APM: Efficient Approximate Graph Pattern Matching System

Yang Yiliu

1155157082

Supervisor: James Cheng

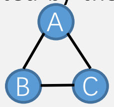
APM: Efficient Approximate Graph Pattern Matching System

Yang Yiliu

Introduction

Suppose there are three people, they are A, B, C, and they all know each other.

If we consider people as vertices and relationships as edges, the whole relationship can be represented by the below graph.



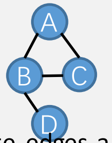
The problem is how many triangles are there in all WhatsApp users? We need an algorithm to solve it in such a large-scale graph.

	Time Cost	Applicable Graph Size	
Exact algorithm	More time	Small-scale graphs	×
Estimation algorithm	Less time	Large-scale graphs	✓

Neighborhood Sampling

Neighborhood sampling is a common method to estimate pattern number in a graph.

Consider all edges as a stream (randomly): (A,B), (B,D), (A,C), (B,C)



Each estimation procedure:

- Step1: Randomly get one edge $e1$. Delete edges appear before $e1$ in the stream.
- Step2: Randomly select one edge $e2$ from the neighbors of $e1$. Delete edges appear before $e2$ in the stream.
- Step3: Since three vertices are all sampled, check whether unsampled missing edge exists.

In short, above 3 steps can be considered as **SampleEdge**, **SampleNeighborEdge**, **ClosePattern** respectively.

Example:

- Step 1: Select (A,B) as $e1$ with probability 1/4. Now stream: (A,B), (B,D), (A,C), (B,C)
- Step 2: Select (A,C) as $e2$ with probability 1/3. Now stream: (A,B), (B,D), (A,C), (B,C)
- Step 3: The remaining edge is (B,C), which exists in the stream. That means in this estimation, there is a probability of (1/4 x 1/3 = 1/12) to form a triangle. Take 12 as return value.

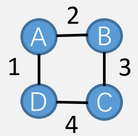
By doing such estimation several times, the result of $\frac{\sum Estimation\ return\ values}{Estimation\ times}$ will be close to the exact triangle number.

Implementation Method

4-vertex-pattern

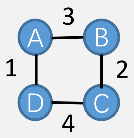
Take rectangle as an example.

The sampling procedure is more complex than triangle.



Rectangle-Type-I

- Step1: SampleEdge
- Step2: SampleNeighborEdge
- Step3: SampleNeighborEdge
- Step4: ClosePattern



Rectangle-Type-II

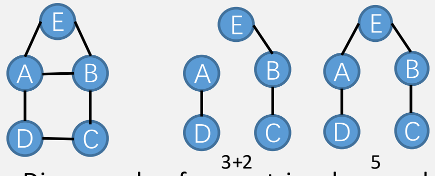
- Step1: SampleEdge
- Step2: SampleEdge
- Step3: ClosePattern

When the edge stream is (A,B), (B,C), (C,D), (A,D), the algorithm will detect type-I.

When the edge stream is (A,B), (C,D), (B,C), (A,D), the algorithm will detect type-II.

General Patterns

General patterns include not only type-I and type-II, but also the combination of type-I and type-II.



Discovered from triangle and rectangle sampling, different sampling type comes from different order of appearance of the edges.

We can separate types by vertices number. If the pattern contains 5 vertices, it can be divided into 5=2+3 or 5=5. Thus, such an algorithm can be generated.

Type-I

- Step1: SampleEdge
- Step2: SampleNeighborEdge
- Step3: SampleNeighborEdge
- Step4: SampleNeighborEdge
- Step5: ClosePattern

Type-II

- Step1: SampleEdge
- Step2: SampleEdge
- Step3: SampleNeighborEdge
- Step4: ClosePattern

Similarly, for 6-vertex-pattern, equations are 6=2+2+2, 6=3+3, 6=4+2, 6=6.

Before estimation, the system can automatically design estimation method.

Therefore, this system support general graph pattern matching.

Performances

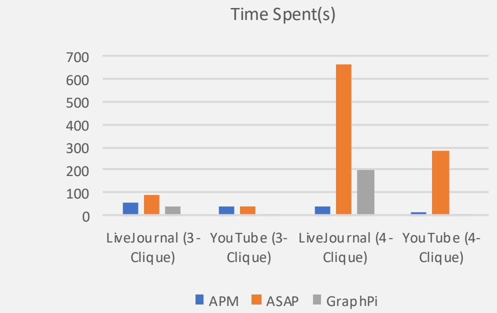
ASAP is also an approximate graph pattern matching system, but do not automatically support general patterns.

GraphPi is an exact graph pattern matching system.

We tested these three systems with LiveJournal and YouTube graph datasets. Here are the data of tested graphs.

Graph Datasets	|V|	|E|
LiveJournal	4847571	68993773
YouTube	1134890	2987624

Here we set ASAP and APM with 95% confidence, 5% error rate. All systems are tested with 8 threads.



References

A. P. Iyer, Z. Liu, X. Jin, S. Venkataraman, V. Braverman, and I. Stoica, “{ASAP}: Fast, approximate graph pattern mining at scale,” 2018, pp. 745–761.

T. Shi, M. Zhai, Y. Xu, and J. Zhai, “GraphPi: high performance graph pattern matching through effective redundancy elimination,” 2020, pp. 1–14.

Leskovec, J., and Krevl, A. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.