Chosen Algorithm explanation step by step:

* First, I am looping through input text file word by word – **Time Complexity O(n)**
  + And Storing KeyWord1 indexes in List **- Space complexity O(n)** (worst case – all words in file are KeyWord1)
  + And storing KeyWord2 occurrences at each word index in Dictionary – **Space complexity O(n)** 
    - Example Input – [the canal the man canal panama canal] – KeyWord2 is Canal
    - KeyWord2Occurence Dictionary [{0, 0}, {1, 1}, {2, 1}, {3, 1}, {4, 2}, {5, 2}, {6, 3}]
* Then, looping through KeyWord1 index list – **Time complexity O(n)**
  + Example: let’s take Keyword1 at index = 2 in above example input and find out all occurrences possible with KeyWord2 within Range 3
  + Look for all KeyWords2 occurrences in dictionary after KeyWord1 index within given range
    - Dictionary Index to check KeyWord2 occurrences = (2 + 3 -1) = 5
    - Dictionary value at range end index Dictionary [5] = 2 (number of KeyWord2’s present till this point)
  + Look for all KeyWord2 occurrences in dictionary before KeyWord1 index within given range
    - Dictionary Index to check at KeyWord2 occurrences = (2 - 3) = less than zero So set index value as -1. To handle these kinds of cases added {-1, 0} in dictionary as first element because occurrence always zero in this case.
    - Dictionary value at range start index Dictionary [-1] = 0 (number of KeyWord2’s present till this point)
  + **BothKeyWordsOccurenceWithinRange =** Number of KeyWord’2 occurrences till range end – Number of KeyWord2’s occurrences at start of range (2-0) = 2
* Final Time Complexity = O(n) + O(n) = O(2n) which can simplify as **O(n)**
* Final Space Complexity = O(n) + O(n) = O(2n) which can simplify as **O(n)**