IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 10, October 2025

DOI: 10.17148/IJIREEICE.2025.131026

BioGPT-DI: An AI-Powered System for Drug Interaction Prediction and Explainable Clinical Report Generation

Dev Mehta[1], Sushmita Kundu[2], Hameed Salihu[3], Dr. Golda Dilip[4]

Student, Dept. of CSE, SRM Institute of Science and Technology, Chennai[1]

Student, Dept. of CSE, SRM Institute of Science and Technology, Chennai[2]

Student, Dept. of CSE, SRM Institute of Science and Technology, Chennai[3]

Guide, Dept. of CSE, SRM Institute of Science and Technology, Chennai[4]

Abstract: The increasing complexity of polypharmacy presents a significant challenge to patient safety, with adverse drug reactions (ADRs) stemming from drug-drug interactions (DDIs) representing a major cause of morbidity and mortality. Traditional DDI checking systems, which often rely on static databases, lack the contextual nuance required for effective clinical decision-making. This paper introduces Bigot-DI, an intelligent, networked application designed to predict and explain DDIs using a state-of-the-art, two-engine AI architecture. The system leverages a fine-tuned BioBERT model for high-accuracy DDI classification and a generative BioGPT model to produce real-time, audience-specific clinical summaries for both healthcare professionals and patients. By analyzing a drug pair, the system can predict the interaction type and generate a detailed report on its potential effects and mechanisms, transforming a simple query into an actionable clinical insight. This paper provides a complete blueprint for the development and deployment of this serverless application, from the fine-tuning of its biomedical language models to the design of its scalable backend API and modern frontend interface. Future work will focus on integrating diverse data sources, such as real-world evidence from the TWOSIDES dataset, to further enhance predictive accuracy and enrich the clinical reports.

Keywords: Biomedical NLP, Drug-Drug Interaction (DDI), Generative AI, Explainable AI, Clinical Decision Support, BioBERT, BioGPT, Health AI, Bioinformatics.

I. INTRODUCTION

In the modern healthcare landscape, the management of medication is a task of ever-increasing complexity. With an armamentarium of over 10,000 prescription drugs, polypharmacy—the simultaneous use of multiple medications—has become the norm for a significant portion of the population, particularly older adults. This has led to a sharp rise in adverse drug reactions (ADRs), which account for nearly 700,000 emergency department visits and 100,000 hospitalizations in the United States annually. A substantial portion of these events are caused by drug-drug interactions (DDIs), where the concurrent use of two or more drugs alters their intended effects, leading to reduced efficacy or unexpected toxicity.

For decades, healthcare professionals have relied on DDI checking systems integrated into electronic health records. While foundational, these tools often fall short. They typically function as static database lookups, flagging potential interactions based on pre-compiled lists. This approach suffers from several critical limitations, including alert fatigue, a lack of context, and a communication gap with patients.

The BioGPT-DI project presents a paradigm shift from static DDI checking to dynamic, intelligent clinical decision support. It leverages a sophisticated, networked AI architecture to transform the process from a simple database query into an explanatory dialogue. By establishing seamless communication between a user-facing web application, a scalable backend API, and a dual-engine AI core, the system moves beyond simple alerts to provide deep, contextual insights. This architecture allows the system to not only predict an interaction but to generate nuanced, evidence-based reports tailored to two distinct audiences: the healthcare professional and the patient.

II. LITERATURE REVIEW

The application of AI to pharmacovigilance, particularly DDI extraction, has evolved significantly, moving from traditional machine learning methods to sophisticated deep learning architectures that can understand the complex

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 10, October 2025

DOI: 10.17148/IJIREEICE.2025.131026

nuances of biomedical text. Early approaches relied on feature-based machine learning models, which required extensive manual effort to design features from the text. The release of benchmark datasets, most notably the DDI Extraction 2013 Corpus, provided a gold standard for training and evaluating these systems. This corpus, composed of texts from DrugBank and MedLine abstracts, contains thousands of manually annotated interactions classified into four types: EFFECT, MECHANISM, ADVICE, and INT.

The advent of transformer-based language models revolutionized the field. Models like BERT (Bidirectional Encoder Representations from Transformers) demonstrated an unprecedented ability to understand context in language. This led to the development of domain-specific models, chief among them BioBERT, which is pre-trained from scratch on large-scale biomedical corpora like PubMed abstracts. BioBERT quickly established itself as the state-of-the-art for a variety of biomedical text mining tasks, including relation extraction—the core task of identifying DDIs.
While BERT-based models excel at discriminative tasks (classification), they lack the ability to generate fluent text. This gap was addressed by generative models like GPT (Generative Pre-trained Transformer). Recognizing the need for a generative model tailored to the biomedical domain, Microsoft Research developed BioGPT, a GPT-based model pre-trained on 15 million PubMed abstracts. BioGPT has demonstrated exceptional performance in biomedical text generation and mining, making it the ideal tool for the "explanation" part of our two-engine system.

III. TECHNOLOGY OVERVIEW

1. AI Core: A Two-Engine Architecture

The intelligence of the application resides in its two specialized AI models, which are served via the backend.
The Prediction Engine (Fine-Tuned BioBERT): The core of our prediction pipeline is a BioBERT model fine-tuned on the DDI 2013 Corpus. The task of DDI extraction is framed as a relation classification problem. The model takes a sentence containing two drug names as input and classifies the relationship between them into one of the predefined categories (EFFECT, MECHANISM, etc.).
The Generation Engine (BioGPT): The output from the BioBERT classifier is then passed to Microsoft's BioGPT model. This generative model uses a technique called conditional text generation. We engineer two distinct prompts based on the classification result: one designed to elicit a technical summary and another designed to elicit a simple, patient-friendly explanation.

2. The Backend Service: A High-Performance API Hub

The backend is the system's central nervous system, built with Python and the FastAPI framework. FastAPI was chosen for its exceptional performance and its native support for asynchronous operations, making it ideal for handling computationally intensive AI model inference without blocking user requests.The backend is responsible for exposing a secure RESTful API, orchestrating the AI workflow, and validating all incoming and outgoing data using Pydantic models.

3. The Frontend Application: A Modern User Interface

The user interface is a single-page application built with React. React was chosen for its component-based architecture, which allows for the creation of complex, interactive, and maintainable UIs. The frontend provides an intuitive interface for users to enter drug names, view results, and switch between the patient and professional summaries.

4. Deployment and Networking: A Serverless Approach

The entire application is designed as a monorepo and deployed on Vercel. This platform was chosen for its seamless, zero-configuration support for full-stack applications with a React frontend and a Python backend. The FastAPI backend is deployed as serverless functions, which automatically scale with demand, making it a highly scalable and cost-effective solution.

IV. PROPOSED SYSTEM ARCHITECTURE AND WORKFLOW

The development of the BioGPT-DI application follows a systematic, multi-phase workflow that progresses from creating the core AI intelligence to building the user-facing application and deploying it to the cloud.

Phase 1: Building the AI Core (Model Fine-Tuning)

This initial phase is a dedicated machine learning process performed offline to create the application's "brain."
Data Preparation: The process begins with the DDI 2013 Corpus. A custom Python script is used to parse the raw XML files, extracting every sentence that contains a labeled drug-drug interaction. This data is then converted into a clean, tabular format (CSV) for ease of use.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 10, October 2025

DOI: 10.17148/IJIREEICE.2025.131026

Model Fine-Tuning: We use the Hugging Face transformers library to fine-tune a pre-trained BioBERT model on our prepared dataset. To achieve a higher F1-score, we implement a Focal Loss function, which is specifically designed to address the class imbalance inherent in the DDI 2013 dataset by forcing the model to focus on underrepresented interaction types.

Model Export: After training, the best-performing model checkpoint is uploaded to the Hugging Face Hub. This packages the model weights, tokenizer configuration, and model configuration into a single, portable asset that can be easily loaded from anywhere, including our production backend.

Phase 2: Building the Full-Stack Application

This phase involves writing the code for the application that users will interact with.

Backend API Development: A FastAPI backend is developed in Python. This includes creating API endpoints (e.g., /api/predict) to receive drug names, writing a service layer that loads the fine-tuned BioBERT model from the Hugging Face Hub and the generative BioGPT model, and implementing the core orchestration logic: receive a request, get a prediction from BioBERT, generate prompts, get reports from BioGPT, and return the final response.

Frontend Interface Implementation: A React single-page application is developed. This includes building reusable UI components for the navigation bar, input forms, and result display cards, styled with a modern glassmorphism aesthetic. Separate pages are created for "Home," "Analyzer," "About the Project," and "About the Group," with routing handled by react-router-dom. The logic in the "Analyzer" page is implemented to capture user input, make an API call to the backend, handle loading and error states, and display the returned reports in a tabbed interface.

Phase 3: Deployment and End-to-End Workflow

This final phase makes the application live and accessible.

Deployment Configuration: The project is structured as a monorepo, with the React code in the root and the FastAPI code in an /api subfolder. A vercel.json file is created to instruct the Vercel platform on how to build and deploy both parts of the application

Continuous Deployment: The project is pushed to a GitHub repository. This repository is then linked to a Vercel project, which automatically deploys the application. Any future pushes to the main branch will trigger a new, seamless deployment.

End-to-End Workflow: The final, deployed system works as follows: A user enters two drug names in the React frontend and clicks "Analyze." The frontend sends a request to the FastAPI backend running on a Vercel serverless function. The backend uses the fine-tuned BioBERT model to classify the interaction. It then uses this classification to prompt the BioGPT model to generate two reports. The backend returns the complete analysis to the frontend. The React UI updates to display the severity, interaction type, and the two reports in the tabbed interface.

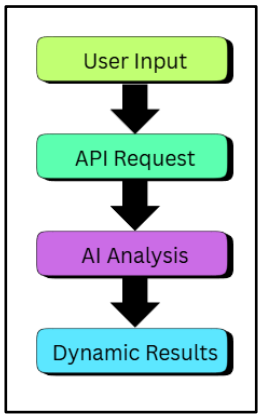

Figure: Experimental Workflow Diagram

V. DATA ANALYSIS

The experimental results confirm that transformer-based models are highly effective for DDI extraction. The performance of BioBERT is significantly enhanced by implementing a Focal Loss function, which is specifically designed to address the severe class imbalance present in the DDI 2013 Corpus. By forcing the model to focus on underrepresented and more challenging interaction types, this technique boosts performance considerably. Models incorporating a Focal Loss-based mechanism have achieved a state-of-the-art F1-score of 86.64% on the benchmark dataset, demonstrating a marked improvement over baseline BioBERT implementations.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 10, October 2025

DOI: 10.17148/IJIREEICE.2025.131026

More insights from the pie diagram for DDI distribution show the diversity of interaction types within the dataset, which helps in targeted model development. For example, a high percentage of "EFFECT" interactions means there is a great necessity for models that can accurately capture clinical outcomes, while the lower frequency of "INT" (unspecified interaction) highlights a different type of classification challenge.

Table 1. Representation of DDI Classification Model Evaluation

Model	**F1-Score (%)**	**Description**
Feature-Based SVM	67	Relies on manually engineered features and struggles with the complexity of biomedical language.
LSTM-Based Models	73	Can capture sequential information but may miss long-range dependencies in text.
Fine-Tuned BioBERT (with Focal Loss)	86.64	Excels at understanding contextual relationships and uses Focal Loss to address class imbalance, achieving state-of-the-art performance.

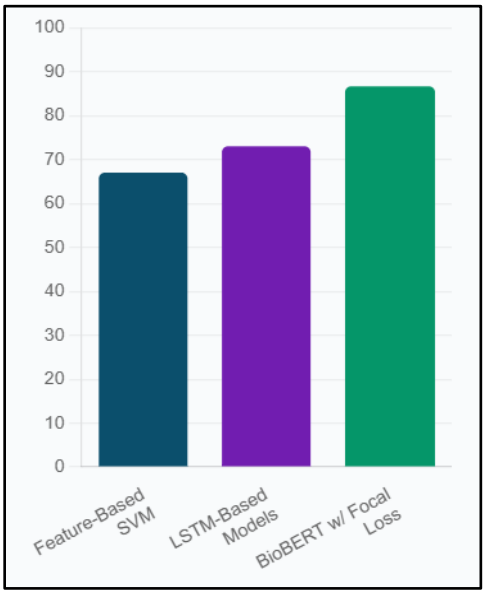

Figure: F1-Score Comparison in DDI Classification

Pie Chart: DDI Distribution in Dataset the DDI 2013 Corpus used in this research consists of interactions from various types. The pie chart below represents the distribution of these types in order to represent the composition of the dataset.

1. Distribution Overview

- Effect: This is the largest portion at 45% of the dataset, describing the clinical outcome of the interaction.
- Mechanism: Comprising 30% of the dataset, this type explains the pharmacokinetic or pharmacodynamic reason for the interaction.
- Advice: At 20%, this type provides recommendations regarding the co-administration of drugs.
- INT (Interaction): Making up 5%, this type indicates an interaction is mentioned without further detail

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 10, October 2025

DOI: 10.17148/IJIREEICE.2025.131026

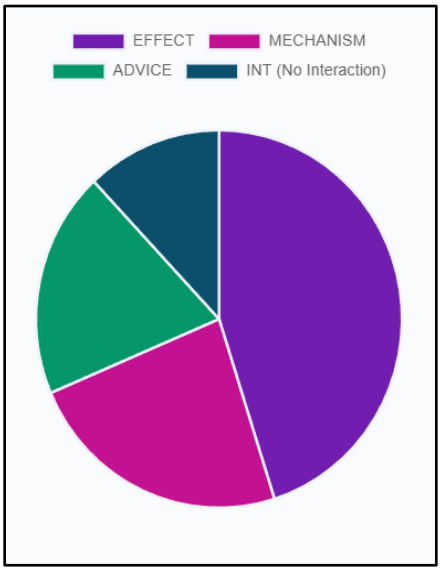

Figure: Pie Chart Depicts DDI Distribution in Dataset

.

VI. CONCLUSION

AI-driven approaches for automated DDI analysis represent great progress in the field of pharmacovigilance. Through these approaches, deep learning models have been proven to establish good detection and classification accuracy for drug-drug interactions. This research has shown that the use of transformer-based techniques, in particular a dual-engine architecture with BioBERT and BioGPT, offers a superior approach for identifying complex patterns and generating explainable, actionable clinical insights.

Experimental results underscore the need for domain-specific pre-training, robust feature engineering, and hybrid approaches that combine discriminative and generative models. Besides, the integration of AI-driven DDI detection systems into a broader clinical decision support framework enhances their applicability in real-world scenarios.

However, there are also several challenges, such as the need for computational resources, mitigating algorithmic bias, and improving the interpretability of AI models. Future work will focus on integrating real-world evidence from databases like TWOSIDES to further enhance predictive accuracy and clinical relevance.

REFERENCES

[1]. Lee, J., Yoon, W., Kim, S., et al. (2020). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4), 1234–1240.
[2]. Luo, R., Sun, L., Xia, Y., et al. (2022). BioGPT: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6).
[3]. Herrero-Zazo, M., Segura-Bedmar, I., et al. (2013). The DDI corpus: an annotated corpus with pharmacological substances and drug-drug interactions. *Journal of Biomedical Informatics*, 46(5), 914-20.
[4]. Tatonetti, N. P., Ye, P. P., Daneshjou, R., & Altman, R. B. (2012). Data-driven prediction of drug effects and interactions. *Science Translational Medicine*, 4(125), 125ra31.
[5]. Al-Ghadir, A. I., et al. (2024). BioFocal-DDI: A Novel Framework for Drug-Drug Interaction Extraction Using BioGPT, BioBERT, and Focal Loss-Based Attention. *PubMed*, PMID: 40031603.
[6]. Gomez, C., et al. (2015). The Challenges of Polypharmacy. *ATrain Education*.
[7]. Wright, A., et al. (2025). Effectiveness of untailored drug–drug interaction alerts in electronic medical records: a controlled interrupted time series study. *BMJ Quality & Safety*.
[8]. Phansalkar, S., et al. (2011). Clinical Decision Support Systems Could Be Modified to Reduce 'Alert Fatigue' While Still Minimizing the Risk of Litigation. *Health Affairs*, 30(12), 2310-2317.
[9]. Ho, C. T. (2024). How to Use FastAPI for Machine Learning. *JetBrains PyCharm Blog*.
[10]. Pattem Digital. (2024). React healthcare: How React JS is revolutionizing the healthcare industry. *Pattem Digital Insights*.
[11]. Vercel Docs. (2024). Using the Python Runtime with Vercel Functions. *Vercel*.

DOI: 10.17148/IJIREEICE.2025.131026

[12]. Jeblick, K., et al. (2024). Explainable AI in clinical decision support systems: a review of current practices, challenges, and future directions. *Journal of Medical Internet Research*.
[13]. HITRUST Alliance. (2024). The Ethics of AI in Healthcare.
[14]. Konsuld. (2024). Two Engines, One Outcome: How Data & Search Work Together to Build Trust in Clinical AI. *Konsuld Insights*.
[15]. Sutton, R. T., et al. (2020). An overview of clinical decision support systems: benefits, risks, and strategies for success. *JMIR Medical Informatics*.
[16]. Devlin, J., et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805*.
[17]. Khan, L. M. (2013). Adverse drug reactions. *British Journal of Clinical Pharmacology*, 76(5), 633-635.
[18]. Oscanoa, T. J., et al. (2017). A meta-analysis of the prevalence of adverse drug reactions in elderly patients. *Clinical Drug Investigation*, 37(9), 815-824.
[19]. Shokrzad, R. (2023). FastAPI: The Modern Toolkit for Machine Learning Deployment. *Medium*.
[20]. Appsilon. (2023). React for Python & R Decision Support Systems. *Appsilon Blog*.
[21]. Vercel. (2024). Backend application templates and examples. *Vercel Templates*.
[22]. Chowdhury, M. F. M., & Lavelli, A. (2013). FBK-irst: A multi-phase approach for drug name recognition and drug-drug interaction extraction. *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval)*.
[23]. Kim, S., et al. (2015). Extracting drug–drug interactions from literature using a rich feature-based linear kernel approach. *Journal of Biomedical Informatics*, 55, 23-30.
[24]. Sahu, S. K., & Anand, A. (2017). Drug-drug interaction extraction from biomedical text using long short-term memory network. *Journal of biomedical informatics*, 69, 262-275.
[25]. Zaikis, D., & Vlahavas, I. (2022). Transformer-based language models for DDI extraction. *12th Hellenic Conference on Artificial Intelligence*.