TellTail: Fast Scoring and Detection of Dense Subgraphs

Bryan Hooi,1 Kijung Shin,2 Hemank Lamba,3 Christos Faloutsos3

1National University of Singapore
2KAIST
3Carnegie Mellon University

bhooi@comp.nus.edu.sg, kijungs@kaist.ac.kr, hlambda@cs.cmu.edu, christos@cs.cmu.edu

Proof of Safe Pruning (Theorem 1)

Let \( S^* = \arg \max_{S \subseteq \mathcal{V}} \text{TELLTAIL}(S) \) and \( s = \lfloor n/2 \rfloor \). Define:

\[
\Delta(S) = (s^\beta - (s-1)^\beta) \text{TELLTAIL}(S) + \mu_0(s^\alpha - (s-1)^\alpha)
\]

Theorem 1 (Safe Pruning). For any node \( i \), if \( i \in S^* \), we have:

\[
\text{Dev}_i \geq \Delta(S^*) \geq \Delta(S) \quad \forall \quad S \subseteq \mathcal{V}
\]

Thus, we can prune nodes with deviation less than \( \Delta(S) \), for any \( S \).

Proof. Define \( k_s = |S^*| \), and \( \gamma = k_s^\alpha - (k_s - 1)^\alpha \), and \( \delta = k_s^\beta - (k_s - 1)^\beta \).

Since \( \text{Dev}_i \) is the sum of positive modularity terms along the \( i \)th row of \( B \), \( \text{Dev}_i \) is an upper bound for how much adjusted mass the \( i \)th row can contribute to \( \hat{\varepsilon}(S^*) \). Hence:

\[
\text{TELLTAIL}(S^*) \geq \frac{\hat{\varepsilon}(S^*) + \mu_0 \cdot k_s^\alpha}{k_s^\beta - \delta} \geq \frac{\hat{\varepsilon}(S^*) - \text{Dev}_i - \mu_0 \cdot (k_s - 1)^\alpha}{(k_s - 1)^\beta}
\]

\[
\Longrightarrow \frac{\hat{\varepsilon}(S^*) - \text{Dev}_i - \mu_0 \cdot (k_s - 1)^\alpha}{(k_s - 1)^\beta} \geq \frac{\hat{\varepsilon}(S^*) - \text{Dev}_i - \mu_0 \cdot k_s^\alpha + \mu_0 \cdot \gamma}{k_s^\beta - \delta}
\]

\[
\Longrightarrow -\delta \cdot \hat{\varepsilon}(S^*) + \mu_0 \cdot \delta \cdot k_s^\alpha \geq -k_s^\beta \cdot \text{Dev}_i + k_s^\beta \cdot \mu_0 \cdot \gamma
\]

\[
\Rightarrow \text{TELLTAIL}(S^*) \leq \frac{\text{Dev}_i - \mu_0 \gamma}{\delta}
\]

\[
\Rightarrow \text{Dev}_i \geq \delta \cdot \text{TELLTAIL}(S^*) + \mu_0 \cdot \gamma
\]

\[
\Rightarrow \text{Dev}_i \geq \Delta(S^*)
\]

The second inequality \( \Delta(S^*) \geq \Delta(S) \) \( \forall \quad S \subseteq \mathcal{V} \) follows from \( \text{TELLTAIL}(S^*) \geq \text{TELLTAIL}(S) \), since this implies that \( \Delta(S^*) \geq \Delta(S) \) \( \forall \quad S \subseteq \mathcal{V} \).

Proof of Consistency (Theorem 2)

Theorem 2 (Consistency). Let \( G \) be a graph drawn from \( \mathcal{G} \), and fix \( k \). Define any fixed \( z > 0 \), and let \( y_n = -\sigma(\hat{\mu}_n)(1 - z)/\xi \). Then as \( n \to \infty \), there exists a sequence\(^1\) of \( N_n \to \infty \), \( \epsilon_n \to 0 \) such that:

\[
\frac{1 - \hat{F}(\hat{\mu}_n + y_n)}{1 - \hat{F}(\hat{\mu}_n + y_n)} \to 1
\]

where \( \to \) denotes convergence in probability.

Fix \( N \) and let \( n \to \infty \). Consider an \( n \)-node graph \( G \sim \mathcal{G} \) and random subsets \( S_1, \ldots, S_N \) of size \( k \). For any \( i,j \), the probability that \( S_i \) and \( S_j \) are completely disjoint is \( 2^{-k} \) \( \to 1 \) as \( n \to \infty \) (recall that \( k \) is fixed). Extending this to all of the pairs of \( i,j \) by union bound, the probability that all of \( S_1, \ldots, S_N \) are disjoint goes to \( 1 \) as \( n \to \infty \). This implies that with high probability, \( S_1, \ldots, S_N \) are disjoint and hence are i.i.d. samples of size \( k \) from \( \mathcal{G} \), and their masses (denoted by \( m_1, \ldots, m_N \)) are i.i.d. samples from \( F \).

Note that the event in which \( S_1, \ldots, S_N \) are non-disjoint has probability converging to zero, and since the statement we want to prove is about convergence in probability, this event can be ignored.

At this point, \( m_1, \ldots, m_N \) is an i.i.d. sample from a distribution \( F \), and our algorithm TAIL estimates a GP distribution using maximum likelihood from this sample. (Smith 1987) shows that under these conditions, the maximum likelihood procedure produces consistent estimators of the tail probabilities of the distribution \( F \). Formally:

\[
1 - \hat{F}(\hat{\mu}_n + y_n) \to 1
\]

i.e. \( \hat{F} \) converges to \( F \), measured in terms of relative error with respect to the CCDF \( 1 - F \).

TellTail: Estimator for \( \mu_0 \)

Let \( A \) be the adjacency matrix, \( d \) be the column vector of node degrees, \( B = A - d \cdot d^T / (2n) \), and \( s = \lfloor n/2 \rfloor \). Then

\(^1\)\( N_n, \epsilon_n, \hat{\mu}_n \) denote the original variables \( (N, \epsilon, \hat{\mu}) \) indexed over runs corresponding to different values of \( n \).
we will show that the following provides a reasonable approximation for $\mu_0$, based on a Central Limit Theorem-based approximation:

$$
\hat{\mu}_0 = s^{-\alpha}(p_2S_1 + z_{1-\epsilon}(p_2S_2 + p_3(S_3 - 2S_2)) + p_4(S_2^2 + 2S_2 - 3S_2) - (p_2S_2)^2)^{1/2}
$$

(10)

where: $S_1 = \sum_{i<j}B_{ij}$, $S_2 = \sum_{i<j}B_{ij}^2$, $S_3 = \sum_{i=1}^n(\sum_{j=1}^n B_{ij})^2$, $p_e = \prod_{j=1}^{n/j}(s-j+1)/\prod_{j=1}^{n-j+1}(n-j+1)$, and $z_{1-\epsilon}$ is the $(1-\epsilon)$-quantile of a standard normal distribution.

To show this, we first give a lemma:

**Lemma 1.** The mean and variance of $\bar{e}(S)$ are:

$$
\mathbb{E}(%e(S)) = p_2S_1 + p_3(S_3 - 2S_2)
$$

(11)

+ $p_4(S_2^2 + 2S_2 - 3S_2) - (p_2S_2)^2$.

Proof. Define random variables $Z_{ij} = B_{ij}1\{i \in S, j \in S\}$.

Note that $\bar{e}(S) = \sum_{i<j}Z_{ij}$, substituting this into $\mathbb{E}\bar{e}(S)$ and expanding with further computation gives the result.

We now derive our estimator (10) for $\mu_0$. Consider subgraphs of size $s = [n/2]$. For subgraphs $S$ of this large size, $\bar{e}(S)$ can be expressed as a sum over individual edges:

$\bar{e}(S) = \sum_{i<j}Z_{ij}$, which recalling the Central Limit Theorem, suggests approximating $\bar{e}(S)$ with a normal distribution. From our initial definition of the GP distribution and parameters, $\mu(s)$ is the minimum value in the GP distribution’s support. Since we threshold the data at its intercept $\mu$, $\bar{e}(S)$ is the $(1-\epsilon)$-quantile of the subgraph mass distribution. For $s = [n/2]$, this distribution is approximately Gaussian, so the corresponding $(1-\epsilon)$-quantile is:

$$
\mu(n/2) \approx \mathbb{E}(\hat{e}(S)) + z_{1-\epsilon}\sqrt{\text{Var}(\hat{e}(S))}
$$

(12)

The power-law plot for $\mu$ passes through the point $(s, \mu(s))$, so substituting into the Dense Subgraph Power Law, its intercept is $\mu_0 = \mu(s)/(s)^\alpha$. Combining this with (11) gives the final result.

**Computing $S_1, S_2, S_3$**

Recall that $S_1 = \sum_{i<j}B_{ij}$, $S_2 = \sum_{i<j}B_{ij}^2$, $S_3 = \sum_{i=1}^n(\sum_{j=1}^n B_{ij})^2$.

Computing $S_1$ to $S_3$ naively is $O(n^2)$, by this can be sped up to linear time using matrix operations.

**Lemma 2.** $S_1$ to $S_3$ can be computed in $O(m)$ time.

Proof. However, $B = A - d \cdot d^T/(2m)$, a sum of a sparse and a low rank matrix. This substitution allows us to rewrite the expressions for $S_1$ to $S_3$:

$$
S_1 = \frac{\sum_{i=1}^n d_i^2}{4m}
$$

(13)

$$
S_2 = \sum_{i<j}A_{ij}^2 - \frac{d^TAd}{2m} + \frac{(\sum_{i=1}^n d_i^2)^2 - \sum_{i=1}^n d_i^4}{8m}
$$

(14)

$$
S_3 = \sum_{i=1}^n d_i^4
$$

(15)

The only terms that require matrix operations are $\sum_{i<j}A_{ij}^2$ and $d^TAd$. Both can be computed in $O(m)$ time using standard sparse matrix operations.

**Subgraph Mass Distribution Plots in Real Data**

Figure 1 plots the empirical distribution of subgraph masses for all 8 of our original datasets as plotted in Table III of the paper, and all 5 of the Twitter subset graphs.

**Proof of NP Completeness**

In this section, we show that maximizing TellTail is NP-complete. Let $\hat{e}_G(S)$ denote the adjusted mass of subset $S$ with respect to the graph $G$, i.e. $\hat{e}_G(S) = e(S) - d(S)^2/(4m)$, all with respect to $G$. Define the following two problems:

**Problem 1 (Modularity).** Given a graph $G$, does there exist a subset $S$ of its nodes such that $\hat{e}_G(S) > 0$?

The NP-hardness of modularity maximization was first shown by (Brandes et al. 2007), though this differs from the formulation here. The NP-hardness of this exact formulation was shown by (Dinh, Li, and Thai 2015). They do this by reducing the known NP-hard PARTITION problem, of partitioning $n$ integers $x_1, \ldots, x_n$ into two subsets of equal sum, to the MODULARITY problem. To do this, they show that given $x_1, \ldots, x_m$, we can construct a graph such that a subset $S$ of positive modularity exists iff there exists a partition of $x_1, \ldots, x_m$ into two halves of equal sum, which completes the reduction and establishes the NP-hardness of MODULARITY.

The problem we are interested in is:

**Problem 2 (TellTailProb).** Given a graph $G$, does there exist a subset $S$ of its nodes such that $\text{TellTail}_{G'}(S) > 0$?

We now show our main NP-completeness result:

**Theorem 3.** TellTailProb is NP-complete.

Proof. Given a subset $S$ of the nodes of $G$, we can compute TellTail$(S)$ in polynomial time. Thus, TellTailProb can be verified in polynomial time, and is therefore in NP. It remains to establish that TellTailProb is NP-hard.

We do this by reducing Modular to TellTailProb: given an algorithm $A'$ that solves TellTailProb, we show that it can be used as a subroutine in a polynomial-time algorithm $A$ to solve Modular. Then, since we know that Modular is NP-hard, this would imply that TellTailProb is NP-hard as well.

Consider an instance of Modular with graph $G$. Construct a new graph $G'$ by adding $r$ extra nodes to $G$, in which the extra nodes have no edges attached to them. Note then that for any subset $S$ of the nodes of $G$, the adjusted mass of $S$ is the same when computed with respect to either $G$ or $G'$; this is because in the formula $\hat{e}(S) = e(S) - d(S)^2/(4m)$, none of the terms $e(S), d(S)$ or $m$ differ between $G$ and $G'$.

Consider Eq. (10) for $\mu_0$. Each of the $p_i$ is at most 1, and the $S_i$ are all constant as we increase $r$, which we can verify.
The GP distribution fits the empirical distribution of subgraph masses much more closely than other distributions. Black crosses indicate the empirical distribution of subgraph masses for subgraphs of size \( k = \lfloor \sqrt{n} \rfloor \), in the form of its complementary CDF. The colored curves are the best fit GP, Poisson, and Gaussian to the empirical distribution. The Poisson curve is far to the left of the empirical distributions because Poisson distributions greatly underestimate the number of dense subgraphs that we should observe.
from Eq. (6) to (8) of the original paper. Then, since $G'$ has $n + r$ nodes, we have from Eq. (10) that $\mu_0 \leq \left(\frac{\alpha + \epsilon}{2}\right)^{-\alpha} B$, for some $B > 0$ that does not depend on $r$.

Set $r > 2(4mn^\alpha B)^{(1/\alpha)}$. Then in $G'$, we have:

$$
\mu_0 \leq \left(\frac{n + r}{2}\right)^{-\alpha} B < \left(\frac{r}{2}\right)^{-\alpha} B = (4mn^\alpha B)^{(1/\alpha) - \alpha} B = \frac{1}{4mn^\alpha}
$$

Define the algorithm $A(G)$ for \textsc{Modularity} that given $G$, constructs $G'$, runs our subroutine for solving \textsc{TellTailProb} on $G'$, and outputs $A'(G')$. We claim that $A(G)$ correctly solves \textsc{Modularity}. To show this, we consider two cases:

- **case 1:** there exists $S$ such that $\tilde{e}_G(S) > 0$. Then since $e_G'(S)$ is a fraction with an integer in the numerator and a denominator of $4m$, we have $e_G(S) \geq 1/(4m)$. Then, recalling that $e_G(S) = e_G'(S)$,

$$
\textsc{TellTail}_{G'}(S) = \frac{\tilde{e}_G(S) - \mu_0|S|^\alpha}{|S|^\beta} \geq \frac{1}{4m} - \frac{\mu_0|S|^\alpha}{|S|^\beta} \geq 0 \quad (4mn^\alpha B)^{(1/\alpha) - \alpha} B = \frac{1}{4mn^\alpha}
$$

Thus $A(G)$ outputs the correct result in this case: $A'(G')$ will return ‘true’ since $\textsc{TellTail}_{G'}(S) > 0$, so $A(G)$ will return ‘true’, which is correct since $\tilde{e}_G(S) > 0$.

- **case 2:** there does not exist such an $S$; i.e. $\tilde{e}_G(S) \leq 0$ for all $S$. Then for all $S$,

$$
\textsc{TellTail}_{G'}(S) = \frac{\tilde{e}_G(S) - \mu_0|S|^\alpha}{|S|^\beta} \leq \frac{\tilde{e}_G(S)}{|S|^\beta} \leq 0
$$

Thus $A(G)$ returns ‘false’, which is the correct result in this case as well.

In conclusion, $A(G)$ is a correct, polynomial time algorithm for \textsc{Modularity}, assuming we have a subroutine $A'$ that solves \textsc{TellTailProb}. Since \textsc{Modularity} is NP-hard by (Dinh, Li, and Thai 2015), this implies that \textsc{TellTailProb} is NP-hard as well. Since we also showed that \textsc{TellTailProb} is in NP, this implies that it is NP-complete.

### Accuracy Plots on Synthetic Data

For space reasons, accuracy results for identifying injected subgraphs on synthetic data was omitted from the main paper, and are included here in Figure 2.

### Additional Results for Baseline Methods

For space reasons, results varying $\alpha$ for the Edge Surplus method were omitted for the main paper, and are included here as Tables 1 and 2.

### References


Figure 2: **TAILF outperforms baselines in accuracy** for identifying injected subgraphs.

<table>
<thead>
<tr>
<th></th>
<th>TAILF</th>
<th>Average</th>
<th>Modularity</th>
<th>Expansion</th>
<th>Conduct.</th>
<th>Surp.(1/3)</th>
<th>Surp.(1/2)</th>
<th>Surp.(2/3)</th>
<th>Susp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Football</td>
<td>1.00</td>
<td>0.00</td>
<td>0.96</td>
<td>0.71</td>
<td>0.55</td>
<td>0.99</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PolIE</td>
<td>1.00</td>
<td>0.00</td>
<td>0.94</td>
<td>0.87</td>
<td>0.86</td>
<td>0.16</td>
<td>0.75</td>
<td>0.86</td>
<td>0.38</td>
</tr>
<tr>
<td>PolUK</td>
<td>0.96</td>
<td>0.46</td>
<td>0.87</td>
<td>0.54</td>
<td>0.87</td>
<td>0.46</td>
<td>0.71</td>
<td>0.87</td>
<td>0.29</td>
</tr>
<tr>
<td>Olympics</td>
<td>1.00</td>
<td>0.00</td>
<td>0.85</td>
<td>0.74</td>
<td>0.67</td>
<td>0.99</td>
<td>1.00</td>
<td>1.00</td>
<td>0.63</td>
</tr>
<tr>
<td>Rugby</td>
<td>0.97</td>
<td>0.00</td>
<td>0.85</td>
<td>0.90</td>
<td>0.80</td>
<td>0.97</td>
<td>0.94</td>
<td>0.94</td>
<td>0.55</td>
</tr>
</tbody>
</table>

Table 1: **TAILF outperforms baselines** in identifying ground truth communities.

<table>
<thead>
<tr>
<th></th>
<th>TELL TAIL</th>
<th>TELL TAIL+</th>
<th>Average</th>
<th>Modularity</th>
<th>Expansion</th>
<th>Conduct.</th>
<th>Surp.(1/3)</th>
<th>Surp.(1/2)</th>
<th>Surp.(2/3)</th>
<th>Susp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Football</td>
<td>0.78</td>
<td>0.67</td>
<td>0.11</td>
<td>0.19</td>
<td>0.06</td>
<td>0.11</td>
<td>0.13</td>
<td>0.15</td>
<td>0.20</td>
<td>0.13</td>
</tr>
<tr>
<td>PoliticsIE</td>
<td>0.90</td>
<td>0.82</td>
<td>0.42</td>
<td>0.39</td>
<td>0.02</td>
<td>0.41</td>
<td>0.42</td>
<td>0.41</td>
<td>0.40</td>
<td>0.39</td>
</tr>
<tr>
<td>PoliticsUK</td>
<td>0.82</td>
<td>0.84</td>
<td>0.81</td>
<td>0.75</td>
<td>0.01</td>
<td>0.36</td>
<td>0.51</td>
<td>0.53</td>
<td>0.81</td>
<td>0.76</td>
</tr>
<tr>
<td>Olympics</td>
<td>0.68</td>
<td>0.72</td>
<td>0.20</td>
<td>0.21</td>
<td>0.11</td>
<td>0.15</td>
<td>0.22</td>
<td>0.29</td>
<td>0.44</td>
<td>0.22</td>
</tr>
<tr>
<td>Rugby</td>
<td>0.47</td>
<td>0.49</td>
<td>0.43</td>
<td>0.24</td>
<td>0.36</td>
<td>0.17</td>
<td>0.25</td>
<td>0.03</td>
<td>0.03</td>
<td>0.28</td>
</tr>
</tbody>
</table>

Table 2: **TELL TAIL and TELL TAIL+ outperform baselines** in identifying ground truth communities.