سومین شکاف به جایگاه قواعد دامنه مربوط است. در پژوهش‌های پرامپتی، دانش در متن دستور یا نمونه‌ها قرار می‌گیرد [۶، ۷، ۹]؛ در روش‌های قاعده‌محور تصمیم خارج مدل است؛ و در سامانه‌های ترکیبی، مدل با همان منطق LLM کلاسیک یا بازیابی نقش دروازه را دارد [۱۰، ۱۲]. مقایسه مستقیم قاعده قطعی خارج از به‌صورت دستور درون پرامپت، تحت شرایط ثابت، در مجموعه مطالعات مرورشده مشاهده نشد. دو روش برای سنجش همین تمایز تعریف شده‌اند Prompt-only و Hybrid.

متن‌باز متوسط در استقرار کاملاً محلی و بدون تنظیم LLM چهارمین شکاف، محدود بودن شواهد درباره یک GPT- یا ChatGPT است. بخشی از مطالعات از خدمات تجاری مانند BGL دقیق برای طبقه‌بندی خطبه‌خط را آموزش یا تنظیم کرده‌اند [۵، ۱۰، ۱۵] و Mistral یا LLaMA استفاده کرده‌اند [۶، ۹]، بخشی ۴o-mini خودنظارتی بهره گرفته است [۲۷]. پژوهش حاضر LogBERT مطالعه محلی سال ۲۰۲۶ از محلی و وزن‌های ثابت ارزیابی می‌کند [۲۵، ۲۶] Ollama را با Qwen۲٫۵:۷B-Instruct.

شامل BGL برآیند این شکاف‌ها جایگاه پژوهش را چنین مشخص می‌کند: یک خط لوله بازتولیدپذیر برای JSON استخراج و حذف برچسب، نرمال‌سازی حساس به معنا، کش نسخه‌بندی‌شده در سطح قالب، خروجی اعتبارسنجی‌شده و ثبت منبع تصمیم؛ سپس مقایسه دو معماری که فقط در محل اعمال دانش قطعی تفاوت دارند. این صورت‌بندی ادعای برتری پیشینی هیچ مسیر را مطرح نمی‌کند و نتیجه باید از آزمایش کامل و یکسان دو روش به‌دست آید.

۲-۶ جمع‌بندی

در این فصل مفاهیم لاگ، ناهنجاری، واحد تحلیل، تجزیه و نرمال‌سازی، مدل‌های ترتیبی، مدل‌های BGL، Qwen۲٫۵، LLM، زبانی پیش‌آموزش‌دیده، LLM به تنظیم BERT تعریف شد. مرور پیشینه نشان داد مسیر پژوهش از پیش‌بینی توالی و مدل‌های پرامپت، بازیابی و سامانه‌های چندجزئی گسترش یافته است [۱–۲۴].

مقایسه مطالعات نشان داد که نوع واحد تحلیل، نیاز یا عدم نیاز به آموزش، سازوکار کنترل هزینه و محل استقرار برای تفسیر نتایج ضروری‌اند. شکاف استخراج‌شده از منابع، نبود مقایسه مستقیم و کنترل‌شده بین محلی در طبقه‌بندی Qwen و انتقال دانش آن به پرامپت، همراه با کش قالب و LLM قطعی بیرون guard است. فصل بعد باید معماری نرم‌افزار، قواعد نرمال‌سازی، ساختار پرامپت‌ها، کلید کش BGL خطبه‌خط، پروتکل اجرای دو روش و شیوه محاسبه معیارها را به‌صورت عملیاتی تشریح کند، Ollama تنظیمات.

۲-۷ منابع فصل دوم

[۱] Du, M.; Li, F.; Zheng, G.; Srikumar, V. DeepLog: Anomaly Detection and Diagnosis from System Logs through Deep Learning. ACM CCS, ۲۰۱۷. DOI: ۱۰,۱۱۴۵/۳۱۳۳۹۵۶,۳۱۳۴۰۱۵.

[۲] Guo, H.; Yuan, S.; Wu, X. LogBERT: Log Anomaly Detection via BERT. arXiv:۲۱۰۳,۰۴۴۷۵, ۲۰۲۱.

[۳] Chen, S.; Liao, H. BERT-Log: Anomaly Detection for System Logs Based on Pre-trained Language Model. Applied Artificial Intelligence, ۳۶(۱), ۲۰۲۲. DOI: ۱۰,۱۰۸۰/۰۸۸۳۹۵۱۴,۲۰۲۲,۲۱۴۵۶۴۲.

[۴] Han, X.; Yuan, S.; Trabelsi, M. LogGPT: Log Anomaly Detection via GPT. ۲۰۲۳.

[۵] Guan, W.; Cao, J.; Qian, S.; Gao, J.; Ouyang, C. LogLLM: Log-based Anomaly Detection Using Large Language Models. ۲۰۲۴.

[۶] Qi, J.; Huang, S.; Luan, Z.; Fung, C.; Yang, H.; Qian, D. LogGPT: Exploring ChatGPT for Log-Based Anomaly Detection. ۲۰۲۳.

[۷] Liu, Y.; Tao, S.; Meng, W.; Wang, J.; Ma, W.; Chen, Y.; Zhao, Y.; Yang, H.; Jiang, Y. Prompt Engineering Towards Zero-Shot and Interpretable Log Analysis. ۲۰۲۴. Earlier version: Interpretable Online Log Analysis Using Large Language Models with Prompt Strategies, ۲۰۲۳.

[٨] Pan, J.; Wong, S. L.; Yuan, Y. RAGLog: Log Anomaly Detection using Retrieval Augmented Generation. ٢٠٢٣.

[٩] Park, J.; Choi, E.; Lee, J. Hybrid Prompting for LLM-based Log Anomaly Detection. Journal of KIISE, ٥٢(١٢), ٢٠٢٥. DOI: ١٠.٥٦٢٦/JOK.٢٠٢٥.٥٢.١٢.١٠٦٧.

[١٠] Hadadi, F.; Xu, Q.; Bianculli, D.; Briand, L. LLM meets ML: Data-efficient Anomaly Detection on Unstable Logs. ٢٠٢٤–٢٠٢٥.

[١١] Hadadi, F.; Xu, Q.; Bianculli, D.; Briand, L. Anomaly Detection on Unstable Logs with GPT Models. ٢٠٢٤.

[١٢] Ma, L.; Yang, W.; Li, Y.; Fei, B.; Zhou, M.; Li, S.; Jiang, S.; Xu, B.; Xiao, Y. AdaptiveLog: An Adaptive Log Analysis Framework with the Collaboration of Large and Small Language Model. arXiv:٢٥٠١,١١٠٣١, ٢٠٢٥.

[١٣] Almodovar, C.; Sabrina, F.; Karimi, S.; Azad, S. LogFiT: Log Anomaly Detection using Fine-Tuned Language Models. ٢٠٢٥.

[١٤] Lim, Y. F.; Zhu, J.; Pang, G. Adapting Large Language Models for Parameter-Efficient Log Anomaly Detection. ٢٠٢٥.

[١٥] Zhang, Z.; Li, S.; Zhang, L.; Ye, J.; Hu, C.; Yan, L. LLM-LADE: Large language model-based log anomaly detection with explanation. Knowledge-Based Systems, ٣٢٦, Article ١١٤٠٦٤, ٢٠٢٥.

[١٦] Sun, Y.; Keung, J. W.; Yang, Z.; Liu, S.; Yu, H. K. SemiRALD: A semi-supervised hybrid language model for robust Anomalous Log Detection. Information and Software Technology, ١٨٣, ١٠٧٧٤٣, ٢٠٢٥. DOI: ١٠,١٠١٦/j.infsof.٢٠٢٥,١٠٧٧٤٣.

[١٧] Pospieszny, P.; Mormul, W.; Szyndler, K.; Kumar, S. ADALog: Adaptive Unsupervised Anomaly Detection in Logs with Self-attention Masked Language Model. arXiv:٢٥٠٥,١٣٤٩٦, ٢٠٢٥.

[١٨] Zhong, A.; Mo, D.; Liu, G.; Liu, J.; Lu, Q.; Zhou, Q.; Wu, J.; Li, Q.; Wen, Q. LogParser-LLM: Advancing Efficient Log Parsing with Large Language Models. arXiv:٢٤٠٨,١٣٧٢٧, ٢٠٢٤.

[١٩] Le, V.-H.; Zhang, H. Log Parsing with Prompt-based Few-shot Learning. arXiv:٢٣٠٢,٠٧٤٣٥, ٢٠٢٣.

[٢٠] Sui, Y.; Wang, X.; Cui, T.; Xiao, T.; He, C.; Zhang, S.; Zhang, Y.; Yang, X.; Sun, Y.; Pei, D. Bridging the Gap: LLM-Powered Transfer Learning for Log Anomaly Detection in New Software Systems. ٢٠٢٥.

[٢١] Gao, Y.; Luo, T.; Huang, K.; Tang, J.; Li, X. LogLAA: an adaptive integrated log anomaly analysis framework. Cybersecurity, ٩:١٤١, ٢٠٢٦. DOI: ١٠,١١٨٦/s٤٢٤٠٠-٠٢٦-٠٠٥٧٣-٨.

[٢٢] De la Cruz Cabello, M.; Prince Sales, T.; Machado, M. R. AIOps for log anomaly detection in the era of LLMs: A systematic literature review. Intelligent Systems with Applications, ٢٨, Article ٢٠٠٦٠٨, ٢٠٢٥.

[٢٣] Liu, Y.; Pei, C.; Xu, L.; et al. OpsEval: A Comprehensive Benchmark Suite for Evaluating Large Language Models' Capability in IT Operations Domain. FSE Companion, ٢٠٢٥. DOI: ١٠.١١٤٥/٣٦٩٦٦٣٠.٣٧٢٨٥٧٢.

[٢٤] Xu, R.; Ding, K. Large Language Models for Anomaly and Out-of-Distribution Detection: A Survey. arXiv:٢٤٠٩,٠١٩٨٠, ٢٠٢٥.

[٢٥] Qwen Team. Qwen٢,٥ Technical Report. arXiv:٢٤١٢,١٥١١٥, ٢٠٢٤.

[٢٦] Ollama. API Introduction; Structured Outputs; Usage Metrics. Official documentation. Accessed ١٤ August ٢٠٢٦.

[٢٧] De la Cruz Cabello, M.; Prince Sales, T.; Machado, M. R. Log anomaly detection in AIOps: A real-world implementation using Large Language Models. Systems and Soft Computing, ٨, Article ٢٠٠٤٧٥, ٢٠٢٦.