BÁO CÁO TỔNG KẾT ĐỒ ÁN MÔN HỌC

**Môn học: Cơ chế hoạt động của mã độc**

**Tên chủ đề:** A Superficial Analysis Approach for Identifying Malicious Domain Names Generated by DGAMalware

*Mã nhóm: G6 Mã đề tài: S39*

**Lớp**: NT230.O21.ANTT

1. **THÔNG TIN THÀNH VIÊN NHÓM:**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** | **Email** |
| 1 | Nguyễn Long Vũ | 21522800 | 21522800@gm.uit.edu.vn |
| 2 | Lê Huy Hiệp | 21522067 | 21522067@gm.uit.edu.vn |
| 3 | Nguyễn Việt Khang | 21522198 | 21522198@gm.uit.edu.vn |

1. **TÓM TẮT NỘI DUNG THỰC HIỆN:[[1]](#footnote-1)**
   1. **Chủ đề nghiên cứu trong lĩnh vực Mã độc:** *(chọn nội dung tương ứng bên dưới)*

þ Phát hiện mã độc

¨ Đột biến mã độc

¨ Khác: …………………………………

* 1. **Liên kết lưu trữ mã nguồn của nhóm:**

Mã nguồn của đề tài đồ án được lưu tại: ………………………………………….

*(Lưu ý: GV phụ trách phải có quyền truy cập nội dung trong Link)*

* 1. **Tên bài báo tham khảo chính:**

|  |
| --- |
| Satoh, A., Fukuda, Y. (Member, IEEE), Hayashi, T., and Kitagata, G., "A Superficial Analysis Approach for Identifying Malicious Domain Names Generated by DGA Malware" IEEE Open Journal of the Communications Society, vol. 1, pp. 1837-1849, Nov. 2020.  doi: 10.1109/OJCOMS.2020.3038704 |

* 1. **Dịch tên Tiếng Việt cho bài báo:**

|  |
| --- |
| Một phương pháp tiếp cận phân tích hình thái để xác định các tên miền độc hại được tạo ra bởi DGAMalware |

* 1. **Tóm tắt nội dung chính:**

|  |
| --- |
| Một trong những mối đe dọa an ninh nghiêm trọng nhất đối với mạng máy tính là phần mềm độc hại. Để ngăn chặn thiệt hại do phần mềm độc hại gây ra, các quản trị viên phải nhanh chóng xác định và loại bỏ các thiết bị bị nhiễm trong mạng của họ. Tuy nhiên, nhiều họ phần mềm độc hại sử dụng thuật toán tạo tên miền (DGA) để tránh bị phát hiện. DGA là kỹ thuật trong đó tên miền được thay đổi thường xuyên để che giấu liên lạc giữa máy bị nhiễm và máy chủ điều khiển.  Nhóm tác giả đề xuất một phương pháp ước tính tính ngẫu nhiên của các tên miền bằng cách phân tích hình thái chuỗi ký tự của chúng. Phương pháp này dựa trên quan sát rằng tên miền lành tính do con người tạo ra thường phản ánh ý định cụ thể của người đăng ký, chẳng hạn như tên tổ chức, sản phẩm hoặc nội dung. Ngược lại, các tên miền độc hại do DGA tạo ra thường bao gồm các chuỗi ký tự vô nghĩa để tránh xung đột với tên miền đã đăng ký trước; do đó, có sự khác biệt rõ ràng trong chuỗi ký tự giữa tên miền tạo động và tên miền do con người tạo ra.  Phương pháp này khai thác sự khác biệt này để phân biệt giữa tên miền lành tính và độc hại, giúp nâng cao hiệu quả trong việc phát hiện và ngăn chặn các mối đe dọa từ phần mềm độc hại. |

* 1. **Tóm tắt các kỹ thuật chính được mô tả sử dụng trong bài báo:**

|  |
| --- |
| 1. Giảm nhiễu (Noise reduction) **Vai trò:** Bước giảm nhiễu nhằm xác định và loại bỏ các tên miền lành mạnh và những tên miền cấu hình sai lệch từ danh sách tên miền ban đầu. Điều này giúp tinh giản dữ liệu, chỉ giữ lại những tên miền có khả năng cao là do DGA tạo ra.  **Nhiệm vụ:**   * **So sánh với danh sách trắng:** Xác định các tên miền lành mạnh bằng cách so sánh với các danh sách trắng. Các danh sách trắng bao gồm các tên miền được sử dụng bởi các ứng dụng hoặc dịch vụ cụ thể gây ra số lượng lớn thông báo NXDOMAIN. * **Loại bỏ tên miền cấu hình sai lệch:** Các tên miền vi phạm các quy định DNS được coi là kết quả của cấu hình sai lệch sẽ bị loại bỏ khỏi danh sách. Điều này bao gồm các tên miền không tuân thủ các tiêu chuẩn DNS. * **Chuyển tiếp tên miền nghi ngờ:** Các tên miền còn lại sau khi lọc bỏ được coi là có thể được tạo ra bởi DGA và sẽ được chuyển đến các bước phân tích tiếp theo.  2. Lựa chọn tên miền con (Subdomain selection) **Vai trò:** Lựa chọn tên miền con nhằm xác định các tên miền một cách hiệu quả bằng cách chọn tên miền con (subdomain) có chuỗi dài nhất từ miền (domain) d. Điều này tận dụng quan sát rằng các thuật toán sinh tên miền tự động (DGA) thường tạo ra các tên miền dài để tránh xung đột với các tên miền đã được đăng ký trước đó.  **Nhiệm vụ:**   * **Xác định chuỗi dài nhất trong subdomain:** Đối với mỗi tên miền d, xác định chuỗi dài nhất trong các cấp độ subdomain.   Ví dụ, khi miền d là "xjjjvqpoh.com.ai", chuỗi lớn nhất là "xjjjvqpoh" ở subdomain d3.   * **Phân tích cấu trúc tên miền:** Tận dụng đặc điểm rằng DGA thường tạo ra các tên miền dài hơn để tránh xung đột với các tên miền ngắn hơn, vốn có khả năng cao hơn là đã được các tổ chức hợp pháp sử dụng. * **Lọc và chọn lọc:** Giữ lại các tên miền có subdomain dài nhất để chuyển đến các bước phân tích tiếp theo, loại bỏ các tên miền ngắn hơn có khả năng là hợp pháp.  3. Ước lượng dựa trên từ điển (Dictionary-Based Estimation) **Vai trò:** Ước lượng dựa trên từ điển sử dụng phương pháp dựa trên từ điển để ước lượng tính ngẫu nhiên của các miền con. Phương pháp này giúp xác định xem miền con có phải là do con người tạo ra hay là kết quả của thuật toán sinh tên miền tự động.  **Nhiệm vụ:**   * **Phân nhóm chuỗi miền con:** Chuỗi miền con *did\_i*di được chia thành các nhóm từ *ww*w bằng cách sử dụng từ điển. Các nhóm từ này đại diện cho các từ có nghĩa trong ngôn ngữ tự nhiên. * **Ước lượng tính ngẫu nhiên:** Tính ngẫu nhiên của miền con *did\_i*di được ước lượng từ các đặc điểm của nhóm từ *ww*w. Các đặc điểm này bao gồm tần suất xuất hiện của các từ trong từ điển, độ dài của các từ, và tính liên kết giữa các từ. * **Phân loại miền con:** Nếu kết quả cho thấy miền con *did\_i*di là một chuỗi ký tự có ý nghĩa (tức là, các từ trong nhóm *ww*w có ý nghĩa trong ngôn ngữ tự nhiên), thì miền *dd*d, chứa miền con này, được xác định là một miền lành mạnh do con người tạo ra. * **Tiếp tục phân tích:** Các miền con không có ý nghĩa sẽ được tiếp tục phân tích trong các bước tiếp theo để xác định tính độc hại.  4. Ước lượng dựa trên tìm kiếm trên Web (Web-search-based estimation) **Vai trò:** Ước lượng dựa trên tìm kiếm trên Web sử dụng các công cụ tìm kiếm để thu thập thông tin bổ sung từ các nguồn trực tuyến về các tên miền đã được xác định. Điều này giúp xác nhận và cập nhật thông tin, đảm bảo độ chính xác và tin cậy của kết quả phân tích.  **Nhiệm vụ:**   * **Thực hiện truy vấn tìm kiếm:** Sử dụng các công cụ tìm kiếm để tìm kiếm thông tin về các tên miền trong danh sách đã lọc, thu thập dữ liệu từ các nguồn trực tuyến. * **Thu thập dữ liệu bổ sung:** Lấy thông tin từ các kết quả tìm kiếm để kiểm chứng và bổ sung cho dữ liệu hiện có, bao gồm các bài viết, báo cáo, hoặc nhận xét từ các nguồn đáng tin cậy. * **Xác thực thông tin:** So sánh và đối chiếu thông tin thu thập được từ web với dữ liệu ban đầu để đảm bảo tính chính xác và độ tin cậy của các tên miền đã được phát hiện. * **Cập nhật và điều chỉnh:** Dựa trên các thông tin mới nhất từ web, cập nhật và điều chỉnh danh sách tên miền độc hại để đảm bảo rằng dữ liệu luôn được cập nhật và phản ánh chính xác tình hình hiện tại.   Bốn bước này kết hợp với nhau tạo thành một quy trình phân tích toàn diện để xác định các tên miền độc hại do DGA tạo ra, từ việc làm sạch và chuẩn hóa dữ liệu, lựa chọn các tên miền quan trọng, sử dụng các từ điển chuyên ngành, đến việc thu thập và xác thực thông tin từ các nguồn trực tuyến. Quy trình này đảm bảo tính chính xác, hiệu quả và độ tin cậy cao trong việc phát hiện và phân loại các tên miền độc hại. |

* 1. **Môi trường thực nghiệm của bài báo:**

|  |
| --- |
| * Cấu hình máy tính:   + CPU: Xeon Silver 411- (8core 2.10Hz)  + GPU: NVIDIA GeForce RTX 2080 Ti (11GB GDDR6)  + RAM: 96GB DDR4-2666  + SSD: Seq. Read and Write up to 560MB/sec and 530MB/sec  + Kernel: Linux 3.10.0-957.1.3.el7.x86\_64  + Software TensorFlow 2.0.0, CUDA Toolket 10.0, cuDNN 7.6.5   * Các công cụ hỗ trợ sẵn có: … <dùng cho giai đoạn nào> * Ngôn ngữ lập trình để hiện thực phương pháp: Python * Đối tượng nghiên cứu (chương trình phần mềm dùng để kiểm tra tính khả thi của phương pháp/tập dữ liệu – nếu có): ….<tác giả dùng tập dữ liệu ABC bao gồm 500 chương trình phần mềm chứa lỗ hổng bảo mật XYZ,….> * Tiêu chí đánh giá tính hiệu quả của phương pháp: <vd: số lượng lỗ hổng được phát hiện, thời gian chạy, coverage….> |

* 1. **Kết quả thực nghiệm của bài báo:**

|  |
| --- |
| **Kết luận về nghiên cứu của bài báo**  **1. Phương pháp đề xuất:**   * Phương pháp đề xuất nhằm xác định các tên miền độc hại được tạo động (dynamically generated malicious domains) bằng cách ước tính tính ngẫu nhiên của chuỗi ký tự của chúng. * Quan sát rằng tên miền lành tính do con người tạo ra thường phản ánh ý định của người đăng ký tên miền (ví dụ: tên tổ chức, sản phẩm, hoặc nội dung). * Ngược lại, các tên miền độc hại được tạo động thường bao gồm các chuỗi ký tự vô nghĩa để tránh xung đột với các tên miền đã được đăng ký trước đó. Do đó, có sự khác biệt rõ ràng trong chuỗi ký tự giữa tên miền do con người tạo ra và tên miền được tạo động.   **2. Hiệu quả của phương pháp:**   * Phương pháp này sử dụng tính ngẫu nhiên để phát hiện phần mềm độc hại mà không cần kiến thức trước về các thuật toán tạo tên miền độc hại (DGAs). * Trong các thí nghiệm, phương pháp đạt được độ nhạy (recall) và độ chính xác (precision) lần lượt là 0.9960 và 0.9029 khi sử dụng các tập dữ liệu đã gán nhãn. * Phương pháp cũng cho thấy hiệu quả cao khi áp dụng với các tập dữ liệu thu thập từ mạng lưới trong khuôn viên trường đại học.   **3. Lợi ích của phương pháp:**   * Phương pháp này giúp xác định nhanh chóng và loại bỏ các máy bị nhiễm phần mềm độc hại khỏi mạng lưới bằng cách sử dụng các truy vấn DNS cho các tên miền độc hại được phát hiện. * Đảm bảo ba cấp độ hiệu quả: * Khả năng hoạt động chỉ bằng cách tham chiếu các nhật ký hệ thống được ghi lại bởi RDNSs, không bị giới hạn bởi các mạng có giao tiếp mã hóa. * Loại bỏ vấn đề về sự thay đổi khái niệm trong các tập dữ liệu khi so sánh với các kỹ thuật phát hiện dựa trên ML và DL, giúp duy trì mức hiệu suất cao. * Được sử dụng trong nhiều kịch bản khác nhau do tính linh hoạt cao của phương pháp này. * Có thể tích hợp một số khía cạnh của phương pháp này vào các phương pháp đã được công bố khác để cải thiện hiệu suất phát hiện phần mềm độc hại.   **4. Hướng nghiên cứu tương lai:**   * Đánh giá hiệu quả của phương pháp đề xuất đối với các giao tiếp được quan sát trong thời gian dài trong các mạng lưới lớn. * Nghiên cứu tác động khi kết hợp kỹ thuật này với các phương pháp khác để cải thiện hiệu suất phát hiện phần mềm độc hại. |

* 1. **Công việc/tính năng/kỹ thuật mà nhóm thực hiện lập trình và triển khai cho demo:**

|  |
| --- |
| **Công việc mà nhóm thực hiện cho đề tài dựa trên phân tích phương pháp nhóm đã tham khảo trong bài báo**  1. Thực hiện lựa chọn tên miền phụ đã được gán nhãn.  2. Thực hiện lựa chọn tên miền con  3. Giảm nhiễu các tên miền này thông qua danh sách whitelist và blacklist  4. Sau khi có kết quả từ bước 2, nhóm tiếp tục lấy các tên miền này thực hiện quá trình lọc các tên miền này thông qua từ điển  5. Tiếp tục thực hiện quá trình lọc không qua tìm kiếm web và đưa ra kết luận với các label đã được gán ban đầu, từ đó đưa ra được tính chính xác của mô hình.  **Công việc mà nhóm đã thực hiện cho đề tài này**  1. Thực hiện lựa chọn tên miền phụ đã được gán nhãn.  2. Thực hiện lựa chọn tên miền con  3. Giảm nhiễu các tên miền này thông qua danh sách whitelist và blacklist và gán nhãn cho chúng ( good, bad và unknown )  4. Sau khi có kết quả từ bước 2, nhóm tiếp tục lấy các tên miền này thực hiện quá trình lọc các tên miền này thông qua từ điển  5. Tiếp tục thực hiện quá trình lọc không qua tìm kiếm web và đưa ra kết luận với các label đã được gán ban đầu, từ đó đưa ra được tính chính xác của mô hình. |

* 1. **Các khó khăn, thách thức hiện tại khi thực hiện:**

|  |
| --- |
| Độ chính xác của mô hình khi nhận diện tên miền độc hại còn khá thấp. Cần phải cải tiến độ chính xác nếu muốn áp dụng vào thực tế để nhận diện các tên miền độc hại. |

1. **TỰ ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH SO VỚI KẾ HOẠCH THỰC HIỆN:**

|  |
| --- |
| 80% |

1. **NHẬT KÝ PHÂN CÔNG NHIỆM VỤ:**

|  |  |  |
| --- | --- | --- |
| **STT** | **Công việc** | **Phân công nhiệm vụ** |
| 1 | Thu thập dataset | Nguyễn Long Vũ |
| 2 | Triển khai chọn các tên miền phụ | Nguyễn Long Vũ |
| 3 | Lọc các tên miền thông qua blacklist và whitelist | Nguyễn Việt Khang |
| 4 | Ước lượng dựa trên từ điển | Nguyễn Long Vũ |
| 5 | Ước lượng dựa trên tìm kiếm web | Lê Huy Hiệp |
| 6 | Thống kê kết quả và đưa ra kết luận | Lê Huy Hiệp |
| 7 | Viết báo cáo, làm slide | Nguyễn Việt Khang |

BÁO CÁO TỔNG KẾT CHI TIẾT

**Phần bên dưới của báo cáo này là tài liệu báo cáo tổng kết - chi tiết của nhóm thực hiện cho đề tài này.**

* 1. **Phương pháp thực hiện**

**I. Vấn Đề và Thách Thức:**

Trong môi trường mạng hiện nay, nguy cơ từ các loại malware, đặc biệt là malware DGA, đang ngày càng trở nên nguy hiểm hơn. Malware DGA thường tạo ra các tên miền độc hại một cách động, tự động thay đổi liên tục để tránh bị phát hiện, điều này làm cho việc phát hiện và ngăn chặn trở nên phức tạp. Trong khi đó, mỗi ngày, có hàng triệu truy vấn DNS được thực hiện trên mạng, tạo ra một lượng lớn dữ liệu cần được phân tích để phát hiện các tên miền độc hại.

Thách thức chính đặt ra là làm thế nào để hiệu quả phát hiện và loại bỏ các tên miền độc hại được tạo ra bởi malware DGA từ các truy vấn DNS một cách tự động và nhanh chóng, đồng thời tránh được các tên miền bất hợp pháp. Điều này đặt ra nhu cầu cho một hệ thống phát hiện mạnh mẽ có khả năng xử lý dữ liệu lớn, đồng thời có thể phân biệt giữa các tên miền độc hại và tên miền hợp pháp một cách chính xác và đáng tin cậy.

Để giải quyết vấn đề này, báo báo đề xuất một phương pháp ước lượng tính ngẫu nhiên của các tên miền bằng cách phân tích sơ bộ chuỗi ký tự của chúng. Phương pháp này dựa trên quan sát rằng các tên miền được tạo ra bởi con người thường phản ánh ý định của người đăng ký tên miền, trong khi các tên miền độc hại được tạo động thường chứa các chuỗi ký tự vô nghĩa. Bài báo không đòi hỏi bất kỳ thông tin trước nào về các thuật toán sinh tên miền (DGA) và thay vào đó, sử dụng một phương pháp dựa trên đơn giản để phân biệt giữa các tên miền hợp pháp và độc hại.

Phương pháp được thiết kế để xử lý một lượng lớn dữ liệu DNS, giảm thiểu số lượng dữ liệu nhiễu và tăng hiệu suất phát hiện. Điều này đặc biệt quan trọng khi áp dụng trong môi trường mạng lớn với hàng triệu truy vấn DNS được thực hiện hàng ngày.

Trong phần tiếp theo , nhom sẽ trình bày chi tiết về phương pháp được đề xuất của bài báo bao gồm cả các bước và thuật toán được sử dụng, cũng như kết quả của các thí nghiệm đã thực hiện để đánh giá hiệu suất của phương pháp này.

**II. Phương pháp thực hiện**

Phương pháp của nhóm được thiết kế để giải quyết vấn đề phát hiện các tên miền độc hại được tạo ra động bằng cách sử dụng các kỹ thuật phân tích sơ bộ chuỗi ký tự. Đây là một phương pháp hiệu quả không yêu cầu bất kỳ thông tin trước nào về các thuật toán sinh tên miền (DGA), mà thay vào đó sử dụng sự khác biệt trong cấu trúc của tên miền để phân biệt giữa các tên miền hợp pháp và độc hại.

Phương pháp bao gồm bốn bước chính:

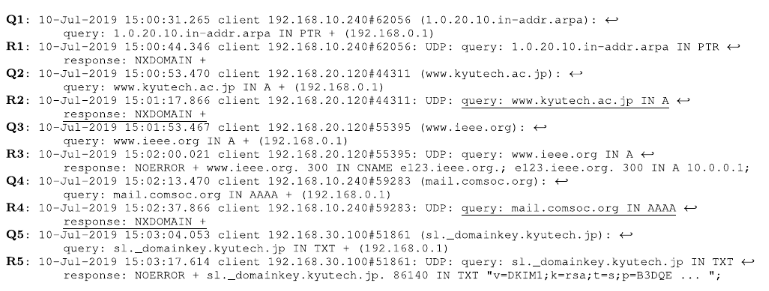
**a) Giảm Nhiễu (Noise Reduction):**

Trong bước này, xác định các tên miền lành tính bằng cách so sánh chúng với danh sách trắng, loại bỏ các tên miền vi phạm các quy định DNS và chỉ giữ lại các tên miền có thể được tạo ra bởi DGA cho các bước phân tích tiếp theo.

Phân đoạn giảm nhiễu trong phương pháp đề xuất là một bước quan trọng nhằm loại bỏ các tên miền không cần thiết và tập trung vào các tên miền có khả năng cao là kết quả của hoạt động của malware. Đây là một bước tiền đề quan trọng để làm giảm khối lượng công việc cần xử lý trong các bước phân tích sau đó, giúp tăng hiệu suất của toàn bộ quá trình phát hiện malware.

Sau khi đã tìm được dataset phù hợp, nhóm tiến hành xác định các tên miền benign thông qua so sánh với các danh sách trắng (whitelist). Các tên miền trong whitelist là những tên miền được sử dụng bởi các ứng dụng hoặc dịch vụ cụ thể gây ra các thông báo NXDOMAIN đại trà, chẳng hạn như phần mềm chống virus endpoint và các danh sách dựa trên DNS (DNSBL).

Bước tiếp theo là loại bỏ các tên miền vi phạm các quy định DNS, được coi là kết quả của việc cấu hình sai lệch và không đáng tin cậy. Các tên miền còn lại được xem xét là có thể được tạo ra bởi DGA và được chuyển đến các bước phân tích tiếp theo.



*Ví dụ về nhật ký truy vấn cho một hệ thống DNS*

**b) Lựa chọn tên miền con (Subdomain Selection):**

Trong bước này, nhóm tiến hành lựa chọn các tên miền con dựa trên nguyên tắc của thuật toán sinh tên miền (DGA). Đặc điểm quan trọng của các tên miền do malware tạo ra là chúng thường có chuỗi ký tự dài và ngẫu nhiên. Bằng cách sử dụng quan sát này, nhóm tận dụng để chọn ra các tên miền con có độ dài lớn nhất từ miền cha. Việc này giúp tăng khả năng bắt giữ các tên miền do malware tạo ra, đồng thời giảm thiểu khả năng bắt nhầm các tên miền benign.

Để xác định miền, bước này chọn tên miền con di có chuỗi dài nhất từ miền d, trong đó i là chỉ mục của cấp độ tên miền con.

Ví dụ, khi miền d là xjjjvqpoh.com.ai, chuỗi dài nhất là xjjjvqpoh của tên miền con d3 . Bước này tận dụng quan sát rằng các thuật toán sinh tên miền (DGA) thường tạo ra các tên miền dài tương đối vì phải tránh xung đột với các tên miền đã đăng ký và các tên miền ngắn hơn có khả năng cao bị chiếm bởi các tổ chức hợp pháp.

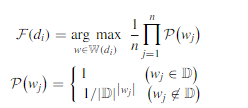
**c) Ước lượng dựa trên từ điển (Dictionary-Based Estimation):**

Trong bước này, nhóm sử dụng từ điển để ước lượng tính ngẫu nhiên của các tên miền con. Cụ thể, nhóm phân loại các từ miền con dựa trên sự xuất hiện của chúng trong từ điển. Các từ miền con phổ biến trong từ điển thường là các từ có ý nghĩa, trong khi các từ không xuất hiện trong từ điển có thể được coi là ngẫu nhiên hoặc do malware tạo ra. Việc này giúp nhóm phát hiện ra các tên miền con có tính ngẫu nhiên cao, đồng thời loại bỏ các tên miền con có xu hướng xuất hiện trong ngôn ngữ tự nhiên.

Bước này ban đầu chia chuỗi tên miền con thành các nhóm từ *ww*w bằng cách sử dụng từ điển. Độ ngẫu nhiên của tên miền con di được ước lượng từ các đặc điểm của nhóm từ *ww*w. Nếu kết quả cho thấy tên miền con *did\_i*di là một chuỗi ký tự có ý nghĩa, thì miền *dd*d, chứa tên miền con này, được xác định là một miền benign được tạo ra bởi con người.

Từ điển được sử dụng cho bước này là từ điển tiếng Anh. Xét đến cách ký hiệu miền, chúng tôi thay thế các biểu diễn không phải chữ cái bằng các biểu diễn chữ cái trong các từ điển cho các ngôn ngữ này.

Chuỗi tên miền con được chia thành một tập từ *ww*w dựa trên hai điều kiện: (1) số từ là tối thiểu, trong khi độ dài của từ là tối đa và (2) ưu tiên được đặt cho các từ có trong từ điển bằng cách phân biệt với sự lựa chọn cực kỳ nghiêm ngặt bằng công thức sau:



Trong đó W(di ) là tập hợp tất cả các phân đoạn ứng cử của tên miền con di , *ww*w là nhóm từ ứng cử *w1,...,wj,...,* ,∣wj ∣ là độ dài của từ wj , và ∣D∣ là tổng số từ trong từ điển D. Ngoài ra, P(wj ) là tính chọn lọc của từ wj , dựa trên việc từ wj có trong từ điển D hay không.

Các điều kiện đầu tiên và thứ hai ở trên chủ yếu tương ứng với các phần *1n\frac{1}{n}*n1 và P(wj ) của công thức này. Bước này chọn ra kết quả tốt nhất từ các phân đoạn được tạo ra bởi sáu từ điển. Nếu kết quả cho thấy n=1, tức là nếu tên miền con di được bao gồm trong từ điển D, thì miền d được xác định là benign.

Vì một chuỗi tên miền do con người tạo ra được đặc trưng bởi một số lượng từ ít và dài, chúng tôi ước lượng độ ngẫu nhiên của tên miền con di như sau:



Trong đó uk là một trọng số cho hàm Lk (w), sắp xếp các từ theo thứ tự giảm dần dựa trên độ dài và sau đó cho độ dài của từ thứ k trong tập từ *ww*w. Ví dụ, khi nhóm từ *ww*w bao gồm các từ kyutech, local, domain và name, L2 (w) và L4 (w) đưa ra kết quả lần lượt là 6 và 4, đó là độ dài của domain và name. Lưu ý rằng các từ không có trong từ điển được gán độ dài là 0. Nếu điều kiện yα >thα được thỏa mãn, thì miền d được xác định là benign.

**d) Ước lượng dựa trên tìm kiếm trên Web (Web-search-based Estimation):**

Trong bước này, nhóm thực hiện truy vấn trên công cụ tìm kiếm Web để xác định tính ngẫu nhiên của các tên miền con dựa trên sự xuất hiện của chúng trên các trang web. Bằng cách này, nhóm có thể đánh giá tính phổ biến và ý nghĩa của các tên miền con trong môi trường trực tuyến. Điều này giúp nhóm cung cấp thông tin bổ ích về các tên miền có nguồn gốc và tính chất của chúng, từ đó hỗ trợ quá trình phát hiện và loại bỏ các tên miền do malware tạo ra một cách hiệu quả hơn.

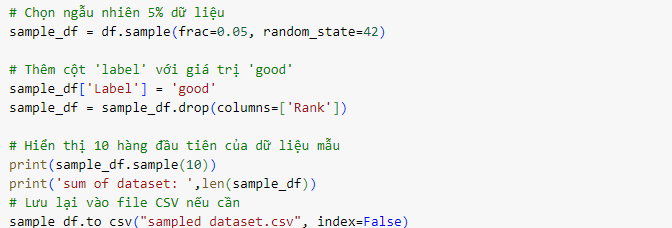
* 1. **Chi tiết cài đặt, hiện thực**

***Cách cài đặt và hiện thực môi trường***

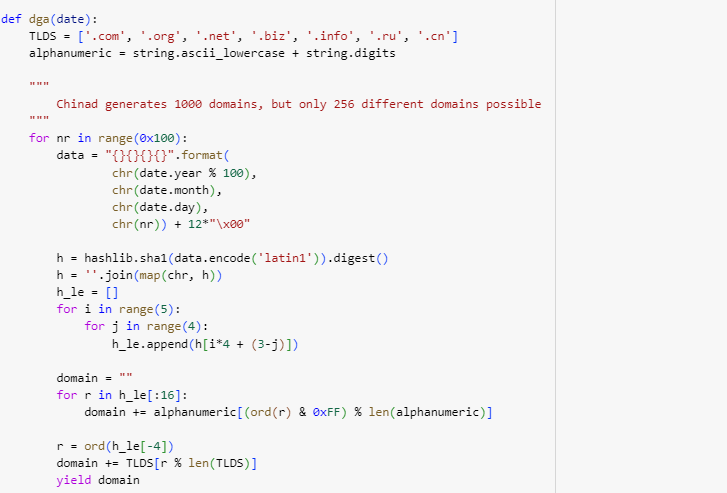
Triển khai thực hiện phân tích dataset trên google colab.

**I. Thu thập dataset**

Dataset: dựa theo bài báo, nhóm thu thập khoảng 50,000 URL được dán nhãn là good( tên miền lành tính. ) được lưu lại với tên file là sampled\_dataset.csv



Ngoài ra, còn thu thập các tên miền được dán nhãn là bad ( tên miền độc hại dựa trên quá trình sau:

*Hàm dga kiểm tra tên miền có phải là Domain Generation Algorithm hay không ?*

Hàm dga() trong đoạn code là một hàm được sử dụng để tạo ra các tên miền giả mạo từ một thuật toán sinh tên miền độc hại. DGA là viết tắt của "Domain Generation Algorithm", là một phương pháp được sử dụng trong mã độc để tạo ra một loạt các tên miền mới một cách tự động, nhằm tránh việc các tên miền cụ thể bị chặn hoặc cấm.

Dưới đây là giải thích chi tiết về cách hoạt động của hàm dga():

**1. Khởi tạo các biến**: Hàm này bắt đầu bằng việc khai báo một số biến cơ bản, bao gồm một danh sách các đuôi tên miền (TLDS), một chuỗi ký tự chứa các ký tự chữ cái và chữ số (alphanumeric).’

**2. Vòng lặp chính**: Trong vòng lặp này, một loạt các tên miền giả mạo được tạo ra. Mỗi lần lặp, một tên miền mới được tạo ra bằng cách sử dụng một thuật toán mã hóa cụ thể.

**3. Thuật toán sinh tên miền giả mạo**: Trong mỗi vòng lặp, một chuỗi dữ liệu được tạo ra từ các thành phần khác nhau như năm, tháng, ngày và số nguyên dương nhỏ hơn 256. Chuỗi dữ liệu này sau đó được mã hóa bằng thuật toán SHA-1, và sau đó được chuyển đổi thành một chuỗi ký tự mới, đại diện cho tên miền giả mạo.

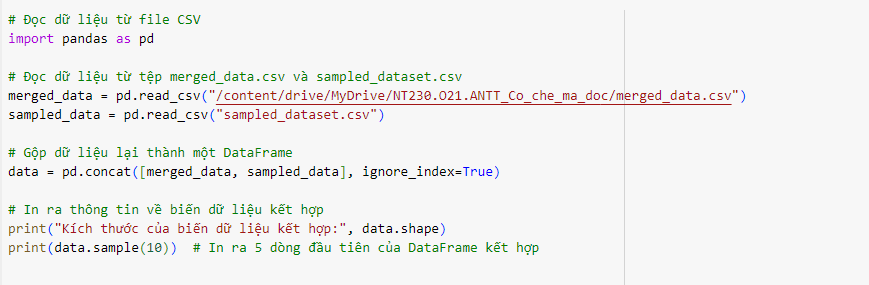
**4. Kết thúc và trả về**: Sau khi tạo ra tên miền giả mạo, nó được trả về bằng cách sử dụng câu lệnh yield, cho phép hàm trả về một giá trị mới trong mỗi lần gọi mà không cần phải duy trì trạng thái của toàn bộ hàm.

Ở đây nhóm chọn 4 kĩ thuật Chinad, Locky, Newgoz và Nymain để nạp vào các dataframe riêng biệt



* Mỗi DataFrame sau khi được nạp vào được lấy mẫu ngẫu nhiên 12,500 mẫu từ các tệp CSV ban đầu để tạo ra một tập dữ liệu có kích thước nhất định.
* Các DataFrame được nối lại với nhau bằng cách sử dụng phương thức concat() để tạo ra một DataFrame duy nhất chứa tất cả các dữ liệu từ các loại tên miền khác nhau.
* DataFrame kết quả được ghi vào tệp "merged\_data.csv" mà không bao gồm cột chỉ mục (index) để thuận tiện sử dụng.
* In ra thông báo xác nhận rằng dữ liệu đã được hợp nhất và lưu vào tệp "merged\_data.csv".

Sau khi đã có được danh sách 50,000 URL được đánh giá là bad thông qua các kĩ thuật phát hiện DGA, thực hiện gộp các URL đã được dán nhãn good ở file sampled\_dataset.csv ( good domain) với file mered\_data.csv ( bad domain ) thành dataset hoàn chỉnh ( khoảng 100,000 URL đã được dán nhãn )

*Đoạn code gộp URL lại thành dữ liệu kết hợp*

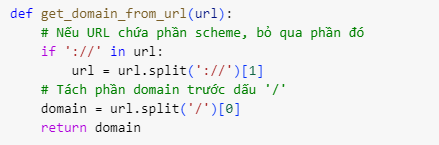
Sau khi đã có được dataset hoàn chỉnh, tiến hành các kĩ thuật phân tích dữ liệu tiếp theo.

**II. Triển khai thực nghiệm**

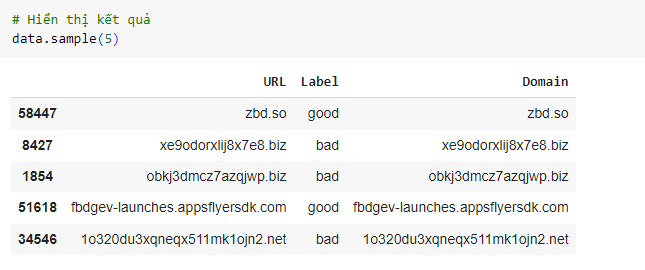
**1. DNSBL**

Nhóm sẽ thực hiện lọc dataset xem chúng có nằm trong blacklist hay whitelist hay không ?

Trước tiên, cần tách domain từ URL để có thể thực hiện được việc kiểm tra.



Kiểm tra với 5 domain ngẫu nhiên:



Tiếp đến, ta sẽ thực hiện kiểm tra xem các domain có nằm trong blacklist, whitelist hay không, thực hiện dán nhãn lần lượt: bad, good, unknow.



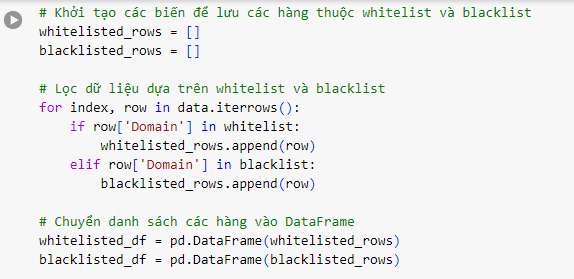
Thực hiện kiểm tra với khoảng 20 domain ngẫu nhiên

A screenshot of a computer

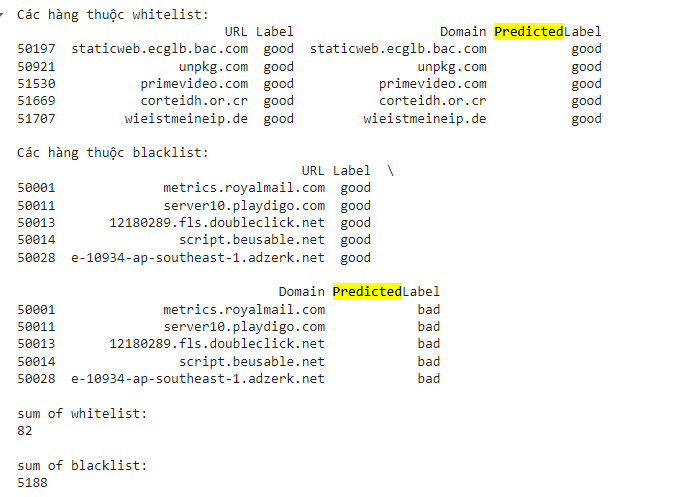
Description automatically generated

Ngay lập tức ta thấy đã có thêm một trường dữ liệu với tên là PredictedLabel được thêm vào.

Nhóm thực hiện tạo thêm 2 danh sách: dnsblacklist và dnswhitelist để chuyển các domain nằm trong 2 danh sách này.

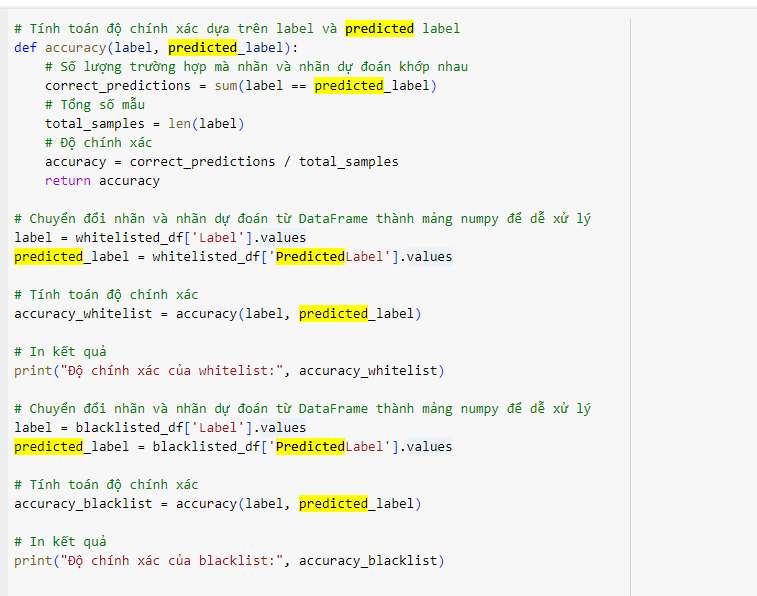


Kết quả thu được:



Như vậy có khoảng 5188 domain nằm trong blacklist, 82 domain nằm trong whitelist.

Kiểm tra xem kết quả này có đúng hay không

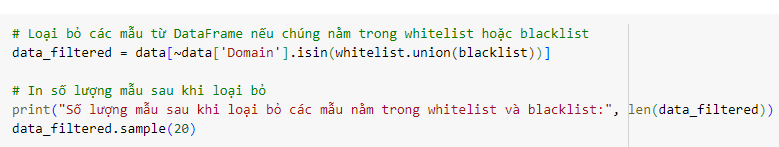


Kết quả:



Độ chính xác của black list thấp là do 1 số domain là độc hại chưa được cập nhật nhãn lại trên database, nên khi so sánh với blacklist hiện nay thì đã không còn chính xác.

Tiếp đến, ta cần loại bỏ đi các domain đã được liệt kê trong blacklist hoặc whitelist.



Sau khi loại bỏ, còn lại khoảng 94,730 domain



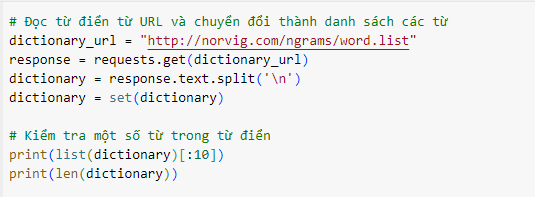
Sau khi đã có kết quả từ triển khai DNSBL, ta tiến hành đánh giá các domain còn lại dựa trên từ điển

**2. Ước lượng dựa trên từ điển**

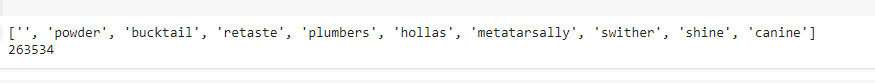
Ở đây nhóm triển khai đối với từ điển tiếng anh, không triển khai các tiếng như Tây Ban Nha, Pháp,... như trong bài báo.

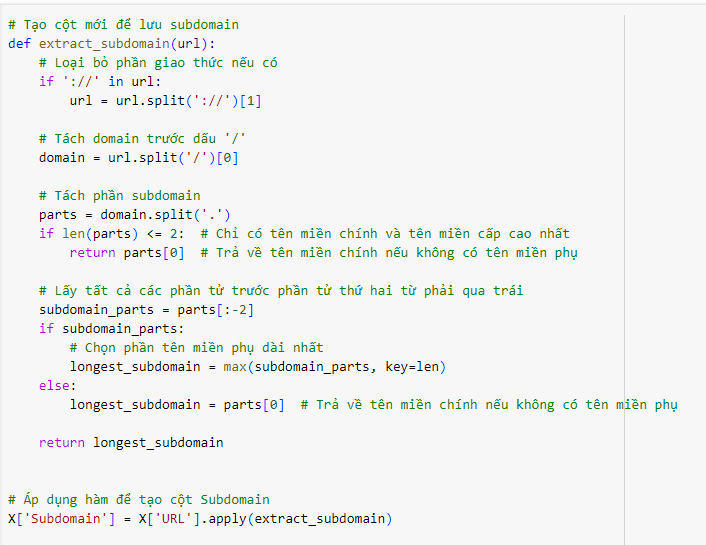
Danh sách các từ nằm trong từ điển: <http://norvig.com/ngrams/word.list>

Bước đầu tiên, ta đọc từ điển từ URL và chuyển đổi thành các từ:



Kết quả

Tiếp theo, ta tạo một cột mới trong dataframe để lưu trữ subdomain của mỗi URL

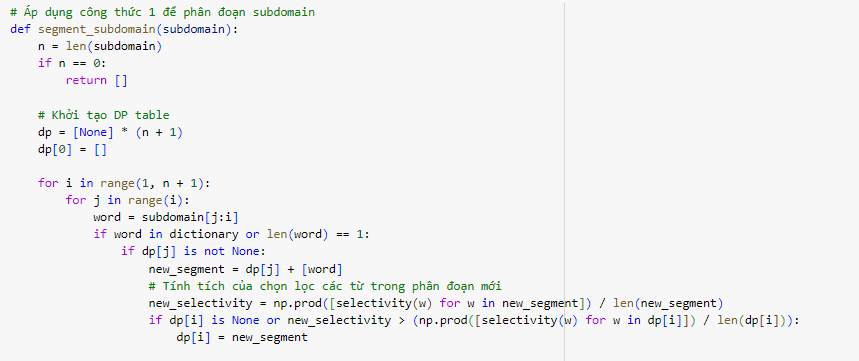


Thực hiện kiểm tra đối với khoảng 10 url ngẫu nhiên

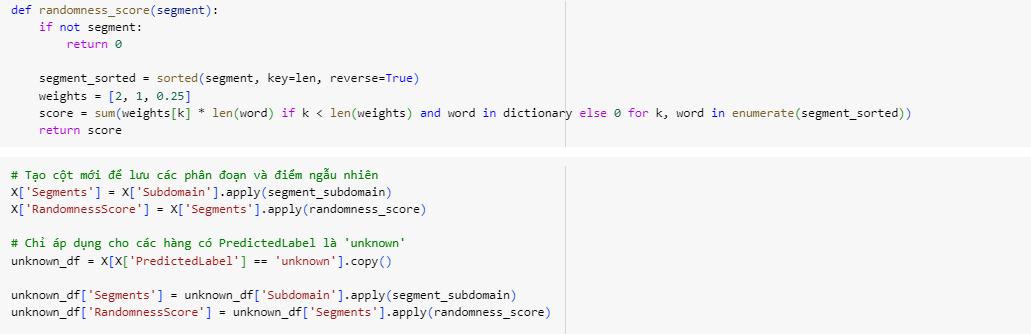
A screen shot of a computer

Description automatically generated

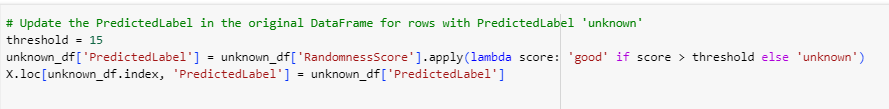
Ta lại đến với giai đoạn phân đoạn subdomain



Tạo các randomnesscore ( điểm ngẫu nhiên )



Đoạn code cập nhật lại giá trị PredictedLabel



Đầu tiên, các hàng từ DataFrame con "unknown\_df" được lọc ra. Sau đó, một ngưỡng (threshold) được sử dụng để quyết định liệu một URL được dự đoán là "tốt" hay "không xác định" dựa trên điểm số ngẫu nhiên (RandomnessScore).

Nếu điểm số lớn hơn ngưỡng, thì "PredictedLabel" được cập nhật thành good , ngược lại là unknown. Cuối cùng, cột "PredictedLabel" trong DataFrame gốc "X" được cập nhật với các giá trị mới từ "unknown\_df".

Hiển thị kết quả cuối cùng sau khi đã thực hiện phân đoạn subdomain, tạo randomnesscore và cập nhật lại giá trị PredictedLabel

Như vậy, quá trình ước lượng tên miền dựa trên từ điển đã hoàn thành.

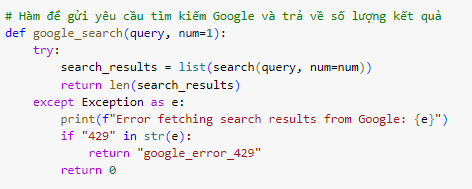
Tiếp đến, ta sẽ thực hiện ước lượng dựa trên tìm kiếm web đối với các domain có predictedLabel là ‘unknown’

**3. Ước lượng dựa trên tìm kiếm web**

Để thực hiện bước kiểm tra tên miền lành tính hay độc hại của một subdomain bằng cách tìm kiếm trên web và phân tích kết quả tìm kiếm, nhóm thực hiện các bước sau:

Ở đây nhóm chọn khoảng 4 công cụ tìm kiếm là google, bing, duckduckgo,

**1. Google**

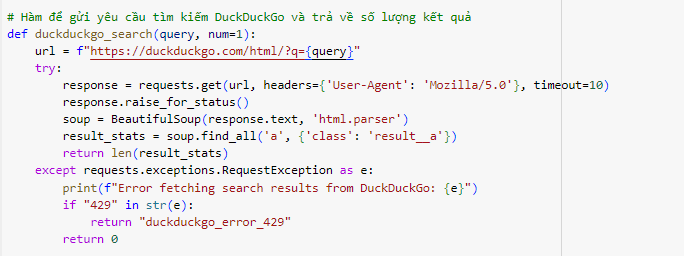
  
*Thực hiện tìm kiếm trên Google và trả về số lượng kết quả tìm kiếm*

**2. Bing**



*Thực hiện tìm kiếm trên Bing và trả về số lượng kết quả tìm kiếm.*

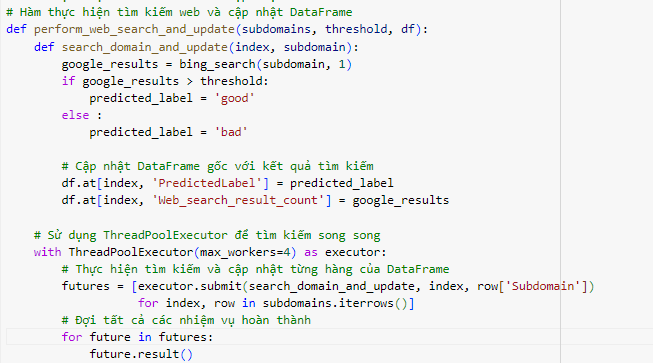
**3. Duckduckgo**

*Thực hiện tìm kiếm trên DuckDuckGo và trả về số lượng kết quả tìm kiếm.*

**4. Yahoo**

 *Thực hiện tìm kiếm trên Yahoo và trả về số lượng kết quả tìm kiếm.*

Thực hiện tìm kiếm web và cập nhật dataframe



Nhóm hực hiện một loạt các tìm kiếm web đồng thời cho mỗi subdomain trong DataFrame. Mỗi subdomain được đưa vào một luồng riêng để tìm kiếm thông qua ThreadPoolExecutor.

Kết quả từ mỗi tìm kiếm được sử dụng để dự đoán nhãn cho subdomain tương ứng, sau đó cập nhật DataFrame ban đầu với nhãn dự đoán và số lượng kết quả tìm kiếm được trả về từ công cụ tìm kiếm web.

* 1. **Kết quả thực nghiệm**

Sau khi hoàn thành các bước được thực hiện trong bài báo, cuối cùng nhóm thu được một danh sách các domain đã có đầy đủ các thông tin để so sánh với nhãn đã dán ban đầu.

Từ đó có thể kiểm tra được tính chính xác của mô hình và các điểm cần cải thiện trong tương lai.

Kết quả được đánh giá như sau:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precison | Recall | F1-score | Support |
| Good | 1.00 | 0.20 | 0.33 | 50,000 |
| Bad | 0.56 | 1.00 | 0.71 | 50,000 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Accuracy |  |  | 0.6 | 100,000 |
| Macro avg | 0.78 | 0.60 | 0.52 | 100,000 |
| Weighted | 0.78 | 0.60 | 0.52 | 100,000 |

Báo cáo phân loại và ma trận nhầm lẫn cung cấp cái nhìn tổng quan về hiệu suất của mô hình phân loại:

**1. Lớp "good"**:

* + **Độ chính xác (Precision)** chỉ là 100%, nhưng **độ phủ (Recall)** thấp, chỉ 20%. Điều này ngụ ý rằng mô hình chỉ dự đoán đúng một phần nhỏ các trang web thực sự là "good" và bỏ lỡ nhiều trang web "good" khác.
  + F1-score là chỉ số kết hợp giữa Precision và Recall, và ở đây chỉ là 33%, cho thấy hiệu suất tổng thể của mô hình đối với lớp "good" khá thấp.
  + Số lượng mẫu trong lớp "good" là 50,000.

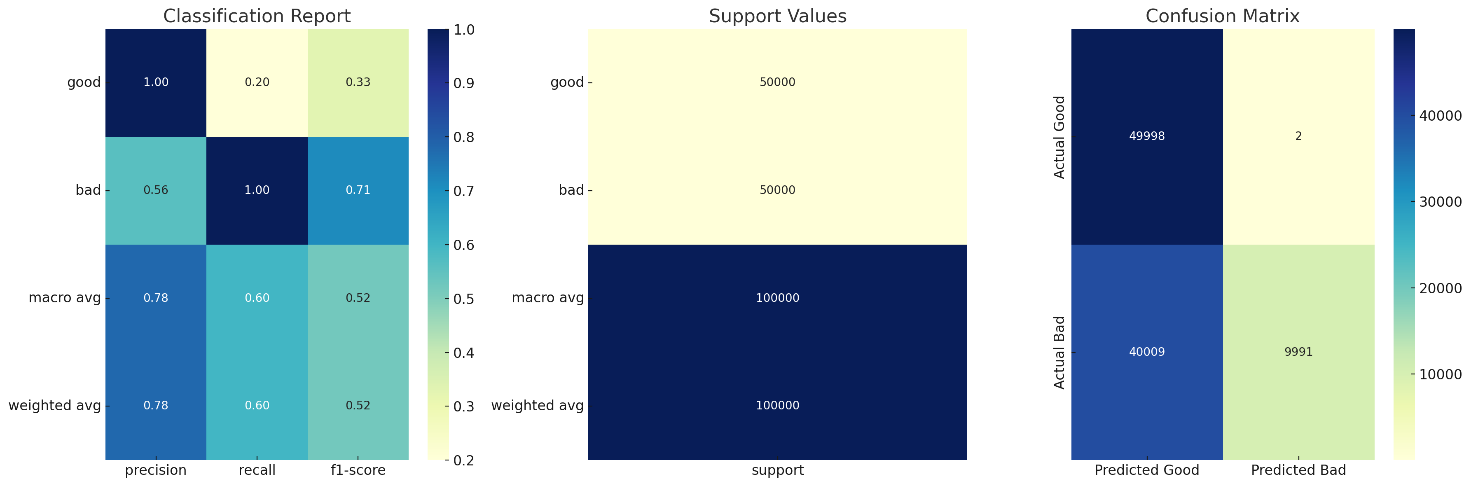
**2. Lớp "bad"**:

* + Độ chính xác là 56% và độ phủ là 100%. Điều này ngụ ý rằng mô hình có xu hướng dự đoán chính xác nhiều trang web thực sự là "bad" hơn là dự đoán trang web "good".
  + F1-score cho lớp "bad" là 71%, chỉ số này cho thấy mô hình có hiệu suất tốt hơn đối với lớp này so với lớp "good".
  + Số lượng mẫu trong lớp "bad" cũng là 50,000.

Tổng quan, mô hình có xu hướng dự đoán chính xác hơn cho lớp "bad" so với lớp "good". Tuy nhiên, mô hình cần cải thiện đáng kể để cân bằng giữa độ chính xác và độ phủ cho cả hai lớp, đồng thời tăng cường hiệu suất phân loại tổng thể.

**Confussion matrix**

* **True Positives (TP)**: 49998, Số lượng miền "good" được phân loại chính xác là "good".
* **True Negatives (TN)**: 9991, Số lượng miền "bad" được phân loại chính xác là "bad".
* **False Positives (FP)**: 2, Số lượng miền "bad" được phân loại sai là "good".
* **False Negatives (FN)**: 40009, Số lượng miền "good" được phân loại sai là "bad".



* 1. **Hướng phát triển**

Hướng Phát Triển Tiềm Năng của Đề Tài trong Tương Lai

Mở rộng và cải thiện dữ liệu huấn luyện: Một hướng phát triển quan trọng là thu thập và sử dụng các bộ dữ liệu lớn hơn và đa dạng hơn. Việc này sẽ giúp mô hình học máy phân biệt tốt hơn giữa các tên miền lành tính và độc hại, từ đó tăng độ chính xác và độ nhạy của hệ thống.

Tích hợp với các phương pháp phát hiện khác: Kết hợp phương pháp hiện tại với các kỹ thuật tiên tiến như học sâu (Deep Learning) hoặc học máy (Machine Learning) sẽ cải thiện hiệu suất phát hiện. Điều này đặc biệt quan trọng để phát hiện các mẫu tên miền phức tạp mà phương pháp hiện tại có thể bỏ sót.

Ứng dụng trong môi trường thực tế: Đánh giá hiệu quả của phương pháp trong các mạng lưới lớn và phức tạp, đặc biệt là các mạng lưới có lưu lượng truy cập cao và các yêu cầu bảo mật nghiêm ngặt, sẽ giúp xác định khả năng ứng dụng thực tế của hệ thống.

Nhận Xét về Tính Ứng Dụng của Đề Tài

Khả năng phát hiện nhanh chóng và chính xác: Phương pháp đề xuất có khả năng phát hiện nhanh chóng và chính xác các tên miền độc hại được tạo ra bởi phần mềm độc hại sử dụng thuật toán tạo tên miền (DGA), giúp giảm thiểu rủi ro và thiệt hại cho các hệ thống mạng.

Tính linh hoạt cao: Phương pháp này có thể được áp dụng trong nhiều kịch bản khác nhau, bao gồm các mạng lưới doanh nghiệp, các hệ thống mạng lớn như trường đại học, và các hệ thống mạng công cộng. Điều này chứng tỏ tính linh hoạt và khả năng thích ứng cao của phương pháp.

Tích hợp dễ dàng: Phương pháp này có thể dễ dàng tích hợp vào các hệ thống phát hiện và phòng chống phần mềm độc hại hiện có, từ đó cải thiện hiệu suất và khả năng bảo vệ của các hệ thống này.

**HẾT**

1. Ghi nội dung tương ứng theo mô tả [↑](#footnote-ref-1)