Learning the PE Header, Malware Detection with Minimal Domain Knowledge

Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security (2017)

Edward Raff · Jared Sylvester · Charles Nicholas

-2021.02.16-

김 정 우

목차

Introduction

- **목표 : 최소한의 'Domain Knowledge'를 사용해서 악성코드 탐지기를 학습시키자**

- Domain Knowledge?
  - 시스템을 구현할 때 필요한 선험적인 지식 (Model을 구현할 때 사용하는 Feature)

- 'Domain Knowledge' 기반의 악성코드 탐지는 어려운 편
  - PE file 파싱은 번거롭고 시간이 많이들 수 있음
  - 사용되는 Domain Knowledge를 줄이기위해 Pe Header 분석을 통해 악성코드 탐지를 수행

- Domain Knowledge필요 없는 n-gramming기법도 존재

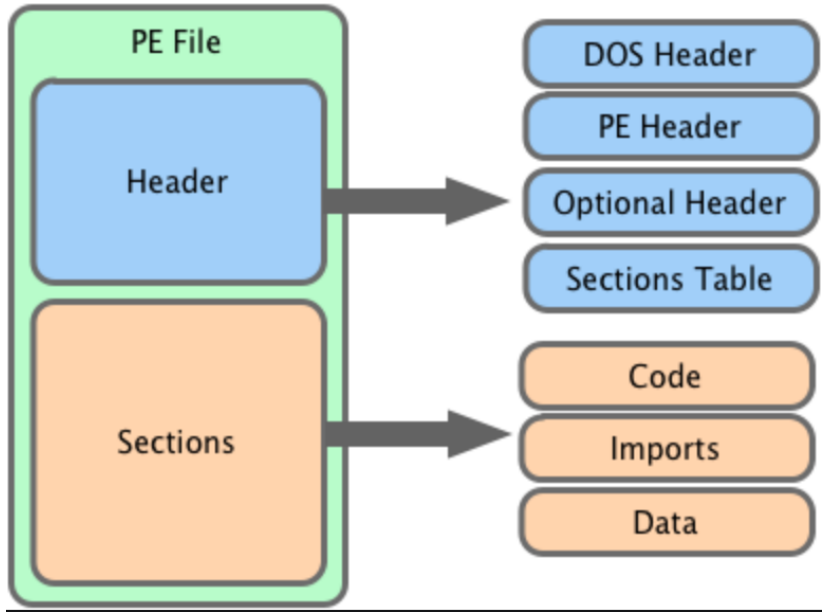
- 'Minimal domain knowledge' 을 통해 악성코드 탐지를 할 수 있어야 함

Introduction

- **목표 : 최소한의 'Domain Knowledge'를 사용해서 악성코드 탐지기를 학습시키자**

- 악성코드 탐지기에 학습시킬 Feature (Minimal domain knowledge)
  - Windows PE file의 pe header
  - MS-DOS
  - Optional header

- 악성코드 탐지기를 구현하고,
  Baseline을 정해서 이들과 성능 비교



Related Work

- PE-Miner Project , Pe header 분석을 통한 탐지
  - 189개의 feature, 그 중 73개는 DLL imports 정보
- DLL imports and function imports 목록기반 탐지
- 결정트리기반의 악성코드 탐지
  - Function, header field기반
- N-gramming을 통한 악성코드 탐지
- HMM(Hidden Markov Models)기반의 악성코드 탐지
- RNN기반의 악성코드 탐지
  - 동적 분석 결과를 통해 얻어낸 고수준 feature

Baseline Approach

- 연구 목표와 비교하기 위한 기준 모델

1. Byte N-Gramming Approach

2. PortEX를 통해 feature 추출해 트리기반의 악성코드 탐지기 구현
   - 115개의 feature 추출
   - 결정 트리, SVM은 성능이 좋지 않아 제외
   - Import table 정보 제외

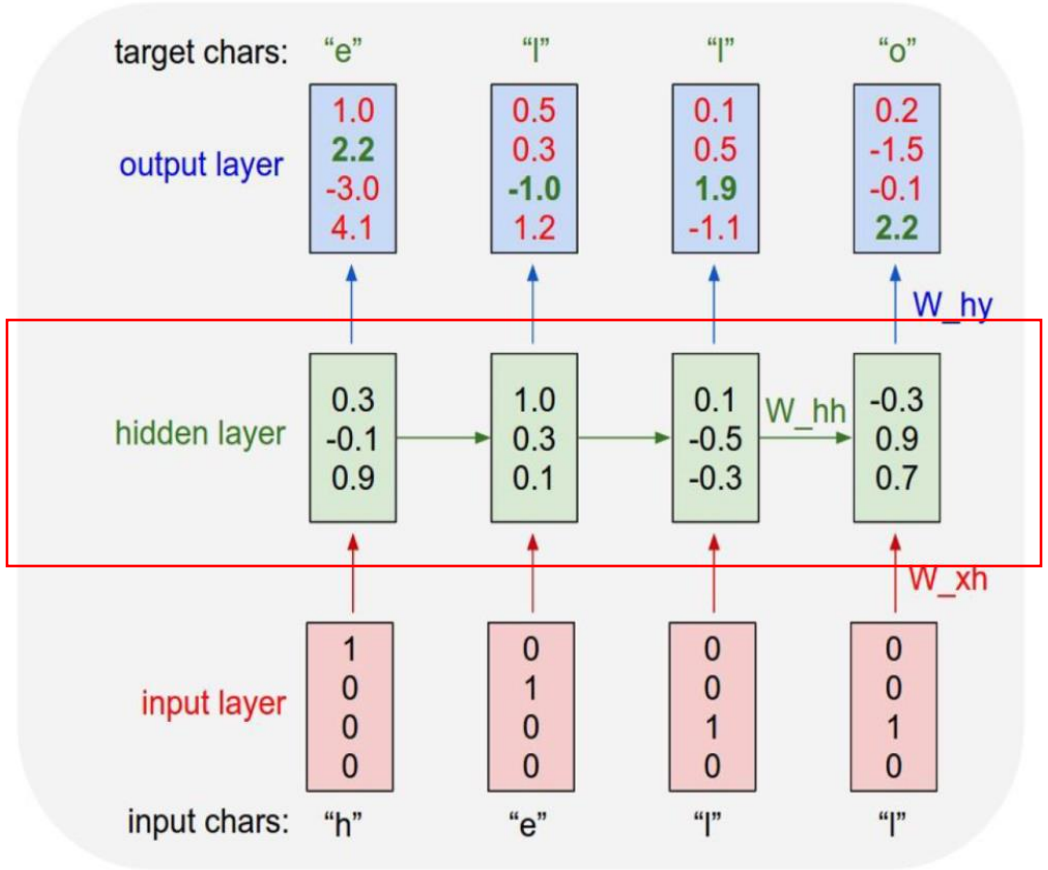
Neural Network Approach

- **Classifier using Minimal domain knowledge**
- RNN / LSTM

- 순차적인 데이터를 분석하는 인공신경망
  - 시퀀스에 대한 확률 모델

- Hidden layer를 여러겹 쌓을 수 있음

- RNN의 단점을 보완하는 LSTM, GRU
  - 은닉상태를 처리하는 방안을 개선
  - RNN cell의 구현 형태가 다름



Neural Network Approach

- Attention LSTM architecture

- Attention 매커니즘
  - 출력 데이터를 예측하는 매 시점(time step)마다, 전체 입력 문장을 Attention Value에 기반해 다시 한 번 참고

- 어텐션 함수(Attention Function)
  - Attention(Q, K, V) = Attention Value

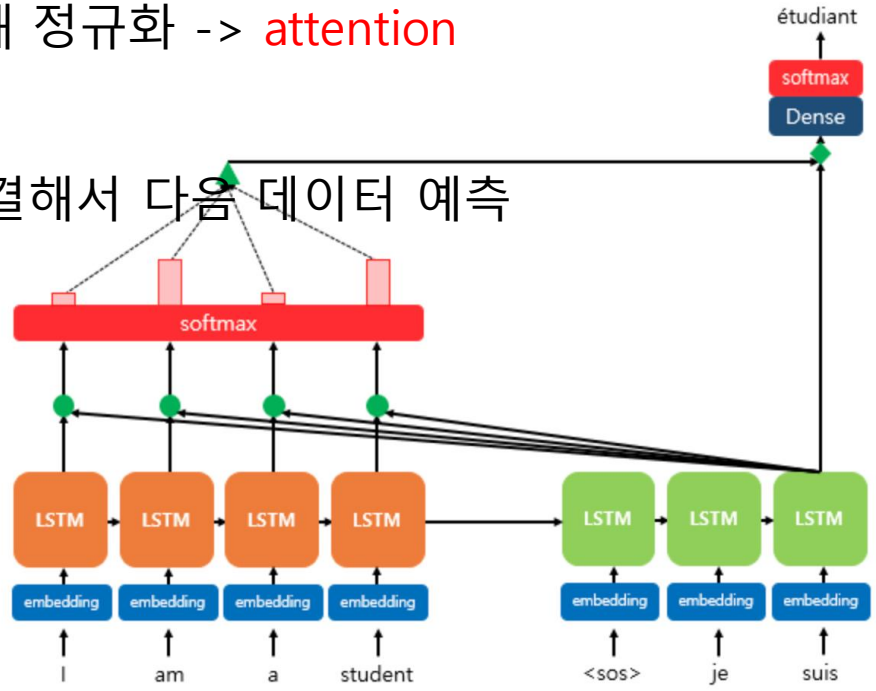
Q = Query : t 시점의 디코더 셀에서의 은닉 상태
K = Keys : 모든 시점의 인코더 셀의 은닉 상태들
V = Values : 모든 시점의 인코더 셀의 은닉 상태들

Neural Network Approach

- Attention score
  - 인코더의 모든 은닉 상태 각각이 디코더의 현 시점의 은닉 상태 와 얼마나 유사한지를 판단

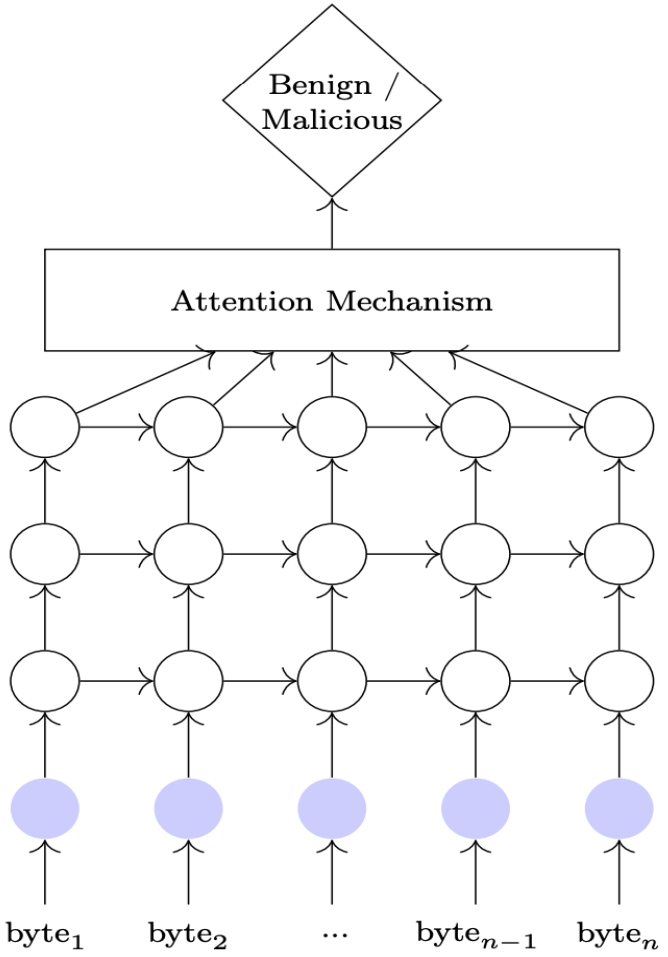
- Q와 K의 내적을 통해 값을 구해고 이를 softmax를 통해 정규화 -> attention weights

- 1단계의 결과를 통해 가중합(V)을 구하고 이를 Q와 연결해서 다음 데이터 예측



Neural Network Approach

- 'Minimal Domain Knowledge'를 사용한 악성코드 탐지기
- Attention LSTM architecture
  - domain knowledge free approaches
- equivalent information
  - PE header에 해당하는 328byte를 추출



Experiment Methodology and Results

- Extra Random Trees (ET) and Random Forests (RF)
  - domain knowledge baselines trained on PE header features
- Logistic Regression on byte 3-grams (LR)
  - domain knowledge free approach
- FC and LSTM

Experiment Methodology and Results

- Group B is supposed to better represent the common types of files found on client machines.
- Group A benign data can lead to significant over-fitting on properties unique from MS
- TEST SET
  - Group A, Group B, Open Malware
- Training model with Group B
  - And test with Group A, Group B, Open Malware

Table 1. Breakdown of the number of malicious and benign training and testing examples in each data group, along with the sources from which they were collected. "Misc." comprises `portablefreeware.com`, Cygwin and MinGW.

	training		testing	
Group A	malicious	benign	malicious	benign
Virus Share	175,875	—	43,967	—
Open Malware	—	—	81,733	—
MS Windows	—	268,236	—	21,854
Misc.	—	1,195	—	—
total	*175,875*	*269,431*	*125,700*	*21,854*
Group B				
Industry Partner	200,000	200,000	40,000	37,349
total	*200,000*	*200,000*	*40,000*	*37,349*

Experiment Methodology and Results

- Group A로 훈련시킨 모델의 테스트 결과
  - 과적합으로 인해 Group B에서는 성능이 낮음 -> 그렇다고 해당 접근법이 틀린 것은 아님

Table 2. Performance of Random Forest and Extra Trees using header-based features. Models trained on only the Group A training data.

	Extra Trees		Random Forest	
	Accuracy	AUC	Accuracy	AUC
Group A Test	99.1%	0.999	99.5%	0.999
Group B Test	69.9%	0.723	71.5%	0.728
Open Malware	99.9%	—	95.5%	—

Experiment Methodology and Results

•Model result
•바이너리 데이터와 타 데이터의 차이점
  • 변조, 노이즈에 취약함, 아주 작은 비트차이가 매우 다른 동작방식을 야기할 수 있음

•그럼에도 불구하고 높은 성능을 신경망을 보여줄 수 있다.

Table 3. Performance of all methods on the test sets. Accuracy scores are balanced so classes have equal contribution.

	Group A Test		Group B Test		Open Malware
	Accuracy (%)	AUC (%)	Accuracy (%)	AUC (%)	Accuracy (%)
Fully Connected	90.8	97.7	83.7	91.4	89.9
LSTM	84.2	96.7	77.5	86.7	79.7
Extra Tree	86.4	97.2	80.7	86.1	85.5
Random Forest	78.9	96.8	82.3	91.2	64.4
LR 3-grams	71.2	91.4	77.8	87.3	61.5
WFByte N-Grams	87.3		94.5		81.1

+ PE header의 데이터가 다른 데이터에 대해 일반화가 잘 될 수 있음(추측)

Inferring Relative Feature Importance

- 악성코드 탐지에 중요한 Feature
  - Feature importance from tree model
  - Attention weights of the LSTM model
- 중첩되서 사용된 Feature들이 존재
  - Black Box와 같은 NN을 분석 가능
- Export table 사이즈는 중첩 X
- MS-DOS header는 영향력이 낮음

Table 4. Most important features, as determined by the Extra Trees algorithm. Relative Importance (RI) is in the last column, with 1.0 for the most important feature. Byte ranges expressed as [min, max) when the feature is multiple bytes in length. For features that may have a different location for 32 and 64-bit binaries, the 32-bit location is specified first, followed by the 64-bit location.

Header Field	Byte Location	RI
IMAGE_FILE_DLL	87th, 3rd MSB	1.00
Certificate Table's size	[220, 224)/[236, 240)	0.42
TERMINAL SERVER AWARE	159th. MSB	0.34
Export Table's size	[188, 192)/[204, 208)	0.33
CLR Runtime Header's size	[300, 304)/[316, 320)	0.23
Subsystem field	[156, 158)	0.21
Debug table's size	[236, 240)/[252, 256)	0.18
CLR Runtime Header's offset	[296. 300)/[312, 316)	0.15
Import Table's size	[196, 200)/[212, 216)	0.12
NO SEH	158th, MSB	0.09

Conclusion

- 원시 바이트('Minimal domain knowledge') 데이터를 통해 인공신경망을 학습 가능
- 새로운 데이터에도 성능이 높고 견고한 기법
- Malware family classification에도 적용될 수 있을거라 예상