Détection de Fraude Financière par Graphes

Projet Académique ECE - Groupe 42

Malak El Idrissi & **Joe Boueri**

Intelligence Artificielle & Finances - 2026

Introduction

Contexte de la Fraude Financière

- **Volume croissant** des transactions financières numériques
- **Complexité accrue** des schémas de fraude
- **Impact économique** : milliards d'euros perdus annuellement
- **Réglementation stricte** : AML/CFT (Anti-Money Laundering / Combating the Financing of Terrorism)

Enjeux de la Détection

Problématique

Pourquoi les Graphes ?

Les approches traditionnelles basées sur les règles et les statistiques présentent des limites :

Approche Traditionnelle	Approche par Graphes
Analyse transaction par transaction	Analyse des relations entre entités
Détection de patterns simples	Détection de structures complexes

Objectifs du Projet

Trois Types de Fraude à Détecter

1. Cycles de Blanchiment

- Boucles de transferts masquant l'origine des fonds
- Retour aux sources après plusieurs transactions

2. Smurfing / Schtroumpfage

- Fractionnements de montants vers un compte pivot
- Évitement des seuils de déclaration

Cycles de Blanchiment

Définition

Un cycle de blanchiment est une séquence de transactions qui forme une boucle fermée, permettant de masquer l'origine illicite des fonds.

```
A → B → C → D → A
```

Caractéristiques

- **Boucle fermée** : le dernier transfert revient à l'expéditeur initial
- **Complexité variable** : de 3 à N nœuds

Smurfing / Schtroumpfage

Définition

Technique consistant à fractionner de grosses sommes en multiples petits montants transférés vers un compte pivot, pour éviter les seuils de déclaration.

Caractéristiques

- **Fractionnement** : montants < seuil réglementaire
- **Compte pivot** : collecte des fonds fractionnés
- **Multiples sources** : plusieurs comptes émetteurs

Anomalies de Réseaux

Définition

Comportements atypiques dans la structure des transactions qui dévient des patterns normaux d'activité financière.

Types d'Anomalies

Centralité Anormale

- Nœuds avec un degré de connexion inhabituel
- Hubs artificiels créés pour la fraude

Approche Algorithmique

Algorithmes Implémentés

1. Détection de Cycles - Algorithme de Johnson

- **Complexité** : O((V + E)(c + 1)) où c = nombre de cycles
- **Avantages** : efficace pour graphes de taille moyenne
- **Application** : identification des boucles de blanchiment

2. Détection de Communautés - Algorithme de Louvain

Architecture Technique

Stack Technologique

Langage Principal

- **Python 3.9+** : langage de référence pour la data science

Bibliothèques Graphes

- **NetworkX** : création, manipulation et analyse de graphes
- **igraph** : algorithmes de graphes performants (optionnel)

Traitement de Données

Implémentation

Structure du Code

```
src/
├── fraud_detector.py    # Moteur de détection principal
│   ├── FraudDetector     # Classe principale
│   ├── detect_cycles()   # Détection de cycles
│   ├── detect_smurfing() # Détection de smurfing
│   └── detect_anomalies()# Détection d'anomalies
│
└── utils.py              # Fonctions utilitaires
    ├── load_data()       # Chargement des données
    ├── build_graph()     # Construction du graphe
    └── visualize()       # Visualisation
```

Résultats

Exemples de Détection

Cycles Détectés

- **Nombre moyen** : 5-15 cycles par dataset de test
- **Longueur** : 3 à 7 nœuds principalement
- **Précision** : > 85% sur données synthétiques

Smurfing Identifié

- **Seuil de détection** : 5+ transactions fractionnées

Conclusion

Résumé du Projet

✅ **Détection de cycles** : Algorithme de Johnson implémenté avec succès
✅ **Détection de smurfing** : Identification des fractionnements suspects
✅ **Anomalies de réseaux** : Analyse de centralité et communautés

Perspectives

Améliorations Futures

Questions ?

Merci de votre attention

Malak El Idrissi & Joe Boueri
ECE - Intelligence Artificielle & Finances - 2026