M.Sc Project Final

Interactive Malware Analysis Using Roberta Based Model

Utkarsha Shukla (THA079MSISE018)

Supervisor:

Er. Om Prakash Mahato

Department of Electronics and Computer Engineering
Institute of Engineering, Thapathali Campus

August, 2024

Presentation Outline

- Motivation
- Background
- Problem Statement
- Objective of Project
- Scope of Project
- Originality of Project
- Potential Applications
- Literature Review
- Methodology
- Results
- Discussion and Analysis
- Discussion and Analy
- Discussion and Analysis
- Tentative Timeline (Gantt Chart)
- References (IEEE format)

Background[1]

- Malware has become a pervasive threat in the digital world, causing significant damage to individuals, organizations, and governments like financial lossed, data breach, etc.
- Modern Malware is increasingly sophisticated in nature, new and advanced techniques are required to evade detection, such as polymorphism and advanced obfuscation methods.
- Traditional malware detection methods struggle to detect new malwares and generates high false positive rate thus relying heavily on static signatures.

Background[2]

- Machine learning models can extract and learn from various features of malware, such as system calls, network traffic, file characteristics,etc.
- These models can generalize from the training data to detect new malware variants.

Motivation

•There is an imperative need to enhance the accuracy, precision and efficiency of malware detection and analysis processes due to growing complexity of modern malware.

•SecureBERT model have the capacity to recognize relevant features from complex data distributions which are in textual format which may result in the contribution to stronger cybersecurity defenses against sophisticated malware threats and their impact on digital infrastructures.

Problem Statement

- Modern malware exhibits intricate behaviors: polymorphic, metamorphic, fileless, stealthy techniques.
- Traditional signature-based methods struggle to accurately identify and classify such malware, yielding many false positive.
- Sophisticated variants often evade detection, leaving infrastructures vulnerable.
- Through machine learning and AI, robust solutions could be developed to accurately detect, analyze and mitigate sophisticated malware threats.

Objective of Project

The main objectives are:

- Utliize SecureBERT to enhance the classification of complex android malware threats.
- Generate context and analyze for various malware types , aiding in comprehensive threat analysis and better understanding of malware behavior through texts.

Scope of Project

Capability

- Utilize SecureBERT for enhanced accuracy, context generation in malware analysis.
- Train and fine tune models using diverse datasets to identify and classify various types of malware.

Limitations .

- Textual dataset dependent.

Originality of Project

The novel task has been performed using SecureBERT which is a Roberta based domain and is trained through articles, websites, research paper related to cybersecurity subject The model processes data in context manner through malwares which contain features in textual format which enables to classify the real time texts, articles which contain malwares which may ultimately contribute to cybsersecurity domain.

Potential Applications

- Cyber Security: Enhance malware detection and analysis capabilities.
- Threat Intelligence: provide valuable insights to security professionals and help them stay ahead of emerging threats by continuous analyzation and classification of new malware variants
- Forensic Analysis: gather evidence and reconstruct attack scenarios.
- Incident Response: help security teams understand the nature of the malware along with potential impact, and the necessary mitigation steps required.

Literature Review[1]

Paper	**Authors**	**Methodology**	**Results**	**Conclusion**	**Flaws**	**Strengths**
MeMalDet: A memory analysis-based malware detection framework using deep autoencoders and stacked ensemble under temporal evaluations (2024)	Pascal Manirinho, Abdun Naser Mahmood, Mohammad Jabed Morshed Chowdhury	The model has utilized deep auto encoders and stacked ensemble approach to analyze memory dumps, learning normal system behavior and focusing on malware attacks.	enhance malware detection accuracy by leveraging memory analysis techniques, deep autoencoders, achieved accuracy of 99.78%	Highlight the effectiveness of their approach in recognizing common patterns indicative of malware across different variations and instances	The sources lack detailed discussion on the limitations of memory analysis techniques	Innovative use of memory analysis techniques, deep autoencoders, and stacked ensemble methods

Literature Review[2]

Paper	**Authors**	**Methodology**	**Results**	**Conclusion**	**Flaws**	**Strengths**
A Transformer-Based Framework for Payload Malware Detection and Classification (2024)	Kyle Stein, Arash Mahyari,Guillermo Francia III, Eman El-Sheikh	The model utilizes transformers to learn complex patterns from raw payload bytes of network packets. It utilized self attention mechanism to analyze sequential data.	Transformer based model using raw payload bytes can effectively detect and classify malware in network traffic,achieved accuracy and f1 score of 79.57%	The proposed transformer-based model detect and classified malware using raw payload bytes	Over reliance on survey results and subjective feelings, rather than objective data to address the rising crime rate.	transfomer-based framework can significantly improve malware detection and classification

Literature Review[3]

Paper	Authors	Methodology	Results	Conclusion	Flaws	Strengths
MADRAS-NET: A deep learning approach for detecting and classifying android malware using Linknet (2024)	Yi Wang, Shanshan Jia	Utilized AndMal-2020 dataset for training and evaluation. It incorporated static and dynamic malware features like permissions(static) and api calls (dynamic)	The proposed framework can achieve superior performance in detecting and classifying Android malware, outperforming existing approaches,achieved 99.59% accuracy,0.997 AUC	The framework enhances the accuracy and robustness of Android malware detection and classification.	need for further validation of the model's performance across diverse malware types.	Ability to organize an argument as a coherent line of reasoning composed of multiple supporting claims.

Literature Review[4]

Paper	Authors	Methodology	Results	Conclusion	Flaws	Strengths
Automated malware detection using machine learning and deep learning approaches for android applications (2023)	S.Poornima, R.Mahalakshmi	Feature extraction was performed using CICAndMal2017 dataset, categorizing the data into signature based and behavior based, DBN has been used for classification.	Emphasizes the importance of analyzing different machine learning models for creating an effective real-world malware detection system, achieved 99.83% accuracy	The approach significantly enhances device security and privacy by generating high accuracy in the sytem through DBN network.	High accuracy might lead to overfiting of the model.	a compelling approach for automated malware detection in Android applications, backed by strong evidence and impressive accuracy.

Literature Review[5]

Paper	**Authors**	**Methodology**	**Results**	**Conclusion**	**Flaws**	**Strengths**
MalBERTv2: Code Aware BERT-Based Model for Malware Identification (2023)	Abir Rahali,Moulay A. Akhloufi	BERT based architecture has been utilized to incorporate pretokenization and feature extraction to improve malware accuracy.	MalBERTv2 combines the feature analyzer and MalBERT components to achieve state of the art performance,achived accuracy of 0.9957(MixG-Androzoo),F1-Score(0.9762)	Integration of code-aware features with BERT architecture improves the model's performance.	Lack of comparison with other state of art models beyond accuracy metrices.	combination of code-aware features and BERT architecture, supported by high frequency, F1 score, and precision

Methodology [1]

BERT (Bidirectional Encoder Representation from Transformers)

- A deep learning model for natural language understanding developed by Google AI.
- Bidirectional Context: Considers both left and right context simultaneously .
- Transformer based encoder model.
- Masked Language Model (MLM): Predicts masked words based on context.
- Next Sentence Prediction (NSP): Determines if one sentence follows another.
- The model uses smaller training batch and fewer training steps for optimization.

Methodology [2]

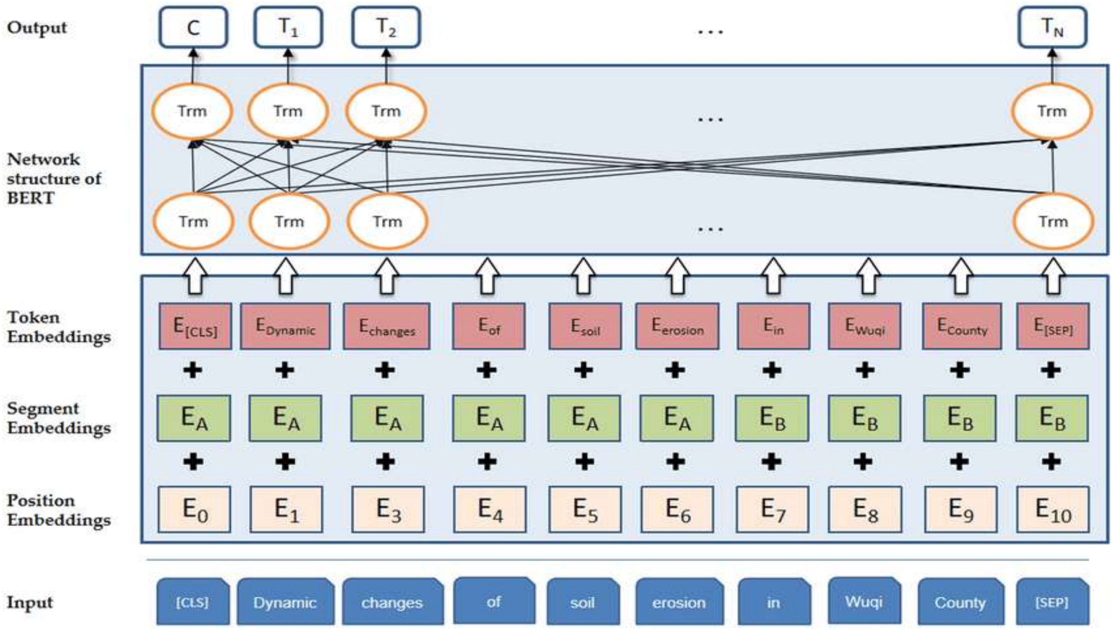


Fig.1 BERT Architecture,Source: Adapted from [6]

Methodology [3]

ROBERTa (A Robustly Optimized BERT Pretraining Approach)

- An Enhanced Version of BERT developed by Facebook AI.
- Optimized for better performance on AI tasks .
- Trained on larger datasets as compared to BERT model.
- Uses same architecture as BERT, but with adjusted hyperparameters like longer learning rates, large batch size.
- The model uses mini-batches and more training steps as compared to BERT models.

Methodology [4]

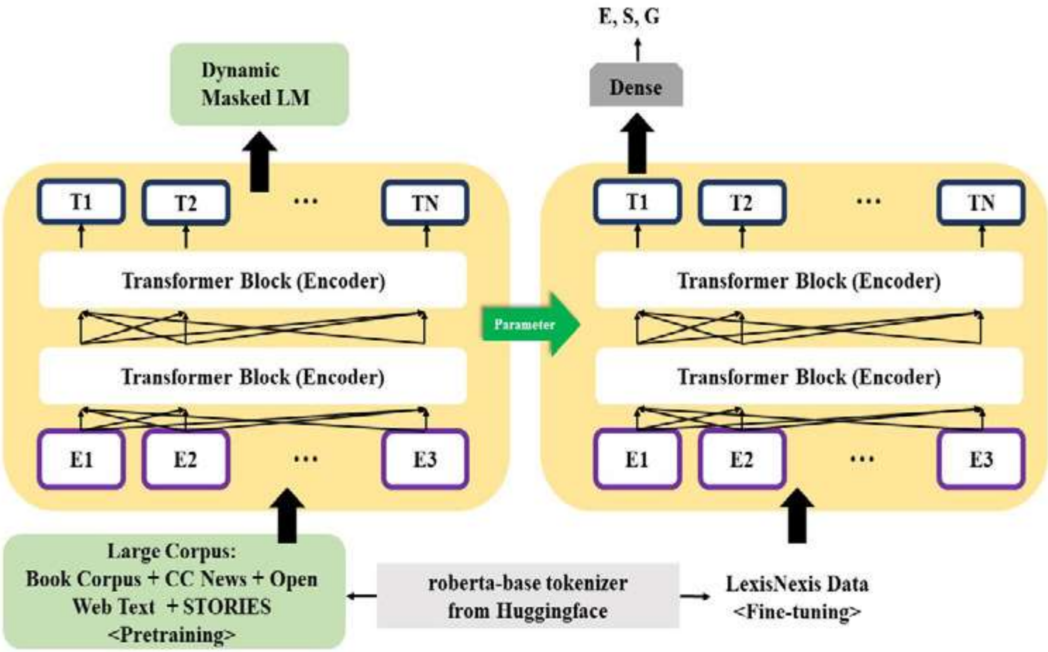


Fig.2 RoBERTa Architecture,Source: Adapted From [8]

Methodology [5]

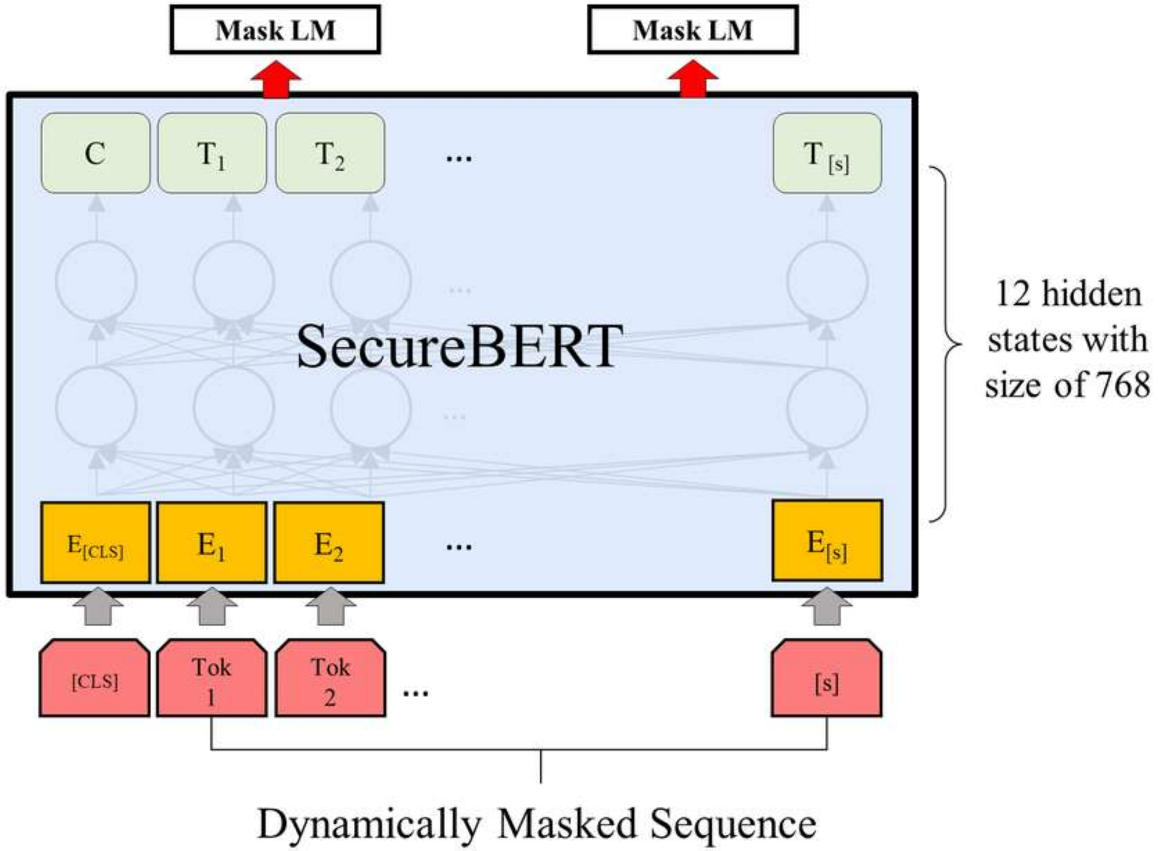


Fig.3 SecureBERT Architecture,Source: Adapted from [9]

Methodology [6]

SecureBERT

- SecureBERT is a domain specific language model for cybersecurity.
- Built on the ROBERTa architecture.
- Trained on extensive cybersecurity-related texts.
- The model is evaluated using the Standard Masked Language Model (MLM) tests
- Outrperforms models like RoBERTa and SciBERT in predicting masked words through MLM.
- Effective in interpreting cybersecurity related texts.

Methodology [7]

SecureBERT

- Input Layer: Processes tokenized cybersecurity-related input data.
- Transformer Layer: 12 hidden layers utilizing self-attention mechanisms.
- Captures contextual relationships within text.
- Approximately 123 million parameters.
- Capable of processing complex cybersecurity information.
- Noise Injection introduced during training to enhance robustness which improves adaptability to varied cybersecurity contexts.

Methodology [8]

SecureBERT

- Tailored for cybersecurity terminology.
- Expanded vocabulary with approximately 50,265 tokens.

Methodology [9]

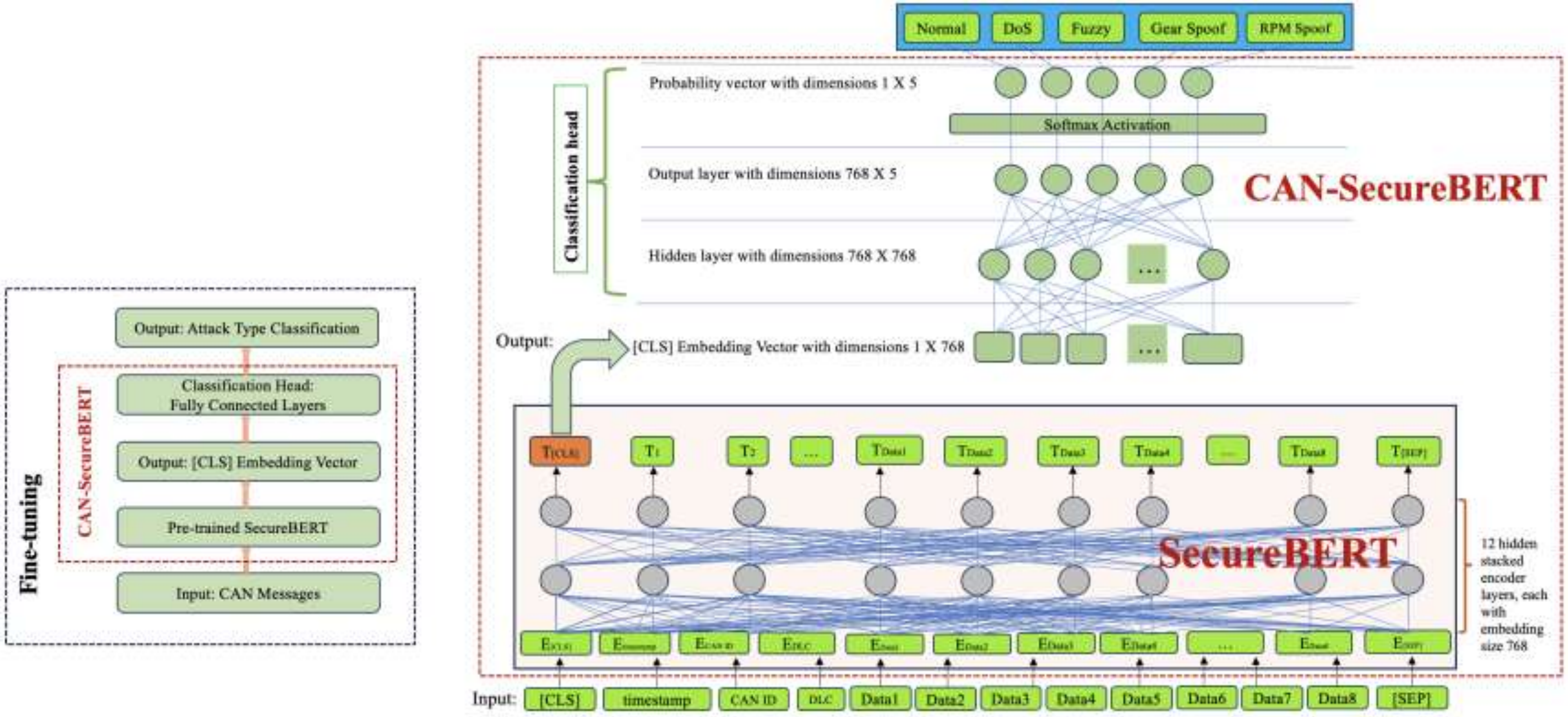


Fig.4 SecureBERT Detailed Architecture and Flow, Source: Adapted From [9]

Methodology [10]

Table 2.1: The statistics of collected cybersecurity corpora for training the SecureBERT.

Type	No. Documents
Articles	8,955
Books	180
Survey Papers	515
Blogs/News	85,953
Wikipedia (cybersecurity)	2,156
Security Reports	518
Videos (subtitles)	134
Total	**98,411**

Vocabulary size	1,674,434 words
Corpus size	1,072,798,637 words
Document size	2,174,621 documents (paragraphs)

Table 2.2: The resources collected for cybersecurity textual data.

Websites
Trendmicro, NakedSecurity, NIST, GovernmentCIO Media, CShub, Threatpost, Techopedia, Portswigger, Security Magazine, Sophos, Reddit, FireEye, SANS, Drizgroup, NETSCOUT, Imperva, DANIEL MIESSLER, Symantec, Kaspersky, PacketStorm, Microsoft, RedHat, Tripwire, Krebs on Security, SecurityFocus, CSO Online, InfoSec Institute, Enisa, MITRE
Security Reports and Whitepapers
APT Notes, VNote, CERT, Cisco Security Reports , Symantec Security Reports
Books, Articles, and Surveys *Tags: cybersecurity, vulnerability, cyber attack, hack*
ACM CCS: 2014-2020 , IEEE NDSS (2016-2020), IEEE Oakland (1980-2020) ACM Security and Privacy (1980-2020), Arxiv , Cybersecurity and Hacking books
Videos (YouTube)
Cybersecurity courses, tutorial, and conference presentations

Fig.5: Cybersecurity corpora for training SecureBERT,Source: Adapted from [10]

Methodology [11]

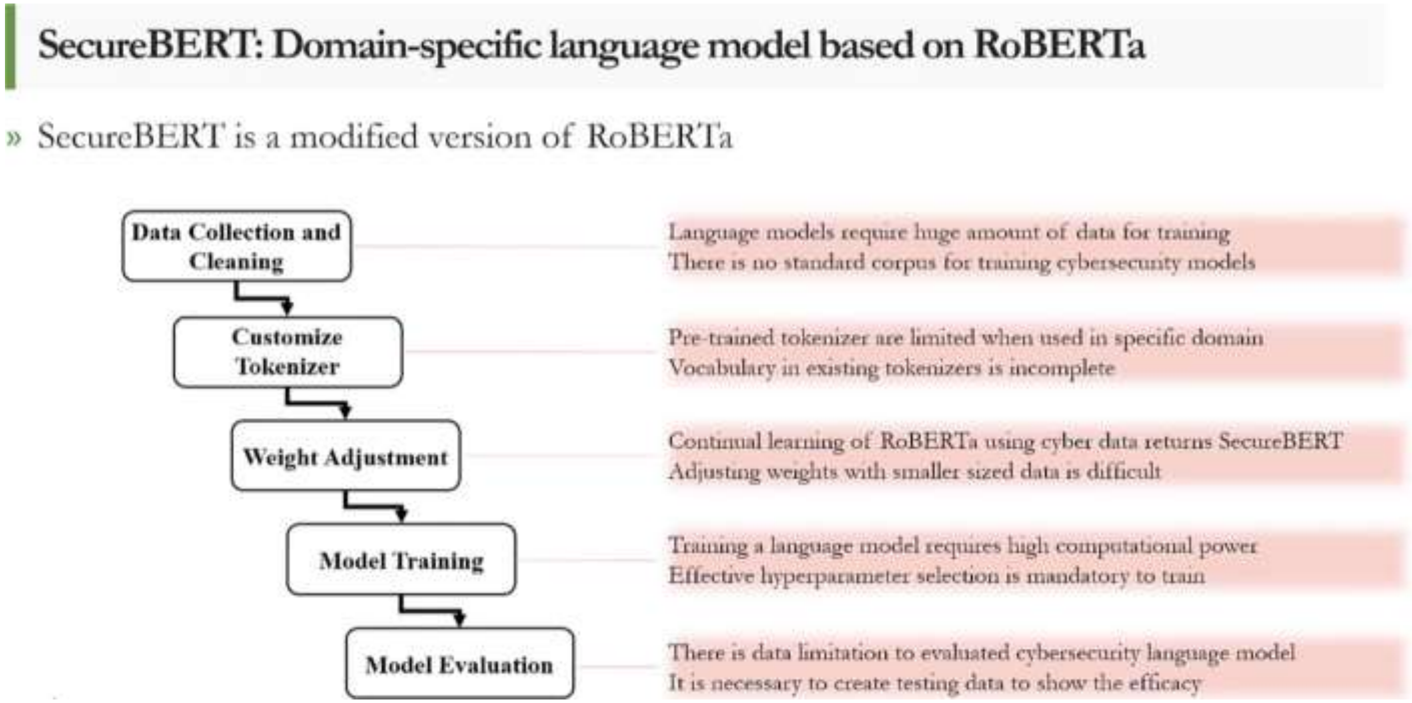


Fig.6: Explanation of SecureBERT,Source: Adapted from [10]

Methodology [12]

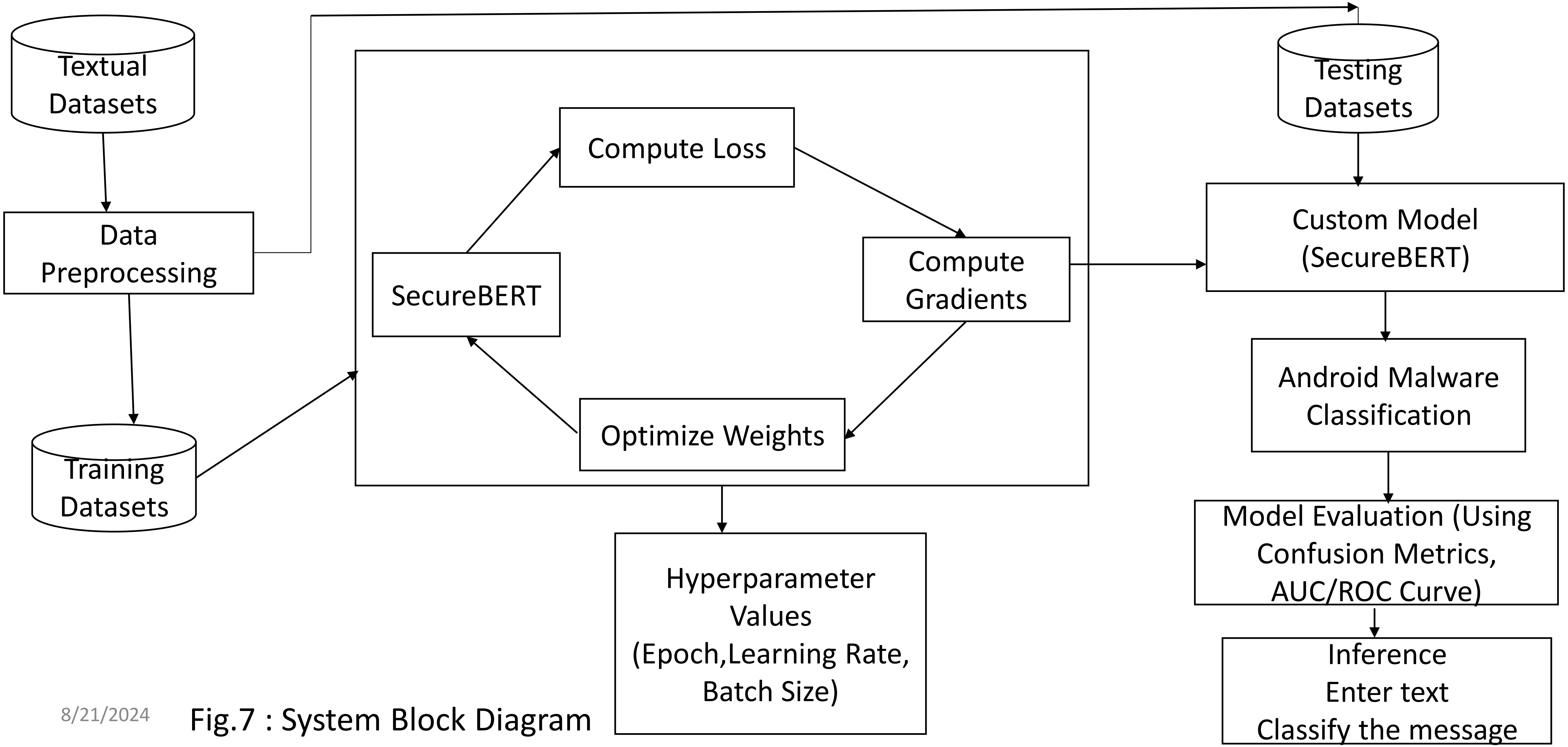


Fig.7 : System Block Diagram

Methodology [13]

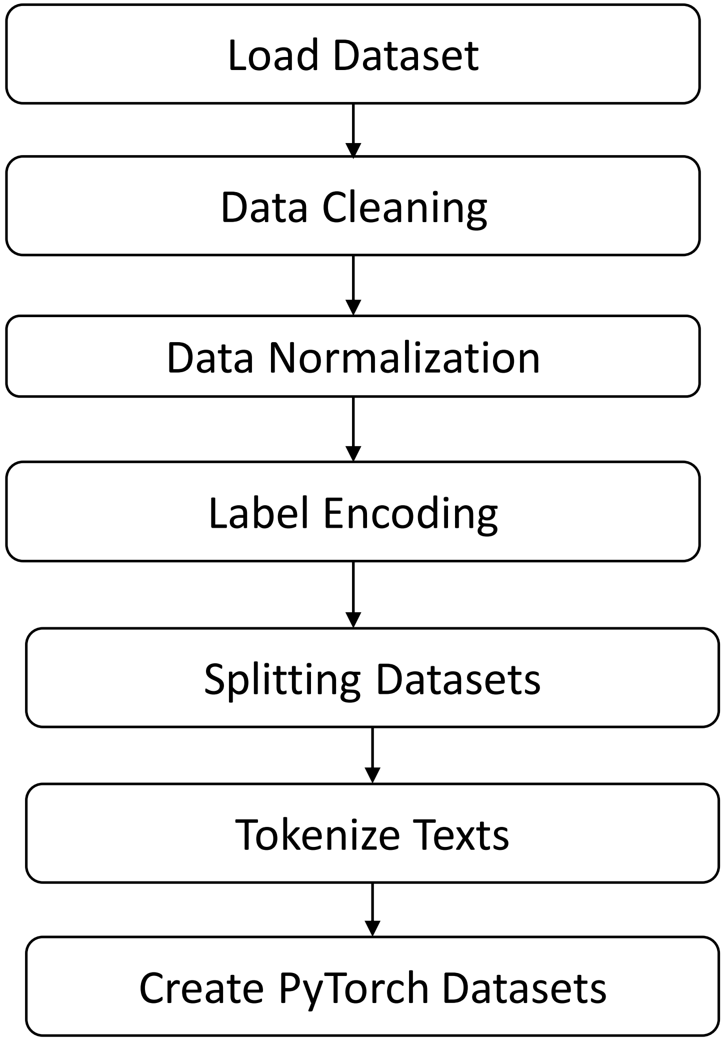


Fig.8: Preprocessing Steps

Methodology [14]

Preprocessing Steps:

- **Load Dataset:** Import data from files or databases.
- **Data Cleaning:** Remove errors, handle missing values, and eliminate duplicates.
- **Data Normalization:** Standardize texts or scale numerical values.
- **Label Encoding:** Convert categorical labels into numerical formats.
- **Splitting Datasets:** Divide data into training, validation and test sets.
- **Tokenize Texts:** Breakdown texts into words or subwords for model processing.

Methodology [15]

Preprocessing Steps:

- **Create Pytorch Datasets:** Convert processed data into PyTorch-compatible datasets for training and evaluation.

Methodology [16]

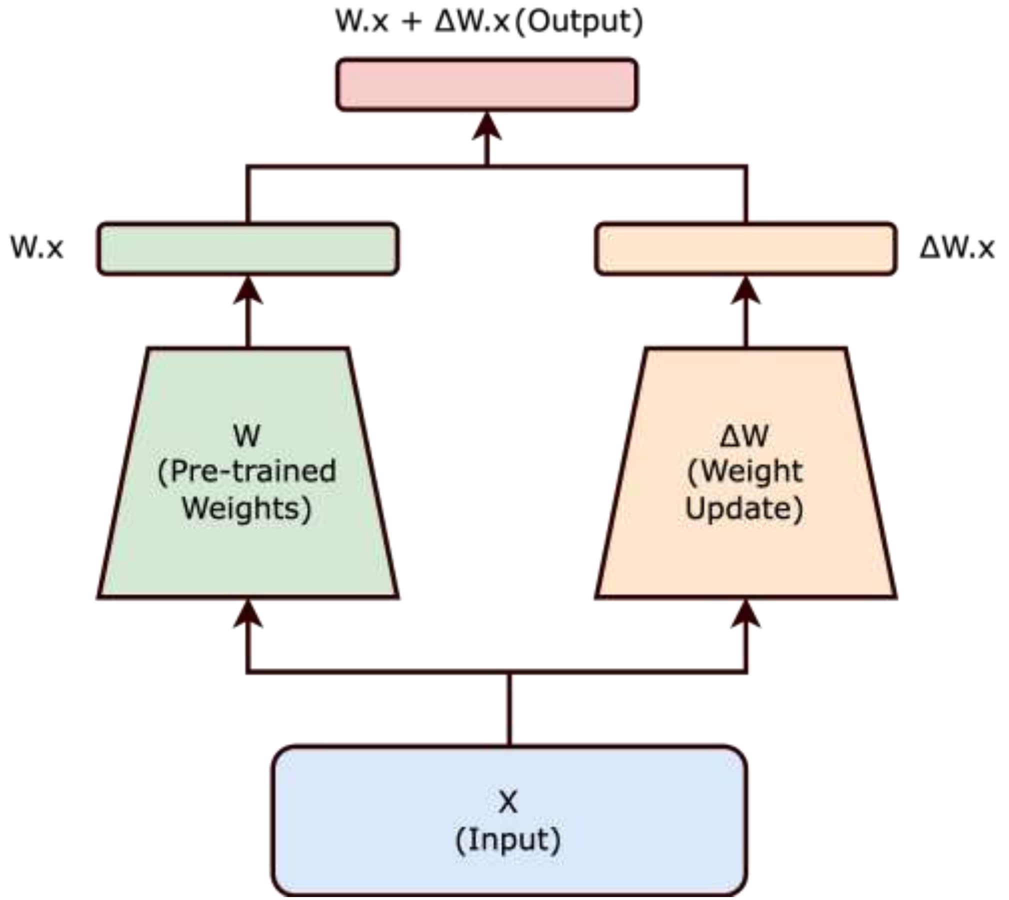


Fig.9: Finetuning using LoRA(Low Rank Adaptation)

Methodology [17]

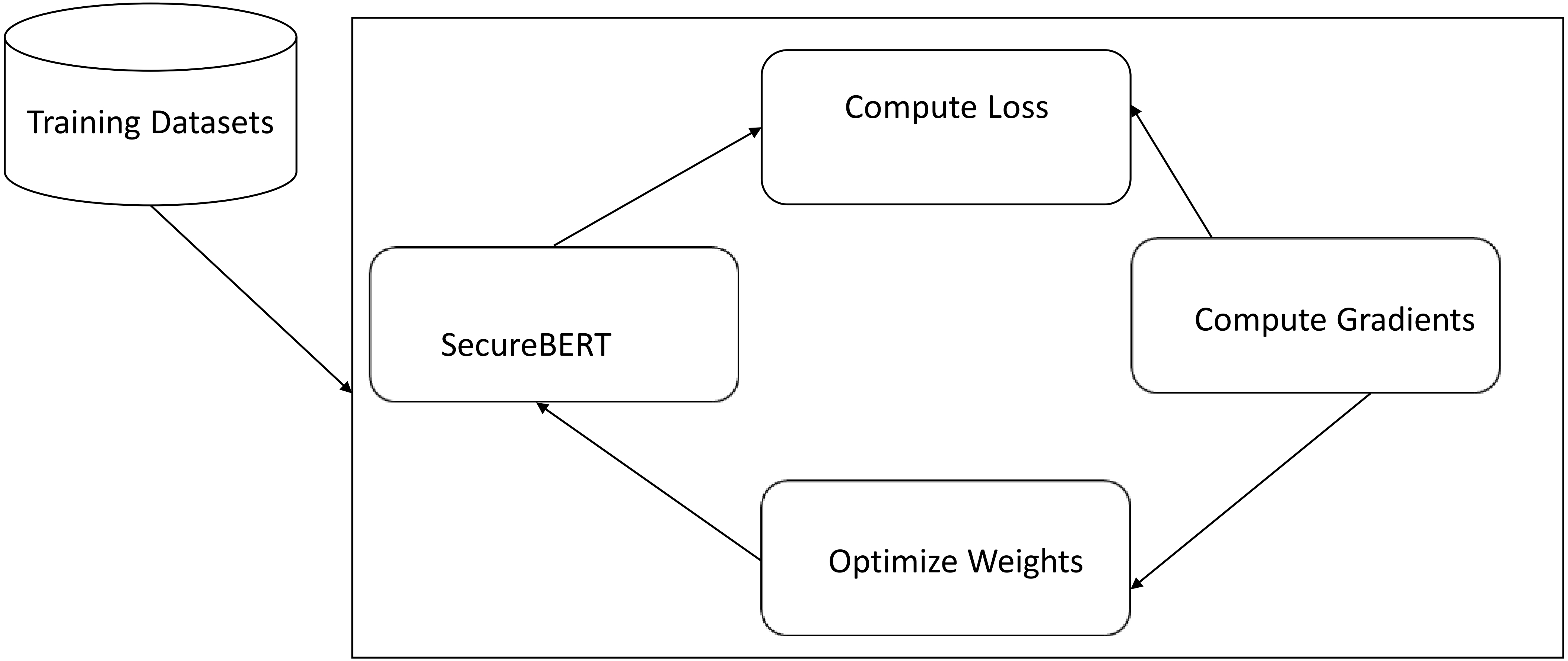


Fig.10: Model Finetuning Steps

Methodology [18]

Model Training:

- The training process begins with preprocessed training datasets and their corresponding labels into the SecureBERT model.
- The models compute the loss by comparing predicted values to actual labels. Gradients are calculated to determine necessary adjustments, and the model weights are optimized (fine-tuned) to minimize the loss, adapting the pre-trained models to the new dataset.
- The process is iteratively repeated for multiple epochs, continuously refining the models' performance and enhancing their accuracy in malware classification tasks.

Methodology [19]

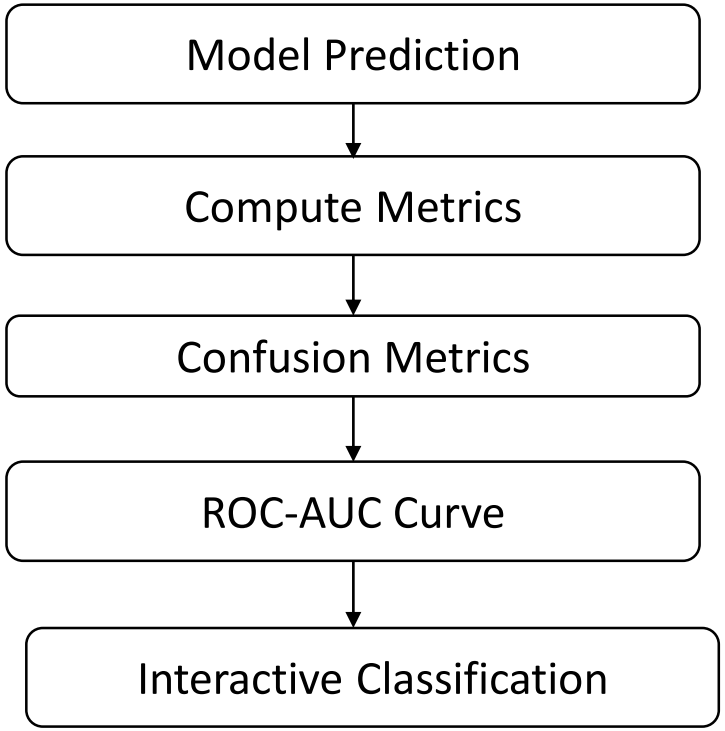


Fig.11: Postprocessing Steps

Methodology [20]

Postprocessing Steps:

- **Model Evaluation:** Use the trained model to predict labels for the test data.
- **Compute Metrics**: Calculate performance metrics:accuracy, precision, recall, F1score,etc.
- **Confusion Metri**cs: Generate a confusion matrix to visualize true/false positives and negatives.
- **ROC-AUC Curve**: Plot ROC and evaluate AUC to evaluate the model's distinguishing ability.
- **Interactive Classification (Inference)**: A message is entered by user through which prediction of label is classified.

Methodology [21]

Datasets: Malware DB Dataset: It is a comprehensive dataset specifically designed to provide annotate malware articles:

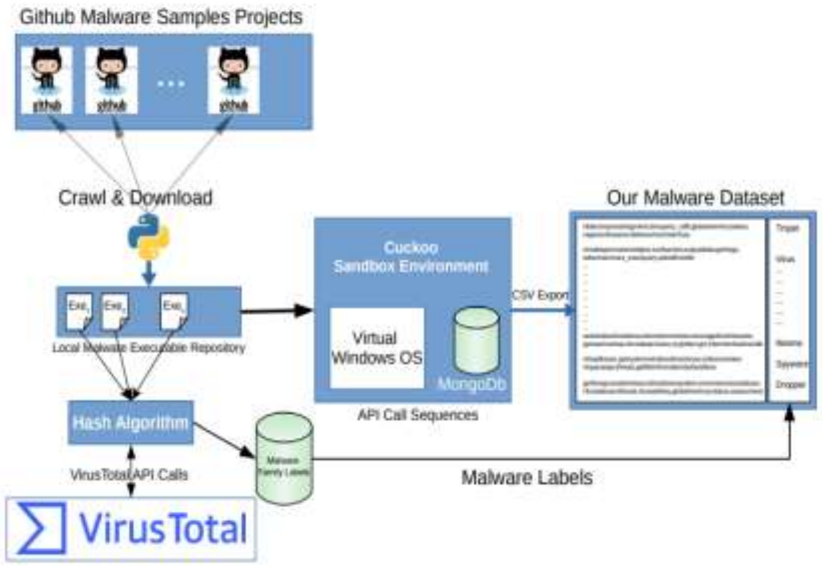
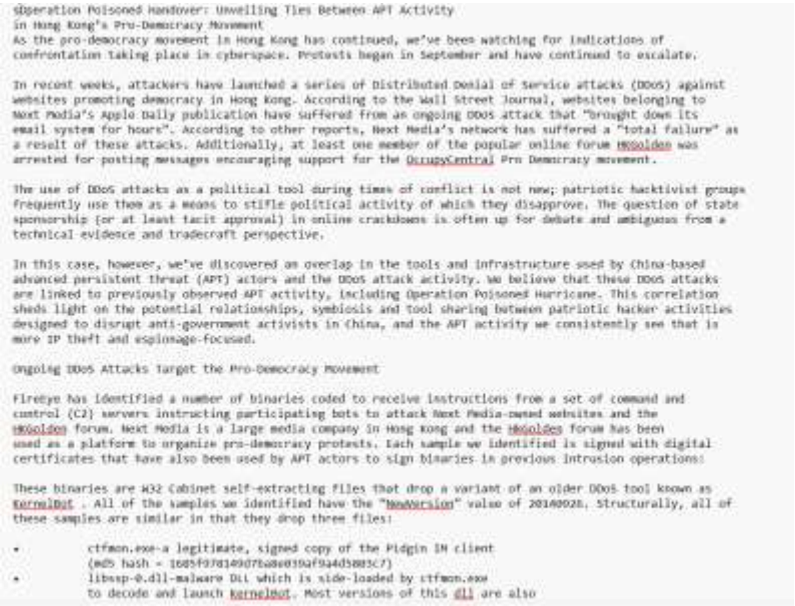


Fig.12: Malware DB Dataset Preparation



Operation Poisoned Handover: Unveiling Ties Between APT Activity in Hong Kong's Pro-Democracy Movement

As the pro-democracy movement in Hong Kong has continued, we've been watching for indications of confrontation taking place in cyberspace. Protests began in September and have continued to escalate.

In recent weeks, attackers have launched a series of Distributed Denial of Service attacks (DDoS) against websites promoting democracy in Hong Kong. According to the Wall Street Journal, websites belonging to Next Media's Apple Daily publication have suffered from an ongoing DDoS attack that "brought down its email system for hours". According to other reports, Next Media's network has suffered a "total failure" as a result of these attacks. Additionally, at least one member of the popular online forum HKGolden was arrested for posting messages encouraging support for the OccupyCentral Pro Democracy movement.

The use of DDoS attacks as a political tool during times of conflict is not new; patriotic hacktivist groups frequently use them as a means to stifle political activity of which they disapprove. The question of state sponsorship (or at least tacit approval) in online crackdowns is often up for debate and ambiguous from a technical evidence and tradecraft perspective.

In this case, however, we've discovered an overlap in the tools and infrastructure used by China-based advanced persistent threat (APT) actors and the DDoS attack activity. We believe that these DDoS attacks are linked to previously observed APT activity, including Operation Poisoned Hurricane. This correlation sheds light on the potential relationships, symbiosis and tool sharing between patriotic hacker activities designed to disrupt anti-government activists in China, and the APT activity we consistently see that is more IP theft and espionage-focused.

Ongoing DDoS Attacks Target the Pro-Democracy Movement

FireEye has identified a number of binaries coded to receive instructions from a set of command and control (C2) servers instructing participating bots to attack Next Media-owned websites and the HKGolden forum. Next Media is a large media company in Hong Kong and the HKGolden forum has been used as a platform to organize pro-democracy protests. Each sample we identified is signed with digital certificates that have also been used by APT actors to sign binaries in previous intrusion operations:

These binaries are W32 Cabinet self-extracting files that drop a variant of an older DDoS tool known as KernelBot . All of the samples we identified have the "NewVersion" value of 20140926. Structurally, all of these samples are similar in that they drop three files:

- ctfmon.exe-a legitimate, signed copy of the Pidgin IM client (md5 hash = [illegible])
- libssp-0.dll-malware DLL which is side-loaded by ctfmon.exe to decode and launch KernelBot. Most versions of this dll are also

Fig.13: Malware DB Dataset Sample

Methodology [22]

Datasets:

Androzoo: Collection of 24,476,148 Android APKs from various sources including Google Play.

markets	text
anzhi	The Android malware identified by the hash '0000003B455A6C7AF837EF90F2EAFFD856E3B5CF49F5E27191430328DE2FA670', with an additional signature '9C14D537A7ADB4CFC43D291352F73E05E0CCDD4A' and '3EDFC78AB53521942798AD55
PlayDrone	The Android malware identified by the hash 0000014A634DB98F85038B833A8DFC50D5FB13A464E0B25994E439AEF830CD70, along with signatures C3EBEC52C9388BF67479FF1385A56C59B3E39E81 and 0A146750FB447CF3859C9CB659AB04F1,
play.googl	000001A94F46A0C3DDA514E1F24E675648835BBA5EF3C3AA72D9C378534FCAD6 C0444D784685EFE5F6D9F28683B24B5873E509CB EC82771AE018B93AD784A1FD2B625216 1/1/1980 0:00 52469861.0 com.firstchoice.myfirstchoice 1206145.0 0
play.googl	000002B63FAD4B030787F6DE4081DC1E12325026EB7DDAD146C52F5F4FC2D525 DD723B32EDD9F70AADBD66846621967157DF9BD4 985E601C17F0A9346590AE92A5AD664E 1/1/1980 0:00 4300370.0 com.deperu.sitiosarequipa 10000.0 0.0 12,
play.googl	000003D3981DC548A772A30D688F424CFB88561A63A2DD888E7CF55171442946 6AE9F138F7E0C63E5D58CC7E82FB05A50F041637 F4789023733E41EE883208ACBC956020 1/1/1980 0:00 12958838.0 com.safetravels.safetravelsmain 400125.0 0
appchina	00000439A3FFA123C3F9BC45E5E821351B1A5C276871B36447AB80C74261F354 375D6FFD167B85E7E2935B1B927F87D8E44A9AB4 9283C74DD8356C188B6D94B88B8FDD9B 10/25/2011 2:30 1044597.0 bmthx.god102409paperi 6.0 1.0 4/27/201
play.googl	0000049D8911607971A3336DE5CF36F4799D679D6BB9EF014CBFE73578A6E3EA 433307FF815B8BA4A88BC3EE3EC2C912FFBA063B B03BEEAFB1975881F11395C4F5F6E2ED 1/1/1980 0:00 3161615.0 com.bmi.calculatorplus 11.0 0.0 7/13/2020 7
PlayDrone	0000090C2169EFF7E9A889CA1953CF39C887E0FD23A9BEF6E44A387339842C40 E86150BBF222D038F607BA91AD5E6E70C4C86C2A 39DC3F83F0F1E15A791A3964738388E1 8/21/2014 23:38 2541606.0 eremeev.dev.montecarloradio 8.0 0.0 5/19
play.googl	00000989F3E215BA9FC3BDD5B56AF751343B540C1026BF42AEA8FEBF68874ECC 81EA59FE2C95EDEF3662CAF165FF36AB410FA158 71FFF1BA55D7F6BF0AB00C15F6B5BC99 9/2/2015 10:34 1375862.0 kr.ac.snjc.library 3.0 0.0 3/17/2016 4:48 227
play.googl	000009BE655C2111A0ADA2B46EBD2AD59946D46A4780C3D4D34E58E57E0AFED9 47F9BA16C8E8EC99826FF7C051C9E5A55B50F954 E08EB262E4B439D027722014EBA56016 1/1/1981 1:01 11605703.0 com.fun.video.chat.theprincefamilycall 3.(
play.googl	00000A753D553DE825BBD8E84D62B3D5965EA29E775504FCB4F31570E6B0E0B9 088592B41814930929A8EE504FF298248FFFBB9C E9A51B9409807C5DDBB50C60A7F11D66 1/1/1981 1:01 22490734.0 com.rcdnc.cafezinho 68.0 0.0 11/29/2023
play.googl	00000B7064B3750F9B4C6CB9FF2C8852DFF67B29CC8CEF6D72FA4744E573041B FF63649BAAB9913C95F943FCCE16F7B6CD9FDE6A B8E8A8BED6FB447F6B75050F80F66ECC 1/1/1981 1:01 10510322.0 com.resultsdirect.eventsential.branded.cue
play.googl	00000CABAEF95BC6586666EAF020DD1A71E7A12D80033A047B66563DC7B936BA 3719D3B82545DB31B33B77B30F299BEF32DDE579 DB3B03486C135DBF8E30C84C72F966D6 1/1/1981 1:01 142134854.0 com.boombitgames.Tiny2 3000448.0 0.(
play.googl	00000D27239CED784030954898E6BF328528993E209C226E447F91A8E146876B 50EA4541EFE2EC5E6C8638DE20DA7E7D10262D21 25BCACFC9E79EE61054C97D9D38B6142 1/1/1981 1:01 15562852.0 com.springtechsolutions.sunnahradio 22.0 (
play.googl	00000DA875B5A96549C8D23D74FDA8419166EDE7441E2562BAA49928CBA0BE07 A56F0E18CC7AC55DA291A31E6209F17D0D80AFEB 45116F2A069C5B29FE7171C86825C08C 1/1/1981 1:01 143417826.0 app.stretchminder.android 73.0 0.0 6/17,
play.googl	00000F8EE4F64324BA04356745E946152C2F43ECB63372E89DB79830BADAF1BA ACBDBE1199C224A9F27A23EBEE333BA78180B38C 9ECD6BAF09ECB56DB7C1E3B1CE0C1E79 1/1/1980 0:00 5345848.0 com.rbsoftware.pfm.personalfinancemana
play.googl	00001037C7C73206C99EC558943F81EB40C22614281A3472E3FB69730F0DABC8 8C921A024441BF275313AAF760CDBBE8BE8C396A DDC95C58F124E5E01D2C04971CB628F0 1/1/1981 1:01 14736254.0 com.newaadharguide 3.0 0.0 10/23/2022 1
play.googl	0000103AF1B89F26FF77209541EFF57DA99891371944408A92E875658355B9DB 2DD82AD3DC5CBA1BFF01BD0714C59F75F7F3F65D 08094F033E8F442882592E0F509A024C 1/1/1981 1:01 6650179.0 com.dudesolutions.sdcm01 38.0 0.0 4/30/202
play.googl	00001091AE8F704E461A37576EE67CCCD1F286D07C6C176449FEC86B6984E05F F0F3A2F67DA2ABC34E02254F1A1CFA7512E036F3 C5E3E0C04F46B1009E8E665C6359D136 1/1/1980 0:00 12798854.0 com.indoorway.android.fluctus 14.0 0.0 9/1
play.googl	000010F33B578B4B5C33724520E38BD44485705DC1B6ADDE4975EBBA7F114355 DDE9CA9CC8C9FD67B27E68A637D1015E3852AD4E 8CEE4A742DD05B509007C540CBB86BEE 1/1/1980 0:00 4727630.0 com.fearless.teengirl 3.0 0.0 11/5/2018 1
play.googl	00001112E046D8BBF8B5529A9ECB39920F828209EF4ABFEB95AAB46D41F56A7D 4852E9A9B19DB6CC776AB811EAE6ABD5F845A64B 2A6ECF7A0D56056D653F5B3EDDAFC8A5 8/17/2016 15:42 16004883.0 com.appautomatic.ankulua.trial 39.0 0
play.googl	0000120C5998A69C7907706CA1BDBBF98CCD884891D68CDE4D87BB530AEAF015 9DB718A09CE7DEEFFAA2192A6A19A428D917CFDB 1190C2067E50EF869D9179AC18834E8B 2/8/2021 13:59 10936955.0 com.placz.cricketpakistan 2.0 0.0 3/16/
play.googl	00001438D92DC8AAA7B1BCEDDC4BEBC12E3104C02142FEAE1F929E3A0C2BC719 6AA10B10EE30A7525528F5A0A639836E181648E6 90E704E0F6189111796CD7B2F358719F 1/1/1981 1:01 138916497.0 com.northcube.sleepcycle 6446.0 0.0 5/5,
play.googl	0000143EF8D00E3A65C5C8C380221D00678FED906FDC2EBC483D1987457C7B2B DAF886288EB27F9C0B866EB19A357E1E866AF4DC 2C640DD9FE09FAC938F4282C24942516 1/19/2016 13:28 1882706.0 com.kbf.app27730661 70101.0 0.0 3/23/
PlayDrone	000014F7037586315DE348D21337B90B83A1C887E247DA8E4CC043702E36DFBA b597a72faf8641fef10a448d42dbda5452c01d10 21ae85a4879db03c122b122a0dde62c4 3/5/2012 17:08 132603.0 cinema.release.dates 4.0 1.0 4/28/2014 0:45 11
play.googl	000016735720DFABEE77B5A884629F8CB1578CCBAE079CF1839B8E8AE84C6B07 46BFEBF9DBB59D5B086EE4049DD86C8E2C48630E 817E4DC9202A9C05E52E0196B3B88D70 1/1/1981 1:01 43961338.0 freshpicks2.android.app 12.0 0.0 4/8/2023

Fig.14: Androzoo features

Methodology [23]

Datasets:

Drebin: Provides tagged Android malware samples for easier navigation and research.

text	class
	S
In our Android application, we utilize the `SmsManager` class to send SMS messages with	S
In our Android application, we send SMS messages using the `SmsManager` class with the	S
Ljava.lang.Class.cast Ljava.net.URLDecoder READ_PHONE_STATE Landroid.content.Cont	S
Ljava.net.URLDecoder READ_PHONE_STATE ClassLoader Ljava.lang.Class.getDeclaredFie	S
READ_PHONE_STATE READ_SMS Ljavax.crypto.spec.SecretKeySpec android.intent.actior	S
onServiceConnected bindService ServiceConnection android.os.Binder READ_PHONE_ST	S
Ljava.lang.Class.cast Ljava.net.URLDecoder READ_PHONE_STATE Landroid.content.Cont	S
SEND_SMS android.telephony.SmsManager READ_PHONE_STATE RECEIVE_SMS READ_S	S
SEND_SMS READ_PHONE_STATE READ_SMS android.intent.action.BOOT_COMPLETED a	S
transact attachInterface android.os.Binder SEND_SMS Ljava.lang.Class.getCanonicalNan	S
SEND_SMS android.telephony.SmsManager READ_PHONE_STATE ClassLoader Ljava.lang	S
READ_PHONE_STATE android.intent.action.BOOT_COMPLETED KeySpec HttpGet.init Sec	S
transact onServiceConnected bindService attachInterface ServiceConnection android.os.	S
READ_PHONE_STATE ClassLoader Ljava.lang.Class.getDeclaredField TelephonyManager.	S
SEND_SMS READ_PHONE_STATE TelephonyManager.getLine1Number android.telephony	S
SEND_SMS android.telephony.SmsManager READ_PHONE_STATE Landroid.content.Cont	S
SEND_SMS Ljava.lang.Class.getCanonicalName android.telephony.SmsManager READ_PI	S
SEND_SMS READ_PHONE_STATE Ljava.lang.Class.getField RECEIVE_SMS READ_SMS andr	S
READ_PHONE_STATE RECEIVE_SMS READ_SMS Ljavax.crypto.spec.SecretKeySpec androi	S
transact onServiceConnected bindService attachInterface ServiceConnection android.os.	S
SEND_SMS android.telephony.SmsManager READ_SMS INTERNET TelephonyManager.ge	S
SEND_SMS android.telephony.SmsManager READ_PHONE_STATE RECEIVE_SMS Telepho	S
SEND_SMS android.content.pm.Signature READ_PHONE_STATE ClassLoader RECEIVE_SN	S
transact onServiceConnected bindService attachInterface ServiceConnection android.os.	S
SEND_SMS android.telephony.SmsManager READ_PHONE_STATE Landroid.content.Cont	S

Fig.15: Drebin features

Methodology [24]

Datasets:

CICMalDroid 2017: Comprehensive dataset with over 17,341 samples, categorized into Adware, Scareware, SMS, Riskware, and Benign.

Label	Message
ADWARE_SELFMITE	Flow ID: 172.217.2.106-10.42.0.151-443-36635-6, Source: 10.42.0.151:36635, Destination: 172.217.2.106:443, Protocol: 6.0, Timestamp: 14/06/2017 01:54:51, Duration
RANSOMWARE_SIMPLOCKER	Flow ID: 172.217.1.162-10.42.0.211-443-40670-6, Source: 10.42.0.211:40670, Destination: 172.217.1.162:443, Protocol: 6.0, Timestamp: 16/06/2017 03:55:43, Duration
ADWARE_SELFMITE	Flow ID: 172.217.1.174-10.42.0.151-443-57273-6, Source: 10.42.0.151:57273, Destination: 172.217.1.174:443, Protocol: 6.0, Timestamp: 24/08/2017 01:10:10, Duration
SMSMALWARE_ZSONE	Flow ID: 216.58.219.234-10.42.0.151-443-38357-6, Source: 10.42.0.151:38357, Destination: 216.58.219.234:443, Protocol: 6.0, Timestamp: 24/08/2017 01:10:26, Durati
SMSMALWARE_ZSONE	Flow ID: 172.217.10.138-10.42.0.42-443-58647-6, Source: 10.42.0.42:58647, Destination: 172.217.10.138:443, Protocol: 6.0, Timestamp: 16/08/2017 04:29:13, Duration
SCAREWARE_VIRUSSHIELD.	Flow ID: 180.149.134.142-10.42.0.211-80-59193-6, Source: 10.42.0.211:59193, Destination: 180.149.134.142:80, Protocol: 6.0, Timestamp: 28/08/2017 05:17:14, Durati
RANSOMWARE_SIMPLOCKER	Flow ID: 10.42.0.211-103.7.30.118-35524-80-6, Source: 10.42.0.211:35524, Destination: 103.7.30.118:80, Protocol: 6.0, Timestamp: 27/06/2017 03:44:37, Duration: 304
BENIGN	Flow ID: 192.168.1.100-10.42.0.42-8004-59388-6, Source: 10.42.0.42:59388, Destination: 192.168.1.100:8004, Protocol: 6.0, Timestamp: 16/08/2017 04:07:02, Duration
RANSOMWARE_SIMPLOCKER	Flow ID: 172.217.2.106-10.42.0.151-443-48575-6, Source: 10.42.0.151:48575, Destination: 172.217.2.106:443, Protocol: 6.0, Timestamp: 14/06/2017 01:54:51, Duration
BENIGN	Flow ID: 180.149.136.194-10.42.0.151-80-36214-6, Source: 10.42.0.151:36214, Destination: 180.149.136.194:80, Protocol: 6.0, Timestamp: 24/08/2017 01:46:30, Durati

Fig.16: CicMaldroid 2017 features

Methodology [25]

Datasets:

Ransomware: The dataset consists of network monitoring records of android devices which determine the types of ransomware along with benign which have been transacted in the user network.

label	text
1	The file with the name '0' and hash '0124e21d-018c-4ce0-92a3-b9e205a76bc0.dll' has properties including a file size of 79755c51e413ed3c6be4635fd729a6e1 bytes, multiple zero values indicating the absence of specific permissions or intents, a
1	The file with the name '1' and hash '05c8318f98a5d301d80000009c316005.vertdll.dll' has properties including a file size of 95e19f3657d34a432eada93221b0ea16 bytes, multiple zero values indicating the absence of specific permissions or intent
1	The file with the name '2' and hash '06054fba-5619-4a86-a861-ffb0464bef5d.dll' has properties including a file size of 85c32641d77a54e19ba8ea4ab305c791 bytes, multiple zero values indicating the absence of specific permissions or intents, a
1	The file with the name '3' and hash '075822ac99a5d301660400009c316005.adhapi.dll' has properties including a file size of 62e3b959d982ef534b66f819fe15f085 bytes, multiple zero values indicating the absence of specific permissions or intent
1	The file with the name '4' and hash '090607dd9ba5d301ca0900009c316005.SensorsNativeApi.V2.dll' has properties including a file size of ae38c5f7d313ad0ff3bfb8826476767f bytes, multiple zero values indicating the absence of specific permiss
1	The file with the name '5' and hash '0aedb43f9ba5d3014e0600009c316005.wlanapi.dll' has properties including a file size of 28c98e000d02f8bfd9d050eb4e4c5d2e bytes, multiple zero values indicating the absence of specific permissions or inter
1	The file with the name '6' and hash '0bc194f9-b102-4833-85bd-603e216a9274.dll' has properties including a file size of 706463ecc8e48e9e3a2a12a0cfcb8858 bytes, multiple zero values indicating the absence of specific permissions or intents, a
1	The file with the name '7' and hash '0c7f9fdc9ba5d301c60900009c316005.wscsvc.dll' has properties including a file size of 39da352fad220e83ce64de8dccb9736b bytes, multiple zero values indicating the absence of specific permissions or intent
1	The file with the name '8' and hash '1.0.154_chromesetup_154_59.exe' has properties including a file size of e11e70ba243800626d17e3ffa6c9fb71 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory us
1	The file with the name '9' and hash '1035f45d9ca5d3015b0a00009c316005.srclient.dll' has properties including a file size of a5a106d5d03e6e59b0282e72bc9420f1 bytes, multiple zero values indicating the absence of specific permissions or inten
1	The file with the name '10' and hash '109b6ed1-e133-407f-b839-3fdd1ebc0d85.dll' has properties including a file size of 212fdb291e0b5d8f34771bc1338c20b3 bytes, multiple zero values indicating the absence of specific permissions or intents, a
1	The file with the name '11' and hash '112ff2d99ca5d301590b00009c316005.mscoreei.dll' has properties including a file size of e313747a38c6bc267c9167d92bd2cb7f bytes, multiple zero values indicating the absence of specific permissions or int
1	The file with the name '12' and hash '11b25aa499a5d301370400009c316005.pngfilt.dll' has properties including a file size of a2f43fac128db67452726d096fb77768 bytes, multiple zero values indicating the absence of specific permissions or inten
1	The file with the name '13' and hash '128906669ca5d301730a00009c316005.TimeBrokerClient.dll' has properties including a file size of d2e6d99c9624a3f6ac7c8679f0e23e00 bytes, multiple zero values indicating the absence of specific permissi
1	The file with the name '14' and hash 'MSBuildTaskHost.resources (10).dll' has properties including a file size of fd3313b819e4d121cb1551362420609c bytes, multiple zero values indicating the absence of specific permissions or intents, a memory
1	The file with the name '15' and hash 'MSBuildTaskHost.resources (11).dll' has properties including a file size of 462da78784b24814d0538fa6dc2ced28 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory
1	The file with the name '16' and hash 'MSBuildTaskHost.resources (12).dll' has properties including a file size of d002e0973af380781e85fdc8b0ddf683 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory
1	The file with the name '17' and hash 'MSBuildTaskHost.resources (13).dll' has properties including a file size of 333669b5ac0e3418e9d72393a2be2d8c bytes, multiple zero values indicating the absence of specific permissions or intents, a memory
1	The file with the name '18' and hash 'MSBuildTaskHost.resources (14).dll' has properties including a file size of 5570b99ac52c4fd941331508b0ec2786 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory
1	The file with the name '19' and hash 'MSBuildTaskHost.resources (15).dll' has properties including a file size of ec9de465ca85f523ea84c07368741fcc bytes, multiple zero values indicating the absence of specific permissions or intents, a memory
1	The file with the name '20' and hash 'MSBuildTaskHost.resources (16).dll' has properties including a file size of de12836d9fb3050b1c4c5c47770624ee bytes, multiple zero values indicating the absence of specific permissions or intents, a memory
1	The file with the name '21' and hash 'MSBuildTaskHost.resources (17).dll' has properties including a file size of 7bc94f59e94e9a6a38d83ead7b1eac34 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory
1	The file with the name '22' and hash 'MSBuildTaskHost.resources (18).dll' has properties including a file size of 328f0ce88bd8a40fe85f0ba5d8e5f703 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory
1	The file with the name '23' and hash 'MSBuildTaskHost.resources (19).dll' has properties including a file size of 5025f28b6052b19ea0a1c4332407d06b bytes, multiple zero values indicating the absence of specific permissions or intents, a memory
1	The file with the name '24' and hash 'MSBuildTaskHost.resources (2).dll' has properties including a file size of 493f30fa92f8a9328eb0fe7602d14967 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory us
1	The file with the name '25' and hash 'MSBuildTaskHost.resources (20).dll' has properties including a file size of e6868c88fa398c92624e071018fb72d1 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory

Fig.17: Ransomware features

Methodology [26]

Datasets:

TUANDROMD: It is the dataset used for classification tasks in the field of cybersecurity, specifically for distinguishing between malicious software (malware) and legitimate software (goodware).

text	Label
In our Android application, we utilize the ACCESS_NETWORK_STATE permission to check network connectivity, the CA	malware
ACCESS_NETWORK_STATE BATTERY_STATS INTERNET READ_PHONE_STATE RECEIVE_BOOT_COMPLETED RECEIVE_S	malware
ACCESS_NETWORK_STATE DISABLE_KEYGUARD GET_TASKS INTERNET KILL_BACKGROUND_PROCESSES READ_PHONE	malware
BATTERY_STATS INTERNET READ_PHONE_STATE RECEIVE_BOOT_COMPLETED RECEIVE_SMS SEND_SMS Ljavax/crypt	malware
ACCESS_WIFI_STATE CHANGE_WIFI_STATE GET_TASKS KILL_BACKGROUND_PROCESSES RECEIVE_BOOT_COMPLETED	malware
ACCESS_NETWORK_STATE INTERNET READ_EXTERNAL_STORAGE READ_PHONE_STATE RECEIVE_BOOT_COMPLETED	malware
ACCESS_NETWORK_STATE INTERNET READ_EXTERNAL_STORAGE READ_PHONE_STATE RECEIVE_BOOT_COMPLETED	malware
ACCESS_NETWORK_STATE INTERNET READ_PHONE_STATE RECEIVE_BOOT_COMPLETED WAKE_LOCK Ljava/lang/refl	malware
ACCESS_WIFI_STATE CHANGE_WIFI_STATE GET_TASKS KILL_BACKGROUND_PROCESSES RECEIVE_BOOT_COMPLETED	malware
ACCESS_NETWORK_STATE DISABLE_KEYGUARD GET_TASKS INTERNET KILL_BACKGROUND_PROCESSES READ_PHONE	malware
ACCESS_WIFI_STATE CHANGE_WIFI_STATE GET_TASKS KILL_BACKGROUND_PROCESSES RECEIVE_BOOT_COMPLETED	malware
ACCESS_NETWORK_STATE BIND_DEVICE_ADMIN CAMERA GET_ACCOUNTS GET_TASKS INTERNET READ_CONTACTS I	malware
ACCESS_WIFI_STATE CHANGE_WIFI_STATE GET_TASKS KILL_BACKGROUND_PROCESSES RECEIVE_BOOT_COMPLETED	malware
ACCESS_NETWORK_STATE INTERNET READ_PHONE_STATE RECEIVE_BOOT_COMPLETED WAKE_LOCK Ljava/lang/refl	malware
ACCESS_WIFI_STATE CHANGE_WIFI_STATE GET_TASKS KILL_BACKGROUND_PROCESSES RECEIVE_BOOT_COMPLETED	malware
ACCESS_NETWORK_STATE DISABLE_KEYGUARD GET_TASKS INTERNET KILL_BACKGROUND_PROCESSES READ_PHONE	malware
ACCESS_NETWORK_STATE CAMERA GET_TASKS INTERNET READ_EXTERNAL_STORAGE READ_PHONE_STATE RECEIVE	malware
ACCESS_NETWORK_STATE DISABLE_KEYGUARD GET_TASKS INTERNET KILL_BACKGROUND_PROCESSES READ_PHONE	malware
ACCESS_CHECKIN_PROPERTIES ACCESS_COARSE_LOCATION ACCESS_FINE_LOCATION ACCESS_LOCATION_EXTRA_CO	malware
ACCESS_NETWORK_STATE CAMERA GET_ACCOUNTS GET_TASKS INTERNET READ_CONTACTS READ_EXTERNAL_STOR	malware
ACCESS_NETWORK_STATE DISABLE_KEYGUARD GET_TASKS INTERNET KILL_BACKGROUND_PROCESSES READ_PHONE	malware
ACCESS_WIFI_STATE CHANGE_WIFI_STATE GET_TASKS KILL_BACKGROUND_PROCESSES RECEIVE_BOOT_COMPLETED	malware
ACCESS_NETWORK_STATE DISABLE_KEYGUARD GET_TASKS INTERNET KILL_BACKGROUND_PROCESSES READ_PHONE	malware
ACCESS_WIFI_STATE CHANGE_WIFI_STATE GET_TASKS KILL_BACKGROUND_PROCESSES RECEIVE_BOOT_COMPLETED	malware
DISABLE_KEYGUARD GET_TASKS INTERNET KILL_BACKGROUND_PROCESSES RECEIVE_BOOT_COMPLETED SYSTEM_A	malware
ACCESS_WIFI_STATE CHANGE_WIFI_STATE GET_TASKS KILL_BACKGROUND_PROCESSES RECEIVE_BOOT_COMPLETED	malware

Fig.18: TUANDROMD features

Methodology [27]

Datasets:

Trojan Detection: The data contains the records of the traffics like Trojan Horse and Benign so the detection of Trojan and Benign can be done using Binary Classification

Class	text
Trojan	73217 10.42.0.42-121.14.255.84-49975-80-6 10.42.0.42 49975 121.14.255.84 80 6 17/07/2017 01:18:33 10743584 4 4 372.0 672.0 372.0 0.0 93.0 186.0 672.0 0.0 168.0 336.0 97.17427629364651 0.744630469683151 1534797.714285714 3734
Trojan	72089 172.217.6.226-10.42.0.42-443-49169-17 10.42.0.42 49169 172.217.6.226 443 17 17/07/2017 10:25:25 254217 6 7 3191.0 5246.0 1350.0 38.0 531.8333333333334 645.2160620030058 1350.0 30.0 749.4285714285714 678.206423134544
Benign	96676 10.42.0.1-10.42.0.42-53-37749-17 10.42.0.42 37749 10.42.0.1 53 17 30/06/2017 07:16:12 1023244 1 1 30.0 179.0 30.0 30.0 30.0 0.0 179.0 179.0 179.0 0.0 204.2523581863172 1.954568020921696 1023244.0 0.0 1023244.0 1023244.0 0.0
Trojan	42891 10.42.0.1-10.42.0.42-53-41352-17 10.42.0.42 41352 10.42.0.1 53 17 13/07/2017 03:48:44 286483 1 1 40.0 106.0 40.0 40.0 40.0 0.0 106.0 106.0 106.0 0.0 509.6288435963042 6.981217035565811 286483.0 0.0 286483.0 286483.0 0.0 0.0
Benign	169326 10.42.0.151-107.22.241.77-44353-443-6 10.42.0.151 44353 107.22.241.77 443 6 05/07/2017 10:47:35 65633087 12 10 767.0 5622.0 403.0 0.0 63.91666666666667 130.00171910285252 1448.0 0.0 562.1999999999999 649.538947596
Trojan	34510 10.42.0.211-10.42.0.1-6021-53-17 10.42.0.211 6021 10.42.0.1 53 17 11/07/2017 05:01:34 251336 1 1 37.0 182.0 37.0 37.0 37.0 0.0 182.0 182.0 182.0 0.0 871.3435401215902 7.957475252251966 251336.0 0.0 251336.0 251336.0 0.0 0.0
Trojan	59506 10.42.0.42-74.217.63.24-38871-443-6 10.42.0.42 38871 74.217.63.24 443 6 14/07/2017 01:48:47 3096 3 0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 968.9922480620156 1548.0 439.8204178980326 1859.0 1237.0 3096.0 1548.0 439.8204
Benign	98047 10.42.0.42-66.231.239.96-49387-443-6 10.42.0.42 49387 66.231.239.96 443 6 02/07/2017 08:24:19 473 1 2 46.0 31.0 46.0 46.0 46.0 0.0 31.0 0.0 15.5 21.920310216782973 162790.6976744186 6342.494714587738 236.5 132.2289680
Trojan	44044 10.42.0.1-10.42.0.42-53-34743-17 10.42.0.42 34743 10.42.0.1 53 17 13/07/2017 04:03:57 557409 1 1 25.0 79.0 25.0 25.0 25.0 0.0 79.0 79.0 79.0 0.0 186.57754001101523 3.588029615596447 557409.0 0.0 557409.0 557409.0 0.0 0.0 0.0
Trojan	84350 172.217.10.1-10.42.0.151-443-51786-6 172.217.10.1 443 10.42.0.151 51786 6 04/07/2017 10:07:47 314 2 0 55.0 0.0 55.0 0.0 27.5 38.8908729653 0.0 0.0 0.0 0.0 175159.23566879 6369.4267515924 314.0 0.0 314.0 314.0 314.0 314.0 0.0
Benign	140308 180.76.184.128-10.42.0.42-80-47733-6 10.42.0.42 47733 180.76.184.128 80 6 05/07/2017 03:30:54 518061 3 3 551.0 0.0 551.0 0.0 183.66666666666663 318.1199983234838 0.0 0.0 0.0 0.0 1063.5813157137866 11.58164772102127 10
Benign	149339 172.217.9.227-10.42.0.42-443-38191-6 10.42.0.42 38191 172.217.9.227 443 6 05/07/2017 09:35:37 46143837 9 9 448.0 4792.0 242.0 0.0 49.77777777777778 99.18389206138488 1418.0 0.0 532.4444444444443 671.8467293793858 11
Benign	90887 10.42.0.42-112.80.248.220-39888-443-6 10.42.0.42 39888 112.80.248.220 443 6 30/06/2017 04:54:01 34659660 26 28 6359.0 9213.0 1452.0 0.0 244.5769230769231 492.9489363475225 1460.0 0.0 329.0357142857142 515.197731342
Trojan	30597 10.42.0.211-10.42.0.1-62945-53-17 10.42.0.211 62945 10.42.0.1 53 17 11/07/2017 03:57:53 663034 1 1 30.0 118.0 30.0 30.0 30.0 0.0 118.0 118.0 118.0 0.0 223.21630564948404 3.016436562830865 663034.0 0.0 663034.0 663034.0 0.0
Benign	114034 140.205.230.8-10.42.0.42-80-35291-6 140.205.230.8 80 10.42.0.42 35291 6 02/07/2017 04:50:25 7 2 0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 285714.28571428574 7.0 0.0 7.0 7.0 7.0 7.0 0.0 7.0 7.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 40
Benign	98060 10.42.0.42-66.198.24.250-35120-443-6 10.42.0.42 35120 66.198.24.250 443 6 02/07/2017 08:20:58 4745734 40 58 1578.0 64471.0 552.0 0.0 39.44999999999999 96.17237462507433 1460.0 0.0 1111.5689655172414 574.853821604193
Benign	139010 216.58.217.68-10.42.0.42-443-46251-6 10.42.0.42 46251 216.58.217.68 443 6 05/07/2017 02:03:16 72918838 20 20 4556.0 5871.0 1368.0 0.0 227.8 449.3343146431056 1418.0 0.0 293.54999999999995 453.13498337459527 142.9945
Trojan	13077 172.217.12.162-10.42.0.211-443-60065-6 10.42.0.211 60065 172.217.12.162 443 6 11/07/2017 12:02:58 566018 31 42 2920.0 57213.0 978.0 0.0 94.19354838709675 292.52879281133323 1418.0 383.0 1362.2142857142856 220.345575
Benign	124498 10.42.0.1-10.42.0.42-53-64235-17 10.42.0.42 64235 10.42.0.1 53 17 02/07/2017 10:18:27 1482 1 1 36.0 246.0 36.0 36.0 36.0 0.0 246.0 246.0 246.0 0.0 190283.4008097166 1349.527665317139 1482.0 0.0 1482.0 1482.0 0.0 0.0 0.0 0.0 0.0
Benign	116573 202.77.129.230-10.42.0.42-80-56764-6 10.42.0.42 56764 202.77.129.230 80 6 02/07/2017 05:36:30 4542518 3 4 804.0 1964.0 804.0 0.0 268.0 464.1896164284591 1460.0 0.0 491.0 688.3051648796484 609.3536668429272 1.54099554
Benign	131470 208.80.154.224-10.42.0.42-80-47994-6 10.42.0.42 47994 208.80.154.224 80 6 04/07/2017 08:12:20 5410908 4 4 402.0 490.0 402.0 0.0 100.5 201.0 490.0 0.0 122.5 245.00000000000003 164.85218377396177 1.4784949217395675 7729
Benign	113980 10.42.0.42-104.193.88.109-57723-80-6 10.42.0.42 57723 104.193.88.109 80 6 02/07/2017 04:50:03 21856196 2 0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0915072320910738 21856196.0 0.0 21856196.0 21856196.0 21856196.0 2185
Benign	157416 10.42.0.42-54.230.51.28-44346-443-6 54.230.51.28 443 10.42.0.42 44346 6 05/07/2017 02:13:33 143 2 0 31.0 0.0 31.0 0.0 15.5 21.920310216782973 0.0 0.0 0.0 0.0 216783.2167832168 13986.013986013986 143.0 0.0 143.0 143.0 143.
Trojan	79012 205.185.216.10-10.42.0.42-80-33806-6 10.42.0.42 33806 205.185.216.10 80 6 17/07/2017 02:35:22 1397782 2 0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.4308382852261654 1397782.0 0.0 1397782.0 1397782.0 1397782.0 1397782.0 0
Benign	146071 172.217.10.129-10.42.0.42-443-49472-6 10.42.0.42 49472 172.217.10.129 443 6 05/07/2017 06:37:23 65339441 9 8 653.0 5638.0 250.0 0.0 72.55555555555556 107.51408176503102 1418.0 0.0 704.75 681.6339088647848 96.2818154
Benign	148759 10.42.0.42-98.139.180.149-35305-443-6 10.42.0.42 35305 98.139.180.149 443 6 05/07/2017 09:11:36 115680306 17 18 813.0 7881.0 297.0 0.0 47.823529411764696 84.69004907168672 1460.0 0.0 437.8333333333337 625.5383798

Fig.19: Trojan features

Methodology [28]

Datasets: It simulates real time data.

Synthetic Dataset

Banking Tr	I received a message prompting me to enter my banking credentials on a suspicious-looking app.
Ransomwa	My files are encrypted and inaccessible; I'm being asked to pay a ransom to unlock them, which indicates a ransomware attack.
Spyware	IÃ¢Â€Â™m seeing unexpected behavior on my device, such as unauthorized access to my personal data, which might be due to spyware.
Download	This app is secretly downloading other apps onto my phone, and some of them seem suspicious.
SMS Trojai	I noticed a lot of premium text messages being sent from my phone without my approval, likely due to an SMS Trojan.
Download	This app is secretly downloading other apps onto my phone, and some of them seem suspicious.
Worm	The app is self-replicating and has started infecting other devices, which indicates that it is a worm.
Ransomwa	I received a ransom note demanding payment to regain access to my files, which means my device has been hit by ransomware.
Cryptojack	My device is overheating and running much slower; it seems like a cryptojacker is using my CPU to mine cryptocurrency.
SMS Trojai	I noticed a lot of premium text messages being sent from my phone without my approval, likely due to an SMS Trojan.
Cryptojack	The performance of my phone has significantly decreased, possibly because a cryptojacker is running mining operations.
Adware	Ever since I installed this app, my phone is flooded with unwanted ads and my device performance has dropped.
Keylogger	I suspect that a keylogger might be capturing my keystrokes since I'm seeing unexpected logins on my accounts.
Worm	This app is replicating itself across my network and causing other devices to become infected, suggesting it's a worm.
Download	I noticed additional malware being installed on my phone without my consent, likely because of a downloader app.
Worm	This app is replicating itself across my network and causing other devices to become infected, suggesting it's a worm.

Fig.20: Synthetic Datasets

Methodology [29]

APT Notes Dataset: It is a collection of documents and notes related to APT (Advanced Persistent Threat).

	combined	Sentence	Malware/Attack Type
0	WickedRose_andNCPH "Wicked Rose" And The Ncph Hacking Group iDefense https://app.box.co	The hacking group known as 'Wicked Rose' an	Wicked Rose (Hacking Group)
1	Fritz_HOW-CHINA-WILL-USE-CYBER-WARFARE(Oct-01-08) How China Will Use Cyber Warfare Ja	Jason Fritz's report titled 'How China Will Use	Chinese Cyber Warfare
2	556_10535_798405_Annex87_CyberAttacks Russian Cyberwar On Georgia Georgia Gov https://a	The Russian cyberwar on Georgia in 2008 mar	Russian Cyberwar (Georgia)
3	Ashmore_Impact-of-Alleged-Russian-Cyber-Attacks(Jan-18-09) Impact Of Alleged Russian Cyber	William C. Ashmore's 2009 report addresses t	Russian Cyber Attacks
4	ghostnet Tracking Ghostnet: Investigating A Cyber Espionage Network Information Warfare Moni	'Tracking Ghostnet' investigates a complex cy	Ghostnet (Cyber Espionage)
5	Case_Study_Operation_Aurora_V11 Case Study: Operation Aurora Triumfant https://app.box.co	No clear malware type identified.	Unknown
6	Aurora_Botnet_Command_Structure The Command Structure Of The Aurora Botnet Damballa ht	No clear malware type identified.	Unknown
7	McAfee_Operation_Aurora Combating Aurora McAfee https://app.box.com/s/jhy5k76ox6z8sy6t	No clear malware type identified.	Unknown
8	Aurora_HBGARY_DRAFT Operation Aurora: Detect, Diagnose, Respond HBGary https://app.box.c	No clear malware type identified.	Unknown
9	HBGary_Operation_Aurora Operation Aurora HBGary https://app.box.com/s/fjb89qr1vnk2ox0vll	No clear malware type identified.	Unknown
10	how_can_u_tell_Aurora How Can I Tell If I Was Infected By Aurora? McAfee https://app.box.con	No clear malware type identified.	Unknown
11	in-depth_analysis_of_hydraq_final_231538 In-Depth Analysis Of Hydraq: The Face Of Cyberwar E	No clear malware type identified.	Unknown
12	Shadowserver_shadows-in-the-cloud Shadows In The Cloud: Investigating Cyber Espionage 2.0 Sh	No clear malware type identified.	Unknown
13	WashingtonPost_2010-Defense-official-discloses-cyberattack(08-24-2010) Defense official discl	No clear malware type identified.	Unknown
14	MSUpdaterTrojanWhitepaper The Msupdater Trojan And Ongoing Targeted Attacks Seculert, Zsc	No clear malware type identified.	Unknown
15	w32_stuxnet_dossier W32.Stuxnet Dossier Symantec https://app.box.com/s/rpdy3pk00bmkhgm	No clear malware type identified.	Unknown
16	wp-global-energy-cyberattacks-night-dragon Global Energy Cyberattacks: Night Dragon McAfee h	No clear malware type identified.	Unknown
17	Alerts DL-2011 Alerts-A-2011-02-18-01 Night Dragon Attachment 1 Night Dragon: Specific Protec	No clear malware type identified.	Unknown
18	Stuxnet_Under_the_Microscope Stuxnet Under The Microscope ESET https://app.box.com/s/2m	No clear malware type identified.	Unknown
19	C5_APT_ADecadeInReview Advanced Persistent Threats: A Decade In Review Command Five Pty	No clear malware type identified.	Unknown
20	shady_rat_vanity Operation Shady Rat: Unprecedented Cyber-Espionage Campaign And Intellectu	No clear malware type identified.	Unknown
21	HTran_and_the_Advanced_Persistent_Threat Htran And The Advanced Persistent Threat Dell Sec	No clear malware type identified.	Unknown
22	wp-operation-shady-rat Revealed: Operation Shady Rat McAfee https://app.box.com/s/a086wzc	No clear malware type identified.	Unknown
23	wp_dissecting-lurid-apt The Lurid Downloader Trend Micro https://app.box.com/s/7s9bvquu64v	No clear malware type identified.	Unknown
24	C5_APT_SKHack Sk Hack By An Advanced Persistent Threat Command Five Pty Ltd https://app.bo	No clear malware type identified.	Unknown
25	tb_advanced_persistent_threats Alleged Apt Intrusion Set: 1.Php Group Zscaler, ThreatLabz https	No clear malware type identified.	Unknown

Fig.21: APT Notes Datasets

Methodology [30]

Datasets:

Final Dataset

SCAREWAI	On 27/06/2017 at 03:27:43, a suspicious flow with ID 10.42.0.211-123.125.125.56-59243-443-6 was detected, where the source IP 123.125.125.56 (port 443) communicated with the destination IP 10.42.0.211 (port 59243) using protocol 6.0. Th
RANSOMV	The file with hash '0124e21d-018c-4ce0-92a3-b9e205a76bc0.dll' has properties including a file size of 79755c51e413ed3c6be4635fd729a6e1 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory usage of
ADWARE	Your browser has been redirected to BestDeals.com for the best online shopping discounts!
ADWARE	In our Android application, we utilize the ACCESS_NETWORK_STATE permission to check network connectivity, the CAMERA permission to access the device's camera for taking pictures using Camera.open and Camera.takePicture, the GET_ACCC
RANSOMV	The file with hash '05c8318f98a5d301d80000009c316005.vertdll.dll' has properties including a file size of 95e19f3657d34a432eada93221b0ea16 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory usag
SMSMALW	In our Android application, we utilize the SmsManager class to send SMS messages with the SEND_SMS permission, register and unregister broadcast receivers using Context.registerReceiver and Context.unregisterReceiver for handling the RECEI
TROJAN	This malware utilizes several Android permissions and functionalities to compromise user privacy and device security. By leveraging ACCESS_NETWORK_STATE, it monitors network connectivity, while BATTERY_STATS allows it to access battery u
TROJAN	On March 17, 2021, at 8:02 AM, the application with the package name com.firstchoice.myfirstchoice generated a significant data entry with a unique identifier 000001A94F46A0C3DDA514E1F24E675648835BBA5EF3C3AA72D9C378534FCAD6, re
BENIGN	On June 16, 2017, at 03:55:47, a suspicious network flow was observed between IP addresses 10.42.0.211 (source) and 172.217.2.174 (destination), with source port 51023 and destination port 443, using protocol 6 (TCP). The flow, which lasted f
RANSOMV	The file with hash '090607dd9ba5d301ca0900009c316005.SensorsNativeApi.V2.dll' has properties including a file size of ae38c5f7d313ad0ff3bfb8826476767f bytes, multiple zero values indicating the absence of specific permissions or intents, a
TROJAN	On April 6, 2021, at 11:00 AM, the application with the package name com.firstchoice.myfirstchoice processed data associated with the unique identifier 000001A94F46A0C3DDA514E1F24E675648835BBA5EF3C3AA72D9C378534FCAD6, which in
BENIGN	On June 16, 2017, at 04:00:36, a network flow was detected between IP address 10.42.0.211 (source) and 199.59.148.73 (destination), using source port 33772 and destination port 443, with TCP protocol 6. The flow lasted for 453,887 millisecon
RANSOMV	The file with the hash '11b25aa499a5d301370400009c316005.pngfilt.dll' has properties including a file size of a2f43fac128db67452726d096fb77768 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory u
SMSMALW	A meeting is scheduled for 10 AM tomorrow, as noted in the message from evanmccann@gmail.com, received on April 19, 2024, at 12:21 AM, with a message length of 26 characters.. This message was sent on unknown date at unknown time by
SPYWARE	The application requests permissions such as ACCESS_NETWORK_STATE, BATTERY_STATS, and INTERNET, and utilizes Java methods like Cipher.doFinal and TelephonyManager.getNetworkCountryIso to manage network states and obtain the co
SPYWARE	On January 1, 1980, at 0:00, the application with the package name com.fearless.teengirl version 3.0, having an identifier of 000010F33B578B4B5C33724520E38BD44485705DC1B6ADDE4975EBBA7F114355 DDE9CA9CC8C9FD67B27E68A637D10
RANSOMV	The file with hash 'MSBuildTaskHost.resources (12).dll' has properties including a file size of d002e0973af380781e85fdc8b0ddf683 bytes, multiple zero values indicating the absence of specific permissions or intents, a memory usage of 0 bytes, 81
SCAREWAI	On June 27, 2017, at 03:17:21, a network flow was recorded between the source IP 23.203.49.224 (port 443) and the destination IP 10.42.0.211 (port 34092) using protocol 6.0. The flow, lasting 13 seconds, included 2 forward packets with a total
TROJAN	The application requests permissions such as ACCESS_NETWORK_STATE, INTERNET, READ_PHONE_STATE, RECEIVE_BOOT_COMPLETED, and WAKE_LOCK, and utilizes Java methods like Method.invoke and Cipher.doFinal to manage network sta
ADWARE	On January 1, 1980, at 00:00, the application com.fearless.teengirl version 3.0, with identifiers 000010F33B578B4B5C33724520E38BD44485705DC1B6ADDE4975EBBA7F114355 DDE9CA9CC8C9FD67B27E68A637D1015E3852AD4E 8CEE4A742DD0
SPYWARE	On June 16, 2017, at 03:55:40, a network flow was recorded from the source IP 10.42.0.211 (port 43070) to the destination IP 239.255.255.250 (port 1900) using protocol 17 (UDP). The flow lasted 598,272 seconds and included 6 forward packet
ADWARE	On June 14, 2017, at 01:54:51, a network flow was observed between source IP 10.42.0.151 (port 36635) and destination IP 172.217.2.106 (port 443), using protocol 6 (TCP). The flow lasted 22,287 seconds and included 1 forward packet and 1 ba
RANSOMV	The file with the hash 'MSBuildTaskHost.resources (16).dll' has properties including a file size of de12836d9fb3050b1c4c5c47770624ee bytes, multiple zero values indicating the absence of specific permissions or intents, a memory usage of 0 byte
SPYWARE	On February 8, 2021, at 13:59, the application com.placz.cricketpakistan version 2.0, identified by the hexadecimal strings 0000120C5998A69C7907706CA1BDBBF98CCD884891D68CDE4D87BB530AEAF015 9DB718A09CE7DEEFFAA2192A6A19A42
TROJAN	On January 1, 1981, at 01:01, the application com.northcube.sleepcycle, identified by the hexadecimal strings 00001438D92DC8AAA7B1BCEDDC4BEBC12E3104C02142FEAE1F929E3A0C2BC719 6AA10B10EE30A7525528F5A0A639836E181648E6 9
ADWARE	On June 16, 2017, at 04:04:52, a network flow was recorded between source IP 10.42.0.211 (port 45130) and destination IP 23.53.114.188 (port 443), using protocol 6 (TCP). The flow lasted 3 seconds and included 2 forward packets, with no data

Fig.22:Combined Malware Features

Methodology [31]

Confusion Metrics:

True Positive (TP):Correctly predicted positive instances (malware samples).

True Negative (TN): Correctly predicted negative instances (non-malware samples).

False Positive (FP): Non-malware samples incorrectly classified as malware.

False Negative (FN):Malware samples incorrectly classified as non-malware.

Methodology [32]

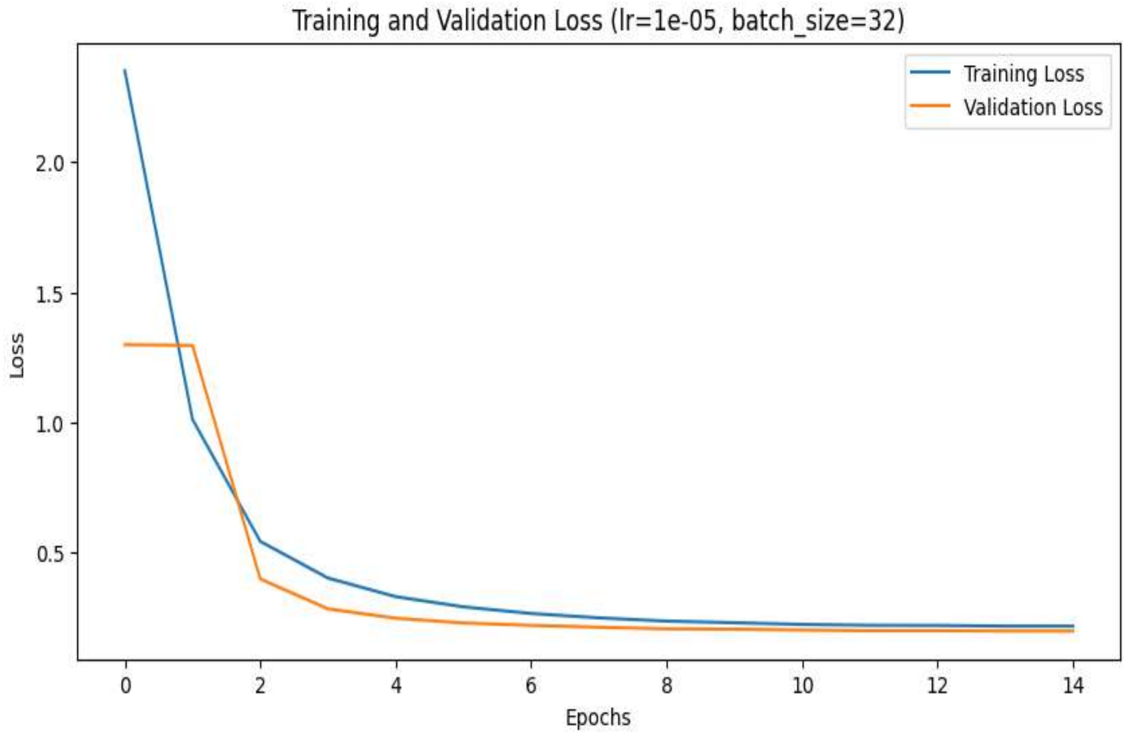
ROC-AUC:

- ROC Curve: A graphical plot that illustrates the diagnostic ability of a binary classifier as its discrimination threshold is varied.
- AUC(Area under the curve): The measure of the ability of a classifier to distinguish between classes.

The higher the AUC, the better the model is at predicting positives as positives and negatives as negatives.

Results[1]

Scenarios and Output (Best Case) I:



Epoch	Training Loss	Validation Loss	Accuracy	Precision	Recall	F1
1	1.660700	1.298315	0.765157	0.717394	0.765157	0.714884
2	0.694100	0.400591	0.906574	0.903809	0.906574	0.903211
3	0.446300	0.285729	0.914849	0.909724	0.914849	0.911580
4	0.403700	0.249736	0.920867	0.915783	0.920867	0.917656
5	0.267400	0.231484	0.930495	0.925657	0.930495	0.927428
6	0.258900	0.222350	0.930946	0.925851	0.930946	0.927795
7	0.259700	0.215159	0.930645	0.924804	0.930645	0.927240
8	0.271100	0.209107	0.931398	0.926000	0.931398	0.928147
9	0.218800	0.207782	0.932150	0.927134	0.932150	0.928894
10	0.242200	0.204563	0.932601	0.926886	0.932601	0.929253
11	0.238500	0.201759	0.932902	0.927344	0.932902	0.929635
12	0.260500	0.201864	0.932902	0.927428	0.932902	0.929561
13	0.237600	0.200394	0.933053	0.927488	0.933053	0.929719
14	0.191900	0.200505	0.933053	0.927627	0.933053	0.929775
15	0.194100	0.200191	0.933203	0.927876	0.933203	0.929928

Fig.23:Loss PLOT (Best Case)

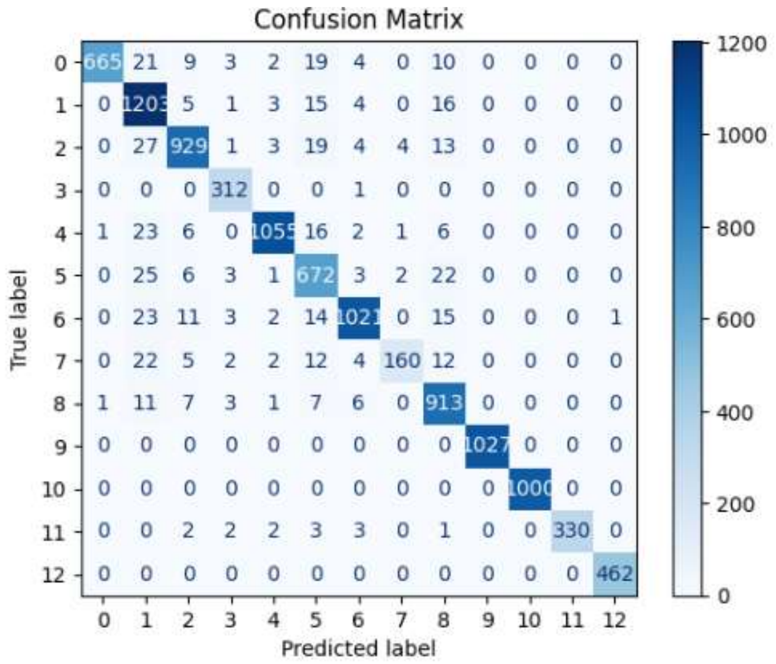
Fig. 24 Table (Best Case)

Results[2]

Scenarios and Output (Best Case):

Classification Report:

	precision	recall	f1-score	support
scareware	0.91	0.95	0.93	1000
ransomware	0.95	0.96	0.95	1635
adware	0.91	0.96	0.93	1365
smsmalware	0.94	0.94	0.94	1047
trojan	0.97	0.95	0.96	1377
benign	0.90	0.94	0.92	1062
spyware	0.97	0.94	0.95	1077
polymorphic	0.98	0.78	0.87	247
downloader	1.00	0.95	0.97	1260
cryptojacker	1.00	1.00	1.00	1000
worm	1.00	1.00	1.00	1269
fake app	1.00	1.00	1.00	356
keylogger	1.00	1.00	1.00	463
accuracy			0.96	13158
macro avg	0.96	0.95	0.96	13158
weighted avg	0.96	0.96	0.96	13158



Accuracy: 0.9533

Fig.25: Classification Report (Best Case)

Fig.26: Confusion Matrix

Results[3]

Scenarios and Output (Best Case) :

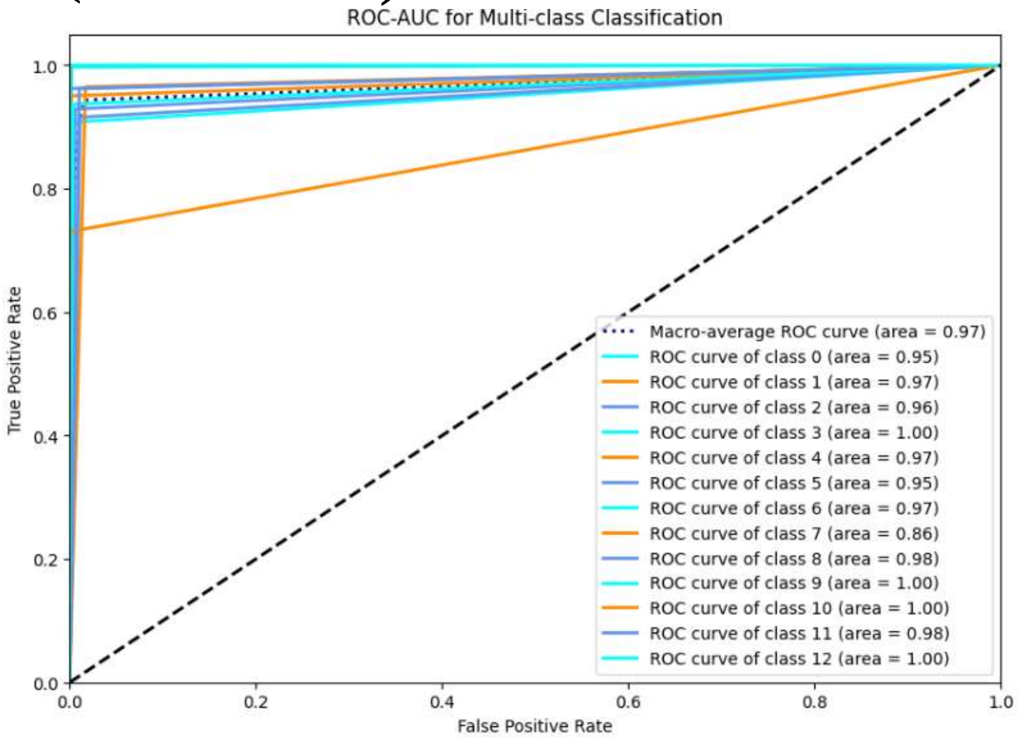


Fig.27: AUC-ROC Curve

Results[4]

Scenarios and Output (Best Case (Inference)) :

```
Welcome to Malware Message Classification
Enter a message to classify or type 'exit' to quit.
Message:  I noticed that this malware is spreading to other devices in my network, which is typical behavior of a worm.
Predicted Label: worm
Message:  This app is replicating itself across my network and causing other devices to become infected, suggesting it's a worm.
Predicted Label: worm
Message:  A network scan revealed unusual traffic and multiple infections across different devices. This suggests the presence of a worm spreading through vulnerabilities.
Predicted Label: worm
Message:  Multiple systems on your network are infected and spreading malware independently. This pattern suggests the presence of a worm.
Predicted Label: worm
Message:  Your network is congested with traffic, and several devices show signs of infection. The behavior is consistent with a worm that spreads by exploiting network vulnerabilit
es.
Predicted Label: worm
Message:  exit
Exiting the program.
```

Fig.28: Best Case (Inference)

Results[5]

Scenarios and Output (Worst Case) :

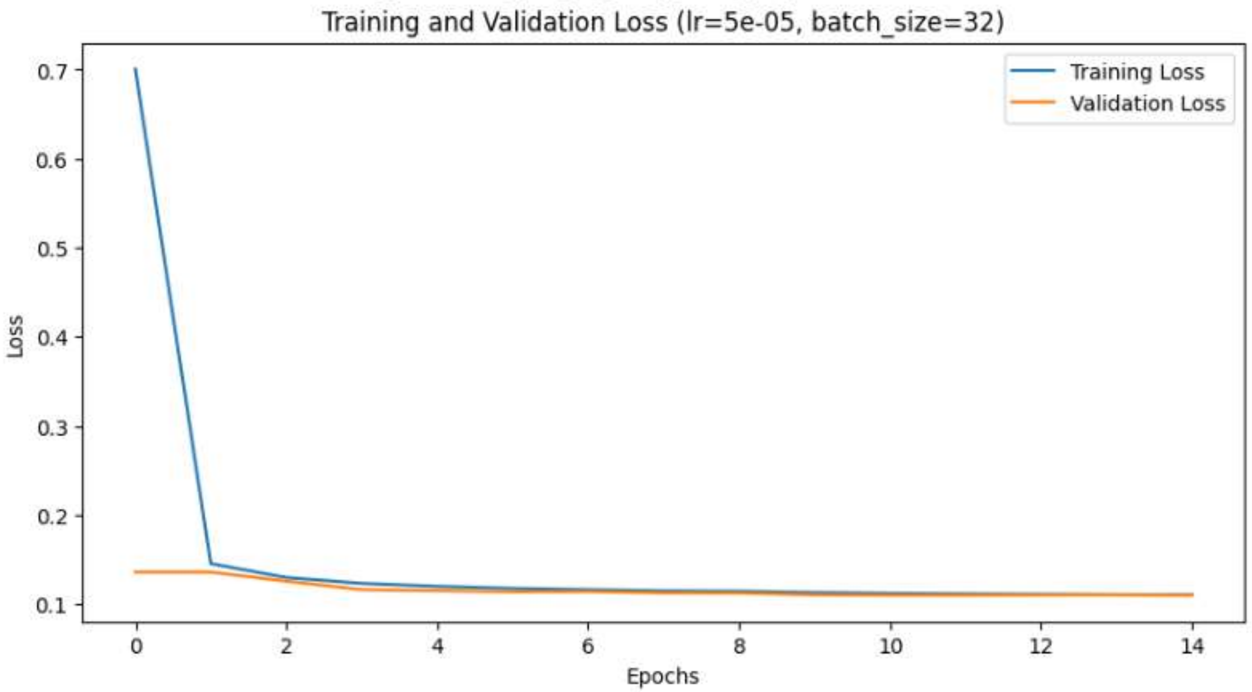


Fig.29: Loss Plot (WorstCase)

[9870/9870 1:58:54, Epoch 15/15]

Epoch	Training Loss	Validation Loss	Accuracy	Precision	Recall	F1
1	0.173400	0.136197	0.956018	0.957808	0.956018	0.956157
2	0.122700	0.125977	0.956968	0.959360	0.956968	0.957293
3	0.127400	0.116459	0.957728	0.960298	0.957728	0.957984
4	0.105900	0.115382	0.956398	0.957793	0.956398	0.956455
5	0.110300	0.114094	0.957823	0.959623	0.957823	0.957943
6	0.103400	0.114455	0.958298	0.963167	0.958298	0.959146
7	0.080300	0.112811	0.957253	0.957683	0.957253	0.957091
8	0.134300	0.112662	0.957633	0.959190	0.957633	0.957680
9	0.149000	0.110726	0.957443	0.958582	0.957443	0.957423
10	0.117700	0.110265	0.956493	0.957325	0.956493	0.956531
11	0.125600	0.110242	0.958298	0.959743	0.958298	0.958442
12	0.115400	0.110577	0.956873	0.958036	0.956873	0.956968
13	0.096900	0.110575	0.956778	0.957882	0.956778	0.956800
14	0.131200	0.110187	0.956683	0.957426	0.956683	0.956627
15	0.111600	0.110167	0.956398	0.957118	0.956398	0.956353

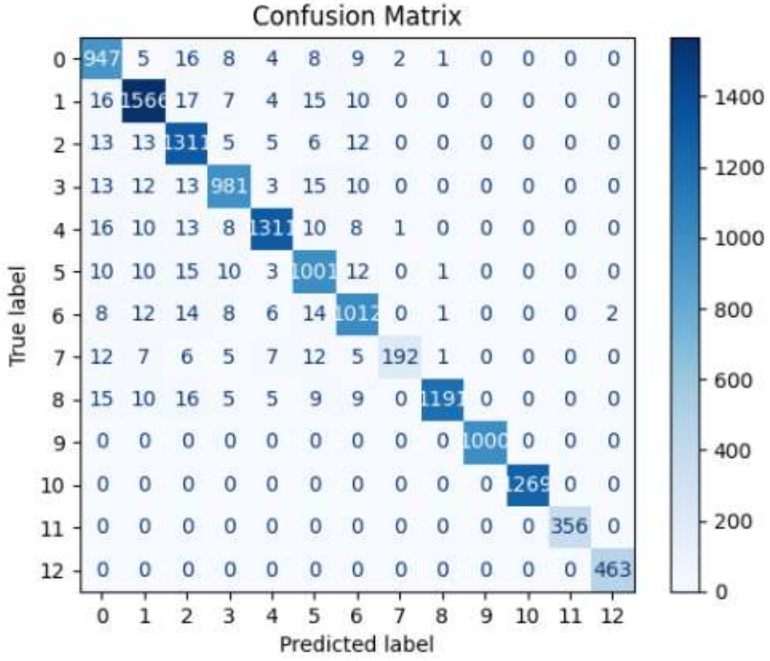
Fig.30: Training Table(Worst Case)

Results[6]

Scenarios and Output (Worst Case):

Classification Report:

	precision	recall	f1-score	support
scareware	0.90	0.95	0.92	1000
ransomware	0.95	0.96	0.95	1635
adware	0.92	0.96	0.94	1365
smsmalware	0.95	0.94	0.94	1047
trojan	0.97	0.95	0.96	1377
benign	0.92	0.94	0.93	1062
spyware	0.93	0.94	0.94	1077
polymorphic	0.98	0.78	0.87	247
downloader	1.00	0.95	0.97	1260
cryptojacker	1.00	1.00	1.00	1000
worm	1.00	1.00	1.00	1269
fake app	1.00	1.00	1.00	356
keylogger	1.00	1.00	1.00	463
accuracy			0.96	13158
macro avg	0.96	0.95	0.96	13158
weighted avg	0.96	0.96	0.96	13158



Accuracy: 0.9576

Fig.: 31 Classification Report (Worst Case)

Fig.32: Confusion Metrics (Worst Case)

Results[7]

Scenarios and Output (Worst Case):

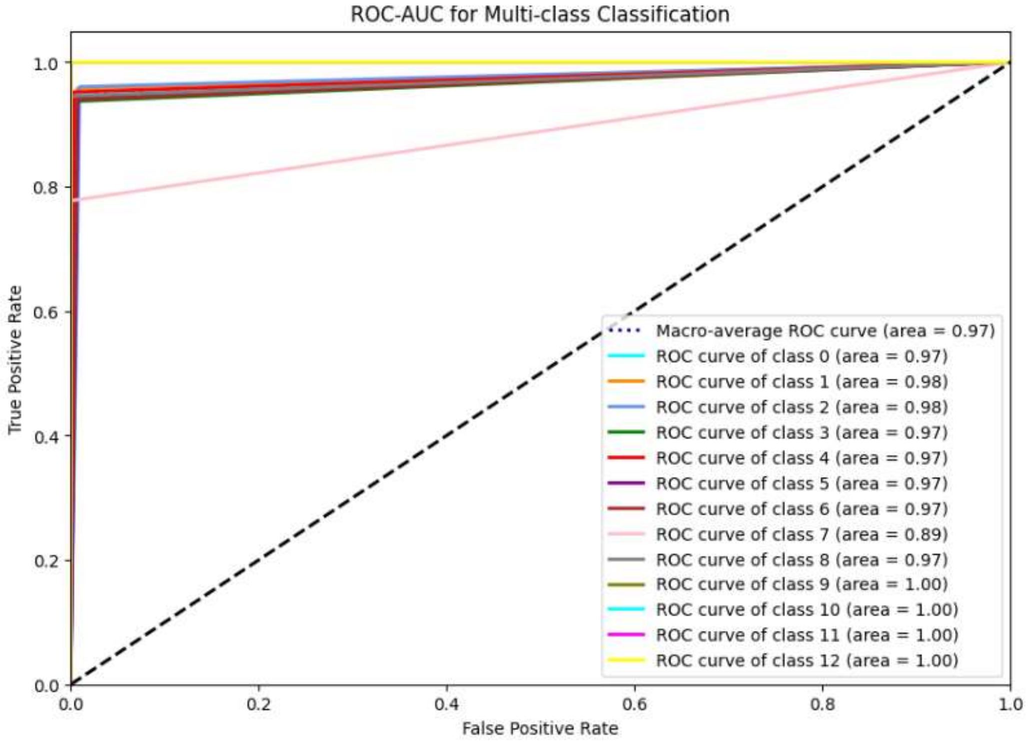
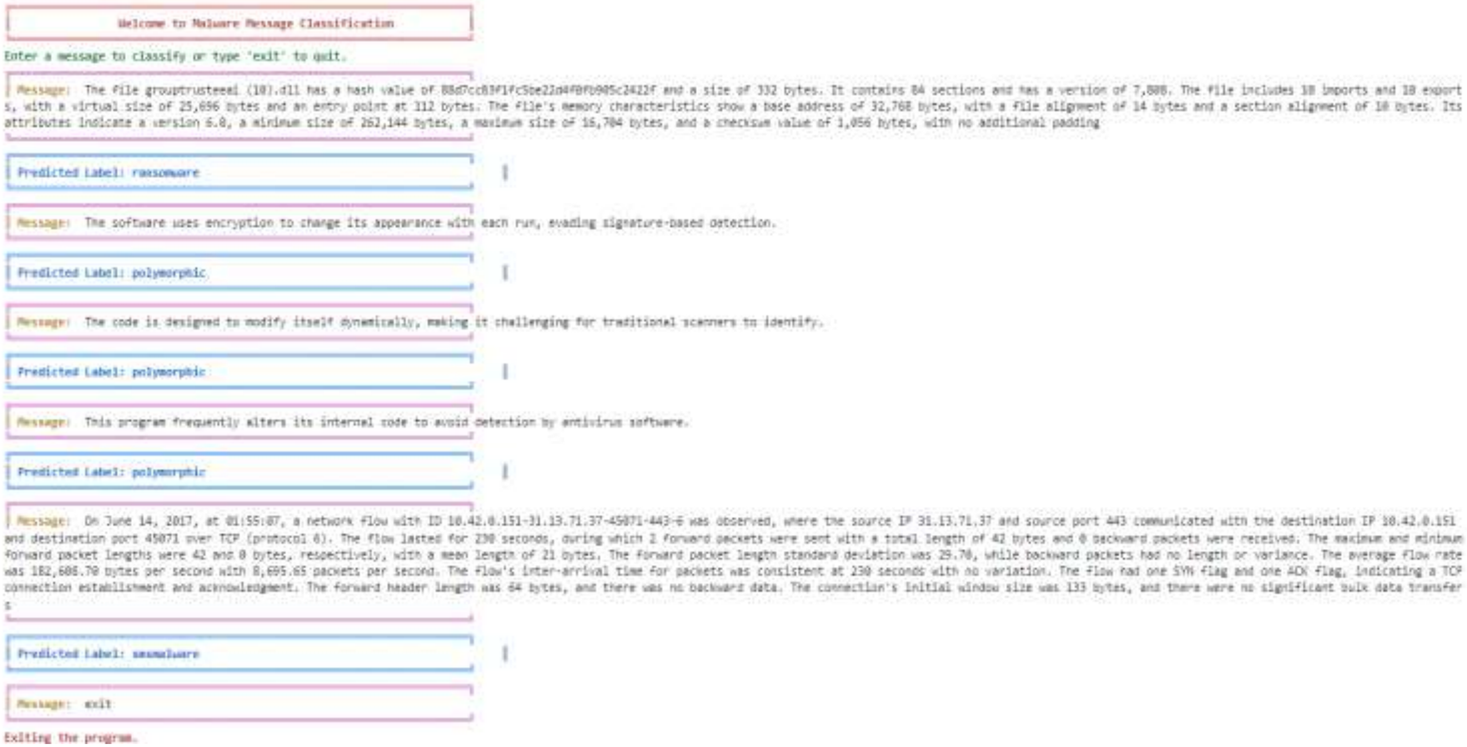


Fig.33: AUC-ROC Curve

Results[8]

Scenarios and Output (Worst Case (Inference)) :



```
Welcome to Malware Message Classification
Enter a message to classify or type 'exit' to quit.
Message:  The file grouptrusteeai (10).dll has a hash value of 88d7cc83f1fc5be22d4f0fb905c2422f and a size of 332 bytes. It contains 84 sections and has a version of 7,808. The file includes 18 imports and 18 exports, with a virtual size of 25,696 bytes and an entry point at 112 bytes. The file's memory characteristics show a base address of 32,768 bytes, with a file alignment of 14 bytes and a section alignment of 10 bytes. Its attributes indicate a version 6.0, a minimum size of 262,144 bytes, a maximum size of 16,704 bytes, and a checksum value of 1,056 bytes, with no additional padding
Predicted Label: ransomware
Message:  The software uses encryption to change its appearance with each run, evading signature-based detection.
Predicted Label: polymorphic
Message:  The code is designed to modify itself dynamically, making it challenging for traditional scanners to identify.
Predicted Label: polymorphic
Message:  This program frequently alters its internal code to avoid detection by antivirus software.
Predicted Label: polymorphic
Message:  On June 14, 2017, at 01:55:07, a network flow with ID 10.42.0.151-31.13.71.37-45071-443-6 was observed, where the source IP 31.13.71.37 and source port 443 communicated with the destination IP 10.42.0.151 and destination port 45071 over TCP (protocol 6). The flow lasted for 230 seconds, during which 2 forward packets were sent with a total length of 42 bytes and 0 backward packets were received. The maximum and minimum forward packet lengths were 42 and 0 bytes, respectively, with a mean length of 21 bytes. The forward packet length standard deviation was 29.70, while backward packets had no length or variance. The average flow rate was 182,608.70 bytes per second with 8,695.65 packets per second. The flow's inter-arrival time for packets was consistent at 230 seconds with no variation. The flow had one SYN flag and one ACK flag, indicating a TCP connection establishment and acknowledgment. The forward header length was 64 bytes, and there was no backward data. The connection's initial window size was 133 bytes, and there were no significant bulk data transfers
Predicted Label: smsmalware
Message:  exit
Exiting the program.
```

Fig.34: Worst Case (Inference)

Discussion and Analysis[1]

- Theoretical Output : 0.95-1 accuracy
- Simulated Output:

Model Name	Best Case (15 epoch)	Worst Case (15 epochs)
SecureBERT	0.9533	0.9576

Table 1: Best Case Accuracy and Worse Case Accuracy

- Best Case Scenario : Learning Rate (1e-5, Batch Size:32,10 epochs)
- Worst Case Scenario: Learning Rate (5e-5,Batch Size: 32, 15 epochs

Potential Reasons for Disrepancies:

1. Class Imbalance.
2. Model Training and Evaluation.

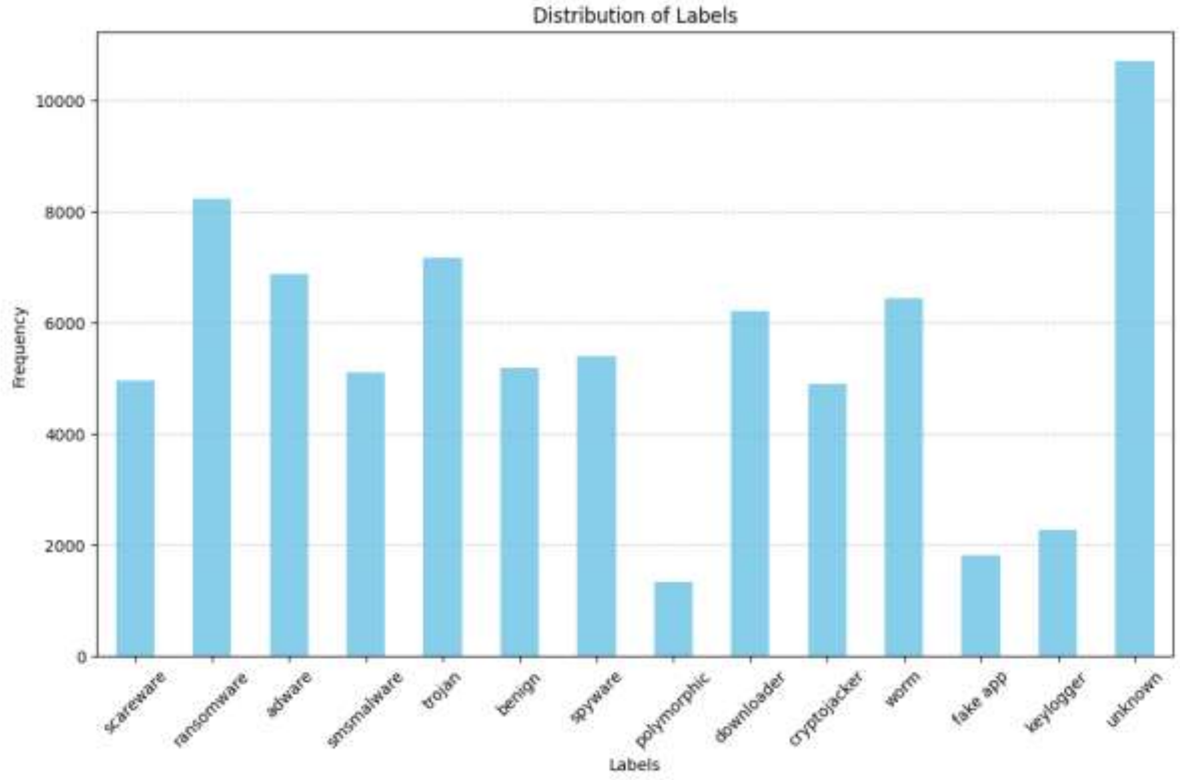
Discussion and Analysis[2]

Potential Sources of Errors:

- Data Quality.
- Model Training and Evaluation
- Model Complexity Mismatch
- Class Imbalance.

Discussion and Analysis[3]

Error Analysis

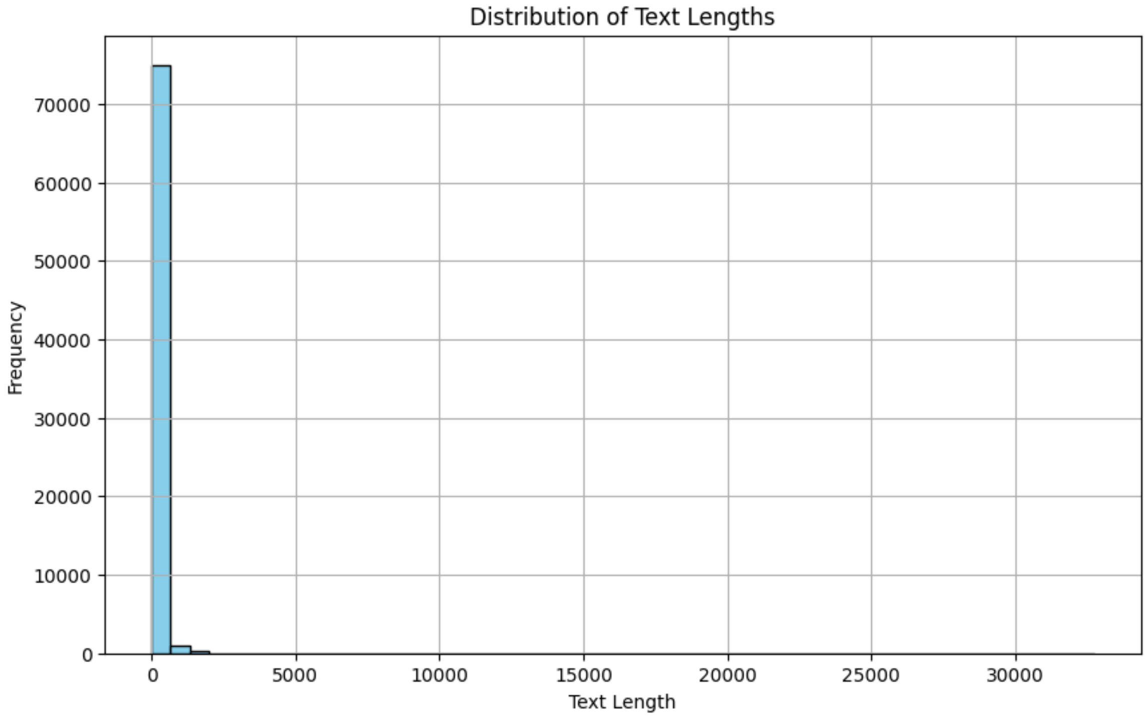



```
Label Distribution:
scareware: 4952
ransomware: 8221
adware: 6865
smsmalware: 5098
trojan: 7167
benign: 5178
spyware: 5385
polymorphic: 1320
downloader: 6204
cryptojacker: 4897
worm: 6422
fake app: 1811
keylogger: 2269
unknown: 10690
```

Fig.35 Label Distribution of different malwares used in datasets

Discussion and Analysis[4]

Error Analysis



Text Length Characteristics:
Minimum Length: 0
Maximum Length: 32759
Mean Length: 160.21
Median Length: 110.0
Standard Deviation of Length: 581.51

Fig.36 Distribution of Text lengths

Discussion and Analysis[4]

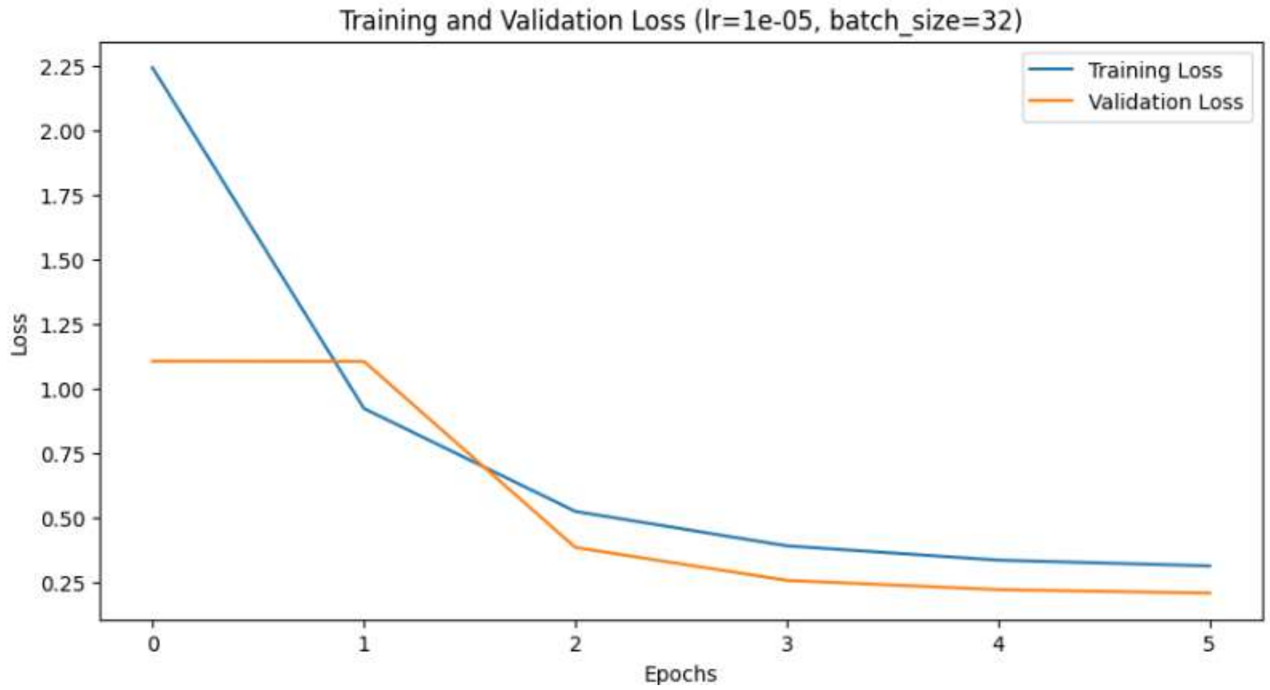
Hyperparameter Tuning

Parameter	Values
Learning Rate	1e-4,1e-5,2e-5,3e-4,3e-5
Training Batch Size	16,32,64,128
Epochs	3,4,5,6,10,15

Table 2: Parameters used for Hyperparameter Tuning

- Hyperparameter Tuning involves adjusting various parameters to find the optimal settings that affect the model’s accuracy and generalization ability.

Discussion and Analysis[5]



Epoch	Training Loss	Validation Loss	Accuracy	Precision	Recall	F1
1	1.426400	1.107367	0.765705	0.706801	0.765705	0.714075
2	0.625800	0.384912	0.909435	0.914985	0.909435	0.909250
3	0.458700	0.257342	0.926668	0.930368	0.926668	0.927417
4	0.386700	0.221378	0.933390	0.936084	0.933390	0.933846
5	0.314000	0.207784	0.935957	0.938795	0.935957	0.936424
6	0.370500	0.204298	0.936690	0.939680	0.936690	0.937163

Fig.37 Plots, table used while training model at different phases

Discussion and Analysis[6]

Comparison with State of Art workers:

- MALBERT

Accuracy: 0.9757(MixG-Androzoo), 0.9240(MixG-VirusShare)

F1-score: 0.9762 (MixG-Androzoo), 0.9247(MixG-VirusShare)

- MALBERTv2

Accuracy Range: 0.8224 to 0.9376 across datasets

Future Enhancement[1]

- **Possible enhancements in dataset**
- Increase Dataset size: Expand the dataset with more diverse datasets.
- Balanced Dataset: Ensure dataset is well balanced among different classes.
- **Selection of improved instruments**
- Ensemble methods: Combining multiple models and traditional machine learning classifiers.
- Experiment with real-time system: Implement real-time detection systems. Testing malware in controlled environment.

Future Enhancement[2]

- **Transfer Learning from related domains**: Utilize transfer learning from other cybersecurity domains.
- **Experiment with real-time system:** Implement real-time detection systems. Testing malware in controlled environment.

Conclusion[1]

- **Effective Malware Classification:** various type of malwares along with their characteristics were analyzed in textual format.
- **Data Augmentation and diversity:** Datasets of different categories were applied which were different from each other.
- **Interactive Classification Interface:** The interface provided an intuitive interface for users to input data and receive instant results.
- **Finetuning:** SecureBERT models were finetuned using LoRA method, through which considerable accuracy was achieved.
- **Analyze Model Performance Across Diverse Dataset** (Partially Met)
- **Adressing Class Imbalances** (Partially Met)

Tentative Timeline

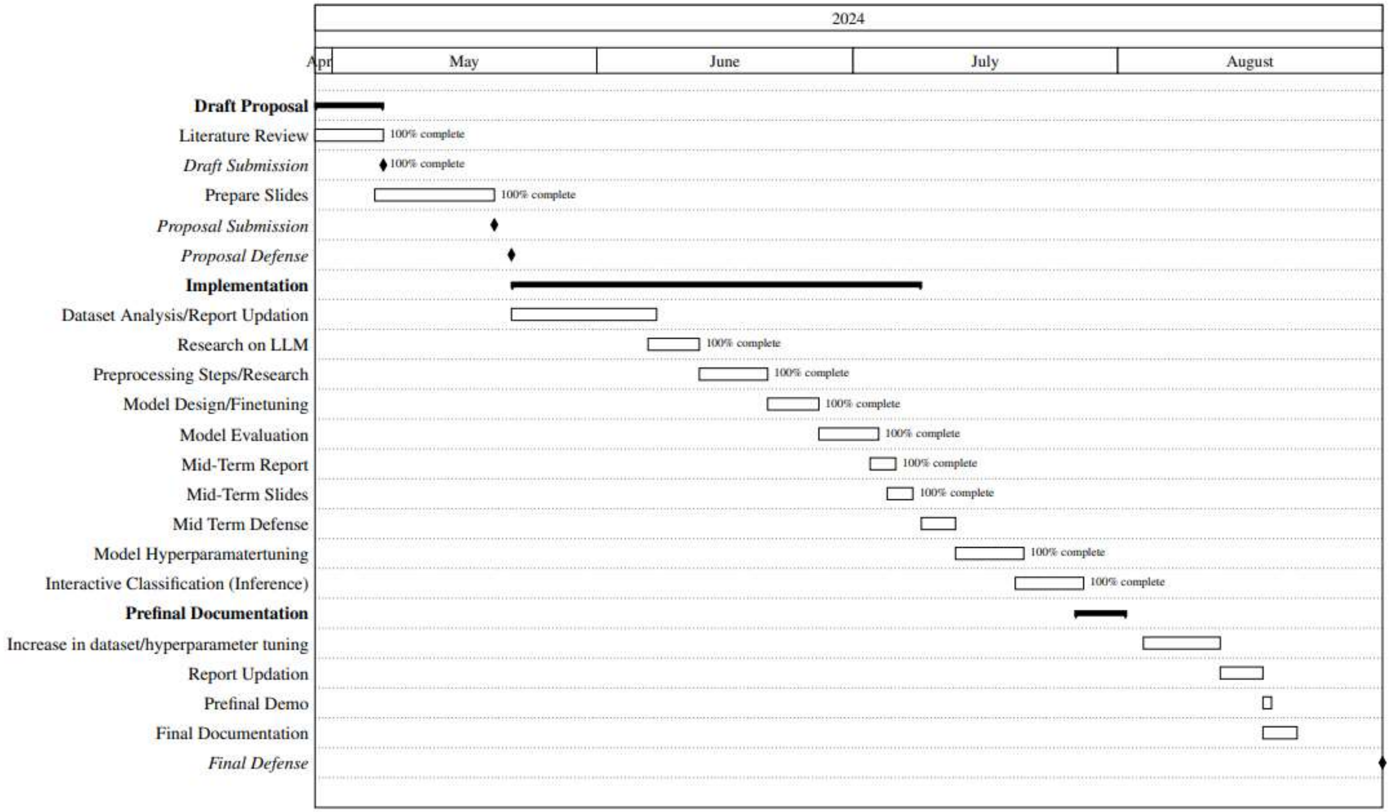


Fig.38: Tentative Timeline Chart

References

[1] Einfochips. Malware detection using machine learning techniques. Einfochips Blog, 2023. https://www.einfochips.com/blog/ malware-detection-using-machine-learning-techniques/.

[2] N. Sahin. Malware detection using transformers-based model gpt-2, 2021.

[3] Ahmet Selman Bozkir, Ersan Tahillioglu, Murat Aydos, and Ilker Kara. Catch them alive: A malware detection approach through memory forensics, manifold learning, and computer vision. Computers & Secu rity, 102166, 2021.

[4] I. Zborovska, E. Zatsarinnaya, A. Zaytsev, and D. Artemenko. Method ology of creating the innovation clusters in the system of regional entrepreneurship. Przeglad Organizacji, 3:17–24, 2020.

[5] TechTarget. What is the bert language model? TechTarget, 2024. Accessed: 2024-07-20.

[6] DS Stream. Roberta vs bert: tion of transformer models. Exploring the evolu https://dsstream.com/ roberta-vs-bert-exploring-the-evolution-of-transformer-models/ #:~:text=BERT%3A%20BERT%20uses%20a%20smaller,sequence% 20length%20of%20512%20tokens., 2024.

[7] Word2vec. https://en.wikipedia.org/wiki/Word2vec, May 2024. Accessed: 2024-05-29.

References

[8] Max Kuhn and Kjell Johnson. Feature Engineering and Selection: A Practical Approach for Predictive Models. CRC Press, Boca Raton, FL, 2019.

[9] Author Name. Understanding lora with python implementation. Medium, 2023. Accessed: YYYY-MM-DD.

[10] R. Hu, A. Zhang, E. Liu, and et al. Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2104.07640, 2021

[11] Word2vec. https://en.wikipedia.org/wiki/Word2vec, May 2024. Accessed: 2024-05-29.

[12] Max Kuhn and Kjell Johnson. Feature Engineering and Selection: A Practical Approach for Predictive Models. CRC Press, Boca Raton, FL, 2019.

[13] Author Name. Understanding lora with python implementation. Medium, 2023. Accessed: YYYY-MM-DD.

[14] Bhavin Jawade. Understanding lora: Low-rank adaptation for finetuning large models, June 2023. Accessed: 2024-08-20.

References

[15] R. Hu, A. Zhang, E. Liu, and et al. Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2104.07640, 2021.

[16] B. Catak and Y. Yazi. A benchmark api call dataset for windows pe malware classification. In Proceedings of the 2020 IEEE European Symposium on Security and Privacy (EuroSP), pages 411–425. IEEE, 2020.

[17] Bai Liang, Christian Hlauschek, Yajin Zhou, Xuan Wang, and Yuanchun Xue. Androzoo: Collecting millions of android apps for the research community. https://androzoo.uni.lu/, 2016.

[18] S. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, K. Rieck, and ¨ C. Siemens. Drebin: Efficient and explainable detection of android malware. https://drebin.mlsec.org/, 2024. [Accessed: Jun. 1, 2024].

[19] H. Shiravi, A. Shiravi, and A. A. Ghorbani. A realistic dataset for anomaly-based network intrusion detection. https://www.unb.ca/ cic/datasets/andmal2017.html, July 2017. Accessed: 2024-06- 04.

[20] amdj3dax. Ransomware detection data set, 2023. Accessed: 2024-07- 28.

[21] subhajournal. Trojan detection, 2024. Accessed: 2024-07-28.

References

[22] Wikipedia contributors. Receiver operating characteristic, 2024. Ac cessed: 05-Jul-2024.

Thank you

Any Queries?