SEMANTIC EXTRACTION FROM CYBERSECURITY REPORTS USING NLP

SYNOPSIS/PROPOSAL

Submitted By:

Anukriti (2K17/IT/027)

Anurag Mudgil (2K17/IT/029)

Armaan Dhanda (2K17/IT/031)

PROBLEM STATEMENT

Detecting which sentences are relevant to malware and cyber-security using Natural Language Processing has a lot of potential to benefit security researchers but surprisingly, not much work has been done in applying NLP to the security domain. In our project, we are planning to classify sentences as being relevant to malware or irrelevant to malware using different approaches. Using classification and annotation of the malware related text we will be able to better analyze the behavior of the malware and its capabilities. We can segregate malware related text from the irrelevant text. An application of this model can be the analysis of behavioral reports of malware, thus we can cluster similar types of malware easily and better understand its attributes.

PLAN OF EXECUTION (PROPOSAL)

❖ COLLECTION OF DATA (Dataset to be used):

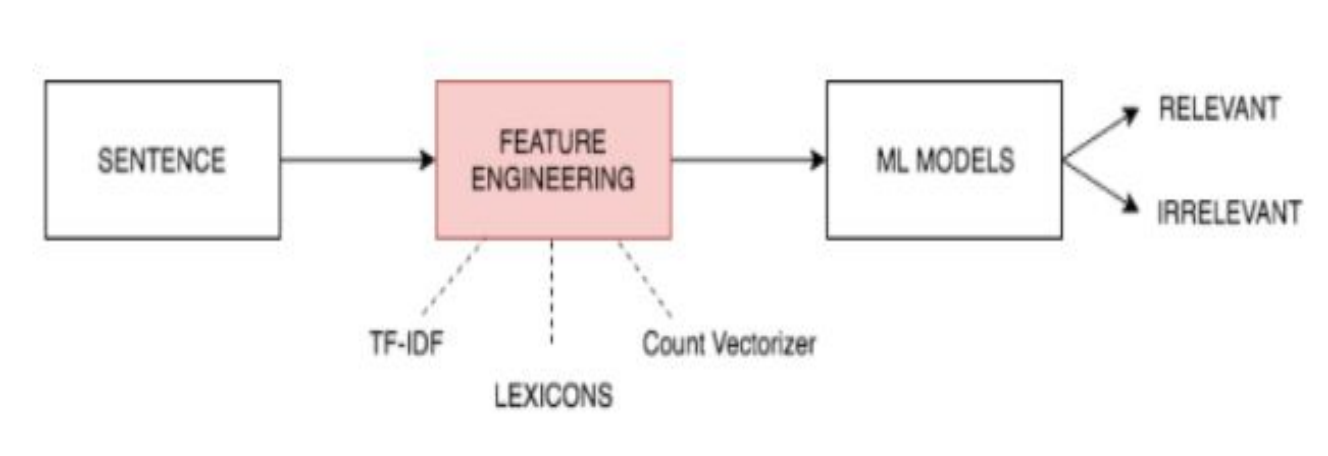
We will be using annotated malware and Advanced Persistent Threat (APT) reports with attribute labels from the Malware Attribute Enumeration and Characterization (MAEC) vocabulary (Kirillov et al., 2010). The APT reports in our dataset will be taken from APTnotes, a GitHub repository of publicly released reports related to APT groups (Blanda, 2016). It provides a constant source of APT reports for annotations with consistent updates. We will be choosing reports from recent years for annotation purposes.

Preparation of dataset through data pre-processing:

The APT reports from APTnotes are in PDF format, hence we will be using the PDFMiner tool to convert the PDF files into plaintext format. We also will manually remove the non-sentences, such as the cover page or document header and footer, before the annotation. Hence only complete sentences will be considered for subsequent steps.

❖ FEATURE ENGINEERING:

On observation, we found out that the dataset is imbalanced and relevant sentences are very few as well which makes the task of feature engineering quite crucial for which the following features are used to extract information from the data.



1. **Lexicons:** Words obtained from The MAL: A Malware Analysis Lexicon : A list of malware related words (3000 words) and scraping books like Practical Malware Analysis by Michael Sikorski.
2. **CountVectorizer:** Tokenize a collection of text documents and build a vocabulary of known words, but also encode new documents using that vocabulary. An encoded vector is returned with a length of the entire vocabulary and an integer count for the number of times each word appears in the document.
3. **TF-IDF Vectorizer:** This is an acronym that stands for "Term Frequency – Inverse Document".
   a. Term Frequency: This summarizes how often a given word appears within a document.
   b. Inverse Document Frequency: This downscales words that appear a lot across documents.
4. **Singular-Value Decomposition:** SVD is a matrix decomposition method used for reducing a matrix to its constituent parts in order to make certain subsequent matrix calculations simpler. It is used for dimensionality reduction.
5. **Word Embeddings:** Word embedding is a vector representation of the words that are used to capture the context of a word in a document followed by its semantic and syntactic similarity. We will be using pre-trained Stanford GloVe or Google-News-Word2Vec embeddings or some domain-specific malware embeddings extracted from the APT Threats.

❖ CHOICE OF MODEL/CLASSIFICATION ALGORITHM:

The aim of our models is to extract the malware related tokens from the sentence and classify them into one of the relevant or irrelevant. This can be divided into 2 major subtasks:

1. **MALWARE THREAT CLASSIFICATION:**
   For this task, we will be trying out different models and will ultimately provide a detailed analysis between all our approaches. Some of the techniques that we have planned to implement are as follows:

   Traditional Machine Learning Models: Multinomial Naive Bayes, XGBoost, Random Forest, Logistic Regression as well as Ensemble of CRF (Conditional Random Fields) and Naive Bayes.

   Deep Learning Models: Bidirectional Long Short Term Memory with hyperparameter tuning (Parameter Optimization Algorithms)

2. **MALWARE TOKEN IDENTIFICATION:**
   Conditional Random Fields is a class of discriminative models best suited for prediction tasks where contextual information or state of the neighbours affect the current prediction. CRFs find their applications in named entity recognition, part of speech tagging, gene prediction, object detection problems, to name a few. The bag of words (BoW) approach works well for multiple text classification problems. Hence, it comes to rescue for problems such as entity recognition, part of speech identification where word sequences matter as much, if not more. Thus, we will be implementing our CRF model which, given: some feature extractors, weights associated with the features as well as previous labels can easily predict the current label.

❖ PERFORMANCE METRICS:

Combination of below mentioned metrics:

- ❖ Accuracy
- ❖ Precision
- ❖ F1-Score
- ❖ Recall

- ❖ Area under ROC curve

TECHNOLOGY STACK

- ❖ **FOR CLASSIFIER/MODEL:**
  - **Libraries:** pandas, itertools, bs4 (BeautifulSoup), sklearn (classifiers, metrics, model_selection), matplotlib (visualization) etc.
  - **Tools:** Anaconda, WEKA
  - **Language:** Python

SOME BASE RESEARCH PAPERS AS REFERENCES:

(PDF) Flytxt_NTNU Identifying and Classifying Malware Text Using Conditional Random Fields and Naïve Bayes Classifiers

HCCL: An End-to-End System for Sequence Labeling from Cybersecurity Reports

(PDF) Devising Malware Characteristics using Transformers

Digital Operatives: Using dependency features for malware NLP

UMBC: Understanding Text about Malware

MalwareTextDB: A Database for Annotated Malware Articles