**7.1 Heap Memory analysis in JVM**

This section shows several charts that were generated using Memory Analyzer tool in Eclipse (MAT).This tool profiled the memory heap consumption of the main thread application and all of its child classes. The *Indexer* class mainly depends on *Pair* class which had a significant portion of the allocated memory.

The component reports below provide breakdown charts for the dominant classes *Pair* & *String* . The first two charts contrast the memory utilization of class *Pair* replacing document reference from filepath strings to integer IDs. The memory elimination on class-level has consistently optimized memory usage on thread-level. To prove that, we have conducted same analysis on *String* object type before and after optimization. According to the second pair, the ratio of (string memory: application memory) has remained the nearly the same.  
Before optimization, it was nearly 60% of the heap allocated for the main thread (233/387) and even after optimization, it remained around same allocation percentage (165/260), however, the main thread heap size has been reduced from 387MB to 260MB in one go.

Figure (n): Memory analyzer component reports.

|  |  |
| --- | --- |
| Before optimization | After optimization |
|  |  |
|  |  |