Descriptive Adverse Drug Reaction Prediction via Hybrid Retrieval and Low-λ MMR Re-ranking

Anonymous ACL submission

Abstract

Adverse drug reaction (ADR) prediction involves two complementary tasks: structured inference (classifies the occurrence of specific ADR outcomes) and unstructured inference (generates narrative descriptions of those outcomes using preferred terminology). Existing methods predominantly focus on structured ADR prediction, and use supervised learning based on hand-engineered features yielding non-generalized detection, while unstructured ADR narrative prediction remains largely unexplored. In this work, we propose a novel ADR prediction framework that jointly predicts structured and unstructured ADR outcomes. Our framework leverages large language model (LLM) for semantic representation and ADR knowledge retrieval in a three-stage pipeline to simultaneously predict both structured and unstructured outcomes in a generalized manner. First, we fine-tune an ADR-specific embedding model on top of a benchmark foundation model to align the embedding space with domain-specific ADR terminology. Second, we construct a novel hybrid retrieval pipeline that integrates BM25 lexical matching with dense vector similarity search to ensure high recall. Third, we apply a Maximal Marginal Relevance (MMR) re-ranking strategy to balance relevance and diversity. Evaluation on three held-out FDA Adverse Event Reporting System (FAERS) quarterly test sets demonstrates that our method achieves a two-fold improvement in top-1 classification accuracy for structured ADR prediction, and a 32% improvement in recall for unstructured narrative descriptions compared with baselines.

1 Introduction

Adverse drug reactions (ADR) pose a significant risk to patient safety and have been consistently reported as a leading cause of death (Makary and Daniel, 2016; American Society of Pharmacovigilance, 2025). The scale of this risk is evident in large-scale pharmacovigilance systems such as the U.S. FDA's Adverse Event Reporting System (FAERS) which recorded more than 1.25 million serious ADR cases in 2022 including nearly 175,000 fatalities accessed in 2024 (Kommu et al., 2024). Figure 1 depicts an example of the ADR prediction task, consisting of the ADR structured label (*e.g.*, "Death") and unstructured narratives describing the label in preferred clinical terminology, given a patient's information and treatment. As the number of approved and experimental drugs continues to increase, early and accurate prediction of ADR outcomes has become increasingly critical for minimizing patient risk and improving clinical outcomes (Mullard, 2025).

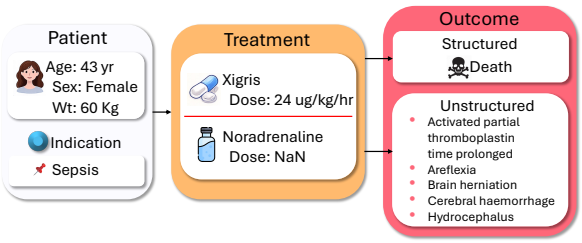

Figure 1: An Example of Structured and Unstructured ADR Outcomes

Recent reports have discussed the need for methods that accurately predict structured ADR labels, as well as extracting narrative descriptions aligned with preferred clinical terminology that can easily inform practitioners about potential ADRs (Golder et al., 2025). However, existing approaches primarily focus on structured ADR label prediction that are based on traditional machine learning methods with limited generalization capabilities. Moreover, there is a lack of approaches for predictive or explanatory unstructured ADR outcomes (Deimazar and Sheikhtaheri, 2023). To date, no unified framework has effectively integrated structured ADR prediction with unstructured narrative extraction within a single pipeline. This disconnect limits the interpretability and practical use cases of ADR pre-

diction in real-world pharmacovigilance settings, motivating the need for hybrid approaches that jointly address structured outcome inference and unstructured narrative generation.

In this study, we propose a large language model (LLM) based retrieval-based pipeline for the prediction of structured and unstructured ADR outcomes in three stages. In the first stage, we leverage the FAERS system to compile a dataset out of quarterly ADR samples and fine-tune an ADR-specific embedding model on top of a open-source benchmark foundation model BGE-M3 to align the embedding space with domain-specific ADR terminology. In the second stage, we develop a novel hybrid retrieval method by combining lexical matching via BM25 and vector-based semantic search, which leverages the high precision of lexical matching and the high recall of semantic similarity to deliver accurate performance across diverse query types (Wang et al., 2025; Lamsiyah et al., 2023). In the third stage, rather than fine-tuning a costly re-ranking model, we optimize a low λ of Maximal Marginal Relevance (MMR) that emphasizes diversity, thereby making it beneficial for ADR prediction where capturing a broad spectrum of ADR outcomes is desirable. (Xia et al., 2015; Coppolillo et al., 2024) at inference time.

Our experiments show that our framework significantly outperforms open-source LLM baselines: for structured ADR classification, our approach achieves 63.3% prediction accuracy on Top-1 retrieval and 86.2% on Top-5 retrievals, substantially exceeding GPT-OSS-20B (37.5%) and GPT-OSS-120B (38.3%) by approximately two-fold; for unstructured, our framework outperforms baselines with a 32% recall improvement, a two-fold gain with top-1 retrieval, and four-fold gain with top-5 retrievals for unstructured narrative prediction. These results demonstrate strong robustness and effectiveness of the proposed model-free, retrieval-based solution for ADR predictions.

Our key contributions can be summarized as follows:

- Unlike existing works, our work focuses on the extraction of unstructured narrative descriptions in preferred clinical terminology, in addition to the structured ADR labels in a generalized manner by leveraging LLMs.
- We propose a novel, three-stage framework[1] for generalized detection of ADR outcomes consisting of fine-tuning an ADR-specific embedding model, a hybrid retrieval strategy, and an MMR-driven re-ranking.
- Our systematic assessment on both structured ADR classification and unstructured narrative extraction demonstrates substantial performance gains over baseline approaches.

[1]Our framework will be open-sourced.

2 Related Work

Existing ADR prediction research has largely concentrated on two separate tasks. The first focuses on the extraction of ADR-related information from free-text sources such as clinical notes, patient reports, and spontaneous narrative reports. These tasks are typically addressed through natural language processing (NLP) techniques such as named entity recognition (NER), relation extraction, and text classification have been developed to detect ADR from unstructured data like social media, clinical notes, or pharmacovigilance narratives (Sarker and Gonzalez, 2015; Ben Abacha et al., 2015; Tiftikci, 2019) More recent studies have incorporated deep learning models and contextual language representations to improve robustness against linguistic variability and noise in free-text data (Haq et al., 2023). While effective at detecting ADR mentions, these methods primarily support information extraction rather than predictive inference or narrative generation. In contrast, the second line of task involves structured ADR occurrence classification using supervised learning models that heavily rely on the specifically engineered features as structured input and annotated ADR outcomes, thereby limiting generalization beyond predefined inputs (Farnoush et al., 2024; Choudhury et al., 2020; Bresso et al., 2021).

Advances in language model fine-tuning, particularly BERT and its biomedical variants, have improved the representation of ADR-related semantics and enhanced downstream prediction tasks (Guo and Choo, 2025). LLMs can also be leveraged to generate unstructured ADR narratives in preferred clinic terminology. This shift enables more effective exploitation of rich, unstructured healthcare data, despite the inherent challenges posed by complex domain knowledge and heterogeneous data structures.

3 Datasets

ADR Data Extraction, Transformation, and Loading (ETL). FAERS releases a quarterly data

dump as a compressed archive that contains a set of plain-text files together with PDF files for processing instructions. For each quarter, the archive includes five core tables that we use to reconstruct individual ADR cases identified by a unique caseID.

The ETL Workflow includes 1) Data extraction - ADR cases are extracted from the zip archive is unpacked. Each text file is read line-by-line and parsed according to separate defined in FDA's specifications. 2) Data transformation – Key alignment–records from the five tables are joined on caseID to form a single, patient-centric record. 3) Outcome handling – the ADR outcome (OUTC) provides a categorical label for seven ADR outcomes (Table 1). Because the source data are incomplete, many reports lack an outcome code. These incomplete records are retained for model training (they still contribute valuable drug-exposure and reaction information) but are excluded from evaluation of classification accuracy. Meanwhile, a unique outcome code was assigned to a case given the highest severity class. 4) Missing-value imputation – When a field is missing (e.g., dose frequency), we insert a sentinel value (NULL) that the downstream model can learn to ignore. 5) Data formatting – every ADR case is constructed as instruction and outcome pair both in jsonl format (Section 4.1) and written to a column-archetype Parquet store, enabling efficient downstream vector-indexing and BM25 indexing.

Preparation of training includes splitting the dataset into training and testing sets. The training set includes quarterly releases from 2004Q1 to 2024Q2. To maximize exposure to diverse drug-reaction patterns, missing outcome codes ADR cases are included. The testing set is prepared for performance evaluation covering quarterly releases from 2004Q3 through 2025Q1. We restrict to ADR cases that possess a complete outcome label (a unique OUTC present) and a non-empty reaction narrative, ensuring a fair comparison of structured-classification and unstructured-generation metrics.

By systematically extracting and merging the 5 core FAERS tables, the ETL pipeline yields a high-quality, patient-level ADR corpus that serves as the foundation for our hybrid RAG system.

Code	Description
DE	Death
LT	Life-Threatening
HO	Hospitalization-Initial or Prolonged
DS	Disability
CA	Congenital Anomaly
RI	Required Intervention to Prevent Permanent Impairment/Damage
OT	Other Serious (Important Medical Event)

Table 1: Classification of Seven Structured ADR Outcomes

4 Our Approach

4.1 Construction of ADR Instruction and Outcome

Firstly, structured ADR tabular data are prepared from quarterly FAERS releases dating back to 2004. Each ADR case comprises patient demographics, drug administration details, medication indications, ADR narratives expressed in preferred terms, and ADR outcome classifications. Each case is formulated as an instruction-outcome pair. The instruction component consists of 3 distinct fields that collectively capture the semi-structured information associated with each case: 1) Patient demographics, including age, age code, gender, weight, and weight unit; 2) Drug administration, detailing the drug name, route, dose, frequency, formulation; and 3) Medication indications, describing therapeutic purpose for which the drug was prescribed. We refer to this three-component construct as the ADR triplet extraction hereafter. To preserve the column-wise relationships inherent in the original FAERS data while enabling downstream information-retrieval and generative tasks, we store each instruction–outcome pair in a line-delimited JSON (JSONL) file. Each line contains a self-contained JSON object with the following schema:

{""patient": ""age": "43.0", "age_cod": "yr", "gndr_cod": "f", "wt": "60.0", "wt_cod": "kg"","treatment": ""drug-name": "xigris", "route": "other", "dose": "24 ug/kg/hr/other"; "drugname": "nora-drenaline", "route": nan, "dose": nan", "indi_pt":"sepsis"}

Each ADR case outcome comprises 2 complementary predictions. The unstructured component

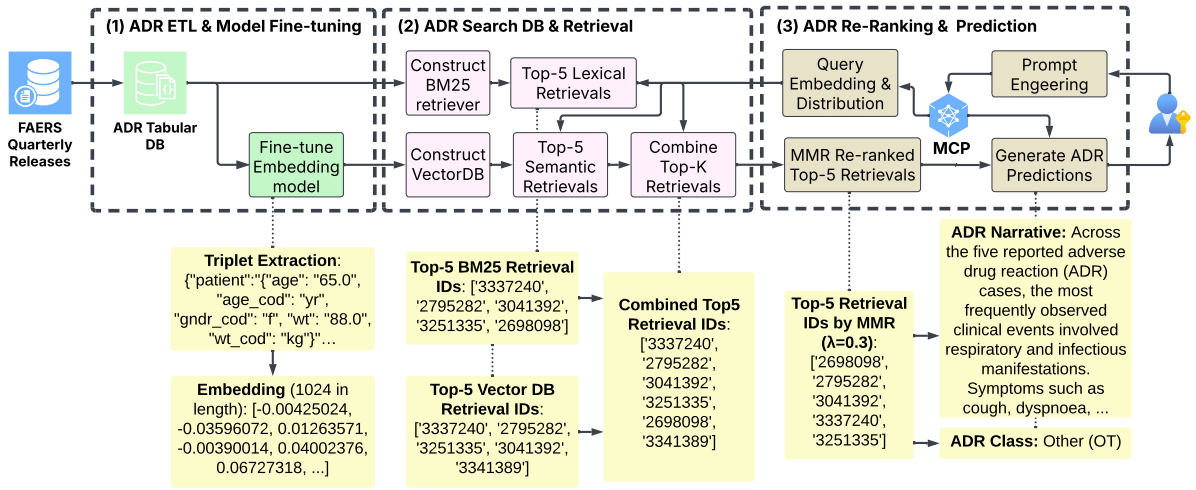

Figure 2: Overview of 3-stage framework: (1) ADR ETL & Model Fine-tuning, (2) ADR Search DB construction & Retrieval, and (3) ADR Re-ranking & Prediction. The example illustrates how structured triplets are embedded, top-k lexical and semantic samples are combined, followed by MMR-based re-ranking to predict both unstructured narrative and structured ADR class.

is an ADR narrative represented as a free-text description expressed in MedDRA preferred terms, while the structured component is an outcome code encoded as a single categorical label corresponding to the most severe outcome reported for the case. When multiple outcome codes are present, the one with the highest severity ranking is retained.

Just as the instruction is stored as a line-delimited JSON object, the ADR outcome is encoded in the same JSONL format. An illustrative example is shown below:

{"pt": "activated partial thromboplastin time prolonged; areflexia; brain herniation; cerebral haemorrhage; hydrocephalus; platelet count decreased; prothrombin time prolonged; pupil fixed", "uni_code": "DE"}

The JSONL representation is the query format for both BM25/vector hybrid retrieval stage and subsequent generation step in the hybrid RAG pipeline.

4.2 Fine-tuning Embedding Model

RAG pipeline depends on high-quality retrieved passages to guide the downstream LLM in ADR prediction. Instead of fine-tuning an entire open-source LLM for the ADR task (a computationally intensive and difficult process to keep up with new quarterly FAERS releases), we adopt a more pragmatic strategy by fine-tuning a sentence-level embedding model that can be updated quickly and yields hallucination-free evidence for the LLM (Kraišniković et al., 2025; Ng et al.).

In a vector-search setting, each ADR triplet extraction is encoded into a fixed-length dense vector; similarity between a query and candidate records is then computed with a normalized dot-product (i.e., cosine similarity). Because no publicly available embedding model has been specifically trained for ADR data, we first evaluated the state-of-the-art models on the Massive Text Embedding Benchmark (MTEB) [2] and selected the open-source BAAI/bge-m3 model [3] as the foundation model. BGE-M3 can ingest up to 8,192 tokens and produces 1,024-dimensional embeddings, which is sufficient to capture the rich information contained in an ADR triplet extraction.

Supervised fine-tuning used 500k randomly sampled ADR cases reported from the FAERS corpus (2004Q1–2024Q2), with 100k (20%) ADR cases set aside for validation. Early-stopping and hyper-parameter tuning are configured on single NVIDIA H100 (80GB) GPU. The whole training run over 3 epochs for a total of 150k optimization steps. The objective is to optimize the contrastive soft-hit learning by maximizing similarity for true instruction-outcome pairs while minimizing it for mismatchs. After each epoch the model was evaluated on the validation set using 3 retrieval metrics calculated on the top-K results: NDCG, Mean Re-

[2] https://huggingface.co/spaces/mteb/leaderboard

[3] https://huggingface.co/BAAI/bge-m3

ciprocal Rank (MRR), and Mean Average Precision (MAP). The 3rd epoch checkpoint achieved the highest scores across all three metrics and was selected as the final ADR-specific embedding model. The entire fine-tuning completed in 7 hours, producing an ADR-adapted vector space that enables fast, up-to-date retrieval of ADR triplet extractions for semantic search.

4.3 Construction of Vector Database

To support fast similarity search over millions of ADR triplet embeddings, we store the vectors in a LanceDB database that provides Approximate Nearest-Neighbor (ANN) indexes, specifically a joint indexing approach with Inverted File Index, Hierarchical Navigable Small World, and Product Quantization (IVF-HNSW-PQ) [4] with 256 partitions and 128 sub_vectors. This setting eliminates the need for exhaustive linear scans of the entire vector space and optimizes performance trade-offs.

All triplet extraction embeddings generated by the fine-tuned ADR embedding model (Section 4.2) are ingested into LanceDB as a persistent column-wise vector field. When a query is issued, the top-K most similar ADR cases were returned , ranked by the descending cosine similarity order. Because the BGE-M3 model outputs unit-norm embeddings (‖v‖=1), cosine similarity reduces to a simple dot product. Consequently, the retrieval step can be performed with a single inner-product operation, further accelerating the ANN search. The result is a ranked list of ADR cases that can be passed to the subsequent hybrid retrieval fusion and MMR re-ranking stages.

4.4 Construction of BM25 Retriever

Using the same training corpus of 500K instruction–response pairs for embedding model finetuning, we constructed a BM25-based lexical retrieval pipeline as the first-stage retriever of proposed hybrid RAG framework. Each instruction–outcome pair was indexed as a unified retrieval unit. The corpus was indexed using the BM25 ranking function, which scores relevance based on term frequency, inverse document frequency, and document length normalization. At inference time, inputs of ADR were processed using the same pipeline and matched against the BM25 index to retrieve the top 5 candidate ADR outcomes. This lexical retrieval stage provides high-precision matching for drug names and administration details, indication of medication in preferred terms, forming a robust candidate set to fuse with dense vector retrievals, MMR re-ranking, and generation components within the ADR-focused RAG pipeline.

4.5 Hybrid Retrieval Combination and MMR Re-ranking

To integrate lexical and semantic retrieval signals, we employed a hybrid retrieval strategy that combines sparse BM25 retrieval and dense vector similarity search. For each query, the top 5 ADR candidate cases were independently retrieved from the BM25 index and the dense vector database, yielding a fused candidate set after duplicate removal. The merged candidates were subsequently re-ranked using the MMR algorithm (Carbonell and Goldstein, 1998), which iteratively selects ADR candidates by balancing relevance to the query and diversity among the already selected results. Specifically, at each selection step, an ADR candidate d_i is chosen to maximize

$$MMR_{d_i} = \lambda sim(d_i, q) - (1 - \lambda) \max_{d_j \in S}(sim(d_i, d_j))$$

where q denotes the ADR query, S is the set of previously selected ADR cases, *sim()* represents the similarity function, and tunable $\lambda \in [0, 1]$ controls the trade-off between relevance and diversity. Higher values of λ prioritize query relevance, whereas lower values emphasize diversity by penalizing redundancy. By tuning λ, the MMR re-ranking step improves coverage of diverse and rare ADR cases while maintaining high relevance. The final re-ranked top 5 ADR candidates were then used as contextual inputs for downstream ADR prediction within the hybrid RAG pipeline.

4.6 Local LLM inference on OOT ADR

We deployed local Ollama server with Nvidia Spark miniserver to support on-premises inference with open-source LLMs, specifically GPT-OSS-20B and GPT-OSS-120B. The Ollama service runs as a containerized inference endpoint, offering a lightweight HTTP REST API for customized prompt-based ADR prediction. Running inference locally reduces external dependencies, and ensures reproducibility with fixed model versions for consistent evaluation. Due to the slow inference speed, we use 3,517 (3%) triple extractions randomly sampled ADR cases from OOT set between 2024Q3 and 2025Q1 as context in customized prompt to

[4] https://docs.lancedb.com/indexing

generate ADR predictions and return both unstructured ADR narratives in preferred term and structured ADR classification (See customized prompt for open-source LLM inference in Appendix 8.

4.7 Proxy Ground Truth of ADR

ADR cases reported to FAERS are not systematically reviewed by clinical pharmacologists, raising data-quality concerns (Veronin et al., 2020; Giunchi et al., 2023). Since manual verification millions of heterogeneous ADR cases is infeasible. We propose a proxy ground-truth method based on the semantic similarity between the predicted and reported ADR representations. Specifically, we encode the predicted ADR outcome (both unstructured and structured outcomes) and the reported ADR outcome using the fine-tuned ADR embedding model described in Section 4.2. The cosine similarity between the two embeddings is computed, and a threshold of 0.4 is used to determine the relevance: similarity ≥ 0.4 is labeled as correct (1), and similarity < 0.4 as incorrect (0). This similarity-based proxy provides a scalable, automated evaluation without exhaustive expert review while reflecting semantic alignment between predicted and reported ADR cases. It is employed as the evaluation target for the unstructured prediction evaluation reported in Section 5.4.

4.8 Agentic AI Orchestration for ADR prediction

To perforem inference with the hybrid pipelines, user inputs are first integrated and converted into a JSONL that encodes the triplet extraction from processed FEARS data. The query is then dispatched in parallel to a BM25 index for sparse lexical matching and to a dense vector DB for semantic search using fine-tuned bge-m3 embeddings. Each retrieval engine returns its top-5 results, yielding up to 10 candidate evidences. After de-duplication, the remaining retrievals are concatenated and passed to a MMR re-ranking module with $\lambda = 0.3$. The top-5 retrievals are serialized into a single JSON-formatted context and appended to the original query to form the prompt for downstream LLM inference. This assembled JSONL entry is sent to a locally hosted LLM via a REST API. To mitigate low-quality ADR predictions, the same cosine-similarity threshold (0.4) used during retrieval is applied, such that any of the top-5 retrievals falling below this threshold are flagged as low confidence.

To orchestrate the multi-step ADR prediction, an Model Context Protocol (MCP) server is implemented to coordinate the execution across agents, ensuring correct sequencing and preservation of contextual state throughout the pipeline. The system returns a final ADR summary along with provenance links, providing a transparent, step-by-step, evidence-based ADR prediction that integrates hybrid retrieval, diversity-aware MMR re-ranking, and a context-preserving agentic reasoning layer.

5 Evaluation

5.1 Evaluation of Finetuned Embedding Model

Compared with the base model, the fine-tuned bge-m3 embedding model yields consistent performance gains (Kusupati et al., 2024). Using the Enrichment Factor (EF)-defined as the ratio of the observed hit rate to the expected hit rate of a random baseline (1/100,000 for a Top-1 hit) - we evaluated performance on 20% of training set (500K randomly sampled ADR cases), comprising 100K randomly sampled ADR cases. The finetuned model achieves EF values well above random hit after just 3 training epochs with 150K fine-tuning steps.

Overall, the finetuned model enriches the top-1 cosine-similarity hit rate by a factor of 214× over random chance and maintains strong enrichment across higher top-K thresholds, including EF values of 178× at Top-3 and 160× at Top-5. Comparable gains are observed across other retrieval and accuracy metrics, demonstrating that embedding fine-tuning substantially amplifies semantic signal and retrieval effectiveness for large-scale ADR data.

5.2 Redundancy Analysis of Retrievals

To assess how MMR influences redundancy in the retrieved evidence, we computed the average pairwise cosine similarity among the top-5 items returned the mean of cosine similarity from queries under each retrieval pipeline. Because cosine similarity captures the degree of overlap between ADR triplet extraction embeddings, higher values indicate greater redundancy.

Figure 3 shows that the mean pairwise similarity is essentially identical for the BM25-only, vector-only, and the higher-λ ($\lambda = 0.5, 0.7$) MMR pipelines. In contrast, the low-$\lambda = 0.3$ behaves differently, such as Top-1 retrieval with 0.3 yields the highest similarity (0.457), confirming that

	Finetuned Metrics	EF
Cosine-Accuracy@1	0.00214	214
Cosine-Accuracy@3	0.00535	178.3
Cosine-Accuracy@5	0.008	160
Cosine-Precision@1	0.00214	214
Cosine-Precision@3	0.00178	59.3
Cosine-Precision@5	0.0016	32
Cosine-Recall@1	0.00214	214
Cosine-Recall@3	0.00535	178.3
Cosine-Recall@5	0.008	160
NDCG@10	0.00476	101.60
MRR@10	0.00666	227.38
MAP@100	0.00628	121.06

Table 2: Evaluation of Fine-tuned ADR Embedding Model

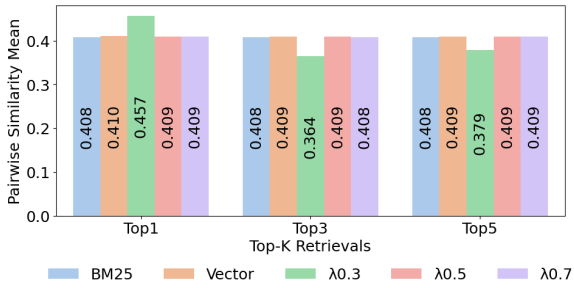

Figure 3: Relevance and Redundancy Analyses with MMR

the most-relevant document is preserved after re-ranking. However, Top-3 and Top-5 retrievals with the same $\lambda = 0.3$ setting produces substantially lower similarities (0.364 and 0.379, respectively), indicating that the subsequent items are more diverse and less redundant. Thus, operating MMR in a low-λ regime maintains relevance at the first rank while deliberately diversifying the remaining slots. This reduction in redundancy explains the improved NDCG and recall observed for hybrid RAG-based unstructured ADR prediction when $\lambda = 0.3$ is used in Section 5.4

5.3 Impact of MMR on Structured ADR Classification

We evaluate the impact of three MMR weighting parameters λ values of 0.3, 0.5 and 0.7 on the fused retrievals and compared them with two hybrid-retrieval BM25 and vector and open-source baselines for structured ADR prediction across 3 sequential OOT quarters.

We first assess the structured ADR classification accuracy by baseline open-source LLMs on 3% OOT data given the lengthy LLM inference. The baseline models achieve average 0.373 (GPT-OSS-20B) and 0.385 (GPT-OSS-120B) with minimal variance quarter-to-quarter variance.

Next we reported accuracies for Top-1, Top-3, and Top-5 retrievals produced by the two hybrid pipelines and by the 3 MMR-re-ranked sets (λ = 0.3, 0.5, 0.7) The results are summarized in Table 3. For top-1 performance (e.g. 24Q3@1, 24Q4@1, and 25Q1@1), both hybrid pipelines achieves near-perfect accuracy , reaching 100% for top-1 24Q3@1, 24Q3@3, and 24Q3@5. However, in later quarter (24Q4 and 25Q1), accuracy declines substantially, highlighting the necessity of continual updates of ADR cases from FAERS. Notably, the performance degradation in newer quarter is mitigated by MMR-based re-ranking. Specifically, Top-3 retrievals re-ranked with MMR (λ = 0.5) achieve the highest accuracy among all five pipelines for the 3rd OOT quarter (25Q1); Top-5 retrievals re-ranked with MMR ($\lambda = 0.3$) yield the best accuracy for the 2nd OOT quarter (24Q4); and Top-5 retrievals re-ranked with MMR (λ = 0.7) show the highest accuracy for the 3rd OOT quarter (25Q1).

Overall, all evaluated pipelines outperform the strongest baseline model, confirming that even a single, well-ranked retrieval from our pipeline can improve structured ADR classification accuracy from 39.7% to 100%. Given its superior performance on near-term data, the vector-based retrieval pipeline is recommended for ADR classification when the evaluation period is close to the training distribution. However, due to the diversity-promoting nature of MMR, its Top-1 accuracy is generally lower than that of pure BM25 or vector-based pipelines. Among all methods, pure vector search yields the highest accuracy for quarters closest to the training period, whereas MMR-re-ranked pipelines consistently outperform others in more temporally distant quarters, where data drift is more pronounced. Collectively, these findings demonstrate that MMR re-ranking not only improves ranking quality but also helps surface rare and future ADR cases that would otherwise be missed under temporal drift.

5.4 Impact of MMR on Unstructured ADR Prediction

To understand the impact, three λ settings on unstructured ADR predictions, we examine the recall and NDCG at the Top-1, Top-3, and Top-5 cut-offs

	BM25	Vector	λ0.3	λ0.5	λ0.7
24Q3@1	0.979	**1.000**	0.547	0.546	0.544
24Q4@1	0.476	**0.485**	0.417	0.417	0.420
25Q1@1	0.435	**0.448**	0.400	0.397	0.400
24Q3@3	0.995	**1.000**	0.846	0.843	0.843
24Q4@3	0.700	**0.716**	0.711	0.706	0.709
25Q1@3	0.675	0.691	0.692	**0.695**	0.694
24Q3@5	0.998	**1.000**	0.939	0.939	0.939
24Q4@5	0.797	0.810	**0.820**	0.816	0.816
25Q1@5	0.778	0.789	0.803	0.804	**0.806**

Table 3: Top-K Classification by Quarter

(Table 5). Using the proxy ground truth described earlier, the baseline LLMs achieve the following recall rates on ADR narratives: 0.168 for GPT-OSS-20B and 0.183 for GPT-OSS-120B (each model generates a single prediction, i.e., Top-1). Table 4 shows that after MMR re-ranking with λ = 0.3, the hybrid pipeline achieves a Top-1 recall of 0.241, which is 32% higher than the best baseline (GPT-OSS-120B, 0.183). By contrast, the Top-1 recall of the BM25-only and vector-only pipelines remains below the baseline, reflecting the advantage of the combined lexical-dense re-ranking. When we expand the evaluation to the Top-3 and Top-5 results, the λ = 0.3 hybrid configuration continues to dominate. Its Top-5 recall reaches 0.816 which is 4.46X the recall of the strongest baseline LLM. The λ = 0.5 and λ = 0.7 settings also improve recall relative to the baselines with smaller gains than those observed with λ = 0.3. Overall, these results demonstrate that a modest lexical weight (λ = 0.3) in the MMR re-ranking step not only boosts NDCG across the top ranks but also dramatically increases the recall of relevant ADR narratives, especially when multiple top-ranked items are considered.

	BM25	Vector	λ0.3	λ0.5	λ0.7
Top1	0.156	0.157	**0.241**	0.158	0.157
Top3	0.469	0.470	**0.495**	0.471	0.469
Top5	0.782	0.782	**0.816**	0.782	0.782

Table 4: Top-K Recall Rates

Table 5 reports scores for the various hybrid-search pipelines, illustrating how relevance is distributed among the top-ranked results. Across all pipelines, NDCG consistently rises from the Top-1 to the Top-5 cut-off, indicating that the most relevant ADR narratives tend to appear near the top of the ranked lists.

Among the configurations examined, the hybrid setting with λ = 0.3 achieves the highest performance across all retrieval pipelines. It yields a Top-1 NDCG of 0.500, which improves to 0.506 at Top-3 and reaches 0.638 at Top-5. The relatively low λ places greater emphasis on recall while still promoting the most relevant ADR narratives to the top positions. In contrast, the BM25-only, vector-only, and hybrid variants with λ = 0.5 or 0.7 produce comparable but uniformly lower NDCG scores, indicating that a stronger relevance bias in favor of higher λ reduces the quality of the highest-ranked ADR hits. These results demonstrate that diversified retrieval followed by MMR re-ranking with λ = 0.3 yields the most robust ordering, to which NDCG is especially sensitive.

	BM25	Vector	λ0.3	λ0.5	λ0.7
Top1	0.418	0.420	**0.500**	0.421	0.421
Top3	0.492	0.492	**0.506**	0.494	0.492
Top5	0.608	0.609	**0.638**	0.610	0.610

Table 5: Top-K NDCG Metrics

6 Conclusion and Future Work

This study presents a use case of hybrid Retrieval-Augmented Generation (RAG) framework that jointly supports structured and unstructured ADR prediction by integrating a customized ADR embedding model with during-inference MMR re-ranking using an optimized low-λ setting. The approach bridges representation learning and relevance optimization, where low-λ MMR favors retrieval diversity while preserving sufficient relevance. Although this slightly reduces raw similarity scores, the resulting broader evidence coverage improves detection of rare or less obvious adverse reactions, reduces redundancy of highly similar evidence.

As future work, we plan to extend our framework by incorporating relationship-aware information retrieval methods, such as GraphRAG, to explicitly model relationships among patients, drugs, indications, and ADR outcomes. By enabling multi-hop subgraph retrieval, this approach has the potential to better surface rare ADR signals that are difficult to capture with flat ADR case retrieval. We further intend to investigate the impact of MMR re-ranking with different λ values on subgraph retrieval, with the goal of improving performance for both structured and unstructured ADR prediction tasks.

7 Limitations

The quality and diversity of ADR data are critical for meaningful downstream tasks. While we utilized the largest ADR data source from the FAERS quarterly releases, this is just one of many sources of ADR data. There are numerous other ADR datasets available globally, some are in languages other than English. Incorporating these datasets could enhance the retrieval pipeline robustness and generalizability to less common adverse-reaction patterns. However, accessing such data is challenging due to the highly regulated nature of healthcare information. Additionally, parsing and integrating real-time ADR data from social media platforms remains an open research question, which could further enrich the data pool.

While our study demonstrated promising results with fine-tuning efforts on ADR embedding and retrieval re-ranking using a subset of 500K samples from training data, ideally, the fully set of training data should be used to finetune embedding model. The beg-m3 embedding model used in this study is limited to 8,192 input tokens. In real-world scenarios, such as extensive medication historical records for a patient, the concatenated triplet extraction could span multiple pages. Fortunately, the latest embedding models in the MTEB benchmark allow larger input tokens up to 131,072 tokens and return embeddings with dimensions up to 4096 which can alleviate the input limit concern.

References

American Society of Pharmacovigilance. 2025. Predicting adverse drug event prevalence: A data-driven approach.

Asma Ben Abacha, Md. Faisal Mahbub Chowdhury, Aikaterini Karanasiou, Yassine Mrabet, Alberto Lavelli, and Pierre Zweigenbaum. 2015. Text mining for pharmacovigilance: Using machine learning for drug name recognition and drug–drug interaction extraction and classification. *Journal of Biomedical Informatics*, 58:122–132.

Emmanuel Bresso, Pierre Monnin, Cédric Bousquet, François-Elie Calvier, Ndeye-Coumba Ndiaye, Nadine Petitpain, Malika Smaïl-Tabbone, and Adrien Coulet. 2021. Investigating ADR mechanisms with Explainable AI: a feasibility study with knowledge graph mining. *BMC Medical Informatics and Decision Making*, 21(1):171.

Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336, Melbourne Australia. ACM.

Olivia Choudhury, Yoonyoung Park, Theodoros Salonidis, Aris Gkoulalas-Divanis, Issa Sylla, and Amar k. Das. 2020. Predicting Adverse Drug Reactions on Distributed Health Data using Federated Learning. *AMIA Annual Symposium Proceedings*, 2019:313–322.

Erica Coppolillo, Giuseppe Manco, and Aristides Gionis. 2024. Relevance Meets Diversity: A User-Centric Framework for Knowledge Exploration Through Recommendations. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, pages 490–501, New York, NY, USA. Association for Computing Machinery.

Ghasem Deimazar and Abbas Sheikhtaheri. 2023. Machine learning models to detect and predict patient safety events using electronic health records: A systematic review. *International Journal of Medical Informatics*, 180:105246.

Alireza Farnoush, Zahra Sedighi-Maman, Behnam Rasoolian, Jonathan J. Heath, and Banafsheh Fallah. 2024. Prediction of adverse drug reactions using demographic and non-clinical drug characteristics in FAERS data. *Scientific Reports*, 14:23636.

Valentina Giunchi, Michele Fusaroli, Manfred Hauben, Emanuel Raschi, and Elisabetta Poluzzi. 2023. Challenges and Opportunities in Accessing and Analysing FAERS Data: A Call Towards a Collaborative Approach. *Drug Safety*, 46(10):921–926.

Su Golder, Dongfang Xu, Karen O'Connor, Yunwen Wang, Mahak Batra, and Graciela Gonzalez Hernandez. 2025. Leveraging Natural Language Processing and Machine Learning Methods for Adverse Drug Event Detection in Electronic Health/Medical Records: A Scoping Review. *Drug Safety*, 48(4):321–337.

David Guo and Kim-Kwang Raymond Choo. 2025. Applications of Federated Large Language Model for Adverse Drug Reactions Prediction: Scoping Review. *Journal of Medical Internet Research*, 27(1):e68291. Company: Journal of Medical Internet Research Distributor: Journal of Medical Internet Research Institution: Journal of Medical Internet Research Label: Journal of Medical Internet Research Publisher: JMIR Publications Inc., Toronto, Canada.

Hasham Ul Haq, Veysel Kocaman, and David Talby. 2023. *Mining Adverse Drug Reactions from Unstructured Mediums at Scale*, pages 361–375. Springer International Publishing, Cham.

Sharath Kommu, Christopher Carter, and Philip Whitfield. 2024. Adverse Drug Reactions. In *StatPearls [Internet]*. StatPearls Publishing.

Ceca Kraišniković, Robert Harb, Markus Plass, Wael Al Zoughbi, Andreas Holzinger, and Heimo Müller. 2025. Fine-tuning language model embeddings to reveal domain knowledge: An explainable artificial intelligence perspective on medical decision making. *Engineering Applications of Artificial Intelligence*, 139:109561.

Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, and Ali Farhadi. 2024. Matryoshka Representation Learning. *arXiv preprint*. ArXiv:2205.13147 [cs].

Salima Lamsiyah, Abdelkader El Mahdaouy, Said Ouatik El Alaoui, and Bernard Espinasse. 2023. Unsupervised query-focused multi-document summarization based on transfer learning from sentence embedding models, BM25 model, and maximal marginal relevance criterion. *Journal of Ambient Intelligence and Humanized Computing*, 14(3):1401–1418.

Martin A. Makary and Michael Daniel. 2016. Medical error—the third leading cause of death in the US. *BMJ*, 353:i2139. Publisher: British Medical Journal Publishing Group Section: Analysis.

Asher Mullard. 2025. 2024 FDA approvals. *Nature Reviews Drug Discovery*, 24(2):75–82. Bandiera_abtest: a Cg_type: News Publisher: Nature Publishing Group.

Karen Ka Yan Ng, Izuki Matsuba, and Peter Chengming Zhang. RAG in health care: A novel framework for improving communication and decision-making by addressing LLM limitations. 2(1):AIra2400380. Publisher: Massachusetts Medical Society.

Abeed Sarker and Graciela Gonzalez. 2015. Portable automatic text classification for adverse drug reaction detection via multi-corpus training. *Journal of Biomedical Informatics*, 53:196–207.

Özgür Arzucan He Yongqun Hur Junguk Tiftikci, Mert. 2019. Machine learning-based identification and rule-based normalization of adverse drug reactions in drug labels. *BMC Bioinformatics*, 20.

Michael A. Veronin, Robert P. Schumaker, and Rohit Dixit. 2020. The Irony of MedWatch and the FAERS Database: An Assessment of Data Input Errors and Potential Consequences. *Journal of Pharmacy Technology*, 36(4):164–167. Publisher: SAGE Publications Inc.

Mengzhao Wang, Boyu Tan, Yunjun Gao, Hai Jin, Yingfeng Zhang, Xiangyu Ke, Xiaoliang Xu, and Yifan Zhu. 2025. Balancing the Blend: An Experimental Analysis of Trade-offs in Hybrid Search. *arXiv preprint*. ArXiv:2508.01405 [cs].

Long Xia, Jun Xu, Yanyan Lan, Jiafeng Guo, and Xueqi Cheng. 2015. Learning Maximal Marginal Relevance Model via Directly Optimizing Diversity Evaluation Measures. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pages 113–122, New York, NY, USA. Association for Computing Machinery.

8 Appendix

You are an expert pharmacovigilance assistant. Given a patient profile, treatments (drugs, doses, routes), and indication ("indi_pt") as <query triple extraction>, your task is to generate possible adverse drug reactions (ADRs). The output MUST be a single JSON line containing: "pt": semicolon-separated preferred terms (MedDRA PT) representing possible ADRs as in the examples "uni_code": one of the 7 standardized serious outcome codes as below:

DE – Death

LT – Life-threatening

HO – Hospitalization (initial or prolonged)

DS – Disability

CA – Congenital anomaly

RI – Required intervention to prevent permanent impairment or damage

OT – Other (anything not above)

Your reasoning must NOT appear in the output. Output only the JSON line. Here are 2 examples of input and output.

Input example 1:

> {"patient":{"age": "72", "age_cod": "yr","gndr_cod": "f", "wt": "58", "wt_cod": "kg"}, "treatment": "{"drugname": "vectibix", "route": "intravenous (not otherwise specified)", "dose": "320 milligram, q2wk (biweekly)"}; {"drugname": "cisplatin","route": "intravenous","dose" :"70 mg/m2 q3wk"}", "indi_pt": "asthma; lung cancer metastatic"}

output example 1:

> {"pt": "nausea; vomiting; nephrotoxicity; fatigue; anaemia; tinnitus; electrolyte imbalance","uni_code":"HO"}

Input example 2:

> {"patient":{"age": "5","age_cod": "yr","gndr_cod": "m","wt":

"20","wt_cod": "kg"}, "treatment": "{"drugname": "amoxicillin", "route":"oral", "dose":"250 mg tid"}", "indi_pt": "otitis media"}

Output example 2:

{"pt": "bruising; haemorrhage; epistaxis; anaemia; international normalised ratio increased","uni_code":"RI"}