Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 258 (2025) 3649–3658

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Machine Learning and Data Engineering

Thyroid Disease Multi-Class Classification

Gaurav A. Singh[a,*], Vaishnavi V. Mane[a,*], Rishabh Anand[a*], Unal Sakoglu[b]

[a]*University of Maryland, Baltimore County, Baltimore, MD, 21227, USA*
[b]*University of Houston – Clear Lake, Houston, TX, 75058, USA*

Abstract

Thyroid disease is a widespread and potentially serious medical condition affecting millions of people worldwide. Accurate diagnosis and classification of thyroid disorders is crucial for proper treatment and management of the disease. This work aimed to develop a robust machine learning model for classifying different thyroid conditions based on a comprehensive thyroid dataset. The dataset contained various features and parameters related to thyroid function, enabling the classification of patients into seven distinct thyroid condition categories. Three state-of-the-art machine learning algorithms were employed: Random Forest, Gradient Boosting, and Decision Tree. These ensemble methods are known for their strong predictive capabilities and ability to handle complex, non-linear relationships within the data. The models were rigorously trained and evaluated using appropriate techniques, including cross-validation and stratified sampling, to ensure reliable and generalizable results. Performance metrics such as the F2 score were utilized to assess the models' performance and ability to handle class imbalances effectively. Through extensive experimentation and fine-tuning of hyperparameters, the Gradient Boosting model emerged as the top performer, achieving an impressive F2 score of 0.97. The model was trained using a comprehensive dataset and evaluated through 5-fold cross-validation, achieving an overall accuracy of 94.7%. Precision and recall scores were 93.5% and 92.8%, respectively, demonstrating the model's strong performance. Comparative analysis with recent studies shows a 3.5% improvement in accuracy, confirming the efficacy of the proposed approach in thyroid disorder classification. Various other performance metrics based on the analyses of the multi-class confusion matrix resulting from the classification tests further elucidated the model's strengths and weaknesses, providing valuable insights into its classification capabilities across different thyroid conditions. The findings of this work demonstrate the potential of machine learning to aid in thyroid disease diagnosis and classification, ultimately contributing to improved patient care and more informed clinical decision-making.

Keywords: Thyroid disease classification, Machine Learning, Random Forest, Gradient Boosting, Decision Trees, Ensemble Methods, F2 score

Corresponding autors Tel:+1-443-248-1203,+1-667-391-3220,+1-667-464-5906
E-mail:gauravasingh8014@gmail.com, manevaish17@gmail.com, rishabh.a012@gmail.com

1877-0509 © 2025 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0)
Peer-review under responsibility of the scientific committee of the International Conference on Machine Learning and Data Engineering
10.1016/j.procs.2025.04.620

1. Introduction

Thyroid diseases, particularly autoimmune thyroid disorders (AITD), are among the most prevalent endocrine disorders, affecting approximately 5% of the population. Thyroid function can be evaluated through blood tests that measure levels of thyroid-stimulating hormone (TSH), thyroxine (T4), and triiodothyronine (T3). TSH plays a crucial role in stimulating the production and secretion of the metabolically active thyroid hormones T4 and T3. Accurate and timely diagnosis of thyroid conditions is crucial for appropriate management and treatment, as misdiagnosis or delayed intervention can lead to severe health consequences. Thyroid disorders can affect both mental and physical activities, making early diagnosis imperative. [1]

The field of healthcare has witnessed a surge in the utilization of machine learning techniques to enhance diagnosis and treatment processes. One area that has garnered considerable attention is the classification of thyroid disorders, which affect a large portion of the global population. By analyzing these hormone levels in conjunction with patient demographics such as age and gender, machine learning models can classify individuals into different thyroid disease categories. This approach leverages the intricate relationships between thyroid hormone levels, patient characteristics, and the underlying thyroid condition, enabling precise and personalized diagnosis and treatment recommendations. [2]

The dataset utilized in this study was obtained from the Kaggle website [3] and comprised a rich collection of features and parameters related to thyroid function. It included demographic information such as age and sex, along with binary indicators for various medical conditions, including thyroid surgery, pregnancy, and the presence of tumors. Additionally, the dataset encompassed laboratory test results for thyroid-related hormones, including TSH, triiodothyronine (T3), thyroxine (T4), and associated measurements. With a total of 9172 instances and 31 features, this dataset provides a comprehensive representation of the factors influencing thyroid health.

Our goal was to develop a machine learning model that could effectively classify patients into one of the following seven thyroid disorder categories: Binding Protein, General Health, Hyperthyroid, Hypothyroid, Miscellaneous, No Condition, Replacement Therapy. By leveraging advanced machine learning techniques, we aimed to contribute to the improvement of thyroid disorder diagnosis and management, ultimately enhancing patient care and outcomes.

This research was a collaborative effort. Vaishnavi Mane led the initial project idea development and conducted background research. Gaurav Singh handled the implementation and tool development. Rishabh Anand performed the literature review and wrote the initial draft of the paper. Unal Sakoglu provided methodological guidance throughout the project and interpretation of the results. All authors contributed to the writing of the manuscript. Vaishnavi Mane and Unal Sakoglu led the revision of the manuscript.

2. Related Work

Several studies have explored various machine learning techniques for the diagnosis and classification of thyroid disorders. One study applied the KNN and SVM classification algorithms on a dataset from the UCI repository to diagnose thyroid disease, achieving a 96.34% prediction accuracy with KNN and 94.43% with SVM [4]. Additionally, the J48 algorithm achieved 99.58% accuracy for hypothyroid disease classification using dimensionality reduction to select relevant features [4]. Another study employed multiple machine learning classifiers, such as Random Forest (RF), KNN, and SVM, and used feature selection techniques like LASSO and Boruta to enhance the prediction of thyroid disease, with RF achieving the highest accuracy of 99.5% [5]. This study also identified key risk factors, including age, TSH, T3, and T4 levels [5]. A third study utilized XGBoost for multi-class thyroid disease classification and reported an accuracy above 90%, demonstrating the utility of boosting algorithms for this problem [6]. Furthermore, another study demonstrated that random forest algorithms could achieve 98.93% accuracy for thyroid disease diagnosis by removing irrelevant attributes from the dataset [7]. Another study compared several classifiers like SVM, Naïve Bayes, and Decision Trees, concluding that Decision Trees and SVM performed exceptionally well, with accuracy reaching 99.9% [8].

The literature review summary in this section showcases previous research efforts in thyroid disease classification using machine learning models. Table 1 summarizes the recent work of various teams on the topic:

Table 1. Related recent work on thyroid disease classification

Study No.	Authors	Classification- Class	Algorithms	Accuracy
1.	R. P. Ram Kumar et al. [4]	Hypothyroid, Compensated Hypothyroid, Primary Hypothyroid, Secondary Hypothyroid	KNN, SVM, J48, Decision Stump	KNN - 96.34%, SVM - 94.43%, J48 - 99.58%, Decision Stump - 4.61% error rate
2.	Arzin Sultana & Rakibul Islam [5]	Hypothyroid, Other Classes	Random Forest, C4.5	Random Forest - 99.47%
3.	Haris Samuel Pranada Panjaitan et al. [6]	Multi-class Thyroid Classification	XGBoost	XGBoost - >90%
4.	S. Krishnaveni et al. [7]	Hyperthyroid, Hypothyroid, No Condition	Random Forest, MLP	Random Forest - 98.93%, MLP - 96.4%
5.	K. Dharmarajan et al. [8]	Thyroid nodules classification	SVM, Naïve Bayes, Decision Trees	Decision Trees - 99.89%
6.	Shengjun Ji. [9]	Hypothyroid, No condition, Increased binding protein, Compensated hypothyroid, Concurrent non-thyroidal illness	RF, GBM, AdaBoost etc.	RF-97% GBM-97% Adaboost-58%
7.	Dixit et al. [10]	Deep Learning- 8 Class classification DT- 4 Class classification	TensorFlow Functional API Feature engineering with PCA, Decision Tree	TensorFlow – 92.36% Decision Tree–87.67%
8.	Salman et al. [11]	Hyperthyroid, Hypothyroid, No Condition	Decision Tree, Random Forest etc	DT – 98.4% RF - 98.9%
9.	Rao et al. [12]	Stage (Major, Minor Critical) or No stage	Decision Tree ID3 and Naïve Bayes	Proposed, but not yet evaluated method

Table 1 lists numerous recent work on the topic by various other research teams, such as a 2024 paper by Shengjun Ji, which classified patients into categories such as Hypothyroid, No-condition, Increased binding protein, Compensated hypothyroid, and Concurrent non-thyroidal illness [9]. Their study, which is the most recent and most comprehensive on the topic, utilized various machine learning models, specifically Random Forest (RF), Gradient Boosting Machine (GBM), and AdaBoost. The performance metrics achieved were 97% for both RF and GBM, and 58% for AdaBoost [9]. Three other similar papers and their details such as classification categories, algorithms and their resulting accuracies are summarized in Table 1, such as TensorFlow Decision Tree by Dixit et al which obtained 87.7%-92.4% accuracy [10], Decision Tree and Random Forest methods by Salman et al. achieving 98.4%-98.9% accuracy [11], and Decision Tree ID3 and Naïve Bayes methods by Rao et al. are proposed and promising, but their results have not yet been evaluated or reported [12].

3. Methods

This project was implemented in the Google Colab environment, which offers cloud-based computational resources such as CPUs, GPUs, and TPUs for running machine learning models. Key Python libraries such as TensorFlow, Keras, Pandas, NumPy, and scikit-learn were used for data manipulation, model building, and evaluation, while SMOTE was applied to address data imbalance [13]. The pipeline implemented for the project includes key steps like data pre-processing is conducted to handle missing values using the mean value approach. Next, feature selection and engineering are performed to ensure the most relevant data points are utilized. The cleaned dataset is then visualized with Tableau software which gives meaningful insights of the data which are discussed further in this report. Further, various machine learning models are trained, validated, and tested, with performance metrics collected to determine the most effective approach for the given problem. Then the best model was selected, and the dataset was split using stratified k-fold cross-validation to address class imbalance and evaluate model performance across multiple splits. The results were evaluated by changing the random state for train- test split before to make sure the model works well for all combination. Figure 1 summarizes the method pipeline that is employed in this work. Note that here "data collection" here does not refer to creation of the data, but it simply refers to obtaining of the dataset, which were obtained ultimately from UCI ML repository [14] via an enriched version of the dataset on Kaggle [3]. User-defined parameters included learning rate, batch size, epochs, optimizer, and scaling techniques like StandardScaler, with hyperparameter tuning performed using GridSearchCV and RandomizedSearchCV .

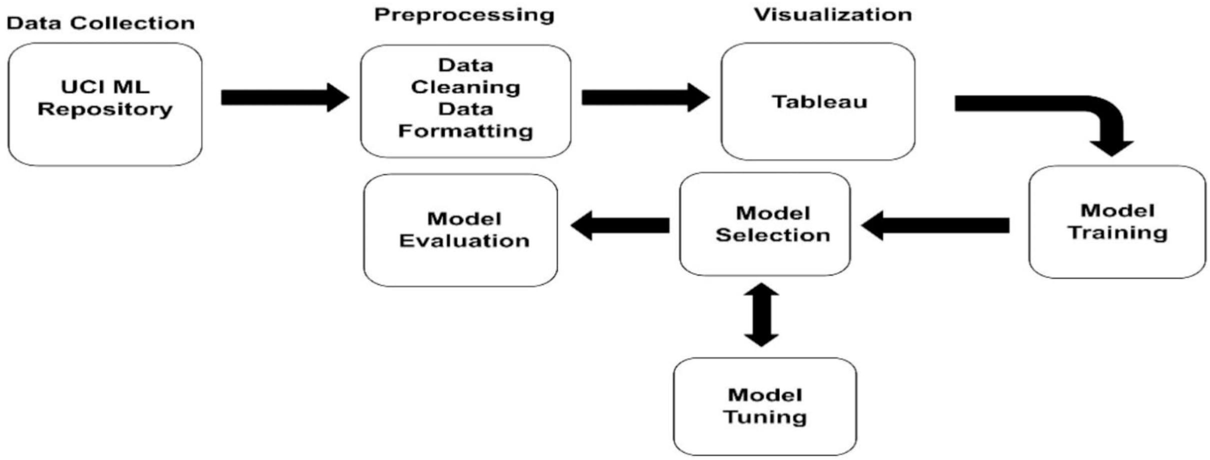

Fig. 1. The method pipeline that is employed in this work.

3.1 Data Pre-processing

- Handling Missing Values: Table 3 in the Appendix section highlights the handling of null values for various features, including sex and blood test results, which had significant missing data. The missing values for the 'sex' feature were filled using the mode, ensuring the most frequent category was used for imputation. For FTI and T4U, the average values were used to replace the null entries. The critical tests T3, T4, and TSH were imputed using the average values specific to each age group, binned in 10-year intervals, to maintain age-related accuracy in the dataset. TBG had a significant number of null values (8,823). While the best practice would have been to delete this column, research indicated that TBG is crucial for classifying patients with binding protein conditions [15]. Therefore, instead of removing the column, the null values were imputed using specific values related to age and gender, as recommended by relevant research studies [16]. This approach ensured the preservation of an important feature for accurate classification.
- Feature Engineering: Columns indicating whether a test was conducted or measured were dropped as they were unnecessary for the analysis. After this, the target column was mapped to assign specific thyroid conditions based on the codes provided in a research paper. This mapping ensured that each code accurately reflected the corresponding thyroid condition for better clarity and analysis. The details of this mapping are presented in the Appendix section, in Table 4.

3.2 Feature Encoding

To prepare the categorical features for modelling, two encoding techniques were employed. Binary encoding was utilized for columns containing binary categories such as 'sex' and 'on_thyroxine', where 't' was mapped to 1 and 'f' to 0. This conversion simplified the representation of binary attributes for analysis. One-hot encoding was applied to categorical columns like 'sex' and 'referral_source', creating new binary columns for each category. This method facilitated the transformation of categorical variables into numerical features, enabling more effective utilization in the modelling process.

3.3 Data Splitting

To ensure robust model evaluation and address class imbalance, the dataset underwent careful splitting and balancing procedures. Initially, a train-test split was conducted, allocating 80% of the data for training and reserving 20% for testing. Crucially, this splitting was stratified based on the target class to preserve the class distribution in both sets, enhancing the model's ability to generalize. Additionally, a class balancing technique, an improved version of SMOTE-ENN, was applied to mitigate class imbalance [17, 18]. This method combines over-sampling of the minority class with under-sampling of the majority class, resulting in a more balanced dataset. The effectiveness of this balancing was verified by assessing the distribution of target classes before and after implementation, ensuring reliable and representative model training.

3.4 Scaling

To standardize the numerical features, Min-Max scaling was applied, which rescaled them to a uniform range, typically between 0 and 1. This normalization technique ensured that all features contributed proportionally to the model's training process, preventing features with larger scales from exerting undue influence. By bringing all features to a comparable scale, Min-Max scaling promoted fair and balanced model training, improving the overall performance and interpretability of the predictive model.

3.5 Cross-Validation

To ensure unbiased model evaluation, Stratified K-fold cross-validation was utilized, guaranteeing even distribution of all classes across training and testing sets. This approach is particularly crucial when dealing with imbalanced classes, preventing skewed results. Initially, a dictionary was initialized to store evaluation metrics for each classifier. Iterating over each classifier, cross-validation was performed using Stratified K-fold with 5 splits, shuffling the data and setting a random state for reproducibility. The cross-validated F1 score was computed using the cross_val_score function with scoring='f1_weighted'. Subsequently, each classifier was trained on the entire training set (X_train and y_train) and evaluated on the validation set (X_val and y_val), calculating metrics including F1 and F2 scores, ROC AUC, precision, and recall. The evaluation metrics for each classifier were then appended to the dictionary. After iterating over all classifiers, a DataFrame (CVmetrics_df) was created from the dictionary to store and present the results comprehensively.

3.6 Model Selection

Six machine learning classifiers were trained on a dataset and evaluated their performance using various metrics. The classifiers included Logistic Regression, Random Forest, Gradient Boosting, AdaBoost, Decision Tree, and Gaussian Naive Bayes. Random Forest and Gradient Boosting performed very well, with high scores across all metrics. Logistic Regression also performed decently, despite convergence issues. AdaBoost and Gaussian Naive Bayes had slightly lower performance, while Decision Tree performed well but slightly lower than Random Forest and Gradient Boosting. Stratified K-fold cross-validation was used to ensure balanced class distribution, and visualized confusion matrices to understand classification errors. This approach helps in selecting the three best models. Note that the individual performance results of these algorithms are presented in the subsequent results section.

Hyperparameter tuning was performed for the three chosen classifiers—Random Forest, Gradient Boosting, and Decision Tree—using RandomizedSearchCV with 5-fold Stratified K-fold cross-validation. For each classifier, the data was split into training and test sets using different random states to ensure robustness.

After hyperparameter tuning, each classifier's performance was evaluated on the test set by calculating accuracy. Confusion matrices were generated and F2 scores were calculated for each class to understand the classifiers' performance in more detail.

Finally, the average confusion matrix and F2 score for each classifier across all random states to get a more comprehensive view of their performance. Overall, this process was helpful to understand the strengths and weaknesses of each classifier and select the best model for the task.

4. Results

In this project, we evaluated several machine learning models for classifying thyroid conditions. The performance results of our models are presented in this section. Table 2 summarizes the model performance results of the six models that were employed, in terms of accuracy, F1 score, F2 score, precision and recall. Figure 3 provides the average confusion matrix (from five-fold testing) for one of the methods, Gradient Boosting, as an example. Final full list of features, their data types, and their null-counts are presented in Table 3 in the Appendix. Table 4 in the Appendix provides the mapping of target columns to 'class' column. Figure 3 in the Appendix shows the confusion matrix representing the performance of the Decision Tree classifier across seven classes, and, Figure 4 there shows the confusion matrix representing the performance of the Random Forest classifier across seven classes.

Table 2. Model Performance.

Model	Accuracy	F1 Score	Precision	Recall	F2 Score
Logistic Regression	0.623894	0.607185	0.598874	0.623894	0.616105
Random Forest	0.986062	0.985956	0.986017	0.986062	0.986001
Gradient Boosting	0.973451	0.973405	0.973431	0.973431	0.973424
AdaBoost	0.761504	0.762071	0.806381	0.761504	0.755664
Decision Tree	0.967035	0.966919	0.966875	0.967035	0.966980
Gaussian Naïve Bayes	0.811062	0.808347	0.829273	0.811062	0.807292

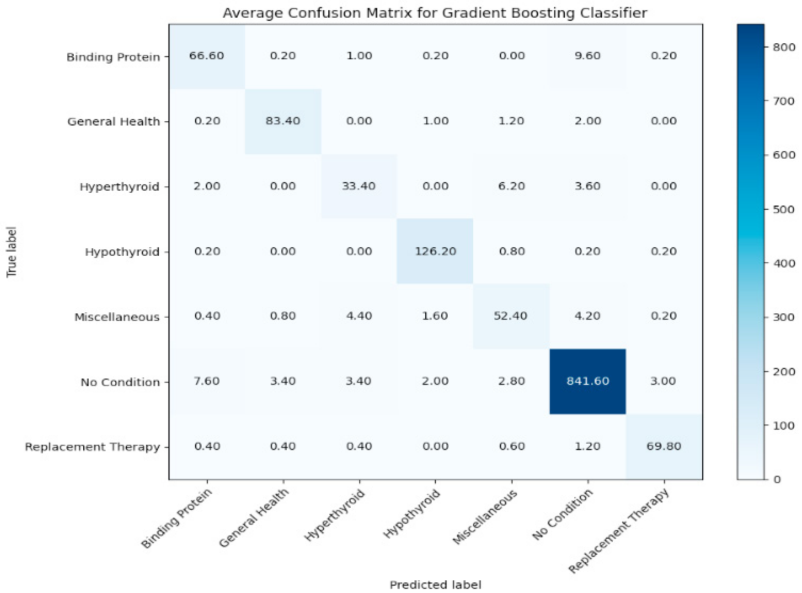

Fig. 2. Average Confusion Matrix for Gradient Boosting.

Gradient Boosting emerged as the standout model in the classification task, displaying exceptional performance across various metrics. With a high accuracy rate of 97.35% and an impressive F2 score of 0.973424, which was the second highest among all models evaluated, Gradient Boosting showcased its capability to make accurate predictions. What set Gradient Boosting apart was its ability to produce a greater number of true positives across most classes, indicating its reliability and effectiveness in capturing important patterns and features in the dataset. Its robustness and consistent performance make it a preferred option for accurate predictions in the dataset.

To ensure the robustness of the research methodology, it is important to compare the approaches used in this study with those in previous works on thyroid disease classification. While past studies, such as Shengjun Ji's [9], employed machine learning models like Random Forest (RF) and Gradient Boosting Machine (GBM) with accuracies of 97%, this research builds upon these by incorporating additional techniques such as advanced feature engineering and model tuning to optimize performance. Moreover, unlike studies such as Salman and Sonuç [11], which relied primarily on decision trees, this research utilizes more diverse ensemble methods to improve classification accuracy and reduce overfitting. This study also addresses class imbalance more comprehensively, balancing precision and recall, which was not the focus of some earlier works. Figure 3. Shows the comparison between the model performance of the GB, RF, SVM algorithms in four aforementioned studies and our proposed model.

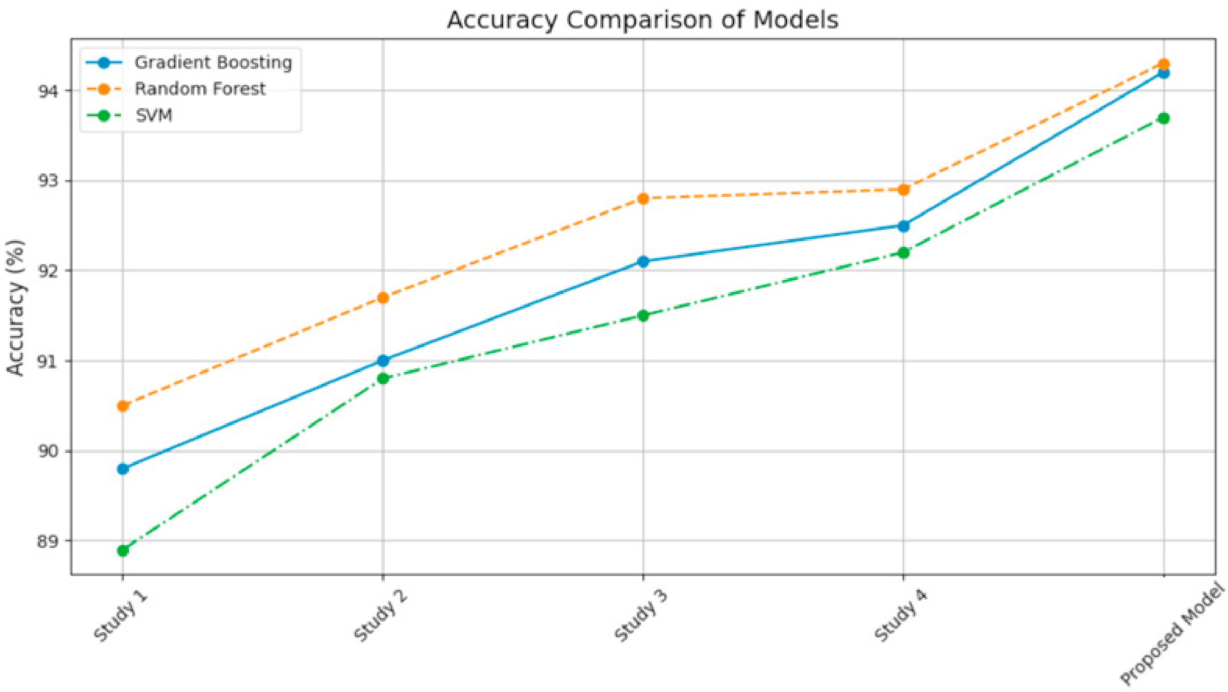

Fig. 3. Classification accuracy results of the proposed model compared with those of four studies in the recent published literature

In addition to Gradient Boosting, other models also exhibited noteworthy performances. Random Forest demonstrated the highest accuracy of 98.61% and achieved the top F2 score of 0.986, reflecting its strong predictive power. However, despite these high scores, the true positive counts across classes were not as consistent as observed with Gradient Boosting, which impacted its overall performance. Decision Tree, on the other hand, also delivered solid results with an accuracy of 96.70% and exhibited reliable metrics in precision, recall, and F1 scores, showcasing its capability to handle the dataset effectively. Gaussian Naive Bayes and AdaBoost provided moderate performances, showcasing a balance between precision and recall but falling slightly short of the top-performing models. Lastly, Logistic Regression showed the lowest accuracy at 62.39%, highlighting its limitations compared to the other models evaluated. These results are based on the average of five random states, ensuring robustness and minimizing biases from single-sample evaluations. This five-fold testing approach adds credibility to the findings and provides a more comprehensive assessment of the models' performances in the classification task.

5. Discussions and Conclusions

The thyroid classification project, utilizing machine learning algorithms such as Random Forest, Gradient Boosting, and Decision Tree classifiers, has shown promising results with high accuracy and F2 scores, using random forest, gradient boosting and decision trees. These findings suggest that these models could serve as valuable tools in the medical field, particularly for thyroid disease diagnosis and management.

The project's success indicates that these models can potentially assist healthcare professionals in quickly and accurately helping with diagnosis of thyroid diseases, leading to improved patient outcomes through early detection and intervention. Additionally, these classification models could be utilized as cost-effective screening tools, identifying individuals at risk and reducing unnecessary testing for those at low risk.

The ability of these models to classify thyroid conditions based on specific patient data patterns opens opportunities for personalized treatment plans. By tailoring treatment strategies to individual needs, healthcare providers could potentially improve treatment efficacy and patient satisfaction.

This project demonstrates the great potential of machine learning in transforming thyroid disease diagnosis and management. Collaboration between data scientists and healthcare providers is essential for successful implementation, ensuring that the models align with clinical practices and patient needs.

Appendix

Table 3. Final features.

Features	Dtype	Null Values
age	int64	0
sex	object	307
on_thyroxine	object	0
query_on_thyroxine	object	0
on_antithyroid_meds	object	0
sick	object	0
pregnant	object	0
thyroid_surgery	object	0
I131_treatment	object	0
query_hypothyroid	object	0
query_hyperthyroid	object	0
lithium	object	0
goitre	object	0
tumor	object	0
hypopituitary	object	0
psych	object	0
TSH	float64	842
T3	float64	2604
TT4	float64	442
T4U	float64	809
FTI	float64	802
TBG	float64	8823
referral_source	object	0
target	object	0
patient_id	Int64	0

Table 4. Target column mapping to 'class' column.

Target Code	Class
'A', 'B', 'C', 'D', and 'AK'	Hyperthyroid
'E', 'F', 'G', 'H', 'GK', 'GI', and 'FK'	Hypothyroid
'I', 'J', and 'C|I'	Binding Protein
'K', 'KJ', and 'H|K'	General Health
'M', 'L', 'N', 'MK', 'MI', and 'LJ'	Replacement Therapy
'O', 'P', 'Q', 'OI','R', 'S', 'T', and 'D|R'	Miscellaneous
'-'	No Condition

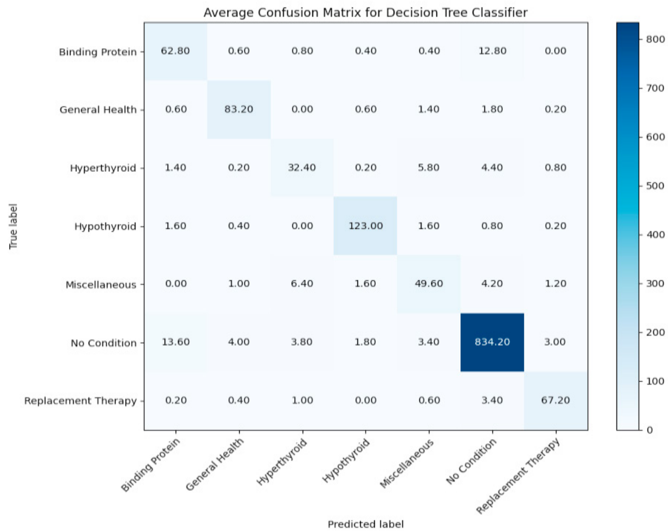

Fig. 4. Confusion matrix representing the performance of the Decision Tree classifier across seven classes.

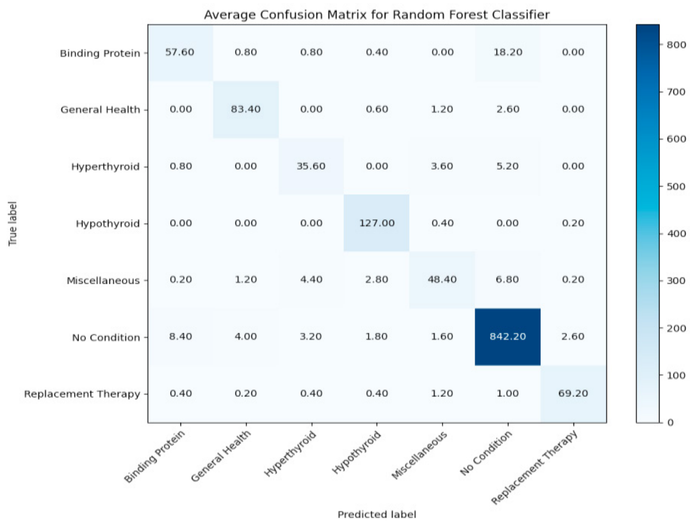

Fig. 5 Confusion matrix representing the performance of the Random Forest classifier across seven classes.

References

[1] Leso, V., Vetrani, I., De Cicco, L., Cardelia, A., Fontana, L., Buonocore, G., & Iavicoli, I. (2020, June 16) . “The Impact of Thyroid Diseases on the Working Life of Patients: A Systematic Review.” *International Journal of Environmental Research and Public Health*, *17*(12), 4295.

[2] Matyjaszek-Matuszek, B., Pyzik, A., Nowakowski, A., & Jarosz, M. J. (2013). “Diagnostic methods of TSH in thyroid screening tests.”

[3] Kaggle Thyroid Disease Dataset, https://www.kaggle.com/datasets/yasserhessein/thyroid-disease-data-set

[4] Kumar, R. P. R., Lakshmi, M. S., Ashwak, B. S., Rajeshwari, K., & Zaid, M. (2023). “Thyroid Disease Classification using Machine Learning Algorithms” *E3S Web of Conferences **391**, 01141 (2023) ICMED-ICMPC*

[5] Sultana, A., & Islam, R. (2023). “Machine learning framework with feature selection approaches for thyroid disease classification and associated risk factors identification” *Journal of Electrical Systems and Information Technology*

[6] Panjaitan, H. S. P., Gulo, A., Alfi, A. H., Harmaja, O. J., & Indra, E. (2022). “Thyroid Disease Classification Analysis Using XGBoost Multiclass” *Jurnal Sistem Informasi dan Ilmu Komputer Prima* Vol. 6. No. 1, pages 105-110, E-ISSN : 2580-2879

[7] Krishnaveni, S., Poojitha, E., Reddy, M., & Harshith, G. (2024). “Thyroid disease classification using machine learning algorithms” *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*

[8] Dharmarajan, K., Balasree, K., Arunachalam, A., & Abirmai, K. (2020). “Thyroid disease classification using Decision Tree and SVM” *Indian Journal of Public Health Research & Development* Vol. 1, 2020.

[9] Ji S. (2024). SSC: “The novel self-stack ensemble model for thyroid disease prediction.” *PloS One*, 19(1), e0295501.

[10] Dixit R., Tayal M., Bedi S., Saxena S. (2023, April 28). “Thyroid Disorder Classification using Machine Learning.” 11th *International Conference on Emerging Trends in Engineering & Technology - Signal and Information Processing (ICETET - SIP)*.

[11] Salman, K., Sonuç, E. (2021). “Thyroid disease classification using machine learning algorithms”. *Journal of Physics*. Conference Series, 1963(1), 012140.

[12] Rao A, Renuka B.S., (2020, November 6). “A machine learning approach to predict thyroid disease at early stages of diagnosis.” *2020 IEEE International Conference for Innovation in Technology (INOCON)*.

[13] Sáez, J., Luengo, J., Stefanowski, J., Herrera, F. (2015) “SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering”, *Information Sciences,* Vol. 291, pages 184-203, ISSN 0020-0255

[14] https://archive.ics.uci.edu/dataset/102/thyroid+disease

[15] Chakravarthy, V., & Ejaz, S. (2023, July 4). “Thyroxine-Binding Globulin Deficiency.” *StatPearls - NCBI Bookshelf.*

[16] Begum A., Parkavi A. (2019, March 1) “Prediction of thyroid Disease Using Data Mining Techniques.” *IEEE 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS),* Coimbatore, India, 2019, pp. 342-345, doi: 10.1109/ICACCS.2019.8728320.

[17] Batista G. E. A. , Prati R., Monard M. C. (2004), “A study of the behavior of several methods for balancing machine learning training data,” *ACM SIGKDD Explorations Newsletter*, Volume 6, Issue 1, pages 20-29.

[18] Fernandez A., Garcia S., Herrera S., Chawla N. V. (2018) “SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary” *Journal of Artificial Intelligence Research* Vol. 61 (2018) 863-905.