Journal of Advanced Artificial Intelligence
Volume 1 – No.7, June 2025

Network Slice Recognition With Explainable Machine Learning

Debmalya Ray
Data Science Dept., International Institute of Information Technology,
Bangalore, India

ABSTRACT
Fifth-generation (5G) and beyond networks various emerging applications such as AR/VR/XR, e-Health, live video streaming, and automated vehicles are expected to have diverse and strict quality of service (QoS) requirements.

Network slicing will prioritize virtualized and dedicated logical networks over common physical infrastructure and encourage flexible and scalable networks. It enables the creation of multiple virtual networks, each operating on a shared physical infrastructure, to meet various application requirements. This approach allows for customized network environments tailored to specific needs, such as different Quality of Service (QoS) levels, security protocols, and performance characteristics.

This paper also envisages the usage of Explainable AI which plays a significant role in making machine learning models more transparent and understandable. In the context of network slicing and other telecom applications which enhance the interpretability and trustworthiness of machine learning models. This is essential for effective decision making and maintaining high service standards.

Keywords
Supervised Learning, Feature Engineering, Feature Extraction, Python, Network Slicing, Telecom, Classification Problems, Explainable Artificial Intelligence.

1. INTRODUCTION
The telecommunications industry involves the provision of communication services over networks, including mobile and fixed-line communications. In the context of 5G and beyond, network slicing and machine learning are crucial for managing complex and dynamic network environments.

Applications:

Quality of Service (QoS) Management: Machine learning models help ensure that each network slice meets its specific QoS requirements.

Security and Anomaly Detection: Enhancing network security by detecting and mitigating threats within network slices.

Our motivation is to take one such task to build a machine learning model that will be able to proactively detect and eliminate threats based on incoming connections thereby selecting the most appropriate network slice, even in case of a network failure.

Dataset Description LTE/5g - User Equipment categories or classes to define the performance specifications

Packet Loss Rate - number of packets not received divided by the total number of packets sent.

Packet Delay - The time for a packet to be received.

Slice type - network configuration that allows multiple networks (virtualized and independent)

GBR - Guaranteed Bit Rate Healthcare - Usage in Healthcare (1 or 0)

Industry 4.0 - Usage in Digital Enterprises (1 or 0)

IoT Devices - Usage Public Safety – Usage for public welfare and safety purposes (1 or 0)

Smart City & Home - usage in daily household chores

Smart Transportation - usage in public transportation

Smartphone - whether used for smartphone cellular data

2. LITERATURE REVIEWS
Fundamentals of Network Slicing:

Overview and Architecture: Early research often focuses on the fundamental concepts of network slicing, including its architecture and benefits. Studies such as those by **Xia et al. (2018)** and **Zhang et al. (2019)** outline the basic principles of network slicing in 5G networks and describe how slices can be customized to meet specific application requirements.

Challenges and Requirements: Papers like **Fitzek and Reisslein (2020)** discuss the challenges in implementing network slicing, including the need for efficient resource management and the complexity of slicing different network functionalities.

Overview of Network Slicing and ML Integration:

Role of ML in Network Slicing: Machine learning (ML) plays a crucial role in the management and optimization of network slices, given the dynamic and complex nature of 5G networks. ML techniques can be applied to predict network demand, optimize resource allocation, and automate the orchestration of network functions. Reviews by **Bertocchi et al. (2019)** and **Zhang et al. (2020)** discuss how ML enhances the efficiency of network slicing by enabling adaptive and real-time decision-making processes. The integration of ML helps in dealing with the uncertainties and variability inherent in network traffic, ensuring that each slice meets its specific performance requirements.

Dynamic Network Selection:

Context-Aware Network Selection: Reviews by **Garcia and Lloret (2021)** discuss how ML models are utilized for dynamic network selection, where the model predicts the optimal network type for a device based on current environmental conditions and user needs. This approach often involves reinforcement learning, where the model continuously learns and adapts based on feedback from its environment.

Multi-Criteria Decision Making: Some literature also delves into multi-criteria decision-making algorithms combined with ML for network selection, where multiple factors such as signal strength, user mobility, and application type are considered simultaneously to make network type recognition decisions.

Security and Anomaly Detection in Network Recognition:

Anomaly Detection: Literature reviews such as those by **Zhang et al. (2018)** explore how ML techniques are applied to detect anomalies in network type recognition. These anomalies could indicate security breaches or malfunctions in the network. ML models are trained to recognize the typical behavior of network types, enabling them to flag unusual activity that might suggest an attack or other issues.

Intrusion Detection Systems: Other reviews, like those by **Patel et al. (2019),** focus on how ML models are integrated into intrusion detection systems (IDS) to identify unauthorized network types or access points, enhancing the security of network operations.

3. SIGNIFICANCE OF THE STUDY

The significance of studying network slice recognition using machine learning (ML) techniques can be summarized as follows:

Optimization of Network Resources:

Efficient Resource Allocation: ML techniques enable more accurate recognition and management of network slices, ensuring that resources like bandwidth, processing power, and storage are optimally allocated. This leads to better utilization of network infrastructure and minimizes wastage.

Support for Diverse Applications:

Customized Service Delivery: Different applications have varying requirements for Quality of Service (QoS), latency, and bandwidth. ML-driven slice recognition allows networks to cater to these diverse needs by dynamically creating and managing customized slices, ensuring each application receives the necessary resources.

Cost Efficiency:

Reduced Operational Costs: Automating network slice recognition with ML reduces the need for manual intervention, leading to lower operational costs and reduced human error.

Infrastructure Savings: Optimized resource allocation through ML reduces the need for additional infrastructure investments, making network management more cost-effective.

Facilitation of 5G and Future Networks:

Key to 5G Success: Network slicing is a fundamental component of 5G networks, enabling support for a wide range of services. ML enhances the efficiency and scalability of network slicing, making it crucial for the success of 5G and future network technologies.

Future-Ready Networks: ML-based slice recognition prepares networks to handle the growing complexity and demands of next-generation applications and services.

Innovation and Research Advancement:

Pioneering New Solutions: The study of ML in network slice recognition drives innovation, pushing the boundaries of what is possible in telecommunications and contributing to advancements in both telecommunications and machine learning fields.

4. METHODS

Network slice recognition using machine learning (ML) techniques involves identifying and managing different network slices to optimize performance and resource allocation. Here's how ML can be applied to network slice recognition.

Data Collection:

Traffic Data: Collect data related to network traffic, including metrics such as bandwidth, latency, packet loss, jitter, and user mobility.

Feature Extraction: Identify key features from the collected data that can help distinguish between different network slices**.**

Data Normalization: Normalize the data to ensure consistency in scale, which improves the performance of ML models.

Feature Engineering:

Feature Extraction: Extracting relevant features from the collected data that can help distinguish between different network slices.

Normalization: Normalizing the features to ensure consistent scale and improve the performance of ML models.

Anomaly Detection for Slice Integrity:

Outlier Detection: Implement anomaly detection methods to monitor network slices for unusual behaviour or performance degradation. Techniques like Isolation Forests and One-Class SVMs can identify traffic patterns that deviate from expected norms.

Proactive Management: Detecting anomalies can trigger proactive adjustments to slice configurations, ensuring that performance and security standards are maintained.

Model Training:

Training Process: Train the model on the labelled training data, using an iterative process to minimize the error (e.g., loss function) between the model's predictions and the actual labels.

Hyperparameter Tuning: Optimize the model's hyperparameters (e.g., learning rate, regularization parameters, tree depth) using techniques such as grid search or random search to improve performance.

Cross-Validation: Employ cross-validation (e.g., k-fold cross-validation) to assess the model's performance across different subsets of the training data, reducing the risk of overfitting.

Model Evaluation:

Accuracy: Measuring the accuracy of the model in correctly recognizing and classifying network slices.

Precision and Recall: Evaluating the model's precision and recall to ensure it correctly identifies slices and minimizes false positives and negatives.

F1 Score: Using the F1 score as a balance between precision and recall to assess overall model performance.

Confusion Matrix: Helps visualize performance with counts of true positives, false positives, true negatives, and false negatives.

ROC Curve and AUC: Useful for understanding trade-offs between true positive and false positive rates.

5. PLOTS AND FIGURES

Figure 1 shows the sizes of the trains and tests used for model training and evaluation.

```
print(train_dataset.shape, test_dataset.shape)

(31583, 17) (31584, 16)
```

Figure 1 Train and Test Datasets

Figure 2 is vital for preparing input variables for ML models in network slicing. Techniques such as correlation matrices, heatmaps, and scatter plots help in understanding the relationships between features. Effective feature selection and engineering, based on correlation analysis, enhance model performance and interpretability. Tools and libraries like Pandas, Seaborn, and Scikit-Learn facilitate the analysis and processing of feature correlations.

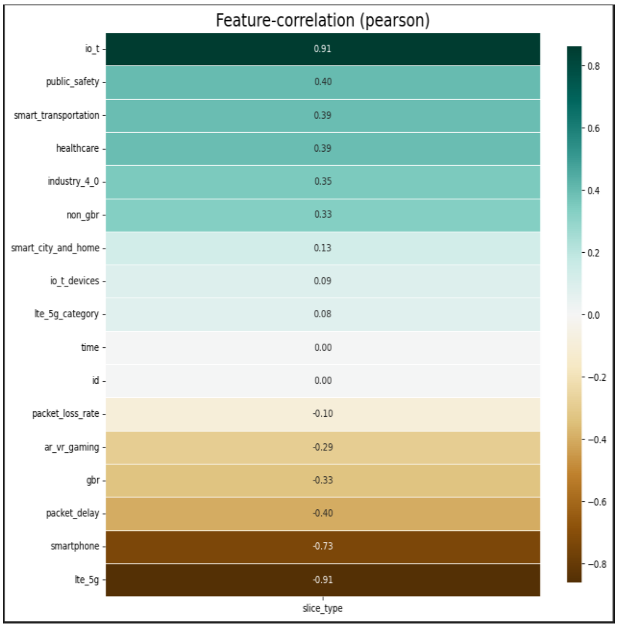

Figure 2 Correlation Among Variables

Figure 3 explains the feature selection techniques. Feature importance in the context of network slicing recognition using machine learning (ML) techniques refers to identifying which features (input variables) most significantly impact the model's predictions. Network slicing is a method in 5G networks that allows multiple virtual networks to be created on a shared physical infrastructure, each tailored to specific service requirements.

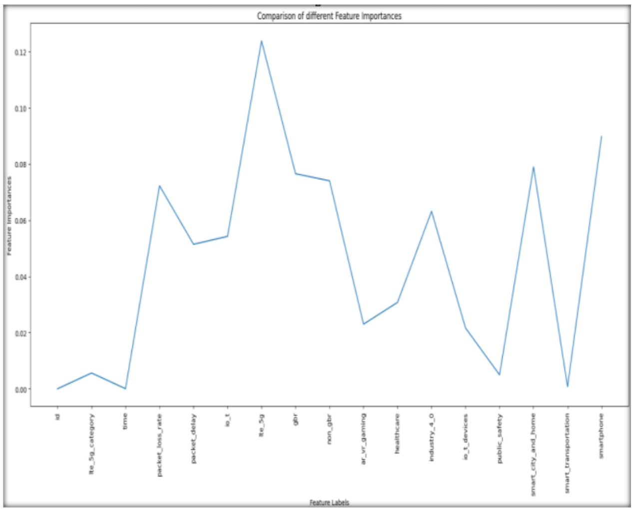

Figure 3 Feature Selection Techniques

Figure 4 explains the feature correlation mapping. It can be a valuable tool in network slicing when using machine learning (ML). It helps visualize and understand the relationships between various network features, which can significantly impact the performance and accuracy of ML models designed for network slicing.

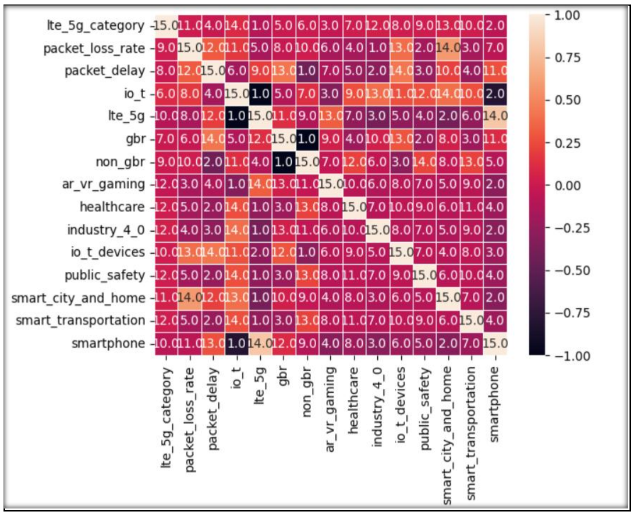

Figure 4 Feature Correlation Mapping

Figure 5 explains details of model selection techniques.It for network slicing recognition using machine learning techniques involves choosing a model that can accurately classify or predict the appropriate network slices based on real-time traffic and network conditions. The choice of model depends on the specific task (classification, clustering, reinforcement learning), data characteristics, and operational constraints like real-time processing and scalability. The selected model should be capable of handling the complexity and dynamism of modern networks, ensuring efficient and accurate slice recognition for optimal network performance.

id	pipeline_name	search_order	ranking_score	mean_cv_score	standard_deviation_cv_score	percent_better_than_baseline	high_variance_cv
1	Random Forest Classifier w/ Label Encoder + Im...	1	4.440892e-16	4.440892e-16	0.000000e+00	100.000000	False
3	Extra Trees Classifier w/ Label Encoder + Impu...	3	4.440892e-16	4.440892e-16	0.000000e+00	100.000000	False
2	LightGBM Classifier w/ Label Encoder + Imputer...	2	1.137570e-06	1.137570e-06	2.102637e-11	99.999993	False
5	XGBoost Classifier w/ Label Encoder + Imputer ...	5	1.418487e-04	1.418487e-04	4.158776e-07	99.999153	False
4	Elastic Net Classifier w/ Label Encoder + Impu...	4	3.454684e-04	3.454684e-04	5.857096e-06	99.997936	False
6	Logistic Regression Classifier w/ Label Encode...	6	3.973698e-04	3.973698e-04	7.944801e-06	99.997626	False
0	Mode Baseline Multiclass Classification Pipeline	0	1.674177e+01	1.674177e+01	2.945292e-03	0.000000	False

Figure 5 Model Leadership Board

Figure 6 displays the best diagram created for data transformation and it's classification during pipeline creation.

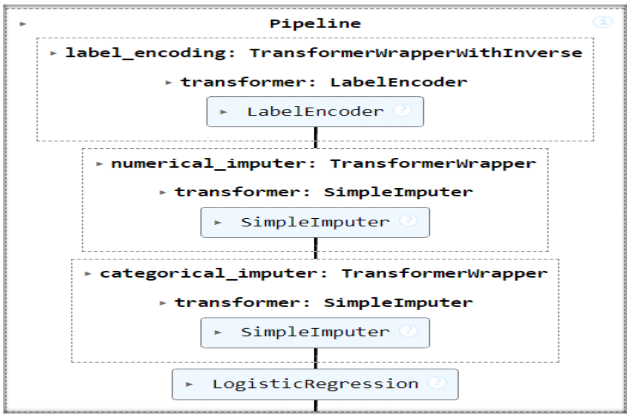

Figure 6 Best Pipeline Created

6. RESULTS

The final results predicted from the dataset using the best pipeline algorithm is mentioned as follows:

Table 1 : Final Results

SLICE TYPE	COUNT
Slice 1	9839
Slice 2	4294
Slice 3	4240

7. COMPARATIVE ANALYSIS – MODEL

In the context of network slicing recognition using machine learning, before selecting the best model, it's common to perform a comparative analysis using different algorithms. The comparative analysis involves evaluating multiple models based on a set of predefined metrics, often referred to as foundation metrics, which are critical to assessing model performance in a consistent and systematic way.

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
lr	Logistic Regression	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	5.1580
nb	Naive Bayes	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.1220
dt	Decision Tree Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.2590
ridge	Ridge Classifier	1.0000	0.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.0640
rf	Random Forest Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.7950
ada	Ada Boost Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.8710
gbc	Gradient Boosting Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	6.0580
et	Extra Trees Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.7450
xgboost	Extreme Gradient Boosting	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.5710
lightgbm	Light Gradient Boosting Machine	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.5260
catboost	CatBoost Classifier	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	15.3480
lda	Linear Discriminant Analysis	0.8668	0.9690	0.8668	0.8810	0.8649	0.7700	0.7811	0.0840
knn	K Neighbors Classifier	0.7858	0.9102	0.7858	0.7855	0.7856	0.6472	0.6473	1.2610
qda	Quadratic Discriminant Analysis	0.5319	0.0000	0.5319	0.2829	0.3694	0.0000	0.0000	0.0680
dummy	Dummy Classifier	0.5319	0.5000	0.5319	0.2829	0.3694	0.0000	0.0000	0.0580
svm	SVM - Linear Kernel	0.5256	0.0000	0.5256	0.6419	0.4228	0.2469	0.3203	1.4640

Figure 7: Comparative Analysis of Model Selected

8. PLOTS DERIVED FROM METRICS

A ROC curve is a graphical representation used to evaluate the performance of a binary classification model. It plots the True Positive Rate (TPR) against the False Positive Rate (FPR) across different threshold values.

Here's a quick breakdown of the terms:

True Positive Rate (TPR): Also known as Sensitivity or Recall, this is the proportion of actual positives that are correctly identified by the model.

True Positive Rate =

$$\frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}.$$

False Positive Rate (FPR): This is the proportion of actual negatives that are incorrectly classified as positives by the model.

False Positive Rate =

$$\frac{\text{False Positives}}{\text{False Positives} + \text{True Negatives}}.$$

Confusion Matrix:

A Confusion Matrix is a table used to evaluate the performance of classification model.

It provides insight into how well your model is performing by comparing the actual labels with the predicted labels. For a multi-class classification problem like network slice classification, the following metrics can be derived:

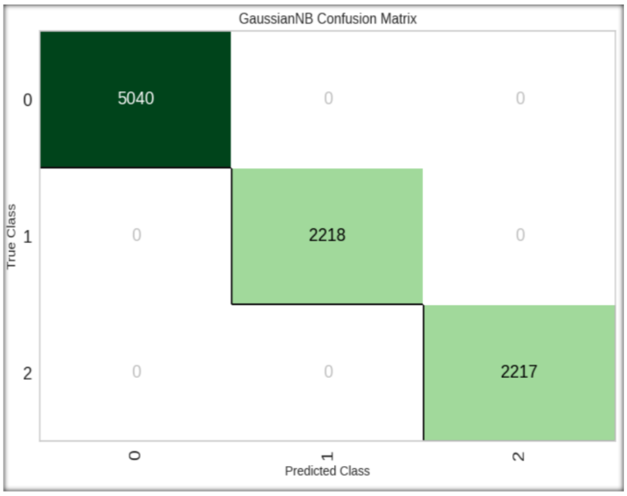

Figure 8: Confusion Matrix for Multi-Class Classification Problem

Variable Importance using XAI – Explainable AI:

Explainable AI (XAI) refers to techniques and methods that make the behavior and decisions of machine learning models understandable to humans. When dealing with complex models like neural networks, decision trees, or ensemble methods, it becomes crucial to explain why a model made a certain prediction. One of the key aspects of XAI is understanding variable (or feature) importance, which helps in interpreting the impact of each input feature on the model's predictions.

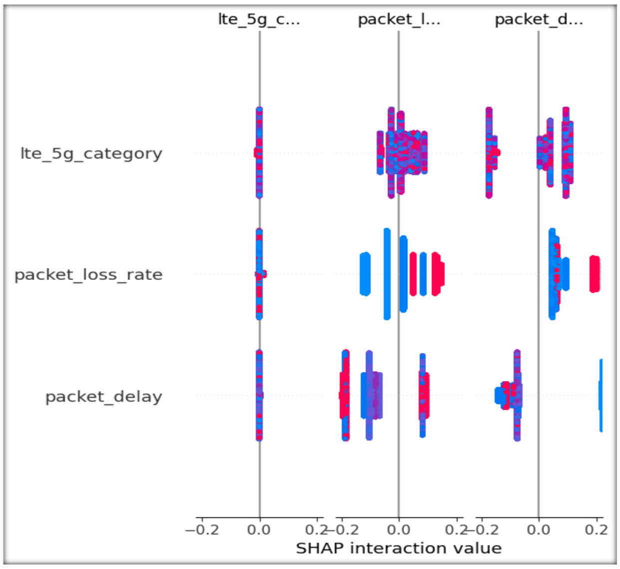

Figure 9: Variable Importance using SHAP XAI

The waterfall plot provides a detailed view of how the SHAP values (representing the contribution of each feature) combine to move from the base value (expected model output) to the final predicted value.

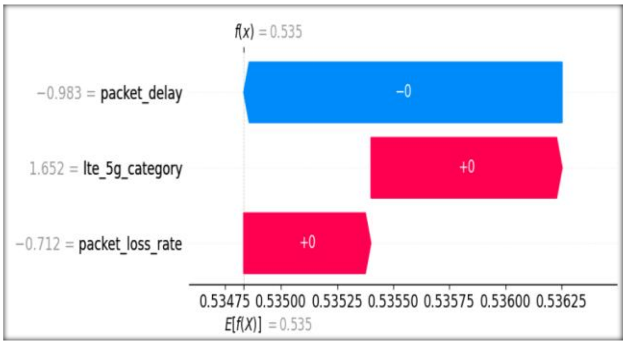

Figure 10: Waterfall Model Using SHAP

Interpretation of model using LIME for recognizing Network Slicing:

Using LIME (Local Interpretable Model-agnostic Explanations) to explain machine learning models for network slicing recognition involves generating explanations for the predictions made by a model that classifies network traffic into different slices.

In a network slicing environment, different types of network traffic are allocated to different slices based on their requirements. For example:

- **Slice 1:** High bandwidth (e.g., video streaming)
- **Slice 2:** Low latency (e.g., gaming)
- **Slice 3:** Massive IoT (e.g., smart sensors)

A machine learning model might be trained to classify incoming traffic into one of these slices based on features like bandwidth usage, latency, packet size, etc.

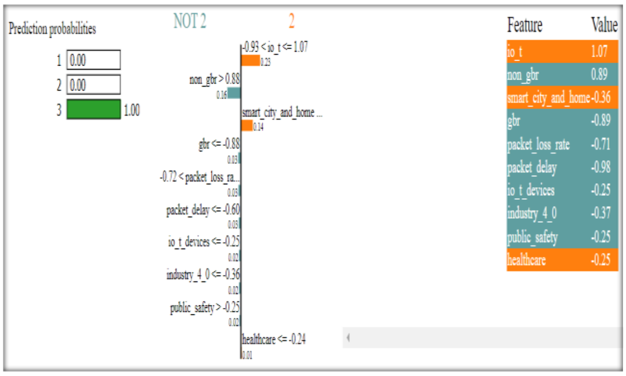

Figure 11: Recognizing Network Slicing Using LIME

9. CONCLUSIONS

We focus on the applications of machine learning (ML) techniques in network slicing. We begin by providing background information on network slicing to establish a foundational understanding. Following this, we offer an overview of some common ML techniques utilized in network slicing.

We then review the literature on this topic, organizing the contributions into three main categories: traffic forecasting, admission control, and resource allocation. For each category, we highlight key lessons learned from the research. Finally, we discuss some open challenges in the field and propose potential solutions that could be explored in future work.

1. Enhancing Trust and Transparency:

Interpretability of Decisions: Explainable machine learning (XML) models in network slice recognition provide transparency in decision-making processes. This transparency is crucial for network operators to understand why a particular slice was classified or managed in a specific way, building trust in the system's outputs.

Regulatory Compliance: XML helps ensure that network slice management aligns with regulatory requirements, particularly in industries like telecommunications where accountability and compliance are paramount. By making the decision-making process more transparent, XML aids in meeting legal and ethical standards.

2. Balancing Complexity and Explainability

Trade-offs in Model Selection: While complex deep learning models often yield higher accuracy in network slice recognition, they tend to be less interpretable. XML approaches seek to balance this by using models or techniques that provide a reasonable trade-off between accuracy and explainability, such as decision trees, rule-based systems, or interpretable neural networks.

Simplified Model Interpretations: Techniques like SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-agnostic Explanations) are employed to interpret complex models. These methods allow for understanding which features influenced specific decisions, thereby making even black-box models more interpretable.

10. ABBREVIATIONS

The following abbreviations are used in this manuscript:

For example:

ML: Machine Learning

CV: Computer Vision

SL: Supervised Learning

RF: Random Forest

DT: Decision Tree

DL: Deep Learning

SVM: Support Vector Machine

XAI: Explainable Artificial Intelligence

XML: Extensible Markup Language

SHAP: Shapley Additive Explanations

LIME: Local Interpretable Model-Agnostic Explanations

QOS: Quality of service

11. AUTHORS CONTRIBUTIONS

All authors take part in the discussion of the work described in this paper. They read and approved the final manuscript.

12. REFERENCES

[1] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca and J. Folgueira, "Network Slicing for 5G with SDN/NFV: Concepts, Architectures, and Challenges," in IEEE Communications Magazine, vol. 55, no. 5, pp. 80-87, May 2017, doi: 10.1109/MCOM.2017.1600935

[2] Zhang, Q.; Liu, F.; Zeng, C. Online Adaptive Interference-Aware VNF Deployment and Migration for 5G Network Slice. IEEE/ACM Trans. Netw. 2021, 29, 2115–2128. [CrossRef]

[3] Zhang, H.; Liu, N.; Chu, X.; Long, K.; Aghvami, A.; Leung, V. Network Slicing Based 5G and Future Mobile Networks: Mobility. In Resource Management, and Challenges; IEEE Communications Magazine: 2017. Available online: https://ieeexplore.ieee.org/ abstract/document/8004168

(accessed on 17 March 2023).

[4] Barakabitze, A.A.; Ahmad, A.; Mijumbi, R.; Hines, A. 5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges. Comput. Netw. 2020, 167, 106984. [CrossRef]

[5] Babbar, H.; Rani, S.; AlZubi, A.A.; Singh, A.; Nasser, N.; Ali, A. Role of Network Slicing in Software Defined Networking for 5G: Use Cases and Future Directions. IEEE Wirel. Commun. 2022, 29, 112–118. [CrossRef]

[6] A. Thantharate, R. Paropkari, V. Walunj and C. Beard, "DeepSlice: A Deep Learning Approach towards an Efficient and Reliable Network Slicing in 5G Networks," *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, New York, NY, USA, 2019, pp. 0762-0767, doi: 10.1109/UEMCON47517.2019.8993066

[7] Hurtado Sánchez, Johanna Andrea, Katherine Casilimas, and Oscar Mauricio Caicedo Rendon. 2022. "Deep Reinforcement Learning for Resource Management on Network Slicing: A Survey" *Sensors* 22, no. 8: 3031. https://doi.org/10.3390/s22083031

[8] Domeke, Afra, Bruno Cimoli, and Idelfonso Tafur Monroy. 2022. "Integration of Network Slicing and Machine Learning into Edge Networks for Low-Latency Services in 5G and beyond Systems" *Applied Sciences* 12, no. 13: 6617. https://doi.org/10.3390/app12136617

[9] Hurtado Sánchez, Johanna Andrea, Katherine Casilimas, and Oscar Mauricio Caicedo Rendon. 2022. "Deep Reinforcement Learning for Resource Management on Network Slicing: A Survey" *Sensors* 22, no. 8: 3031. https://doi.org/10.3390/s22083031

[10] Venkatapathy, Sujitha, Thiruvenkadam Srinivasan, Han-Gue Jo, and In-Ho Ra. 2023. "An E2E Network Slicing Framework for Slice Creation and Deployment Using Machine Learning" Sensors 23, no. 23: 9608. https://doi.org/10.3390/s23239608

[11] Mlika, Z., Cherkaoui, S. Network slicing for vehicular communications: a multi-agent deep reinforcement learning approach. *Ann. Telecommun.* **76**, 665–683 (2021). https://doi.org/10.1007/s12243-021-00872-w

[12] Abood, M. S., Wang, H., He, D., Kang, Z., & Kawoya, A. (2023). Intelligent Network Slicing in V2X Networks – A Comprehensive Review. *Journal of Artificial Intelligence and Technology*, *3*(2), 75–84. https://doi.org/10.37965/jait.2023.0208

[13] Zahoor, Sumbal, Ishtiaq Ahmad, Mohamed Tahar Ben Othman, Ali Mamoon, Ateeq Ur Rehman, Muhammad Shafiq, and Habib Hamam. 2022. "Comprehensive Analysis of Network Slicing for the Developing Commercial Needs and Networking Challenges" *Sensors* 22, no. 17: 6623. https://doi.org/10.3390/s22176623

[14] Thantharate, Anurag, Ankita Vijay Tondwalkar, Cory Beard, and Andres Kwasinski. 2022. "ECO6G: Energy and Cost Analysis for Network Slicing Deployment in Beyond 5G Networks" *Sensors* 22, no. 22: 8614. https://doi.org/10.3390/s22228614

[15] Rakkiannan, T., Ekambaram, G., Palanisamy, N. *et al.* An Automated Network Slicing at Edge with Software Defined Networking and Network Function Virtualization: A Federated Learning Approach. *Wireless Pers Commun* **131**, 639–658 (2023). https://doi.org/10.1007/s11277-023-10450-z.

[16] Abbas, Khizar, Muhammad Afaq, Talha Ahmed Khan, Adeel Rafiq, and Wang-Cheol Song. 2020. "Slicing the Core Network and Radio Access Network Domains through Intent-Based Networking for 5G Networks" *Electronics* 9,no.10:1710.https://doi.org/10.3390 /electronics9101710