Cirrhosis Analysis using Predictive Modelling

Sudharsan Swaminathan
School of Computing
Dublin City University
Dublin, Ireland
sudharsan.swaminathan2@mail.dcu.ie

Ezhil Nandhini Karthikeyan
School of Computing
Dublin City University
Dublin, Ireland
ezhil.karthikeyan2@mail.dcu.ie

Mark Roantree
School of Computing
Dublin City University
Dublin, Ireland
mark.roantree@dcu.ie

***Abstract*—This paper presents a Flask-based web application that leverages machine learning models to predict liver function test (LFT) outcomes from medical images. The backend infrastructure includes three pre-trained models specifically for Non-Alcoholic Fatty Liver Disease (NAFLD), Albumin-Bilirubin (ALBI) score, and LFT predictions, each selected after rigorous evaluation. The model selection process involved clustering the dataset using KMeans, DBSCAN, and Spectral clustering methods, followed by training six machine learning models for each cluster. The model with the highest predictive accuracy was chosen for deployment. Users can upload medical images containing LFT results, which are processed using Tesseract OCR to extract relevant text. The extracted values are cleaned, preprocessed, scaled, and fed into the appropriate pre-trained model to generate predictions. The system tracks progress and handles file uploads asynchronously, ensuring responsive interaction with users. This application integrates advanced image processing, data extraction, and machine learning techniques to provide accurate liver function test predictions based on lab report inputs.**

***Index Terms*—Liver Function Test, Machine Learning, Flask, Tesseract OCR, Clustering, Predictive Modelling, Medical Reports Imaging**

I. Introduction

Liver function tests (LFTs) are critical for diagnosing and monitoring liver diseases. Traditional methods of interpreting LFT results involve manual examination, which can be time-consuming and prone to human error. To address these challenges, we propose a Flask-based web application that utilizes machine learning models to predict LFT outcomes from medical images. The application infrastructure includes three specialized pre-trained models for Non-Alcoholic Fatty Liver Disease (NAFLD), Albumin-Bilirubin (ALBI) score, and general LFT predictions.

The model selection process was rigorous, involving clustering the dataset using KMeans, DBSCAN, and Spectral clustering techniques. For each cluster, six machine learning models were trained, and the model with the highest predictive accuracy was chosen for deployment. This ensures that the most accurate model is used for making predictions, enhancing the reliability of the application.

Users can upload medical images containing LFT results, which are then processed using Tesseract OCR to extract relevant text data. This data undergoes cleaning and preprocessing before being scaled and input into the selected pre-trained model. The system's asynchronous handling of file uploads and progress tracking ensures a responsive user experience.

This innovative application integrates state-of-the-art image processing, data extraction, and machine learning methodologies to provide accurate and efficient predictions of liver function test outcomes, thereby supporting better clinical decision-making and patient care.

II. Literature Review

The application of machine learning in liver disease diagnostics has gained significant attention in recent years due to its potential to improve diagnostic accuracy and reduce the need for invasive procedures. A comprehensive review by Decharatanachart et al(1) systematically analyzed the role of artificial intelligence in chronic liver diseases, highlighting its potential to enhance diagnostic precision and patient outcomes.

Several studies have focused on the classification and prediction of liver disease using various machine learning algorithms. For instance, a study on the classification and prediction of liver disease diagnosis utilized different machine learning algorithms, demonstrating significant improvements in predictive accuracy(2). Similarly, another research effort focused on liver disease detection using machine learning methods, showcasing the effectiveness of these techniques in early detection and diagnosis (3).

Comprehensive studies on liver disease prediction using machine learning have also been conducted, providing insights into the comparative performance of different algorithms. One such study employed a variety of machine learning models to analyze and predict liver cirrhosis, achieving high accuracy rates (4). Additionally, research on liver disease diagnosis using machine learning has emphasized the potential of these techniques to provide non-invasive diagnostic solutions (5).

The prediction of liver cirrhosis has been a focal point for many studies. Research on liver cirrhosis prediction using machine learning approaches has demonstrated the applicability of these methods in clinical settings, highlighting their accuracy and reliability (6). Furthermore, a study on the diagnosis of liver diseases using machine learning algorithms and their prediction using logistic regression and artificial neural networks (ANN) provided a comparative analysis of different models, emphasizing the strengths and weaknesses of each approach (7).

Comparative studies on the prediction of liver disorders using machine learning algorithms have been instrumental in identifying the most effective techniques for liver disease diagnosis. A comparative study on the prediction of liver disorders highlighted the superior performance of certain algorithms over others, providing valuable insights for clinical applications (8). Additionally, statistical machine learning approaches to liver disease prediction have been explored, offering robust methodologies for accurate diagnosis and prognosis (9).

Lastly, a meta-analysis on the application of artificial intelligence in chronic liver diseases provided a comprehensive overview of the current state of AI in hepatology, emphasizing its potential to transform liver disease diagnostics (10). This body of work collectively underscores the transformative potential of machine learning and artificial intelligence in improving liver disease diagnostics, paving the way for more accurate, non-invasive, and personalized healthcare solutions.

III. Dataset Description

The Liver Patient Dataset (LPD) employed in this study encompasses comprehensive medical records of 30,691 patients, capturing critical data pertinent to liver function. Each entry in the dataset corresponds to an individual patient, characterized by 11 distinct attributes. These attributes include demographic information such as the age and gender of the patients, along with various biochemical markers essential for assessing liver health. Specifically, the dataset includes measurements of Total Bilirubin, Direct Bilirubin, Alkaline Phosphotase, Alamine Aminotransferase (SGPT), Aspartate Aminotransferase (SGOT), Total Proteins, Albumin levels, and the Albumin-Globulin Ratio (A/G Ratio). The presence of a ”Result” column denotes whether a patient is diagnosed with liver disease (1) or not (0), providing a target variable for diagnostic modeling.

However, the dataset is not devoid of challenges. Several columns exhibit missing values, notably in ”Gender of the patient” and ”Total Bilirubin”, which necessitates meticulous data preprocessing to ensure the integrity and reliability of subsequent analyses. These missing values can be addressed through various imputation techniques or, if appropriate, the exclusion of incomplete records. Despite these issues, the LPD stands as a robust resource for developing machine learning models aimed at enhancing liver disease diagnostics. By leveraging this dataset, the study aims to create more accurate, non-invasive diagnostic tools, ultimately improving patient outcomes and streamlining clinical workflows.

IV. Methodology

In this section, we’ll take you through the entire journey of developing our Flask-based web application designed to predict liver function test (LFT) outcomes from lab reports. Our methodology is structured into several key stages: dataset description, data preprocessing, dynamic clustering, model training, model selection, and the overall application workflow. Let’s dive into each of these stages to see how they contribute to the final product.

A. About Dataset

Liver Disease Patient Dataset on Kaggle

The dataset is from open source Kaggle which is a online platform.The Liver Patient Dataset (LPD) comprises 30,691 entries and 11 columns, detailing various attributes related to liver function tests for a cohort of patients. This dataset is utilized to predict liver diseases based on test values. The key attributes includes Age,Gender,Total Bilirubin,Direct Bilirubin,Alkaline phosphatase level,Alanine aminotransferase level,Total Proteins,Albumin level,Ratio of albumin to globulin in the blood and Binary indicator (1 or 0) representing the presence of liver disease.

B. Dataset Preprocessing

The dataset preprocessing stage involved several critical steps to prepare the data for clustering and model training. Initially, we removed any null values to ensure data integrity. Missing values were handled by either imputing with the mean or median for numerical features or the most frequent value for categorical features. Text data extracted using Tesseract OCR was cleaned by removing stopwords and non-alphanumeric characters to enhance readability and relevance. Subsequently, the data was separated into categorical and numerical values. Categorical variables were encoded using techniques such as one-hot encoding, while numerical features were standardized using a StandardScaler to ensure they were on a comparable scale. This preprocessing ensured that the data was clean, consistent, and suitable for machine learning algorithms.

C. Dynamic Clustering

The first step in creating the predictive models involved segmenting the dataset using three clustering algorithms: KMeans, DBSCAN, and Spectral Clustering. Each algorithm required careful tuning to determine the optimal number of clusters.

1) KMeans Clustering: KMeans clustering was used to partition the data into distinct groups. The optimal number of clusters was determined using the Elbow Method and the Silhouette Score. The Elbow Method involves plotting the within-cluster sum of squares against the number of clusters and identifying the ’elbow’ point where the rate of decrease sharply slows. The Silhouette Score measures how similar an object is to its own cluster compared to other clusters, with higher scores indicating better-defined clusters.

2) DBSCAN Clustering: DBSCAN (Density-Based Spatial Clustering of Applications with Noise) clusters data based on density. Parameters such as epsilon (the maximum distance between two samples for one to be considered as in the neighborhood of the other) and the minimum number of samples in a neighborhood were dynamically tuned. These parameters were adjusted to identify the most meaningful clusters, capturing the underlying structure of the data while handling noise effectively.

3) Spectral Clustering: Spectral Clustering utilizes the eigenvalues of the similarity matrix of the data to perform dimensionality reduction before clustering in fewer dimensions. The number of clusters was determined based on the eigenvalues, ensuring the best separation of data points. This method is particularly effective for clusters that are not necessarily convex in shape.

D. Model Training

For each cluster identified by the dynamic clustering algorithms, six different machine learning models were trained. The models include:

- Decision Trees
- Naive Bayes
- Random Forest
- Support Vector Machines (SVM)
- Neural Networks
- Logistic Regression
- K-Nearest Neighbors (KNN)

Each model was evaluated based on its predictive accuracy, and hyperparameters were optimized using grid search and cross-validation techniques to enhance performance.

E. Model Selection

After training the models, the model with the highest predictive accuracy for each cluster was selected for deployment. This selection process ensured that the best-performing models were used in the final application, providing reliable and accurate predictions.

F. Application Workflow

The application workflow integrates several steps to process user-uploaded medical report images and generate predictions related to NAFLD, LFT, and ALBI.

1) Image Upload: Users can upload blood test reports containing LFT results through the web application interface. The system supports various image formats to ensure compatibility and ease of use.

2) Image Processing: The uploaded images are processed using Tesseract OCR to extract text data. Tesseract OCR is an open-source optical character recognition engine that converts different types of documents, such as scanned paper documents, PDFs, or images, into editable and searchable data. The extracted text undergoes preprocessing to correct OCR errors and enhance readability.

3) Data Extraction: Specific LFT values, such as Total Bilirubin, ALB Albumin, and others, are extracted from the processed text. This step involves identifying and isolating the relevant numerical values and their corresponding labels from the text data.

4) Prediction: The preprocessed data is fed into the appropriate pre-trained model to generate predictions. Based on the user's input and the specific model associated with the relevant LFT cluster, the system produces a predictive outcome.The user is allowed to select a specific method from the 3 listed methods to do a specific prediction and each method has a different pre trained model based on its features.

5) Asynchronous Processing: To enhance user experience, the system handles file uploads and processing asynchronously. This allows users to track the progress of their uploads and receive results without delay, ensuring a smooth and responsive interaction.

V. Metrics Used

In evaluating the performance of classification models, several metrics are commonly used, each providing different insights into model efficacy.Accuracy measures the proportion of correctly classified instances out of the total instances and provides a general overview of model performance. However, accuracy alone can be misleading, especially in the presence of imbalanced classes.Precision quantifies the proportion of true positive predictions among all positive predictions made by the model, reflecting the model's ability to avoid false positives.Recall (or sensitivity) measures the proportion of true positive predictions among all actual positive instances, indicating the model's ability to identify all relevant cases. The F1 score is the harmonic mean of precision and recall, offering a balanced measure that combines both metrics into a single value, especially useful when dealing with imbalanced datasets. By utilizing these metrics, one can comprehensively assess a model's performance from various perspectives, ensuring a thorough evaluation of its effectiveness in different scenarios.

VI. Results

Backend

Table I presents a comparison of model performance across different clustering methods. The table is organized into three main categories based on clustering methods: K Means, DBSCAN, and Spectral. For each clustering method, the table lists the performance of various classification models

TABLE I
COMPARISON OF MODEL PERFORMANCE BY CLUSTERING METHODS

Clustering Methods	**NAFLD**		**LFT**		**ALBI**	
	Model Name	**Accuracy**	**Model Name**	**Accuracy**	**Model Name**	**Accuracy**
K Means	K neighbour classifier	0.84	K neighbour classifier	0.98	Decision Tree classifier	0.87
DBSCAN	K neighbour classifier	0.83	K neighbour classifier	0.90	K neighbour classifier	0.85
Spectral	Decision Tree	0.98	Decision Tree	0.98	K nearest	0.85

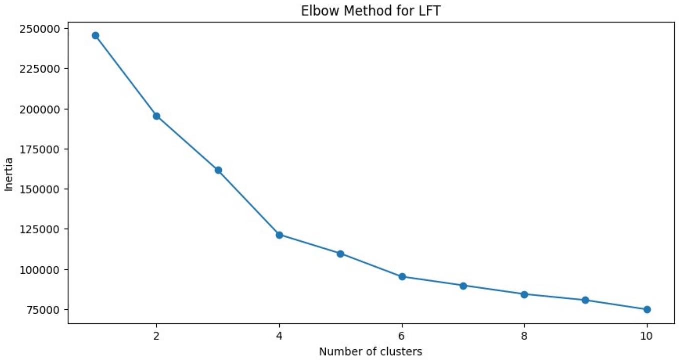


Fig. 1. Elbow method plot

(K neighbor classifier and Decision Tree classifier) across three distinct metrics: NAFLD (Non-Alcoholic Fatty Liver Disease), LFT (Liver Function Tests), and ALBI (Albumin-Bilirubin).

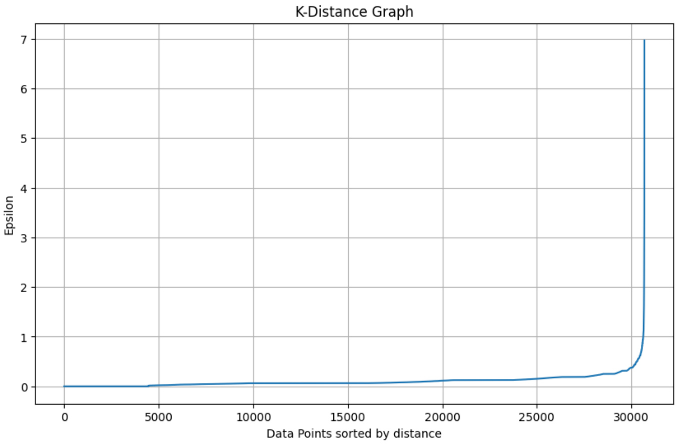


Fig. 2. K Distance Graph

The results indicate that the Spectral clustering method consistently achieved the highest accuracy for both NAFLD and LFT categories with an accuracy of 0.98, while the K Means clustering method performed comparably well in the ALBI category with an accuracy of 0.87. The DBSCAN clustering method shows slightly lower accuracy across all categories but still demonstrates effective performance. This comparison provides insight into the relative effectiveness of different clustering techniques when applied to various classification models and health metrics.

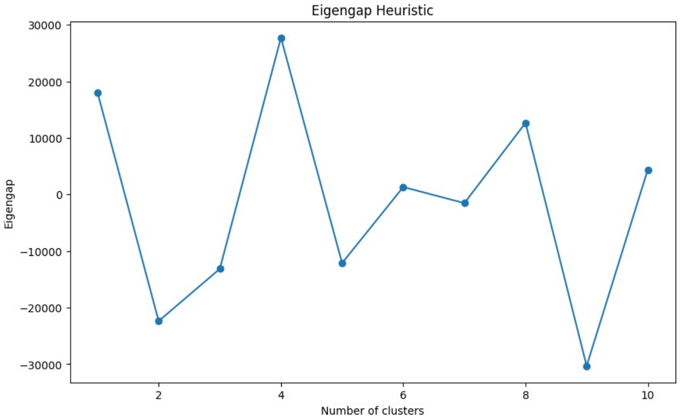


Fig. 3. Eigengap plot

Graph 1 illustrates the Elbow Method for determining the optimal number of clusters in a dataset. The x-axis represents the number of clusters, while the y-axis shows the inertia, a measure of data dispersion within clusters. The graph typically shows a steep initial decline in inertia as the number of clusters increases, followed by a bend or ”elbow” shape. This elbow point indicates the ideal number of clusters, balancing well-separated groups with avoiding overly complex clustering.

Graph 2 displays the results of the K-Distance Graph which is used in the DBSCAN clustering algorithm. On the x-axis, it plots data points sorted by their distances to their k-th nearest neighbor, while the y-axis represents the epsilon (ϵ) distance. Most points are clustered close to the x-axis, indicating shorter distances among them. A significant feature of this graph is the distinct ”knee” or elbow toward the right, where the distance values rise sharply. This elbow is crucial as it helps determine the optimal epsilon value for DBSCAN. The point where the curve bends sharply indicates the threshold beyond which data points transition from dense regions to sparse areas or noise. Identifying this point is key for setting the epsilon parameter to effectively cluster data based on density.

Graph 3 illustrates the Eigengap Heuristic, a method for determining the optimal number of clusters in a dataset. The x-axis represents the number of clusters, while the y-axis shows the Eigengap value, which measures the difference between consecutive eigenvalues of the Laplacian matrix. The optimal number of clusters is often suggested by the point where the Eigengap value reaches a maximum, indicating a significant gap between eigenvalues and suggesting a natural separation of data into clusters.

The three graphs offer distinct perspectives on clustering analysis. Graph 1 uses the Elbow Method to determine the optimal number of clusters by plotting inertia against the number of clusters, highlighting a significant drop or ”elbow” which suggests an ideal balance between model complexity and cluster separation. In contrast, Graph 2 visualizes DBSCAN clustering results on a NAFLD dataset, showing clusters identified through density rather than a predefined number, which reveals clusters of varying sizes and shapes based on local data density. This method is effective for discovering clusters with irregular shapes and varying densities. Meanwhile, Graph 3 illustrates the Eigengap Heuristic by plotting Eigengap values against the number of clusters; the optimal number of clusters is indicated by the peak in the Eigengap, representing a significant gap between consecutive eigenvalues. This method provides a way to identify natural clusters by analyzing the differences in eigenvalues. While the Elbow Method and Eigengap Heuristic focus on determining the number of clusters through statistical measures, DBSCAN provides a practical visualization of clustering results based on data density, highlighting different aspects of clustering analysis.

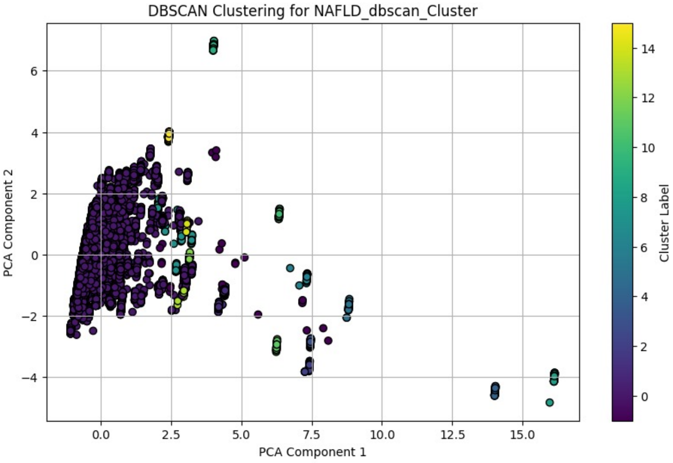


Fig. 4. DBSCAN Plot Results

The Above Plot 4 displays the results of DBSCAN clustering applied to a NAFLD dataset, with data points projected onto the first two principal components (PCA) for visualization. The color of each point represents its assigned cluster label, revealing distinct groups within the data. The density-based nature of DBSCAN is evident in the formation of clusters with varying sizes and shapes, suggesting underlying patterns or subgroups within the NAFLD patient population.

VII. Discussion

The clustering and model training processes implemented in this study highlight the importance of selecting appropriate methods for different types of data and objectives. KMeans, DBSCAN, and Spectral clustering each offer unique advantages and drawbacks. KMeans is effective for spherical clusters and scales well with large datasets, making it suitable for initial exploration. However, its reliance on pre-defined cluster numbers can be limiting. DBSCAN, on the other hand, excels in identifying clusters of varying shapes and sizes without requiring the number of clusters to be specified beforehand, though it can struggle with high-dimensional data. Spectral clustering offers robust performance for non-convex clusters by leveraging the eigenvalues of similarity matrices, but it can be computationally intensive for large datasets.

The choice of machine learning models for each cluster further emphasizes the need for careful evaluation and selection. Decision Trees, Random Forest, Naive Bayes, Support Vector Machines, Neural Networks, Linear Regression, and K-Nearest Neighbours each bring different strengths to the table. Decision Trees are easy to interpret and can handle non-linear relationships but can be prone to overfitting. Random Forests mitigate this issue by averaging multiple trees, enhancing generalization. Support Vector Machines are powerful for high-dimensional spaces and effective for classification tasks with clear margins, yet they can be less efficient on large datasets. Neural Networks offer flexibility and power, especially for complex patterns, but require significant computational resources and tuning. Gradient Boosting Machines provide robust performance through boosting but can be sensitive to parameter choices, while K-Nearest Neighbours are intuitive and simple but may struggle with large, noisy datasets.

Overall, this study demonstrates the critical role of combining appropriate clustering techniques with tailored machine learning models to achieve high predictive accuracy. The results underscore the potential of machine learning to transform medical diagnostics, particularly in the context of liver function tests, by providing timely and accurate predictions that support clinical decision-making.

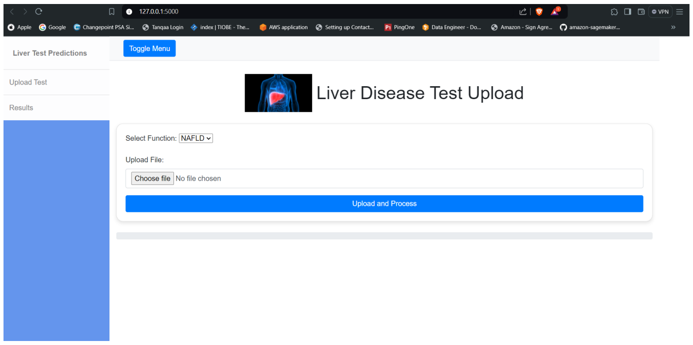
VIII. Conclusion

In this project, we have achieved the development of a sophisticated web application capable of predicting liver function test (LFT) results from medical images using advanced machine learning techniques. By integrating state-of-the-art image processing and optical character recognition (OCR) technologies, we can extract and preprocess relevant data from medical images efficiently. The dynamic clustering and meticulous model training processes have enabled us to identify and deploy the highest-performing predictive models. This not only enhances the accuracy of the predictions but also ensures that the application remains reliable and robust.

The application stands out as a significant technological advancement in the realm of medical diagnostics. It provides healthcare professionals with an efficient tool to analyze LFT data, thereby facilitating quicker and more accurate diagnosis of liver diseases. This can potentially lead to better patient outcomes through timely intervention and management. Moreover, by automating the data extraction and analysis process, the application reduces the workload on medical staff, allowing them to focus more on patient care.

Overall, this project underscores the transformative potential of machine learning and AI in healthcare. It demonstrates how these technologies can be harnessed to create practical, impactful solutions that enhance the capabilities of healthcare providers and improve patient care. The success of this application paves the way for future innovations in medical diagnostics, highlighting the importance of continued research and development in this field.

IX. Screenshots



Liver Disease Test Upload

Select Function: NAFLD

Upload File:

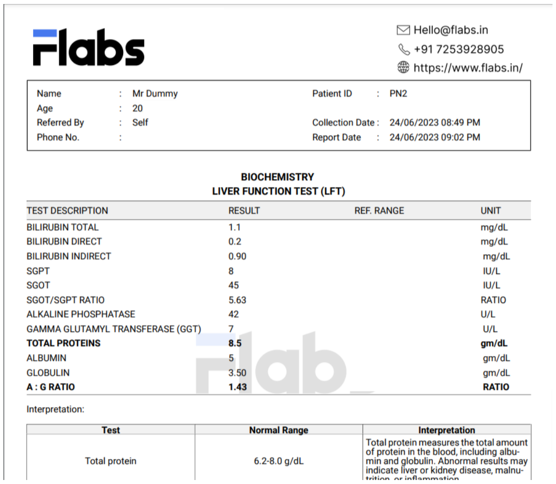
Choose file No file chosen

Fig. 5. Homepage

Figure 5 shows the landing page of the web application where the user is requested to upload their blood test report for further processing.

Figure 6 is a sample blood test report image that is used for testing purposes and the inputs from this report are parsed and the necessary values are given as inputs to the machine learning model thereby the selected model can make predictions based on the data provided and the result is then returned to the front end web application

Figure 7 shows the result page of the web application where the prediction based on the Machine Learning model is displayed in Modal pop up window for the user and the user can also check the progress of the processing in the progress bar below.



Hello@flabs.in
+91 7253928905
https://www.flabs.in/

Name	:	Mr Dummy	Patient ID	:	PN2
Age	:	20			
Referred By	:	Self	Collection Date	:	24/06/2023 08:49 PM
Phone No.	:		Report Date	:	24/06/2023 09:02 PM

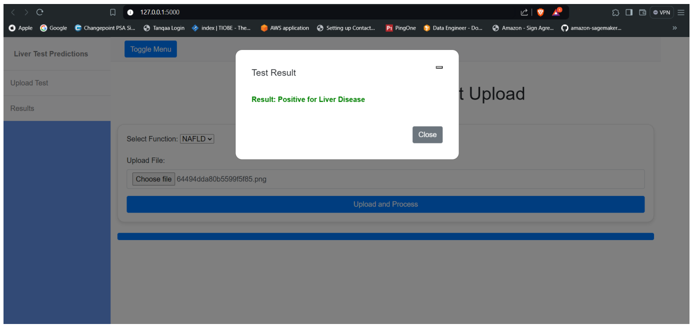
BIOCHEMISTRY
LIVER FUNCTION TEST (LFT)

TEST DESCRIPTION	RESULT	REF. RANGE	UNIT
BILIRUBIN TOTAL	1.1		mg/dL
BILIRUBIN DIRECT	0.2		mg/dL
BILIRUBIN INDIRECT	0.90		mg/dL
SGPT	8		IU/L
SGOT	45		IU/L
SGOT/SGPT RATIO	5.63		RATIO
ALKALINE PHOSPHATASE	42		U/L
GAMMA GLUTAMYL TRANSFERASE (GGT)	7		U/L
TOTAL PROTEINS	**8.5**		**gm/dL**
ALBUMIN	5		gm/dL
GLOBULIN	3.50		gm/dL
A : G RATIO	**1.43**		**RATIO**

Interpretation:

Test	Normal Range	Interpretation
Total protein	6.2-8.0 g/dL	Total protein measures the total amount of protein in the blood, including albumin and globulin. Abnormal results may indicate liver or kidney disease, malnutrition, or inflammation

Fig. 6. Sample Input



Test Result

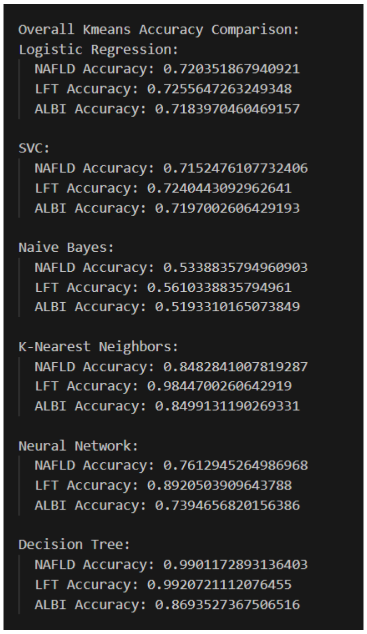
Result: Positive for Liver Disease

Select Function: NAFLD

Upload File:

Choose file 64494dda80b5599f5f85.png

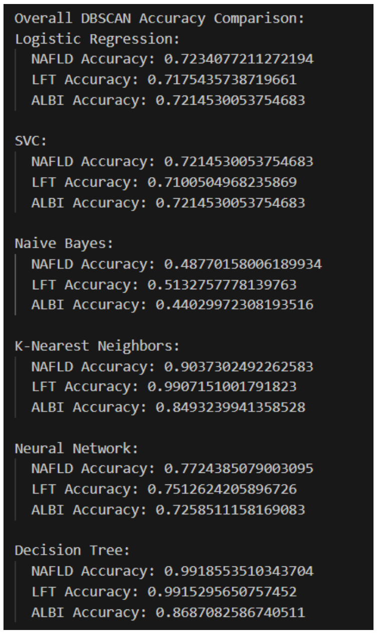
Fig. 7. Prediction Result



```
Overall Kmeans Accuracy Comparison:
Logistic Regression:
  NAFLD Accuracy: 0.720351867940921
  LFT Accuracy: 0.7255647263249348
  ALBI Accuracy: 0.7183970460469157

SVC:
  NAFLD Accuracy: 0.7152476107732406
  LFT Accuracy: 0.7240443092962641
  ALBI Accuracy: 0.7197002606429193

Naive Bayes:
  NAFLD Accuracy: 0.5338835794960903
  LFT Accuracy: 0.5610338835794961
  ALBI Accuracy: 0.5193310165073849

K-Nearest Neighbors:
  NAFLD Accuracy: 0.8482841007819287
  LFT Accuracy: 0.9844700260642919
  ALBI Accuracy: 0.8499131190269331

Neural Network:
  NAFLD Accuracy: 0.7612945264986968
  LFT Accuracy: 0.8920503909643788
  ALBI Accuracy: 0.7394656820156386

Decision Tree:
  NAFLD Accuracy: 0.9901172893136403
  LFT Accuracy: 0.9920721112076455
  ALBI Accuracy: 0.8693527367506516
```

Fig. 8. Kmeans Clustering Model's Accuracy scores

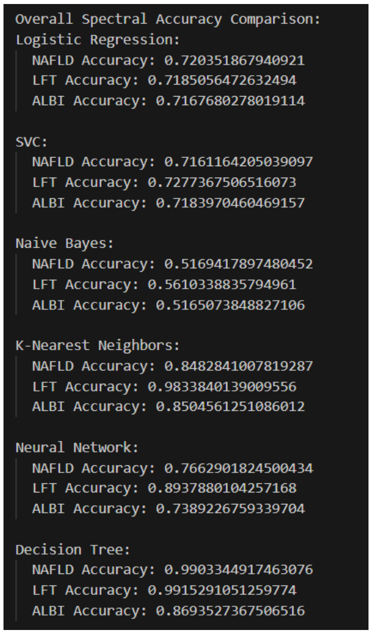
Figure 8 Shows the accuracy scores of the Models after KMeans clustering.



```
Overall DBSCAN Accuracy Comparison:
Logistic Regression:
  NAFLD Accuracy: 0.7234077211272194
  LFT Accuracy: 0.7175435738719661
  ALBI Accuracy: 0.7214530053754683

SVC:
  NAFLD Accuracy: 0.7214530053754683
  LFT Accuracy: 0.7100504968235869
  ALBI Accuracy: 0.7214530053754683

Naive Bayes:
  NAFLD Accuracy: 0.48770158006189934
  LFT Accuracy: 0.5132757778139763
  ALBI Accuracy: 0.44029972308193516

K-Nearest Neighbors:
  NAFLD Accuracy: 0.9037302492262583
  LFT Accuracy: 0.9907151001791823
  ALBI Accuracy: 0.8493239941358528

Neural Network:
  NAFLD Accuracy: 0.7724385079003095
  LFT Accuracy: 0.7512624205896726
  ALBI Accuracy: 0.7258511158169083

Decision Tree:
  NAFLD Accuracy: 0.9918553510343704
  LFT Accuracy: 0.9915295650757452
  ALBI Accuracy: 0.8687082586740511
```

Fig. 9. DBSCAN Clustering Model's Accuracy scores

Figure 9 Shows the accuracy scores of the Models after DBSCAN clustering.



```
Overall Spectral Accuracy Comparison:
Logistic Regression:
  NAFLD Accuracy: 0.720351867940921
  LFT Accuracy: 0.7185056472632494
  ALBI Accuracy: 0.7167680278019114

SVC:
  NAFLD Accuracy: 0.7161164205039097
  LFT Accuracy: 0.7277367506516073
  ALBI Accuracy: 0.7183970460469157

Naive Bayes:
  NAFLD Accuracy: 0.5169417897480452
  LFT Accuracy: 0.5610338835794961
  ALBI Accuracy: 0.5165073848827106

K-Nearest Neighbors:
  NAFLD Accuracy: 0.8482841007819287
  LFT Accuracy: 0.9833840139009556
  ALBI Accuracy: 0.8504561251086012

Neural Network:
  NAFLD Accuracy: 0.7662901824500434
  LFT Accuracy: 0.8937880104257168
  ALBI Accuracy: 0.7389226759339704

Decision Tree:
  NAFLD Accuracy: 0.9903344917463076
  LFT Accuracy: 0.9915291051259774
  ALBI Accuracy: 0.8693527367506516
```

Fig. 10. Spectral Clustering Model's Accuracy scores

Figure 10 Shows the accuracy scores of the Models after Spectral clustering.

FUTURE WORKS

Future research can build on this study by exploring several avenues to enhance clustering analysis and its applications. One area of interest is the integration of clustering methods to leverage their complementary strengths. Combining the Elbow Method or Eigengap Heuristic with DBSCAN or other density-based methods could improve clustering performance and robustness, particularly in datasets with complex structures or varying densities. Advanced ensemble methods that combine multiple clustering results could offer more reliable insights.

Another promising direction is the application of clustering techniques to large-scale and high-dimensional datasets. Implementing scalable algorithms and dimensionality reduction techniques, such as t-SNE or UMAP, can address the computational challenges and improve the interpretability of clustering results. Evaluating these methods in real-world scenarios, such as genomics, finance, or social networks, could provide valuable insights and validate their effectiveness.

Additionally, incorporating domain knowledge and leveraging hybrid approaches that combine clustering with supervised learning or feature engineering could enhance the relevance and accuracy of clustering results. For instance, integrating clustering with predictive models could help identify latent structures that are informative for subsequent analysis.

Finally, exploring the impact of different distance metrics and similarity measures on clustering outcomes can further refine method selection and application. Understanding how various metrics influence cluster formation and separation will aid in selecting the most appropriate methods for diverse datasets.

Overall, continued advancements in clustering methodologies and their applications hold the potential to unlock deeper insights from complex data and address emerging challenges in various fields.

To make the Flask application better at uploading files and extracting text, we can make a few key improvements. First, allow users to upload several files at once and support more file types like PDF and DOCX. It's also important to set a limit on file sizes and give clear error messages if a file is too big or not supported. For extracting text, we can use Optical Character Recognition (OCR) for image files and advanced tools like Apache Tika or PyMuPDF for more complex documents. If something goes wrong, the app should provide helpful error messages and suggestions. Using a background task manager like Celery can make the app more responsive by handling uploads and text extraction in the background, with progress indicators to show users how things are going. Additionally, integrating databases from cloud services and hosting the application on a cloud-based server can streamline maintenance and scalability.

On the front end, we can make the interface more modern and user-friendly. The design should be responsive, meaning it works well on different devices and screen sizes. Organize the layout clearly so users can easily find where to upload files and see the text results. Add visual cues like checkmarks and loading spinners to guide users through the process. Interactive features like drag-and-drop for file uploads and real-time previews of extracted text will make the app more engaging. Allow users to download the extracted text in different formats like TXT or PDF. Ensure the design is consistent with a good color scheme and offer options for light and dark modes to cater to different preferences. Adding small animations, like smooth transitions and hover effects, can make the user experience more enjoyable.

References

[1] H. S. Yadav and R. K. Singhal, "Classification and prediction of liver disease diagnosis using machine learning algorithms," in *2023 2nd International Conference for Innovation in Technology (INOCON)*. IEEE, 2023, pp. 1–6.

[2] A. Sharma, A. Dahiya, D. Rastogi, P. Mahajan, and P. Nagrath, "Liver disease detection using machine learning methods," in *International Conference on Computer Vision and Internet of Things 2023 (ICCVIoT'23)*, vol. 2023. IET, 2023, pp. 244–250.

[3] V.-A. Hoang, D.-T. Nguyen, H.-T. Bui, T.-L. Le, D.-H. Vu, B.-G. Tran, G.-A. Pham, H. N. Luu, T.-H. Tran, H. Vu *et al.*, "Comprehensive study of liver disease prediction using machine learning," in *2023 1st International Conference on Health Science and Technology (ICHST)*. IEEE, 2023, pp. 1–6.

[4] L. D. Sawant, R. Ritti, N. Harshith, A. Kodipalli, T. Rao, and B. Rohini, "Analysis and prediction of liver cirrhosis using machine learning algorithms," in *2023 3rd International Conference on Intelligent Technologies (CONIT)*. IEEE, 2023, pp. 1–5.

[5] M. Minnoor and V. Baths, "Liver disease diagnosis using machine learning," in *2022 IEEE World Conference on Applied Intelligence and Computing (AIC)*. IEEE, 2022, pp. 41–47.

[6] I. Hanif and M. M. Khan, "Liver cirrhosis prediction using machine learning approaches," in *2022 IEEE 13th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*. IEEE, 2022, pp. 0028–0034.

[7] S. Sontakke, J. Lohokare, and R. Dani, "Diagnosis of liver diseases using machine learning," in *2017 international conference on emerging trends & innovation in ICT (ICEI)*. IEEE, 2017, pp. 129–133.

[8] M. F. Rabbi, S. M. Hasan, A. I. Champa, M. AsifZaman, and M. K. Hasan, "Prediction of liver disorders using machine learning algorithms: a comparative study," in *2020 2nd International Conference on Advanced Information and Communication Technology (ICAICT)*. IEEE, 2020, pp. 111–116.

[9] F. Mostafa, E. Hasan, M. Williamson, and H. Khan, "Statistical machine learning approaches to liver disease prediction," *Livers*, vol. 1, no. 4, pp. 294–312, 2021.

[10] P. Decharatanachart, R. Chaiteerakij, T. Tiyarattanachai, and S. Treeprasertsuk, "Application of artificial intelligence in chronic liver diseases: a systematic review and meta-analysis," *BMC gastroenterology*, vol. 21, pp. 1–16, 2021.