Practical 2

For a given set of training data examples stored in a .CSV file, implement and demonstratethe Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.

CANDIDATE-ELIMINATION Learning Algorithm

The CANDIDATE-ELIMINTION algorithm computes the version space containing allhypotheses from H that are consistent with an observed sequence of training examples.

Initialize G to the set of maximally general hypotheses in H
Initialize S to the set of maximally specific hypotheses in H
For each training example d, do

- If d is a positive example
  - Remove from G any hypothesis inconsistent with d
  - For each hypothesis s in S that is not consistent with d
    - Remove s from S
    - Add to S all minimal generalizations h of s such that
      - h is consistent with d, and some member of G is more general than h
    - Remove from S any hypothesis that is more general than another hypothesis in S

- If d is a negative example
  - Remove from S any hypothesis inconsistent with d
  - For each hypothesis g in G that is not consistent with d
    - Remove g from G
    - Add to G all minimal specializations h of g such that
      - h is consistent with d, and some member of S is more specific than h
    - Remove from G any hypothesis that is less general than another hypothesis in G

CANDIDATE- ELIMINTION algorithm using version spaces

Training Examples:

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

Algorithmic steps:

Initially : G0 = [[?, ?, ?, ?, ?, ?]
S0 = [ϕ, ϕ, ϕ, ϕ, ϕ, ϕ]

For instance 1 : Positive output (convert specific to general)
Instance 1: <'sunny','warm','normal','strong','warm ','same'>

G1 = G
S1 = <'sunny','warm','normal','strong','warm ','same'>

For instance 2 : Positive output (convert specific to general)
Previous hypothesis: <'sunny','warm',**'normal'**,'strong','warm ','same'>
Instance 2: <'sunny','warm',**'high'**,'strong','warm ','same'>

G2 = G1
S2 = ['sunny','warm',**?**,'strong','warm ','same'] = New hypothesis

For instance 3 : Negative output (convert general to specific)
Previous hypothesis: <'sunny','warm',**?**,'strong','warm ','same'>
Instance 3: <**'rainy','cold','high'**,'strong','warm ',**'change'**>
Instance 2: <'sunny','warm','high','strong','warm ','same'>

G3 = [<'sunny', ?, ?, ?, ?, ?>, <?, 'warm', ?, ?, ?, ?>, <?, ?, ?, ?, ?, ?>,
<?, ?, ?, ?, ?, ?>, <?, ?, ?, ?, ?, ?>, <?, ?, ?, ?, ?, 'same'>]
S3 = S2

For instance 4 : Positive output (convert specific to general)

Previous hypothesis: <'sunny','warm',**?**,'strong','warm ','same'>

Instance 4: <'sunny','warm',**'high'**,'strong',**'cool'**,**'change'**> (s4 compare with s3)

Previous G3 = [['sunny', ?, ?, ?, ?, ?], [?, 'warm', ?, ?, ?, ?], [?, ?, ?, ?, ?, ?], [?, ?, ?, ?, ?, ?], [?, ?, ?, ?, ?, ?], [?, ?, ?, ?, ?, 'same']]

G4 = [['sunny', '?', '?', '?', '?', '?'], ['?', 'warm', '?', '?', '?', '?']] ("same" is ? in S4)
S4 = ['sunny','warm',?,'strong', ?, ?]

At last, by synchronizing the G4 and S4 algorithm produce the output.

Output:

Final Specific_h:
['sunny' 'warm' '?' 'strong' '?' '?']

Final General_h:
[['sunny', '?', '?', '?', '?', '?'],
['?', 'warm', '?', '?', '?', '?']]