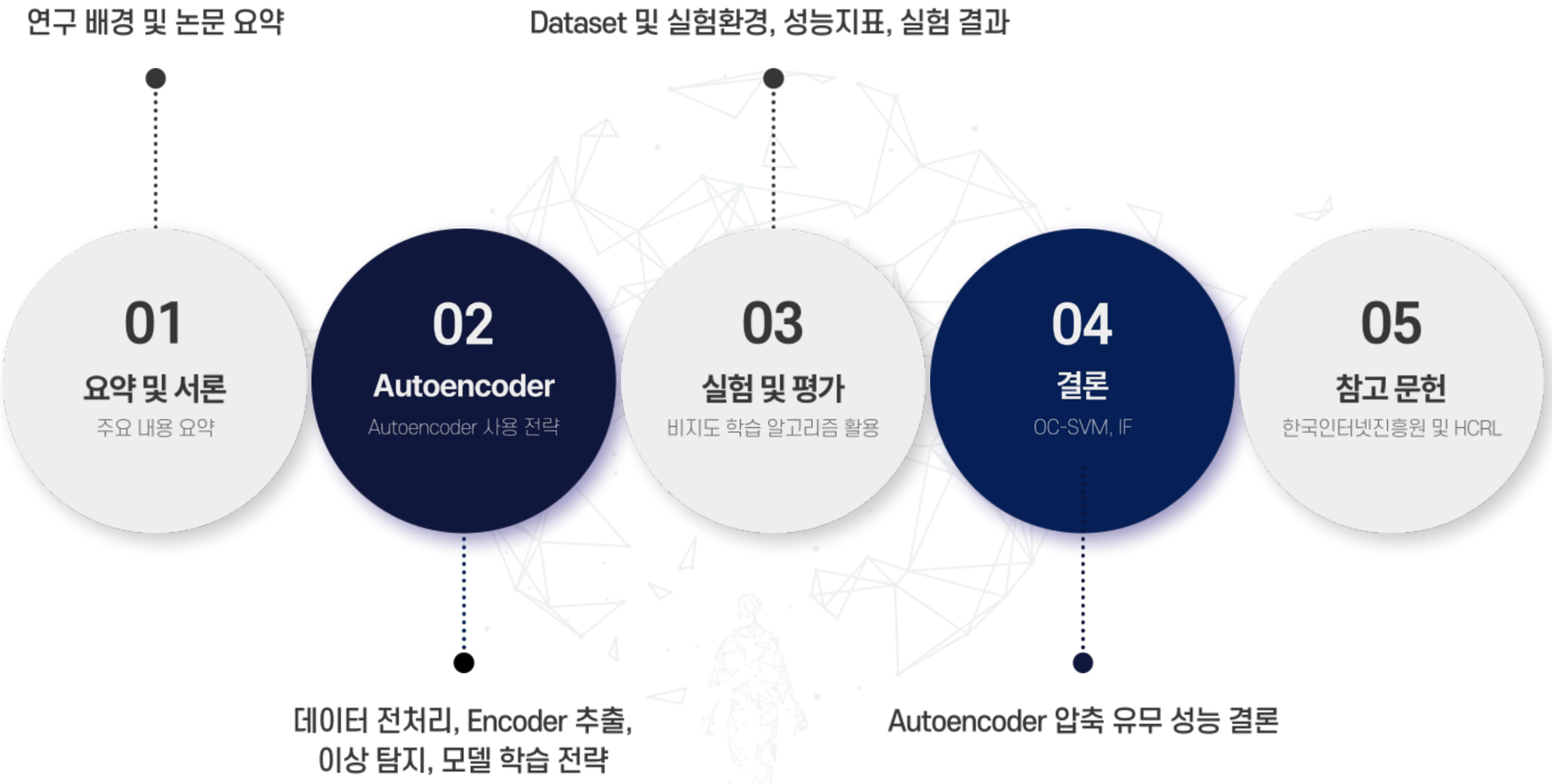
Autoencoder 기반 경량 피쳐를 이용한 IVN 이상징후 탐지 방법

Anomaly detection method of IVN using lightweight features based on Autoencoder

Do-Hee Kang, Software, Hallym University (Undergraduate Student)

목차 (Contents)


연구 배경 및 논문 요약
01
요약 및 서론
주요 내용 요약
02
Autoencoder
Autoencoder 사용 전략
데이터 전처리, Encoder 추출,
이상 탐지, 모델 학습 전략
Dataset 및 실험환경, 성능지표, 실험 결과
03
실험 및 평가
비지도 학습 알고리즘 활용
04
결론
OC-SVM, IF
Autoencoder 압축 유무 성능 결론
05
참고 문헌
한국인터넷진흥원 및 HCRL

01
요약 및 서론

주요 내용 요약

요약(Abstract)

Abstract

- 현대의 차량에 적용되어 있는 CAN Bus 는 보안성이 결여됨.
- 내 · 외부에서 CAN Bus 의 연결을 통해 위협을 초래할 수 있음.

- 본 논문에서는,
  - 차량 내부 네트워크 CAN Bus 에서의 **이상징후 탐지 알고리즘**을 제안함.
  - 피쳐 경량화를 위해 Autoencoder 기반의 데이터 전처리 과정을 적용함.
  - 머신러닝 알고리즘 2개를 적용한 결과, 이상징후 탐지 성능과 학습에 걸리는 시간 자체를 유의미하게 줄이는 것을 확인함.

1. 서론 (Introduction)

1. Introduction

- 차량의 ICT기술 발전과 함께 운전자의 편의성 및 운전 보조를 위한 다양한 기술들이 적용되고 있음.
- 차량에 탑재되는 차량 내 · 외부 네트워크 기술을 통해 Connected Car 및 Vehicle to Everything (V2X) 등이 적용되고 있음.
- 하지만, 운전자의 편의성 및 운전 보조를 위한 차량 네트워크 기술은 사이버 보안에 있어 심각한 위협이 됨. [1]
  - 자율주행 및 미래 차량에 대한 보안성 향상을 위해 다양한 연구와 기술들이 개발되고 있음.
  - 차량 내부 네트워크 (IVN) 중 CAN Bus 는 차량 내부 센서들 간의 성능에 집중하여 개발된 프로토콜이며,
    - 보안성이 고려되지 않아 IVN 에서의 공격 발생 시 이를 파악하기에 제한적인 부분이 있음.
  - 본 논문에서는 IVN 에서의 효과적인 침입탐지(Intrusion Detection)을 위한 딥러닝 기반의 전처리 방법을 제안함.

ICT : Information and Communication Technology

02
Autoencoder

Autoencoder 사용 전략

2. Autoencoder

- IVN 으로써 CAN Bus 에서의 이상 탐지 및 침입탐지를 수행함.
  - 피쳐 경량화를 위해 Autoencoder 를 이용하여 입력 데이터 크기를 축소시킴.
  - 축소된 입력 데이터는 머신러닝 알고리즘인 OC-SVM 에 입력값으로 적용함[2][3].

2.1 데이터 전처리

- CAN Bus traffic 은 수집 시,
  - message 의 식별자인 Arbitration Id, 데이터의 크기인 DLC, 데이터 페이로드를 나타내는 Data field 로 구성됨.
- Autoencoder 에 적용하기 위해,
  - Arbitration Id 는 10 진수의 값으로, DLC 와 Data field 는 각각 bit 단위로 변환하여 입력 벡터를 구성함.
    - 예시 : Arbitration Id 가 16 진수 값으로 '0×123' 이고, DLC 값이 10 진수로 '8' 일때,

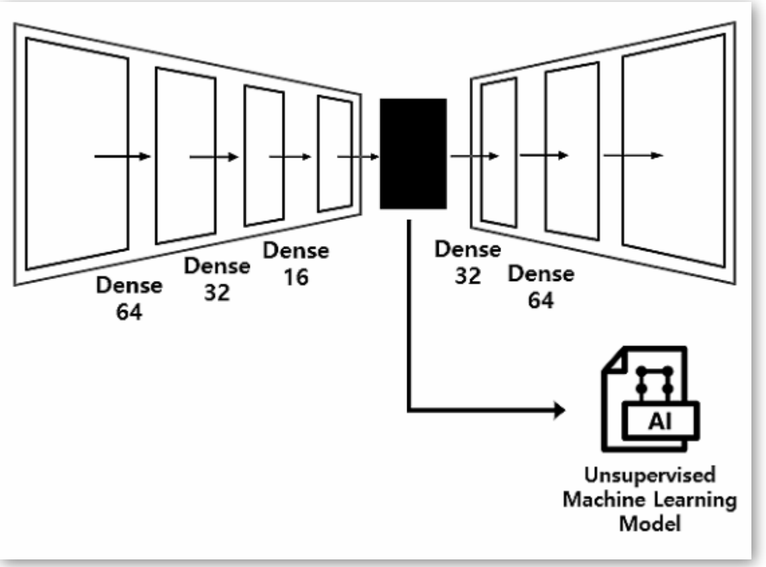
      ⇒ 변환된 벡터로써 Arbitration Id 는 '291', DLC 는 '1000'로 구성됨.
  - Arbitration Id 는 10 진수로 변환되기 때문에,
    - Autoencoder 및 머신러닝 알고리즘에서 계산이 용이하도록 MinMaxScaler 로 정규화를 진행함.

IVN : In-vehicle networking (차량 내부 네트워크) , DLC : Data length code

2. Autoencoder

2.2 Encoder 추출

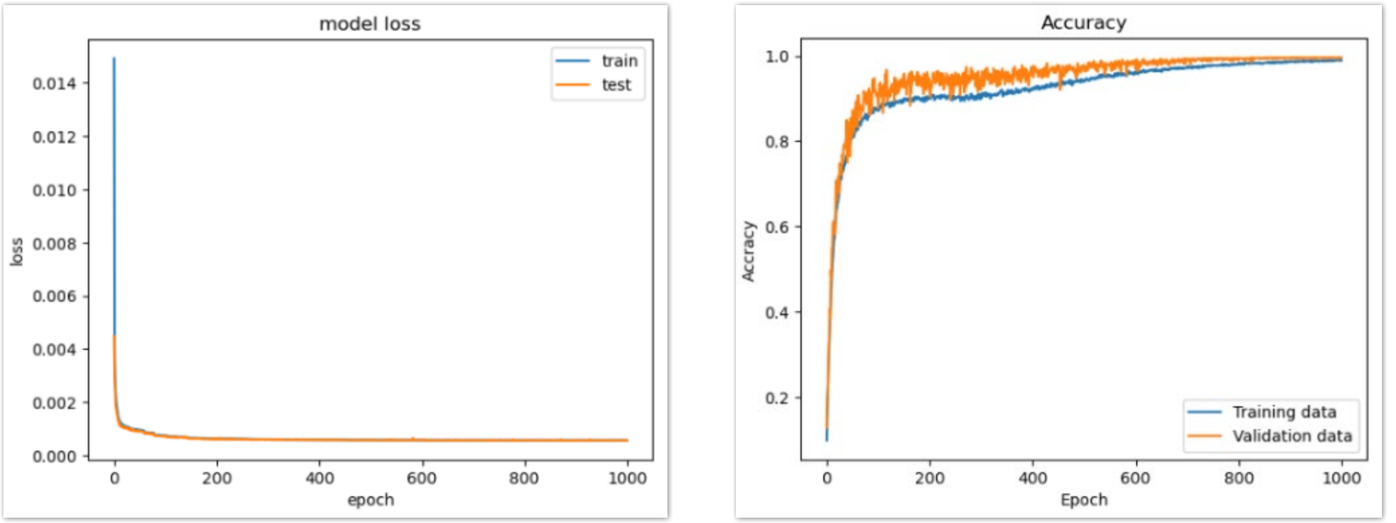
- 제안하는 방법, CAN Bus 에서의 경량화된 이상 탐지를 위해 Autoencoder 를 학습 한 후 Code Layer 를 포함한 Encoder 만을 추출함.
  - Encoder 의 학습을 위해 Fig.1 과 같이, Code layer 를 포함하여 Encoder (FC 64-32-16), Decoder (FC 32-64)를 구성함.
  - Autoencoder 모델에서의 과적합을 방지하기 위해 Encoder 와 Decoder 의 FC 중간 레이어에 BatchNomalization 을 적용함.
  - 해당 Autoencoder 를 설정한 epoch 수 만큼 수행한 후, code layer 를 포함하는 Encoder 를 추출함.



[Flg 1. Autoencoder Model]

FC : Fully Connected Layer

2. Autoencoder



[Fig.2 Autoencoder model loss and accuracy]

2.3 이상 탐지

- 학습된 Autoencoder 에서의 Encoder 를 통해 정상 CAN Traffic 의 정보를 함축적으로 표현하는 데이터를 추출할 수 있음.
- Encoder 의 Code layer 를 통한 출력값은 정상 및 비정상을 탐지할 수 있는 머신러닝 알고리즘에 적용되어 이상징후를 탐지함.
- 이상징후 탐지를 위해 Unsupervised Learning Algorithm 중 OC-SVM 과 IF를 사용함.

OC-SVM : One class Support Vector Machine , IF : Isolation Forest

2. Autoencoder

2.4 모델 학습 전략

- 학습 전략은 크게 두 가지 부분으로 구성됨.
  - 첫번째 단계는 FC 로 구성된 Autoencoder 학습임.
    - Autoencoder 학습 시 정상 데이터를 기반으로 Autoencoder 모델의 weight 값을 학습함.
  - 학습이 완료된 경우, 두번째 단계로써 Autoencoder 의 Encoder 부분과 함께 OC-SVM 또는 IF 를 연결하여 학습을 수행함.

- 학습에 사용하는 데이터셋은 모두 정상으로 처리된 데이터만을 사용하며, Train 및 Test 를 위해 데이터셋을 7:3 비율로 적용함.

03
실험 및 평가

비지도 학습 알고리즘 활용

3. 실험 및 평가

3.1 Dataset 및 실험환경

- 실험을 위해 HCRL CAR HACKING : ATTACK & DEFENSE CHALLENGE 2020 DATASET 을 사용함. [4]
  - 해당 데이터 셋은 Timestamp, Arbitration ID, DLC, Data, Class, SubClass 로 구성됨.

- 본 실험에서는 비지도 학습 알고리즘 적용을 위해 정상 데이터만으로 모델의 학습을 진행함.
- 실험은 Window 10 Enterprise 64-bit 운영체제, Intel(R) i5-12600KF CPU, NVIDIA GeForce RTX 3070 Ti, RAM 32GB 환경에서 수행함.

3.2 성능지표

- OC-SVM 및 IF 알고리즘의 성능을 평가하기 위해 혼동행렬을 사용함.
  - 해당 혼동행렬을 바탕으로 정확도 (Accuracy), 정밀도(Precision), 재현율 (Recall), F1-Score 를 계산함.
  - 혼동행렬에서의 정상 데이터에 대해 정상으로 탐지했을 경우 True positive (TP)로 설정했으며, 실제 공격 데이터에 대해 공격으로 탐지했을 경우 True negative (TN)로 설정함.

3. 실험 및 평가

3.3 실험 결과

- 본 논문에서 제안한 Autoencoder 에서의 Encoder 적용 여부에 따른 학습 결과를 Table 1 에 나타냄.
- OC-SVM, IF 모델은 데이터를 압축했을 때
  - 학습 시, 62.49%, 58.17%의 시간을 감소시킴.
  - 테스트 시, 4.17%, 51.99%의 시간을 감소시킴.
- 게다가, OC-SVM 및 IF 모델은 데이터를 압축했을 때 정확도 및 성능지표가 향상됨을 확인할 수 있음.

	Training-time	Predict-time	accuracy	precision	recall	F1-score
theory	Encode data					
OC-SVM	12381.874375	6482.443503	0.8706	1.0000	0.8431	0.9184
IF	108.136745	79.943310	0.8569	0.8569	1.0000	0.9229
	Original data					
OC-SVM	33008.681408	6764.735523	0.7965	0.7766	0.8630	0.8175
IF	258.541455	166.511220	0.8568	0.8568	1.0000	0.9229

[Table 1. Unsupervised Learning 학습에 따른 평가 성능]

04
결론

OC-SVM, IF

4. 결론

- 본 연구에서는 Autoencoder 를 통한 데이터 압축과 비지도 학습 모델 실험 결과를 나타냄.
- 2 개의 비지도 학습 모델은 Autoencoder 의 데이터 압축 유무에 따라, 시간 단축 면에서 높은 성능을 나타냄.

- 향후 연구 계획으로 보다 효과적인 데이터 압축 알고리즘을 통해 경량화된 침입탐지 알고리즘을 연구할 계획임.

5. 참고 문헌

[1] 한국인터넷진흥원, “자율주행차 보안모델”, 한국인터넷진흥원, 2022. (최종 열람일 : 2023 년 3 월 30일). https://www.kisa.or.kr/2060205/form?postSeq=14&page=1#fnPostAttachDownload

[2] Yan, Binghao, and Guodong Han. "Effective feature extraction via stacked sparse autoencoder to improve intrusion detection system." IEEE Access 6 (2018): 41238-41248.

[3] 민병준, 유지훈, 김상수, 신동일 and 신동규. (2021). 오토 인코더 기반의 단일 클래스 이상 탐지 모델을 통한 네트워크 침입 탐지. 인터넷정보학회논문지, 22(1), 13-22.

[4] HCRL, “CAR HACKING: ATTACK & DEFFENSE CHALLENGE 2020”, HCRL, 2020 (최종 열람일 : 2023 년 3 월 30 일) https://ocslab.hksecurity.net/Datasets/carchallenge2020

THE END

Anomaly detection method of IVN using lightweight features based on Autoencoder

감사합니다.