Graduation Project

Healthcare Predictive Analytics Project

Made by

Abdelrahman Ayman Farid

Loai Waleed Hataba

Tasneem Ahmed Ali

Mahmoud Ibrahim Tawfik

Muhamed Elsayed Anas

Mostafa Tarek Mahmoud

Supervised by

ENG:Muhamed Ehab

Table of Contents

1. Introduction

This documentation describes a comprehensive Machine Learning API system that provides predictive capabilities for various health conditions and general machine learning tasks. The API is built using FastAPI and provides endpoints for:

- Cardiovascular disease risk prediction
- Diabetes risk prediction
- Asthma diagnosis prediction
- Schizophrenia risk evaluation
- Image classification
- Text sentiment analysis

The system is designed to be modular, allowing easy addition of new models while maintaining consistent interfaces.

2. System Overview

Architecture

The system follows a client-server architecture with:

- **Frontend**: API endpoints exposed via FastAPI
- **Backend**: Model management and prediction logic
- **Data**: Pre-trained models stored in the `saved_models` directory

Key Components

1. **Model Manager**: Central component that loads and manages all available models
2. **Base Model Class**: Abstract class defining the interface for all models

3. **Specific Model Implementations**: Concrete implementations for each prediction task

4. **API Endpoints**: RESTful endpoints for interacting with the models

Technology Stack

- Python 3.8+

- FastAPI (web framework)

- Uvicorn (ASGI server)

- Scikit-learn (machine learning)

- TensorFlow/Keras (deep learning)

- Joblib (model serialization)

- Pydantic (data validation)

3. API Endpoints

Endpoint	Method	Description	Input	Output
`/`	GET	Welcome message	None	Basic info
`/api/health`	GET	Health check	None	System status
`/api/models`	GET	List all models	None	Model info list
`/api/models/{model_id}`	GET	Get specific model info	Model ID	Model details
`/api/models/{model_id}/predict`	POST	Predict with image model	Image file	Prediction result
`/api/models/{model_id}/predict/text`	POST	Predict with text model	JSON text	Prediction result
`/api/models/{model_id}/predict/tabular`	POST	Predict with tabular model	JSON features	Prediction result
`/api/cardio/predict`	POST	Cardiovascular prediction	JSON features	Prediction result
`/api/diabetes/predict`	POST	Diabetes prediction	JSON features	Prediction result
`/api/asthma/predict`	POST	Asthma prediction	JSON features	Prediction result
`/api/schizo/predict`	POST	Schizophrenia risk prediction	JSON features	

4. Models Documentation

4.1 Cardiovascular Disease Predictor

Model ID: `cardio-predictor`

Description

Predicts cardiovascular disease risk based on health metrics and lifestyle factors.

Features

- `age`: Age in days (converted to years internally)

- `gender`: 1 = female, 2 = male

- `height`: Height in cm

- `weight`: Weight in kg

- `ap_hi`: Systolic blood pressure

- `ap_lo`: Diastolic blood pressure

- `cholesterol`: Cholesterol level (1: normal, 2: above normal, 3: well above normal)

- `gluc`: Glucose level (1: normal, 2: above normal, 3: well above normal)

- `smoke`: Smoking status (0: no, 1: yes)

- `alco`: Alcohol consumption (0: no, 1: yes)

- `active`: Physical activity (0: no, 1: yes)

Output

- `prediction`: "high risk" or "low risk"

- **confidence**: Probability score (0-1)

- **processing_time**: Time taken for prediction

4.2 Diabetes Predictor

Model ID: `diabetes-predictor`

Description

Predicts diabetes risk based on health indicators.

Features

- **Pregnancies**: Number of pregnancies

- **Glucose**: Plasma glucose concentration

- **BloodPressure**: Diastolic blood pressure (mm Hg)

- **SkinThickness**: Triceps skin fold thickness (mm)

- **Insulin**: 2-Hour serum insulin (mu U/ml)

- **BMI**: Body mass index

- **DiabetesPedigreeFunction**: Diabetes pedigree function

- **Age**: Age in years

Output

- **prediction**: "diabetic" or "non-diabetic"

- **confidence**: Probability score (0-1)

- **processing_time**: Time taken for prediction

4.3 Asthma Predictor

Model ID: `asthma-predictor`

Description

Predicts asthma diagnosis based on health metrics and environmental factors.

Features

- `PatientID`: Unique patient identifier

- `Age`: Patient age

- `Gender`: Patient gender

- `Ethnicity`: Ethnic background

- `EducationLevel`: Education level

- `BMI`: Body mass index

- `Smoking`: Smoking status

- `PhysicalActivity`: Physical activity level

- `DietQuality`: Quality of diet

- `SleepQuality`: Sleep quality

- `PollutionExposure`: Pollution exposure level

- `PollenExposure`: Pollen exposure level

- `DustExposure`: Dust exposure level

- `PetAllergy`: Pet allergy status

- `FamilyHistoryAsthma`: Family history of asthma

- `HistoryOfAllergies`: History of allergies

- **Eczema**: Eczema status

- **HayFever**: Hay fever status

- **GastroesophagealReflux**: GERD status

- **LungFunctionFEV1**: FEV1 lung function measure

- **LungFunctionFVC**: FVC lung function measure

- **Wheezing**: Wheezing symptoms

- **ShortnessOfBreath**: Shortness of breath

- **ChestTightness**: Chest tightness

- **Coughing**: Coughing symptoms

- **NighttimeSymptoms**: Nighttime symptoms

- **ExerciseInduced**: Exercise-induced symptoms

Output

- **prediction**: "asthma" or "no asthma"

- **confidence**: Probability score (0-1)

- **processing_time**: Time taken for prediction

4.4 Schizophrenia Risk Predictor

Model ID: `schizo-predictor`

Description

Evaluates schizophrenia proneness based on behavioral indicators using a trained classifier with five risk levels.

Features

Parameter	Type	Description	Example Values
Age	integer	Patient age in years	25
Gender	string	Biological sex	"Male", "Female"
Marital_Status	string	Current relationship status	"Single", "Married"
Fatigue	integer	Fatigue level (1-5 scale)	3
Slowing	integer	Psychomotor slowing (1-5 scale)	2
Pain	integer	Pain sensitivity (1-5 scale)	4
Hygiene	integer	Self-care habits (1-5 scale)	3
Movement	integer	Motor abnormalities (1-5 scale)	1

Output

- `prediction`: Risk level category

- `confidence`: Highest probability score

- `class_probabilities`: Dictionary of all risk levels with probabilities

- `processing_time`: Time taken for prediction (seconds)

4.5 Image Classifier

Model ID: `image-classifier`

Description

Classifies images into categories (currently a dummy implementation).

Output

- `prediction`: Predicted class (e.g., "cat")

- `confidence`: Confidence score (0-1)

- `processing_time`: Time taken for prediction

4.6 Text Sentiment Analyzer

Model ID: `text-sentiment`

Description

Analyzes text sentiment (currently a dummy implementation).

Output

- `prediction`: "positive" or "negative"

- `confidence`: Confidence score (0-1)

- `processing_time`: Time taken for prediction

5. Installation Guide

Prerequisites

- Python 3.8+
- pip package manager

Steps

1. Clone the repository
2. Create a virtual environment:

```
python -m venv venv
source venv/bin/activate   # On Windows: venv\Scripts\activate
```

3. Install dependencies:

```
pip install -r requirements.txt
```

4. Ensure model files are in the `saved_models` directory

Required Files

- `saved_models/cardio_model.pkl`
- `saved_models/cardio_scaler.pkl`
- `saved_models/diabetes_pipeline.pkl`
- `saved_models/asthma_diagnosis_model.h5`
- `saved_models/schizomodel.pkl`

6. Usage Examples

Cardiovascular Prediction

```
curl -X POST "http://localhost:8000/api/cardio/predict" \
-H "Content-Type: application/json" \
-d '{
    "age": 18000,
    "gender": 1,
    "height": 165,
    "weight": 70,
    "ap_hi": 120,
    "ap_lo": 80,
    "cholesterol": 1,
    "gluc": 1,
    "smoke": 0,
    "alco": 0,
    "active": 1
}'
```

Diabetes Prediction

```
curl -X POST "http://localhost:8000/api/diabetes/predict" \
-H "Content-Type: application/json" \
-d '{
    "Pregnancies": 2,
    "Glucose": 120,
    "BloodPressure": 70,
    "SkinThickness": 30,
    "Insulin": 80,
    "BMI": 25.5,
    "DiabetesPedigreeFunction": 0.4,
    "Age": 35
}'
```

Image Classification

```
curl -X POST "http://localhost:8000/api/models/image-classifier/predict" \
-H "Content-Type: multipart/form-data" \
-F "file=@cat.jpg"
```

7. Error Handling

The API returns appropriate HTTP status codes:

- `200 OK`: Successful request

- `400 Bad Request`: Invalid input parameters

- `404 Not Found`: Model not found

- `500 Internal Server Error`: Prediction failed

Error responses include a JSON object with error details:

```
{
    "detail": "Error message"
}
```

8. Security Considerations

1. **CORS**: Configured to allow all origins (adjust for production)
2. **Input Validation**: All inputs are validated using Pydantic models
3. **File Uploads**:
   a. Restricted to specific file types for image uploads
   b. Files are saved in a temporary uploads directory
4. **Environment Variables**: Consider using environment variables for sensitive configuration

9. Performance Metrics

Typical response times:

- Tabular models: <500ms

- Image classification: ~500ms (dummy implementation)

- Text analysis: ~200ms (dummy implementation)

System requirements:

- Minimum 2GB RAM

- Recommended 4GB RAM for better performance

10. Future Enhancements

1. **Model Versioning**: Support for multiple model versions

2. **Batch Processing**: Endpoints for batch predictions

3. **Authentication**: JWT-based authentication

4. **Rate Limiting**: Prevent abuse of the API

5. **Model Monitoring**: Track prediction accuracy over time

6. **Swagger/OpenAPI**: Enhanced API documentation

7. **Docker Support**: Containerized deployment

8. **More Models**: Additional health prediction models

Conclusion

This ML Models API provides a robust platform for health risk prediction and general machine learning tasks. The modular design allows for easy extension with additional models while maintaining consistent interfaces. The comprehensive documentation enables developers to quickly integrate the API into their applications.