

数据结构

优缺点

排序列表Array/List

使用二分法查找，不平衡。

HashMap/TreeMap

性能高，内存消耗大，几乎是原始数据的三倍。

Skip List 跳跃表

可快速查找词语，在lucene、redis、Hbase等均有实现。相对于TreeMap等结构，特别适合高并发场景。

Trie

适合英文词典，如果系统中存在大量字符串且这些字符串基本没有公共前缀，则相应的trie树将非常消耗内存。

Double Array Trie

适合做中文词典，内存占用小，很多分词工具均采用此种算法。

Ternary Search Tree

三叉树，每一个node有3个节点，兼具省空间和查询快的优点。

Finite State Transducers (FST)

一种有限状态转移机，内存开销较小。