Concept Learning

Today:

Learning: inducing general functions from examples

Inductive Bias

Concept Learning

What is a Concept?

Concept Learning

What is a Concept?

-A general category of things

-A boolean-valued function

Concept Learning

Concept Learning: Inferring a boolean-valued function from training examples of its inputs and outputs

Concept Learning

A Concept Learning Task:

“Days in which Aldo enjoys his favorite water sport”

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

+ Cloudy

Concept Learning

- Hypothesis Representation

Concept Learning

- Hypothesis Representation
  - Simple representation: Conjunction of constraints on the 6 instance attributes
    - indicate by a “?” that any value is acceptable
    - specify a single required value for the attribute
    - indicate by a “∅” that no value is acceptable

Example:

h = (?, Cold, High, ?, ?, ?)

indicates that Aldo enjoys his favorite sport on cold days with high humidity (independent of the other attributes)

Concept Learning

- $h(x)=1$ if example x satisfies all the constraints
  $h(x)=0$ otherwise
- Most general hypothesis: (?, ?, ?, ?, ?, ?)
- Most specific hypothesis: ($\emptyset$, $\emptyset$, $\emptyset$, $\emptyset$, $\emptyset$, $\emptyset$)

Concept Learning

- Notation
  - Set of instances X
  - Target concept $c : X \rightarrow \{0,1\}$ (*EnjoySport*)
  - Training examples $\{x , c(x)\}$
  - Data set $D \subset X \times \{0,1\}$
  - Set of possible hypotheses H
  - $h \in H \qquad h : X \rightarrow \{0,1\}$

Concept Learning

Goal: Find h / $h(x)=c(x)$

Concept Learning

Goal: Find h / $h(x)=c(x)$

Really? Can we say that h(x)=c(x)?

Concept Learning

- Inductive Learning Hypothesis

  Any hypothesis *h* found to approximate the target function *c* well over a sufficiently large set *D* of training examples *x*, will also approximate the target function well over other unobserved examples in *X*

Concept Learning

Concept Learning as Search

Find $h \in H$ / $h(x)=c(x)$

Concept Learning

Concept Learning as Search

Find $h \in H \; / \; h(x)=c(x)$

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

+ Cloudy

Concept Learning

Concept Learning as Search

- Distinct instances in *X* : 3.2.2.2.2.2 = 96

Concept Learning

Concept Learning as Search

- Distinct instances in *X* : 3.2.2.2.2.2 = 96
- Distinct hypotheses
  - syntactically 5.4.4.4.4.4 = 5120
  - semantically 1 + (4.3.3.3.3.3) = 973

Concept Learning

- General-to-Specific Ordering of hypotheses

  h_1=(sunny,?,?,Strong,?,?) h_2=(Sunny,?,?,?,?,?)

Concept Learning

- General-to-Specific Ordering of hypotheses

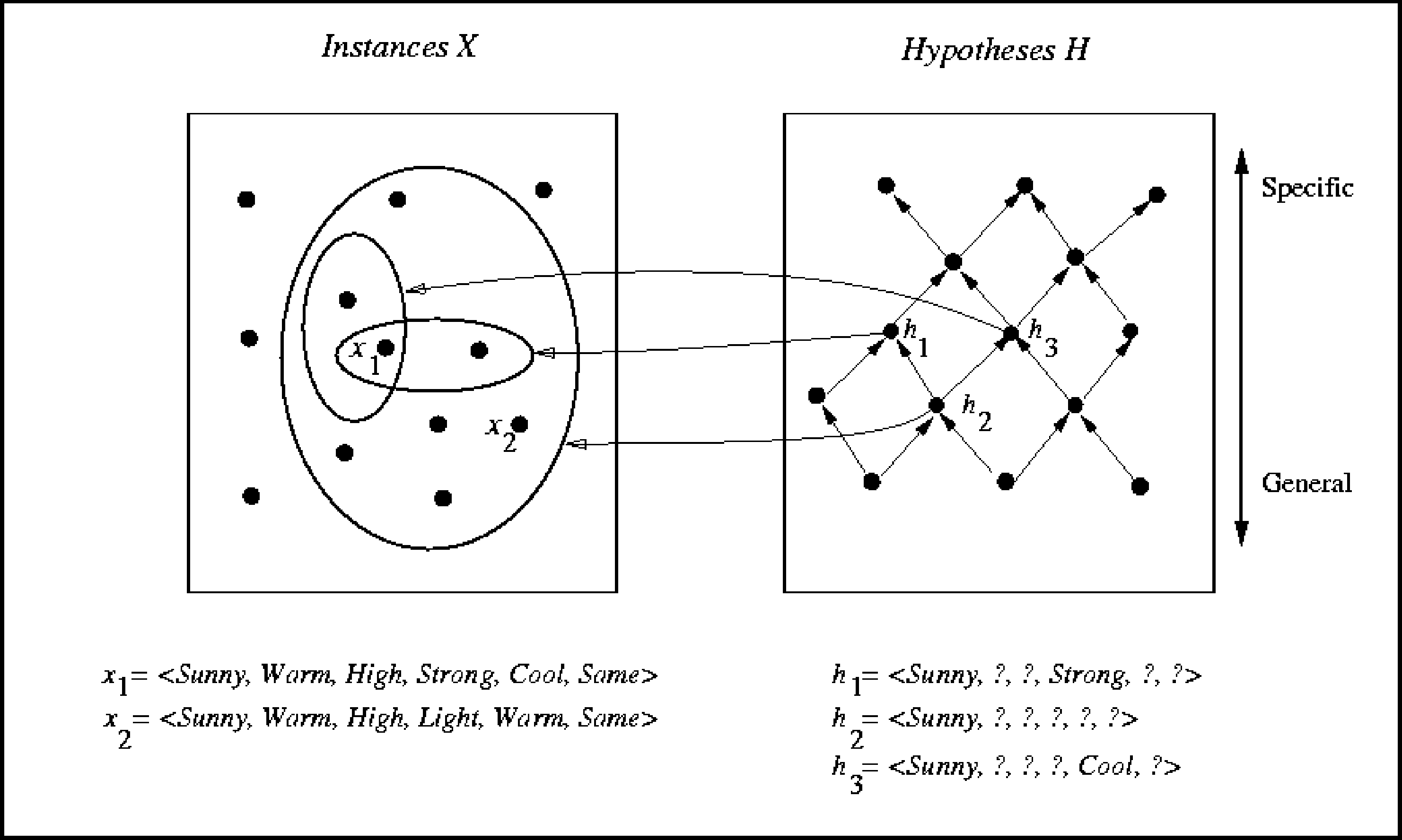
h_1=(sunny,?,?,Strong,?,?) h_2=(Sunny,?,?,?,?,?)

Definition: h_2 is *more_general_than_or_equal_to* h_1 (written $h_2 \geq_g h_1$) if and only if

$$(\forall x \in X)\ [\ h_1(x)=1 \rightarrow h_2(x)=1]$$

$\geq_g$ defines a partial order over the hypotheses space for *any* concept learning problem

Concept Learning



Concept Learning

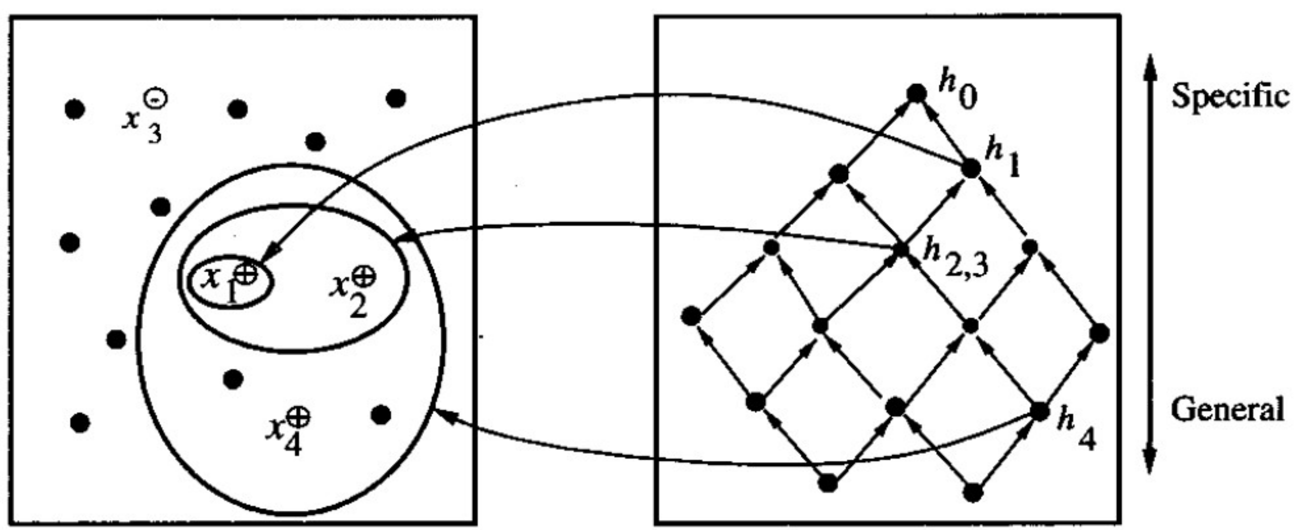
Finding a Maximally Specific Hypothesis

– *Find-S* Algorithm

1. Initialize h to the most specific hypothesis in H
2. For each positive training instance x
   - For each attribute constraint a_i in h
     If the constraint a_i is satisfied by x
     Then do nothing
     Else replace a_i in h by the next more general constraint that is satisfied by x
3. Output hypothesis h

TABLE 2.3
FIND-S Algorithm.

Concept Learning



h_0 = <∅, ∅, ∅, ∅, ∅, ∅>

x_1 = <*Sunny Warm Normal Strong Warm Same*>, +	h_1 = <*Sunny Warm Normal Strong Warm Same*>
x_2 = <*Sunny Warm High Strong Warm Same*>, +	h_2 = <*Sunny Warm ? Strong Warm Same*>
x_3 = <*Rainy Cold High Strong Warm Change*>, -	h_3 = <*Sunny Warm ? Strong Warm Same*>
x_4 = <*Sunny Warm High Strong Cool Change*>, +	h_4 = <*Sunny Warm ? Strong ? ?* >

Concept Learning

- Questions left unanswered:
  - Has the learner converged to the correct concept?
  - Why prefer the most specific hypothesis?
  - Are the training examples consistent?

Concept Learning

Version Spaces and the Candidate-Elimination Algorithm

- The Candidate-Elimination Algorithm outputs a description of the set of all hypotheses consistent with the training examples
- Representation
  - Consistent hypotheses

$$\text{Consistent}(h,D) \equiv (\forall \{x,c(x)\} \in D) \;\; h(x) = c(x)$$

Concept Learning

– Version Space

$VS_{H,D} \equiv \{h \in H \mid Consistent(h,D)\}$

– The List-Then-Eliminate Algorithm

- Initialize the version space to *H*
- Eliminate any hypothesis inconsistent with any training example

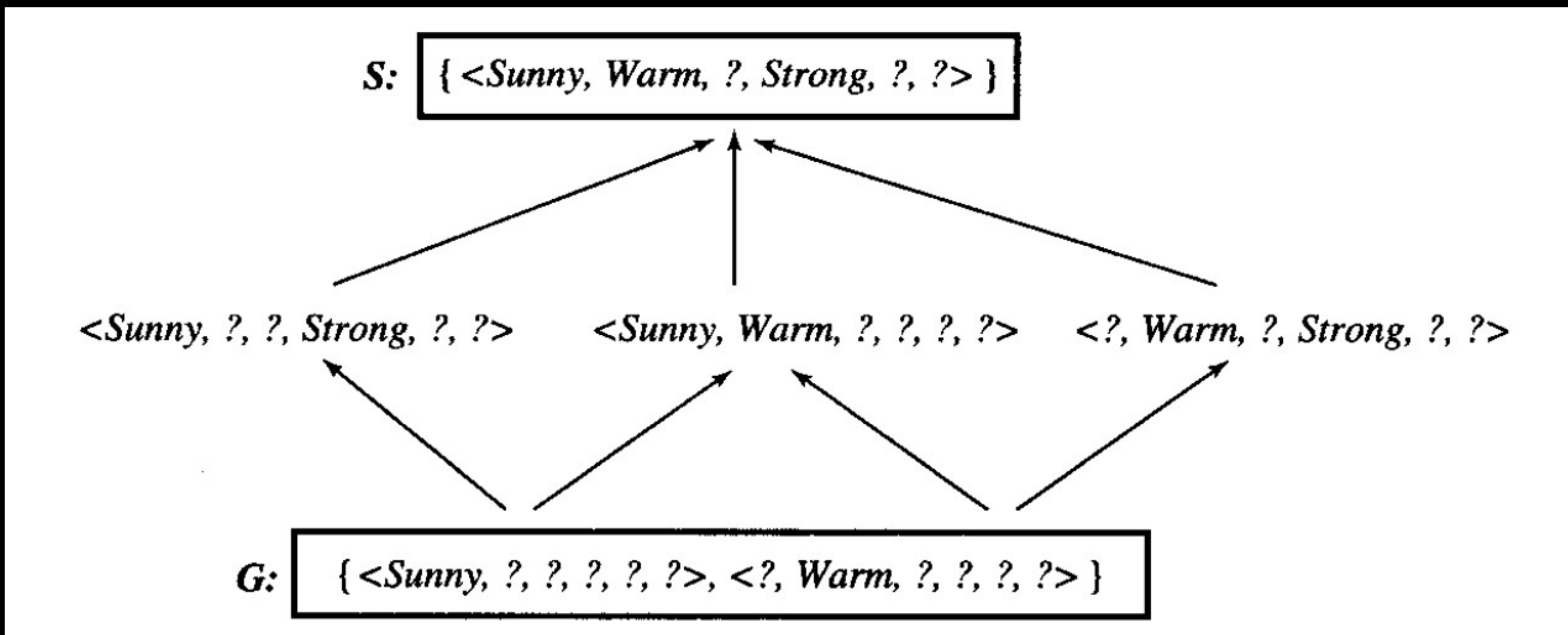
$\Rightarrow$ the version space shrinks to the set of hypothesis consistent with the data

Impractical!

Concept Learning

- Compact Representation for Version Spaces
  - General Boundary $G(H,D)$: Set of maximally general members of H consistent with D
  - Specific Boundary $S(H,D)$: set of minimally general (i.e., maximally specific) members of H consistent with D

Concept Learning



Concept Learning

- Theorem: Version Space Representation
  - For all X, H, c and D such that S and G are well defined,

$$VS_{H,D} \equiv \{h \in H \mid (\exists\, s \in S)\ (\exists\, g \in G)\ (g \geq_g h \geq_g s)\}$$

Concept Learning

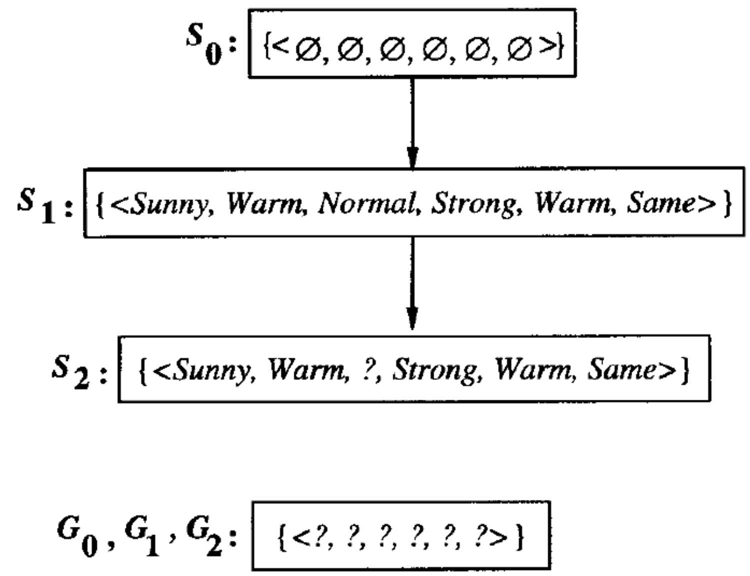
- Candidate-Elimination Learning Algorithm

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

TABLE 2.5
CANDIDATE-ELIMINATION algorithm using version spaces. Notice the duality in how positive and negative examples influence S and G.

Concept Learning

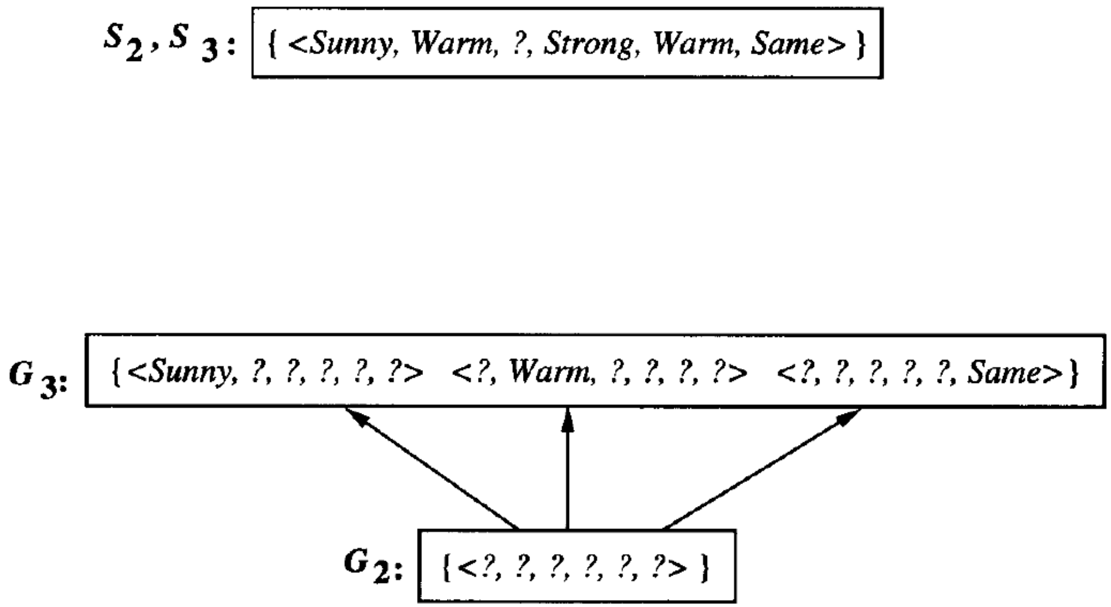


Training examples:

1. <*Sunny, Warm, Normal, Strong, Warm, Same*>, *Enjoy Sport* = *Yes*
2. <*Sunny, Warm, High, Strong, Warm, Same*>, *Enjoy Sport* = *Yes*

FIGURE 2.4
CANDIDATE-ELIMINATION Trace 1. S_0 and G_0 are the initial boundary sets corresponding to the most specific and most general hypotheses. Training examples 1 and 2 force the S boundary to become more general, as in the FIND-S algorithm. They have no effect on the G boundary.

Concept Learning



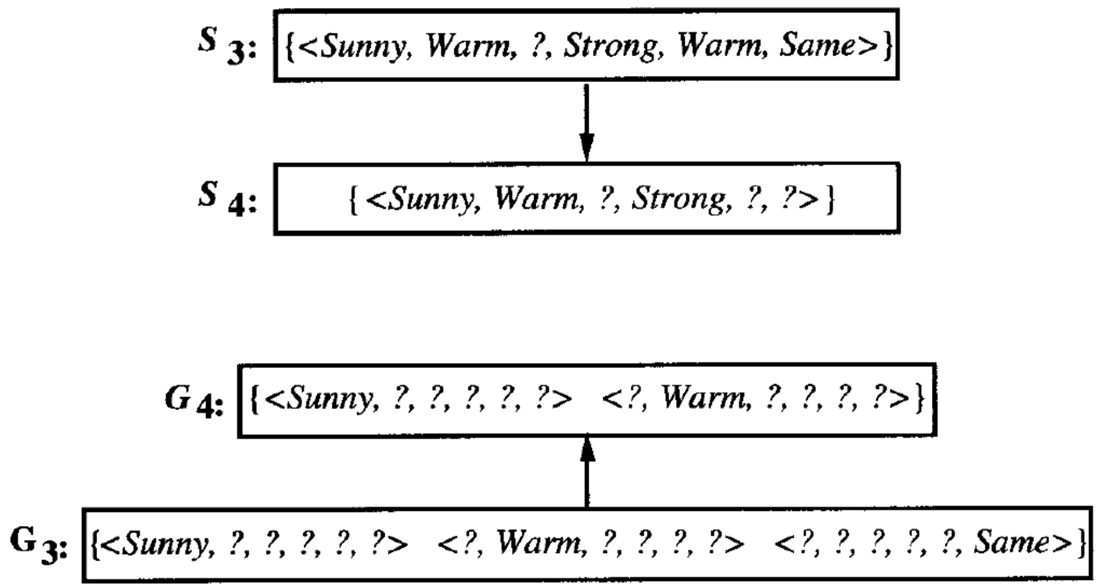
Training Example:

3. *<Rainy, Cold, High, Strong, Warm, Change>, EnjoySport=No*

FIGURE 2.5

CANDIDATE-ELIMINATION Trace 2. Training example 3 is a negative example that forces the G_2 boundary to be specialized to G_3. Note several alternative maximally general hypotheses are included in G_3.

Concept Learning



Training Example:

4. <*Sunny, Warm, High, Strong, Cool, Change*>, *EnjoySport* = *Yes*

FIGURE 2.6
CANDIDATE-ELIMINATION Trace 3. The positive training example generalizes the S boundary, from S_3 to S_4. One member of G_3 must also be deleted, because it is no longer more general than the S_4 boundary.

Concept Learning

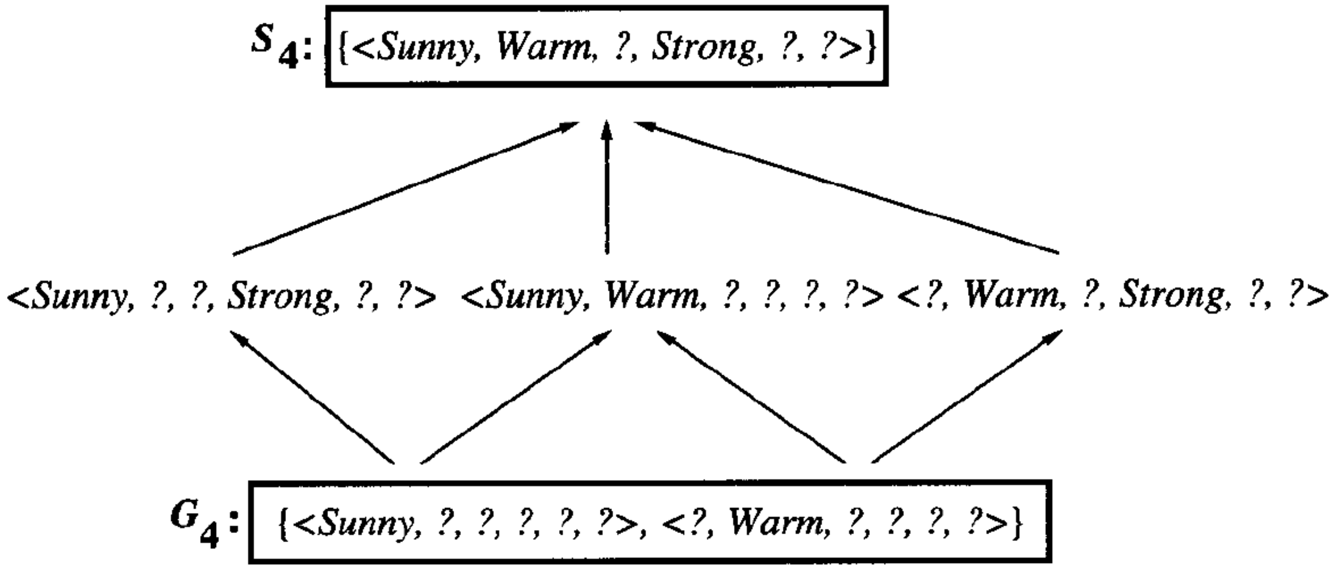


FIGURE 2.7
The final version space for the *EnjoySport* concept learning problem and training examples described earlier.

Concept Learning

- Remarks
  - Will the Candidate-Elimination converge to the correct hypothesis?
  - What training example should the learner request next?
  - How can partially learned concepts be used?

Concept Learning

Instance	*Sky*	*AirTemp*	*Humidity*	*Wind*	*Water*	*Forecast*	*EnjoySport*
A	Sunny	Warm	Normal	Strong	Cool	Change	?
B	Rainy	Cold	Normal	Light	Warm	Same	?
C	Sunny	Warm	Normal	Light	Warm	Same	?
D	Sunny	Cold	Normal	Strong	Warm	Same	?

TABLE 2.6
New instances to be classified.

A=*yes* B=*no* C=1/2 *yes* - 1/2 *no* D=1/3 *yes* - 2/3 *no*

Concept Learning

Inductive Bias

– The hypothesis space previously considered for the *EnjoySport* task is biased. For instance, it does not include disjunctive hypothesis like:

Sky=Sunny or Sky=cloudy

Can a hypothesis space that includes every possible hypothesis be used ?

Concept Learning

An unbiased H must contain the power set of X

PowerSet (X) = the set of all subsets of X

$|\text{Power Set}(X)| = 2^{|X|}$ ($= 2^{96} \sim 10^{28}$ for *EnjoySport*)

- Unbiased Learning of *EnjoySport*

  H = Power Set (X)

Concept Learning

For example, “Sky=Sunny or Sky=Cloudy” $\in H$:

$$(\text{Sunny},?,?,?,?,?) \vee (\text{Cloudy},?,?,?,?,?)$$

Suppose x_1, x_2, x_3 are positive examples and x_4, x_5 negative examples

$$\Rightarrow \quad S:\{(x_1 \vee x_2 \vee x_3)\} \quad G:\{\neg(x_4 \vee x_5)\}$$

In order to converge to a single, final target concept, every instance in X has to be presented!

Concept Learning

– Voting?

Each unobserved instance will be classified positive by exactly half the hypotheses in the version space and negative by the other half !!

Concept Learning

- The Futility of Bias-Free Learning

  A learner that makes no a priori assumptions regarding the target concept has no rational basis for classifying unseen instances