산업제어시스템 보안위협 탐지 AI 경진대회(HAICon 2021)
코드 설명자료

1. 요약

- **팀 소개**
  - 팀명: GNOEYHEAT
  - 팀원 소개

    GNOEYHEAT: (김태형, 고려대학교 산업경영공학부 대학원생, 융합 데이터 분석 및 인공지능 연구실 - 한성원 교수님)

    히우우웅: (노희웅, 고려대학교 산업경영공학부 학부생)

    후라이냠냠: (김지홍, 고려대학교 산업경영공학부 학부생)
- **문제해결을 위한 전략 및 수행했던 내용**
  - Preprocessing: Duplicated categorical variable drop, 1차 차분.
  - Modeling: Stacked GRU, Stacked LSTM, 1D CNN, Autoencoder based on RNN, Transformer, SCINet.
  - Post-processing: Lowpass filter, Moving average, Range check, Gray area.
  - Ensemble: Voting, Blending.
  - 2-stage: Isolation Forest, Gaussian Mixture Model.
- **참고한 논문 또는 최신 기법**
  - Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin , Attention is All You Need, NeurIPS 2017.
  - Minhao Liu, Ailing Zeng, Zhijian Xu, Qiuxia Lai, Qiang Xu, Time Series is a Special Sequence: Forecasting with Sample Convolution and Interaction, 2021.

2. 데이터 전처리방법

- **문제해결을 위한 전략 및 수행했던 내용**
  - 문제해결을 위해 다양한 전략을 수행하였으나 시간 부족으로 좋은 효과를 얻지 못함. 때문에, 사용 이유에 대해 간략하게 작성함.
  - Duplicated categorical variable drop: 동일한 값을 갖는 변수로 학습에 영향을 미치지 않는 변수를 제거함.
  - 1차 차분: 추세를 제거한 변수를 생성하여 학습에 사용함. 학습 데이터의 구간에 따른 편향성을 제거하고자 함.
- **탐색적 데이터 분석(EDA) 방법과 주요 결과**
  - 값이 동일한 변수를 제거하고 학습을 진행했으나 지우기 전/후 차이가 크지 않았기 때문에 최종 모델은 모든 변수를 학습한 모델임.
  - 위의 문제해결을 위한 전략을 수행하기 위해 training set과 validation set의 차이를 찾고자 함. Train 1&2 dataset은 validation 및 test dataset과 유사하나 이에 대한 분석 결과를 모델링에 활용하지는 못함.

3. 모델 구성 및 학습 방법

- **모델 구성**
  - Baseline으로 제공된 3층의 Stacked GRU 모델 사용을 사용함.
  - 차이를 갖는 하이퍼 파라미터는 다음과 같음.
  - N_HIDDENS: 200,
  - batch_size: 4096,
  - epochs: 200,
- **사용을 고려한 다른 후보 모델과 최종 모델의 선정 방법 또는 이유**
  - Stacked GRU, Stacked LSTM, 1D CNN: Many-to-One 모델로 다양한 baseline을 실험하고자 함. Naive한 1D CNN보다 RNN 계열이 쉽게 예측 성능에 도달함.
  - Autoencoder based on RNN: Many-to-Many 모델로 VAE 등의 다양한 방법을 적용하고자 하였으나 Validation의 성능이 떨어져 사용하지 않음.

- Transformer, SCINet: 코드 공유 게시판을 참조함. Input sequence가 크게 길어야 할 이유를 찾지 못하여 RNN 기반의 모형을 사용함. SCINet의 Idea를 참고하여 모델링하였으나 시간 부족으로 제거함.

- **모델 학습을 위해 사용한 성능 메트릭의 선정 방법 또는 이유**

  - HAICon2020 수상작에서 많이 사용한 MSE loss를 사용함. L1 loss또한 사용하였으나 학습의 수렴이 잘 되지 않고 모델의 성능이 좋지 못하였음.

- **학습 성능 개선을 위해 추가로 수행한 내용, Anomaly scoring 방법**

  - Isolation Forest, Gaussian Mixture Model: Training dataset으로 meta dataset을 만들어 Anomaly Score를 산출함. 기존의 평균값으로 만들어진 score가 아닌 학습된 모형을 통해 score를 산출할 수 있음. 기존의 평균값을 통한 score보다 상대적으로 Validation score를 평활하게 만들어 주는 효과를 가짐. 시간 부족으로 최종 점수를 올리는데 사용하지 못함.

- **이상 판단을 위한 threshold 결정 방법과 이유**

  - Validation set을 기준으로 eTaPR score가 가장 높은 값의 threshold를 선택하였으나 이상 탐지에 대한 recall값이 매우 낮아 임의로 조정하며 # of detected anomalies와 anomaly instance의 개수를 보며 약 0.008 정도의 threshold를 결정함.

4. 성능 최적화 방법

- **성능 최적화(하이퍼 파라미터, random seed 변경 등)을 위해 수행한 내용**

  - Lowpass filter, Moving average, Range check, Gray area: HAICon2020의 Post-processing 방법을 사용함. 이전 대회에서 검증된 방법으로 성능을 쉽게 올릴 수 있음.

  - Lowpass=0.1, moving_average: 60, range_check: 30.

- **그 외의 성능 최적화를 위해 사용한 다른 방법들**

  - Voting, Blending: Seed 값과 Threshold에 영향을 최소화하기 위해 사용함.

- **시도한 방법 중에 유효한 것과 그렇지 못한 방법에 대한 구분 및 비교**

  - Lowpass filter는 예측 score의 변동성을 제거하는데 매우 유효함. 유효하지 않은 방법

에 대한 구체적인 검증을 하지 못함.

5. 결과

- **최종 모델에 대한 주요 예측결과와 이상 탐지결과**
  - Test data에서 약 7500건 정도의 이상치를 평균적으로 탐색함.
  - Seed 값에 따라 학습이 잘 이루어지지 않을 때 더 많은 epoch으로 학습하면 후처리 방법에 따른 비슷한 성능을 가지는 것으로 확인함.
  - Threshold 값에 민감할 수 있으나 Validation의 anomaly 개수를 확인하면 도움이 됨.
- **최종 모델에 대한 성능 분석을 통한 장단점**
  - 적절한 baseline으로 복잡한 모델링 이전에 cost가 적은 후처리 방법을 통해 좋은 성능을 보장할 수 있는 장점을 가짐.
  - 하이퍼 파라미터가 많아지며 threshold에 대해 매우 민감한 단점을 가짐.
- **예기치 않는 결과나 실패했던 방법(전처리, 모델 구성, 최적화 등)**
  - 위에서 서술한 모든 방법.
- **이상탐지 성능을 추가로 개선할 수 있을 것으로 기대되는 부분이나 아이디어**
  - 위에서 서술한 모든 방법.

6. 건의사항

없습니다. 좋은 대회 참여할 수 있었습니다. 감사합니다.