Machine Learning-Based Intrustion Detection for Internet of Medical Things (IoMT)

Nicole Smitheman  
Umair Bilal Chaudhry  
Computer and Information Systems (Conversion)

*Abstract*— The Internet of Medical Things (IoMT) is becoming an increasingly vital part of the healthcare industry. By providing the ability to wirelessly communicate between a network of devices, IoMT allows professionals to collect and monitor health information such as heart rate, and make informed medical decisions based on this data. Examples include implantable medical devices (IMDs) such as pacemakers, infusion pumps and smart imaging machines. IoMT devices often lack adequate security mechanisms, with many devices utilising unauthorised, unencrypted networks. Prior research has utilised machine learning algorithms for intrusion detection in communication data. However, there are current limitations due to the common use of non-IoMT datasets, making generalisation limited. Datasets are often imbalanced, and accuracy is lower for minority attack classes. This paper details the use of machine learning for intrusion detection in IoMT systems by detecting anomalies or suspicious activity in communication patterns by utilising an IoMT dataset. This can be used to accurately identify real-time intrusions so that they can be handled promptly, thus improving cybersecurity of these devices.

Keywords—Healthcare, Internet of Medical Things, IoMT, Anomaly Detection, Cybersecurity, Threat Detection, Machine Learning

# **Introduction**

Over recent years, technology in the healthcare sector has become increasingly prominent. Technology plays a large role in storing and managing health data, monitoring patient vitals, planning and implementing treatments, assisting during surgery, and especially over recent years, enabling the expansion of the internet of medical things (IoMT). IoMT aligns with the concept of the internet of things (IoT) but directly refers to the healthcare field. Although IoMT has brought many benefits with it, additional connectivity brings heightened cybersecurity risks. If an adversary is successful in compromising a device or network, there may be a loss of data integrity, patient confidentiality, or data availability (Thomasian and Adashi, 2021). In the healthcare industry, this could mean a risk to the lives of many patients.

There are more prominent cybersecurity risks associated with IoMT devices, and if we go into more detail and look specifically at implantable medical devices (IMD), a branch of IoMT, we can highlight a number of specific common vulnerabilities. These vulnerabilities include man-in-the-middle attacks, code injection, inserting system malware, denial-of-service and side channel attacks to name a few (Kwarteng and Cebe, 2022). If the network of the IoMT device is unsecure, this could allow an adversary to eavesdrop, monitor, and potentially even change the data or treatment of the IMD. A laboratory experiment showed that it was even possible to shock a patient with a pacemaker. The experiment also showed that it was feasible to slowly deplete the battery of the devices over time by forcing the device to perform energy depleting actions. This could be life threatening if the drained battery was not detected in time or may result in the patient needing further surgery to replace the drained device. (Camara, Peris-Lopez and Tapiador, 2015). This would be a type of denial-of-service attack.

As attacks grow more sophisticated, there has been growing concern around whether IoMT devices are adequately secured. An example of this was with Dick Cheney, a previous American vice president who made the decision to have his implantable defibrillator replaced with one that does not have the ability to wirelessly connect to Wi-Fi due to security concerns. (Camara, Peris-Lopez and Tapiador, 2015).

Table 1: Table showing what cybersecurity concepts could be at risk without adequate IoMT security (Camara, Peris-Lopez and Tapiador, 2015).

|  |  |
| --- | --- |
| ***Cybersecurity concept*** | ***How it could be broken in IoMT*** |
| Confidentiality | Eavesdropping on patient data if not encrypted, potential to determine type of device or treatment or release confidential data. |
| Availability | Denial-of-service attacks, ransomware, battery-drainage, data flooding, communication disruption, delay of treatment. |
| Data integrity | Intercepting and changing data or communication, changing treatment or even dosage levels. |
| Authentication | Weak authentication protection could lead to attacker being able to access devices that are prohibited. |
| Non-repudiation | Deleting proof of access log, or if there are inadequate procedures in place we may not be able to trace user. |
| Authorization | Privilege escalation, leading to the possibility of changing treatment or turning off the device. |

The main aim of this paper is to use supervised machine learning techniques, namely Random Forest, Isolation Forest, SMOTE and XGBoost, to accurately detect anomalies in IoMT device communication that may indicate suspicious activity or potential cybersecurity threats. The remainder of the paper covers a literature review of related work in section II, the methodology used including a dataset description in section III, an evaluation of the machine learning models, and the performance results of the trained models in IV followed by a conclusion of findings and future work in section V.

# **Related work**

This section covers a brief overview of key research papers, all of which aim to improve and evaluate current intrusion detection technology, specifically in context to machine learning and IoMT communication. These papers overview current cybersecurity solutions and their limitations, as well as possible improvements in real-world threat detection accuracy.

## Tree-based classifiers and filter-selection techniques

The first paper reviewed, by Balhareth and Ilyas (2024) provides a thorough overview of the IoMT cybersecurity risks and the current limitations of machine learning-based intrusion detection techniques. A key limitation highlighted is the lack of quality labelled datasets available that are relevant to IoMT environments. This can affect the quality of the machine learning models produced and the ability to generalise the findings to real-life scenarios. The paper explores combining both tree-based classifiers with filter-based feature selection in an attempt to improve model accuracy. Currently their work is focused on binary classification for intrusion detection, but in the future they aim to be able to expand the models to additionally classify the type of intrusion captured, such as spoofing or malware injection.

## Enhancing Intrusion Detection Systems in IoMT

The next study by Domenech et al (2025) also details current limitations with the use of machine learning-based intrusion detection when applying to IoMT. The paper explains that many current Intrusion Detection Systems (IDS) for IoMT use are trained using Internet of Things (IoT) datasets, but this leads to lower accuracy levels when it comes to detecting attacks. IoT can have different communication environments, devices and attack patterns in comparison to IoMT so should not be generalised to an IoMT system. The authors detail how this can cause a large drop in model performance, such as a drop in the F1 score by up to 66.87% when using models from one dataset on another dataset. This paper summarises that using datasets specifically from IoMT communication can significantly improve attack detection accuracy levels, making models more generalisable to real-life.

## Surveying machine learning-based IDS techniques (Si-Ahmed, Al-Garadi and Boustia, 2022)

This paper discusses the use of machine and deep learning methods for IoMT intrusion detection. Emphasis is placed on the fact that traditional cybersecurity methods like cryptography are not alway suitable for detecting modern day attacks such as zero-day attacks. Even machine learning-based methods need to be constantly revised to remain effective against the ever-expanding risk of cyber threat. An adequate cybersecurity solution needs not to be overcomplicated as many IoMT devices have low capacity and power due to factors like the need to preserve battery-life of the device, and the small size of many devices. The authors state that third-party device security also needs to be considered, as a common practise is to transmit the data to these external devices for storage and processing.

Expanding from these concerns, the paper provides a survey of various IDS methods such as both signature-based, anomaly-based and hybrid detection techniques. The authors dive into the different machine learning methods and their effectiveness, ranging from decision trees to convolutional neural networks (CNNs). As with the previous related works mentioned, the authors of this paper also state that the majority of data used in machine learning in this sector is outdated and may not be so applicable to real-world scenarios. In summary, the paper raises awareness of the need for more robust, versatile IDS methods that can be applied to IoMT and is capable of being able to adapt to ever-evolving cyber threats.

## RCLNet for anomaly-based intrusion detection

Over time, IDS’ have slowly started to become more advanced, especially due to the rising sophistication of machine learning models. Another method of intrusion detection by Shaikh et al (2024) involves RCLNet, which uses a combination of feature selection with Random Forest, SAALM (Self-Adaptive Attention Layer Mechanism) and CNN-LSTM in order to detect traffic communication anomalies. The model was trained with the same dataset that is being used in this study, the WUSTL-EHMS-2020 dataset. This is useful for comparison when evaluating the performance metrics of my model. The authors report that the proposed model achieved an accuracy of 99.78%, an excellent metric for potential in intrusion detection. The WUSTL-EHMS-2020 dataset being used in this paper reinforces that this source is representative and reliable for use in IoMT intrusion detection, and allows for a direct comparison between other machine-learning models.

From this literature review, several gaps in current research can be identified. Many studies make use of binary classification, where they class data as either normal or attack data. This project aims to additionally identify specific attack types, namely spoofing and data alteration. Another limitation is the reoccurent use of IoT datasets, in which communication data patterns may behave differently to patterns with IoMT devices, lowering generalisability. To handle this limitation and improve model performance, this study uses a dataset that focuses on IoMT-specific communication data.

# **Methodology**

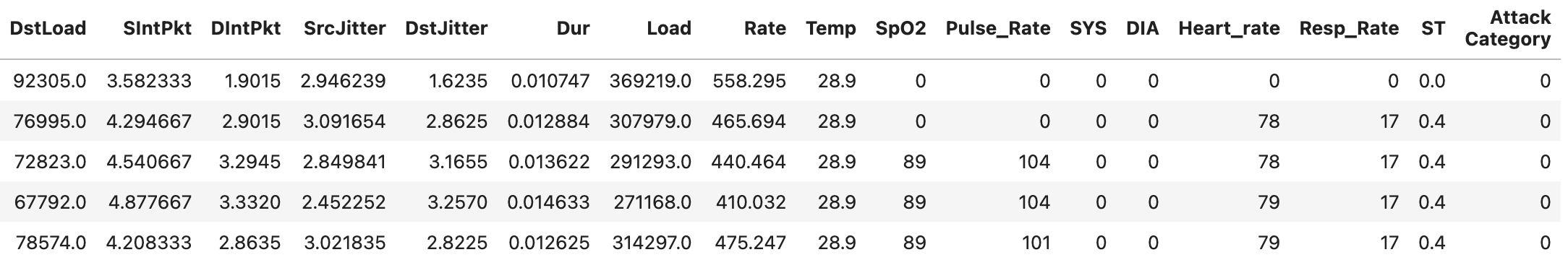
## Data description and processing

The dataset being used for the project is the WUSTL EHMS 2020 dataset (Ghubaish, 2020) and is based on live data from an Enhanced Healthcare Monitoring System. The dataset is made for the use of machine learning concerning not just IoT, but IoMT specifically. Initially it contained 16,318 records, and 45 columns with features like patient vitals, source and destination addresses, and attack categories. The dataset contains both normal and attack data to allow for a comprehensive comparison, and contains multiple attack categories.

During the data processing stage, the first step was to ensure there were no null values, of which none were present. Any unnecessary columns were then dropped, such as the columns where the values were unchanging or features like the MAC addresses which were irrelevant in this case. This was to help reduce unnecessary noise in the data. After processing the unnecessary columns, there were 19 features remaining to be used for training. Categorical non-numerical values (attack category and flags) were then converted into numerical values using label encoding.

Finally, the dataset was then split to prepare it for model training into 80% training data and the remaining 20% was used as the test set to ensure a sufficient evaluation of the machine learning model trained. A sample of the dataset with some remaining features is displayed in figure 1.

*Figure 1: Sample of the cleaned dataset, displaying some of the selected features used for training the model.*



*Figure 2: Heatmap displaying feature correlation between dataset features and the target variable, attack category.*

Figure 2 displays a correlation heatmap between the remaining features and the target variable, attack category. Features displayed in red show a stronger positive correlation to the attack category in comparison to the blue, which represents a weaker correlation. The diagram shows that the most strongly correlated features are Flgs (the protocol flags), followed by DstJitter (Destination Jitter) and DIntPkt (Destination Inter-Packet Time), suggesting that these features play an important role in highlighting if communication patterns are normal or anomalous.

## Supervised machine learning techniques

Supervised machine learning was used to detect communication anomalies via classification and then evaluated for effectiveness using a range of metrics. Supervised learning was chosen as models can be trained based on the labelled dataset, allowing these models to draw out patterns. The technique is also prominent in IoMT literature, making it a good comparison during evaluation.

# *Random forest*

Random forest was the baseline supervised learning algorithm chosen for anomaly detection and classification. The algorithm uses a range of decision tree predictions to make an overall prediction (Vargaftik et al, 2019). Random forest is a popular algorithm due to its ease of use and resistance to overfitting data. These factors mean that Random Forest tends to perform well when evaluating trained models with metrics such as accuracy and F1-score. Due to the class imbalance within the dataset, with a large majority of data in the normal class, and a much smaller sample size of attack data, class weights were balanced before training the data so that the model knows to focus on underrepresented attack classes to help improve classification accuracy. This was done by setting the class\_weight = ‘balanced’ parameter with scikit-learn.

# *Hybrid model with anomaly detection – Isolation Forest*

With the aim to further enhance anomaly detection, Random Forest was then combined with Isolation Forest to create a hybrid model. Isolation forest was trained using only the ‘normal’ data in the dataset and was then used to provide a score to the remaining data. This score was used to create an additional dataset feature called an ‘anomaly score’, which measures the extent to which each dataset column is abnormal. The algorithm has a threshold that identifies data as either normal or anomalous depending on the score (Liu, Ting and Zhou, 2008). When the dataset is then trained again with Random Forest, the aim is to help the model classify the data correctly by providing additional information about the dataset with the extra feature, and this combination may identify additional patterns.

# *Random Forest with Synthetic Minority Over-Sampling (SMOTE)*

The baseline Random Forest algorithm was then combined with SMOTE in order to focus on the large class imbalance in the data, again with there being a lower ratio of attack data in comparison to normal data. SMOTE is a technique that can be used to even out the distribution of classes by producing more samples of the underrepresented data. SMOTE does not add any more information to the dataset, but simply balances the classes. Using SMOTE alongside Random Forest should, in theory, improve model performance amongst the minority classes (Fernandez et al, 2018).

# *XGBoost (eXtreme Gradient Boosting) model*

This model is unrelated to Random Forest, and was implemented to allow for comparison between the Random Forest and various hybrid methods. XGBoost uses ensemble learning, in a process that combines multiple separate models as base learners, such as decision trees, to improve overall performance (Chen and Guestrin, 2016). Each model builds from the previous tree outcome, also correcting previous errors. Decision trees can be weak individually and are often prone to overfitting. XGBoost increases the strength of the ‘weak’ learners by building on the mistakes of the base learner (Chen and Guestrin, 2016). This model often produces good performance despite class imbalances and may perform better than Random Forest alone.

# *Adjusting thresholds*

In an aim to improve detection evaluation metrics, in particular spoofing detection, thresholds were adjusted on the original standalone Random Forest model and on the best performing hybrid or alternative model which was XGBoost. Adjusting thresholds can be useful, especially in cases where there is a large class imbalance in the dataset. In standard algorithms, there is an assumption that the dataset classes are balanced, and when they are not balanced the models may not perform as they should (Hong, Ghosh and Srinivasan, 2016). In this case, the thresholds of both models were adjusted from the default to 0.25 for the spoofing class in an attempt to raise the model performance, as there was an underperformance when it came to correctly identifying this feature. The thresholds for other features remained at the default value.

# Evaluation

To ensure a comprehensive, well-rounded evaluation of the various models tested, several evaluation metrics were chosen for comparison. These metrics can be used to evaluate how effective the model can distinguish between normal communicative data and anomalous communicative data during an attack. The evaluation metrics assessed accuracy, precision, F1-score and recall. A confusion matrix was also produced for some of the best performing models in order to visualise the rate of true and false positives and negatives produced by each, allowing for a comparison in error rate. Although accuracy can provide a good initial view into model performance, the metric on its own may not provide enough of a complete view, especially if there is a class imbalance between the features being measured. In this dataset, the majority of the data is classed as ‘normal’ data. This can be an issue because for example, if the model predicts all of the data as normal it could still achieve a high accuracy even though it will not detect any attack data. The other evaluation metrics are important because even if a model scores a high accuracy, it may not have a sufficient precision or F1-score for the individual classification features.

Following the evaluation of each model, further steps were performed with the base Random Forest Model and XGBoost. Precision-Recall curves were used to visualise differences between precision and recall. PR-AUC scores were also calculated to gain further insight into model performance.

Stratified K-Fold Cross-Validation was also performed on XGBoost, the best performing model, in order to provide a comprehensive evaluation. This technique involves testing the model on different splits of the data, in this case 5 times (where k = 5), and taking the average F1 score. This technique is particularly useful in datasets like this one, where there is a class imbalance as it ensures that each class is still accounted for in each split. It provides a more thorough evaluation because it involves evaluating the model multiple times, even with the same dataset. This increases confidence in the model performance when being used on new data (Prusty, Patnaik and Dash, 2022).

# **Results**

This section overviews the evaluation outcomes of applying several machine learning algorithms to the WUSTL-EHMS-2020 dataset. The first section details the outcome of the model evaluations, followed by some examples of the confusion matrices to display the proportion of correct predictions. The paper then details the outcome of threshold tuning and XGBoost cross-validation. Finally, the last part of the results section highlights the most important features needed to accurately classify potential attack data with different models. Additional supporting figures such as precision-recall curves are detailed in the appendix section.

## Model Performance evaluations

# Baseline Random Forest

Figure 3 below shows the classification report for the Random Forest model, using numerous evaluation metrics. The evaluation shows that the model performed strongly in predicting ‘normal’ and attack data categorised as ‘data alteration’, with recall of 100% and 99%, respectively. However, the model was limited when accurately categorising spoofed communication, with a recall of only 0.18, and an F1- score of only 0.3. These scores show that the majority of spoofed communication data was not correctly classified. Despite the model being limited in regards to spoofed communication, it still achieved an overall accuracy of 94% indicating that the model performed to a fair standard generally. However, there is still room for improvement.

*Figure 3: Classification report showing the evaluation matrices, accuracy, precision, recall, f1-score and support between each class.*

A useful method of visualising the performance of the model is by creating a confusion matrix, as shown in Figure 4. The matrix visually breaks down the model’s predictions, showing the proportion of correctly and incorrectly classified data for each class. The model performed well when classifying normal and attack data in the ‘data alteration’ category, correctly identifying 2851 out of 2855 normal instances and 183 out of 184 data altercation instances. However, for the spoofing data, the model only showed 40 true positives, with the other 185 instances being incorrectly classified as normal data. These findings are consistent to the classification report findings in Figure 3, which also identified this limitation in the model.

*Figure 4: Confusion matrix showing the rates of predicted and actual classifications by the Random Forest model.*

# II. Random Forest with SMOTE

Incorporating SMOTE into the Random Forest algorithm improved spoofing detection performance overall, increasing recall from 0.18 to 0.45 and the spoofing F1 score from 0.30 to 0.48. This came at the cost of accuracy, with it decreasing to 93% and spoofing precision from 0.91 to 0.52. Overall, this SMOTE-Random Forest Hybrid produced a higher performance than Random Forest alone, but at the cost of other evaluation metrics.

# *III.Random Forest with Isolation Forest*

Using a hybrid algorithm resulted in a very similar outcome to the baseline Random Forest Model, with the spoofing precision increasing from 0.91 to 0.93, but every other metric remaining the same. This indicates that using this hybrid does not provide a significant enough performance improvement to be useful.

# *IV.XGBoost*

XGBoost was the best performing algorithm overall, with a 96% accuracy, spoofing recall of 0.43 and macro F1-score of 0.85. This model produced the lowest amount of misclassified spoofing classes. This shows that XGBoost may be the best model to use in a real-life setting to accurately identify anomalous data.

Figure 5 below shows a confusion matrix for XGBoost and confirms the classification report statistics. Like the other models, there is good performance for both normal and data alteration attack data, with 2843 out of 2855 normal and all 184 data altercation points being correctly classified. There is still an underperformance for spoofing data, with only 96 out of 255 cases identified correctly, but in comparison to other models there is a significant improvement.

*Figure 5: Confusion matrix showing the rates of predicted and actual classifications by the XGBoost model.*

# *V.Threshold tuning*

After adjusting the thresholds for Random Forest and XGBoost, there was a notable increase in model performance for both algorithms. Random Forest spoofing recall increased from 0.18 to 0.43, but this was still outperformed by XGBoost which saw a further increase from 0.43 to 0.56 making it the best performing overall model. However, adjusting thresholds resulted in a lower precision in both models, and the accuracy for both models remained unchanged, making the effects of adjusting the thresholds limited when looking at the overall performance, even with improved spoofing detection. Different thresholds were also tested on both models, and 0.25 was found to be the optimal threshold for the highest overall performance increase.

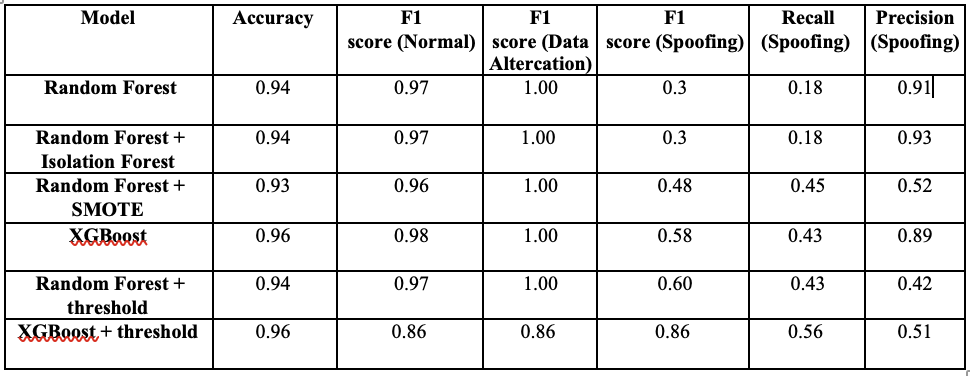
# *VI.Stratified K-Fold Cross-Validation*

Stratified K-Fold Cross Validation was performed on XGBoost as it was the best performing model, and XGBoost produced a mean macro F1-score of 0.852, across a 5 fold validation. This further validates XGBoost being the best performing model, and suggests that it is the most generalisable to real-life communication and across alternative datasets.

# *Overall model comparison*

Table 2 summarises model performance across all algorithms, focusing particularly on accuracy, F1 scores and spoofing recall and precision. XGBoost was the best performing model with the highest accuracy of 96% and better overall statistics in comparison to the baseline Random Forest model and its hybrids.

*Table 2: Table displaying a comparison of all tested models, showing accuracy, macro F1 score, recall for spoofed data and PR-AUC scores for spoofing data*



# Feature importance

Figure 6: Bar graph displaying the top 10 most important features when identifying attack data with Random Forest.

*Figure 7: Bar graph displaying the top 10 most important features when identifying attack data with XGBoost.*

Figure 6 and 7 show the top 10 most important features used by the Random Forest model and XGBoost when identifying attack data. Feature importance is determined by the amount that a particular feature helps to decrease classification error (Breiman, 2001). The top three most important features were DIntPkt (Destination Inter-Packet Time), DstJitter (Destination Jitter) and Dur (Duration) for Random Forest. DIntPkt and DstJitter are related to the network, rather than factors like temperature, which relates to patient vitals. From these results, we can conclude that network-based data may be more important for identifying attack data and anomalous communication.

On the other hand, the XGBoost model displays notable differences with the top features being Flgs, SrcLoad and SIntPkt. The features that were important in Random Forest are not highly ranked. This difference in feature importance shows that different algorithms can distinguish different patterns in the same dataset, varying the model performance and outcome. Random Forest focused mainly on network-based features, whereas XGBoost focused on categorical data and network load. Notably, XGBoost’s most important feature, Flgs was previously ranked as having the highest correlation in the correlation heatmap shown in figure 2, but this feature is not even within the top 10 most important features for Random Forest.

# **Discussion and conclusion**

## Model performance on an imbalanced dataset

Overall, the Random Forest model applied to the WUSTL-EHMS-2020 dataset demonstrated a strong performance when detecting attack communication traffic. Most classes were detected accurately, demonstrating that the model is suitable for classifying normal and data alteration traffic, especially for classes that are adequately represented in the dataset. However, the unbalanced nature of the dataset caused a drop in performance for the minority classes. This issue was previously highlighted in the literature review by Balhareth and Ilyas (2024) and Domenech et al (2025), where the authors mentioned that there was currently a lack of high-quality datasets specific to IoMT. This is currently an issue that may be holding back machine learning model performance. Alternative models explored did make a difference to spoofing attack data detection, especially when adjusting thresholds and using XGBoost as an alternative. This shows that this issue can be improved with adjustments and exploring different models and algorithms.

## Detecting specific attack types

There was a notable limitation across all models when it came to identifying spoofing attack data and this was a more prominent issue in some of the models in comparison to others. The evaluation metrics displayed a poor model performance in correctly categorising this type of communication data, even when the model displayed good performance for identifying other attack types. A reason for this could be that there was a limited number of spoofing attack data points in the dataset, possibly making it more difficult for the model to correctly identify feature patterns specific to this type of traffic. Using hybrid models such as isolation forest, adjusting thresholds and testing other algorithms like XGBoost improved spoofing detection, but there is still growth potential. It may also be useful in the future to use multiple datasets where available to broaden the amount that the model can learn and make it more generalisable to different attack types.

Overall, this project successfully trained and evaluated multiple supervised machine learning models. Attack data was successfully identified, but specific limitations were revealed when it came to identifying rarer attack types. This reiterates that it is important to thoroughly evaluate machine learning models, as accuracy metrics are often not enough alone to reveal the true generalisability of the model to real-life scenarios.

## XGBoost as the best-performing algorithm

XGBoost was the best performing model tested, suggesting that it would be the most suitable overall for intrusion detection in IoMT systems. Its strong performance may be due to the use of gradient boosting, where each new tree developed corrects the errors of the previous tree. This technique can produce more accurate outcomes as it allows for the detection of more subtle patterns in the data that weaker models may fail to distinguish. The Flgs feature was ranked with the highest importance to the algorithm, which aligns with the earlier feature correlation heatmap and further reinforces the conclusion of XGBoost being the most suitable algorithm.

## Effect of threshold tuning

Adjusting thresholds did improve some areas of model performance, notably the recall for both Random Forest and XGBoost. However, this came with the cost of reduced performance in other areas like precision. In the real world, focusing on increasing recall could be justified in the sense that there may be less risk of missing an attack. However, this may lead to many false alarms and would require careful reflection as to whether this is more advantageous than using the baseline thresholds.

## Practical implications

The results of this study have various practical applications to real world intrusion detection systems, specifically within IoMT. The improvement in spoofing communication detection demonstrates that existing systems have the potential to be refined through exploring various adjustments and hybrid models, even with a class imbalance within datasets that they may be trained with. In a healthcare setting, improving model performance even a small amount can make a substantial difference to patient’s lives.

Secondly, the XGBoost algorithm is a fairly light-weight algorithm, which is important in IoMT as many devices in this framework need to be simple and need low system processing due to requirements to extend battery life for as long as possible. It may also be useful in the future to explore algorithms that are even more lightweight, such as LightGBM (Ket et al, 2017), which also uses gradient boosting but has an extra focus on being specifically lightweight.

Another real-world implication of this study is again highlighting what previous literature emphasises (Balhareth and Ilyas, 2024), that there is a need for more IoMT datasets, with better representation of high-quality attack data. Currently, the majority of data is IoT-related and there is very little data on specific IoMT devices such as implantable medical devices. This leads to model underperformance in real-life practice. There is a growing necessity for additional datasets, especially in a world where there is an ever-increasing risk of cyberthreat.

Additionally, the actual experimental set-up increases the project credibility due to the use of one of the few IoMT datasets currently available, making the study reproducible and comparable with similar research. The algorithms tested such as Random Forest and XGBoost are also reliable, well-known algorithms, alongside well-accepted evaluation metrics. This makes my work easily comparable and opens the possibility for future expansion.

# **Future work**

Although the various algorithms, especially XGBoost, scored strongly in the evaluation using the WUSTL-EHMS-2020 dataset, there is still room for performance improvement in several areas. One major limitation was the poorer performance when detecting attack data from the spoofing class specifically, expected due to the small number of instances in the dataset. It would be useful for future work to focus on representing these rarer data types either by training the models with alternate IoMT datasets or using alternative techniques more advanced than SMOTE to make up for the class imbalance.

While several algorithms were explored in this project, it would be beneficial to expand the project to compare further alternative models, such as deep learning, to improve detection potential. These algorithms have the potential to detect more complex communication traffic produced by IoMT devices, and performance may be improved by combining the different strengths of distinct models. Testing these on other datasets would again be useful to gain insight into robustness and applicability when it comes to using the models across environments.

Another useful direction could be to focus more on specific IoMT devices, such as implantable medical devices, to observe whether the communication traffic varies significantly between different types of devices. However, currently there is a lack of sufficient datasets, likely due to the sensitivity of healthcare related data, making it more limited because of factors such as patient confidentiality. This limitation could be partially overcome by generating synthetic data that mimics real IoMT communication, allowing for alternative models to be trained without the risk of exposing confidential patient information.

##### **Acknowledgment**

I would like to express my true gratitude towards my project supervisor, Umair Bilal Chaudhry, for guiding me throughout the project. His feedback and support were essential to its successful completion.

##### **References**

1. Balhareth, G. and Ilyas, M. (2024). Optimized Intrusion Detection for IoMT Networks with Tree-Based Machine Learning and Filter-Based Feature Selection. *Sensors*, 24(17), p.5712. doi:https://doi.org/10.3390/s24175712.
2. Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), pp.5–32. doi:https://doi.org/10.1023/A:1010933404324
3. Camara, C., Peris-Lopez, P. and Tapiador, J.E. (2015). Security and privacy issues in implantable medical devices: A comprehensive survey. *Journal of Biomedical Informatics*, 55, pp.272–289. doi:https://doi.org/10.1016/j.jbi.2015.04.007
4. Chen, T. and Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System*. [online] Available at: https://arxiv.org/pdf/1603.02754.
5. Doménech, J., León, O., Siddiqui, M.S. and Pegueroles, J. (2025). Evaluating and enhancing intrusion detection systems in IoMT: The importance of domain-specific datasets. *Internet of Things*, [online] 32, p.101631. doi:https://doi.org/10.1016/j.iot.2025.101631
6. Fernandez, A., Garcia, S., Herrera, F. and Chawla, N.V. (2018). SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary. *Journal of Artificial Intelligence Research*, 61, pp.863–905. doi:https://doi.org/10.1613/jair.1.11192.
7. Ghubaish, A. (2020). *WUSTL EHMS 2020 Dataset for Internet of Medical Things (IoMT) Cybersecurity Research*. [online] Wustl.edu. Available at: <https://classes.engineering.wustl.edu/~jain/ehms/index.html> [Accessed 25 June. 2025]
8. Hong, C., Ghosh, R. and Srinivasan, S. (2016). *Dealing with Class Imbalance using Thresholding*. [online] arXiv.org. Available at: <https://arxiv.org/abs/1607.02705>? [Accessed 6 Aug. 2025].
9. Hosain, Y. and Çakmak, M. (2025). XAI-XGBoost: an innovative explainable intrusion detection approach for securing internet of medical things systems. *Scientific Reports*, [online] 15(1). doi:https://doi.org/10.1038/s41598-025-07790-0.
10. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. and Liu, T.Y., 2017. LightGBM: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*, 30. Available at: <https://arxiv.org/abs/1712.01047> [Accessed 13 Aug. 2025].
11. Kwarteng, E. and Cebe, M. (2022). A survey on security issues in modern Implantable Devices: Solutions and future issues. *Smart Health*, 25, p.100295. doi:https://doi.org/10.1016/j.smhl.2022.100295
12. Liu, F.T., Ting, K.M. and Zhou, Z.-H. (2008). Isolation Forest. *2008 Eighth IEEE International Conference on Data Mining*. doi:https://doi.org/10.1109/icdm.2008.17.
13. Prusty, S., Patnaik, S. and Dash, S.K. (2022). SKCV: Stratified K-fold cross-validation on ML classifiers for predicting cervical cancer. *Frontiers in Nanotechnology*, 4. doi:https://doi.org/10.3389/fnano.2022.972421.
14. Shaikh, J.A., Wang, C., Muhammad, W.U.S., Arshad, M., Owais, M., Alnashwan, R.O., Chelloug, S.A. and Muthanna, M.S.A., 2024. *RCLNet: an effective anomaly-based intrusion detection for securing the IoMT system*. Frontiers in Digital Health, 6, p.1467241. Available at: <https://www.frontiersin.org/articles/10.3389/fdgth.2024.1467241/full> [Accessed 3 Jul. 2025].
15. Si-Ahmed, A., Al-Garadi, M.A. and Boustia, N. (2022). Survey of Machine Learning Based Intrusion Detection Methods for Internet of Medical Things. *arXiv:2202.09657 [cs]*. [online] Available at: <https://arxiv.org/abs/2202.09657> [Accessed 2 Jul 2025]
16. Thomasian, N.M. and Adashi, E.Y. (2021). Cybersecurity in the Internet of Medical Things. *Health Policy and Technology*, 10(3), p.100549. doi:https://doi.org/10.1016/j.hlpt.2021.100549
17. Vargaftik, S., Eytan, D. and Rubinstein, B.I.P., 2019. A tree-based approach for multi-class anomaly detection. *arXiv preprint arXiv:1909.11877*. Available at: <https://arxiv.org/pdf/1909.11877> [Accessed 12 Aug. 2025].

**APPENDIX**

**APPENDIX A: FULL CLASSIFICATION REPORTS**

**Table A1: Random Forest**

Accuracy: 0.94

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-score** | **Support** |
| **Normal** | 0.94 | 1.00 | 0.97 | 2855 |
| **Data Alteration** | 1.00 | 0.99 | 1.00 | 184 |
| **Spoofing** | 0.91 | 0.18 | 0.30 | 225 |
| **Macro Avg** | 0.95 | 0.72 | 0.75 | 3264 |
| **Weighted Avg** | 0.94 | 0.94 | 0.92 | 3264 |

**Table A2: Random Forest with Threshold Tuning**

Accuracy: 0.93

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-score** | **Support** |
| **Normal** | 0.96 | 0.97 | 0.96 | 2855 |
| **Data Alteration** | 1.00 | 0.99 | 1.00 | 184 |
| **Spoofing** | 0.51 | 0.43 | 0.47 | 255 |
| **Macro Avg** | 0.82 | 0.80 | 0.81 | 3264 |
| **Weighted Avg** | 0.93 | 0.93 | 0.93 | 3264 |

**Table A3: Random Forest with Isolation Forest**

Accuracy: 0.94

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-score** | **Support** |
| **Normal** | 0.94 | 1.00 | 0.97 | 2855 |
| **Data Alteration** | 1.00 | 0.99 | 1.00 | 184 |
| **Spoofing** | 0.93 | 0.18 | 0.30 | 225 |
| **Macro Avg** | 0.96 | 0.72 | 0.75 | 3264 |
| **Weighted Avg** | 0.94 | 0.94 | 0.92 | 3264 |

**Table A4: Random Forest with SMOTE**

Accuracy: 0.93

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-score** | **Support** |
| **Normal** | 0.96 | 0.97 | 0.96 | 2855 |
| **Data Alteration** | 1.00 | 1.00 | 1.00 | 184 |
| **Spoofing** | 0.52 | 0.45 | 0.48 | 225 |
| **Macro Avg** | 0.82 | 0.81 | 0.81 | 3264 |
| **Weighted Avg** | 0.93 | 0.93 | 0.93 | 3264 |

**Table A5: XGBoost**

Accuracy: 0.96

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-score** | **Support** |
| **Normal** | 0.96 | 1.00 | 0.98 | 2855 |
| **Data Alteration** | 1.00 | 1.00 | 1.00 | 184 |
| **Spoofing** | 0.89 | 0.43 | 0.58 | 225 |
| **Macro Avg** | 0.95 | 0.81 | 0.85 | 3264 |
| **Weighted Avg** | 0.95 | 0.96 | 0.95 | 3264 |

**Table A6: XGBoost with Threshold Tuning**

Accuracy: 0.95

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-score** | **Support** |
| **Normal** | 0.97 | 0.98 | 0.97 | 2855 |
| **Data Alteration** | 1.00 | 1.00 | 1.00 | 184 |
| **Spoofing** | 0.66 | 0.56 | 0.60 | 225 |
| **Macro Avg** | 0.88 | 0.84 | 0.86 | 3264 |
| **Weighted Avg** | 0.95 | 0.95 | 0.95 | 3264 |

**APPENDIX B: ADDITIONAL RESULTS**

**Figure B1: Precision-Recall curve with AUC score for Random ForestA graph of a forest

AI-generated content may be incorrect.**

**Figure B2: Precision-Recall curve with AUC score for XGBoostA graph of a graph with lines and numbers

AI-generated content may be incorrect.**

**Table B1: Stratified K-Fold Cross-Validation for XGBoost**

|  |  |
| --- | --- |
| **Fold** | **Macro F1-Score** |
| **1** | 0.8452 |
| **2** | 0.8613 |
| **3** | 0.8606 |
| **4** | 0.8575 |
| **5** | 0.8372 |

Mean F1-score: 0.8524

Standard Deviation: 0.0096