021)。

CCL25-Eval 任务 10 专注于从中文社交媒体文本中提取四元组 (目标、论点、针对群体、仇恨)。由于中文仇恨言论 (Pavlopoulos et al., 2020) 的微妙性和语境依赖性质, 四元组元素的相互依赖性, 以及高质量标注数据的有限性, 这项任务特别具有挑战性。STATE ToxiCN 研究 (Bai et al., 2025) 凸显了这些困难, 表明即使是最先进的模型如 GPT-4o 也只能获得平均分 15.63, 而经过微调的开源模型如 Qwen2.5-7B 达到 35.365, 但仍需要进一步优化。

为了解决这些挑战, 我们提出了一种新颖的 SRAG-MAV 框架, 该框架协同结合了任务重构 (TR)、自检索增强生成 (SRAG) 和多轮累积投票 (MAV)。我们的方法将四元组提取简化为三元组提取, 通过受检索增强生成 (RAG) (Lewis et al., 2020) 启发的动态检索增强上下文理解, 并通过基于平行缩放法则 (PARSCALE) (Chen et al., 2025) 原理的多轮推理确保输出的稳定性。我们的贡献包括:

- 提出了一种新颖的 SRAG-MAV 框架, 该框架集成了 TR、SRAG 和 MAV, 展示了在 FGCHSR 上的卓越性能, 并且可以适应其他结构化 NLP 任务。
- 进行了全面的实验, 验证了我们方法的有效性和稳健性, 并评估了各个组件的性能贡献。
- 在 <https://github.com/king-wang123/CCL25-SRAG-MAV> 发布的代码, 促进了可重复性并推动了仇恨言论检测及其他相关自然语言处理领域的进一步研究。

Corresponding author: Jing Li (jingli.phd@hotmail.com).

©2025 China National Conference on Computational Linguistics

This work is licensed under a Creative Commons Attribution 4.0 International License. License details: <http://creativecommons.org/licenses/by/4.0/>.

2 方法论

2.1 系统概述

我们的系统结合了 TR、SRAG 和 MAV，从中文社交媒体文本中提取细粒度的仇恨言论四元组，如图 1 所示。工作流程将任务简化为三元组提取，通过检索增强上下文理解，并通过迭代投票稳定输出。

最初，我们将四元组数据集转换为三元组数据集，以降低任务复杂性，这与 TR 的原则相一致。然后，我们使用检索模型将所有训练输入编码到一个向量数据库中，从而在训练和推理阶段实现高效检索。

训练阶段：对于每个输入，我们从训练集中检索出与之最相似的样本（不包括输入自身），将检索到的样本与输入连接以形成提示。这些提示及其对应的三元组输出，构成用于微调的新训练样本。

推理阶段：对于每个测试输入，我们检索与其最相似的前 k 个训练样本来构建 k 个提示。模型在多轮迭代中为这些 k 个提示生成三元组，直到最频繁的三元组的出现频率超过阈值 τ ，此时 MAV 将其选为最终的三元组答案。然后，通过从目标群体中推断出仇恨性，将选定的三元组转换为四元组。

这个流程，如图 1 所示，在保持简单性的同时，兼顾了上下文丰富性 (Lewis et al., 2020) 和输出稳定性，其中 k 和 τ 是关键超参数。

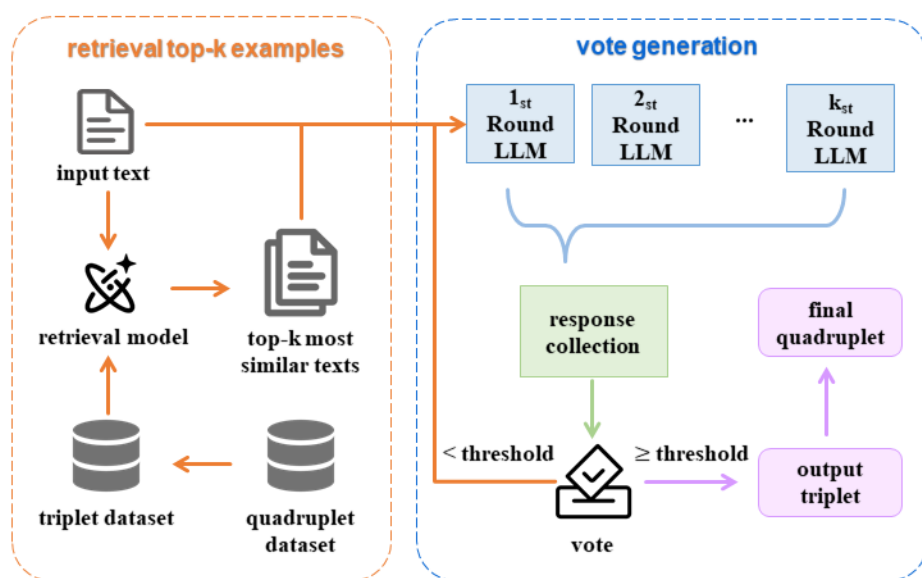


Figure 1: SRAG-MAV 的系统架构，描述了从输入文本到最终四元组输出的工作流程。该过程包括：(1) 将四元组数据集转换为三元组数据集，(2) 检索前 k 个相似样本并将其与输入文本连接以构建提示，(3) 投票选择超过阈值 τ 的三元组答案，否则继续迭代推理直到达到阈值，(4) 将选定的三元组转换为最终的四元组输出。

2.2 任务重组 (TR)

对训练数据的分析揭示了目标群体与仇恨性标签之间的强相关性：“无仇恨”情况仅发生在目标群体为“无仇恨”的情形下；否则，仇恨性被标记为“仇恨”。利用这一模式，我们将原始的四元组提取任务重新建模为三元组提取任务。此简化降低了结构化生成的复杂性，因为仇恨性标签可以从目标群体中确定性地推断出来，从而提高了大型语言模型 (LLMs) 的效率和准确性。

图 2 提供了一个具体的数据示例，以说明 TR 如何在保持结构化输出完整性的同时简化提取过程。

2.3 自我检索增强生成 (SRAG)

检索增强生成 (RAG) 已成为 NLP 中的一个强大范式，被广泛应用于问答、对话系统和知识密集型文本生成等任务 (Lewis et al., 2020; Gao et al., 2023)。通过检索集成外部知识，RAG

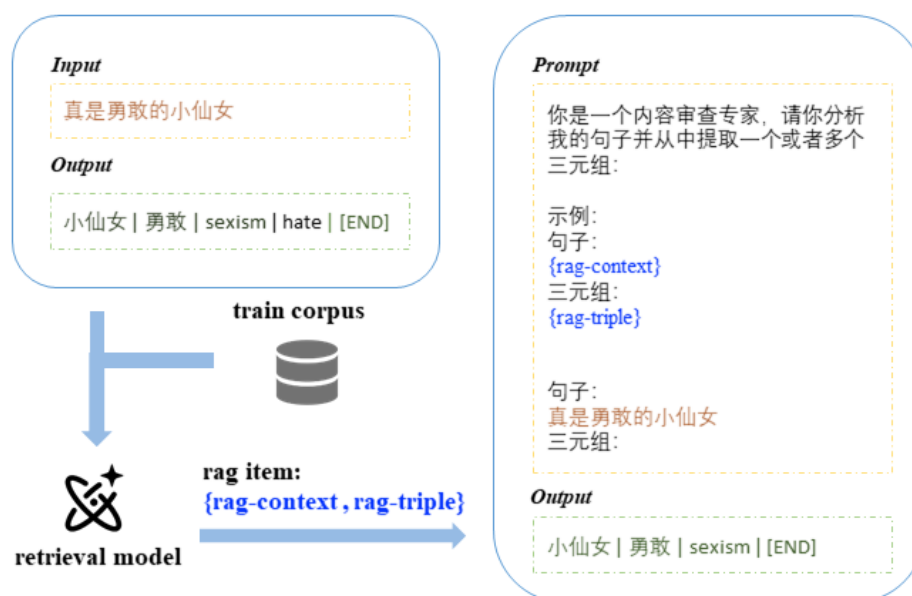


Figure 2: TR 和 SRAG 的示意图：检索模型从训练语料库中检索类似文本，将这些文本连接成一个提示，并生成相应的三元组。

提高了生成输出的上下文相关性和事实准确性 (Ram et al., 2023)。最近的进展进一步优化了 RAG，以处理结构化数据并提高低资源环境中的鲁棒性 (Zhang et al., 2023)。然而，FGCHSR 带来了独特的挑战，包括缺乏高质量的外部语料库和生成结构化四元组的复杂性。

为了解决这些挑战，我们提出了自我检索增强生成 (SRAG) 框架，该框架通过使用训练集本身作为检索语料库来调整 RAG 范式。SRAG 利用语义上相似的注释示例来指导三元组的生成，确保上下文相关的输出而无需依赖外部资源。SRAG 流程包括：

1. 语料库构建：我们使用 bge-large-zh-v1.5 模型 (Xiao et al., 2023) 生成训练集文本的嵌入，以余弦相似度为基础构建检索语料库。
2. 动态检索：对于每个输入文本，我们在训练期间检索除输入自身以外的最相似样本，而在推理期间检索前 k 个最相似样本。
3. 提示生成：检索到的样本与输入文本结合，创建结构化的提示，引导模型生成符合任务的三元组，如图 2 所示。

SRAG 创新性地将训练集作为一个动态检索语料库，利用类似的标注示例来实现少样本学习，以增强任务理解和输出准确性。与传统 RAG 不同，SRAG 不需要外部数据，使其特别适合资源有限的环境和特定领域的任务，如 FGCHSR。

2.4 多轮累积投票 (MAV)

并行缩放定律 (PARSCALE) (Chen et al., 2025) 表明，对输入应用多样的变换以生成多个变体，然后进行并行推理和带可学习参数的结果聚合，可以显著提升大语言模型 (LLM) 的性能。这种方法在不需要对模型重新训练的情况下提高了鲁棒性和准确性，使其在资源受限的环境中处理复杂任务时更为高效。

受到 PARSCALE 对多元输入并行处理的强调的启发，我们引入了多轮累积投票 (MAV) 作为 FGCHSR 的一种创新改编，它通过 SRAG 检索的例子生成多样化的提示，并通过投票机制选择最佳三元组输出。MAV 流程包括：

1. 多样化提示：通过 SRAG，从三元组数据集中检索与每个输入文本最相似的前 k 个样本，并将每个检索到的样本与输入文本连接起来，构建 k 个不同的提示。

2. 多轮推理：对每个提示迭代地执行推理，在多次迭代中生成并累积三元组结果的频率，直到最频繁的三元组超过阈值 τ 。
3. 投票机制：选择达到频率阈值 τ 的三元组输出，并将其转换为四元组作为最终结果。

MAV 因其成本效益高而突出，仅需额外的推理时间资源，而不需重新训练或参数调整。其灵活性在第 3.2 节中得以展示，其中通过逐步提高阈值来改善结果，允许根据可用计算资源进行动态调整。此外，其简单的实现提高了可靠性，使得 MAV 在受限条件下特别有效。

3 实验

3.1 实验设置

我们在 STATE ToxiCN 数据集 (Bai et al., 2025) 上评估了我们的方法，该数据集包含 4,000 个训练样本和 1,602 个测试样本，使用 4 个 $\times$ NVIDIA L40S 40GB GPU 进行计算。基础模型 Qwen2.5-7B (Team, 2024) 使用 LLaMA-Factory 框架 (Zheng et al., 2024) 进行训练，并使用 vLLM (Kwon et al., 2023) 进行推理。对于检索，我们使用了 bge-large-zh-v1.5 模型 (Xiao et al., 2023)，生成参数在微调时配置为温度 0.7，在 MAV 推理时为 0.1。MAV 配置使用了 top- k 值为 10 和投票阈值 τ 为 200。评估指标包括：

- 严格分数：用于精确四元组匹配的 F1 分数。
- 软得分：针对部分匹配的 F1 得分，要求目标群体和仇恨特征相同，并且目标和论点相似度超过 50 %。
- 平均分：硬评分和软评分的平均值。

基准线取自 STATE ToxiCN 基准测试 (Bai et al., 2025)，并与我们的方法进行比较以验证其有效性。

3.2 实验结果

我们进行了三个实验，包括一个模型比较以基准测试我们的系统与基线的对比，以及一个 MAV 参数敏感性分析来评估阈值变化对性能的影响，和消融研究以评估每个组件的贡献。

3.2.1 模型比较

Model	Hard Score	Soft Score	Average Score
mT5-base	16.60	38.61	27.605
Mistral-7B	23.72	45.62	34.670
LLaMA3-8B	24.27	46.08	35.175
Qwen2.5-7B	23.70	47.03	35.365
ShieldLM-14B-Qwen	23.59	45.58	34.585
ShieldGemma-9B	23.49	47.14	35.315
Ours	26.66	48.35	37.505

Table 1: 在 STATE ToxiCN 测试集上的性能比较。基线模型的结果直接引用自 STATE ToxiCN 论文 (Bai et al., 2025)，代表了原始监督微调 (SFT) 的结果。我们的方法在所有指标上显著优于这些基线。

表 1 展示了我们的系统在 STATE ToxiCN 测试集上实现了 26.66 的 Hard Score，48.35 的 Soft Score，和 37.505 的 Average Score，显著超过了所有使用普通监督微调 (SFT) 训练的基线模型。我们的方法带来了显著的改进，尤其是在 Hard Score 上，其体现了精确的四元组匹配，并表明在 FGCHSR 中表现强劲。

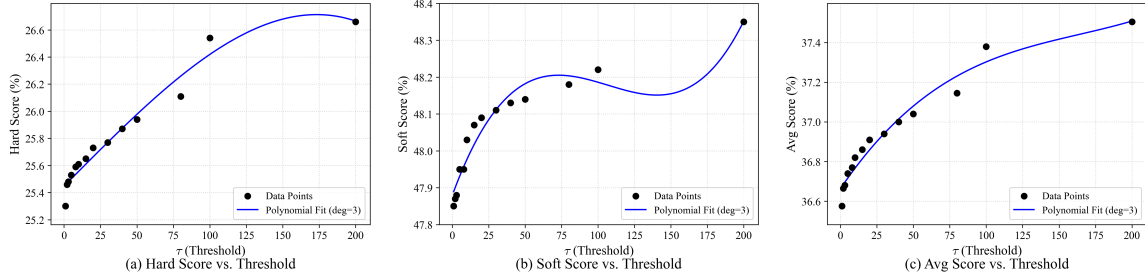


Figure 3: MAV 阈值参数（顶部- $k=10$ ）对 STATE ToxiCN 测试集的影响。图形展示了阈值（ τ ）数值与所得硬得分（a）、软得分（b）和平均得分（c）之间的关系。

3.2.2 MAV 参数敏感性分析

图 3 详细分析了 MAV 阈值（ τ ）参数对模型性能的影响，测试的阈值为 [1, 2, 3, 5, 8, 10, 15, 20, 30, 40, 50, 80, 100, 200]。Hard Score 呈现出显著增长，从 25.30 增加到 26.66，增加了 1.36 分，尤其是在更高的阈值下有明显的增长（例如，从 $\tau = 80$ 的 26.11 增长到 $\tau = 200$ 的 26.66），这强调了该方法在实现精确四元组匹配方面的有效性。Soft Score 也稳步提高，从 47.85 增加到 48.35，提高了 0.50 分，这表明部分匹配准确性的增强。平均分从 36.575 上升到 37.505，提高了 0.93 分，表明在两个指标上均衡提升。结果验证了 MAV 在通过累积投票稳定输出方面的作用，Hard Score 的显著增长突显了其在精细化仇恨言论识别中的优越性。

模型训练提供了基础能力，而推理策略通过引入适度的计算开销进一步释放模型的潜力，两者相辅相成。

3.2.3 消融研究

Configuration	Hard Score	Soft Score	Average Score
Base Model	23.70	47.03	35.365
+ TR	24.33	47.35	35.840
+ TR + SRAG	25.30	47.85	36.575
+ TR + SRAG + MAV	26.66	48.35	37.505

Table 2: 消融研究的结果，展示了任务重构（TR）、自检索增强生成（SRAG）和多轮累积投票（MAV）对整体性能的增量贡献。

如表 2 所示，消融研究从高层次角度展示了每个组件在增强模型性能方面的有效性。从基本模型（通过标准 SFT 训练的 Qwen2.5-7B）开始，引入 TR 简化了模型的输出结构，导致性能的显著提升，这突显了其在简化任务方面的有效性。在此基础上，增加 SRAG 通过利用上下文检索进一步加强了模型，结果显示出明显的改进，突显了其在优化预测方面的作用。最终加入 MAV 提供了最显著的增强，通过迭代推理显著提高了模型的稳定性和准确性，这强调了 MAV 在整体性能中关键的贡献。

4 结论

对于 CCL25-Eval 任务 10，我们开发了一个新颖的 SRAG-MAV 框架，旨在有效检测和减少社交媒体上有害内容的传播。我们的方法在 STATE ToxiCN 测试集 (Bai et al., 2025) 上实现了 26.66 的硬分数、48.35 的软分数和 37.505 的平均分数，显著优于基线模型，例如 GPT-4o（平均分数 15.63）和微调的 Qwen2.5-7B（平均分数 35.365）。TR 将四元组提取任务简化为三元组提取，降低了复杂性；SRAG 通过利用训练集作为检索语料库来增强上下文理解；MAV 通过迭代提示生成和投票确保输出稳定性。这些组件协同工作，我们的切除实验显示了每个模块带来的增量性能提升。

我们系统的开源实现 (<https://github.com/king-wang123/CCL25-SRAG-MAV>) 促进了可重复性和进一步的研究。然而，局限性包括模型的特定领域性能、对纯文本数据的依赖以及 MAV

的高投票阈值增加了计算成本。未来的工作将探索跨域迁移学习以增强泛化能力 (Toraman et al., 2022), 采用多模态方法结合文本和图像以获得更丰富的上下文 (Gomez et al., 2020; Das et al., 2020), 并优化 MAV 的计算效率以扩大其适用性。

5

致谢

本工作部分由中国国家自然科学基金 (62476070)、深圳市科技计划 (JCYJ20241202123503005, GXWD20231128103232001, ZDSYS20230626091203008, KQTD2024072910215406) 及广东省科技厅 (2024A1515011540) 资助。

我们感谢 CCL25-Eval 的组织者提供的平台, STATE ToxiCN 数据集的提供者对我们实验的支持, 以及审稿人提供的宝贵反馈。

References

- Zewen Bai, Shengdi Yin, Junyu Lu, Jingjie Zeng, Haohao Zhu, Yuanyuan Sun, Liang Yang, and Hongfei Lin. 2025. State toxicon: A benchmark for span-level target-aware toxicity extraction in chinese hate speech detection. arXiv preprint arXiv:2501.15451.
- Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In Proceedings of the 13th International Workshop on Semantic Evaluation, pages 54–63.
- Mouxian Chen, Binyuan Hui, Zeyu Cui, Jiayi Yang, Dayiheng Liu, Jianling Sun, Junyang Lin, and Zhongxin Liu. 2025. Parallel scaling law for language models. arXiv preprint arXiv:2505.10475.
- Abhishek Das, Japsimar Singh Wahi, and Siyao Li. 2020. Detecting hate speech in multimodal memes. arXiv preprint arXiv:2012.14891.
- Thomas Davidson, Dana Warmusley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In Proceedings of the International AAAI Conference on Web and Social Media, volume 11, pages 512–515.
- Paula Fortuna and Sérgio Nunes. 2018. A survey on automatic detection of hate speech in text. ACM Computing Surveys, 51(4):1–30.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997.
- Raul Gomez, Jaume Gibert, Lluís Gomez, and Dimosthenis Karatzas. 2020. Exploring hate speech detection in multimodal publications. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 1470–1478.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In Proceedings of the 29th Symposium on Operating Systems Principles, pages 611–628.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In Advances in Neural Information Processing Systems, volume 33, pages 9459–9474.
- Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2021. Hatexplain: A benchmark dataset for explainable hate speech detection. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 35, pages 14867–14875.
- John Pavlopoulos, Jeffrey Sorensen, Lucas Dixon, Nithum Thain, and Ion Androutsopoulos. 2020. Toxicity detection: Does context really matter? In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4296–4305.

- Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.
- Feiliang Ren, Longhui Zhang, Shujuan Yin, Xiaofeng Zhao, Shilei Liu, Bochao Li, and Yaduo Liu. 2021. A novel global feature-oriented relational triple extraction model based on table filling. *arXiv preprint arXiv:2109.06705*.
- Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678.
- Qwen Team. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.
- Cagri Toraman, Furkan Şahinuç, and Eyup Yilmaz. 2022. Large-scale hate speech detection with cross-domain transfer. In *Proceedings of the 13th Language Resources and Evaluation Conference*, pages 2215–2225.
- Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93.
- Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2023. C-pack: Packed resources for general chinese embeddings. *arXiv preprint arXiv:2309.07597*.
- Wenjie Yin and Arkaitz Zubiaga. 2021. Towards generalisable hate speech detection: a review on obstacles and solutions. *PeerJ Computer Science*, 7:e598.
- Longhui Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, Meishan Zhang, and Min Zhang. 2023. A two-stage adaptation of large language models for text ranking. *arXiv preprint arXiv:2311.16720*.
- Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyang Luo. 2024. Llama-factory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410.