Event Type Classification and Trend Analysis of Cybersecurity News Using Dynamic Topic Modeling

by Adewale Akinfaderin

B.S. in Applied Physics (Geophysics), November 2008, University of Lagos
M.S. in General Physics, August 2012, North Carolina Central University
M.S. in Physics, August 2014, Florida State University

A Praxis submitted to

The Faculty of
The School of Engineering and Applied Science
of The George Washington University
in partial satisfaction of the requirements
for the degree of Doctor of Engineering

August 31, 2023

Praxis directed by

Amir Etemadi
Associate Professor of School of Engineering and Applied Science

The School of Engineering and Applied Science of The George Washington University certifies that Adewale Akinfaderin has passed the Final Examination for the degree of Doctor of Engineering as of July 29, 2023. This is the final and approved form of the Praxis.

Event Type Classification and Trend Analysis of Cybersecurity News Using Dynamic Topic Modeling

Adewale Akinfaderin

Praxis Research Committee:

Amir Etemadi, Associate Professor of School of Engineering and Applied Science, Praxis Director

Rebecca Yassan, Professorial Lecturer in Engineering and Applied Science, Committee Member

Donald Widener, Professorial Lecturer in Engineering and Applied Science, Committee Member

© Copyright 2023 by Adewale Akinfaderin
All rights reserved

Dedication

I would like to dedicate this praxis research to my wife, my kids, and my parents. Your sacrifices, support, and confidence in my abilities have been my compass, helping me get through difficulties and pushing me to new heights.

Acknowledgments

I wish to thank Dr. Amir Etemandi for his invaluable guidance and support throughout the different phase of this Doctoral Praxis research.

Abstract

Event Type Classification and Trend Analysis of Cybersecurity News Using Dynamic Topic Modeling

In the present interconnected era, where social media, online infrastructure, cloud computing systems, and automated processes have become integral to our daily lives, technology is advancing rapidly. However, this progress brings a corresponding rise in cybersecurity incidents. Attackers are continuously adapting, employing new attack methods, tools, and strategies that enable them to breach even the most sophisticated and tightly controlled environments. This poses a greater risk of significant damage or covert operations going unnoticed. The proliferation of data and interconnected devices further expands the potential targets for such attacks. Therefore, it is crucial to allocate more resources and attention to cybersecurity, especially in vulnerable sectors.

The cost of cybersecurity attacks is expected to increase by a factor of 3.5 times over a ten-year period from 2015 to 2025. This substantial increase far surpasses the economic repercussions brought about by most natural disasters within a single year. This occurrence not only signifies the largest shift of economic resources in history but also presents a substantial threat to the driving forces behind innovation and investment.

In this praxis, we conducted research focused on developing predictive models to classify cybersecurity text corpus into different event types. Our objective was to showcase the effectiveness of cutting-edge deep learning techniques and Large Language Models (LLMs) such as Bidirectional Long Short-Term Memory (Bi-LSTM) and Bidirectional Encoder Representations from Transformers (BERT) in assisting organizations in proactively address-

ing evolving cybersecurity incidents. Additionally, we employed Dynamic Embedded Topic Models to offer cybersecurity teams insights into emerging trends by identifying latent topics and word probabilities over time.

The results from this praxis shows that the proposed models plays a vital role in assisting various important stakeholders within the cybersecurity domain, such as cybersecurity engineers and information officers. They provide valuable support by accurately classifying vast amounts of data and uncovering significant trends. This, in turn, empowers these stakeholders to make proactive decisions and implement robust cybersecurity strategies.

Table of Contents

Dedication **iv**

Acknowledgments **v**

Abstract **vi**

List of Figures **xi**

List of Tables **xii**

List of Acronyms **xiii**

Chapter 1: Introduction **1**
1.1 Background 1
1.2 Research Motivation 2
1.3 Problem Statement 3
1.4 Thesis Statement 4
1.5 Research Objectives 5
1.6 Research Questions and Hypotheses 6
1.6.1 Research Questions 6
1.6.2 Research Hypotheses 6
1.7 Scope of Research 6
1.8 Limitations of Research 7
1.9 Praxis Organization 8

Chapter 2: Literature Review **10**
2.1 Introduction 10
2.2 Cybersecurity Management 11
2.2.1 Cybersecurity Risk Management 12
2.2.2 Cybersecurity Events Management 14
2.2.3 Cybersecurity Events Dataset and Modeling Techniques 14
2.3 Natural Language Processing (NLP) for Text Classification . 16
2.3.1 Text Pre-processing 17
2.3.2 Vectorization: Term Frequency - Inverse Document Frequency (TF-IDF) 18
2.4 Supervised Machine Learning Methods for Natural Language Processing 19
2.4.1 Naïve Bayes Classifier 20
2.4.2 eXtreme Gradient Boosting (XGBoost) 21
2.4.3 Deep Neural Networks and Bidirectional Long Short-Term Memory 22
2.4.4 Bidirectional Encoder Representations from Transformers (BERT) 23

2.5 Dynamic Topic Modeling 25
2.5.1 Dynamic Topic Model in Embedding Spaces 26
2.6 Summary and Conclusion 27

Chapter 3: Methodology **29**
3.1 Introduction 29
3.2 Data Collection and Data Labeling 30
3.2.1 Data Cleaning and Labeling Consolidation 32
3.3 Exploratory Analysis, Data Pre-processing and Feature Transformation 36
3.3.1 Textual Data Preprocessing 39
3.4 Vector Representation of Texts: Word Embeddings 40
3.5 Model Development 42
3.5.1 Unsupervised Machine Learning Model: Dynamic Embedded Topic Models (D-ETM) 43
3.5.2 Supervised Machine Learning Models: Bi-LSTM and BERT 46
3.6 Conclusion 48

Chapter 4: Results **50**
4.1 Introduction 50
4.2 Model One: Deep Learning Models for Classifying Cybersecurity Event Types 51
4.2.1 Model 1A: Bi-LSTM Model for Classifying Cybersecurity Event Types 51
4.2.2 Model 1B: BERT Model for Classifying Cybersecurity Event Types 53
4.2.3 Explainability of Black Box Models 54
4.2.4 Hypothesis One 61
4.3 Model Two: Classical Machine Learning Models for Classifying Cybersecurity Event Types 63
4.3.1 Hypothesis Two 64
4.4 Model Three: Dynamic Embedded Topic Models for Identifying and Analyzing Cybersecurity Trends 67
4.4.1 Qualitative Test Case: Trend Analysis for Malware . 69
4.4.2 Hypothesis Three 70
4.5 Summary and Conclusion 71

Chapter 5: Discussion and Conclusions **73**
5.1 Discussion 73
5.2 Conclusion 73
5.3 Contributions to Body of Knowledge 75
5.4 Future Research Recommendation 76

Bibliography **78**

List of Figures

2.1 The original Transformer model architecture showing the Encoder and the Decoder. Adapted from (Vaswani et al., 2017) 24

3.1 The end-to-end process of the methodology showing the raw text extraction process, the feature transformation and data preprocessing pipeline, the model training and development pipeline, and the model evaluation and analysis section. 29
3.2 Sample of the CASIE dataset in JSON format 31
3.3 Distribution of Consolidated Cybersecurity Articles for Each Year 34
3.4 Cybersecurity Articles Counts for the Cybersecurity Event Labels 35
3.5 Box Plot representing the word counts for each cybersecurity event label 37
3.6 Box Plot representing the word counts for each cybersecurity event year 37
3.7 Unigrams and Bigrams with their top frequency 38
3.8 Sample Cybersecurity Article before Preprocessing 40
3.9 Sample Cybersecurity Article after Preprocessing 40
3.10 Sample Cybersecurity Article before Vector Representation . . . 41
3.11 Sample Cybersecurity Article after Vector Representation 42
3.12 Generative Process for D-ETM. Adapted from (Dieng et al., 2019; Dieng et al., 2020). 44
3.13 Bi-LSTM Model used in this praxis showing a 100-dimension word embeddings from the input data, the backward and forward layers, the combined dense and dropout layers, and the final softmax output layer. 47
3.14 Summary of the Bi-LSTM Model Used in this Praxis 48
3.15 Summary of the BERT Model Used in this Praxis 49

4.1 Training and Testing Accuracy and Loss for the Bi-LSTM model for Classifying Cybersecurity Articles into Event Types 52
4.2 In General machine learning, the evaluation metrics are obtained by using the predictions on the test data and the ground truth label. However, additional cybersecurity stakeholders like the cyber risk manager or chief information security officer (CISO) can demand explainability when using ML for Cybersecurity event classification. Adapted from (Lipton, 2018) 55
4.3 Published Cybersecurity News Article with an "Attack" Event Type 56
4.4 Model-Agnostic Post-hoc Local Explanation of Use Case One . . 57
4.5 Published Cybersecurity News Article with an "Discover" Event Type 58
4.6 Model-Agnostic Post-hoc Local Explanation of Use Case Two . . 58
4.7 Published Cybersecurity News Article with an "Impact" Event Type 59

4.8 Model-Agnostic Post-hoc Local Explanation of Use Case Three . 60
4.9 Published Cybersecurity News Article with an "Patch" Event Type 61
4.10 Model-Agnostic Post-hoc Local Explanation of Use Case Four . 61
4.11 Contingency Table Layout comparing the prediction performance of XGboost Model and Bi-LSTM Model in context of the McNemar's Test. 65
4.12 Image showing the time-dependent evolution of the top nine words from a "Malware" topic. The words were learned using the D-ETM algorithm. 68
4.13 The evolution of word probabilities for six distinct topics. 69

List of Tables

3.1 List and Information about the Data Sources I, II, III (Satyapanich et al., 2020; Ahmed et al., 2021; Trong et al., 2020). 33
3.2 Event Classification Label Mapping for Datasets from Data Sources I and II. The third data source (CyberED) already contains the label categories defined above. 35

4.1 Model Training Metrics of the Bi-LSTM model for Classifying Cybersecurity Articles into Event Types 52
4.2 Comparing Bi-LSTM and BERT Model Evaluation Metrics for Classifying Cybersecurity NEws Articles 54
4.3 Prediction Statistics and Results of Hypothesis Tests for Bidirectional LSTM Model Classifier for Classifying Cybersecurity Event Types 62
4.4 Machine Learning model valuation showing results for the different classical and Deep Learning Model Types. 64
4.5 Prediction Statistics and Results of Hypothesis for comparing Classical Machine Learning Algorithm (Model 1) and Deep Learning Algorithm (Model 2) for Cybersecurity Event type Classification 66
4.6 Model Performance of D-ETM on Cybersecurity News Articles. The Lower the Perplexity, the Better. Higher is Better for the Topic Quality, Topic Diversity and Topic Coherence. 67
4.7 Performance Statistics and Results of Hypothesis Test for Dynamic Embedded Topic Model for Identifying Trends from Cybersecurity News Articles with Unique Topics 70

List of Acronyms

AI Artificial Intelligence

CNN Convolutional Neural Network

BERT Bidirectional Encoder Representations from Transformers

Bi-LSTM Bidirectional Long Short-Term Memory

CBOW continuous bag-of-words

D-ETM Dynamic Embedded Topic Model

DTM Dynamic Topic Model

ETM Embedded Topic Model

GenAI Generative Artificial Intelligence

HPO Hyperparameter Optimization

k-NN k-Nearest Neighbors

LDA Latent Dirichlet Allocation

LLM Large Language Model

ML Machine Learning

NB Naive Bayes

NLP Natural Language Processing

RNN Recurrent Neural Network

SVM Support Vector Machines

TF-IDF Term Frequency - Inverse Document Frequency

XGBoost eXtreme Gradient Boosting

Chapter 1: Introduction

1.1 Background

In today's interconnected world, where social media, online infrastructure, cloud computing systems, and automated processes becomes the building block of everyday lives, technology continues to progress at an astounding pace. However, with this advancement comes a surge in cybersecurity-related events. Attackers are constantly evolving, employing new attack methods, tools, and strategies that allow them to infiltrate even the most sophisticated and tightly controlled environments, inflicting more significant damage or operating undetected (Bendovschi, 2015). The expansion of data and the proliferation of connected devices only widen the potential targets for such attacks. It is imperative to allocate increased resources and attention to cybersecurity, particularly in vulnerable sectors like healthcare.

The cost related to cybersecurity attacks is expected to rise by a factor of 3.5 times from 2015 to 2025. The scale of this growth greatly exceeds the economic consequences typically associated with most natural disasters within a single year (Hawdon, 2021). Furthermore, this trend not only marks the most significant shift of economic resources ever recorded but also poses a considerable risk to the factors that drive innovation and investment. An example is the cyberattack on Anthem Inc. in February 2015, during which cyberterrorists gained access to 80 million member and employee records, therefore revealing the weakness of the US healthcare system (Garner, 2017; Gomez et al., 2015).

In this praxis, we presented research where we explored predictive models for classifying cybersecurity text corpus into different event types and to

demonstrate the use of Large Language Model (LLM) such as Bidirectional Long Short-Term Memory (Bi-LSTM) and Bidirectional Encoder Representations from Transformers (BERT) to help organizations stay ahead of cybersecurity events as they evolve. The use of Dynamic Embedded Topic Model (D-ETM) provides cybersecurity team a view into the cybersecurity trends based on identified latent topics and word probabilities over time, which enables team to detect patterns that indicates potential threats and take the necessary mitigation strategies to alleviate these threats.

1.2 Research Motivation

Organizations face extensive difficulties in protecting their sensitive data and important information because of the rapidly evolving landscape of cybersecurity threats. For successful incident response and mitigation, early identification of cybersecurity events and trends are essential. It is often difficult for traditional rule-based strategies or knowledge-based detection systems to keep up with the advanced nature and technicalities of cybersecurity threats, necessitating the use of novel methods for classifying cybersecurity events and analyzing dynamic trends from textual data.

NLP has emerged as an important tool in various industries, offering potential applications for automatic analysis of large corpus of texts contained in cybersecurity-related documents (Sarker et al., 2021; Georgescu, 2020). By leveraging several NLP techniques, such as cyber event classification, topic modeling,and named entity recognition, it becomes possible to analyze and develop intelligence from large volume of unstructured texts from cybersecurity-related systems such as incident reports, news articles, and security alerts (Ukwen and Karabatak, 2021). This praxis focuses on using supervised NLP techniques as predictive models to classify important cy-

bersecurity events and unsupervised NLP techniques for discovering latent topics and identifying trends from cybersecurity texts.

In recent years, large language models with recurrent architectures like Bi-LSTM and transformer architectures such as BERT have emerged as the prevailing deep learning techniques for effectively analyzing and comprehending sequential data. The D-ETM is a powerful probabilistic model utilized in Natural Language Processing (NLP) to uncover hidden topics within a collection of documents. This advanced model extends the capabilities of the Latent Dirichlet Allocation (LDA) topic model by incorporating word embeddings as input, enabling the encoding of semantic relationships between texts. Furthermore, D-ETM introduces time-dependent dynamics to track the evolution of these latent topics over a specific time period. In this praxis, we showcase the effectiveness of cutting-edge large language models such as Bi-LSTM and BERT for classifying cybersecurity events, while leveraging D-ETM to extract crucial topics and identify trends over time from cybersecurity news articles.

1.3 Problem Statement

Cybersecurity incidents are left undetected and unattended for so long (Onwubiko and Ouazzane, 2020), leading to disruptions in business operations and a decline in an organization's capacity to safeguard its core activities (Paté-Cornell and Kuypers, 2021).

Effective cybersecurity incident management relies on the foundation of comprehensive planning, preparedness, and readiness. This entails establishing a collaborative framework that involves cybersecurity incident classification and facilitates information sharing and communication among stakeholders involved in managing cyber incidents (Onwubiko and Ouazzane,

2020). In recent years, there has been a substantial rise in cybersecurity attacks and a heightened focus on cybersecurity concerns. This is exemplified by notable incidents such as the Target, Home Depot, and Anthem breaches, followed by a shift towards banking fraud and ransomware attacks (Paté-Cornell and Kuypers, 2021).

The Anthem breach, initially believed to be sophisticated, was later revealed to be uncomplicated upon closer examination. It was detected when a database administrator noticed suspicious activity tied to their credentials, indicating a possible intrusion. It was later discovered that the attackers were able to exploit Anthem's inadequate protection measures such as insufficient monitoring controls (Garner, 2017). The lack of an effective automated monitoring control led to Anthem being confronted with multiple civil-class lawsuits, ultimately leading to a settlement where they agreed to pay $115 million in damages (Shankar and Mohammed, 2020; Garner, 2017). This incident highlighted the need for stronger security practices in safeguarding sensitive information.

1.4 Thesis Statement

A predictive model is required to classify cybersecurity event types and provide cybersecurity stakeholders an early and trustworthy view into cybersecurity events.

Organizations can obtain crucial information to prevent or mitigate the impact of cyber incidents by extracting timely intelligence from publicly accessible data sources, including forums, traditional news, and social media (Marinho and Holanda, 2023). This intelligence provides valuable insights into emerging threats and their potential consequences. The importance of providing stakeholders with an early and trustworthy view into cybersecurity

events lies in the proactive defense and collaborative nature of cybersecurity operations. By utilizing predictive models, organizations can effectively monitor and analyze real-time cybersecurity data, enabling stakeholders to promptly identify emerging threats and allocate resources efficiently (Naseer et al., 2021). A reliable and accurate predictive model empowers cybersecurity stakeholders to stay one step ahead of adversaries, enabling them to adapt their defense strategies and develop effective countermeasures.

1.5 Research Objectives

The primary objectives of this praxis research are to contribute to the overall goal of enhancing cybersecurity practices by exploiting the use of novel algorithms for improving cybersecurity event classification and mitigating the risks associated with evolving cyber threats from analyzing trends. The following are the detailed research objectives:

1. To develop predictive models for classifying cybersecurity text corpus into different event types.

2. To explore the use of Large Language Models (LLMs) such as Bi-LSTM and BERT for improving cybersecurity event classification.

3. To utilize D-ETM to analyze cybersecurity trends based on identified latent topics and word probabilities over time.

4. To provide cybersecurity teams with insights and patterns to detect potential threats and develop effective mitigation strategies.

5. To contribute to the field of cybersecurity by advancing knowledge and understanding of evolving cybersecurity events and trends.

1.6 Research Questions and Hypotheses

The research objectives of this study provide the foundation for the research questions and hypotheses explored in this praxis.

1.6.1 Research Questions

RQ1: How well can a deep learning model accurately classify cybersecurity news articles into events with model explanations?

RQ2: Can a deep learning model perform better than a classical tree-based ensemble model for classifying cybersecurity events?

RQ3: Can the dynamic embedded topic model identify a set of unique topics and analyze temporal-dependent cybersecurity trends from news articles?

1.6.2 Research Hypotheses

H1: An explainable predictive model can identify cybersecurity event types with 70% accuracy.

H2: The model using deep learning algorithm and tree-based ensemble learning algorithm have the same performance.

H3: A dynamic embedded topic model can identify 70% of the unique topics used to identify cybersecurity trends.

1.7 Scope of Research

The scope of the research encompasses the development and evaluation of predictive models for classifying cybersecurity event types and extracting trends from cybersecurity news articles. It includes the exploration of deep learning techniques, such as Bi-LSTM and BERT, as well as the application

of D-ETM for topic extraction and trend analysis. This research exploration consists of data collection from cybersecurity news articles, model training using both supervised and unsupervised approaches, and the testing and verification of the hypotheses associated with the models. The training dataset consists of 9,566 cyber news articles exclusively used to train the supervised models, while the testing dataset comprises 3,189 articles that are unseen by the trained model and serve as an independent evaluation set.

1.8 Limitations of Research

Over the past few years, deep learning models have exhibited exceptional performance across diverse NLP tasks. Consequently, employing deep learning for cybersecurity events classification appears to be a promising approach. Nonetheless, the remarkable achievements of deep learning models are often attributed to their training on huge amount of data (Rusk, 2016). The requirement for substantial datasets in training deep learning models is aimed at avoiding overfitting and optimizing evaluation metrics such as model testing accuracy. This research heavily relies on the availability and quality of the 12,755 cybersecurity news articles used for training and testing the models. One limitation is that the dataset used may not represent the entire spectrum of cybersecurity events, as it is limited to the sources and cybersecurity news articles included in the dataset.

Since the research mostly examines publicly accessible data, privacy, data ownership, and potential biases in the dataset are ethical issues that need to be considered. Addressing these ethical concerns and ensuring that the data utilized for developing the models is ethically sourced, handled, and compliant with applicable laws and regulations are of utmost importance.

The generalizability of the models created in this study to other specialized cybersecurity domains or sectors may be limited. The particular traits and nuances of various contexts, such as variances in the sorts of cybersecurity events, industry-specific language, or regulatory constraints, may have an impact on the effectiveness of the models. It is crucial to take into account the particular environment in which these models are used.

1.9 Praxis Organization

This praxis has five chapters, which are as follows:

In Chapter 1, the background and research motivation are presented, providing the context for the study. The research objectives, research questions, and research hypotheses are outlined to guide the praxis. The chapter concludes by discussing the scope of the research and acknowledging its limitations.

In Chapter 2, we presented a literature review on cybersecurity management, risk management, incident detection, natural language processing (NLP) techniques, supervised machine learning methods for NLP, and temporal-dependent dynamic topic modeling. The primary goal of the review is to provide a concise overview of the existing research in these areas and analyze the scholarly articles that contribute to the understanding of these topics.

Chapter 3 covers the data collection process for the cybersecurity news dataset used in this praxis, including labeling and cleaning. It explores the dataset through descriptive analysis and visualizations, and presents a feature transformation pipeline using word embeddings. The chapter also discusses the models used in the training pipeline, such as dynamic topic modeling, Bi-LSTM, and BERT, and concludes by summarizing the

methodology and its relevance to testing the research hypothesis.

Chapter 4 covers the extensive model results for classifying cybersecurity news articles, including deep learning algorithms, model interpretability technique, classical machine learning models, and the Dynamic Embedded Topic Model (D-ETM) for extracting latent topics and understanding trends.

In the final chapter, Chapter 5, we discussed the results and presented a conclusion based on examination of the results. Additionally, we extensively explored the contributions made to the current knowledge base and provide valuable suggestions for prospective future research in the domain.

Chapter 2: Literature Review

2.1 Introduction

This chapter contains a comprehensive literature review on topics related to this praxis such as cybersecurity management, cybersecurity risk management and cybersecurity incident detection, a high level introduction of supervised NLP techniques and temporal-dependent dynamic topic modeling with vector representation of texts using classical methods and in embedding spaces. The aim of this review is to present a summary of published literature related to these topics and analyze scholarly articles in these areas.

The chapter starts with a comprehensive review of cybersecurity management, including cybersecurity risk management and cybersecurity events management. Cybersecurity management is important in every aspect of any organization that depends on digital technologies. As the number of cyber threats and attacks increases, these organizations must put in real-time efforts to protect their assets, company's and customer information, and intellectual property. Furthermore, we provide an overview of the current state of cybersecurity management, including an exploration of the challenges organizations face, the impact of evolving digital technologies and threats, and the best practices for cybersecurity management.

Then the chapter continues with a detailed exploration of literature on cybersecurity events datasets and modeling techniques. Cybersecurity events are incidents that threatens the security of an organization's information system, data, and networks. The classification of cybersecurity event types is essential for the cybersecurity team of an organization to be able to

identify, evaluate, and develop measured response to cyber incidents in a fast and effective manner. This section also includes a review of existing traditional and machine learning methods used by researchers to analyze cybersecurity trends from texts, using data from blogs, tweets, academic research papers, and other social media forums.

Next, we provide a summary of current research on supervised machine learning and NLP techniques for text classifications including text preprocessing, vector representation of texts, classical Machine Learning (ML) methods, and algorithms for text classification like the Bi-LSTM and the BERT model. BERT is a large language model that was pre-trained on large amounts of data and can be fine-tuned for solving language modeling tasks such as text classification and named entity recognition amongst others.

The chapter also presented a review of the current research works on unsupervised machine learning methods with emphasis on Topic Modeling. It examines dynamic topic models, which are probabilistic models used in analyzing trends and discovering latent topics contained in textual documents with a temporal dependence. We explored the theoretical background of temporal based dynamic topic model in embedding spaces to solve the problem of the inability of classical topic models to learn latent topics that are interpretable.

Finally, the chapter wraps up by presenting a brief overview of the literature's findings and the prospects of the current study in this praxis.

2.2 Cybersecurity Management

Modern enterprises must manage cybersecurity because they depend more and more on digital technologies. The management of risk, access control, data protection, and incident response are the main tenets of cyber-

security (Hughes and Cybenko, 2013). Regular security audits, the use of secure passwords and encryption, the installation of firewalls and intrusion detection systems, employee training and awareness programs, and regular security assessments are all examples of cybersecurity management strategies (Collier et al., 2014). Yet, many companies still struggle to manage cybersecurity risks due to continually changing threats, a lack of qualified personnel, and growing IT infrastructure complexity. Organizations may manage their cybersecurity risks and safeguard their digital assets by putting best practices into practice and remaining current with new technologies and threat information (Meszaros and Buchalcevova, 2017).

2.2.1 Cybersecurity Risk Management

Information security management must effectively manage cybersecurity risk. It entails locating, evaluating, and reducing threats to the privacy, accuracy, and accessibility of data and information systems (Eling et al., 2021). Onwubiko et al offered a thorough methodology for managing cybersecurity incidents (Onwubiko and Ouazzane, 2020). The researchers used equivalence mapping in their framework to compile the terminology of cybersecurity management into a single, uniform lexicon. Their study revealed inadequacies in the field of cybersecurity risk management. However, a limiting factor is that standardization might make it challenging to reach agreement on some lexicons, particularly when adopting them in several geographical places (Onwubiko and Ouazzane, 2020).

The fact that threats and technologies are always changing presents one of the biggest challenges businesses confront when managing cybersecurity risks. Organizations struggle to keep up with the constantly evolving methods that cyber-criminals use to access systems and steal critical data.

Additionally, as cloud computing and Internet of Things (IoT) devices are used more often, new attack mediums are created that need to be secured by enterprises. The lack of skilled cybersecurity experts is another issue that should be considered. There is a significant need for experienced cybersecurity engineers and researchers in this industry, yet there are few of them available. Organizations may find it challenging to locate and hire the personnel they require to handle their cybersecurity threats (Eling et al., 2021; Onwubiko and Ouazzane, 2020). Organizations must keep up with the newest technology and threat information to handle these issues and make sure their cybersecurity risk management methods are successful. To remain ahead of changing threats, they must also routinely evaluate their security posture and make any necessary adjustments (Eling et al., 2021; Onwubiko and Ouazzane, 2020).

Businesses can increase their cybersecurity by taking a more methodical and analytical approach. The likelihood of new attack scenarios can be calculated using a probabilistic method based on historical data. Pate-Cornell et al article's objective is to estimate the risk based on prior cyber incidents and utilize that information to create a scenario analysis of possibly significant current and future occurrences. For the purpose of managing cyber risk, statistical and probabilistic methodologies are trustworthy models for quantitative risk assessments. Their models quantifies cyber risk beyond heuristics, employing existing data to introduce a probabilistic approach and extending it to evaluate the probabilities of novel attacks (Paté-Cornell and Kuypers, 2021).

2.2.2 Cybersecurity Events Management

Data breaches, ransomware, and cyber-attacks on vital infrastructure take place everyday and they have made cybersecurity a problem for governments, companies, and individuals. The current demand is for cybersecurity solutions that can handle a lot of textual data, patterns, and trend identification (Ghasiya and Okamura, 2022). In order to get information about the rapidly increasing number of cybersecurity events identified in texts, cybersecurity event type classification is required. This task have been used to find event trigger phrases in the cybersecurity domain (Trong et al., 2020).

Because the techniques used in cybersecurity attacks are evolving and firms are at danger from hidden weaknesses, cybersecurity events have grown more complex. Computer emergency response teams (CERTs) and security operation centers (SOCs) must assess huge volume of security alerts produced by security systems and devices and those recorded in vulnerability databases in order to lessen the effects and damages brought on by such incidents (Ishii et al., 2020). Due to the fact that they are dependent on rigid rules or signatures, security information and event management systems are unable to detect complex and unforeseen assaults on their own (Ishii et al., 2020).

2.2.3 Cybersecurity Events Dataset and Modeling Techniques

Many datasets have been introduced for predicting cyber-related incidents. Ghasiya and Okamura used the BeautifulSoup Python package to scrape 18 English-language newspapers' websites between April 2018 and March 2019 to gather news stories about cybersecurity (Ghasiya and Oka-

mura, 2021; Ghasiya and Okamura, 2022). They gathered 4159 articles, with the United States having the most (1615), and South Korea having the fewest (87). The Non-negative Matrix Factorization (NMF) topic modeling method was used in this extensive study, to examine hundreds of cybersecurity news stories from six nations and 18 major publications (Ghasiya and Okamura, 2021; Ghasiya and Okamura, 2022; Sra and Dhillon, 2005).

In the cybersecurity events prediction research by Ishii et al, they employed 405 security texts from both internal and external sources: 45 security event reports that their research team kept track of from January 2016 to March 2019 and 120 blog entries on the Japanese-language Trend Micro Security Blog. To evaluate and categorize Japanese text, they used topic models based on LDA. They also showed that using a seeded topic model can yield more precise results than a topic model in a supervised settings(Ishii et al., 2020).

Trong et al developed deep learning models using a new dataset they created for event prediction. To achieve this, they characterized manual annotation for 30 key cybersecurity event types. For event prediction, they used two classes of models: sentence-level models (models that solely use the local context information in the triggers' phrases) and document-level models (i.e., the models that further considers extracting contextual linguistic information from document-level corpus) (Trong et al., 2020). Ravi et al suggested a framework and strong mechanism based on deep learning to stop cybercriminals from evading detection and to resist adversarial learning.Their suggested framework employs cutting-edge adversarial defensive techniques and is resilient and robust. They were able to demonstrate that their method performs better with a higher detection rate than several typical architectures. Their research emphasizes the effectiveness of robust

and sophisticated black-box detection methods in thwarting adversarial learning (Ravi et al., 2021). However, their methods lacks explainability because of the complexity of the model framework.

In order to comprehend how the latent topics in documents are evolving over time, Sleeman et al provided insights into the use of dynamic topic models for analyzing documents in the cybersecurity domain (Sleeman et al., 2021). They showed a method for finding latent topics that are easier for people to understand by using an ontology of cybersecurity ideas taken from Wikipedia. The output of the dynamic topic model is represented as a knowledge graph, which was utilized for inference and information search. The limitation of their research is that the trend analysis is done using published academic paper and Wikipedia articles that are not real-time and very domain-specific. Also, Dynamic Latent Dirichlet Allocation (D-LDA) used in this research does not offer a better qualitative performance (topic perplexity, quality, coherence and diversity) than Dynamic Embedded Topic Models (D-ETM) (Blei and Lafferty, 2006; Dieng et al., 2019).

2.3 Natural Language Processing (NLP) for Text Classification

NLP involves the processing and analysis of speech and textual data (Amini et al., 2019). One of the core NLP tasks is text classification, which is categorizing or labeling text documents based on their content. There are many ways to classify texts, and we will cover some of the widely used NLP techniques for text classification such as text pre-processing methods and vectorization techniques (Hapke et al., 2019).

2.3.1 Text Pre-processing

Text pre-processing is a foundational stage in natural language processing that entails converting unprocessed textual data into a machine-readable format that machine learning algorithms can more readily understand and evaluate. The following are often included in the pre-processing steps (Anandarajan et al., 2019; Hapke et al., 2019):

- Tokenization: This step involves breaking the text into separate words or tokens. This can be done using whitespace or punctuation as delimiters, or using more sophisticated NLP libraries like NLTK to identify named entities or multi-word expressions.

- Lowercasing: To limit the amount of unique words the model needs to learn, this stage entails changing all words to lowercase. This can lessen the influence of spelling changes and increase the model's accuracy.

- Stopword removal: Stop words are often used words like "the," "a," "is," "are," "and," "in," etc. that have little meaning and can be eliminated without harming the text's overall meaning.

- Stemming and Lemmatization: They are text pre-processing methods used in NLP to break down words to their root or base form. Stemming is the process of stripping words to reveal their original word form. The stem of words like "sleeping," "sleeps," and "sleep" is "sleep," for instance. Lemmatization is a more complex process that includes mapping words to their dictionary equivalents. In order to ascertain a word's base form, one must take into account both its context and its part of speech. The lemma of "far", "farther", and "farthest" would

all be "far", for instance. The base form of a word is produced through lemmatization, however it is often slower and more computationally expensive than stemming.

- Removal of special characters and punctuations: This stage entails deleting special symbols, emojis, and other non-alphanumeric characters that might not add to the text's meaning. It also includes the removal of email addresses, URLs, and other particular forms of information that might not add to the text's general meaning.

2.3.2 Vectorization: Term Frequency - Inverse Document Frequency (TF-IDF)

The Term Frequency - Inverse Document Frequency (TF-IDF) weighing method is a method for vectorizing texts. This algorithm is effective, and its implementation is simple (Ullman, 2011; Hapke et al., 2019). For a given term/word, x, that appears in a document/sentence, D, in a corpus, C, the term frequency (TF), the inverse document frequency (IDF) and TF-IDF is given as (Silge and Robinson, 2017):

$$\text{TF}(x,D) = \frac{\text{Count of the occurrences of the word "x" in the given sentence}}{\text{Count of the total words in the given sentence}} = \frac{n_x}{n_{xD}} \tag{2.1}$$

$$\text{IDF}(x,D,C) = \log_e\left(\frac{\text{Count of the total number of sentences}}{\text{Count of sentences with word, x, in it}}\right) = \log_e\left(\frac{N_D}{N_{xD}^C}\right) \tag{2.2}$$

$$\text{TF-IDF}(x,D,C) = \text{TF}(x,D) \times \text{IDF}(x,D,C) = \frac{n_x}{n_{xD}} \times \log_e\left(\frac{N_D}{N_{xD}^C}\right) \tag{2.3}$$

Therefore, the higher the frequency of a word in a sentence, the more likely the TF would go up. Similarly, as the number of sentences that contain the word goes up, the IDF (and hence the TF-IDF) goes down. However, the limitation of TF-IDF is that it computes sentence similarity directly in the word-count space and this process can be slow for large vocabularies.

2.4 Supervised Machine Learning Methods for Natural Language Processing

NLP has advanced significantly over time thanks to supervised machine learning. In this area, machine learning models are trained to identify patterns in textual data using supervised learning, which enables them to understand and interpret natural language. The first supervised machine learning applications in NLP used rule-based techniques that depended on manually created linguistic rules. The intricacy and variety of real language, however, presented a challenge for these approaches, prompting academics to investigate statistical models (Ullman, 2011; Hapke et al., 2019).

The use of Naive Bayes classifiers for text categorization tasks like spam filtering was one of the first widely adopted successful applications of supervised machine learning in NLP. These classifiers categorize new text using probabilistic models based on patterns discovered from labeled training data (Ullman, 2011; Hapke et al., 2019). Support Vector Machines (SVM) became more popular for text classification and other NLP applications around the middle of the 2000s (Hearst et al., 1998). By maximizing the margin between classes, SVM, a form of supervised learning algorithm, can learn complex decision boundaries.

Convolutional neural networks and recurrent neural networks, two examples of deep learning techniques, have recently transformed the NLP

domain (Patel and Patel, 2021). On a variety of tasks, including text classifications, sentiment analysis, machine translation, and named entity recognition, these models have produced cutting-edge results. Overall, supervised machine learning has been an essential tool for developing NLP, enabling scientists, researchers, and engineers to create models that can comprehend and analyze languages in a way that was previously impractical (Goodfellow et al., 2016).

2.4.1 Naïve Bayes Classifier

The Naïve Bayes method is a popular probabilistic machine learning approach for text classification applications including spam detection, sentiment analysis, and document categorization. It relies on the principles of Bayes theorem, assuming that the attributes of the input data are conditionally independent, given the class label (Berrar, 2018). In a typical classification scenario, the Naïve Bayes algorithm assesses the likelihood of each class label given the input features, subsequently opting for the class with the highest probability as the predicted label. (Rish et al., 2001). The Naïve Bayes algorithm is a simple and intuitive model, with decent computational efficiency and baseline accuracy. The underlying principle for a Naïve Bayes classifier is the Bayes theorem:

$$P(L|\mathbf{X}) = P(L)\frac{P(\mathbf{X}|L)}{P(\mathbf{X})} \tag{2.4}$$

where L is the label representing the classification event types; Attack, Discover, Impact, Patch. $\mathbf{X}$ is the extracted feature from the text using the TF-IDF vectorization, P(L$|\mathbf{X}$) is the posterior probability, P($\mathbf{X}$) is the evidence, P($\mathbf{X}$ |L) is the likelihood and P(L) is the prior probability. Therefore, given predictor variables, $\mathbf{X} = \{x_1, x_2, x_3, ..., x_n\}$, we would like to define the posterior

probability among a set of possible event types labels, $L = \{l_1, l_2, l_3, l_4\}$ using Bayes rule. Due to its reliance on the assumption of statistical independence among predictor variables, Naïve Bayes allows us to rephrase the likelihood and posterior as follows:

$$P(\mathbf{X}|l_i) \propto \prod_{k=1}^{n} P(x_k, |l_i) \tag{2.5}$$

$$P(l_j|\mathbf{X}) \propto P(l_j) \times \prod_{k=1}^{n} P(x_k, |l_i) \tag{2.6}$$

Calculating $P(x_k \mid l_i)$ depends on the distribution of the feature variables. If the features are word counts, they will be a multinomial distribution. For continuous variables like TF-IDF vectors, it will be a Gaussian distribution. Naïve Bayes also performs well with high dimensional data points because it uses a deterministic approach to efficiently find the parameters for the probability distribution of the feature variables.

2.4.2 eXtreme Gradient Boosting (XGBoost)

eXtreme Gradient Boosting (XGBoost) is an algorithm utilized for machine learning applications like regression and classification (Chen and Guestrin, 2015). Since its initial introduction by Chen and Guestrin in 2015, it has gained widespread acceptance in both academic, research, and professional settings. XGBoost is a tree-based ensemble model that combines the predictions of various decision trees (weak learners) to compute the most likely prediction (strong learner). The approach makes use of a gradient boosting framework, adding new trees to the model iteratively, each tree fixing mistakes produced by the one before it (Chen and Guestrin, 2016). As a result, the model becomes more reliable and accurate, and have been used in many machine learning competitions over the years.

The handling of missing values, which can be a prevalent problem in real-world datasets, is one of XGBoost's important characteristics (Aydin and Ozturk, 2021). The method can function with a variety of datasets because it can handle both categorical and numerical data. L1 and L2 regularization, which impose penalties on the model's parameters, are two regularization methods included in XGBoost to avoid overfitting (Shilong et al., 2021). By doing so, the model becomes simpler and is less likely to fit the data noise (Wang and Lu, 2020; Bhattacharya et al., 2020).

2.4.3 Deep Neural Networks and Bidirectional Long Short-Term Memory

Neural networks consist of interconnected nodes or neurons that collaborate to identify patterns and draw inferences from input data (Zou et al., 2009). The neuron serves as the fundamental building block of a neural network, receiving input from other neurons and producing an output based on continually updated learning biases and weights (Goodfellow et al., 2016). The weights are multiplied by the input to the neuron, and the total of these weighted inputs is then sent via an activation function, which gives the model non-linearity. The output of the activation function serves as output of the neuron, which can be passed on to other neurons in the network.

Deep neural networks (DNNs) contains multiple layers, with each layer made up of a collection of neurons processing the incoming data. The output layer generates the final predictions after the input layer gets the raw data, such as an image or a list of words. The hidden layers extract increasingly complex and abstract properties from the incoming data. To minimize the difference between the expected output and actual output, a neural network undergoes training via backpropagation. During this process, the network's

weights and biases are iteratively adjusted and updated for the neurons (Goodfellow et al., 2016).

Numerous types of neural networks exist, each of which is designed to solve a specific type of problem. Convolutional Neural Network (CNN) are frequently used for image classification tasks, whereas Recurrent Neural Network (RNN) are used for sequence prediction tasks like natural language processing (Li et al., 2021; Gu et al., 2018; Medsker and Jain, 2001; Mikolov et al., 2010). Many machine learning tasks have used neural networks to achieve cutting-edge performance. However, training them can be computationally expensive, and achieving satisfactory performance demands substantial amounts of data. To mitigate overfitting and achieve robust generalization performance, Hyperparameter Optimization (HPO) is also employed (Goodfellow et al., 2016).

The Bi-LSTM is a type of RNN commonly used for handling sequential data tasks like NLP tasks, speech recognition, and time series prediction (Liu and Guo, 2019). Unlike a standard LSTM, which processes the input sequence in a forward manner, with each time step relying solely on the previous one, the Bi-LSTM processes the sequence in both forward and backward directions (Xu et al., 2019). This unique approach allows the model to capture meaningful relationships and dependencies between past and future context, making it particularly effective for tasks involving sequential data.

2.4.4 Bidirectional Encoder Representations from Transformers (BERT)

BERT is a pre-trained model used for a variety of NLP applications, including text similarity, sentiment analysis, text classification, and question-

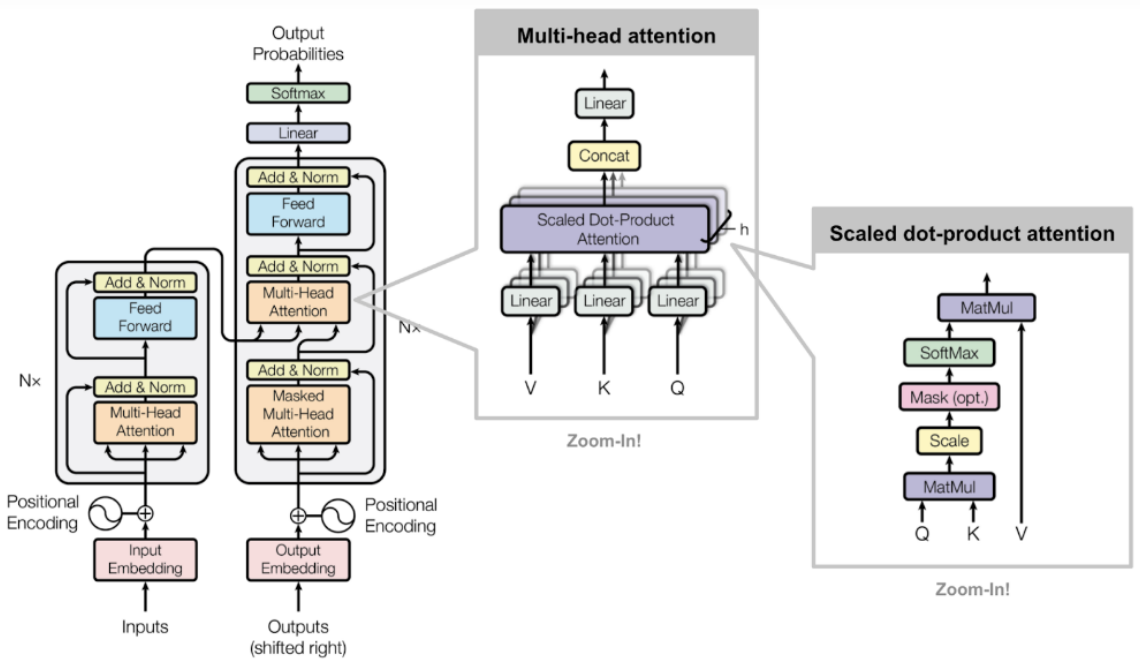

Figure 2.1: The original Transformer model architecture showing the Encoder and the Decoder. Adapted from (Vaswani et al., 2017)

answering. BERT model captures word context and meaning by training on an extensive amount of unlabeled text with hundreds of millions of parameters (Devlin et al., 2018). Vaswani et al. proposed the Transformer architecture in 2017 and it serves as the foundation for the BERT model architecture (Vaswani et al., 2017). The Transformer design makes use of the self-attention method to let the model concentrate on various segments of the input sequence. BERT has a two-stage training procedure in which it is initially adjusted for particular downstream tasks using a large corpus of unlabeled text and a masked language modeling job.

BERT is trained by utilizing two tasks during the pre-training phase: masked language modeling and next sentence prediction. A set fraction of the input tokens are randomly masked in the masked language modeling phase, and the model is trained to predict the masked tokens. The model is then trained to determine if two sentences are sequential in the source text

when performing the next sentence prediction task. BERT can be fine-tuned for different NLP tasks after pre-training. In fine-tuning, the pre-trained model's last layer is swapped with a task-specific layer, and training occurs on a labeled dataset designed for that particular task. (Devlin et al., 2018).

BERT has undergone numerous developments and enhancements since it was first introduced. The debut of RoBERTa by Facebook Artificial Intelligence (AI) in 2019 was one of the major developments (Liu et al., 2019; Briskilal and Subalalitha, 2022). RoBERTa performed better than BERT because it used a larger pre-training corpus and more training time. ALBERT (A Lite BERT), which was released in 2019, was another development (Lan et al., 2019). It used parameter sharing techniques to minimize the model's parameters, making it more memory-efficient and quick.

The significant computational cost needed for pre-training and fine-tuning the model is one of the major drawbacks of BERT and most LLMs (Rogers et al., 2021). For fine-tuning, BERT also needs a lot of labelled data, which might not be available for all use cases. BERT might also struggle on tasks that call for knowledge outside of the linguistic corpus utilized for pre-training (Rogers et al., 2021). BERT has emerged as a important modeling tool in the field of NLP, enabling researchers and developers to complete a variety of tasks with state-of-the-art outcomes. Further increases in the performance and efficacy of BERT models can be anticipated thanks to robust developments in model architecture and efficient training techniques (Pati et al., 2022).

2.5 Dynamic Topic Modeling

Dynamic Topic Model (DTM) are probabilistic models that have been used in analyzing and discovering latent topics over time in textual data (Blei and Lafferty, 2006). DTMs are used to model evolving topics in documents over

time and find applications in diverse areas, including COVID-19 Twitter narratives, financial market competition, and urban geo-topics (Sha et al., 2020; Greene and Cross, 2017; Yao and Wang, 2020).

Dynamic Topic Model was first applied to understand the temporal evolution of topics in news articles (Blei and Lafferty, 2006). This DTM extends the Latent Dirichlet Allocation (LDA) model which doesn't have any temporal dependence, by adding a temporal dimension for modeling the evolution of the latent topics over time. With qualitative experimental results, the authors showed how DTM can accurately track the evolution of latent topics in large, unstructured news articles over time (Blei and Lafferty, 2006).

With the proliferation of financial documents and the need to have a structured understanding of digitized documents, an important application of DTM are in the analysis of of financial documents. Research have been done to track the evolution in the languages used in financial news articles, with an aim to correlate them with the performance and changes in the activity of the financial market (Morimoto and Kawasaki, 2017; Zhu et al., 2022). Experimental results have shown that using dynamic topic models for monitoring the evolution of financial market news and social media activities can significantly improve market activity prediction models (Wang et al., 2012). For documents with multiple time scales, continuous-time dynamic topics models have been developed to identifying rapidly changing topics (Wang et al., 2012).

2.5.1 Dynamic Topic Model in Embedding Spaces

Topic models, like Latent Dirichlet Allocation (LDA) (Blei and Lafferty, 2006), are used to discover latent topics in a document set. LDA assumes that each document contains a distribution over latent topics, which, in

turn, are characterized by a distribution over words. Despite being a powerful and ubiquitous model, LDA has many shortcomings. In LDA, the Dirichlet topic distribution cannot capture the correlation that exists between topics. LDA is unable to model the time evolution of topics and the topic quality diminishes with very large datasets (Chi et al., 2018). The bag of words, which assumes that the words are interchangeable, doesn't take the structure of the sentence into consideration. To solve the problem of the inability of LDA to learn interpretable topics from large amounts of data, the Embedded Topic Model (ETM) was introduced (Dieng et al., 2019). D-ETM is a method that combines Dynamic Topic Models (DTM) with word embeddings as inputs to learn latent topics and produce word probabilities of these topics over time (Dieng et al., 2020).

2.6 Summary and Conclusion

With a focus on modern NLP techniques and unsupervised dynamic topic modeling, this literature review section provided a thorough analysis of journal articles, conference papers, and books on cybersecurity management, cybersecurity risk management, cybersecurity events management, and the datasets and techniques using for classifying cybersecurity events and their trends. According to the literature analysis, despite having access to current cybersecurity technologies, organizations still have significant issues with understanding the evolution of topics in cybersecurity trends and having highly precise and accurate models for event type classifications. NLP methods can be used to classify and understand trends in cybersecurity texts, but the approaches that have been discovered by the literature review are either limited by the dataset or uses methods with lower quantitative performance. The newly developed Dynamic Embedded Topic (D-ETM) and

state-of-the-art large language models like BERT will be used in this praxis to develop a more accurate classification model and identify interpretable latent topics and trends from cybersecurity news.

Chapter 3: Methodology

3.1 Introduction

The goal of the study is to develop and implement supervised and unsupervised machine learning algorithms for classifying cybersecurity events and identifying cybersecurity trends from published news respectively. To achieve this, we used many data pre-processing steps and vector representation of texts as input for developing the machine learning models. Figure 3.1 is a diagram showing the end-to-end process of the methodology used in this praxis from how the cybersecurity data is collected and processed to how the models are developed and evaluated to analyze trends and classify the cybersecurity events.

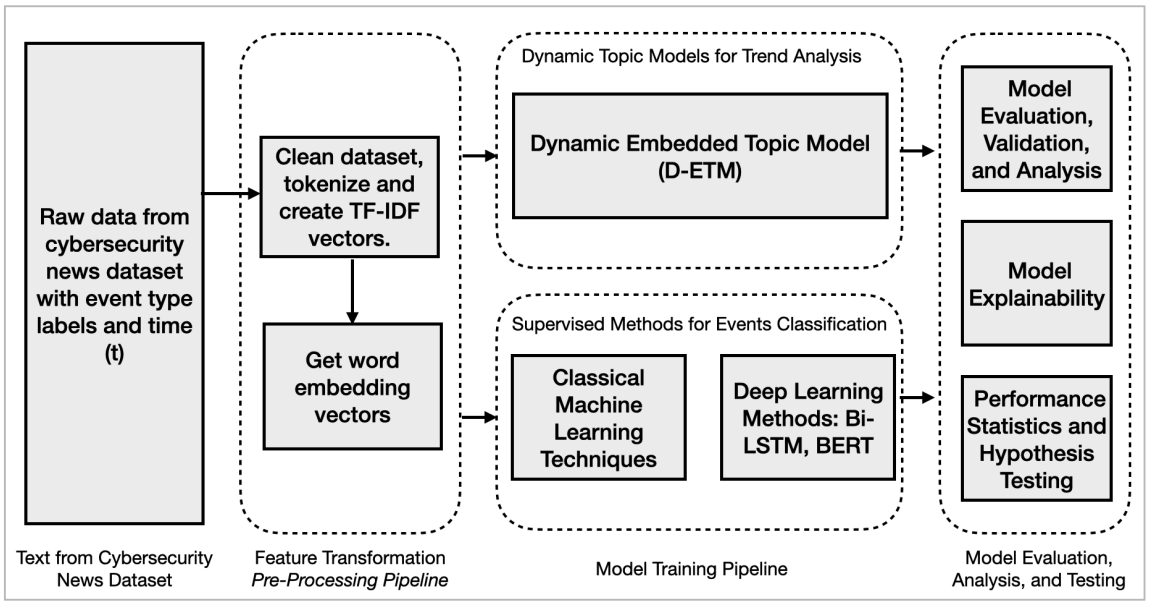

Figure 3.1: The end-to-end process of the methodology showing the raw text extraction process, the feature transformation and data pre-processing pipeline, the model training and development pipeline, and the model evaluation and analysis section.

The chapter starts with a discussion of the data collection process of raw cybersecurity news dataset with labels representing the different type of events and the year the article was published. Next, we described the consolidation, cleaning, and concatenation process involved in taking the data from the raw state to a processed dataframe that can be used for modeling. After this, the chapter continues with the descriptive and exploratory analysis of the dataset including the word counts and box plots showing additional statistical information. Next, our feature transformation pipeline is described. This pipeline takes the dataframe, tokenize it and creates vector representations of the input texts using word embeddings. These embeddings act as vector representations that are learned from textual data by capturing semantic similarities between words. This groundbreaking approach to representing words and documents has been instrumental in enabling the successful application of deep learning to complex NLP tasks. Next, the chapter explored the development and intuition around the models used in the training pipeline, specifically the dynamic topic modeling in embedding spaces for cybersecurity trend analysis and the deep learning methods such as the Bi-LSTM network and the BERT model. Finally, the section concludes by summarizing the methodology and explaining how each components of the proposed analysis methodology are important for testing the hypothesis in this praxis.

3.2 Data Collection and Data Labeling

The data utilized in this study originates from three main sources, comprising cybersecurity news articles along with their associated metadata. This metadata encompasses crucial information such as publication dates extracted from the news content and labels assigned through manual anno-

tation.

The first dataset is from the CyberAttack Sensing and Information Extraction (CASIE) publication, where the authors extracted details about cybersecurity events from texts for use with a semantic model (Satyapanich et al., 2020). It contains a corpus of 1,000 English news items from 2017 to 2019 that include both cyber-attack and vulnerability-related incidents with extensive and event-based annotations. The annotated CASIE datasets are in json format and contains the content, the cyberevents and other relevant information like the date, labels, and sub-labels.

Figure 3.2: Sample of the CASIE dataset in JSON format

The second main data source, curated by academic researchers, comprises a collection of cybersecurity news articles obtained from "The Hacker News" website. A comprehensive total of 3742 news articles was gathered and appropriately labeled (Ahmed et al., 2021). The news categories were classified by consulting with experts and reaching a consensus regarding

various types of cyber threats. The data encompasses crucial details such as the article title, the corresponding link, the complete article content, and the associated labels.

The third primary source utilized in this study is the Cybersecurity Event Detection (CyberED) dataset, which offers a valuable resource for identifying event trigger phrases specifically within the cybersecurity domain (Trong et al., 2020). The authors of this dataset have created human annotations for 30 significant cybersecurity event types, surpassing the number of event types found in previous datasets. This expanded coverage allows for improved modeling of document-level information, ultimately enhancing performance in the field. The CyberED dataset consists of a vast collection of 8014 articles pertaining to cybersecurity events, accompanied by important metadata including labels, trigger words, and the respective publication year.

3.2.1 Data Cleaning and Labeling Consolidation

In this study, we utilized datasets from the three different sources described in the previous subsection. First, for the CASIE dataset, we extracted individual files from a JSON file. We transformed each JSON file into a dataframe, added it to a list, and finally concatenated all the dataframes together. The resulting dataframe contained details such as contents, cyberevent, sourcefile, and info. To extract the year and labels, we performed string manipulation using the Python programming language on the cyberevent and date fields.

Next, the Mendeley dataset was in Excel format, which made it easy to convert into a dataframe. We simply transformed it without any complications. Lastly, for the CyberED dataset, we converted it from JSON format

Name	Source	Format	Time Span	Number of Records
Cybersecurity News Dataset I	CyberAttack Sensing and Information Extraction (CASIE)	JSON Format	2017 - 2018	1000
Cybersecurity News Dataset II	Curated cybersecurity data on Mendeley from "The Hacker News" website	Excel File	2013 - 2021	3742
Cybersecurity News Dataset III	Cybersecurity Event Detection (CyberED) dataset	JSON Format	2010 - 2019	8014

Table 3.1: List and Information about the Data Sources I, II, III (Satyapanich et al., 2020; Ahmed et al., 2021; Trong et al., 2020).

into dataframes. We detokenized the sentences and extracted the year of publication from the "id" field using string manipulation techniques. To combine all three datasets, we focused on three important features: article texts, year of publication, and labels. The resulting dataset, after concatenation and cleaning, is represented in Figure 3.3. It's important to note that the distribution of article counts in the dataset is not uniform. This variation is due to the differing number of articles from each data source and the years they cover, as shown in Table 3.1.

To define our labels, we used cybersecurity event taxonomy defined by Simmons et al. (2009) and Trong et al. (2020) (Trong et al., 2020; Simmons et al., 2009). The cybersecurity events categories used are frequent and have a significant influence in the cybersecurity management community. The categories and definition of the categories are as follows:

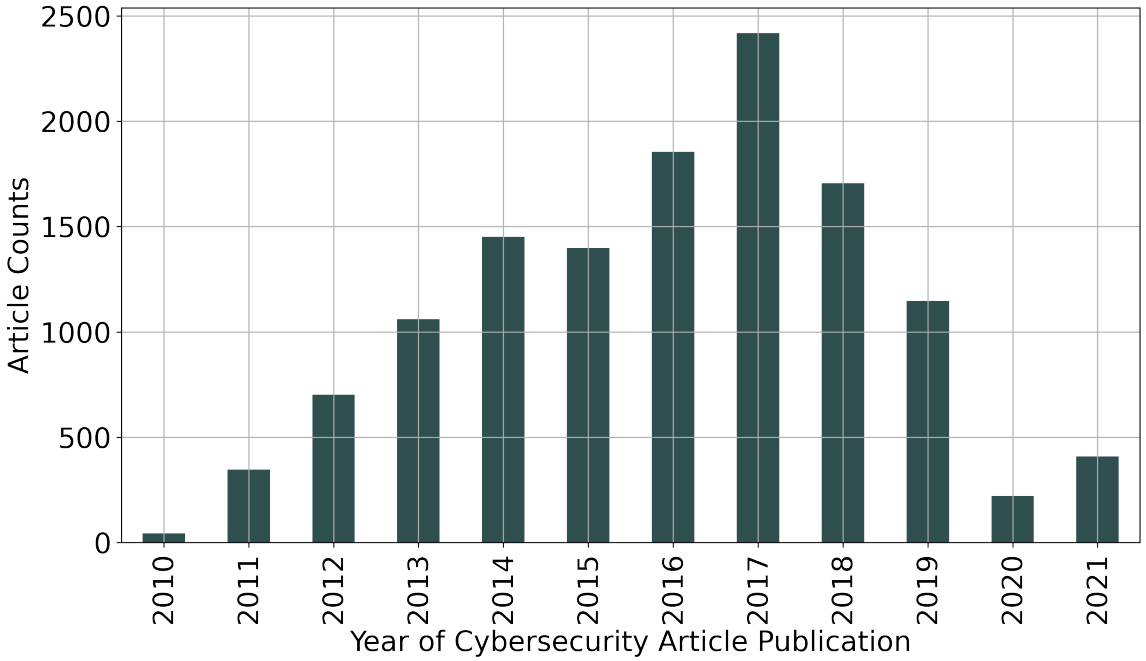

Figure 3.3: Distribution of Consolidated Cybersecurity Articles for Each Year

- Attack: This is a type of cybersecurity event that occurs when an attacker leverages a vulnerability to impact the systems in some way (such as user compromise, viruses, spyware, worms, or denials of service).

- Discover: This is a cybersecurity event that occurs when a cybersecurity stakeholder (e.g. hackers, engineers, cybersecurity analysts) notices or makes reference to a vulnerability in a piece of software or a system (e.g., kernel leakage, buffer overflow, back door)

- Impact: This is a type of cybersecurity event that documents or explores what happens to the system as a result of the assault (such as disruption or information breach).

- Patch: This is a type of cybersecurity event that occurs when a known vulnerability is fixed or instructions are provided by some party (such

Cybersecurity News Dataset I (CASIE)		Cybersecurity News Dataset II (Mendeley Data)	
Original Label	**Converted Label**	**Original Label**	**Converted Label**
Databreach	Attack	Vulnerability	Discover
DiscoverVulnerability	Discover	Malware	Attack
Phishing	Attack	Cyber_Attack	Attack
Ransom	Attack	Data_Breaches	Attack
PatchVulnerability	Patch	–	–

Table 3.2: Event Classification Label Mapping for Datasets from Data Sources I and II. The third data source (CyberED) already contains the label categories defined above.

as software firms).

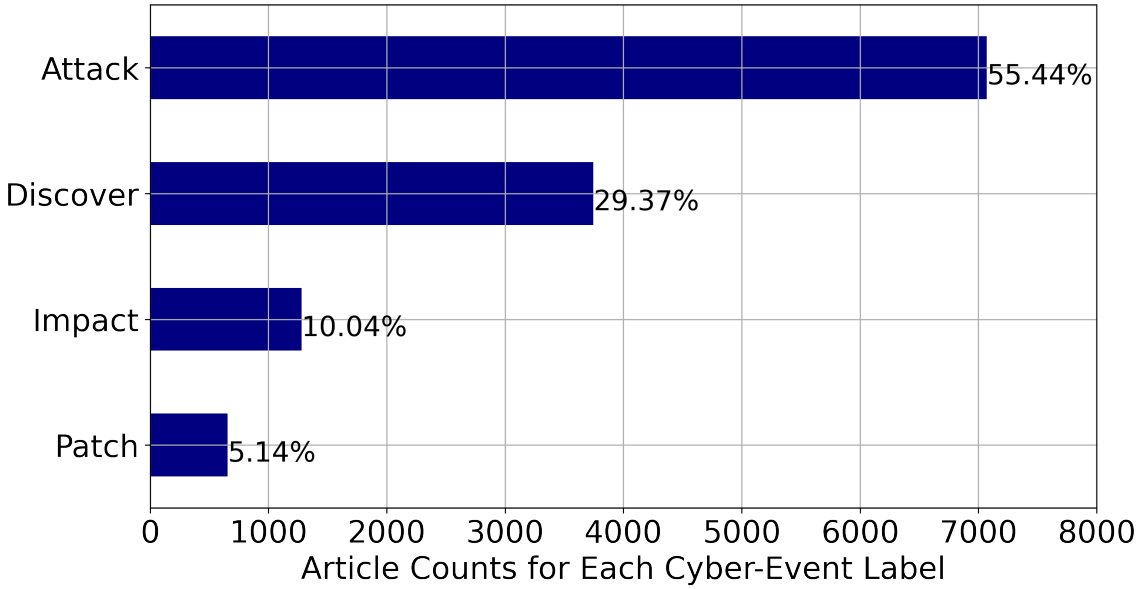

Figure 3.4: Cybersecurity Articles Counts for the Cybersecurity Event Labels

To define the labels that will be used for the cybersecurity events classification models, we mapped the labels defined in the datasets from Data Sources I and II to the labels defined above. Five different labels from the CASIE were mapped to three of the event labels we defined in this section. The four labels in the Mendeley Data were mapped and converted to two of

the event labels defined. Table 3.2 shows the event label mappings for the datasets from the two data sources. The third dataset (CyberED) contains the defined labels above. Figure 3.4 shows the count of the cybersecurity articles for each label after cleaning and mapping. The distribution shows that there is class imbalance from the labels. This imbalance occurs when there are disproportionately more or fewer occurrences of one class (label) than another in a dataset.

3.3 Exploratory Analysis, Data Pre-processing and Feature Transformation

To identify patterns, trends, and relationships that might be present in the textual data, exploratory analysis entails evaluating and comprehending the texts (Eisenstein, 2019). It is an important phase in the analysis of data since it aids in identifying potential issues, outliers, and anomalies in the data that can then guide more in-depth investigation and modeling. Gaining a deeper knowledge of the data and producing insights that can help inform decision-making are the objectives of exploratory analysis. In order to find trends and patterns, this method frequently requires displaying the data using charts, graphs, and other visual representations. The Figure 3.5 and 3.6 are box plots with the distribution of the word counts for each cybersecurity event labels and year. The first quartile (Q1), median (Q2), third quartile (Q3), minimum value, and maximum value are the five summary statistics used to illustrate the distribution of the data in a box plot. The whiskers, shown as lines extending from the box, display the minimum and maximum values of the word counts from the texts. In some instances, outliers are seen as extra dots or asterisks.

Box plots can be used to determine a dataset's central tendency, distri-

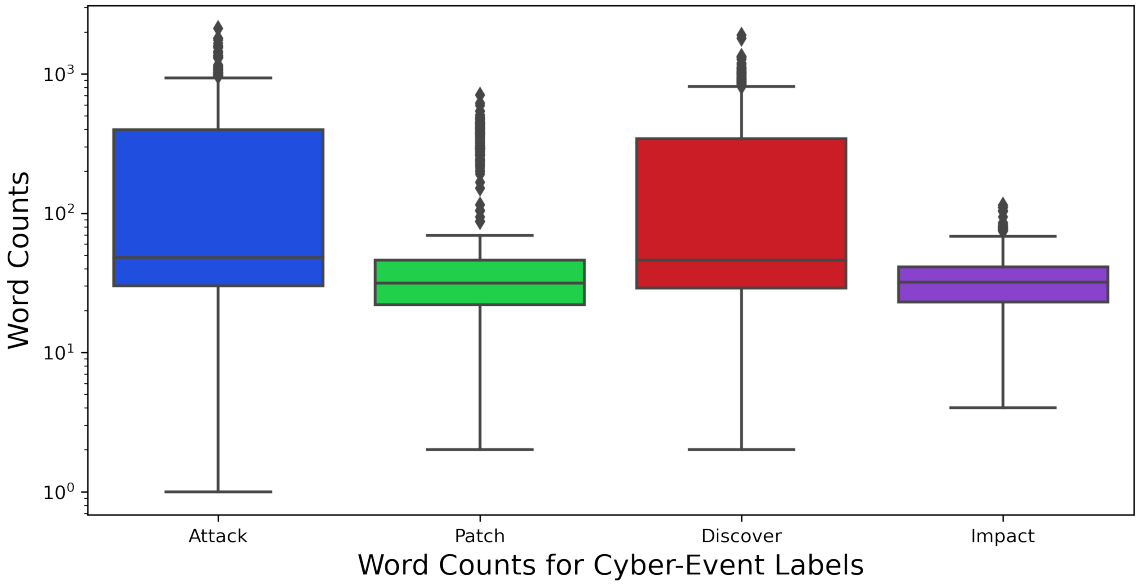

Figure 3.5: Box Plot representing the word counts for each cybersecurity event label

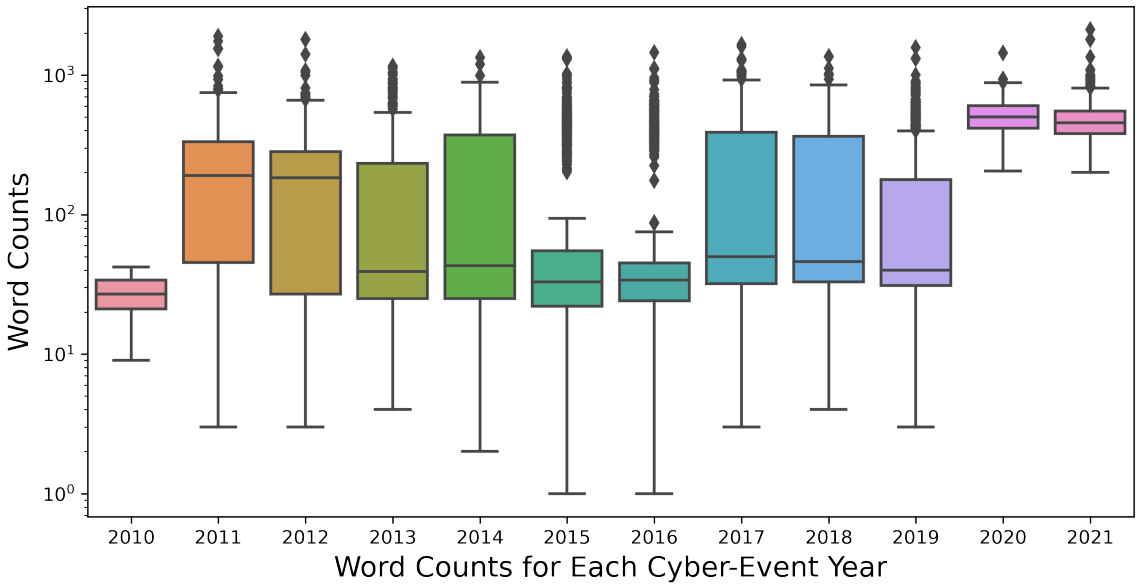

Figure 3.6: Box Plot representing the word counts for each cybersecurity event year

bution, and probable outliers. When comparing the distributions of several datasets or analyzing the distribution of a single dataset with a lot of values, they are especially helpful. The "Patch" and "Impact" event types shows a

larger steep and consistency in the distribution of the word count, unlike the "Attack" and "Impact" labels. In the box plot representing the word counts by year, much shorter articles were written in the earlier year (2010) and the cybersecurity event articles written in latter years (2020 and 2021) are longer and consistent.

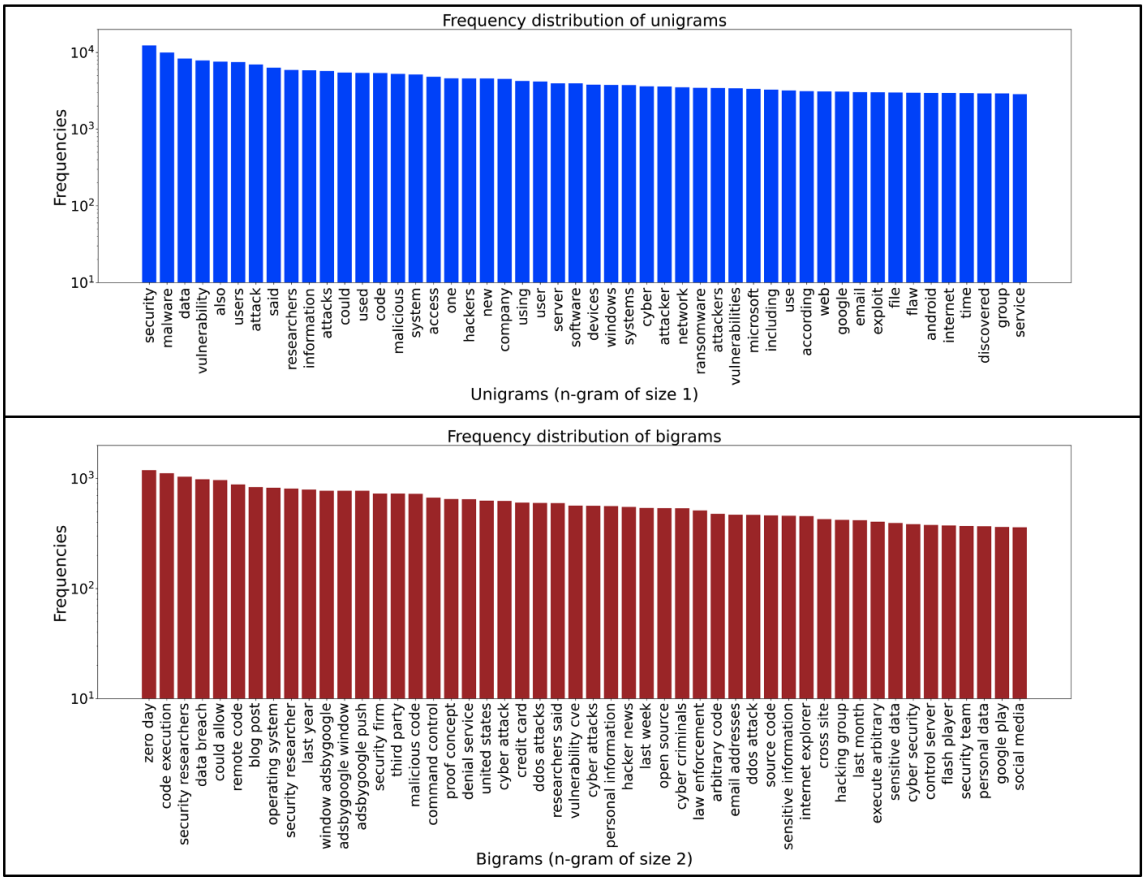

Figure 3.7: Unigrams and Bigrams with their top frequency

Unigrams and Bigrams are two methods that can be used for evaluating and processing text input in NLP (Eisenstein, 2019). Unigrams are single words or tokens in a text whereas bigrams are pairs of adjacent words in a text For instance, the words "There," "was," "a," "ransomware," and "attack" are unigrams in the sentence "There was a ransomware attack," while the words "There was," "was a," "a ransomware," and "ransomware attack" are

bigrams.

Unigrams can be used to analyze the frequency of specific words in a document and to pinpoint essential terms or ideas. With techniques like bag-of-words representation, where a text is represented as a collection of individual words or tokens, they are frequently utilized. On the other hand, bigrams are helpful for recording the context and connections among words in a text. Bigram analysis makes it possible to spot frequent words and expressions, which can help us understand the text's meaning and purpose better (Tan et al., 2002). Bigrams are frequently employed in approaches like n-gram modeling, which examine word sequences in a text to forecast the likelihood of upcoming words in the sequence. Figure 3.7 shows the frequency distribution of unigrams and Bigrams in the cybersecurity news dataset. The top unigrams are "security", "malware", "data", and "vulnerability". The top bigrams are "zero day", "code execution", "security researchers" and "data breach".

3.3.1 Textual Data Preprocessing

In NLP applications, text data is the main source of information, so it is crucial to process and clean it before using it for additional analysis and modeling. Text preprocessing involves converting raw text data into a form that machine learning algorithms can better understand and interpret. These steps are aimed at making the data transform from human-readable to machine-readable format (Eisenstein, 2019; Hapke et al., 2019). In this praxis, we took the first step in enhancing the data quality for modeling by making the text lowercase, extracting texts in brackets, removing the URL links, removing punctuation and other special characters, and removing words with mixture of alphabets and numbers.

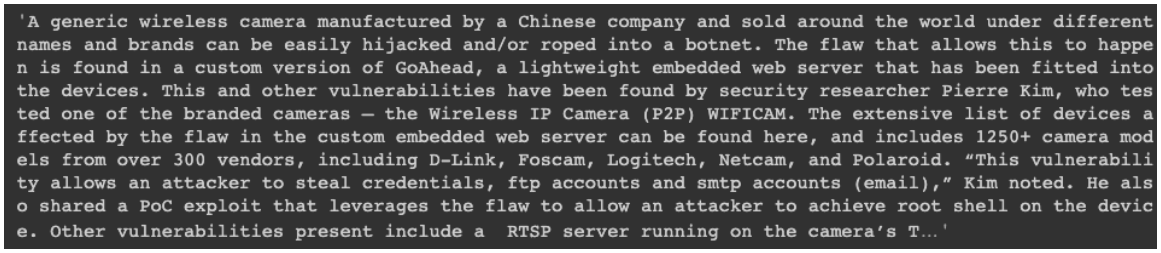

Figure 3.8: Sample Cybersecurity Article before Preprocessing

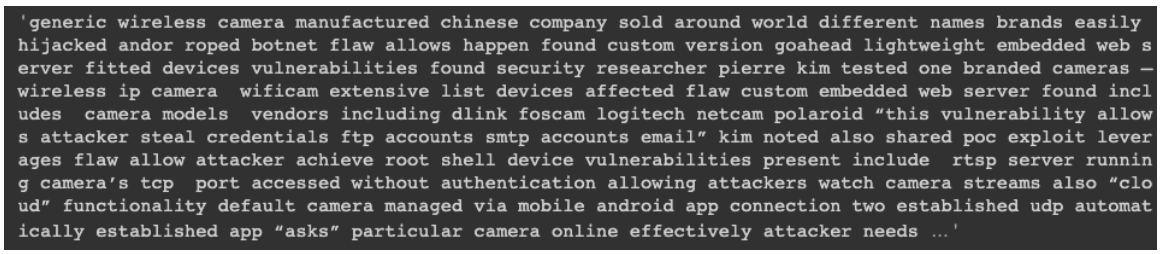

Figure 3.9: Sample Cybersecurity Article after Preprocessing

After the previous preprocessing step, we applied the stop-word removal technique on the texts. To achieve this, we import the module containing the stopwords from Natural Language Toolkit (nltk) library and download the English stop words (Bird, 2006; Hardeniya, 2015). Next, we defined and deconstructed the cleaned cybersecurity texts. Lastly, using a list comprehension to verify whether a word is not in the stop words set after changing its case, we build a new list of words devoid of any stop words. Figure 3.8 and 3.9 are sample texts showing what they look like before and after going through the preprocessing pipeline.

3.4 Vector Representation of Texts: Word Embeddings

In NLP, word embeddings plays an important role because they represent words as vectors in a high-dimensional space, resulting in the clustering of words with similar meanings. Bengio et al. introduced the method in 2003 (Bengio et al., 2003), but it became more well-known after Mikolov

et al published the Word2Vec algorithm in 2013 (Mikolov et al., 2013). Research on word embeddings has advanced significantly since then, leading to modifications to the original Word2Vec algorithm and the creation of advanced methods. The GloVe (Global Vectors for Word Representation) algorithm, developed by Pennington et al. in 2014, is one such advancement (Pennington et al., 2014). GloVe creates word embeddings using a co-occurrence matrix to capture the statistical correlations between words. Word embeddings vectors created by GloVe have been demonstrated to perform better than Word2Vec on a variety of tasks by using a co-occurrence matrix to capture the statistical links between words (Pennington et al., 2014; Stein et al., 2019).

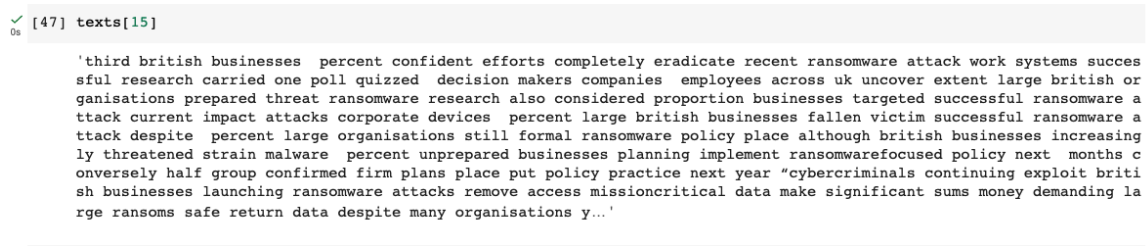

Figure 3.10: Sample Cybersecurity Article before Vector Representation

In this praxis, we used the GloVe method for generating word embeddings and obtaining the vector representation of the cybersecurity texts. The GloVe was trained by using the aggregated word-to-word co-occurrence statistics from a larger corpus. After extracting vectors using GloVe, the vectors were padded using the zero padding method. This process adds extra token of zero sequences of variable length to make all the vectors representing each cybersecurity article uniform in length. The Figure 3.10 and 3.11 is sample text from the cybersecurity news article before and after representing it with vectors.

```
✓ [48] train_padded_sentences[15]
0s

array([  646,  1489,   548,   686,  3425,  1076,   664,  9380,   291,
          35,     7,   321,    28,   629,   178,  1049,    20,  7937,
       33672,  2665,  2326,   156,   380,   229,   894,  4177,  2066,
         334,  1489,  1450,  2804,    71,    35,   178,     6,  1223,
        8355,   548,    79,   629,    35,     7,   566,   571,    15,
         942,    24,   686,   334,  1489,   548,  3674,   139,   629,
          35,     7,  1339,   686,   334,  1450,   123,  9969,    35,
         793,   447,   689,  1489,   548,  1594,  2177,  2032,     2,
         686, 11730,   548,  3206,  1588, 33673,   793,   415,   262,
       21275,  1296,    47,   299,   116,  1739,   447,   762,   793,
        2030,   415,    97, 14739,  2930,    38,  1489,   548,  1179,
          35,    15,   675,    16,  7938,     3,   144,   806,  7027,
         340,  1813,   334,  4050,   626,  1541,     3,  1339,    90,
        1450,   276,   107,   847,  1588,  1977,   693,    34,   108,
        2804,   142, 13025,    10,  2542, 21276,  1021,     1,  8356,
        2931,  9381,  5821,  2967,   200,   431,  2625,   383,  1977,
         447,   286,     7,   156, 17191,  2543,  2712,  5535,    35,
         949,    30,  7302, 33674,  7937,     6, 11731,  2066,    35,
          15,    54,   942,    24,    28,  1753,   623,  4115,   806,
         239,    49,    24,  1906,   548,    94,    24,    49,   291,
          35,     7,    20,   646,   548,   380,    94,    24,    54,
         565,  1450,  3674,   139,    35,     7,   686,  1805,  2777,
          24,    54, 33675,  5535,    35,   132,  2968,  2232,   400,
        1703,   509,   146,   942,     3,  5674,  9382,  7028,  9383,
         182,  1057,   874,  3426,   363,    90,   548,  2289,    35,
          70, 21277,   942,    34,  1451,  1076,   512,   447,  1240,
         371,   229,    90,   162, 13026,  2542, 21276,   538,     0,
           0,     0,     0,     0,     0,     0,     0,     0,     0,
           0,     0,     0,     0,     0,     0,     0,     0,     0,
           0,     0,     0,     0,     0,     0,     0,     0,     0,
           0,     0,     0,     0,     0,     0,     0,     0,     0,
           0,     0,     0,     0,     0,     0,     0,     0,     0,
           0,     0,     0,     0,     0,     0,     0,     0,     0,
           0,     0,     0,     0,     0,     0,     0,     0,     0,
           0,     0,     0,     0,     0,     0,     0,     0,     0,
           0,     0,     0,     0,     0,     0,     0,     0,     0.
```

Figure 3.11: Sample Cybersecurity Article after Vector Representation

3.5 Model Development

In this praxis research, we used three major machine learning methods to fulfill our thesis statement. The core objective of our study is to develop a predictive model that can effectively classify different types of cybersecurity events and offer cybersecurity stakeholders an early and reliable understanding of the evolving trends in such events over time. To achieve this, we will explore the two supervised machine learning models and one

unsupervised machine learning model in more detail in this section.

3.5.1 Unsupervised Machine Learning Model: Dynamic Embedded Topic Models (D-ETM)

Dynamic topic modeling (DTM) is a group of methods for examining how topics change over time. These techniques help you understand the time-dependent representation of topics (Blei and Lafferty, 2006). D-ETM is a generative model of texts that uses word embeddings as the vector representation of texts for input together with the dynamic latent Dirichlet allocation (D-LDA) (Dieng et al., 2019; Dieng et al., 2020). The Embedded Topic Model (ETM) uses the continuous bag-of-words (CBOW) model architecture of word embeddings. The likelihood of the ith word in document d is represented by using the Softmax function:

$$w_{id} \sim \text{softmax}\left(\beta^{\top}\alpha_{id}\right) \tag{3.1}$$

where β is the $A \times B$ embedding matrix with columns containing the vocabulary representation, $\beta_x \in \mathbb{R}^A$, and α_{id} is the sum of all the embedding vectors representing by the context (β_x for each word x). To be able to develop a time-dependent topic model in embedding space, we will be using the D-ETM (Dieng et al., 2019). Within D-ETM, every word is represented by a categorical distribution, derived from the inner product of the word embedding and the embedding representation for each time step. The probability of a word belonging to a topic is determined by the exponential of the inner product between the word embedding, denoted as β_x, and the topics embedding at a

specific time step, indicated as $\alpha_j^{t_d}$.:

$$p\left(w_{in} = x \mid y_{in} = j, \alpha_j^{(t_d)}\right) \propto \exp\left\{\beta_x^\top \alpha_j^{(t_d)}\right\} \tag{3.2}$$

Using Markov Chain in the D-ETM allows for learning from seamless topic trajectories, resulting in topic embeddings evolving with Gaussian noise characterized by variance γ^2,

$$p\left(\alpha_j^{(t)} \mid \alpha_j^{(t-1)}\right) = \mathcal{N}\left(\alpha_j^{(t-1)}, \gamma^2 I\right) \tag{3.3}$$

The prior distribution of θ_d, which is dependent on a latent variable λ_{td}, is:

$$p\left(\theta_d \mid \lambda_{t_d}\right) = \mathcal{LN}\left(\lambda_{t_d}, c^2 I\right) \tag{3.4}$$

where $p\left(\lambda_t \mid \lambda_{t-1}\right) = \mathcal{N}\left(\lambda_{t_1}, \delta^2 I\right)$ and $\mathcal{LN}$ is the Logistic-Normal distribution. The generative process of D-ETM is shown in Figure 3.12.

Algorithm 1 D-ETM Generative Process

1. Draw initial topic embedding $\alpha_j^{(0)} \sim \mathcal{N}(0, I)$
2. Draw initial topic proportion mean $\lambda_0 \sim \mathcal{N}(0, I)$
3. **for** time step $t = 1, \ldots, T$ **do**
4. Draw topic embeddings $\alpha_j^{(t)} \sim \mathcal{N}(\alpha_j^{(t-1)}, \delta^2 I)$
5. Draw topic proportion means $\lambda_t \sim \mathcal{N}(\lambda_{t-1}, \delta^2 I)$
6. **end for**
7. **for** each document d in dataset D $(d \in D)$ **do**
8. Draw topic proportion $\theta_d \sim \mathcal{LN}(\lambda_{t_d}, c^2 I)$
9. **for** each word in the document d $(i \in d)$ **do**
10. Draw topic assignment $y_{id} \sim \text{Cat}(\theta_d)$
11. Draw word $w_{id} \sim \text{Cat}(\text{softmax}(\beta^\top \alpha_{y_{id}}^{(t_d)}))$
12. **end for**
13. **end for**

Figure 3.12: Generative Process for D-ETM. Adapted from (Dieng et al., 2019; Dieng et al., 2020).

The D-ETM's inference algorithm employs variational inference to approximate the posterior distribution $q_x(\theta, \lambda, \alpha)$. The variational parameters are optimized by minimizing the Kullback-Leibler (KL) divergence between this approximation and the actual posterior distribution (Dieng et al., 2020):

$$D_{KL}(p||q) = \mathbb{E}_q \left[\log \frac{p(D, \theta, \lambda, \alpha)}{q_x(\theta, \lambda, \alpha)} \right] \tag{3.5}$$

3.5.1.1 Evaluating Topic Models

To evaluate the predictive performance of the topic models, we used the perplexity score on the test set, given the D-ETM parameters learned during the model training. The perplexity exhibits a consistent decrease as the likelihood of the test data increases (Steyvers et al., 2004; Wallach et al., 2009). For a good model, a high likelihood would be assigned to the held-out test documents, and thus, a low perplexity.

$$\text{Perplexity}\,(D_{\text{test}}) = -\frac{\sum_n \log\left(p\left(w_n\right)\right)}{\sum_n w_n} \tag{3.6}$$

Three other metrics will be used in this praxis. The first is the Topic Coherence (TC), which is a metric used for estimating human-level interpretability at the topic model level. This is achieved by taking the normalized pointwise mutual information (NPMI) between two topics randomly drawn from the same document (Mimno et al., 2011). The NPMI is given as:

$$\text{NPMI}\,(w_i) = \sum_j^{N-1} \frac{\log \frac{p\left(w_i, w_j\right)}{p(w_i)p\left(w_j\right)}}{-\log p\left(w_i, w_j\right)} \tag{3.7}$$

The second metric is known as topic diversity (TD). It represents the proportion of unique words present in the top 25 words across all topics. The

higher the topic diversity, the less redundant the topics are (Lau et al., 2014). The final metric is the topic quality (TQ). This is a better interpretable metric that combines the product of the topic coherence (TC) and topic diversity (TD) (Dieng et al., 2019):

$$TQ = TC \times TD \tag{3.8}$$

3.5.2 Supervised Machine Learning Models: Bi-LSTM and BERT

The Bi-LSTM is a sequential model comprising two LSTMs, where one processes input in the forward direction and the other in the backward direction (Liu and Guo, 2019; Goodfellow et al., 2016). The Bi-LSTM is one of the deep learning models used for event type classification in this praxis. The input sequence is processed in accordance with usual procedure by the forward LSTM layer, with each time step's output being a function of the input and the prior hidden state. The input sequence is processed by the backward LSTM layer in reverse order, with each time step's output being a function of the input and the subsequent hidden state (Liu and Guo, 2019).

The Figure 3.12 is the Bi-LSTM model used in this praxis. Word embedding vectors with a dimensions of 100 and 300 representing the cybersecurity texts were fed into the Bi-LSTM layer consisting of the forward and backward layers. The concatenation (last) layer combines the output of both layers before sending it to a dropout layer, which is a regularization technique for combating overfitting. Finally, the output is sent to a dense layer before the softmax layer that calculates the probabilities of each event type. The Bi-LSTM model summary used in this praxis is shown in Figure 3.13. This includes the number of parameters and the description of each layer in the model.

Recently published by researchers (Vaswani et al., 2017; Devlin et al.,

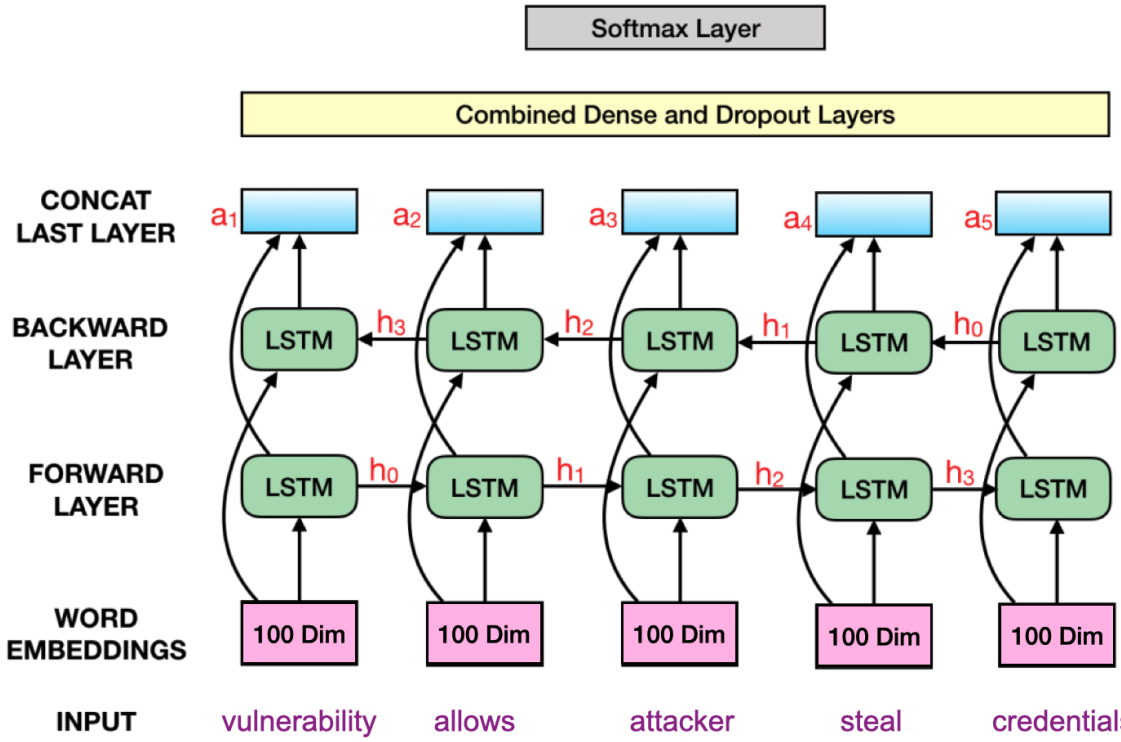

Figure 3.13: Bi-LSTM Model used in this praxis showing a 100-dimension word embeddings from the input data, the backward and forward layers, the combined dense and dropout layers, and the final softmax output layer.

2018), BERT has ignited the machine learning community by demonstrating profound results in a variety of NLP tasks, such as Texts Classification, Neural Machine Translation (NMT), Question Answering (SQuAD v1.1), and Natural Language Inference (NLI) (Rajpurkar et al., 2016). BERT's bidirectional training enables a more comprehensive understanding of language context and flow than unidirectional models (Devlin et al., 2018).

In this praxis, we fine-tuned BERT for our classification tasks by adding a small layer to the pre-trained BERT-Large model. This task is done similar to doing a classification task for a predicting the next sentence, where a classification layer is added on top of the Transformer model output. The Figure 3.14 is the summary of the BERT model used in this praxis. The total number of trained parameters for the BERT model is approximately

```
Model: "sequential_10"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_10 (Embedding)    (None, 1063, 100)         7343800

 bidirectional_10 (Bidirecti  (None, 1063, 128)        84480
 onal)

 global_max_pooling1d_10 (Gl  (None, 128)              0
 obalMaxPooling1D)

 batch_normalization_10 (Bat  (None, 128)              512
 chNormalization)

 dropout_30 (Dropout)        (None, 128)               0

 dense_30 (Dense)            (None, 128)               16512

 dropout_31 (Dropout)        (None, 128)               0

 dense_31 (Dense)            (None, 128)               16512

 dropout_32 (Dropout)        (None, 128)               0

 dense_32 (Dense)            (None, 4)                 516

=================================================================
Total params: 7,462,332
Trainable params: 7,462,076
Non-trainable params: 256
_________________________________________________________________
```

Figure 3.14: Summary of the Bi-LSTM Model Used in this Praxis

335 million. The model has a dropout layer and a dense layer in addition to the pre-trained BERT model.

3.6 Conclusion

In this chapter, we reviewed the different important methodology used in this praxis. First, we presented the end-to-end process of the methodology which includes the raw text extraction process, the data pre-processing step, data exploration and feature transformation process before exploring the methods for model development and evaluation. These methods such as the supervised machine learning method (Bi-LSTM, BERT model) and the unsupervised model (D-ETM) will enable engineers and other cybersecurity

```
Model: "functional_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_word_ids (InputLayer)  [(None, 80)]              0
_________________________________________________________________
tf_bert_model (TFBertModel)  ((None, 80, 1024), (None, 335141888
_________________________________________________________________
tf_op_layer_strided_slice (T [(None, 1024)]            0
_________________________________________________________________
dropout_73 (Dropout)         (None, 1024)              0
_________________________________________________________________
dense (Dense)                (None, 4)                 4100
=================================================================
Total params: 335,145,988
Trainable params: 335,145,988
Non-trainable params: 0
```

Figure 3.15: Summary of the BERT Model Used in this Praxis

stakeholders to analyze trends over time from cybersecurity articles and classify cybersecurity event types which are interpretable. These methods will help the cybersecurity industry avoid the expense of hours spent analyzing big texts for information about cyber-related events and improve the fidelity and interpretability of cybersecurity event type classifiers.

Chapter 4: Results

4.1 Introduction

In this chapter, we documented and discussed the extensive results of the models developed in this study for classifying event types from cybersecurity news articles and identifying and understanding trends from the documents. The first set of models are two deep learning algorithms, Bi-LSTM and BERT classifiers, for categorizing cybersecurity news articles into different cyber events. In this section, we presented the model results during training at different epochs and the performance of the model on the test dataset. Next, we demonstrate how model agnostic explainability techniques can be used to interpret the models. Explaining the results of black-box machine learning models applied to cybersecurity data helps major stakeholders with improving adoption, increases the comfort level with model predictions and augments the understanding of uncertainties that are intrinsic to complex machine learning models. For the second set of models, we present the performance with classical tree-based and non-tree based supervised machine learning models using several text representation techniques and compared the performance on the same unseen testing dataset. The last model is the D-ETM for extracting latent topics from unlabeled cybersecurity articles and understanding the trend of these topics over time. This D-ETM model offers a better qualitative performance over other techniques because of the representation of texts via word embeddings. Finally, we presented a test case for the D-ETM model based on the characterization of the uncovered latent topics.

4.2 Model One: Deep Learning Models for Classifying Cybersecurity Event Types

For our first set of models, we developed two deep learning models for classifying the four event types in cybersecurity news articles. The deep learning models are the Bi-LSTM and the BERT model.

4.2.1 Model 1A: Bi-LSTM Model for Classifying Cybersecurity Event Types

The first deep learning model is the Bi-LSTM classifier for classifying the cybersecurity newspaper article into four different types: Attack, Patch, Discover, and Impact. The model was trained and validated using 9566 and 3189 cybersecurity news articles respectively, with 56% labeled as "Attack", 29% labeled as "Discover", 10% labeled as "Impact", and 5% labeled as "Patch". The 9566 cyber news articles are the training dataset reserved for training the Bi-LSTM model and the 3189 articles are the testing dataset which is unseen by the trained model. The texts were first represented using the GloVe word embeddings (Pennington et al., 2014), an unsupervised learning algorithm for vector representation of words. The 100-dimensional pretrained word vectors with 6 billion tokens was downloaded and used to create word embedding matrix for each token in the cybersecurity news articles. These embedding matrix are then fed into the input of the Bi-LSTM model for model training. The model was executed for 5 epochs, with a batch size of 128 cybersecurity news articles and a learning rate of 0.001. The categorical cross-entropy was used as the loss function and the ADAM (adaptive moment estimation) (Kingma and Ba, 2014) was used as the optimizer.

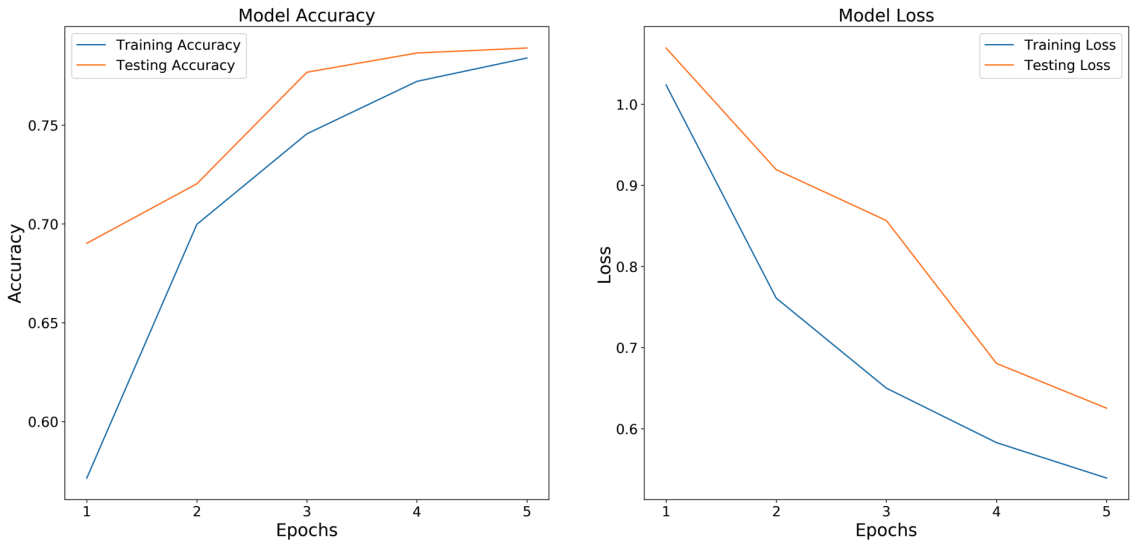

Figure 4.1: Training and Testing Accuracy and Loss for the Bi-LSTM model for Classifying Cybersecurity Articles into Event Types

Epochs	Training Loss	Training Accuracy	Testing Loss	Testing Accuracy
1	1.0237	0.5714	1.0692	0.6902
2	0.7610	0.6998	0.9193	0.7203
3	0.6500	0.7456	0.8566	0.7767
4	0.5828	0.7721	0.6805	0.7865
5	0.5392	0.7839	0.6253	0.7890

Table 4.1: Model Training Metrics of the Bi-LSTM model for Classifying Cybersecurity Articles into Event Types

The Figure 4.1 shows the training and testing/validation accuracy and loss for the Bi-LSTM model classifying algorithm. In Table 4.1, we show the numerical values of these loss and accuracy for the different epochs. The similarities of the model results on both the training dataset and testing dataset is an indication that the model is not overfitting on the training set.

4.2.2 Model 1B: BERT Model for Classifying Cybersecurity Event Types

The second deep learning model used for classifying cybersecurity news articles into event types is the BERT model. As a special type of masked language model, the BERT model enforces a bidirectional learning from corpus of texts by masking a word in the sentence and using the model to bidirectionally predict the masked word on either side. For this praxis, we fine-tuned the pretrained BERT Large model from hugging on our classification tasks. The BertTokenizer, which is a tokenization technique based on word-piece, was used from the Transformer library to create a word and sub-word representation vector of the corpus (Kamps et al., 2020; Devlin et al., 2018). The pre-trained BERT transformer layer was use alongside a dropout layer with a dropout probability of 0.1 and an output dense layer with a softmax activation function. The fine-tuned model has approximately 335 million trainable parameters. To train the model by fine-tuning BERT, we used a batch size of 16 tokenized cybersecurity news articles and trained for 10 epochs with an early stopping callback added to it to monitor the validation loss and reduce overfitting.

The Table 4.2 is the classification report table containing both the evaluation metrics from the performance of the Bi-LSTM and BERT models on the unseen testing dataset. The table shows the Precision, Recall, and F1-Score for the different event types and the support being the number of testing examples for each event type. From the results table, BERT slightly outperform Bi-LSTM model with an Accuracy of 0.7971, a Weighted Average

	Models	Precision	Recall	F1-Score	Support
Attack	Bi-LSTM	0.8353	0.8701	0.8523	1778
	BERT	0.8423	0.8774	0.8595	
Discover	Bi-LSTM	0.7513	0.7917	0.7710	912
	BERT	0.7594	0.7967	0.7776	
Impact	Bi-LSTM	0.6272	0.5255	0.5719	333
	BERT	0.6439	0.5424	0.5888	
Patch	Bi-LSTM	0.7423	0.4337	0.5475	166
	BERT	0.7475	0.4458	0.5585	
Accuracy	Bi-LSTM		0.7890		3189
	BERT		**0.7971**		
Macro-Average	Bi-LSTM	0.7390	0.6553	0.6857	3189
	BERT	0.7483	0.6656	0.6961	
Weighted Average	Bi-LSTM	0.7847	0.7890	0.7839	3189
	BERT	**0.7931**	**0.7971**	**0.7923**	

Table 4.2: Comparing Bi-LSTM and BERT Model Evaluation Metrics for Classifying Cybersecurity NEws Articles

Precision, Recall, and F1-Score of 0.7931, 0.7971, and 0.7923, respectively.

4.2.3 Explainability of Black Box Models

The concept of machine learning explainability lacks a universally accepted definition, but it is commonly regarded as an essential component that encompasses crucial principles such as trust, causality, informativeness, and transparent decision-making (Doshi-Velez and Kim, 2017). In supervised machine learning, stakeholders typically prioritize evaluation metrics that demonstrate the model's performance such as model accuracy and F1-Score. However, there are situations where stakeholders require insights into the model's decision-making process, leading to the inevitable need for model explainability (Lipton, 2018). Explainability can be understood as the extent to which a human can understand the rationale behind a particular decision. From the perspective of a machine learning practitioner, it refers to the level at which the model's outcomes can be consistently predicted. Machine learning explainability plays a vital role in enhancing

adoption, fostering a higher level of comfort with model predictions, and deepening the understanding of the inherent uncertainties associated with machine learning models (Molnar, 2020).

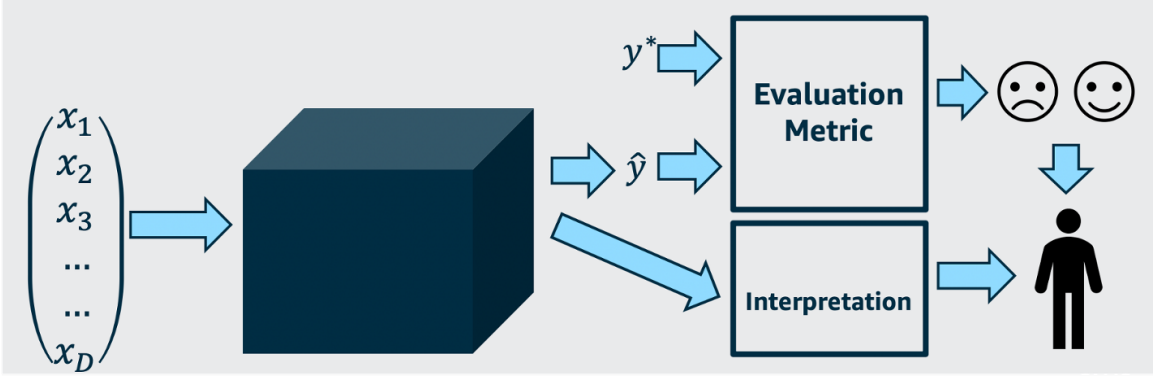

Figure 4.2: In General machine learning, the evaluation metrics are obtained by using the predictions on the test data and the ground truth label. However, additional cybersecurity stakeholders like the cyber risk manager or chief information security officer (CISO) can demand explainability when using ML for Cybersecurity event classification. Adapted from (Lipton, 2018)

Generally, explainability approaches are classified into two groups. The first group focuses on individual explanation (local explainability) while the second summarizes the entire model behavior on a population level (global explainability). These explainability can be model-specific and model-agnostic. The former is limited to specific models which by definition are intrinsic to models that are interpretable. The latter, on the other hand, is applied to black box machine learning models after they have been trained (post-hoc) (Molnar, 2020). These model-agnostic methods usually work by using a proxy model to dynamically learn an approximation of the representation of a complex, black-box model from the outputs of the model. In this praxis, we will focus on a model agnostic local explainable AI method known as LIME (Local Interpretable Model-agnostic Explanations) (Molnar, 2020; Ribeiro et al., 2016). We will use LIME on the output of our model to explain different use case. The LIME method does not have access to

the ML model internals and parameters such as model weights or other structural information (Ribeiro et al., 2016).

4.2.3.1 Explainable AI use case one: Cybersecurity News Article with an "Attack" Event Type

The first use case is a cybersecurity news article published on how Pennsylvania Democrats' computer network was hit by a ransomware attack, but no other state agencies were affected, and efforts used to resolve the issue without evidence of data compromise or targeted caucus system. This article is labeled with "Attack" as the cyber event.

```
Democrats in Pennsylvania's state Senate were locked out of their computer network early Friday morning due to a
ransomware attack, NBC News reports. According to an unidentified state official who spoke with NBC, the Democrat
ic senators in Harrisburg use their own computer network and "there is no indication that other state agencies of
the Republicans have been affected". As of about 5 p.m. Friday, both law enforcement agencies and Microsoft were
working with the state Democrats to free their network. In a statement sent to reporters via text message and obt
ained by The Hill, state party officials said, "there is currently no indication that the caucus system was targe
ted or that any data has been compromised". Recently, ransomware attacks have struck everywhere from hospitals an
d universities to San Francisco's transit system. Last summer, the congressional IT desk warned representatives i
n Washington DC to be careful of potential ransomware and phishing threats, but the hacks on the DNC were unrelat
ed. In many cases, the payment demanded is only in the tens of thousands of dollars, and occasionally ransomware
can be spammed without a specific target, but the affected computer systems are encrypted and inaccessible until
the hackers release a key. If a network's data is backed up offsite, the target can occasionally circumvent the r
ansom altogether – albeit with some increased security. A spokesperson for the Pennsylvania Democrats declined to
say to NBC News whether that was possible in this case, or whether the attackers had revealed any motives
```

Figure 4.3: Published Cybersecurity News Article with an "Attack" Event Type

The local explanation using LIME shows the prediction probabilities for the news article. These probabilities are - Attack: 0.96, Discover: 0.02, Impact: 0.01, Patch: 0.01. The prediction probability is commensurate with the original label which is "Attack". The explanation shows the probabilities of the top words that contributes to this article being an "Attack" event type. It also highlights the explanation probabilities for the other three class labels.

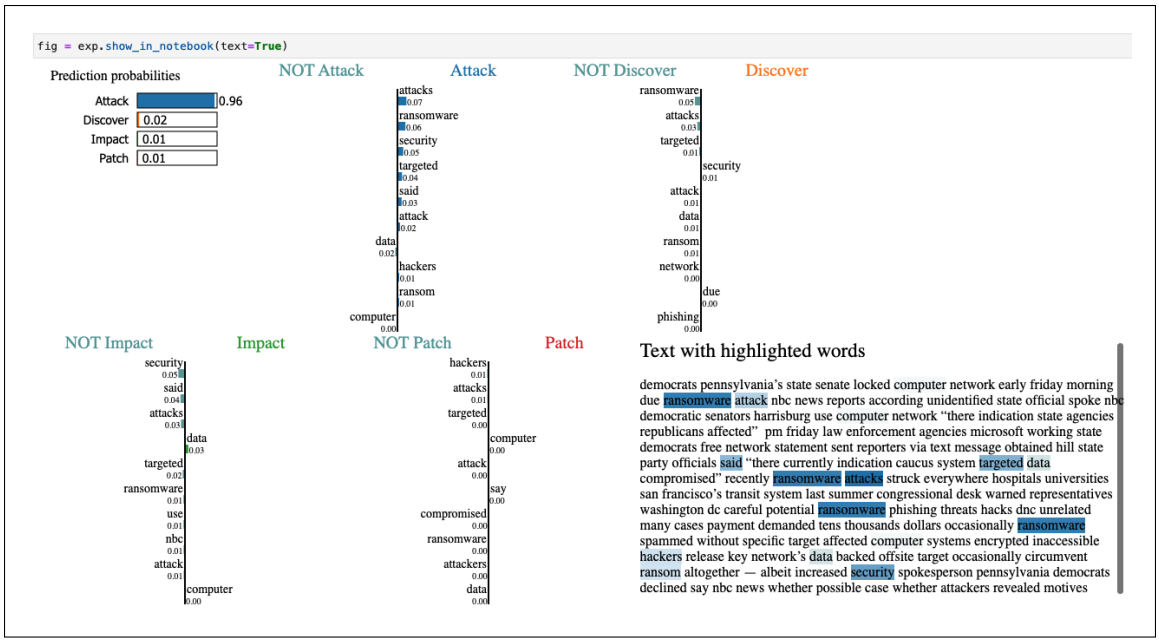

Figure 4.4: Model-Agnostic Post-hoc Local Explanation of Use Case One

4.2.3.2 Explainable AI use case two: Cybersecurity News Article with a "Discover" Event Type

The second use case is a cybersecurity news article published about the discovery of a critical vulnerability in Intel processors' remote management features. This vulnerability enabled attackers to remotely control affected computers, but specific details remain undisclosed, and Intel released firmware updates to address the issue. This article is labeled with "Discover" as the cyber event.

The local explanation using LIME shows the prediction probabilities for the news article. These probabilities are - Attack: 0.08, Discover: 0.90, Impact: 0.00, Patch: 0.02. The prediction probability is consistent with the original label which is "Discover". The explanation shows the probabilities of the top words that contributes to this article being a "Discover" event type. The texts with the highlighted word is the clean and processed texts

Updated: Since the below-reported vulnerability is highly critical and it would take a few weeks for sysadmins to protect their enterprise network, the research t
eam has not yet disclosed the technical details of the vulnerability.

Meanwhile, I have talked with Maksim Malyutin, a member of Embedi research team who discovered the vulnerability in March, and updated my article based on the inf
ormation provided by him.

A critical vulnerability has been discovered in the remote management features on computers shipped with Intel processors for past seven years (and not decade), w
hich could allow attackers to take control of the computers remotely, affecting all Intel systems, including PC, laptops, and servers, with AMT feature enabled.

As reported earlier, this critical flaw (CVE-2017-5689) is not a remote code execution, rather Malyutin confirmed to The Hacker News that it's a logical vulnerabi
lity that also gives remote attackers an opportunity to exploit this bug using additional tactics.

This elevation of privilege bug resides in the Intel Management Engine (ME) technologies such as Active Management Technology (AMT), Small Business Technology (SB
T), and Intel Standard Manageability (ISM), according to an advisory published Monday by Intel.

These remote management features allow a systems administrator to remotely manage large fleets of computers over a network (via ports 16992 or 16993) in an organi
zation or an enterprise.

Since these functions are present only in enterprise solutions, and mostly in server chipsets, Intel claims that the vulnerability doesn't affect chips running on
Intel-based consumer PCs.
But Malyutin told us that "Intel-based consumer PCs with official support of Intel vPro (and have Intel AMT feature enabled) could also be at risk," and "there is
also a chance of attacks performed on Intel systems without official Intel AMT support."

According to the Intel advisory, the vulnerability could be exploited in two ways:

An unprivileged network attacker could gain system privileges to provisioned Intel manageability SKUs: Intel AMT and ISM. However, Intel SBT is not vulnerable to
this issue.
An unprivileged local attacker could provision manageability features gaining unprivileged network or local system privileges on Intel manageability SKUs: Intel A
MT, ISM, and SBT.

Figure 4.5: Published Cybersecurity News Article with an "Discover" Event Type

that includes tokenizing, stemming, and removal of stop words.

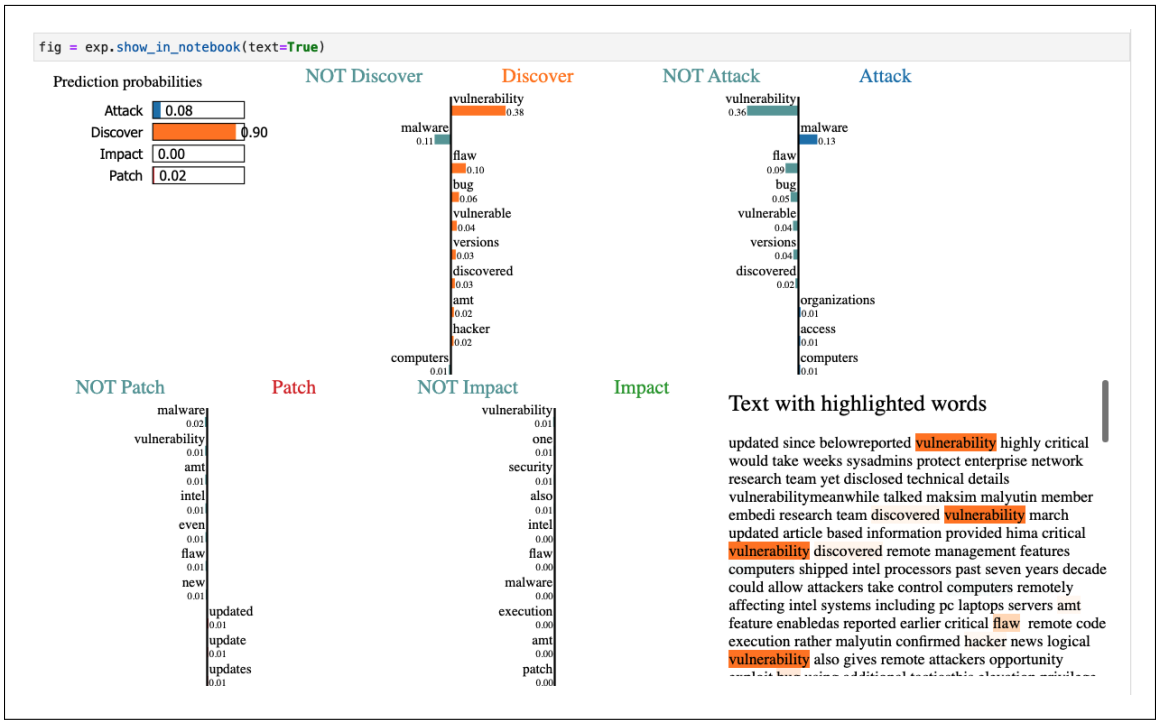

Figure 4.6: Model-Agnostic Post-hoc Local Explanation of Use Case Two

4.2.3.3 Explainable AI use case three: Cybersecurity News Article with an "Impact" Event Type

The third use case is a cybersecurity news article published about how Sony Pictures Entertainment was hit by a massive data breach during the holiday season, resulting in a network shutdown and the leak of five unreleased movies by hackers known as "Guardians of Peace." This article is labeled with "Impact" as the cyber event type.

```
What a horrible start the holiday season in U.S. Over Tha
nksgiving weekend, Sony Pictures Entertainment suffered a
massive data breach as " Guardians of Peace " hacked - in
to Sony Pictures' computer system that brought the studi
o's network to a screeching halt . Following the hack, ha
ckers leaked five unreleased Sony movies to Torrent file
- sharing website during Black Friday.
```

Figure 4.7: Published Cybersecurity News Article with an "Impact" Event Type

The local explanation using LIME shows the prediction probabilities for the news article. These probabilities are - Attack: 0.25, Discover: 0.05, Impact: 0.69, Patch: 0.01. The highest prediction probability is consistent with the original label which is "Impact". The explanation shows the probabilities of the top words that contributes to this article being an "Impact" event type. Top words such as "breached" and "leaked" contributed to the article describing the impact of a cyber event.

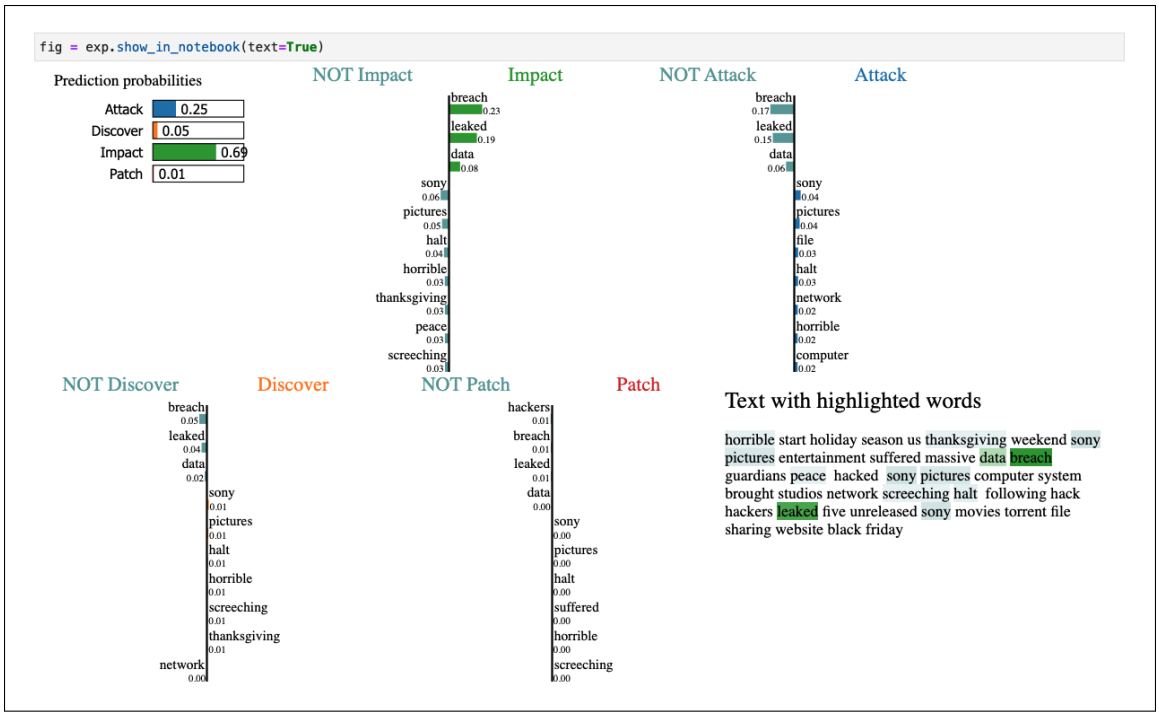

Figure 4.8: Model-Agnostic Post-hoc Local Explanation of Use Case Three

4.2.3.4 Explainable AI use case three: Cybersecurity News Article with a "Patch" Event Type

The third use case is a cybersecurity news article about how Apple released an emergency patch, to fix critical vulnerabilities, including one that could enable hackers to execute code on the Wi-Fi chip. This article is labeled with "Patch" as the cyber event type.

The local explanation using LIME shows the prediction probabilities for the news article. These probabilities are - Attack: 0.18, Discover: 0.22, Impact: 0.05, Patch: 0.55. The highest prediction probability is consistent with the original label which is "Patch". The explanation shows the probabilities of the top words that contributes to this article being a "Patch" event type. Top words such as "released", "fixes" and "patch update" contributed to the article describing a cyber event related to patches.

```
Less than a week after Apple released iOS 10.3 with over
100 bug fixes and security enhancements; the company has
just pushed an emergency patch update – iOS 10.3.1 – to a
ddresses a few critical vulnerabilities, one of which cou
ld allow hackers to " execute arbitrary code on the Wi –
Fi chip . "
```

Figure 4.9: Published Cybersecurity News Article with an "Patch" Event Type

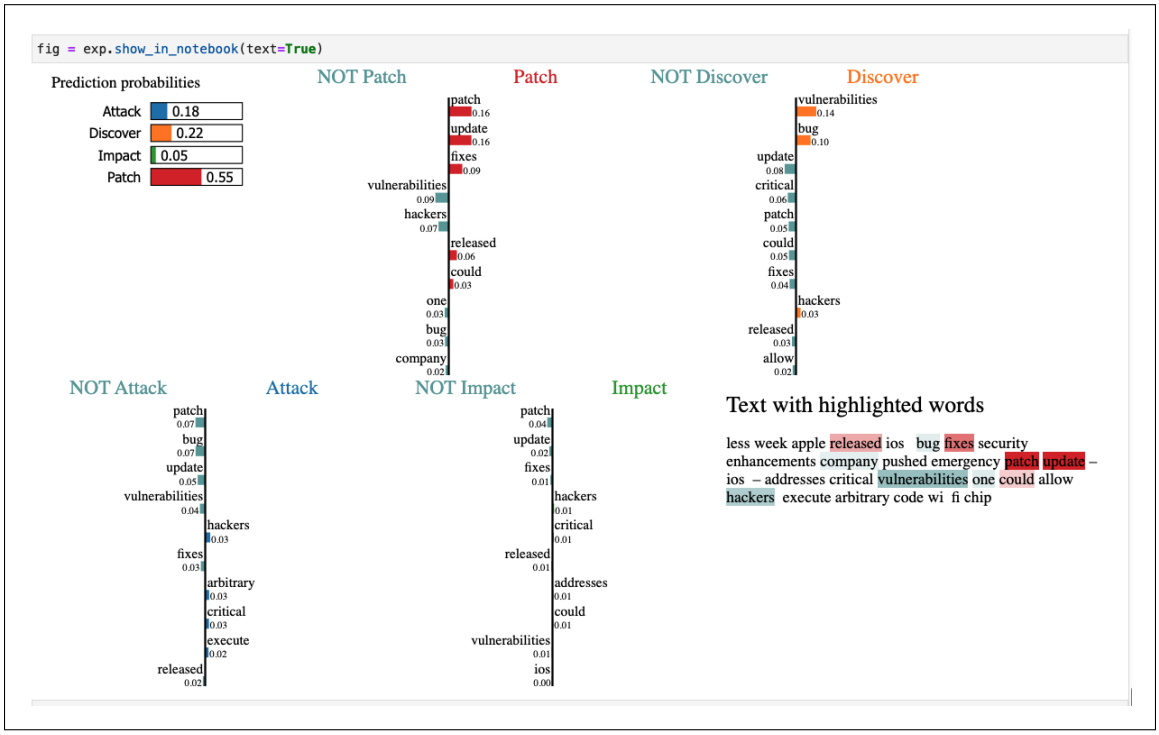

Figure 4.10: Model-Agnostic Post-hoc Local Explanation of Use Case Four

4.2.4 Hypothesis One

H1: An explainable predictive model can identify cybersecurity event types with 70% accuracy

The first hypothesis evaluates the performance of the deep learning model (Bi-LSTM) for classifying cybersecurity event types. The predicted

results from evaluation on the test data, which was unseen by the model during training, was compared with the output of the true/actual labels to generate the prediction statistics and values of the evaluation metrics. The Table 4.2 shows these prediction statistics and hypothesis for this model. The testing precision, recall, and F1 score are the weighted average values from the individual metrics for each event type. For the test, we used the one-sample proportion z-test to compare the sample's proportion to a known population proportion. The test was done using an alpha value of 0.05 ($\alpha = 0.05$) with a 70% accuracy.

Prediction Statistics			
True Positives	2515	Testing Accuracy	0.7890
False Positives	671	Testing Precision	0.7847
True Negatives	3185	Testing Recall	0.7890
False Negative	674	Testing F1 Score	0.7839
One Proportion Z-Test			
$H_0 = 0.70$ $H_1 > 0.70$			
Sample Proportion	0.7890	95% Confidence Interval for Proportion	(0.7744, 0.8028)
Z-Statistic	10.91821	P-Value	0.0000

Table 4.3: Prediction Statistics and Results of Hypothesis Tests for Bidirectional LSTM Model Classifier for Classifying Cybersecurity Event Types

A significance level of $\alpha = 0.05$ was used for all the hypothesis tests in this praxis. The table shows that the model accuracy on the test set is 0.7890 and this first hypothesis is to test using the one-sample proportion Z-test to evaluate that the model has at least a 70% (0.70) accuracy. With the P-Value of 0.000, we reject the null hypothesis that the explainable predictive model can identify cybersecurity event types with 70% accuracy, and conclude that the predictive model (Bi-LSTM) is sufficient enough to classify cybersecurity event types from news articles.

4.3 Model Two: Classical Machine Learning Models for Classifying Cybersecurity Event Types

We evaluated classical machine learning methods using traditional NLP text preprocessing and word representation techniques on the cybersecurity news articles. For the classical preprocessing steps, we made the text lowercase, removed links, special characters, and punctuation, removed stop words before we finally applied stemming to the words in the corpus. For Text representation methods for the classical ML techniques, we used the CountVectorizer and Term Frequency - Inverse Document Frequency (TF-IDF) technique. A collection of text documents is transformed into a matrix representation using CountVectorizer so that each document is represented by a vector of word counts. The text is then tokenized to create a structured vocabulary list, and the frequency of each word in each document is counted. TF-IDF is used in NLP to evaluate the relevance of a phrase/word by taking into account its frequency within a document as well as its uniqueness throughout the complete corpus of document.

The two classical machine learning methods used are the Multinomial Naive Bayes (NB) method and the eXtreme Gradient Boosting (XGBoost) algorithm. The Multi-NB was trained first using the CountVectorizer representation and then the TF-IDF. The XGBoost model was trained using only the TF-IDF vectors. The XGBoost model pipeline used the following parameters for the training paradigm: Learning Rate = 0.1, Maximum Depth = 7, Number of Estimators = 80, Minimum Child Weight = 5, Subsample Rate = 0.7. Both the Multi-NB and XGBoost model were trained and evaluated on the same data used in the Section 4.1 of this chapter. The training data and unseen testing data consists of 9566 and 3189 Cybersecurity news

articles, respectively.

Model Type	Text Representation	Accuracy	F1-Score	Precision	Recall
Multi-NB	CountVectorizer	0.7197	0.6636	0.6739	0.7197
Muti-NB	TF-IDF	0.6754	0.6053	0.5752	0.6754
XGBoost	TF-IDF	0.7695	0.7555	0.7603	0.7695
Bi-LSTM	Word Embedding	0.7890	0.7839	0.7847	0.7890
BERT	WordPiece Embedding	0.7971	0.7923	0.7931	0.7971

Table 4.4: Machine Learning model valuation showing results for the different classical and Deep Learning Model Types.

The Figure 4.4 is a table that compares the classical ML methods (Multi-NB and XGBoost) with the deep learning based models (Bi-LSTM and BERT). The classical techniques used a text preprocessing together with the CountVectorizer and TF-IDF for text representations. The deep learning based model used the word and wordpiece embedding techniques for test representation. The word embedding uses a pretrained unsupervised algorithm called GloVe, and it uses word co-occurrences over the entire corpus of data (Pennington et al., 2014). The pretrained GloVe vectors used to initialize the embedding layer has a dimension of 100. The wordpiece is a type of sub-word tokenization technique that identifies and merges character pairs in a manner that maximizes the likelihood of the training data when added to the vocabulary (Song et al., 2021). The DL-base model outperforms the classical techniques based on the Accuracy and the Weighted average Precision, Recall, and F-1 Scores.

4.3.1 Hypothesis Two

H2: The model using deep learning algorithm and tree-based ensemble learning algorithm have the same performance.

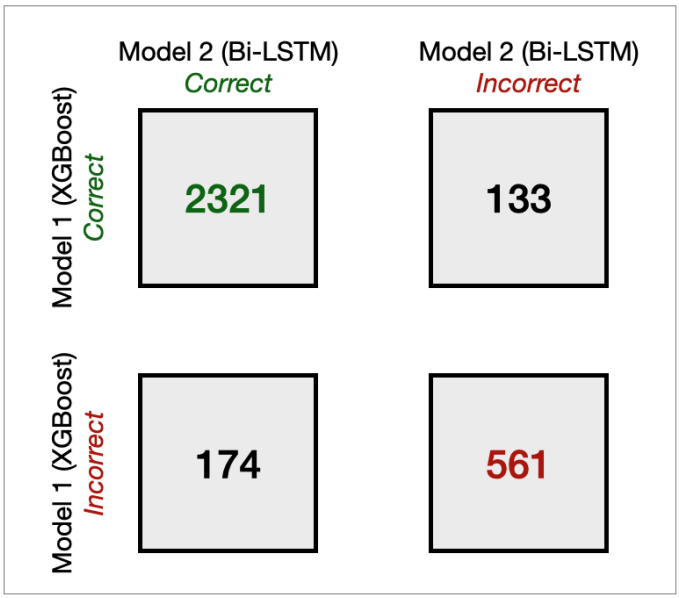

Figure 4.11: Contingency Table Layout comparing the prediction performance of XGboost Model and Bi-LSTM Model in context of the McNemar's Test.

The second hypothesis compares the performance of the deep learning algorithm (Bi-LSTM) to that of the tree-based ensemble learning algorithm for classifying cybersecurity events. To compare the two models, we will use the McNemar's test, which is a statistical test that is used to compare the proportions of two sets of paired categorical data (Lachenbruch, 2014; **?**). When the same individuals or subjects are measured or classed under two separate conditions or treatments, it is commonly used. The test determines whether there is a statistically significant difference in the marginal proportions of the two categories under consideration. The McNemar's test assumes that the test data is randomly sampled, assumes independence, and that the groups are mutually exclusive. Since the test is focused mostly on the predictive differences between the two classifiers, we'll need to prepare a contingency table to represent the frequencies of the two models.

Prediction Statistics			
Model 1: Correct Model 2: Correct	2321	Model 1: Correct Model 2: Incorrect	133
Model 1: Incorrect Model 2: Correct	174	Model 1: Incorrect Model 2: Inorrect	561
McNemar's Test With Edwards Correction			
H_0: Model 1 = Model 2 H_1: Model 2 > Model 1			
Test Statistics $\chi^2 = 5.212$			
P-Value = 0.0224			

Table 4.5: Prediction Statistics and Results of Hypothesis for comparing Classical Machine Learning Algorithm (Model 1) and Deep Learning Algorithm (Model 2) for Cybersecurity Event type Classification

The Figure 4.11 sets up the McNemar's contingency table with the sum of when Model 1 and Model 2 made the correct or incorrect prediction on the test data. The off-diagonal terms are the most important on the table because they show the count of the samples where the models made correct or incorrect predictions relative to one another. For this test, we formulate that Model 2 does not perform better than Model 1 and vice versa. For this praxis, we used the variant of the McNemar's test that uses the Edward continuity correction (Edwards, 1948; Dietterich, 1998). This corrected method increases the accuracy of the test especially in cases where we have discrete frequencies. The computed χ^2 for the McNemar's test with Edward correction is 5.212 and the computed P-Value is 0.0224. This P-Value is below the set significance threshold, alpha ($\alpha = 0.05$). Therefore, we will reject the null hypothesis that both the classical machine learning algorithm and deep learning algorithm have similar predictive performance. Thus, the deep learning based model performed better than the classical method.

4.4 Model Three: Dynamic Embedded Topic Models for Identifying and Analyzing Cybersecurity Trends

To quantify and identify the trends in cybersecurity news events over time, we used the D-ETM model. The D-ETM is a dynamic topic model that utilizes embedding vectors to represent words and topics. The cybersecurity news corpus curated for this praxis spans a time period of 12 years (from 2010 to 2021). We pre-processed the data for D-ETM using the standard techniques such as the removal of special characters, numbers, and punctuation before applying a tokenization scheme to it. Due to the length of some of the cybersecurity news articles, we split the news articles into multiple paragraphs and used the individual paragraphs as standalone documents for the quantitative analysis. We also removed stop words from the document. The data was further split into a train/test/validation size of 85%/10%/5%. This led to us having 45,557 documents for our training set, 5362 documents for our test set, and 2681 documents for our validation set.

	Dynamic Embedded Topic Model (D-ETM)		
Epochs	10	200	400
Min DoF	20	10	10
Model Perplexity	1935.5	1636.9	**1399.6**
Topic Quality	0.0536	0.1662	**0.3532**
Topic Diversity	0.3196	0.6359	**0.7436**
Topic Coherence	0.1677	0.2614	**0.4751**

Table 4.6: Model Performance of D-ETM on Cybersecurity News Articles. The Lower the Perplexity, the Better. Higher is Better for the Topic Quality, Topic Diversity and Topic Coherence.

For our experiment, we built a document vector embedding with 300 dimensions using Glove (Pennington et al., 2014). For the model, we employed D-ETM (Dieng et al., 2019; Dieng et al., 2020) which uses embedding

representations of words and topics. For learning the neural network model parameters, we utilized a batch size of 100 and used the Adaptive Moment Estimation (ADAM) algorithm for gradient descent (Kingma and Ba, 2014; Yi et al., 2020). Additionally, we constructed a feed-forward inference network that is fully connected with a ReLU activation function, consisting of two layers, each containing 800 hidden units, to handle topic proportions. We used the following hyperparameters; dropout rate of 1.0 to compute the mean and log variance, weight decay = 1.2e-6, learning rate = 0.01, number of bad hits allowed = 10, and randomseed = 2019. We ran the algorithm for multiple epochs (10, 200, 400) and output 50 topics, and 60 words for the corpus of document. The Table 4.6 is the evaluation performance of the D-ETM on the different epochs we used for model training. The main metrics are the Perplexity evaluated on the document completion task, the topic quality, topic coherence, and the topic diversity. The best results were gotten after we have trained the model for 400 epochs.

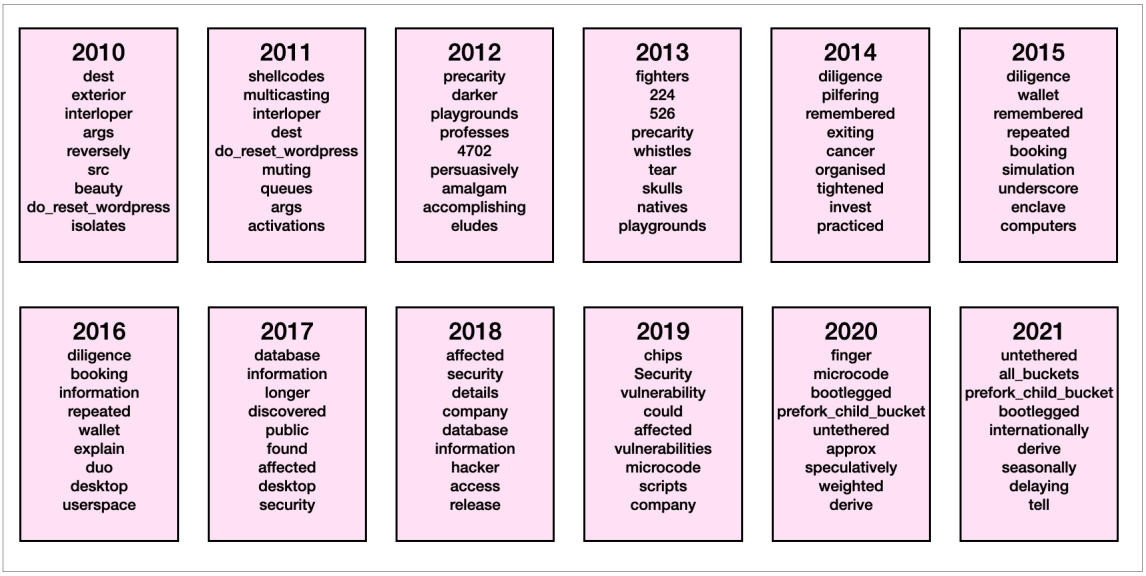

Figure 4.12: Image showing the time-dependent evolution of the top nine words from a "Malware" topic. The words were learned using the D-ETM algorithm.

4.4.1 Qualitative Test Case: Trend Analysis for Malware

In our qualitative use case, we verified that the identified evolution of topics over time aligns with significant historical events. For this uses case, we used a topic on "Malware" as shown in Figure 4.12. For example, the "526" word in 2013 represents The European Union Agency for Cybersecurity Regulation number, also known as ENISA. The ENISA-526 regulation was established in 2013 to address network and information security challenges. ENISA's goal is to provides advice, solutions, and expertise to meet network and information security requirements, supports cooperative responses to cross-border cybersecurity incidents, and develops cybersecurity certification schemes (Remac, 2017).

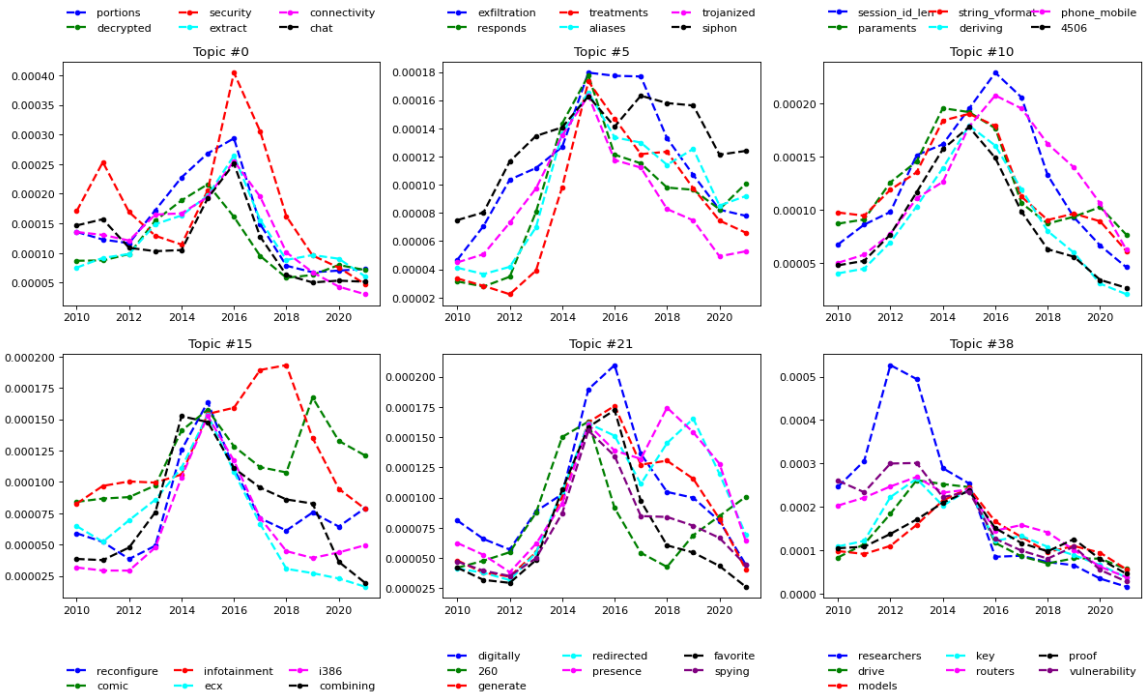

Figure 4.13: The evolution of word probabilities for six distinct topics.

The word "prefork-child-bucket" is a part of a malware structure used by attackers and discovered as a vulnerability in Apache HTTP Server software

in 2019 before the proliferation in 2020 and 2021 (Lu et al.,). In 2015 and 2016, malware attacks to steal private keys to access cryptocurrency wallets were on the increase and this is another example shown in the "malware" topic evolution (Hampton and Baig, 2015).

4.4.2 Hypothesis Three

H3: A dynamic topic model can identify 70% of the unique topics used to identify cybersecurity trends

The third hypothesis evaluates the performance of the dynamic topic model for identifying unique topics for identifying cybersecurity trends. The Table 4.7 shows the prediction statistics and hypothesis for this D-ETM model. The key statistics are the Perplexity, Topic Quality, Topic Coherence, and Topic Diversity. We focused on the Topic Diversity, which is the fraction of unique words found in the top 25 words of all topics. The higher the topic diversity, the less redundant the topics are (Lau et al., 2014). For the test, we used the one-sample proportion z-test to compare the sample's proportion to a known population proportion. The test was done using an alpha value of 0.05 ($\alpha = 0.05$) with a 70% Topic Diversity. A significance level of $\alpha = 0.05$

Prediction Statistics			
Model Perplexity	1399.6	Topic Quality	0.3532
Topic Coherence	0.4751	Topic Diversity	0.7436
One-Sample Proportion Z-Test			
$H_0 = 0.70$ $H_1 > 0.70$			
Sample Proportion	0.7436	95% Confidence Interval for Proportion	(0.7319, 0.7553)
Z-Statistic	6.9669	P-Value	0.0000

Table 4.7: Performance Statistics and Results of Hypothesis Test for Dynamic Embedded Topic Model for Identifying Trends from Cybersecurity News Articles with Unique Topics

was used for conducting this hypothesis test. The obtained table reveals that the topic diversity in the test set is 0.7436. The aim of this hypothesis test is to utilize the one-sample proportion Z-test to assess whether the model achieves a minimum topic diversity of 70% (0.70). With a P-value of 0.000, we reject the null hypothesis, which states that the dynamic topic model (D-ETM) is capable of identifying only 70% of the distinct topics used to analyze cybersecurity trends. Consequently, we can confidently conclude that the D-ETM adequately captures temporal-dependent trends in cybersecurity events.

4.5 Summary and Conclusion

We have developed two groups of quantitative models, namely deep learning algorithms and classical machine learning algorithms, to effectively classify cybersecurity event types. Furthermore, we have also implemented an unsupervised model known as the dynamic embedded topic model for identifying key trends from cybersecurity news articles. Throughout our praxis research, we collected results from various model training experiments and evaluated the predictive performance. The obtained results were analyzed, enabling us to gain valuable insights and contribute to the field of cybersecurity analysis.

In the first and second model, the null hypothesis was successfully rejected. This means that an explainable predictive model (Bi-LSTM) is sufficient enough to classify cybersecurity event types from news articles and that the deep learning based model performed better than the classical machine learning based method. For the first model, we introduced the concept of machine learning model explainability and used four different use cases to outline the post-hoc local explanations of the classifier predictions

using LIME. In the third model, we also successfully rejected the null hypothesis which implies that the D-ETM is sufficient enough for identifying temporal-dependent trends in cybersecurity events. The three models can help key cybersecurity stakeholders like cyber engineers and information officers to classifying large cybersecurity documents and news articles into key event types and identify the trends of cybersecurity events over time. A detailed discussion of the impact of this model results in the cybersecurity field will be presented in the next chapter.

Chapter 5: Discussion and Conclusions

5.1 Discussion

The primary objective of this praxis research is to investigate the effectiveness of deep learning and generative probabilistic models in classifying cybersecurity documents based on event types, while providing valuable insights into potential threats and corresponding solutions for cybersecurity professionals and stakeholders. To achieve this, two deep learning algorithms, namely the Bi-LSTM (based on recurrent architecture) and BERT (based on transformer architecture), along with the Dynamic Embedded Topic Model (D-ETM), were employed. The goal was to assess the models' capabilities in classifying documents and extracting unique cybersecurity topics with their associated word probabilities over time.

Both the Bi-LSTM and BERT models were successfully trained and evaluated using a testing dataset specifically set aside for evaluation purposes. The D-ETM model, on the other hand, effectively learned from the entire document corpus to identify unique topics and track their trends over time. In the following sections, we will delve into the conclusive parts about the models developed in this praxis, explaining their performance and providing insightful interpretations of the obtained results.

5.2 Conclusion

This praxis research aimed to enhance the ability of cybersecurity professionals to gain early and reliable insights into cybersecurity events. While organizations commonly rely on an incident response process to address

such events, manual detection can be challenging and time-consuming. To overcome this, text mining techniques have been proposed to monitor public data sources and identify relevant incidents (Ignaczak et al., 2021). However, traditional text mining approaches like k-Nearest Neighbors (k-NN) and SVM may lack generalizability.

To address these limitations, this research introduced a novel predictive model that utilizes large language models and D-ETM for topic modeling. By applying these advanced techniques, the model enables the identification of cybersecurity events and extraction of latent topics from publicly available cybersecurity documents (Le Sceller et al., 2017). This innovative approach provides cybersecurity engineers and stakeholders with an early view and valuable insights into cybersecurity events, facilitating the identification of trends, development of mitigation strategies, and proactive actions.

To summarize, we successfully developed and implemented two sets of quantitative models: deep learning algorithms and classical machine learning algorithms, with the objective of effectively classifying cybersecurity events into specific types, including 'Attack', 'Discover', 'Impact', and 'Patch'. Additionally, an unsupervised model called the D-ETM was introduced to uncover significant trends within cybersecurity news articles. Through rigorous experimentation and comprehensive evaluation, we have collected and meticulously analyzed the results obtained in this praxis research.

The first set of models demonstrated the efficacy of the explainable predictive model in accurately classifying cybersecurity event types found in news articles, surpassing the performance of traditional classical machine learning approaches. Moreover, to enhance the interpretability of the models, we introduced the concept of machine learning model explainability, providing post-hoc local explanations of classifier predictions using the LIME

method across various use cases. These explainable models offer valuable insights to cybersecurity professionals and stakeholders, enabling them to understand the underlying factors driving the classification decisions.

In the third model, D-ETM proved to be a reliable tool for identifying time-dependent trends within the realm of cybersecurity events. By analyzing large volumes of cybersecurity documents and news articles, the D-ETM model successfully revealed prevailing trends over time, empowering cybersecurity experts to gain a comprehensive understanding of the evolving landscape. These three models collectively provide essential support to key stakeholders in the cybersecurity field, including cybersecurity engineers and information officers, by facilitating the accurate classification of extensive datasets and the identification of crucial trends, ultimately enabling proactive decision-making and effective cybersecurity strategies.

5.3 Contributions to Body of Knowledge

This research contributes to the field of cybersecurity intelligence by leveraging advanced techniques such as large language models and dynamic embedded topic models. The developed models, including deep learning algorithms like Bi-LSTM and BERT, offer improved accuracy and efficiency in classifying cybersecurity event types from news articles, surpassing traditional machine learning approaches. This contribution enhances the understanding of cybersecurity events and enables organizations to make informed decisions and take proactive measures to protect sensitive information and mitigate potential risks.

Additionally, the introduction of the D-ETM provides valuable insights into the temporal dynamics of cybersecurity events by identifying key trends in cybersecurity news articles. This unsupervised model enhances the exist-

ing knowledge base by offering a data-driven approach to understanding the evolution of cybersecurity events over time. By detecting emerging patterns and trends, organizations can proactively respond to potential threats and adapt their cybersecurity strategies accordingly. This research not only advances the field of cybersecurity intelligence but also has practical implications for organizations seeking to enhance their cybersecurity intelligence capabilities.

5.4 Future Research Recommendation

Looking ahead, the future of this research path holds great potential for further advancements in the field of cybersecurity. One key area for improvement lies in the continuous enhancement of the existing cybersecurity events classification and topic extraction tasks. This can be achieved by exploring more diverse and contextual datasets that encompass a wider range of cybersecurity events and scenarios. By incorporating additional data sources, such as social media platforms, multilingual datasets, academic research reports on cybersecurity events, forums, and incident reports, researchers can obtain a more comprehensive understanding of cybersecurity events and improve the accuracy and generalizability of the classification models.

Future research in this field should explore various downstream natural language processing (NLP) tasks, such as text summarization and cybersecurity entity extraction, in addition to improving the classification models. The ability of cybersecurity professionals to quickly understand the main insights and take the right steps is made possible by text summarization, which plays a critical role in reducing large amounts of cybersecurity-related information into short summaries. Investigating cutting-edge Generative

Artificial Intelligence (GenAI) methods can present potential directions for creating summarized version of cybersecurity texts that accurately captures cybersecurity events.

Overall, the future outlook for this research path is defined by continuous improvement, exploration of diverse datasets from different data sources, and the application of more robust generative AI techniques for other downstream tasks. By addressing these key areas, researchers can advance the field of cybersecurity intelligence, empower organizations with more accurate and comprehensive insights, and contribute to a safer and more secure digital landscape.

Bibliography

Ahmed, M. F., Anwar, M. T., Tanvir, S., Saha, R., and Shoumo, S. Z. H. (2021). Cybersecurity news article dataset. *Mendeley Data*, V1.

Amini, H., Farahnak, F., and Kosseim, L. (2019). Natural language processing: an overview. *Frontiers in Pattern Recognition and Artificial Intelligence*, pages 35–55.

Anandarajan, M., Hill, C., and Nolan, T. (2019). Practical text analytics. *Maximizing the Value of Text Data.(Advances in Analytics and Data Science. Vol. 2.) Springer*, pages 45–59.

Aydin, Z. E. and Ozturk, Z. K. (2021). Performance analysis of xgboost classifier with missing data. *Manchester Journal of Artificial Intelligence and Applied Sciences (MJAIAS)*, 2(02):2021.

Bendovschi, A. (2015). Cyber-attacks–trends, patterns and security countermeasures. *Procedia Economics and Finance*, 28:24–31.

Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Berrar, D. (2018). Bayes' theorem and naive bayes classifier. *Encyclopedia of bioinformatics and computational biology: ABC of bioinformatics*, 403:412.

Bhattacharya, S., Maddikunta, P. K. R., Kaluri, R., Singh, S., Gadekallu, T. R., Alazab, M., and Tariq, U. (2020). A novel pca-firefly based xgboost classification model for intrusion detection in networks using gpu. *Electronics*, 9(2):219.

Bird, S. (2006). Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Blei, D. M. and Lafferty, J. D. (2006). Dynamic Topic models. *In Proceedings of the 23rd international conference on Machine learning*, (1):113–120.

Briskilal, J. and Subalalitha, C. (2022). An ensemble model for classifying idioms and literal texts using bert and roberta. *Information Processing & Management*, 59(1):102756.

Chen, T. and Guestrin, C. (2015). Xgboost: Reliable large-scale tree boosting system. In *Proceedings of the 22nd SIGKDD Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA*, pages 13–17.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

Chi, R., Wu, B., and Wang, L. (2018). Expert identification based on dynamic lda topic model. In *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, pages 881–888. IEEE.

Collier, Z. A., DiMase, D., Walters, S., Tehranipoor, M. M., Lambert, J. H., and Linkov, I. (2014). Cybersecurity standards: Managing risk and creating resilience. *Computer*, 47(9):70–76.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dieng, A. B., Ruiz, F. J., and Blei, D. M. (2020). Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Dieng, A. B., Ruiz, F. J. R., and Blei, D. M. (2019). The dynamic embedded topic model. *arXiv preprint arXiv:1907.05545*, pages 35–45.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7):1895–1923.

Doshi-Velez, F. and Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Edwards, A. L. (1948). Note on the "correction for continuity" in testing the significance of the difference between correlated proportions. *Psychometrika*, 13(3):185–187.

Eisenstein, J. (2019). Introduction to natural language processing. adaptive computation and machine learning series. *MIT Press*, 55:56.

Eling, M., McShane, M., and Nguyen, T. (2021). Cyber risk management: History and future research directions. *Risk Management and Insurance Review*, 24(1):93–125.

Garner, R. L. (2017). Evaluating solutions to cyber attack breaches of health data: How enacting a private right of action for breach victims would lower costs. *Ind. Health L. Rev.*, 14:127.

Georgescu, T.-M. (2020). Natural language processing model for automatic analysis of cybersecurity-related documents. *Symmetry*, 12(3):354.

Ghasiya, P. and Okamura, K. (2021). Investigating cybersecurity news articles by applying topic modeling method. In *2021 International Conference on Information Networking (ICOIN)*, pages 432–438. IEEE.

Ghasiya, P. and Okamura, K. (2022). A hybrid approach to analyze cybersecurity news articles by utilizing information extraction & sentiment analysis methods. *International Journal of Semantic Computing*, 16(01):135–160.

Gomez, J., Sensato, C., Konschak, C., and Divurgent, C. (2015). Cybersecurity in healthcare.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

Greene, D. and Cross, J. P. (2017). Exploring the political agenda of the european parliament using a dynamic topic modeling approach. *Political Analysis*, 25(1):77–94.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., et al. (2018). Recent advances in convolutional neural networks. *Pattern recognition*, 77:354–377.

Hampton, N. and Baig, Z. A. (2015). Ransomware: Emergence of the cyber-extortion menace.

Hapke, H., Howard, C., and Lane, H. (2019). *Natural Language Processing in Action: Understanding, analyzing, and generating text with Python*. Simon and Schuster.

Hardeniya, N. (2015). *NLTK essentials*. Packt Publishing.

Hawdon, J. (2021). Cybercrime: Victimization, perpetration, and techniques. *American Journal of Criminal Justice*, 46(6):837–842.

Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., and Scholkopf, B. (1998). Support vector machines. *IEEE Intelligent Systems and their applications*, 13(4):18–28.

Hughes, J. and Cybenko, G. (2013). Quantitative metrics and risk assessment: The three tenets model of cybersecurity. *Technology Innovation Management Review*, 3(8).

Ignaczak, L., Goldschmidt, G., Costa, C. A. D., and Righi, R. D. R. (2021). Text mining in cybersecurity: A systematic literature review. *ACM Computing Surveys (CSUR)*, 54(7):1–36.

Ishii, M., Matsuura, S., Mori, K., Tomoishi, M., Jin, Y., and Kitaguchi, Y. (2020). A study of classification of texts into categories of cybersecurity incident and attack with topic models. In *ICISSP*, pages 639–646.

Kamps, J., Kondylidis, N., Rau, D., et al. (2020). Impact of tokenization, pretraining task, and transformer depth on text ranking. In *TREC*.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lachenbruch, P. A. (2014). Mcnemar test. *Wiley StatsRef: Statistics Reference Online*.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Lau, J. H., Newman, D., and Baldwin, T. (2014). Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 530–539.

Le Sceller, Q., Karbab, E. B., Debbabi, M., and Iqbal, F. (2017). Sonar: Automatic detection of cyber security events over the twitter stream. In *Proceedings of the 12th International Conference on Availability, Reliability and Security*, pages 1–11.

Li, Z., Liu, F., Yang, W., Peng, S., and Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*.

Lipton, Z. C. (2018). The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57.

Liu, G. and Guo, J. (2019). Bidirectional lstm with attention mechanism and convolutional layer for text classification. *Neurocomputing*, 337:325–338.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Lu, Y., Shiveley, R., and Ruby, J. Scaling the performance of apache web server on intel®-based platforms. *White Paper Apache Web server Open Source Software Solutions, Intel*.

Marinho, R. and Holanda, R. (2023). Automated emerging cyber threat identification and profiling based on natural language processing. *IEEE Access*.

Medsker, L. R. and Jain, L. (2001). Recurrent neural networks. *Design and Applications*, 5:64–67.

Meszaros, J. and Buchalcevova, A. (2017). Introducing ossf: A framework for online service cybersecurity risk management. *computers & security*, 65:300–313.

Mikolov, T., Karafiát, M., Burget, L., Cernockỳ, J., and Khudanpur, S. (2010). Recurrent neural network based language model. In *Interspeech*, volume 2, pages 1045–1048. Makuhari.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.

Mimno, D., Wallach, H., Talley, E., Leenders, M., and McCallum, A. (2011). Optimizing semantic coherence in topic models. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pages 262–272.

Molnar, C. (2020). *Interpretable machine learning*. Lulu. com.

Morimoto, T. and Kawasaki, Y. (2017). Forecasting financial market volatility using a dynamic topic model. *Asia-Pacific financial markets*, 24:149–167.

Naseer, A., Naseer, H., Ahmad, A., Maynard, S. B., and Siddiqui, A. M. (2021). Real-time analytics, incident response process agility and enterprise cybersecurity performance: A contingent resource-based analysis. *International Journal of Information Management*, 59:102334.

Onwubiko, C. and Ouazzane, K. (2020). SOTER: A Playbook for Cybersecurity Incident Management. *IEEE Transactions on Engineering Management*.

Paté-Cornell, M.-E. and Kuypers, M. A. (2021). A probabilistic analysis of cyber risks. *IEEE Transactions on Engineering Management*, 70(1):3–13.

Patel, R. and Patel, S. (2021). Deep learning for natural language processing. In *Information and Communication Technology for Competitive Strategies (ICTCS 2020) Intelligent Strategies for ICT*, pages 523–533. Springer.

Pati, S., Aga, S., Jayasena, N., and Sinclair, M. D. (2022). Demystifying bert: System design implications. In *2022 IEEE International Symposium on Workload Characterization (IISWC)*, pages 296–309. IEEE.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Ravi, V., Alazab, M., Srinivasan, S., Arunachalam, A., and Soman, K. P. (2021). Adversarial Defense: DGA-Based Botnets and DNS Homographs Detection Through Integrated Deep Learning. *IEEE Transactions on Engineering Management*.

Remac, M. (2017). The european union agency for network and information security (enisa)-regulation (eu) no 526/2013 of the european parliament and of the council of 21 may 2013 concerning the european union agency for network and information security (enisa).

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Rish, I. et al. (2001). An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46.

Rogers, A., Kovaleva, O., and Rumshisky, A. (2021). A primer in bertology: What we know about how bert works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Rusk, N. (2016). Deep learning. *Nature Methods*, 13(1):35–35.

Sarker, I. H., Furhad, M. H., and Nowrozy, R. (2021). Ai-driven cybersecurity: an overview, security intelligence modeling and research directions. *SN Computer Science*, 2:1–18.

Satyapanich, T., Ferraro, F., and Finin, T. (2020). Casie: Extracting cybersecurity event information from text. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8749–8757.

Sha, H., Hasan, M. A., Mohler, G., and Brantingham, P. J. (2020). Dynamic topic modeling of the covid-19 twitter narrative among us governors and cabinet executives. *arXiv preprint arXiv:2004.11692*.

Shankar, N. and Mohammed, Z. (2020). Surviving data breaches: A multiple case study analysis. *Journal of Comparative International Management*, 23(1):35–54.

Shilong, Z. et al. (2021). Machine learning model for sales forecasting by using xgboost. In *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pages 480–483. IEEE.

Silge, J. and Robinson, D. (2017). *Text mining with R: A tidy approach*. " O'Reilly Media, Inc.".

Simmons, C., Ellis, C., Shiva, S., Dasgupta, D., and Wu, Q. (2009). Avoidit: A cyber attack taxonomy. *University of Memphis, Technical Report CS-09-003*.

Sleeman, J., Finin, T., and Halem, M. (2021). Understanding cybersecurity threat trends through dynamic topic modeling. *Frontiers in Big Data*.

Song, X., Salcianu, A., Song, Y., Dopson, D., and Zhou, D. (2021). Fast wordpiece tokenization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2089–2103.

Sra, S. and Dhillon, I. (2005). Generalized nonnegative matrix approximations with bregman divergences. *Advances in neural information processing systems*, 18.

Stein, R. A., Jaques, P. A., and Valiati, J. F. (2019). An analysis of hierarchical text classification using word embeddings. *Information Sciences*, 471:216–232.

Steyvers, M., Smyth, P., Rosen-Zvi, M., and Griffiths, T. (2004). Probabilistic author-topic models for information discovery. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 306–315.

Tan, C.-M., Wang, Y.-F., and Lee, C.-D. (2002). The use of bigrams to enhance text categorization. *Information processing & management*, 38(4):529–546.

Trong, H. M. D., Le, D. T., Veyseh, A. P. B., Nguyen, T., and Nguyen, T. H. (2020). Introducing a new dataset for event detection in cybersecurity texts. *In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Ukwen, D. O. and Karabatak, M. (2021). Review of nlp-based systems in digital forensics and cybersecurity. In *2021 9th International symposium on digital forensics and security (ISDFS)*, pages 1–9. IEEE.

Ullman, J. (2011). *Mining of massive datasets*. Cambridge University Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wallach, H. M., Murray, I., Salakhutdinov, R., and Mimno, D. (2009). Evaluation methods for topic models. In *Proceedings of the 26th annual international conference on machine learning*, pages 1105–1112.

Wang, C., Blei, D., and Heckerman, D. (2012). Continuous time dynamic topic models. *arXiv preprint arXiv:1206.3298*.

Wang, X. and Lu, X. (2020). A host-based anomaly detection framework using xgboost and lstm for iot devices. *Wireless Communications and Mobile Computing*, 2020:1–13.

Xu, G., Meng, Y., Qiu, X., Yu, Z., and Wu, X. (2019). Sentiment analysis of comment texts based on bilstm. *Ieee Access*, 7:51522–51532.

Yao, F. and Wang, Y. (2020). Tracking urban geo-topics based on dynamic topic model. *Computers, Environment and Urban Systems*, 79:101419.

Yi, D., Ahn, J., and Ji, S. (2020). An effective optimization method for machine learning based on adam. *Applied Sciences*, 10(3):1073.

Zhu, Y., Lu, X., Hong, J., and Wang, F. (2022). Joint dynamic topic model for recognition of lead-lag relationship in two text corpora. *Data Mining and Knowledge Discovery*, 36(6):2272–2298.

Zou, J., Han, Y., and So, S.-S. (2009). Overview of artificial neural networks. *Artificial neural networks: methods and applications*, pages 14–22.