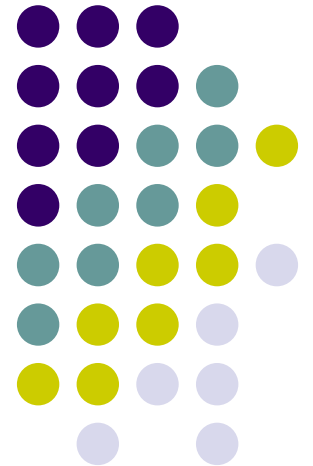


Concept Learning

Zhang Hongxin
zhx@cad.zju.edu.cn

State Key Lab of CAD&CG, ZJU
2008-02-21



Overview



- Learning from examples
- General-to-specific ordering of hypotheses
- Version spaces and candidate elimination algorithm
- Inductive bias

Introduction



- What is concept learning?
 - Induce Boolean function from a sample of positive/negative training examples.
 - Infer the general definition of some concept, given examples labeled as members or nonmembers of the concept.
- Concept learning in daily life
 - 根据人证物证判断犯罪嫌疑人是否有罪
 - 根据笔试面试决定是否录用
 - And more ...



A Demo Task – *Enjoy Sport*

- **Concept:**
 - days on which my friend Tom enjoys his favorite water sports
- **Task:**
 - predict the value of "Enjoy Sport" for an arbitrary day based on the values of the other attributes

Training Data

Attributes	ID	Sky	Temp	Humidity	Wind	Water	Forecast	Enjoy
	1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
	2	Sunny	Warm	High	Strong	Warm	Same	Yes
Example	3	Rainy	Cold	High	Strong	Warm	Change	No
	4	Sunny	Warm	High	Strong	Cool	Change	Yes



Representing Hypothesis

- Hypothesis h is described as a conjunction of constraints on attributes
- Each constraint can be:
 - A specific value : e.g. Water=Warm
 - A don't care value : e.g. Water=?
 - No value allowed (null hypothesis): e.g. Water= \emptyset
- Example: hypothesis h

	Sky	Temp	Humid	Wind	Water	Forecast	
<	Sunny	?	?	Strong	?	Same	>



A Demo Task – *Enjoy Sport*

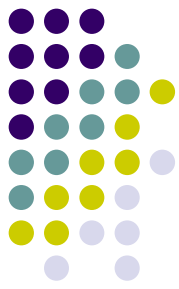
- **Given:**

- **Instance Space X :** Possible days described by the attributes=[Sky, Temp, Humidity, Wind, Water, Forecast]
- **Target function c :** Enjoy Sport $X \rightarrow \{0,1\}$
- **Hypotheses Space H :**
 - conjunction of literals e.g. < Sunny ? ? Strong ? Same >
- **Training examples D :**
 - positive and negative examples of the target function, < $x_1, c(x_1)$ >, ... , < $x_n, c(x_n)$ >

- **Determine:**

- A hypothesis h in H such that $h(\mathbf{x})=c(\mathbf{x})$ for all \mathbf{x} in X .

The Inductive Learning Hypothesis



- Any hypothesis found to approximate the target function well over a sufficiently large set of training examples will also approximate the target function well over other unobserved examples.
- Find the hypothesis that best fits the training data
- 根据已知推断未知，假定已知满足某种规律

Number of Instances, Concepts, Hypotheses



- ◆ Sky: sunny, cloudy, rainy
- ◆ Air Temp: warm, cold
- ◆ Humidity: normal, high
- ◆ Wind: strong, weak
- ◆ Water: warm, cold
- ◆ Forecast: same, change

#distinct instances : $3*2*2*2*2*2 = 96$

#distinct concepts : 2^{96}

#syntactically distinct hypotheses : $5*4*4*4*4*4=5120$

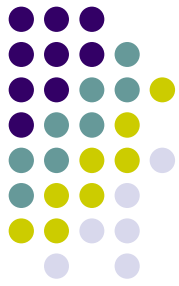
#semantically distinct hypotheses : $1+4*3*3*3*3*3=973$

Perspective



- Concept learning can be formulated as a **searching:**
 - *through a predefined space of potential hypotheses for the hypothesis that **best fits** the training examples.*
- *General-to-specific* ordering
$$h_j \geq_g h_k \iff (\forall x \in X)[(h_k(x) = 1) \rightarrow (h_j(x) = 1)]$$
 - Example : $\langle \text{Sunny}, ?, ?, ?, ?, ? \rangle \geq \langle \text{Sunny}, ?, ?, \text{Strong}, ?, ? \rangle$
 - Introduce a **hierarchy** structure into hypotheses space, which leads to efficient searching strategy.

Algorithms



Algorithm	Order	Strategy	N/P
FIND-S	Specific-to-general	Top-down	Positive
LIST-THEN-ELIMINATE	General-to-Specific	Bottom-up	Negative
CANDIDATE-ELIMINATION	Bi-directional	Bi-directional	Both



FIND-S

- $h_0 = \langle 0, 0, 0, 0, 0, 0 \rangle$
- $h_1 = \langle \text{Sunny, Warm, Normal, Strong, Warm, Same} \rangle$
- $h_2 = \langle \text{Sunny, Warm, ?, Strong, Warm, Same} \rangle$
- $h_3 = \langle \text{Sunny, Warm, ?, Strong, Warm, Same} \rangle$
- $h_4 = \langle \text{Sunny, Warm, ?, Strong, ?, ?} \rangle$

Training examples:

1. $\langle \text{Sunny, Warm, Normal, Strong, Warm, Same} \rangle$, Enjoy Sport = **Yes**
2. $\langle \text{Sunny, Warm, High, Strong, Warm, Same} \rangle$, Enjoy Sport = **Yes**
3. $\langle \text{Rainy, Cold, High, Strong, Warm, Change} \rangle$, Enjoy Sport = **No**
4. $\langle \text{Sunny, Warm, High, Strong, Cool, Change} \rangle$, Enjoy Sport = **Yes**

Report the *most specific* hypothesis



LIST-THEN-ELIMINATE

- $h_0 = \langle ?, \quad ?, \quad ?, \quad ?, \quad ?, \quad ? \rangle$
- $h_1 = \langle \text{Sunny}, ?, \quad ?, \quad ?, \quad ?, \quad ? \rangle$ or
 $\langle ?, \quad \text{Warm}, ?, \quad ?, \quad ?, \quad ? \rangle$ or
 $\langle ?, \quad ?, \quad \text{Normal}, ?, \quad ?, \quad ? \rangle$ or
 $\langle ?, \quad ?, \quad ?, \quad \text{Weak}, ?, \quad ? \rangle$ or
 $\langle ?, \quad ?, \quad ?, \quad ?, \quad \text{Cold}, ? \rangle$ or
 $\langle ?, \quad ?, \quad ?, \quad ?, \quad ?, \quad \text{Same} \rangle$

$\langle \text{Rainy}, \text{Cold}, \text{High}, \text{Strong}, \text{Warm}, \text{Change} \rangle, \text{Enjoy Sport} = \mathbf{No}$

Report the *most general* hypothesis



CANDIDATE-ELIMINATION (1)

S0: {<0, 0, 0, 0, 0, 0>}



S1: {<Sunny, Warm, Normal, Strong, Warm, Same>}



S2: {<Sunny, Warm, ?, Strong, Warm, Same>}

G0, G1, G2: {<?, ?, ?, ?, ?, ?>}

Training examples:

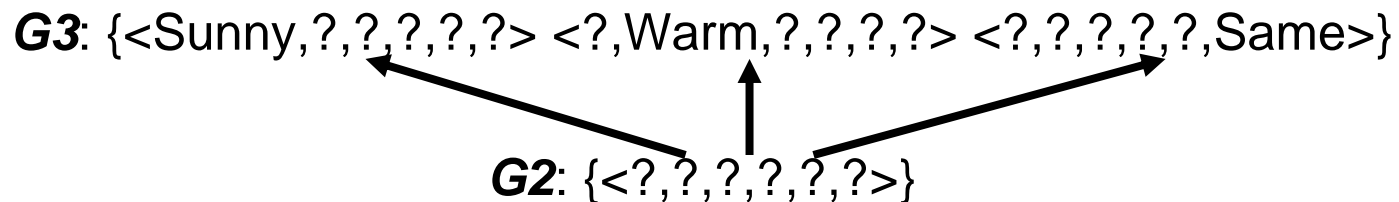
1. <Sunny, Warm, Normal, Strong, Warm, Same>, Enjoy Sport = **Yes**
2. <Sunny, Warm, High, Strong, Warm, Same>, Enjoy Sport = **Yes**

遇到正例与**Find-s**方法类似



CANDIDATE-ELIMINATION (2)

S2, S3: {<Sunny, Warm, ?, Strong, Warm, Same>}



Training examples:

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

遇到反例与**List-then-eliminate**方法类似



CANDIDATE-ELIMINATION (3)

S3: {<Sunny, Warm, ?, Strong, Warm, Same>}



S4: {<Sunny, Warm, ?, Strong, ?, ?>}

G4: {<Sunny, ?, ?, ?, ?, ?> <?, Warm, ?, ?, ?, ?>}



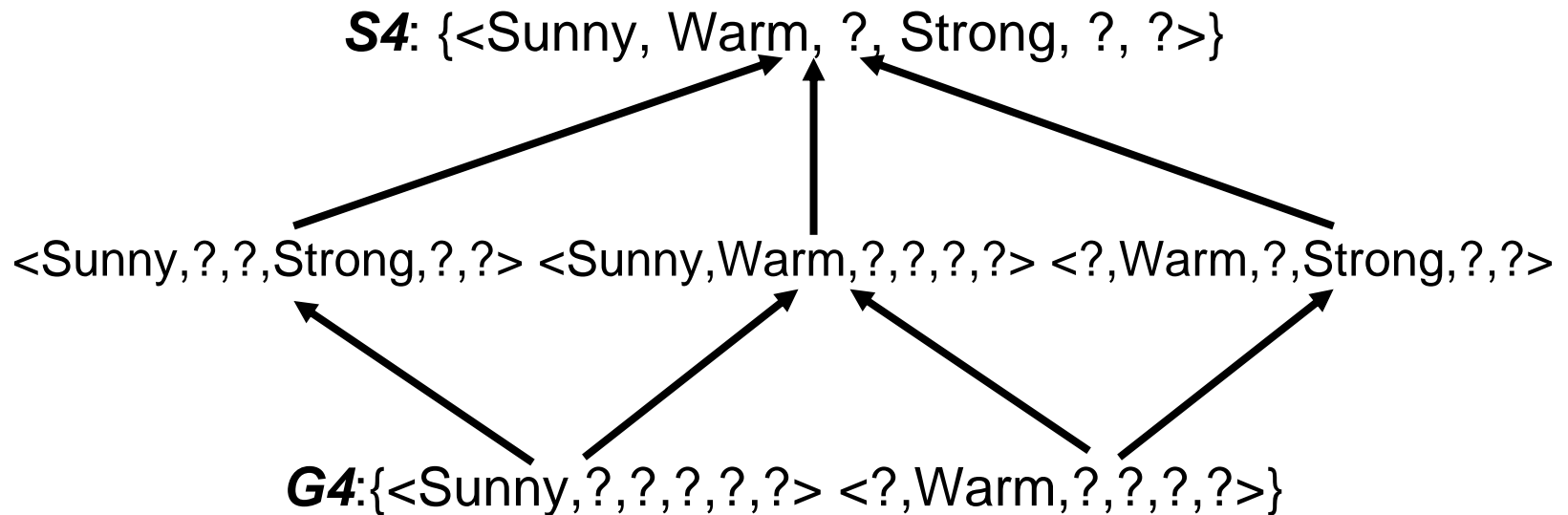
G3: {<Sunny, ?, ?, ?, ?, ?> <?, Warm, ?, ?, ?, ?> <?, ?, ?, ?, ?, Same>}

Training examples:

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



Final Version Space



Report the *version space* – all possible hypotheses

<Sunny,Warm,Normal,Strong,Cool,Change>

<Rainy,Cold,Normal,Light,Warm,Same>

<Sunny,Warm,Normal,Light,Warm,Same>

<Sunny,Cold,Normal,strong,Warm,Same>

Remarks



- **Convergence Condition**
 - Noise Free (No Errors)
 - The target concept **DOES** exist in the searching hypotheses space ***H***
- **What Training Example Should the Learner Request Next?**
 - Satisfy half the hypotheses in the current version space
 - Fastest Convergence, Least Sample Needed, Best Uncertainty Elimination
- **How Can Partially Learned Concepts Be Used?**
 - Absolutely Accept <Sunny,Warm,Normal,Strong,Cool,Change>
 - Absolutely Deny <Rainy,Cold,Normal,Light,Warm,Same>
 - Pending
 - <Sunny,Warm,Normal,Light,Warm,Same>
 - <Sunny,Cold,Normal,Strong,Warm,Same>



Inductive Bias (归纳的偏差)

- Bias Vs. Unbiase
- The **Futility** (无用性) of Bias-Free Learning
 - Too large searching space
 - The convergence is impossible
 - Rational inference is impossible
- Inductive bias of CANDIDATE-ELIMINATION algorithm
 - The target concept c is contained in the given hypothesis space H .
 - Inductive System == Deductive System + **Inductive Bias**

Summary



- Concept learning can be cast as **Searching** through predefined hypotheses space.
- The **general-to-specific** partial ordering of hypotheses leads to efficient searching strategy, such as CANDIDATE-ELIMINATION algorithm.
- A practical concept learning methods must employ **inductive bias**. Otherwise, they can only classify the observed training examples.
- Version spaces and the CANDIDATE-ELIMINATION algorithm provide a useful **conceptual framework** for studying concept learning. However, their correctness rely on the noise-free training examples and the ability of provided hypotheses space to express the unknown target concepts.



EnjoySport revisit

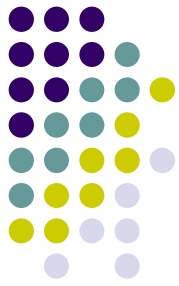
- **Given:**

- Instances **X** : Possible days, each described by the attributes
 - Sky (*Sunny, Cloudy, and Rainy*)
 - Temp (*Warm and Cold*)
 - Humidity (*Normal and High*)
 - Wind (*Strong and Weak*)
 - Water (*Warm and Cool*)
 - Forecast (*Same and Change*)
- Hypotheses **H** : Each hypothesis is described by a **conjunction** of constraints. These constraints may be “?” (any value), “0” (no value), or a specific value.

设想编程调试时候的情景

- **Determine:**

- A hypothesis h in **H** such that $h(x) = c(x)$ for all x in **X** .



Thank you