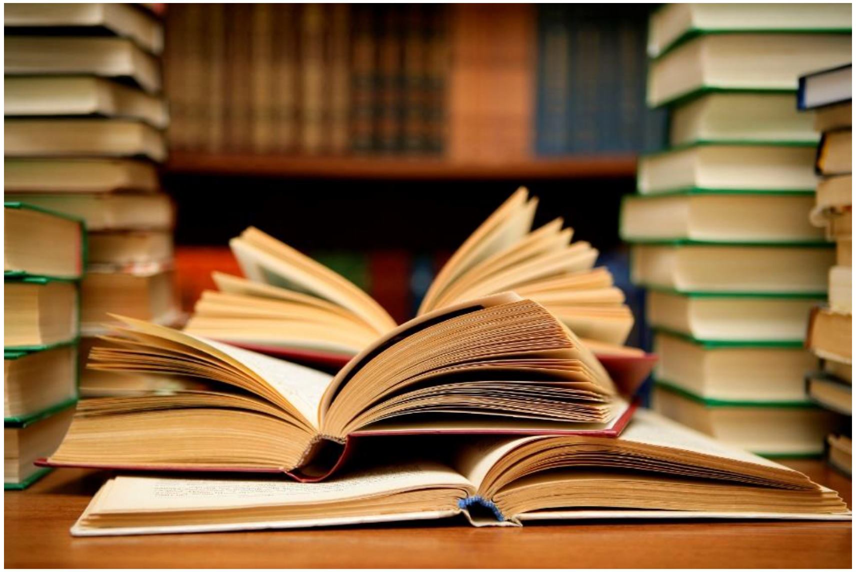
Telecooperation Lab
Prof. Dr. Max Mühlhäuser

TK3: Ubiquitous Computing

Chapter 3: Context-aware Computing
Part 1.5: ML Basics
Lecturer: Dr. Immanuel Schweizer

Sources

- Slides by Frederik Janssen (KE, TK)
- Sources used to prepare the slides
  - Wikipedia (http://en.wikipedia.org) for figures and algorithm properties
  - VL “Maschinelles Lernen” by Prof. Johannes Fürnkranz (the lectures given in all the different semesters are available under http://www.ke.tu-darmstadt.de/lehre)
  - VL “Künstliche Intelligenz” by Prof. Johannes Fürnkranz
  - Various other internet sources



Example: Pipeline

Sensor Data

- Noise Data
- Weather Data
- Street Data
- Building Data

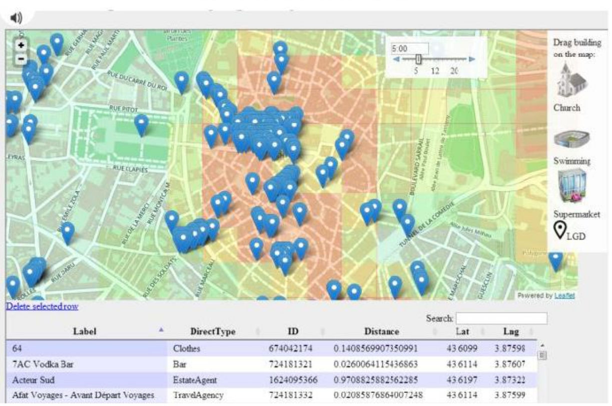
Classification

- Feature Generation
- Decision Tree Classification

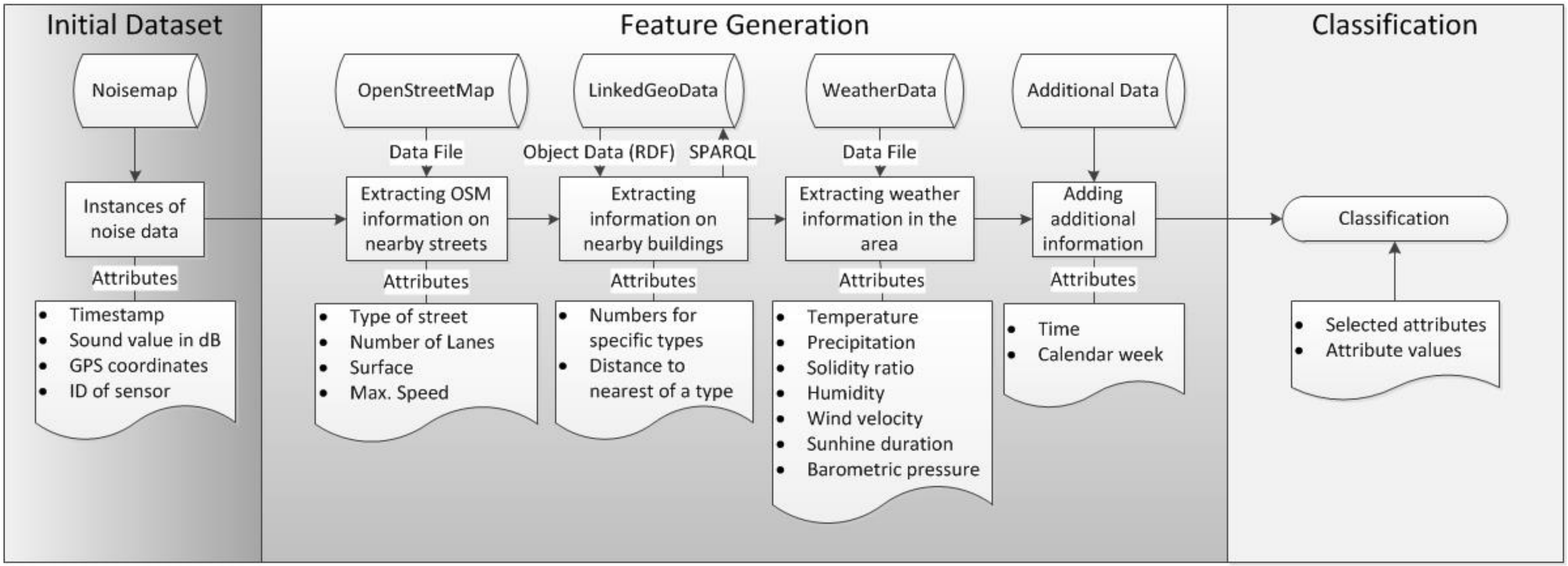
Evaluation and Optimization

- Feature Optimization
- Feature Selection

Visualization



Example: Feature Extraction



Sensor Data Analysis

- Segmentation
  - Sensors usually produce data continuously
  - Choose time-granularity to predict the context
  - Detect segments in the data signal, e.g., using local thresholds
- Raw sensor data
  - Contains redundant information
  - May confuse the classifier
- Features
  - Calculated based on the raw sensor data
  - Usually domain specific
  - Examples: mean, variance, standard deviation, zero crossings, FFT coefficients, integration, ...
- Machine Learning
  - Training set consisting of features and class labels
  - Method learns to predict the class label given the features
  - Methods: k-NN, SVN, decision trees, neuronal nets, ...

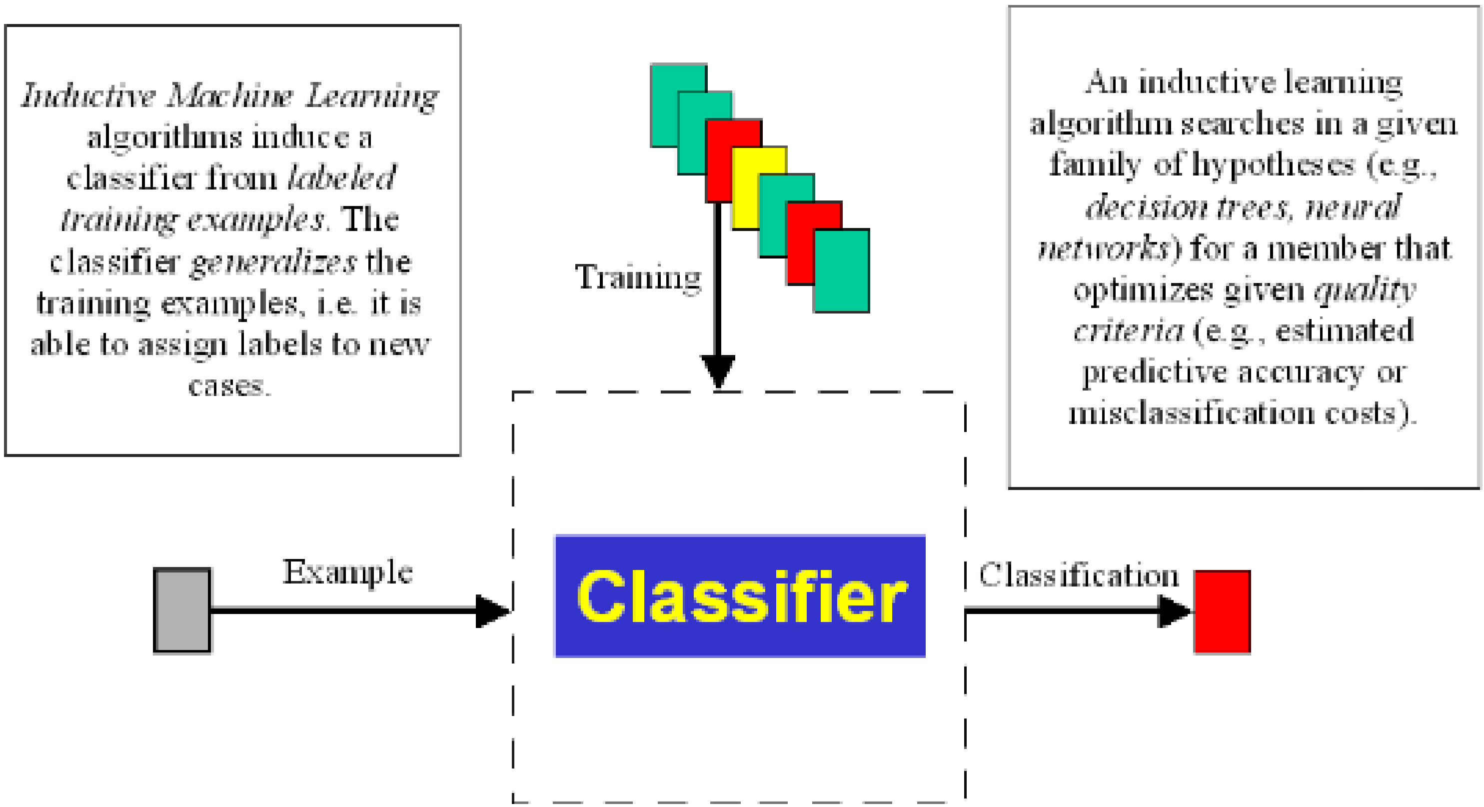
Machine Learning

- 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

ML: Classification

- 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 plus a label
    - Label: the outcome that was observed in this situation (usually the number of labels (or classes) can be greater than 2, here, we most often assume binary classification
- 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



Data Representation

- Each example (or instance) is described with values for a fixed number of attributes (also called features)
  - 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
- Note that attribute values can also be missing (usually marked with a "?")

Example

Day	Temperature	Outlook	Humidity	Windy	Play Golf?
07-05	26	sunny	high	false	no
07-06	28	sunny	high	true	no
07-07	29	overcast	high	false	yes
07-09	23	rain	normal	false	yes
07-10	20	overcast	normal	true	yes
07-12	12	sunny	high	false	no
07-14	8	sunny	normal	false	yes
07-15	25	rain	normal	false	yes
07-20	18	sunny	normal	true	yes
07-21	18	overcast	high	true	yes
07-22	20	overcast	normal	false	yes
07-23	19	rain	high	true	no
07-26	11	rain	normal	true	no
07-30	16	rain	high	false	yes

today	9	sunny	normal	false	?
tomorrow	13	sunny	normal	false	?

Concept Representation

- Several different ways to represent the learned concept
  - Interpretable models (i.e., decision trees or rule sets)
  - Hyperplanes (SVMs or neural networks)
  - Coefficients (weights) of a linear model
  - Table of conditional probabilities (naive bayes)
  - None at all (lazy learning)
- Two important criteria when searching for a good concept
  - Overfitting Avoidance
  - Occam’s Razor
- Two ways of representation:
  - Learn a single model
  - Learn many models and combine them (ensemble learning or meta learning)
    - Bagging, boosting, stacking, ...
  - Advantages: confidence in classification is higher due to the combination of many models
  - Disadvantages: takes more time to learn and more storage space

Supervised Learning

- A teacher provides correct labels for all examples
- Usually the last attribute encodes the class (or target) attribute
- Different settings:
  - All examples are available (batch learning)
  - Iterative learning or stream mining: examples arrive one by one
- Two ways to deal with labeled data:
  - Eager learning: a model (concept) is learned on the labeled data
    - This model is used to classify unseen examples
  - Lazy learning: no model is learned
    - When a new unseen training example should be classified, the classification is build directly on the training data

Day	Temperature	Outlook	Humidity	Windy	Play Golf?
07-05	26	sunny	high	false	no
07-06	28	sunny	high	true	no
07-07	29	overcast	high	false	yes
07-09	23	rain	normal	false	yes
07-10	20	overcast	normal	true	yes
07-12	12	sunny	high	false	no
07-14	8	sunny	normal	false	yes
07-15	25	rain	normal	false	yes
07-20	18	sunny	normal	true	yes
07-21	18	overcast	high	true	yes
07-22	20	overcast	normal	false	yes
07-23	19	rain	high	true	no
07-26	11	rain	normal	true	no
07-30	16	rain	high	false	yes
today	9	sunny	normal	false	?
tomorrow	13	sunny	normal	false	?

Unsupervised Classification

- No information of the target class of the examples is given
  - Algorithms try to find regularities in the data
  - Somewhat hard to evaluate
  - Related to density estimation in statistics

- Methods:
  - Clustering: hierarchical clustering, centroid-based clustering, distribution-based clustering, density-based clustering, etc.
  - Association Rule Learning: Find a relation in a huge database (Diapers -> Beer)
  - Subgroup Discovery: Find a interesting set of instances in a database

Semi-supervised Learning

- In semi-supervised learning only a part of the training data is labeled (typically only a small amount is labeled)
- However, the learner has access to all data (labeled and unlabeled)
- Different techniques:
  - Self-training: the learning algorithms learns a model on the labeled data and uses this model to classify the unlabeled examples, then the model is re-learned
  - Active learning: the learner suggests unlabeled examples that can then be labeled by an expert/a teacher, only those examples are requested to be labeled that are important for the algorithm, i.e., from which the learner is able to improve the model the most (e.g., decision boundary)
  - Co-training: two classifiers are learned on two different views on the data (ideally conditionally independent), the most confident predictions on the unlabeled data are then used to label more and more training data
    - Note that this is a very strong technique: in the original publication 95% of 788 web pages were labeled correctly with the use of only 12 labeled pages (Blum and Mitchell, 1998)

Regression

- Contrary to classification in regression the target variable is numeric (therefore also called function learning)
  - A lot of previous work was done in the statistics community, rather new topic in the machine learning community
- Different ways to deal with this situation:
  1. Either by discretizing the target variable and use standard classification algorithms, or
  2. By working on the numeric targets directly

- Main disadvantage of the first case: we do not know the right number of classes in advance
- Main disadvantage of the latter: we have to adapt the algorithms

Data Stream Mining

- Also called online learning, i.e., examples arrive one by one
- The model is refined with each new example
- Important observations:
  - When should old examples be excluded from the current model (concept drift)
  - When should all examples be skipped and a completely new model should be learned (concept shift)
- Algorithms:
  - Windowing: Only keep a window of the examples
  - ID5R: decision tree learner for iterative learning, refines the decision tree with each example by rotating it (to keep it balanced)

Frameworks

- Many frameworks for machine learning are available, often open source
- Still the most popular one: weka
  - (http://www.cs.waikato.ac.nz/ml/weka/)
- Nice alternative: Rapid Miner (http://rapid-i.com/)
- These two frameworks provide implementations of a variety of different algorithms
- But there are also many frameworks that are specialized for a certain task
  - NTLK: framework for natural language processing (http://www.nltk.org/)
  - SVMlight: implementation of a fast support vector machine (http://svmlight.joachims.org/)
  - LibSVM for weka (http://www.csie.ntu.edu.tw/~cjlin/libsvm/)
  - Theano for deep learning (http://deeplearning.net/software/theano/)
  - And many more...