Group 12 (Zach, S. Felix, K. Logan, L. Magan, G.)

CompSci 469

12/14/2024

Dr. Zhen

Final Report

Supervisory Control and Data Acquisition (SCADA) systems play a critical role in industries such as manufacturing, transportation, water management, renewable energy, oil and gas, and power distribution. These systems were first utilized in the 1960s to provide companies with a more efficient way to monitor and control their equipment. The introduction of microprocessors and programmable logic controllers (PLCs), helped to automate process monitoring. By the 1980s, advancements in personal computers (PCs) and Local Area Networking (LAN) allowed SCADA systems to integrate with other networks, facilitating communication between vendors and companies while enabling the connection of more devices (Johnson, M. 2019). Today, SCADA systems efficiently oversee and control operations, making global data access seamless and fast for individuals and organizations.

SCADA security from possible cyber threats is essential because, in turn, it would also protect the Industrial Control Systems (ICS). To attack any of the two systems could affect the economy, the environment, and / or the safety of people with the systerm. For example, an attacker in 2021 targeted the website for the Oldsmar water plant in Florida. They injected malicious code on the website to then collect a user’s CPU, browser, input methods, camera, accelerometer, microphone, time zone, locations, videos, and screen dimensions. This website attack became dubbed the “watering hole attack”. Not only did they collect data, but the same hacker was able to adjust the sodium hydroxide amounts in the water supply by performing an attack on the SCADA system of the water treatment plant. They increased the dosage so that it would be dangerous for a person to drink the water (Lakshmanan, R. 2021). SCADA systems are used for essential companies and infrastructures like the water plant. Accessing the system can be detrimental to not only a company’s finances but also to the health of the populace.

For our project, we have two main objectives. One is to develop a Support Vector Machine (SVM) based model for detecting anomalies and malicious events. The other is to evaluate the model’s performance on SCADA datasets. An SVM-based model uses a supervised learning algorithm to find the defining line between different classes of data. Most importantly, it identifies the best possible boundary to separate data. We are using this type of model to be able to detect attacks on a SCADA dataset.

One of the datasets used for this project was the gas pipeline dataset from Tommy Morris’ Industrial Control System Cyber Attack Datasets. This dataset included sensor readings and annotations that helped distinguish between normal operations and anomalies caused by either malicious activities or natural disturbances. Because SCADA systems play such a critical role, ensuring the dataset was clean and ready for analysis felt like an important step, and one we wanted to get right.

Morris identifies seven attack types within his datasets, which he outlines in Morris and Gao (2013). According to that paper, Naive Malicious Response Injection (NMRI) “lacks sophistication.” This attack essentially injects random values into a SCADA system, which although theoretically dangerous, is usually easy to detect and address.

Complex Malicious Response Injection (CMRI) builds upon NMRI, requiring knowledge of the SCADA system’s ordinary readings. By simulating trends in the data that could realistically occur, a CMRI attack can create a feedback loop from the SCADA system, potentially causing a serious failure. For instance, injecting data that simulates a decrease in gas pressure could prompt the system to respond by increasing gas pressure—even when the pressure never actually fell.

Response injection attacks (like NMRI and CMRI) falsify readings to trigger a response from the system, while command injection attacks directly send configuration or control commands. Malicious State Command Injection (MSCI) attacks send malicious commands to devices such as switches, actuators, and valves within a SCADA system, potentially destabilizing the system. Malicious Parameter Command Injection (MPCI) alters automation parameters, which can lead to failures based on the system’s response to ordinary readings. Malicious Function Code Injection (MFCI) involves injecting malicious code into a SCADA system to disrupt communication or extract sensitive information.

A Denial of Service (DOS) attack disables some portions of the SCADA system, leading to failures. These attacks range from disabling software on system endpoints to shutting off switches or valves manually. Lastly, reconnaissance attacks collect information about a SCADA system, including details about its network, control system, and device characteristics.

NMRI attacks are straightforward to detect since they inject values far outside of normal system parameters. Other attacks, however, require more nuanced detection methods. For example, abnormal deviations in data packet frequency (Fiah, 2023) or irregular system patterns (Altaha and Hong, 2022) are good indicators of malicious activity.

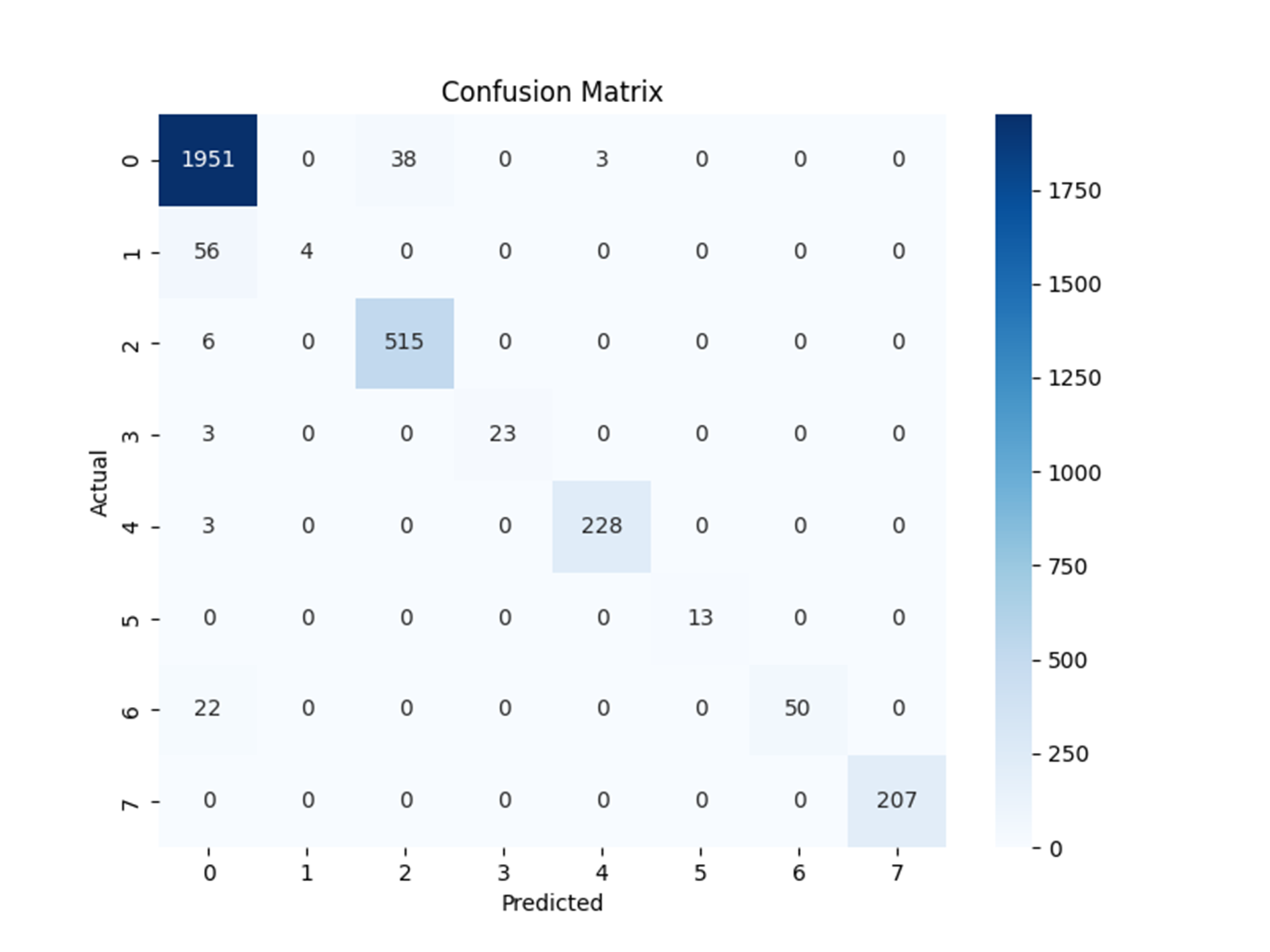
Currently, some of the most prevalent anomaly detection techniques are signature-based detection, anomaly-based detection, and hybridization of those methods. Signature-based detection excels at identifying known attacks. As explained by Hagar (2024), a signature-based Intrusion Detection System (IDS) identifies malicious activity by identifying known patterns attributed to those attack types. Signature-based identification is efficient and effective, but only against known attacks. It also requires frequent maintenance to account for newly identified attacks, and it tends to mark normal data as malicious (false positives).

Anomaly-based detection works by detecting any abnormalities in the data, which covers known and unknown attack types. While it isn’t as precise as signature-based detection, its ability to identify unknown attacks makes it necessary in most applications. A hybridization of the two allows an IDS to detect most types of attacks.

For the actual anomaly detection, we chose SVM. The reason behind this choice was simple: SVM is known for handling high-dimensional data well and for being resistant to overfitting. Both qualities are invaluable when dealing with SCADA data, which can be noisy and complex. At its core, SVM works by creating a hyperplane that separates data points from different classes while maximizing the margin between them. This approach helps reduce errors and ensures the model can handle unseen data effectively. As Mahmood and Hu (2016) note, SVM’s ability to construct such robust decision boundaries makes it particularly suitable for challenging datasets like those found in SCADA systems.

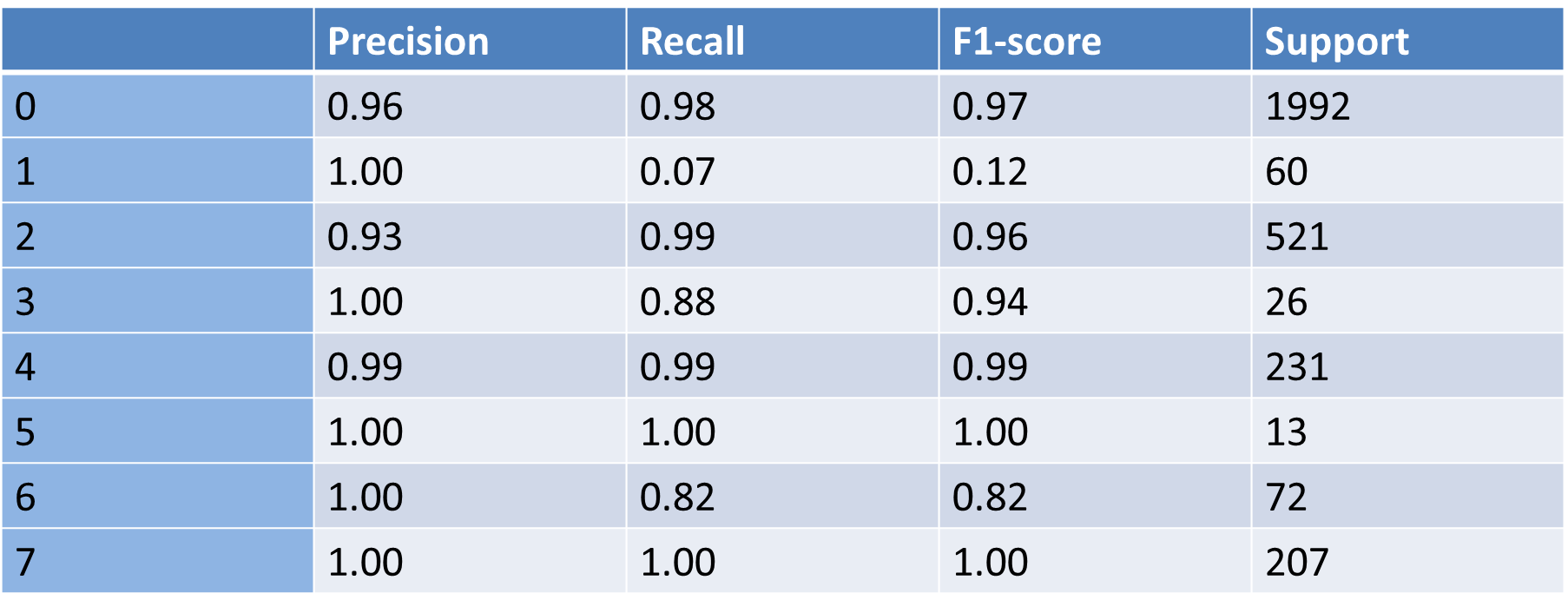
Machine learning methods such as SVM are critical for detecting anomalies in SCADA systems. SVM excels at identifying patterns in data and uses labeled datasets to draw clear boundaries between normal and anomalous data, ensuring high accuracy and reliability (Obonna et al., 2023). SVM is particularly effective in handling large, high-dimensional datasets, a common characteristic of SCADA systems. To further enhance its performance, preprocessing techniques such as Principal Component Analysis (PCA) address imbalances in datasets. These steps ensure that SVM can process SCADA data, which is typically noisy and unevenly distributed (Liu et al., 2022). By integrating SVM into SCADA systems, it becomes possible to detect sophisticated attacks, such as man-in-the-middle (MitM) and false data injection. SVM enhances security by monitoring data exchanges between sensors and controllers, ensuring stable and secure operations in critical infrastructure systems (Obonna et al., 2023; Liu et al., 2022).

The confusion matrix below illustrates the SVM model's initial performance across the different classes.

**Figure 1: Confusion Matrix for the Water Storage Tank Dataset**

We evaluated the model using several metrics: precision, recall, F1-score, accuracy, and Area Under the Curve (AUC). Precision-measured how well the model identified actual anomalies, while recall captured the proportion of anomalies detected out of all actual anomalies. The F1-score, as a balance of precision and recall, became a key indicator of the model’s overall performance. Accuracy gave us a broad view of correctness, and the AUC helped us understand how well the model handled the trade-off between true positives and false positives.

The results from the confusion matrix indicate high accuracy for Class 0 (non-anomalous data), with 1951 correct predictions out of 1992 instances. However, the model struggled with Class 1 (NMRI), achieving a recall of only 0.07, suggesting difficulty identifying anomalies in this class.



In addition to the gas pipeline dataset, we also utilized one other dataset to broaden the evaluation of the SVM model: 10% Random Sample of the Water Storage Tank Dataset:  
This dataset includes SCADA logs from a simulated water storage system and covers both normal operations and attack scenarios, such as data injection and control disruptions. Using a 10% random sample ensured the dataset was manageable while still representative of broader trends. This dataset offered insights into vulnerabilities in water management SCADA systems.

To ensure consistency, we applied the same preprocessing steps to both datasets:

* Outlier Removal: Values below the 1st percentile and above the 99th percentile were filtered out to reduce noise and extreme values.
* Feature Scaling: Numerical features were standardized to have a mean of zero and a standard deviation of one.
* Categorical Encoding: Categorical data was converted into numerical representations using one-hot encoding or label encoding.

These preprocessing steps ensured the datasets were cleaned, normalized, and ready for training. Applying the same pipeline to all three datasets allowed for fair comparisons when evaluating the SVM model.

A key part of making SVM effective is ensuring the data is properly prepared for training. Without the right preprocessing steps, the model’s performance could be compromised. To address this, we focused on data preprocessing techniques to improve model accuracy. To start, we removed outliers to maintain data integrity. Outliers, if left unaddressed, can cause skewed results and mislead the model, therefore, we decided to filter out values below the 1st percentile and above the 99th percentile. After that, we adjusted every numerical feature to have a mean of zero and a standard deviation of one, to ensure no one feature dominated the others during training. These adjustments might seem minor, but as Shang and Zeng (2016) point out, preprocessing steps like scaling and noise removal are game-changers for improving model accuracy in environments like SCADA.

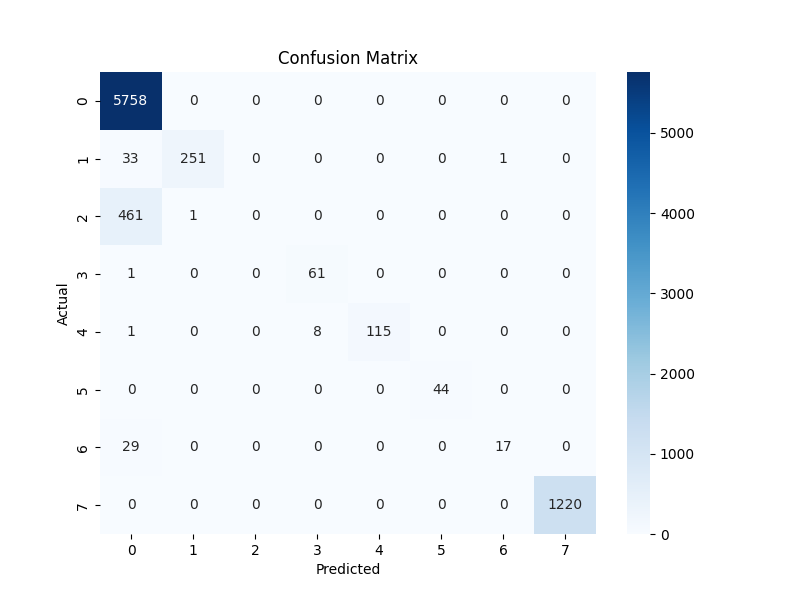
Another key step was encoding categorical data into numerical formats. Using one-hot encoding and label encoding, we converted non-numerical data into forms the SVM could process. The “Results” column, which identified whether a data point was normal or anomalous, was converted into integers to streamline the process. These transformations might sound tedious, but they were essential for making sure the data was not only ready but also optimized for machine learning.

To enhance SVM’s ability to deal with non-linear relationships, we applied an RBF (Radial Basis Function) kernel. This kernel projects the data into a higher-dimensional space, allowing for better separation of normal and anomalous data points. Tuning the hyperparameters—C (which controls regularization) and gamma (γ, which influences the kernel)—was crucial. We used cross-validation to find the sweet spot where the model wasn’t underfitting or overfitting. It’s worth mentioning that machine learning in SCADA systems, as Yasakethu and Jiang (2013) emphasize, thrives on adaptability, which is exactly what SVM with the RBF kernel offers.

Once the preprocessing was complete, we split the dataset into training and testing subsets 70% for training and 30% for testing. During training, we used Python’s scikit-learn library to implement the SVM model with the RBF kernel. After the training was finished, the model was saved as a .pkl file for easy reuse in evaluations. Testing the model involved using the processed testing data to make predictions, which were then compared against actual labels in the dataset. This step helped reveal the model’s strengths and weaknesses, giving us insights into how well it could generalize. Phillips, Gamess, and Krishnaprasad (2020) point out that testing on separate datasets is vital because it highlights the true potential and limitations of SCADA anomaly detection systems.

To dive deeper into the model's performance, we visualized the results using a confusion matrix. The confusion matrix highlights where the model performed well and where it struggled, providing more context behind the metrics like accuracy and F1-score. From the confusion matrix, Class 0 and Class 7 stood out as the best-performing classes. Class 0 had 5,758 correct predictions, and Class 7 had 1,220 correct predictions, both with little to no misclassifications. These results are encouraging since they show the SVM effectively recognized the majority of classes in the dataset.

However, the model had trouble with Class 2, where 461 out of 462 instances were misclassified as Class 0. This significant misclassification suggests that Class 2’s features either overlap too much with Class 0 or that the model didn’t have enough training data for Class 2. Similarly, Class 6 showed weaker performance, with only 17 correct predictions out of 46, indicating the model struggled to generalize for this minority class.

**Figure: Confusion Matrix**

Overall, the SVM performed impressively, achieving 96% accuracy on the initial testing set. Still, there was a slight bias, as the model occasionally classified normal data as anomalous. This reinforced just how challenging it can be to strike a balance between precision and recall, especially in imbalanced datasets. Even so, the high F1-score reassured us that the model was effective at identifying anomalies without creating too many false alarms.

Moreover, throughout the project, we encountered some challenges. Data quality issues such as missing labels and the presence of outliers meant we had to tread carefully during preprocessing. Balancing precision and recall also proved tricky, as improving one often came at the expense of the other. Additionally, the model’s reliance on labeled data highlighted a significant limitation. As Beaudet and Escudero (2022) discuss, SCADA systems often need methods that can adapt to changing conditions and unseen anomalies. This makes reliance on labeled training data somewhat restrictive. Generalizing the model to handle new types of anomalies was another concern, as it wasn’t clear how the SVM would perform outside the dataset we had.

Despite these challenges, the project was a success in demonstrating the potential of SVM for detecting SCADA anomalies. By carefully tuning the model and preparing the dataset, we were able to achieve strong performance. This project reaffirmed that, when paired with good preprocessing practices, SVM can serve as a reliable and robust tool for anomaly detection in critical systems.

In conclusion, the SVM model was effective, but not perfect, in detecting SCADA anomalies. Looking at the confusion matrix, we could see that the model struggled in some cases and excelled in others. In addition, we have obtained a 96% accuracy when comparing the actual and predicted values. Although the model wasn’t perfect when testing with our data set, it is the best model to use when dealing with high-dimensional datasets. However, its performance depends on the characteristics of the data and other issue-specific needs. SVMs are also robust to overfitting, effective with non-linear relationships, and are less impacted by outliers. They do have some limitations though such as scalability issues, parameter tuning complexity, limited multiclass performance, and lack of probabilistic outputs. Some machine learning model alternatives include random forests, gradient boosting, neural networks, and logistic regression. These alternatives may prove to be less effective than SVM because they may over-fit and require significant tuning because they are computationally excessive (Samia, N. Saha, S Haque, A. 2024).

To improve our SVM model, we could integrate with other algorithms like AdaBoost for better performance. AdaBoost, or adaptive boosting, is a machine learning algorithm that combines multiple weak classifiers to create a strong one. Essentially, it trains the weak classifiers on different subsets of the training data. During each session, the algorithm gives higher weights to the weak samples from the previous session. In other words, AdaBoost gives a way to reduce the error and increase the accuracy of any machine learning technique (Data Science Wizards, 2023). Another way to improve our SVM model is to test on more diverse datasets. Our project would have required much more time to be able to properly process, train, and evaluate our model on more diverse data sets. If we had, the work would have been rushed and incomplete, thus being of lower quality. It is recognized that we could use other diverse datasets from Kaggle or other sources to improve the model. Model’s improve from learning from data and using training algorithms. In more simple terms, the more the merrier.

In future works, we would explore unsupervised or semi-supervised techniques to detect anomalies in SCADA systems. Unsupervised techniques is a type of machine learning that learns from data without human supervision. They are given unlabeled data and allowed to train without explicit guidance or instruction. This includes exclusive clustering and hierarchical clustering. Semi-supervised techniques is a type of machine learning that uses both labeled and unlabeled data. This includes self-training, multi-view training, and label propagation. Another future exploration would be to do real time anomaly detection systems. This is a technology that continuously monitors incoming data streams and identifies unusual or outlier data points as they occur. This allows for immediate alerts and responses to potential issues within a system without needing to wait for data batch processing. SVM, which is what our model uses, is a type of anomaly detection but it's not in “real time”.

References

Yasakethu, S. L. P., & Jiang, J. (2013). Intrusion detection via machine learning for SCADA system protection. *1st International Symposium for ICS & SCADA Cyber Security Research 2013*.

Shang, W., & Zeng, P. (2016). Intrusion detection algorithm based on OCSVM in industrial control system. *Security and Communication Networks, 9*(10), 1040-1049.

Shon, T., & Moon, J. (2007). A hybrid machine learning approach to network anomaly detection. *Information Sciences, 177*(18), 3799-3821.

Phillips, B., Gamess, E., & Krishnaprasad, S. (2020). An evaluation of machine learning-based anomaly detection in a SCADA system using the Modbus protocol. *Proceedings of the 2020 ACM Southeast Conference*, 188-196.

Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications, 60*, 19-31.

Asplund, M., & Nadjm-Tehrani, S. (2016). Timing-based anomaly detection in SCADA networks. *International Journal of Embedded Systems, 8*(1), 14-23.

Beaudet, A., & Escudero, C. (2022). Malicious anomaly detection approaches robustness in manufacturing ICSs. *SN Computer Science, 3*(2), 145.

Thomas H. Morris and Wei Gao. Industrial Control System Cyber Attacks. 2013. DOI: 10.14236/ewic/ICSCSR2013.3

Fiah, Eric Kudjoe, "Anomaly detection in SCADA systems using machine learning" (2023). *Theses and Dissertations*. 5824. https://scholarsjunction.msstate.edu/td/5824

Altaha, M., & Hong, S. (2022). Anomaly Detection for SCADA System Security Based on Unsupervised Learning and Function Codes Analysis in the DNP3 Protocol. *Electronics*, *11*(14), 2184. <https://doi.org/10.3390/electronics11142184>

Hagar, S. (2024, February 16). *Intrusion detection systems: Signature vs. anomaly-based*. Retrieved December 15, 2024, from [https://cybersecuritycaucus.com](https://cybersecuritycaucus.com/)

Staff, “What is SCADA?” (2024). *SCADA International*. <https://scada-international.com/what-is-scada/>

Johnson, M. (2019, May 14). “The Birth and Development of SCADA.” *The Earth Awards* ,[theearthawards.org/a-brief-history-of-the-scada-system/#:~:text=SCADA%20systems%20were%20first%20utilized,equipment%2C%20especially%20over%20long%20distances](http://theearthawards.org/a-brief-history-of-the-scada-system/#:~:text=SCADA%20systems%20were%20first%20utilized,equipment%2C%20especially%20over%20long%20distances)

Lakshmanan, R. (2021, May 20.) “Watering hole attack was used to target Florida Water Utilities.” *The Hacker News*, <https://thehackernews.com/2021/05/watering-hole-attack-was-used-to-target.html>

Obonna, U. O., Opara, F. K., Mbaocha, C. C., Obichere, J.-K. C., Akwukwaegbu, I. O., Amaefule, M. M., & Nwakanma, C. I. (2023). “Detection of man-in-the-middle (MitM) cyber-attacks in oil and gas process control networks using machine learning algorithms.” *Future Internet,*

Liu, F., Zhang, S., Ma, W., & Qu, J. (2022). “Research on attack detection of cyber-physical systems based on improved support vector machine.” *Mathematics*,

Anomaly Detection via One Class SVM for Protection of SCADA Systems*Yasakethu, S. L. P., & Jiang, J. (2013). Anomaly detection via one class SVM for protection of SCADA systems. In 2013 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (pp. 82-88). IEEE.*

Analysis of SCADA Data Preprocessing Methods for Wind Power Farms*Zhang, Y., & Wang, J. (2022). Analysis of SCADA data preprocessing methods for wind power farms. In 2022 IEEE 4th International Conference on Power, Intelligent Computing and Systems (ICPICS) (pp. 1-5). IEEE.*

Anomaly Detection in SCADA Systems: A State Transition Modeling*Zhang, Y., & Wang, J. (2022). Anomaly detection in SCADA systems: A state transition modeling. IEEE Transactions on Industrial Informatics, 18(6), 4112-4121.*

Improving Anomaly Detection in SCADA Network Communication with Attribute Extension*Anwar, M., Lundberg, L., & Borg, A. (2022). Improving anomaly detection in SCADA network communication with attribute extension. Energy Informatics, 5, Article number: 69.*

A Review of Research Works on Supervised Learning Algorithms for SCADA Intrusion Detection*Almseidin, M., & Reed, M. J. (2021). A review of research works on supervised learning algorithms for SCADA intrusion detection. Sustainability, 13(17), 9597.*

Data Preprocessing Techniques*Zhang, Z., & Zhu, Q. (2018). Data preprocessing techniques. In Data Mining: Concepts and Techniques (pp. 83-124). Springer.*

An Efficient One-Class SVM for Anomaly Detection in the Internet of Things*Yang, K., Kpotufe, S., & Feamster, N. (2021). An efficient one-class SVM for anomaly detection in the Internet of Things. arXiv preprint arXiv:2104.11146.*

Machine Learning in Modern SCADA Systems: Opportunities and Challenges*Khan, R., & Salah, K. (2022). Machine learning in modern SCADA systems: Opportunities and challenges. IEEE Internet of Things Journal, 9(5), 3497-3512.*

Intrusion Detection in SCADA Systems Using Machine Learning Techniques*Pan, S., Morris, T. H., & Adhikari, U. (2015). Intrusion detection in SCADA systems using machine learning techniques. In 2015 7th International Symposium on Resilient Control Systems (ISRCS) (pp. 110-117). IEEE.*

Samia, N., Saha, S., Haque, A. (2024, December 1). “Predicting and Mitigating Cyber Threats Through Data Mining and Machine Learning”. *Science Direct*, <https://www.sciencedirect.com/science/article/pii/S0140366424002962#:~:text=They%20found%20that%20the%20Support,attackers%2C%20with%2065.2%25%20accuracy>.

Data Science Wizards (2023, July 7). “Understanding the AdaBoost Algorithm”. *Medium*, <https://medium.com/@datasciencewizards/understanding-the-adaboost-algorithm-2e9344d83d9b>