國立中正大學電訊傳播研究所碩士論文

指導教授 :管中祥、簡廷軒

UF-FAE：整合 Union-Find 演算法與機器學習於分散式帳本應用之洗錢偵測系統框架

UF-FAE: A Money Laundering Detection System Framework Integrating Union-Find Algorithm and Machine Learning for Distributed Ledger Applications

研究生 ： 洪禎 撰

西元 2025 年 X 月

關鍵字:

UF-FAE (Fraud-Attributed Embedding)

Union-Find Algorithm

Anti-Money Laundering (AML)

Graph-Based Fraud Detection

Social Network Analysis

Computational Communication

一、緒論

1.1 研究背景

數位金融與電子支付在疫情期間普及率大幅提高，金融犯罪亦趨於複雜化(Abhinaya, 2024)，但各銀行維持獨立帳本、缺乏共識與同步化，在金流路徑上形成監控斷點(Blind spot)，犯罪者利用此缺口迅速跳轉資金逃避監管。至今洗錢佔全球 GDP 約 5% (Deprez et al., 2025; Feedzai, 2023; Tian et al., 2025)。

洗錢（Money Laundering）是將犯罪所得透過複雜的金融操作與合法交易渠道掩飾其非法來源，使其得以重新進入合法經濟體系，常見手段包括多層轉帳、貿易偽報等(Reuter & Truman, 2004; United Nations Office on & Crime, 2025)，洗錢行為不僅提供犯罪組織資金，與犯罪造成的傷害呈現顯著正相關。參與洗錢的組織其犯罪傷害平均增加近 49%，為犯罪組織提供動力來源(Morgan, 2024)，研究指出，全球反洗錢（Anti-Money Laundering, AML）市場預計將以年均 16.2% 的速度成長，2030 年達到 42.39 億美元(Grand View, 2025)，其危害對金融市場、人均 GDP、政府對貪腐與法治的管控能力、私營部門貸款機會，以及失業率產生顯著影響(Batool et al., 2024)。

Mirenda et al. (2022)指出，受黑手黨滲透的年輕且效率偏低企業雖然營收顯著提升，但未增加生產投入，卻導致財務健康惡化，而反洗錢政策雖打擊非法資金，卻可能因流動性收縮對合法小企業與地方經濟造成負面衝擊，顯示政策需配套流動性措施以降低副作用(Slutzky et al., 2020)。Becker (1968)認為，提高洗錢犯罪之破獲率較為重要，故本研究認為，應著重從於金流偵測之方法，提高洗錢防治效率。

洗錢流程可以分為三個階段 ：1.置入(Placement) 將非法資金引入金融體系; 2.分層(Layering)透過一連串交易混淆資金來源; 3.整合(Integration)將清洗過的資金納入合法經濟循環(United Nations Office on & Crime, 2025)。最常被使用的洗錢手段為小額拆分轉帳(Smurfing)，主要分成兩種模式:1.**Scatter-Gather**：將資金透過多個中間帳戶（mules）轉移後，最終集中至一個收款帳戶；2.**Gather-Scatter**：從多個來源帳戶集中資金至一個帳戶，再分發至多個目的帳戶，常見於加密貨幣（Cryptocurrency）交易(Starnini et al., 2021)。圖 1 及表 1 為 Altman et al. (2023)以 **8 種常見洗錢行為模式**作為代理人基模型(Agent base model, ABM)模擬。

表 1： 常見洗錢模式

模式代號	名稱	說明
(a)	Fan-out 模式	單一帳戶 v 向多個帳戶轉出資金（v 對多個節點發出邊），常見於資金分散。
(b)	Fan-in 模式	多個帳戶向同一帳戶 v 匯入資金（多邊指向 v），用於集中可疑資金。
(c)	Gather-Scatter 模式	v 先從多個來源接收資金，再轉出給多個目標，結合 Fan-in 與 Fan-out 特徵。
(d)	Scatter-Gather 模式	v 將資金散出到中介帳戶，最終由中介帳戶再將資金集中到目標 u，模擬混合路徑。
(e)	Simple Cycle 模式	洗錢資金形成封閉迴圈，最終返回原始帳戶，為明顯的異常訊號。
(f)	Random 模式	可控帳戶之間隨機流動，未返回起點，意圖模擬合法資金流向以混淆視聽。
(g)	Bipartite 模式	一組帳戶對另一組帳戶進行資金轉移，呈現平行式、批次性洗錢。
(h)	Stack 模式	Bipartite 的堆疊擴展，形成層層轉移的複雜結構，以掩蓋資金源頭與去向。

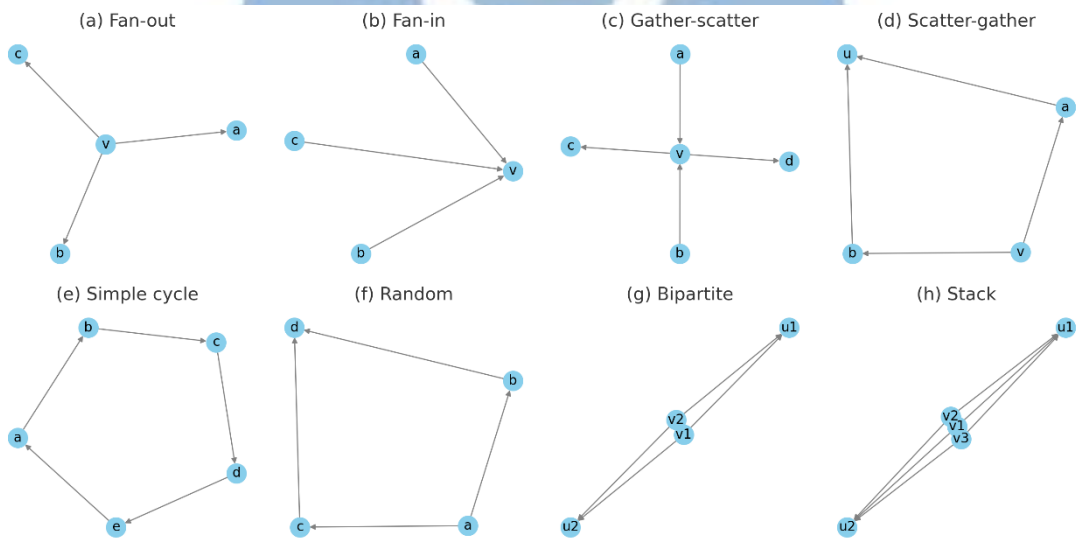


圖 1： 常見洗錢網絡圖

由此可見，洗錢手段雖日益複雜，但並非不可追蹤的無解問題，Watts and Strogatz

(1998)提出了小世界網絡(Small-World Network, SWN)，以節點和邊的圖論(Graph Theory)建立模型，用於分析人際關係之社會網絡（Social Network），本研究認為藉由社會網絡為基礎採用圖論演算法（Graph Theory Algorithms）能快速有效的追蹤金流。

疫情同時加速了電子支付的蓬勃發展與洗錢的惡化程度(Abhinaya, 2024)，電子支付與加密貨幣具有高度相關性(Hajr et al., 2023)，區塊鏈技術（Blockchain Technology）透過去中心化（Decentralization）提升支付安全性與透明性，其互操作性協議更有助於舊有系統與多鏈整合，推動全球數位資產與金融創新發展為金融機構與消費者帶來效益(Movva & Dasarraju, 2024)。

2023 年，來自可疑地址流向混幣器的資金大幅下滑，從 2022 年的 10 億美元降至 5.043 億美元。但如 Lazarus 等加密貨幣犯罪組織已經開始採用新的混幣器服務，如 YoMix，於 2023 年流入量比前一年成長超過 5 倍。約三分之一的 YoMix 流入資金來自於加密貨幣駭客活動相關的地址(Chainalysis Research, 2024)。2025 年，美國聯邦調查局(FBI)轄下網路犯罪投訴中心(IC3)公告，Bybit 遭駭事件造成約 15 億美元的加密資產遭竊，為迄今史上規模最大的加密貨幣竊盜案，並經追蹤確認是北韓 TraderTraitor 駭客組織所為(Federal Bureau of Investigation, 2025)。

加密貨幣是以比特幣(Bitcoin, BTC)為代表的區塊鏈（Blockchain）。其採用工作量證明(Power of Work, Pow)之共識機制（Consensus Mechanism）為首要代表，透過分散式運算（Distributed Computing）與分散式帳本（Distributed ledger Technology, DLT）確保交易不可竄改且全網可驗證，每筆交易皆以前一筆交易的雜湊值(Hash Value)串連成可驗證的金流結構(Nakamoto, 2008)，為本研究提供了對反金融犯罪系統的啟發，若建立快速可追蹤的金流網路，有助於提升金融偵測的效率。

區塊鏈技術在 AML 領域應用逐漸受到關注。金融機構嘗試運用區塊鏈技術提升 AML 效率，如透過不可篡改的交易紀錄，強化追蹤資金流向，利於偵測可疑交易(Huang & Trangle, 2020)。但礙於現行金融體制下直接導入加密貨幣使用的共識機制的可行性不高，AML 仍以依賴靜態規則進行篩選，難以掌握複雜的資金跳轉模式(Aidoo, 2025; Oad et al., 2021; Tian et al., 2025)。目前模型亦多假設集中式交易圖，機構間因法規限制資料共享，導致 AML 在金流跨機構偵測的效果有限而形成資料孤島（Data Silo）(Lucinity, 2024; Tian et al., 2025)。

縱然區塊鏈技術利於洗錢防制，但帳戶地址本身僅為加密雜湊的字串，而區塊鏈驗證機制只能反應交易的順序，而非現實世界中的人際網絡與其相關行為，因此本研究引進 Union-Find 演算法進行分群，藉由觀測交易邊(如 Sender-Receiver)之連結關係，將具相關性的交易帳戶動態歸納至同一交易網絡群組，捕捉近似真實社會網絡的結構，並以圖論指標來表示交易帳戶間的網絡和行為量化特徵，增強機器學習的金流偵測效率。

Aidoo (2025)指出，區塊鏈技術在強化交易可追蹤性、改善跨機構資料共享與提升 AML 效率具備潛力，但受到法規與擴展性（Scalability）問題所限制。為解決金融機構間資料孤島與即時追蹤困難的問題，本研究認為 DLT 是可行的解決方案，DLT 為一種可由多個節點共同維護並同步資料的數位記錄機制，具去中心化、一致性與抗竄改等特性，其創新應用被視為重塑金融基礎設施之關鍵技術(Financial Conduct, 2017; Treleaven et al., 2017; World Economic, 2016)。

區塊鏈由於其匿名性導致加密貨幣成為另一種新興的洗錢與犯罪手段，BTC 等加密貨幣匿，使犯罪者更容易進行非法活動且透過掩藏 IP、混合交易等方式逃避偵查(B, 2025)，以勒索軟體 WannaCry 為例，犯罪者要求被害人以 BTC 作為勒索電腦的贖金，但透過將公共記錄與 BTC 圖譜視覺化，能精確階露資金從受害者到犯罪者的流轉路徑，並指出儘管具備偽匿名性，仍可透過跨平台情報整合追蹤犯罪套現行為(Turner et al., 2019)。

區塊鏈同時成為助長和防治洗錢的雙面刃，自 2008 年 BTC 問世後，加密貨幣如雨後春筍般不但推出，如可保存完整資料的 IOTA Tangle 的 DLT 可支援高頻、低延遲，具備應用於跨機構 AML 系統之潛力(Sealey et al., 2022; Treleaven et al., 2017)，Becker (1968)指出比起加重刑度，應提高抓獲犯罪的機率。因此本研究認為跨機構間若藉由區塊鏈技術達到 DLT 並解決資料孤島問題可提高 AML 效率。因此提出以下研究問題：在多銀行、多用戶的交易網絡環境中，如何設計一種具有低時間複雜度（Time Complexity）、可即時運行且具擴展性的金融犯罪偵測演算法?

現實世界中，交易資料間具有關聯性，因此區塊鏈亦適用於使用圖資料結構（Graph Data Structure）來進行分析，其模式可分為靜態圖、動態圖、可解釋性以及實務應用四種類型，以詐欺網路為例，異常偵測（Anomaly Detection）如機會型詐欺或組織型詐欺(Akoglu et al., 2014)。以圖資料結構進行異常偵測所面臨的問題包括大數據分析（Big Data Analytics）的計算量問題(Abhinaya, 2024; Akoglu et al., 2014)，交易網路若無法在短時間內完成查詢和標記，洗錢行為即可能早已完成資金跳轉的所有流程(Tian et al., 2025)。而本研究之實驗設計假設於金融機構間採用 DLT 分享帳本，故資料較現行金融機構內部大數據之計算量更加龐大，因此本研究認為需要進行高速及高效率的演算法設計（Algorithm Design）因應快速洗錢。

由於區塊鏈及加密貨幣多採用共識機制進行人工驗證，僅靠人工手段並不足以應對快速洗錢，因此機器學習(Machine Learning, ML)成為現今 AML 的必要方法之一。Alotibi et al. (2022)指出傳統規則機制對加密交易偵測效果差，因此深度學習（Deep Learning）與機器學習能大幅提升識別可疑加密貨幣交易的準確率。

機器學習泛用於 AML 應對複雜洗錢手段(Aidoo, 2025)，但依賴過去的資料，一旦遭遇到新的資料型態將難以應對(Deprez et al., 2025; Pareja et al., 2020)。Neo4j (2021)指出資

料科學家以嚴謹的機器學習模型來偵測詐欺，但經常忽略網絡結構的重要性。Abhinaya (2024)認為圖論為基礎的偵測方法在處理不同來源的數據、模擬真實金融方面更擅長揭露異常行為。結合圖資料結構是更即時準確地偵測犯罪行為的方法，例如在圖資料庫中的節點（Node）代表資料點(IP 或裝置等)，邊（Edge） 表示節點間的互動關係，如交易紀錄和共用資訊(Neo4j, 2023)。

假設情境在跨銀行機構可共享資料的情況下運用 DLT 進行實驗設計，跨機構的交易資料勢必遠比現在龐大，採用機器學習方法雖然有效，但大數據必然對機器學習演算法的時間效率產生重大負面影響，導致無法做實際運用(Song et al., 2024)，因此如何使用圖資料結構以及演算法設計迅速有效地做金融偵測成為本研究的最重要項目之一。本研究以人工智慧（Artificial Intelligence, AI）、演算法設計及區塊鏈技術作為啟發，以 Union-Find with Path Compression 演算法實現分散式高效率的演算法設計。

1.2 研究目的

區塊鏈技術逐漸由傳統鏈式結構演進至 DAG（Directed Acyclic Graph）架構，IOTA 2.0 即為此類平台代表之一，透過平行化交易與零手續費機制提升系統吞吐量與資源效率，並廣泛應用於物聯網(Internet of Things, IoT)與即時資料交換等高頻場景(Bai et al., 2025)。IOTA 2.0 之 DAG 架構主要關注於交易拓撲（Transaction Topology）與可擴展性，對於模擬如社會網絡般的複雜人際互動關係仍有所侷限，難以有效揭示跨群帳戶行為與詐欺擴散脈絡。

於 AML 應用中，交易網絡通常具備高頻率、多樣參與者與強時間敏感性等特徵，若僅依賴靜態圖建模與傳統分類方法，常難即時捕捉詐欺群體的演化與網絡異常結構。因此，若能有效建構交易圖中的節點群組與動態關聯，並進一步擷取行為與結構特徵，將有助於實現具即時性與可擴展性的 AML 偵測架構。

本研究提出一套結合 Union-Find 路徑壓縮演算法與圖結構特徵融合的群組識別流程，目標為建構具結構可解釋性與實作效率的 AML 分析系統。具體研究目的如下：

1. 藉由 Union-Find 演算法作為分群骨架，改善 DAG 型交易網絡中節點間的以真實交易網路的動態關聯；
2. 擷取群組內交易圖之中心性指標與網絡統計量，並結合原始交易屬性進行特徵融合與輸出建模；
3. 評估本研究提出之方法在洗錢交易識別任務中相較於基準機器學習模型的分類效果與運算效率是否能提高；

4. 探討該架構於 IOTA 2.0 類 DAG 區塊鏈平台下對於 TPS 提升與安全性增強的潛在貢獻。

以上研究目的假設在 DLT 的情境下，由金融監管單位和機構共同維護其去中心化分散式系統，以 Union-Find 為演算法藍圖，並使用遞迴(Recursion)計算圖論指標的 UF-FAE 演算法來達到獲取個別族群及其行為特徵等量化指標的目的，並預期量化指標能有效增加機器學習對於金流異常偵測的效率。

二、相關研究

2.1 跨行轉帳與金流監控問題

金融機構因監管與隱私限制無法即時共享交易資料，形成資料孤島(Kumar, 2023; Talend, 2022; TechTarget, 2024)，而現行多採批次處理，導致跨行交易網絡圖更新延遲，使可疑資金常在風控警示前即完成跳轉(Lucinity, 2024)。傳統 AML 架構倚賴靜態規則與單筆特徵，對日益複雜、跨日多跳的小額拆單顯得力不從心；近半金融從業者亦坦言目前需要動態、跨機構策略才能跟上犯罪腳步(Aidoo, 2025; Feedzai, 2023; Tian et al., 2025)。雖有 BTS 結合區塊鏈與門檻規則(Oad et al., 2021)、FPC 隨機投票機制(Mamache et al., 2021)及 GARG-AML 的散佈-聚合圖模型(Deprez et al., 2025)，但它們分別受限於無 DAG 最佳化、拓撲依賴性與忽略金額時間特徵；而私有鏈式 CDD(Xu et al., 2021)與動態 GCN(Pareja et al., 2020)又分別因硬體門檻與可解釋性不足，在跨行大規模場景難以落地。綜上所述，本研究認為若要真正縮短偵測落差、扭轉被動局面，關鍵在於突破資料孤島、整併多機構交易圖並以可擴展、低延遲的圖論演算法為核心作為 DLT 帳本，方能構築面向未來的金流防線。

DLT 架構涵蓋鏈式區塊鏈、DAG、哈希圖（Hashgraph）等資料記錄方式，其中 DAG 架構因其支援多筆交易並行處理、零手續費與高吞吐量的特性，逐漸受到 IoT 與高頻交易應用的重視。以 IOTA 為代表的 DAG 型平台，即為一種兼顧擴展性與效率的區塊鏈設計。

然而 IOTA2.0 DAG 架構在處理 AML 問題時，以交易資料作為節點，以驗證引用順序為邊，透過 MCMC 以及 DFS 建立隨機引用前兩筆交易作為驗證機制，並非以社會網絡關係為主，加上 MCMC 演算法易偏向主鏈進行驗證，導致部分節點形成孤塊，有遭到駭客攻擊等風險。

2.2 分散式帳本技術

DLT 是指一種透過網路中多個節點共同維護的資料記錄架構，資料一旦被記錄即不可隨意篡改，並可於各節點間同步更新。不同於傳統由中央伺服器集中控制的資料庫系統，DLT 將紀錄權力分散至所有參與者，確保資料的完整性、透明性與抗竄改性(World Economic, 2016)。

DLT 並不等同於區塊鏈，而是包含區塊鏈、DAG、哈希圖等資料結構的上位概念。其中，區塊鏈是將交易資料依時間順序打包為區塊並串接形成線性鏈，而 DAG 則允許多筆交易並行進行，透過引用先前的交易達成共識，提升可擴展性與處理效率(Treleaven et al., 2017)。

根據 Financial Conduct (2017)之定義，DLT 是一種可同時於多處記錄資產交易資料之數位系統，無需中央管理者亦能確保各筆交易一致性與真實性。該技術具備去中心化、高可用性、抗故障與可追蹤等特性，特別適合應用於金融、供應鏈管理、數位身份驗證及物聯網等領域。

近年來，DLT 技術逐步受到金融業界關注，作為資料共享與資金流動透明化的潛在解方。特別是在跨機構合作與 AML 等場景中，DLT 能突破資料孤島的限制，實現即時資訊整合與風險預警。作為 DLT 一種具代表性的技術架構，IOTA 採用 DAG 為核心資料結構，在維持交易完整性的同時，實現高頻低延遲之資料傳遞與驗證機制，為本研究之模擬與演算法設計奠定基礎。

為強化 AML 分析之可解釋性與群組關係辨識能力，本研究不直接使用 DAG 原生架構，而是針對交易資料作為邊，建構無向圖（Undirected Graph），並應用 Union-Find 路徑壓縮方法進行弱連通元件（Weakly Connected Components, WCC）分群，進而萃取交易網絡的結構特徵。此方法在保留 DLT 架構的非中心化特質與高頻資料處理需求的同時，亦能提高模型對異常行為的偵測效能與時間敏感性。

2.3 區塊鏈共識機制的演進：PoW → PoS → Solana 與 IOTA

BTC 問世以來，便以其不可竄改的公開帳本和 PoW 機制，讓礦工們以 SHA-256 雜湊函式競爭解題，以獲取打包區塊和獎勵的權利(Nakamoto, 2008)。PoW 雖具有安全性，但存在電力和算力成本昂貴的問題，其每秒交易數(TPS)僅約為 7，交易速度緩慢進而降低用戶之使用意願(Hinzen et al., 2018)。

為解決 BTC 的龐大成本問題，以太坊(Ethereum, ETH)自 2022 年開始，由 PoW 過渡至權益證明（Proof of Stake, PoS），宣布進入 ETH2.0 階段。PoS 不依賴電力及算力成本，而是根據持幣量和和質押(Staking)時間決定驗證權限，可減少 99%的能源成本，TPS 的理論值可達 100000(Buterin, 2013, 2020)。

但 PoS 存在權力向少數持幣者集中的缺陷，導致壟斷和治理中心化風險，2022 年 LUNA 事件即因大量 LUNA 幣被少數主體控制，最終使市場失去信心而瓦解(Badev & Watsky, 2023)。基於上述問題，Jonas et al. (2021)提出了 IOTA Tangle 2.0，利用 DAG 形成交易網絡，並非使用傳統鏈式結構，每筆交易需引用兩筆先前的交易來形成網絡達到去中心化目的，並且推出 On Tangle Feeless Parallel Consensus(OTFPC)機制，配合 Mana 經濟模型，鼓勵用戶持續參與和驗證交易，Mana 為另一種加密貨幣，用於訪問 IOTA 的創建區塊，為可消耗的資源，且可用於驅動智能合約、去中心化金融(DeFi)等服務，持有 IOTA 可以藉由驗證(Validating)及委託(Delegating)的方式產生 Mana 代幣，為防止壟斷，持有 IOTA 所產生的 Mana 會隨著時間逐漸產生邊際衰退效應，若不將 Mana 代幣再度投入創建區塊或其他應用，生成的 Mana 會逐漸變少，以刺激用戶不斷重複投入區塊鏈的參與。

本研究認為 IOTA 2.0 經濟模型上存在中心化風險，首先是 Mana，若透過中心化交易所（Centralized Exchange, CEX）持有 IOTA，所產生的 Mana 並不會流進用戶個人的資產中，而是被 CEX 控制，如 Coinbase 及 Binance 兩大交易所皆並未告知用戶 IOTA 所產生之 Mana 流入 CEX 中並且加以控制。再者，OTFPC 採用偏向主鏈的隨機深度優先搜尋（Randomized Depth-First Search, Randomized DFS）作為驗證路徑，部分交易節點可能未被納入主鏈而變成孤塊(Orphaned transactions)，存在被駭客攻擊等風險。

Solana 以其極高的 TPS 聞名，但其系統卻頻繁遭遇詐欺與資安攻擊事件，問題核心多來自應用層的安全治理不足與基礎設施過度集中所致，其中 Rug Pull 案件發生相當頻繁，Rug Pull 為惡意開發者透過承諾高額收益來吸引使用者投資，待募集到目標存放在智慧合約中的資產後，快速移除並掩蓋其痕跡(Cernera et al., 2023)，而 Solana 之使用者和其 Rug Pull 案件同時快速增加(Alhaidari et al., 2025)。

Solana 採用 PoS 機制，在正常條件下提供高速交易，但在網路分割或遭受攻擊時，存在鏈停擺或回滾的風險(Rifat Hossain et al., 2024)，實務上，節點運營集中於少數主體，容易因節點異常導致停電或網路中斷而全鏈停止出塊(CryptoManiaks Editorial, 2024)。

Solana 等區塊鏈技術與加密貨幣雖具潛力，卻同時放大其安全治理上的弱點，顯示在區塊鏈應用於金融領域時，仍需結合異常偵測機制，Neo4j (2021)指出 Weakly Connected Components（Union Find）可作用於辨識共用資訊的互動頻繁之潛在詐欺集團，Abhinaya (2024)認為圖論為基礎的偵測方法（如 GNN）在處理不同來源的數據、模擬真實金融方面更擅長揭露異常行為。

鑒於上述，本研究將原先的 IOTA 2.0 之圖論架構轉化為用於金流追蹤系統，結合

Union-Find with Path Compression 演算法之應用，建立一套具備高可擴展性，即時性強，且能精確偵測異常交易的風控系統。

2.4 圖論演算法作為金流偵測之基礎

分析複雜的大數據資料時，發現其異常資訊與了解其整體結構同等重要，而資料探勘領域中專門用來發掘罕見事件的分支稱為異常偵測(Akoglu et al., 2014)。金融領域中，圖論演算法常被使用於辨識可疑洗錢交易。Wang and Dong (2009)以最小生成樹（Minimum Spanning Tree, MST）為基礎將資料分群後，尋找並識別異常群組。Li et al. (2017)則提出基於 Spark GraphX 架構的 TD-Louvain 演算法，透過同步與更新第一及第二階段資料，解決社群交換（Community Swap）及延遲歸屬（Ascription Lag）問題，精確識別潛在洗錢集團。

圖資料結構本身由節點及邊組成，節點與邊的重要性常透過圖論中心性分析量化，如 Brin and Page (1998)提出的 PageRank 演算法，利用機率及圖論評估節點重要性，對後續研究產生深遠影響。本研究目的在於追求高效金流偵測，因此採用以 Union-Find 為基礎的 WCC 無向圖演算法。WCC 透過節點間的關係將帳戶分成獨立子圖，便於後續分析及視覺化。Neo4j (2021)亦指出，WCC 可有效辨識潛在詐欺集團。儘管 WCC 為無向圖，但透過交易金流方向(發送方及接收方)，可將 WCC 轉為更具分析價值的有向圖(Directed Graph）結構，本研究利用 Python NetworkX 套件進行此有向圖的視覺化。

為即時處理大規模交易網絡，實務上需要考慮分散式演算法的效能及可行性。Union-Find 演算法結合路徑壓縮（Path Compression）在單機環境中可達到接近常數階時間複雜度 O(α(n))(Tarjan, 1975)，本研究在後續章節以拓撲學(Topology)進行數學證明。然而，區塊鏈系統具備分散式系統(Distributed System)的特性，理論上可延伸至分散式模型(Bulk Synchronous Parallel, BSP）。BSP 模型強調「批次同步」（Bulk Synchronous）與「超步」（Superstep）概念，為以 Union-Find 為單機執行環境延伸至分散式系統。

本研究初步以單機架構運用 NetworkX 進行 WCC 分群及中心性計算等圖論前處理，後續則可進一步將本地 parent label 結構及聚合流程平行化至 Spark、GraphFrames 等平台，於每個超步進行 pointer-jumping 同步及收斂檢查，實現真正的分散式 Union-Find。每個 Worker 處理局部節點後，透過全域 parent label 同步，直至路徑收斂。

因此，本研究最終之設計理念，是將 Union-Find 的高效運算與 BSP 架構結合，並於聚合過程中納入時間、金額等異質特徵，以增強 AML 分析的解釋性與效能。此架構不僅適合多核心、雲端、大規模資料處理環境，亦符合區塊鏈情境中高頻、多機構交易圖的應用需求。本研究據此設計 UF-FAE 演算法，詳見後續之 Algorithm 3。

2.5 機器學習在 AML 系統的應用

以下為機器學習方法在 AML 系統中的應用簡介表格，包含相關文獻引用：

表 2： 機器學習於之 AML 應用整理表格

方法	簡介與特點	引用來源
邏輯迴歸 (Logistic Regression)	適用二元分類問題，模型結構簡單，高解釋性；透過 Sigmoid 函數進行機率預測，支援正則化（Regularization）避免過度擬合（Overfitting）。	Cox (1958); (Jensen & Iosifidis, 2023; Pan, 2024)
決策樹 (Decision Tree)	模型推論透明度高，適合法規解釋需求；可同時處理數值與類別型特徵，易於建立 baseline 模型。	Quinlan (1986); Martínez-Sánchez et al. (2020)
隨機森林 (Random Forest)	採用大量隨機決策樹集成預測，降低過度擬合風險；具特徵重要性分析能力，容忍異常與缺失值，適用於大規模資料。	Breiman (2001); Raiter (2021); Weber et al. (2019)
支援向量機 (Support Vector Machine, SVM)	在高維及不平衡資料下具高度穩定性與非線性建模能力；適用於風控場景，透過核方法處理複雜特徵，具良好泛化能力。	Cortes & Vapnik (1995); Pambudi et al. (2019)
線性支援向量分類器（Linear Support Vector Classifier, LinearSVC）	SVM 的線性版本，適合處理高維稀疏特徵；執行效率高，具備強化邊界樣本的容錯能力，適合 AML 實務部署需求。	Cortes and Vapnik (1995); (Pambudi et al., 2019)

邏輯迴歸

邏輯迴歸廣泛用於處理二元分類問題。透過 sigmoid 函數將特徵加權總和轉換為介於 0 與 1 之間的機率輸出，進而預測樣本屬於特定類別（如「正常交易」或「可疑交易」）的可能性。Cox (1958)提出二元分類的統計建模框架，但其在機器學習領域中被正式納入監督式學習（Supervised Learning）分類器體系，其主要優點在於其模型結構簡單、訓練效率高，且具備高度可解釋性，特別適用於風控領域，可明確解釋各項特徵對預測結果的邏輯影響。此外，模型可透過最大概似然估計（Maximum Likelihood Estimation）求得最適化參數，並支援不同類型的正則化（如 L1、L2），以避免過度擬合。至今仍活躍於 AML、詐欺偵測、信用評分等金融領域的研究上(Jensen & Iosifidis, 2023; Pan, 2024)。

首先計算特徵變數的加權總和，形成一組線性組合：

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

其中 β_i為模型係數，x_i為輸入特徵。

為了將上述線性輸出映射為機率值（範圍在 0 與 1 之間），使用 Sigmoid 函數進行轉換，其數學定義如下：

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

此函數會將任意實數 z 映射至 (0,1) 區間，詮釋為樣本屬於某一特定類別的機率。

決策樹

決策樹廣泛應用於分類與回歸，其運作邏輯類似於判斷流程圖（Flowchart），模型從根節點(root node)出發，根據資料中的某一特徵進行條件式切分，並沿分支(Branch)遞迴劃分樣本，直到抵達葉節點（Leaf node）並輸出預測結果，以條件遞進為核心的邏輯，使決策樹在模型解釋性上具有優勢，監管部門或政策制定者理解模型如何對交易樣本進行分類判定。(Quinlan, 1986)。

建立過程中需選擇最佳特徵與分裂點，常見評估指標包括：資訊增益（Information Gain）、熵（Entropy）及基尼不純度（Gini Impurity）。決策樹具備下列優勢，使其成為金融領域中的實用建模選擇：

- 高可解釋性，模型推論過程透明，適用於需要符合法規解釋義務的 AML 系統；
- 可同時處理數值與類別型特徵，無需複雜特徵轉換；
- 可應對非線性資料關係與特徵交互影響；
- 適合中小型資料集，能快速建立 Baseline 模型作為風險篩選依據。

由於單棵決策樹在資料龐大或高維特徵情境下，易出現過擬（overfitting）與模型不穩定問題，故實務上常進一步發展為隨機森林或梯度提升樹(GBDT)等集成方法。在 AML 研究上，墨西哥金融機構的實證研究顯示決策樹模型對於資金來源、帳戶活動特徵具高度敏感性，能有效輔助風險評分(Martínez-Sánchez et al., 2020)。

隨機森林

隨機森林屬於「Bagging（Bootstrap Aggregating）」方法的一種。其核心思想為透過建構大量相互獨立、具隨機性的決策樹構成森林，並以投票（分類任務）或平均（迴歸任務)方式進行最終預測，以提升模型的準確性與穩定性，同時降低過度擬合(overfitting)的風險(Breiman, 2001)。

在模型建構過程中，隨機森林會進行以下兩層隨機化處理：

1. **樣本隨機抽樣（Bootstrap Sampling）**：從原始訓練資料中以放回抽樣方式產生多個子樣本集；
2. **特徵子集隨機選取（Random Feature Subspace）**：每棵決策樹在每次節點分裂時，僅從隨機選取的特徵子集中挑選最佳分裂變數，而非考慮全部特徵。

這種雙重隨機機制可使模型具有較低方差與良好泛化能力，同時在處理高維度特徵資料、非線性邊界與不平衡資料上具備優勢。

隨機森林的主要特點包括：

- **高準確性**：透過多棵樹的投票機制，可有效降低單棵決策樹的波動與誤判；
- **特徵重要性衡量能力**：可計算每個特徵對分類決策的貢獻程度，具備一定的可解釋性；
- **對異常值與缺值具容忍性**，且不易受極端樣本影響；
- **易於平行計算與實作**，適用於大規模交易資料環境。

隨機森林在 AML 中已被研究證實具備優異表現。Raiter (2021)針對合成金融交易資料，進行多種監督式學習模型之效能比較，結果顯示在多項指標上均優於其他機器學習方法。對於異常模式與複雜樣本邊界的掌握能力，使其特別適合應用於金融詐欺與洗錢行為之監控。

Weber et al. (2019)則聚焦於 BTC 網絡中的非法交易識別任務，使用 Elliptic 所提供之標註交易圖資料集，將 Random Forest 與 MLP、GCN 等模型進行實證比較。實驗結果顯示，在缺乏圖神經網路時，Random Forest 為最穩定且精準的基線方法，能有效偵測鏈上交易中的潛在可疑資金流向。該研究亦突顯了 Random Forest 在可擴展性與解釋性上的相對優勢，為傳統金融監理部門提供了具備部署潛力的機器學習工具。

隨機森林相對於單棵決策樹較難完整可視化決策過程，但在實務應用中，透過其特徵重要度分析與集體決策機制，仍可在一定程度上滿足金融監理對模型可解釋性的要求。使其成為 AML 與詐欺偵測任務中被廣泛採用的分類演算法之一。

支援向量機

SVM 最早應用於二元分類任務，主要目標為：在特徵空間中尋找一條最優超平面（optimal hyperplane），將資料分為兩類，並使兩側樣本距離此平面最遠，以提升模型對未知樣本的泛化能力(Cortes & Vapnik, 1995)。若資料並非線性可分，SVM 可引入鬆弛變數與懲罰參數 C，構成軟間隔（soft margin）；進一步，透過核技巧（kernel trick）可將資料映射至高維空間，使非線性資料在新空間中可被線性分隔。常見核函數包含多項式

核（polynomial kernel）、高斯徑向基函數核（RBF kernel）等。

SVM 在處理高維資料、特徵數遠大於樣本數（如金融詐欺資料）時具備高度穩定性。其主要優勢包括：

- **分類邊界明確且具理論保證**，適合風控或法遵場景；
- **對少數異常樣本具魯棒性**，在類別極度不平衡的情境下仍能維持判別力；
- **具備強大非線性建模能力**（透過核方法），適合處理複雜交易行為與時序變化特徵。

SVM 在 AML 中雖為經典模型之一，但常因參數選擇敏感性與資料不平衡而表現不穩。為克服此一限制，Pambudi et al. (2019)提出模型優化方法，結合隨機欠抽樣（Random Under Sampling, RUS）與交叉驗證調參（cross-validation based hyperparameter tuning），有效提升了 SVM 在可疑交易分類任務中的整體效能。

SVM 對參數敏感，對於大量樣本資料而言，訓練時間與記憶體需求可能迅速上升。另外其決策邊界不如決策樹或隨機森林直觀可解釋，限制了其在部分監管環境下的使用。

本研究使用 Linear Support Vector Classifier（LinearSVC）作為比較模型之一。SVM 為經典的二元分類演算法，特別適合處理高維稀疏特徵（high-dimensional sparse features）與異常分佈不均的資料，具備良好的泛化能力與抗過度擬合效果。考量到本研究融合了多種圖結構特徵（如中心性、度數）與類別型交易屬性（如支付方式、地點等 One-Hot 向量），資料特徵空間維度較高，LinearSVC 能夠有效對此類特徵進行線性分隔與分類。

此外，LinearSVC 搭配 hinge loss 函數所構成的最大間隔超平面，有助於強化模型對邊界樣本的容錯能力，對於反洗錢（AML）場景中難以明確標定的可疑交易，能提供更具魯棒性的決策邊界。相較於非線性模型，LinearSVC 執行效率較高，適合處理大規模交易資料與實際部署需求。透過與其他分類模型（如 Logistic Regression、Decision Tree、Random Forest）之比較，我們可進一步評估 LinearSVC 在高維交易特徵環境下的可行性與應用潛力。

三、研究方法

3.1 研究架構

本研究的整體架構如圖所示，首先以 SAML-D 原始資料集作為基礎，透過統計分析進行初步的資料分布與異常檢驗，並輔以 K-Means 對金額特徵（Amount）進行聚類，以掌握潛在的交易模式。接著引入 UF-FAE 演算法進行分群：先利用 Union-Find 演算法結合路徑壓縮建立 WCC，再將無向圖轉換為有向圖，計算節點與群組層級的圖論指標。同時，針對原始資料中的類別型與數值型欄位進行特徵工程編碼，以確保其能與圖論特徵共同進入後續的機器學習模型。

在建模階段，本研究設計了三條平行實驗路徑：（1）僅以純圖論指標作為輸入的機器學習；（2）僅以 SAML-D 原始資料作為輸入的機器學習；（3）多模態特徵融合後的機器學習。藉由不同特徵組合的比較，能夠驗證 UF-FAE 演算法在揭露洗錢交易模式上的貢獻。最終，所有模型的效能結果將進行消融實驗分析，透過比較 AUC、Precision、Recall 與 F1-score 等指標，評估多模態融合與單模態特徵的相對優劣，並據此得出演算法效能的結論。

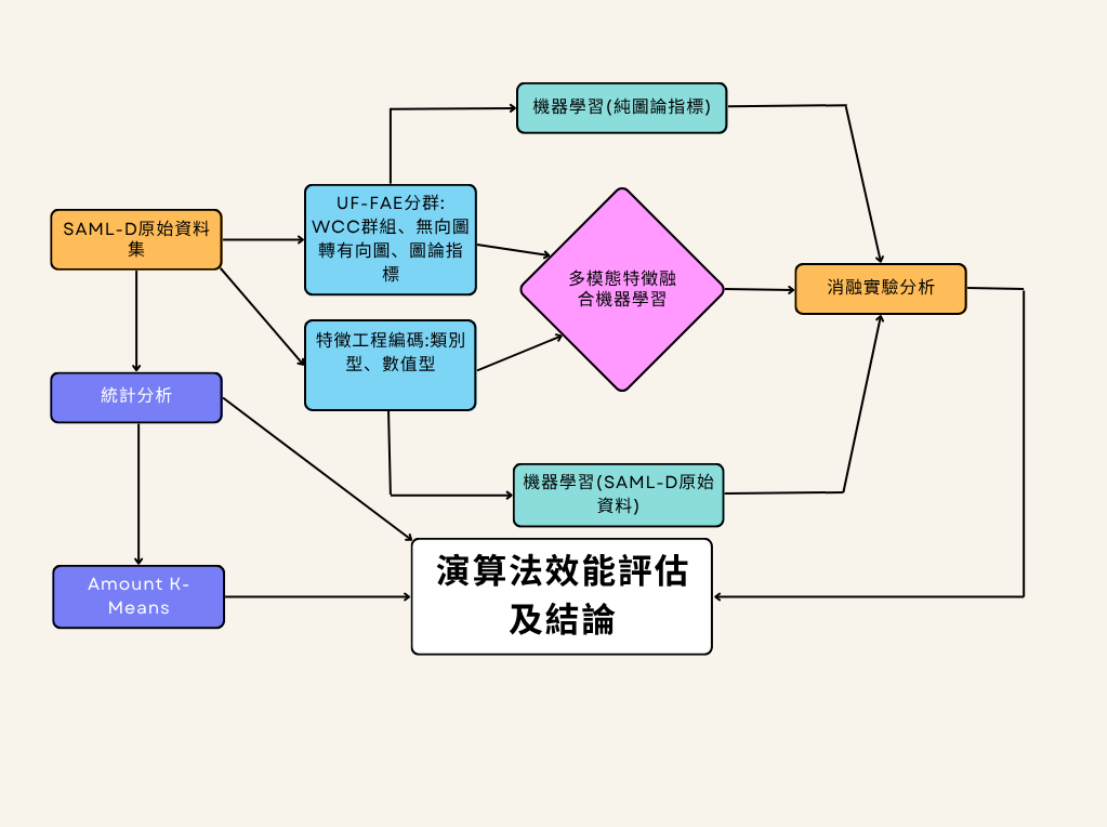


圖 2： 研究架構

為驗證本研究提出之 UF-FAE（Union-Find-based Feature-Augmented Embedding）演算法中各組成要素對整體模型效能的影響，特別設計消融實驗。消融實驗透過逐步移除演算法中的特定特徵或模組，觀察其對洗錢交易偵測準確度的改變，以評估各部分的貢獻度與必要性。

如演算法三所示，UF-FAE 演算法首先利用 Union-Find 搭配路徑壓縮將交易圖切分為多個 WCC，藉此降低大規模交易網絡的計算複雜度。接著，針對每個子圖萃取多種圖論指標及交易統計特徵，並整合為特徵向量，作為機器學習模型的輸入。

Step.1 圖資料結構建構

從資料集中取出發送方與接收方帳戶，建立有向圖：$\mathrm{G} = (\mathrm{V}, \mathrm{E})$，其中節點集合為 $V = a_1, a_2, \ldots, a_n$ ，邊集合為$\mathrm{E} = (\mathrm{u}, \mathrm{v}) \mid u = \mathrm{Sender},\ \mathrm{v} = \mathrm{Receiver}$，

Step.2 Union-Find 與弱連通子圖分群

接著透過 Union-Find 演算法配合路徑壓縮（path compression）將圖切分為若干 WCC 子圖：

- 每個節點初始化一個 parent；
- 若兩節點有邊相連，則進行集合合併。

將整張有向圖 G 拆成 k 個弱連通子圖 $\mathrm{G_1}, \mathrm{G_2}, \ldots, \mathrm{G_k}$

Step.3 節點分類

從子圖中節點的入度與出度，將節點分類為：$\deg_{\mathrm{in}}(\mathrm{v}) = 0 \Longrightarrow v \in \mathrm{Sender}$以及 $\deg_{\mathrm{out}}(v) = 0 \Longrightarrow v \in \mathrm{Receiver}$ 作為異常模式偵測的基礎。

Step.4 圖論指標計算

對每個子圖 $\mathrm{G_i}$ 計算以下圖論指標：

雙向邊比例 (Bidirectional Edge Ratio) :

$$\mathrm{BidirectRatio}(\mathrm{G_i}) = \frac{|\,(\mathrm{u}, \mathrm{v}) \in E_i \mid (\mathrm{v}, \mathrm{u}) \in E_i\,|}{|\mathrm{E_i}|}$$

度中心性 (Degree Centrality) : 節點 v 的度定義為：

$$\deg(\mathrm{v}) = |\, u\colon (\mathrm{u}, \mathrm{v}) \in E \mid + w\colon (\mathrm{v}, \mathrm{w}) \in E$$

接近中心性 (Closeness Centrality) :將子圖視為無向圖，避免方向性影響連通性，定義為：

$$C_{closeness}(v) = \frac{1}{\sum_{u \neq v} d(v,u)}$$

其中$d(v,u)$ 為節點 v 到 u 的最短路徑距離。

中介中心性 (Betweenness Centrality):

$$C_{betweenness}(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

Step.5 子圖結構統計

針對每個子圖 G_i 計算結構統計特徵：

- 節點數 $|V_i|$
- 邊數 $|E_i|$
- 雙向邊比例

3.2 資料蒐集

本研究採用之資料集為 Kaggle 平台提供之 Synthetic Transaction Monitoring Dataset for AML (SAML-D)，為基於 AML 之模擬資料，具備真實交易樣態之統計結構，適用於金融詐欺與異常偵測任務。本資料集以多銀行、多客戶、多樣交易管道為架構，涵蓋高頻微額轉帳、跨國轉帳與混淆資金來源等情境，符合本研究設計之高維、動態、跨域金流圖模型需求。此資料集包含 12 個特徵和 28 種類型（分為 11 種正常類型和 17 種可疑類型）。這些特徵是根據現有資料集、學術文獻以及對反洗錢專家的訪談篩選出來的。資料集包含 9,504,852 筆交易，0.1039% 為洗錢交易，共 9873 筆。其中，Is_laundering 欄位為監督式學習所需之標籤（label），標示該筆交易是否與洗錢行為有關(Oztas et al., 2023)。原始資料以 CSV 格式儲存，無缺漏值與重複紀錄，屬品質完整之開放資料。每筆交易包含以下欄位：

表 3 : SAML-D 資料集原始欄位列表及說明

欄位名稱	說明
Time	交易時間戳記
Date	交易發生之年月日
Sender_account	發款帳戶編號
Receiver_account	收款帳戶編號
Amount	轉帳金額
Payment_currency	發款幣別
Received_currency	收款幣別
Sender_bank_location	發款方銀行所屬地區
Receiver_bank_location	收款方銀行所屬地區
Payment_type	支付類型（例：現金、線上、加密貨幣等）
Laundering_type	模擬洗錢手法類型（若屬可疑交易）
Is_laundering	是否為可疑洗錢交易（1 表示可疑，0 表示正常）

3.3 統計分析

本研究先行以 Python3.10 之 polars、numpy、pandas、scipy 套件進行統計分析，並以 matplotlib 做資料視覺化，下圖呈現正常交易與洗錢交易之統計分布圖:

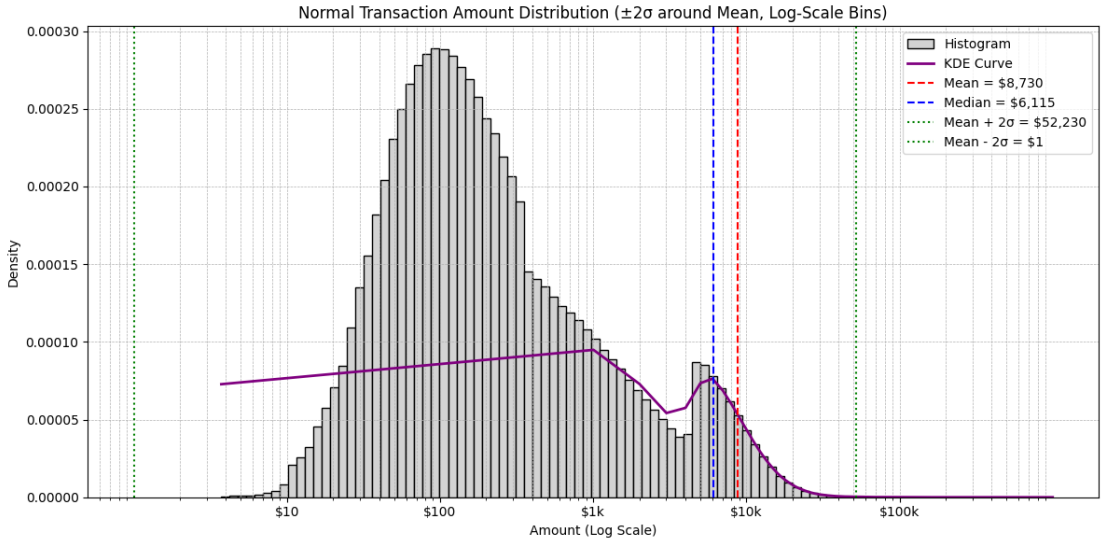


圖 3 : SAML-D 之交易金額統計分布圖

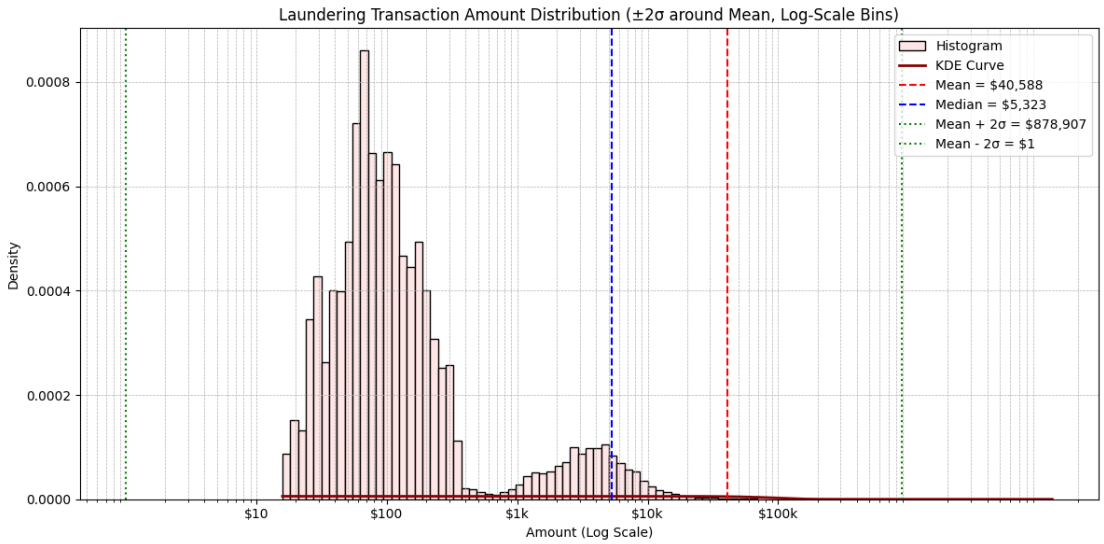


圖 4: SAML-D 之洗錢標註之交易金額統計分布圖

3.3.1 洗錢類型描述性統計 ：

表 4： 洗錢之支付類型筆數與百分比

支付類型	交易筆數	百分比
Cross-border	2628	26.62%
Cash Deposit	1405	14.23%
Cash Withdrawal	1334	13.51%
ACH	1159	11.74%
Credit card	1136	11.51%
Debit card	1124	11.38%
Cheque	1087	11.01%

表 5： 跨國洗錢與跨幣種洗錢交易筆數

項目	數值
總交易筆數	9,873
跨幣種交易筆數	3,657

項目	數值
跨幣種交易比例	37.04%
跨國洗錢交易筆數	3,045
跨國洗錢平均金額	$18,978.18

表 6： 信用卡洗錢相關統計表

項目	數值
信用卡洗錢交易筆數	1,136
信用卡洗錢平均金額	$53,666.89
信用卡交易總筆數	1,136
跨幣別的信用卡交易筆數	138
跨幣別的信用卡交易比例	12.15%

表 7 : ACH 洗錢交易比數統計表

項目	數值
ACH 洗錢交易筆數	1,159
ACH 洗錢平均金額	$64,623.30
ACH 洗錢交易總筆數	1,159
跨幣別的 ACH 洗錢交易筆數	168
跨幣別的 ACH 洗錢交易比例	14.50%

3.3.2 統計檢定

本研究針對 SAML-D 資料集進行顯著性檢定，結果顯示「交易方式」與「是否為洗錢交易」具有顯著關聯（$\chi^2 = 13,831.72, p < .001$），而洗錢交易的金額分布也與非洗錢交易有顯著差異（$T = -7.552, p < .001$；$U = 4.84e+10, p < .001$）。此外，不同付款類型之間的交易金額也存在顯著差異（$ANOVA\ F = 8902.77, p < .001$），指出付款方式在洗錢行為中可能扮演關鍵角色。

3.3.3 Amount K-Means 分析

因正常與洗錢交易間具有顯著差異，本研究後續使用本研究使用 scikit-learn 之 K-Means 分群演算法，針對金額進行無監督學習，嘗試發掘在數值分佈上可能存在的結構。考量到原始金額分布具有較大變異與偏態，因此在建模前先使用 StandardScaler 將金額資料標準化處理，以避免群聚過程中受到極端值影響。在選定適當群數方面，採用肘部法（Elbow Method）作為指標。透過計算不同群數（k=1 至 k=10）下的群內平方誤差總和（Sum of Squared Errors, SSE），並繪製出對應的趨勢圖，可觀察 SSE 隨著群數增加而遞減的變化情形。理論上，當群數過少時，模型難以捕捉異質結構，導致誤差過高；而群數過多則可能產生過度擬合，因此應尋找 SSE 曲線開始明顯趨緩的「肘部」位置作為最佳群數。

從圖 5 所示結果可見 SSE 在 k = 3 處出現明顯轉折，顯示此時模型已能有效分群，而再增加群數所帶來的 SSE 改善幅度趨於平緩。故本研究使用分為三群進行後續分析。

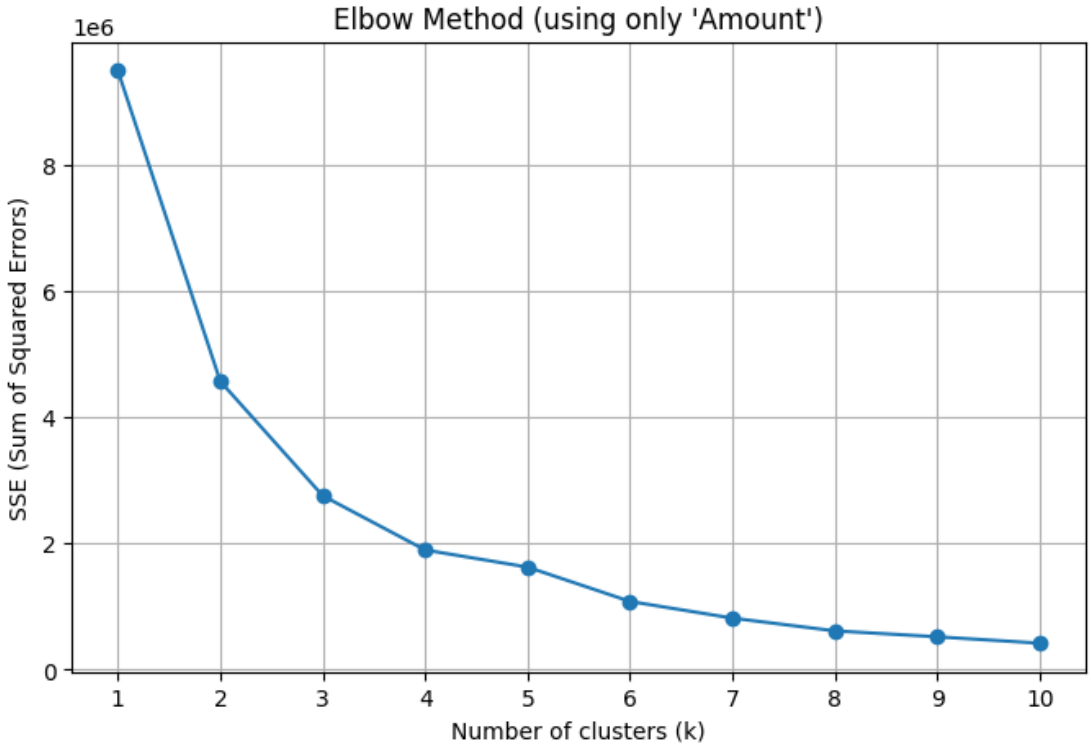


圖 5： 肘部法 SSE 曲線圖

基於先前肘部法分析結果顯示最佳群數落在 k=3，使用 K-Means 演算法對標準化後的交易金額 (Amount) 進行分群。經分群後，系統將所有交易劃分為三個具代表性的金額群體，分別代表低額、中額與高額的交易行為樣態。

為更具體觀察這三群在金額上的差異，本研究繪製各群交易金額的箱型圖（Boxplot），如圖 6 所示。從視覺化結果可觀察到三個群體的中位數、四分位距（IQR）、以及極端值的分布情況具有明顯差異，顯示模型能有效區分不同金額區間的交易類型。

其中，**Cluster 0** 顯著集中於低金額範圍，可能對應日常正常交易；**Cluster 1** 則分布於中間金額帶，代表中度金額活動；而 **Cluster 2** 呈現較大變異與高額異常值，可能潛藏不尋常的高金額洗錢行為。該群的極端上限（whiskers）明顯偏高，值得進一步搭配 Is_laundering 標籤驗證其潛在風險。

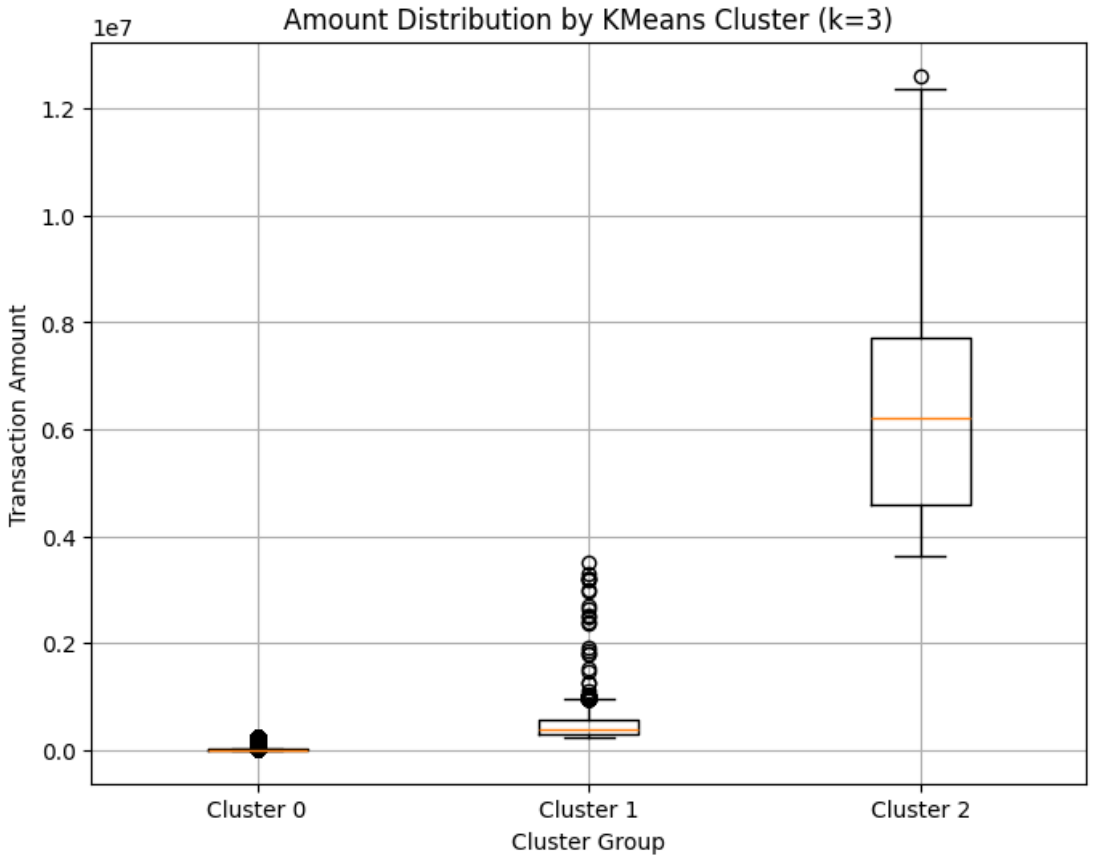


圖 6: 分群之交易金額箱型圖

為初步了解各群體在金額上的統計特徵，本文針對三個群體進行金額欄位的摘要統計，包括交易筆數（Count）、平均值（Mean）、標準差（Std）、最小值（Min）、中位數（Median）與最大值（Max），以描繪各群在金額分佈上的整體輪廓。如表 8 所示，三個群體在統計指標上存在顯著差異，顯示 K-Means 演算法成功捕捉到金額分佈中的潛在異質性：

- **Cluster 0** 為低金額群，交易金額集中於相對較小區間，標準差亦較低，可能對應日常小額交易活動。
- **Cluster 1** 呈現中間程度金額，具有適中變異性，可能為企業或跨國轉帳類型。

- **Cluster 2** 則包含金額明顯較高的交易，最大值遠高於其他群體，並呈現較大標準差，暗示潛在的高風險交易行為或異常活動。

本階段結果除可作為非監督式學習（Unsupervised Learning）異常偵測的基礎，也提供後續交叉分析（如各群的洗錢比率）與視覺化探索的資料依據。分析結果已另存為含 Cluster 標籤的資料表，便於後續操作與報告產出(Deprez et al., 2025)。

表 8： 分群之描述性統計

Cluster	筆數 (Count)	平均金額 (Mean)	標準差 (Std)	最小值 (Min)	中位數 (Median)	最大值 (Max)
0	9,489,333	8,017.20	10,010.69	3.73	6,103.01	230,139.97
1	15,488	452,371.96	205,848.39	230,155.77	393,558.85	3,507,884.10
2	31	6,661,100.00	2,608,700.00	3,634,000.00	6,213,200.00	12,618,000.00

Cluster 0 佔比極高（超過 99.8%），代表主要為小額與中額交易，且標準差也屬溫和，為日常型態。
Cluster 1 為高金額交易，筆數雖少但平均金額逼近 50 萬，可能為企業或異常轉帳。
Cluster 2 為極高金額群，筆數極少（僅 31 筆），但均值超過 **660 萬元**，最大值高達 **1,261 萬元**，可能為洗錢行為集中地。

為進一步分析於非監督式學習分群結果中所識別的潛在高風險交易樣態，本文針對 K-Means 分群中金額最顯著偏高的群體（即 Cluster 2）進行聚焦式探勘。該群僅佔全體交易筆數之極少比例（31 筆），但平均金額高達 660 萬元，最大值更超過 1,261 萬元，遠高於其他群體，具有明顯異常特徵。從 Cluster 2 中抽取關鍵交易欄位進行剖析，包括交易日期與時間（Date, Time）、帳戶來源與接收者（Sender_account, Receiver_account）、金額（Amount）、支付方式（Payment_type）及雙方地理位置（Sender_bank_location, Receiver_bank_location），作為風險交易清單的核心欄位（表 9）。

表 9 : Cluster 2 之洗錢交易資料

Date	Time	Sender_account	Receiver_account	Amount	Payment_type	Sender_location	Receiver_location
2022-10-12	16:12:51	5057689301	3267846600	6,213,931.56	Credit card	UK	UK
2022-11-05	23:51:26	4197215987	7510608716	5,971,168.57	Debit card	UK	UK

Date	Time	Sender_account	Receiver_account	Amount	Payment_type	Sender_location	Receiver_location
2022-11-09	17:39:28	6498511324	3990508657	4,240,667.45	Debit card	UK	UK
2022-11-13	17:39:28	8561558553	6498511324	9,348,125.28	Debit card	UK	UK
2022-11-23	09:46:18	5061861850	520713149	4,198,253.76	ACH	UK	UK
2022-11-29	09:46:18	1534642148	5061861850	11,837,365.32	ACH	UK	UK
2022-11-30	18:10:29	2149901234	5804990951	4,692,191.66	Debit card	UK	UK
2022-12-21	08:40:55	9126482714	254333209	4,739,084.71	Debit card	UK	UK
2022-12-28	08:40:55	1075524474	9126482714	9,939,367.70	Debit card	UK	UK
2023-01-03	04:21:55	2887411233	7219174047	4,509,020.91	Cheque	UK	UK
2023-01-05	20:05:23	2258780243	4271610216	6,475,296.65	Cross-border	Nigeria	UK
2023-01-18	17:20:37	4216161511	8808180090	6,687,866.73	Credit card	UK	UK
2023-02-07	13:47:09	6684984541	5444185928	3,722,429.57	Cheque	UK	UK
2023-02-17	10:56:31	1794587906	7976247492	7,208,996.55	ACH	UK	UK
2023-03-02	11:13:22	7699316196	717432649	3,633,963.35	Cross-border	Switzerland	UK
2023-03-09	11:13:22	2636186377	7699316196	6,812,158.18	Cross-border	UK	Switzerland
2023-03-31	14:02:39	8422910317	6451219507	4,355,569.17	Debit card	UK	UK
2023-04-06	14:02:39	6051338057	8422910317	7,982,973.18	Debit card	UK	UK

Date	Time	Sender_account	Receiver_account	Amount	Payment_type	Sender_location	Receiver_location
2023-04-29	11:14:02	9858304544	1874802060	5,611,048.00	Credit card	UK	UK
2023-05-03	13:45:43	717880712	8058727145	4,435,401.96	ACH	UK	UK
2023-05-07	13:45:43	7113150348	717880712	12,618,498.40	ACH	UK	UK
2023-05-16	23:53:22	7283781763	3486140760	4,075,453.13	Debit card	UK	UK
2023-05-22	23:53:22	976249309	7283781763	7,471,377.04	Debit card	UK	UK
2023-06-02	08:45:45	5929867889	1885068430	4,720,982.29	Cheque	UK	UK
2023-06-03	16:34:53	9098263330	6832796065	5,881,588.23	Debit card	UK	UK
2023-06-09	08:45:45	5649129207	5929867889	10,267,422.68	Cheque	UK	UK
2023-06-25	13:11:57	3106127807	2358122027	6,347,718.67	Cheque	UK	UK
2023-07-01	19:39:41	9911799503	648937549	4,709,139.53	Cheque	UK	UK
2023-07-05	19:39:41	2309271621	9911799503	12,358,785.74	Cheque	UK	UK
2023-08-13	09:47:53	6674085239	5665240975	6,213,230.54	Credit card	UK	UK
2023-08-20	06:55:37	4917020979	1285232494	9,216,360.49	Debit card	UK	UK

進一步交叉驗證標記資料發現，**該 31 筆 Cluster 2 交易全數標記為洗錢行為**（Is_laundering = 1），顯示該群不僅在金額上明顯異常，亦具有極高的風險性質。此結果驗證了非監督式學習金額分群可作為潛在洗錢行為的前置篩選機制。

表 10 : Cluster 2 之金額交易資料統計(全被標註為洗錢)

指標	數值
交易筆數	31
平均金額	$6,661,285.68
最大金額	$12,618,498.40
最小金額	$3,633,963.35
總交易金額	$206,499,856.10
跨國交易筆數	3（9.7%）
常見支付方式前 3 名	Debit card（11）、Cheque（9）、ACH（6）

3.4 資料預處理

隨著區塊鏈之公有鏈之交易資料量成長快速，如何有效進行資料預處理已成為未來演算法設計與機器學習模型表現的關鍵之一。本研究採用基於圖論演算法的分群與特徵融合流程進行資料預處理，具體步驟如下：

首先利用 Polars 高效載入大規模交易資料，並藉由 Networkx 建立有向圖結構，將每筆交易的 Sender 與 Receiver 視為圖中的一條有向邊。接著，根據圖的 WCC 進行分群，每個群組對應一個局部金流社群，這有助於分辨多組彼此獨立的金流流向，並降低資料維度，提升分析效率。

為強化後續模型之判別能力，本研究更進一步於每個分群內萃取多種圖結構性指標作為新增特徵，包括：

- **群組節點數/邊數（node/edge count）**：反映局部網路規模。
- **雙向連結比例（bidirect ratio）**：評估群組內交易的互動性與複雜度。
- **中心性指標（closeness、betweenness）**：計算 Sender/Receiver 於其所屬群組中的近接中心性與介數中心性，幫助識別網路中的潛在樞紐與異常節點。

本研究認為此類特徵可顯著提升區塊鏈詐欺偵測與資金流追蹤等下游任務的模型效能。考慮到圖結構在區塊鏈應用中的重要性，研究中僅針對節點數超過三者之群組計算中心性，避免小規模群體數值偏離。最終，將所有圖指標融合回原始交易資料，並以 parquet 格式儲存，確保下游大數據機器學習流程之高效運算與可重複性。

3.4.1 Union-Find

Union-Find（Disjoint Set Union，DSU）是一種經典的資料結構，用於維護元素集合之間的連通關係。最早可追溯至 Galler and Fisher (1964)的集合合併演算法，後由 Tarjan (1975)提出 path compression 與 union-by-rank 的優化策略，並證明其攤銷時間複雜度 $O(\alpha(n))$可接近至常數階，其中 $\alpha(n)$ 為反 Ackermann 函數。後續多項研究(Cormen et al., 2022; Hopcroft & Ullman, 1979; Sedgewick & Wayne, 2011)皆指出此演算法在實務應用中幾乎等同常數時間，為處理動態連通性問題（Dynamic Connectivity Problem）的核心方法。

Union-Find 主要由兩個基本操作構成：

1. **Find(x)**：尋找元素 x 所屬集合的代表元（root 或 parent），藉此判斷兩元素是否位於同一群組。
2. **Union(x, y)**：將分屬不同集合的兩元素合併，使其代表元指向同一節點。

為提升效能，Union-Find 常搭配兩項結構優化：

- **路徑壓縮（Path Compression）**：在執行 Find 時，將沿途節點直接連至根節點，以降低後續查找深度(Tarjan, 1975)。
- **秩合併（Union by Rank）**：合併時令秩（樹高或節點數）較小者掛於較大者之下，避免生成過深樹(Cormen et al., 2022)。

結合這兩項策略，Union-Find 在任意 m 次 find 與 union 操作的總時間為 $O(m\,\alpha(n))$ (Tarjan, 1975)，在實際應用中 $\alpha(n) \leq 5$，可視為常數時間。此特性使 Union-Find 被廣泛應用於圖論演算法、社會網絡分析與大規模資料分群(Cormen et al., 2022; Holm et al., 2001)，特別適用於處理高頻率的集合合併與查詢操作。

在本研究中，Union-Find 被用作 UF-FAE 架構之**分群核心演算法**。藉由持續觀測交易資料中 Sender 與 Receiver 之間的連結關係，系統可動態合併具關聯性的帳戶節點，形成可演化的金流群組。當新交易邊導致群組合併時，該事件即視為潛在可疑行為，並觸發後續的圖論特徵擷取與異常偵測流程。此方法能在維持近常數運算效率的同時，有效捕捉區塊鏈交易網絡的結構變化。

3.4.2 使用拓撲學證明 Disjoint 與 Union-Find 等價

因需要解釋並證明不相交集的資料結構與不相交集的劃分及其與併查集演算法的連結，且本質上併查集藉由執行併運算來維護等價關係符合拓撲原理(Topology principles)，本研究將涉及定義等價關係、建立商空間以及證明分割與併查集樹結構上，Disjoint 和 Union-Find 為相等的。

- 目標：證明 Disjoint Sets = Union-Find = π_0（路徑連通分支）
- 觀點：電腦科學 + 拓撲學
- 核心概念：Union-Find 動態維護圖的 π_0（1 維單純形複形）。

定義一：父指標函數與森林

- 設 $X = x_1, x_2 \ldots x_n$
- 父指標函數：$p(x) = x$ if x is root; else $p(x) = \text{parent}(x)$
- 邊集合：$E = \{(x, p(x)) | x \in X, p(x) \neq x\}$
- 森林結構：$G = (X, E)$，每棵樹對應一個不相交集合。

$$p(x) = \begin{cases} x, & if\ x\ is\ root \\ p(parent(x)), & otherwise \end{cases}$$

```
def find(x):
    if parent[x] != x:
        parent[x] = find(parent[x])     # <== 對應 p(parent(x))
    return parent[x]
```

p(x) 理論上是一個函數映射。

find(x) 是它的實作函數，並在過程中執行 **path compression**。

定義二：拓撲建模

- 從森林 G 建立 1 維單純形複形 K。
- 定義路徑連通關係：
- $x \sim y \Leftrightarrow \exists (x = v_0, v_1, \ldots, v_k = y)$，使得$(v_i, v_{i+1}) \in E(K), \quad \forall i$。

- $\pi^{0(K)} = X/\sim$：K 的連通分支集合。

該關係在離散數學上具自反性、對稱性與傳遞性，因此是一個**等價關係**。其等價類集合記作：

$$\pi^{0(K)} = X/\sim \ ，$$

即為拓撲空間 K 的「路徑連通分支」（path-connected components）。

因為每次執行 find 時都讓 parent[x] = root，使得實際的 p(x) 變成幾乎恆定映射（幾乎所有節點直接指到根）。

在拓撲學詮釋裡，就是：

Path compression = homotopy collapse
⇒ 空間被壓平成 π_0 等價類，時間複雜度降到 $O(\alpha(m,n))$。

定義三：Find 與 Union 操作

- $\mathrm{Find}(x)$：若 $p(x) = x$ 則回傳 x；否則 $\mathrm{Find}(p(x))$。
- $\mathrm{Union}(x,y)$：$r_x = \mathrm{Find}(x)$，$r_y = \mathrm{Find}(y)$
- 若 $r_x \neq r_y$，則將秩較小者指向秩較大者。
- PathCompression：$p(x) \leftarrow \mathrm{Find}(p(x))$。

引理一 Find 與路徑連通等價

- $\mathrm{Find}(x) = \mathrm{Find}(y) \Leftrightarrow x \sim y$
- ⇒ 同根節點 ⇒ 存在路徑 ⇒ 連通。
- ⇐ 經 Union 建立的邊 ⇒ 同屬一棵樹 ⇒ 同根節點。

引理二 Path Compression 不改變 π_0

- 壓縮後：$\pi^{0(K')} = \pi^{0(K)}$

原因是：樹內的邊收縮不會改變連通關係（樹為可收縮空間）

引理三 Union 操作的拓撲對應

- 若兩個根節點 r_1, r_2 連接：
- $\pi^{0(K_{after})} = \pi^{0(K_{before})} - 1$
- 理由：新增邊(r_1, r_2)合併兩個連通分支，其他分支未受影響，因此 π_0 減少 1。

定理 — 拓撲等價性

- 對任意時間 t：$\mathcal{P}_{uf}(t) \cong \pi^{0(K_t)}$
- 映射 $\Phi_t(S_i) = C_i$，其中 C_i 為 K_t 的第 i 個連通分支。
- 引理 1：UF 分割 = 路徑連通分割。
- 引理 2：Path Compression 保留 π_0。
- 引理 3：Union 使 π_0 減一。

推論 — 動態連通性詮釋

- Find → 查詢節點的連通分支。
- Union → 合併兩個分支。
- Path Compression → 樹內邊收縮操作。
- ⇒ Union-Find = 動態維護 $\pi_0(K)$ 的資料結構。

結論-由拓撲等價導出三層 DAG 結構

由上可知，若證明 **Disjoint Set 所維護的集合分割**與 **Union-Find 資料結構所維護的集合分割** 在數學上等價，即 $P_{Disjoint} = P_{Union-Find}$則兩者在執行所有集合操作（Find、Union）的過程中皆維持相同的集合演化與分割狀態。因此，**Disjoint Set 的時間與空間複雜度分析結果**可直接套用於 **Union-Find**，換言之：

$$T_{Disjoint}(n) = T_{Union\text{-}Find}(n) = O(\alpha(n)),$$

- Union–Find 分割 = $\pi_0(K)$
- Union-Find 以 $O(\alpha(n))$ 維護動態連通性

在任意時間 t，$P_{UF}(t) \cong \pi_0(K_t)$ 成立，則 Union–Find 不只是資料結構，而是 π_0 的離散實作。這一同構在不同操作層次上會生成三個對應的 DAG：

1. 原始因果 DAG — 由時間序列與方向邊 (sender → receiver) 構成，
   為拓撲空間 K_0 的實體表示。
2. Rank-Filtration DAG G_{UF} — 由 Union 操作的秩序序列產生：
   每次 Union 使 π_0 -1，形成:

$$K_0 \subset K_1 \subset \cdots \subset K_t, \quad (u,v) \in E_{rank} \Leftrightarrow \text{Union}(u,v) \text{ with } r(u) \geq r(v)$$

因 rank 只增不減，G_{UF} 必為 DAG。

3. 同倫商 DAG G'— 將群內強連通成分 (SCC) 在同倫意義下收縮為單點，
   得到商空間 $K' = K/\simeq$其邊集

$$E' = \{(C_i, C_j) \mid \exists (u,v) \in E,\ u \in C_i, v \in C_j\}$$

   若 $H_1(G') = 0$，則 G' 為同倫有向無環商(Directed Acyclic Quotient up to Homotopy, DAQH)，代表群際層的偏序拓撲。

因此，3.4.2 中的拓撲證明不僅說明 Disjoint 與 Union-Find 在結構上等價，也揭示了 π_0 的三種表現形式：

層級	對應 DAG	拓撲對應	描述
靜態層	原始 DAG	K_0	實際交易／因果流
演化層	Rank-Filtration	K_t	結構生成過程
同倫層	DAQH	$K' = K/\simeq$	穩定群際拓撲

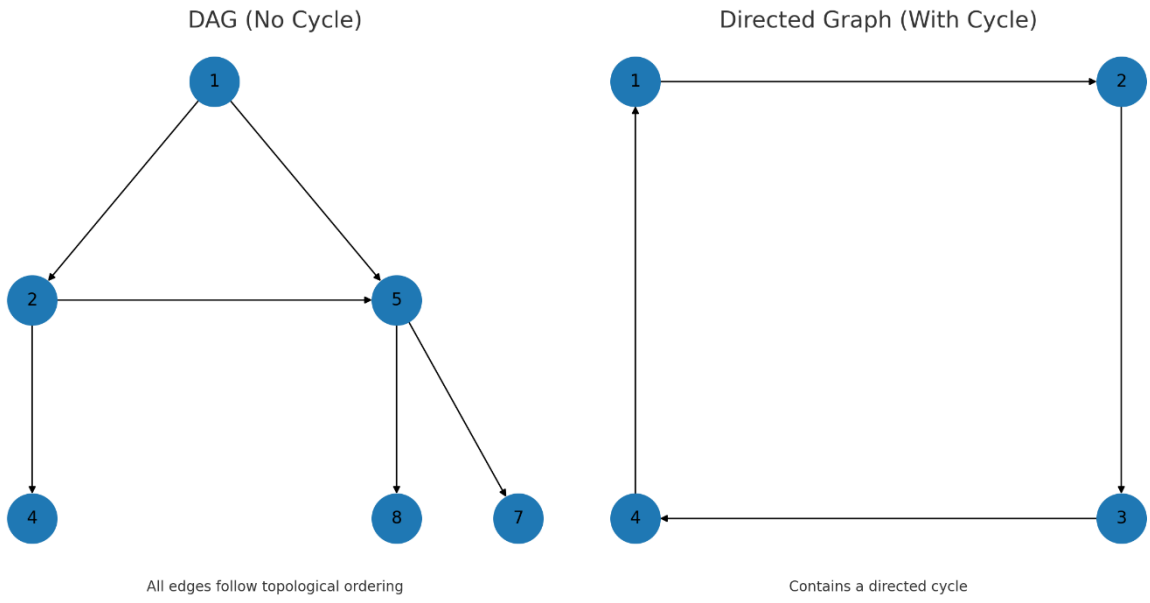
```
def find(x):
    if parent[x] != x:
        parent[x] = find(parent[x])
    return parent[x]

def union(x, y):
    root_x = find(x)
    root_y = find(y)
```

```
    if root_x != root_y:          # 關鍵條件
        parent[root_x] = root_y
```

條件 if root_x != root_y 就是反環保證：

- 若兩個節點已屬於同一集合，find() 回傳相同 root → 不再合併。
- 若不同集合，合併只在根與根之間進行 → 無法生成自迴圈。



Union–Find 資料結構中，部分操作序列中樹的高度可能暫時超過$O(\log n)$複雜度。但因 **find(x) 操作結合「路徑壓縮（path compression）」** 會在之後的操作中自動將樹高度壓縮至近乎常數。具體來說，每當執行 find(x) 時，演算法會遞迴尋找根節點並同時讓沿途所有節點直接指向該根：

$$\text{parent}[v] = \text{root}(v)$$

使其平均成本維持在$O(\alpha(m,n))$, Union–Find 的森林結構永遠不可能形成環，理由如下：

1. **唯一路徑性**：每個節點恰有一個父節點，且僅有根節點自指。
2. **不交併（Disjointness）**：union(x, y) 只在兩個不同集合（不同根）之間建立指向，因此新邊必然連接兩棵不相交的樹，絕不可能回指形成閉環。
3. **路徑壓縮只縮短，不新增邊**：find() 只會改寫既有邊為「指向根節點」，這相當於拓撲上的邊收縮（Edge contraction），而非新增邊，因此同樣不可能產生有向環。

3.4.3 一維特徵向量的構成

將輸入特徵整合為一維向量 ：

$$X = [x1, x2, \dots, xn]$$

其中 X 由兩個部分所串接 ：

$$\mathrm{X} = \left[\mathrm{X}_{\mathrm{num}} \parallel \mathrm{X}_{\mathrm{cat}}^{(\mathrm{OHE})}\right]$$

X_{num} ： 數值特徵

$X_{cat}^{(OHE)}$ ： 對類別特徵經 One-Hot 編碼後的向量

(A) 數值特徵

1. 交易金額：Amount
2. 圖論群組統計：group_node_count、group_edge_count、group_bidirect_ratio
3. 發款/收款節點中心性：sender_degree、receiver_degree、sender_closeness、receiver_closeness、sender_betweenness、receiver_betweenness

註：Date/Time 僅用於**時間序切分**形成 timestamp，不納入模型特徵，避免時間洩漏。

(B) 類別特徵（OHE 編碼後）

- 幣別：Payment_currency、Received_currency
- 地點：Sender_bank_location、Receiver_bank_location
- 支付方式：Payment_type

標籤變數為 ：

$$\mathrm{y} \in \{0,1\}$$

註：是否為洗錢交易，1 為洗錢交易，對應欄位 `Is_laundering` 。

3.4.4 類別型特徵編碼流程（Algorithm 2）

String Indexer :

將類別型變數 $C \in c_1, c_2, \ldots, c_k$ 映射為整數索引，對應關係為：

$$Index(ci) = i \; for \; i \in 0,1, \ldots, k-1$$

例如：UK → 0、UAE → 1、China → 2。

One-Hot Encoder :

將整數索引轉換為獨熱向量（One-Hot Encoding）形式：

$$\text{OneHot(i)} = [0, \ldots, 0, 1, 0, \ldots, 0]$$

其中第 i 位為 1，其餘位元為 0。

例如當 $UK_{idx} = 2$ 時，One-Hot 向量為 ：

$$[0,0,1,0, \ldots, 0]$$

Vector Assembling :

將 Xnum與所有 One-Hot 向量拼接，得到最終輸入向量 X。

Algorithm 2 Categorical Feature Encoding

Require: Categorical variable $C \in \{c_1, c_2, \ldots, c_k\}$
Ensure: Numerical feature vector $\mathbf{x}$

1: **StringIndexer:**
2: Map each category c_i to integer index i where $i = 0, 1, \ldots, k-1$.
3: Example: UK $\rightarrow$ 0, UAE $\rightarrow$ 1, China $\rightarrow$ 2.
4: **OneHotEncoder:**
5: Convert integer index i to one-hot vector:

$$\text{OneHot}(i) = [0, \ldots, 0, 1, 0, \ldots, 0]$$

where the i-th position is 1 and all others are 0.
6: Example: if $UK_{idx} = 2$, then $\text{OneHot}(2) = [0, 0, 1, 0, \ldots, 0]$.
7: **VectorAssembler:**
8: Concatenate all numerical features and one-hot vectors into a single feature vector:

$$\mathbf{x} = [x_1, x_2, \ldots, x_n].$$

9: **return** $\mathbf{x}$.

演算法二: Categorical Feature Encoding Algorithm 流程

3.4.5 機器學習模型與損失函數

本研究以四種經典監督式分類模型建構於前述一維特徵向量 X 上，對應之目標函數（損失）如下。

Logistic Regression

以 sigmoid 將線性組合映射為機率 $\hat{p}_i$ ，採用二元交叉熵損失：

$$\hat{p}_i = \sigma(\mathrm{w}^{\mathrm{T}}\mathrm{x} + \mathrm{b}) = \frac{1}{1 + \mathrm{e}^{-(\mathrm{w}^{\mathrm{T}}\mathrm{x}+\mathrm{b})}}$$

採用二元交叉熵損失 ：

$$\mathrm{L}_{CE} = -\frac{1}{N} = \sum_{i=1}^{N} [y_i \log \hat{p}_i + (1 - y_i)\log(1 - \hat{p}_i)].$$

Linear SVM（線性支援向量機）

以最大化間隔為目標：

$$L_{SVM} = \left[\frac{1}{2}\| \mathrm{w} \|^2 + \mathrm{C}\sum_{\mathrm{i}=1}^{\mathrm{n}} \max\left(0,\ 1 - \mathrm{y_i}(\mathrm{w}^{\mathrm{T}}\mathrm{x_i} + \mathrm{b})\right)\right]$$

採用 Hinge Loss 之原始型式

$$\mathrm{HingeLoss}\big(y, f(\mathbf{x})\big) = \max\big(0, 1 - yf(\mathbf{x})\big)$$

Decision Tree（決策樹）/ Random Forest（隨機森林）

以**節點不純度**作為分裂準則

$$\mathrm{H(S)} = -\sum_{\mathrm{i}=1}^{\mathrm{k}} p_i \log_2 p_i$$

基尼指數：

$$\text{Gini}(S) = 1 - \sum_{i=1}^{k} p_i^2$$

其中：

- k：分類問題的總類別數。
- p_i：樣本屬於第 i 的機率（比例）。

在本研究的二元分類任務（是否為洗錢交易，Is_laundering ∈ {0,1}）中，k=2：

- p_0：屬於正常交易的比例。
- p_1：屬於洗錢交易的比例。

因此，熵與基尼值分別衡量一個資料子集在二分類下的不純度，用於決定決策樹的最佳分裂點。

3.5 小世界交易網路之分群與節點角色分析

為進一步剖析交易網絡中潛在的異常行為，本研究以 Union-Find 對完整交易資料進行弱連通分群處理，將所有帳戶視為節點、交易行為視為邊，利用 SAML-D 之帳戶間的交易關係，由原 WCC 無向圖轉變為建立出有向圖，並進行高效路徑壓縮的並查集運算。最終共劃分出 **15,592 個子圖群組**，每一群代表一個可達的交易圈層。

接著對每個群組統計其內部節點的入度與出度，辨識群內是否存在僅收款或僅付款之節點，以此判定該群之潛在角色分工。初步結果顯示，多數群組中同時存在大量出金節點（Sender）與收金節點（Receiver），顯示資金可能在群內被反覆流動或層層轉移，有典型洗錢「拆分—混合—再合併」之結構性特徵。揭示出 AML 圖譜中的小世界特性與節點間角色異質性，更為後續圖論指標（如中介中心性、雙向交易比例等）提供群組內部的語義基礎，使得本研究所提之多模態特徵融合架構，具備更強的解釋力與擴展性。

為深入探討在交易網絡中是否存在**節點數量龐大的高風險子群組**，本研究進一步統計每個透過 Union-Find 分群後所產生子圖（群組）的節點數，並依節點數進行降冪排序。使用者可依需求輸入欲觀察的「前 N 名（含並列）」群組，進行聚焦分析。

在取得節點數最多的前 N 名群組後，系統會針對每個群組進行洗錢交易數量的統計。此步驟透過將交易發送端帳戶對應至其所屬群組，並比對標註為 Is_laundering = 1

的交易筆數。分析結果呈現每個高節點群組中是否存在洗錢交易，並額外標示「完全沒有洗錢交易」的安全群組數。

此一分析策略有助於發掘潛藏於大規模交易圈中的可疑行為，並聚焦有限資源於高風險群組進行後續監控與建模。根據實驗結果顯示，節點數前 3% 的群組(含並列節點數名次，共 465 群)中皆出現洗錢交易，共 6236 筆，覆蓋所有洗錢交易 63.12%，發現顯示節點密度高的網絡子圖中，潛藏違規行為的機率顯著提升，值得金融單位優先關注與介入監控。

此方法亦具備良好彈性與擴展性，可作為多模態 AML 系統中的「初階風險熱點篩選器」，協助決策者自海量資料中快速定位潛在威脅區域，進一步接軌機器學習或深度學習模型進行精準預測，範例圖以及 Algorithm1 如下:

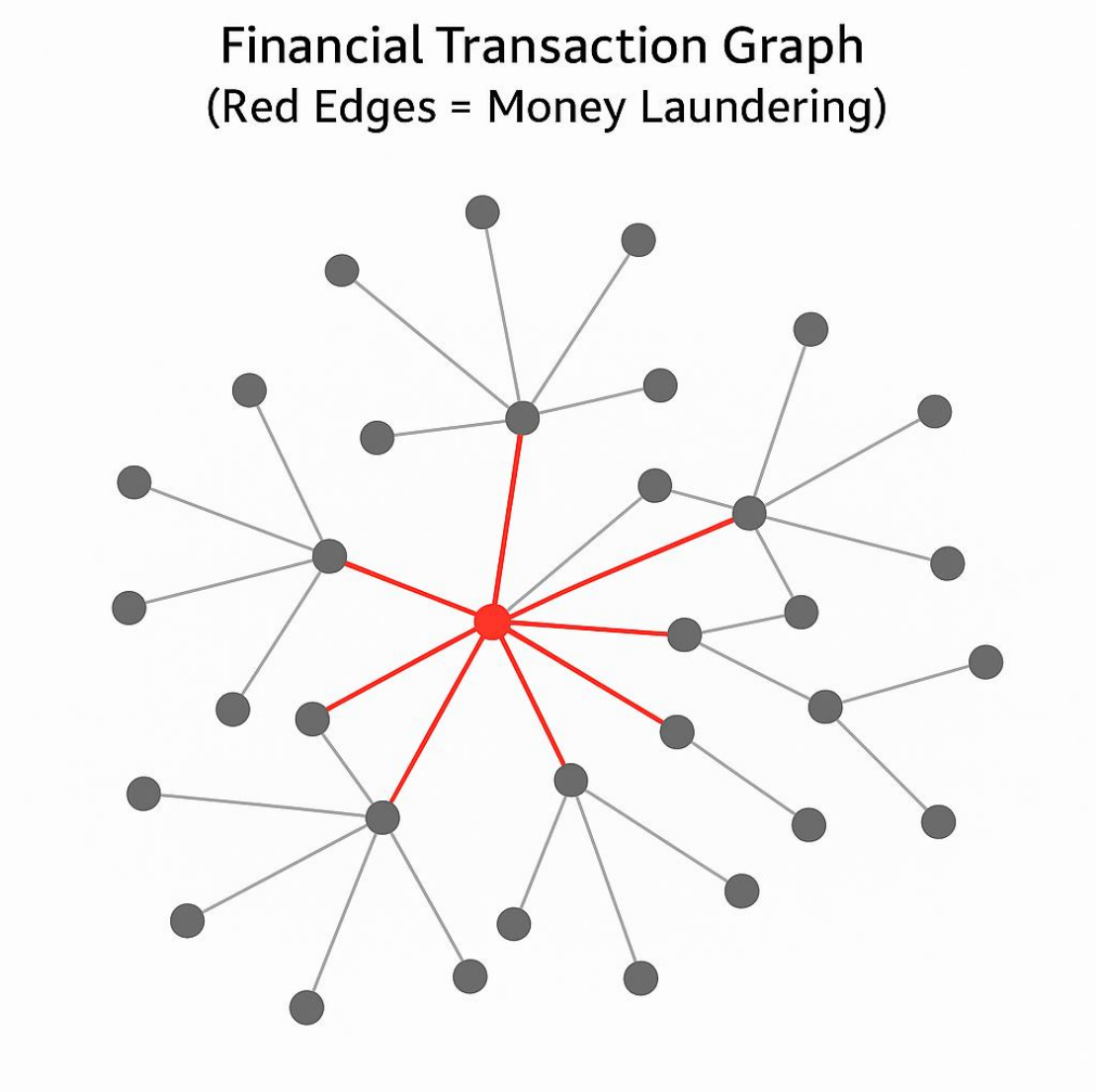


圖 7： 交易群組社會網絡示意圖

設交易資料集為一有向圖 $G = (V, E)$，其中節點 V 代表帳戶，邊 $E \subseteq V \times V$ 代表帳戶間的交易關係。令交易邊 $(u, v) \in E$ 表示帳戶 u 向帳戶 v 進行一筆交易。

利用 Union-Find 結構將圖中節點依連通性劃分為若干不相交的子集（群組）。初始時，每個節點為獨立集合，即

$$v \in V, \quad \text{parent}(v) = v.$$

對每條邊 (u, v)，執行 $\text{union}(u, v)$, 將 u 與 v 所屬集合合併，使得所有可達帳戶最終被聚合至同一集合。

透過路徑壓縮優化的操作，確定每個節點所屬集合代表元（根節點）：

$$\text{root}(v) = \text{find}(v)$$

使得

$$\forall u, v \in V, \text{root}(u) = \text{root}(v) \Leftrightarrow u, v$$

在同一交易連通分量中.

最後針對每個子集合 $C \subseteq V$，統計在該群組內發生的洗錢交易數目：

$$L(C) = \sum_{(u,v)\in E,\ u,v\in C} 1_{\text{Is\_laundering}(u,v)=1}$$

Algorithm 1 UF-AML Clustering Algorithm

Input: Transaction dataset D with edges (u, v) and laundering labels *Is_laundering*.
Initialize: Each account is in its own cluster.
for all transaction edges $(u, v) \in D$ **do**
    Union the clusters of accounts u and v.
end for
for all accounts **do**
    Find the root cluster representative.
end for
Calculate laundering transaction counts per cluster.
Output: Cluster assignments and laundering statistics.

本研究中，機器學習並非直接參與演算法的運作，而是作為驗證工具，用以**證明圖論指標對於洗錢偵測的效率能夠提升**。透過 UF-AML（Union-Find-based Anti-Money Laundering）分群演算法，我們能在交易網絡中建立弱連通群組（WCC），並在每個群組中計算多項圖論指標（如中心性、互惠性、Fan-in/Fan-out 比例等）。指標再與原始交易屬性融合後輸入至機器學習模型，比較「含圖論特徵」與「僅用原始特徵」模型間的效能差異。結果顯示，圖論特徵能顯著提升分類模型的 Precision 與 Recall，證實 UF-AML 架構在揭露潛在可疑群組上的有效性。

3.6 UF-FAE 架構與演算法定位

本研究提出之 UF-FAE 演算法，屬於啟發式的混合近似演算法（Heuristic-based Hybrid Approximation Algorithm）(Lozano & García-Martínez, 2010; Máximo & Nascimento, 2019)。其主要特點如下：

特點	說明
雜交性（Hybridization）	結合兩種或多種不同的啟發式技術，使演算法同時具備全域探索（Diversification）與區域強化（Intensification）能力。
啟發式導向（Heuristic-driven）	透過問題特性與經驗法則引導搜尋方向，快速找到「足夠好」的可行解，而非追求絕對最優解。
近似性（Approximation）	所獲得之解僅為最優解的近似值，但能在理論上或實驗上證明其效能接近最優。

結合 Union-Find 資料結構的結構性啟發（Structural heuristic）與機器學習模型的統計近似（Statistical approximation），藉由維護交易節點間的不相交集合結構，動態追蹤 DLT 網路中群組之連通關係，在此基礎上萃取群組層級的圖論特徵，作為啟發式初始解（Initial heuristic solution）供後續分類模型進行特徵近似與異常識別。

UF-FAE 的設計兼具結構可解釋性與運算效率，其核心演算法以 Union-Find 為骨架，在攤銷時間複雜度 O(α(n)) 下維護群組合併與查找操作，同時透過機器學習模型對多維特徵空間進行近似映射與分類推論。此混合式設計不僅保留拓撲結構資訊，亦能在高頻交易環境中實現具近似最優效率之洗錢行為偵測，因此可視為兼具結構導向啟發**與**資料導向近似（Data-driven approximation）之混合演算法框架。

3.6.1 Hybrid Approximation 與 Multimodal Learning 之區辨

雖然本研究提出之 UF-FAE 架構在概念上同時結合了結構性演算法與統計式學習模型，但其定位仍以啟發式混合近似演算法為核心，而非純粹的多模態機器學習（Multimodal Machine Learning）系統。兩者在理論基礎與整合層次上存在明顯區別。

啟發式混合近似演算法屬於演算法設計與最佳化理論（Optimization Theory）範疇，其核心目標為在可接受的時間複雜度下尋找接近最優解（near-optimal solution）。這類方法強調不同啟發式技術的策略性結合，以兼顧全域探索（Diversification）與區域強化（Intensification），常見於求解 NP-hard 問題之近似與搜尋任務(Blum & Roli, 2003; Lozano & García-Martínez, 2010; Máximo & Nascimento, 2019)。

UF-FAE 之架構上，Union-Find 作為結構性啟發的角色地位，透過路徑壓縮維持交易網絡之動態連通關係;而機器學習模組則作為統計機制,於特徵空間中進行分類與推論。此設計旨在以低攤銷時間並同時達成結構可解釋性與近似最佳化效率，應屬混合近似策略。

其次，多模態機器學習則屬於表示學習（Representation Learning）與資料融合（Data Fusion）範疇，著重於整合不同來源或型態的資料，以學習具跨模態語意的統一表徵(Baltrusaitis et al., 2019)。本研究在資料層面上，雖同時使用了圖論特徵（graph-based topological features）與交易屬性（transactional attributes）作為模型輸入，形成具多模態特徵融合（Multimodal feature fusion）的設計，但此部分僅屬於資料層次的特徵整合，並非跨感知或語意對齊的多模態學習架構。因此 UF-FAE 在理論定位上屬於「**結構導向啟發與資料導向近似並行的混合近似演算法框架**」，其中多模態特徵僅作為輸入層的因子實驗設計，以增強分類器的判斷能力，而非模型本體的設計哲學。

四、實驗結果

本研究提出的核心演算法為 UF-FAE（Union-Find-based Feature-Augmented Embedding），結合 Union-Find 路徑壓縮，將大規模交易網絡切分為弱連通子圖，並透過圖論指標與統計特徵進行高效能特徵萃取。為了驗證各組成要素對整體模型效能的貢獻，本研究進行消融實驗，逐步移除特定特徵或模組，並評估其對偵測準確度的影響。

此外，本研究同時採用多種統計分析、非監督式學習及監督式學習之機器學習來探索交易數據的分佈特性與關聯結構，為後續機器學習模型的建立與優化提供依據。機器學習部分則採用多種分類模型，包括邏輯迴歸、決策樹、隨機森林與線性支援向量機，並以交叉驗證評估其分類效能。本章節將詳細呈現 UF-FAE 的消融實驗結果，以及各機器學習模型的比較，全面驗證本研究方法的有效性與可行性。

研究流程完整解決現有區塊鏈 ETL（Extract-Transform-Load）常見的數據異質性與互動性資料遺失問題，兼顧可擴展性與分析精度。此外，透過圖結構特徵之自動化融合，能更精確描述金流路徑與節點角色，為後續異常偵測、社群發現等多元區塊鏈應用提供強大資料支持，結果如下:

Algorithm 3 UF-FAE: Union-Find based Feature-Augmented Embedding

Require: Transaction records $T = \{(s_i, r_i, a_i, t_i)\}$, where s_i and r_i are sender and receiver, a_i is amount, t_i is timestamp

Ensure: Feature-augmented embedding matrix $X \in R^{n \times d}$, where n is the number of groups

Initialize disjoint sets for all accounts using Union-Find;
for all transaction $(s_i, r_i, a_i, t_i) \in T$ **do**
    Perform Union operation: Union(s_i, r_i);
end for
Group transactions by connected components (clusters);
for all group G_j **do**
    Extract structural features:
    - Number of nodes / edges
    - Degree statistics (avg/max/min)
    - Betweenness or Closeness centrality (optional)

    Extract transactional features:
    - Mean / Std of transaction amounts
    - Temporal features (e.g., time gap statistics)
    - Entropy of sender / receiver diversity

    Concatenate all features into vector x_j;
end for
Stack all x_j vectors into feature matrix X;
(Optional) Normalize or embed X for downstream ML training;
return X;

演算法三 : UF-FAE 流程

4.1 多模態特徵融合與時間序列驗證之機器學習模型設計

本研究進一步設計一套結合圖論特徵（如節點數、邊數、中心性指標）與原始交易欄位（如金額、支付方式、幣別、地點）的**多模態特徵融合機器學習模型**，並採用嚴格時間序列切分（Time-Series Split）策略進行訓練與驗證。此設計模擬真實金融反洗錢監控情境，即僅使用歷史資料進行模型訓練，避免未來資訊洩漏，以提升實際部署效能的可信度。

資料來源為經圖結構擴充後的 SAML-D_with_graph_centrality 資料集，透過 PySpark 進行特徵處理與管線化建模。首先，將 Date 與 Time 欄位合併為 timestamp，並轉換為整數類型以作為時間序列排序依據。資料依照 timestamp 進行升冪排序後，取前 80% 作為訓練集、後 20% 作為測試集，確保時間上的資訊不可逆滲透。

在特徵設計方面，類別變數（如幣別、地點與支付方式）先經 StringIndexer 編碼，再以 OneHotEncoder 展開為向量格式；數值特徵則直接納入合併，最終整合為統一的 features 特徵向量。模型部分，本研究選用四種常見分類器進行比較分析，分別為：

- 邏輯迴歸
- 決策樹
- 隨機森林
- 線性支援向量機

每種模型皆使用相同特徵集與資料切分方式，並評估其在測試資料上的四項指標：AUC、Precision、Recall 與 F1-Score。除此之外，也針對具備特徵重要性評估功能的模型（如決策樹與隨機森林）進行前 10 名關鍵特徵排名輸出。

透過此一 pipeline，可系統化比較不同演算法於多模態融合 AML 資料上的實際效能，同時檢驗圖論結構特徵是否對模型預測帶來實質提升。此架構亦為日後接軌深度學習模型（如 GNN、GraphSAGE）提供一套可擴展的 baseline 參照，以下為資料愈處理之後延伸圖資料結構之檔案表格:

表 11: 資料預處理後之特徵融合資料欄位表

欄位名稱	中文說明
Time	時間
Date	日期
Sender_account	付款帳戶
Receiver_account	收款帳戶
Amount	金額
Payment_currency	支付幣別
Received_currency	收款幣別
Sender_bank_location	發款銀行所在地
Receiver_bank_location	收款銀行所在地
Payment_type	支付方式
Is_laundering	是否為洗錢交易（標註）
Laundering_type	洗錢類型
group_id	所屬圖群組編號
group_node_count	群組內帳戶數（節點數）
group_edge_count	群組內交易數（邊數）
group_bidirect_ratio	雙向交易比例
sender_degree	發送者的度數
receiver_degree	接收者的度數
sender_closeness	發送者的接近中心性

欄位名稱	中文說明
receiver_closeness	接收者的接近中心性
sender_betweenness	發送者的中介中心性
receiver_betweenness	接收者的中介中心性

4.2 消融實驗(UF-FAE)

本研究提出的核心演算法為 UF-FAE（Union-Find-based Feature-Augmented Embedding），結合 Union-Find 路徑壓縮，將大規模交易網絡切分為弱連通子圖，並透過圖論指標與統計特徵進行高效能特徵萃取。為了驗證各組成要素對整體模型效能的貢獻，本研究進行消融實驗，逐步移除特定特徵或模組，並評估其對偵測準確度的影響。

此外，本研究同時採用多種統計分析、非監督式學習及監督式學習之機器學習來探索交易數據的分佈特性與關聯結構，為後續機器學習模型的建立與優化提供依據。機器學習部分則採用多種分類模型，包括邏輯迴歸、決策樹、隨機森林與線性支援向量機，並以交叉驗證評估其分類效能。本章節將詳細呈現 UF-FAE 的消融實驗結果，以及各機器學習模型的比較，全面驗證本研究方法的有效性與可行性。

研究流程完整解決現有區塊鏈 ETL（Extract-Transform-Load）常見的數據異質性與互動性資料遺失問題，兼顧可擴展性與分析精度。此外，透過圖結構特徵之自動化融合，能更精確描述金流路徑與節點角色，為後續異常偵測、社群發現等多元區塊鏈應用提供強大資料支持，下一段落呈現結果 ：

本研究以四種機器學習模型進行特徵融合、原始欄位以及 SAML-D 延伸之純圖資料結構進行分析，皆採用 8:2 之比例進行前後時序分析，結果如下表:

模型	分組	AUC(ROC)	C1 Precision	C1 Recall	C1 F1	Weighted Precision	Weighted Recall	Weighted F1	時　間（秒）	Top 特徵（依重要性排序）
Logistic Regression	多模態	0.9866	0.7467	0.5558	0.6372	0.9992	0.9993	0.9992	246.77	receiver_closeness, receiver_betweenness, sender_betweenness
Decision Tree	多模態	0.8621	0.9799	0.8637	0.9182	0.9998	0.9998	0.9998	227.93	sender_degree, sender_betweenness, receiver_degree

模型	分組	AUC(ROC)	C1 Precision	C1 Recall	C1 F1	Weighted Precision	Weighted Recall	Weighted F1	時　間（秒）	Top 特徵（依重要性排序）
Random Forest	多模態	0.9913	1.0000	0.0070	0.0139	0.9989	0.9989	0.9983	883.70	receiver_degree, sender_degree, sender_betweenness
SVM (LinearSVC)	多模態	0.9320	1.0000	0.0019	0.0037	0.9989	0.9989	0.9983	272.35	receiver_closeness, receiver_betweenness, sender_betweenness
Logistic Regression	原始	0.7281	1.0000	0.0042	0.0084	0.9989	0.9989	0.9983	66.65	*（無顯著解釋性特徵）*
Decision Tree	原始	0.4461	1.0000	0.0247	0.0483	0.9989	0.9989	0.9984	91.37	Amount, Payment_type_vec_6
Random Forest	原始	0.7364	0.0000	0.0000	0.0000	0.9977	0.9989	0.9983	675.01	Amount, Payment_type_vec_6, Payment_type_vec_4
SVM (LinearSVC)	原始	0.6890	0.0000	0.0000	0.0000	0.9977	0.9989	0.9983	208.29	*（無顯著解釋性特徵）*
Logistic Regression	圖論	0.9786	0.9373	0.5166	0.6661	0.9994	0.9994	0.9993	91.14	receiver_closeness, receiver_betweenness, sender_betweenness
Decision Tree	圖論	0.9913	0.9803	0.8838	0.9296	0.9998	0.9998	0.9998	65.22	sender_degree, sender_betweenness, receiver_degree
Random Forest	圖論	0.9984	1.0000	0.9127	0.9544	0.9999	0.9999	0.9999	571.89	sender_degree, sender_betweenness, receiver_degree
SVM (LinearSVC)	圖論	0.8980	0.0000	0.0000	0.0000	0.9977	0.9989	0.9983	122.04	receiver_closeness, receiver_betweenness, sender_betweenness

分析與結論建議

本研究針對三組特徵組合（純原始、純圖論、多模態）與四種模型進行消融實驗，觀察其在 Class 1（洗錢交易）識別的效能差異，結果顯示：

多模態特徵（原始 + 圖論）

- **Decision Tree** 在多模態情境中表現最優，**Recall 達 0.8637、F1-score 為 0.9182**，遠高於其他模型，顯示能有效捕捉潛在洗錢行為。
- **Logistic Regression** 雖整體表現良好（AUC = 0.9866），但在 Class 1 的 Recall 僅 0.5558，顯示對少數異常類別的捕捉仍有不足。
- **Random Forest 與 SVM** 雖在 Class 0 上準確率極高，但 Class 1 的 Recall 嚴重偏低（RF = 0.0070；SVM = 0.0019），幾近無效，呈現嚴重**過度擬合於多數類別**的現象。
- 就執行時間而言，Logistic Regression 與 Decision Tree 均為效率與效能兼具的首選。

純原始特徵

- 所有模型在此分組下對 Class 1 的識別表現極差，**Recall 全數小於 0.03**，F1-score 幾乎無預測力。
- 雖然 Weighted Precision 與 F1-score 看似極高，但實際上主要來自 Class 0 的正確預測，具**明顯誤導性**。
- 特徵貢獻方面，大多模型僅依賴少數類別特徵（如 Payment_type_vec_6），而「金額（Amount）」的權重貢獻幾乎為 0，顯示其資訊不足、難以構成有效預測依據。

純圖論特徵

- 圖論特徵在 Class 1 的識別上幾乎**壓倒性優於其他分組**。
- **Random Forest 表現最強**：Recall 高達 **0.9127**，F1-score 為 **0.9544**，顯示圖結構能有效揭露潛在洗錢群體的特性。
- **Decision Tree** 同樣穩定，Recall 為 0.8838，準確率與效率皆優，**可視為本研究主推的模型選擇之一**。
- **SVM** 即使納入圖論特徵，仍無法識別 Class 1，進一步驗證線性模型在非線性異常行為辨識上之侷限。
- **Logistic Regression** 仍維持中等表現，Recall 雖不如 RF/DT，但具備可解釋性與穩定性，**適合做為風控輔助工具**。

整體觀察總結

比較面向	表現最佳	說明
Class 1 Recall	Random Forest（圖論）	高達 0.9127，對洗錢行為捕捉能力最強
整體穩定性	Decision Tree（圖論 + 多模態）	效能穩定、F1 高且訓練時間低
可解釋性	Logistic Regression（多模態）	模型透明，利於金融機構風控應用
耗時最少	Decision Tree（圖論）	僅 65 秒即可訓練完成，效能與效率兼具

儘管 Logistic Regression 為線性模型，對於高度非線性的洗錢行為結構存在建模侷限，但在結合多模態特徵後，其在 Class 1 上仍表現出令人驚豔的偵測能力（Precision = 0.7467，F1-score = 0.6372）。考量本研究資料具高度不平衡性（洗錢交易僅佔全部 0.1% 以下），此結果顯示多模態特徵融合對於異常行為捕捉的實質幫助，同時維持模型可解釋性，極具實務應用價值。

指標	面向
訓練+預測時間	UF-FAE 的快速可部署性
AUC	模型區分能力，與 baseline 相比是否更穩定
Precision	預測為詐欺時準確度（減少誤報）
Recall	捕捉所有詐欺者能力（提高敏感性）
F1	綜合指標，適用於樣本不平衡的 AML 問題

表 15: 機器學習指標涵義

AUC（Area Under Curve）：

$$\text{AUC} = \int_0^1 \text{TPR}\left(\text{FPR}^{-1}(x)\right)\, dx$$

Precision（精確率）：

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall（召回率）：

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

F1 Score :

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- 對於決策樹類模型，特徵重要性為：

$$\text{Importance}_j = \sum_{\text{split on } j} \frac{\text{samples at split}}{\text{total samples}} \cdot \text{information gain}$$

- 對 Logistic Regression / SVM，則取：

$|w_j|$ （係數的絕對值）

來表示特徵 j 的影響力。

在不引入原始交易欄位與類別欄位的情況下，研究僅以圖論結構所萃取之特徵（如節點度數、群內邊數、雙向交易比例、接近中心性與中介中心性等）進行機器學習訓練，結果顯示其仍具極高的偵測效能。特別是在 **Random Forest 模型**中，不僅 AUC 高達 **0.9984**、**F1 Score 達 0.9999**，同時也展現出極高的精確度與召回率。

更值得關注的是，其運算效率也相當優異。相比融合原始資料與類別欄位的多模態模型版本需耗時超過 **900 秒**，**僅用圖論指標的版本訓練與預測僅需約 569 秒**，大幅減少 30% 以上的執行時間。這種高效能低資源消耗的特性，尤其適用於需要近即時反應的大型交易系統或資安監控場景。

此外，圖論指標所反映的結構性異常（如極端中介中心性、過高節點連結度）往往是洗錢行為的本質表徵，具有高度可遷移性與跨平台適用性。這也說明即使在不完全揭露交易細節的情況下，透過網絡結構特徵依然能達成高準度的洗錢偵測，有效補足傳統特徵在資料不完整或加密情境下的不足。綜上所述，「僅使用圖論特徵」的模型在本研究中不僅達成近乎完美的分類表現，更兼具**運算效率與資料隱私彈性**，具備極高的實務應用潛力與推廣價值，本研究證實採用 Union-Find 與 WCC 進行特徵工程之前處理具備良好的演算法效率於 AML 系統。

五、結論

由於 Union–Find 結構在時間複雜度上已達 $O(\alpha(n))$ 的理論最優上界(Fredman & Saks, 1989; Tarjan, 1975)，若試圖進一步嘗試降低其連通性查詢或集合合併成本，均需突破現有計算複雜度理論基礎。UF-FAE 因此不以超越該下界為目標，而著重於將此結構性最優演算法與機器學習技術結合，以提升在動態分群與異常交易識別任務上的表現。

5.1 實務建議

本研究以非機器學習方法為基礎，結合圖論特徵與金額異常偵測邏輯，成功辨識出高風險交易群體，其全數皆為洗錢交易，顯示資料驅動的聚類技術具備高度潛力。然而，為全面強化反洗錢機制之實務應用與即時反應能力，仍需整合更高階的技術框架。

先前研究指出，洗錢交易的金額平均顯著高於一般交易（平均值達 $36,571.56，相較於正常交易僅 $8,682.22），而「跨國交易（cross-border）」與「信用卡交易」亦最常與洗錢活動產生關聯。統計檢定結果顯示此類關係具高度顯著性（$p < 0.001$）與本研究結果相符(Ramanujam, 2025)。

統計方法雖可提供初步偵測依據，卻難以掌握潛藏於大規模非結構化資料中的複雜關聯與時間演化模式。因此，**本研究特別提出結合演算法設計、區塊鏈技術與 AI 架構的 AML 解決方案構想**，具備以下多重優勢：

1. **區塊鏈（Blockchain）** 可提供交易全紀錄可追溯與不可竄改性，適合建構跨機構共享的帳戶圖譜與時間演化結構；
2. **圖論與深度學習（如 GNN）** 可揭露傳統方法無法偵測的關鍵節點與群體異常行為；
3. **非監督式學習聚類結合時間動態分析（如狀態空間模型）** 可即時標記高風險交易與群組異動，達成接近即時的預警系統。

本研究採用 Union-Find 演算法和 WCC 進行小世界網絡之分群後進行圖論指標分析能大幅增加識別洗錢速度，儘管 WCC 為無向圖，但仍可藉由 Sender 及 Receiver 確認其因果關係並轉為有向圖，而未來可藉由 Union-Find 後續延伸之 Kruskal 最短路徑演算法來做犯罪金流風險評估之方法，Kruskal 若使用路徑壓縮，時間複雜度可最佳化至 $O(|E|\alpha(|V|))$ (Kruskal, 1956)，作為未來脂實務風險評估之考量。

基於上述觀察，建議未來研究與金融監理機構應持續探索 AI 與區塊鏈整合之應用潛力，並導入實時資料串流處理框架（如 Spark Streaming）與分散式學習架構，實例上，SCARFF 系統以 Apache Kafka 與 Spark 為基礎建構串流處理流程，藉由滑動窗口(Sliding

window approach) 來處理概念飄移（concept drift），並且用於動態更新兩個分類器之動態規劃演算法，顯示即時資料處理技術對於金融詐欺風險評估具可行性(Carcillo et al., 2018)，與本研究提出的架構構想高度一致，如當滑動窗口飄移至新資料加入圖資料結構後，後續可藉 Union-Find 做及時快速的群組動態更新，當有小世界網絡群組發生合併時，可採用特徵融合交互學習，反之則採用對比學習，有利於深度學習之應用，因此可打造具備可擴展性與自適應能力的下一代反洗錢監控系統。

5.2 學術建議

本研究之研究結果顯示出相較於大數據分析與機器學習對於原始欄位表格之分類具有一定程度上的限制，原始欄位表格無法呈現帳戶間所形成之社會網絡脈絡，研究結果顯示相對於原始欄位及特徵融合，使用由社會網絡組成之純粹的圖資料結構在演算法設計上比直接對原始欄位進行機器學習分析來得更有效率，彰顯族群研究之重要性，根據 Simon (1957)提出的有限理性框架(Bounded Rationality)，行為人會在資訊不足、環境複雜和時間限制下，僅求「足夠好」(Satisficing)的決策。傾向選擇當下即可完成目的的行為，犯罪者洞悉跨行交易可降低暴露風險，偏好選擇快速的資金轉移策略(Gilmour, 2016; Tiwari et al., 2023)，而理性選擇理論（Rational Choice Theory）指出行為人是否犯罪取決於對成本與收益的衡量(Becker, 1968; Cornish & Clarke, 1987; Reuter & Riccardi, 2024)。

本研究採用理性選擇理論解釋洗錢行為背後的行動動機。雖然理性選擇理論主要以「個體」為分析單位，但犯罪組織本身亦是由多個基於利害計算而合作的個體所構成，因此組織犯罪仍可透過個體理性選擇的聚合來加以說明。換言之，本研究並非將犯罪組織視為具有單一意志的整體行動者，而是強調組織內部的合作、角色分工與資源交換皆可追溯至個體層次的理性評估。未來研究可考慮結合**社會交換理論**（Social Exchange Theory），以進一步說明個體在群體中的互信形成機制及合作策略如何累積而生成穩定的犯罪網絡結構。(新增 RCT 理論的論述補充)

使用者在做重要決定時會經過認知上複雜的決策過程(Simon, 1957)，而洗錢者會共用某些資金轉移模板，不論是手動操作還是使用自動化工具，顯示其背後具有行動上的協調性與集體策略規劃，為群體決策過程的直接表現(Oggier et al., 2020)。

本研究認為單獨使用機器學習對原始欄位進行大數據分析效果應結合小世界網絡之族群行為特性，有利於掌握洗錢犯罪者之科技使用行為及習慣
(Khan & Akcora, 2022)，Ramanujam (2025)對 SAML-D 資料集進行統計分析，指出了金額

為最有力之預測變數，但解釋力僅有 0.1%，而跨國交易及信用卡交易最常與洗錢具有顯著關聯，而機器學習可採用貝葉斯決策之關聯規則(Association Rules)進行未來的預測，適用於金融風險偵測領域中(Ethem, 2014)。

因此，無論是了解統計分析上的描述性統計及變項關聯，抑或是大數據分析和機器學習進行分類和預測，採用特徵融合之方法結合社會網絡分析，能夠更好地解釋洗錢犯罪者所採用的科技使用行為，並且利用圖資料結構的因果關係呈現，彰顯出機器學習與人為決策結合之重要性。

六、研究限制

本研究所提出之 **UF-FAE** 架構，其核心假設為資料集必須滿足**因果有向性（Causal directionality）與時間單調性（Temporal monotonicity）兩項條件，以確保整體結構可被建模為 DAG**。在此前提下，UF-FAE 才能透過 Union–Find 維護連通分支 π_o 的動態演化，並於拓撲層形成具偏序性質的 Rank-Filtration DAG 與同倫商空間。若資料不具明確的時間戳或事件排序，則無法定義 filtration 序列:

$$K_0 \subset K_1 \subset \cdots \subset K_t,$$

UF-FAE 將退化為靜態 Union–Find 結構，無法呈現拓撲演化與群組融合的時間關係。同樣地，若資料缺乏明確方向性或存在環狀結構（Cyclic dependencies），則 DAG 假設失效，偏序拓撲與同倫壓縮皆不再成立。因此，UF-FAE 僅適用於可推導出因果一致性（causal–temporal coherence）的資料，亦即：

$$\forall (u,v) \in E,\ (u \to v) \Rightarrow t(u) < t(v)$$

當無法滿足此條件時，需先重建 pseudo-time 或進行強連通成分收縮（SCC contraction），方可重新建立有效的 DAG 結構與拓撲層分析。

參考文獻

Abhinaya, A. (2024). INFLUENCE OF THE COVID-19 PANDEMIC ON FINANCIAL CRIMES.

International Research Journal of Modernization in Engineering Technology and Science, *6*(6), 3427-3439. https://www.irjmets.com/uploadedfiles/paper/issue_6_june_2024/59496/final/fin_irjmets1719342103.pdf

Aidoo, S. (2025). The Role of Blockchain in AML Compliance: Potential Applications and Limitations.

Akoglu, L., Tong, H., & Koutra, D. (2014). Graph-based Anomaly Detection and Description: A Survey. *arXiv preprint arXiv:1404.4679v2*. https://arxiv.org/abs/1404.4679

Alhaidari, A., Kalal, B., Palanisamy, B., & Sural, S. (2025). *SolRPDS: A Dataset for Analyzing Rug Pulls in Solana Decentralized Finance* Proceedings of the Fifteenth ACM Conference on Data and Application Security and Privacy, Pittsburgh, PA, USA. https://doi.org/10.1145/3714393.3726487

Alotibi, J., Almutanni, B., Alsubait, T., Alhakami, H., & Baz, A. (2022). Money Laundering Detection using Machine Learning and Deep Learning. *International Journal of Advanced Computer Science and Applications*, *13*. https://doi.org/10.14569/IJACSA.2022.0131087

Altman, E., Blanuša, J., Niederhäusern, L. v., Egressy, B., Anghel, A., & Atasu, K. (2023). *Realistic synthetic financial transactions for anti-money laundering models* Proceedings of the 37th International Conference on Neural Information Processing Systems, New Orleans, LA, USA.

B, R. (2025). Money laundering through Cryptocurrency and Its Arrangements. *Lex Publica (Appthi Journal)*, *1*(1), 65-74. https://journal.appthi.org/index.php/lexpublica/article/download/167/164/395

Badev, A., & Watsky, C. (2023). Interconnected DeFi: Ripple Effects from the Terra Collapse. *Finance and Economics Discussion Series*(2023-044), 1-39. https://doi.org/10.17016/feds.2023.044

Bai, Y., Lee, S., & Seo, S. H. (2025). A Survey on Directed Acyclic Graph-Based Blockchain in Smart Mobility. *Sensors (Basel)*, *25*(4). https://doi.org/10.3390/s25041108

Baltrusaitis, T., Ahuja, C., & Morency, L. P. (2019). Multimodal Machine Learning: A Survey and Taxonomy. *IEEE Trans Pattern Anal Mach Intell*, *41*(2), 423-443. https://doi.org/10.1109/TPAMI.2018.2798607

Batool, F., Sharjeel, M., & Omer, M. (2024). The Evolving Impact of Money Laundering on Financial Markets.-5. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4882815

Becker, G. S. (1968). Crime and Punishment: An Economic Approach. *Journal of Political Economy*, *76*(2), 169-217. http://www.jstor.org/stable/1830482

Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Comput. Surv.*, *35*(3), 268–308. https://doi.org/10.1145/937503.937505

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5-32. https://doi.org/10.1023/A:1010933404324

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, *30*(1), 107-117. https://doi.org/https://doi.org/10.1016/S0169-7552(98)00110-X

Buterin, V. (2013). Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform. In.

Buterin, V. (2020). *A rollup-centric Ethereum roadmap*. https://ethereum-magicians.org/t/a-rollup-centric-ethereum-roadmap/4698

Carcillo, F., Dal Pozzolo, A., Le Borgne, Y.-A., Caelen, O., Mazzer, Y., & Bontempi, G. (2018). SCARFF : A scalable framework for streaming credit card fraud detection with spark. *Information Fusion*, *41*, 182-194. https://doi.org/10.1016/j.inffus.2017.09.005

Cernera, F., Morgia, M. L., Mei, A., & Sassi, F. (2023). *Token spammers, rug pulls, and sniper bots: an analysis of the ecosystem of tokens in Ethereum and in the binance smart chain (BNB)* Proceedings of the 32nd USENIX Conference on Security Symposium, Anaheim, CA, USA.

Chainalysis Research, T. (2024). *Money Laundering Activity Spread Across More Service Deposit Addresses in 2023, Plus New Tactics from Lazarus Group* (Chainalysis Blog, Issue. https://www.chainalysis.com/blog/2024-crypto-money-laundering/

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to Algorithms* (4th ed.). MIT Press.

Cornish, D. B., & Clarke, R. V. (1987). Understanding Crime Displacement: An Application of Rational Choice Theory. *Criminology*, *25*(4), 933-948. https://doi.org/10.1111/j.1745-9125.1987.tb00826.x

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273-297. https://doi.org/10.1007/BF00994018

Cox, D. R. (1958). The Regression Analysis of Binary Sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, *20*(2), 215-242. http://www.jstor.org/stable/2983890

CryptoManiaks Editorial, T. (2024). The Complete List of Solana Outages and Failures: Why Does SOL Keep Failing? *CryptoManiaks*. https://cryptomaniaks.com/crypto-news/solana-outage-list-failures-sol-blockchain-mainnet

Deprez, B., Baesens, B., Verdonck, T., & Verbeke, W. (2025). *GARG-AML against Smurfing: A Scalable and Interpretable Graph-Based Framework for Anti-Money Laundering*. https://doi.org/10.48550/arXiv.2506.04292

Ethem, A. (2014). *Introduction to Machine Learning*. MIT Press. https://ieeexplore.ieee.org/book/6267367

Federal Bureau of Investigation. (2025). North Korea Responsible for $1.5 Billion Bybit Hack

[Public Service Announcement]. *Internet Crime Complaint Center (IC3) Public Service Announcement*. https://www.ic3.gov/psa/2025/psa250226

Feedzai. (2023). The Future of AML: New Insights from Feedzai's 2023 Report. Retrieved August 24, from https://www.feedzai.com/blog/the-future-of-aml-new-insights-from-feedzais-2023-report

Financial Conduct, A. (2017). *Distributed Ledger Technology: Feedback Statement on Discussion Paper DP17/3*. https://www.fca.org.uk/publication/feedback/fs17-04.pdf

Fredman, M., & Saks, M. (1989). *The cell probe complexity of dynamic data structures* Proceedings of the twenty-first annual ACM symposium on Theory of computing, Seattle, Washington, USA. https://doi.org/10.1145/73007.73040

Galler, B. A., & Fisher, M. J. (1964). An improved equivalence algorithm. *Commun. ACM*, *7*(5), 301–303. https://doi.org/10.1145/364099.364331

Gilmour, N. (2016). *Improving the prevention of money laundering in the UK: A situational crime prevention approach* [PhD thesis, University of Portsmouth]. Portsmouth, UK.

Grand View, R. (2025). *Anti-money Laundering Market Size, Share, & Trends Analysis Report By Component, By Product, By Deployment, By Enterprise Size, By End Use, By Region, And Segment Forecasts, 2025–2030*. https://www.grandviewresearch.com/industry-analysis/anti-money-laundering-market

Hajr, L., Katamoura, S., & Mirza, A. (2023). Bitcoin Cryptocurrency and Electronic Commerce in Saudi Arabia. *Sage Open*, *13*(4). https://doi.org/10.1177/21582440231218513

Hinzen, L., John, K., & Saleh, F. (2018). Bitcoin's Fatal Flaw: The Limited Adoption Problem. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3268328

Holm, J., Lichtenberg, K. d., & Thorup, M. (2001). Poly-logarithmic deterministic fully-dynamic algorithms for connectivity, minimum spanning tree, 2-edge, and biconnectivity. *J. ACM*, *48*(4), 723–760. https://doi.org/10.1145/502090.502095

Hopcroft, J. E., & Ullman, J. D. (1979). *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley.

Huang, C.-C., & Trangle, A. (2020). Anti-Money Laundering and Blockchain Technology. *AML Case Study*.

Jensen, R. I. T., & Iosifidis, A. (2023). Fighting Money Laundering With Statistics and Machine Learning. *IEEE Access*, *11*, 8889-8903. https://doi.org/10.1109/access.2023.3239549

Jonas, T., William, S., Alex, S., Navin, R., Hans, M., & Serguei, P. (2021). *IOTA 2.0 Incentives and Tokenomics Whitepaper*. https://www.iota.org/research/academic-papers

Khan, A., & Akcora, C. G. (2022). Statistical Graph Mining: From Social Networks and Event Logs to Blockchain. In *CIKM 2022 Tutorial*.

Kruskal, J. B. (1956). On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. *Proceedings of the American Mathematical Society*, *7*(1), 48-50. https://doi.org/10.2307/2033241

Kumar, S. (2023). *Data Silos -A Roadblock for AIOps*. https://doi.org/10.13140/RG.2.2.31931.18721

Li, X., Cao, X., Qiu, X., Zhao, J., & Zheng, J. (2017, 13-16 Aug. 2017). Intelligent Anti-Money Laundering Solution Based upon Novel Community Detection in Massive Transaction Networks on Spark. 2017 Fifth International Conference on Advanced Cloud and Big Data (CBD),

Lozano, M., & García-Martínez, C. (2010). Hybrid metaheuristics with evolutionary algorithms specializing in intensification and diversification: Overview and progress report. *Computers & Operations Research*, *37*(3), 481-497. https://doi.org/10.1016/j.cor.2009.02.010

Lucinity. (2024). Data Silos and AML Compliance. *Lucinity Blog*. https://www.lucinity.com/blog/data-silos-and-aml-compliance

Mamache, H., Mazué, G., Rashid, O., Bu, G., & Potop-Butucaru, M. (2021). *Resilience of IOTA Consensus*. arXiv preprint arXiv:2111.07805. https://arxiv.org/abs/2111.07805

Martínez-Sánchez, J., Cruz-García, S., & Venegas-Martínez, F. (2020). Money laundering control in Mexico: A risk management approach through regression trees (data mining). *Journal of Money Laundering Control*, *ahead-of-print*. https://doi.org/10.1108/JMLC-10-2019-0083

Máximo, V. R., & Nascimento, M. C. V. (2019). Intensification, learning and diversification in a hybrid metaheuristic: an efficient unification. *Journal of Heuristics*, *25*(4–5), 539–564. https://doi.org/10.1007/s10732-018-9373-1

Mirenda, L., Mocetti, S., & Rizzica, L. (2022). The Economic Effects of Mafia: Firm Level Evidence. *American Economic Review*, *112*(8), 2748-2773. https://doi.org/10.1257/aer.20201015

Morgan, A. (2024). *Money laundering and the harm from organised crime: Results from a data linkage study*. https://doi.org/10.52922/sp77628

Movva, S., & Dasarraju, V. K. (2024). Impact of Blockchain on FinTech and Payment Systems. *Journal of Technology and Systems*, *6*, 78-86. https://doi.org/10.47941/jts.2026

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. In.

Neo4j. (2023). *Accelerate Fraud Detection With Graph Databases: How Graph Design Patterns Help You Identify and Investigate Suspicious Activity*. https://neo4j.com

Neo4j, I. (2021). *Financial Fraud Detection with Graph Data Science* (Neo4j Graph Data Science Whitepaper, Issue. https://neo4j.com/wp-content/uploads/2021/03/Financial-Fraud-Detection-with-Graph-Data-Science-Whitepaper.pdf

Oad, A., Razaque, A., Tolemyssov, A., Alotaibi, M., Alotaibi, B., & Zhao, C. (2021). Blockchain-Enabled Transaction Scanning Method for Money Laundering Detection. *Electronics*, *10*(15). https://doi.org/10.3390/electronics10151766

Oggier, F., Datta, A., & Phetsouvanh, S. (2020). An ego network analysis of sextortionists. *Social Network Analysis and Mining*, *10*. https://doi.org/10.1007/s13278-020-00650-x

Oztas, B., Cetinkaya, D., Adedoyin, F., Budka, M., Dogan, H., & Aksu, G. (2023). *Enhancing Anti-Money Laundering: Development of a Synthetic Transaction Monitoring Dataset* 2023 IEEE International Conference on e-Business Engineering (ICEBE),

Pambudi, B. N., Hidayah, I., & Fauziati, S. (2019, 5-6 Dec. 2019). Improving Money Laundering Detection Using Optimized Support Vector Machine. 2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI),

Pan, E. (2024). Machine Learning in Financial Transaction Fraud Detection and Prevention. *Transactions on Economics, Business and Management Research*, *5*, 243-249. https://doi.org/10.62051/16r3aa10

Pareja, A. H., Rossi, R. A., Rao, J. L., Hasani, R., Bronstein, M. M., & Ahmed, N. K. (2020). EvolveGCN: Evolving Graph Convolutional Networks for Dynamic Graphs. Proceedings of the AAAI Conference on Artificial Intelligence,

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, *1*(1), 81-106. https://doi.org/10.1007/BF00116251

Raiter, O. (2021). Applying Supervised Machine Learning Algorithms for Fraud Detection in Anti-Money Laundering. https://doi.org/10.17613/2g0z-0814

Ramanujam, B. (2025). Statistical in Sights in to Anti-Money Laundering: Analyzing Large-Scale Financial Transactions. *International Journal of Engineering Research & Technology (IJERT)*, *14*(04). http://www.ijert.org

Reuter, P., & Riccardi, M. (2024). Introduction to Special Issue on "Understanding Money Laundering: Empirical and Theoretical Insights into Offenders, Typologies, and Determinants of Criminal Behaviour". *European Journal on Criminal Policy and Research*, *30*. https://doi.org/10.1007/s10610-024-09604-x

Reuter, P., & Truman, E. M. (2004). *Chasing dirty money: The fight against money laundering*. Institute for International Economics.

Rifat Hossain, M., Nirob, F. A., Islam, A., Rakin, T. M., & Al-Amin, M. (2024). A Comprehensive Analysis of Blockchain Technology and Consensus Protocols Across Multilayered Framework. *IEEE Access*, *12*, 63087-63129. https://doi.org/10.1109/access.2024.3395536

Sealey, N., Aijaz, A., & Holden, B. (2022). *IOTA Tangle 2.0: Toward a Scalable, Decentralized, Smart, and Autonomous IoT Ecosystem*. https://doi.org/10.48550/arXiv.2209.04959

Sedgewick, R., & Wayne, K. (2011). *Algorithms* (4th ed.). Addison-Wesley.

Simon, H. A. (1957). *Models of Man, Social and Rational: Mathematical Essays on Rational Human Behavior in a Social Setting*. John Wiley & Sons.

Slutzky, P., Villamizar-Villegas, M., & Williams, T. (2020). Drug Money and Bank Lending: The Unintended Consequences of Anti-Money Laundering Policies. *SSRN Electronic*

Journal. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3280294

Song, K., Dhraief, M. A., Xu, M., Cai, L., Chen, X., Mithal, A., & Chen, J. (2024). *Identifying Money Laundering Subgraphs on the Blockchain* Proceedings of the 5th ACM International Conference on AI in Finance, Brooklyn, NY, USA. https://doi.org/10.1145/3677052.3698635

Starnini, M., Tsourakakis, C. E., Zamanipour, M., Panisson, A., Allasia, W., Fornasiero, M., Puma, L. L., Ricci, V., Ronchiadin, S., Ugrinoska, A., Varetto, M., & Moncalvo, D. (2021, 2021//). Smurf-Based Anti-money Laundering in Time-Evolving Transaction Networks. Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track, Cham.

Talend. (2022). Data Silos, Why They're a Problem & How to Fix It. *Talend Resources*. https://www.talend.com/resources/what-are-data-silos/

Tarjan, R. E. (1975). Efficiency of a good but not linear set union algorithm. *Journal of the ACM (JACM)*, *22*(2), 215-225. https://doi.org/10.1145/321879.321884

TechTarget. (2024). Why data silos matter: Settling ownership of data issues. *SearchDataManagement (TechTarget)*. https://www.techtarget.com/searchdatamanagement/tip/Why-data-silos-matter-Settling-ownership-of-data-issues

Tian, Z., Ding, Y., Yu, X., Gong, E., Liu, J., & Ren, K. (2025). *Towards Collaborative Anti-Money Laundering Among Financial Institutions* Proceedings of the ACM on Web Conference 2025,

Tiwari, M., Ferrill, J., Gepp, A., & Kumar, K. (2023). Factors influencing the choice of technique to launder funds: The APPT framework. *Journal of Economic Criminology*, *1*. https://doi.org/10.1016/j.jeconc.2023.100006

Treleaven, P., Brown, R. G., & Yang, D. (2017). Blockchain technology in finance. *Computer*, *50*(9), 14-17. https://doi.org/10.1109/MC.2017.3571042

Turner, A., McCombie, S., & Uhlmann, A. (2019). A target-centric intelligence approach to WannaCry 2.0. *Journal of Money Laundering Control*, *22*, 646-665. https://doi.org/10.1108/JMLC-01-2019-0005

United Nations Office on, D., & Crime. (2025). Money laundering. https://www.unodc.org/unodc/en/money-laundering/overview.html

Wang, X., & Dong, G. (2009, 30 Nov.-1 Dec. 2009). Research on Money Laundering Detection Based on Improved Minimum Spanning Tree Clustering and Its Application. 2009 Second International Symposium on Knowledge Acquisition and Modeling,

Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, *393*(6684), 440-442. https://doi.org/10.1038/30918

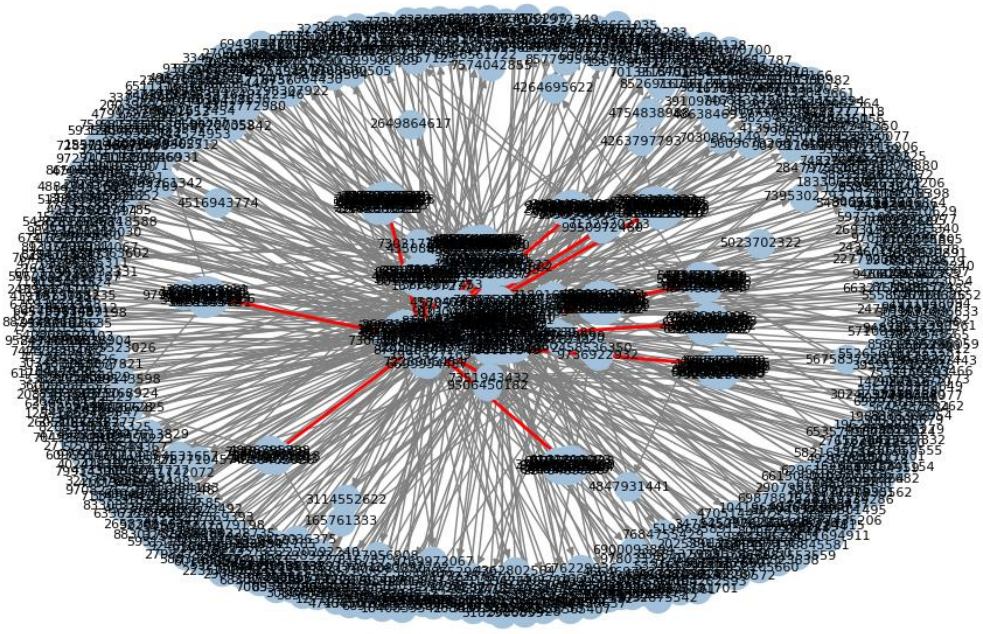
Weber, M., Domeniconi, G., Chen, J., Weidele, D. K. I., Bellei, C., Robinson, T., & Leiserson, C. E. (2019, 2019/08). Anti-Money Laundering in Bitcoin: Experimenting with Graph

Convolutional Networks for Financial Forensics. KDD ’19 Workshop on Anomaly Detection in Finance, Anchorage, AK, USA.

World Economic, F. (2016). *The future of financial infrastructure: An ambitious look at how blockchain can reshape financial services*. https://www3.weforum.org/docs/WEF_The_future_of_financial_infrastructure.pdf

Xu, C., Liu, C., Nie, D., & Gai, L. (2021). How Can a Blockchain-Based Anti-Money Laundering System Improve Customer Due Diligence Process?(Journal of Forensic and Investigative Accounting, 2021).

附錄:



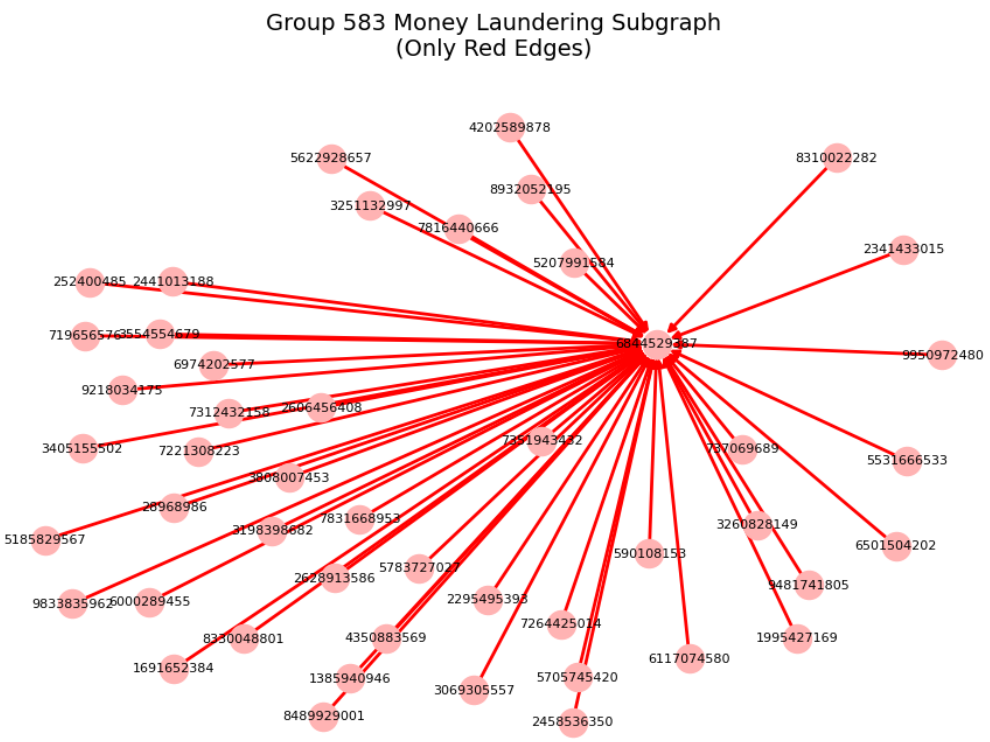


圖 8：洗錢交易網絡示意圖

研究工具

為應對本研究中 SAML-D 洗錢資料集所涉及的高維特徵、多筆交易紀錄以及圖論分群與時間序列機器學習模型等複雜計算需求，本研究選擇使用 Google Cloud Platform (GCP) 所提供之 Dataproc 雲端叢集服務作為主要運算環境。

GCP Dataproc 是一種基於 Apache Spark、Apache Hadoop 及 Hive 等開源生態系構建的彈性雲端運算平台，能夠快速部署分散式叢集以進行大數據處理、機器學習模型訓練、資料轉換與圖計算任務。相較於傳統本地端架設環境，Dataproc 具備以下優勢：

- **彈性擴展性（Elastic Scaling）**：可依據任務規模動態調整叢集節點數，節省成本並加速訓練。
- **高速分散式運算**：整合 Spark 平台進行記憶體內運算，適合處理大規模交易圖資料與即時特徵提取任務。
- **原生整合 GCS（Google Cloud Storage）**：支援高效存取 .parquet 格式資料及模型權重儲存。
- **安全與管理性佳**：支援 IAM 權限控制、日誌追蹤與任務自動化。

本研究中 Dataproc 被用於執行如 Union-Find 分群、圖中心性指標萃取、PySpark 特徵處理流程建構、時間序列切分與多模型交叉驗證等核心分析工作。透過將運算資源遷

移至雲端，研究者得以縮短訓練時間、確保環境穩定性，並支援未來大規模監理實務部署的可行性。所有大規模資料處理與特徵工程流程，皆於 Google Cloud Dataproc 平台執行，並根據資料量與處理需求設置如下資源配置：

- 叢集類型：Dataproc 2.1-debian11（含 Jupyter、Zeppelin 可視化元件）
- 區域與可用區：us-central1, us-central1-a
- Master 節點：n2-standard-4，200GB
- Worker 節點：n2-highmem-4，4 台，每台 200GB
- GCS bucket 持久化：saml-d
- 空閒自動關閉（max idle 30 分鐘），減少資源浪費

此資源配置可確保在多執行緒和巨量資料的情境下，所有 PySpark、資料預處理與特徵工程能於可接受時間內完成，同時亦可動態調整叢集大小，以因應不同階段的運算需求。

GitHub

https://github.com/scuranger0625/UF-FAE

https://github.com/scuranger0625/Anti-Money-Laundering-Transaction-Data-SAML-D-