Malware Detection on Highly Imbalanced data through Sequence Modeling - Summary - Sidharth Anil

Presentation

The core theme of the paper is how Sequence Modeling performs on Highly imbalanced data. It is made relevant by the fact that there are many contexts in the world where there is a high imbalance between malware and goodware, and a model that functions well taking this into considerations hence could have more merit. The presentation started off with a few introductory principles in malware detection system. This includes the rule-based system being outdated due to high false positive rates and a need for skilled individuals to come up with the heuristics. Furthermore some machine learning concepts that were used in the paper were briefly discussed including NLP, RNN, LSTM and BERT.

One of the principles the primary premise of the paper revolves around is how models designed for Natural Language Processing(NLP) could be utilized for malware analysis. Two such models that were implemented and evaluated by the authors of the paper are LSTM and BERT. Brief explanations were provided on both LSTM and BERT, on their design and how it applies to the context of Malware Analysis using dynamic analysis logs. LSTM is a type of neural network that containts a cell that retains historical information with the help of 3 gates - input, forget and output. BERT is a model that falls under Masked Language Model(MLM). A basic explanation of BERT is that it would cover up a few random words in a sentence and then train the model to predict these words using bi-directional information. Another key point in BERT is that the model can be pre-trained using large general datasets and can later on be fine-tuned for specific contexts by adding an additional output layer with minimal training required.

Dynamic analysis of Android applications provides a high-level sequential activity log that can be utilized for determining whether an application is malicious or not. The authors of the paper acquired the dataset from Palo Alto Network's Wildfire service, out of which the sequential higher level activity logs were extracted. The sequential nature of the logs, and the log entries miles apart providing context for each other and indicating a certain type of behaviour is very similar to how natural languages work. When analyzing an essay, a sentence from the first paragraph may completely alter the context and hence meaning of a claim made in the subsequent paragraphs. This behavioural pattern is the reason why Natural Language Models were tried out in the domain of malware analysis.

Experiment Results

Different types of Natural Language Models were tried out and their performance evaluated on unbalanced data. The results of each of these experiments were presented to compare between them

and understand the situations in which a model performed well. Some of the key points are mentioned below.

- A bag-of-words analysis (non sequential) using a Term-Frequency-Inverse Document Frequency had an F1-score, accuracy and precision of 0.96 on fairly balanced data.
- An LSTM Model with a sequence length of 50 had an F1-score of 0.985 and Accuracy of 0.975 on balanced data
- On unbalanced data, LSTM did not perform well enough. For a dataset with 2% malware samples, LSTM (with DeepLog) achieved an F1-score of 0.65.
- A BERT model pre-trained on natural language data with 3 epochs of fine tuning achieved an F1-score of 0.91 on a 2% malware dataset
- A BERT model that was trained from scaratch on Android malware datasets achieved an F1-score of 0.880 on a 2% malware dataset.

Discussion

- The class discussed about the real world imbalance between the occurence of malware and goodware, and the importance of tuning the machine learning model to this. It was also noted that the distribution would vary from context-to-context. That is, a model running in the labs of an antivirus company would be facing an excess of malcious files (due to suspicious files being filtered and sent to them), whereas the distribution would be completely towards the opposite direction in a regular user's machine that would encounter much more goodware than malware. This would imply the need for different models in different situations.
- Another key point of discussion was why low false positive rate is important for a malware detection system. The class went over how multiple fasle positive announcements, would make the user lose trust in the detection system. Counterpoints were made on prioritizing detection of as many threats as possible even if it compromises usability to an extent. The discussion concluded on the idea that security was built not as the end-goal for a product but as way to support the product. Therefore, in most regular contexts, usability shouldn't be largely affected by security. Unless, it is a high security requiring context (like top-secret government applications) in which case the rules might change. Context Matters
- The imbalance between the frequency of benign tokens in the logs compared to the tokens that contribute to maliciousness mgiht cause issues for machine learning models. Statistical values like TFIDF ensures that more weight is not given for benign tokens, just because they appear more frequently.