K-Distance Binary Label Counting in Large Scale Graph

ABSTRACT

We propose to count the label within 2-out of each vertex over large scale graph data. The input is a label and a graph. Our specific target is to count the neighbors of given label within 2-out of each vertex in the graph.

PVLDB Reference Format:

. K-Distance Binary Label Counting in Large Scale Graph. PVLDB, 14(1): XXX-XXX, 2020.

doi:XX.XX/XXX.XX

PVLDB Artifact Availability:

The source code, data, and/or other artifacts have been made available at http://vldb.org/pvldb/format_vol14.html.

1 INTRODUCTION

In WechatPay scenario, users are labeled as fraud or not (binary). For user u, the number of fraud 2-distance friends of u is an important feature of u in the corresponding risk control model.

1.1 Our contributions

First exact solution

2 RELATED WORK

To the best of our knowledge, this is the first work on exact label counting over k-distance neighbors.

Existing works on this problem are all approximate: Hyperloglog, HNF ...

Existing system for this problem: vertex-centric computing (pregel, giraph...); Flink...; Spark...; Parameter Server (Angel of Tencent).

3 PRELIMINARIES

In this section, we define our problem. Before the formal definition, we present some important concepts.

Definition 3.1 (Data Graph). A data graph is defined as a 2-tuple $G = (V, E, L)$, where V denotes the vertex set, E is the edge set and L is a binary labeling function over vertex, i.e., $V \mapsto \{0, 1\}$.

. Without loss of generality, G is assumed to be an undirected and unweighted simple graph, namely, there is no loop (edge that connects a vertex to itself) and at most one edge connecting a pair of vertices. We use $N_G(v)$ to denote the neighbor set of v in G.

Definition 3.2 (k-Hop Neighbor). Given a data graph $G = (V, E, L)$, and two vertices v_1, $v_2 \in V$, if the distance between v_1 and v_2 is k (the length of shortest path between v_1 and v_2), then v_1 and v_2 are k-hop neighbor of each other. We use $N_G^k(v)$ to denote the set of k-hop neighbors of v. Particularly, we use $N_G^0(v) = \{v\}$. Apparently, $N_G(v) = N_G^1(v)$.

We define the k-distance neighbor set of vertex v, denoted as $N_G^{\leq k}(v)$, as the set of neighbors of which the distance to v is not more than k, namely,

$$N_G^{\leq k}(v) = \bigcup_{i=0}^{k} \left(N_G^i(v)\right)$$

With the concepts as above, we formally define our problem.

Definition 3.3 (Problem Definition). Given a graph $G = (V, E, L)$ and a distance limit parameter k, for each vertex in V, output the number of vertices of label 1 in the k-distance neighbor set of v. We use $C(v)$ to denote this number, namely,

$$C(v) = \sum_{v' \in N_G^{\leq k}(v)} \left(L(v')\right)$$

Actuallyčň we can try 2-distance label counting first.

4 BASIC APPROACH

- Algorithm Design for acceleration, BFS, multiple BFS
- Concurrent with multiple threads
- Parallel with mutliple processes/machines
- Hardware level acceleration, such as GPU, FPGA, NUMA.... and so on

Proceedings of the VLDB Endowment, Vol. 14, No. 1 ISSN 2150-8097.
doi:XX.XX/XXX.XX