International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 09 | Sept – 2025 DOI: 10.55041/ISJEM05027
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Drug Overdose Prediction Using Machine Learning

[1]P. BINDHU PRIYA, [2]MUDILI LAVANYA
[1]Assistant Professor, [2]2 MCA Final Semester Master of Computer Applications,
Sanketika Vidya Parishad Engineering College, Visakhapatnam, Andhra Pradesh, India.

Abstract:
Drug overdose is now the leading cause of death for those under 50 in the World.Inadequate data present a challenge for city officials, which prevents them from investigating the scale of the opioid overdose crisis.Various factors need to be considered in the prediction model for estimating the level of drug consumption, type of drug, and the location of the affected area.The aim of this project is to investigate several prediction and analysis models for forecasting drug use and overdoses by considering diverse data obtained from different sources, including sewage-based drug epidemiology, healthcare data, social networks data mining, and police data.Such analysis will help to formulate more effective policies and programs to combat fatal opioid overdoses.This project aims to explore, develop, and evaluate various prediction and analysis models to forecast drug usage trends and overdose risks by integrating diverse datasets. The data sources include sewage-based drug epidemiology, healthcare records, social media and network data mining, and police and law enforcement reports. Machine learning algorithms and statistical modeling will be employed to estimate the level of drug consumption, identify high-risk substances, and pinpoint geographical regions most affected by the crisis.

Index Term: Drug Overdose Prediction, Opioid Crisis, Machine Learning, Predictive Analytics, Sewage-based Epidemiology, Data Mining, Healthcare Data, Social Media Analysis, Law Enforcement Data, Public Health Policy.

I. INTRODUCTION

Narcotic drug consumption is a serious global problem with harmful effects on health, society, and the economy. The misuse of substances like heroin and cocaine can lead to addiction, mental health disorders, and increased risk of infectious diseases. It is also associated with higher crime rates, loss of workplace productivity, and heavy healthcare costs. Early detection and prevention are therefore crucial in addressing this issue effectively. Traditional approaches such as surveys and interviews are often slow, costly, and unable to capture complex behavioral patterns. With advancements in Artificial Intelligence (AI) and Machine Learning (ML), new solutions have become possible. ML techniques can analyze large datasets and identify hidden patterns linked to drug use. By studying demographic, psychological, and behavioral data, these models can predict individuals at higher risk. Such predictions help healthcare professionals, educators, and policymakers take timely preventive measures. Ultimately, ML offers an efficient and data-driven approach to reduce the impact of narcotic drug abuse.

1.1 EXISTING SYSTEM

- Surveys and Questionnaires conducted by healthcare organizations, law enforcement agencies, and researchers.
- Manual Data Analysis governments and institutions study statistical reports on drug usage trends in different demographics.
- This is time-consuming manual surveys and evaluations take a long time to process
- Rehabilitation and Counseling Assessments experts assess individuals based on personal history, family background, and behavior patterns.
- Late Detection often, drug use is identified only after addiction has developed, making intervention

less effective.

1.1.1 CHALLENGES

- **Data Quality and Availability** – Drug consumption datasets are often incomplete, noisy, or confidential, making accurate prediction difficult. High dimensionality of demographic, psychological, and behavioral features also adds complexity.
- **Model Reliability and Interpretability** – Issues like overfitting/underfitting, difficulty in extracting meaningful features, and the "black-box" nature of machine learning models make it challenging to build trustworthy and explainable systems.
- **Computational and Deployment Challenges** – Preprocessing, training, and testing are time-consuming and resource-heavy. Deploying the system in real-world healthcare environments requires fast predictions, high accuracy, and the ability to handle many requests simultaneously.
- **Domain and Practical Challenges** – Late detection of addiction, bias in self-reported data, and the constantly evolving nature of drug trends make it hard to maintain accurate and up-to-date prediction models.

1.2 PROPOSED SYSTEM

- **Data Collection and Preprocessing** – Gather demographic and psychological details, clean and normalize the input, and convert it into a structured dataset ready for analysis.
- **Machine Learning Model** – Implement a **Random Forest Classifier** that learns patterns from historical data and improves prediction accuracy through ensemble learning.
- **Risk Prediction and Classification** – Predict the likelihood of drug usage and classify individuals into **Low, Moderate, or High Risk** categories with probability scores.
- **User-Friendly Interface** – Provide a Python Tkinter-based GUI for easy data entry, secure authentication, and clear result visualization with intuitive indicators.
- **Scalability and Integration** – Support future integration with healthcare databases, social networks, and law enforcement systems, making the solution adaptable and scalable.

1.2.1 ADVANTAGES

- **Early Risk Detection** – The system predicts the likelihood of drug abuse at an early stage, helping healthcare professionals and counselors take preventive measures before addiction worsens.
- **High Accuracy with Machine Learning** – By using the **Random Forest Classifier**, the system achieves reliable and accurate predictions, reducing errors compared to traditional manual assessments.
- **Time and Effort Saving** – Unlike surveys and manual evaluations, the automated system quickly analyzes large datasets and generates results in seconds, saving both time and resources.
- **User-Friendly Interface** – The GUI built with **Python Tkinter** allows even non-technical users (like healthcare staff) to easily input data and view risk predictions with clear visual indicators.
- **Scalability and Integration** – The system can be scaled to handle more users and integrated with external databases (healthcare records, law enforcement data, etc.), making it practical for large-scale deployment.
- **Supports Data-Driven Policy Making** – By providing reliable predictions and insights into drug consumption trends, the system helps policymakers design effective intervention strategies.
- **Enhanced Security** – Personal and sensitive information is protected with authentication and access control, ensuring data privacy and trust in the system.

II. LITERATURE REVIEW

2.1 Architecture

The system architecture consists of four main components:

- Input Layer – Collects demographic and psychological data (age, gender, personality traits, impulsivity, etc.) through a Tkinter-based GUI.
- Preprocessing Layer – Cleans, normalizes, and structures the data for model training and prediction.
- Machine Learning Layer – Uses a Random Forest Classifier trained on drug consumption datasets to classify users into Low, Moderate, or High Risk categories.
- Output Layer – Displays prediction results on the GUI with clear indicators (color codes and probability scores).

A backend database supports user authentication and storage of historical predictions, ensuring security and reliability.

2.2 ALGORITHM

The system uses a **Random Forest Classifier** for predicting drug overdose risk.

- **Data Preprocessing** – Clean and normalize demographic and psychological data for model readiness.
- **Feature Selection** – Identify key traits (age, gender, personality scores, impulsivity, etc.) influencing drug risk.
- **Model Training** – Build multiple decision trees on subsets of data; combine results using majority voting to improve accuracy.
- **Prediction** – Classify individuals into **Low, Moderate, or High Risk** categories with probability scores.
- **Result Display** – Show predictions through a Tkinter GUI with clear visual indicators, and optionally store results in a secure database.

2.3 TECHNIQUES

- **Data Preprocessing** – Cleaning, normalization, and handling of missing values to make input data consistent and model-ready.
- **Random Forest Classifier** – Core machine learning algorithm that builds multiple decision trees and aggregates their results for higher accuracy and robustness.
- **Feature Selection** – Extraction of critical attributes such as age, gender, education, personality traits, impulsivity, and sensation-seeking scores to improve prediction quality.
- **Tkinter GUI** – A simple, user-friendly Python interface for collecting input and displaying prediction results clearly with risk-level indicators.
- **Database Connectivity** – Secure integration with MySQL for storing user login details and historical predictions for future reference.
- **Validation and Testing** – Applying unit testing, integration testing, and system testing to ensure correctness, reliability, and performance of the model.
- **Security Measures** – User authentication and access control to protect sensitive data and maintain privacy.
- **Scalability Features** – The system is designed to handle more data and users in the future, with the possibility of integration into larger healthcare systems.

- **Stateless CPU Inference with Torch**: The model runs on CPU using torch. float32 to ensure compatibility on machines without GPUs. The inference process is stateless and loads the model only once per runtime session.

2.4 TOOLS

Several tools were selected to streamline development and ensure efficient prediction and user interaction:

- **Python** – The primary programming language used for implementing the machine learning model, data preprocessing, and backend logic.
- **Scikit-learn** – Provides the implementation of the **Random Forest Classifier** and other ML utilities such as data preprocessing, model training, and evaluation.
- **Tkinter** – Python's standard GUI toolkit used to design a simple, interactive, and user-friendly interface for data input and prediction display.
- **MySQL** – A relational database used for storing user credentials (login/registration) and historical prediction results securely.
- **Joblib** – Utilized for saving and loading trained ML models and preprocessing artifacts, ensuring quick reusability without retraining.
- **Pandas & NumPy** – Used for dataset handling, numerical computation, and efficient data manipulation.
- **PIL (Python Imaging Library)** – Provides support for image handling and background images within the GUI interface.
- **Operating System Libraries (OS, UUID)** – Help in managing file paths, unique file handling, and smooth integration between different modules.

2.5 METHODS

- Data Collection & Input – User enters demographic and psychological details through a Tkinter-based GUI.
- Data Preprocessing – Raw data is cleaned, normalized, and structured; missing or invalid values are handled.
- Feature Extraction & Selection – Important features such as age, gender, education, impulsivity, and sensation-seeking scores are identified for prediction.
- Model Training (Random Forest) – Multiple decision trees are built on subsets of the training data; final results are derived using majority voting to improve accuracy.
- Model Testing & Validation – Evaluate performance using test data; ensure accuracy, reliability, and minimize overfitting.
- Risk Prediction – Classify users into Low, Moderate, or High Risk categories with probability scores to aid decision-making.
- Result Display – Predictions are displayed on the GUI with intuitive color codes (green, yellow, red) for easy understanding.
- Database Storage – User credentials and historical predictions are securely stored in a MySQL database for future access.

III. METHODOLOGY

3.1 INPUT

This project is designed to predict the risk of drug overdose based on demographic and psychological attributes of individuals using Machine Learning, specifically the Random Forest Classifier. The system takes structured user data as input through a Tkinter-based GUI and processes it for accurate prediction.

The types of input used in the system include:

- Demographic Information: Age, Gender, Education, Country, and Ethnicity.
- Psychological Traits: Personality scores such as Nscore (Neuroticism), Escore (Extraversion), Oscore (Openness), Ascore (Agreeableness), and Cscore (Conscientiousness).
- Behavioral Attributes: Impulsivity score and Sensation Seeking (SS) score.
- Drug History (if any): Information about prior narcotic usage.

The application is implemented using a modular structure for better maintainability:

- Main Application (app.py / GUI logic): Handles user interaction, login/registration, and form input.
- Preprocessing Module: Cleans and validates input data, ensuring proper formats and handling missing values.
- Model Module: Loads the trained Random Forest model, processes the input features, and generates risk predictions.
- Database (MySQL): Stores user credentials and prediction history for secure access.

This modular approach allows easy deployment, reliable risk assessment, and ensures smooth operation even on systems with limited computational resources.

.

Figure:1 User Interface Screen

Fig 2: Prompting the input Screen

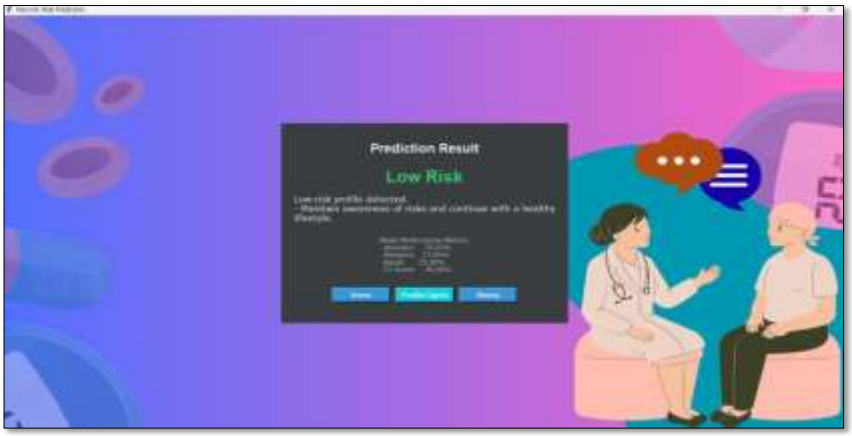

Fig 3: Output Screen

3.2 METHOD OF PROCESS

The drug overdose prediction system follows a structured methodology that integrates user input, data preprocessing, machine learning model prediction, and result visualization to provide accurate and reliable risk assessment. The process works as follows:

1. **User Input Acquisition**

- Users enter demographic and psychological details (age, gender, education, personality traits, impulsivity, sensation-seeking, etc.) through a Tkinter-based GUI.
- Secure login and registration features ensure authenticated access.

2. **Data Preprocessing**

- Input data is validated, cleaned, and normalized to remove inconsistencies.
- Missing or invalid values are handled, and categorical attributes are encoded into numerical form.

3. **Feature Selection and Transformation**

- Significant features influencing drug consumption patterns (e.g., impulsivity, sensation seeking, neuroticism) are selected.
- Features are transformed into model-ready formats for accurate processing.

4. **Model Training and Prediction**

- A **Random Forest Classifier** is trained on historical drug consumption datasets.
- When new user data is provided, the trained model predicts the likelihood of drug overdose.
- Results are categorized into **Low, Moderate, or High Risk** levels with probability scores.

5. **Result Visualization**

- Predictions are displayed on the Tkinter GUI with intuitive color codes:
  - Green → Low Risk
  - Yellow → Moderate Risk
  - Red → High Risk
- This makes interpretation simple for healthcare professionals or end users.

6. **Storage and History Management**

- A MySQL database securely stores user credentials and historical predictions.
- Users can retrieve past records for monitoring and analysis.

7. **Error Handling and Feedback**

- Input validation ensures incorrect or incomplete data is flagged with clear error messages.
- Feedback from users can be used to refine system performance and improve reliability.

3.3 OUTPUT

The output of the drug overdose prediction system is a risk classification that indicates the likelihood of an individual being prone to drug consumption or overdose. By leveraging the Random Forest Classifier, the system provides accurate predictions based on demographic, psychological, and behavioral attributes entered by the user.

The results are displayed through a Tkinter-based graphical interface in a clear and interactive manner:

- Risk Level Classification – Users are categorized into Low Risk, Moderate Risk, or High Risk groups based on the model's prediction.
- Visual Indicators – Color codes (Green → Low, Yellow → Moderate, Red → High) help users easily interpret results.
- Probability Scores – Alongside the risk category, the system may provide probability values to indicate the model's confidence in the prediction.
- Result History – Predictions can be stored in a secure MySQL database for future retrieval and tracking of individual cases.

This output empowers healthcare professionals, counselors, and researchers with actionable insights, allowing them to take preventive measures, provide early interventions, and design effective rehabilitation strategies.

IV. RESULTS

The Text-to-Image Generator successfully transforms user-provided text prompts into visually compelling images using a combination of transformer-based text encoders and latent diffusion models. Through the integration of models like CLIP (for text understanding) and a diffusion-based U-Net (for image synthesis), the system generates high-quality outputs that are coherent with the input description. Upon testing with a variety of prompts across different themes (e.g., "A futuristic robot in a desert", "A serene village in water color style"), the results demonstrated the model's ability to preserve semantic relevance and artistic quality. The diffusion process efficiently denoised latent space representations to produce visually appealing images that matched the intended style and content of the prompts. The web interface ensured that users could interactively submit prompts, view real-time results, and regenerate outputs with modified inputs. Overall, the system delivered accurate, creative, and responsive performance suitable for applications in design, storytelling, and AI-driven content generation.

V. DISCUSSIONS

This project shows how powerful AI tools can turn text into images. Users can enter a description, and the system creates a matching image using advanced models in the background. Even though the technology behind it is complex, the website makes it easy for anyone to try.The system responds quickly and produces clear images based on the user's input. The design is simple and user-friendly, focusing on providing a smooth experience. While it doesn't store images or keep a history, it works well for its main goal—turning text into visuals.

VI. CONCLUSION

The Text-to-Image Generator project successfully combines advanced AI models—transformers for understanding text and diffusion models for generating images—into a user-friendly web application. It allows users to input simple descriptions and receives visually meaningful images that reflect their prompts. The project demonstrates how cutting-edge machine learning can be made accessible to everyday users through thoughtful design and integration. While there is room for further improvement, such as adding download options or improving generation speed, the current system proves that AI can turn imagination into visuals with just a few words.

VII. FUTURE SCOPE

The Text-to-Image Generator has strong potential for improvement and expansion. Key future developments include:

- Higher-Quality Images: Upgrading to newer models like SDXL can improve image clarity and detail.
- Style-Based Generation: Allowing users to choose art styles (e.g., sketch, painting) for more creative control.
- Voice Input: Adding speech-to-text support for hands-free prompt entry.
- User Accounts and History: Saving generated images under user profiles for easy access and continued use.
- Mobile Optimization: Making the system accessible on mobile devices to reach a wider audience**.**

VIII. ACKNOWLEDGEMENT

P.Bindhu Priya working as Assistant Professor in Master of Computer Applications at Sanketika Vidya Parishad Engineering college, Visakhapatnam, Andhra Pradesh with 13 years of experience in Master of Computer Applications (MCA), accredited by NAAC. Her area of interest is in Computer Programming Using C, Computer Organisation, Software Engineering, Artificial Intelligence, Internet of Things (IoT) and Distributed Systems.

Mudili Lavanya is pursuing her final semester MCA in Sanketika Vidya Parishad Engineering College, accredited with A grade by NAAC, affiliated by Andhra University and approved by AICTE. With interest in Artificial intelligence. A. Twinkle Vanishree has taken up her PG project on TEXT-TO-IMAGE GENERATOR USING DEEP LEARNING and published the paper in connection to the project under the guidance of Mrs. P. Bindhu Priya, Assistant Professor, SVPEC.

VIII REFERENCES

1. **The leading neighborhood-level predictors of drug overdose: a mixed machine learning and spatial approach**
https://www.sciencedirect.com/science/article/abs/pii/S0376871621006384
2. **Using machine learning to predict opioid overdoses among prescription opioid users**
https://www.valueinhealthjournal.com/article/S1098-3015(18)31985-5/fulltext
3. **Enhancing timeliness of drug overdose mortality surveillance: A machine learning approach**
https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0223318
4. **Developing and validating a machine-learning algorithm to predict opioid overdose in Medicaid beneficiaries in two US states**
https://www.thelancet.com/journals/landig/article/PIIS2589-7500(22)00062-0/fulltext
5. **Identifying Predictors of Opioid Overdose Death at a Neighborhood Level With Machine Learning**
https://academic.oup.com/aje/article/191/3/526/6433428
6. **Machine learning based opioid overdose prediction using electronic health records**
https://pmc.ncbi.nlm.nih.gov/articles/PMC7153049/0
7. **Detection of overdose and underdose prescriptions—An unsupervised machine learning approach**
https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0260315
8. **Machine learning takes a village: Assessing neighbourhood-level vulnerability for an overdose and infectious disease outbreak**
https://www.sciencedirect.com/science/article/pii/S0955395921003005
9. **Machine learning takes a village: assessing neighbourhood-level vulnerability for an overdose and infectious disease outbreak**
https://www.sciencedirect.com/science/article/pii/S0955395921003005
10. **Predicting opioid overdose risk of patients with opioid prescriptions using electronic health records based on temporal deep learning**
https://www.sciencedirect.com/science/article/pii/S153204642100054X
11. **Discovering the Unclassified Suicide Cases Among Undetermined Drug Overdose Deaths Using Machine Learning Techniques**
https://onlinelibrary.wiley.com/doi/abs/10.1111/sltb.12591
12. **Enhancing timeliness of drug overdose mortality surveillance: A machine learning approach**
https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0223318
13. **Developing and validating a machine-learning algorithm to predict opioid overdose in Medicaid beneficiaries in two US states**
https://www.thelancet.com/journals/landig/article/PIIS2589-7500(22)00062-0/fulltext

14. Identifying Predictors of Opioid Overdose Death at a Neighborhood Level With Machine Learning

https://academic.oup.com/aje/article/191/3/526/6433428

15. Machine learning based opioid overdose prediction using electronic health records

https://pmc.ncbi.nlm.nih.gov/articles/PMC7153049/0

16. Detection of overdose and underdose prescriptions—An unsupervised machine learning approach

https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0260315

17. Machine learning takes a village: Assessing neighbourhood-level vulnerability for an overdose and infectious disease outbreak

https://www.sciencedirect.com/science/article/pii/S0955395921003005

18. Machine learning takes a village: assessing neighbourhood-level vulnerability for an overdose and infectious disease outbreak

https://www.sciencedirect.com/science/article/pii/S0955395921003005

19. Predicting opioid overdose risk of patients with opioid prescriptions using electronic health records based on temporal deep learning

https://www.sciencedirect.com/science/article/pii/S153204642100054X

20. Discovering the Unclassified Suicide Cases Among Undetermined Drug Overdose Deaths Using Machine Learning Techniques

https://onlinelibrary.wiley.com/doi/abs/10.1111/sltb.12591