By: Nicole Poliak & Naveh Nissan

ADR Detect

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

Project Proposal

Motivation

Adverse Drug Reactions (ADRs) are a leading cause of patient harm and hospitalization, yet they are often underreported and buried within unstructured clinical text. Automatically detecting ADRs from these texts is essential for improving pharmacovigilance, enhancing patient safety, and reducing healthcare costs.

This project aims to explore and compare two NLP approaches for ADR detection:

General-purpose Large Language Models (LLMs) used in zero- or few-shot settings & Sentence embedding methods paired with a traditional classifier. By benchmarking both methods, we assess their effectiveness in enabling safer, AI-assisted medical decision-making.

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.

Two Paths:

Zero-/Few-shot LLM prompting:
GPT-4o-mini, GPT-4o, Phi-4-mini-instruct, LLaMA-3.2-3B-Instruct

Sentence embedding + classifier:
BioBERT, SBERT, InstructorXL → Logistic Regression

Key Challenges

01

Noisy and Ambiguous Text:
Patient-written reviews often include unrelated symptoms, making it hard to distinguish actual ADRs from coincidental mentions.

02

Domain-Specific Complexity:
Clinical and pharmaceutical texts contain medical jargon, abbreviations, and phrasing that may confuse general models.

03

Label Imbalance:
Only ~24% of the dataset contains ADR-positive sentences, which poses a challenge for both LLMs and classifiers.

Dataset Overview & Usage Strategy

Public Datasets

- ADE Corpus V2 : Expert-annotated PubMed case reports (23,516 sentences).
- PsyTAR: Patient-reported reviews from askapatient.com (6,009 sentences)

Data Type

Combined, structured and labeled dataset with three columns:

- Text: Sentence from medical abstract or patient review.
- Label: Binary (1 = ADR, 0 = non-ADR).
- Dataset: Dataset origin (ADE / PsyTAR).

Usage Strategy

- Embedding-based Models:

→ Train/dev/test split (60/20/20) using BioBERT, SBERT, InstructorXL.

- LLMs (GPT-4 models, Phi-4-mini-instruct, LLaMA-3.2-3B-Instruct):

→ 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).

Evaluation

Evaluation Metrics

- **Accuracy:** Overall percentage of correctly predicted labels.
- **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.
- **ROC-AUC:** Embeddings only
- **Confusion Matrix**

Evaluation Strategy

Embedding-Based Models:

- Train/dev/test split: 60/20/20, stratified by label.
- Input: Sentence → Embedding (InstructorXL, BioBERT, SBERT) → Logistic Regression
- Baseline: Naïve Bayes + Bag-of-Words.

LLM Models:

- Zero-shot & few-shot prompting only.
- Evaluated without fine-tuning.
- Prompts include task description + labeled examples (few-shot).
- Baseline: GPT-4o-mini zero-shot prompt.

Interim report

Description

Task

- **Input:** Sentence from biomedical literature (PubMed) / sentence from patient reviews (askapatient.com).
- **Output:** Binary label (0 = non-ADR, 1 = contains ADR).
- **Goal: Compare two paradigms for biomedical sentence classification:**

Zero-/Few-shot LLM prompting:
GPT-4o-mini, GPT-4o, Phi-4-mini-instruct, LLaMA-3.2-3B-Instruct

Sentence embedding + classifier:
BioBERT, SBERT, InstructorXL → Logistic Regression

Datasets:

- ADE Corpus V2: 23,516 expert-annotated sentences from biomedical literature.
- PsyTAR: 6,009 patient drug reviews.

Structure:

- Text: Sentence describing a clinical case.
- Label: ADR present (1) / not present (0).

Description

Usage:

- Embedding models: 60/20/20 stratified train/dev/test split.
- LLMs: Zero-shot and few-shot prompting (no fine-tuning).

Evaluation:

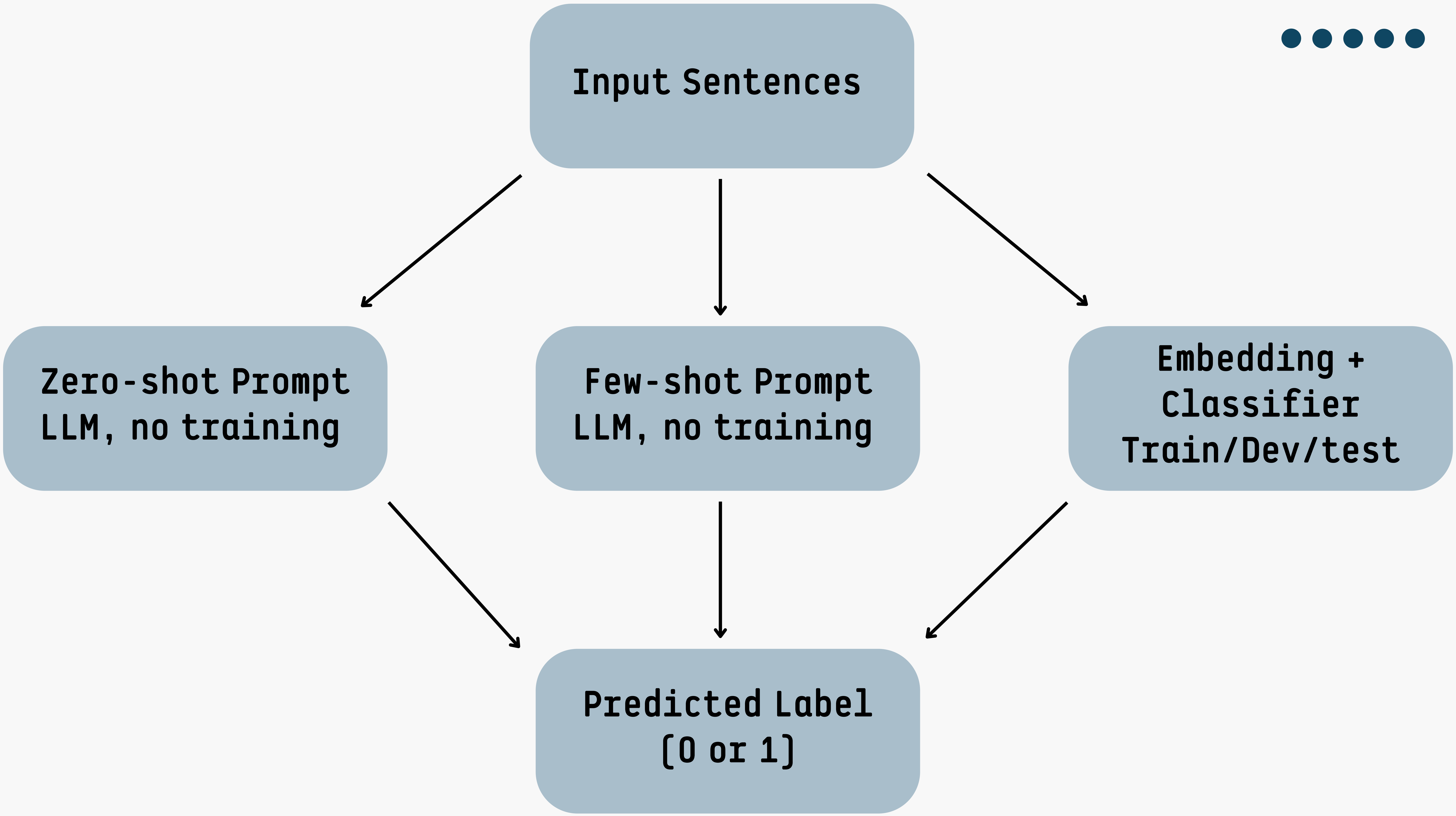
- Metrics: Accuracy, Precision, Recall, F1-Score, ROC-AUC (embeddings only).
- Embedding Models: Classifier trained on extracted embeddings.
- Baseline = Naïve Bayes + Bag-of-Words.
- LLMs: Zero/few-shot without training.
- Baseline = GPT-4o-mini zero-shot performance.

Prior Art

Source / Title	Approach / Model	Data	Metrics	Results
ModernBERT vs LLMs **for Detecting Adverse Drug Reactions** **Simmering.dev, 2025**	Comparison between: ModernBERT-base and ModernBERT-large as the structured language models with Llama 3.2-3B-instruct as the LLM.	ADE-Benchmark Corpus (23.5k labeled sentences)	Recall, Precision, F1 Score, Speed, Cost	**1a-modernbert-base:** f1-score - 86.0, recall - 90.3, precision - 82.2 **1b-modernbert-large:** f1-score - 89.2, recall - 91.8, precision - 86.8 **2-DSPy-25-threads-Llama-3.2-3B-Instruct:** f1-score - 80.7, recall - 87.9, precision - 74.6
LLMs are not Zero-Shot Reasoners **for Biomedical Information Extraction** **ACL Anthology, 2025**	Various open LLMs— BioMistral and Llama-2 models —using standard prompting, Chainof-Thought (CoT), Self-Consistency based reasoning and Retrieval-Augmented Generation (RAG).	14 different classification datasets from the BigBio collection like: PsyTAR dataset	F1 Score	Standard prompting (VANILLA); highest average F1 scores: **BioMistral-7B** (36.48%), **Llama-2-70B-Chat-AWQ** (40.34%), **Llama-2- 7b-chat-hf** (34.92%).
Using LLMs to Extract ADR's **from Short Text** **SCITEPRESS, 2025**	Evaluate various LLMs and ML approaches for ADR extraction and detection, Using multiple ADR datasets and a range of prompt formulations.	ASU-CHOP dataset, SMM4H dataset, WEB-RADRSMM4H dataset and ADE Corpus v2	Recall, Precision, F1 Score	**GPT-4o-mini** (Precision - 1.0, Recall - 0.91, F1 - 0.95) **GPT-4** (Precision - 1.0, Recall - 0.97, F1 - 0.98) **Llama** (Precision - 1.0, Recall - 1.0, F1 - 1.0)

Steps

Step	Description	Input → Output	Method/Tool	Metrics
1a. Preprocessing	Prepare text (for BoW & TF-IDF)	Raw sentence → Cleaned text	Lowercasing & punctuation removal	None (text preparation only)
1b. Explorative Data Analysis	Analyze dataset structure & class distribution	Data → Summary stats, visualizations	Pandas, Matplotlib, Seaborn	• Label distribution • Sentence/word length stats • Top frequent terms
2a. Baseline model	Simple lexical baseline using BoW features	Cleaned text → Binary label	CountVectorizer + Naïve Bayes	• Accuracy • Precision • Recall • F1-Score • ROC-AUC
2b. InstructorXL Embedding	Extract task-aware semantic features	Instruction + sentence ➝ Dense vector	Instructor-XL	Evaluated in step 3a
2c. SBERT / BioBERT	Extract semantic features	Raw sentence → Dense vector	sentence-transformers / BioBERT / SBERT	Evaluated in step 3a
3a. Classification (Embeddings)	Train classifier on embeddings	Embedding ➝ Binary label	Logistic Regression	• Accuracy • Precision • Recall • F1-Score • ROC-AUC
3b. LLM Prompting	Prompt LLM to return label	Sentence + prompt → Binary label	GPT-4o-mini, GPT-4o, GPT-4.1, Phi-4-mini-instruct, LLaMA-3.2-3B-Instruct (zero/few-shot, no train)	• Accuracy • Precision • Recall • F1-Score
4. Evaluation	Compare model outputs to true labels	Predictions + Labels ➝ Scores	sklearn.metrics, seaborn, visualizations	• Confusion Matrix & ROC-AUC Curve (embedding models) • Metric comparison across models • Zero-/Few-shot performance gap (LLMs)



Input Sentences

Zero-shot Prompt
LLM, no training

Few-shot Prompt
LLM, no training

Embedding +
Classifier
Train/Dev/test

Predicted Label
(0 or 1)

Exploration & Baseline

Dataset

- ADR classification dataset combining ADE Corpus V2 and PsyTAR
- 26,867 sentences, 2 classes (ADR = 1, Non-ADR = 0) after cleaning & removing duplicates
- Minimal text cleaning: lowercasing, punctuation removal
- Mean sentence length: 17.2 ± 8.91 words (max 155 words)
- Notable class imbalance: ~76% Non-ADR, ~24% ADR → downsampled majority class to balance (6,431)

Baseline Results

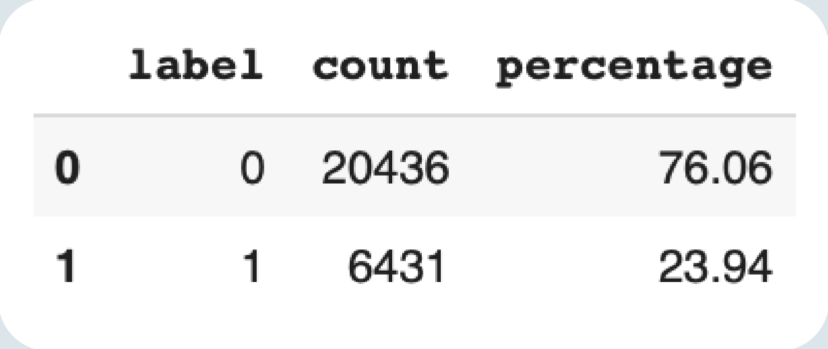
1. Bag-of-Words (CountVectorizer) + Multinomial Naive Bayes

- 60/20/20 stratified train-dev-test split
- Performance → Test: Accuracy = 0.76, Precision = 0.73, Recall = 0.81, F1 = 0.77

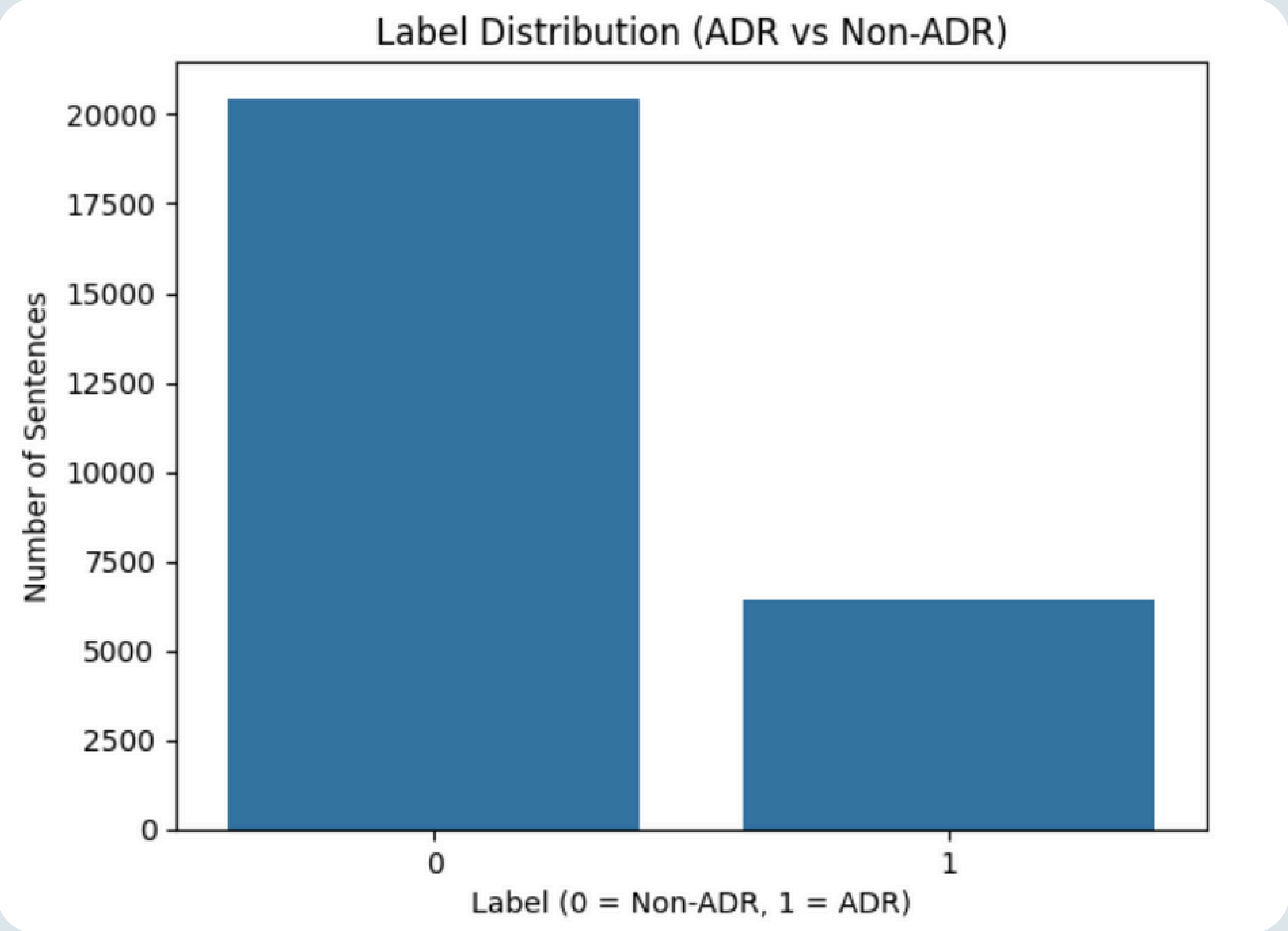
2. GPT-4o-mini Zero-Shot

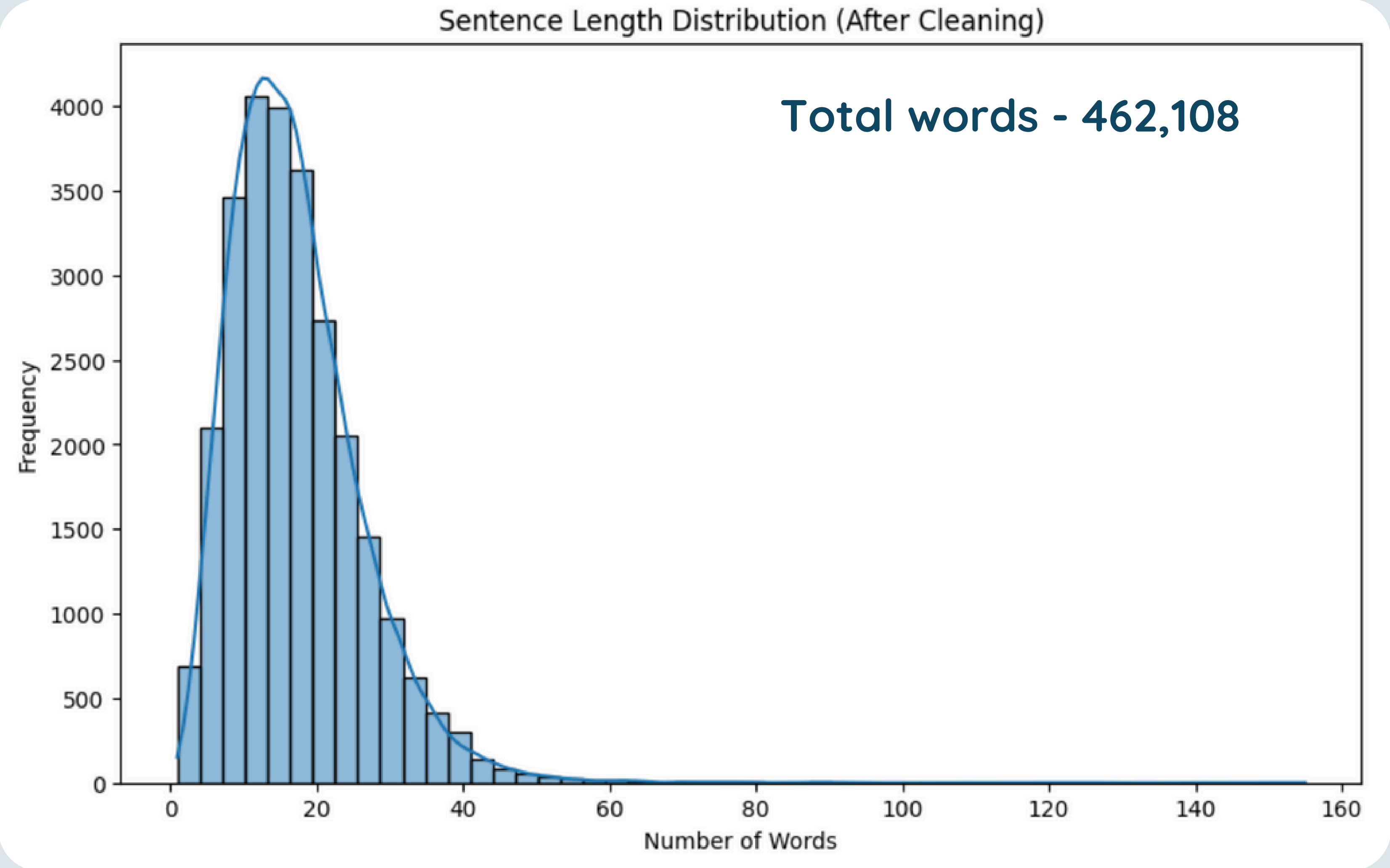
- Trained only on test data (20%)
- Performance → Accuracy = 0.83, Precision = 0.79, Recall = 0.90, F1 = 0.84

EDA Visualizations

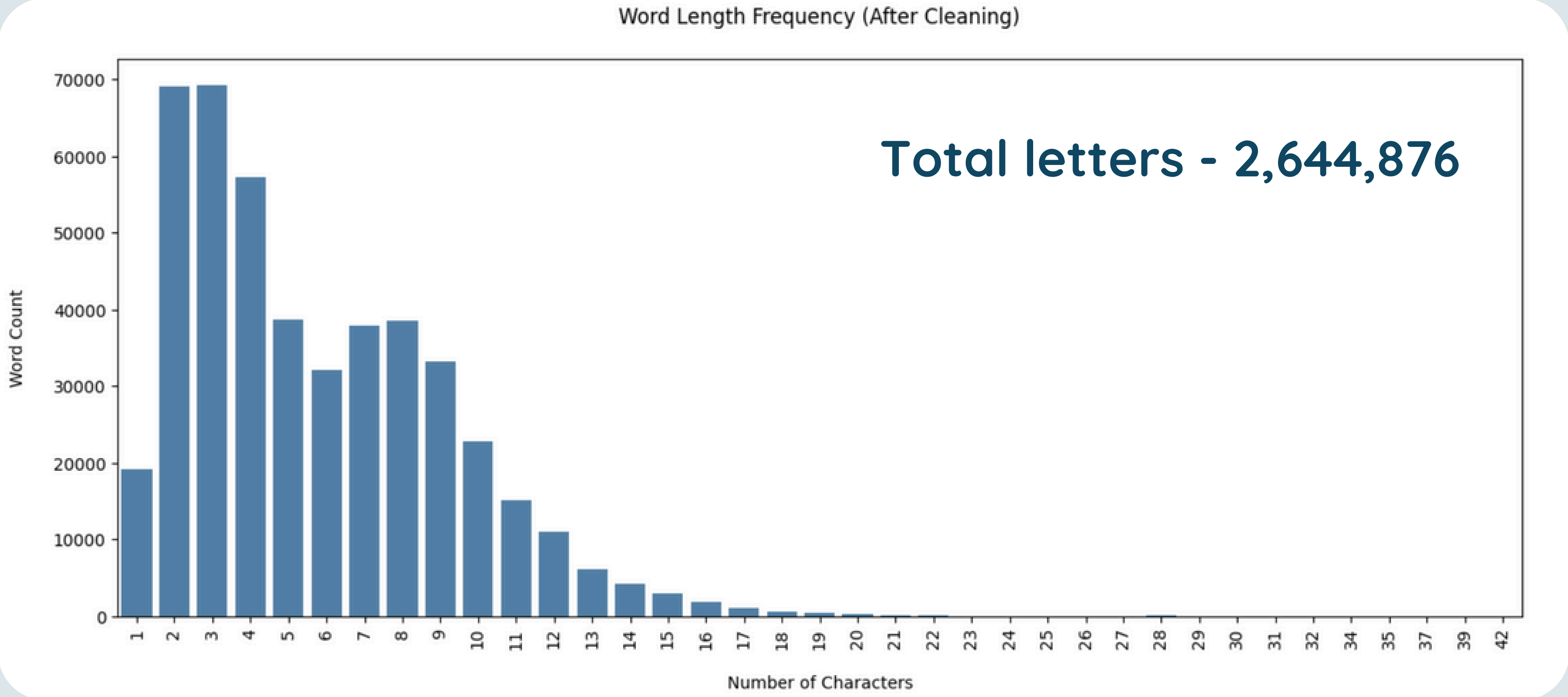


	label	count	percentage
0	0	20436	76.06
1	1	6431	23.94





Total words - 462,108



Total letters - 2,644,876

Insights & Recommendations

- From Simmering.dev (2025): ModernBERT models outperformed a few-shot LLaMA → **Our embedding models approach may outperform zero-/few-shot LLMs.**
- From ACL Anthology (2025): From ACL Anthology (2025): Advanced prompting (CoT, Self-Consistency) didn't always improve over basic prompting in open source LLMs with zero-shot → **Keep LLM prompting simple and controlled when relevant.**
- From SCITEPRESS (2025): GPT-4 and Llama showed better performance → **supports including them as a competitive LLM models, even without fine-tuning.**
- Mean sentence length = 17.2 words (Non-ADR has more outliers) → **Sentences are short enough to fit within BERT and LLM token limits.**
- Strong class imbalance (~76% Non-ADR) required downsampling → LLMs and BERT/InstructorXL models will be evaluated on the downsampled dataset.
- Duplicate rows were removed before & after cleaning.
- **Level of Sensitivity (Recall) is the primary metric** for our task - FN are more important than FP

Final Report

Introduction •••••

What are ADRs?

Adverse drug reactions: Unexpected negative effects of a medication despite being used at its normal dose, and can range from mild discomfort to life-threatening conditions.

Why it matters:

- A major cause of patient harm and hospitalization.
- Timely detection improves drug safety and healthcare decisions.

Challenges:

- Manual detection is slow and inconsistent.
- Automated detection faces noisy patient text and must understand medical terminology, context and causality.

Why is it a problem?

ADRs are underreported, implicit, and buried in complex or informal language.

Who benefits:

Healthcare providers, pharmaceutical companies and patients.

Introduction

- Our project tackles the challenge of automatically detecting ADRs in short medical texts by focusing on sentence-level classification.
- This is a key component of the larger problem of mining ADRs from unstructured clinical data.
- We aim to compare two prominent NLP paradigms: general-purpose Large Language Models (LLMs) using zero- and few-shot prompting (different prompt formulations) and sentence embedding models combined with a classifier.
- By evaluating their performance on a unified ADR-labeled dataset, we investigate which method better supports scalable, accurate ADR detection.

Formal Task Specification

Input:

A single sentence from a medical abstract or patient review.

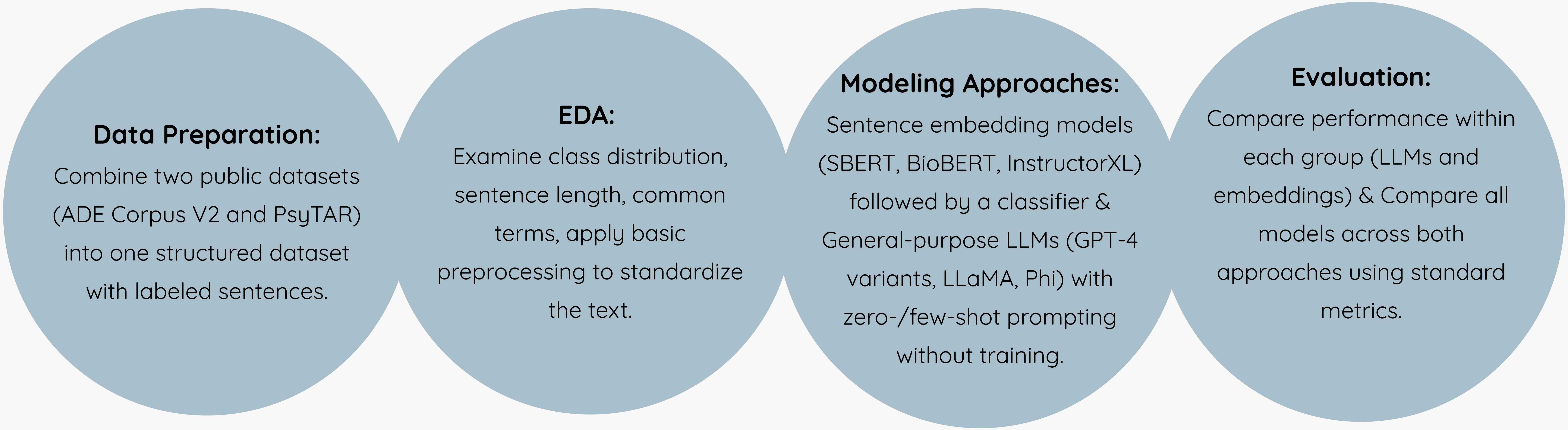
Output:

A binary label
1 → Sentence describes an ADR
0 → Sentence doesn’t describe an ADR

Evaluation Metrics:

- **Accuracy:** Overall correctness of predictions.
- **Precision:** Correct ADR predictions out of all ADR-labeled predictions.
- **Recall (Sensitivity):** Correct ADR predictions out of all actual ADR cases.
- **F1-Score:** Harmonic mean of precision and recall — balances false positives and false negatives.

High-Level Plan



Data Preparation:
Combine two public datasets (ADE Corpus V2 and PsyTAR) into one structured dataset with labeled sentences.

EDA:
Examine class distribution, sentence length, common terms, apply basic preprocessing to standardize the text.

Modeling Approaches:
Sentence embedding models (SBERT, BioBERT, InstructorXL) followed by a classifier & General-purpose LLMs (GPT-4 variants, LLaMA, Phi) with zero-/few-shot prompting without training.

Evaluation:
Compare performance within each group (LLMs and embeddings) & Compare all models across both approaches using standard metrics.

Prior Art

Source / Title	Task solved	Approach / Model	Data	Metrics	Results
ModernBERT vs LLMs for Detecting Adverse Drug Reactions **Simmering.dev, 2025**	Adverse event classification	Fine-tuning ModernBERT, few-shot learning with Llama 3.2-3B and fine-tuning Llama 3.2-3B.	ADE-Benchmark Corpus (23.5k labeled sentences)	Recall, Precision, F1 Score, Speed, Cost	**1a-modernbert-base:** f1-score - 86.0, recall - 90.3, precision - 82.2 **1b-modernbert-large:** f1-score - 89.2, recall - 91.8, precision - 86.8 **2-DSPy-25-threads-Llama-3.2-3B-Instruct:** f1-score - 80.7, recall - 87.9, precision - 74.6
LLMs are not Zero-Shot Reasoners for Biomedical Information Extraction **ACL Anthology, 2025**	Benchmark LLM performance in Medical Classification and Named Entity Recognition	Various open LLMs—BioMistral and Llama-2 models—using standard prompting, Chainof-Thought (CoT), Self-Consistency based reasoning and Retrieval-Augmented Generation (RAG).	14 different classification datasets from the BigBio collection like: PsyTAR dataset	F1 Score	**Standard prompting (VANILLA);** highest average F1 scores: BioMistral-7B (36.48%), Llama-2-70B-Chat-AWQ (40.34%), Llama-2- 7b-chat-hf (34.92%).
Using LLMs to Extract ADR's from Short Text **SCITEPRESS, 2025**	ADR extraction and detection	Evaluate various LLMs and ML approaches for ADR extraction and detection, Using multiple ADR datasets and a range of prompt formulations.	ASU-CHOP dataset, SMM4H dataset, WEB-RADRSMM4H dataset and ADE Corpus v2	Recall, Precision, F1 Score	**GPT-4o-mini** (Precision - 1.0, Recall - 0.91, F1 - 0.95) **GPT-4** (Precision - 1.0, Recall - 0.97, F1 - 0.98) **Llama** (Precision - 1.0, Recall - 1.0, F1 - 1.0)

Data Preparation & Description

Source Datasets

- ADE Corpus V2 – Expert-annotated PubMed case reports (~23,500 sentences) ADE Corpus V2 dataset
- PsyTAR – Patient drug reviews from askapatient.com (~6,000 sentences) PsyTAR dataset

Fields

Combined, structured and labeled dataset with three columns:

- **Text:** Single sentence describing medical event or drug use.
- **Label:** Binary (1 = ADR present, 0 = not present).
- New field was created: ‘**dataset**’ - origin of sentence (ADE or PsyTAR)

Labeling

- ADE Corpus V2: Manually labeled by domain experts for ADR presence.
- PsyTAR: Manually labeled by four annotators following detailed guidelines, with consistency ensured through double coding and inter-annotator agreement (Kappa = 78%).

Preparation Process

- From PsyTAR, extracted drug, ADR label, and text columns.
- Combined drug name and text in the shape: ‘drug: sentence’.
- Converted blank ADR labels to 0 (Non-ADR).
- Combined with ADE dataset by aligning text and label fields.
- Added a ‘dataset’ column to indicate sentence origin (ADE or PsyTAR).

EDA Steps ●●●●●

- **Dataset Composition & Origin:**
  - A combined dataset with 29525 rows × 3 columns (text, label, dataset)
- **Class Distribution (Per Dataset):**
  - ADE Corpus V2: Non-ADR - 16,695 sentences & ADR - 6,821 sentences.
  - PsyTAR: Non-ADR - 3,841 sentences & ADR - 2,168 sentences.
- **Duplicate Removal:** Identified and dropped 2,639 duplicate rows.
- **Updated Class Distribution:** ADR - 6,431 (23.94%), Non-ADR - 20,436 (76.06%) → Clear class imbalance observed.
- **Sentence Length Analysis:**
  - Calculated number of words per sentence & calculate summary statistics: Mean - 17.2 words, Min - 1 word, Max - 156 words.
  - ADR sentences tend to be slightly longer and more descriptive than non-ADR ones.
- **Text Cleaning:** converted all text to lowercase, removed punctuation, rechecked and removed any new duplicate texts introduced by cleaning.
- **Corpus Size:** total words - 462,108 & total characters - 2,644,876.
- **Handling Imbalance:** Applied random downsampling to the Non-ADR class.
- **Final dataset:** 6,431 ADR + 6,431 Non-ADR = 12,862 sentences.

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.

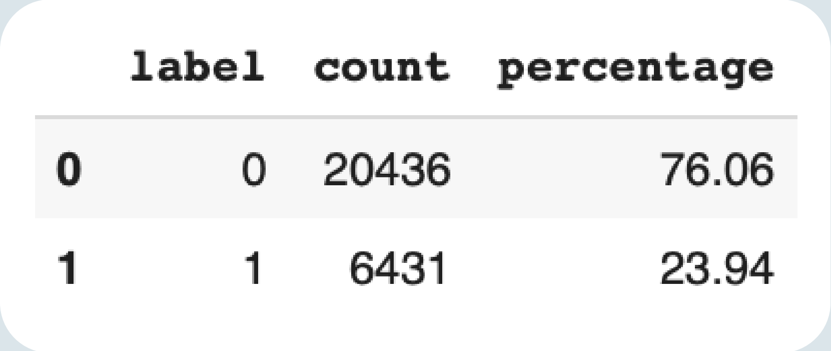
Positive Example (ADR Present):

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

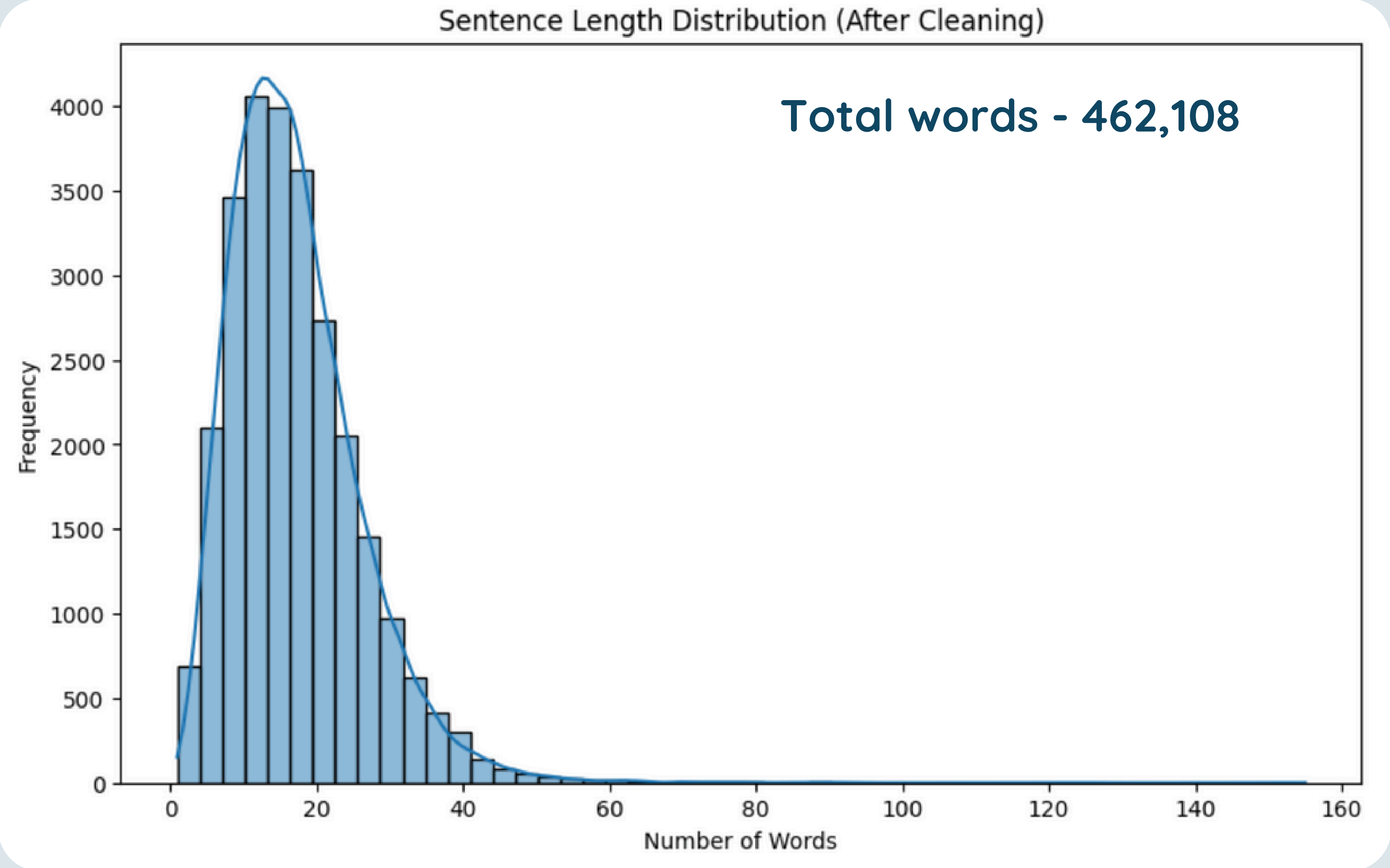
Output: Label: 1

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

EDA Visualizations

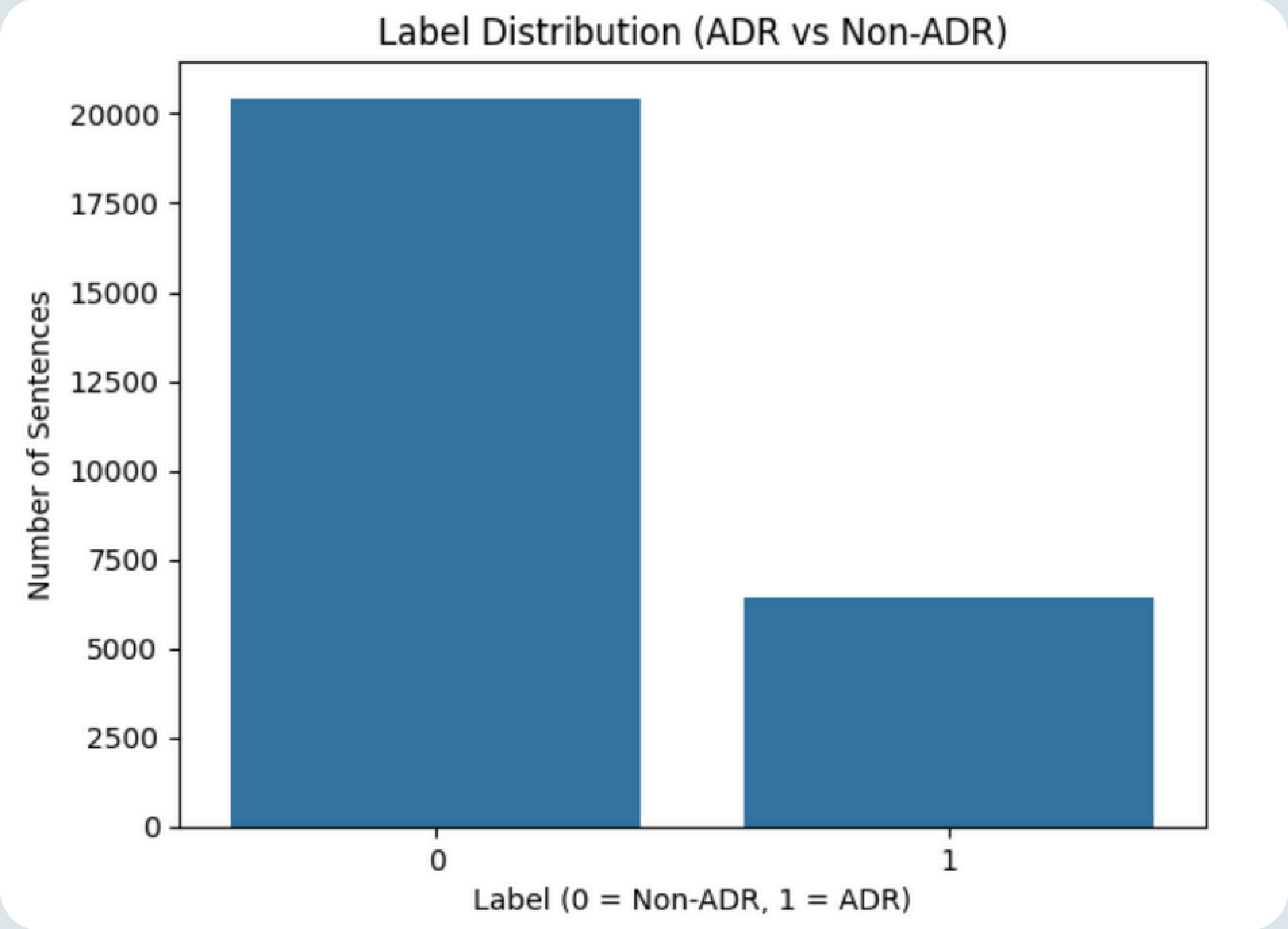


	label	count	percentage
0	0	20436	76.06
1	1	6431	23.94

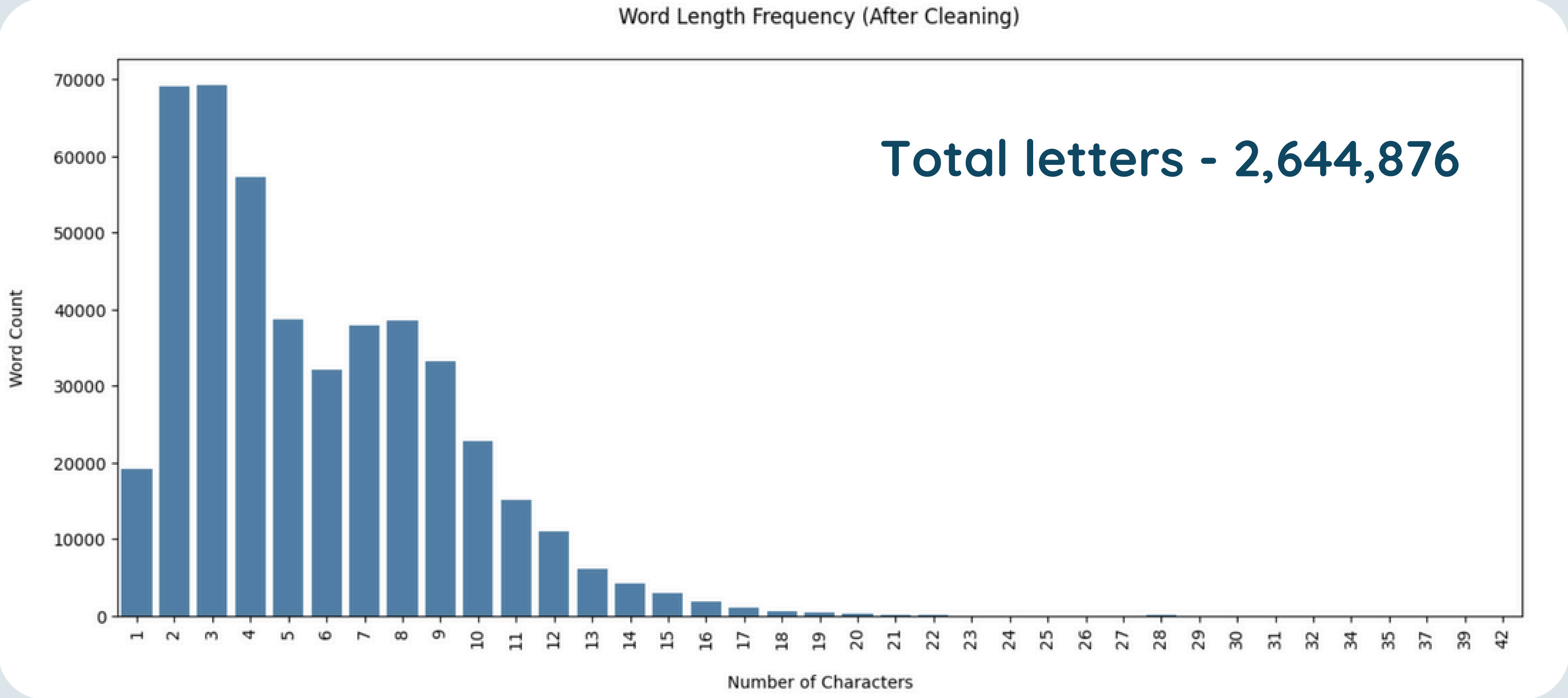


Sentence Length Distribution (After Cleaning)

Total words - 462,108



Label Distribution (ADR vs Non-ADR)



Word Length Frequency (After Cleaning)

Total letters - 2,644,876

Models & processing pipelines

Embedding-Based	LLM-Based
Baseline: Naïve Bayes (BoW + Naïve Bayes)	Baseline: GPT-4o-mini (Zero-shot prompt)
BioBERT + logistic regression	GPT-4o-mini (Few-shot prompt)
SBERT + logistic regression	GPT-4o (Zero&Few shot prompt)
InstructorXL + logistic regression	Phi-4-mini-instruct (Zero&Few shot prompt)
	LLaMA-3.2 3B-Instruct (Zero&Few shot prompt)

Training & Platform Details:

- Embedding-based models trained using stratified **60/20/20 split: train/dev/test.**
- LLM-based models were evaluated using only **test data.**
- Platform: **Google Colab Pro+ with A100 GPU.**
- LLM models accessed via:
  - Azure OpenAI (GPT, Phi)
  - Hugging Face (LLaMA)

LLM Inference Settings:

- max_tokens=5, temperature=0.0-0.1, top_p=1.0
- All models prompted with **short deterministic outputs (0 or 1)**
- Few-shot prompts used **4–8 balanced examples**

Embedding Model Configurations:

- max_iter=1000 in Logistic Regression
- batch_size=16 during BioBERT embedding

Evaluation

- **Classification task** → Binary labels (ADR / Non-ADR)
- **Metrics:** Accuracy, Precision, Recall, F1-Score, ROC-AUC, Confusion Matrix

- **Baselines:** Accuracy, Precision, Recall, F1-Score, ROC-AUC (only for BoW + Naïve Bayes) Confusion Matrix.
- **LLMs:** Only classification metrics & Confusion Matrix (no ROC-AUC)
- **Embeddings:** Class probabilities → ROC & AUC analysis

Set	Computed Metrics
Train (60%)	Accuracy, Precision, Recall, F1
Dev (20%)	Accuracy, Precision, Recall, F1
Test (20%)	Accuracy, Precision, Recall, F1, ROC-AUC (BOW + embeddings only), Confusion Matrix

Code Organization & Repository Overview

[Link to GitHub]

Data File

- **combined_dataset.csv:** Final dataset combining ADE & PsyTAR
- Columns: text: drug-related sentence, label: Binary (1 = ADR present, 0 = not ADR), dataset: Source origin (ADE or PsyTAR)

Code Notebooks

- **data_preparation.ipynb:** Combines datasets
- **adr_classification_pipeline.ipynb:** EDA + baseline models + embedding-based models + Zero-/few-shot classification + Evaluation

Presentation file

- ADR Detection from Text: Embeddings vs. LLMs

Results files

- llm_results.csv
- overall_results.csv
- embeddings_and_baseline_results.csv

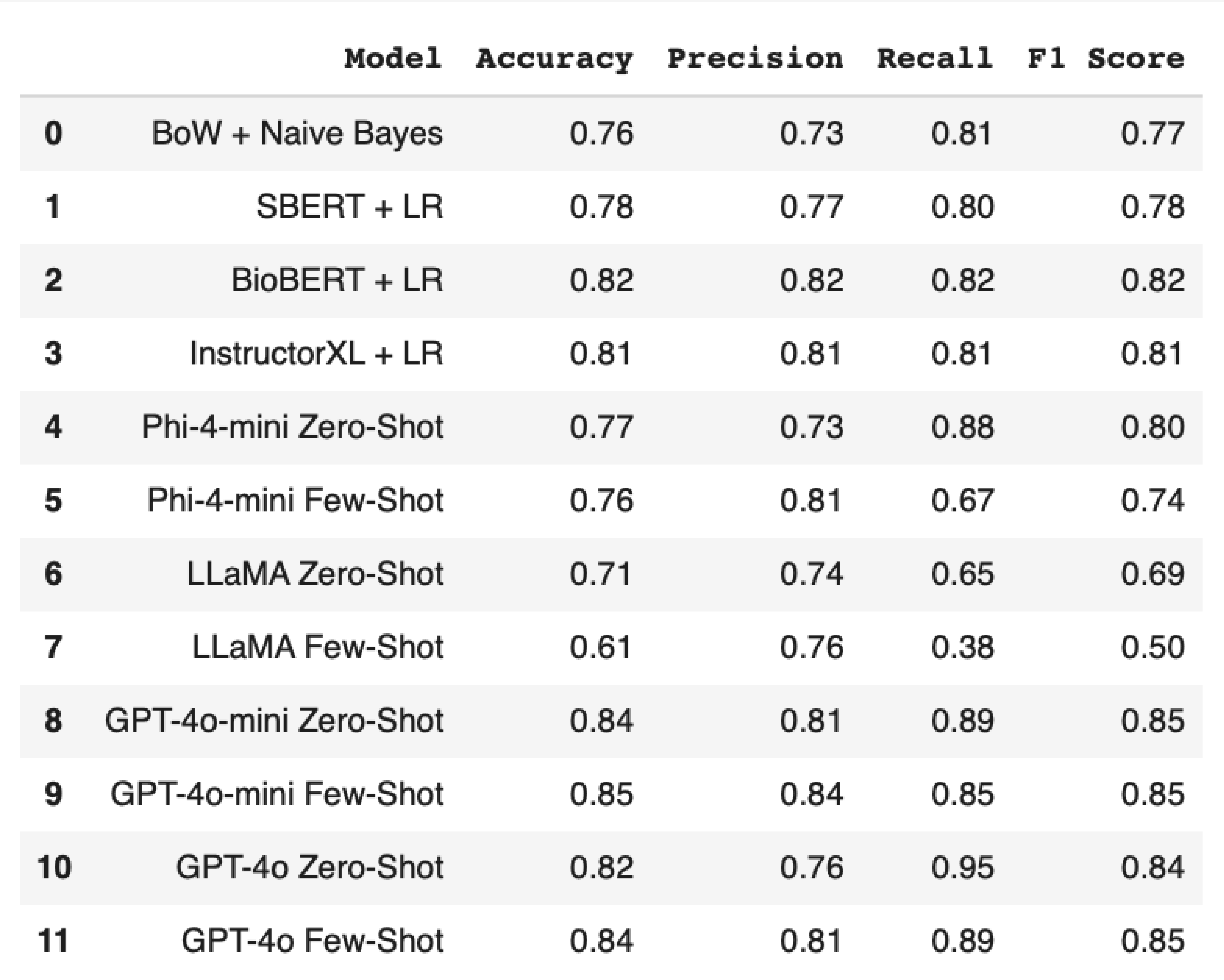
Baselines Results

BoW + Naïve Bayes

Metric	Train Score	Dev Score	Test Score
Accuracy	0.89	0.76	0.76
Precision	0.86	0.73	0.73
Recall	0.94	0.82	0.81
F1 Score	0.9	0.77	0.77

Metric	GPT-4o-mini Score
Accuracy	0.84
Precision	0.81
Recall	0.89
F1 Score	0.85

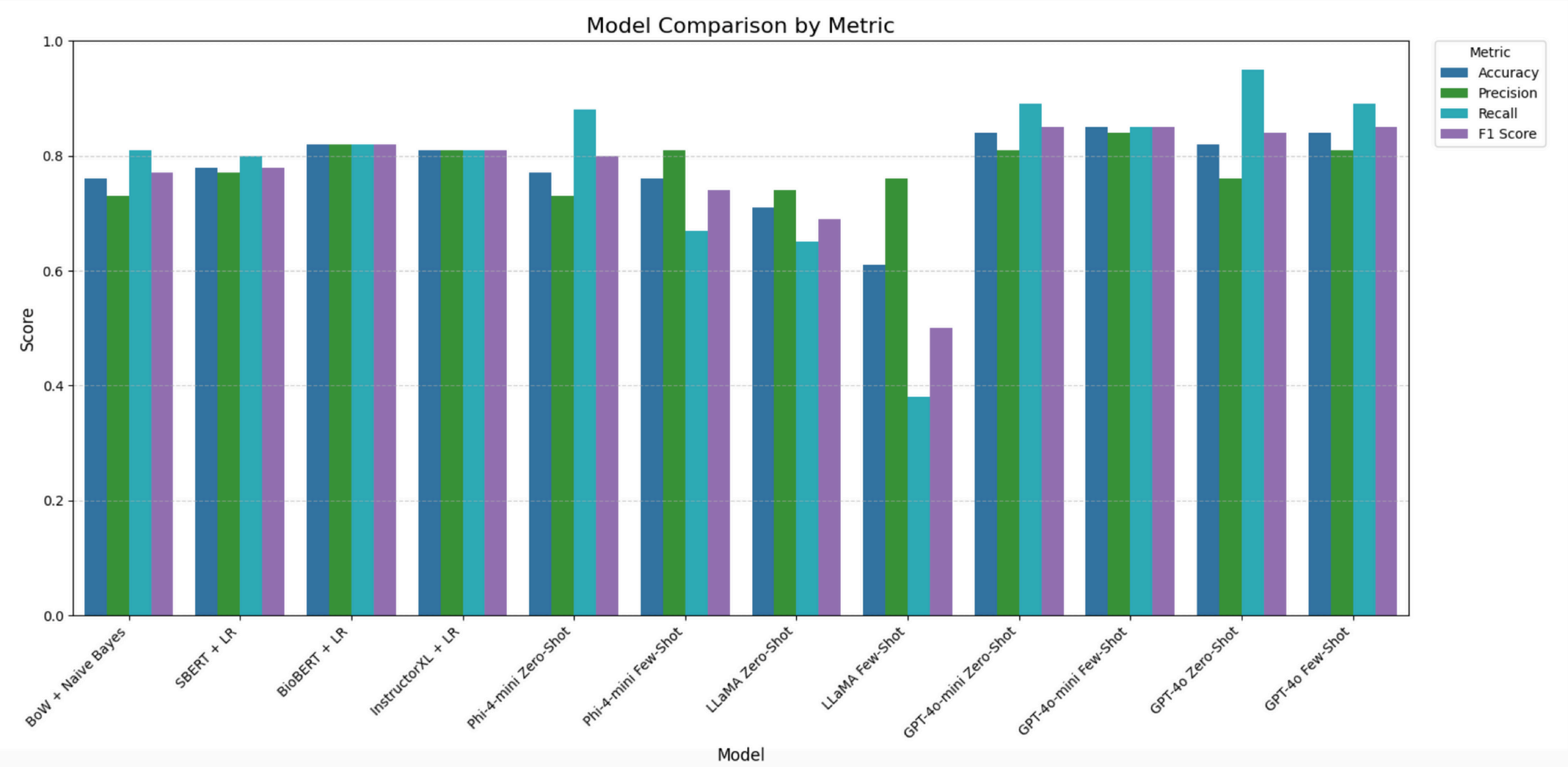
Main Results



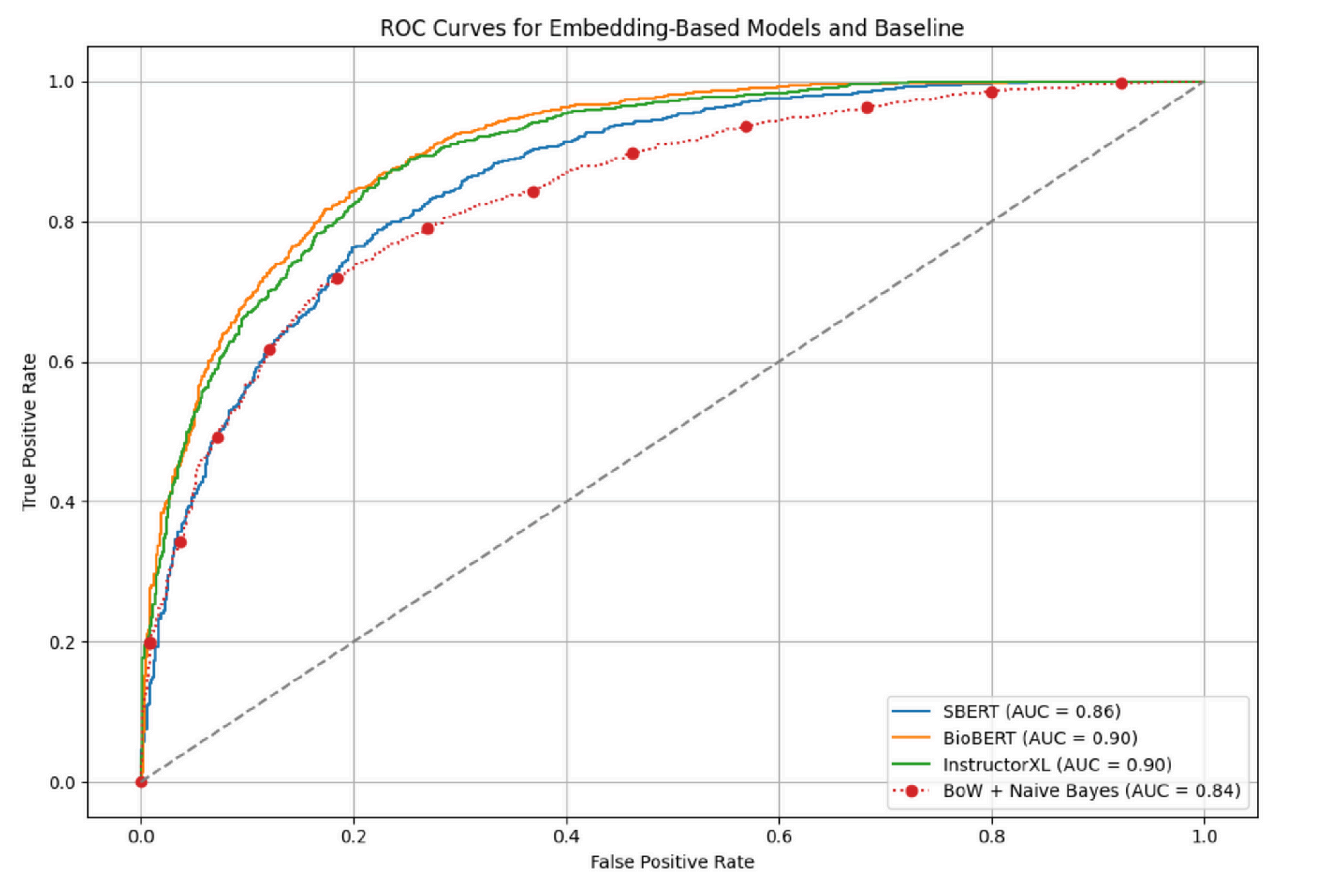
	Model	Accuracy	Precision	Recall	F1 Score
0	BoW + Naive Bayes	0.76	0.73	0.81	0.77
1	SBERT + LR	0.78	0.77	0.80	0.78
2	BioBERT + LR	0.82	0.82	0.82	0.82
3	InstructorXL + LR	0.81	0.81	0.81	0.81
4	Phi-4-mini Zero-Shot	0.77	0.73	0.88	0.80
5	Phi-4-mini Few-Shot	0.76	0.81	0.67	0.74
6	LLaMA Zero-Shot	0.71	0.74	0.65	0.69
7	LLaMA Few-Shot	0.61	0.76	0.38	0.50
8	GPT-4o-mini Zero-Shot	0.84	0.81	0.89	0.85
9	GPT-4o-mini Few-Shot	0.85	0.84	0.85	0.85
10	GPT-4o Zero-Shot	0.82	0.76	0.95	0.84
11	GPT-4o Few-Shot	0.84	0.81	0.89	0.85

- **GPT-4o Zero-Shot had the best Recall** (0.95) and good f1-score (0.84).
- **GPT-4o-mini (zero/few) and GPT-4o Few-shot had the highest f1-score** (0.85).
- **BioBert had the highest scores for embeddings** over all metrics (0.82).
- **LLaMA Few-Shot showed poor performance** (F1 = 0.50) and 14.15% invalid responses, limiting reliability.

Main Results



Main Results

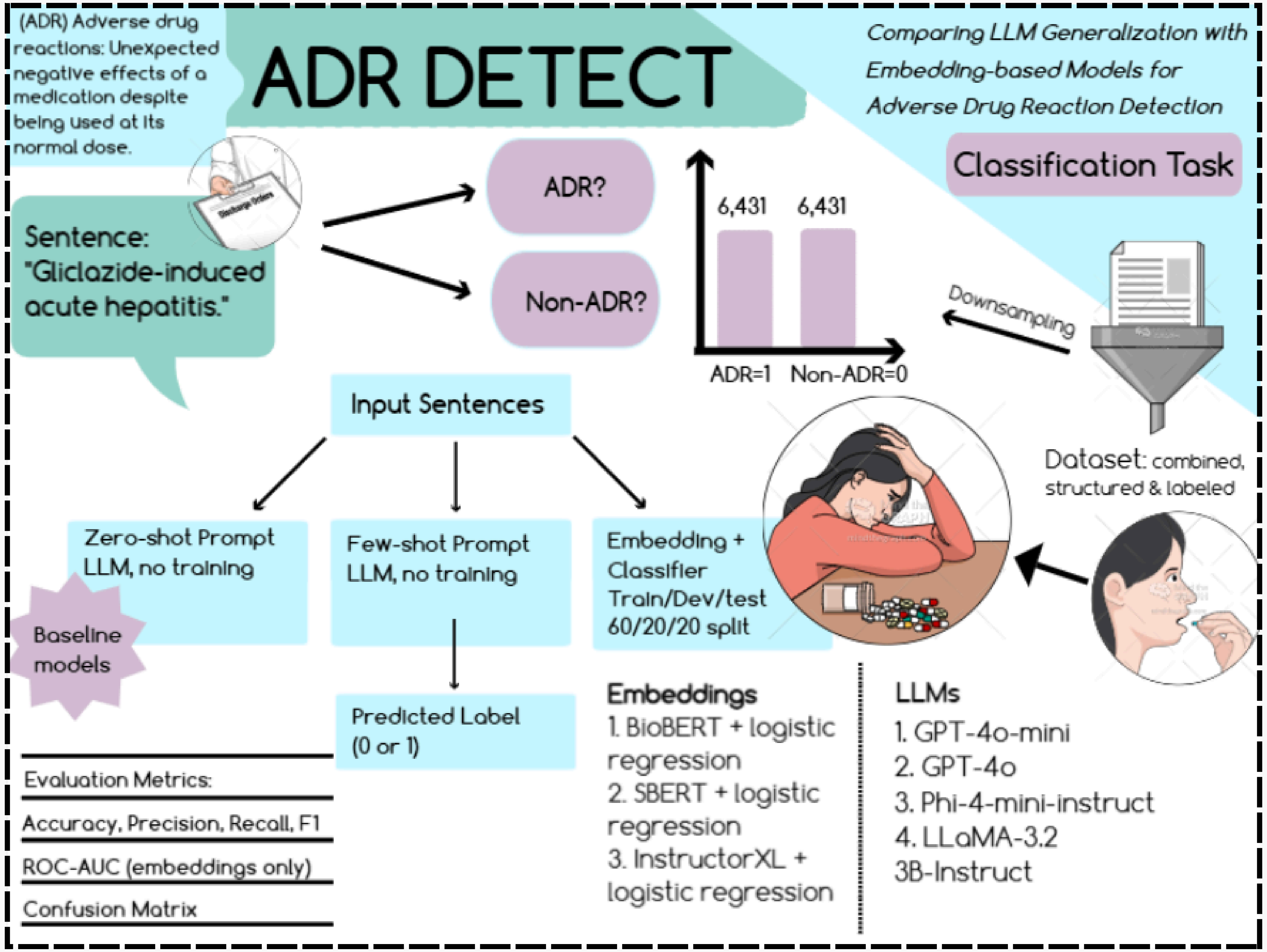


- **BioBERT and InstructorXL achieved the highest AUC** (0.90), suggesting strong separation between classes due to domain-specific and instruction-tuned embeddings.
- Despite its simplicity and lack of semantic understanding, the baseline model achieved a solid AUC of 0.84.

Conclusions

- Models prioritizing recall are most suitable for ADR detection to avoid missing critical cases.
- GPT-4o Zero-Shot is the optimal choice: it requires no training, delivers exceptional recall, and performs reliably across all metrics.
- Traditional models like BoW + Naive Bayes (F1 = 0.77) serve as a solid baseline, but were clearly outperformed by modern LLMs and embeddings.
- Prompting strategy impacts performance — the prompts were adjusted multiple times to improve results.
- Zero-Shot often outperformed Few-Shot, showing that more examples don’t always improve results.
- The LLaMA model performed worse than expected, but this is understandable: prior research typically uses fine-tuned or enhanced LLaMA variants, which were beyond the scope of our setup.

Visual Abstract Slide



(ADR) Adverse drug reactions: Unexpected negative effects of a medication despite being used at its normal dose.

ADR DETECT

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

Classification Task

Sentence: "Gliclazide-induced acute hepatitis."

ADR?

Non-ADR?

Dataset: combined, structured & labeled

Input Sentences

Zero-shot Prompt LLM, no training

Few-shot Prompt LLM, no training

Embedding + Classifier Train/Dev/test 60/20/20 split

Baseline models

Predicted Label (0 or 1)

Evaluation Metrics:

Accuracy, Precision, Recall, F1

ROC-AUC (embeddings only)

Confusion Matrix

Embeddings

1. BioBERT + logistic regression
2. SBERT + logistic regression
3. InstructorXL + logistic regression

LLMs

1. GPT-4o-mini
2. GPT-4o
3. Phi-4-mini-instruct
4. LLaMA-3.2 3B-Instruct

Thank you