Alma mater studiorum
Università di Bologna

Corso di laurea magistrale in informatica

Progetto per il corso di Blockchain and Cryptocurrencies

Componenti del gruppo:

Andrea Accornero	1097521
Giuseppe Pio Salcuni	1100090

Anno Accademico 2023/2024

Indice

1 Introduzione

Questo progetto è parte della prova d'esame per il corso "*Blockchain and Cryptocurrencies*. Abbiamo affrontato un problema di Antiriciclaggio (AML), che è cruciale quando si tratta di transazioni con criptovalute, a causa della loro natura decentralizzata e meno monitorata. In questo caso, ci siamo concentrati sulla classificazione delle transazioni illecite, che è l'obiettivo principale di questo tipo di compito. Questo compito è molto importante per le istituzioni finanziarie per ottenere la conformità ai requisiti legali per monitorare attivamente e segnalare attività sospette. In uno studio effettuato da un altro studente l'obiettivo era creare un confronto tra diversi approcci, per scoprire quali processi sono più adatti a risolvere questo tipo di problema. Weber et al. nel loro paper (Weber et al., 2019) hanno indagato questo campo cercando di applicare alcune tecniche di machine learning e deep learning per identificare possibili transazioni sospette e etichettare i nodi che le hanno effettuate come illecite. In lavori precedenti sono stati utilizzati modelli di machine learning per classificare la natura di un nodo, ottenendo risultati piuttosto buoni ad esempio con l'uso di classificatori Random Forest, che supera persino l'approccio Graph Convolutional Network. È stato anche utilizzato un GCN temporale (Evolve GCN (Pareja et al., 2020)), ma con un miglioramento molto piccolo delle prestazioni. Altri lavori, anche più recentemente, hanno affrontato lo stesso problema migliorando i risultati. È il caso di (Lo et al., 2023), che hanno utilizzato l'incorporamento dei nodi auto-supervisionato in GNNs. Lo studio, condotto da un altro studente, ha coinvolto l'implementazione di diverse soluzioni di reti grafiche, tra cui GNN incorporate in PyTorch Geometric (come GCN, GAT, GATv2, Chebyshev e GraphSage) e un'implementazione manuale di GAT. L'obiettivo principale era classificare i nodi come leciti o illeciti utilizzando un approccio di apprendimento transduttivo. Sebbene questo approccio richieda il riaddestramento dell'intero modello quando viene aggiunto un nuovo nodo al grafo, è stato

seguito poiché le reti neurali grafiche differiscono nel modo in cui gestiscono le informazioni del grafo e potrebbero influenzare i risultati finali. Sono stati condotti sei esperimenti basati sulle architetture GNN menzionate sul Dataset Ellittico, confrontando i risultati per valutare l'efficacia delle diverse strategie e tecniche utilizzate. Si è scoperto che l'attenzione è stata fondamentale per ottenere buone prestazioni, con implementazioni come GAT e GATv2 che hanno mostrato miglioramenti significativi rispetto all'approccio GCN semplice.

Questa parte del nostro studio rappresenta il punto in cui abbiamo condotto le nostre ricerche. L'attenzione è stata posta sulla ricreazione delle caratteristiche del dataset, con particolare enfasi sulla trasformazione del dataset originale opaco dell'Elliptic. Il problema principale riscontrato è stato il mancato accesso alle descrizioni delle caratteristiche nel dataset originale, il che ha reso difficile l'interpretazione e l'analisi dei dati. Tuttavia, abbiamo ipotizzato che fosse possibile partire dal grafo delle transazioni del dataset Elliptic originale, eliminare tutte le caratteristiche e tentare di crearne di nuove basate esclusivamente sulle proprietà del grafo stesso. Esempi di queste nuove caratteristiche potrebbero includere i gradi dei nodi, il coefficiente di clustering dei nodi, misure di centralità, e altre metriche relative alla struttura del grafo. Riteniamo che questa approccio potrebbe aiutarci a ottenere una migliore comprensione delle transazioni e a migliorare le prestazioni dei nostri modelli. Successivamente, abbiamo pianificato di riapplicare tutte le tecniche di classificazione basate sulle reti neurali grafiche utilizzate nel nostro studio, valutando le prestazioni dei modelli sui nuovi dati caratterizzati dalle nuove features aggiunte. Questo ci avrebbe permesso di determinare se l'integrazione delle nuove caratteristiche avrebbe portato a un miglioramento nell'identificazione delle transazioni illecite, mantenendo comunque la conoscenza pregressa sulla classificazione delle transazioni.

2 Dataset

2.1 Descrizione

Il Dataset Elliptic mappa le transazioni di Bitcoin a entità reali appartenenti a categorie lecite (scambi, fornitori di portafogli, minatori, servizi leciti, ecc.) rispetto a categorie illecite (truffe, malware, organizzazioni terroristiche, ransomware, schemi Ponzi, ecc.). Il compito su questo dataset consiste nella classificazione dei nodi illeciti e leciti nel grafo.

Questo dataset fornisce un quadro unico delle transazioni Bitcoin, consentendo agli analisti di identificare e comprendere meglio i pattern e le dinamiche dietro le transazioni sia legittime che illecite. L'obiettivo principale è distinguere tra entità e transazioni legittime, come scambi e fornitori di portafogli, e attività sospette o illegali, come truffe o attività associate a gruppi terroristici. Questa distinzione è fondamentale per le autorità di regolamentazione e le istituzioni finanziarie nell'implementare misure di sicurezza e di conformità normativa per prevenire il riciclaggio di denaro e altre attività illecite nel contesto delle criptovalute.

2.2 Contenuto

Il dataset è composto da 3 file csv: il primo file mappa i nodi con le loro etichette (lecito/illecito/sconosciuto), il secondo file introduce gli archi tra 2 nodi utilizzando i loro ID di transazione e l'ultimo ha gli ID dei nodi con 166 caratteristiche. Un nodo nel grafo rappresenta una transazione, un arco rappresenta un flusso di Bitcoin tra una transazione e l'altra. Ciascun nodo ha 166 caratteristiche ed è stato etichettato come creato da un'entità "lecita" (ad esempio, quelle effettuate da scambi di criptovalute regolamentati), "illecita" (ad esempio, quelle effettuate da mercati neri) o "sconosciuta". Ci

sono 203.769 nodi e 234.355 archi nel grafo, abbiamo il 2% (4.545) dei nodi che sono etichettati come illeciti e il 21% (42.019) sono etichettati come leciti. Le transazioni rimanenti non sono etichettate in merito a lecito o illecito. Ci sono anche 49 distinti passaggi temporali. Le prime 94 caratteristiche rappresentano informazioni locali sulla transazione e le restanti 72 caratteristiche sono caratteristiche aggregate ottenute utilizzando informazioni sulla transazione a una distanza di un passo temporale all'indietro/in avanti dal nodo centrale.

3 Analisi Esplorativa dei Dati

3.1 Introduzione

Nella presente sezione, condurremo un'analisi esplorativa approfondita del Dataset Ellittico. Questa fase cruciale del processo di analisi dei dati ci consentirà di ottenere una comprensione più profonda della struttura dei dati, delle relazioni tra le variabili e dei pattern emergenti. Attraverso una combinazione di visualizzazioni, statistiche descrittive e analisi dei modelli, esploreremo le caratteristiche dei nodi e degli archi, la distribuzione delle etichette di classe, le tendenze temporali e altro ancora. L'obiettivo è quello di acquisire una panoramica completa del dataset che ci consenta di prendere decisioni informate durante le fasi successive dell'analisi e della modellazione dei dati.

3.2 Implementazione

Abbiamo condotto le nostre analisi iniziando a raggruppare le transazioni in base alle categorie di classe e contando il numero di transazioni per ciascuna classe, ottenendo una panoramica chiara e immediata della distribuzione dei dati. Questo ci consente di identificare le classi prevalenti nel dataset e di comprendere meglio la proporzione di transazioni licite, illecite e sconosciute. Le etichette per le classi forniscono ulteriori informazioni sulle categorie di transazioni che stiamo esaminando.

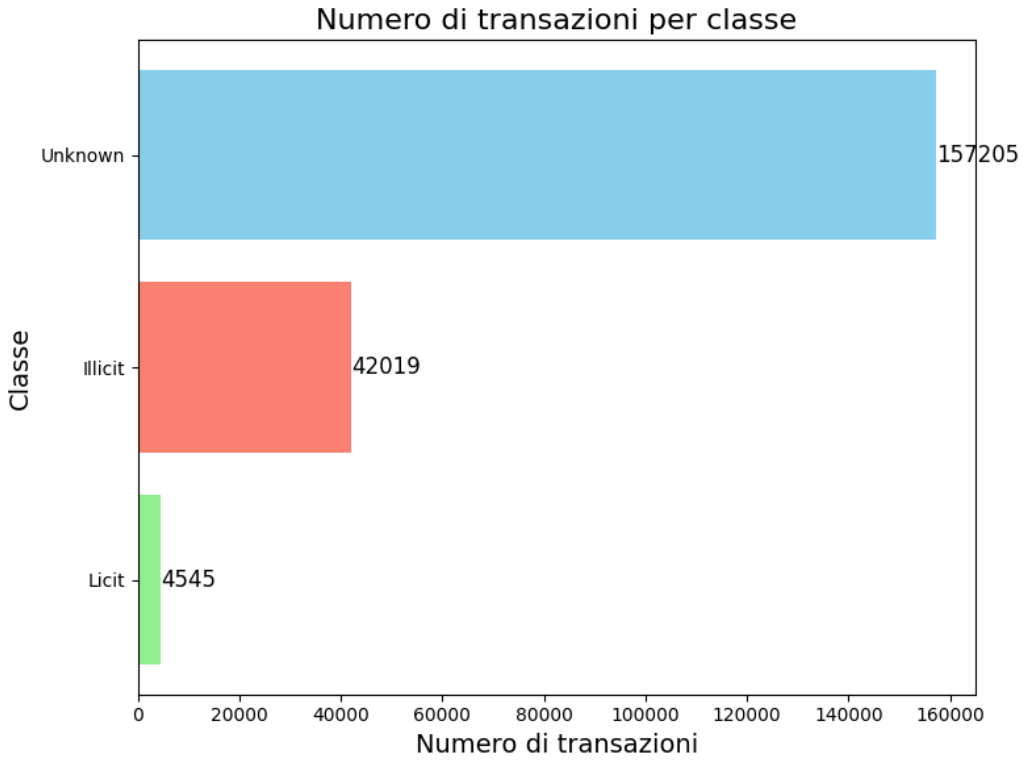

Figura 1: Numero di transazioni per classe

Successivamente abbiamo rappresentato il numero di transazioni per ciascun "Time Step", cioè per ciascun intervallo temporale nel dataset:

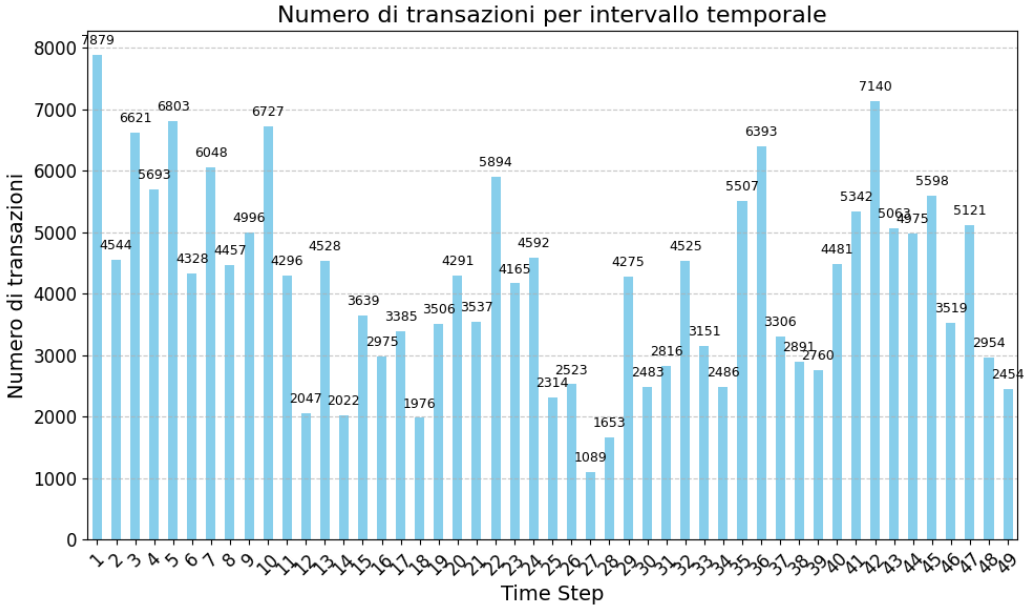

Figura 2: Numero di transazioni per TimeStep

Il grafico utilizza un diagramma a barre per mostrare il conteggio delle transazioni per ogni intervallo temporale. Ogni barra rappresenta un time step e l'altezza della barra corrisponde al numero di transazioni in quel periodo di tempo. Sopra ogni barra è stata aggiunta un'etichetta numerica che rappresenta il conteggio esatto delle transazioni per quel time step. Questo fornisce un modo diretto per interpretare i dati e comprendere la distribuzione delle transazioni nel corso del tempo.

Per uno studio più approfondito abbiamo deciso di utilizzare la matrice di correlazione per esaminare le relazioni tra le caratteristiche nel dataset delle transazioni Bitcoin. Poiché ogni transazione è descritta da numerose caratteristiche (ad esempio, volume di output, tasso di transazione, numero di input/output, ecc.), la matrice di correlazione ci permette di visualizzare rapidamente se e in che misura queste caratteristiche sono correlate tra loro.

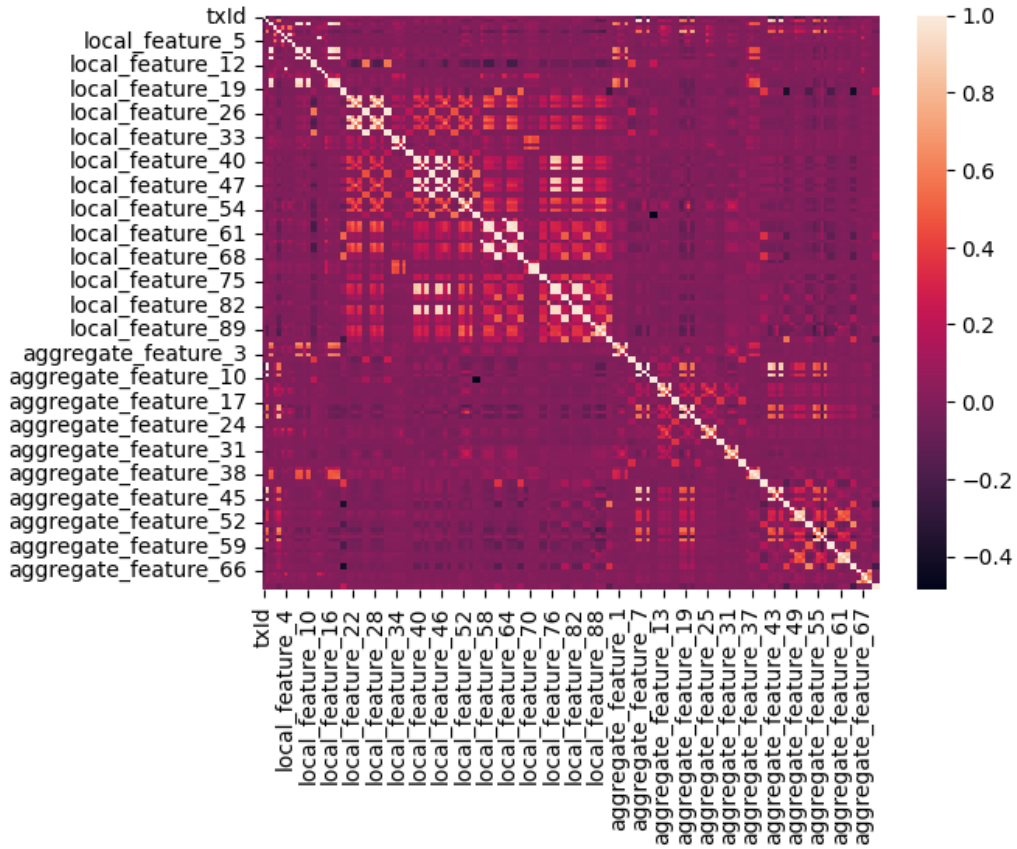

Figura 3: Matrice di correlazione

In seguito abbiamo combinato le informazioni per poter migliorare le analisi e abbiamo deciso di rappresentare il numero di transazioni per intervallo temporale per ogni classe

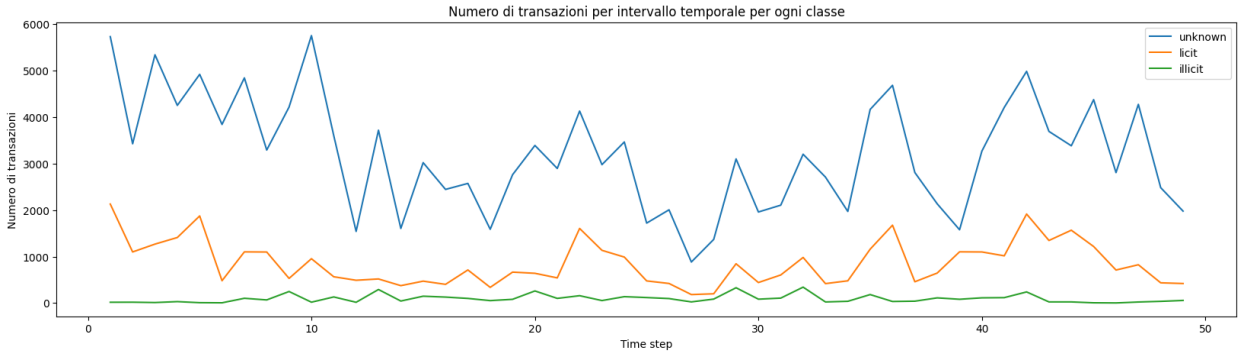

Figura 4: Transizioni per ogni classe

Grazie alla visualizzazione del grafico ci siamo resi conto del livello di attività sospetta e legittima nel corso del tempo nel dataset delle transazioni Bitcoin. Abbiamo esaminato ogni periodo di tempo e contato quante transazioni sono state etichettate come illegali e quante come legali operando nel seguente modo:

- Calcolo del numero di transazioni per ciascun timestamp
- Trova i timestamp estremi
- Visualizzazione dei dati

Alla fine, mostriamo i risultati, indicando i momenti in cui ci sono state più e meno transazioni illegali e legali.

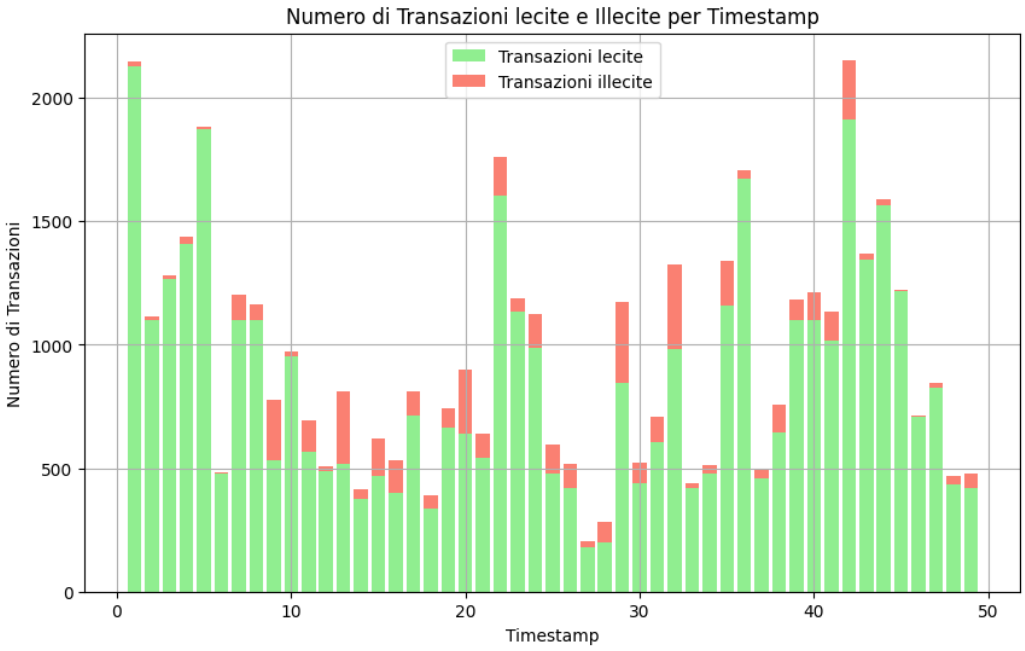

Figura 5: Numero di transazioni per Timestamp

Ogni barra rappresenta un timestamp e mostra il numero totale di transazioni effettuate in quel periodo. Le barre sono divise in due colori: il verde chiaro rappresenta le transazioni lecite e il rosso chiaro rappresenta le transazioni illecite. Questo grafico consente di confrontare visivamente la distribuzione delle transazioni lecite e illecite per ogni intervallo temporale. Possono essere identificati facilmente i periodi con un numero significativamente maggiore di transazioni illecite rispetto a quelle lecite, e viceversa.

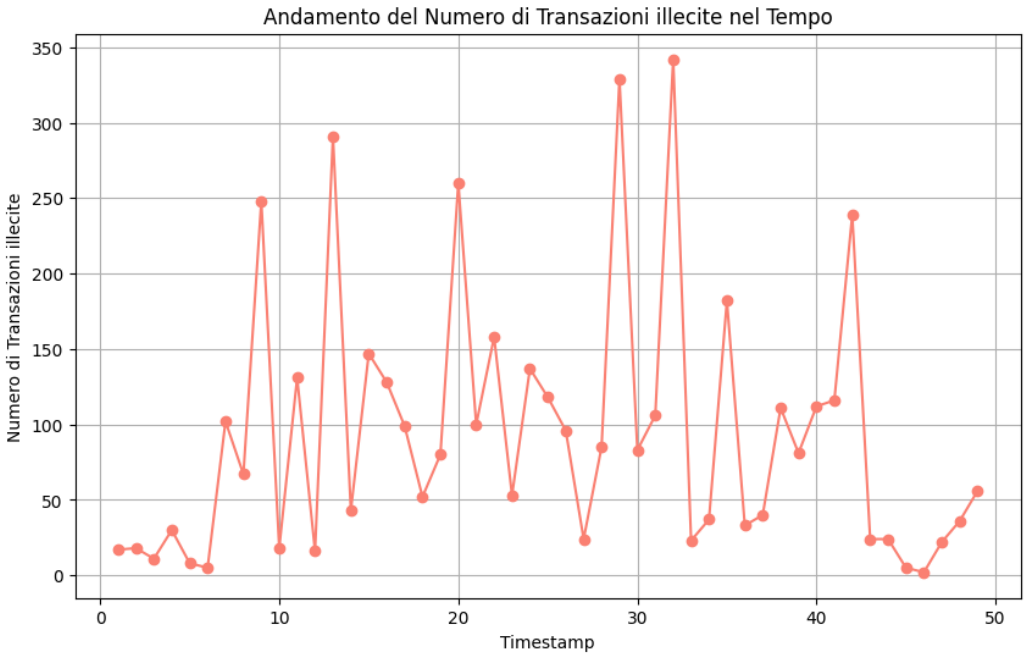

Figura 6: Andamento Numero di transazioni illecite nel tempo

Questo grafico mostra l'andamento temporale del numero di transazioni illecite. Ogni punto sulla linea rappresenta un timestamp e corrisponde al numero di transazioni illecite in quel periodo. Questo grafico fornisce una visualizzazione chiara delle variazioni nel numero di transazioni illecite nel corso del tempo. È possibile identificare trend temporali, picchi o cali nel numero di transazioni illecite nel corso del tempo.

In conclusione abbiamo realizzato altri due grafici:

- **Grafico delle transazioni illecite al passo temporale specifico**. Ogni nodo nel grafico rappresenta una transazione illecita, e i collegamenti tra i nodi indicano il flusso di Bitcoin tra le transazioni. I nodi sono colorati in rosso chiaro per distinguere le transazioni illecite. Questo grafico offre una visualizzazione immediata delle transazioni sospette o illegali che si verificano in un dato momento.

- **Grafico delle transazioni lecite al passo temporale specifico**. I nodi nel grafico rappresentano le transazioni lecite, con i collegamenti che indicano il flusso di Bitcoin tra di esse. I nodi sono colorati in verde chiaro per identificare le transazioni lecite. Questo grafico fornisce una visione delle transazioni legittime che si verificano contemporaneamente alle transazioni illecite.

Per confrontare direttamente le transazioni illecite e lecite che si verificano contemporaneamente, offrendo una panoramica completa delle attività sospette e legittime abbiamo combinato i due grafici

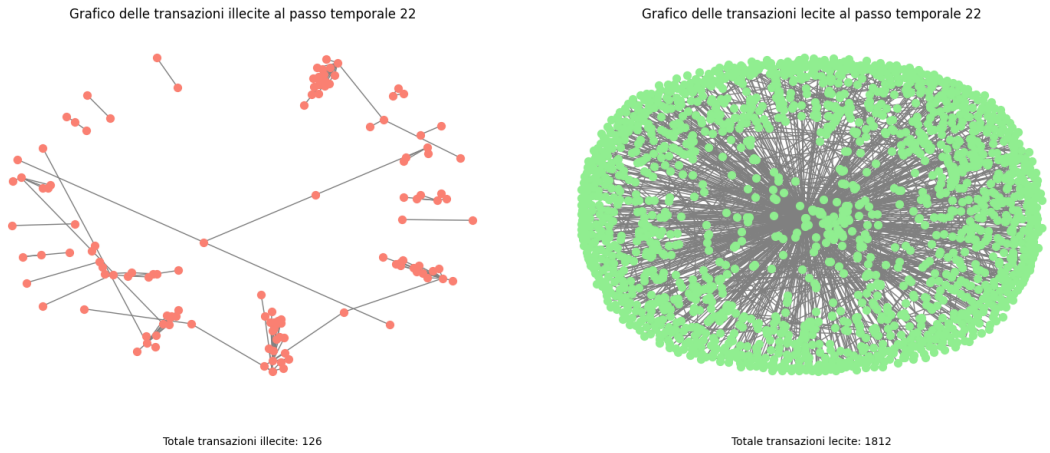

Figura 7: Grafico combinato delle transazioni illecite e lecite

4 Implementazione

In questa sezione illustriamo le nostre scelte implementative utilizzate per la rigenerazione delle caratteristiche, approfondendo anche il processo di addestramento del modello.

4.1 Rete.ipynb

Questo script si occupa di preparare i dati per l'analisi e l'addestramento aggiungendo uteriori caratteristiche.

Dopo aver creato il grafo, abbiamo introdotto nuove misure di centralità. Queste misure forniscono ulteriori informazioni sull'importanza e sull'influenza dei nodi all'interno della rete. In particolare:

- **Degree Centrality**: Valuta il numero di archi che connettono un nodo agli altri.
- **Closeness Centrality**: Misura la vicinanza di un nodo agli altri nodi.
- **Betweenness Centrality**: Valuta il ruolo di un nodo nei percorsi più brevi tra gli altri nodi.
- **Eigenvector Centrality**: Valuta l'importanza di un nodo in base ai suoi collegamenti con nodi importanti.
- **Pagerank Centrality**: Valuta la centralità di un nodo basandosi sui suoi collegamenti in entrata.
- **Harmonic Centrality**: Misura la velocità con cui un nodo può raggiungere gli altri nodi nella rete.

Successivamente, abbiamo esteso la nostra analisi includendo misure di clustering, che ci permettono di comprendere meglio la struttura della rete e l'agglomeramento dei nodi in gruppi densamente connessi. Nello specifico:

- **Clustering Coefficient**: è una misura della propensione dei suoi vicini ad essere collegati tra loro. Indica la probabilità che i vicini di un nodo siano anche tra loro connessi. Un alto clustering coefficient per un nodo indica che i suoi vicini tendono a formare gruppi o cluster.
- **Local Clustering Coefficient**:è una media dei clustering coefficient dei suoi vicini diretti. Questa misura fornisce una stima della densità di connessioni tra i vicini diretti di un nodo.

Le misure di centralità e di clustering vengono aggiunte come caratteristiche ai nodi del grafo, arricchendo così i dati e rendendo il modello di machine learning più informativo durante l'addestramento. Le caratteristiche estratte vengono organizzate in un DataFrame pandas, adatto per l'addestramento del modello. Le colonne vengono convertite nel formato corretto per un'elaborazione ottimale. Infine, i dati preparati vengono salvati in nuovi file CSV nella stessa cartella per l'utilizzo futuro nell'addestramento e nella valutazione del modello.

4.2 Addestramento del modello

Il dataset di addestramento è stato diviso utilizzando il 65% di tutti i campioni per scopi di addestramento utilizzando gli altri 15% e il 20% per scopi di convalida e test rispettivamente. È anche importante sottolineare che è garantito che le proporzioni delle etichette siano mantenute lungo le divisioni, evitando così un potenziamento ingiustificato dei risultati.
Dallo studio precedente sono stati condotti sei esperimenti rappresentativi dell'approccio **GNN** (Graph Neural Network): GCN (Graph Convolutional

Network), GAT (Graph Attention Network), GATv2, GAT personalizzato, rete di Chebyshev e rete GraphSage. Per tutti i modelli esaminati sono stati utilizzati gli stessi iperparametri:

- **Binary Cross Entropy**come funzione di perdita;
- Ottimizzatore **Adam** con un tasso di apprendimento di 1e-3;
- **4000** epoche.

4.2.1 Metriche

Le metriche di valutazione sono fondamentali per valutare l'efficacia dei modelli nella classificazione delle transazioni illecite e per comprendere come si comportano in termini di precisione, completezza e bilanciamento tra i due. La scelta di utilizzare le stesse metriche del paper originale consente una valutazione diretta e comparativa dei risultati ottenuti.

- **Precision**: si concentra sugli errori di Falsi Positivi (in questo caso transazioni lecite classificate come illecite). La precisione è efficace in situazioni in cui è importante evitare i falsi positivi.
- **Recall**: si concentra invece sugli errori di Falsi Negativi, fornendo un'indicazione delle previsioni positive non identificate correttamente, ovvero transazioni illecite classificate erroneamente come lecite.
- **F1-score**: cerca di trovare un equilibrio tra *precision* e *recall* calcolando la loro media armonica. Questo punteggio è utile per valutare le prestazioni complessive del modello, tenendo conto sia degli errori di falsi positivi che di falsi negativi.
- **Micro-avg F1 score**: è incluso per completezza e indica la proporzione di osservazioni correttamente classificate rispetto a tutte le osser-

vazioni (sia lecite che illecite). Questo punteggio fornisce una visione generale della capacità predittiva del modello su tutto il dataset.

5 Risultati

Di seguito vengono riportati i risultati dei test eseguiti.

I modelli utilizzati sul dataset che include solo le caratteristiche aggiuntive derivanti dal grafo producono prestazioni scarse. Ciò è attribuibile al fatto che, concentrandosi sulla classe "illecita" (0), i modelli non riescono ad identificare correttamente le istanze positive.

All'opposoto i modelli mostrano un'elevata capacità di classificazione per la classe "lecita" (1), ottenendo risultati eccellenti.

Tuttavia, ciò è influenzato anche dallo sbilanciamento del dataset, in cui le classi sono distribuite in modo disuguale, con solo il 2% delle istanze illegali e il 21% delle istanze legali.

In generale, viene un risultato di 0.9 per quanto riguarda "F1 micro avg", questo influenzato dai risultati sul target 1.

Tabella 1: Graph Features (Target 1)

Model	***Precision***	**Recall**	**f1**	**f1 micro avg**
GCN Convolution	0.904	1.0	0.95	0.904
GAT Convolution	0.904	1.0	0.95	0.904
SAGE Convolution	0.904	1.0	0.95	0.904
Chebyshev Convolution	0.904	1.0	0.95	0.904
GATv2 Convolution	0.904	1.0	0.95	0.904

In aggiunta, abbiamo condotto numerosi test integrando le nostre caratteristiche derivate dal grafo al set di base di Elliptic. Questo ha portato a un

Tabella 2: Graph Features (Target 0)

Model	***Precision***	**Recall**	**f1**	**f1 micro avg**
GCN Convolution	0.0	0.0	0.0	0.904
GAT Convolution	0.0	0.0	0.0	0.904
SAGE Convolution	0.0	0.0	0.0	0.904
Chebyshev Convolution	0.0	0.0	0.0	0.904
GATv2 Convolution	0.0	0.0	0.0	0.904

aumento del rumore nei vari modelli e, di conseguenza, a una diminuzione delle prestazioni in quasi tutti i casi.

Attraverso correlazione si sono testati i modelli eliminando le features alatamente correlate(sia le coppie sia la singola features), impostando una soglia di 0.9. Anche questi ultimi test hanno portato aggiunta di rumore aggiungendo le features relative al grafo.

Tabella 3: Features Base + graph features

Model	***Precision***	**Recall**	**f1**	**f1 micro avg**
GCN Convolution (tx)	0.952	0.312	0.469	0.933
GCN Convolution (tx+agg)	0.884	0.385	0.536	0.936
GAT Convolution (tx)	0.776	0.258	0.388	0.922
GAT Convolution (tx+agg)	0.83	0.712	0.766	0.958
SAGE Convolution (tx)	0.949	0.711	0.813	0.969
SAGE Convolution (tx+agg)	0.94	0.777	0.851	0.974
Chebyshev Convolution (tx)	0.951	0.752	0.84	0.973
Chebyshev Convolution (tx+agg)	0.946	0.828	0.833	0.979
GATv2 Convolution (tx)	0.865	0.756	0.807	0.966
GATv2 Convolution (tx+agg)	0.895	0.857	0.875	0.977

Tabella 4: Feature Base corr < 0.9 (escludendo la coppia)

Model	***Precision***	**Recall**	**f1**	**f1 micro avg**
GCN Convolution (tx)	0.842	0.446	0.583	0.939
GCN Convolution (tx+agg)	0.82	0.475	0.602	0.94
GAT Convolution (tx)	0.826	0.663	0.735	0.955
GAT Convolution (tx+agg)	0.869	0.696	0.773	0.961
SAGE Convolution (tx)	0.947	0.764	0.846	0.973
SAGE Convolution (tx+agg)	0.941	0.823	0.878	0.978
Chebyshev Convolution (tx)	0.959	0.772	0.856	0.975
Chebyshev Convolution (tx+agg)	0.949	0.839	0.89	0.98
GATv2 Convolution (tx)	0.901	0.8	0.848	0.973
GATv2 Convolution (tx+agg)	0.886	0.852	0.869	0.975

Tabella 5: Feature Base corr < 0.9 (escludendo la coppia) + graph features

Model	***Precision***	**Recall**	**f1**	**f1 micro avg**
GCN Convolution (tx)	0.941	0.308	0.464	0.932
GCN Convolution (tx+agg)	0.883	0.322	0.472	0.931
GAT Convolution (tx)	0.787	0.466	0.585	0.937
GAT Convolution (tx+agg)	0.741	0.615	0.672	0.943
SAGE Convolution (tx)	0.95	0.736	0.83	0.971
SAGE Convolution (tx+agg)	0.929	0.766	0.84	0.972
Chebyshev Convolution (tx)	0.948	0.765	0.847	0.974
Chebyshev Convolution (tx+agg)	0.932	0.804	0.863	0.976
GATv2 Convolution (tx)	0.786	0.77	0.778	0.958
GATv2 Convolution (tx+agg)	0.906	0.785	0.841	0.972

Tabella 6: Feature Base corr < 0.9 (escludendo la singola feature)

Model	***Precision***	**Recall**	**f1**	**f1 micro avg**
GCN Convolution (tx)	0.846	0.447	0.585	0.940
GCN Convolution (tx+agg)	0.806	0.503	0.620	0.941
GAT Convolution (tx)	0.832	0.661	0.737	0.955
GAT Convolution (tx+agg)	0.821	0.730	0.773	0.959
SAGE Convolution (tx)	0.950	0.769	0.850	0.974
SAGE Convolution (tx+agg)	0.953	0.824	0.884	0.979
Chebyshev Convolution (tx)	0.958	0.765	0.851	0.974
Chebyshev Convolution (tx+agg)	0.957	0.855	0.903	0.982
GATv2 Convolution (tx)	0.884	0.801	0.841	0.971
GATv2 Convolution (tx+agg)	0.893	0.864	0.879	0.977

Tabella 7: Feature Base corr < 0.9 (escludendo la singola feature) + graph features

Model Modello	***Precision*** Precision	**Recall** Recall	**f1** F1-score	**f1 micro avg** Accuracy
GCN Convolution (tx)	0.953	0.278	0.430	0.930
GCN Convolution (tx+agg)	0.847	0.436	0.576	0.938
GAT Convolution (tx)	0.804	0.462	0.586	0.938
GAT Convolution (tx+agg)	0.753	0.626	0.683	0.944
SAGE Convolution (tx)	0.934	0.721	0.814	0.969
SAGE Convolution (tx+agg)	0.936	0.783	0.852	0.974
Chebyshev Convolution (tx)	0.960	0.749	0.842	0.973
Chebyshev Convolution (tx+agg)	0.944	0.835	0.886	0.979
GATv2 Convolution (tx)	0.881	0.754	0.813	0.967
GATv2 Convolution (tx+agg)	0.901	0.830	0.864	0.975

6 Conclusioni

I risultati dei nostri esperimenti sulla rete non hanno raggiunto un livello di miglioramento rispetto ai lavori precedenti; al contrario, hanno evidenziato una diminuzione delle prestazioni. Tuttavia, sono emerse diverse soluzioni potenziali per migliorare questa situazione. Una di queste consiste nel bilanciare ulteriormente il dataset, al fine di ridurre l'effetto dello sbilanciamento sulle performance dei modelli. Inoltre, potremmo esaminare altre metriche sui grafi al fine di ottenere una valutazione più completa delle caratteristiche delle reti e delle loro relazioni.

Riferimenti bibliografici

[1] Progetto Simone Marasi, Documentazione

[2] Eliptic, Dataset

[3] M Weber et al., , "Anti-Money Laundering in Bitcoin: Experimenting with Graph Convolutional Networks for Financial Forensics"in ArXiv, 2019