

# ScholarWorks@GSU

## Artificial Intelligence Approaches for Financial Cybercrime Information Analysis

Item Type	Dissertation
Authors	Gao, Chunlan
Citation	Gao, Chunlan. "Artificial Intelligence Approaches for Financial Cybercrime Information Analysis." PhD diss., Georgia State University, 2025. <a href="https://doi.org/10.57709/r00t-bs40">https://doi.org/10.57709/r00t-bs40</a>
DOI	<a href="https://doi.org/10.57709/r00t-bs40">https://doi.org/10.57709/r00t-bs40</a>
Download date	2026-05-08 11:37:45
Link to Item	<a href="https://hdl.handle.net/20.500.14694/15853">https://hdl.handle.net/20.500.14694/15853</a>

Artificial Intelligence Approaches for Financial Cybercrime Information Analysis

by

Chunlan Gao

Under the Direction of Yubao Wu, Ph.D.

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2025

## ABSTRACT

Artificial Intelligence (AI) has become an indispensable tool in combating cybercrime. Through machine learning and deep learning techniques, AI systems can automatically detect fraudulent activities by recognizing patterns, identifying anomalies, and predicting potential security threats. While most existing research on financial fraud focuses on structured datasets such as transaction records and financial statements, this dissertation targets a more complex and challenging data source—financial fraud-related content on Telegram. Telegram data is loosely formatted, combining structured patterns (e.g., hashtags, prices) with unstructured text, slang, emojis, and embedded images, which poses unique challenges for automated analysis and classification.

This dissertation presents four interrelated studies on AI-driven financial fraud detection. The first introduces AutoCut-2D, a feature selection method that adaptively determines cut-off thresholds across multiple dimensions of feature importance, improving prediction accuracy and significantly reducing computational cost. The second study focuses on fraud category classification of Telegram messages, comparing diverse embedding and machine learning techniques to enhance model performance. The third extends this research into multimodal learning, integrating BERT-based textual embeddings with Swin Transformer-based visual embeddings through attention-based fusion, achieving substantially higher accuracy than using either modality alone in identifying fraudulent advertisements.

The fourth study advances the research toward Knowledge Base Construction (KBC) from Telegram messages. A weakly supervised extraction pipeline is proposed to derive structured triples such as (brand, original price, discount price) from loosely formatted content using a combination of rule-based heuristics and machine learning methods. This KBC framework effectively bridges the gap between unstructured communication and structured financial intelligence, providing a scalable and interpretable foundation for automated cy-

bercrime investigation and future knowledge-driven fraud analysis.

INDEX WORDS: Cybercrime, Multimodal, Data Imbalance, Telegram, Financial Frauds, Knowledge Base Construction, BERT, TF-IDF

Copyright by  
Chunlan Gao  
2025

Artificial Intelligence Approaches for Financial Cybercrime Information Analysis

by

Chunlan Gao

Committee Chair:

Yubao Wu

Committee:

Xiaojun Cao

Kai Qian

Rajshekhar Sunderraman

Electronic Version Approved:

Office of Graduate Services

College of Arts and Sciences

Georgia State University

12 2025

## DEDICATION

I sincerely thank my advisor, Professor Yubao Wu, for his insightful guidance, continuous support, and invaluable feedback throughout the preparation of this dissertation. I am deeply grateful for his mentorship, patience, and encouragement, which have profoundly shaped my research and professional growth. I also wish to express my heartfelt appreciation to my family—my parents, sister, husband, and children—for their unwavering love, understanding, and encouragement. Their constant support has been the driving force that keeps me moving forward.

## ACKNOWLEDGMENTS

I express my appreciation to my advisors, committee members, and lab members for their support and assistance throughout my PhD journey.

## TABLE OF CONTENTS

ACKNOWLEDGMENTS . . . . .		v
LIST OF TABLES . . . . .		ix
LIST OF FIGURES . . . . .		x
<b>1 CHAPTER 1 INTRODUCTION . . . . .</b>		<b>1</b>
1.1 Improving Predictive Performance and Reducing Computational Time in Feature Selection . . . . .		5
1.2 Category Prediction of Financial Fraud using messages in Telegram		6
1.3 Recognizing the Financial Fraud Type of Telegram Advertisement Posts Based on the Multimodal Learning . . . . .		8
1.4 Weakly Supervised Knowledge Base Construction for Telegram Gift- Card Fraud Messages . . . . .		10
<b>2 CHAPTER 2 IMPROVING PREDICTIVE PERFORMANCE AND RE- DUCING COMPUTATIONAL TIME IN FEATURE SELECTION . . . . .</b>		<b>13</b>
2.1 Introduction . . . . .		13
2.2 Related Work . . . . .		15
2.3 Experiments and Results . . . . .		21
2.3.1 <i>Dataset Specification</i> . . . . .		21
2.3.2 <i>Feature Selection Methods</i> . . . . .		23
2.3.3 <i>Performance Measure</i> . . . . .		27
2.3.4 <i>Results</i> . . . . .		27
2.4 Conclusion . . . . .		34
<b>3 CHAPTER 3 CATEGORY PREDICTION OF FINANCIAL FRAUD USING MESSAGES IN TELEGRAM . . . . .</b>		<b>37</b>
3.1 Introduction . . . . .		37

3.2	Related Work . . . . .	38
3.3	Methodology Overview . . . . .	40
	3.3.1 <i>Data Collection and Extraction</i> . . . . .	41
	3.3.2 <i>Feature Extraction Methods</i> . . . . .	44
	3.3.3 <i>Classification Models</i> . . . . .	45
3.4	Experiment and Results . . . . .	49
	3.4.1 <i>Dataset Collection</i> . . . . .	49
	3.4.2 <i>Data Preprocessing</i> . . . . .	50
	3.4.3 <i>Experiment Results</i> . . . . .	53
3.5	Conclusion . . . . .	56
4	<b>CHAPTER 4 RECOGNIZING THE FINANCIAL FRAUD TYPE OF TELEGRAM ADVERTISEMENT POSTS BASED ON THE MULTI-MODAL LEARNING</b> . . . . .	<b>57</b>
4.1	Introduction . . . . .	57
4.2	Related Work . . . . .	60
	4.2.1 <i>Multimodal Fusion Techniques</i> . . . . .	61
	4.2.2 <i>Traditional Machine Learning Approaches</i> . . . . .	62
	4.2.3 <i>Deep Learning Models</i> . . . . .	63
4.3	Methodology Overview . . . . .	64
	4.3.1 <i>Model Architecture</i> . . . . .	64
	4.3.2 <i>Fusion Strategies</i> . . . . .	66
	4.3.3 <i>Training and Evaluation</i> . . . . .	67
4.4	Experiments and Results . . . . .	68
	4.4.1 <i>Dataset</i> . . . . .	68
	4.4.2 <i>Implementation Details</i> . . . . .	69
	4.4.3 <i>Performance Comparison</i> . . . . .	70
	4.4.4 <i>Misclassification Analysis in Swin Transformer + BERT Model</i> . . . . .	73
4.5	Conclusion . . . . .	74

<b>5</b>	<b>WEAKLY SUPERVISED KNOWLEDGE BASE CONSTRUCTION FOR TELEGRAM GIFT-CARD FRAUD MESSAGES . . . . .</b>	<b>75</b>
5.1	Introduction . . . . .	75
5.2	Related Work . . . . .	78
5.3	Methodology . . . . .	80
	<i>5.3.1 System Overview . . . . .</i>	80
	<i>5.3.2 Telegram Parsing and Candidate Generation . . . . .</i>	80
	<i>5.3.3 Refined Weak Labeling . . . . .</i>	81
	<i>5.3.4 Triple Export for Knowledge Base Construction . . . . .</i>	82
5.4	Experimental and Results . . . . .	83
	<i>5.4.1 Dataset . . . . .</i>	83
	<i>5.4.2 Models and Features . . . . .</i>	83
	<i>5.4.3 Detailed Feature Engineering . . . . .</i>	83
	<i>5.4.4 Model Comparison . . . . .</i>	85
	<i>5.4.5 Human–Model Agreement . . . . .</i>	85
	<i>5.4.6 Interpretation and Knowledge Base Analysis . . . . .</i>	85
5.5	Conclusions . . . . .	87
<b>6</b>	<b>CONCLUSION AND FUTURE WORK . . . . .</b>	<b>90</b>
6.1	Summary of Research . . . . .	90
6.2	Major Findings and Contributions . . . . .	91
6.3	Discussion and Implications . . . . .	91
6.4	Conclusion . . . . .	93
<b>7</b>	<b>PUBLICATIONS . . . . .</b>	<b>94</b>
	<b>REFERENCES . . . . .</b>	<b>95</b>

**LIST OF TABLES**

Table 2.1	Result Comparison . . . . .	28
Table 3.1	Seconds for different feature extraction methods . . . . .	53
Table 4.1	Performance Compare . . . . .	72
Table 5.1	Comparison with Representative Approaches in Financial-Fraud and Telegram Analysis. . . . .	79
Table 5.2	Dataset Statistics for Telegram Gift-Card Messages . . . . .	83
Table 5.3	Typical Extracted Discount Ranges by Brand (Selected Examples) . .	86

## LIST OF FIGURES

Figure 2.1	Ransomedata Distribution . . . . .	22
Figure 2.2	Telegram Distribution . . . . .	23
Figure 2.3	Feature Importance With Random Forest . . . . .	28
Figure 2.4	Feature importance selection with Elbow Method Visually . . . . .	29
Figure 2.5	Training time comparison across models and strategies. . . . .	30
Figure 2.6	Accuracy comparison across models and strategies. . . . .	31
Figure 2.7	Accuracy Comparison: MI + 2D Elbow vs. MI + Kneed Elbow . . . . .	32
Figure 2.8	Accuracy and training time for RFE (top) and SFS (bottom) across selected feature dimensions. . . . .	34
Figure 3.1	The Diagram of the Proposed Method. . . . .	41
Figure 3.2	The Category on Web. . . . .	43
Figure 3.3	The schemas of the two tables. . . . .	43
Figure 3.4	The Data Distribution . . . . .	51
Figure 3.5	The Data Distribution after Pre-processing . . . . .	52
Figure 3.6	Performance Analysis . . . . .	54
Figure 3.7	Confusion Matrix for NN . . . . .	55
Figure 4.1	Model Architecture . . . . .	65
Figure 4.2	Early Fusion method . . . . .	66
Figure 4.3	Text Message From Criminal . . . . .	68
Figure 4.4	Image From Criminal . . . . .	69
Figure 4.5	Data Distribution . . . . .	70
Figure 4.6	Performance of different Configurations . . . . .	72

Figure 4.7	Example of correct and mistake . . . . .	73
Figure 5.1	Overall pipeline of the Weakly Supervised KBC system. . . . .	80
Figure 5.2	Performance comparison of structure-only, semantic-only, and fusion features. . . . .	84
Figure 5.3	Human-model agreement versus weak-label threshold. . . . .	86

## CHAPTER 1

### CHAPTER 1 INTRODUCTION

In the digital era, the proliferation of financial transactions on online platforms has significantly increased the exposure of individuals and institutions to fraud-related threats Ratna et al. (2024). Financial fraud, including, but not limited to, credit card scams, bank account drops, identity theft, phishing, money laundering, gift card abuse, and promo code exploitation, has become one of the most prevalent and damaging forms of cyber-enabled crime Barker et al. (2008). Unlike other types of cybercrimes, financial fraud directly targets monetary assets and exploits trust in digital financial infrastructures Wronka (2023). These activities often result in severe economic losses, reputational damage, and a growing mistrust in online financial systems Reurink (2019). As fraudsters continuously adapt their tactics, there is an urgent need for intelligent, adaptive, and scalable solutions to detect, analyze, and prevent financial fraud.

In recent years, one of the most pressing challenges in this domain is that cyber criminals are using instant messaging applications to spread financial fraud Shehabat et al. (2017); Europol (2017). Those instant messaging applications include Telegram, Signal, WhatsApp, Wickr, and Discord. They support real-time communication and enable users to send texts, images, and files to individuals or groups. Those applications have been widely used by cybercriminals to spread fraudulent content under the protection of encryption and anonymity.

Telegram has become a popular tool for cybercriminals to share stolen financial data, promote illegal services, and communicate anonymously with potential clients or accom-

plices Soudijn & Zegers (2012). It hosts a large number of public and private channels where financial fraud advertisements are posted in various formats, including texts, promotional banners, screenshots of fraudulent transactions, and fabricated identification documents Garkava et al. (2024). The semi-open nature of Telegram enables fraudulent actors to target global audiences while evading detection. As a result, the platform has evolved into a rich yet challenging source of financial fraud data. Given its rapid adoption by fraudsters and the growing volume of illicit activity, studying financial fraud on Telegram has become an urgent necessity. This raises the question of how to effectively analyze and extract useful information from such data.

Artificial Intelligence (AI) has emerged as a transformative force in combating financial fraud Bello & Olufemi (2024). Its ability to process vast and heterogeneous data sources and uncover subtle patterns makes it an ideal tool to detect fraudulent transactions, identify suspicious actors, and analyze communication patterns on digital platforms Udeh et al. (2024). AI methods such as machine learning (ML), deep learning (DL), natural language processing (NLP), and computer vision have already shown significant promise in financial contexts Verma & Pandiya (2024). Consequently, AI-based methods are particularly suitable for extracting and analyzing such information Papasavva et al. (2024).

Most existing research on financial fraud detection concentrates on structured data, such as financial statements, transactional logs, and bank account records. These data sets follow well-defined schemas, which makes them suitable for conventional machine learning and deep learning techniques. For example, in the work by Craja et al. (2020), the authors

utilized structured financial ratios and managerial commentary as input features for deep neural networks to detect anomalies in company financial reports. Similarly, a systematic review of the literature by Hernandez Aros et al. (2024) examined more than 100 articles and concluded that structured data, particularly financial disclosures, dominate the fraud detection landscape. Fissette (2017) analyzed 1,727 annual reports and identified fraud in 402 of them.

Although most studies rely on structured data, a smaller subset of research has explored the use of unstructured data, such as posts on social networks and online communications, for fraud detection, recognizing its potential to reveal behavioral patterns and contextual signals not captured in traditional data sets Isson (2018). So, AI techniques are a good choice for information extraction and analysis Papasavva et al. (2024). Recent advancements in multimodal Artificial intelligence have significantly improved the detection of fraud schemes by effectively integrating both textual and visual cues Bello & Olufemi (2024). For example, the integration of contextualized text embeddings (e.g., BERT) with visual features extracted from models such as Vision Transformer (ViT) allows systems to simultaneously evaluate linguistic and visual evidence Wang et al. (2023). Attention-based fusion techniques enhance performance by assigning dynamic weights to each modality according to its relevance to the detection task Gao et al. (2019).

In contrast, our work focuses on semi-structured data derived from Telegram, an encrypted messaging platform increasingly exploited by cybercriminals to disseminate financial fraud content. Unlike structured databases, Telegram messages combine structured metadata

(e.g., timestamps, sender IDs) with highly unstructured components such as normal text, slang-filled text, emojis, and embedded images. This hybrid nature poses unique challenges for automated fraud detection, requiring customized preprocessing, feature extraction, and classification strategies. In addition to textual content and images, Telegram fraud messages often exhibit non-linguistic structural features that convey implicit meaning or emphasis. These include table-like layouts (e.g., columns and rows), font casing (e.g., UPPER CASE for urgency), use of emojis, alignment, font types and colors, as well as special symbols or grading scales (e.g., “A++” or “AAA”). Such stylistic elements can serve as visual signals of credibility, urgency, or category, and are critical for accurately interpreting and structuring fraud-related information.

In this dissertation, we present four interrelated studies on AI-driven financial fraud analysis:

- Improving Predictive Performance and Reducing Computational Time in Feature Selection.
- Category Prediction of Financial Fraud using messages in Telegram.
- Recognizing the type of financial fraud in Telegram advertisement posts based on multimodal learning.
- Weakly supervised knowledge base construction for Telegram gift-card fraud messages.

## 1.1 Improving Predictive Performance and Reducing Computational Time in Feature Selection

Feature selection is a fundamental preprocessing step in machine learning, especially when dealing with high-dimensional data sets such as those derived from text-based financial fraud messages Kumar & Minz (2014); Khalid et al. (2014). It aims to reduce noise, improve learning efficiency, and enhance model interpretability by selecting the most informative features from a larger set of candidates Li et al. (2017). Traditional methods such as Mutual Information (MI) and Chi-square ( $\text{Chi}^2$ ) are commonly used to score and rank features based on their relevance to the target variable. In addition, tree-based models—such as Random Forest and XGBoost—provide built-in mechanisms for evaluating feature importance based on criteria like information gain or Gini impurity reduction. These models have been effectively used in cybersecurity applications such as ransomware detection to identify discriminative features Al-Khater et al. (2020); Moore et al. (2017); Lyu et al. (2023).

These methods typically require a manually defined cutoff point (e.g., selecting the top- $k$  features Rajbahadur et al. (2021); Zhang et al. (2020), using a fixed threshold Donoho & Jin (2008), which may lead to suboptimal performance or require empirical tuning. Sometimes, researchers use forward selection and backward elimination Sutter & Kalivas (1993) or recursive feature elimination Granitto et al. (2006); Chen & Jeong (2007) to get the Feature sets. From these methods, researchers can get good accuracy without empirical tuning. All these methods need to repeatedly train and evaluate the machine learning model at each step of the iteration. These methods, however, are often computationally expensive and

time-consuming, particularly when applied to high-dimensional datasets Khoshgoftaar et al. (2014).

To address this problem, we have previously employed the 'Elbow' method, which visually identifies a turning point in the feature importance curve to find the optimal "elbow" point based on curvature analysis. But this method also needs the user to find the cutpoint visually, which may be sensitive to the smoothness or scale of the score distribution.

In our work we developed an AutoCut-2D method, which is a data-driven and fully automated feature selection strategy that determines the cutoff point by analyzing the distribution of feature importance scores in a two-dimensional projection. AutoCut-2D integrates feature ranking and selection into a unified framework, eliminating the need for heuristic thresholding or manual intervention. It is particularly well-suited for the semi-structured and noisy nature of Telegram fraud data, where thousands of sparse textual features can obscure meaningful patterns. Our experiments demonstrate that AutoCut-2D not only improves classification performance but also reduces computational cost by selecting a minimal yet highly effective subset of features.

## **1.2 Category Prediction of Financial Fraud using messages in Telegram**

Telegram is a cross-platform instant messaging application Sutikno et al. (2016). It offers end-to-end encryption to ensure messages remain private during transmission and storage. Users can create and join various chat groups, including private chats, group chats, and public channels, making them suitable for organizations and social media publishers. Telegram has

a relatively loose censorship policy. That is, it only censors contents related to terrorism or child porn. Therefore, Telegram has become a popular platform for vendors to advertise illicit goods.

Financial Fraud-related products are one of the illicit goods Zhu et al. (2021). As we know, financial fraud has a crucial impact on all users around the world. So, our research focuses on the analysis of financial fraud-related posts to protect individuals and organizations and maintain the integrity of financial systems. While most existing studies focus on determining whether a post involves financial fraud, they do not delve into the subtypes of fraud—yet each subtype has its modus operandi and consequences Bozhenko et al. (2022). For example, credit-card scams often rely on stolen credentials to make unauthorized purchases, gift card fraud lures consumers with deeply discounted vouchers that erode trust, and money laundering schemes seek to obscure the origin of illegally obtained funds, posing systemic risks Hashim et al. (2020); Suh et al. (2019). Our research goes beyond binary fraud detection by performing fine-grained classification of financial fraud-related posts, enabling tailored interventions to protect individuals and organizations and to maintain the integrity of financial systems.

Financial fraud-related products include various types, such as credit card (CVV dumps), bank account drops, and gift card fraud. Good classification of these frauds can help assist in online surveillance. However, due to the limited research on the telegram, there are not enough datasets for us to do the research. So, we decided to collect the data and label it manually. The purpose of this study is to train effective machine-learning models using the

data we have collected and organized.

Our main contributions are:

I) We collect the post data from more than ten Telegram channels. We manually categorized them into six classes related to Financial Fraud following the "We The North" website.

II) We develop and test different machine learning methods to classify the posts in the datasets. We obtain a high level of accuracy using the developed machine learning methods.

### **1.3 Recognizing the Financial Fraud Type of Telegram Advertisement Posts Based on the Multimodal Learning**

Upon examining various Telegram channels, we found that criminals rely not just on text-based messages; they frequently post images containing sensitive details such as credit card numbers, checks, and bank account information. In many cases, these images are accompanied by text that explains or expands upon the visual content—criminals might provide instructions alongside photos of forged checks, for example, or offer clarifications on how to use stolen credit card data. Given this close interweaving of textual and visual information, purely text-based classification methods can no longer keep pace. To accurately detect and address such fraudulent activities, it is essential to adopt a more comprehensive approach that accounts for both text and images in tandem. The multimodal approaches are therefore highly suited to our research. By 'multimodal' we refer to methods that integrate data from multiple sources, such as text and images, in the same task Ngiam et al. (2011). This enables a more comprehensive understanding and recognition of complex scenarios and content.

In this study, we compare two different multimodal fusion techniques for fraud classification: concatenation and attention-based fusion. The concatenation method combines text and image features into a single vector, treating both modalities equally without considering their relative importance Noreen et al. (2020). In contrast, the attention-based fusion method assigns different weights to the modalities, allowing the model to focus more on the most informative modality depending on the context Jiang et al. (2020). These two approaches are evaluated and compared across five categories of fraud: Credit Card (CVV Dumps), Bank Account Drops, Personal Information Fraud, Money Transfer Fraud, and Promo Abuse/Discount Fraud.

The goal of this paper is to determine the optimal model for classifying Telegram-based fraud by comparing different fusion techniques, image embedding methods, and their impacts on overall performance. Through this comparison, we aim to enhance the accuracy of fraud detection systems and provide insights into how various multimodal fusion strategies and image embedding approaches influence the effectiveness of fraud classification.

Building upon our initial exploration of text-only and image-only classification methods, our experiments revealed that multimodal approaches significantly enhance classification performance. By leveraging complementary information from both text and image modalities, the multimodal strategy addressed the limitations of single-modality methods, achieving superior results across all evaluated metrics.

The Swin Transformer outperformed traditional CNN-based models like ResNet-50, thanks to its hierarchical architecture and ability to model long-range dependencies. Fusion strate-

gies also played a crucial role, with attention-based techniques further improving performance by dynamically prioritizing salient features.

#### **1.4 Weakly Supervised Knowledge Base Construction for Telegram Gift-Card Fraud Messages**

Beyond classification tasks, a central objective of this dissertation is to advance the automated extraction of structured financial intelligence from the loosely formatted communications that characterize illicit activity on Telegram. Among these communications, gift-card fraud messages are particularly challenging: they frequently contain inconsistent typography, embedded emojis, non-standard currency tokens, fragmented multi-line structures, and multiple denominations and brands listed within a single post. These characteristics severely limit the applicability of conventional supervised information extraction methods, which rely heavily on large volumes of clean, well-annotated data. To address these challenges, we developed a weakly supervised knowledge base construction (KBC) framework tailored specifically for gift-card fraud advertisements circulating on Telegram. The system transforms raw HTML chat exports into structured discount tuples of the form (brand, original price, discounted price, discount rate), enabling a level of financial and behavioral insight not achievable with coarse-grained classification methods alone.

The proposed framework introduces several conceptual and technical contributions that address the inherent difficulties of weakly structured Telegram data. First, we designed a Telegram-specific candidate generation module capable of identifying potential price-discount relations across fragmented and non-contiguous lines. This module leverages reg-

ular expressions, structural proximity cues, and a comprehensive brand-alias dictionary to capture complex multi-denomination patterns present in underground gift-card markets. Second, we formulated a probabilistic weak-labeling scheme that integrates multiple heuristic dimensions—including validated price drops, discount-related keywords, context-aware rate plausibility checks, and structural cues such as emojis, capitalization, and same-line or cross-line alignment. This weak labeling process provides supervision without any manually annotated data, significantly reducing annotation cost while maintaining confidence in the extracted candidates.

Third, the framework incorporates a refinement classifier trained on both structural and semantic features, enabling improved discrimination between true and spurious discount relations. Structural features—such as emoji ratio, uppercase ratio, and line distance—emerged as highly predictive on their own, achieving strong performance ( $F1 \approx 0.89$ ). Weak labels demonstrated over 80% agreement with human judgment after threshold calibration, validating the reliability of the heuristic supervision signals. The final refined tuples generate a high-resolution knowledge base that reveals brand-specific discount patterns, denomination-dependent pricing behaviors, and realistic underground market trends. These structured outputs provide investigators and analysts with a detailed representation of illicit pricing dynamics, illustrating how weak supervision can overcome data sparsity, formatting inconsistencies, and annotation constraints common in cybercrime environments.

Taken together, this line of work demonstrates a scalable, domain-adaptable methodology for extracting structured financial knowledge from noisy, loosely formatted messaging

data. By showing that high-quality discount triples can be derived without relying on large annotated corpora, this framework fills a critical methodological gap in the study of underground economies on Telegram and forms a foundational component of this dissertation’s broader goal of advancing multimodal, weakly supervised analysis for financial cybercrime intelligence.

The remainder of this paper is organized as follows: Section 2 reviews the feature selection methods applied to Telegram-based financial fraud data. Section 3 summarizes the system and experimental results from our previous work, “Category Prediction of Financial Fraud Related Posts in Telegram Group Chats.” Section 4 introduces the multimodal learning approach described in “Recognizing the Financial Fraud Type of Telegram Advertisement Posts Based on Multimodal Learning,” highlighting how text–image fusion improves classification accuracy. Section 5 presents the weakly supervised knowledge base construction framework developed in “Weakly Supervised Knowledge Base Construction for Telegram Gift-Card Fraud Messages,” focusing on extracting structured triples from loosely formatted posts. Finally, Section 6 concludes the paper and outlines directions for future research.

## CHAPTER 2

### CHAPTER 2 IMPROVING PREDICTIVE PERFORMANCE AND REDUCING COMPUTATIONAL TIME IN FEATURE SELECTION

#### 2.1 Introduction

In today's digitized world, fraudulent activities have grown increasingly sophisticated and pervasive, leading to significant financial losses and eroded trust for financial institutions, e-commerce platforms, and individuals alike Daraojimba et al. (2023). To effectively identify and combat fraud, various advanced machine learning and data mining techniques are widely employed to build robust fraud detection models Al-Hashedi & Magalingam (2021). However, fraud datasets often present unique challenges, characterized by high dimensionality, massive data volumes, redundant features, and highly imbalanced class distributions. These characteristics pose considerable hurdles to model training and performance Thudumu et al. (2020).

One important fraudulent activity in our life is Ransomware. Ransomware threats are growing fast around the world nowadays Masum et al. (2022a). Ransomware attacks are targeting our infrastructure, business, industry, and everywhere. These attacks directly impact our daily lives, such as supply chains and the security of our Nation. We need to find ways to battle these attacks and detect them, prevent them from attacks, protect ourselves from ransomware, and reduce their cybersecurity risk Garg et al. (2018); Gonzalez & Hayajneh (2017); Faruk et al. (2022). The machine learning (ML) models can discover patterns in data sets to predict the future based on data in the past which helps tackle today's

data protection and cyber-attack prevention Masum et al. (2022b); Ryan (2021); Urooj et al. (2021). Similarly, another prominent and evolving threat is Telegram financial fraud, where malicious actors exploit the platform’s communication capabilities to orchestrate scams, phishing attempts, and illicit financial schemes La Morgia et al. (2021). Both ransomware and Telegram financial fraud underscore the urgent need for advanced analytical capabilities to identify and mitigate these evolving threats.

It’s against this backdrop that Feature Selection emerges as an indispensable step in the field of fraud detection Zhou et al. (2020). It’s not merely about reducing data dimensionality; more crucially, by identifying and retaining the most informative features—those that best differentiate between legitimate and fraudulent behaviors and distinguish different fraudulent activities—it dramatically enhances model accuracy, efficiency, and interpretability Linardatos et al. (2020).

Feature selection is a fundamental preprocessing step in machine learning, especially when dealing with high-dimensional datasets such as those derived from text-based financial fraud messages Kumar & Minz (2014); Khalid et al. (2014). It aims to reduce noise, improve learning efficiency, and enhance model interpretability by selecting the most informative features from a larger set of candidates Li et al. (2017). Traditional methods such as Mutual Information (MI) and Chi-square ( $\text{Chi}^2$ ) are commonly used to score and rank features based on their relevance to the target variable. In addition, tree-based models—such as Random Forest and XGBoost—provide built-in mechanisms for evaluating feature importance based on criteria like information gain or Gini impurity reduction. These models have

been effectively used in cybersecurity applications such as ransomware detection to identify discriminative features Al-Khater et al. (2020); Moore et al. (2017); Lyu et al. (2023).

In our study, we tried to evaluate and compare various feature selection techniques for dimensionality reduction within cybercrime-related datasets. The datasets we are using include:

- Ransomware numerical features, which were got from publicly available online sources.
- Telegram-based textual messages, gathered and curated by our research team.

Given the heterogeneous nature of these datasets—ranging from structured numerical indicators to unstructured text—we employ a diverse set of feature selection methods to reduce high-dimensional input spaces. The goal is to enhance classification performance while improving interpretability and computational efficiency. Through systematic evaluation, we highlight the trade-offs and effectiveness of filter, wrapper, and projection-based techniques across different cybercrime detection scenarios.

Session 2.2 describes related work, session 2.3 talks about the experiment and results, and session 2.4 discusses the major findings and summarizes the conclusion.

## **2.2 Related Work**

Feature selection and dimensionality reduction are crucial in machine learning, especially when dealing with large datasets like cybersecurity logs and online communication records Abdulhammed et al. (2019); Buczak & Guven (2015). They help to reduce noise, make

training more efficient, and improve how well models generalize Bolón-Canedo et al. (2015, 2016).

There are three main categories of these methods:

- Filter-based methods are chosen for their simplicity and scalability in feature selection.

Two key examples are Mutual Information (MI) and the Chi-squared ( $\chi^2$ ) test. They are particularly useful for identifying informative features in tasks such as malware and spam detection.

- Mutual Information (MI): MI quantifies the dependency between two variables, indicating how much knowing one reduces uncertainty about the other. In feature selection, a higher MI score signifies a stronger relationship between a feature and the target variable, making the feature more informative for classification. MI is valuable for capturing non-linear relationships, unlike correlation coefficients. For example, MI can uncover subtle connections between system calls and malicious behavior in malware detection Lee & Kim (2015); Bose et al. (2008); Suarez-Tangil et al. (2013).
- Chi-squared ( $\chi^2$ ) Test: The  $\chi^2$  test assesses the independence of two categorical variables. In feature selection, it determines if a feature's presence or absence is significantly associated with the class label. A high  $\chi^2$  value suggests dependence, indicating the feature's relevance for distinguishing between classes. For instance, in spam detection, the  $\chi^2$  test can reveal if certain keyword frequencies differ significantly between legitimate and spam emails Zhang et al. (2004). This method

is efficient and well-suited for large datasets with categorical features Ilyas et al. (2004).

- Wrapper methods evaluate the performance of different feature subsets by training and validating a model on each candidate subset. Unlike filter methods that rely solely on statistical characteristics of the data, wrapper methods incorporate the learning algorithm itself as part of the evaluation process, making them more tailored and potentially more accurate for the specific classification task.

These methods are typically more computationally intensive because they involve repeated training and validation cycles, especially on large datasets or when using complex models. However, their advantage lies in their ability to capture interactions among features and adapt the selection process to the characteristics of the chosen model.

Two widely used wrapper techniques are:

- Recursive Feature Elimination (RFE): RFE works by recursively removing the least important feature(s) based on the model's weights or feature importance scores, and then re-fitting the model to the remaining features. This process continues until the desired number of features is reached. It is especially effective when used with models that expose meaningful coefficients, such as linear SVMs or logistic regression. In security applications, RFE has been employed to improve the accuracy of malware classification systems and detect network intrusion pat-

terns by isolating the most relevant behavioral indicators Resende & Drummond (2018).

- Sequential Feature Selection (SFS): SFS builds up (forward selection) or prunes down (backward elimination) the feature set by adding or removing one feature at a time based on the model’s validation performance Brodley & Utgoff (1995). It is particularly useful when domain knowledge is limited or when the optimal number of features is unknown. SFS has proven effective in domains where dimensionality reduction is critical, such as real-time fraud detection and anomaly classification, where computational efficiency and interpretability are both important Alimi et al. (2020).

- Projection-based methods transform the original high-dimensional feature space into a lower-dimensional representation by projecting the data onto a new set of features (called components or latent factors). Unlike filter or wrapper methods that select a subset of existing features, projection methods generate entirely new features, which are often linear or non-linear combinations of the original ones. These methods are particularly effective in handling sparse, noisy, or highly correlated datasets, as they uncover latent structures and reduce dimensionality while preserving as much information as possible.

Two widely used projection-based techniques in machine learning and data mining are:

- Principal Component Analysis (PCA): PCA is an unsupervised linear transfor-

mation method that identifies directions (principal components) in the data along which variance is maximized. Mathematically, PCA finds the eigenvectors of the covariance matrix of the data and projects the data onto the top- $k$  eigenvectors corresponding to the largest eigenvalues Salem & Hussein (2019). The resulting components are orthogonal and ordered by the amount of variance they explain. PCA is widely used in text classification to reduce the dimensionality of TF-IDF matrices, especially when dealing with thousands of correlated terms Kadhim et al. (2014). It helps improve model generalization, reduce overfitting, and speed up training without substantial loss in accuracy Howley et al. (2005).

- Non-negative Matrix Factorization (NMF): NMF decomposes a non-negative matrix  $X$  into the product of two lower-rank non-negative matrices  $W$  and  $H$ , such that  $X \approx WH$  Gillis (2020). This factorization leads to parts-based, additive representations that are particularly interpretable. In document clustering and topic modeling, NMF is used to discover latent topics and reduce text dimensionality Chen et al. (2019). Unlike PCA, which can produce negative entries and orthogonal components, NMF constraints make it more suitable for sparse, positive-valued datasets such as bag-of-words or TF-IDF vectors Berry & Kogan (2010).

Projection-based methods offer the advantage of compact data representations that retain essential structure, making them well-suited for visualization, clustering, and downstream classification tasks Benner et al. (2015). However, since the transformed

features may lack intuitive interpretability (especially in PCA), these methods are generally used when performance and compression are prioritized over feature transparency Salahuddin et al. (2022).

Historically, feature selection often relied on empirical judgment to decide how many features to retain Huang (2015). Researchers and practitioners would manually analyze various metrics and model performances to determine an optimal number Kira & Rendell (1992); Arauzo-Azofra et al. (2011); Salman et al. (2024). This process was not only time-consuming but also highly subjective and difficult to reproduce Theng & Bhoyar 2024.

In Our approach, we take a significant leap forward by leveraging the "elbow" method to automatically identify the ideal number of features to keep. The elbow method, well-known in clustering for finding the optimal number of clusters, involves plotting a metric (like explained variance or model error) against the number of features Shi et al. (2021). The "elbow point" on this graph signifies a point of diminishing returns, where adding more features provides less significant improvement Darban & Valipour (2022). When we determine the optimal number of features, there are two common methods are used:(1) Visual Method: Plot the evaluation metric (e.g., accuracy or variance) against the number of features, and identify the point where the curve starts to level off—this is the "elbow point" Saputra et al. (2020). (2) KneeLocator (Automated): An algorithmic approach that detects the elbow by calculating the point of maximum curvature, providing a reproducible and objective solution Shi et al. (2021).

In our research, we use the visual method to find the "elbow point" for the ransomware

dataset due to its limited number of features. We compared KneeLocator with AutoCut-2D for the text data from Telegram chat. AutoCut-2D is a second-order derivative-based method designed to automatically identify the optimal cutoff point (i.e., elbow point) in performance curves. Given a sequence of performance scores (such as classification accuracy) sorted by the number of selected features, AutoCut-2D computes the second derivative of the curve to measure the rate of change in slope. The position where the second derivative reaches its maximum is considered the elbow point, as it reflects the point of maximum curvature. This approach is particularly useful for detecting the transition between the performance gain phase and the saturation phase in high-dimensional feature selection tasks.

## 2.3 Experiments and Results

### 2.3.1 Dataset Specification

- Ransomware Data

The ransomware dataset we used has 138047 rows x 57 columns. The first 56 columns are dependent feature variables. The last column, “legitimate” is labeled as the target variable, where 0 and 1 indicate ransomware and legitimate software, respectively. The distribution is shown in the 2.1. All 56 features of the dataset are extracted from the tested executable binary files. Some String type features must be dropped, such as Name identity of software and MD5 (Message-Digest algorithm 5 for encryption or fingerprint function for a file). Malware makes use of some unused fields to hide encrypted malicious code in the PE file used by the OS loader to manage the wrapped

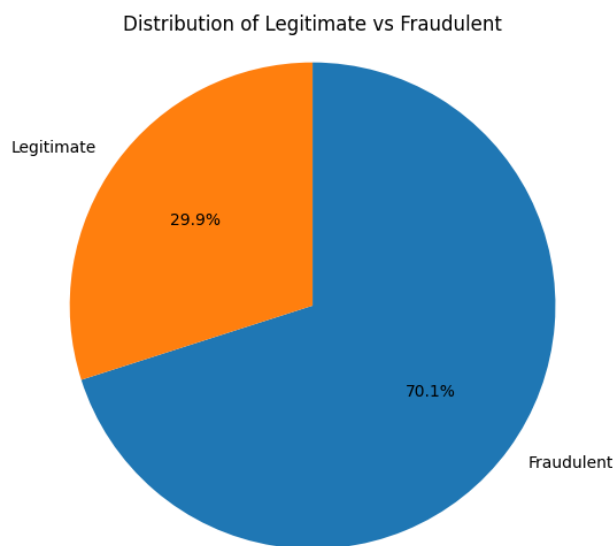


Figure 2.1 Ransomedata Distribution

executable code. A PE file consists of a PE header and sections (code, data, and imports of libraries) are there.

Some features, such as `SectionsMaxEntropy` and `ResourcesMaxEntropy`, reflect the randomness and disorder of a file with Shannon's entropy value (a scale of 0-8), which have close correlations with the target variable. The higher the entropy, the more likely it is that the code is encrypted. A typical ordinary (not packed/compressed/ encrypted) binary file has an entropy below 6, and a packed malware has an entropy above 7. Some other features don't make any contributions to the malware classification.

- Data From Telegram

We collected a dataset of 2,170 Telegram posts from public channels focused on financial activity. Each message was labeled with one of six financial fraud categories: (1) Credit Card (CVV Dumps), (2) Bank Account Drops, (3) Personal Information, (4) Gift Card

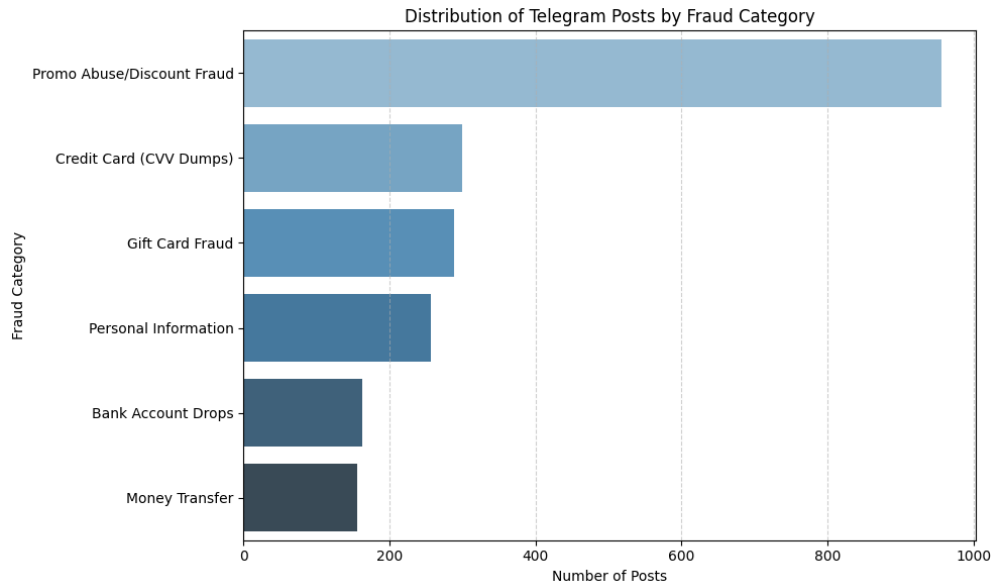


Figure 2.2 Telegram Distribution

Fraud, (5) Money Transfer, and (6) Promo Abuse and Discount Fraud.

The raw HTML content was downloaded from selected channels and parsed using the BeautifulSoup library to extract message text. After cleaning and filtering, 2,117 valid messages were retained. The resulting dataset was stored in a structured format for downstream classification and analysis 2.2.

### ***2.3.2 Feature Selection Methods***

In scenarios involving thousands of features, selecting a reduced set of important dimensions can significantly improve training efficiency and reduce computational overhead without sacrificing accuracy.

Feature selection aims to identify the most relevant features from high-dimensional data, reducing redundancy while retaining predictive power. In this study, we evaluate multiple categories of feature selection techniques, including filter-based, wrapper-based, and

projection-based methods. Each method is briefly described below, along with its associated formulation.

### 2.3.2.1 Filter-Based Methods

- **Mutual Information (MI)** Mutual Information measures the amount of information one random variable contains about another. For a feature  $X_i$  and class label  $Y$ , the mutual information is defined as:

$$MI(X_i, Y) = \sum_{x \in X_i} \sum_{y \in Y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \quad (2.1)$$

A higher MI score indicates greater dependency between the feature and the target.

- **Chi-squared ( $\chi^2$ )** The Chi-squared test evaluates the independence between a feature and the label using observed and expected frequencies:

$$\chi^2 = \sum \frac{(O - E)^2}{E} \quad (2.2)$$

Where  $O$  and  $E$  are the observed and expected frequencies of term occurrences, respectively.

### 2.3.2.2 Wrapper-Based Methods

- **Recursive Feature Elimination (RFE)** RFE recursively removes the least important features based on the weight coefficients from a trained model. At each iteration, the model is refit and ranked by:

$$\text{Feature Importance} \propto |w_i| \quad (2.3)$$

Where  $w_i$  is the weight of the  $i$ -th feature in a linear model (e.g., SVM).

- Sequential Feature Selection (SFS)

SFS adds (forward) or removes (backward) features incrementally by evaluating model performance on subsets. The optimal subset of size  $k$  is defined as:

$$X_{\text{best}} = \arg \max_{X \subseteq \mathcal{F}, |X|=k} \text{Accuracy}(X) \quad (2.4)$$

where  $\mathcal{F}$  is the full set of candidate features and  $X$  is a subset of exactly  $k$  features.

### 2.3.2.3 Projection-Based Methods

- Principal Component Analysis (PCA) PCA reduces dimensionality by projecting the data onto orthogonal components that capture maximum variance:

$$Z = XW, \quad \text{where } W \text{ are the top } k \text{ eigenvectors of } X^T X \quad (2.5)$$

We retain enough components to preserve 80%–95% of the total variance.

- Non-negative Matrix Factorization (NMF) NMF factorizes the input matrix  $X$  into two non-negative matrices  $W$  and  $H$  such that:

$$X \approx WH, \quad W \geq 0, H \geq 0 \quad (2.6)$$

Where  $W$  captures the latent feature representation.

### 2.3.2.4 Elbow Point Detection

We incorporate two elbow-based methods to automatically determine the number of features to retain:

- AutoCut-2D (Proposed).

This method assumes that the sorted feature importance scores (e.g., Mutual Information scores) form a decreasing curve. The elbow point is identified as the point where the second derivative reaches its minimum, indicating a change in curvature:

$$\text{Elbow Index} = \arg \min_i \left| \frac{d^2 S}{di^2} \right| \quad (2.7)$$

where  $S$  is the sorted list of feature scores. This approach captures where the rate of score decrease significantly slows, indicating diminishing marginal importance.

- KneeLocator

The KneeLocator algorithm uses geometric curvature to identify the inflection point on the score curve. It models the curve as a function and finds the point where the distance between the curve and the straight line connecting the endpoints is maximized:

$$\text{Knee Index} = \arg \max_i \left( d(i) = \frac{|(x_i, y_i) - \text{line}(x_i)|}{\|\text{line}\|} \right) \quad (2.8)$$

This method is useful for curves that exhibit a clear geometric knee, where the score improvement flattens sharply.

These methods allow us to analyze both numeric and textual cybercrime data under different feature reduction strategies. The evaluation compares their effectiveness in classification accuracy and computational efficiency.

### ***2.3.3 Performance Measure***

To assess feature selection performance, we use a Support Vector Machine (SVM) and a logistic classifier for all experiments. Evaluation is based on two key metrics: classification accuracy and training time. Accuracy measures how well the model predicts the correct category, while training time reflects computational efficiency. These two criteria allow us to compare trade-offs between precision and resource consumption across different feature selection strategies.

### ***2.3.4 Results***

#### ***2.3.4.1 Ransomware Dataset***

We use the embedded feature importance method in the Random forest to get the importance value. The arrangement of the original feature importance for each feature is shown in Figure 2.3.

We can see from the bar picture that different features are of special importance. We will choose the features with high importance. Then one problem is how we determine where we should stop? How many features should we choose? Some use the “square root”, or ‘log2’ of the total features, but without explanation. The elbow method is applied to find the turning point in the feature importance graph where the slope of the curve turns flat, which indicates that the features after the turning point can be reduced.

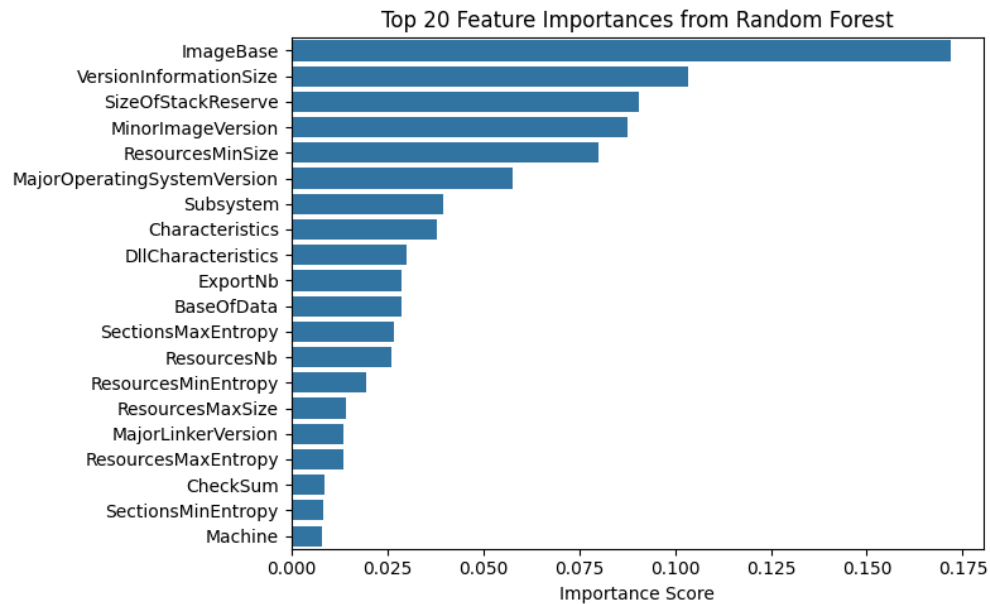


Figure 2.3 Feature Importance With Random Forest

#### 2.3.4.2 Decide the final with an Elbow method

We draw a Line chart from the important values and related features shown in Figure 2.4. From the picture, we can see the cutoff point we can set it to the 'Checksum'. We chose the first 18 important features. Since this is the point where the value reduces significantly. After this point, the 35 other features do not make too much contribution to the importance of the classification.

Logistic regression was applied to evaluate performance. Table 2.1 compares accuracy and precision before and after preprocessing.

Table 2.1 Result Comparison

	Before Preprocessing	After Preprocessing
Accuracy (%)	69.77	93.86
Precision (%)	69.77	92.96

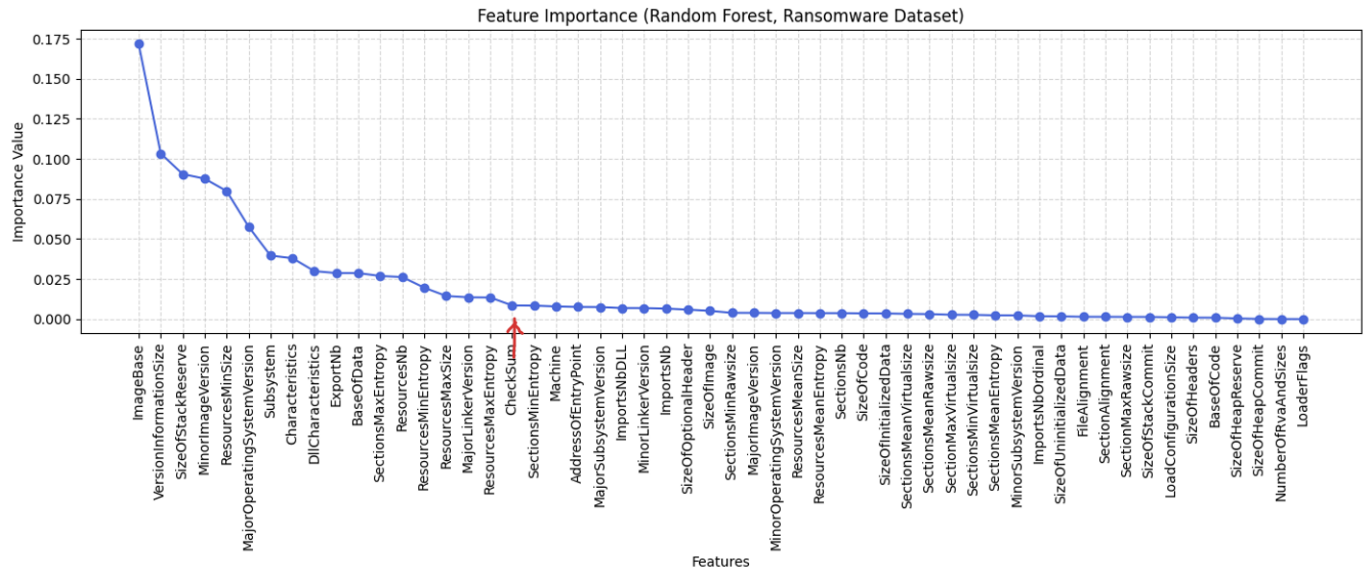


Figure 2.4 Feature importance selection with Elbow Method Visually

### 2.3.4.3 Financial Fraud Dataset

After converting the Telegram posts into TF-IDF vectors with 5000 dimensions, we applied a series of feature selection methods and evaluated the classification results using an SVM classifier.

- Initial Comparison of Feature Selection and Dimensionality Reduction Strategies To explore the trade-offs between training efficiency and classification accuracy, we first conducted a comparative evaluation of multiple feature selection and dimensionality reduction strategies using a Support Vector Machine (SVM) classifier. The methods tested include traditional filter-based approaches (Mutual Information (MI), Chi-squared (Chi<sup>2</sup>)), our proposed 2D Elbow cut-off strategy, and unsupervised dimensionality reduction techniques such as Principal Component Analysis (PCA) and Non-

negative Matrix Factorization (NMF).

Figures 2.5 and 2.6 present the training time and accuracy performance, respectively, across these methods.

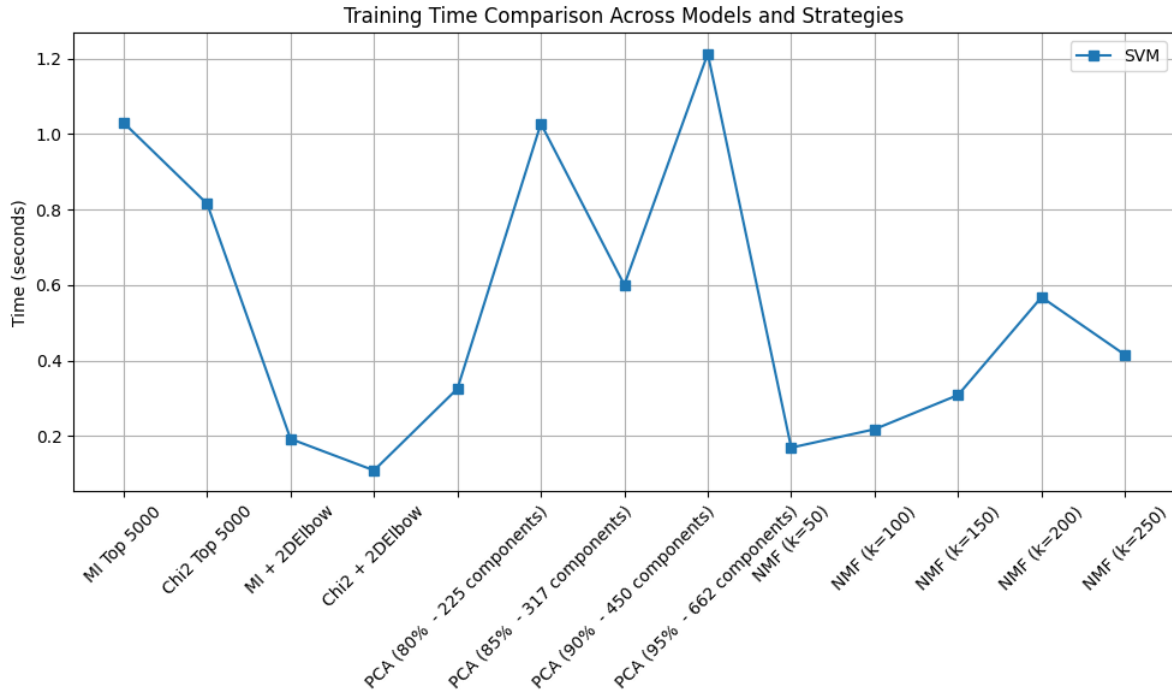


Figure 2.5 Training time comparison across models and strategies.

From the training time comparison in Figure 2.5, we observe that selecting the top 5000 features using MI or Chi2 incurs the highest computational cost (above 1.0 seconds). In contrast, using the 2D Elbow method significantly reduces training time—down to approximately 0.12 seconds for Chi2 and 0.19 seconds for MI—while retaining a strong performance.

PCA-based strategies show a gradual increase in training time as the number of retained components increases (from 225 to 662). Similarly, NMF-based approaches show

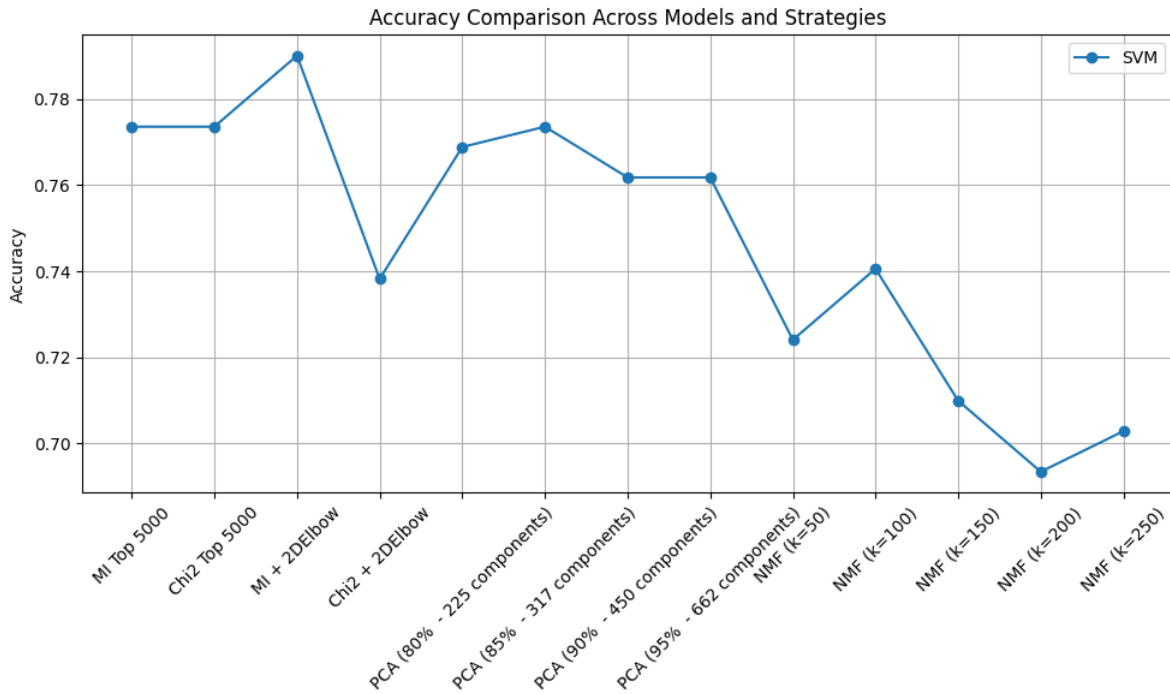


Figure 2.6 Accuracy comparison across models and strategies.

increasing training time with larger values of 0.567886.

In terms of classification performance shown in Figure 2.6, MI + 2D Elbow achieves the best accuracy (around 0.789), followed by the baseline MI and Chi2 top-5000 approaches. PCA methods show relatively stable accuracy (roughly 0.76–0.77) across different component thresholds. However, NMF’s accuracy decreases steadily with larger  $k$ , dropping to as low as 0.695 at  $k = 200$ .

These results suggest that MI + 2D Elbow offers a favorable trade-off between training efficiency and predictive power, making it a strong candidate for high-dimensional feature selection in subsequent evaluations.

- Comparing Elbow Detection Methods — 2D Elbow vs. Kneed

To further investigate automated strategies for determining the optimal number of features, we compared our proposed 2D Elbow method with the widely used Kneed Elbow algorithm, both applied on Mutual Information (MI) scores. The comparison focuses on their impact on classification accuracy when used in conjunction with an SVM classifier.

Figure 2.7 presents the results of this comparison.

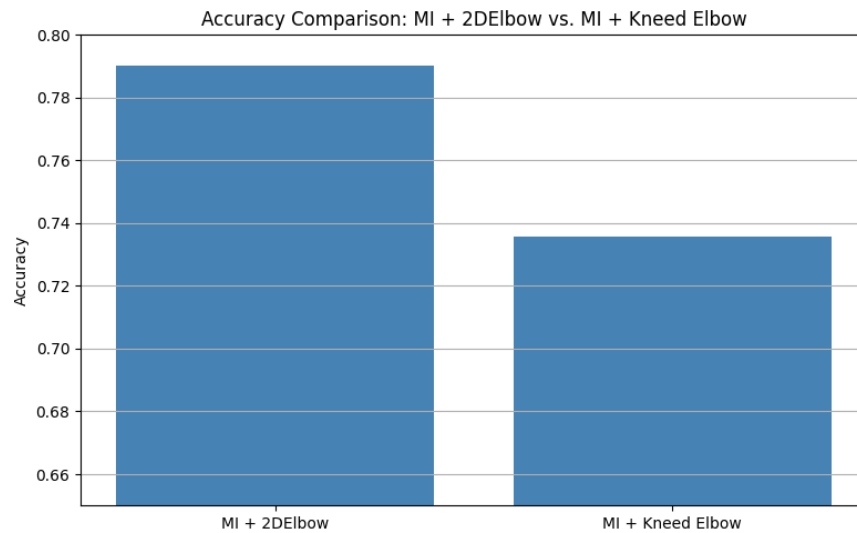


Figure 2.7 Accuracy Comparison: MI + 2D Elbow vs. MI + Kneed Elbow

As shown in Figure 2.7, the 2D Elbow method clearly outperforms the Kneed Elbow method in terms of classification accuracy. Specifically, MI + 2D Elbow achieves an accuracy of approximately 0.789, while MI + Kneed Elbow reaches only about 0.737. This 5% improvement highlights the benefit of using curvature-based second derivative analysis over geometric estimation when detecting inflection points in sorted MI score distributions.

These findings suggest that the 2D Elbow strategy provides a more effective and reliable cutoff point for high-dimensional feature selection and should be preferred over Kneed in downstream tasks. In future steps, we will further integrate this technique with other classifiers and data modalities to assess its robustness and generalizability.

- Comparison of Wrapper-Based Feature Selection Methods — RFE vs. SFS

In this step, we evaluate wrapper-based feature selection techniques, specifically Recursive Feature Elimination (RFE) and Sequential Feature Selection (SFS), to assess their effectiveness in improving classification performance while considering computational cost.

Figure 2.8 summarizes the results, showing classification accuracy and training time across different feature counts for both RFE and SFS strategies.

For RFE, performance improves consistently as more features are included. The accuracy rises from 0.5943 (with 50 features) to 0.6085 (at 350 features and beyond), while training time significantly decreases, from 12.55 seconds (50 features) to only 0.11 seconds (500 features). The best balance of performance and cost occurs around 350- 450 features.

In contrast, SFS yields slightly lower accuracies and dramatically higher computation times. The best accuracy (0.6014) is achieved with 150 or 200 features, but at the cost of over 2400 to 3200 seconds of runtime. Even with only 50 features, SFS takes more than 740 seconds, making it computationally infeasible for large-scale applications.

RFE		Features: 50		Accuracy: 0.5943		Time: 12.55s
RFE		Features: 100		Accuracy: 0.5967		Time: 8.83s
RFE		Features: 150		Accuracy: 0.5967		Time: 8.11s
RFE		Features: 200		Accuracy: 0.6014		Time: 6.21s
RFE		Features: 250		Accuracy: 0.6038		Time: 6.06s
RFE		Features: 300		Accuracy: 0.6038		Time: 5.03s
RFE		Features: 350		Accuracy: 0.6085		Time: 4.27s
RFE		Features: 400		Accuracy: 0.6085		Time: 3.13s
RFE		Features: 450		Accuracy: 0.6085		Time: 2.27s
RFE		Features: 500		Accuracy: 0.6085		Time: 0.11s
SFS		Features: 50		Accuracy: 0.5920		Time: 740.69s
SFS		Features: 100		Accuracy: 0.5991		Time: 1559.11s
SFS		Features: 150		Accuracy: 0.6014		Time: 2414.60s
SFS		Features: 200		Accuracy: 0.6014		Time: 3284.33s
Best SFS Result => Features: 150, Accuracy: 0.6014, Time: 2414.60s						

Figure 2.8 Accuracy and training time for RFE (top) and SFS (bottom) across selected feature dimensions.

## 2.4 Conclusion

In this project, we investigated and compared various feature selection and dimensionality reduction techniques on two distinct cybercrime-related datasets: (1) a ransomware detection dataset based on static binary features, and (2) a Telegram financial fraud dataset composed of unstructured social media messages.

Dataset. The ransomware dataset contained over 138,000 Portable Executable (PE) files, including both malware and benign samples. From each file, we extracted a set of 56 static structural features, such as file size, number of sections, imported APIs, and entropy-based measurements. To identify the most informative subset of features, we applied a Random

Forest-based importance ranking, followed by our proposed AutoCut-2D strategy. This method automatically determines the optimal number of features by identifying the elbow point in the sorted importance curve using second-order derivatives. As a result, we selected 18 high-impact features, which achieved comparable accuracy to the full feature set while significantly reducing computation time and improving model interpretability.

Financial Fraud Dataset. We also analyzed a high-dimensional text dataset collected from Telegram groups focused on illicit financial activities. After applying TF-IDF vectorization, we evaluated a range of dimensionality reduction techniques, including:

- Filter-based selection: Mutual Information (MI) and Chi-squared (Chi2), with both fixed- $k$  and elbow-based cutoffs.
- Wrapper-based methods: Recursive Feature Elimination (RFE) and Sequential Feature Selection (SFS).
- Unsupervised methods: Principal Component Analysis (PCA) and Non-negative Matrix Factorization (NMF).

Each method was assessed using a Support Vector Machine (SVM) classifier, focusing on both classification performance and training efficiency. Key results include:

- The MI + 2D Elbow method consistently outperformed other techniques, achieving the highest accuracy (approximately 0.789) with minimal training time.
- PCA maintained robust accuracy across different component thresholds, whereas NMF showed declining performance as  $k$  increased.

- Wrapper-based methods (RFE and SFS) delivered reasonable accuracy, but SFS in particular suffered from prohibitive computational cost.

and Future Work. Our findings highlight the value of automated elbow-based feature selection strategies such as AutoCut-2D in both structured (ransomware) and unstructured (Telegram) domains. These methods offer a favorable trade-off between accuracy and computational efficiency, and support the development of scalable cybercrime detection systems.

In future work, we aim to:

- Integrate deep learning-based feature extractors (e.g., BERT, CNN);
- Combine structural, semantic, and contextual signals in a multimodal framework;
- We further explore alternative feature embedding methods for the Telegram dataset and evaluate a variety of classification algorithms to assess their impact on model performance.

## CHAPTER 3

### CHAPTER 3 CATEGORY PREDICTION OF FINANCIAL FRAUD USING MESSAGES IN TELEGRAM

#### 3.1 Introduction

Telegram is a cross-platform instant messaging application Sutikno et al. (2016). It offers end-to-end encryption to ensure that messages remain private during transmission and storage. Users can create and join various types of chat groups, including private chats, group chats, and public channels, making them suitable for organizations and social media publishers. Telegram has a relatively loose censorship policy. That is, it only censors the contents that are related to terrorism or child porn. Therefore, Telegram has become to popular platform for vendors to advertise illicit goods.

Financial fraud-related products are one of the illicit goods Zhu et al. (2021). As we know financial fraud has a crucial impact on all users around the world. So, our research focuses on the analysis of financial fraud-related posts to protect individuals and organizations and maintain the integrity of financial systems.

Financial fraud-related products include various types, such as credit card (CVV dumps), bank account drops, and gift card fraud. Accurate classification of these fraud types can assist online surveillance. However, due to the limited research on Telegram, there are not enough publicly available datasets. Therefore, we collected and manually labeled our own data. So, we decided to collect the data and label it manually. The purpose of this study is to train effective machine-learning models using the data we have collected and organized.

In this paper, we first review the related work in Section 3.2. In Section 3.3, we describe the methodology used in our study. The experimental results are presented in Section 3.4. Finally, we conclude the paper in Section 3.5.

Our main contributions are:

I) We collect the post data from more than ten Telegram channels. We manually categorized them into 6 classes related to Financial Fraud following the "We The North" website.

II) We develop and test different machine learning methods to classify the posts in the dataset. We obtain a high level of accuracy using the developed machine learning methods.

## 3.2 Related Work

Ghosh and colleagues present an Automated Tool for Onion Labeling (ATOL) in their work. ATOL is a sophisticated system designed to navigate the Tor network and compile a database of keywords associated with illegal activities on concealed "onion" sites. The system utilizes Term Frequency Inverse Corpus Frequency (TFICF) and a clustering method similar to K-Means to categorize these hidden web pages. By employing clustering, ATOL assigns "thematic labels" to the content of these sites, identifying the main topics and informing the selection of search keywords Ghosh et al. (2017).

Tavabi et al. analyze an extensive collection of messages from 80 forums on the deep and dark web (d2web) spanning over a year. They reveal the dynamic nature of discussions within these forums and identify content similarities. Hidden Markov Models (HMM) are used to uncover underlying states between forums, effectively capturing the fluctuating nature of

forum content, which is crucial as forum themes may shift away from overtly criminal topics while still facilitating illicit activities Eason et al. (1955).

Bhalerao and colleagues propose a graph-based approach to unveil criminal supply chains in their research. They gather data from English-language "Hack" forums and Russian-language "Antichat" forums, focusing on commercial posts resembling the advertisement posts discussed in their paper. The study constructs an interaction graph ( $G = (U, E)$ ), where each node  $u \in U$  represents a forum user, and each edge  $(u_a, u_b) \in E$  signifies a transaction where user  $u_a$  sells a product to user  $u_b$ . The term-frequency inverse document-frequency (TF-IDF) technique is employed to convert forum words into vectors based on significance, and various classifiers are used to identify supply chains Tavabi et al. (2019).

In the study Mackey et al. (2020), the authors develop a machine learning framework to identify Instagram posts associated with illicit online drug sales. Data is collected over three months from Instagram's website, creating a dictionary based on word frequency for data feature extraction. The analysis employs decision trees, random forests, support vector machines, and LSTM (Long Short-Term Memory) deep learning models, all of which demonstrate strong performance in classifying the targeted content.

In Jarynowski et al. (2021), Andrzej J and Alexander S use the deep learning models LSTM and BERT to analyze mild adverse events associated with a specific COVID-19 vaccine.

While these studies differ in their methodologies and focus areas, they all contribute to the broader goal of understanding and combating illicit activities online. It's interesting to

note that while each study emphasizes a particular model or feature extraction technique, they collectively showcase the diversity of approaches in this field. In our project, we focused our research on Telegram’s illicit activities and compared different models to find the best model for this special area.

### 3.3 Methodology Overview

In this section, we present our computational strategy for sourcing data from Telegram public channels, coupled with our methodology for scrutinizing text components within individual messages. Figure 1 shows the diagram of our proposed method.

From Figure 1, we can see the process for processing and classifying data from a Telegram channel. Initially, the raw data in a .zip format is processed using a data parsing method. We use MySQL database management software to create database tables for storing the seller information, text, images, and IDs. The Python program was developed to parse the raw data and insert the data records into the database tables. This database table data is exported as a CSV file for manual labeling. Subsequently, the data undergoes manual labeling. It is categorized into two types: those containing images and those without. For text data without images, features are extracted using techniques including BERT, TF-IDF, and Word2Vec, and classified using machine learning algorithms including SVM, Adaboost, and Neural Networks.

Finally, the classifier that achieves the highest accuracy is selected by comparing the performance of different algorithms. In essence, this process involves extracting valuable

information from raw data and analyzing it through machine learning techniques to achieve the most accurate data classification.

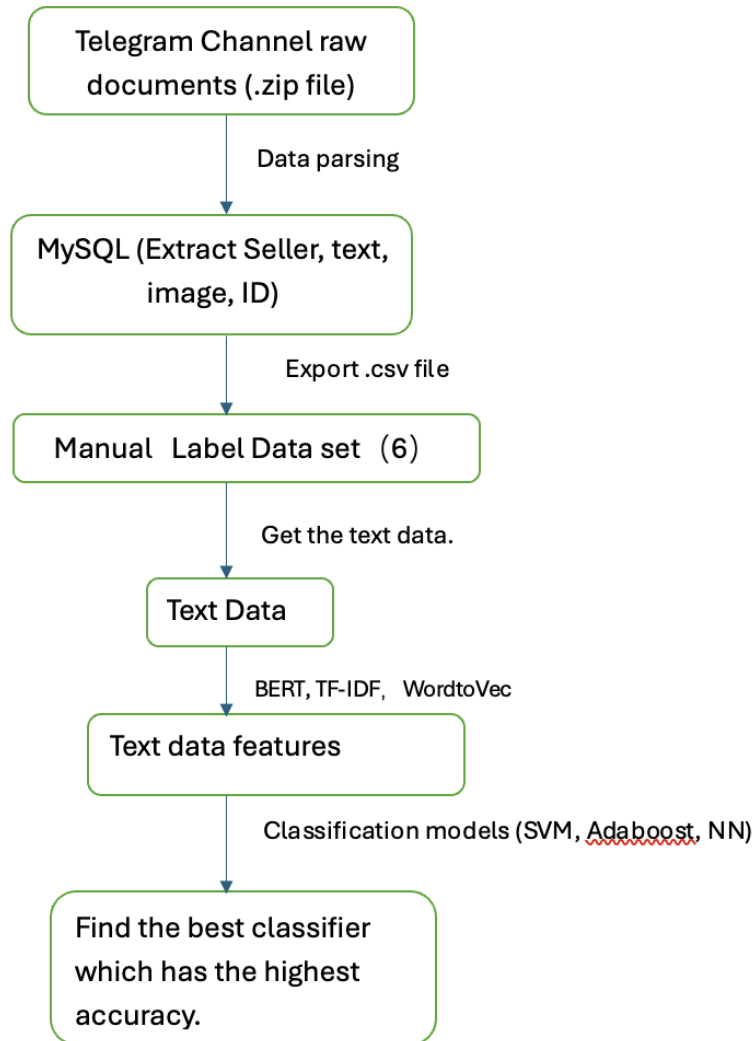


Figure 3.1 The Diagram of the Proposed Method.

### 3.3.1 Data Collection and Extraction

We developed a sophisticated program aimed at efficiently organizing downloaded data from Telegram chats. The program extracts vital details such as the posting channel, publisher,

posting time, and content. Once extracted, the data is meticulously structured and stored in the MySQL database. To manually classify, we download the table and save it as a .csv file.

### *3.3.1.1 Data collection*

We search for relevant content on Telegram based on common classification names found on the dark web and the internet. Figure 2 shows the common classification names. Once confirmed as relevant content, we download it locally. Then, we use DBeaver, a database management software, to create the table schema, and we developed Python programs to extract chat records from the channels and insert the data records into the database table. Figure 3 is the Entity-Relationship Diagram (ERD) in our DBeaver.

The database comprises two tables: `message` and `channels`. The `message` table stores the details of messages, which include content, time of posting, and links to associated images. Each message is connected to a specific channel through the `fk_channels_id` foreign key.

On the other hand, the `channels` table stores the channels, encompassing the channel name, time of creation, and the number of members. These two tables are linked via the foreign key in the `message` table, which references the primary key in the `channels` table.

This relational setup allows us to efficiently track the origins of each message, associating them with their respective channels, thus providing context for the messages posted by the sellers.

We store our text data in the `postDetail`, while the images are stored in the `image_source`. In this project, we are focusing on text data analysis.

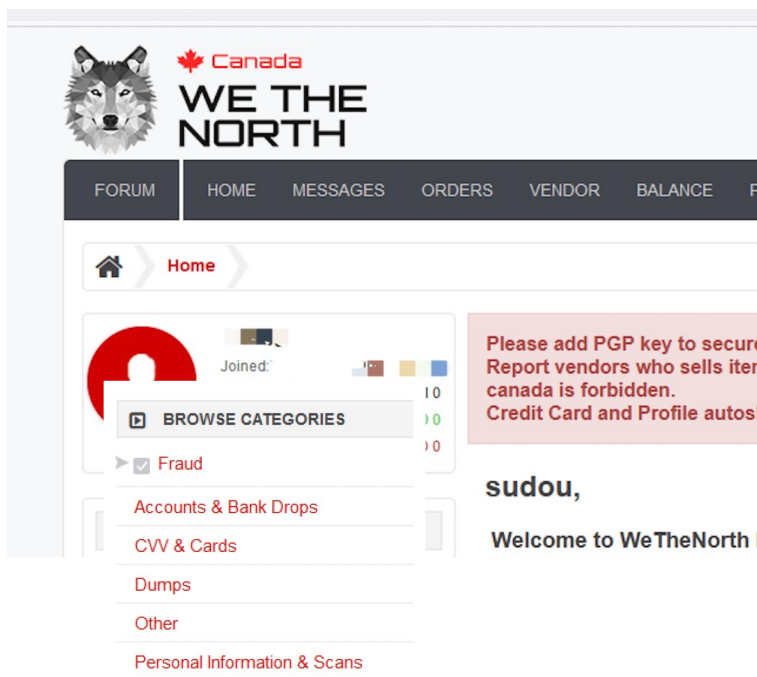


Figure 3.2 The Category on Web.

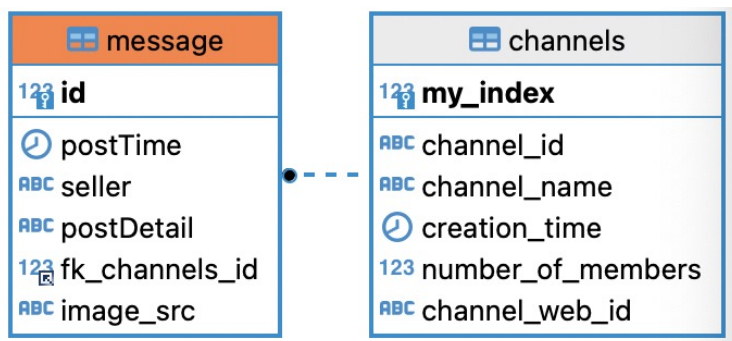


Figure 3.3 The schemas of the two tables.

### 3.3.1.2 Final dataset

After we store the table locally as a CSV file, our team engages in a manual categorization process to further refine and organize the information. The data originates from ten unique Telegram channels, each representing content from 6 distinct domains. The categorized

domains include Credit Cards (CVV Dumps), Bank Account Drops, Personal Information, Gift Card Fraud, Promo Abuse, Discount Fraud, as well as Wire Transfer. We collected a total of 2,579 samples.

### ***3.3.2 Feature Extraction Methods***

#### *3.3.2.1 Term Frequency-Inverse Document Frequency (TF-IDF)*

The TF-IDF model represents a text by quantifying the frequency of individual words (uni-grams), pairs of words (bi-grams), or even longer sequences (n-grams) within the text. This measurement, termed "term frequency," assesses how often each term occurs.

To address the influence of commonly occurring words, the term frequency is normalized by the inverse document frequency. This metric evaluates how frequently a term appears across all entries, or documents, in the corpus. Consequently, words that occur frequently across multiple documents are assigned lower weights, while those that are relatively rare are assigned higher weights Havrlant & Kreinovich (2017).

#### *3.3.2.2 Word Embeddings: Word2Vec*

Word2vec is a natural language processing (NLP) technique that involves generating vector representations for words. These vectors are designed to encapsulate both the meaning of a word and its contextual usage. The algorithm behind word2vec achieves this by analyzing extensive text corpora to estimate the representations of words, providing a valuable tool for understanding semantic relationships within language Lilleberg et al. (2015).

### *3.3.2.3 BERT(sentence Vectors)*

BERT, which stands for the Bidirectional Encoder Representations from Transformers Alparthi & Mishra (2020). It introduces a significant technical advancement by employing bidirectional training of the Transformer, a widely used attention model, for language modeling. This approach differs from earlier methods that considered text sequences either from left to right or combined both left-to-right and right-to-left training. The research demonstrates that a language model trained bidirectionally can possess a more profound understanding of language context and coherence compared to single-direction language models He et al. (2019).

### *3.3.3 Classification Models*

Classification models wield significant power within the realm of machine learning, facilitating the organization of data into predetermined categories or classes. Their versatility renders them indispensable across a myriad of applications. Each model boasts its unique set of strengths and weaknesses within a particular scenario. So, we selected different models for comparison.

#### *3.3.3.1 Support Vector Machines (SVMs)*

Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification and regression tasks Steinwart & Christmann (2008). The primary objective of SVM is to find a hyperplane in a high-dimensional space that best separates the data points into different classes. In a two-dimensional space, this hyperplane is a line, while in a higher-

dimensional space, it becomes a hyperplane Huang et al. (2018).

The key idea of SVM is to maximize the margin between the two classes, which is the distance between the hyperplane and the nearest data point from each class. There are two types of classifiers utilized in Support Vector Machines (SVM): linear and nonlinear Yang (2019); Fung & Mangasarian (2001). Let's look into each:

- **Linear Classifier:** In a linear classifier, the goal is to find a hyperplane that separates the data points into different classes. This hyperplane is a linear decision boundary that maximizes the margin between classes. Linear SVMs are effective when the data can be separated by a straight line or plane in the feature space. Consider a given training set  $\{(\mathbf{x}_k, y_k)\}_{k=1}^N$ . A large margin classifier between two classes of data can be represented by any hyperplane defined as the set of points  $\vec{x}$  satisfying (1):

$$\vec{w} \cdot \vec{x} - b = 0 \tag{3.1}$$

The  $\vec{w}$  represents the normal vector to the hyper-plane. And the  $b$  is the bias term.

- **Nonlinear Classifier:** Nonlinear classifiers employ techniques to handle datasets that cannot be effectively separated by a linear boundary. They utilize kernel functions to map the input data into a higher-dimensional feature space where it can be linearly separated. This allows SVMs to handle complex, nonlinear relationships between features Patle & Chouhan (2013); Prajapati & Patle (2010).

The decision function for SVM with the kernel trick can be expressed as (2):

$$f(x) = \vec{w} \cdot \phi(x) + b \quad (3.2)$$

Where  $f(x)$  is the decision function that maps input  $x$  to an output,  $\vec{w}$  is the weight vector,  $\phi(x)$  is the feature mapping function, which transforms the input  $x$  into a higher-dimensional space. This is where the kernel trick comes into play.  $b$  is the bias term.

The most common types of kernels include the polynomial kernel function, the sigmoid kernel function, and the Radial Basis Function (RBF) kernel. The RBF is function was approved to be more preferable Havrlant & Kreinovich (2017); Liu et al. (2011); Prajapati & Patle (2010). So we chose to use RBF in our project.

### 3.3.3.2 Adaboost: Ensemble Learning with Weighted Weak Classifiers

AdaBoost is an ensemble learning algorithm that combines the predictions of multiple weak classifiers to create a strong classifier. The algorithm assigns weights to data points and adjusts them during each iteration to focus on the misclassified points Schapire (2013). For a given training set  $\{(\mathbf{x}_k, y_k)\}_{k=1}^N$ , the Algorithm 1 shows how the Adaboost works in the multi-classification Mahesh et al. (2022).

In this Algorithm,  $N$  is the total number of observations,  $M$  is the number of iterations in the algorithm,  $w_i$  represents the weight of the  $i$ -th observation,  $G_m(x)$  is the classifier at iteration  $m$  predicting a ranking over all classes,  $\text{err}_m$  represents the weighted error of classifier  $G_m$  in iteration  $m$ ,  $\alpha_m$  is the weight computed for classifier  $G_m$ ,  $K$  is the number

---

**Algorithm 1** AdaBoost.MR (Multi-class Ranking)
 

---

Initialize: For each observation  $i$ , set initial weight  $w_i = \frac{1}{N}$ .  $m = 1$  to  $M$  Train classifier  $G_m(x)$  using current weights  $w_i$  to predict ranking over all classes. Calculate weighted error:

$$\text{err}_m = \frac{\sum_{i=1}^N w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^N w_i}$$

Compute classifier weight:

$$\alpha_m = \log\left(\frac{1 - \text{err}_m}{\text{err}_m}\right) + \log(K - 1)$$

Update weights:

$$w_i \leftarrow w_i \times \exp(\alpha_m \times I(y_i \neq G_m(x_i)))$$

Output the final model:

$$G(x) = \underset{y}{\operatorname{argmax}} \sum_{m=1}^M \alpha_m G_m(x, y)$$


---

of classes,  $I(\text{condition})$  represents the indicator function, equal to 1 if the condition is true, otherwise 0 and  $G(x)$  is the final model as a weighted combination of all classifiers, where the argument maximizing over  $y$  represents the predicted class.

### 3.3.3.3 Neural Networks (NN)

Neural Networks (NN) are a class of machine learning models inspired by the structure and functioning of the human brain Fung & Mangasarian (2001). They consist of interconnected nodes, also known as neurons, organized in layers Prieto et al. (2016); Aggarwal et al. (2018).

**Basic Structure:** A neural network is composed of layers: an input layer, one or more hidden layers, and an output layer. Each layer contains nodes, and each node is connected to every node in the next layer.

**Activation Function:** Nodes in a neural network apply an activation function to the weighted sum of their inputs. Common activation functions include the sigmoid, hyperbolic

tangent (tanh), and rectified linear unit (ReLU).

Feedforward Process: During the Feedforward process, input data is propagated through the network, layer by layer, producing an output Yang (2019). The output  $y$  of a neural network with  $L$  layers is calculated as follows:

$$y = f_L(W_L \cdot f_{L-1}(W_{L-1} \cdot \dots \cdot f_1(W_1 \cdot \vec{x} + b_1) + b_{L-1}) + b_L) \quad (3.3)$$

where  $\vec{x}$  is the input vector,  $W_i$  is the weight matrix for layer  $i$ ,  $b_i$  represent the bias vector for layer  $i$ ,  $f_i$  is the activation function for layer  $i$  and  $y$  is the predicted output.

## 3.4 Experiment and Results

### 3.4.1 Dataset Collection

We developed a program for efficiently organizing the data downloaded from Telegram chats. The program extracts vital details such as the posting channel, publisher, posting time, and content. Once extracted, the data is meticulously structured and stored in a CSV file, which is then saved locally. Subsequently, our team engages in a manual categorization process to further refine and organize the information. The data originates from ten unique Telegram chat channels, each representing content from one of these six distinct domains. We have a total of 2,579 samples.

The categorized domains include:

- Credit Card (CVV Dumps): The seller displays ads concerning credit cards.
- Bank Account Drops: a bank account operated by a lawbreaker, serving as an in-

termediary in a financial scheme designed to cleanse and legitimize illegally obtained money.

- Personal Information: The seller illicitly sells someone else's personal details, as SSN, email, bills, and so on.
- Gift Card Fraud: the seller steals money from prepaid cards or exploits the card system to obtain funds without paying for them, and sells gift cards at very low prices.
- Wire Transfer: Entails unlawfully acquiring funds by misleading an individual or organization into transferring money through a wire transfer.
- Promo Abuse and Discount Fraud: exploitation of promotional offers, discounts, or coupon codes by individuals or groups in ways not intended by the issuer.

Any remaining chat information that doesn't fall into these six categories has been dropped.

Figure 4 shows the data distribution of the raw data.

### ***3.4.2 Data Preprocessing***

Data preprocessing is vital in data analysis and machine learning because it cleans and standardizes raw data, making it usable for models. This process involves handling missing values, normalizing scales, and creating useful features, which enhances the accuracy and efficiency of models. Since our raw data is text data, so we also need tokenization, stemming, lemmatization, and the removal of stop words to prepare the unstructured text data for analysis.

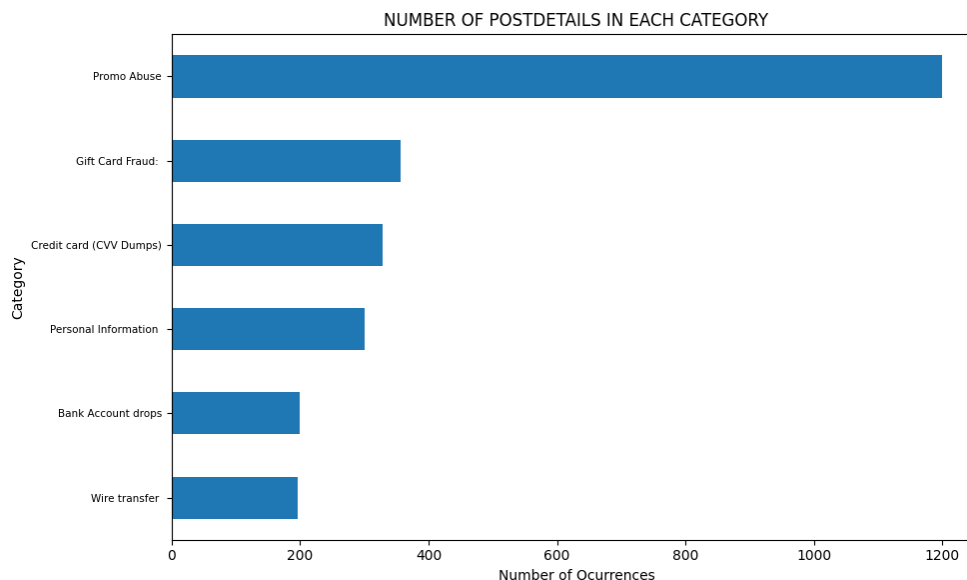


Figure 3.4 The Data Distribution

From the initial examination of the dataset, it is evident that the category labeled "Abuse & Discount Fraud" encompasses over 1200 samples, significantly outnumbering those in other categories. This substantial disparity highlights a classic case of data imbalance, which can skew the performance of predictive models. Upon conducting a thorough review, we also identified a considerable number of duplicate entries within this dataset.

To address these issues, we began by normalizing the data. This normalization process involved several key steps:

- Removing punctuation marks such as periods, exclamation points, and various special characters (e.g., ., ! \$() \* % @);
- Eliminating stop words that add little semantic value;
- Converting all text to lowercase to ensure uniformity;

- Implementing natural language processing techniques such as tokenization, stemming, and lemmatization.

These steps help in reducing noise and variability in the text data, making it more amenable to analysis.

Following the normalization, we tackled the issue of duplicate records. By using the remove duplicate method, we identified and removed these redundant entries. This cleansing step was crucial in refining our dataset, thus ensuring that our subsequent analyses and model training would not be adversely affected by these duplicative data points. Figure 5 is the newly curated dataset. This visualization offers a clear view of the dataset's composition after our intensive cleaning process, providing a more balanced and accurate basis for further data analysis and modeling.

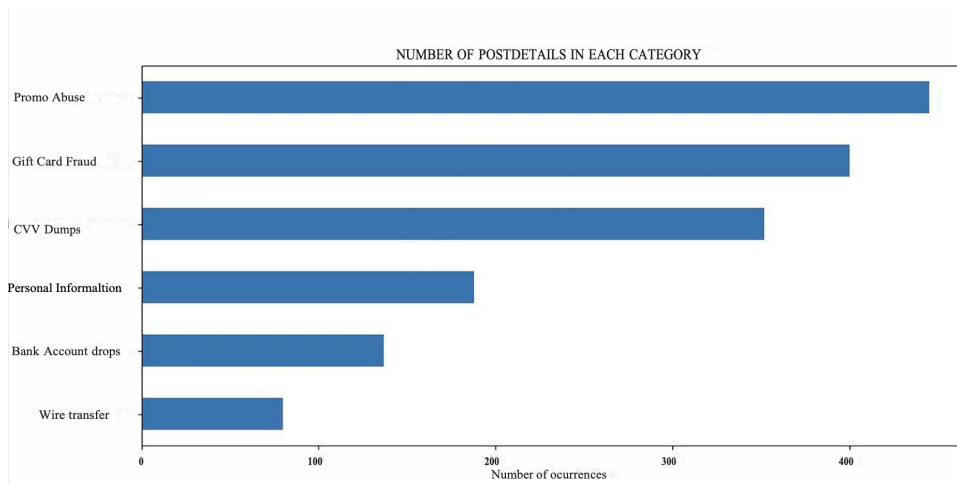


Figure 3.5 The Data Distribution after Pre-processing

### 3.4.3 Experiment Results

We conducted a comprehensive comparison of various feature extraction techniques to assess their performance when applied to the same model framework. Additionally, we evaluated the accuracy of identical vector methods across different modeling approaches to identify any significant variations in performance. To further deepen our analysis, we meticulously documented the time consumed by each feature extraction method when integrated with three distinct models. This detailed examination not only allowed us to discern the most efficient vector methods but also provided insights into the computational demands of each combination, facilitating more informed decision-making for future model implementation and optimization strategies. Table 1 shows the time consumption for different feature extraction methods.

Table 3.1 Seconds for different feature extraction methods

	TF-IDF	Word2Vec	BERT
Time	2.78	2.80	5.13

From the table, we can figure out that when we use TF-IDF, it is the fastest to finish the classification. When we changed to Word2Vec, it was a little slower than TF-IDF but faster than using BERT. This happened mostly because BERT comes with significant computational complexity and longer processing times due to its deep Transformer architecture, while TF-IDF does not capture any semantic or contextual information, making it less suitable for tasks requiring deep text understanding. It implies that while BERT may provide rich contextual embeddings, it comes at the cost of increased computational time compared

to the simpler TF-IDF and Word2Vec methods. This table complements the earlier performance chart by highlighting a trade-off between computational efficiency and possibly the effectiveness or complexity of the feature extraction.

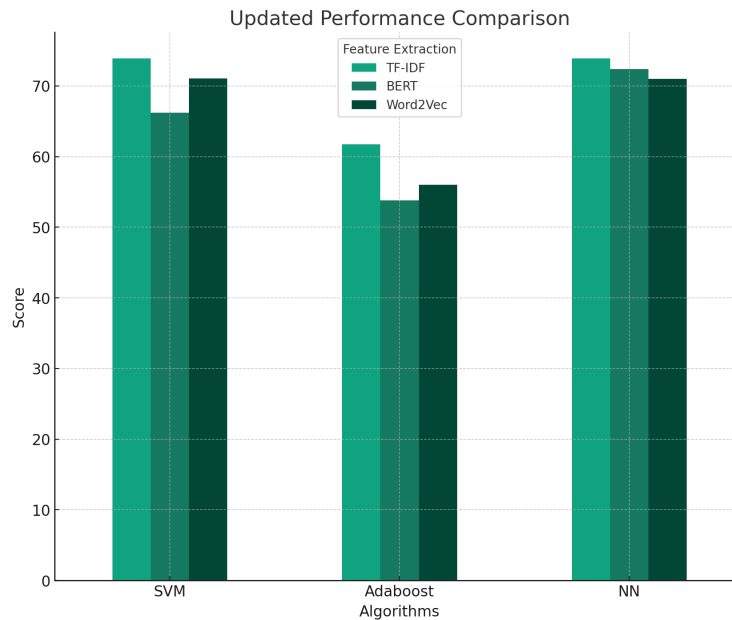


Figure 3.6 Performance Analysis

Figure 6 shows the final comparison of the accuracy. The performance scores are plotted on the y-axis, while the algorithms are distributed along the x-axis. In this chart, BERT consistently shows lower performance compared to the other two feature extraction methods (TF-IDF and Word2Vec) across all three machine learning algorithms (SVM, AdaBoost, and Neural Networks). Specifically:

- In the SVM category, BERT scores are below TF-IDF and slightly above Word2Vec.
- In AdaBoost, BERT scores similarly to TF-IDF but well below the performance achieved by Word2Vec.

- In Neural Networks, BERT's performance is the lowest compared to TF-IDF and Word2Vec.

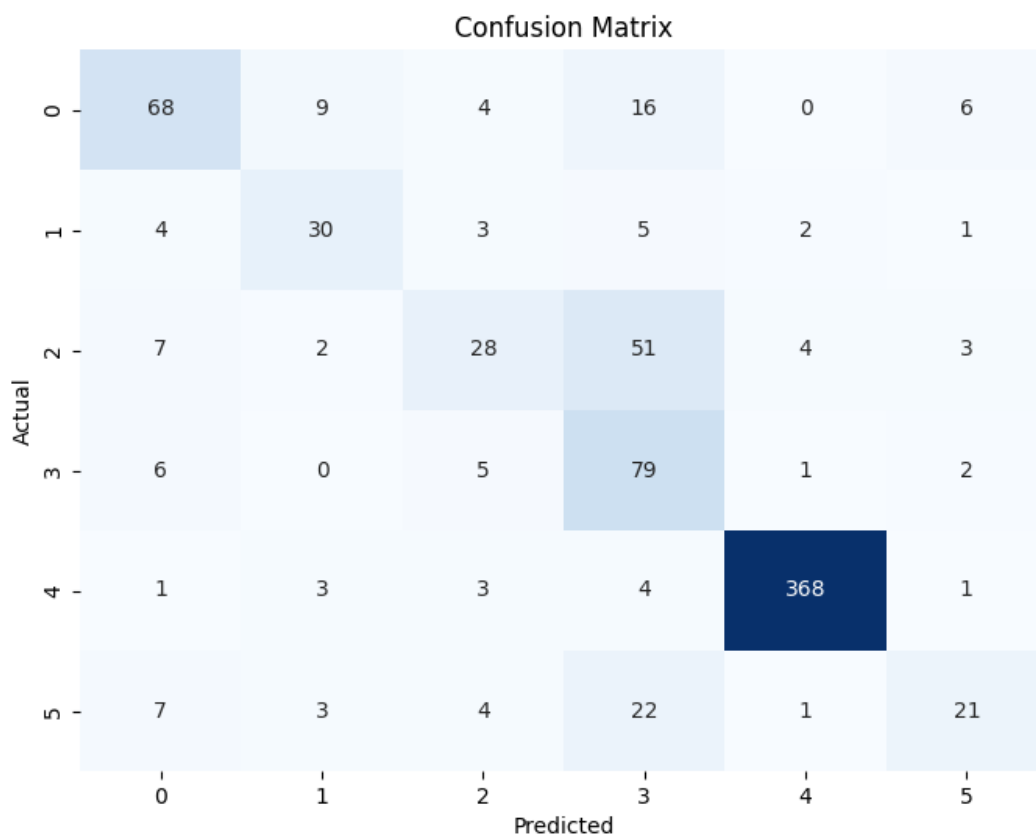


Figure 3.7 Confusion Matrix for NN

The confusion matrix for NN is shown in Figure 7, shows the confusion matrix for NN with the TF-IDF method. From this particular matrix, class '4', which represents the Gift Card Fraud, has a high accuracy with a dominant '368' correct predictions, whereas other classes show more dispersion and higher misclassification rates.

### 3.5 Conclusion

In this paper, we collected the newest data related to financial fraud from more than ten Telegram channels. We manually categorize them into six classes related to Financial Fraud. We focus our attention on the textual part of the dataset. From the result, we can observe that NN and SVM generally achieve higher relative accuracy, with TF-IDF emerging as the most effective method for feature extraction. BERT appears to be the most time-consuming approach and has the lowest accuracy. The BERT embedding is very slow, mostly because BERT comes with significant computational complexity and longer processing times due to its deep Transformer architecture, while TF-IDF does not capture any semantic or contextual information. It implies that while BERT may provide rich contextual embeddings, it comes at the cost of increased computational time compared to the simpler TF-IDF and Word2Vec methods. We can conclude that even though BERT was shown to be powerful in some classifications, our dataset showed that it performed mediocly overall, with poor speeds and accuracy compared to TF-IDF and Word2Vec.

## CHAPTER 4

### CHAPTER 4 RECOGNIZING THE FINANCIAL FRAUD TYPE OF TELEGRAM ADVERTISEMENT POSTS BASED ON THE MULTIMODAL LEARNING

#### 4.1 Introduction

The rapid growth and widespread use of instant messaging applications such as Telegram, Signal, and Wire have contributed to the proliferation of diverse financial fraud schemes, as criminals leverage these secure channels to carry out illicit activities.

Telegram which was founded by Nikolai and Pavel Durov in 2013 Saribekyan & Margvelashvili (2017) is a popular instant messaging application. It is a cloud-based, cross-platform service. The servers of Telegram are distributed worldwide with several data centers Akbari & Gabdulhakov (2019). Until 2024, it has more than 950 million monthly active users Durov's Channel (2024). As we know, the Paris Public Prosecutor's Office issued an arrest warrant for Durov in connection to an investigation opened on July 8, 2024 into organized crime, drug trafficking, fraud, and the distribution of pornographic images of minors on the Telegram platform Wikipedia contributors (2023); CNBC (2024). This high-profile arrest underscores the severity of financial crimes and other illegal activities that take place on Telegram, highlighting the urgent need for stronger monitoring measures.

In Telegram, criminals exploit the anonymity of the platforms to engage in a range of financial fraud schemes, including credit card fraud (CVV dumps), personal information trading, bank account drops, illicit money transfers, and promo abuse/discount fraud. These fraud schemes inflict serious damage on individuals, society, and the financial ecosystem. On

a personal level, victims often suffer direct monetary losses, plummeting credit scores, and the stress of long battles with financial institutions. In a broader sense, widespread fraud undermines trust in digital communication platforms and online commerce, causing people to hesitate before making legitimate transactions. Banks and payment processors also face higher operating costs as they struggle to combat fraud—costs that eventually get passed on to regular customers. If left unchecked, these fraudulent activities threaten economic stability and erode public confidence in digital financial services, underscoring the urgent need for robust detection methods.

In these frauds, credit card fraud (CVV dumps) typically poses the most immediate threat due to its capacity for rapid, large-scale losses. Personal information trading comes next, as it can fuel identity theft on a massive scale. Bank account drops rank third, enabling criminals to funnel illegal funds with relative ease. Illicit money transfers follow as they contribute to the layering and hiding of criminal proceeds across multiple accounts. Lastly, promo abuse/discount fraud —though still harmful—tends to have a less catastrophic financial impact compared to the other threats. Because these threats vary in both nature and urgency, it is imperative to establish a comprehensive classification framework capable of accurately differentiating among the various types of financial fraud. Such a system would enable authorities and financial institutions to prioritize their response efforts, deploy resources more strategically, and ultimately disrupt these criminal operations more effectively—protecting individuals and the broader economy from escalating harm.

Upon examining various Telegram channels, we found that criminals rely not just on

text-based messages; they frequently post images containing sensitive details such as credit card numbers, checks, and bank account information. In many cases, these images are accompanied by text that explains or expands upon the visual content—criminals might provide instructions alongside photos of forged checks, for example, or offer clarifications on how to use stolen credit card data. Given this close interweaving of textual and visual information, purely text-based classification methods can no longer keep pace. To accurately detect and address such fraudulent activities, it is essential to adopt a more comprehensive approach that accounts for both text and images in tandem. Then multimodal approaches are therefore highly suited to our research. By ‘multimodal,’ we refer to methods that integrate data from multiple sources—such as text, images in the same task Ngiam et al. (2011). This enables a more comprehensive understanding and recognition of complex scenarios and content.

In this study, we compare two different multimodal fusion techniques for fraud classification: concatenation and attention-based fusion. The concatenation method simply combines text and image features into a single vector, treating both modalities equally without considering their relative importance Noreen et al. (2020). In contrast, the attention-based fusion method assigns different weights to the modalities, allowing the model to focus more on the most informative modality depending on the context Jiang et al. (2020). These two approaches are evaluated and compared across five categories of fraud: Credit Card (CVV Dumps), Bank Account Drops, Personal Information Fraud, Money Transfer Fraud, and Promo Abuse/Discount Fraud.

The goal of this paper is to determine the optimal model for classifying Telegram-based fraud by comparing different fusion techniques, image embedding methods, and their impacts on overall performance. Through this comparison, we aim to enhance the accuracy of fraud detection systems and provide insights into how various multimodal fusion strategies and image embedding approaches influence the effectiveness of fraud classification.

Building upon our initial exploration of text-only and image-only classification methods, our experiments revealed that multimodal approaches significantly enhance classification performance. By leveraging complementary information from both text and image modalities, the multimodal strategy addressed the limitations of single-modality methods, achieving superior results across all evaluated metrics.

The Swin Transformer outperformed traditional CNN-based models like ResNet-50, thanks to its hierarchical architecture and ability to model long-range dependencies. Fusion strategies also played a crucial role, with attention-based techniques further improving performance by dynamically prioritizing salient features.

## **4.2 Related Work**

Fraud classification has become an increasingly important area of research, especially in the context of online platforms where fraudulent activities continue to proliferate. Various techniques have been developed to classify fraud into distinct categories, enabling more precise detection and prevention mechanisms. This section provides an overview of key approaches in fraud classification, with a focus on multimodal data fusion, traditional machine learning,

and deep learning techniques.

#### *4.2.1 Multimodal Fusion Techniques*

Multimodal fusion techniques, which integrate data from multiple sources such as text, images, and metadata, have shown great promise in improving fraud classification accuracy. A common approach in fraud detection involves concatenating features from different modalities (e.g., combining text and image data into a single vector). This approach has been widely applied in domains such as fake news and rumor detection, where both text and visual content need to be analyzed Bondielli & Marcelloni (2019). The concatenation methods are straightforward but do not account for the varying importance of different modalities Costa Pereira et al. (2014).

In recent years, more advanced techniques like attention-based fusion have been explored. In these methods, the system assigns dynamic weights to different modalities based on their relevance to the task. For instance, Han et al. (2023) proposed a dual-attention mechanism for rumor detection that assigns different importance to text and images, achieving better classification results than simple concatenation methods (MDPI) Han et al. (2023). Similarly, Jiang et al. (2020) applied mutual attention mechanisms to enhance multimodal fusion for fake news detection on social media, demonstrating improved performance by focusing on the most relevant features (SpringerLink) Jiang et al. (2020). These, where images (e.g. screenshots of illicit activities) and textual descriptions (e.g. ads for fraudulent schemes) play critical roles.

Concatenation-based methods offer a simple and computationally efficient approach to

multimodal fusion, but they often fail to capture complex interactions between modalities. On the other hand, attention-based models dynamically weigh the importance of each modality and have been shown to outperform concatenation in tasks involving diverse data types. Studies by Han et al. (2023) and Jiang et al. (2020) have demonstrated the effectiveness of attention mechanisms in improving classification accuracy by focusing on the most informative aspects of both text and image data (MDPI) (SpringerLink). In fraud detection, the choice of fusion technique can have a significant impact on performance, especially when the modalities contribute unequally to the classification task.

Our experimental data also contains both images and text. Inspired by their findings, we plan to apply this novel attention-based fusion approach to our Telegram dataset and compare the results with those obtained using concatenation. In our research, we compare relatively simple concatenation and attention-based fusion methods because the text commonly found on Telegram does not consist of fully descriptive sentences like the examples above; instead, Telegram texts often include everyday colloquial language. Therefore, we aim to determine which fusion method is better suited for our project.

#### ***4.2.2 Traditional Machine Learning Approaches***

Traditional machine learning models, such as Support Vector Machines (SVMs) and decision trees, have long been used for fraud detection Bansal et al. (2022). These models rely heavily on manually crafted features, such as transaction patterns, user behavior, or text-based indicators of fraud. Zhang et al. (2016) demonstrated the effectiveness of SVMs in detecting credit card fraud by modeling the transactional behaviors of users. However, the reliance on

manually engineered features can be a limitation, especially when dealing with complex or multimodal data.

### ***4.2.3 Deep Learning Models***

Deep learning approaches, particularly those utilizing neural networks, have gained popularity in fraud classification due to their ability to automatically learn high-level features from data Sengupta et al. (2020). Techniques like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have been used for image-based and text-based fraud detection, while multimodal networks incorporate both text and visual data to improve detection accuracy Yin et al. (2017); Chauhan et al. (2018); Jogin et al. (2018).

For example, Singhal Singhal et al. (2019) proposed the SpotFake framework, which integrates text and image data using CNNs and attention-based mechanisms to classify fake news. This approach can be adapted for fraud classification by training models on datasets containing fraudulent advertisements, social media posts, or suspicious transactions (SpringerLink) Singhal et al. (2019). Additionally, Khattar et al. (2019) developed a multimodal variational autoencoder for detecting fake news by leveraging both textual and visual content, a methodology that could be extended to identify fraudulent activities across multiple data types Khattar et al. (2019).

Recent advancements in transformer-based architectures, such as the Vision Transformer (ViT) Han et al. (2022) and Swin Transformer Liu et al. (2021), have further enhanced image-based classification tasks. ViT processes images as sequences of patches, leveraging the power of self-attention mechanisms Han et al. (2022), while Swin Transformer introduces a hierar-

chical design that captures both local and global features efficiently Liu et al. (2021). These models have shown superior performance over traditional CNNs and can be integrated with text-based embeddings, such as those generated by BERT, for multimodal fraud detection. Incorporating these state-of-the-art architectures into our framework will allow us to explore their efficacy in improving classification accuracy for Telegram data.

### 4.3 Methodology Overview

#### 4.3.1 Model Architecture

In this section, we outline our computational approach for sourcing data from public Telegram channels and detail our methodology for analyzing both the textual and image components of individual messages. Fig. 4.1 illustrates the diagram of our proposed method. Our model consists of four primary components:

1. **Data Collection:** We developed a program designed to efficiently organize and process downloaded data from Telegram chats. The program extracts essential details, including the posting channel, publisher, posting time, and content. After extraction, the data is carefully structured and stored in a MySQL database. For manual classification, the database table can be exported and saved as a CSV file for further analysis.
2. **Text Embedding:** We employ BERT as text embedding methods. The pre-trained model we are using is "bert-base-uncased". Through this model, BERT captures contextual information with a fixed output dimension of 768.
3. **Image Embedding:** CNN (ResNet50), Vision Transformer, and Swin Transformer are

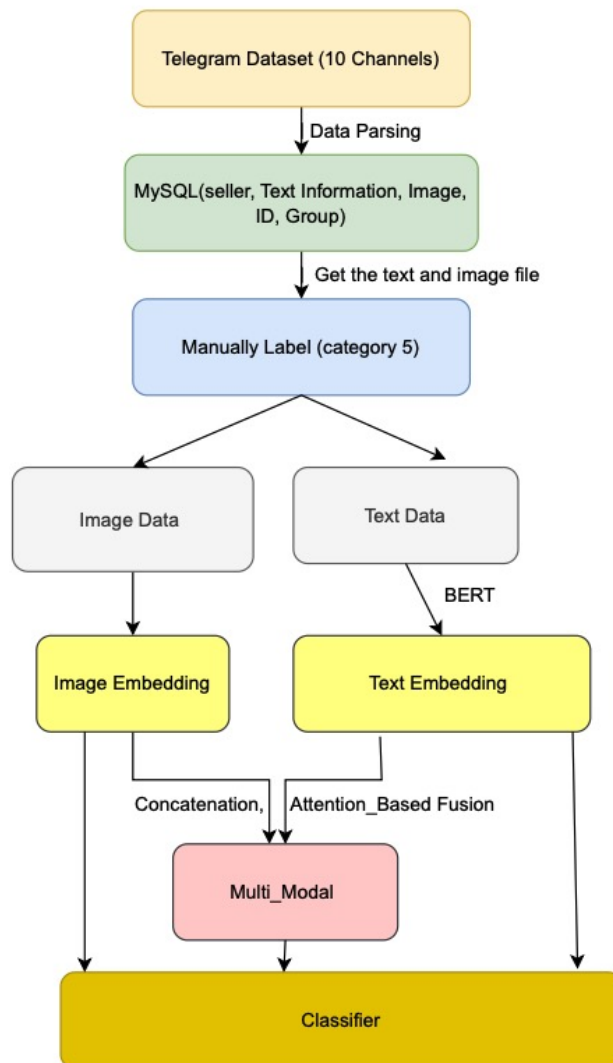


Figure 4.1 Model Architecture

used for image embeddings. CNN provides a 2048-dimensional feature vector, while ViT and Swin Transformer output 512-dimensional embeddings.

4. Fusion Mechanisms: Two fusion methods, simple concatenation and attention-based fusion, are applied to combine text and image features, respectively. The concatenation method merges the embeddings linearly, while attention-based fusion learns dynamic

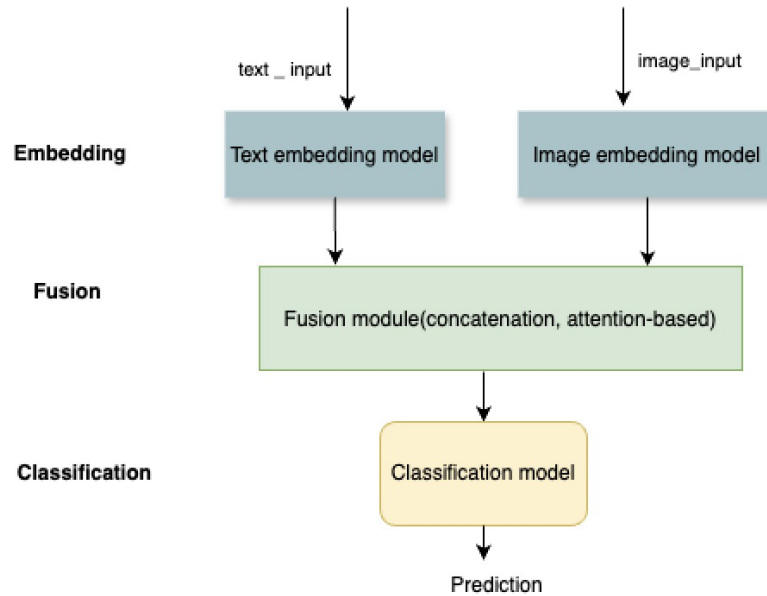


Figure 4.2 Early Fusion method

weights for each modality.

### 4.3.2 Fusion Strategies

There are 4 main strategies for multimodal fusion, they are: Early fusion, Intermediate fusion, Late fusion, and Hybrid fusion. In our project, we use the Early fusion method. In the Early fusion, we combined different modalities (text and image) at the input level. Fig. 4.2 shows the early fusion method in our project. In our project, two kinds of fusion methods are used to make a comparison.

Concatenation Fusion: Text and image embeddings are concatenated along their feature dimensions to form a unified multimodal representation as shown in (4.1) Karathanassi et al. (2007).

$$\text{fused\_features} = \text{concat}(\text{text\_emb}, \text{image\_emb}) \quad (4.1)$$

Where *fused\_features* is a combined feature vector,  $\text{concat}(\cdot)$  is the concatenation operation appends the image feature vector to the text feature vector, *text\_emb* text feature vector, *image\_emb* is the individual image feature vectors.

Attention-Based Fusion: Text and image embeddings are combined using learned attention weights. These weights allow the model to emphasize more informative features from each modality dynamically Mohla et al. (2020), which we can see in (4.2)

$$\text{fused\_features} = w_{\text{text}} \cdot \text{text\_emb} + w_{\text{image}} \cdot \text{image\_emb} \quad (4.2)$$

Where *fused\_features* represents the final combined feature vector that integrates information from both text and image modalities,  $w_{\text{text}}$  is the weight assigned to the text embeddings *text\_emb*,  $w_{\text{image}}$  is the weight assigned to the image embeddings *image\_emb*, *text\_emb* text feature vector, *image\_emb* is the individual image feature vectors.

### 4.3.3 Training and Evaluation

The model is trained using a cross-entropy loss function with the Adam optimizer. Each embedding method and fusion strategy is evaluated on a multi-class classification task. Metrics such as accuracy and F1-score are used to assess performance.

Evaluation Metrics: The performance evaluation includes Accuracy, which measures overall prediction correctness but may be less informative for imbalanced datasets; and F1-Score, which balances precision and recall, making it suitable for different kind of datasets.

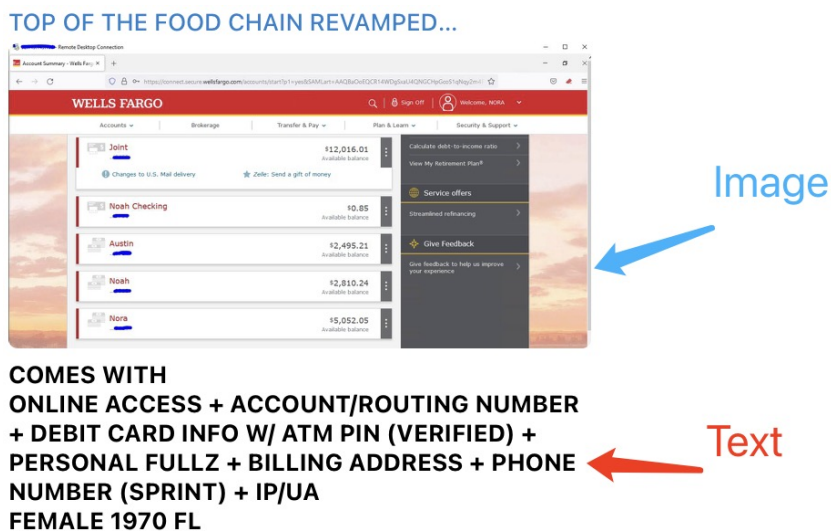


Figure 4.3 Text Message From Criminal

## 4.4 Experiments and Results

### 4.4.1 Dataset

We collected the data from 10 channels in Telegram. We keep the data that has both text and image pairs representing various fraud categories.

- Text Data: We are using the advertising language directly posted by criminals in Telegram channels. Some of the examples are shown in Fig. 4.3. We can see some words related to some illegal activities, such as 'presenting dump checking service' related to the dumps, and 'Target Account with balance' related to discount fraud!
- Image Data: Screenshots or advertisements related to the schemes posted by criminals in Telegram channels. In the Fig. 4.4. There are four example images in this figure; most of them are related to the bank account drops.

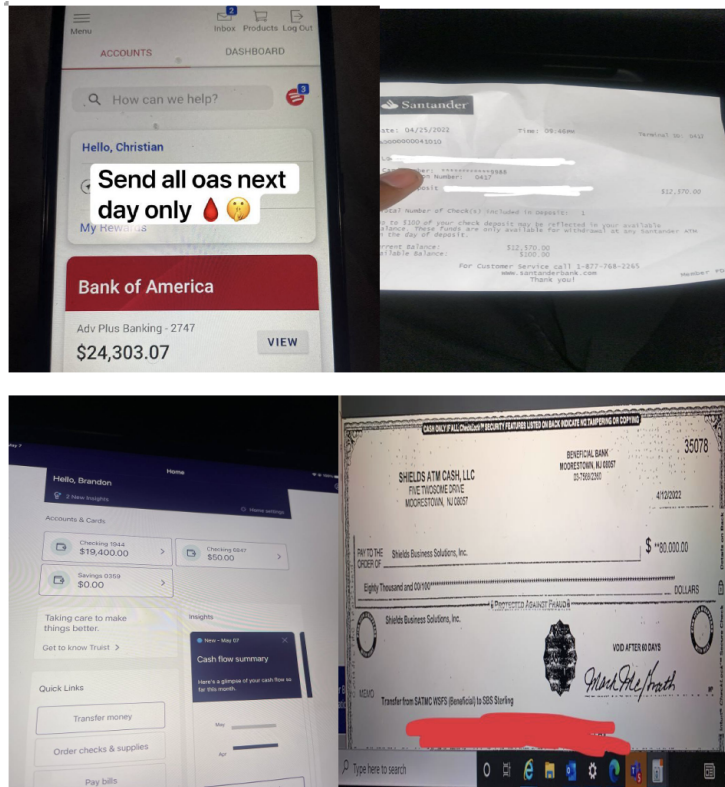


Figure 4.4 Image From Criminal

- Final dataset: After we store the table locally as a CSV file, our team engages in a manual categorization process to further refine and organize the information. The data originates from ten unique Telegram channels, each representing content from 5 distinct domains. The categorized domains include Credit Cards (CVV Dumps), Bank Account Drops, Personal Information, Promo Abuse and Discount Fraud, as well as Money transfer. The organization of the dataset is shown in Fig. 4.5.

#### 4.4.2 Implementation Details

- Framework: Python with PyTorch.

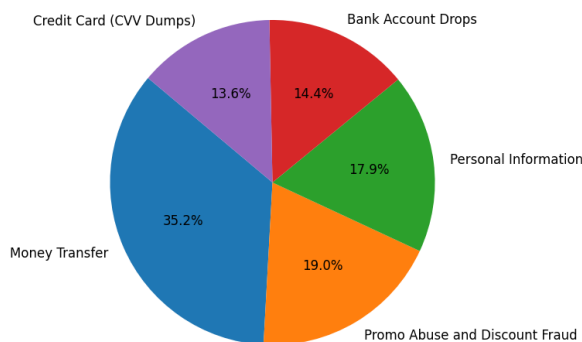


Figure 4.5 Data Distribution

- **Embeddings:** Precomputed using BERT (bert-base-uncased) for text, and CNN(ResNet50), Swin Transformer (Microsoft/swin-base-patch4-window7-224), and ViT (google/vit-base-patch16-224-in21k) for images.
- **Fusion:** Multimodal representations from the fusion layer are input into a neural network classifier. Single concatenation and attention-based fusion methods are used in our project to make a comparison
- **Validation:** 5-fold cross-validation is used for evaluation.
- **Early Stop:** We use the early stop method to find the best epoch for our project.
- **Learning Rate:** We start the training with a higher value, and then we lower it as the number of epochs increases (We start from a value of 0.01, then lower it to 0.005, and finally, to 0.0001).

#### 4.4.3 Performance Comparison

The table 4.1 compares the performance of different configurations, evaluating models on Accuracy and F1-Score metrics. It includes both single-modality setups (Text or Image

Only) and multi-modal setups combining image embeddings with BERT text embeddings.

We can tell that:

1. **Single-Modality Configurations:** Among single-modality setups, Text Only (BERT) achieves the best performance with an Accuracy of 85.5% and F1-Score of 83.8%, demonstrating the strength of BERT for text embedding. In contrast, Image Only configurations (ResNet50, ViT, and Swin Transformer) perform worse, with Swin Transformer slightly outperforming the others (Accuracy: 77.5%, F1-Score: 73.8%). This indicates that image embeddings alone are less effective than text embeddings in this task.
2. **Multi-Modal Configurations:** Combining image embeddings with BERT significantly enhances performance over single-modality setups. Swin Transformer + BERT (Attention-Based) achieves the best results with an Accuracy of 89.5% and F1-Score of 89.2%, demonstrating the effectiveness of attention-based fusion. Similarly, ResNet50 + BERT (Attention-Based) performs well (Accuracy: 88.1%, F1-Score: 87.9%). However, ViT + BERT (Attention-Based) shows a trade-off, achieving a high Accuracy of 89%.
3. **Impact of Fusion Strategies:** Attention-Based Fusion consistently outperforms Concatenation across all models. For example, Swin Transformer + BERT sees significant improvements in both Accuracy (from 86.5% to 89.5%) and F1-Score (from 84.2% to 89.2%). Similarly, ResNet50 + BERT achieves notable gains, with Accuracy increasing from 85.2% to 88.1% and F1-Score from 84.7% to 87.9%.

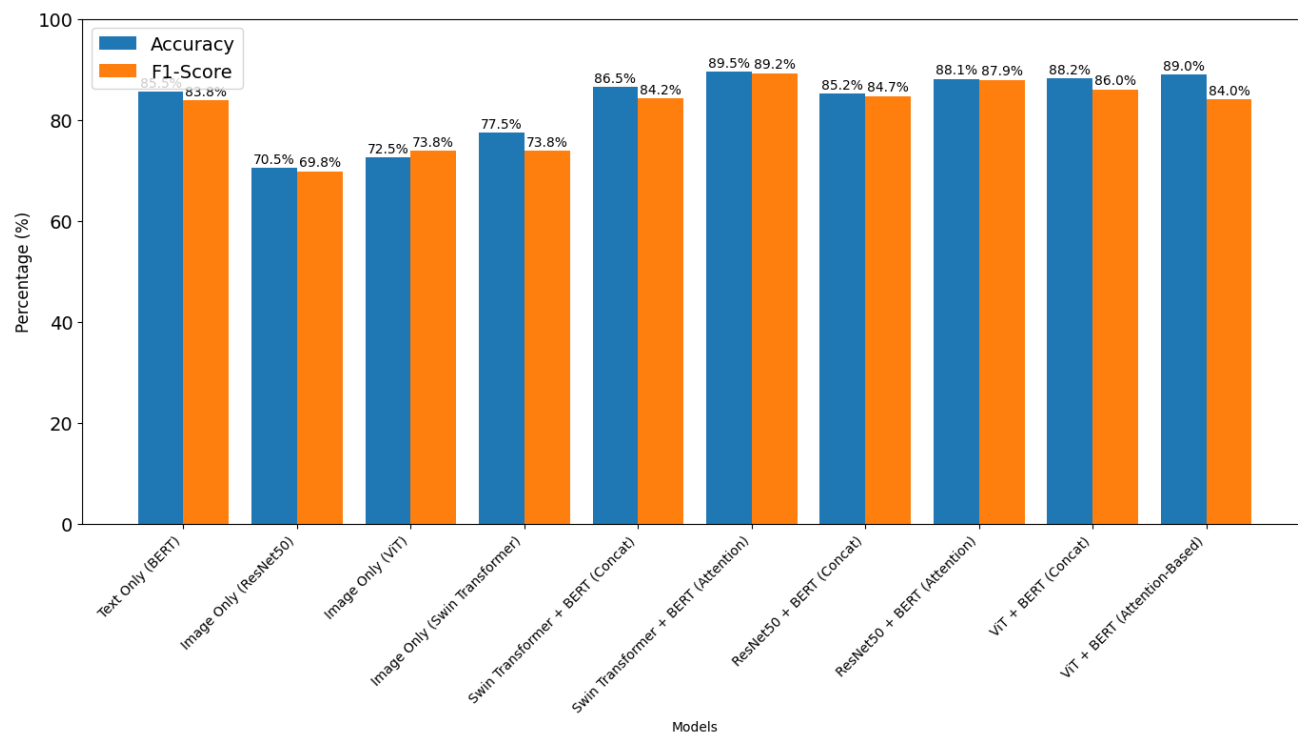
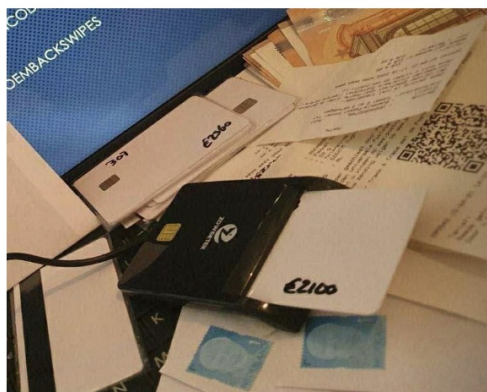


Figure 4.6 Performance of different Configurations

From Figure 4.6, we can see more clearly, multi-modal approach achieves better results.

Table 4.1 Performance Compare

Configuration	Accuracy	F1-Score
Text Only (BERT)	85.5%	83.8%
Image Only (ResNet50)	70.5%	69.8%
Image Only (ViT)	72.5%	73.8%
Image Only (Swin Transformer)	77.5%	73.8%
Swin Transformer + BERT (Concatenation)	86.5%	84.2%
Swin Transformer + BERT Attention	89.5%	89.2%
ResNet50 + BERT (Concatenation)	85.2%	84.7%
ResNet50 + BERT (Attention)	88.1%	87.9%
ViT + BERT Concatenation	88.2%	86.0%
ViT + BERT Attention-Based	89%	84.0%



Text: Message me for all  
 Promo Abuse and Discount Fraud

Sample 1



Text: Fresh don't matter ✓  
 Credit Card

Sample 2

Figure 4.7 Example of correct and mistake

#### 4.4.4 Misclassification Analysis in Swin Transformer + BERT Model

Although the Swin Transformer + BERT model shows high accuracy and overall performs well, there are some instances where certain content has been misclassified. In Figure 4.7, there are two samples of the classification. For sample 1, there are lots of cards; the text message for this image is "Message me for all", the real result should be credit card fraud, but the prediction is promo and Discount Fraud. That means this model defines this sample as some product. For sample 2, even text for the same is "Fresh don't matter", which has nothing to do with credit cards, our model can still figure out that this is a credit card fraud. We believe that overly complex image backgrounds and insufficiently clear text descriptions are the main reasons for the errors in samples 1 and 2.

## 4.5 Conclusion

This research highlights the effectiveness of multimodal approaches in fraud classification on Telegram. By combining BERT for text embeddings with image embeddings generated from ResNet50, Swin Transformer, and ViT, and leveraging attention-based fusion, we achieved state-of-the-art performance. An ablation study underscored the significance of each component, while cross-validation ensured the reliability of our evaluation. Notably, the combination of Swin Transformer embeddings with attention-based fusion delivered the best results, displaying the critical role of global feature extraction in processing image data.

## CHAPTER 5

### WEAKLY SUPERVISED KNOWLEDGE BASE CONSTRUCTION FOR TELEGRAM GIFT-CARD FRAUD MESSAGES

#### 5.1 Introduction

Telegram and similar encrypted messaging platforms are increasingly utilized to advertise and trade discounted or resold digital assets Garkava et al. (2024), specifically gift cards from major retailers such as Amazon, Vanilla, and Walmart. These assets often function as semi-anonymous payment instruments within underground markets. While some activity may involve legitimate bulk resellers, a significant proportion is associated with gray or illicit finance, including the laundering of funds from credit-card fraud, coupon abuse schemes, and the exchange of stolen digital balances for cryptocurrency La Morgia et al. (2021). The automated detection and analysis of these fraudulent transactions in open messaging channels represent a pressing challenge for digital forensics and financial crime research Rawat et al. (2023).

Automatically extracting *valid* discount relations from Telegram posts is complex. Three practical difficulties are observed: (i) The message text is *loosely formatted* and often spans multiple lines, featuring irregular typography, embedded emojis, and mixed currency notations (e.g., “500USD = 280 USDT ”); (ii) A single promotional message frequently lists multiple brands and denominations (e.g., “Amazon 500 / 250 / 100” within one post) without standardized separators; and (iii) Manual annotation of thousands of heterogeneous messages is resource-intensive, prone to subjective bias, and ultimately infeasible for dynamic, real-

world forensic pipelines. Consequently, traditional supervised learning methods encounter substantial bottlenecks when applied to this dynamic, weakly structured data environment.

Traditional fraud detection systems primarily focus on binary classification: determining whether a message or an account is fraudulent or benign Roy et al. (2025); West & Bhattacharya (2016); Dargahi Nobari et al. (2017). While effective for initial triage, these methods offer limited investigative insight into the financial mechanics of the illicit trade. They fail to capture the nuances of criminal pricing strategies, such as brand-specific price floors, denomination-dependent discount rates, or the cryptocurrency conversion rates used for payment (e.g. USD to USDT). Knowledge Base Construction (KBC) shifts the analytical focus from simple detection to detailed structural comprehension. By extracting structured triples, we facilitate deeper forensic analysis, allowing investigators to track underground market trends, identify suspicious pricing anomalies across different brands, and build predictive models based on concrete financial relationships rather than merely on textual patterns. This structured approach moves beyond a simple 'yes/no' determination, providing the essential 'what, and how much' necessary for comprehensive cybercrime intelligence.

To address the limitations of manual annotation and the volatility of the Telegram environment Kumar et al. (2023), we propose an end-to-end weakly supervised learning framework designed to extract structured discount relations from these advertisements. The system automatically parses chat exports, identifies potential price–discount pairs, and utilizes a heuristic-based weak labeling process to generate initial supervision signals without any human annotation. These labels are subsequently used to train a discriminative classifier

that refines the predictions and outputs the most plausible discount triples of the form (brand, original\_price, discounted\_price).

Our main contributions are summarized as follows:

- **Telegram-Specific Candidate Generation:** We developed a generation module capable of identifying price expressions within the complex Telegram chat format, handling both same-line patterns (“\$500 = 280”) and cross-line associations (“Amazon GC \$500” followed by “sell 280”), alongside canonicalizing common brand aliases.
- **Refined Weak-Labeling Scheme:** We designed a multi-cue scoring function that integrates price-drop constraints, discount keywords, message aesthetics (emojis/capitalization), and positional cues to compute a probabilistic score for each candidate pair, thereby providing a robust soft supervision signal.
- **Human–Model Agreement Calibration:** A human validation study demonstrated that setting the weak-label threshold to  $T = 0.80$  maximizes consistency ( $\sim 80.6\%$ ) between the automated labels and human judgment, effectively balancing supervision signal quality.
- **Feature Analysis and Interpretability:** We evaluated structure-only, semantic-only, and feature-fused models, confirming that structural features alone (emoji ratio, uppercase ratio, line patterns) are highly predictive indicators for fraudulent discount detection, with fusion providing modest supplementary gains.

This work introduces one of the first weakly supervised, end-to-end pipelines tailored for fine-grained knowledge extraction from Telegram-based gift-card fraud posts. The methodology is readily extendable to related financial-fraud categories, including coupon abuse and digital money-transfer schemes.

## 5.2 Related Work

Research into financial-fraud detection has traditionally relied on supervised machine learning applied to transaction metadata or social-media content West & Bhattacharya (2016); Roy et al. (2025). Supervised classifiers, including Random Forests and deep neural networks, have proven effective for phishing, credit-card misuse, and money-laundering detection Shah et al. (2020). However, these methods require large and accurately annotated datasets, which are often difficult and expensive to obtain in underground or illicit online domains. Moreover, supervised models tend to struggle with the rapidly evolving tactics and domain shifts commonly observed in such environments. Recent studies have therefore explored *weakly supervised* or *distantly supervised* approaches, in which heuristic or rule-based signals are used to generate approximate labels, reducing the dependence on costly manual annotation while maintaining scalability Wu et al. (2018).

Weak supervision Ratner et al. (2017) presents a compelling alternative, utilizing noisy or heuristic rules to generate supervisory signals in lieu of expensive manual annotation. The Snorkel framework Ratner et al. (2020) formalized the combination of multiple *labeling functions* via a probabilistic label model, and demonstrated that weakly labeled data can

Table 5.1 Comparison with Representative Approaches in Financial-Fraud and Telegram Analysis.

<b>Approach</b>	<b>Supervision Paradigm</b>	<b>Granularity</b>	<b>Annotation</b>
Traditional ML on transactions	Fully Supervised	Record-level	
Social-media classifiers	Supervised / Heuristic	Post-level	
Snorkel / Fonduer	Weak Supervision	Relation-level	
Telegram network analysis Kumar et al. (2023)	Manual / Keyword	Channel-level	
<b>Our system</b>	<b>Weak Supervision</b>	<b>Message-level Triples</b>	

train high-performing discriminative models. Fonduer Wu et al. (2018) extended this concept to multimodal knowledge base construction from richly formatted documents, leveraging both textual and structural cues. These works establish that weak supervision is a viable path for scaling information extraction tasks when gold-standard data is unavailable.

In the cybercrime domain, previous analytical efforts largely focused on studying Telegram network structures or applying manual keyword filtering to identify suspicious channels Roy et al. (2025); West & Bhattacharya (2016); Dargahi Nobari et al. (2017) . While some previous work primarily addressed the classification of Telegram messages into financial-fraud categories Gao & Wu (2024, 2025), the current study extends this line of research toward content-level understanding. We focus on extracting fine-grained information from individual posts, such as brands, original and discounted prices, and discount rates, using a weakly supervised approach that combines structural and semantic features. Our approach introduces a message-level weakly supervised pipeline that extracts fine-grained discount triples—composed of brand, original price, and discounted price—from the loosely formatted Telegram advertisements, a format distinct from traditional web pages or structured documents.

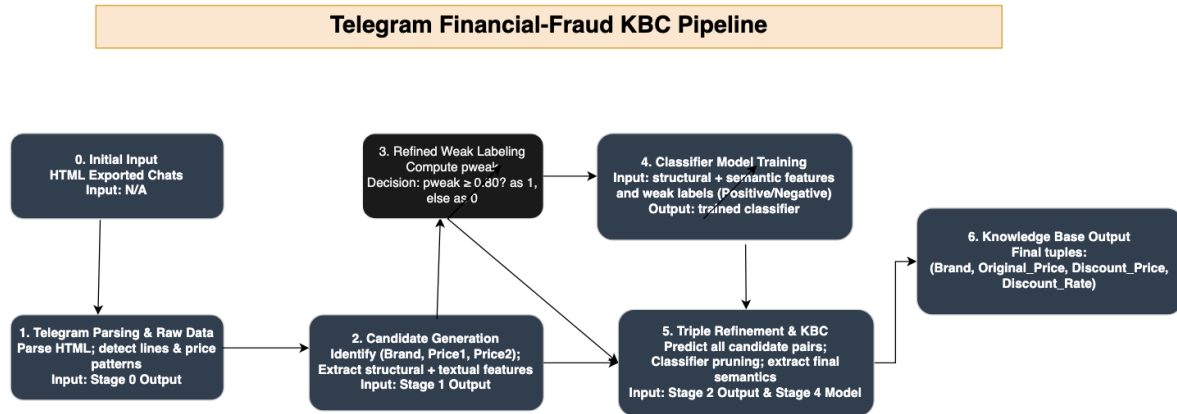


Figure 5.1 Overall pipeline of the Weakly Supervised KBC system.

## 5.3 Methodology

### 5.3.1 System Overview

The proposed end-to-end pipeline (Fig. 5.1) converts Telegram HTML exports into structured discount triples via five sequential stages: HTML parsing, candidate generation, probabilistic weak labeling, classifier model training, and final triple refinement. Each input message is segmented into lines, and potential price-discount pairs are identified using regular expressions combined with structural proximity cues.

### 5.3.2 Telegram Parsing and Candidate Generation

The initial step is a structured parsing of the raw Telegram exports, which handles the multi-line nature of the messages. The parser segments the chat log into individual messages and each message into sequential lines, yielding  $(text\_id, line\_no, line\_text)$  entries.

**Brand and Price Pattern Recognition:** We utilize customized regular expressions to lo-

cate monetary expressions, categorized as: (1) Original Price candidates (high-denomination values like “\$500”, “1000usd”) and (2) Discount Price candidates (lower values often associated with explicit sale keywords or crypto terms like “USDT”). A crucial utility is the Brand Alias Dictionary, which normalizes abbreviations (e.g., “amz”, “vvisa”) to canonical brand names.

**Cross-Line Relation Capture:** Given the conversational and fragmented chat structure, a single discount relation frequently spans across non-contiguous lines:

*Line n: Amazon GC \$500, \$250 stock now.*

*Line n + 1: Price 280, 150.*

Our generator employs a sequential window (up to three lines) to associate an Original Price on line  $n$  with a Discount Price on line  $m \geq n$ . A candidate pair  $c = (P_{orig}, P_{disc})$  is generated only if a preliminary Price Drop Validation ( $P_{orig} > P_{disc}$ ) holds and they occur within this localized window. This mechanism ensures that structurally separated but semantically linked price pairs are effectively captured. The generator also extracts key structural indicators for each pair, including `same_line`, `same_column` (e.g., prices aligned vertically but not explicitly linked), and `has_emoji`.

### ***5.3.3 Refined Weak Labeling***

Rather than relying on simple binary rules, we assign a probabilistic weak score to each candidate based on aggregated heuristic cues, as detailed in Algorithm 2. The multi-cue scoring function,  $s$ , serves as the core of our weak supervision, aggregating confidence signals

for a candidate pair  $c = (P_{orig}, P_{disc})$ . The raw score  $s$  (analogous to the Python variable ‘votes’) is calculated as a weighted linear combination of multiple indicators:

$$s(c) = \sum_i w_i \cdot I_i$$

The indicator weights ( $w_i$ ) are empirically determined based on domain knowledge. For instance, a confirmed price drop receives a weight of +1.0, while an invalid price drop receives a severe penalty of -2.0. Similarly, the same\_column structural feature, which strongly indicates a listing of unrelated denominations, receives a penalty of -0.8. The aggregated score  $s$  is then linearly normalized and clipped to the range  $[0, 1]$  to produce the weak probability  $p_{\text{weak}}$ :  $p_{\text{weak}} = \text{clip}((s + 3)/6, 0, 1)$ , where the constants 3 and 6 scale the typical vote range of  $[-3, 3]$  to the probability range. Candidates meeting the threshold  $T = 0.80$  are selected as high-confidence positive training examples.

#### ***5.3.4 Triple Export for Knowledge Base Construction***

Each successfully labeled record is structured as a tuple (brand, original\_price, discount\_price, discount\_rate). This unit constitutes a domain-specific knowledge item that explicitly preserves multiple denominations for the same brand, enabling fine-grained analysis of discount ratio variation across card values. This design choice prevents information loss that would occur in a flattened relational representation. The final refined triples are exported to `kbc_triples_final_refined.xlsx` for forensic ingestion.

Table 5.2 Dataset Statistics for Telegram Gift-Card Messages

<b>Metric</b>	<b>Count</b>	<b>Notes</b>
Messages parsed	~ 795	Distinct Telegram posts
Brands detected	4	Amazon, Vanilla, Walmart, Sephora
Candidate pairs	~ 2,710	Price/discount combinations
Positive weak labels	~ 20.4%	Using threshold $T = 0.80$

## 5.4 Experimental and Results

### 5.4.1 Dataset

Our evaluation utilized several hundred Telegram gift-card messages gathered from open channels promoting brands including Amazon, Vanilla, Walmart, Sephora. Following the candidate generation step, a total of approximately 2,710 candidate pairs were extracted.

Table 5.2 summarizes the dataset scale.

### 5.4.2 Models and Features

We benchmarked the performance of Random Forest, Support Vector Machine (SVM), and an attention-based feature fusion model. The models were trained using three comparative feature sets: (1) structure-only, (2) semantic-only, and (3) fused features.

### 5.4.3 Detailed Feature Engineering

The input features for the discriminative models were categorized into two main groups for comparative evaluation:

- **Structural Features (S-Features):** These features characterize the layout and stylistic presentation of the advertisement, which often serves as a strong indicator of fraudulent

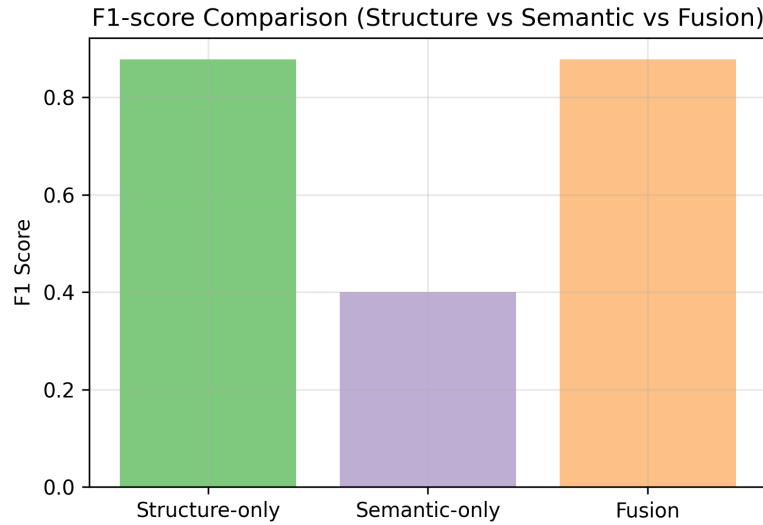


Figure 5.2 Performance comparison of structure-only, semantic-only, and fusion features.

intent. Key S-Features include: (a) Emoji Ratio(density of emojis), (b) Uppercase Ratio (density of capitalized text), (c) Line Distance between the original and discounted price mentions, (d) Presence of Separator Symbols (e.g., '=', '—'), and (e) Message Length.

- Semantic Features (T-Features): These features capture the contextual meaning of the surrounding text. We utilized MiniLM embeddings to generate a vector representation of the text snippet localized around the candidate pair. This embedding captures relevant contextual semantics, enabling generalization beyond exact keyword matching.

The Fused Model combined both S-Features and T-Features, processing the concatenated vector through an attention layer designed to dynamically weigh the contribution of structural versus semantic signals prior to final classification.

#### ***5.4.4 Model Comparison***

The Random Forest model demonstrated the highest overall performance metrics, followed closely by the SVM and the attention-based fusion model. The strong performance of Random Forest is consistent with its established capacity to handle high-dimensional, mixed-type feature spaces and its resilience against noise inherent in the weakly labeled training data, a factor frequently leveraged in fraud detection applications. Crucially, the structure-only features achieved strong  $F_1$  and recall values on their own, confirming the central role that formatting, layout, and symbol patterns play in encoding salient discount information within Telegram advertisements.

#### ***5.4.5 Human–Model Agreement***

A systematic threshold sweep across the range of 0.4 to 0.8 showed that the highest alignment between the probabilistic weak labels and independent human annotations was achieved at  $T = 0.80$ , yielding an agreement rate of  $\sim 80.6\%$  (Fig. 5.3). This calibration confirms that the aggregated heuristic cues effectively capture the pattern recognition skills applied by human analysts, providing a high-quality supervision signal.

#### ***5.4.6 Interpretation and Knowledge Base Analysis***

The utility of our KBC system is best demonstrated by the actionable knowledge generated. The extracted tuples enable analysis of brand-specific discount trends.

**Brand-Specific Discount Trends:** We analyzed the observed discount rates across the most common brands in our knowledge base. Table 5.3 presents a summary of the typical

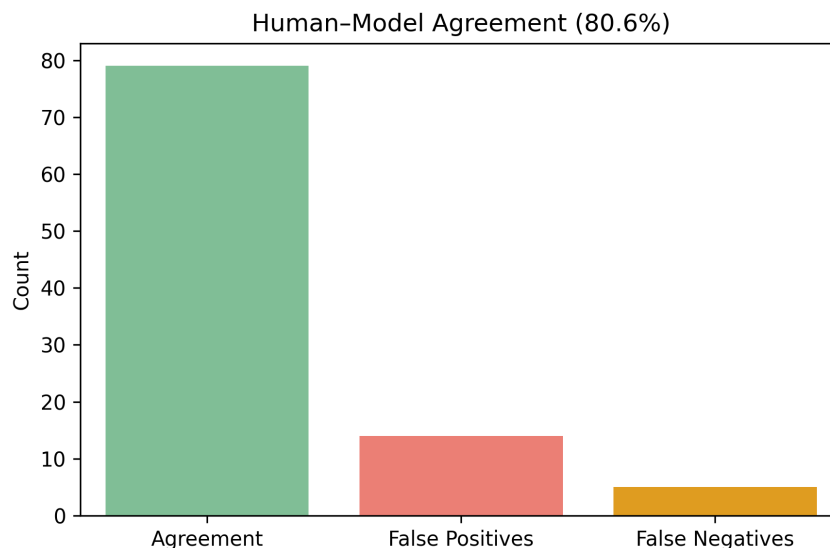


Figure 5.3 Human-model agreement versus weak-label threshold.

ranges observed in the underground market data.

Table 5.3 Typical Extracted Discount Ranges by Brand (Selected Examples)

Brand	Original Price Range (USD)	Observed Discount Rate
Amazon	100 - 2000	40% - 65%
Vanilla	250 - 2000	50% - 75%
Walmart	50 - 250	35% - 55%

The empirical data validates hypotheses from financial forensics: highly liquid, general-purpose cards (such as Vanilla) are often subject to more aggressive discounting than cards tied to specific retailers (like Walmart), likely reflecting a greater urgency for rapid cash-out or higher perceived risk. Furthermore, we consistently observed that larger denominations carried a slightly higher discount percentage than smaller ones within the same brand, validating the need for the full (brand, original\_price, discount\_price) tuple structure for forensic utility.

## 5.5 Conclusions

We presented a weakly supervised knowledge base construction system specifically designed for extracting structured financial intelligence from Telegram gift-card fraud messages. The system leverages automated parsing, heuristic-driven probabilistic labeling, and classifier-based refinement. The resultant structured tuples (brand, original\_price, discount\_price, discount\_rate) function as granular knowledge units, capturing both multiple denominations and varying discount strengths per brand. Our experimental analysis confirmed that structural features provide a high degree of predictive accuracy in this domain, with feature fusion offering marginal gains.

In future work, we plan to generalize this framework to encompass related financial-fraud domains, including promo-abuse and complex money-transfer schemes. We also intend to develop an automated method for decomposing the extracted tuples into canonical (subject, relation, object) triples, facilitating seamless integration into larger knowledge graph systems. Specifically, our immediate next steps will focus on enhancing the theoretical rigor and robustness of the system:

- **Formal Probabilistic Labeling Model:** We will replace the current empirically weighted weak-label scoring function with a formal Probabilistic Label Model (e.g., utilizing a dependency graph model similar to Snorkel). This will allow us to systematically quantify and de-noise the label uncertainty introduced by each heuristic cue, providing a more theoretically sound foundation for the weak supervision pipeline.

- **Robustness and Evasion Analysis:** We plan to conduct a thorough analysis of the system’s robustness against adversarial changes in message formatting, such as the systematic removal of emojis or the introduction of novel obfuscation techniques. This will involve developing methods to maintain high extraction performance even as fraudsters adapt their communication patterns.
- **Benchmarking and Contribution Quantification:** To rigorously quantify the benefits of our approach, we will establish stronger baselines. This includes formally comparing the final classifier’s performance against a highly-tuned Rule-Based System (based purely on the weak-label heuristics) and a small-scale Fully Supervised Baseline to accurately measure the performance gain attributed to the weak supervision and refinement steps.

---

**Algorithm 2** Refined Weak Labeling with Structural Cues
 

---

**Require:** Candidate set  $C$ , Threshold  $T = 0.80$

- 1: Define Positive Cues ( $P_{\text{Cues}}$ ) and Negative Cues ( $N_{\text{Cues}}$ )
- 2: **for** each candidate  $c = (P_{\text{orig}}, P_{\text{disc}}) \in C$  **do**
- 3:   Initialize raw score  $s \leftarrow 0.0$
- 4:   Extract text context  $T_{\text{txt}}$  and structural features  $F_{\text{struct}}$   
    {Aggregating Positive Votes}
- 5:   **if**  $P_{\text{disc}} < P_{\text{orig}}$  **then**
- 6:      $s \leftarrow s + 1.0$  {Confirmed Price Drop}
- 7:   **end if**
- 8:   **if**  $T_{\text{txt}}$  contains any  $w \in P_{\text{Cues}}$  **then**
- 9:      $s \leftarrow s + 1.0$  {Discount Keyword Match}
- 10:   **end if**
- 11:   **if**  $10 \leq \text{rate} \leq 70$  **then**
- 12:      $s \leftarrow s + 1.0$  {Discount Rate Plausibility}
- 13:   **end if**
- 14:   **if**  $F_{\text{struct}}$  includes same-line indicator **then**
- 15:      $s \leftarrow s + 0.9$  {High Positional Proximity}
- 16:   **end if**
- 17:   **if**  $F_{\text{struct}}$  includes emoji indicator **then**
- 18:      $s \leftarrow s + 0.3$  {Aesthetic Flag}
- 19:   **end if**  
    {Applying Penalties (Negative Votes)}
- 20:   **if**  $P_{\text{disc}} \geq P_{\text{orig}}$  **then**
- 21:      $s \leftarrow s - 2.0$  {Invalid Price Drop (Strong Penalty)}
- 22:   **end if**
- 23:   **if**  $T_{\text{txt}}$  contains any  $w \in N_{\text{Cues}}$  **then**
- 24:      $s \leftarrow s - 1.0$  {Negative Keyword (e.g., "sold", "stock")}
- 25:   **end if**
- 26:   **if**  $F_{\text{struct}}$  includes same-column indicator **then**
- 27:      $s \leftarrow s - 0.8$  {Structural Misalignment Penalty}
- 28:   **end if**
- 29:   **if** Brand is Unknown AND rate is not percentage-based **then**
- 30:      $s \leftarrow s - 0.5$
- 31:   **end if**
- 32:   **if**  $\text{length}(T_{\text{txt}}) < 10$  **then**
- 33:      $s \leftarrow s - 0.3$  {Too short context penalty}
- 34:   **end if**  
    {Normalize and Assign Label}
- 35:    $p_{\text{weak}} \leftarrow \text{clip}((s + 3)/6, 0, 1)$
- 36:    $c.\text{label} \leftarrow (p_{\text{weak}} \geq T)$
- 37: **end for**
- 38: **return** Labeled candidates

---

## CHAPTER 6

### CONCLUSION AND FUTURE WORK

#### 6.1 Summary of Research

This dissertation presented a comprehensive study on artificial intelligence approaches for detecting and understanding financial fraud activities in Telegram, one of the most active communication platforms for underground markets. Unlike traditional financial datasets that are highly structured, Telegram data is loosely formatted, containing a mix of textual, visual, and symbolic elements. The research aimed to address the analytical challenges posed by this type of data through a series of interconnected studies focusing on feature selection, classification, multimodal learning, and knowledge base construction.

The first part introduced AutoCut-2D, an adaptive feature selection method that determines optimal cutoff points across multiple importance metrics. This approach improved model interpretability and reduced computational complexity while maintaining high predictive accuracy. The second study focused on classifying Telegram financial fraud posts into multiple categories using a combination of traditional machine learning and embedding-based representations. The third extended the investigation into multimodal analysis by integrating BERT-based textual embeddings and Swin Transformer visual embeddings through attention-based fusion, resulting in a significant performance gain over single-modality models. Finally, the dissertation developed a weakly supervised knowledge base construction (KBC) framework that automatically extracts structured triples—such as (brand, price, discount)—from loosely formatted Telegram messages. Together, these studies provide a

foundation for building automated, interpretable, and scalable systems for cybercrime analysis.

## 6.2 Major Findings and Contributions

The key contributions of this dissertation are summarized as follows:

- Developed AutoCut-2D, a two-dimensional adaptive feature selection technique that enhances both model efficiency and interpretability.
- Built a robust classification pipeline for identifying financial fraud types in Telegram messages using diverse embedding and machine learning techniques.
- Proposed a multimodal learning framework that combines textual and visual representations via attention-based fusion, achieving superior detection accuracy.
- Designed and implemented a weakly supervised KBC pipeline capable of extracting structured knowledge from loosely formatted Telegram data.
- Demonstrated the effectiveness of integrating structural and semantic representations for real-world cybercrime intelligence analysis.

## 6.3 Discussion and Implications

This research provides both theoretical and practical advancements in AI-based financial fraud detection. From an academic perspective, the proposed models contribute to multimodal representation learning and weak supervision methodologies. Practically, the developed frameworks can assist cybersecurity analysts, law enforcement agencies, and financial

institutions in automating fraud monitoring, detecting abnormal behaviors, and generating structured intelligence from open-source platforms. The combination of feature selection, multimodal learning, and knowledge extraction offers a promising paradigm for data-driven cybercrime investigation.

Although the results are promising, several limitations remain. The study relies primarily on publicly available Telegram data, which may not fully represent the diversity of private or encrypted communications. The weakly supervised approach introduces potential labeling noise that could affect model precision. Additionally, multimodal fusion was evaluated on limited visual samples, and extending it to larger datasets may require further optimization and computational resources.

Future research will extend this work in several directions. First, the knowledge base construction pipeline will be expanded to support multilingual and cross-platform data sources, enabling broader coverage of global fraud activities. Second, self-supervised and continual learning techniques will be explored to improve model robustness against evolving fraud patterns. Third, the integration of reasoning and inference mechanisms into the constructed knowledge base will be investigated to enable automated hypothesis generation and fraud scenario analysis. Lastly, the research can be extended to other domains of cybercrime, such as cryptocurrency scams and dark web marketplaces, further enhancing the applicability of AI-driven intelligence systems.

## 6.4 Conclusion

In conclusion, this dissertation establishes a unified and interpretable AI framework for analyzing loosely formatted Telegram data to detect and understand financial fraud. By combining feature selection, multimodal fusion, and weakly supervised knowledge extraction, it contributes both conceptual innovation and practical tools to the field of cybercrime intelligence. The findings not only advance academic understanding of multimodal and weakly supervised learning but also provide actionable insights for real-world fraud prevention and investigation.

## Acknowledgment

This work was supported in part by the Department of Computer Science at Georgia State University. The authors thank colleagues and reviewers for their insightful feedback, and acknowledge the assistance of the research group members in data preparation and evaluation.

## CHAPTER 7

### PUBLICATIONS

1. Chunlan Gao, Zhang, Y., Lo, D., Shi, Y., & Huang, J. (2022). "Improving the Machine Learning Prediction Accuracy with Clustering Discretization." In IEEE CCWC, 2022.
2. Chunlan Gao, Shahriar, H., Lo, D., Shi, Y., & Qian, K. "Improving the Prediction Accuracy with Feature Selection for Ransomware Detection." In IEEE Compsac, 2022.
3. Chunlan Gao, Yong Shi. "Prediction Performance Analysis for ML Models Based on Data Imbalance and Bias." ACM Southeast Conference, 2024.
4. Chunlan Gao, Yubao Wu. "Category Prediction of Financial Fraud Related Posts in Telegram Group Chats" ICSC, 2024.
5. Chunlan Gao, Yubao Wu. "Recognizing the Financial Fraud Type of Telegram Advertisement Posts Based on the Multimodal Learning." ISDFS, 2025.
6. Chunlan Gao, Yubao Wu. "AutoCut-2D: Enhancing Secure AI for Telegram Financial Fraud Detection via Elbow-Based Feature Selection" DSC, 2025.
7. Chunlan Gao, Yubao Wu. "Weakly Supervised Knowledge Base Construction for Telegram Gift-Card Fraud Messages" ISDFS, 2026 (accepted).

## REFERENCES

- Abdulhammed, R., Musafar, H., Alessa, A., Faezipour, M., & Abuzneid, A. 2019, *Electronics*, 8, 322
- Aggarwal, C. C., et al. 2018 (Springer)
- Akbari, A., & Gabdulhakov, R. 2019, *Surveillance and Society*, 17
- Al-Hashedi, K. G., & Magalingam, P. 2021, *Computer Science Review*, 40, 100402
- Al-Khater, W. A., Al-Maadeed, S., Ahmed, A. A., Sadiq, A. S., & Khan, M. K. 2020, *IEEE access*, 8, 137293
- Alaparthi, S., & Mishra, M. 2020, arXiv preprint arXiv:2007.01127
- Alimi, O. A., Ouahada, K., & Abu-Mahfouz, A. M. 2020, *IEEE Access*, 8, 113512
- Arauzo-Azofra, A., Aznarte, J. L., & Benítez, J. M. 2011, *Expert systems with applications*, 38, 8170
- Bansal, M., Goyal, A., & Choudhary, A. 2022, *Decision Analytics Journal*, 3, 100071
- Barker, K. J., D'amato, J., & Sheridan, P. 2008, *Journal of financial crime*, 15, 398
- Bello, O. A., & Olufemi, K. 2024, *Computer science & IT research journal*, 5, 1505
- Benner, P., Gugercin, S., & Willcox, K. 2015, *SIAM review*, 57, 483
- Berry, M. W., & Kogan, J. 2010, *Text mining: applications and theory* (John Wiley & Sons)
- Bolón-Canedo, V., Sánchez-Marroño, N., & Alonso-Betanzos, A. 2015, *Knowledge-based systems*, 86, 33
- . 2016, *Progress in Artificial Intelligence*, 5, 65

- Bondielli, A., & Marcelloni, F. 2019, *Information sciences*, 497, 38
- Bose, A., Hu, X., Shin, K. G., & Park, T. 2008, in *Proceedings of the 6th international conference on Mobile systems, applications, and services*, 225–238
- Bozhenko, V. V., Mynenko, S. V., & Shtefan, A. 2022
- Brodley, C. E., & Utgoff, P. E. 1995, *Machine learning*, 19, 45
- Buczak, A. L., & Guven, E. 2015, *IEEE Communications surveys & tutorials*, 18, 1153
- Chauhan, R., Ghanshala, K. K., & Joshi, R. 2018, in *2018 first international conference on secure cyber computing and communication (ICSCCC)*, IEEE, 278–282
- Chen, X.-w., & Jeong, J. C. 2007, in *Sixth international conference on machine learning and applications (ICMLA 2007)*, IEEE, 429–435
- Chen, Y., Zhang, H., Liu, R., Ye, Z., & Lin, J. 2019, *Knowledge-Based Systems*, 163, 1
- CNBC. 2024, Who is Telegram Founder Pavel Durov and Why Was He Arrested?, <https://www.cnbccom/2024/08/26/who-is-telegram-founder-pavel-durov-and-why-was-he-arrested.html>
- Costa Pereira, J., Coviello, E., Doyle, G., Rasiwasia, N., Lanckriet, G. R., Levy, R., & Vasconcelos, N. 2014, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36, 521
- Craja, P., Kim, A., & Lessmann, S. 2020, *Decision Support Systems*, 139, 113421
- Daraojimba, R. E., Farayola, O. A., Olatoye, F.-m. O., Mhlongo, N., & Oke, T. T.-l. 2023, *Finance & Accounting Research Journal*, 5, 342
- Darban, Z. Z., & Valipour, M. H. 2022, *Expert Systems with Applications*, 200, 116850

- Dargahi Nobari, A., Reshadatmand, N., & Neshati, M. 2017, in Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2035–2038
- Donoho, D., & Jin, J. 2008, Proceedings of the National Academy of Sciences, 105, 14790
- Du Rove's Channel. 2024, Du Rove's Channel, <https://www.t.me>
- Eason, G., Noble, B., & Sneddon, I. N. 1955, Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences, 247, 529
- Europol. 2017, Internet Organised Crime Threat Assessment (IOCTA 2017\)) (European Union Agency for Law Enforcement Cooperation (Europol))
- Faruk, M. J. H., Masum, M., Shahriar, H., Qian, K., & Adnan, M. I. 2022, in Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), 5369–5377
- Fisette, M. V. M. 2017
- Fung, G., & Mangasarian, O. L. 2001, in Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '01) (San Francisco, California: Association for Computing Machinery), 77–86
- Gao, C., & Wu, Y. 2024
- Gao, C., & Wu, Y. 2025, in 2025 13th International Symposium on Digital Forensics and Security (ISDFS), 1–6
- Gao, P., Jiang, Z., You, H., Lu, P., Hoi, S. C., Wang, X., & Li, H. 2019, in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 6639–6648
- Garg, D., Thakral, A., Nalwa, T., & Choudhury, T. 2018, in Proceedings of the 2018 International Conference on Advances in Computing, Communications and Engineering

(ICACCE)

- Garkava, T., Moneva, A., & Leukfeldt, E. R. 2024, Trends in Organized Crime, 1
- Ghosh, S., Das, A., Porras, P., Yegneswaran, V., & Gehani, A. 2017, in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1793–1802
- Gillis, N. 2020, Nonnegative matrix factorization (SIAM)
- Gonzalez, D., & Hayajneh, T. 2017, in 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), Vol. 2018-January
- Granitto, P. M., Furlanello, C., Biasioli, F., & Gasperi, F. 2006, Chemometrics and intelligent laboratory systems, 83, 83
- Han, H., Ke, Z., Nie, X., Dai, L., & Slamun, W. 2023, Applied Sciences, 13, 4886
- Han, K. et al. 2022, IEEE transactions on pattern analysis and machine intelligence, 45, 87
- Hashim, H. A., Salleh, Z., Shuhaimi, I., & Ismail, N. A. N. 2020, Journal of Financial Crime, 27, 1143
- Havrlant, L., & Kreinovich, V. 2017, International Journal of General Systems, 46, 27
- He, J., Zhao, L., Yang, H., Zhang, M., & Li, W. 2019, IEEE Transactions on Geoscience and Remote Sensing, 58, 165
- Hernandez Aros, L., Bustamante Molano, L. X., Gutierrez-Portela, F., Moreno Hernandez, J. J., & Rodríguez Barrero, M. S. 2024, Humanities and Social Sciences Communications, 11, 1
- Howley, T., Madden, M. G., O'Connell, M.-L., & Ryder, A. G. 2005, in International Confer-

- ence on Innovative Techniques and Applications of Artificial Intelligence, Springer, 209–222
- Huang, S., Cai, N., Pacheco, P. P., Narrandes, S., Wang, Y., & Xu, W. 2018, *Cancer Genomics Proteomics*, 15, 41
- Huang, S. H. 2015, *Artif. Intell. Res.*, 4, 22
- Ilyas, I. F., Markl, V., Haas, P., Brown, P., & Abounaga, A. 2004, in *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, 647–658
- Isson, J. P. 2018, *Unstructured data analytics: how to improve customer acquisition, customer retention, and fraud detection and prevention* (John Wiley & Sons)
- Jarynowski, A., Semenov, A., Kamiński, M., & Belik, V. 2021, *medRxiv*, 2021
- Jiang, N., Tian, F., Li, J., Yuan, X., & Zheng, J. 2020, *Applied Intelligence*, 50, 2301
- Jogin, M., Madhulika, M., Divya, G., Meghana, R., Apoorva, S., et al. 2018, in *2018 3rd IEEE international conference on recent trends in electronics, information & communication technology (RTEICT)*, IEEE, 2319–2323
- Kadhim, A. I., Cheah, Y.-N., & Ahamed, N. H. 2014, in *2014 4th international conference on artificial intelligence with applications in engineering and technology*, IEEE, 69–73
- Karathanassi, V., Kolokousis, P., & Ioannidou, S. 2007, *International Journal of Remote Sensing*, 28, 2309
- Khalid, S., Khalil, T., & Nasreen, S. 2014, in *2014 science and information conference*, IEEE, 372–378
- Khattar, D., Goud, J. S., Gupta, M., & Varma, V. 2019, in *The world wide web conference*, 2915–2921

- Khoshgoftaar, T. M., Gao, K., Napolitano, A., & Wald, R. 2014, *Information Systems Frontiers*, 16, 801
- Kira, K., & Rendell, L. A. 1992, in *Machine learning proceedings 1992* (Elsevier), 249–256
- Kumar, A., Meel, M., et al. 2023, in *Proc. of the Digital Forensics Workshop*
- Kumar, V., & Minz, S. 2014, *SmartCR*, 4, 211
- La Morgia, M., Mei, A., Mongardini, A. M., & Wu, J. 2021, arXiv preprint arXiv:2111.13530
- Lee, J., & Kim, D.-W. 2015, *Expert Systems with Applications*, 42, 2013
- Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. 2017, *ACM computing surveys (CSUR)*, 50, 1
- Lilleberg, J., Zhu, Y., & Zhang, Y. 2015, in *2015 IEEE 14th International Conference on Cognitive Informatics Cognitive Computing (ICCI\*CC) (IEEE)*, 136–140
- Linardatos, P., Papastefanopoulos, V., & Kotsiantis, S. 2020, *Entropy*, 23, 18
- Liu, Q., Chen, C., Zhang, Y., & Hu, Z. 2011, *Artificial Intelligence Review*, 36, 99
- Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., & Guo, B. 2021, in *Proceedings of the IEEE/CVF international conference on computer vision*, 10012–10022
- Lyu, Y., Feng, Y., & Sakurai, K. 2023, *Information*, 14, 191
- Mackey, T. K., Li, J., Purushothaman, V., Nali, M., Shah, N., Bardier, C., Cai, M., & Liang, B. 2020, *JMIR Public Health and Surveillance*, 6, e20794
- Mahesh, T., Kumar, V. D., Kumar, V. V., Asghar, J., Geman, O., Arulkumaran, G., & Arun, N. 2022, *Computational Intelligence and Neuroscience*
- Masum, M., Faruk, M. J. H., Shahriar, H., Qian, K., Lo, D., & Adnan, M. I. 2022a, in *2022*

- IEEE Conference on Emerging Wireless Technologies and Enterprise, 03–16–03–22
- Masum, M., Faruk, M. J. H., Shahriar, H., Qian, K., Lo, D., & Adnan, M. I. 2022b, in 2022 IEEE 12th annual computing and communication workshop and conference (CCWC), IEEE, 0316–0322
- Mohla, S., Pande, S., Banerjee, B., & Chaudhuri, S. 2020, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 92–93
- Moore, K. L., Bihl, T. J., Bauer Jr, K. W., & Dube, T. E. 2017, *The Journal of Defense Modeling and Simulation*, 14, 217
- Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A. Y., et al. 2011, in *ICML*, Vol. 11, 689–696
- Noreen, N., Palaniappan, S., Qayyum, A., Ahmad, I., Imran, M., & Shoaib, M. 2020, *IEEE access*, 8, 55135
- Papasavva, A., Johnson, S., Lowther, E., Lundrigan, S., Mariconti, E., Markovska, A., & Tuptuk, N. 2024, arXiv preprint arXiv:2409.19022
- Patle, A., & Chouhan, D. S. 2013, in 2013 International Conference on Advances in Technology and Engineering (ICATE), 1–9
- Prajapati, G. L., & Patle, A. 2010, in 2010 3rd International Conference on Emerging Trends in Engineering and Technology, 512–515
- Prieto, A., Prieto, B., Ortigosa, E. M., Ros, E., Pelayo, F., Ortega, J., & Rojas, I. 2016, *Neurocomputing*, 214, 242
- Rajbahadur, G. K., Wang, S., Oliva, G. A., Kamei, Y., & Hassan, A. E. 2021, *IEEE Trans-*

- actions on Software Engineering, 48, 2245
- Ratna, S., Saide, S., Putri, A. M., Soleha, A., & Andini, P. R. 2024, *Procedia Computer Science*, 234, 1538, seventh Information Systems International Conference (ISICO 2023)
- Ratner, A., Bach, S., Ehrenberg, H., Fries, J., Wu, S., & Ré, C. 2017, in *Hazy Research Technical Report*
- Ratner, A., Bach, S. H., Ehrenberg, H., Fries, J., Wu, S., & Ré, C. 2020, *The VLDB Journal*, 29, 709
- Rawat, R., Oki, O., Chakrawarti, R. K., Adekunle, T. S., Lukose, J. M., & Ajagbe, S. A. 2023, *Informatica*, 47
- Resende, P. A. A., & Drummond, A. C. 2018, *ACM Computing Surveys (CSUR)*, 51, 1
- Reurink, A. 2019, *Contemporary topics in finance: A collection of literature surveys*, 79
- Roy, S. S., Vafa, E. P., Khanmohamaddi, K., & Nilizadeh, S. 2025, in *34th USENIX Security Symposium (USENIX Security 25)*, 4839–4858
- Ryan, M. 2021, *Ransomware Revolution: the rise of a prodigious cyber threat*, Vol. 85 (Springer)
- Salahuddin, Z., Woodruff, H. C., Chatterjee, A., & Lambin, P. 2022, *Computers in biology and medicine*, 140, 105111
- Salem, N., & Hussein, S. 2019, *Procedia Computer Science*, 163, 292
- Salman, H. A., Kalakech, A., & Steiti, A. 2024, *Babylonian Journal of Machine Learning*, 2024, 69
- Saputra, D. M., Saputra, D., & Oswari, L. D. 2020, in *Sriwijaya international conference on*

- information technology and its applications (SICONIAN 2019), Atlantis Press, 341–346
- Saribekyan, H., & Margvelashvili, A. 2017, Diakses tanggal, 15
- Schapiro, R. E. 2013, in *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik* (Springer), 37–52
- Sengupta, S., Basak, S., Saikia, P., Paul, S., Tsalavoutis, V., Atiah, F., Ravi, V., & Peters, A. 2020, *Knowledge-Based Systems*, 194, 105596
- Shah, D., Harrison, T., Freas, C. B., Maimon, D., & Harrison, R. W. 2020, in *2020 IEEE international conference on big data (Big Data)*, IEEE, 4341–4350
- Shehabat, A., Mitew, T., & Alzoubi, Y. 2017, *Journal of strategic security*, 10, 27
- Shi, C., Wei, B., Wei, S., Wang, W., Liu, H., & Liu, J. 2021, *EURASIP journal on wireless communications and networking*, 2021, 1
- Singhal, S., Shah, R. R., Chakraborty, T., Kumaraguru, P., & Satoh, S. 2019, in *2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM)*, IEEE, 39–47
- Soudijn, M. R., & Zegers, B. C. T. 2012, *Trends in organized crime*, 15, 111
- Steinwart, I., & Christmann, A. 2008, *Support Vector Machines* (Springer Science Business Media)
- Suarez-Tangil, G., Tapiador, J. E., Peris-Lopez, P., & Ribagorda, A. 2013, *IEEE communications surveys & tutorials*, 16, 961
- Suh, J. B., Nicolaidis, R., & Trafford, R. 2019, *International Journal of Law, Crime and Justice*, 56, 79
- Sutikno, T., Handayani, L., Stiawan, D., Riyadi, M. A., & Subroto, I. M. I. 2016, *Whatsapp*,

- Viber, and Telegram: which is the best for instant messaging?
- Sutter, J. M., & Kalivas, J. H. 1993, *Microchemical journal*, 47, 60
- Tavabi, N., Bartley, N., Abeliuk, A., Soni, S., Ferrara, E., & Lerman, K. 2019, in *Companion Proceedings of the 2019 World Wide Web Conference*, 206–213
- Theng, D., & Bhoyar, K. K. 2024, *Knowledge and Information Systems*, 66, 1575
- Thudumu, S., Branch, P., Jin, J., & Singh, J. 2020, *Journal of big data*, 7, 1
- Udeh, E. O., Amajuoyi, P., Adeusi, K. B., & Scott, A. O. 2024, *World Journal of Advanced Research and Reviews*, 22, 1746
- Urooj, U., Al-Rimy, B. A. S., Zainal, A., Ghaleb, F. A., & Rassam, M. A. 2021, *Applied Sciences*, 12, 172
- Verma, R., & Pandiya, D. K. 2024, *International Journal of Global Innovations and Solutions (IJGIS)*
- Wang, D.-Q., Feng, L.-Y., Ye, J.-G., Zou, J.-G., & Zheng, Y.-F. 2023, *MedComm–Future Medicine*, 2, e43
- West, J., & Bhattacharya, M. 2016, *Computers & security*, 57, 47
- Wikipedia contributors. 2023, Arrest and indictment of Pavel Durov, [https://en.wikipedia.org/wiki/Arrest\\_and\\_indictment\\_of\\_Pavel\\_Durov](https://en.wikipedia.org/wiki/Arrest_and_indictment_of_Pavel_Durov), [Accessed: Jan-2025]
- Wronka, C. 2023, *Journal of Financial Crime*, 30, 97
- Wu, S., Hsiao, H., Wang, F., Li, C., & Ré, C. 2018, in *Proceedings of the 2018 International Conference on Management of Data (SIGMOD)*
- Yang, G. 2019, *Advances in Neural Information Processing Systems*, 32–34

Yin, W., Kann, K., Yu, M., & Schütze, H. 2017, arXiv preprint arXiv:1702.01923

Zhang, L., Zhu, J., & Yao, T. 2004, ACM Transactions on Asian Language Information Processing (TALIP), 3, 243

Zhang, X., Fan, M., Wang, D., Zhou, P., & Tao, D. 2020, IEEE Transactions on Neural Networks and Learning Systems, 32, 3005

Zhou, Y., Cheng, G., Jiang, S., & Dai, M. 2020, Computer networks, 174, 107247

Zhu, X., Ao, X., Qin, Z., Chang, Y., Liu, Y., He, Q., & Li, J. 2021, The Innovation, 2