<주요 Q&A>

GNN(Graph Neural Network)을 이용한 악성코드 탐지

Q1. 그래프 에서 분석 가능한 데이터에 는 제한이 없는지요?

방향성이 없는 그래프는 모두 사용할 수 있습니다. 다만 노드와 연결 수가 많아짐에 따라 연산량이 증가하기 때문에, 현실적으로는 그래프 크기에 제약이 따릅니다.

Q2. 악성코드를 탐지하는 데 GNN을 효율적으로 활용하는 방법에 대해서 질문드립니다 이와 관련하여 중점적으로 검토하고 점검해야 할 사항들에 대해서 질문드립니다

발표에서도 언급이 되겠지만, 코드에서 그래프 구조 (CFG 등)을 추출할 수 있는지, 추출 시간이 얼마나 걸리는지에 대한 고려가 필요합니다. 생각보다 시간이 많이 걸리는 소수의 코드들이 있습니다.

Q3. GNN을 사용하여 악성코드를 탐지하는 작업을 하는 동안 경험하는 문제점이나 애로사항을 해결하고 극복할 수 있는 방법에 대해서 문의드립니다.

그래프 생성시 오류와 시간이 너무 많이 걸리는 경우가 있어, 여기에 대한 대응책이 필요합니다. 저희는 그래프 정보를 필요로 하지 않는 다른 탐지기를 함께 사용하는 식으로 해결을 했습니다.

Q4. 악성코드를 탐지하는 GNN의 학습 데이터는 어떤 형태의 데이터로 학습을 해야할까요?

노드의 벡터화를 위해 각 노드에 해당하는 basic block의 asm code와 CFG 정보가 필요합니다.

Q5. 실시간 트래픽 분석에 사용할 수 있을 정도로 속도나 오탐 등 문제점들은 없는지요?

실시간 이 어떤 시간 구간인지에 따라 다르겠지만, 저희 work flow에서는 문제없이 사용할 정도의 성능을 보였습니다. 다만, CFG의 노드 수가 너무 많은 경우는 시간을 고려하여 제외해야 했습니다. 단, 추론시간 보다는 CFG 생성시간이 너무 오래 걸려서 였습니다.

Q6. 혹시 발표자료 관련된 논문링크가 있을까요?

논문은 아직 없습니다만 고려하고 있습니다.

Q7. basic block의 asm code와 CFG 정보 가 있다면 악성코드 문제는 해결 할 수 있다는 건가요?

완전한 해결 보다는, 변종 악성 코드에 강건할 수 있는 기술에 대한 소개로 보시면 좋을 것 같습니다 ^^

Q8. CFG 생성에 필요한 정보가 무엇인지? 실제 검증된 사례가 있는 것인지요?

네 발표에서 산학연구과제 결과를 일부 소개 드립니다.

Q9. CFG 생성에 필요한 정보가 궁금합니다.

바이너리 코드와 도구 (저희는 공개 코드인 anger 사용)가 필요합니다.

Q10. AI 기술이 놀랍게 진보하고 있는데 그에 대한보안은 어떻게 되는지 궁금합니다.

Q11. 악성코드를 AI기술을 활용해서 탐지해서 걸러낼 수도 있지만, 해커가 기술을 악용한다면 또 따른 문제가 되지 않을까 생각됩니다.

맞습니다. 저희 연구실에서 자동 악성코드 변종 생성 기술도 연구한 바 있는데... 여러 가지 이슈 때문에 논문화하진 못했습니다.

Q12. 안녕하세요, Graph node 또는 sub graph 단위로 샘플링 후에 training 하시는 걸까요?

아니요, 샘플링을 하기에는 어느 노드에 악성 부분이 있는지 모르기 때문에 전체 그래프를 사용합니다.

Q13. GNN을 이용 가능성만 report 하셔서 여쭤보는데요, 기존의 악성코드 탐지보다 얼마나 발전된 기술인지 설명해주실 수 있나요?

기존 AI기반 SOTA 기법들과 성능 면에서는 유사합니다. 다만 그 특성상 변종코드에 좀 더 강건하리라 기대하는데, 추가 연구가 필요한 부분입니다.

Q14. 임베딩과정에서 사용한 Word2Vec 알고리즘에서는 일반적으로 Skipgram 방식이 CBOW보다 뛰어난 것으로 알고 있는데요, Asm2Vec과 본 연구에서는 CBOW를 채택한 이유가 따로 있을까요?

맞습니다. 코드에서도 CBOW 구조가 실험적으로 좀 더 나은 성능을 보였습니다