ADR Detect

Interim report

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

By: Nicole Poliak 206479982 & Naveh Nissan 322292491

Description

Task

- Input: Sentence from PubMed case report
- Output: Binary label (0 = non-ADR, 1 = contains ADR)
- Goal: Compare two paradigms for biomedical sentence classification:
  (1) zero-/few-shot LLMs (GPT-4, Claude)
  (2) embedding-based classifiers (TF-IDF, SBERT+ Logistic Regression)

Data

- Dataset: ADE Corpus V2 (23,517 expert-annotated sentences from biomedical literature)
- Structure:
  - Text: Sentence describing a clinical case
  - Label: ADR present (1) / not present (0)
- Usage:
  - Embedding models: 60/20/20 stratified train/dev/test split
  - LLMs: Zero-shot and few-shot prompting (no fine-tuning)

Description

Evaluation

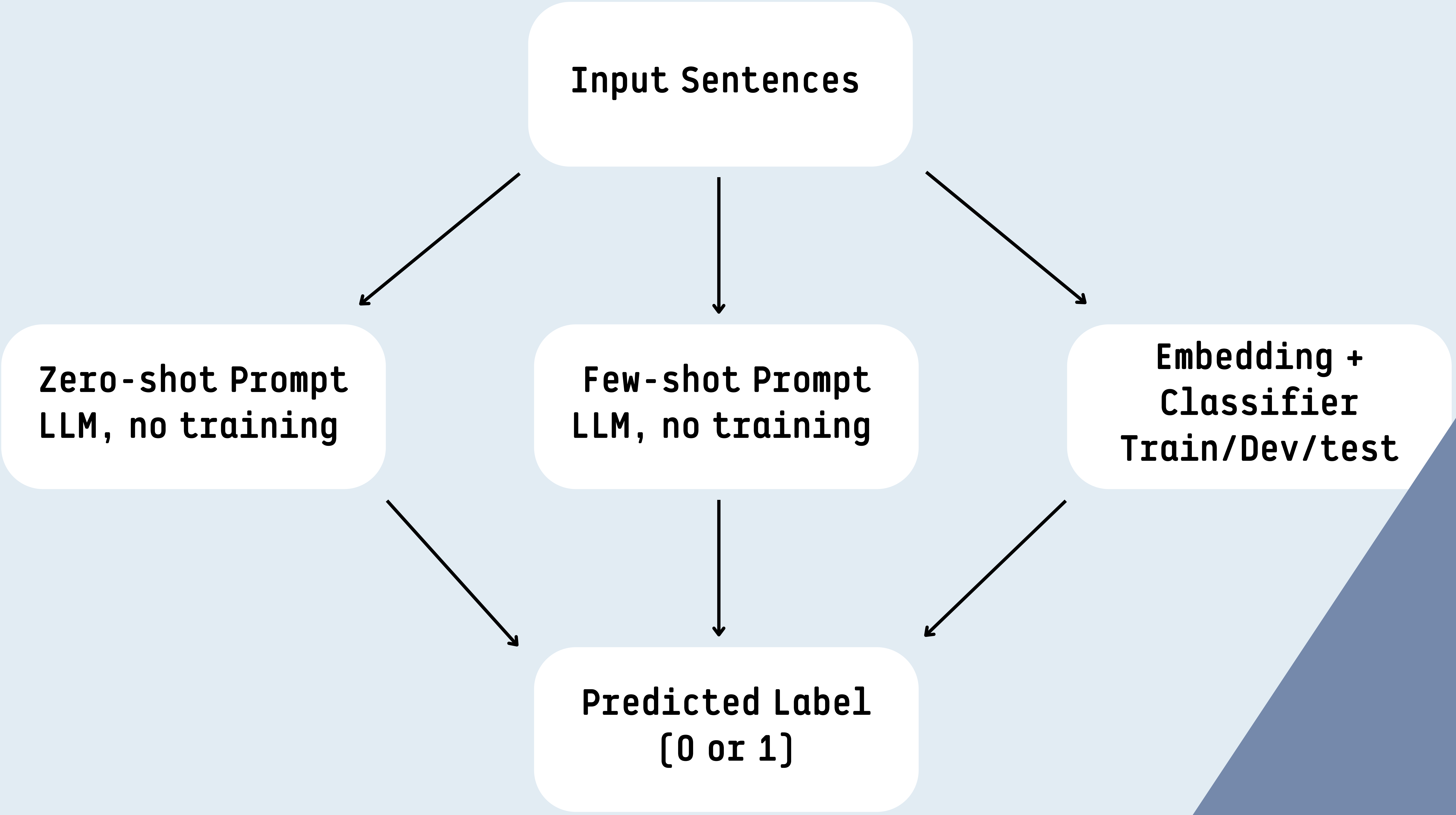
- Metrics: Accuracy, Precision, Recall, F1-Score
- Embedding Models: Classifiers trained on extracted embeddings
  - Baseline = Naïve Bayes + Bag-of-Words
- LLMs: Zero/few-shot without training
  - Baseline = GPT-4 Zero-shot performance

Prior Art

Source / Title	Approach / Model	Data	Metrics	Results
ModernBERT vs LLMs for Detecting Adverse Drug Reactions **Simmering.dev, 2025**	Comparison between: fine-tuned ModernBERT & few-shot Llama 3.2-3B using DSPy	ADE-Benchmark Corpus (23.5k labeled sentences)	Recall, Precision, F1 Score, Speed, Cost	Fine-tuned LLaMA 3.2-3B outperformed ModernBERT and few-shot LLMs
LLMs are not Zero-Shot Reasoners for Biomedical Information Extraction **ACL Anthology, 2025**	Evaluation of LLMs (BioMistral, Llama-2) using: standard prompting, CoT, Self-Consistency and RAG	PsyTAR dataset for ADR + Withdrawal Symptoms classification	F1 Score	Standard prompting outperformed advanced techniques; highlighted limitations of zero-shot LLMs in biomedical tasks
Using LLMs to Extract ADR's from Short Text **SCITEPRESS, 2025**	Application of GPT-4 with: zero-shot, few-shot and CoT prompting for ADR extraction	ASU-CHOP dataset, SMM4H dataset, WEB-RADRSMM4H dataset (tweets) and ADE Corpus v2	Recall, Precision, F1 Score	GPT-4 achieved high F1, competitive with prior state-of-the-art, in ADR binary classification on short texts

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
2b. TF-IDF Vectorization	Extract lexical features	Cleaned text → Sparse vector	TfidfVectorizer	No direct metrics (assessed in 3a)
2c. SBERT Embedding	Extract semantic features	Raw sentence → Dense vector (768D)	sentence-transformers/ all-MiniLM-L6-v2	No direct metrics (assessed in 3a)
3a. Classification (Embeddings)	Predict ADR label	Vector → Binary label	Logistic Regression	• Accuracy • Precision • Recall • F1-Score • ROC-AUC
3b. LLM Prompting	Prompt model to return label	Sentence + prompt → Binary label	GPT-4 / Claude (Zero-/Few-shot prompting)	• Accuracy • Precision • Recall • F1-Score
4. Evaluation	Compare model outputs to true labels	Predictions + gold labels → Scores	sklearn.metrics, 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 with clinical sentences
- 20,895 sentences, 2 classes (ADR(1), Non-ADR (0)) after cleaning & duplicates removal
- Minimal text cleaning: lowercasing, punctuation removal
- Mean sentence length: 17.8 ± 8.6 words (max 122 words)
- Notable class imbalance: ~80% Non-ADR, ~20% ADR → downsampled majority class to balance (4,271)

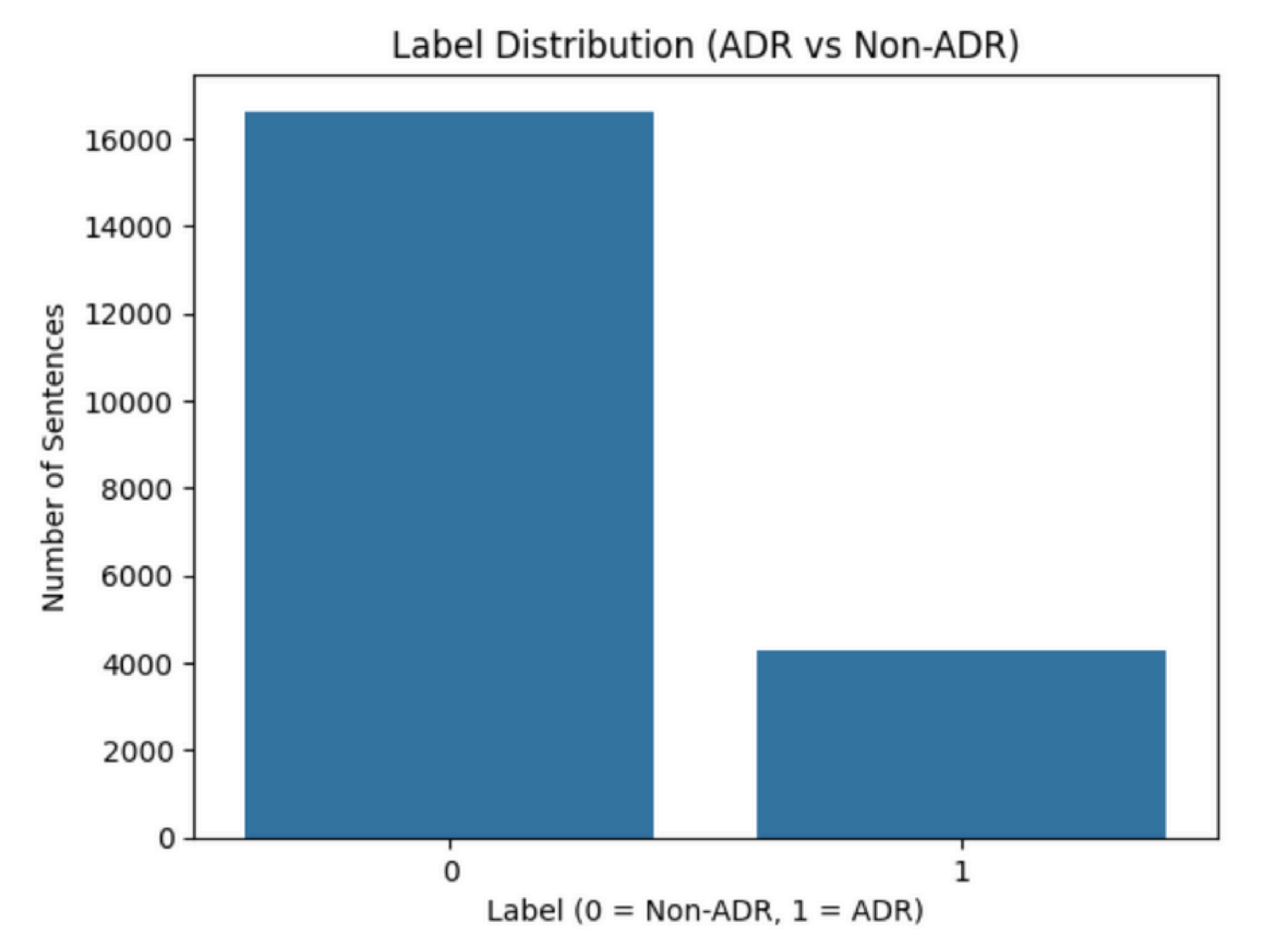
Baseline

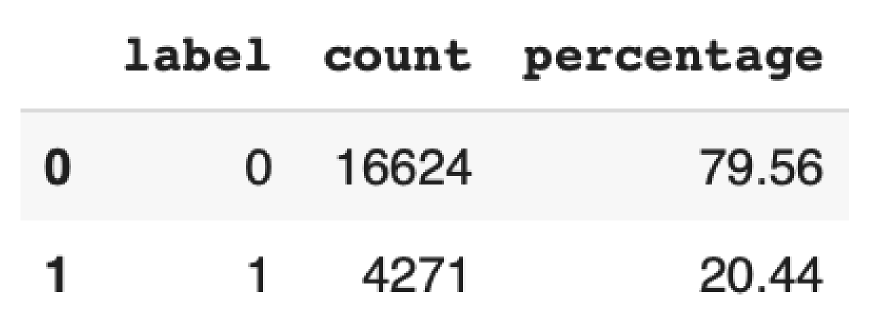
- Bag-of-Words (CountVectorizer) + Multinomial Naive Bayes
- 60/20/20 stratified train-dev-test split
- Performance →

  Dev: Accuracy = 0.77, Precision = 0.73, Recall = 0.85, F1 = 0.79

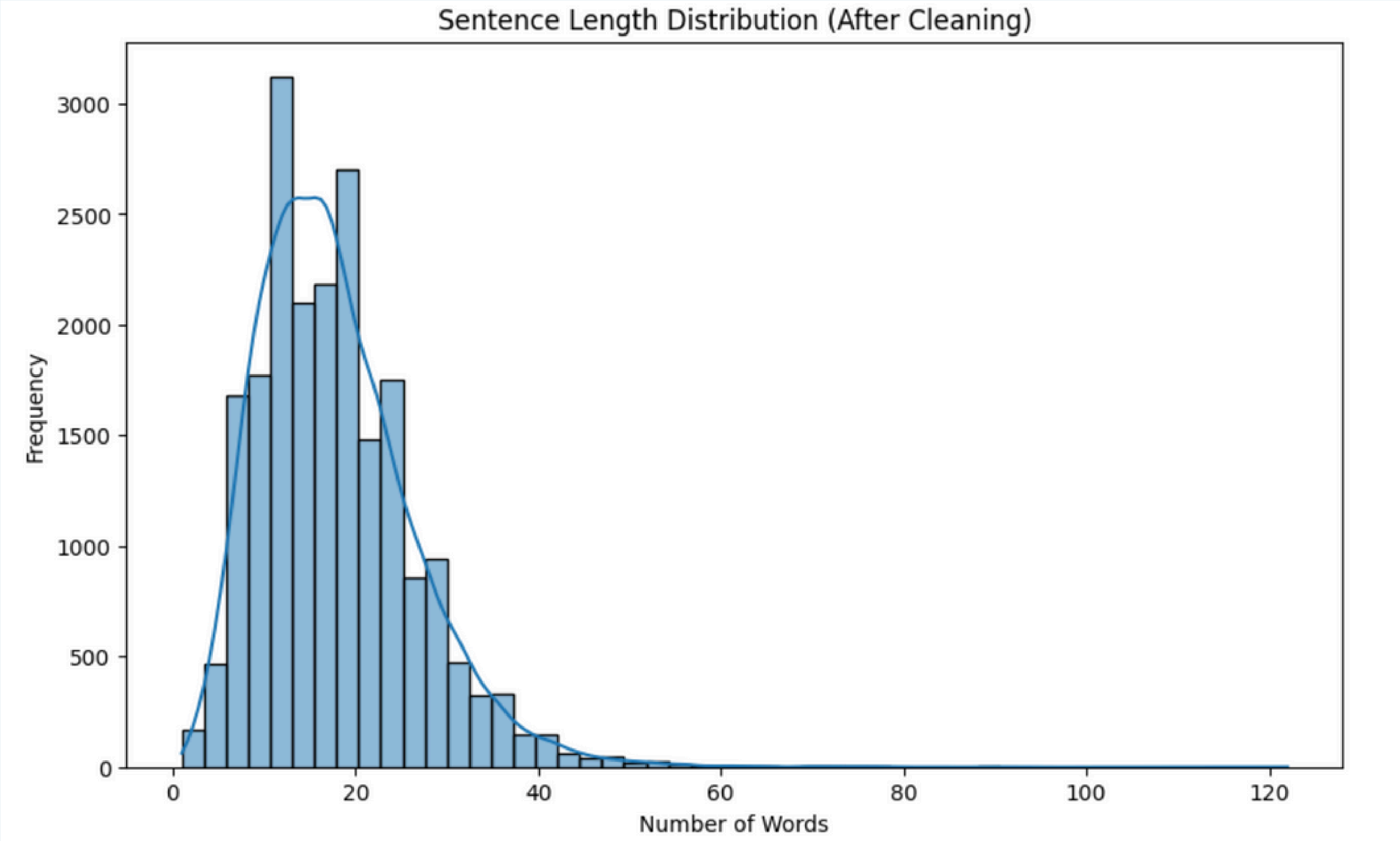
  Test: Accuracy = 0.77, Precision = 0.73, Recall = 0.84, F1 = 0.78
- Better performance in precision and recall compared to without downsample attempt

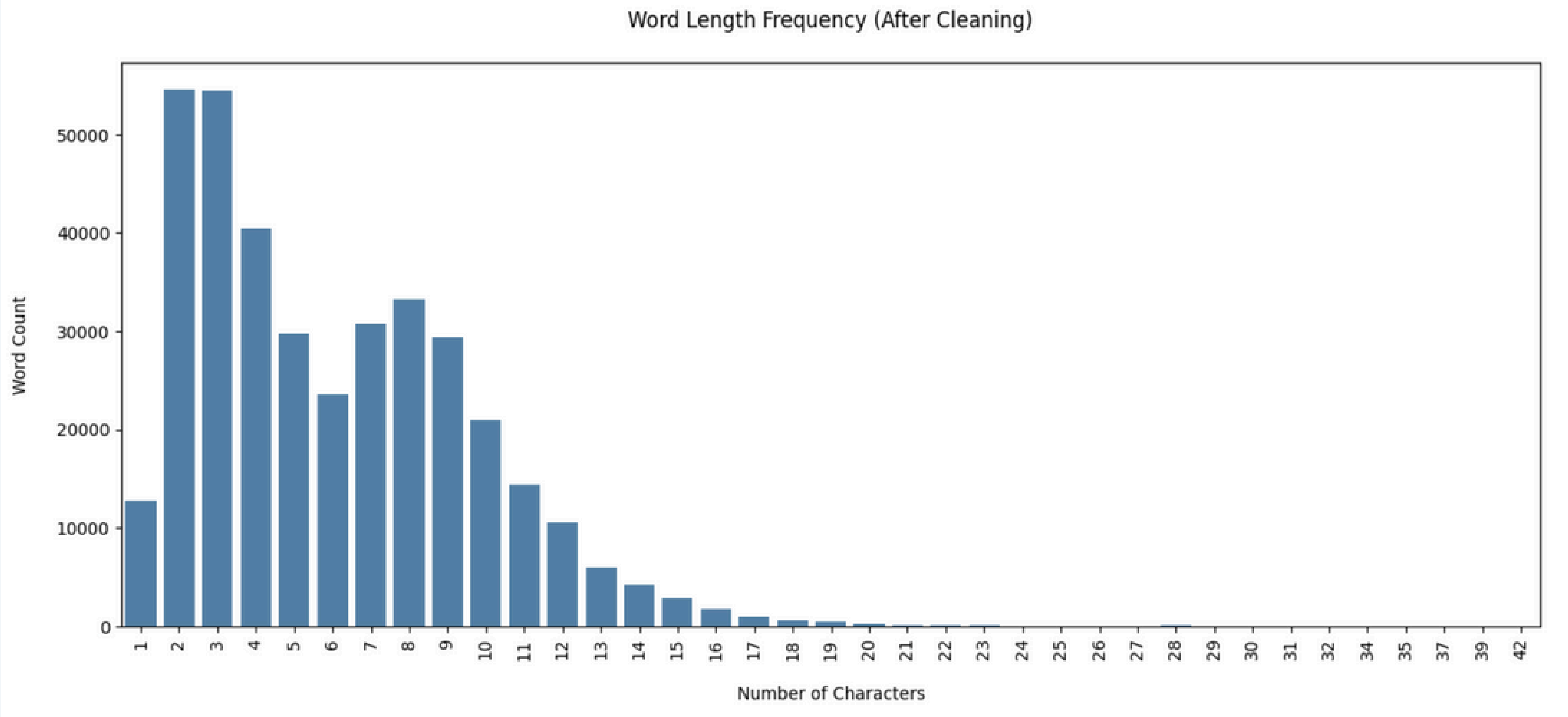
EDA Visualizations





	label	count	percentage
0	0	16624	79.56
1	1	4271	20.44





Total words - 371,956
Total letters - 2,234,037

Insights & Recommendations

01 **From Simmering.dev (2025):** Fine-tuned LLaMA 3.2-3B outperformed ModernBERT and few-shot LLMs → Our SBERT + classifier approach may outperform zero-/few-shot LLMs in precision/efficiency.

02 **From ACL Anthology (2025):** Advanced prompting (CoT, Self-Consistency) didn't always improve over basic prompting → Keep LLM prompting simple and controlled.

03 **From SCITEPRESS (2025):** GPT-4 few-shot prompting improved F1 on ADRs from tweets → supports including GPT-4 few-shot as a competitive LLM model, even without fine-tuning.

04 Mean sentence length = 17.8 words (Non-ADR has more outliers) → Sentences are short enough to fit within SBERT and LLM token limits.

05 Strong class imbalance (~80% Non-ADR) required downsampling → LLMs and SBERT models will be evaluated on the downsampled dataset.

06 Duplicate rows were removed before & after cleaning.

07 Level of Sensitivity (Recall) is the primary metric for our task - FN are more important than FP