ReportIT: Improving Insider Threat Detection Model Explainability Through Report Retrieval

Sameer Khanna [1]

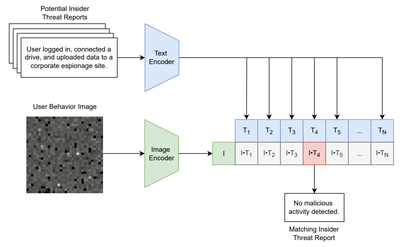
[1]Computer Science, Stanford

Background

- Insider threats are costly, hard to detect and cause catastrophic damage. 30% of confirmed breaches today involve insiders, with an average cost of 11.45 million USD.
- Poor interpretability of machine learning based solutions has led to high skepticism, leading most real-world deployments to not use machine learning techniques despite their higher detection accuracy.
- Insider threat data contains sensitive information, leading compiled datasets to be private in nature. Such data requires expert annotation, making it harder to deploy insider threat detection systems.

Approach

- ReportIT takes the behavior image representations that lead to great insider threat detection performance and generates (or retrieves) a report detailing in plain English the reasoning behind a model's classification.
- The ReportIT image encoder can be finetuned for use as a threat detection model with great label efficiency and competitive accuracy, increasing the ease of setup and the effectiveness of deployed insider threat detection systems in industry.



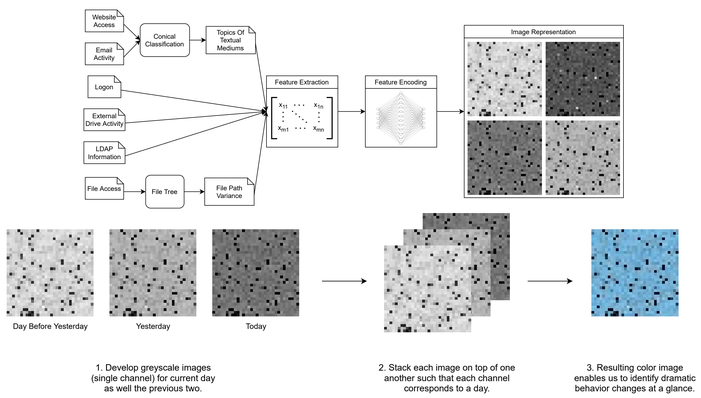
Novelty

There are a couple of things that make this project novel:

- This is the first time contrastive learning has been proposed for insider threat detection.
- To our knowledge, this is the first time image-text contrastive learning has been proposed to be used for image encodings as opposed to true images like natural images or medical images.
- To our knowledge, this is the first time image-text contrastive learning has been proposed for such a highly imbalanced data space, especially one where we there is far greater diversity in the minority class than the majority class.
- We propose two nove'l contrastive learning training methodologies.

Generating Behavior Images

- We follow the approach used by the current state-of-the-art for insider threat detection.
- Features are extracted from log files, enabling us to get a snapshot of the user's behavior.
- A Sparse AutoEncoder is used to project the extracted feature space to 1024 dimensions.
- Data is reshaped to be a 32x32x1 greyscale image.
- This data is combined channel-wise with the previous two day's representations which provides contextual information for the given day. This leaves us with the final 32x32x3 color behavior image.



Report Retrieval

- A behavior image is matched with the correct report by identifying the text encoding with the highest cosine similarity with the given image encoding.

$$\text{Index of Retrieved Report} = arg\,max_i \frac{I \cdot T_i}{|I||T_i|}$$

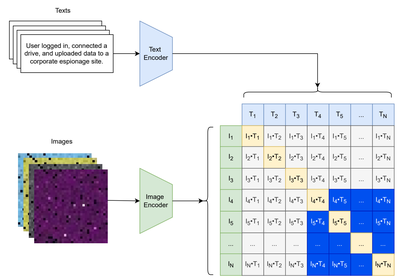
Training Contrastive Learning Models

- The image and text encoders are trained such that the cosine similarity is high for true image-text pairs and low for false image-text pairs. The first part of the loss function improves image-text alignment; the second improves text-image alignment.

$$\mathcal{L} = \frac{1}{2N}\sum_{i=1}^{N} -\log\left(\frac{\exp\left(\frac{I_i \cdot T_i}{|I_i||T_i|}\right)}{\sum_{k=1}^{N}\exp\left(\frac{I_i \cdot T_k}{|I_i||T_k|}\right)}\right) - \log\left(\frac{\exp\left(\frac{I_i \cdot T_i}{|I_i||T_i|}\right)}{\sum_{k=1}^{N}\exp\left(\frac{I_k \cdot T_i}{|I_k||T_i|}\right)}\right)$$

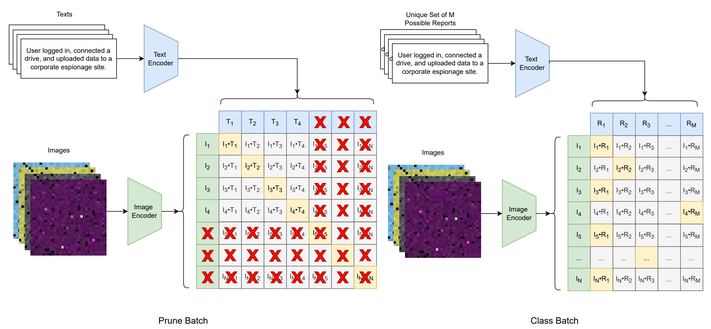
Problem With Traditional Contrastive Learning Approach

- We have low diversity of reports in our majority class and high diversity of reports in our minority class.
- Leads to false negative image-text pairs (shown in blue) if multiple correct reports are in the same training batch ($T_4 = T_5 = ... = T_N$).
- Contrastive learning will try to increase the distance between images and reports it should be aiming to bring close together.



Proposed Solutions

- PruneBatch prunes image-text pairs in the batch, removing false negative pairs in the process. However, this approach removes data from each batch which can increase model brittleness.
- ClassBatch treats the text related to a given image as a class, where the class number corresponds to the index of the given report within the set of all possible reports. Now, each behavior image is compared against all possible reports.



- The loss function for ClassBatch only aims to perform image-text alignment.
- We add a W_{T_i} term that scales the loss to be balanced for each report category.

$$\mathcal{L}_{ClassBatch} = -\frac{1}{N}\sum_{i=1}^{N} W_{T_i} \log\left(\frac{\exp\left(\frac{I_i \cdot R_{R==T_i}}{|I_i||R_{R==T_i}|}\right)}{\sum_{r=1}^{R}\exp\left(\frac{I_i \cdot R_r}{|I_i||R_r|}\right)}\right)$$

Baselines

- The traditional method for obtaining text information describing an image is via treating the task as an image captioning problem, thus our baselines consist of image-captioning models.
- Focus is on evaluating NLP architectures; thus we always use the image encoder ViT.
- For our baselines, we evaluate various text decoder architectures: BERT, BART, GPT-2, and RoBERTa.

Data

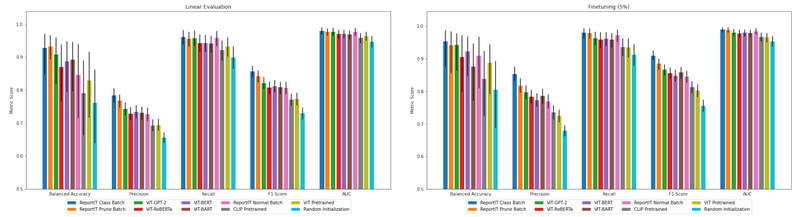
The CERT Insider Threat center together with ExactData LLC analyzed 1,154 actual insider incidents in the United States to create the largest public repository of insider threat scenarios. There are three scenarios of attack that occur in the dataset:

1. User obtains sensitive information they subsequently upload to Wikileaks.
2. User browses job sites looking for a job, stealing confidential information and leaving as soon as they find one.
3. User installs a keylogger on target computer and masquerades as the target.

Text-Guided Learning Improves Image Encodings

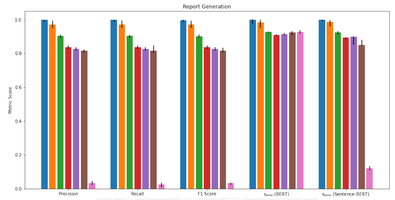
- Here we showcase TSNE visualizations for our various models. ReportIT models, baseline models, as well as pretrained and random initialization ViT encoders.
- Training with text improves encoding quality, with ReportIT-ClassBatch having the best separation between malicious data and benign data.

- The utilization of text leads to great label efficiency as seen in the results for the evaluation and finetuning tasks.
- ReportIT PruneBatch and ClassBatch both outperform all other models.



Contrastive Learning Leads to Accurate Report Generation

- The normal batch configuration is the only ReportIT model to perform worse than the image captioning baselines.
- ViT-GPT2 outperforms the other models, which is to be expected due to GPT2 being specifically designed for the text generation task.
- ClassBatch again outperforms all alternatives.



sameerk@cs.stanford.edu