Disease Prediction from Symptoms

The original dataset was taken from the following link :

<https://people.dbmi.columbia.edu/~friedma/Projects/DiseaseSymptomKB/index.html>

This dataset is a knowledge database of disease-symptom associations generated by an automated method based on information in textual discharge summaries of patients at New York-Presbyterian Hospital admitted during 2004. The first column shows the disease, the second the number of discharge summaries containing a positive and current mention of the disease, and the associated symptom. 

Fig. Dataset

This code implements a simple disease prediction system using a Naive Bayes classifier. The model is trained on a dataset containing symptoms and their corresponding diseases. The symptoms are selected from a predefined list (`l1`), and the diseases are categorized into classes represented by integers. The code uses the Tkinter library for creating a graphical user interface (GUI) to input symptoms and predict the possible disease.

Here's a detailed breakdown of the code:

1. Symptoms and Diseases Lists:

- `l1`: A list of symptoms.

- `disease`: A list of diseases corresponding to the classes in the dataset.

2. Data Loading:

- Training data is loaded from "Training.csv," and testing data is loaded from "Testing.csv."

- The dataset is preprocessed by mapping disease names to integer labels using the `replace` function.

3. Model Training:

- A Multinomial Naive Bayes classifier (`gnb`) is used for training the model.

- The `fit` function is used to train the model on the training data (`X` and `y`).

4. GUI (Tkinter):

- A Tkinter-based GUI is created to input symptoms and obtain disease predictions.

- Entry fields and dropdown menus are provided for users to select symptoms.

- When the "Predict" button is clicked, the `message` function is called to process the input and make predictions.

5. Prediction and Output:

- The `NaiveBayes` function uses the trained model to predict the disease based on the selected symptoms.

- The predicted disease is displayed in a Text widget in the GUI.

6. Accuracy Evaluation:

- The accuracy of the model is evaluated using the testing data, and the results are printed.

7. User Interaction:

- Users select symptoms from dropdown menus and click the "Predict" button to get the predicted disease.

- The GUI provides a simple interface for users to interact with the disease prediction model.

8. Note:

- The code is specific to the dataset structure and assumes a specific format for the training and testing data CSV files.

- The Naive Bayes model is chosen for simplicity, and its performance may vary depending on the dataset characteristics.