A TWO-STAGE FLOW-BASED METHOD FOR CLASSIFYING ENCRYPTED GOOGLE SERVICE TRAFFIC

Khamdamov Utkir and Tojieva Feruza

Department of Info communication engineering, Tashkent University of information technologies named after Muhammad Al-Khwarizmi, Tashkent, Uzbekistan

ABSTRACT

Modern Internet traffic is dominated by a few global platforms, making accurate service identification essential for QoS management and network analytics. Yet pervasive encryption and shared infrastructure increasingly limit port-based methods and DPI. This paper presents a two-stage hierarchical framework for classifying encrypted Google-family traffic (YouTube, Gmail, Google Search) using only flow-level statistical and temporal features extracted with CICFlowMeter. Information Gain is applied to reduce feature redundancy and computational cost. As a baseline, a single-stage Random Forest achieved an overall accuracy of 0.89 but showed strong confusion between Gmail and Google Search. In the proposed framework, Stage 1 separates YouTube vs Other, and only flows predicted as Other are forwarded to Stage 2, where Gmail vs Search are distinguished. This staged design increased the overall accuracy from 0.89 to 0.96, with ROC-AUC of 0.99 for Stage 1 and 0.93 for Stage 2. The results indicate that hierarchical classification effectively mitigates service ambiguity in encrypted, shared-infrastructure environments and supports scalable telecom monitoring.

KEYWORDS

Machine learning, Flow-based, Traffic, Random Forest, Google services, Feature selection, Encrypted traffic, Classification.

1. INTRODUCTION

In recent years, a large share of Internet traffic has become concentrated around the ecosystems of major content providers and global platforms. Among major online platforms, Alphabet's services—particularly YouTube and Google Search—remain some of the most heavily used, since they support everyday activities such as video viewing and information search. Industry traffic reports further suggest that Google-related services generate a substantial share of overall internet traffic [1]. Therefore, accurate separation of Google-ecosystem flows is becoming increasingly important for telecom operators and service providers in practical tasks such as efficient network resource management, maintaining QoS, monitoring, and service-level analytics.

However, identifying modern traffic using traditional approaches is becoming progressively more difficult. Port-based methods often fail to provide sufficient accuracy due to dynamic ports and protocol diversity, while DPI cannot fully rely on packet payload information in environments where encryption technologies such as TLS/QUIC are widely deployed. This challenge is further compounded by the fact that many Google services run on shared infrastructure, including CDN-based delivery, overlapping IP address pools, and DNS-based traffic steering. This issue is especially evident for closely related services such as Gmail and Google Search, whose flow-level statistical characteristics may be similar, thereby increasing classification ambiguity. At the

same time, separating closely related services within the Google ecosystem with high accuracy using a single model is not always straightforward. For this reason, structuring the problem logically and proposing a staged solution is practically well justified.

This paper proposes a two-stage traffic classification method for separating Google services under encrypted-traffic conditions. The core idea is that, in the first stage, flows are separated into more easily distinguishable classes, while in the second stage, confusion between the remaining closely related classes is reduced. In this study, traffic was captured in a laboratory environment; flow features were computed using CICFlowMeter, and to ensure class balance, 10,000 flows were randomly selected from each class. To constrain model complexity and retain informative features, an Information Gain (IG)-based filtering step was applied, and low-value features were removed.

2. RELATED WORK

Studies show that the main approaches to network traffic classification currently in use can generally be divided into three groups: port based, payload based, and flow based approaches.

Regarding studies on port based approaches, Madhukar A. and Williamson C. [2] analyzed P2P-application traffic on the University of Calgary Internet network over a two-year period and evaluated a port based classification method. Their results showed that the port-based approach failed to correctly identify approximately 30–70% of traffic flows, leaving that portion of traffic as “unknown.” This finding demonstrates that, under modern conditions, port-based methods are not sufficiently effective for P2P traffic classification.

Moore A. and Papagiannaki K. analyzed traffic traversing the network of the Genome Campus research center and compared the outcomes of port-based classification with payload-based identification methods. The results indicated that the port number–based approach correctly classified up to 70% of the total traffic volume, while roughly 30% of traffic was misclassified. This study provides scientific evidence that port-based classification is not sufficiently reliable for complex and dynamic application traffic [3]. Overall, the above studies confirm that port-based classification is simple and requires low computational resources; however, due to the widespread use of dynamic and non-standard ports in modern networks, this approach yields low classification accuracy for contemporary Internet traffic.

Regarding studies based on payload-driven approaches, Sen S., Spatscheck O., and Wang D. developed protocol signatures for five P2P applications. Based on the collected signatures, they designed an online filter capable of accurately identifying P2P traffic by analyzing up to the first 10 packets of a connection. Compared with the port-based method, this approach enabled the detection of three times more P2P traffic volume for the Kazaa application [4]. In his study, Zhou Y. [5] demonstrated that payload-based approaches are ineffective for classifying encrypted traffic, and proposed a 1D-CNN–based HexCNN-1D model for classifying VPN and encrypted traffic. These results indicate that, under real-world conditions, the applicability of payload-based methods is limited when traffic is encrypted. Overall, the reviewed studies indicate that constructing accurate and stable flow signatures is not a trivial task. Manual signature extraction requires substantial effort and time, and it often involves trade-offs among the available traffic samples, expert knowledge, and the final classification quality.

In recent years, a number of studies have also focused on classifying network flows without inspecting packet payloads. In particular, there are approaches aimed at identifying email flows within HTTPS, P2P traffic, multimedia streams, and multiplayer FPS game traffic [6]. These studies suggest that approaches based on statistical features derived from packet headers and flow

behavior are practically effective and can enable accurate classification even under encrypted-traffic conditions.

Another recent work proposed a real-time and dynamic traffic classification model for Software-Defined Networking (SDN) environments, called the Dynamic Network Classifier (DNC) [7]. The model was implemented using Docker-based virtual containers and improved accuracy through a Performance Accelerator Algorithm (PAA) that combines multiple classifiers, including C4.5, Random Forest and K-Nearest Neighbor (KNN). The authors built a dataset of 463,874 traffic flows collected from 176 applications and grouped the flows into 10 service classes (e.g., VoIP, Video, OAM, Transactional, Scavenger, Bulk, Best-Effort, etc.). The model was evaluated using compact feature sets with 3, 5 and 7 attributes. The reported results indicate that the PAA ensemble significantly improves classification accuracy and provides an effective solution for adaptive, real-time traffic classification in SDN networks.

Oudah and co-authors addressed the problem of application-level traffic identification under modern constraints, emphasizing that reliance on port numbers or DPI becomes less effective due to encryption and the widespread use of dynamic ports [8]. To overcome these limitations, the authors proposed new flow-statistical features, namely burstiness and idle time, extracted by modifying the tcptrace tool. They then applied a C5.0 decision tree classifier for application-level classification. In evaluations on real, uncontrolled traffic, the method achieved an accuracy of about 79%, supporting the practical usefulness of burstiness and idle-time features for identifying applications in realistic encrypted-traffic conditions.

Another important direction is the evolution of transport protocols, especially QUIC. Luxemburk, Hynek, and Cejka investigated encrypted traffic classification specifically in the QUIC setting and pointed out that QUIC further obscures parts of the handshake and may limit simple identification signals compared to TLS over TCP [9]. Their findings indicate that models validated on TCP/TLS traffic should be re-validated for QUIC and that QUIC-aware datasets and feature checks are essential for realistic modern evaluations.

To support real-time operator requirements, Akem, Fraysse, and Fiore proposed an approach for encrypted traffic classification (ETC) at line rate in programmable switches. Their work builds a Random Forest model using only lightweight features derived from packet sizes and inter-arrival times, and then encodes the trained model into P4-programmable switches [10]. The reported results suggest that accurate encrypted-traffic classification can be achieved with low latency under high-throughput conditions, which is particularly relevant for deployment in backbone and provider networks.

In addition, hierarchical modeling has been explored to address ambiguity among similar traffic types. Li and co-authors presented a hierarchical approach for encrypted traffic classification, showing that decomposing the problem into multiple stages can improve identification of coarse classes and enable finer-grained inference compared to a single flat classifier [11]. This result supports the practical idea that staging the classification process can reduce confusion when different services exhibit similar encrypted-flow behavior.

More recently, Elshewey and co-authors investigated encrypted HTTPS traffic classification and demonstrated that, even when payload is unavailable, combining appropriate preprocessing with machine-learning models can achieve effective separation of multiple encrypted traffic categories [12]. This study further confirms that payload-free encrypted traffic identification remains feasible, but depends strongly on the quality of flow representation and evaluation design.

3. METHODOLOGY

The aim of this study is to identify flows of three Google-ecosystem services YouTube, Gmail, and Google Search under encrypted-traffic conditions without inspecting packet payload. This problem is particularly important in real-world networks because, under a shared infrastructure and overlapping IP space, IP/DNS-based separation does not always provide stable and reliable results. Therefore, it is reasonable to differentiate services using flow-level statistical characteristics derived from packet headers.

The proposed methodology is based on a two-stage classification principle. In the first stage, YouTube flows are separated from the flows of the remaining Google services (YouTube vs Other). In the second stage, flows assigned to the "Other" class in Stage 1 are further distinguished between Gmail and Google Search. This hierarchical structure isolates the "easier-to-separate" class (YouTube) at an early stage, reduces the number of flows forwarded to the next stage, and enables Stage 2 to focus on reducing confusion only between the "closely related" classes (Gmail and Search).

Traffic data were collected in a laboratory environment using separate experimental scenarios. Packets were captured and stored in PCAP format. The recorded PCAP files were converted into flows using CICFlowMeter, and statistical features were computed for each flow. IP addresses and port numbers were not included in the model.

To constrain model complexity, accelerate computation, and reduce the influence of redundant and non-informative features, an Information Gain (IG) based filtering step was applied. IG-based filtering is performed separately for Stage 1 and Stage 2, because the class composition is different (Stage 1: YouTube vs Other; Stage 2: Gmail vs Search), and the most informative features may therefore also differ. In addition, the features ranked by IG were used to train the Random Forest model: each feature was included only if adding it improved the model's accuracy.

Table 1. Feature set used in the two-stage classifier.

Features	**Stage 1**	**Stage 2**
Total.Length.of.Fwd.Packets	+	+
Total.Length.of.Bwd.Packets	+	+
Total.Fwd.Packets	+	+
Total.Backward.Packets	+	+
Packet.Length.Std	+	+
Packet.Length.Mean	+	+
Flow.Duration	+	-
Flow.IAT.Mean	-	+
Idle.Mean	+	+
Idle.Std	+	+
Active.Mean	-	+

4. RESULTS

In this study, a single-stage baseline classification model was first evaluated for classifying Google-family traffic. Feature informativeness was assessed using Information Gain (IG) and machine-learning-based evaluation; 11 statistical features were identified as informative. Using these features, the Random Forest model achieved an overall accuracy of 0.89 and macro-F1 = 0.90 for the YouTube, Gmail, and Google Search classes. These results confirm that the selected

features are sufficiently effective for separating YouTube flows from other traffic and for classifying Google-family traffic in general.

At the same time, the analysis showed that the main confusion occurs between the Gmail and Google Search classes: the F1-score was 0.84 for Gmail, 0.85 for Google Search, and 0.99 for YouTube. Approximately 18% of Gmail flows were classified as Google, while about 11% of Google flows were classified as Gmail. Figure 1 presents the baseline confusion matrix in percentages, and Table 2 summarizes the corresponding classification results.

Table 2. Classification results for the baseline model on Google-family applications.

Class	**Accurancy**	**Recall**	**F1-Score**
Gmail	0.87	0.81	0.84
Google Search	0.82	0.87	0.85
YouTube	0.98	0.99	0.99

As indicated by the confusion matrix, the baseline classification model encounters difficulties in distinguishing between applications within the Google ecosystem whose traffic behavior is similar. Accordingly, in the next step of this study, a two-stage classification model was proposed in order to increase statistical separability between classes and improve classification accuracy for Gmail.

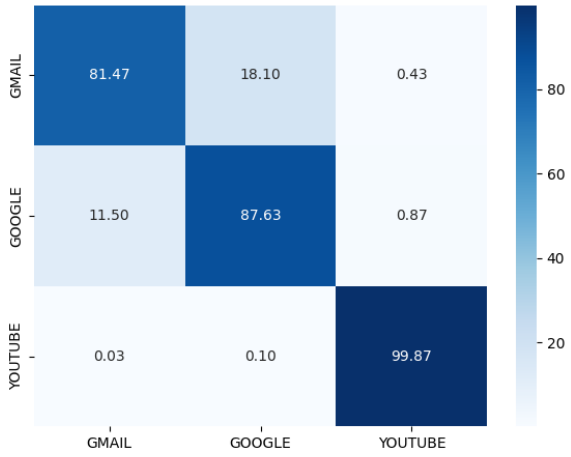

Figure 1. Confusion matrix of the baseline model for Gmail, Google Search, and YouTube applications

A two-stage classification model was applied in this study to improve per-class performance. The main idea of the two-stage approach is to first isolate the most easily separable class from the main traffic stream, and then classify the remaining, more difficult-to-separate classes using a dedicated feature subset. This design can significantly increase overall classification accuracy.

The proposed two-stage model relies on the same set of 11 features: in Stage 1, 9 of these features are used to separate YouTube flows, while in Stage 2, 10 features are used to distinguish Gmail and Google Search among the Non-YouTube flows.

In Stage 1, the Flow. Duration feature (flow duration) was used to separate YouTube flows from Non-YouTube flows. YouTube flows generally exhibit longer durations, larger packet volumes, and a more continuous transmission pattern.However, in Stage 2, Flow.Duration was found to be misleading when distinguishing Gmail and Google Search flows, and therefore it was excluded from the Stage-2 feature set.

For Google Search traffic, web-page loading, refreshing search results, and incremental content loading while the user remains on a page can generate comparatively longer and more continuous data-transfer sessions. As a result, the Google Search class tends to have higher Active.Mean values, meaning that the average duration of active periods is larger than for Gmail. In contrast, Gmail flows are often limited to smaller and shorter transfers such as mailbox synchronization, checking new messages, or sending short emails. Consequently, many Gmail flows contain very short active segments, and their Active. Mean values are concentrated in a lower range.

To assess the reliability of the proposed two-stage model, ROC-AUC metrics were used. The ROC (Receiver Operating Characteristic) curve illustrates how the true positive rate and false positive rate change as the decision threshold varies.

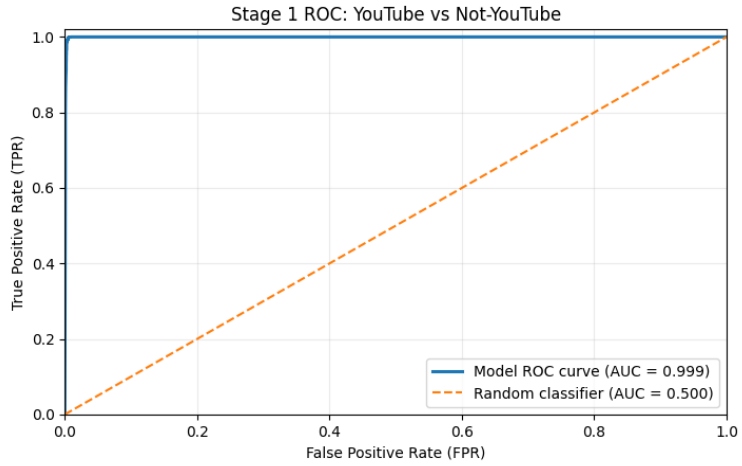

Figure 2. ROC curve for the Stage-1 classification model on Google-family traffic

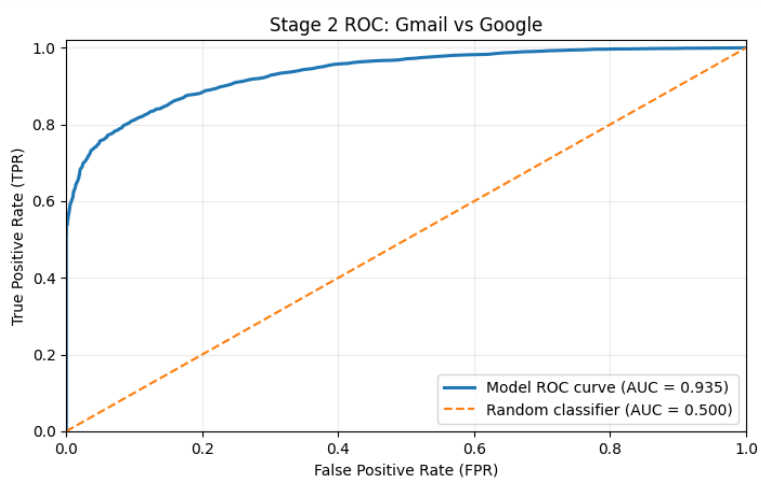

Figure 3. ROC curve for the Stage-2 classification model on Google-family traffic

The area under the curve (AUC) summarizes class separability: AUC = 0.5 corresponds to random guessing, while values closer to 1 indicate stronger discrimination. For the Stage-1 model separating YouTube vs Non-YouTube, the ROC-AUC reached 0.99, confirming that the selected features effectively distinguish these classes. Figures 2 and 3 show the ROC curves for Stage 1 and Stage 2, respectively. In the second stage, the model built to distinguish Gmail and Google

Search traffic achieved ROC-AUC = 0.93. These results indicate that both stages of the proposed two-stage classification model substantially outperform random classification. Evaluation results for the two-stage classification model.

Table 3. Results of the baseline classification model and the proposed two-stage classification model.

Class	Accurancy	Recall	F1-Score
Gmail	0.96	0.94	0.95
Google Search	0.94	0.96	0.95
YouTube	0.99	0.99	0.99

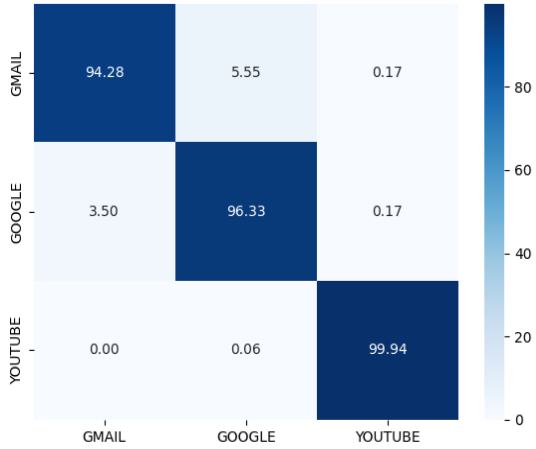

Figure 4. Confusion matrix of the two-stage classification model for Gmail, Google Search, and YouTube applications

Accuracy for the YouTube class increased from 0.98 to 0.99, while the accuracy for Gmail and Google Search increased from 0.86 to 0.96 and from 0.82 to 0.94, respectively. Overall accuracy and overall F1-score improved from 0.89 to 0.96, i.e., by 7.86%, indicating that the proposed two-stage classification approach significantly reduces misclassification between classes within the Google ecosystem and improves overall model performance. In particular, this approach reduces the errors observed with the baseline model and substantially increases the accuracy of Gmail flow identification.

5. CONCLUSIONS

This paper investigated service-level identification of Google-family traffic under encrypted-traffic conditions and proposed a two-stage cascaded classification strategy based solely on flow-level statistical and temporal features. The key motivation is that YouTube, Gmail, and Google Search often share infrastructure and exhibit partially overlapping flow behavior, which can increase confusion in a single flat classifier. To address this, the proposed cascade first separates YouTube vs Non-YouTube, and then refines the Non-YouTube subset into Gmail vs Google Search, allowing each stage to focus on a simpler and more homogeneous decision boundary.

The experimental evaluation demonstrated that the cascaded design yields a clear improvement over the baseline single-stage model. In particular, the overall accuracy increased from 0.89 to 0.96, while the class-wise performance remained strong for YouTube (0.99) and improved notably for Gmail (0.96) and Google Search (0.94), indicating a substantial reduction of

confusion between the most similar classes. In addition to point metrics, the stage-wise ROC–AUC analysis confirmed robust separability across thresholds, achieving 0.99 for the first stage and 0.93 for the second stage. These results suggest that the cascade architecture is an effective way to mitigate misclassification that arises when services share network infrastructure and exhibit similar traffic dynamics.

From a practical standpoint, the proposed approach is appealing because it avoids direct identification methods such as IP/DNS/DPI and instead uses features derived from packet headers and timing information. This property makes the method suitable for encrypted environments where payload visibility is limited and where service-level monitoring must be performed using metadata and flow behavior. Moreover, the two-stage structure provides a modular pathway for deployment: the first stage can operate as a lightweight filter to isolate high-confidence YouTube flows, while the second stage concentrates computational effort on the harder Gmail vs Search discrimination problem.

REFERENCES

[1] Sandvine Incorporated, Waterloo, Ontario, Canada, (2023) "The Global Internet Phenomena Report (H1 2023)", Technical Report.

[2] Williams, N., Zander, S. & Armitage, G., (2006) "A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification", *ACM SIGCOMM Computer Communication Review,* Vol. 36, No. 5, pp. 5–16.

[3] Moore, A.W. & Papagiannaki, K., (2005) "Toward the Accurate Identification of Network Applications", *Proceedings of Passive and Active Network Measurement (PAM),* Springer (LNCS 3431).

[4] Sen, S., Spatscheck, O., & Wang, D. (2004). Accurate, Scalable In-Network Identification of P2P Traffic Using Application Signatures. *Proceedings of the International World Wide Web Conference (WWW 2004),* 512–521.

[5] Zhou, Y., et al. (2023). Identification of Encrypted and Malicious Network Traffic Based on One-Dimensional Convolutional Neural Network (HexCNN-1D). *Journal of Cloud Computing*. doi: 10.1186/s13677-023-00430-w.

[6] Velan, P., Čermák, M., Čeleda, P. & Drášar, M., (2015) "A survey of methods for encrypted traffic classification and analysis", International Journal of Network Management, Vol. 25, No. 5, pp. 355–374.

[7] Mondal, P.K., Aguirre Sanchez, L.P., Benedetto, E., Shen, Y. & Guo, M., (2021) "A dynamic network traffic classifier using supervised ML for a Docker-based SDN network", Connection Science, Vol. 33, No. 3, pp. 527–547, doi: 10.1080/09540091.2020.1870437.

[8] Oudah, H., Ghita, B., Bakhshi, T., Alruban, A. & Walker, D.J., (2019) "Using Burstiness for Network Applications Classification", Journal of Computer Networks and Communications, Vol. 2019, Article ID 5758437, doi: 10.1155/2019/5758437.

[9] Luxemburk, J., Hynek, K. & Čejka, T., (2023) "Encrypted Traffic Classification: The QUIC Case", Proceedings of TMA, doi: 10.23919/TMA58422.2023.10199052.

[10] Akem, A.T.-J., Fraysse, G. & Fiore, M., (2024) "Encrypted Traffic Classification at Line Rate in Programmable Switches with Machine Learning", Proceedings of IEEE/IFIP NOMS, pp. 1–9, doi: 10.1109/NOMS59830.2024.10575394.

[11] Li, Y. et al., (2022) "From traffic classes to content: A hierarchical approach for encrypted traffic classification", Computer Networks, Vol. 212, Article 109017, doi: 10.1016/j.comnet.2022.109017.

[12] Elshewey, A.M. et al., (2025) "Enhancing encrypted HTTPS traffic classification based on stacked deep ensembles models", Scientific Reports, doi: 10.1038/s41598-025-21261-6.

AUTHORS

Khamdamov Utkir was born on March 15, 1977, in the Republic of Uzbekistan. He is a Doctor of Science (DSc) and a professor at the Department of Info communication Engineering, Tashkent University of Information Technologies named after Muhammad al-Khwarizmi. His research interests include digital telecommunications and signal processing.

Feruza Tojiyeva was born on November 4, 1990, in the Republic of Uzbekistan. She is a PhD researcher at the Department of Info communication Engineering, Tashkent University of Information Technologies named after Muhammad al-Khwarizmi. Her research focuses on network traffic classification.