CLASSIFYING OVER-THE-TOP NETWORK TRAFFIC USING DEEP LEARNING ALGORITHMS WITH DESIGN SCIENCE RESEARCH METHODOLOGY IMPLEMENTATION

Faradias Izza Azzahra Faisal[1], Armin Lawi[2], Eliyah Acantha Manapa Sampetoding*[1]
[1]Information Systems, Department of Mathematics, Faculty of Mathematics and Natural Sciences, Hasanuddin University, Jl. Perintis Kemerdekaan Km. 10, Makassar 90245, Indonesia
[2]Bacharuddin Jusuf Habibie Institute of Technology, Jl. Balaikota No. 1, Parepare 91122, Indonesia
Email: izzazzahraa@gmail.com, armin@unhas.ac.id, eliyahacantha@unhas.ac.id

Abstract

Classifying network traffic is the foundational step in analyzing diverse applications reliant on network infrastructure, particularly focusing on identifying Over-The-Top (OTT) application traffic using encryption. This methodology empowers Internet service providers and network operators to manage Quality of Service (QoS) performance effectively. Nonetheless, widespread encryption protocols have rendered traditional traffic identification obsolete. Despite limited work in this area, deep learning algorithms are expected to provide a practical solution. This paper introduces a framework outlining the construction of a classifier architecture through the Design Science Research Methodology (DSRM), suitable for producing information system artifacts. The classifier model is built upon deep learning algorithms—CNN, LSTM, and Bi-LSTM. Applying the DSRM approach to deploy the OTT classifier incorporates a deep learning model, offering performance assessment in terms of accuracy, recall, precision, f1-score, and the AUC-ROC curve. The evaluation results of the three models demonstrated strong performance, with accuracy values ranging from 0.83 to 0.96 on the test data. Specifically, the LSTM model show better performance in classifying OTT applications network traffic, achieving an accuracy of 0.96 and an f1-score of 0.95, surpassing the Bi-LSTM and CNN models.

Keywords: *DSRM, Over The Top, Network Application, Deep Learning Algorithm, Traffic Classification*

**Corresponding Author*

1. INRODUCTION

Design Science Research Methodology (DSRM) is a complementary part of information systems research, which involves the development and evaluation of information technology artefacts to solve identified problems within an organization [1]. Design Science is considered very important and oriented toward successfully creating a technological artefact [2]. Network traffic classification is the process of identifying each flow of network traffic data into specific applications based on its features [3]. This process is crucial for various fields, especially telecommunications and network security. With accurate traffic classification, multiple activities related to network services, such as monitoring, control, and optimization, can be carried out to improve network quality and security[4]. The evolution of cellular communication networks, from 1G to 5G, has encourage the growth of online media services like voice calls, instant messaging, and browsing. This advancement, alongside the birth of over-the-top (OTT) services, has significantly transformed telecommunications.

OTT is a media service in general that is served directly to users by relying on the internet network. OTT is a platform that facilitates service providers to offer video, audio, and other media over an IP network without any technology collaboration with network operators[5]. On the other hand, the task of accurately classifying network traffic is complicated by the increasing use of encrypted protocols, which are becoming standard practice in network security today[6].

This study responds to an urgent need within the field of network traffic classification, particularly in the context of optimizing network performance management for OTT services. With the rapid increase of encrypted protocols, conventional methods have become increasingly ineffective in accurately identifying traffic flows, posing significant challenges to network management and security. This lack of accurate traffic classification not only hinders essential network management functions such as monitoring,

control, and optimization, but also endangers network integrity.

While the focus is primarily on network performance management, it is essential to acknowledge the implications for network security. Without effective traffic classification, networks remain vulnerable to cyber threats such as unauthorized access, data breaches, and denial-of-service attacks. Therefore, there is a pressing need for innovative solutions that can effectively classify encrypted network traffic and mitigate the associated risks[7].

This study employs the Design Science Research Methodology to produce artefacts through research methods and analysis of network traffic data using deep learning algorithms, namely CNN, LSTM, and Bi-LSTM[8]. Deep learning algorithms offer promising alternatives to conventional methods, presenting opportunities to accurately identify encrypted network traffic flows amidst the challenges posed by the increasing use of encrypted protocols[9]. Furthermore, DSRM provides a structured framework for the development and evaluation of technological artefacts, ensuring that the proposed solutions are tailored to address the specific challenges encountered in network traffic classification[1]. The integration of deep learning algorithms and DSRM methodology reflects a collaborative effort to tackle the urgent need for innovative solutions in network traffic classification, particularly in optimizing network performance management for emerging OTT services.

2. LITERATURE REVIEW

2.1. Design Science Research

In the Information Systems field, Design Science Research is a widely accepted approach to conducting research. The literature on this method has increased significantly. DSRM involves the creation and innovation of new artefacts to solve specific organizational problems. Instead of simply explaining or understanding the current situation, this approach focuses on creating and evaluating IT artefacts that can create new and better solutions [10]. According to March and Smith [11], design science consists of two processes: build and evaluate. Build is the process of constructing an artifact for a specific purpose, while evaluate is determining how well an artifact is performing. Artefacts from design science consist of four types: construct, model, method, and instantiation. Creative innovation and rigorous evaluation of new artefacts significantly impact the success of the knowledge base in design science [1].

Peffers *et al.* [2] proposed a six-step design science research methodology: problem identification and motivation, setting targets for solutions, design and development, demonstration, evaluation, and communication, as shown in Figure 1.

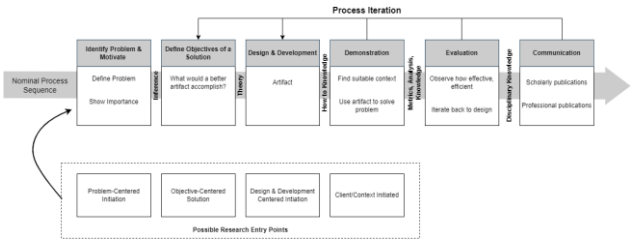


Figure 1. Design Science Research Methodology from Peffers *et al.,*[2]

Although these steps are designed to be followed sequentially, the researcher can follow each step in various ways. The first step of DSRM is problem identification. However, revisiting the problem referred to in the next DSRM step is essential while continuously improving the understanding of the problem[9]. Once a problem has been identified, the next step is to set performance goals for the solution. These objectives must be derived logically from the problem definition. Design and development are stages of artefact development. This artefact can be any design object that incorporates research contributions into its design, followed by defining the desired function and architecture of the artefact [2]. Once an artefact is built, the researcher can understand the performance and phenomena associated with its use to gain insight into the research problem. This approach is equivalent to the fourth stage, namely demonstration [13]. The resources required for the demonstration include practical knowledge of using artefacts to solve problems. The observation and measurement stage is carried out to determine how well the artefacts' performance supports the problem, in which knowledge of relevant measuring instruments and analytical techniques is needed at this stage. Finally, communication is required to disseminate the resulting knowledge in the form of designed artefacts, their uses, uniqueness, design rigor, and effectiveness to researchers and other relevant audiences, including professional practitioners, if needed[2].

2.2. Network Traffic Classification

Network traffic classification is critical for various fields, especially internet service providers[14]. This

task is used to identify the application flow in a network traffic. Network traffic classification is the first step to analysing and identifying various application types [15]. With this technique, internet service providers can manage the performance and quality of the network and internet services as a whole[16].

Along with the need for network traffic classification, the increasing user demand for privacy security and data encryption also increases the amount of encrypted traffic on the internet[17]. The trend of encrypted traffic is becoming standard practice in network security today. This practice creates new challenges for conventional network traffic classification and identification processes. Moreover, the rapid development of the internet and communication devices has led to a larger and more complicated network traffic flow structure[18]. The complexity of this network generates an abundance of large amounts of traffic data and raises new challenges in network management and optimization, including network traffic classification.

2.3. OTT Service

Over-the-top (OTT) refers to services that deliver audio, video, and other media over the Internet by leveraging the infrastructure used by network operators without their involvement in the control or distribution of the content[19]. These services include audio, video, network, instant messaging, file sharing, games, streaming, and others. OTT services are widely used by the public because they have many advantages. For example, OTT-based communication services such as WhatsApp allow users to communicate without additional costs. This advantage causes an expanded use of OTT services, impacting the continued acceleration of data traffic passing through the operator's network.

2.4. Deep Learning

Deep learning is defined as using interconnected deep networks to calculate algorithms that alternately use several layers to produce an output [20]. Most of the deep learning models use artificial neural network architecture. Therefore, this model is often called a deep neural network [21]. Deep learning has a much greater learning capacity when compared to conventional machine learning methods, which is why this method can effectively capture very complicated patterns [16]. Recently, researchers applied deep learning to classify encrypted network traffic used to classify network data according to specific parameters.

Convolutional Neural Network (CNN) is a particular type of neural network for processing data with a grid-like topology [22]. CNN extracts feature from the input data using a layer consisting of convolutional operations [23]. CNN consists of several convolution layers, pooling layers, and fully connected layers. The CNN architecture is inspired by the animal's visual cortex, which is the part of the brain responsible for processing information in graphical form. The CNN architecture is shown in Figure 2.

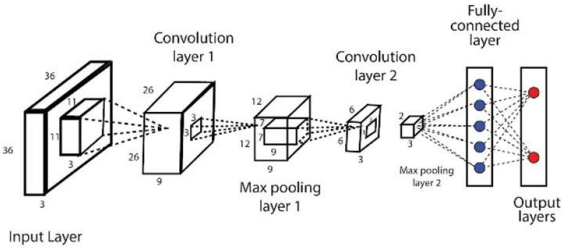


Figure 2. CNN Architecture

Long-Short-Term Memory is a type of Recurrent Neural Network (RNN) used to study, process, and classify sequential data because it can learn long-term dependencies between data. This model was first developed by Hochreiter and Schmidhuber in 1997 [24], in which each repeating node is replaced by a memory cell. Each memory cell contains an internal state as a node with self-connected recurrent edges with a fixed weight of one to ensure that the gradient can pass through many timesteps without vanishing [25]. The LSTM model can process sequential data by traversing all sequence elements in the data using timesteps and traversing information related to data content to update memory, where the last feature is placed on top of the previous feature. This temporal sequential relationship captures feature sequence characteristics better [26]. The LSTM architecture can be seen in Figure 3.

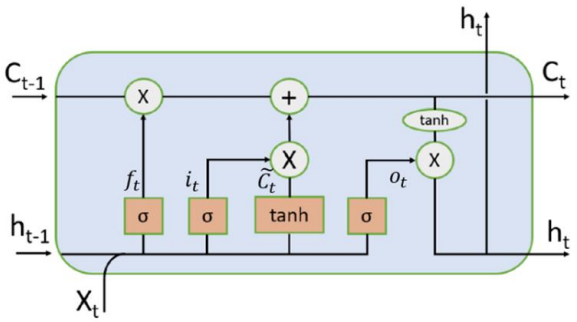


Figure 3. LSTM Architecture

Bidirectional Long Short–Term Memory (Bi-LSTM) is a type of RNN specifically designed to overcome the limitations found in the RNN model and is a development of LSTM. The neural state in Bi-LSTM is

divided into forward and backward states, producing two different forward and backward RNNs[27]. Bi-LSTM follows the process of splitting a neuron from a normal RNN into two directions—one for backward state or negative time direction and one for forward state or positive time direction. Combining the outputs of the two RNNs that convey information from opposite directions makes it possible to capture the context from both ends of the sequence [28]. Bi-LSTM architecture can be seen in Figure 4.

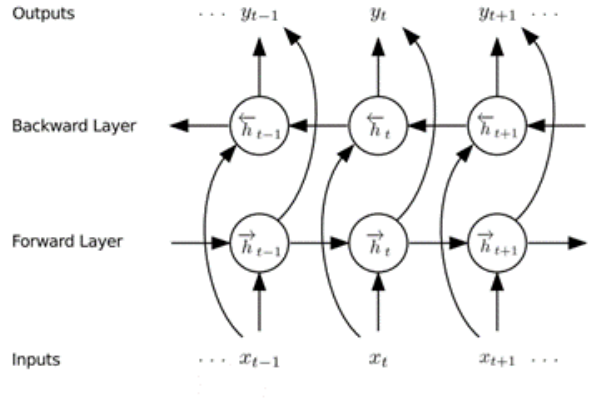


Figure 4. Bi-LSTM Architecture

3. RESEARCH METHODS

The research stages used in this study adopted the DSRM framework by Peffers *et al*. [2], which offers a systematic method for developing and evaluating artefacts designed for specific problem areas. Figure 5 presents our research approach and describes the activities carried out in the DSR cycle.

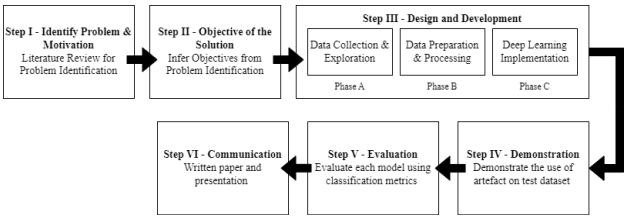


Figure 5. Design Science Research Steps

a. Step 1: Problem Identification and Motivation
   Initially, a thorough literature review was conducted to identify pertinent problems and motivations within the domain. This step serves as the foundation for subsequent stages by establishing the context and scope of the research.
b. Step 2: Objective of the Solution
   Based on the identified problems, clear objectives for the proposed solution were inferred. These objectives are logically derived from the problem definition and serve as guiding principles throughout the research process.
c. Step 3: Design and Development
   This stage consists of three distinct phases:

- Phase A: Data Collection and Explanation, where comprehensive data collection was conducted to gather relevant information and insights into the problem domain.
- Phase B: Data Preparation & Processing involved refining the collected data to ensure it was compatible and suitable for further analysis.
- Phase C: Deep Learning Implementation by utilizing advanced computational techniques to develop and refine the proposed solution, enhancing accuracy and efficiency through the application of deep learning algorithms.

d. Step 4: Demonstration
   The developed artefacts go through rigorous testing and evaluation processes using test datasets to demonstrate its effectiveness and functionality in real-world scenarios.
e. Step 5: Evaluation
   Each deep learning model was evaluated using established classification metrics. This stage provides critical insights into the performance and effectiveness of the proposed solution.
f. Step 6: Communication
   The findings and outcomes of the design science research were communicated through written papers and academic presentations, distributing knowledge of the designed artefacts, its utility, and effectiveness to relevant stakeholders and professional practitioners.

4. RESULTS

4.1. Research Activities Based on the DSRM Framework

The initial activity in this research is problem identification and motivation. At this stage, we conducted a literature study regarding previous actions taken to solve the problem of identifying OTT applications that have been encrypted. The rapid development of the Internet and the current communications industry has contributed to the increasing volume and frequency of network traffic, which is dynamic and complex. Along with this, the growing user demand for privacy security and data encryption has also increased the amount of encrypted traffic on the internet. This practice creates new challenges for conventional network traffic classification and identification processes.

This problem becomes input in the second stage of the DSR method, namely determining the target for the solution. From these problems, the answer is to develop an algorithm that can identify several OTT applications using network traffic data flow. At this stage, the researcher conducts a literature study on algorithms that have been used for the classification and identification of network traffic and then compares the performance of each algorithm with evaluation metrics. The results of this stage become input for the third stage, Design & Development. The design and development stage begins with the requirements gathering process, namely preparing the needs to create hardware and software artefacts, such as libraries and applications used to build classification algorithms and datasets for the training and testing process on the deep learning models. After the needs are met, the development of artefacts in the form of a classification model is carried out through several data analysis processes, such as data exploration, data preparation, and data pre-processing, which consists of feature engineering, feature selection, feature encoding, and data normalization.

Then the implementation of the CNN, LSTM, and Bi-LSTM algorithms is carried out on the transformed data. Each model is deployed to the website application at the end of the third stage. The results of these artefacts' design and development stages become input for demonstrating the use of artefacts using test datasets to produce objective evaluation results. At the demonstration stage, an evaluation is carried out as the fifth stage in the DSR approach. Evaluation is carried out using classification metrics to test each model's performance in classifying and identifying OTT network traffic flow. The design & development, demonstration, and evaluation stages go through several iterations until the model performance evaluation results are considered high enough by performing hyper-parameter tuning in the model development process. The last step is communication carried out through this research by discussing the DSRM approach in classifying OTT application network traffic.

4.2. Metrics for Evaluating Classification Models

In measuring the classification model, a confusion matrix is used to produce a score from the model's prediction results on the test data. Figure 6 displays the confusion matrix for multiclass classification [29].

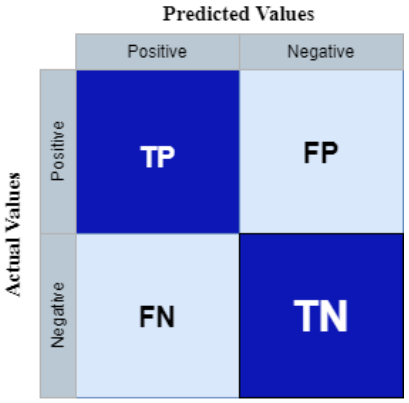


Figure 6. Confusion Matrix

Confusion matrix displays the number of classifications for each class with four types of classification results concerning one target class, namely:

a. True Positive (TP), the amount of positive data that is correctly estimated.
b. True Negative (TN), the amount of negative data that is correctly estimated.
c. False Positive (FP), the amount of negative data that is estimated as positive data.
d. False Negative (FN), the amount of positive data that is estimated as negative data.

Several measures of model performance evaluation can be calculated from the confusion matrix, which provides information on various aspects of the classification. This measure is calculated as the weighted average by class in multiclass classification. The following are the metrics for calculating the performance of the classification model.

a. Accuracy measures how well the classifier can predict the correct target value according to the target. The formula for accuracy is found in Eq. (1).

$$\text{Accuracy}: = \frac{\sum_{i=0}^{N} c_{ii}}{\sum_{i=0}^{N} \sum_{j=0}^{N} c_{ij}} \quad (1)$$

b. Recall is the classifier's ability to identify a particular class correctly. The recall formula is shown in Eq. (2).

$$Recall_{class} := \frac{TP_{class}}{TP_{class} + FN_{class}} \quad (2)$$

c. Precision, is the confidence of the classifier to predict a particular class correctly. The precision formula is shown in Eq. (3).

$$Precision_{class} := \frac{TP_{class}}{TP_{class} + FP_{class}} \quad (3)$$

d. F1-Score is the ability of the classifier to predict certain classes. The F1-score is determined by considering precision and recall. The f1-score formula is shown in Eq. (4).

$$F1 - score_{class} := \frac{2TP_{class}}{2TP_{class} + FN_{class} + FP_{class}} \quad (4)$$

e. AUC – ROC curve is a technique for measuring the performance of a classification model at various threshold settings. ROC (Receiver Operating Characteristic) is a probability curve, while AUC (Area Under Curve) measures the entire two-dimensional area under the ROC curve, as shown in Figure 7. The AUC – ROC curve aims to calculate the ability of a model to differentiate between classes. AUC values range from 0 to 1[30]. Models with an AUC of 1 can classify observations into classes perfectly.

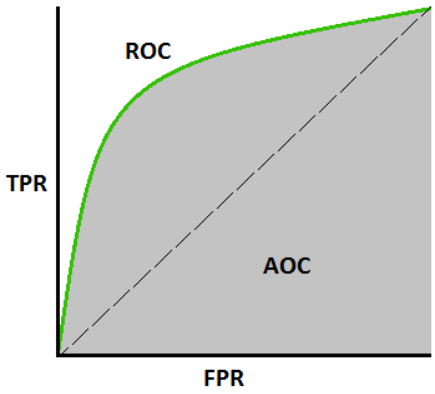


Figure 7. AUC-ROC Curve

4.3. DSRM Implementation in OTT Network Traffic Classification Case Study

OTT network traffic classification refers to the process of identifying various OTT applications by analyzing the received data packets, which plays a vital role in contemporary communication network practices. Accurate network traffic classification is essential for performing advanced network management tasks, including ensuring network QoS (quality of service) and detecting anomalies. There are two conventional methods in traffic classification, namely port-based classification and deep packet inspection (DPI)-based classification. Although these methods can achieve high traffic classification accuracy in some scenarios, conventional methods have limitations due to the prevalence of encrypted data in today's communication networks.

Therefore, a method that can classify encrypted network traffic flow is needed. Through literature study and comparison of several methods shown in Figure 5, we obtained solutions for solving the problem of encrypted network traffic classification using deep learning algorithms: CNN, LSTM, and Bi-LSTM. The deep learning method allows for the classification process directly by studying the representational features of the input data so that it can be a stepping stone for classifying encrypted network traffic with the best performance.

The artefacts produced are models trained using the CNN, LSTM, and Bi-LSTM algorithms with the data transformation process, as shown in Figure 5. This model contains learned parameters, weights, and an algorithmic architecture that allows the model to make predictions on new data without re-training from scratch. After the model-building process, deployment is carried out on the website application. The Activity Diagram for the OTT application network traffic classification website is shown in Figure 8. Users can access the application directly without the need to log in to the application. Furthermore, users can upload network traffic datasets not labeled as applications as input to the model, with conditions that the attributes used are the same as the training data. The web application will display the classification results for each flow according to the input data.

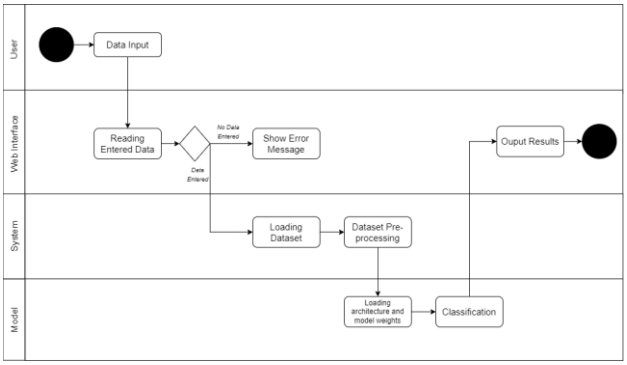
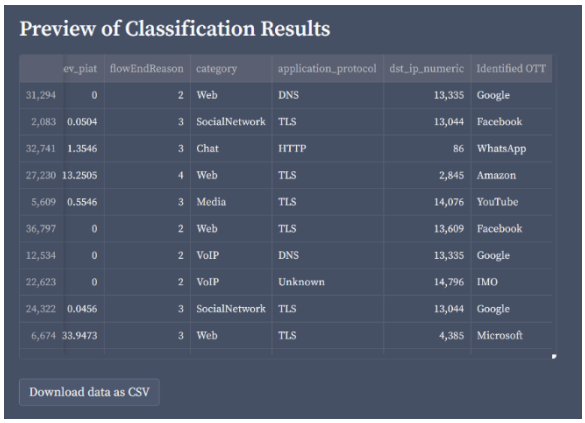


Figure 8. Use Case Diagram

Figure 9 displays the classification test results from the model classification. A preview screen shows a sample data model classification result with the 'Identified OTT' classification target column, followed

by a button to upload the classification results in CSV format.



	ev_piat	flowEndReason	category	application_protocol	dst_ip_numeric	Identified OTT
31,294	0	2	Web	DNS	13,335	Google
2,083	0.0504	3	SocialNetwork	TLS	13,044	Facebook
32,741	1.3546	3	Chat	HTTP	86	WhatsApp
27,230	13.2505	4	Web	TLS	2,845	Amazon
5,609	0.5546	3	Media	TLS	14,076	YouTube
36,797	0	2	Web	TLS	13,609	Facebook
12,534	0	2	VoIP	DNS	13,335	Google
22,623	0	2	VoIP	Unknown	14,796	IMO
24,322	0.0456	3	SocialNetwork	TLS	13,044	Google
6,674	33.9473	3	Web	TLS	4,385	Microsoft

Figure 9. Preview of Classification Results

Then the model performance evaluation visualization display contains accuracy metrics, confusion matrix images, ROC curves, and classification reports from uploaded data. The visualization of the model performance is shown in Figure 10.

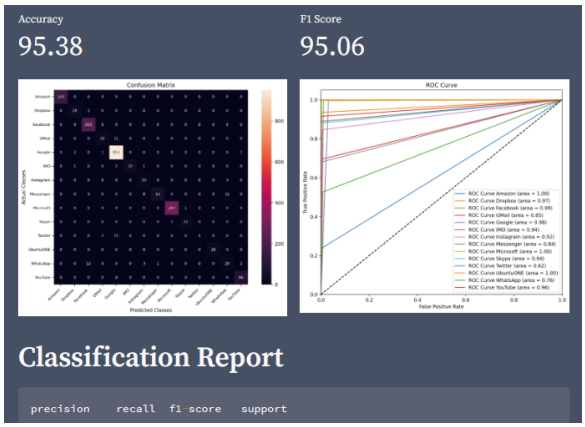


Figure 10. Model performance Evaluation Results

5. CONCLUSION

This study presents research methods and analysis of network traffic data using the DSRM approach. The proposed DSR framework outlines research stages consisting of various activities and phases, including identifying problems and determining solutions, design and development of artefacts, experimentation, evaluation, and communication to disseminate information about the artefacts that have been built. This framework is used to solve problems related to traffic classification of encrypted OTT application networks using the CNN, LSTM, and Bi-LSTM algorithms.

Although this study does not present the complete experiment output, a separate research yielded encouraging outcomes. After completing the steps outlined by the DSRM, the performance evaluation results of the three models showed positive results. The LSTM model has better performance in classifying OTT application network traffic, based on the evaluation value upon the test data, with an accuracy of 0,96 and an f1-score of 0,95, for the Bi-LSTM model with an accuracy of 0,95 and an f1-score of 0,95, and for the CNN model with an accuracy of 0,91 and an f1-score of 0,91.

The findings of this study contribute to advancing the understanding of network traffic analysis amidst the challenges of traffic encryption. By leveraging the DSRM framework and deep learning algorithms, the study demonstrates the potential to improve the accuracy and efficiency of traffic classification of OTT application networks. However, further research is needed to address real-time classification challenges and enhance the practical applicability of the proposed methods.

ACKNOWLEDGMENT

This research was not funded by any grant.

REFERENCES

[1] Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram, “Design Science in Information Systems Research,” *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004, doi: 10.2307/25148625.

[2] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, “A design science research methodology for information systems research,” *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45–77, Dec. 2007, doi: 10.2753/MIS0742-1222240302.

[3] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, “End-to-end encrypted traffic classification with one-dimensional convolution neural networks,” in *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, IEEE, Jul. 2017, pp. 43–48. doi: 10.1109/ISI.2017.8004872.

[4] X. Wang, S. Chen, and J. Su, “Real Network Traffic Collection and Deep Learning for Mobile App Identification,” *Wirel Commun Mob Comput*, vol. 2020, pp. 1–14, Feb. 2020, doi: 10.1155/2020/4707909.

[5] J. S. Rojas, A. Rendon, and J. C. Corrales, “Consumption behavior analysis of over the top services: Incremental learning or traditional methods?,” *IEEE Access*, vol. 7, pp. 136581–136591, 2019, doi: 10.1109/ACCESS.2019.2942782.

[6] I. Akbari *et al.*, “A Look Behind the Curtain: Traffic Classification in an Increasingly Encrypted Web,”

Proceedings of the ACM on Measurement and Analysis of Computing Systems, vol. 5, no. 1, pp. 1–26, Feb. 2021, doi: 10.1145/3447382.

[7] J.H. Kalwar and S. Bhatti, "Deep Learning Approaches for Network Traffic Classification in the Internet of Things (IoT): A Survey," *arXiv preprint*, Feb. 2024.

[8] M. Muntean and F. D. Militaru, "Design Science Research Framework for Performance Analysis Using Machine Learning Techniques," *Electronics (Basel)*, vol. 11, no. 16, p. 2504, Aug. 2022, doi: 10.3390/electronics11162504.

[9] C. Zhang, X. Wang, F. Li, Q. He, and M. Huang, "Deep learning–based network application classification for SDN," *Transactions on Emerging Telecommunications Technologies*, vol. 29, no. 5, May 2018, doi: 10.1002/ett.3302.

[10] A. Alturki, G. G. Gable, and W. Bandara, "A Design Science Research Roadmap," *International Conference on Design Science Research in Information Systems*, pp. 107–123, 2011, doi: 10.1007/978-3-642-20633-7_8.

[11] S. T. March and G. F. Smith, "Design and natural science research on information technology," *Decis Support Syst*, vol. 15, no. 4, pp. 251–266, 1995, doi: 10.1016/0167-9236(94)00041-2.

[12] S. K. Pradhan, H. M. Heyn, and E. Knauss, "Identifying and managing data quality requirements: a design science study in the field of automated driving," *Software Quality Journal*, 2023, doi: 10.1007/s11219-023-09622-8.

[13] J. F. Nunamaker, M. Chen, and T. D. M. Purdin, "Systems development in information systems research," *Journal of Management Information Systems*, vol. 7, no. 3, pp. 89–106, 1990, doi: 10.1080/07421222.1990.11517898.

[14] Z. Okonkwo, E. Foo, Q. Li, and Z. Hou, "A CNN Based Encrypted Network Traffic Classifier," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Feb. 2022, pp. 74–83. doi: 10.1145/3511616.3513101.

[15] N. Sharma and B. Arora, "Review of Machine Learning Techniques for Network Traffic Classification," *SSRN Electronic Journal*, 2020, doi: 10.2139/ssrn.3747605.

[16] M. Shafiq, X. Yu, A. A. Laghari, L. Yao, N. K. Karn, and F. Abdessamia, "Network Traffic Classification techniques and comparative analysis using Machine Learning algorithms," in *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, IEEE, Oct. 2016, pp. 2451–2455. doi: 10.1109/CompComm.2016.7925139.

[17] P. Velan, M. Cermák, P. P. Pavelčeleda, and M. Drašar, "A Survey of Methods for Encrypted Traffic Classification and Analysis," *INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT Int. J. Network Mgmt*, vol. 00, pp. 1–24, 2014, doi: 10.1002/nem.

[18] A. R. Mohammed, S. A. Mohammed, and S. Shirmohammadi, "Machine Learning and Deep Learning Based Traffic Classification and Prediction in Software Defined Networking," in *2019 IEEE International Symposium on Measurements & Networking (M&N)*, IEEE, Jul. 2019, pp. 1–6. doi: 10.1109/IWMN.2019.8805044.

[19] J. S. Rojas, A. Pekar, A. Rendon, and J. C. Corrales, "Smart User Consumption Profiling: Incremental Learning-Based OTT Service Degradation," *IEEE Access*, vol. 8, pp. 207426–207442, 2020, doi: 10.1109/ACCESS.2020.3037971.

[20] B. Dong and X. Wang, "Comparison deep learning method to traditional methods using for network intrusion detection," in *2016 8th IEEE International Conference on Communication Software and Networks (ICCSN)*, IEEE, Jun. 2016, pp. 581–585. doi: 10.1109/ICCSN.2016.7586590.

[21] I. H. Sarker, "Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions," *SN Computer Science*, vol. 2, no. 6. Springer, Nov. 01, 2021. doi: 10.1007/s42979-021-00815-1.

[22] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[23] M. Lotfollahi, R. S. H. Zade, M. J. Siavoshani, and M. Saberian, "Deep Packet: A Novel Approach For Encrypted Traffic Classification Using Deep Learning," *Soft comput*, vol. 24, no. 3, pp. 1999–2012, Sep. 2017.

[24] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[25] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into Deep Learning," *arXiv preprint arXiv:2106.11342*, 2021.

[26] S. Guo, B. Chen, and Y. Su, "Network Anomaly Traffic Detection Method Based on Multi-SAE and LSTM," *J Phys Conf Ser*, vol. 2025, no. 1, p. 012013, Sep. 2021, doi: 10.1088/1742-6596/2025/1/012013.

[27] M. Kumar Ojha, S. Wadhwani, A. Kumar Wadhwani, and A. Shukla, "Deep Convolutional Bidirectional LSTM Model for identifying Ventricular Tachyarrhythmia using ECG Signal Variability," 2021, doi: 10.21203/rs.3.rs-1194607/v1.

[28] S. Cornegruta, R. Bakewell, S. Withey, and G. Montana, "Modelling Radiological Language with

Bidirectional Long Short-Term Memory Networks," Sep. 2016, [Online]. Available: http://arxiv.org/abs/1609.08409

[29] D. A. Rusdah and H. Murfi, "XGBoost in handling missing values for life insurance risk prediction," *SN Appl Sci*, vol. 2, no. 8, p. 1336, Aug. 2020, doi: 10.1007/s42452-020-3128-y.

[30] A. Alshehri and D. AlSaeed, "Breast Cancer Diagnosis in Thermography Using Pre-Trained VGG16 with Deep Attention Mechanisms," *Symmetry (Basel)*, vol. 15, no. 3, p. 582, Feb. 2023, doi: 10.3390/sym15030582.