AI Threat Intelligence

Nikhil Anand, Harshvardhan Singh, Rohan Varma

Team ART (Automated Research Team)

***Abstract*— The client is McAfee, a cybersecurity company. As such, they have to perform constant research in order to keep up with the latest threats. The project aims to streamline their research process. The client has informed us that many cybersecurity researchers tweet about their findings. Accordingly, the project goal is to create a system that scans through tweets and points the clients towards new posts relevant to developments in cybersecurity research. The project will be deemed successful if the system is easily navigable and identifies relevant tweets with an accuracy of 80%.**

Table of Contents

1. Introduction

i. *Premise*

The client for this project is McAfee, a company that focuses on cybersecurity. They produce products that protect computing systems against malware and other threats. One of the largest challenges they face is ensuring that their products are up to date and capable of defending against the latest threats. To meet these requirements in a rapidly evolving cybersecurity landscape, the client continuously performs thorough research; needless to say, this constant search for relevant news is an arduous task. Our aim for this project is to make this research process less labor intensive by automating portions of it.

ii. *Scope*

Automating the client's research process is a very broad problem, and we will be focusing on a specific portion of it. The client has indicated to us that the cybersecurity community often uses Twitter as a platform to notify others about new developments in the field. However, there are millions of new tweets every day and only a small portion of them are relevant to the client. As such, we will be creating a tool that performs the first round of filtering for these tweets. Our goal will be to create an application that scans Twitter and gives the client a list of tweets that are related to new developments in cybersecurity research.

iii. *Definitions, Acronyms, and Abbreviations*

IoC - Indicators of Compromise are pieces of forensic data that are used to identify malicious activity within a system or network. Some examples of these indicators may be unusual web traffic, increase in database read volume and suspicious file changes.

API - Application Programming Interface is the interface used for communication between a client and a server. Often used for client side software development.

UI - User Interface is the visual elements that users view and interact with. This includes all the buttons, icons, and pages that a user may see on a web interface.

AWS - Amazon Web Services is Amazon's on demand cloud platform.

GCP - Google Cloud Platform is a suite of cloud computing services offered by Google.

Pelican Cluster - Pelican Cluster is Oregon State Universities Linux based GPU cluster offered to students for remote computation for machine learning applications.

SOCMINT - Social Media Intelligence refers to the collective tools and techniques that companies use to monitor social media sites and extract useful data about emerging threats and opportunities to gain a competitive advantage.

TWINT - TWINT is an advanced Twitter scraping tool built in Python that allows users to scrape tweets from Twitter profiles without using the official Twitter API.

2. Overall Description

i. *Product Features*

To meet the project goals, the system will have to have the following capabilities:

1. Gain access to tweets on Twitter and tabulate information such as content, poster, user, date, etc.
2. Store information on large amounts of tweets
3. Filter for tweets relevant to the client's cybersecurity focus
4. Display relevant tweets in an easily navigable UI

ii. *Intended Users*

The intended users for our project are McAfee engineers who are utilizing our platform for searching Twitter for emerging threats that would be actionable on their part. These engineers are looking for information about threats that McAfee should be aware of, and would be able to improve their time to response against these threats

iii. Design and Implementation Constraints

A. Model Accuracy
   a) The accuracy of the model is directly related to the quality and size of the dataset.
   b) Manually creating the dataset and classifying tweets as relevant and irrelevant can introduce bias to the model as data curators are not experts in security.
   c) There will be noise in the dataset due to differences in opinions about tweet classification.

B. Compute Resources
   a) The training efficiency will depend directly on the compute power granted to it by McAfee or Oregon State University.
   b) Slow training time will lead to less modifications being tested which may prevent us from fully optimizing our model.

3. System Requirements

In order to meet the product features described above, the end system will have a scraper, data store, tweet classifier, and user interface. Requirements governing what is expected from each of these components are provided below. In order to train the tweet classifier, a training dataset must be acquired. As the data curation process leading to this is also governed by requirements, they have also been listed below.

i. Scraping

A. Summary

This component will be utilizing a scraping tool to bring tweets into the database. It will initially be used procure training data for the model, and later to find tweets for the model to classify

B. Functional Requirements
   a) Scraper can filter tweets based off of hashtags or specified keywords
   b) Scraper will have a dictionary of keywords and hashtags relevant to security tweets
   c) Scraper can search for tweets within a specified time range
   d) Scraper can access tweet content and user information
   e) Scraper can send search results to a specified database

ii. Database

A. Summary

This component will be the repository for tweets the scraper finds as well as a repository for tweets the model classifies as relevant. There will be two databases, one that holds training data for the model, and one for the end product that holds tweets that are to be classified by the model.

B. Functional Requirements
   a) Each database can store at least 30000 tweets
   b) Database can distinguish between unclassified and classified tweets
   c) Database can communicate with scraper, model, and UI

iii. Data Curation

A. Summary

Although not a system component, data curation is a critical process as it is how we will acquire our training set for the model. To ensure efficiency, initial data will be procured by having the scraper filter based off of specified keywords and hashtags. Then, the team will manually label these tweets as relevant or irrelevant

B. Process Requirements
   a) Team members labelling data must be trained to understand what McAfee considers relevant in the security space
   b) Keywords and hashtags used to provide initial filtering of tweets should be informed by McAfee's current research process
   c) Amount of training data should be commensurate with performance requirements for model accuracy. A minimum of 10000 tweets should be procured for training
   d) Training data should include both relevant and irrelevant data

iv. Tweet Classifier

A. Summary

This component will handle classifying the tweets as relevant or irrelevant. It will be trained utilizing the dataset obtained via the data curation process outlined above

B. Functional Requirements

a) Model can classify tweets as relevant or irrelevant with 80% accuracy on test sets
b) Model can draw inputs from database
c) Model can send classification results back to appropriate tables in database

v. UI

A. Summary

This component will tie together the other components into an interface that allows the user to find relevant tweets from a specified timeframe

B. Functional Requirements

a) UI can communicate with scraper, database, and model
b) UI can display tweets classified as relevant by the model
c) UI is easily navigable and uncluttered

4. PERFORMANCE REQUIREMENTS

i. Qualitative Requirements

The main qualitative metric will be the "quality" of the feed provided to the user after classification. If the system can find tweets that a threat researcher thinks is actionable intelligence for research then the system is meeting expectations.

ii. Quantitative Requirements

A. The system should classify tweets relevance with at least 80% accuracy on test sets
B. The system shouldn't take more than 30 seconds to search, classify, and display a feed for tweets in a week's time range.

5. GANTT CHART