ADR Detect

By: Nicole Poliak & Naveh Nissan

Comparing LLM Generalization with Embedding-based Models for Adverse Drug Reaction Detection

Motivation

ADRs (Adverse Drug Reactions) are a serious public health issue, often hidden in free-text clinical notes. Prior work has explored fine-tuned transformer models (e.g., BERT) for ADR classification. Consequently, **we aim to benchmark and compare multiple LLM-based strategies on a common task and dataset.**

Our mission:

1. Understand how well general-purpose LLMs perform without training
2. Evaluate how embedding choice affects sentence-level classification

Project Task

Goal

Compare general-purpose LLMs and embedding-based classifiers for ADR detection

- Input: A sentence from a medical case report.
- Output: Binary label (0/1) indicating whether the sentence describes an Adverse Drug Reaction (ADR).
- NLP Task: Binary sentence classification using LLM embeddings.

Two Paths

- Zero-/Few-shot LLM prompting (e.g., GPT-4, Claude)
- Sentence embedding + classifier (TF-IDF, SBERT, OpenAI embeddings → Logistic Regression)

Key Challenges

01

Generalization Without Training → Can LLMs detect ADRs with zero-/few-shot context and no fine-tuning?

Imbalanced (Only ~29% ADR-labeled sentences)

02

Representation Matters → How does embedding choice impact performance in traditional classifiers?

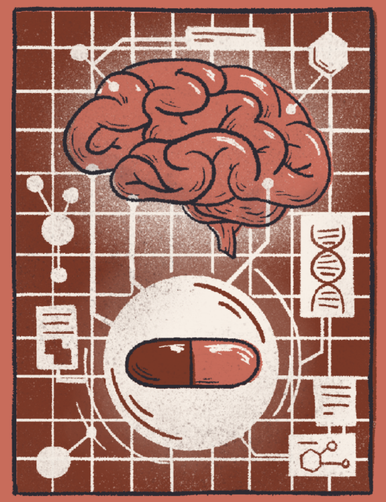
Causal ambiguity (Drugs + effects mentioned without clear link)

03

Data Complexity

Domain-specific phrasing, abbreviations, and medical jargon

Dataset Overview & Usage Strategy



Public Dataset
ADE Corpus V2 – Classification Subsets:
23,517 labeled sentences extracted from 3,000 PubMed case reports.

Labeling Process
Manually annotated by biomedical experts through a multi-stage process.

Data Type
Structured, labeled dataset with two columns:

- Text: Sentence from medical abstract.
- Label: Binary (1 = ADR, 0 = non-ADR).

Usage Strategy

- Embedding-based Models:

→ Train/test split (80/20) using classical classifiers.

- LLMs (GPT-4, Claude, etc.):

→ Evaluate directly via zero-shot and few-shot prompting.
→ No fine-tuning involved.

Input/Output Example

Negative Example (ADR Not Present):

Input
"The total amount of vitamin K received from the enteral feedings ranged from 50 to 115 micrograms/day, which is less than the normal daily intake of 300 to 500 micrograms."

Output
Label: 0

- This sentence only reports a **vitamin K dosage range** without mentioning any harm or negative reaction.

Positive Example (ADR Present):

Input
"Lupus-like syndrome caused by 5-aminosalicylic acid in patients with inflammatory bowel disease."

Output
Label: 1

- The sentence describes a drug **(5-aminosalicylic acid)** causing an adverse effect **(Lupus-like syndrome).**

- **In embedding-based models, these sentences are vectorized and used for training/testing.**
- **In LLMs, examples like this are used in few-shot prompts to evaluate zero-training classification ability.**

Evaluation

Evaluation Metrics

- Precision: Of all predicted ADRs, how many were correct?
- Recall: Of all actual ADRs, how many were detected?
- F1-Score: Harmonic mean of precision and recall – balances both

Evaluation Strategy

Embedding-Based Models

- Train/test split: 80/20, stratified by label
- Input: Sentence → Embedding (TF-IDF, SBERT, OpenAI) → Classifier (e.g., Logistic Regression)

LLM Models (GPT-4, Claude, etc.)

- Zero-shot & few-shot prompting only
- Evaluated without fine-tuning
- Prompts include task description + labeled examples (few-shot)

Evaluation

Baseline & Comparison

Embedding Models

- Baseline: Naïve Bayes + Bag-of-Words
- Compare: TF-IDF, SBERT, OpenAI embeddings with Logistic Regression

LLMs (Zero-/Few-Shot)

- Baseline: GPT-4 Zero-shot prompt
- Compare: Few-shot prompting using labeled examples as context