

## **Experiment NO.: 01**

*Instructor:* Mr Arvind Sardar.

*Name:* Abdul Rehman, Roll No:CS3101

### **AIM**

**Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.**

### **INTRODUCTION TO FIND-S ALGORITHM**

The FINDS algorithm is a search algorithm designed to efficiently search through a sorted list of data elements to find the index of a specific target element. The acronym FINDS stands for "Fast Index- based Non-recursive Discrete Search." This algorithm works by recursively dividing the search space into smaller and smaller subspaces until the target element is found or it is determined that the element does not exist in the list.

The FINDS algorithm is particularly useful when searching through large sorted datasets, where linear search algorithms would be too slow. This algorithm has a time complexity of  $O(\log n)$ , which means that it can efficiently search through datasets with millions or even billions of elements in a reasonable amount of time.

The FINDS algorithm is widely used in computer science and is particularly important in the fields of database management and information retrieval, where efficient search algorithms are essential for managing and querying large datasets.

### **DATASET DESCRIPTION:**

The FINDS algorithm is not a dataset, but rather a search algorithm designed to search through sorted datasets. The algorithm works by recursively dividing the search space into smaller and smaller subspaces until the target element is found or it is determined that the element does not exist in the list.

To use the FINDS algorithm, you would need a sorted dataset to search through, as it is designed specifically for searching through sorted lists of data elements. The dataset can be any type of data that can be sorted, such as numerical data, text data, or any other type of structured data.

### **IMPLEMENTATION CODE FOR FIND-S ALGORITHM**

```
1 import csv
2 a = []
3 with open (' enjoysport . csv ', ' r' ) as csvfile :
4     next ( csvfile )
5     for row in csv . reader ( csvfile ): a.
6         append ( row )
7     print ( a )
8
9 print ( "\ nThe total number of training instances are : ", len ( a ) num_attribute = len ( a
10 [0] ) -1
11
12 print ( "\ nThe initial hypothesis is : ")
13 hypothesis = [ ' 0 ' ] * num_attribute print (
14 hypothesis )
15 for i in range ( 0 , len ( a )):
16     if a[ i ][ num_attribute ] == ' yes ' :
17         print ( "\ n Instance ", i+1 , " is", a[ i] , " and is Positive Instance ")
```

```

18     for j in range (0 , num_attribute ) :
19         if hypothesis [ j ] == ' 0 ' or hypothesis [ j ] == a[ i][ j]:
20             hypothesis [ j ] = a[ i][ j]
21         else :
22             hypothesis [ j ] = ' ?'
23         print ( " The hypothesis for the training instance ",
24               i+1 , " is: ", hypothesis , "\ n" ) if a[
25 i][ num_attribute ] == ' no ' :
26         print ( "\ n Instance ", i+1 , " is",
27               a[ i], " and is Negative Instance Hence Ignored " )
28         print ( " The hypothesis for the training instance ",
29               i+1 , " is: ", hypothesis , "\ n" )
30
31 print ( "\ nThe Maximally specific hypothesis for the training instance is ",
        hypothesis )

```

### Output Of Find-s

```

[['sunny', 'warm', 'normal', 'strong', 'warm', 'same', 'yes'], ['sunny', 'warm', 'high', 'strong', 'warm', 'same', 'yes'], ['rainy', 'cold', 'high', 'strong', 'warm', 'change', 'no'], ['sunny', 'warm', 'high', 'strong', 'cool', 'change', 'yes']]

The total number of training instances are : 4

The initial hypothesis is :
['?', '?', '?', '?', '?', '?']

Instance 1 is ['sunny', 'warm', 'normal', 'strong', 'warm', 'same', 'yes'] and is Positive Instance
The hypothesis for the training instance 1 is: ['sunny', 'warm', 'normal', 'strong', 'warm', 'same']

Instance 2 is ['sunny', 'warm', 'high', 'strong', 'warm', 'same', 'yes'] and is Positive Instance
The hypothesis for the training instance 2 is: ['sunny', 'warm', '?', 'strong', 'warm', 'same']

Instance 3 is ['rainy', 'cold', 'high', 'strong', 'warm', 'change', 'no'] and is Negative Instance Hence Ignored
The hypothesis for the training instance 3 is: ['sunny', 'warm', '?', 'strong', 'warm', 'same']

Instance 4 is ['sunny', 'warm', 'high', 'strong', 'cool', 'change', 'yes'] and is Positive Instance
The hypothesis for the training instance 4 is: ['sunny', 'warm', '?', 'strong', '?', '?']

The Maximally specific hypothesis for the training instance is ['sunny', 'warm', '?', 'strong', '?', '?']

```

Figure 1: Find-s Output

### CONCLUSION

**FINDS** algorithm is a fast and efficient search algorithm that is designed to search through sorted lists of data elements. By dividing the search space into smaller and smaller subspaces, the algorithm can quickly locate the index of a target element, even in very large datasets. Overall, understanding the FINDS algorithm can be very beneficial for computer science students and professionals, as it can be used to optimize and speed up a wide range of search and retrieval tasks.