Fraud Detection by Dense Subgraph Detection

Tong Zhao
Why dense subgraphs?

• Graph-based fraud detection
  • Unsupervised learning.
  • Unexpecting high density is suspicious.
Why dense subgraphs?

• Graph-based fraud detection
  • Unsupervised learning.
  • Unexpected high density is suspicious.
  • Fraudsters’ avoiding effort makes it dense.

follower seller
Hardworking follower seller
Dense Subgraph Detection

• Given a graph \( G = (V, E) \) with vertices \( V \) and edges \( E \subseteq V \times V \).
• Find a subgraph \( S \) such that \( d(S) \) is maximized.
• Edge density (average degree): \( d(S) = \frac{|E(S)|}{|S|} \)
  • The larger, the better.
  • The denser, the better.
Charikar’s greedy algorithm (2000) [1]

Figure from [3].
Fraudar [2] (Based on Charikar’s algorithm)

**Require:** Bipartite $G = (\mathcal{U} \cup \mathcal{W}, E)$; density metric $g$ of the form in (1)

1. procedure FRAUDAR $(G, g)$
2. Construct priority tree $T$ from $\mathcal{U} \cup \mathcal{W}$
3. $\mathcal{X}_0 \leftarrow \mathcal{U} \cup \mathcal{W}$
4. for $t = 1, \ldots, (m + n)$ do
5. $i^* \leftarrow \text{arg max}_{i \in \mathcal{X}_i} g(\mathcal{X}_i \setminus \{i\})$
6. Update priorities in $T$ for all neighbors of $i^*$
7. $\mathcal{X}_t \leftarrow \mathcal{X}_{t-1} \setminus \{i^*\}$
8. end for
9. return $\text{arg max}_{\mathcal{X}_i \in \{\mathcal{X}_0, \ldots, \mathcal{X}_{m+n}\}} g(\mathcal{X}_i)$
10. end procedure

Total runtime: $O((|V| + |E|) \log |V|)$
Implementation

• Fraudar’s source code.
• Written in Python.
• About 300 lines.
• Graphs stored in sparse matrix by SciPy.
Dataset

• Graphs generated by the provided graph generator. [4]
  • Fixed average degree as 20.
  • Changed # of vertices.
• Twitter dataset with 41.7 million users and 1.47 billion follows.
  • Failed.
Performance

• Density of the result is theoretically guaranteed.
  • Charikar’s algorithm is a provable 2-approximation algorithm.

\[ d(S') \geq \frac{1}{2} d(S_{opt}) \]

• \( S' \) denotes the result subgraph by Charikar’s algorithm.
• \( S_{opt} \) denotes the optimal solution.
Performance
Future plan

• Apply Charikar’s algorithm on larger graphs.
• Dense subgraph detection for dynamic graphs.
References


