JOURNAL OF
EMERGING INVESTIGATORS

Article

An improved video fingerprinting attack on users of the Tor network

Trisha Srikanth[1], David Lu[2]

[1] Padua Academy, Wilmington, Delaware

[2] Massachusetts Institute of Technology, Cambridge, Massachusetts

SUMMARY

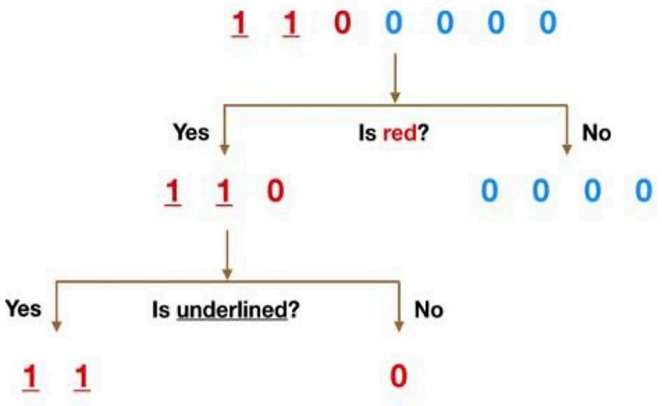
In today's digital age, there is less and less doubt that companies, governments, and hackers track our online patterns and collect personal data. Thus, over 2 million users daily flock to the Tor network, which secures their online identities by encrypting their traffic. Although the Tor network is often considered secure, it is vulnerable to fingerprinting attacks that threaten users' online privacy. Using machine learning, these attacks attempt to classify which web page the victim is visiting based on the page's unique traffic patterns. Since video streaming makes up 80% of total downstream web traffic, we aim to explore how well this content can be fingerprinted in Tor. In this paper, we develop a new video fingerprinting model. Our model is based on a random forest classifier, a supervised machine learning algorithm that assembles decision trees for various samples and classifies based on their majority vote. Our model uses 247 features from video traces to exploit the burst patterns present in video traffic that are unique to each video. Our model is able to distinguish which one of the 50 videos a user is hypothetically watching on the Tor network with 85% accuracy, which outperforms the state-of-the-art, Deep Fingerprinting model's accuracy of 55%. This demonstrates that video fingerprinting poses a serious threat to the privacy of Tor users. Our model performs better as it is adjusted to consider the bursts that are streamed from video traffic's DASH protocol.

INTRODUCTION

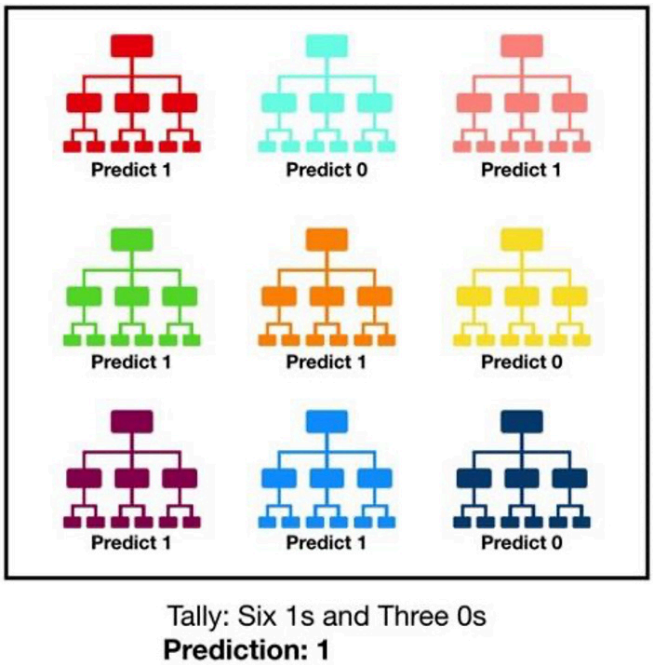
In today's digital age, online privacy is becoming an increasingly important concern, with 4.66 billion active users in 2021 (1). Many internet users have turned to anonymous networks such as Tor to protect their online identities. Accessed by over 2.5 million users daily, Tor protects users' online identities by encrypting their internet traffic and passing it through a series of nodes, or intermediary routers, before reaching the websites' destination (2-3). These nodes act as a three-layer proxy, in which Tor connects at random to one of the publicly listed entry nodes, which is the traffic's entry point into the Tor network. Tor then directs the traffic through a randomly selected middle node and expels the traffic through the third and final exit node (4). Eavesdroppers - hackers or individuals attempting to steal data- cannot directly pinpoint both the user and the website destination during this communication channel and are thus unable to link both ends.

However, website fingerprinting enables adversaries to bypass these safety measures and attack Tor users (2-7). In website fingerprinting, the actor attempts to recognize the encrypted traffic patterns of the specific web page that the targeted user visits, exploiting the fact that the network traffic of each webpage has its own unique pattern (8). These patterns can be learned by a machine learning classifier to categorize websites the victim visits. This attack takes place between the targeted user and the entry node (8). As such, website fingerprinting allows actors to collect information and draw inferences about a user.

There is already appreciable research on website fingerprinting in Tor. Recently, however, Rahman *et al*. investigated whether the techniques from website fingerprinting can also be used to perform video fingerprinting (i.e., identify which videos Tor users are accessing) (7). This question is particularly important because one of the largest sources of web traffic is video streaming, which is made up approximately 60% of the total volume of downstream traffic in 2019 and is projected to make up 82% of all internet traffic by 2022 (3, 6). Rahman *et al*. fine-tuned a convolutional neural network for video fingerprinting and leveraged it to classify videos using the raw packet sequence of traffic instances (7). Their model correctly classified which one of 50 videos a user was accessing with 55% accuracy, which is quite low compared to the performance of website fingerprinting models, which have achieved over 90% accuracy (2, 7-11). As a result, it remains to be seen whether video fingerprinting is a significant threat to users of Tor. If video fingerprinting attacks



Decision Tree Example (12)



Random Forest Model Making a Prediction (12)

can achieve higher accuracies, this would pose a threat to many Tor users who may use the platform for video services.

In this paper, we propose a new video fingerprinting model based on random forests, as it is flexible, simple to use, and less computationally expensive. The random forest classifier relies on a powerful concept: wisdom of the crowds. It contains numerous distinct decision trees that together function as ensemble. Each tree continuously splits data based on a certain criterion until it arrives at a classification. The classification receiving the most votes becomes the classifier's final prediction.

Our model employs over 200 features, which use the characteristics of video streaming traffic. Every video's traffic from our data has unique features - burst size, number of packets, etc.- that differentiate the videos apart; our model utilizes them to make its video predictions.

Our model can correctly identify which 1 of 50 videos a user is accessing with 85% accuracy. This is a substantial improvement to previous work and demonstrates that video fingerprinting is a much larger threat to Tor users' privacy than previously imagined. We also evaluate the effectiveness of our attacking various training scenarios and analyze which features are most important for video fingerprinting. Actors could identify users' identities through the videos they view, thus demonstrating the security and privacy risk this attack poses if exploit.

RESULTS

We extracted a number of features to use in our random forest model, trained the model on test data, and computed its accuracy. We used Rahman *et al.*'s dataset in our evaluation, which consists of traffic generated from YouTube music videos

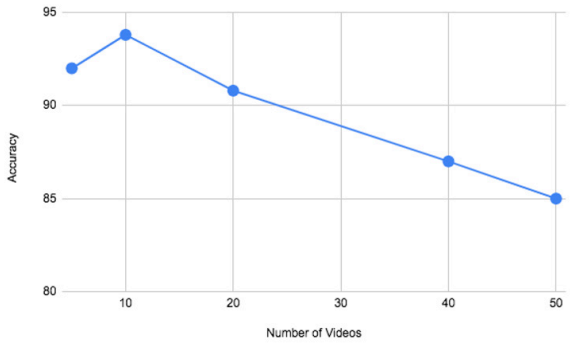


Figure 1: Line graph depicting the model accuracy as the number of videos increases. There is a negative correlation.

(7). In our experiments, we evaluated our model in a closed world setting. This means we assumed that the user cannot access videos which the adversary has not seen before (i.e., there does not exist labels in the testing data which are not present in the training data).

After running our random forest classifier in a closed world setting with 50 total videos, we obtained an accuracy of 85%. This accuracy is significantly higher than the state-of-the-art accuracy of 55%.

Varying the Size of the Closed World

We also investigated how our model performed when we varied the size of the closed world (i.e., the number of videos between which the classifier must distinguish). As the size of the closed world increased, the model's test accuracy decreased. With only 10 videos in the closed world, the model achieved a test accuracy of 93.8%. This value decreased as more videos were added to the dataset; the accuracy of distinguishing between 50 videos was 85% (**Figure 1**). This trend occurred as it is harder for the model to classify a trace as one of 50 videos as opposed to one of a few videos. There are more traces to differentiate.

Varying the Number of Training Instances

We also varied the number of training instances (i.e., the number of examples that the classifier trains on per video) while fixing the size of the closed world to be 50. We found that as the number of training instances increased, the accuracy increased. With only 50 training instances per video, the closed world accuracy was 74.8%; with 100, it was 80.4%; with 200, it was 82.5%; with 400, it was 83.9%; lastly, with 576, it was 85% (**Figure 2**).

Most Important Features

Features are independent variables that models use to make their predictions. The quality of such features has a large influence on the quality of insights obtained. Below, we highlighted the top 20 most important features for our model to gauge which traffic features are significant to video fingerprinting.

We assigned scores to each feature as an indication of

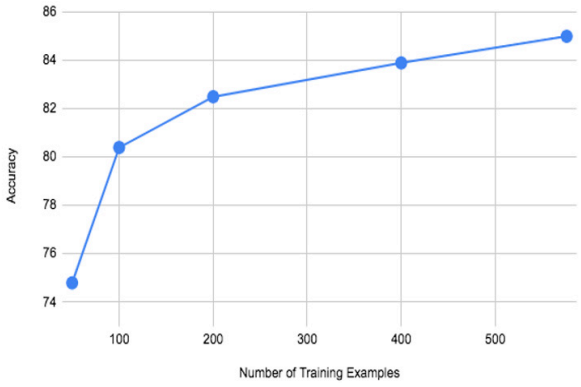


Figure 2: Line graph depicting the model accuracy as the number of training instances increases. There is a positive correlation.

their relative significance when making a prediction, where higher scores suggest more significance. These scores were obtained from the feature importance field of the random forest classifier we trained using sklearn (13).

An "outgoing burst packets" feature refers to the number of packets sent from the user to the video server during a specific 0.1 second interval (a burst). Similarly, an "incoming burst packets" feature refers to the number of packets sent from the server to the user during a 0.1 second interval. A "burst packets" feature refers to the total number of packets sent during a burst. It appears that these burst-base features were important to the classifier during training (especially those in bursts 15-20). Transmission time was also an important feature. Transmission time refers to the time it takes for each video to transmit all of the packets from the server to the user.

Features that were more important played a larger role in the model's video prediction. We found that the most important features were the bursts (**Table 1**). Cumulative features, like the total number of packets, were not as important (**Table 1**).

Feature	Importance	Burst Number
outgoing burst packets	0.0236	16
incoming burst packets	0.0231	16
burst packets	0.0223	17
burst packets	0.0220	16
incoming burst packets	0.0215	15
incoming burst packets	0.0208	17
burst packets	0.0199	15
outgoing burst packets	0.0198	17
burst packets	0.0179	18
burst packets	0.0179	18
outgoing burst packets	0.0170	18
incoming burst packets	0.0161	18
burst incoming divided by total	0.0154	20
incoming burst packets	0.0153	12
burst packets	0.0151	19
burst incoming divided by total	0.0146	18
burst packets	0.0146	12
burst incoming divided by total	0.0145	21
burst packets	0.0136	11
transmission time	0.0134	N/A

Table 1: Table showing top 20 most important features, ordered by feature importance.

DISCUSSION

Our random forest classifier achieved an accuracy of 85%, which was higher than the state-of-the-art accuracy of 55%. One potential reason is that video traffic, which is bursty, is quite different from typical webpage traffic, so using a fine-tuned model pre-trained on website fingerprinting instances may not yield optimal results. Our random forest model, which uses features that are relevant to video traffic sequences, may be better suited to this classification problem. Additionally, our model accuracy is positively correlated with the number of training instances used. This trend is expected because a larger amount of training data enables a classifier to learn more details about each class and therefore achieve higher accuracies. However, it is interesting to find that even with a small amount of training data, our classifier still manages to achieve a high accuracy. The most important features extracted were the burst features. Burst features were more valuable because burst patterns are essentially what makes each video stream unique. The first burst of every video is the client's buffer, so beginning bursts, which are those from 1 to 10, are not as dependent on their videos. Although video fingerprinting has long been overlooked, our results show that it is more of a threat than previously thought. There is considerable research published on website fingerprinting in Tor, from proposing new models of attacks or defenses to simply evaluating the models that have been proposed (11-12). However, video fingerprinting is potentially more invasive. If combined with a website fingerprinting model, video fingerprinting can not only identify the service a user is visiting, such as YouTube, but also identifies which specific video a user is watching. This is particularly concerning since video streaming is a major source of online traffic (6). Attackers may take advantage of this situation to carry out video fingerprinting attacks on Tor users, who are cognizant of the lack of research done on video fingerprinting. Therefore, it is critical that we soon implement defenses to counter these attacks. To that end, more research is needed to delve into the issue of video fingerprinting. With our results on the size of the closed world, the number of training instances, and the feature importance, we hope to provide valuable insight that researchers could use to further the study of video fingerprinting on Tor.

That being said, our attack may not be feasible in the real world just yet. We trained our data in the closed world scenario, where the classifier attempted to distinguish video traffic patterns only from a specific set of traffic patterns. The more realistic open world scenario involves training the classifier to identify a few traffic patterns out of a much larger set of traffic patterns, most of which it has never seen before (8). Real world attacks would take place in the open world, but this scenario is considerably more complex than the closed world scenario and would best be implemented after

the novel classifier attains more experience and modification. This may be accomplished by training on more instances and modifying as necessary until the classifier is experienced enough to identify, rather than merely distinguish, video traffic. If future work on video fingerprinting proves to be successful in the open world, privacy on Tor may indeed be threatened, with video fingerprinting joining the long list of current cyber-attacks on user anonymity.

In future investigations, we plan to modify our classifier to be tested in the more realistic open-world scenario, possibly incorporating unsupervised learning. Unsupervised learning allows the model to learn patterns on its own, a method very suitable for the open-world scenario. We will also test our classifier against existing website fingerprinting defenses or potentially new video fingerprinting-specific defenses.

MATERIALS AND METHODS

Dataset Generation

The dataset that we obtained includes traffic generated by loading popular music videos from YouTube (7). There were 50,000 total instances for these videos, and the video length was restricted to approximately 3 minutes so that the videos were not trivially differentiated by their load times. The videos had the best resolution for the stream during playback, as this is the default setting that most users would maintain. Because of regional restrictions, several video loading attempts were blocked, so captures containing fewer than 3000 packets were removed.

Video Feature Extraction

To extract features from the videos, we used knowledge concerning the nature of video traffic. Video streams have unique bursts, which are short bursts of traffic packets. If their traffic patterns are correlated with their content, then an attacker who can measure these bursty patterns may be able to identify the video (14). Transport-layer encryption hides the content of video streams, but not the traffic's characteristics, like how bursty it is. Video streams have a Dynamic Adaptive Streaming over HTTP (DASH) protocol, which aims to maximize the quality of experience as well as support the interoperability of popular streaming technologies. DASH standardizes these bursty streams by storing the content in segment-files on the server (14). Each file contains a particular encoding of one segment, and so when a streaming session is initiated, the server references the time segments and available encodings to the client. The client then requests for individual segments depending on the presentation plan (14).

To extract features for our video fingerprinting classifier, we exploited the unique characteristics of these bursts, or segments, that were streamed from DASH-video. First, we extracted a number of high-level features. These features included the number of traffic packets that were transmitted from the user to the website destination (incoming packets) and the number of packets transmitted from the destination to the user (outgoing packets). Additionally, they contained the packets' total transmission time for each video and the number of packets transmitted in each 0.1 second time interval burst. We also extracted the number of incoming and outgoing packets in each burst, as well as the number of incoming and outgoing packets in each burst divided by the total number of packets in each burst. Furthermore, we obtained the number of incoming packets transmitted between each outgoing packet, the number of outgoing packets transmitted between each incoming packet, and the total number of packets transmitted per second. Altogether, we obtained 247 features for each video.

Constructing the Classifier

After feature engineering, we explored various machine learning classifiers, like random forests, neural networks, adaptive boosting, and nearest neighbors algorithms. Due to the random forest model's high accuracy compared to other classifiers, we decided to focus only on random forests in our analysis.

We constructed a machine learning classifier using the random forest model. The classifier's parameters included 100 trees, a minimum sample split of 2, and a minimum sample leaf of 1, for video fingerprinting purposes. The minimum sample split parameter is the minimum number of observations in any node in order to split it, while the minimum sample leaf parameter is the minimum number of samples required to be in a node (5).

Within the classifier, we created a dataframe for the features and labels of the data. The data was split into training and testing data, with 90% being training data and 10% being testing data. After training in a closed world scenario, we ran the classifier on the test data and computed its accuracy, creating a confusion matrix to compare the videos the algorithm predicted with the actual videos. We also determined which features of the videos were the most important using feature importance based on Gini impurity.

ACKNOWLEDGMENTS

We would like to thank the SSI (Summer STEM Institute) team for supporting our research.

Received: August 4, 2021
Accepted: November 17, 2021
Published: March 31, 2022

REFERENCES

1. WebsiteSetup Team. "Internet stats & facts." *WebsiteSetup*, 2021.
2. Bhat, Sanjit, *et al*. "Var-CNN: A Data-Efficient Website Fingerprinting Attack Based on Deep Learning." *Proceedings on Privacy Enhancing Technologies*, vol. 2019, no. 4, 2019, pp. 292–310., doi.org/10.2478/popets-2019-0070.
3. Evans, Chris, *et al*. "The sustainable future of video

entertainment: From creation to consumption." *Futuresource Consulting*, August 2020, p. 9.
4. Porup, J. M. "What is the tor browser? And how can it help protect your identity." *CSO*, October 2019, www.csoonline.com/article/3287653/what-is-the-tor-browser-how-it-works-and-how-it-can-help-you-protect-your-identity-online.html#:~:text=Tor%20Browser%20connects%20at%20random,third%20and%20final%20exit%20node.&text=As%20a%20result%2C%20don't,you%20in%20a%20foreign%20tongue.
5. Meinert, Reilly. "Optimizing hyperparameters in random forest classification: What hyperparameters are, how to choose hyperparameter values, and whether or not they're worth your time." *Towards Data Science*, 5 June 2019, www.towardsdatascience.com/optimizing-hyperparameters-in-random-forest-classification-ec7741f9d3f6.
6. NCTA. "Report: Where does the majority of internet traffic come from?" *NCTA — The Internet Television Association*, 17 October 2020, www.ncta.com/whats-new/report-where-does-the-majority-of-internet-traffic-come.
7. Wright, Matthew *et al*. "Poster: Video fingerprinting in tor." *ResearchGate*, 2019 November, DOI: 10.1145/3319535.3363273.
8. Fukui, Takuya. "De-anonymizing tor traffic with website fingerprinting." *Towards Data Science*, 22 April 2017, www.witestlab.poly.edu/blog/de-anonymizing-tor-traffic-with-website-fingerprinting/.
9. Panchenko, Andriy *et al*. "Website fingerprinting at internet scale." *Semantic Scholar*, 1 February 2016, DOI:10.14722/NDSS.2016.23477.
10. Hayes, Jamie, and Danezis, George. "k-fingerprinting: A robust scalable website fingerprinting technique". *25th USENIX Security Symposium (USENIX Security 16)*, 2016, pp. 1187–1203.
11. Wang, Tao *et al*. "Effective Attacks and Provable Defenses for Website Fingerprinting." *23rd USENIX Security Symposium (USENIX Security 14)*, pp. 143–157, 2014.
12. Yiu, Tony. "Understanding Random Forest: How the Algorithm Works and Why it Is So Effective." *Towards Data Science*, 12 June 2019.
13. Pedregosa *et al*. "Scikit-learn: Machine Learning in Python." *JMLR* 12, 2011, pp. 2825-2830, 2011.
14. Schuster, Roei, *et al*. "Beauty and the Burst: Remote Identification of Encrypted Video Streams." *USENIX*, USENIX Association, 1 January 1970, https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/schuster.