19Z601- MACHINE LEARNING

UNIT- 1 INTRODUCTION

INTRODUCTION : Types of Learning - Designing a learning system - concept learning - Find-s Algorithm - Candidate Elimination - Data Preprocessing - Cleaning - Data Scales - Transformation - Dimensionality Reduction. (9)

Presented by

Ms.Anisha.C.D

Assistant Professor

CSE

CANDIDATE ELIMINATION ALGORITHM

- **Input : Set of instances in the training dataset**
- **Output : Hypothesis G and S**

Step 1: Initialize G, to the maximally general hypotheses.

Step 2: Initialize S, to the maximally specific hypotheses.

- Generalize the initial hypothesis for the first positive instance

Step 3: For each subsequent new training instance

- If the instance is positive
  - Generalize S to include the positive instance
  - ➢ Check the attribute value of the positive instance and S
    - If the attribute value of positive instance and S are difference, fill that field value with '?'
    - If the attribute value of positive instance and S are same, then do no change
- If the instance is negative
  - Specialize G to exclude the negative instance,
  - Add to G all minimal specialization to exclude the negative example and be consistent with S
    - If the attribute value of S and the negative instance are different, then fill that attribute value with S value
    - If the attribute value of S and negative instance are same, no need to update 'G' and fill that attribute value with '?
  - Remove from S all inconsistent hypotheses with the negative instance.

Candidate Elimination Method
Problem 2 : Dataset starting with Negative Instance

SIZE	COLOR	SHAPE	CLASS/LABEL
Big	Red	Circle	No
Small	Red	Triangle	No
Small	Red	Circle	Yes
Big	Blue	Circle	No
Small	Blue	Circle	Yes

Candidate Elimination Method
Problem 2 : Dataset starting with Negative Instance

Step 1 : Initialize the General and Specific Hypothesis

S0 : {<∅, ∅, ∅>}

G0: {<?, ?, ?>}

Candidate Elimination Method
Problem 2 : Dataset starting with Negative Instance

Step 2 : The first example is **negative**, the hypothesis at the specific boundary is consistent, hence we retain it, and the hypothesis at the generic boundary is inconsistent hence we write all consistent hypotheses by removing one “?” at a time.

S1: {<∅, ∅, ∅>}

G1: (Small, ?, ?), (?, Blue, ?), (?, ?, Triangle)

Candidate Elimination Method
Problem 2 : Dataset starting with Negative Instance

- The **second example is negative**, the hypothesis at the specific boundary is consistent, hence we retain it, and the hypothesis at the generic boundary is inconsistent hence we write all consistent hypotheses by removing one “?” at a time.

S2: (0, 0, 0)

G2: (Small, Blue, ?), (Small, ?, Circle), (?, Blue, ?), (Big, ?, Triangle), (?, Blue, Triangle)

Candidate Elimination Method
Problem 2 : Dataset starting with Negative Instance

The third example **is positive, the hypothesis at the specific boundary is inconsistent**, hence we extend the specific boundary, and the consistent hypothesis at the generic boundary is retained and inconsistent hypotheses are removed from the generic boundary.

S3: (Small, Red, Circle)

G3: (Small, ?, Circle)

Candidate Elimination Method
Problem 2 : Dataset starting with Negative Instance

The **fourth example is negative**, the hypothesis at the specific boundary is consistent, hence we retain it, and the hypothesis at the generic boundary is inconsistent hence we write all consistent hypotheses by removing one “?” at a time.

S4: (Small, Red, Circle)

G4: (Small, ?, Circle)

Candidate Elimination Method
Problem 2 : Dataset starting with Negative Instance

The **fifth example is positive**, the hypothesis at the specific boundary is inconsistent, hence we extend the specific boundary, and the consistent hypothesis at the generic boundary is retained and inconsistent hypotheses are removed from the generic boundary.

S5: (Small, ?, Circle)

G5: (Small, ?, Circle)