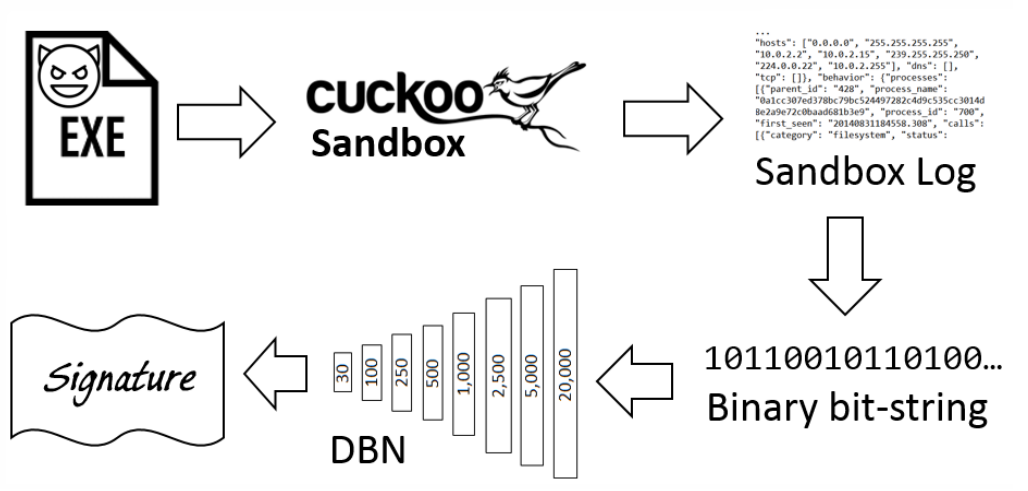
**Malware classification approach**

1. DeepSign: Deep Learning for Automatic Malware Signature Generation and Classification

* Bài báo trình bày phương pháp áp dụng kỹ thuật học sâu để tự động học cách biểu diễn một file malware (signature), sau đó cho vào một bộ phân loại để xác định lớp mà malware đó thuộc về.
* Phương pháp này cho độ chính xác 98.6% trên bộ dữ liệu gồm 6 loại malware
* Các bước thực hiện như sau:



Hình 1: Trình tự các bước học cách biểu diễn một file PE

+ Sử dụng Sandbox cuckoo để ghi lại hành vi của chương trình đầu vào tương ứng (file PE), output là file text chứa các hành vi (các lời gọi hàm, các biến, …) của file PE ban đầu.



Hình 2: Ví dụ file hành vi được sinh ra bởi sandbox cuckoo

+ Biểu diễn file trên dưới dạng vector:

1. Sinh ra tất cả các unigram (1-gram)

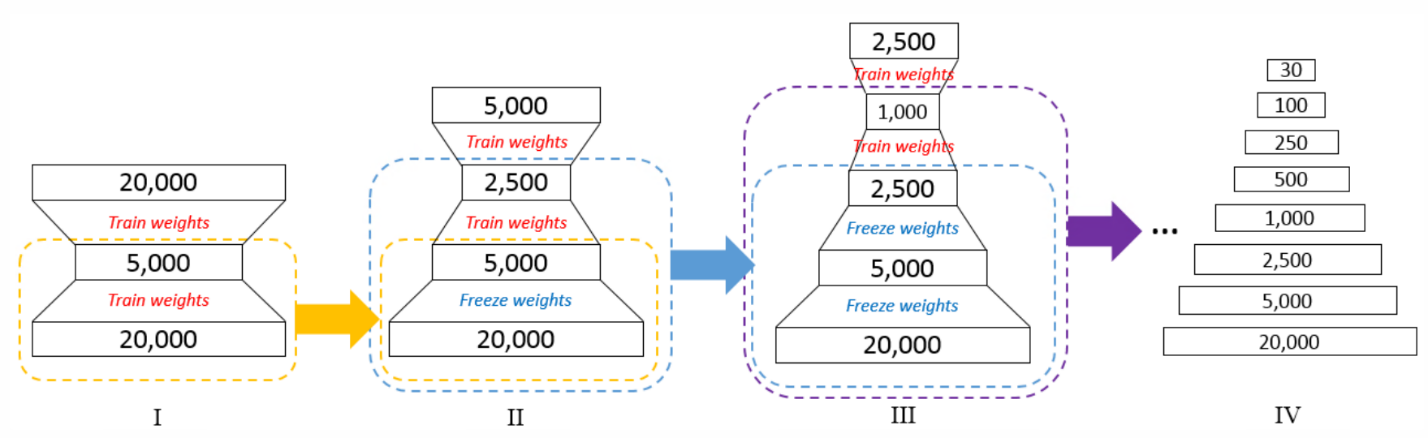
2. Loại những unigram xuất hiện trong tất cả các file

3. Với mỗi unigram, đếm số lượng file chứa unigram đó

4. Chọn ra top-20000 unigram

5. Biểu diễn mỗi file tương ứng với một vector , , nếu unigram thứ i xuất hiện trong file, ngược lại = 0

+ Học ra signature tương ứng với vector ở trên sử dụng denoising autoencoder, gồm 8 layer 20000 – 5000 – 2500 – 1000 – 500 – 250 – 100 – 30. Mỗi bước chỉ trên một layer duy nhất. Kết quả sinh ra vector signature kích thước 30 tương ứng với mỗi file malware



Hình 3: Denoising auto encoder

+ Sử dụng SVM hoặc KNN để phân loại malware dựa trên signature đã xây dựng ở bước trên.

1. Large-Scale Malware Classification Using Random Projections And Neural Networks (2013)

* Huấn luyện trên 2,6 triệu mẫu malwares, đạt tỉ lệ lỗi 0.49%
* Biểu diễn file đầu vào bằng ma trận nhị phân thưa kích thước lớn, giảm chiều dữ liệu trước khi huấn luyện bằng mạng neural



Hình 4: Sơ đồ hệ thống malware classifier

* Với mỗi file PE, sử dụng Microsoft Security Essentials, Forefront Endpoint Protection để chọn features
* Sử dụng Random Projections để giảm chiều dữ liệu

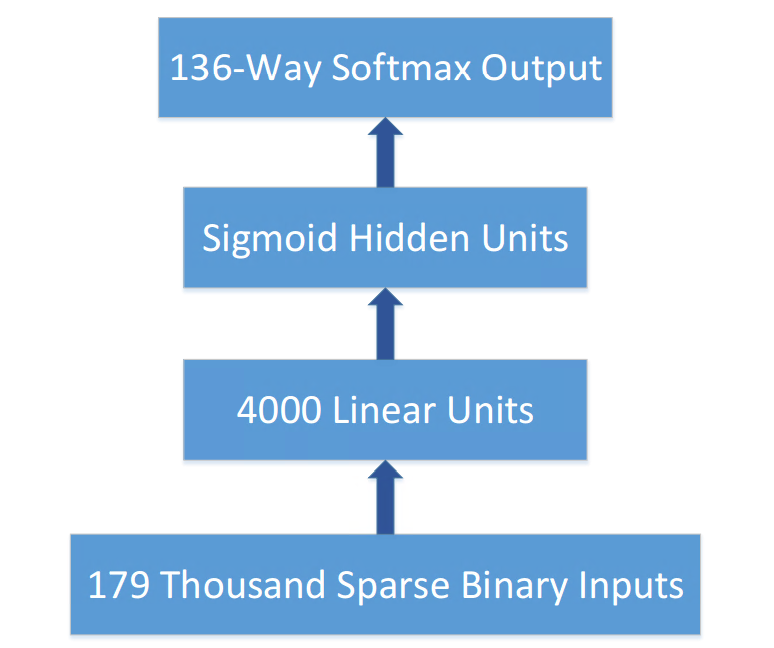
+ là dữ liệu đầu vào, n là số instances, d là số lượng features

+ là dữ liệu sau khi giảm chiều

+ là ma trận Random Projections thỏa mãn

với xác suất lần lượt là

* Coi là input của mạng neural



Hình 5: Kiến trúc mạng neural

1. MtNet: A Multi-Task Neural Network for Dynamic Malware Classification

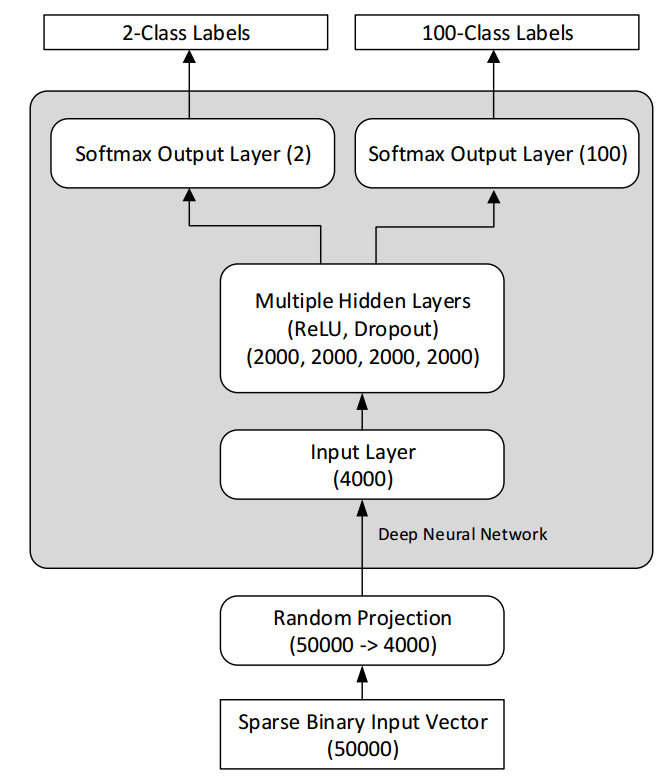
* Đề xuất mô hình Deep learning multi task, huấn luyện trên 4.8 triệu file, đạt tỉ lệ lỗi 0.358%
* Mỗi file PE sẽ được mô phỏng lại bởi anti malware engine, 2 loại thông tin được trích chọn: các lời gọi hàm (API) cùng các biến đi kèm và null-terminated object được khôi phục từ bộ nhớ trong quá trình mô phỏng. Có 114 lời gọi hàm được thống kê.
* 3 tập features được trích chọn từ 2 nguồn thông tin trên:

+ null-terminated object

+ Kết hợp 1 lời gọi hàm và 1 biến

+ Trigram API (kết hợp 3 API liên tiếp)

* Biểu diễn features dưới dạng sparse binary vector
* Sử dụng mutual information để chọn top 50000 features tốt nhất
* Mô hình mạng:



Hình 6: Multi task neural networks

* Object function là sự kết hợp giữa 2 task malware binary classification (malicious or benign) và malware family classification (100 class)

1. Deep Neural Network Based Malware Detection Using Two Dimensional Binary Program Features

* Sử dụng mạng neural huấn luyện trên 400,000 mẫu, đạt độ chính xác 95%.
* Input đầu vào là file binary tương ứng với file PE
* Trích chọn đặc trưng:

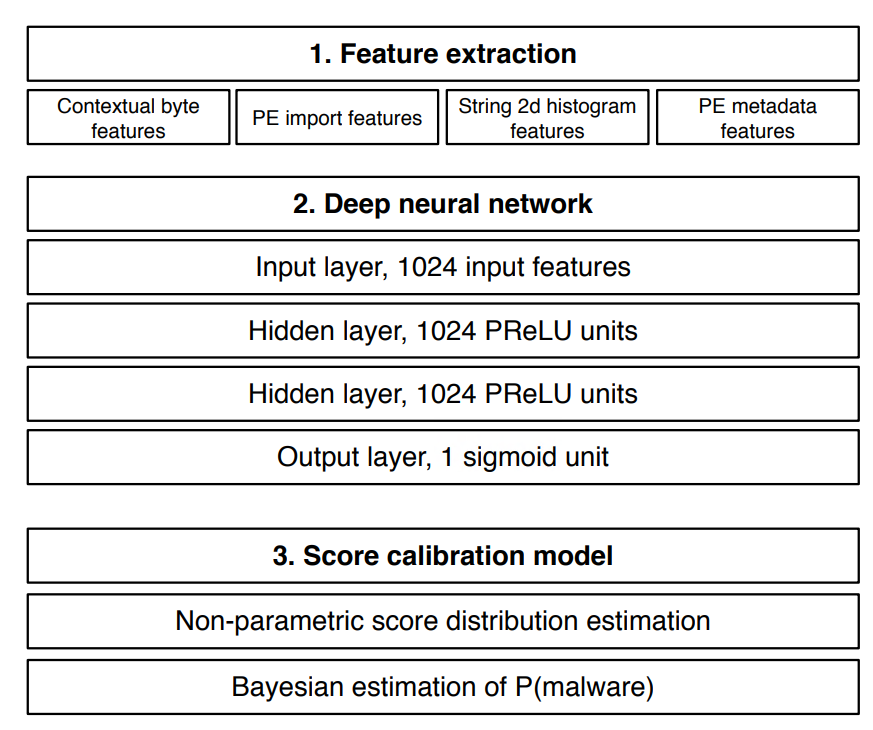
+ Byte/Entropy histogram features:

1. Trượt trên toàn bộ file với kích thước cửa sổ (window) 1024 byte. Với mỗi window, xây dựng vector , trong đó , là byte thứ trong cửa sổ , là giá trị entropy của window
2. Chia [0, 8] thành 16 bin – X, chia đoạn [0, 255] thành 16 bin – Y
3. Đếm số lần xuất hiện của các cặp => vector 256 chiều

+ PE import features: xây dựng vector 256 chiều biểu diễn sự xuất hiện của các API trong file

+ PE meta data: sử dụng thư viện ‘pefile’ xây dựng vector 256 chiều

* Mô hình thực nghiệm:



Hình 7: Mô hình thực nghiệm

* Gọi là giá trị output của mạng neural, tính đặc trưng cho độ tin tưởng file đầu vào là malware

được tính theo phương pháp kernel density estimator