The Journal of Finance and Data Science 10 (2024) 100147

Contents lists available at ScienceDirect

The Journal of Finance and Data Science

journal homepage: www.keaipublishing.com/en/journals/jfds

Research Article

Detecting Hawala network for money laundering by graph mining[☆]

Marzhan Alenova, Assem Utaliyeva, Ki-Joune Li[*]

Pusan National University, Department of Computer Science and Engineering, Busan, 46241, Busan, South Korea

ARTICLE INFO

Keywords:
Hawala network
Money laundering
Graph mining
Anomaly detection

ABSTRACT

Hawala, a traditional but informal money transfer system, has been prevalent in many parts of the world, such as money laundering. Despite the regulatory actions taken by financial institutions, Hawala is still a key node in terror financing schemes and its extent of misuse is unknown. Due to the hidden transactions and limited knowledge about the Hawala, it is difficult for legal enforcement authorities such as financial intelligence units (FIU) of each country to detect and investigate the Hawala network. In this paper, we present a novel approach to detect the potential Hawala instances in the stream of financial transaction data by using graph mining techniques. In order to reflect the properties of Hawala, we apply graph mining methods such as graph centrality, Blackhole metric, and Hidden link metric as well as anomaly detection methods using graph convolutional network. Experiments demonstrate that the proposed method gives a meaningful result in detecting Hawala network and can be used as a complementary tool to the existing transactional monitoring tracks.

1. Introduction

Terrorist financing (TF) and money laundering (ML) pose significant threats to global safety, economies, and financial systems. To avoid legal regulations and investigations, criminals have used various typologies in their financing schemes, and the movement of money is a critical intermediary step in all of them. One of the ways to move funds from one country to another is known as Hawala (Freeman and Ruehsen, 2013). It is a traditional but informal banking system that allows money transfers outside of formal financial institutions, and people offering Hawala services are called Hawaladars. There is no fixed Hawala typology, but in most cases, Hawala does not involve the legal international transfer or physical movement of money, making it an attractive point for criminals (FATF, 2013).

Hawala has been a topic of increasing interest since the 9/11 terrorist attack. It was discovered that Hawala played a substantial role in the financial network of the Al-Qaeda organization (del Cid Gomez, 2010; Sakharova, 2011). This led to the fact that nowadays, Hawala is viewed as one of the leading channels for TF and ML, and it is illegal in many countries (FATF, 2013). The actual extent of its misuse is still unknown, but it is estimated to range in the billions of USD annually (Chêne, 2008). Common definitions of Hawala refer to an alternative money transfer system of sending funds without the physical movement of money and a paper trail, making it an attractive point for criminals (FATF, 2013). The extent of its misuse is still unknown, but it is estimated to range in the billions of USD annually, thus posing a serious threat to global socioeconomic well-being (Chêne, 2008).

[☆] This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2023-00242528).

[*] Corresponding author.

E-mail addresses: alenova@pusan.ac.kr (M. Alenova), assemakx@pusan.ac.kr (A. Utaliyeva), lik@pnu.edu (K.-J. Li).

Peer review under the responsibility of KeAi Communications Co., Ltd.

https://doi.org/10.1016/j.jfds.2024.100147
Received 4 November 2023; Received in revised form 3 October 2024; Accepted 11 November 2024
Available online 16 November 2024

Despite its informal nature, Hawala may involve the use of formal and visible banking services within a single country, depending on the circumstances and availability of banking infrastructure (FATF, 2013). Such domestic transfers may be the only ”visible” and formal part of the Hawala transaction, while the rest remains anonymous and invisible. In addition to anonymity, the speed of Hawala transactions greatly contributes to its widespread use in illicit activities. For example, it has been directly linked to financing terrorist attacks in the United States, Kenya, and Tanzania, as well as laundering the proceeds from drug trafficking in Europe (Wheatley, 2005). Thus, Hawala poses a serious threat to global socioeconomic well-being.

Due to the aforementioned negative effects, several countries and international organizations have introduced regulatory measures against Hawala (Passas, 2005). These regulations primarily focus on monitoring transactions beyond a certain threshold, imposing sanctions, and increasing awareness. However, there is no explicit detection system for Hawala, and financial institutions mostly rely on general detection systems that raise alerts based on fixed rules. This approach generates a high proportion of false positives (over 90 %), which is very resource-intensive (Maxwell et al., 2020). Furthermore, the recent surge in the number of transactions, owing to advancements in the banking sector, has complicated the detection of suspicious transactions for financial institutions.

In recent years, innovative techniques, such as artificial intelligence and machine learning, have been adopted by some banks to improve the operational efficiency of traditional methods (Watkins et al., 2010; Oeben et al., 2019). However, regulating the detection process of the Hawala network has seen little progress due to several challenges. A significant barrier is the absence of publicly available bank transaction data. Additionally, the lack of knowledge about Hawala and its constantly evolving nature poses another obstacle to its effective detection.

In this paper, we provide alternative solutions for the above-mentioned Hawala challenges and propose a framework for detecting suspicious bank accounts. We aim to uncover the ”invisible” part of Hawala only relying on its ”visible” part. Our framework is based on a graph representation of bank transaction data and is composed of two distinct modules: *Hawala-based Ranking* (HR) and *Anomaly-based Ranking* (AR). The HR module is designed to calculate the ranking of bank accounts based on their similarity to known Hawala characteristics, while the AR module determines the ranking of bank accounts based on their anomaly scores obtained through graph anomaly detection.

The proposed framework is not seen as a replacement for the existing rule-based system but as a complementary tool. Detection results can be used as a reference by banking experts, thus assisting their operational efficiency.

After the introduction, this paper is structured as follows. Section 2 presents the background information related to Hawala and discusses the main challenges in its detection. Section 3 presents common Hawala characteristics. Section 4 describes some publicly available financial datasets, Section 4.1 presents a real-world Berka dataset provided by a Czech bank, and Section 4.2 outlines the performed data augmentation steps on that dataset. Section 5 defines the problem of this paper and presents the rank-driven framework with two modules, HR and AR. Section 6 introduces the details of the HR module, while Section 7 describes the AR module. Section 8 describes the results of the proposed framework and finally, Section 10 presents the conclusion.

2. Background

In this section, we present the studies on *Graph Neural Networks* (GNNs), which is a key approach of our work, and introduce the concept of Hawala, describe how it operates, and highlight the main challenges in its detection.

2.1. Graph Neural Networks methods in financial applications

Financial systems, characterized by their complexity, comprise numerous elements and interconnected structures that undergo frequent updates. Graphs are commonly used to represent relational data in the financial domain, including transaction networks, user-item review graphs, and stock relation graphs. Graph Neural Networks (GNNs) have gained popularity for various financial tasks, including attributed graph anomaly detection. DOMINANT (Ding et al., 2019) is an example of a method that uses GNNs to compute anomaly ranking scores by employing a deep GCN-based auto-encoder.

In particular, GNNs have proven effective in detecting fraud through anomaly detection. Examples include Semi GNN Wang et al., (2020) for detecting fraudsters on Alipay; GraphConsis (Liu et al., 2020) and CARE-GNN (Dou et al., 2020) for filtering dissimilar neighbors to identify camouflage fraudsters; and Geniepath (Liu et al., 2018) for designing a novel aggregate method of GNNs to filter graph signals from neighbors of different hops away for detecting financial fraud. However, the application of GNNs is not only limited to fraud detection. It also extends to Anti-money laundering (AML) purposes in Bitcoin (Weber et al., 2019). More information about GNNs' application in financial sector is presented in review paper (Wang et al., 2021).

2.2. Hawala

Hawala is an ancient informal banking system facilitating international fund transfers outside of formal financial institutions. People offering Hawala services are called Hawaladars (Hawala dealers), and it was originally developed for areas lacking or unsafe conventional banking systems (Teichmann and Wittmann, 2022). There is no recorded agreement between Hawaladars and their clients since they often belong to the same ethnic communities, and they may be relatives or acquaintances (Soudjin, 2014). Hawala has often been in demand among migrant workers seeking a swift and affordable means of transferring a portion of their incomes to their home countries. However, in the aftermath of 9/11, it was discovered that the scope of Hawala far exceeded the religio–cultural association.

Fig. 1 shows the illustration of a basic Hawala operation. We suppose that person P in country X wants to send money to a receiver Q in country Y.

1. P contacts a Hawala dealer A in country X and gives him money to send to a receiver Q in country Y.
2. In return, Hawala dealer A gives a password to P.
3. Hawala dealer A contacts her/his partner, Hawala B in country Y and instructs her/him to give the money to the receiver Q. At the same time, P passes the password to the receiver Q.
4. Q gets the money from Hawala dealer B by providing the password to her/him.

Although Hawala transactions can be simply illustrated as Fig. 1, they may easily adapt to different circumstances, and the operation may involve several complications (McCusker, 2005).

2.3. *Challenges in Hawala detection and motivations*

New technologies can enhance the effectiveness of monitoring systems for financial institutions and improve the detection of suspicious activities (FATF, 2021). However, conducting any kind of research related to the financial sector is not a straightforward task, as it faces a variety of challenges.

In this research, the first challenge is the lack of unified knowledge about Hawala typology and the concept drift. Hawala's adaptability and lack of a paper trail hinder law enforcement agencies' efforts to investigate it. This results in difficulties in developing robust Hawala characteristics for its detection and regulation. To tackle this issue, we conducted a thorough analysis of real Hawala cases published by FATF (FATF, 2013) and highlighted its most common characteristics. These highlighted Hawala characteristics are summarized in Section 3.

The second challenge is the absence of publicly available bank transaction data (labeled data). Effective research requires access to large-scale datasets. However, due to confidentiality concerns, financial institutions are hesitant to release these datasets to the research community. This significantly impedes the progress in designing and implementing new systems, particularly AI-based solutions. In this landscape, where financial datasets are so scarce, we have an alternative solution involving a real financial dataset from Czech Bank, presented in Section 4.1.

Table 1 summarizes the key challenges encountered in conducting this research, along with the proposed solutions that were implemented to address them. We discuss the issues listed in Table 1 in detail in the subsequent sections, which serve as the starting points of our work.

3. Common Hawala characteristics

In this section, we explore prevalent understanding of Hawala, which forms the foundational concepts of the proposed framework. Despite the existence of several indicators for identifying Hawala transactions, distinguishing between legal and illegal money transfers remains a complex task due to the sophisticated methods criminals employ to evade detection Chêne (2008). This complexity poses a significant hurdle in establishing an efficient Hawala detection system. To navigate this challenge, we conducted an in-depth analysis of publicly available reports and identified the following distinctive characteristics of Hawala.

3.1. *Blackhole pattern*

Hawaladars commonly resort to smurfing, a technique involving the structuring of funds (Bowers 2009). Structuring is a tactic used by criminals to transfer amounts less than a certain threshold to avoid the possibility of detection. Hawaladars often employ structured deposits or wire transfers under the following circumstances (FATF, 2013).

- Hawaladar mostly receives money from the client in the same country.
- Hawala partners do the net settlement with each other.

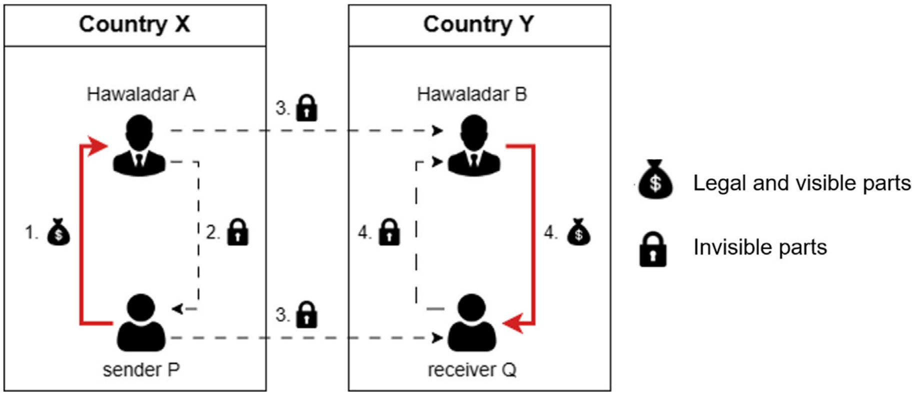

Fig. 1. Example of Hawala transaction.

Table 1
Challenges and proposed solutions.

Challenges	Solution
Lack of unified knowledge on Hawala	Analysis of common Hawala characteristics such as *blackhole patterns* and *hidden links between non-related bank accounts*
Absence of real-world dataset	Data augmentation and utilization of public datasets such as Berka dataset (Czech financial dataset) (Adams et al., 1999)

Through our analysis of Hawala cases provided by the FATF (FATF, 2013), we discovered that smurfing may result in a pattern known as *blackhole*, which is a well-known pattern in fraud detection (Li et al., 2010). This pattern arises when a cluster of bank accounts, exclusively connected by inbound transactions from external accounts, creates a financial sink. To formulate blackhole pattern, we first define *transaction graph* as below.

Definition 1. (Transaction Graph). *transaction graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ *is a directed graph extracted from the transaction list; where the set of nodes* $\mathcal{V}$ *are bank accounts; the set of edges* $\mathcal{E}$ *represent bank operations between two bank accounts; each edge* $e(v_i, v_j) \in \mathcal{E}$ *between nodes* $v_i, v_j \in \mathcal{V}$ *has associated weight,* $w_e(=1)$.

Note that the weight of edge is set to 1 as we do not consider the monetary amount of transaction. If there are several transactions between two accounts, each of them is considered as a separate edge. The definition of blackhole pattern is given in Li et al. (2010) as follows;

Definition 2. (Blackhole pattern). *A set of nodes B,* $B \subseteq V$ *form a blackhole pattern, if and only if the following two conditions are satisfied,*

1. $\|B\| \geq 2$, *and the subgraph* $\mathcal{G}(B)$ *induced by B is a weakly connected, and*
2. $(in(B) - out(B))/\|B\| > \theta$, *where* θ *is a predefined positive threshold.*

Indegree and outdegree of the subgraph, $in(B)$ and $out(B)$, refer to the number of edges entering and leaving the subgraph, and a graph is defined as weakly connected if there is a path between every two vertices in the underlying undirected graph. Fig. 2 shows an example of the blackhole pattern, where the blackhole pattern consists of three bank accounts as a subgraph B highlighted by a dashed circle. Assuming $\theta = 2.0$, we see that $in(B) = 7$ and $out(B) = 0$, and $(in(B) - out(B))/\|B\| = 7/3 > \theta$. Subgraph B is therefore a blackhole as it satisfies definition 1. However, if we include node d to B', $in(B') = 6$ and $out(B') = 1$ and $(in(B') - out(B'))/\|B'\| = 5/4 < \theta$. Subgraph B' is not a blackhole pattern.

Hawala transactions frequently exhibit blackhole patterns. In an example typology of Hawala transactions, cash deposits or wire transfers are split into smaller amounts to avoid attention from banks, as deposits or transfers above a certain threshold (for example USD 10,000) may be reported as an STR (Suspicious Transaction Report) (FATF, 2013). However, this practice can lead to a situation where Hawaladars accumulate a significant volume of minor transactions, resulting in a blackhole pattern. Two case studies are presented below as references to show how Hawala results in blackhole patterns.

- **Case 1**: According to a case presented in (FATF, 2013), a toy company in Los Angeles called Angel Toy Company engaged in structuring currency transactions while producing stuffed toys. Over the course of four years, investigators followed over $8 million in cash deposits into the company's account, none of which were for more than $10,000. The money was later returned to drug traffickers once the stuffed toys were exported and sold in foreign countries to make ”clean” money.
- **Case 2**: An individual from Africa who lived in a European country admitted to conducting Hawala transactions, according to a report (FATF, 2013). The person's account received only cash deposits and small inward transfers. After a few months, the funds were transferred in bulk to Company A in Africa.

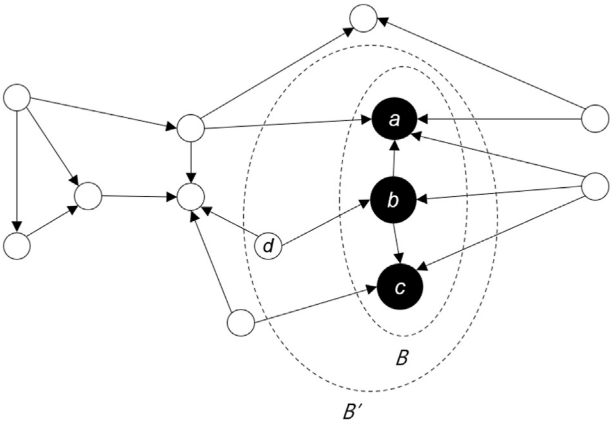

Fig. 2. An illustration of blackhole pattern.

3.2. Hidden link between bank accounts

In most cases, Hawaladars use their relatives or community acquaintances to minimize detection risk in their money transfer business (FATF, 2013). They frequently orchestrate control over personal bank accounts opened by their relatives or open accounts under non-operating businesses. For example in Fig. 2, three bank accounts **a**, **b**, and **c** may, in reality, be managed by one Hawaladar, creating a hidden link among them.

Financial institutions encounter significant challenges when attempting to uncover these hidden links between bank accounts, which can benefit Hawaladars in running their money transfer business on one hand. On the other hand, the use of collective bank accounts also introduces a potential vulnerability. Bank accounts managed by one Hawaladar may show incidents of information reuse, such as identical telephone numbers, home addresses, or IP addresses. Fig. 3 provides an example of information reuse between two unrelated bank accounts, where they have the same IP address. Thus, it may suggest a potential link between them.

FATF Report (FATF, 2020) on Virtual Assets presented case studies indicating that IP address could be a strong indicator of illicit transactions involving ML and TF. This report elaborates on how the IP addresses associated with bank accounts can reveal various transaction patterns indicative of illegal activities. In this research, we prioritize IP address as a main attribute to uncover hidden links between bank accounts. The following two cases illustrate how Hawaladars can manipulate several bank accounts to run their money transfer business.

- **Case 3**: According to a case study provided by AUSTRAC, one remittance dealer showed a substantial increase in activity, including accepting large cash deposits and facilitating large international funds transfers (FATF, 2013). This was inconsistent with the remitter's previous profile, and further analysis revealed significant discrepancies between the information reported by the remitter and financial institutions to AUSTRAC. Shortly after the spike in transaction activity, bank account held by the remittance business stopped receiving deposits. However, AUSTRAC analysts identified additional bank accounts operated by the remittance business, which had been opened under a new company name. The remitter's transaction activity continued to escalate while operating under the new bank accounts.
- **Case 4**: The Hawala service provider and his nephew were found to be operating a money-transfer operation at an Ice Cream shop in Brooklyn, New York (FATF, 2013), according to a case study provided by FATF. The shop held one main bank account, along with 12 different ”feeder” accounts at various banks that received money through different means like deposits and wire transfers. The funds were then transferred to the main account. One of the feeder accounts was opened under a non-operating business name that used the nephew's home address and telephone numbers.

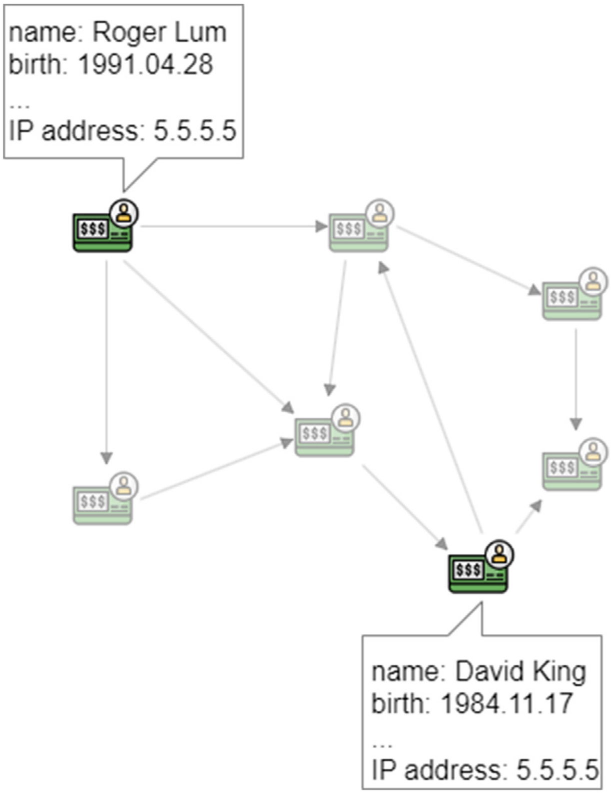

Fig. 3. An illustration of the correlation in information.

4. Dataset

While bank transaction data are crucial in analyzing Hawala and eventually developing solutions to detect Hawala networks, financial data are highly confidential, and it must only be disclosed to authorized individuals. There are fortunately some alternatives such as datasets from Kaggle, which are open to public. These include IEEE-CIS Fraud Detection dataset (Vesta Coorporation, 2019), Credit Card Fraud dataset (Machine Learning Group, 2018), and Bitcoin Blockchain Historical dataset (Day et al., 2019). Each of them has its own advantages as well as disadvantages.

IEEE-CIS Fraud Detection dataset contains about 500K e-commerce transactions, but it doesn't capture transaction interaction between users. The Credit Card Fraud dataset includes over 280K transactions made by credit cardholders. It is highly imbalanced, and most of the dataset features are turned into numerical values obtained via Principal Component Analysis. It also lacks of transaction interaction between users. The Bitcoin Blockchain Historical dataset contains data about blockchain blocks and transactions. Money laundering by Bitcoin or similar cryptocurrency become an important issue and need more attention. However, we will leave it as a future work since very little is known about Hawala network by Bitcoin. When sufficient cases of Hawala by Bitcoin will be collected, we expect that a similar solution one could be developed based on the proposed solution.

4.1. Berka dataset

In a landscape where real-world financial datasets are scarce due to privacy issues, we take a practical approach to this problem by data augmentation using a real dataset, called *Berka* dataset (Adams et al., 1999). Berka dataset is a collection of real anonymized financial information from a Czech Bank, used for the PKDD'99 Discovery Challenge. Originally, the dataset was disclosed to address the problem of loan prediction and improve bank services. It spans a 5-year period from 1993 to 1998 and includes records of various bank operations, such as transfer, deposit, and withdrawal. The original dataset contains 8 raw files including records about bank accounts, clients, transactions, loans, and credit cards.

Several steps of preprocessing were undertaken to make the Berka dataset more up-to-date. In addition, we narrowed the dataset's extensive time frame, focusing on bank operations conducted over two years, specifically from 1993 to 01-01 to 1994-12-31. The following list summarizes the preprocessing steps applied to the original Berka dataset.

- Converting local currency to USD.
- Removing duplicated transactions.
- Incrementing date-related features to 20 years.
- Extracting the gender and birth date of clients from encrypted value.
- Generating the first and last names based on the gender of clients.

Given its publication date, the Berka dataset may not reflect the current state of the banking sector, which has undergone substantial evolution. Consequently, it does not encompass certain features discussed in Sections 3.1 and 3.2, such as IP addresses and the blackhole pattern. To mitigate this limitation, we employed data augmentation technique to enrich the original Berka dataset, thereby enhancing its relevance and applicability for the objectives of this paper.

4.2. Data augmentation on Berka dataset

Fig. 4 outlines the main data augmentation procedures applied to the Berka dataset. To incorporate blackhole patterns into the dataset, we utilized the AMLSim simulator (Suzumura and Kanezashi, 2021) to generate synthetic transaction samples, which were subsequently manually injected into designated bank accounts. These accounts constitute 2.2 % of the overall bank accounts, and the inserted synthetic transactions represent 2.1 % of the total transaction volume. We categorized the chosen accounts into clusters varying in size from 2 to 10, each forming a distinct blackhole pattern. Henceforth, these accounts and their associated transactions are nominally deemed suspicious. The selection of 2.2 % and 2.1 % ratios is predicated on the presumed proportion of illicit bank operations, as indicated in the Europol Spotlight Report (Europol, 2022).

To establish hidden links between bank accounts, we assigned a synthesized IP address to each account within the Berka dataset. Subsequently, we altered half of the injected blackhole patterns, assigning identical IP addresses to certain accounts within each group, thereby mimicking hidden links. To maintain dataset diversity and reflect real-world complexity, we also randomly selected several benign accounts, applying similar modifications. Consequently, the augmented Berka dataset not only encompasses blackhole patterns but also features interlinked bank accounts, creating an intricate network of hidden associations through shared IP addresses.

A subset of bank accounts, specifically those within the blackhole patterns, were identified and nominally labeled as suspicious. These labels are not used in the primary analysis but are set aside for subsequent evaluation of the proposed framework. Here are some important features of the Berka dataset following the preprocessing steps.

- number of bank accounts: 4500
- number of transactions: 202216
- mean transaction value: 1574.57
- median transaction value: 595.02

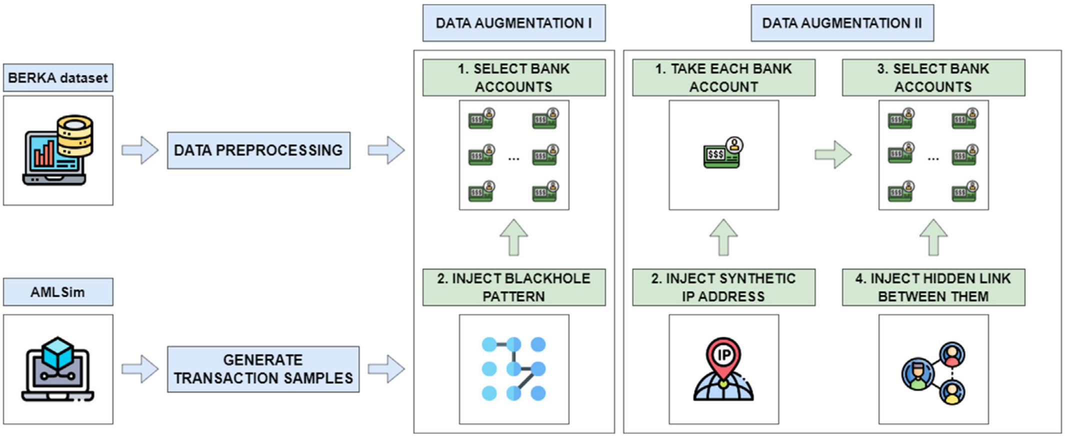
DATA AUGMENTATION I
DATA AUGMENTATION II
BERKA dataset
DATA PREPROCESSING
1. SELECT BANK ACCOUNTS
2. INJECT BLACKHOLE PATTERN
1. TAKE EACH BANK ACCOUNT
2. INJECT SYNTHETIC IP ADDRESS
3. SELECT BANK ACCOUNTS
4. INJECT HIDDEN LINK BETWEEN THEM
AMLSim
GENERATE TRANSACTION SAMPLES
IP

Fig. 4. Data augmentation steps on Berka dataset.

Table 2
Account features.

Type	Feature	Description
Real	account_id	Account ID of client
	birth_date	Birth date of client
	Gender	Gender of client
	initial_balance	Initial balance of client
	client_addr	Address of client
	create_date	Creation date of account
	bank_id	Bank ID
	branch_addr	Address of the bank branch
Synthetic	first_name	First name of client
	last_name	Last name of client
	ip_address	IP address of client

- max transaction value: 8739.73
- min transaction value: 0.02
- number of " suspicious" bank accounts: 103
- number of "suspicious" transaction: 4260
- number of blackhole pattern: 51

We leveraged the augmented Berka dataset, now inclusive of blackhole patterns and hidden links, as the main data for our framework.[1]

All detail features of the dataset and their descriptions are summarized in Tables 2 and 3.

5. Problem setting and framework of proposed solutions

In this section, we define the main problem of the paper and propose the rank-driven framework as a solution.

5.1. Problem setting

The input data is the bank transaction dataset. Specifically, we focus on a graph representation of the bank transactions as defined by 2 in section 3.1. Therefore, the problem of this paper is to detect suspicious bank accounts from a given transaction graph, which represents a bank transaction dataset. Table 4 summarizes the symbolic notations often used in the rest of the paper.

5.2. Proposed framework

To address the problem defined in the previous subsection, we propose a rank-driven framework consisting of two independent rank modules, *Hawala-based Rank* (HR) and *Anomaly-based Rank* (AR), as shown in Fig. 5.

The HR module computes the rank of nodes based on the degree to which a certain node shows the traits of Hawala. Specifically, a node is considered to show the traits of Hawala either if it is involved in either a blackhole pattern, or it has a hidden link to

[1] The augmented dataset is available at https://github.com/STEMLab/Hawala.

Table 3
Transaction features.

Type	Feature	Description
Real	tx_id	Transaction ID
	tx_type	Transaction type
	tx_date	Transaction date
	tx_amount	Transaction amount
	sender_id	Sender ID
	receiver_id	Receiver ID
Synthetic	ip_address	IP address of client

Table 4
Symbolic notations.

Symbol	Description
$\mathcal{G}$	Graph
$\mathcal{V}$	Set of nodes in $\mathcal{G}$
$\mathcal{E}$	Set of edges in $\mathcal{G}$
N	Number of nodes in $\mathcal{G}$
$X \in \mathcal{R}^{N \times D}$	Matrix of node attributes in $\mathcal{G}$
$A \in \mathcal{R}^{N \times N}$	Adjacency matrix of $\mathcal{G}$
$Z \in \mathcal{R}^{N \times F}$	Matrix of node embeddings
PR	PageRank metric of nodes
BM	Blackhole metric of nodes
HLM	Hidden link metric of nodes
S	Anomaly score of nodes
c	Hypersphere center
r	Hypersphere radius

other nodes, which belong to blackhole patterns (See section 3). The HR score is calculated based on three sub-metrics: the Blackhole metric (BM), which is computed by finding blackhole patterns in the graph; the Hidden link metric (HLM), which is calculated by uncovering the hidden link between nodes; and the PageRank (PR), which is a well-known centrality measure. The AR module generates the rank of nodes using an unsupervised graph anomaly detection method. The resulting anomaly scores indicate the degree of non-benign behavior of each node, taking into account both the node's attribute information and the graph structure.

Two independent rank results can serve as a reference for banking experts to improve their operational efficiency. For example, nodes with high scores in both modules could be flagged for further verification. More detailed explanations of HR and AR modules are given in subsequent sections.

6. HR module

In this section, we present the basic ideas of the HR module, sub-metrics, and how they are combined to generate the final HR rank.

6.1. PageRank metric

PageRank, which is one of the most popular ranking metric, does not have a direct connection to Hawala, as opposed to Blackhole (Section 6.2) and Hidden link sub-metrics (Section 6.3), However, we incorporated PageRank as a sub-metric in the HR rank calculation. This is because focusing solely on Hawala-based rank may neglect the global importance of nodes, which reflects their significance based on the characteristics of the graph as a whole.

PageRank, Google's web page ranking system proposed by Brin and Page (1998), is a common ranking system used in graph analysis. It assigns scores to nodes based on the number and quality of links pointing to them. The score of a node is proportional to the sum of the scores of its incoming neighbors and is iteratively computed until it reaches to a convergence. PageRank score for each node $v \in \mathcal{V}$ is initialized with

$$PR_0(v) = \frac{1}{N} \tag{1}$$

where N is the number of nodes in the graph. The iterative process to compute the PageRank (PR) is defined by a simple sum as follows;

$$PR_{iter+1}(v_i) = \sum_{v_j \in Ne(v_i)} \frac{PR_{iter}(P_j)}{\|Ne(v_i)\|} \tag{2}$$

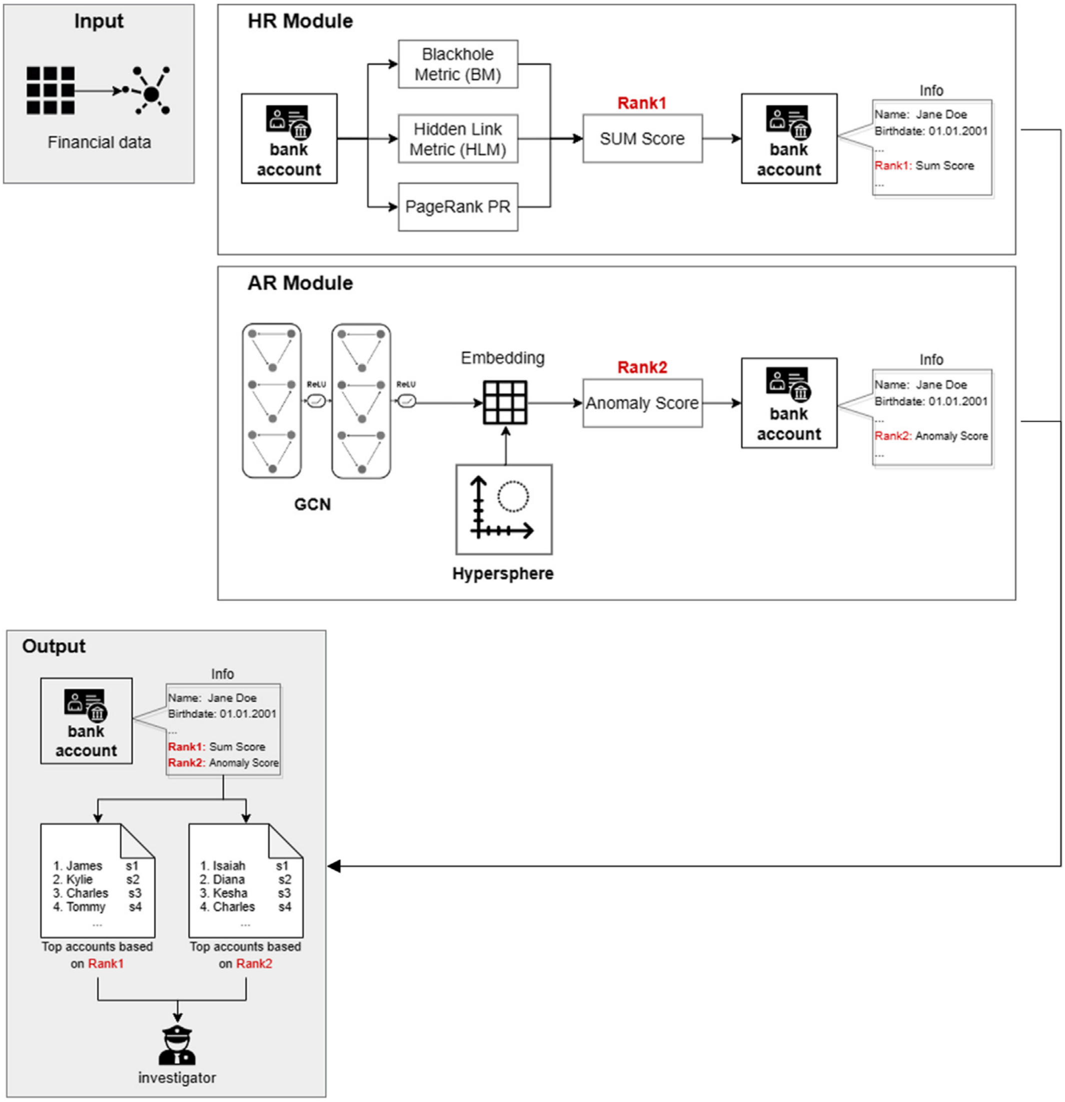

Fig. 5. Proposed framework.

where PR_{iter+1} is a PageRank of a graph node v_i at the $(iter + 1)$-th iteration, and $Ne(v_i)$ is the set of neighboring nodes of v_i. Its definition can be formulated as below.

Definition 3. (PageRank metric). *Given a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, PageRank metric of node v, $v \in \mathcal{V}$, is denoted as $PR(v)$ and computed such that,*

1. *Each $PR(v)$, $v \in \mathcal{V}$ is initialized using Equation 1, then*
2. *New $PR(v)$ is computed using Equation 2 by iterating way until the $PR(v)$ is basically stable.*

6.2. Blackhole metric

The Blackhole metric is based on the blackhole pattern explained in Section 6.2. Firstly, all the nodes that form the blackhole patterns in the graph are identified, and then, they are assigned a score according to the size of the blackhole pattern they belong to. We formulate the definition of Blackhole metric (BM) as follows (Li et al., 2010).

Definition 4. (Blackhole metric). *Given a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and set of blackholes, denoted as Blackholes, Blackhole metric of node v $(\in \mathcal{V})$, is denoted as $BM(v)$ and computed such that,*

1. *If B is a blackhole pattern, then $BM(v) = \|B\|$,*
2. *Else $BM(v) = 0$*

$\|B\|$ indicates the cardinality of B. The computation of BM involves the identification of all the blackhole patterns in the graph. The definition of blackhole pattern is given by definition 2 in section 3.1.

In real world scenarios, detecting all blackhole patterns is a very complex and expensive operation due to its computational cost as the number of combinations of the nodes exponentially increases with the increase of the number of nodes. Instead, we focus on a more practical version of this problem in this paper. Specifically, we simplify the definition of the blackhole pattern by changing the definition to $out(B) = 0$.

There are several well-established algorithms for detecting blackhole patterns, such as the simple Brute-Force algorithm, iBlackhole algorithm, and iBlackholeDC algorithm (Li et al., 2010). The Brute-force algorithm for detecting blackhole patterns involves checking all possible combinations of nodes in a graph. However, due to its computational complexity, this algorithm becomes impractical as the number of nodes in the graph increases.

The iBlackhole algorithm aims to reduce the computational complexity by applying pattern-size-independent pruning rules. It starts by establishing a potential list of nodes, *P*, based on initial pruning rules. Only nodes in this list are considered candidates for forming a blackhole pattern. Next, the algorithm iteratively examines the nodes in *P* and removes irrelevant nodes based on some pruning rules. The results of this pruning form an intermediate candidate list C. Then, list C is further refined using additional pruning rules to remove irrelevant nodes until it becomes a final search list *F*. Finally, the Brute-Force algorithm is applied to *F* to identify all blackhole patterns. By leveraging these pruning rules, the iBlackhole algorithm significantly reduces the search space and computational cost. Li et al. proposed a set of pruning rules for iBlackhole algorithm (Li et al., 2010).

Li et al. (2010) also improved iBlackhole to iBlackhole-DC algorithm, which also leverages the pruning rules, but on top of that, it divides the search space into several smaller ones by using a divide-and-conquer pruning strategy. In this paper, we leveraged the iBlackhole approach, and Algorithm 1 shows the overall procedure of computing the Blackhole metric of nodes.

Algorithm 1 Generation of Blackhole Metric using *iBlackhole* algorithm

```
Input: Transaction graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, maximum number of nodes in blackholes $n$
Output: Blackhole metric of nodes $BM$
$BM \leftarrow \{0|$for each $v \in \mathcal{V}\}$
$Blackholes \leftarrow iBlackhole(n)$    ▷ Find blackholes with sizes of 2 till $n$
for all $B \in Blackholes$ do
    for all $v \in B$ do
        $BM(v) \leftarrow |B|$
    end for
end for
return $BM$
```

6.3. Hidden link metric

Hidden link Metric is based on information reuse between bank accounts explained in Section 3.2. This metric selects an attribute and checks all nodes for information reuse. If a group of nodes shares an identical attribute, they are considered to have a hidden link. The challenge is to choose an attribute that is unique enough to show information reuse between bank accounts. Some attributes like name or date of birth are easy to fake, while others like IP address are more reliable.

In this paper, we leverage the IP address attribute of bank accounts. For example, two accounts that share an identical IP address, are considered to have hidden link regardless of whether they are directly connected or not. To find the information reuse incidents in the graph, we leverage the cosine similarity approach. Therefore, we formulate the definition of *Hidden link metric* as follows.

Definition 5. (Hidden link metric). *Given a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and set of IP addresses of nodes in binary vector form, denoted as bIPs, Hidden link metric of node v_i, $v_j \in \mathcal{V}$, is denoted as $HLM(v_i, v_j)$ and computed such that,*

1. *If $cos(bIPs(v_i), bIPs(v_j)) > 0.95$ for any $v_j \in \mathcal{V}$, then $HLM(v_i) = cos(bIPs(v_i), bIPs(v_j))$*
2. *Else $HLM(v_i) = 0$.*

Algorithm 2 shows the overall procedure of computing the Hidden link metric of nodes using the cosine similarity approach.

Algorithm 2 Generation of Hidden Link Metric Using Cosine Similarity

```
Input: Transaction graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, set of IP addresses of nodes $IP$
Output: Hidden link metric of nodes $HLM$
$HLM \leftarrow \{0|$for each $v \in \mathcal{V}\}$
$bIP \leftarrow ip\_to\_binary(IP)$          ▷ Convert IP addresses to binary format
for all $v_i \in \mathcal{V}$ do
    for all $v_j \in \mathcal{V}$ do
        if $v_i \neq v_j$ then
            $similarity \leftarrow$ cosine_similarity$(bIP(v_i), bIP(v_j))$
            if $similarity > 0.95$ then
                $HLM(v_i) \leftarrow similarity$
                $HLM(v_j) \leftarrow similarity$
            end if
        end if
    end for
end for
return $HLM$
```

6.4. Combining sub-metrics into a metrics

PR, *BM*, and *HLM* are combined into one rank score by leveraging the *Single Usability Metric* (SUM) principle (Sauro and Kindlund, 2005). All three sub-metrics are expressed in different units, and the SUM principle allows for reliably combining them into a single score value. We define the combined rank score as follows.

Definition 6. (Combined rank score). *Given PageRank metric PR, Blackhole metric BM and Hidden link metric HLM, the combined rank score z of nodes is computed such that,*

$z = (z_{PM} + z_{BM} + z_{HLM})/3.$

z_{PM}, z_{BM}, and z_{HLM} are normalized values of each sub-metrics by standardization process such that each sub-metric values is subtracted from a benchmark value and divided them by the standard deviation, respectively. Benchmark serves as a bar, with scores above it being good and scores below it being bad. In Table 5, benchmark values for each sub-metric are shown.

First, *PM* has a benchmark value of 4 as conventionally PageRank scores at or above 4 are considered to be an above-average score and sufficiently influential. Second, in order for network to be considered as a blackhole network, the network should consist of at least two nodes and meet other conditions of blackhole network. It means $BM = 2$ is the minimum benchmark here. Third, our purpose for determining HLM was to determine the affinity between two nodes and cosine similarity was used to calculate the affinity measure in our study. For our experiment, the affinity was measured by a single target feature, which is IP address and therefore *HLM* was set to 0.95 including a small tolerance fraction (0.05).

Note that the combined rank score does not replace individual sub-metrics but rather summarizes them in a more condensed way for better interpretability. The individual sub-metrics can still be explored separately if necessary.

7. AR module

In this section, we introduce the AR module which involves the graph anomaly detection process. In addition, we briefly summarize the application of *Graph Neural Networks* (GNNs) methods in the financial sector in Section 2.1.

The HR Module can be effective in spotting blackhole patterns and hidden relationships between bank accounts but may fail in the face of more complex schemes used by more sophisticated criminals. Therefore, we introduce another complementary ranking module based on graph anomaly detection.

7.1. Graph anomaly detection

Graph anomaly detection identifies abnormal data in a set mostly composed of normal data represented as a graph. It has been an active area of research in recent years, especially in the financial sector. This technique also finds applications in various fields, including medicine, neuroscience, and law enforcement. In its turn, financial data is suitable for graph anomaly detection due to its highly connected structure and we employ a GNN-based technique to identify abnormal nodes in our data.

Table 5
Benchmark values.

Sub-metric	Benchmark value
PageRank metric *PR*	4
Blackhole metric *BM*	2
Hidden link metric *HLM*	0.95

Before moving into further details, it is worth mentioning the common challenge in applying any type of anomaly detection technique to financial datasets. It is a highly imbalanced nature of financial datasets. Illegal transactions are estimated to represent a small fraction, typically less than 5 % of the overall financial transaction volume. To address this issue, we incorporate hypersphere learning into the AR module.

Our AR module involves a GNN-based anomaly detection technique that ranks bank accounts by generating anomaly scores. This method combines the representation power of GNNs with the hypersphere learning objective to identify anomalies in a graph. To overcome the challenge of imbalanced datasets, we use a one-class classification algorithm, which has been shown to be effective for detecting suspicious nodes in highly skewed class distributions (Seliya et al., 2021). Hypersphere learning, combined with node embeddings, helps to single out unusual data, proving useful in various anomaly detection tasks such as differentiating time-sensitive anomalous patterns (Teng et al., 2017), or isolating the spatio-temporal anomalies (Teng et al., 2018).

The GNN automatically extracts typology information from the graph, enabling anomaly detection without extensive knowledge of graph theory. By aggregating neighborhood information, the GNN computes the embedding of each node, and the hypersphere learning objective minimizes the volume of a hypersphere that encloses those node embeddings. Anomalous nodes are identified based on their embedding location outside of the hypersphere. The anomaly detection process of our approach can be summarized as follows.

1. First, the AR module leverages GNN to compute the embedding of each node, which is then used for downstream anomaly detection.
2. Second, it adopts a one-class classification algorithm, more precisely, a hypersphere learning objective to tackle the issue of high imbalance in the dataset.
3. Finally, the output of anomaly detection is the anomaly scores of nodes, which can be used to rank them.

A more detailed explanation of the GNN and hypersphere learning is given in the subsequent subsections 7.2 and 7.3.

7.2. Graph Neural Networks

Graph Neural Networks (GNNs) is a type of deep learning method used for data represented by graphs. They can be directly applied to graphs and have various architectures, including Graph Convolutional Network (GCN) (Kipf and Welling, 2016), GraphSAGE (Hamilton et al., 2017), and Graph Isomorphism Network (GIN) (Xu et al., 2019).

This paper uses Graph Convolutional Networks (GCN), which extend the capabilities of Convolutional Neural Networks to graph data. GCN extracts domain information from the input graph based on the node attributes and structural information. Then, it generates the embedding matrix, which is used for the anomaly detection task. Given a transaction graph $\mathcal{G} = (\mathcal{V},\mathcal{E})$, the input and output of GCN are listed as follows.

- Input: transaction graph $\mathcal{G} = (\mathcal{V},\mathcal{E})$ represented by an adjacency matrix.
- Output: Node embedding matrix, which is denoted as matrix Z with dimension $N \times M$ (M is the number of output classification categories per node).

The input graph needs to be transformed by multiple hidden convolutional layers before producing output. The process of the hidden layer is shown as Equation (3), where $H(0)$, and $H(l)$ represent the input and output of the GCN, respectively.

$$H(l+1) = f(H(l), A) \tag{3}$$

In each hidden layer, the input matrix is computed using Equation (4), where $h_{v_i}^{l+1}$ and $h_{v_i}^{l}$ represent the attribute set for node v_i in layer $l+1$ and layer l, $Ne(v_i)$ represents set of neighboring nodes of node v_i, $W_{R_{v_j}}^{l}$ represents weight parameter to transform the information from node v_j, $\frac{1}{c_{v_i v_j}}$ represents normalization factor, such as degree of node v_i. Moreover, σ represents a non-linear activation function like the ReLU.

$$h_{v_i}^{l+1} = \sigma\left(\sum_{v_j \in Ne(v_i)} \frac{1}{c_{v_i v_j}} h_{v_j}^{l} W_{R_{v_j}}^{l}\right) \tag{4}$$

In its turn, the hidden layers of the GCN processes the input matrix in the following three steps.

- *Send.* Each node sends its attribute information to its neighbor node.
- *Receive.* Each node receives attribute information from its neighboring nodes.
- *Transform.* Collected attributes and structural information are transformed using a non-linear function.

7.3. Hypersphere learning

Hypersphere learning, originally proposed in Support Vector Data Description (SVDD) (Tax and Duin, 2004), involves learning a hypersphere space for all data and detecting which new objects can be contained within this space. This technique is useful for detecting anomalies in imbalanced datasets by describing a boundary around mostly normal data and identifying objects that fall outside of this boundary as anomalies.

Hypersphere learning uses a soft margin to allow the possibility of anomalies and penalizes points outside the hypersphere boundaries. The loss is defined as Equation (5), where *f* maps input data to the output kernel space, *c* and *r* are the center and radius of the hypersphere, and ν is the trade-off between the volume of hypersphere and boundary violations, representing the fraction of anomalies.

To address imbalanced datasets, we employ the penalization principle of hypersphere learning. We constrain the node embeddings, generated via GCN, to lie within a minimum hypersphere boundary, ensuring a small fraction of anomalies. We formulate the definition of the anomaly score using Equation (5).

$$F(r,c,f) = r^2 + \frac{1}{\nu n}\sum max(0, \|f(x_i) - c\|^2 - r^2) \tag{5}$$

Definition 7. (Anomaly Score). *Given a directed graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and *matrix of node embeddings, denoted as Z, anomaly score of node v,* $v \in \mathcal{V}$, *is denoted as S(v) and generated based on the location of its corresponding node embedding Z(v) with respect to the sphere, such that,*

$$S(v) = \|Z(v) - c\|^2 - r^{*2} \tag{6}$$

Based on the generated anomaly scores of nodes, we can establish rankings from the AR module on top of the rankings from the HR module.

8. Experimental Results

This section focuses on the evaluation of the proposed framework, including the experimental setup, evaluation metrics, and performance results.

8.1. Experimental setup

We evaluate our framework using the Berka dataset, which lacks of ground-truth labels, presenting a common challenge for effective evaluation. To overcome this problem, the data augmentation process was carried out as explained in section 4.2. Our approach is to generate a synthetic dataset and inject suspicious patterns for evaluation. The injected accounts exhibit suspicious behavior, such as forming blackholes or having hidden links with other accounts. Thus, we evaluate the framework based on how well it ranks these suspicious accounts. Essentially, if the framework outputs high-ranking scores for these accounts, it is considered effective.

8.2. Evaluation metric

To evaluate and check the accuracy of the proposed approach, we use the following evaluation metrics.

- **AUC**: We use the *Area Under the receiver operating characteristic Curve* (AUC) to evaluate the proposed approach. AUC captures the trade-off between the true positive rate (TP) and the false positive rate (FP). TP measures the detection rate of true suspicious nodes, while FP measures the rate of normal nodes falsely identified as suspicious.
- **Precision@K** and **Recall@K.** AUC can be misleading for highly imbalanced datasets. Therefore, we additionally use Precision@K and Recall@K. Precision@K measures the proportion of true positives in the top K-ranked nodes, while Recall@K measures the proportion of true positives out of all actual ground truth labels. Precision quantifies the number of correct positive decisions made, while recall quantifies the number of correct positive decisions made out of all positive decisions that could have been made.

8.3. Dataset for the experiment

For our experiment, we prepared a synthetic dataset, which was augmented from the Berka dataset. As Berka dataset includes normal bank transaction data, we injected transactions generated by AMLSim (Suzumura and Kanezashi, 2021) for virtual Hawala transactions.

AMLSim is a simulator that generates synthetic bank transaction data with known money laundering patterns by manually setting the initial parameters such number of bank accounts, and the number of transactions. It has been used to test machine learning models and graph algorithms. It incorporates a wide variety of normal and money laundering typologies. The normal ones consist of some simple typologies. Meanwhile, money laundering typologies are more complex and they include the combination or extension of normal typologies.

AMLSim simulator needs five input files to generate transactions. The first input file is the transaction file that specifies the transaction types, which is currently limited to wire transfers only. The second input file is the account information file, which contains data about the bank accounts including initial balance, country and bank affiliation. Table 6 shows the input parameters for account information file.

The third input file contains the degree distribution for each bank account, specifying its incoming and outgoing degrees. This determines the connections (transactions) between accounts within the simulated network. Table 7 shows the input parameters for the file.

The fourth input file defines the normal transaction patterns, including the amount, period, and a number of accounts involved. Table 8 shows the input parameters for the normal model file.

The final input file contains the parameters for the money laundering patterns. These patterns can be defined by various factors such

Table 6
Account input file.

count	min bal.	max bal.	country	bis.type	model	bank id
750	10	20963	US	I	0	AB
750	200	102547	UK	I	1	ST
750	14	2001	FR	I	2	IJ
750	12	5478	CH	I	3	KL
750	100	2343	IT	I	4	OP

Table 7
Degree input file.

count	in-degree	out-degree
0	1	2
1	24	20
2	132	132
…	…	…
97	1	1

Table 8
Normal model input file.

count	type	sche. Id	min acc.	max acc.	min per.	max per.
750	Single	2	1	1	5	20
750	Fan-out	2	5	20	5	20
750	Fan-in	2	5	20	5	20
750	Forward	2	3	3	5	20
750	Mutual	2	2	2	5	20
750	Periodical	2	2	2	5	20

as the type of money laundering pattern, and the number of accounts that are involved in the money laundering scheme. The input file format is shown in Table 9.

8.4. Performance results on the dataset

The performance results with respect to AUC, Precision@K, and Recall@K on top-ranked @K bank accounts are shown in Tables 10 and 11. The values of K are 50, 100, 150, 200, and 300 in our experiment. Tables 10 and 11 show the rank results for the HR and AR modules, respectively. Moreover, we integrated the rank results of both modules and identified the level of intersection between these rankings for top @K bank accounts. Next, we computed the precision and recall scores of the intersecting ranking set for each @K value. The evaluation results are presented in Table 12.

Table 10 shows that most of the bank accounts detected by HR module are on one hand, real suspicious accounts and we can reduce the number of cases that the inspectors of banks or law enforcement agencies have to investigate. For example, only two accounts among fifty do not belong to suspicious ones as precision is 96 %, when $K = 50$. On the other hand, many suspicious accounts may be missing in the results. For example, we can detect about only the half of suspicious accounts when $K = 50$ as recall is 46.6 %. If we want to detect most of the suspicious account, we have to increase K to 300. The determination of K value is therefore up to the inspectors.

Unlike HR, AR module does not show high precision measures. For example, precision is only 68.0 % when $K = 50$, which is much lower that the accuracy of HR module. On the contrary recall is comparable with HR module as we increase K value. As HR module is designed specifically to discover blackhole patterns and hidden links of Hawala network while AR is to detect any anomalies, the detected accounts may not necessarily belong to blackhole patterns or have hidden links but any other abnormal patterns. Although the precision of AR is significantly lower compared to the HR, increasing the value of K enhances the detection of abnormal patterns,

Table 9
Money laundering model input file.

count	type	sch.	min.	max.	min.	max.	min.	max.
		id	account	account	amount	amount	persons	persons
750	Fan-in	2	2	13	0.01	8740	5	60
750	Fan-out	2	2	11	0.01	8740	5	60
750	Cycle	2	2	9	0.01	8740	5	60
750	Random	2	2	7	0.01	8740	5	60
750	Bipartite	2	2	5	0.01	8740	5	60

Table 10
Performance of **HR module** w.r.t. Precision, Recall, F1-score, and AUC.

Rank module	K	Evaluation Metric			
		Precision	Recall	F1	AUC
HR module	50	0.960	0.466	0.627	0.987
	100	0.590	0.572	0.581	0.981
	150	0.513	0.747	0.608	0.978
	200	0.460	0.893	0.607	0.973
	300	0.326	0.951	0.486	0.954

Table 11
Performance of **AR module** w.r.t. Precision, Recall, F1-score, and AUC.

Rank module	K	Evaluation Metric			
		Precision	Recall	F1	AUC
AR module	50	0.680	0.330	0.444	0.989
	100	0.610	0.592	0.601	0.989
	150	0.520	0.757	0.617	0.989
	200	0.425	0.825	0.850	0.989
	300	0.340	0.990	0.506	0.989

including those associated with blackholes. For instance, with *K* set at 300, only two accounts among 200 illicit ones can evade detection by the AR module, as evidenced by a recall of 99.0 %.

In order to overcome the shortcomings of HR and AR modules, we conducted an additional experiment to integrate the results from HR and AR modules. We intersected the accounts detected by both HR and AR modules, and the experimental results obtained from the intersection show great promise. The intersection of HR and AR includes the accounts detected by both modules, which are considered suspicious by HR and AR. Although the intersection rate of the two modules may not be high (the lowest rate is 26.0 % and the highest rate is 60.0 %), the precision and recall scores of the intersecting ranking sets were satisfactory as shown in Table 12. For example, in the case of the top 50 nodes, there were only 13 bank accounts included in both module results, but the precision score was 100 %, indicating only true positive cases. This trend applies to all @K values, and precision scores of intersecting ranking sets are significantly higher than when HR and AR modules are considered separately. It is worth noting that precision is a critical metric as it involves false positive cases, and it is challenging to achieve satisfactory precision results without ground-truth data.

Based on this observation, we can conclude that the intersection set of HR and AR modules offers a more refined rank result with fewer false positives. The fact that the two modules have a low intersection rate is an advantage since it provides a shorter list of bank accounts for inspectors to investigate, with a higher probability that they are non-benign.

9. Discussion

In this section, we discuss the strength and weakness of our methods based on the observation of experiments.

As the investigation on money laundering particularly for terrorist funding becomes an increasing task of law enforcement agents in each country and a large part of the task is relied on manual ways, the automated detection is highly demanded. Fully automated detection is however hard to achieve and we need to increase the ratio of automated detection to reduce the cost of manual investigation. The cost of manual work primarily stems from the process of identifying actual Hawala transactions from the results of automated detection.

In order to understand this more concretely, it is important to focus on the practical meaning of two metrics — precision and recall — as discussed in the previous section. On one hand, increasing recall plays a crucial role in preventing transactions from escaping automated detection. On the other hand, increasing *K* value, the recall of the proposed methods approaches to 1. However, this also leads to an increase in irrelevant transactions with the actual Hawala network, which subsequently increases the amount of manual inspection and consequently increases the cost.

We observed in the experiment that when both HR and AR were applied to produce an intersection result, it was possible to achieve a

Table 12
Performance of HR module & AR module w.r.t. Precision, Recall, and F1.

K	Intersection of HR module & AR module			
	Intersection rate	Precision	Recall	F1
50	26.0 %(13)	1.000	0.126	0.224
100	37.0 %(37)	0.892	0.320	0.471
150	49.3 %(74)	0.776	0.553	0.646
200	54.0 %(108)	0.694	0.728	0.711
300	60.0 %(180)	0.538	0.941	0.685

precision of 54 % while maintaining a recall of 94 % with $K = 300$. This means that only 12 would be missed out of 200 illegal Hawala transactions and the false positive rate would be only 46 %, significantly reducing the cost of manual inspection. It indicates that by examining approximately double of illegal Hawala transactions, 94 % of illegal transactions can be identified by the proposed method, thus significantly reducing the cost incurred by unnecessary manual inspection efforts. In comparison with a use-case study in UK (Demetis, 2018) where only 5 % of suspicious transaction reports were labeled as truly suspect, our method significantly improves the ratio of true positives. It is expected that our method would contribute to significantly reduce the cost for manual inspection.

While our approach is expected to reduce the cost for manual investigation efforts to detect Hawala transactions, there are several weaknesses. First, it is limited to the detection of Hawala network. Second, it is not fully validated with real data due to the confidentiality policy on transaction data. We leave the validation of robustness with real data sets as a future work, which requires legal and institutional arrangements.

To the best of our knowledge, the proposed method is the first approach to detect Hawala network in automated ways and it is difficult to directly compare our approaches with any precedent ones.

10. Conclusion

In this paper, we presented a framework for detecting suspicious bank account, particularly related with Hawala activities. Our framework is summarized as follows:

1. Building the transaction graph with given bank transaction data, where nodes are accounts and edges are bank operations between bank accounts.
2. Computing HR rank of each node in the graph using the specified Hawala characteristics.
3. Computing AR rank of each node in the graph by applying an unsupervised anomaly detection technique.
4. Metrics from HR and AR ranks are provided as a final output to the inspectors of banks and law enforcement agencies for further investigation.

In addition to the framework, we also generated a synthetic dataset by augmenting a bank transaction dataset, called Berka dataset. This dataset includes transactions generated by AMLSim and additional properties related with Hawala activities as well as normal transactions.

The HR and AR modules both performed well on the synthetic dataset that we augmented from Berka dataset within their respective scopes. The HR module had a lower false positive rate compared to the AR module. However, considering that the AR module uses unsupervised learning, it still demonstrated good performance. An interesting observation from our experiment is that high recalls were achieved with relatively smaller sacrifices of precision by intersecting HR and AR than applying HR or AR separately.

An insight to anti-money laundering regulation that we find from our study is related with the measure to report suspicious transaction. For example, the EU Anti-Money Laundering Directive provides criteria, under which obliged entities such as banks have to report to the financial intelligence units (FIU). Cash deposits exceeding USD 10K have to be reported to FIU. However, money laundering for terrorist funding is mostly conducted with many transaction of small amounts to avoid the surveillance by law enforcement agents. The proposed method for detecting Hawala network and black hole transaction patterns is expected to serve as a proper tool.

In conclusion, our framework can be a valuable aid to bank experts and improve their operational efficiency. However, it should be viewed as a complement to existing systems rather than a replacement. This paper highlights the significance of raising awareness about Hawala and its connection to money laundering and terrorist financing. More work should be dedicated to the research and development of the Hawala detection system, as well as its validation on larger datasets, ideally on real datasets.

Declaration of competing interest

The authors declare no conflict of interest.

References

Adams, P., Doman, M., Kuchipudi, P., 1999. The Berka Dataset. https://paperswithcode.com/dataset/the-berka-dataset. (Accessed 2 November 2023).

Bowers, C.B., 2009. Hawala, money laundering, and terrorism finance: mirco-lending as an end to illicit remittance. Denver J. Int. Law policy 37, 379–419.

Brin, S., Page, L., 1998. The anatomy of a large-scale hypertextual web search engine. Comput. Netw. ISDN Syst. 30, 107–117.

Chêne, M., 2008. Hawala Remittance System and Money Laundering. U4 Anti-Corruption Resource Centre.

Day, A., Risdal, M., Dane, S., 2019. Bitcoin Blockchain Historical Data. https://www.kaggle.com/datasets/bigquery/bitcoin-blockchain. (Accessed 2 November 2023).

del Cid Gomez, J.M., 2010. A financial profile of the terrorism of al-qaeda and its affiliates. Perspect. Terrorism 4.

Demetis, D.S., 2018. Fighting money laundering with technology: a case study of Bank X in the UK. Decis. Support Syst. 105, 96–107.

Ding, K., Li, J., Bhanushali, R., Liu, H., 2019. Deep anomaly detection on attributed networks. In: SIAM International Conference on Data Mining 2019.

Dou, Y., Liu, Z., Sun, L., Deng, Y., Peng, H., Yu, P.S., 2020. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In: Conference on Information and Knowledge Management 2020, pp. 315–324.

Europol, 2022. Cryptocurrencies: tracing the evolution of criminal finance. Europol Spotlight Report series. Publication Office of the European Union, Luxembourg, p. 20.

FATF, 2013. The Role of Hawala and Other Similar Service Providers in Money Laundering and Terrorist Financing. FATF.

FATF, 2020. Virtual Assets Red Flag Indicators of Money Laundering and Terrorist Financing. FATF.

FATF, 2021. Opportunities and Challenges of New Technologies for AML/CFT. FATF.

Freeman, M., Ruehsen, M., 2013. Terrorism financing methods: an overview. Perspectives on Terrorism 7, 5–26.

Hamilton, W.L., Ying, R., Leskovec, J., 2017. Inductive representation learning on large graphs. In: Conference on Neural Information Processing Systems 2017.

Kipf, T.N., Welling, M., 2016. Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations 2017.

Li, Z., Hui Xiong, Y.L., Zhou, A., 2010. Detecting blackhole and volcano patterns in directed networks. In: IEEE International Conference on Data Mining.

Liu, Z., Chen, C., Li, L., Zhou, J., Li, X., Song, L., Qi, Y., 2018. Graph Neural Networks with Adaptive Receptive Paths. Association for the Advancement of Artificial Intelligence.

Liu, Z., Dou, Y., Yu, P.S., Deng, Y., Peng, H., 2020. Alleviating the inconsistency problem of applying graph neural network to fraud detection. Special Interest Group on Information Retrieval 2020, 1569–1572.

Machine Learning Group at Université de Bruxelles, 2018. Credit Card Fraud Detection. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud?resource=download. (Accessed 2 November 2023).

Maxwell, W., Bertrand, A., Vamparys, X., 2020. Are Ai-Based Antimoney Laundering (AML) Systems Compatible with European Fundamental Rights? ICML 2020 Law and Machine Learning Workshop.

McCusker, R., 2005. Underground Banking: Legitimate Remittance Network or Money Laundering System? Australian Institute of Criminology.

Oeben, M., Goudsmit, J., Marchiori, E., 2019. Prerequisites and AI challenges for model-based anti-money laundering. In: Proceedings of 2019 International Joint Conference on Artificial Intelligence Workshop, pp. 1–4, 2019.

Passas, N., 2005. Informal Value Transfer Systems, Terrorism and Money Laundering. U.S. Department of Justice.

Sakharova, I., 2011. Al qaeda terrorist financing and technologies to track the finance network. In: Proceedings of 2011 IEEE International Conference on Intelligence and Security Informatics, pp. 20–25.

Sauro, J., Kindlund, E., 2005. A method to standardize usability metrics into a single score. In: Conference on Human Factors in Computing Systems, pp. 401–409.

Seliya, N., Abdollah Zadeh, A., Khoshgoftaar, T.M., 2021. A literature review on one-class classification and its potential applications in big data. Journal of Big Data 8 (1), 1–31.

Soudjin, M., 2014. Hawala and money laundering: potential use of red flags for persons offering hawala services. Eur. J. Crim. Pol. Res. 21, 257–274.

Suzumura, T., Kanezashi, H., 2021. Anti-money laundering datasets. https://github.com/IBM/AMLSim. (Accessed 2 November 2023).

Tax, D.M., Duin, R.P., 2004. Support vector data description. Mach. Learn. 54, 45–66.

Teichmann, F.M.J., Wittmann, C., 2022. The abuse of hawala banking for terrorist financing in German-speaking countries. J. Money Laund. Control.

Teng, X., Lin, Y.-R., Wen, X., 2017. Anomaly detection in dynamic networks using multi-view time-series hypersphere learning. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 827–836.

Teng, X., Yan, M., Ertugrul, A.M., Lin, Y.-R., 2018. Deep into hypersphere: robust and unsupervised anomaly discovery in dynamic networks. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence. IJCAI-18, pp. 2724–2730.

Vesta Coorporation, 2019. IEEE-CIS Fraud Detection. https://www.kaggle.com/competitions/ieee-fraud-detection/overview. (Accessed 2 November 2023).

Wang, D., Lin, J., Cui, P., Jia, Q., Wang, Z., Fang, Y., Yu, Q., Zhou, J., Yang, S., Qi, Y., 2020. A semi-supervised graph attentive network for financial fraud detection. In: International Conference on Data Mining, pp. 598–607.

Wang, J., Zhang, S., Xiao, Y., Song, R., 2021. A review on graph neural network methods in financial applications. J. Data Sci. 20, 111–134.

Watkins, R.C., Reynolds, K.M., Demara, R., Georgiopoulos, M., Gonzalez, A., Eaglin, R., 2010. Tracking dirty proceeds: exploring data mining technologies as tools to investigate money laundering. Police Pract. Res.: Int. J. 4, 163–178.

Weber, M., Domeniconi, G., Chen, J., Weidele, D.K.I., Bellei, C., Robinson, T., Leiserson, C.E., 2019. Anti-money Laundering in Bitcoin: Experimenting with Graph Convolutional Networks for Financial Forensics. Workshop on Anomaly Detection in Finance 2019.

Wheatley, J., 2005. Ancient banking, modern crimes: how hawala secretly transfers the finances of criminals and thwarts existing laws. Journal of International Law 26 (2), 347–378.