Machine Learning: Symbolische Ansätze

TECHNISCHE UNIVERSITÄT DARMSTADT

Introduction

- Machine Learning
  - Problem definition
  - Example Tasks
- Dimensions of Machine Learning Problems
  - Example Representation
  - Concept Representation
  - Learning Tasks
  - Evaluation Scenarios
- Induction of Classifiers
  - Characteristics of this framework
  - A small example
- Learning and Search
  - Generalization and Bias
- Data Mining
  - Motivation
  - Relation to Machine Learning

Was ist Lernen?

TECHNISCHE UNIVERSITÄT DARMSTADT

„Lernen ist der Sammelname für Vorgänge, Prozesse oder nicht unmittelbar beobachtbare Veränderungen im Organismus, die durch Erfahrungen entstehen und zu Veränderungen des Verhaltens führen."

[Bergius,1971]

„Lernen bedeutet Veränderungen in der Wahrscheinlichkeit, mit der Verhaltensweisen in bestimmten Situationen auftreten."

[Hilgard,1973]

„Lernen ist eine Verhaltensänderung, die *nicht* durch Reifungsvorgänge, Verletzungen oder Erkrankungen, Ermüdungsprozesse oder durch Anlagen erklärt werden kann."

[Joerger,1976]

Machine Learning

TECHNISCHE UNIVERSITÄT DARMSTADT

„Learning denotes changes in the system that ... enable the system to do the same task or tasks drawn from the same population more efficiently and more effectively the next time."

[Simon,1983]

„Learning is making useful changes in our minds."

[Minsky,1985]

„Learning is constructing or modifying representations of what is being experienced."

[Michalski,1986]

Machine Learning Problem Definition

TECHNISCHE UNIVERSITÄT DARMSTADT

- Definition (Mitchell 1997)

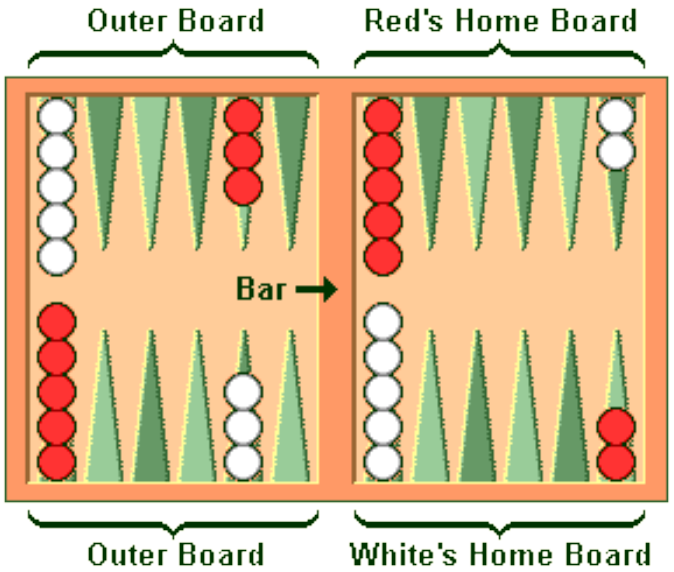
> „A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.“

- Given:
  - a task T
  - a performance measure P
  - some experience E with the task
- Goal:
  - generalize the experience in a way that allows to improve your performance on the task

Learning to Play Backgammon

TECHNISCHE UNIVERSITÄT DARMSTADT

- Task:
  - play backgammon
- Performance Measure:
  - percentage of games won
- Experience:
  - previous games played



TD-Gammon:

- learned a neural network for evaluating backgammon boards
- from playing millions of games against itself
- successively improved to world-champion strength
- `http://www.research.ibm.com/massive/tdl.html`
  GNU Backgammon: `http://www.gnu.org/software/gnubg/`

Recognizing Spam-Mail

TECHNISCHE UNIVERSITÄT DARMSTADT

- Task:
  - sort E-mails into categories (e.g., Regular / Spam)
- Performance Measure:
  - Weighted Sum of Mistakes (letting spam through is not so bad as misclassifying regular E-mail as spam)
- Experience:
  - Handsorted E-mail messages in your folder

In Practice:

- Many Spam-Filters (e.g., Mozilla) use Bayesian Learning for recognizing spam mails

Market Basket Analysis

TECHNISCHE UNIVERSITÄT DARMSTADT

- Task:
  - discover items that are frequently bought together
- Performance Measure:
  - ? (revenue by making use of the discovered patterns)
- Experience:
  - Supermarket check-out data

Myth:

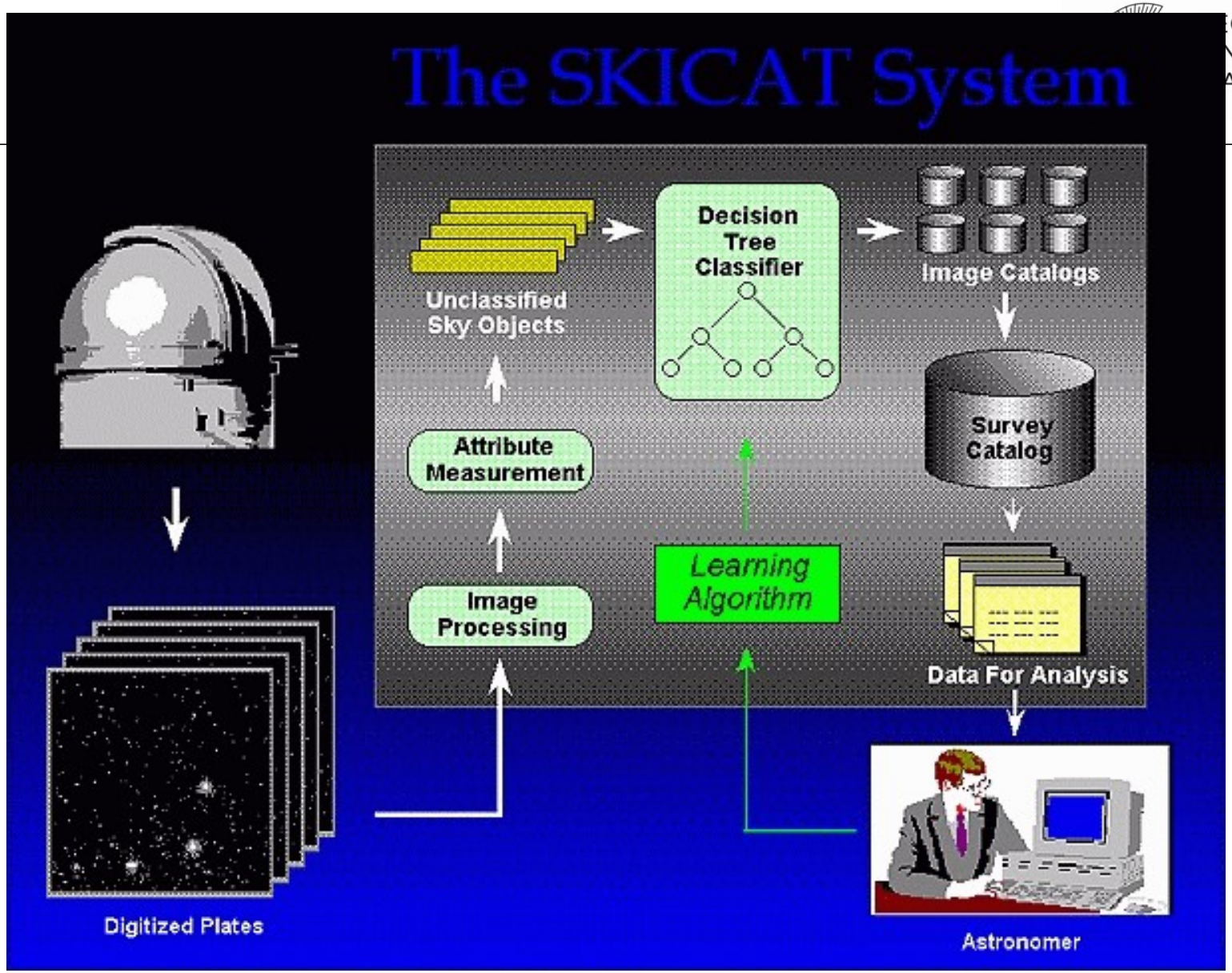
- The most frequently cited result is:

  diapers $\rightarrow$ beer

Learning to Classify Stars

TECHNISCHE UNIVERSITÄT DARMSTADT

- Task:
  - Classification of celestial bodies
- Data:
  - 3000 images (23,040 x 23,040 pixels) of the Palomar Sky Observatory, 3 Terabytes of data,
  - classified into ≈ 10^7 galaxies, ≈ 10^8 stars, ≈ 10^5 quasars
  - representation with 40 attributes (image-processing)
- Method:
  - learning of multiple decision trees
  - combining the best rules of each tree
- SKICAT Performance:
  - 94% accuracy, better than astronomers
  - discovery of 16 new quasars (12/95)


The SKICAT System
Unclassified Sky Objects
Decision Tree Classifier
Image Catalogs
Survey Catalog
Data For Analysis
Learning Algorithm
Attribute Measurement
Image Processing
Digitized Plates
Astronomer

Dimensions of Learning Problems

TECHNISCHE UNIVERSITÄT DARMSTADT

- Example Representation
  - attribute-value data vs. first-order logic
- Type of training information
  - supervised vs. unsupervised learning
- Availability of training examples
  - batch learning vs. on-line learning (incremental learning)
- Concept representation
  - IF-THEN rules, decision trees, neural networks...
- Learning algorithm
  - divide-and-conquer, back-propagation,...
- Evaluation scenario
  - estimating predictive performance, cost-sensitive-learning,

Example Representation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Attribute-Value data:
  - Each example is described with values for a fixed number of attributes
    - **Nominal Attributes:**
      - store an unordered list of symbols (e.g., *color*)
    - **Numeric Attributes:**
      - store a number (e.g., *income*)
    - **Other Types:**
      - ordered values
      - hierarchical attributes
      - set-valued attributes
  - the data corresponds to a single relation (spreadsheed)
- Multi-Relational data:
  - The relevant information is distributed over multiple relations
  - Inductive Logic Programming

Type of Training Information

TECHNISCHE UNIVERSITÄT DARMSTADT

- **Supervised Learning:**
  - A teacher provides the value for the target function for all training examples (labeled examples)
  - concept learning, classification, regression
- **Semi-supervised Learning:**
  - Only a subset of the training examples are labeled (labeling examples is expensive!)
- **Reinforcement Learning:**
  - A teacher provides feedback about the values of the target function chosen by the learner
- **Unsupervised Learning:**
  - There is no information except the training examples
  - clustering, subgroup discovery, association rule discovery

Example Availability

TECHNISCHE UNIVERSITÄT DARMSTADT

- Batch Learning
  - The learner is provided with a set of training examples
- Incremental Learning / On-line Learning
  - There is constant stream of training examples
- Active Learning
  - The learner may choose an example and ask the teacher for the relevant training information

Concept Representation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Most Learners generalize the training examples into an explicit representation
  (called a model, function, hypothesis, concept...)
  - mathematical functions (e.g., polynomial of 3rd degree)
  - logical formulas (e.g., propositional IF-THEN rules)
  - decision trees
  - neural networks
  - ....
- Lazy Learning
  - do not compute an explicit model
  - generalize „on demand“ for a given training example
  - example: nearest neighbor classification

A Selection of Learning Techniques

TECHNISCHE UNIVERSITÄT DARMSTADT

- Decision and Regression Trees
- Classification Rules
- Association Rules
- Inductive Logic Programming
- Neural Networks
- Support Vector Machines
- Statistical Modeling
- Clustering Techniques
- Case-Based Reasoning
- Genetic Algorithms
- ...

Evaluation of Learned Models

TECHNISCHE UNIVERSITÄT DARMSTADT

- Validation through experts
  - a domain experts evaluates the plausibility of a learned model
    - \+ subjective, time-intensive, costly
    - – but often the only option (e.g., clustering)
- Validation on data
  - evaluate the accuracy of the model on a separate dataset drawn from the same distribution as the training data
    - – labeled data are scarce, could be better used for training
    - \+ fast and simple, off-line, no domain knowledge needed, methods for re-using training data exist (e.g., cross-validation)
- On-line Validation
  - test the learned model in a fielded application
    - \+ gives the best estimate for the overall utility
    - – bad models may be costly

Induction of Classifiers

TECHNISCHE UNIVERSITÄT DARMSTADT

The most „popular“ learning problem:

- Task:
  - learn a model that predicts the outcome of a dependent variable for a given instance
- Experience:
  - experience is given in the form of a data base of examples
  - an example describes a single previous observation
    - *instance:* a set of measurements that characterize a situation
    - *label:* the outcome that was observed in this siutation
- Performance Measure:
  - compare the predicted outcome to the observed outcome
  - estimate the probability of predicting the right outcome in a new situation

Induction of Classifiers

TECHNISCHE UNIVERSITÄT DARMSTADT

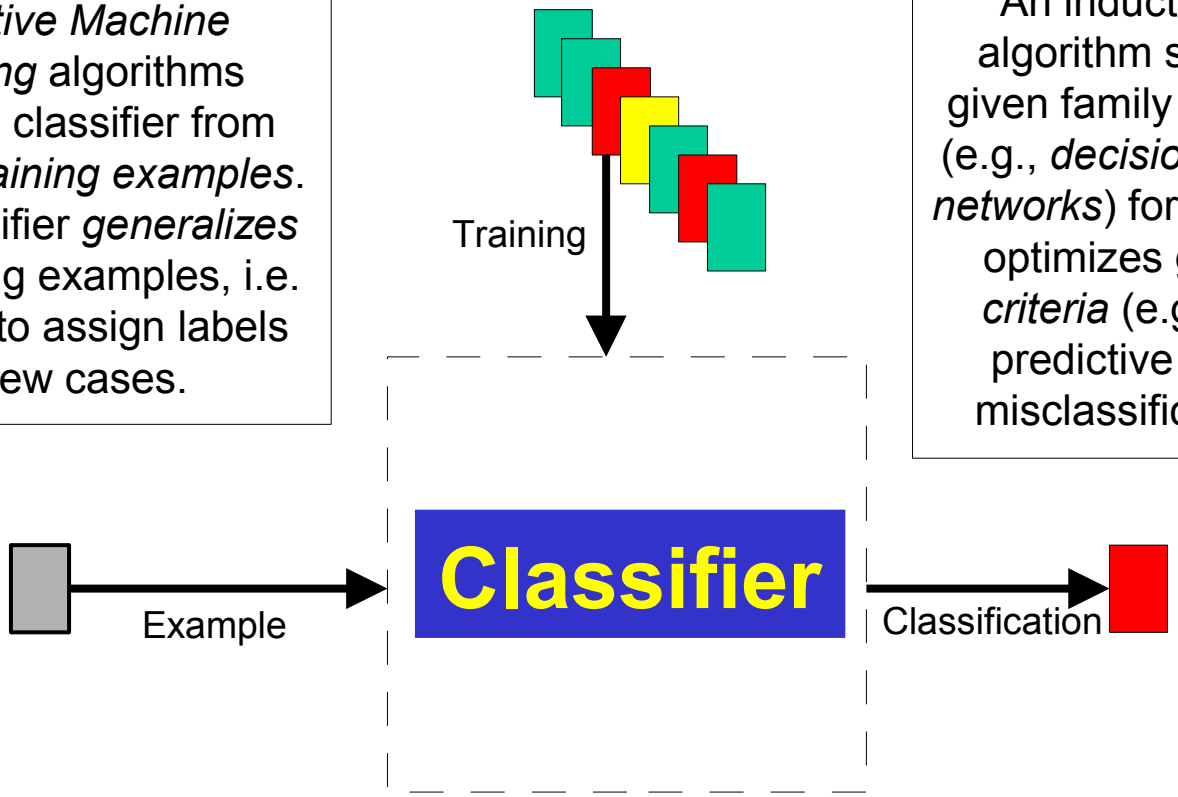
Typical Characteristics

- attribute-value representation (single relation)
- batch learning from off-line data (data are available from external sources)
- supervised learning (examples are pre-classified)
- numerous learning algorithms for practically all concept representations (decision trees, rules, neural networks, SVMs, statistical models,...)
- often greedy algorithms (may not find optimal solution, but fast processing of large datasets)
- evaluation by estimating predictive accuracy (on a portion of the available data)

Induction of Classifiers

TECHNISCHE UNIVERSITÄT DARMSTADT

Inductive Machine Learning algorithms induce a classifier from *labeled training examples*. The classifier *generalizes* the training examples, i.e. it is able to assign labels to new cases.



An inductive learning algorithm searches in a given family of hypotheses (e.g., *decision trees, neural networks*) for a member that optimizes given *quality criteria* (e.g., estimated predictive accuracy or misclassification costs).

A Sample Task

TECHNISCHE UNIVERSITÄT DARMSTADT

Day	Temperature	Outlook	Humidity	Windy	Play Golf?
07-05	hot	sunny	high	false	no
07-06	hot	sunny	high	true	no
07-07	hot	overcast	high	false	yes
07-09	cool	rain	normal	false	yes
07-10	cool	overcast	normal	true	yes
07-12	mild	sunny	high	false	no
07-14	cool	sunny	normal	false	yes
07-15	mild	rain	normal	false	yes
07-20	mild	sunny	normal	true	yes
07-21	mild	overcast	high	true	yes
07-22	hot	overcast	normal	false	yes
07-23	mild	rain	high	true	no
07-26	cool	rain	normal	true	no
07-30	mild	rain	high	false	yes

today	cool	sunny	normal	false	?
tomorrow	mild	sunny	normal	false	?

Rote Learning

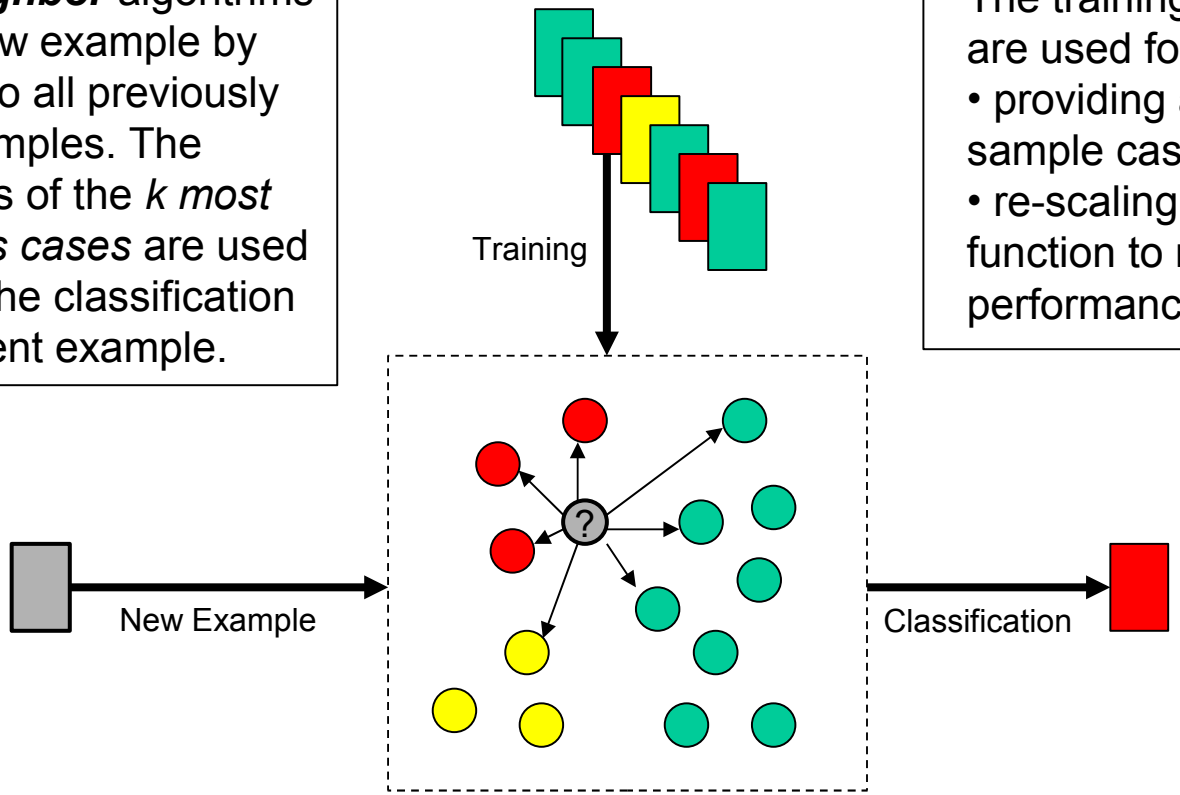
TECHNISCHE UNIVERSITÄT DARMSTADT

Day	Temperature	Outlook	Humidity	Windy	Play Golf?
07-05	hot	sunny	high	false	no
07-06	hot	sunny	high	true	no
07-07	hot	overcast	high	false	yes
07-09	cool	rain	normal	false	yes
07-10	cool	overcast	normal	true	yes
07-12	mild	sunny	high	false	no
07-14	cool	sunny	normal	false	yes
07-15	mild	rain	normal	false	yes
07-20	mild	sunny	normal	true	yes
07-21	mild	overcast	high	true	yes
07-22	hot	overcast	normal	false	yes
07-23	mild	rain	high	true	no
07-26	cool	rain	normal	true	no
07-30	mild	rain	high	false	yes

today	cool	sunny	normal	false	yes
tomorrow	mild	sunny	normal	false	?

Nearest Neighbor Classifier

K-Nearest Neighbor algorithms classify a new example by comparing it to all previously seen examples. The classifications of the *k most similar previous cases* are used for predicting the classification of the current example.



The training examples are used for
- providing a library of sample cases
- re-scaling the similarity function to maximize performance

Nearest Neighbor

TECHNISCHE UNIVERSITÄT DARMSTADT

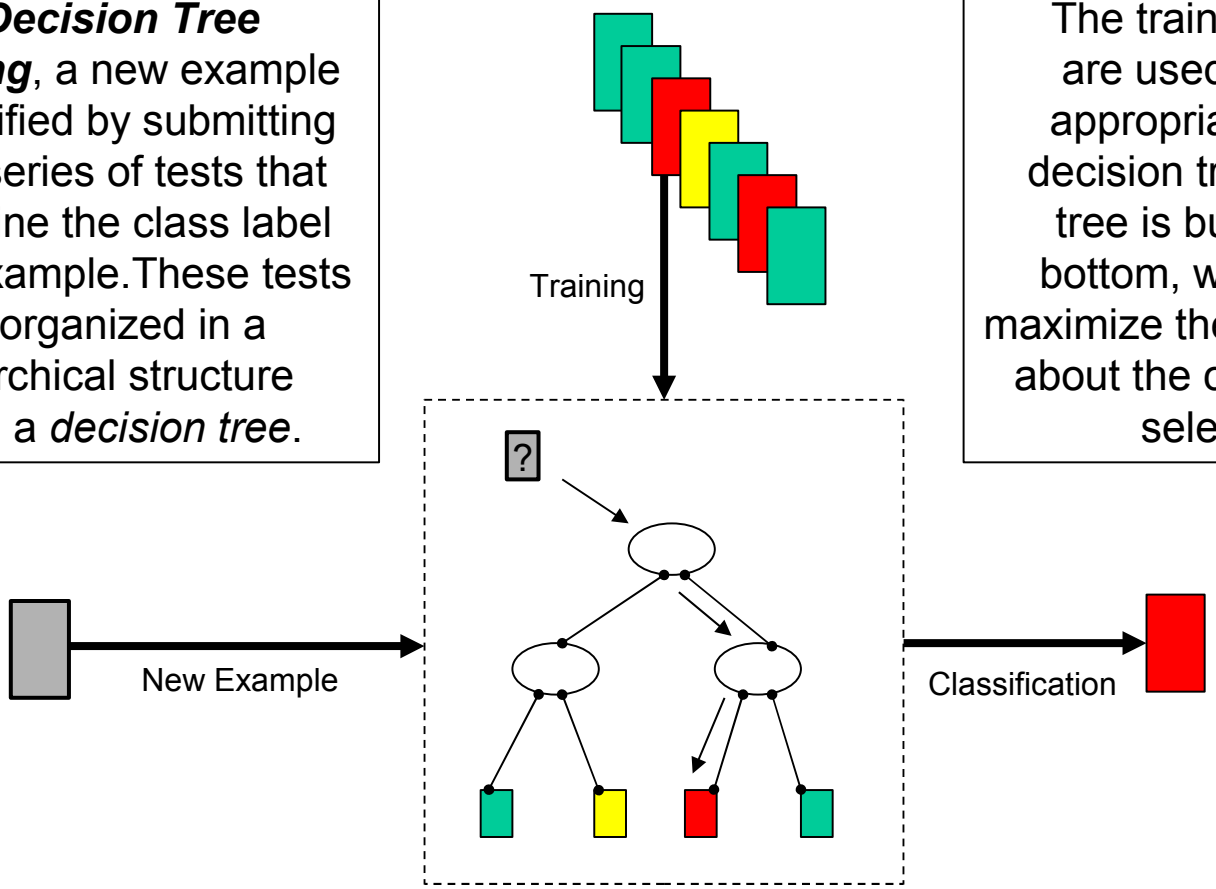
Day	Temperature	Outlook	Humidity	Windy	Play Golf?
07-05	hot	sunny	high	false	no
07-06	hot	sunny	high	true	no
07-07	hot	overcast	high	false	yes
07-09	cool	rain	normal	false	yes
07-10	cool	overcast	normal	true	yes
07-12	mild	sunny	high	false	no
07-14	cool	sunny	normal	false	yes
07-15	mild	rain	normal	false	yes
07-20	mild	sunny	normal	true	yes
07-21	mild	overcast	high	true	yes
07-22	hot	overcast	normal	false	yes
07-23	mild	rain	high	true	no
07-26	cool	rain	normal	true	no
07-30	mild	rain	high	false	yes
tomorrow	mild	sunny	normal	false	yes

Decision Trees

- a decision tree consists of
  - **Nodes:**
    - test for the value of a certain attribute
  - **Edges:**
    - correspond to the outcome of a test
    - connect to the next node or leaf
  - **Leaves:**
    - terminal nodes that predict the outcome
- an example is classified
  1. start at the root
  2. perform the test
  3. follow the corresponding edge
  4. goto 2. unless leaf
  5. predict that outcome associated with the leaf

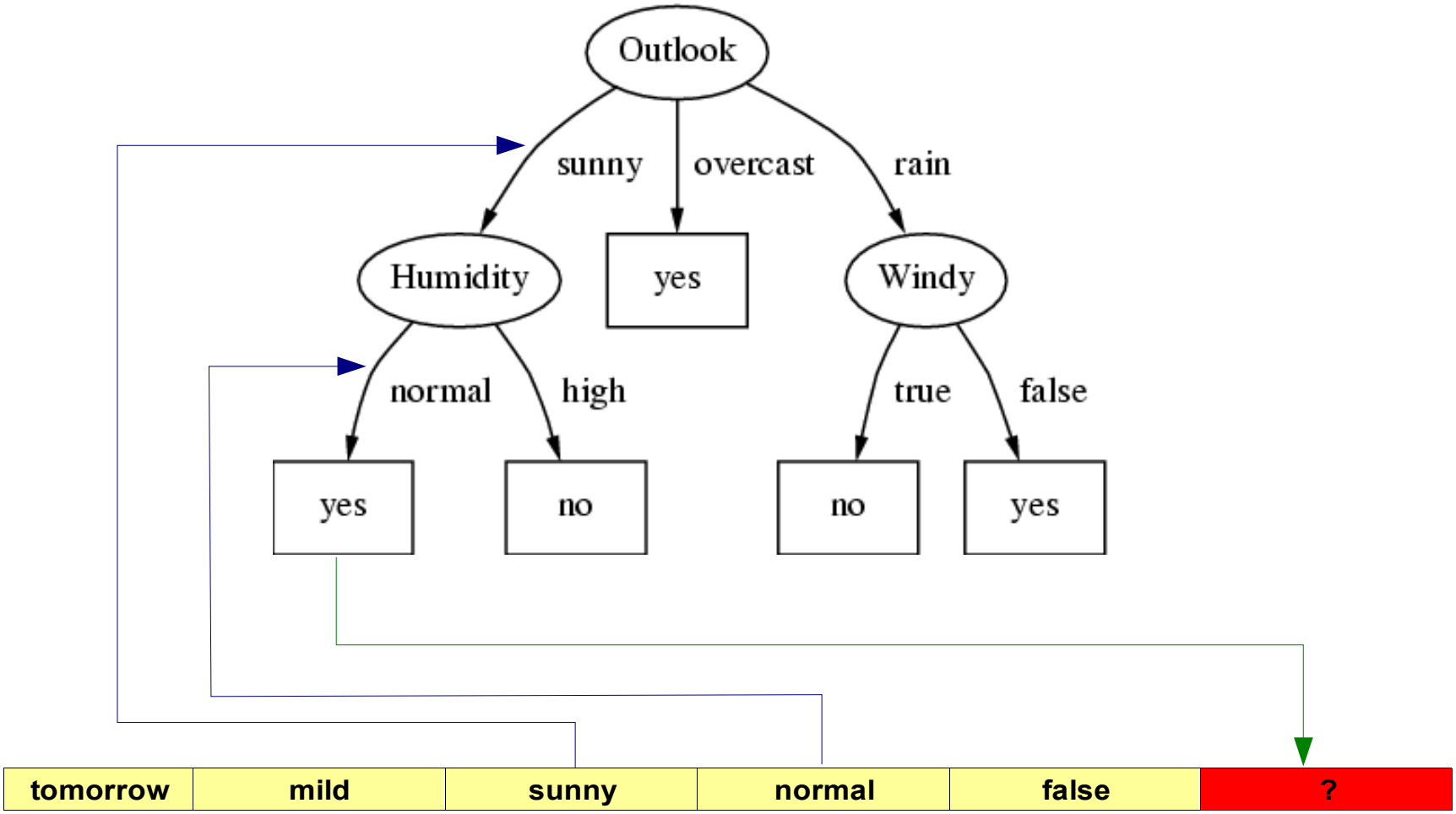
Decision Tree Learning

In ***Decision Tree Learning***, a new example is classified by submitting it to a series of tests that determine the class label of the example.These tests are organized in a hierarchical structure called a *decision tree*.

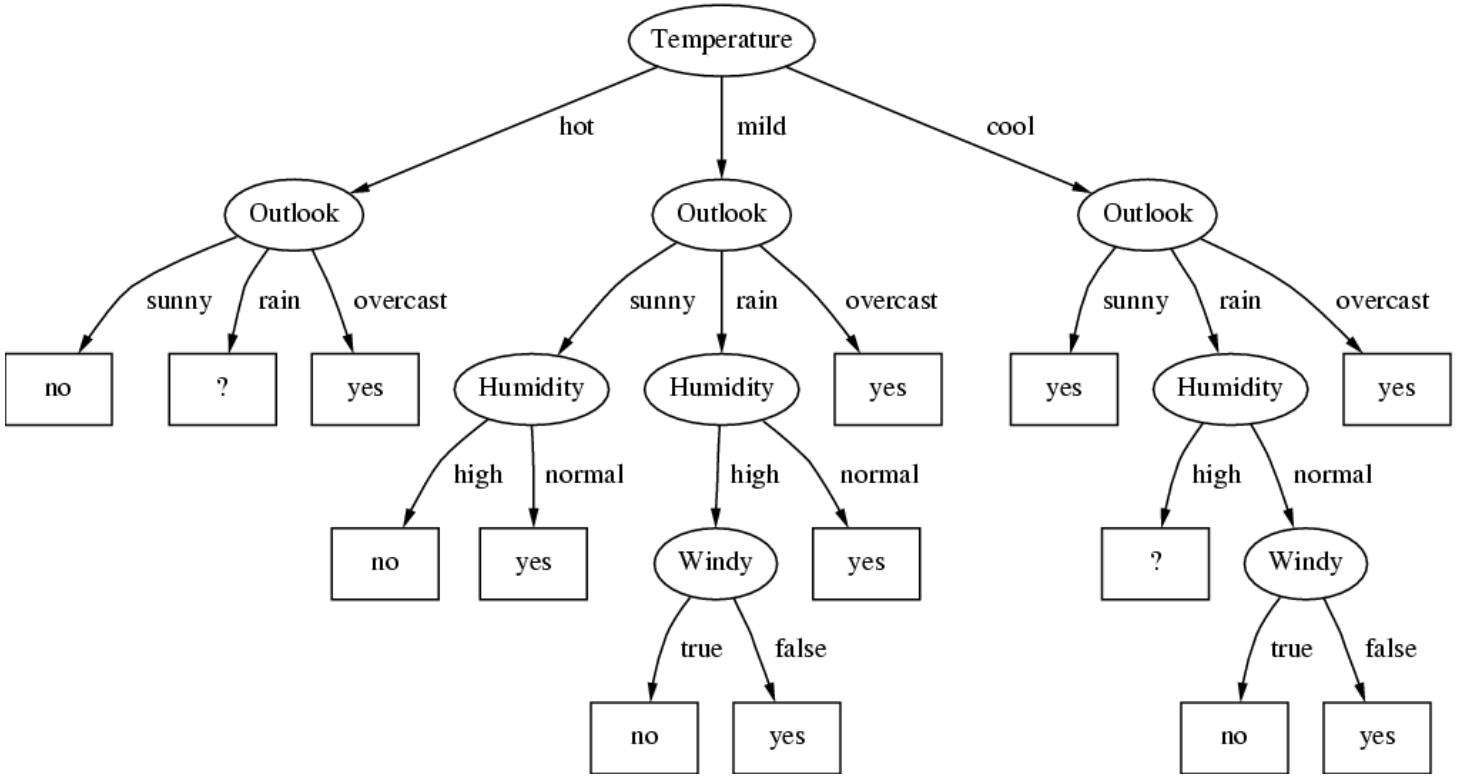


The training examples are used for choosing appropriate tests in the decision tree. Typically, a tree is built from top to bottom, where tests that maximize the information gain about the classification are selected first.

Decision Tree Learning



A Different Decision Tree



- also explains all of the training data
- will it generalize well to new data?

Learning and Search

TECHNISCHE UNIVERSITÄT DARMSTADT

- Learning may be viewed as a search problem
  - **Search space:** the space of all possible hypotheses of the chosen hypothesis class (e.g., all decision trees)
  - **Find:** find a hypothesis that is likely to underly the data
- Different search techniques:
  - exhaustive search
    - enumerate all hypotheses
    - typically infeasible (some hypotheses spaces are even infinite!)
  - greedy search
    - incrementally build up a solution
    - use heuristics to choose the next solution step
  - randomized search
    - e.g., evolutionary algorithms

Bias and Generalization

TECHNISCHE UNIVERSITÄT DARMSTADT

Bias: (Machine Learning Definition)
Any criterion that prefers one concept over another except for completeness/consistency on the training data.

- **Language Bias:**
  Choose a hypothesis representation language
- **Selection Bias:**
  Which hypotheses will be preferred during the search?
- **Overfitting Avoidance Bias:**
  Avoid too close approximations to training data

- Bias is necessary for generalization
  - without bias all complete and consistent hypotheses (those that correctly explain all training examples) are equally likely
  - for any example, half of them will predict one class, the other half the opposite class (*no free lunch theorems*)