Are Industrial ML Image Classifiers Robust to Data Affected by Network QoS Degradation?

Raman Zatsarenko, Chirayu Anil Marathe, Sergei Chuprov, Matthew Hyland, and Leon Reznik
Department of Computer Science, Rochester Institute of Technology, Rochester, NY, USA
Email: rz4983@rit.edu, cm6647@rit.edu, sc1723@rit.edu, mph6083@rit.edu, leon.reznik@rit.edu

Abstract—**In industrial applications, Machine Learning (ML) services are often deployed on cloud infrastructure and require a transfer of the input data over a network, which is susceptible to Quality of Service (QoS) degradation. In this paper we investigate the robustness of industrial ML classifiers towards varying Data Quality (DQ) due to degradation in network QoS. We define the robustness of an ML model as the ability to maintain a certain level of performance under variable levels of DQ at its input. We employ the classification accuracy as the performance metric for the ML classifiers studied. The POWDER testbed is utilized to create an experimental setup consisting of a real-world wireless network connecting two nodes. We transfer multiple video and image files between the two nodes under varying degrees of packet loss and varying buffer sizes to create degraded data. We then evaluate the performance of AWS Rekognition, a commercial ML tool for on-demand object detection, on corrupted video and image data. We also evaluate the performance of YOLOv7 to compare the performance of a commercial and an open-source model. As a result we demonstrate that even a slight degree of packet loss, 1% for images and 0.2% for videos, can have a drastic impact on the classification performance of the system. We discuss the possible ways to make industrial ML systems more robust to network QoS degradation.**

Index Terms—**image processing, network QoS, image classification, robustness, data quality**

I. Introduction

Industrial ML classifiers, such as AWS Rekognition[1], and open-source like YOLO [9], are widely employed for a variety of tasks in imaging and other areas, such as real-time traffic management and decision making in Intelligent Transportation Systems (ITSs) [8], anomalies detection in X-ray medical images [7], and fraudulent transactions identification in financial applications [4]. Despite being designed as standalone applications usually deployed on cloud infrastructure, image ML classifiers require data to get delivered there, for which purpose they typically rely on an intermediate network. The classifier's performance has been shown to heavily depend on the Data Quality (DQ) at their entrance point [4], which emphasizes the Quality of Service (QoS) of the network infrastructure employed for conveying the data from the data source. We have shown for other applications [4], [6], [7] that even minor degradation in network QoS due to the factors such as packet loss, bit errors, network congestion, etc., can significantly affect the data transmitted reducing the DQ, which results in a ML application performance degradation.

In our integral approach [5], we consider an image classifier as an endpoint of the data delivery pipeline from the data source through the communication network to the cloud that makes its performance dependable not only of the DQ but also network QoS. In order to ensure reliable and resilient image classification service, it is important to determine the QoS conditions that ML classifier is able to tolerate while maintaining acceptable performance. This challenge was previously addressed in the context of static image classification using open-source models by Chuprov *et al.* [6]. In this paper, through our empirical study, we investigate the performance of the AWS Rekognition industrial classifier in the domain of video and image classification. Specifically, we transfer video files using UDP protocol between two nodes in a real wireless network established on the POWDER wireless communication research platform [1]. During the data transmission, we simulate varying network conditions caused by either network technological problems or malicious attacks [3] resulting in changing packet loss rate and the size of the receiver buffer socket, which leads to a degraded version of the original data. Then we employ AWS Rekognition over the original and degraded data to quantitatively assess the effects of network-based DQ degradation on the image or video classifier performance. Additionally, we compare AWS Rekognition ability to tolerate corruptions in static imaging data against the open-source YOLOv7 model.

This paper has two major contributions. First, we systematically investigate and analyze the ML applications' robustness to network QoS degradation in the real image and video object detection and classification use case. Second, based on the results we develop, we derive our recommendations on employing the pre-trained image and video classification systems in varied network QoS conditions. Previously, we studied the performance of AWS Rekognition under varied network QoS conditions with medical image data [6]. In the present work, we expand the field of applications and examine the ML robustness towards DQ variations in videos in the context of the ITS. We consider ITS as a specific use case in which the quality of real-time intelligent decisions is vital. In particular, we consider the scenario when the DQ processed by the ML application may highly vary due to changing network conditions, which results in ML application performance drop. We observe a significant drop in the AWS

[1] https://aws.amazon.com/rekognition/

979-8-3503-2969-8/23/$31.00 ©2023 IEEE

Rekognition classification accuracy demonstrated on videos and images that are affected by varied network QoS. We show that AWS Rekognition, if it used "off the shelf", is not robust enough to be leveraged in industrial applications, especially those deployed in real-time systems, such as ITS. To address this challenge, we discuss known techniques that can be employed to enhance the image classifiers' robustness towards the varying DQ.

II. Previous Work

The impact of network QoS degradation on the performance of video classification systems is still a relatively novel research area. In our previous work [7], we studied the robustness of ML classifiers towards low DQ due to network QoS degradation in medical image data. We found that even a small packet loss rate can result in a considerable decline in classification performance for medical images. We determined that, depending on the medical image classification system, only the packet loss of less then 1% could be reliably tolerated. Earlier, we investigated how DQ degradation due to packet loss and varying buffer sizes affects the performance of various open-source image classification models, namely VGG16, Inception, and EfficientNet [6], in the ITS use case. We demonstrated that these image classifiers can endure a packet loss of around 10%. Higher packet losses prevented employing the studied classifiers in real domains, as their classification accuracy dropped more than by a third.

One known approach to make ML application more tolerant to DQ variations is to re-train the model on a data than involves samples of various quality. In [2], we studied how Transfer Learning, which allows to adapt ML models trained on one application domain to the target one, can effectively enhance ML robustness towards DQ variations. We found that state-of-the-art ML image classifiers are usually pre-trained on the high quality data only, and need to be re-trained using the additional set of lower quality images obtained under realistic conditions in order to satisfy the performance requirements posed by the industry. In [4], alongside Transfer Learning we also investigated the feasibility of Federated Learning in improving robustness of road sign images classification in ITS. Our results showed that the aggregation procedure incorporated in Federated Learning enables producing the global model more robust to the input DQ variation in comparison to the conventional centralized ML setup.

III. Empirical Study Methodology

A. Data Collection

To investigate the effects of network degradation on the video classifier's performance, we employ the Berkley Deep Drive data set (BDD100K)[2]. The BDD100K is a large, diverse, crowd-sourced video data set containing over 100,000 videos featuring various scene types such as city streets, residential areas, and highways in varying weather conditions recorded by the vehicle dash cam. We selected 35 videos from BDD100K that contain visible patterns corresponding to "Traffic Lights" and "Road Sign" labels detected by AWS Rekognition. Our collection involves 25 files with visible traffic lights and 10 files with visible road signs. We evaluate the performance of AWS Rekognition based on the confidence score supplied by the system and the classification accuracy we calculate based on the outputs, both for the original data and corrupted data.

B. Wireless Network Configuration Setup

We establish a wireless network in POWDER using the *geni-lib*[3] library for the Python programming language, which allows to generate RSPEC files for network topologies from Python code. We employ the established end-to-end LTE network to transmit video files between two nodes. During the data transmission, we vary such network parameters as packet loss rate and socket buffer size of the receiving node. We investigate packet loss rates up to 20% for the image transmission, and 0.1 to 1% for the videos since higher packet losses make videos too corrupted for their processing by AWS Rekognition. For the buffer sizes, we examine buffers of 128B, 256B, 512B, and 1024B.

C. ML Image and Video Classification Tools

AWS Rekognition is a cloud-based multi-functional ML service launched in 2016. It offers pre-trained computer vision model as a service with on-demand pricing for image and video analysis. Alongside AWS Rekognition, we also employ the YOLO image detection tool [9], which is a family of open-source real-time object detection models that use a single neural network to predict bounding boxes and class probabilities from full images. In our ITS scenario, we employ AWS Rekognition for detecting "Traffic Lights" and "Road Signs" labels in the uploaded images and videos, and YOLO for detecting the road signs in images specifically. First, we upload the original data to AWS Rekognition and process it to evaluate the baseline performance. AWS analyzes each frame in the video and employs proprietary ML techniques to detect and classify the objects in a given video frame. It assigns labels for the objects detected and outputs a confidence score for each object. The confidence score is a number between 0 and 100 that indicates the probability that a given prediction is correct. After determining the baseline performance, we process the data of varied quality by the employed ML systems. We compare the performance demonstrated by AWS Rekognition over classifying the images of various quality with the results provided by YOLOv7 model. AWS Rekognition recommends that applications that are very sensitive to detection errors (false positives) should discard results associated with confidence scores below a certain threshold.

D. Video Transfer and Classification Use Case

After determining the benchmark confidence and classification accuracy scores over the original data, we transmitted our videos over a real-world wireless network established with POWDER. During the data transmission, we vary packet loss

[2] https://bdd-data.berkeley.edu/

[3] https://pypi.org/project/geni-lib/

rate between 0-1% and buffer size at the receiver side between 512B and 1024B to capture the difference in resources available to the receiving node, which enables to obtain videos of various quality. To manipulate these network parameters and to organize the communication, we employ socket connections in Python. Figure 1 demonstrates an example of the visual degradation in a video transmitted with 0.5% packet loss and various buffer size values. After obtaining videos of various DQ, we upload them to AWS Rekognition again to investigate its performance over the data affected by the changes in network QoS.

E. Image Transfer and Classification Use Case

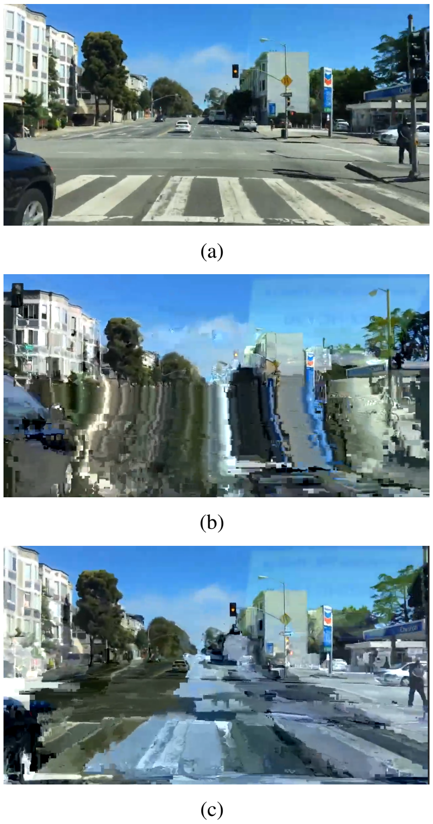
We utilize the same experimental setup to study the effects of packet loss on AWS Rekogntion performance over a static image data. For this, we employed stop and traffic sign images from Open Image V6 dataset[4]. In our research, we compare the performance demonstrated by the commercial AWS Rekognition ML system with the pre-trained YOLOv7 architecture. In our experiments related to static images processing, we employ multiple buffer sizes at the receiver's side, ranging from 128B to 1024B. We study the performance of those classifiers with packet loss rate varying from 0-20%. The comparison of performance demonstrated by a black-box system with a well-known open-source classifier will provide more insights about the robustness of industrial ML systems to the input data of various quality. Further we specifically discuss how we compare the robustness of AWS Rekognition with YOLOv7 when the static image data are provided as an input.

IV. Results

A. Static Image Classification Case

Figure 2(a) demonstrates that image classification accuracy is highly sensitive to DQ degradation caused by packet loss. As illustrated, even a packet loss rate of 1% leads to a significant drop in classification accuracy. Specifically in case of AWS Rekognition, we observe a performance drop of 26% for a receiver with the 128B buffer, a drop of 32% for a receiver with the 256B buffer, and a drop of 24% for a receiver with the 1024B buffer. Overall, Figure 2(a) demonstrates that the ML classifier with a 1024B receiver node tolerates packet loss better when the packet loss rate is less than 1%. However, we are not able to make any conclusion on the influence of the buffer size on the classification accuracy in the general case when the packet loss rate exceeds 1%. In case of YOLOv7, we can see from Figure 2(a) that this model is slightly more robust to DQ degradation due to packet loss than Rekognition when the packet loss rates are less than 5%. However, as in the case with AWS Rekognition, beyond packet loss rate of 1% it is hard to tell whether an increase in resources in the receiving node actually helps the model to make better classification decisions.

Figure 2(b) represents the average confidence scores of the classification decisions made by AWS Rekognition as dependent on the packet loss rate. The figure clearly demonstrates that Rekognition is more confident in its classifications when a buffer size of the receiver is bigger.


(a)

(b)

(c)

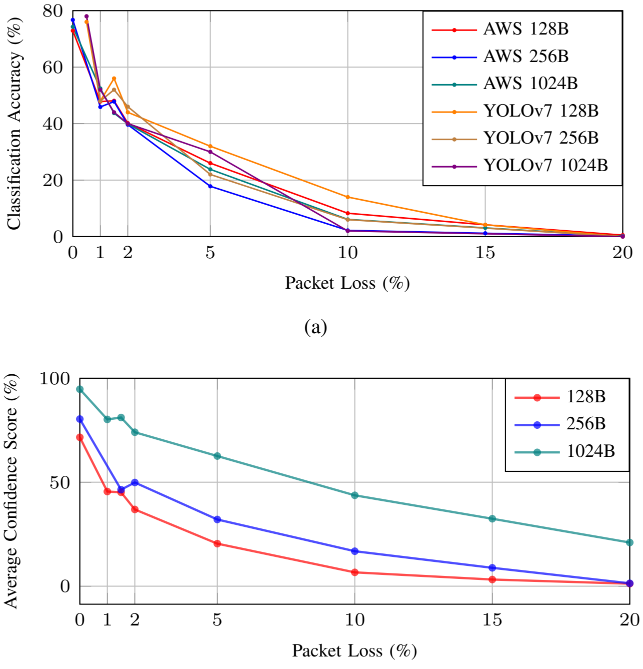
Fig. 1. Example frames from videos affected by network QoS degradation: (a) – original video; (b) – 0.5% packet loss and 512B receiver buffer; (c) – 0.5% packet loss and 1024B receiver buffer;

B. Video Files Classification Case

Figure 3(a) represents the classification accuracy of AWS Rekognition for video data depending on the packet loss rate for the "Traffic Lights" and the "Road Sign" image categories. This figure shows that for the "Traffic Lights" class even a 0.2% packet loss rate is sufficient to drop the performance of AWS Rekognition by 20% for a model with a 1024B receiver and by almost 30% for a model with a 512B receiver. A similar trend is captured for the "Road Sign" class. With a 0.2% rate of packet loss we can observe a drop of more than 20% for a model with a 1024B receiver and a drop of more than 30% for a model with a 512B receiver. Based on this, we can conclude that AWS Rekognition is more sensitive to data loss in the "Road Sign" objects.

Figure 3(b) captures the confidence scores provided by AWS Rekognition when packet loss rates vary. For example, Figure 3(b) shows the confidence score decline of 8% for a packet loss rate of 0.2% on the "Traffic Lights" class with a buffer size of 512B and a decline of 6% for the same packet loss rate and a buffer size of 1024B. Similarly, for the "Road Sign" class we can observe the confidence score decline of 20% for a packet loss rate of 0.2% with a buffer size of 512B and a drop of

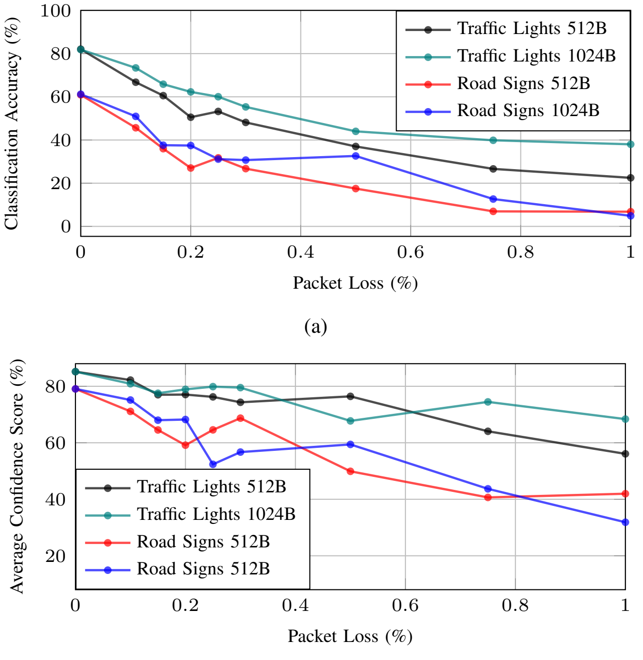
[4] https://www.v7labs.com/open-datasets/open-image-v6



(a)

(b)

Fig. 2. Performance comparison between AWS Rekognition and YOLOv7 over the images classification: (a) – classification accuracy; (b) – average confidence score (only provided by AWS Rekognition)



(a)

(b)

Fig. 3. Performance demonstrated by AWS Rekognition over the videos for "Traffic Lights" and "Road Sign" labels: (a) – classification accuracy; (b) – average confidence score

15% for the same packet loss rate and a buffer size of 1024B. We observe that the drop in the confidence scores is not as significant as the drop in the actual classification accuracy of AWS Rekognition which confirms the higher robustness of the confidence score to the data loss.

V. Conclusion

In this paper, we investigated the impact of data affected by network QoS variations on the performance demonstrated by ML image classifiers, such as AWS Rekognition and YOLOv7, in the real ITS use case. Our results show that even a small packet loss rate, around 1% for images, and an even smaller rate of around 0.2% for videos, resulted in a drastic decline in classification accuracy demonstrated by AWS Rekognition and YOLO. This decline could be explained by the pre-training of image and video ML classifiers on high quality imaging data sets with no corruptions or quality degradation. Consequently, the performance showed by both industrial and open-source classifiers we investigated is unacceptable in most industrial domains. They cannot be employed as an "out of the box" solution to provide effective decisions in real applications like ITS, where the input data quality may vary frequently and dramatically due to numerous factors. The specific DQ variation tolerance thresholds for a particular image and video classifier depend on the concrete scenario and the application requirements. As a general recommendation, we suggest employing the target data for an image classifier re-training, which should include data of various quality instead of using the service as it comes "straight off the shelf".

References

[1] J. Breen, A. Buffmire, J. Duerig, K. Dutt, E. Eide, A. Ghosh, M. Hibler, D. Johnson, S. K. Kasera, E. Lewis *et al.*, "Powder: Platform for open wireless data-driven experimental research," *Computer Networks*, p. 108281, 2021.

[2] S. Chuprov, I. Khokhlov, L. Reznik, and S. Shetty, "Influence of transfer learning on machine learning systems robustness to data quality degradation," in *2022 International Joint Conference on Neural Networks (IJCNN)*, 2022, pp. 1–8.

[3] S. Chuprov, S. Mahajan, R. Zatsarenko, and L. Reznik, "Are industrial ml image classifiers robust to withstand adversarial attacks on videos?" in *2023 Western New York Image and Signal Processing Workshop (WNYISPW)*, 2023, pp. 1–4, unpublished, submitted to this workshop.

[4] S. Chuprov, M. Memon, and L. Reznik, "Federated learning with trust evaluation for industrial applications," in *2023 IEEE Conference on Artificial Intelligence (CAI)*. IEEE, 2023, pp. 347–348.

[5] S. Chuprov, L. Reznik, and G. Garegin, "Study on network importance for ml end application robustness," in *2023 International Conference on Communications (ICC), Rome, Italy*. IEEE, 2023, pp. 1–8.

[6] S. Chuprov, L. Reznik, A. Obeid, and S. Shetty, "How degrading network conditions influence machine learning end systems performance?" in *The 9th International Workshop on Computer and Networking Experimental Research using Testbeds (CNERT)*. IEEE, 2022, pp. 1–6.

[7] S. Chuprov, A. N. Satam, and L. Reznik, "Are ml image classifiers robust to medical image quality degradation?" in *2022 IEEE Western New York Image and Signal Processing Workshop (WNYISPW)*, 2022, pp. 1–4.

[8] S. Chuprov, I. Viksnin, I. Kim, N. Tursukov, and G. Nedosekin, "Empirical study on discrete modeling of urban intersection management system," *International Journal of Embedded and Real-Time Communication Systems (IJERTCS)*, vol. 11, no. 2, pp. 16–38, 2020.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.