ABSTRACT

Improving system performance with caching replacement techniques has always been one of the hot topics in systematic research. Especially in the past decade, with the advance of the big data technology, more and more data are required to be stored and processed in the modern software systems. The caching replacement algorithm is very helpful in improving the performance of a system. The caching algorithm itself is an optimization technique that replaces blocks in cache memory with new ones. Some of the techniques of replacement algorithms are: LRU (Last Recent Used), FIFO (First In First Out), and LFU (Last Frequently Used).

However, such a single shift technique is considered insufficient for the daily workload scenario. Therefore, the LeCar Algorithm was created. LeCar Algorithm uses reinforcement learning that uses regret minimization to change between LRU and LFU policy. Unfortunately, the LeCar Algorithm has problems in overfitting, where LeCar is centered on the FIU workload with a small learning rate and discount rate. So that adjustment of the algorithm is considered impossible. So that a LeCar modification was made where this modification took the concept from LeCar but for policy changes based on the evicted history page of the LRU and LFU.

In this final project, modified LeCar and other algorithms, including LeCar, are compared with three types of datasets. The simulation results show that the LeCar modification can outperform other algorithms. In the FIU dataset scenario, the LeCar modification gets a 3-6% cache hit better than other algorithms, including LeCar. Meanwhile, the modified Synthetic dataset excels when the number of objects and the number of requests is high and competes with LeCar when the number of objects and requests is low. Moreover, the Medisyn modified LeCar dataset excels when the cache size used is small and competes with LRU when the cache size used is significant.

Key Words: Cache Replacement, LRU, LFU, LeCar Modification, Cache.