Project Milestone Document

Title: Improving Co-operative Caching Using Importance Aware Bloom Filter

Milestone: 1

Advisor: Dr. Prof. Hans- Peter Bischof

Student: Shridhar Bhalekar (snb3300@rit.edu)

Problem Statement

Bloom Filter and its variants [2] are widely used data structure in distributed environments. One of the major deployment areas of Bloom Filters is collaborative caching in distributed environments. One of the real world examples of such system is multiple web proxies sharing their caches with other peers. To make sure such a distributed system works correctly these Bloom Filters must perform efficiently.

First part of this project is to compare the Importance Aware Bloom Filter with conventional Bloom Filter. Second part of this project is to compare cooperative caching algorithms. Summary Cache [1] will be modified to use Importance Aware Bloom Filter and it will then be compared with Greedy Forwarding, N-Chance and Robinhood algorithms.

Strategy

Simulation technique will be used for both the parts of this project. For the comparison of Bloom Filter and Importance Aware Bloom Filter [3], simulator will create two identical networks of clients and server. On one system clients and server will be using Bloom Filter while on the other system they will use Importance Aware Bloom Filter. Simulator will distribute data on both networks and fire queries to get the number of False Positives.

For the comparison of Summary Cache (Importance Aware Bloom Filter) with N-Chance, Greedy Forwarding and Robinhood the simulator will create identical networks for all the algorithms. Simulator will distribute data on all networks and simulate query forwarding. During query forwarding it will keep track of the network hops, cache reference and disk reference. For the comparison of these algorithms I will be using ticks (time needed to access disk, cache and network hops) and cache hit/miss ratio.

Previous Milestone Work

In the previous milestone I read the research papers to better understand the cooperative caching algorithms and bloom filters. After understanding the algorithms I decided the comparison metrics for bloom filters and cooperative caching algorithms and designed a simulator for these comparisons. I have successfully completed the Bloom Filter and Importance Aware Bloom Filter comparison. These experiments were executed for varying cache size, bloom filter size and total requests. From the results it was observed that Importance Aware Bloom Filter is better than Bloom Filter in terms of the False Positive rate as the cache size and bloom filter size increases. The results were plotted on graphs to justify the performance of Importance Aware Bloom Filter over Bloom Filter.

Progress

In this milestone I have completed the design and implementation of Greedy Forwarding and N-Chance algorithms along with the simulator code to execute experiments using these algorithms. Currently, I have been working on the implementation of Robinhood algorithm. I have setup GitHub to maintain the code repository. Here's the link to my project https://github.com/snb3300/capstone/. This repository contains the project source code, project documents and some of the intermediate results which were produced while testing.

Obstacles

While implementing these algorithms I had to make minor changes to the design of simulator to make sure it creates a proper network of clients and server and gets the experiment results accordingly.

Future Plans

I am planning to complete the Robinhood algorithm implementation and get initial comparison results of Greedy Forwarding, N-Chance and RobinHood. The next part will be to implement Summary Cache and compare it with other three algorithms.

References

[1] **Li Fan, P. C. (2000). Summary cache: a scalable wide-area web cache sharing protocol. *IEEE/ACM* (pp. 281-293). IEEE/ACM.**

[2] Ming Zhong, P. L. (2008). Optimizing Data Popularity Concious bloom filters. *PODC '08 Proceedings of the twenty-seventh ACM symposium on Principles of distributed computing.* ACM.

[3] Puru Kulkarni, R. B. (2013). Importance-aware Bloom Filter for Set Membership Queries in Streaming Data. *COMSNETS, 2013 Fifth International Conference.* COMSNETS.

[4] S. Tarkoma, C. E. (2012). Theory and Practice of Bloom Filters for Distributed Systems. *IEEE Communications Surveys and Tutorials. Vol. 14, Number 1.* IEEE.