

Program 1: 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 csv file.

Solution: Find-S algorithm is a searching algorithm.

Purpose: To find the maximally specific hypothesis from the set of hypothesis space.

Method: Begin with most specific possible hypothesis in H, then generalize this hypothesis each time it fails to cover an observed positive training example. Notations:

- * D - Training data set
- * X - Set of instances with in training data set
- * x - particular instance in training example
- * H - Set of possible hypothesis
- * h - particular hypothesis described by conjunction of constraints on the attributes
- * ai - constraint attribute of hypothesis, ai can have a value of 0 (no value), or any value(ai=sunny), or ?(any value)
- * c - target concept

Algorithm: 1. Initialize h to the most specific hypothesis in H

1. For each positive training instance x

- For each attribute constraint ai is satisfied by x
- Then do nothing
 - Else replace ai in h by the next more general constraint that is satisfied by x

3. Output hypothesis h

```
import csv
#!/usr/bin/python #list creatin
hypo=['%','%','%','%','%','%']
with open('Training_examples.csv') as csv_file:
    readcsv = csv.reader(csv_file, delimiter=',')
    print(readcsv)
    data=[]
    print("\nThe given training examples are:")
    for row in readcsv:
        print(row)
        if row[len(row)-1] == 'Yes':
            data.append(row)
print("\nThe positive examples are:")
for x in data:
    print(x)
print("\n")
TotalExamples=len(data)
i=0
j=0
k=0
print("The steps of the Find-s algorithm are\n",hypo)
```

```
list=[]
p=0
d=len(data[p])-1
for j in range(d):
    list.append(data[i][j])
hypo=list
for i in range(1,TotalExamples):
    for k in range(d):
        if hypo[k]!=data[i][k]:
            hypo[k]='?'
        else:
            hypo[k]
    print(hypo)
    print("-----")
print("\nThe maximally specific Find-s hypothesis for the given training examples is");
list=[]
for i in range(d):
    list.append(hypo[i])
print(list)
```

OUTPUT

The given training examples are:

['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 positive examples are:

['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same', 'Yes']

['Sunny', 'Warm', 'High', 'Strong', 'Warm', 'Same', 'Yes']

['Sunny', 'Warm', 'High', 'Strong', 'Cool', 'Change', 'Yes']

The steps of the Find-s algorithm are

['%', '%', '%', '%', '%', '%']

['Sunny', 'Warm', '?', 'Strong', 'Warm', 'Same']

['Sunny', 'Warm', '?', 'Strong', '?', '?']

The maximally specific Find-s hypothesis for the given training examples is

['Sunny', 'Warm', '?', 'Strong', '?', '?']