EagleMine: Vision-Guided Mining in Large Graphs Supplementary Document

Wenjie Feng[1], Shenghua Liu[1], Christos Faloutsos[2], Bryan Hooi[2], Huawei Shen[1], Xueqi Cheng[1]
[1]CAS Key Laboratory of Network Data Science & Technology,
Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2]School of Computer Science, Carnegie Mellon University, PA, USA

ABSTRACT

In this supplementary document, we provide the detailed proof of the time complexity of EagleMine algorithm described in our main paper.

Let M be the number of non-empty bins in the histogram $\mathcal{H}$, and $\mathcal{C}$ be the number of clusters. We use gradient-descent to learn parameters in $DistributionFit(\cdot)$ of EagleMine algorithm, so we assume that the number of iterations is T, which is related to the differences between initial and optimal objective values, and $h_{max} = \max \mathcal{H}$, and ρ is the fixed step size in logarithmic scale for raising the water-levels. Then we have:

Theorem .1 (EagleMine Time complexity). *The time complexity of EagleMine algorithm is*
$O(\frac{\log h_{max}}{\rho} \cdot M + \mathcal{C} \cdot T \cdot M)$

A PROOF OF THEOREM 1

Proof. First, WaterLevelTree is invoked by EagleMine as a subprocedure, in which we compare all M non-empty bins with water level r in step 3, and then do binary opening [1] to remove small blobs (noise) by checking non-empty ones in step 4, both of which cost $O(M)$. From steps 5 to 6, for each island, we connect its children (# of islands $< M$) to it, so the time cost equals to the number of links, i.e. $O(M)$. Hence the whole iteration from step 2 to 7 takes $O(\Delta \cdot M)$, where $\Delta = {}^{\log h_{\max}}/_{\rho}$. As a result, we get a tree $\mathcal{T}$, whose height is Δ and width is at most M. The total number of links in that tree are less than $\Delta \cdot M$. Afterwards, the operation of contracting takes $O(\Delta \cdot M)$. In each tree level, the summation of bins in islands is less than M, so the complexity of both pruning and expanding process is also $O(\Delta \cdot M)$.

Consequently, the costs of constructing the water-level tree is $O(\Delta \cdot M)$. In the following steps of EagleMine algorithm, function $DistributionFit(\cdot)$ costs $O(T \cdot M)$, where each gradient-descent cost $O(M)$, the number of training data. Since our algorithm finds C micro-clusters when stops, the subtree with visited nodes by BFS search on $\mathcal{T}$ has C leaves. Due to the contraction of WaterLevelTree algorithm at step 8, each non-leaf node in the subtree has at least two children, hence the subtree has at most $2 \cdot C$ nodes, which means the steps from 5 to 15 have at most $2 \cdot C$ times of choosing the largest island, conducting $DistributionFit(\cdot)$, and applying hypothesis tests. The cost of statistical hypothesis test on each node (island) is linear of the number of bins in the island, which is less than M. During stitching, we only test those islands close to each other in a plane, which costs less than the above process on tree $\mathcal{T}$. Therefore, the time complexity of EagleMine is

$$O(\Delta \cdot M + 2C \cdot (T \cdot M + M)) = O(\frac{\log h_{max}}{\rho} \cdot M + C \cdot T \cdot M)$$

where $C \ll M$. ■

REFERENCES

[1] Rafael C Gonzalez and Richard E Woods. 2007. Image processing. *Digital image processing* (2007).