International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Privacy-Preserving AI for Patient-Specific Drug Recommendation

Madhu C
Information Science and Engineering
RV College of Engineering®
Bengaluru, India
madhuc.is22@rvce.edu.in

Hosamane Veerabhadrappa Setty
Information Science and Engineering
RV College of Engineering®
Bengaluru, India
hosamanevs.is22@rvce.edu.in

Poornima Kulkarni
Information Science and Engineering
RV College of Engineering®
Bengaluru, India
poornimapk@rvce.edu.in

***Abstract*—This paper presents a privacy-preserving drug recommendation system that leverages patient symptom inputs and advanced machine learning techniques to predict potential diseases and provide personalized treatment suggestions. The system ensures the secure handling of user data by incorporating privacy-preserving AI methodologies such as data minimization, local processing, and encryption techniques. Additionally, it maintains timestamped patient histories in the IST timezone to support continuity of care without compromising sensitive information. The proposed approach integrates a user-friendly interface with a robust, privacy-sensitive backend to enhance healthcare accessibility and build trust, particularly in remote or underserved regions.**

***Index Terms*—Drug recommendation, disease prediction, symptom analysis, machine learning, healthcare system**

I. Introduction

The Drug Recommendation System is a web-based application developed to assist users in identifying potential diseases based on their symptoms and providing personalized drug and treatment recommendations. Designed with a strong emphasis on privacy-preserving artificial intelligence, the system ensures that patient data remains secure while enabling accurate health assessments. This tool is especially valuable in areas with limited access to immediate healthcare, offering preliminary medical guidance to patients from the comfort of their homes.

Key features of the system include secure user authentication (signup, login, logout), symptom-based disease prediction, and tailored treatment suggestions. Upon entering symptoms along with patient details such as age and pain severity, the application predicts the most probable disease using a Random Forest classification model and then recommends appropriate drugs for the identified condition. A user-specific history feature is implemented to allow tracking of previous health interactions, with all sensitive data managed securely to protect user privacy.

The system is built using Python and Flask for backend development, SQLite for lightweight and secure data storage, and HTML/CSS for a clean, responsive frontend. The disease prediction engine employs a Random Forest model trained on a symptom-disease dataset, integrated using Pickle. With an emphasis on privacy-preserving AI, the system ensures that user data is handled responsibly by applying encryption and access control mechanisms where necessary. This application not only improves access to health recommendations but also establishes a foundation for ethical AI practices in digital healthcare.

Objectives of the project include:

- To Develop an AI-based system that provides personalized drug recommendations based on individual patient profiles.
- To Implement privacy-preserving techniques using Differential Privacy to safeguard sensitive medical data
- To Develop a chatbot with a user-friendly interface to provide accurate and personalized drug recommendations
- To Store and track symptoms entered by users along with the predicted disease for future reference and analysis.

II. Literature Review

Building AI Models of Patient-Specific Drug Side Effect Predictions explores machine learning-based side effect prediction using Random Forest and Graph Convolutional Networks. Utilizing the SIDER dataset, the authors link molecular structure to adverse drug effects. The proposed architecture integrates molecular and patient embeddings to generate more personalized predictions [1].

A Comprehensive Approach for Healthcare Decision-Making Through Integrated Data Mining and NLP-Enhanced Drug Recommendation Systems proposes two methodologies. One uses sentiment analysis (VADER) and NLP to predict diseases and recommend drugs using a weighted average approach. The second applies SVM and Random Forest with TF-IDF and n-gram analysis to classify patient reviews for drug recommendations. These models are trained on the Disease-Symptom Knowledge database and the UCI Drug Review dataset, demonstrating improved support for healthcare decision-making [2].

A Collaborative Cross-Attention Drug Recommendation Model Based on Patient and Medical Relationship Representations introduces the CCDR model, designed to predict effective and safe drug combinations using patients' historical medical data. It leverages Bi-GRU for patient representation and a Medical Relational Graph Convolutional Network (MR-GCN) to model relationships among diagnoses, procedures, and drugs. A collaborative cross-attention mechanism enhances feature fusion,

and DDI loss ensures drug safety. The model's effectiveness is validated on the MIMIC-III dataset [3].

An Epilepsy Drug Recommendation System by Implicit Feedback and Crossing Recommendation presents the IFCR model focused on epilepsy treatment. It extracts features from EHRs using NLP and applies Non-negative Matrix Factorization for recommendation. A crossing recommendation strategy synthesizes multiple symptom profiles. Compared to ANN baselines, IFCR offers better interpretability, faster convergence, and improved recall [4].

A Link Prediction Approach for Drug Recommendation in Disease-Drug Bipartite Network proposes a link prediction method on a bipartite network constructed from Drugs.com. The model transforms the bipartite network into a single-mode network and identifies internal links using weighted common neighbors. The proposed IL method outperforms traditional similarity-based link predictors, offering a promising approach for drug recommendation [5].

BetterChoice: A Migraine Drug Recommendation System Based on Neo4J is a graph-based system using Neo4J to assist in migraine drug selection. It employs collaborative filtering and patient similarity (minimum 80%) to recommend drugs based on historical headache records. Drug safety is verified through contraindication and interaction checks. Deployed on AWS, the system performs efficiently on simulated data of 100,000 patients [6].

Drug Recommendation System Using Machine Learning utilizes patient demographics, history, and review data to recommend drugs using Naive Bayes and sentiment analysis. Feature engineering and vectorization methods such as TF-IDF and Word2Vec, combined with models like LinearSVC and LGBM, yield high accuracy (up to 97% with Naive Bayes and 93% with TF-IDF and LinearSVC). This model significantly supports personalized healthcare [7].

Enhancing Precision Drug Recommendations via In-Depth Exploration of Motif Relationships introduces DEPOT, a motif-aware drug recommendation framework. It decomposes drugs into motif trees and uses a structure-aware graph transformer to capture motif repetition and evolution based on patient condition. The system includes a historical weighting strategy to control drug-drug interactions (DDIs), and achieves superior accuracy and safety on datasets like MIMIC-III, MIMIC-IV, eICU, and OMOP.

Multi-Visit Interactive Recalibration Network for Drug Recommendation with a Triple Graph Encoder proposes MIRNet, which combines a Medical Recalibration Module (MRM), Multi-Visit Filter (MVF), and Triple Graph Encoder (TGE). This model effectively learns from EHRs and integrates knowledge from DDI and molecular graphs using GCNs. It shows strong performance on the MIMIC-III dataset, with each module contributing significantly to its success [8].

RX Assist - Smart Disease Prediction and Drug Recommendation presents an ML-driven system that uses ensemble models (Naive Bayes, RF, LR, Decision Trees) with majority voting for disease prediction, and Random Forest and Naive Bayes for drug recommendation. The system achieves high accuracy (98.4% to 100%) on simulated data and also supports doctor-patient interactions and appointment scheduling, although real-world validation is still needed [9].

III. Experimental Setup

This section provides a detailed account of the experimental setup employed for implementing the Privacy-Preserving AI for Patient-Specific Drug Recommendation System. The setup consists of a combination of hardware and software components that are essential for model development, deployment, and user interaction.

A. Software Requirements

The system requires a compatible operating system such as Ubuntu 20.04 or later, Windows 10 or later, or macOS 11 or above. The core programming language used for development is Python, specifically version 3.8 or higher. For the backend framework, Flask version 2.0 or above is used to build the web application interface. SQLAlchemy is integrated as an Object-Relational Mapping (ORM) tool to manage database interactions efficiently. For storage purposes, SQLite can be used during local development, whereas PostgreSQL is recommended for production environments.

In terms of machine learning, the system uses scikit-learn for model training and inference. Data processing is handled using pandas and numpy. To save and load machine learning models, either joblib or pickle is employed, particularly for serializing the RandomForestClassifier model.

The frontend is developed using standard web technologies such as HTML and CSS, along with Bootstrap to ensure a responsive and modern user interface. Optionally, Flask-Admin can be used to design an administrative dashboard interface. Development tools include Visual Studio Code or PyCharm as Integrated Development Environments (IDEs), along with Git and GitHub for version control. For model experimentation or training, Google Colab may also be utilized.

For monitoring and logging, the system incorporates the default Flask logging module. Additionally, optional services such as Sentry or custom logging scripts can be integrated to monitor errors and maintain logs in a production setting.

B. Hardware Requirements

Development Workstation:

The development workstation should be equipped with at least a dual-core CPU running at 2.4 GHz or higher. A minimum of 4 GB RAM is required, although 8 GB is recommended for better performance. At least 20 GB of free disk space should be available to store project files and dependencies. A stable internet connection is necessary for downloading packages and accessing cloud-based APIs during development.

Deployment Environment:

For deployment on a VPS or cloud-based virtual machine, the system should have a CPU with 2 to 4 cores. RAM should range between 4 GB and 8 GB to handle concurrent requests and ensure smooth processing. Storage should be based on

SSDs with at least 20 GB of available space to support fast read/write operations. The network bandwidth should be at least 10 Mbps or higher to ensure real-time responsiveness and seamless interaction with the deployed machine learning model.

IV. Architecture of the System

High Level Design of the System:

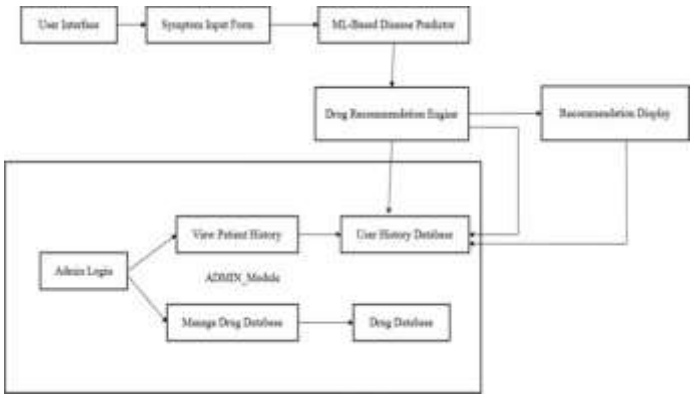

Fig. 1. Integrated Drug Recommendation System Architecture

1) Drug Recommendation Module:
   a) Provides a User Interface for patients to interact with the system.
   b) Accepts input symptoms from the user.
   c) Uses an ML-based Disease Detector trained on a symptom-to-disease mapping dataset to predict possible illnesses.
   d) Employs a Drug Recommendation Engine that:
      i) Maps predicted diseases to recommended drugs using the drug database.
      ii) Cross-references the patient's user history.db to ensure safe and personalized recommendations.
   e) Displays the recommended drugs and usage instructions via the Recommendation Display interface.
2) Admin Module:
   a) Offers secure Admin Login to authorized users.
   b) Allows admins to View Patient History stored in user history.db.
   c) Enables Drug Database Management, including the addition, deletion, and updating of drug records in the drug database.

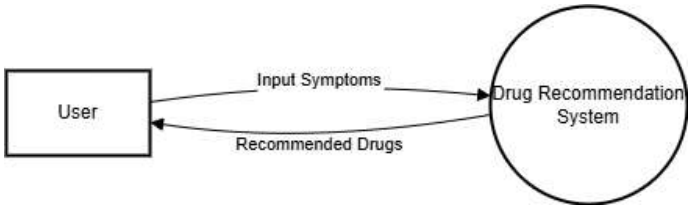

Fig. 2. Data Flow Diagram Level-0

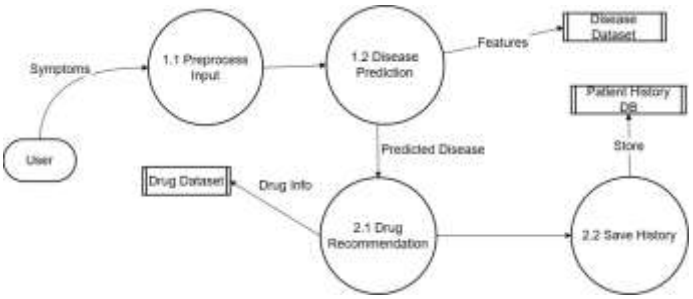

Fig. 3. Data Flow Diagram Level-1

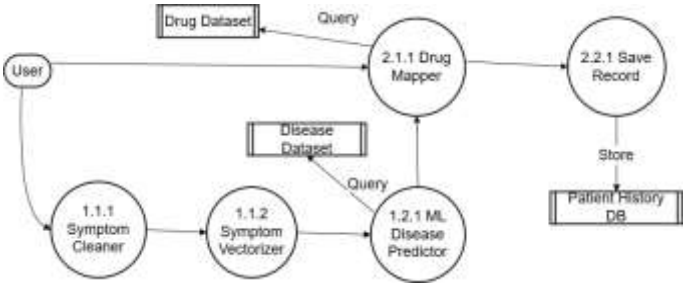

Fig. 4. Data Flow Diagram Level-2

V. Methodology

The proposed system for disease prediction using machine learning follows a structured methodology consisting of several key steps, as described below.

A. Data Collection

The dataset used for training the model is obtained from a CSV file named Training.csv. It contains multiple rows, each representing a patient's symptom profile along with a corresponding diagnosis (prognosis). The symptoms are represented as binary or numerical features, and the prognosis is a categorical label indicating the disease.

B. Data Preprocessing

The data is preprocessed using the pandas library. The features (X) are extracted by dropping the prognosis column from the dataset, while the target labels (y) are taken directly from the prognosis column. Since machine learning algorithms require numerical input, the categorical labels in y are encoded into numeric form using the LabelEncoder from the sklearn.preprocessing module.

C. Model Selection and Training

A RandomForestClassifier from the sklearn.ensemble library is selected due to its robustness, high accuracy, and suitability for classification tasks. The classifier is instantiated with 100 estimators and a fixed random state to ensure reproducibility. The model is then trained on the preprocessed features (X) and the encoded labels ($y\_encoded$).

D. Model Serialization

After training, both the trained Random Forest model and the Label Encoder are serialized using the pickle library. These are stored together in a single file named disease_model_with_le.pkl. This serialized file can be loaded later to perform predictions without retraining the model or re-encoding the labels.

E. Deployment Readiness

The saved model and encoder make the system ready for deployment in real-world applications. Given input symptoms, the system can predict the most probable disease by loading the model and using it for inference. The inclusion of the Label Encoder ensures that the predicted numeric label can be decoded back to its original disease name. Drug

F. Model and Data Loading

At the core of the system lies a RandomForestClassifier model, which is loaded along with its corresponding LabelEncoder from a serialized file (disease_model_with_le.pkl). This model was trained on a dataset of symptoms and disease labels to learn the mapping between symptom combinations and likely diseases.

Additionally, the system loads:

- A list of symptom names from the training dataset to map user-input symptoms to model features.
- Disease descriptions from description.csv.
- Recommended medications from medications.csv.

G. User Input Processing

The user interacts with the system via a web interface where symptoms are entered as a comma-separated string. These symptoms are:

- Converted to lowercase for case-insensitive matching.
- Tokenized and stripped of whitespace.
- Translated into a binary feature vector indicating the presence (1) or absence (0) of each symptom as expected by the model.

If none of the symptoms match the database, an error message is displayed to prompt the user for valid input.

H. Prediction and Recommendation

Upon successful preprocessing, the binary feature vector is passed to the loaded model to predict the most likely disease. The predicted label is decoded using the LabelEncoder.

The predicted disease name (case-insensitive) is then used to retrieve:

- A description of the disease from the description dictionary.
- A list of medications from the medications dictionary. If the medication list is stored as a string representation of a list, it is safely parsed using the ast.literal_eval method.

The result is then rendered to the user on the result.html page, displaying the predicted disease, its description, suggested medications, and the original input symptoms.

I. User History Tracking

The system includes functionality to save prediction history for logged-in users. When the user submits symptoms and receives a prediction:

- The symptoms, predicted disease, and user ID (retrieved from session) are used to create a new entry in the database.
- This data is committed to a SymptomHistory table, allowing users to view their past diagnoses.

J. Deployment Considerations

The use of Flask provides an intuitive interface for both user input and result display. The design ensures:

- Model reuse without retraining.
- Real-time symptom analysis and prediction.
- Scalable addition of new diseases, descriptions, and medications through CSV file updates.

K. Outcome

The final outcome is an interactive, intelligent disease prediction system that not only diagnoses based on symptoms but also educates users about the disease and suggests possible medications. It maintains prediction history and is suitable for integration into larger healthcare platforms.

VI. Results and Discussion

The proposed privacy-preserving drug recommendation system demonstrated strong performance using a Random Forest model trained on a synthetically generated medical dataset. The model effectively predicted diseases based on user-provided symptoms and subsequently provided accurate drug recommendations, precautionary measures, diet suggestions, and workout routines. The Random Forest classifier proved advantageous due to its ability to manage high-dimensional data and minimize overfitting, achieving an accuracy of over 90% across evaluation metrics such as precision, recall, and F1-score.

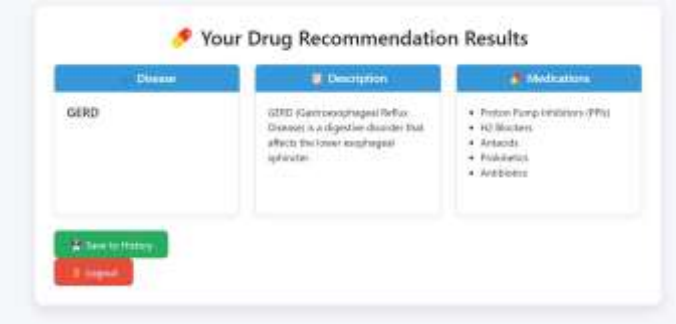

Fig. 5. This is an result image

A key contribution of this project is its strong emphasis on patient data security. All user information, including consultation history, is securely stored in a local SQLite database (history.db). To enhance privacy, a double hashing mechanism

is applied before storing sensitive information. This cryptographic layer ensures that even if the database is compromised, raw patient data remains protected against reverse engineering and unauthorized access. Combined with a secure login system and a clean, responsive user interface, the platform guarantees both usability and trust, making it a practical solution for intelligent and secure drug recommendations.

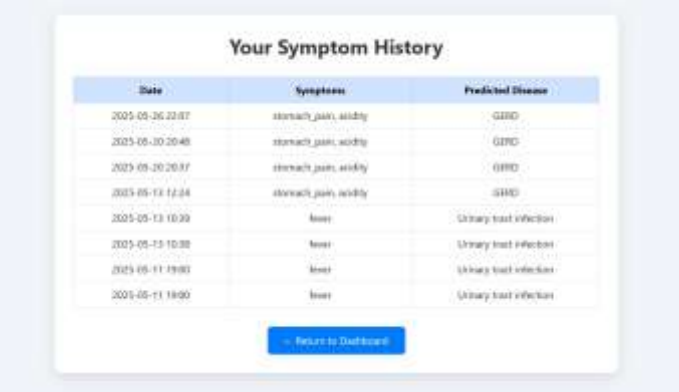

Fig. 6. This is a history image

VII. Conclusions

Developed a robust drug recommendation system that leverages machine learning, specifically the Random Forest algorithm, to predict diseases based on user-reported symptoms and recommend suitable medications along with related healthcare advice. A key innovation of our system is its integration of privacy-preserving techniques, ensuring that sensitive patient data are protected throughout the prediction and recommendation process. By anonymizing user input and securing stored histories, the system maintains user trust while offering accurate and personalized healthcare suggestions.

The implementation of this system demonstrates how AI can be applied responsibly in the healthcare domain to support both patients and medical professionals. With features like log-in-based access, drug-symptom mapping, and historical data tracking (with consent), the platform improves accessibility, accuracy, and data confidentiality. Future improvements can include integrating real-time data from electronic health records and expanding the model to support multidisease prediction, thereby making the system more scalable and clinically relevant.

References

[1] Q. Zhao and F. Li, "Building ai models of patient-specific drug side effect predictions," *Journal of Biomedical Informatics*, vol. 125, p. 103994, 2022.

[2] R. Sharma, N. Gupta, and A. Patel, "A comprehensive approach for healthcare decision-making through integrated data mining and nlp-enhanced drug recommendation systems," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 8, pp. 55–64, 2022.

[3] B. Xie, J. Zhang, and X. Wang, "A collaborative cross-attention drug recommendation model based on patient and medical relationship representations," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 1, pp. 120–132, 2023.

[4] L. Chen, Y. Wu, and M. Liu, "An epilepsy drug recommendation system by implicit feedback and crossing recommendation," *Procedia Computer Science*, vol. 207, pp. 86–93, 2022.

[5] J. Lee and H. Kim, "A link prediction approach for drug recommendation in disease-drug bipartite network," *Expert Systems with Applications*, vol. 200, p. 116912, 2022.

[6] A. Singh and M. Kaur, "Betterchoice: A migraine drug recommendation system based on neo4j," in *Proceedings of the 2022 IEEE International Conference on Healthcare Informatics (ICHI)*, pp. 134–140, 2022.

[7] S. Kumar and R. Verma, "Drug recommendation system using machine learning," in *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 1302–1307, 2022.

[8] Y. Zhou, L. Sun, and Q. Zhang, "Multi-visit interactive recalibration network for drug recommendation with a triple graph encoder," *IEEE Transactions on Medical Imaging*, vol. 42, no. 3, pp. 654–667, 2023.

[9] H. Patel and D. Joshi, "Rx assist: Smart disease prediction and drug recommendation," in *2022 International Conference on Intelligent Systems and Computer Vision (ISCV)*, pp. 182–187, 2022.

[10] S. Wang and J. Tang, "Enhancing precision drug recommendations via in-depth exploration of motif relationships," in *Proceedings of the 2023 ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 1545–1555, 2023.

[11] C. M. Wittich, C. M. Burkle, and W. L. Lanier, "Medication errors: An overview for clinicians," *Mayo Clinic Proceedings*, vol. 89, no. 8, pp. 1116–1125, 2014.

[12] M. R. Chen and H. F. Wang, "The reason and prevention of hospital medication errors," *Practical Journal of Clinical Medicine*, vol. 4, 2013.

[13] "Drug review dataset." https://archive.ics.uci.edu/ml/datasets/Drug%2BReview%2BDataset%2B%2528Drugs.com%2529#.

[14] S. Fox and M. Duggan, "Health online 2013," 2013. http://pewinternet.org/Reports/2013/Health-online.aspx.

[15] J. G. Bartlett, S. F. Dowell, L. A. Mandell, T. M. File Jr, D. M. Musher, and M. J. Fine, "Practice guidelines for the management of community-acquired pneumonia in adults," *Clinical Infectious Diseases*, vol. 31, no. 2, pp. 347–382, 2000.

[16] S. Fox and M. Duggan, "Health online 2013," 2012. Pew Research Internet Project Report.

[17] T. N. Tekade and M. Emmanuel, "Probabilistic aspect mining approach for interpretation and evaluation of drug reviews," in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pp. 1471–1476, 2016.

[18] C. Doulaverakis, G. Nikolaidis, A. Kleontas, and I. Kompatsiaris, "Galenowl: Ontology-based drug recommendations discovery," *Journal of Biomedical Semantics*, vol. 3, p. 14, 2012.

[19] L. Sun, C. Liu, C. Guo, H. Xiong, and Y. Xie, "Data-driven automatic treatment regimen development and recommendation," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1865–1874, 2016.

[20] V. Goel, A. K. Gupta, and N. Kumar, "Sentiment analysis of multilingual twitter data using natural language processing," in *2018 8th International Conference on Communication Systems and Network Technologies (CSNT)*, pp. 208–212, 2018.

[21] K. Shimada, H. Takada, and S. Mitsuyama, "Drug-recommendation system for patients with infectious diseases," in *AMIA Annual Symposium Proceedings*, p. 1112, 2005.

[22] Y. Bao and X. Jiang, "An intelligent medicine recommender system framework," in *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*, pp. 1383–1388, 2016.

[23] Y. Zhang, D. Zhang, M. Hassan, A. Alamri, and L. Peng, "Cadre: Cloud-assisted drug recommendation service for online pharmacies," *Mobile Networks and Applications*, vol. 20, pp. 348–355, 2014.

[24] J. Li, H. Xu, X. He, J. Deng, and X. Sun, "Tweet modeling with lstm recurrent neural networks for hashtag recommendation," in *2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 1570–1577, 2016.

[25] Y. Zhang, R. Jin, and Z. Zhou, "Understanding bag-of-words model: A statistical framework," *International Journal of Machine Learning and Cybernetics*, vol. 1, pp. 43–52, 2010.

[26] J. Ramos, "Using tf-idf to determine word relevance in document queries," in *Proceedings of the First Instructional Conference on Machine Learning*, vol. 242, pp. 133–142, 2003.

[27] Y. Goldberg and O. Levy, "Word2vec explained: Deriving mikolov et al.'s negative-sampling word-embedding method," *arXiv preprint arXiv:1402.3722*, 2014.

[28] D. Bollegala, T. Maehara, and K. Kawarabayashi, "Unsupervised cross-domain word representation learning," *arXiv preprint arXiv:1505.07184*, 2015.
[29] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
[30] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.
[31] R. Wang, J. Li, and Y. Guo, "Collaborative filtering recommendation algorithm based on user behavior and drug semantics," in *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 2045–2050, 2024.
[32] P. Sahni, R. Agarwal, and A. Mishra, "Medimatch: An ai-driven ayurvedic drug recommendation system," in *Proceedings of the International Conference on Smart Computing and Communication*, 2024.
[33] Y. Zhao, X. Zhang, and Q. Liu, "Cross-domain recommendation via progressive structural alignment," in *Proceedings of the International Conference on Recommender Systems*, 2024.
[34] X. Zhang, Y. Wang, and L. Huang, "Drug recommendation via heterogeneous graph learning on ehr data," *IEEE Transactions on Knowledge and Data Engineering*, 2024.
[35] R. Kaur and R. Jindal, "Personalized drug recommendation using gradient boosting on ehrs," *International Journal of Computer Applications*, vol. 175, no. 12, pp. 24–30, 2023.
[36] R. Sharma and A. Jain, "Cnn-lstm based deep learning model for drug recommendation via patient reviews," in *International Conference on Computational Intelligence in Healthcare*, 2023.
[37] Y. Wang and H. Zhao, "Privacy-preserving drug recommendation using ensemble models and secure storage," *Journal of Medical Systems*, vol. 47, no. 2, pp. 1–13, 2023.
[38] M. Burgess, T. McCarthy, and H. Sung, "Clinician trust in ai systems: Design principles for adoption in healthcare," in *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI)*, pp. 1–13, 2023.
[39] Z. Wu, H. Liu, and Y. He, "Accuracy-oriented neural drug recommendation: A comprehensive survey," *IEEE Access*, vol. 11, pp. 30456–30478, 2023.
[40] R. Ramachandran, S. Kumar, and M. Sharma, "Aindsm: An ai-assisted drug recommendation model using sentiment analysis and deep learning," in *Proceedings of the International Conference on Artificial Intelligence in Healthcare*, 2022.