**Exploring Graph based approaches to Network Anomaly Detection**

1. **Abstract**

Anomalies can appear when working on networks of any size. Some examples of anomalies are abnormal login attempts, unreachable web pages when web crawling, and network malware intrusions. In May 2014, a total of 145 million Ebay users had their names, addresses, and encrypted passwords compromised from a cyber attack. Detecting intrusions is essential to protecting a business and its users. Currently, many large companies like Google use statistical techniques to detect anomalies. A machine learning model trained on similarity scores between graphs could provide faster and more accurate detection of network intrusions. Also, graph based deep learning approaches have been shown to perform well in classification problems dealing with graphs. The goal of this research is to investigate the use of graph based machine learning approach in detecting network intrusions. It is our goal that this approach will outperform some of the state of the art network intrusion models.

1. **Introduction**

Network traffic data can be modelled as graphs consisting of nodes and edges. An edge between two nodes represents the nodes being connected. In this case, the nodes consist of information of all devices connected to a network. Edges indicate which devices are connected to other devices. Snapshots can be taken of the network graph and compared to one another to generate a similarity score between the two graphs. A higher similarity score means the graphs are more alike. Subgraphs from the larger graph are taken and compared. An anomaly will be present if the subgraph is not exactly isomorphic but is isomorphic within a certain percentage of the given normal execution structure. This is because to avoid being detected, an intruder will attempt to mimic normal network activity.

Recent studies have shown that machine learning is a better approach to detecting network intrusions. More specifically, since network data can be modelled as graphs, we can leverage the power of graph-based machine learning approaches by incorporating deep learning graph based models to increase the accuracy and precision of our network intrusion detection algorithm. Backed by prior literature, it is our intuition that graph based deep learning models will lead to an increased accuracy of network intrusion detection when compared to other intrusion detection approaches.

1. **Methods**

The dataset for our evaluation will be derived from readily available open source datasets that consist of enterprise level network traffic. This traffic data was collected in a realistic environment consisting of normal system network execution and also some network malicious exploits. This dataset will be used to study our various proposed approaches by modelling them as network graphs. Consequently, machine learning models applied directly to graphs is a challenge in itself. Therefore, the graphs will be embedded into vector representations using different embedding approaches as described in the literature. These embeddings will be used as input to some graph-based deep learning models to produce results with high accuracy. In addition, we will explore and approach in which each embedded graph is compared to other graphs using similarity scores. Additionally, the normal substructure of a graph will be computed at this time. All of the results will be compared to see which is best at detecting network anomalies.

1. **Timeline**

Week 1: Data Processing/Cleaning

Week 2: Research on graph based embedding approaches to determine which embedding

approach is best; Develop code to embed the network graphs

Week 3: Develop code to embed the network graphs

Week 4: Develop code to calculate similarity scores/ Deep learning implementation using graph

embeddings

Week 5: Create deep learning models that takes similarity scores as inputs and outputs whether

there is an anomaly present or not

Week 6: Create similarity scores for graphs & Start writing final report.

Week 7: Train models

Week 8: Train models

Week 9: Test and compare the models

Week 10: Draw conclusions

No change to plans if research must be remote.

1. **Feasibility**

Previous work has been done on network embeddings and calculating similarity scores. Additionally, there has been work using similarity scores to detect anomaly detection. Therefore, this project needs to determine which embedding technique is best for detecting these anomalies in a computer network. Graph based anomaly detection is a more feasible approach since network data can be modelled as graphs. Graph Machine learning has been shown to perform with high accuracy in other areas such as social networks. It is realistic to use a machine learning model rather than algorithms to detect an anomaly. Therefore, this project is reasonable.

1. **Broader Impacts**

The Internet itself can be seen as a large network. Detecting anomalies with this technique may be applicable to all networks such as social and computer networks. As previously mentioned, Google uses statistical and machine learning techniques to detect anomalies when creating their search engine graphs. Therefore, this research could be used on a large scale to detect anomalies and allow the search engine to respond accordingly. In addition, it could be adopted by network administrators of computer networks of all sizes to detect network intrusions.

1. **Trajectory/Future Goals**

My primary interests in technology are cybersecurity and machine learning. I want to work within one or both of these fields after graduating college. Being able to combine my two interests would be an important experience for both fields. In the summer of 2021, I plan to apply for the Department of Defense SMART Scholarship. This research would certainly help distinguish me as a candidate. Furthermore, I am also considering an academic route of pursuing further education and conducting research. This opportunity would allow me to determine whether or not a career in the academic field suits me.