فصل دوم: مروری بر ادبیات و پیشینه تحقیق

مقدمه ۱-۲

لاگ‌های سیستمی گزارش‌های متنی یا نیمه‌ساخت‌یافته‌ای هستند که رخدادهای زمان اجرا، وضعیت اجزای ،سامانه، پیام‌های هشدار و اطلاعات عیب‌یابی را ثبت می‌کنند. این داده‌ها در بیشتر زیرساخت‌های نرم‌افزاری شبکه‌ای و محاسباتی تولید می‌شوند و از منابع اصلی پایش سلامت سامانه، تشخیص خطا، تحلیل رخداد و یافتن علت ریشه‌ای به‌شمار می‌روند. با افزایش مقیاس سامانه‌ها، تعداد پیام‌ها و تنوع قالب‌های لاگ نیز افزایش یافته .و تحلیل دستی دیگر پاسخ‌گوی جریان پیوسته داده نیست [۱۴، ۲۰، ۲۱]

پژوهش تشخیص ناهنجاری لاگ از روش‌های قاعده‌محور و آماری به مدل‌های یادگیری توالی، مدل‌های زبانی LSTM توالی رخدادها را با DeepLog .پیش‌آموزش‌دیده و سپس مدل‌های زبانی بزرگ گسترش یافته است را وارد مسئله می‌کنند؛ و پژوهش‌هایی BERT بازنمایی دوطرفه BERT-Log و LogBERT مدل می‌کند؛ از توان پیروی از دستور و درک Hybrid Prompting و RAGLog، LogGPT، LogPrompt مانند .استفاده می‌کنند [۱–۶، ۱۲، ۱۸، ۱۹، ۲۱] LLM معنایی

مدل زبانی در مطالعات مرورشده نقش واحدی ندارد. در برخی پژوهش‌ها طبقه‌بند مستقیم است؛ در برخی دیگر برای پیش‌بینی رخداد بعدی آموزش داده می‌شود؛ در گروهی برای تولید داده یا توضیح به‌کار می‌رود؛ و در چارچوب‌های ترکیبی، مدل بزرگ فقط نمونه‌های دشوار را دریافت می‌کند. مرور نظام‌مند حوزه لاگ و مرور عمومی تشخیص ناهنجاری و داده خارج از توزیع نیز همین تنوع نقش را به‌عنوان یکی از محورهای .اصلی دسته‌بندی معرفی می‌کنند [۱۴، ۲۷]

به همین دلیل، مقایسه پژوهش‌ها باید افزون بر نام مدل، نوع ورودی، واحد تصمیم، نیاز به آموزش، سازوکار حاصل از طبقه‌بندی F۱ آماده‌سازی، نوع استقرار و روش کنترل هزینه را در نظر بگیرد. برای نمونه، عدد خطبه‌خط با نتیجه یک پنجره یک‌دقیقه‌ای یا توالی مبتنی بر نشست هم‌معنا نیست. همچنین کش تصمیم، بازیابی را LLM نمونه و واگذاری به مدل کوچک سه سازوکار متفاوت‌اند، هرچند هر سه تعداد یا هزینه فراخوانی .کاهش می‌دهند [۳، ۱۰، ۱۲، ۱۳]

محدود است. مدل از طریق Qwen۲٫۵:۷B-Instruct و مدل BGL دامنه پژوهش حاضر به مجموعه‌داده و Hybrid و به‌صورت محلی اجرا می‌شود و تنظیم دقیق انجام نمی‌گیرد. مقایسه میان دو روش Ollama اعمال می‌شوند و LLM صورت می‌گیرد: در روش نخست قواعد قطعی پیش از فراخوانی Prompt-only در روش دوم همان دانش دامنه در متن پرامپت قرار می‌گیرد. در هر دو روش، نرمال‌سازی، کش سطح قالب .و اعتبارسنجی خروجی ثابت نگه داشته می‌شوند

این فصل ابتدا مبانی نظری موردنیاز را تعریف می‌کند. سپس هر ۲۷ مقاله موجود در پوشه پژوهش به‌صورت ،هستند LogPrompt مستقل مرور می‌شود؛ حتی دو مقاله [۱] و [۶] که دو نسخه هم‌پوشان از خط پژوهشی جداگانه معرفی و رابطه آن‌ها تصریح می‌شود. در ادامه، جدول ۲۷ ردیفی، شکل ارتباط مطالعات، تحلیل مقایسه‌ای و شکاف تحقیقاتی ارائه می‌گردد. تمام تعریف‌ها و گزاره‌های مربوط به مطالعات به منابع ارجاع .داده شده‌اند و نتایج آزمایش‌های در حال تکمیل پروژه وارد پیشینه نشده‌اند

مبانی نظری و مفاهیم موردنیاز ۲-۲

ماهیت و ساختار لاگ سیستمی ۱-۲-۲

یک پیام لاگ معمولاً شامل سرآیند و بدنه است. سرآیند می‌تواند زمان، سطح شدت، نام مؤلفه، شناسه فرایند یا و LogPPT گره را در بر گیرد و بدنه، شرح رخداد برنامه و پارامترهای زمان اجرا را بیان کند. مطالعات بدنه را به بخش ثابت یا «قالب لاگ» و بخش متغیر یا «پارامتر» تفکیک می‌کنند. کلمات LogParser-LLM ثابت نوع رخداد را نشان می‌دهند و مقادیری مانند شناسه بلوک، نشانی شبکه، مسیر فایل و زمان اجرا میان نمونه‌ها تغییر می‌کنند [۲۳، ۲۴].

لاگ خام به دلیل وجود مقادیر پویا، اختصارات، کدهای خطا و قالب‌های متفاوت، متن طبیعی کامل نیست. بااین‌حال، روابط میان کلمات ثابت و زمینه عملیاتی حامل اطلاعات معنایی‌اند. مدل‌های زبانی پیش‌آموزش‌دیده LLM و LLM‌ها تلاش می‌کنند این اطلاعات را از متن یا توالی استخراج کنند، درحالی‌که روش‌های آماری معمولاً ابتدا پیام را به شناسه قالب یا بردار فراوانی تبدیل می‌کنند [۴، ۱۴، ۱۸، ۱۹].

مفهوم ناهنجاری و سطوح تحلیل ۲-۲-۲

در تشخیص ناهنجاری، مشاهده‌ای ناهنجار تلقی می‌شود که با الگوی رفتار عادی آموخته‌شده یا قواعد میان تغییر هم‌توزیعی که به ناهنجاری منجر می‌شود و تغییر Ding و Xu تعریف‌شده سازگار نباشد. مرور LLM» برای تشخیص» و LLM» را در دو گروه LLM معنایی داده خارج از توزیع تمایز می‌گذارد و نقش «برای تولید» طبقه‌بندی می‌کند [۲۷]. در حوزه لاگ، ناهنجاری می‌تواند در متن یک پیام، فراوانی قالب‌ها ترتیب رخدادها یا معنای یک پنجره زمانی ظاهر شود [۱۴].

واحد تحلیل در ادبیات شامل خط منفرد، قالب، پنجره زمانی، توالی مبتنی بر گره و نشست است. ADALog بر توالی رخدادها تکیه دارند؛ LogBERT و DeepLog پیام منفرد را مستقل تحلیل می‌کند؛ BERT-Log از پنجره یک‌دقیقه‌ای استفاده می‌کند Hybrid Prompting پنجره لغزان مبتنی بر گره می‌سازد؛ و BGL در واحد تحلیل، نحوه تعریف TP، FP، TN و FN و قابلیت مقایسه نتیجه را تعیین می‌کند [۲۲، ۲۱، ۱۹، ۱۸، ۳].

مجموعه‌داده BGL ۳-۲-۲

در آزمایشگاه ملی لارنس لیورمور Blue Gene/L از ابررایانه BGL یا Blue Gene/L مجموعه‌داده گردآوری شده است. این مجموعه ۴،۷۴۷،۹۶۳ پیام دارد و ۳۴۸،۴۶۰ پیام، حدود ۷،۳۴ درصد، با برچسب هشدار به‌عنوان ناهنجار مشخص شده‌اند [۲، ۱۸، ۱۹]. برچسب اصلی در ابتدای هر خط قرار دارد و برای ارزیابی قابل استفاده است، اما حضور آن در ورودی مدل باعث نشت پاسخ می‌شود؛ بنابراین در پژوهش حاضر پیش از تشکیل متن ورودی حذف می‌گردد.

پنجره زمانی پنج‌دقیقه‌ای با میانگین ۵۶۲ رخداد ایجاد LogBERT. در منابع یکسان نیستند BGL پروتکل‌های پنجره Hybrid Prompting شناسه گره، اندازه پنجره و گام را به‌کار می‌گیرد؛ BERT-Log می‌کند؛ پس از ADALog یک‌دقیقه‌ای و نمونه متوازن هزار توالی عادی و هزار توالی ناهنجار می‌سازد؛ و پاک‌سازی، مجموعه‌ای از پیام‌های یکتای عادی و ناهنجار را در سطح خط بررسی می‌کند [۳، ۱۸، ۱۹، ۲۲].

دو هزار لاگ متوالی از بخش API، به دلیل محدودیت نرخ ChatGPT، مبتنی بر LogGPT در پژوهش K-means نیز برای ساخت خوشه‌ها و پایگاه برداری از نمونه‌گیری و RAGLog آزمون انتخاب شده است [۲]. در باید همراه با واحد «BGL» استفاده می‌شود [۱۲]. این تفاوت‌ها نشان می‌دهد عبارت «ارزیابی روی تحلیل، نمونه‌گیری، پنجره و نسبت کلاس‌ها گزارش شود.

۲-۲-۴ تجزیه، پاک‌سازی و نرمال‌سازی

،تجزیه لاگ فرایند تبدیل پیام خام به قالب ثابت و فهرست پارامترهای متغیر است. روش‌های سنتی از فراوانی را با پرامپت‌تیونینگ و تعداد اندکی نمونه RoBERTa مدل LogPPT .شباهت و قواعد نحوی استفاده می‌کنند را با درخت LLM نیز معناشناسی LogParser-LLM برای برچسب‌گذاری توکن‌های پارامتر تنظیم می‌کند؛ .پیشوند ترکیب می‌کند تا فقط برای الگوهای نیازمند تصمیم فراخوانی انجام شود [۲۳، ۲۴]

خوشه‌بندی‌شده کامل نیست، بلکه نگاشتی قطعی از مقادیر پویا به parser نرمال‌سازی پژوهش حاضر یک مقادیر هگزادسیمال، تاریخ و IP، جانشین‌های ثابت برای تشکیل قالب قابل کش است. شناسه‌های گره، نشانی <NON_ZERO> و <ZERO> زمان، مسیرها و اعداد پویا پوشانده می‌شوند. صفر و غیرصفر به‌ترتیب با متمایز می‌مانند، زیرا برخی رخدادها از وضعیت عددی خود معنا می‌گیرند. این تصمیم با اصل تفکیک قالب .و پارامتر در مطالعات تجزیه سازگار است [۲۳، ۲۴]

کیفیت نرمال‌سازی بر دو متغیر اثر دارد: اطلاعاتی که برای تشخیص باقی می‌ماند و تعداد قالب‌های یکتایی که به مدل فرستاده می‌شود. ماسک بیش‌ازحد ممکن است واژه یا مقدار معنادار را حذف کند؛ ماسک کمتر نیز نیز اثر کیفیت تجزیه بر LogParser-LLM و LogLAA .پیام‌های هم‌معنا را به کلیدهای متعدد تبدیل می‌کند .تحلیل پایین‌دستی و ضرورت کنترل دانه‌بندی قالب را گزارش می‌کنند [۱۷، ۲۳]

۲-۲-۵ پارادایم‌های یادگیری در تشخیص ناهنجاری لاگ

.روش نظارت‌شده از نمونه‌های عادی و ناهنجار برچسب‌خورده برای یادگیری مرز تصمیم استفاده می‌کند نمونه‌ای از تنظیم نظارت‌شده است [۱۸]. روش نیمه‌نظارتی از مقدار محدودی داده BERT-Log و RoBERTa این رویکرد را با SemiRALD برچسب‌خورده و داده بدون برچسب بهره می‌گیرد؛ ترکیب می‌کند [۱۶]. روش خودنظارتی از یک وظیفه کمکی مانند پیش‌بینی توکن یا رخداد BiLSTM در این دسته ADALog و LogFiT، LogBERT ماسک‌شده برای یادگیری الگوی عادی استفاده می‌کند؛ .قرار می‌گیرند [۸، ۱۹، ۲۲]

.تنظیم دقیق کامل، تنظیم پارامترکارا و یادگیری درون‌متنی سه شیوه متفاوت سازگارکردن مدل زبانی با دامنه‌اند Lim و پژوهش LLM-LADE مراحل آموزشی اختصاصی دارند؛ GPT مبتنی بر LogGPT و LogLLM LogGPT و LogPrompt استفاده می‌کنند؛ و PEFT یا سایر روش‌های ReFT، LoRA و همکاران از .بدون تغییر وزن، دستور و نمونه را در پرامپت قرار می‌دهند [۱، ۲، ۴–۷، ۹] ChatGPT مبتنی بر

۲-۲-۶ مدل‌های ترتیبی، Transformer و مدل‌های زبانی

رخداد بعدی LSTM با DeepLog .مدل ترتیبی احتمال رخداد بعدی یا سازگاری کل توالی را برآورد می‌کند با سازوکار توجه امکان Transformer .را از تاریخچه پیش‌بینی و انحراف را علامت‌گذاری می‌کند [۲۱]

با زمینه دوطرفه و دو هدف LogBERT .استفاده از زمینه دورتر و پردازش موازی‌تر را فراهم می‌کند و طبقه‌بند متصل، این معماری را برای لاگ به‌کار گرفته‌اند BERT با بازنمایی BERT-Log خودنظارتی و .[۱۸، ۱۹]

خودبازگشتی بنا می‌شوند و پاسخ را براساس دستور Transformer های دستورمحور عمدتاً بر معماریLLM ،فرایند توسعه یک مدل متن‌باز ۱۲ میلیارد پارامتری را در سه مرحله پیش‌آموزی YuLan .متنی تولید می‌کنند تنظیم دستوری و هم‌ترازی انسانی تشریح می‌کند و قابلیت یادگیری درون‌متنی را یکی از پیامدهای پس‌آموزی می‌داند [۲۶]. این مقاله آشکارساز لاگ نیست، اما برای توضیح منشأ توانایی مدل‌های دستورمحور در پیروی .از پرامپت استفاده می‌شود

۲-۲-۷ نقش‌های LLM در خط لوله تحلیل لاگ

در یک خط لوله لاگ، LLM می‌تواند parser، تبیین‌گر یا هماهنگ‌کننده باشد، تولیدکننده داده، طبقه‌بند. LogPrompt، LogGPT. در مرحله استخراج قالب قرار دارند [۲۳، ۲۴] LogPPT و LogParser-LLM از LLM-LADE. تصمیم تشخیص را مستقیماً از مدل دریافت می‌کنند [۱–۳، ۶] Hybrid Prompting و توضیح رخداد را پس از تشخیص تولید می‌کند LogLAA برای افزایش داده و توضیح استفاده می‌کند و LLM [۷، ۱۷].

در سامانه‌های چندمدلی، LLM لزوماً برای همه نمونه‌ها اجرا نمی‌شود. AdaptiveLog عدم‌قطعیت مدل نتیجه چند مدل یادگیری ماشین را با FlexLog. کوچک را معیار ارجاع به مدل بزرگ قرار می‌دهد [۱۳] را برای ساخت LLM نیز تفسیر رخداد با LogSynergy. بازیابی و کش ترکیب می‌کند [۱۰]، Mistral بازنمایی قابل انتقال میان سامانه‌ها به‌کار می‌برد [۱۵].

۲-۲-۸ Qwen۲٫۵:۷B-Instruct

خانواده‌ای از مدل‌های باز با نسخه‌های پایه و دستورمحور در اندازه‌های ۰٫۵ تا ۷۲ میلیارد پارامتر Qwen۲٫۵ پیش‌آموزی بر حدود ۱۸ تریلیون توکن و پس‌آموزی با بیش از یک میلیون Qwen۲٫۵ است. گزارش فنی حدود ۶٫۵ میلیارد پارامتر Bنمونه تنظیم دستوری و مراحل تقویت یادگیری را شرح می‌دهد. نسخه ۷ برای پیروی از دستور و تولید پاسخ ساخت‌یافته آماده شده است [۲۸] Instruct غیرتعبیه‌ای دارد و نسخه.

در پژوهش حاضر بدون تنظیم دقیق و با وزن‌های ثابت استفاده می‌شود. ثابت Qwen۲٫۵:۷B-Instruct نگه‌داشتن نسخه مدل، کوانتیزه‌سازی، دما و گزینه‌های تولید در دو روش ضروری است؛ زیرا در غیر این صورت اختلاف خروجی می‌تواند به تغییر مدل یا تنظیمات تولید نسبت داده شود، نه محل اعمال قاعده. مقاله برای تصمیم استفاده GPT-٤o-mini برای تعبیه و Qwen۳-Embedding-۸B از Hybrid Prompting کرده است و از نظر نقش مدل با این پژوهش تفاوت دارد [۳].

۲-۲-۹ مهندسی پرامپت، یادگیری درون‌متنی و خروجی

پرامپت شامل تعریف وظیفه، زمینه دامنه، معنای برچسب‌ها، نمونه‌های احتمالی و قرارداد خروجی است. پنج پرسش درباره نوع پرامپت، اندازه پنجره، دانش انسانی، مقایسه با ChatGPT مبتنی بر LogGPT اثر پرامپت ساده، پرامپت دارای دانش دامنه LogPrompt. روش‌های عمیق و تبیین‌پذیری مطرح می‌کند [۲] و زنجیره استدلال را بر تجزیه و تشخیص بررسی می‌کند [۱، ۶].

Few-shot نمونه‌های برچسب‌خورده را در خود پرامپت قرار می‌دهد و به تغییر وزن مدل نیاز ندارد. Hybrid Prompting نمونه‌های عادی مرتبط را از پایگاه برداری بازیابی می‌کند [۱۲] و RAGLog نیز حساسیت OpsEval. نزدیک‌ترین نمونه عادی را همراه با شواهد رخداد و توالی در متن قرار می‌دهد [۳] in-context learning و chain-of-thought، self-consistency وظایف عملیات فناوری اطلاعات به را در یک معیار مستقل بررسی می‌کند [۲۵].

قرارداد خروجی پژوهش حاضر فقط {"label":"۱"} یا {"label":"۰"}، را معتبر می‌داند. متن اضافی امکان تعیین Ollama ثبت و کش نمی‌شود. مستندات INVALID ناقص یا برچسب دیگر با وضعیت JSON و استفاده از دمای صفر برای کاهش نوسان را بیان می‌کنند [۲۹]. اعتبارسنجی در برنامه JSON قالب یا طرح مستقل از پذیرش ظاهری پاسخ انجام می‌شود.

بازیابی، کش و کنترل هزینه استنتاج ۲-۲-۱۰

بازیابی و کش دو سازوکار متفاوت‌اند. بازیابی، نمونه یا دانش مرتبط را برای ساخت زمینه پرامپت پیدا می‌کند؛ از این الگو استفاده می‌کنند [۱۲، ۱۳]. کش، پاسخ معتبر یک ورودی پایدار AdaptiveLog و RAGLog و مدل‌های RAG کش را در کنار FlexLog را نگه می‌دارد تا در مشاهده تکراری مدل دوباره اجرا نشود؛ یادگیری ماشین به‌کار می‌گیرد [۱۰].

در پژوهش حاضر کش در سطح قالب نرمال‌شده است. کلید شامل نسخه پرامپت، نام و نسخه مدل، قالب نرمال‌شده و در صورت فعال بودن، فیلدهای مؤلفه، شدت و دسته است. بنابراین تصمیم یک مدل یا پرامپت قدیمی به اجرای جدید منتقل نمی‌شود. نرخ برخورد، عدم برخورد، کلیدهای یکتا و منبع تصمیم برای تحلیل جداگانه اثر کش ثبت می‌شوند.

کاهش هزینه فراخوانی تا ۷۳ LLM با استفاده از مدل کوچک و ارجاع موارد نامطمئن به AdaptiveLog نیز با درخت پیشوند و تجمیع پیام‌های مشابه، در LogParser-LLM درصد را گزارش می‌کند [۱۳]. انجام داده LLM با میانگین ۳٫۶ میلیون پیام در هر مجموعه به‌طور متوسط فقط ۲۷۲٫۵ فراخوانی LogPub است [۲۳]. این اعداد نشان‌دهنده اهمیت گزارش فراخوانی و زمان در کنار معیارهای طبقه‌بندی هستند.

استنتاج محلی با Ollama ۲-۲-۱۱

Ollama پیش‌فرض API اجرای محلی مدل و دسترسی برنامه‌نویسی از طریق localhost:۱۱٤۳٤/api را می‌تواند زمان کل، زمان بارگذاری، شمار توکن پرامپت و خروجی و مدت ارزیابی API فراهم می‌کند. پاسخ هر بخش را گزارش کند [۲۹]. ثبت این داده‌ها امکان تفکیک زمان بارگذاری مدل از زمان پردازش پرامپت و تولید پاسخ را فراهم می‌سازد.

استقرار محلی از ارسال لاگ به سرویس خارجی جلوگیری و کنترل نسخه مدل را ممکن می‌کند. مطالعه De la Cruz Cabello و همکاران نیز LogBERT را برای محیط دارای محدودیت کاربرد واقعی محرمانگی به‌صورت درون‌سازمانی مستقر کرده است [۲۰]. تفاوت آن مطالعه با پژوهش حاضر در نوع مدل در این پژوهش بدون تنظیم Qwen عادی آموزش می‌بیند، درحالی‌که syslog و آموزش است: مدل آن‌ها روی اختصاصی استفاده می‌شود.

در پژوهش حاضر Prompt-only و Hybrid دو معماری ۲-۲-۱۲

پس از استخراج فیلدها، حذف برچسب و نرمال‌سازی، کش بررسی می‌شود. اگر کلید Hybrid، در روش برای guard موجود نباشد، مجموعه محدود قواعد قطعی با اطمینان بالا اعمال می‌شود. فقط نمونه‌هایی که ارسال می‌شوند. پاسخ معتبر در کش قرار می‌گیرد و منبع تصمیم به‌صورت Qwen آن‌ها تصمیم ندارد به llm یا guard، cache ثبت می‌شود.

قطعی غیرفعال است و دانش دامنه در پرامپت قرار می‌گیرد. نرمال‌سازی guard، Prompt-only در روش و ترتیب داده ثابت می‌مانند. در نتیجه متغیر اصلی مقایسه محل اعمال دانش JSON کش، نسخه مدل، قرارداد تفاوت دارد؛ مقاله [۳] شواهد سطح Hybrid Prompting در این پژوهش با Hybrid قاعده است. اصطلاح رخداد، توالی و نمونه بازیابی‌شده را در یک پرامپت ترکیب می‌کند و تصمیم آن در سطح توالی است.

معیارهای ارزیابی ۲-۲-۱۳

خط عادی با برچسب FP ،خط عادی درست TN ،ناهنجاری درست شناسایی‌شده TP ،در طبقه‌بندی دودویی سهم Precision ،نسبت کل تصمیم‌های درست Accuracy .ناهنجاری ازدست‌رفته است FN ناهنجار و و Precision میانگین هارمونیک F۱ سهم ناهنجاری‌های کشف‌شده و Recall ،پیش‌بینی‌های ناهنجار درست است [۲، ۳، ۱۸] Recall.

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN)$$

$$\text{Precision} = TP / (TP + FP) \quad \text{Recall} = TP / (TP + FN) \quad F۱ = ۲ \times \text{Precision} \times \text{Recall} / (\text{Precision} + \text{Recall})$$

گزارش شود. نرخ F۱ و Recall ،Precision باید همراه Accuracy ،BGL به‌علت نامتوازن بودن منبع تصمیم و نرخ برخورد کش نیز برای این پژوهش ،LLM زمان پاسخ، تعداد فراخوانی ،INVALID با بهبود خود طبقه‌بند guard ضروری‌اند. این شاخص‌ها مانع آن می‌شوند که کاهش فراخوانی ناشی از کش یا اشتباه شود.

انواع الگوهای ناهنجاری در لاگ ۲-۲-۱۴

ناهنجاری رخدادی زمانی رخ می‌دهد که خود یک پیام یا قالب، مستقل از پیرامون آن، نشانه وضعیت غیرعادی باشد. وجود عبارت خطا، توقف مؤلفه، خرابی سخت‌افزار یا کد بازگشتی غیرمنتظره می‌تواند از این نوع باشد. بر همین سطح تمرکز می‌کند و احتمال بازسازی توکن‌های یک پیام را به امتیاز خط تبدیل می‌نماید ADALog طبقه‌بندی خطبه‌خط پژوهش حاضر نیز ابتدا برای این سطح از ناهنجاری تعریف شده است .[۲۲].

،ناهنجاری ترتیبی از کنار هم قرارگرفتن رخدادهایی ایجاد می‌شود که ممکن است هرکدام به‌تنهایی عادی باشند LogBERT با پیش‌بینی رخداد بعدی و DeepLog .اما ترتیب یا فاصله آن‌ها با الگوی عادی سازگار نباشد نیز اطلاعات Hybrid Prompting .با یادگیری زمینه دوطرفه، این نوع انحراف را هدف می‌گیرند [۱۹، ۲۱] توالی را همراه رخدادهای دیده‌نشده در پرامپت قرار می‌دهد [۳].

ناهنجاری کمّی به تغییر غیرعادی تعداد تکرار یک یا چند قالب در یک پنجره مربوط است. روش‌های مبتنی بر بردار شمارش و برخی مدل‌های ترکیبی، فراوانی رخداد را در کنار ترتیب یا معنا استفاده می‌کنند. سه نوع بازنمایی شمارشی، ترتیبی و معنایی را به‌صورت همزمان وارد آشکارساز می‌کند [۱۷] LogLAA. نیست و این محدودیت باید هنگام Qwen در تصمیم خطمحور این پایان‌نامه، شمارش پنجره ورودی مستقیم تحلیل خطا در نظر گرفته شود.

،ناهنجاری معنایی زمانی مطرح می‌شود که واژه‌ها یا مقادیر یک پیام از نظر نحوی مشابه نمونه عادی باشند استفاده LogLLM و RAGLog ،LogPrompt اما مفهوم آن‌ها وضعیت نامطلوبی را بیان کند. مطالعات از بازنمایی زبانی را برای بازیابی یا درک چنین تفاوتی بررسی کرده‌اند [۱، ۴، ۶، ۱۲]. حفظ صفر و غیرصفر در نرمال‌سازی پایان‌نامه برای جلوگیری از حذف بخشی از همین معنای عملیاتی انجام می‌شود.

لاگ ناپایدار به تغییر قالب، واژگان یا توزیع رخدادها پس از به‌روزرسانی نرم‌افزار یا تغییر پیکربندی اشاره این مسئله را با انتقال نسخه، مدل زبانی FlexLog و LogLLM ،LOGEVOL روی GPT دارد. مطالعات در پایان‌نامه به‌عنوان یک مجموعه ثابت ارزیابی BGL .یا ترکیب مدل‌ها بررسی کرده‌اند [۴، ۱۰، ۱۱] می‌شود و آزمون انتقال میان نسخه‌های نرم‌افزار خارج از دامنه است.

چالش‌های روش‌شناختی و اجرایی ۲-۲-۱۵

سهم پیام‌های ناهنجار حدود ۷٫۳۴ درصد BGL نامتوازن بودن کلاس‌ها یکی از چالش‌های اصلی است. در Accuracy است [۲، ۱۹]. در چنین شرایطی، مدلی که بیشتر نمونه‌ها را عادی پیش‌بینی کند ممکن است Hybrid و BERT-Log ناهنجاری پایین بماند. به همین دلیل، مطالعات Recall ظاهراً بالایی داشته باشد، اما یا به‌جای اتکای صرف به آن Accuracy را در کنار F۱ و Recall، Precision معیارهای Prompting گزارش می‌کنند [۳، ۱۸].

چالش دیگری است. اگر پارامتر به‌عنوان کلمه ثابت یا کلمه ثابت به‌عنوان پارامتر parser انتقال خطای برای سنجش LogParser-LLM و LogPPT. تشخیص داده شود، قالب نادرست به مدل پایین‌دستی می‌رسد و معیارهای دانه‌بندی را به‌کار می‌گیرند [۲۳، ۲۴] Parsing Accuracy، Group Accuracy. این مسئله در پایان‌نامه، قواعد نرمال‌سازی باید جدا از کیفیت طبقه‌بندی ممیزی شوند.

نشت داده می‌تواند از ورود برچسب به متن، هم‌پوشانی نمونه‌های آموزش و آزمون یا تکرار قالب یکسان در داده‌های یکتای عادی را میان آموزش، تعیین آستانه و آزمون جدا می‌کند ADALog. دو بخش ایجاد شود قبل از ساخت پرامپت حذف می‌شود و کش فقط در محدوده همان اجرای BGL در این پایان‌نامه برچسب [۲۲]. تعریف‌شده و با کلید نسخه‌بندی‌شده به‌کار می‌رود.

پاسخ مولد ممکن است از قرارداد خروجی پیروی نکند یا میان اجراها تغییر کند. مطالعات پرامپتی نشان می‌دهند متن دستور و درخواست تبیین بر نتیجه اثر دارد [۱، ۲، ۶، ۲۵]. استفاده از دمای صفر، طرح JSON، در پایان‌نامه برای قابل‌اندازه‌گیری کردن این رفتار انجام می‌شود INVALID اعتبارسنجی برچسب و ثبت [۲۹].

از مدل می‌خواهد دلیل تصمیم را تولید کند LogPrompt. تبیین‌پذیری در منابع به چند معنا به‌کار رفته است پس از آشکارسازی LogLAA تشخیص و توضیح را هم‌زمان آموزش می‌دهد [۷]؛ و LLM-LADE [۱۰، ۶]؛ برای توضیح استفاده می‌کند [۱۷]. خروجی پایان‌نامه عمداً دودویی است، بنابراین تبیین در سطح LLM از ثبت منبع تصمیم، قالب نرمال‌شده و قاعده فعال‌شده دنبال می‌شود، نه تولید شرح آزاد.

محرمانگی، بازتولیدپذیری و هزینه نیز به هم مرتبط‌اند. سرویس تجاری امکان تغییر نسخه مدل و محدودیت نرخ دارد، درحالی‌که استقرار محلی کنترل بیشتری بر مدل و داده فراهم می‌کند [۲، ۲۰]. بااین‌حال، استقرار و پارامترهای تولید نیاز دارد تا نتیجه قابل بازتولید Ollama محلی به ثبت سخت‌افزار، کوانتیزه‌سازی، نسخه باشد [۲۸، ۲۹].

پیشینه تحقیق ۲-۳

دامنه و شیوه مرور مطالعات ۲-۳-۱

پیشینه این فصل شامل تمام ۲۷ فایل مقاله موجود در پوشه پژوهش است. مقالات براساس نقش اصلی روش در شش دسته قرار گرفته‌اند: روش‌های ترتیبی و پیش‌آموزش‌دیده؛ آموزش و تنظیم LLM؛ بازیابی و پرامپت؛ کنترل هزینه؛ تجزیه لاگ؛ و مرور، معیار، مدل پایه یا استقرار واقعی. این دسته‌بندی با محورهای مطرح‌شده برای تشخیص ناهنجاری [۲۷] سازگار است LLM در مرور نظام‌مند لاگ [۱۴] و مرور.

هستند. مطابق درخواست پوشش LogPrompt دو فایل [۱] و [۶] نسخه‌های هم‌پوشان یک خط پژوهشی با نام همه مقالات، هر دو به‌صورت مستقل مرور و در جدول درج شده‌اند؛ بااین‌حال هنگام جمع‌بندی شواهد به‌عنوان آشکارساز مستقیم لاگ [۲۶] YuLan و [۲۵] OpsEval دو روش متفاوت شمرده نمی‌شوند. همچنین مقالات در عملیات و معماری مدل متن‌باز استفاده می‌شوند LLM نیستند، اما به‌ترتیب برای مبانی ارزیابی.

روش‌های ترتیبی و مدل‌های زبانی پیش‌آموزش‌دیده ۲-۳-۲

مقاله ۲۱ - DeepLog: Du و شبکه LSTM را مانند یک توالی زبان در نظر گرفتند و برای پیش‌بینی رخداد بعدی از روی تاریخچه رخدادهای عادی آموزش دادند. اگر رخداد واقعی در مجموعه رخدادهای محتمل مدل قرار نگیرد، توالی به‌عنوان مشکوک گزارش می‌شود. مقاله علاوه بر تشخیص، به‌روزرسانی افزایشی مدل و ساخت جریان‌های اجرایی برای کمک به عیب‌یابی را توضیح می‌دهد. [۲۱]

ارزیابی DeepLog روی داده HDFS و OpenStack نشان داد مدل ترتیبی می‌تواند وابستگی رخدادها را تشکیل کلید لاگ و توالی معتبر نیاز دارد، parser بهتر از روش‌های آماری مبنا ثبت کند [۲۱]. این روش به هدف آن تصمیم مستقل برای هر خط نیست. در پژوهش حاضر، ترتیب رخداد به‌عنوان ورودی مستقیم مدل و خط پایه مفهومی خانواده ترتیبی محسوب DeepLog است؛ بنابراین BGL استفاده نمی‌شود و واحد تصمیم خط پروتکل، نه روش هم‌پروتکل می‌شود.

مقاله ۱۹ - LogBERT: Guo، Yuan و Wu چارچوب خودنظارتی LogBERT را با دو وظیفه طراحی کردند: پیش‌بینی کلید لاگ ماسک‌شده و کمینه‌سازی حجم ابرکره بازنمایی توالی‌های عادی. زمینه دوطرفه باعث می‌شود بازنمایی هر رخداد از قبل و بعد توالی استفاده کند و وظیفه ابرکره نیز توالی‌های عادی BERT را در فضای بازنمایی به یکدیگر نزدیک می‌سازد. [۱۹]

LogBERT روی HDFS، BGL و Thunderbird ارزیابی شده است. برای BGL پنجره پنج‌دقیقه‌ای نیز BGL ساخته شده و مقاله از حدود پنج هزار توالی عادی برای آموزش استفاده می‌کند؛ میانگین طول توالی رخداد گزارش شده است [۱۹]. وابستگی به آموزش روی داده عادی و ساخت پنجره تفاوت اصلی آن با ۵۶۲ Qwen بدون تنظیم و تصمیم خطبه‌خط در پژوهش حاضر است.

مقاله ۱۸ - BERT-Log: Chen و Liao بازنمایی و مدل را همراه یک BERT پیام‌های تجزیه‌شده را با شناسه گره BGL توالی‌ها با شناسه بلوک و در HDFS لایه تمام‌متصل برای طبقه‌بندی تنظیم کردند. در اندازه پنجره و گام ساخته می‌شوند. مدل از اطلاعات متن و زمینه توالی برای تفکیک رفتار عادی و ناهنجار استفاده می‌کند. [۱۸]

مقاله F۱ را در پروتکل خود گزارش می‌کند BGL و ۹۹٫۴ درصد برای HDFS برابر ۹۹٫۳ درصد برای وابسته است و مستقیماً با تشخیص خطمحور BGL و پنجره parser، این نتیجه به آموزش نظارت‌شده [۱۸]. برای پژوهش حاضر نشان‌دادن ظرفیت بازنمایی BERT-Log قابل مقایسه نیست. اهمیت fine-tuning بدون ضرورت گزارش دقیق پروتکل پنجره است BGL زبانی در.

مقاله ۸ - LogFiT: Almodovar و همکاران LogFiT را بر پایه مدل زبانی تنظیم‌شده با وظیفه مدل‌سازی توکن ماسک‌شده ارائه کردند. مدل فقط روی لاگ‌های عادی آموزش می‌بیند و در آزمون، توکن‌های پیام یا پیش‌بینی باشد، شواهد ناهنجاری افزایش top-k توالی را پیش‌بینی می‌کند. اگر توکن واقعی خارج از مجموعه می‌یابد. [۸]

بررسی شده و هدف آن کاهش نیاز به نمونه ناهنجار Thunderbird و BGL، HDFS روی LogFiT برچسب‌خورده است [۸]. با وجود استفاده از مدل زبانی، روش مولد دستورمحور نیست و به تنظیم روی لاگ را ثابت نگه می‌دارد و پاسخ دودویی را از Qwen مقصد نیاز دارد. در مقابل، پژوهش حاضر وزن‌های پرامپت دریافت می‌کند.

مقاله ۲۲ - ADALog: Pospieszny و همکاران چارچوبی تطبیقی و بدون نظارت برای تحلیل پیام‌های خام روی پیام‌های عادی تنظیم می‌شود؛ احتمال MLM با وظیفه DistilBERT. و مستقل از توالی پیشنهاد کردند بازسازی توکن‌ها به امتیاز سطح پیام تبدیل و آستانه با صدک خطاهای داده عادی تعیین می‌گردد. روش به parser یا پیوستگی توالی وابسته نیست. [۲۲]

ارزیابی و اثر نرخ ماسک، پاک‌سازی و موقعیت توکن‌ها Spirit و Thunderbird، BGL روی ADALog را با مطالعات حذف مؤلفه بررسی کرده است [۲۲]. نزدیک‌بودن واحد تصمیم آن به سطح خط برای این و امتیاز fine-tuning DistilBERT به ADALog: پایان‌نامه مهم است، اما سازوکار تصمیم متفاوت است و کش قالب استفاده می‌کند JSON، دستورمحور Qwen بازسازی متکی است، درحالی‌که پژوهش حاضر از.

۲-۳-۳ آموزش، تنظیم پارامترکارا و انتقال LLM

مقاله ۵ - Log Anomaly Detection via GPT: Han، Yuan و Trabelsi مدلی با نام LogGPT را برای پیش‌بینی رخداد بعدی آموزش می‌دهد و سپس با یادگیری تقویتی، هدف تولید را GPT ارائه کردند که به تصمیم تشخیص ناهنجاری نزدیک می‌کند. طراحی روش از توان مدل خودبازگشتی برای یادگیری توالی رخداد و از بازخورد پاداش برای اصلاح مرز تصمیم استفاده می‌کند. [۵]

ارزیابی مقاله روی HDFS، BGL و Thunderbird انجام شده است [۵]. اشتراک نام آن با مقاله [۲] نباید به یکسان‌انگاری روش‌ها منجر شود؛ مقاله [۵] روش آموزش و یادگیری تقویتی است، ولی مقاله [۲] را با پرامپت بررسی می‌کند. پژوهش حاضر از نظر نبود آموزش به مقاله [۲] نزدیک‌تر است ChatGPT.

مقاله ۴ - LogLLM: Guan و همکاران LogLLM طراحی کردند با سه جزء اصلی: BERT برای به‌عنوان طبقه‌بند. آماده‌سازی LLaMA برای نگاشت بازنمایی و projector استخراج بردارهای معنایی، یک و فضای BERT با قواعد منظم انجام می‌شود و فرایند آموزشی چندمرحله‌ای برای هم‌ترازکردن بازنمایی به‌کار می‌رود. [۴] LLaMA

و در سناریوهای لاگ ناپایدار ارزیابی شده است Thunderbird و Liberty، BGL، HDFS روش روی می‌تواند اطلاعات معنایی و تصمیم را LLM کوچک‌تر با encoder نشان می‌دهد ترکیب LogLLM. [٤] است. پژوهش حاضر بدون آموزش، مستقیماً قالب LLaMA و projector جدا کند، اما نیازمند آموزش می‌دهد Qwen نرمال‌شده را به.

مقاله ۷ - LLM-LADE: Zhang و همکاران چارچوب سه‌مرحله‌ای شامل افزایش داده، تنظیم پارامترکارا LoRA با Llama۳-۸B برای تولید نمونه و few-shot با پرامپت GPT-٤. و بهینه‌سازی برخط ارائه کردند برای آشکارساز استفاده می‌شود. مدل علاوه بر برچسب ناهنجاری، توضیح متنی تولید می‌کند و دانش حاصل از نمونه‌های جدید در یک پایگاه دانش افزایشی نگه‌داری می‌شود. [۷]

تشخیص و توضیح را در یک فرایند مشترک قرار می‌دهد و سازوکار برخط را برای LLM-LADE سازگاری با تغییرات پیشنهاد می‌کند [۷]. در پژوهش حاضر خروجی برای جلوگیری از ابهام فقط برچسب وجود ندارد. این تفاوت دامنه LoRA است و توضیح تولید نمی‌شود؛ همچنین بهینه‌سازی برخط یا JSON سنجش را به کیفیت دودویی و هزینه استنتاج محدود می‌کند.