Technische Hochschule Rosenheim
Technical University of Applied Sciences

Introduction

Introduction to AI – Part II

Johannes Jurgovsky

Summer term 2023

Introduction

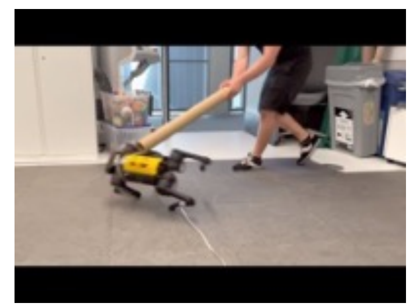
How to solve complex real world problems?

- Object Detection

- Sentiment Classification

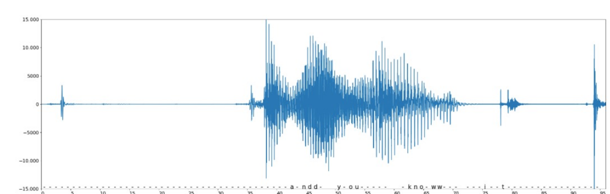
„I bought this drinking bottle for my daughter. After only two weeks use the lid fell apart and now I'm left with a very expensive mug.“

- Continuous Control Tasks



https://danijar.com/project/daydreamer/

- Automatic Speech Recognition (ASR)



Han, Songshen & Xu, Kaiyong & Guo, Songhui & Yu, Miao & Yang, Bo. (2022). Evading Logits-Based Detections to Audio Adversarial Examples by Logits-Traction Attack. Applied Sciences. 12.

Introduction

Central Paradigm

Modeling

- The process of describing an idealized version of the problem in terms of variables and interactions.
- Some variables might have unknown values (parameters) -> the model is <u>incomplete</u>.

```
if nYellowPixels(I,x,y) / nPixelsInWindow(I,x,y) > w
then TrafficSign
```

Learning

- The process of exploiting <u>data</u> to adjust parameters of the <u>model</u> such that its <u>predictions</u> match our expectation.

E.g.: Data is a collection of images containing traffic signs and pixel-wise annotations.

Inference

- The process of calculating the model‘s <u>predictions</u> based on evidence.

E.g.: Given a particular image and a particular pixel location, is this pixel part of a TrafficSign? Involves counting pixels within some window.

Content

Machine Learning

Given description of environment, make a prediction for a given input.

- Environment described as parameterized model fitted on data from the problem domain
- Output: A single numerical value for a given input object from that domain

Incomplete Model

Data

→ **~~Incomplete~~ Model**

- The main driver of success in AI applications

- Instead of specifying a concrete solution to a problem, we specify how to *„transfer“* the information contained in data into a model.

➢ Only brief introduction to establish the fundamentals required for all other chapters.

➢ Three dedicated courses on Machine Learning throughout AAI-study program.

Content

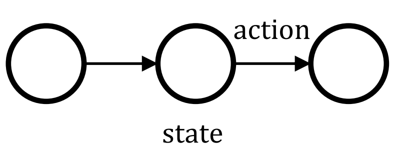
State-based Models

Given a description of the environment, find a solution path to a goal.

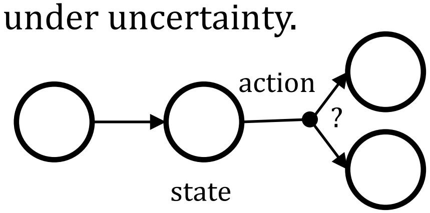
- Environment described in form states, actions, costs and rewards
- Output: Solution path

- Applications:
  - Games: Chess, Go, PacMan, etc.
  - Robotics: Motion planning
  - Natural language generation: Machine translation, image captioning

- Search problems: Full control over states and actions



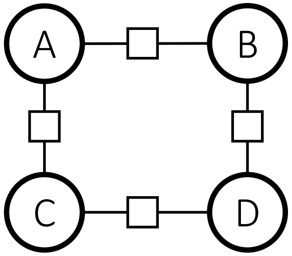
- Adversarial search problems: Playing against a rational opponent.
- Markov decision processes: Playing against nature.
  - Making decisions under uncertainty.



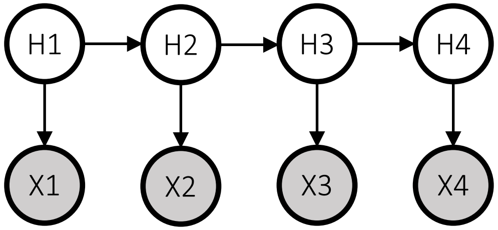
Content

Graphical Models

- Constraint satisfaction problems: Hard constraints on relations between variables.
  - Sudoku: Every row, column and cell must contain all digits 1-9.
  - Scheduling problems: Every person can be at exactly one place at any time.



- Bayesian Networks: Soft dependencies between variables.
  - Object tracking using sensors: Determining an object‘s location (P) from a noisy sensor reading (E) requires a model to permit some “slack“ in the relation between P and E.



- Python

> numpy, scipy

Content

Differentiable Programming

Deep Learning

- Modern models consist of computational building blocks (components)
  - Which computational components exist?
  - How are they assembled to form an architecture?
  - How can we turn an architecture into a model?
  - How can we fit such a model to data?

- Pre-defined architectures
  - Convolutional Networks
  - Recurrent Networks
  - Transformer

- Python
  - Automatic differentiation over computational graphs

  > `PyTorch`

Content

Representation Learning

How can we represent an object from the real-world as a numerical object?

- Features
- Feature engineering
- Automatic feature extraction

Popular algorithm for creating word representations from a text corpus:

- Word2Vec

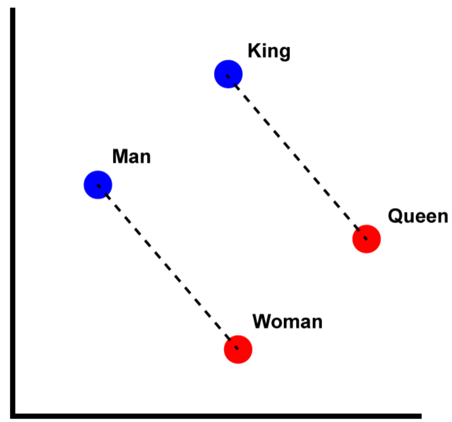


Image source

Python

```
> numpy, nltk
```

Content

Recommender Systems

Given a large collection of items and the preferences of a user, which item should we recommend to her?

- Passive „search“ based on preferences
- Examples of items: Music, Products, Documents, etc.

Common strategies:

- Collaborative filtering
- Content-based filtering

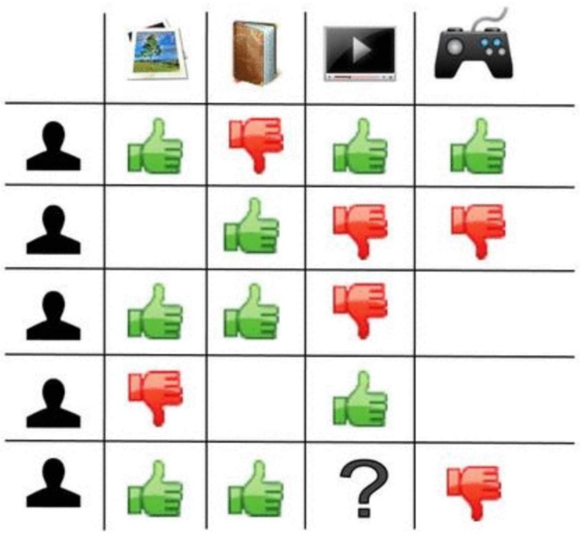


Image source

Content

Metaheuristics

General-purpose algorithms for finding an approximate solution to NP-hard optimization problems.

- Traveling salesperson problem:
  „Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city exactly once and returns to the origin city?"

Common strategies:

- Hill climbing
- Simulated annealing
- Evolutionary algorithms

Example: Nature-inspired algorithm *Ant Colony Optimization*

Python

`> numpy, numba`

Content

Localization & Mapping

Mobile robots, autonomous cars, unmanned aerial vehicles (UAV) need to

- Estimate a map of their environment
- Localize themselves inside the estimated map

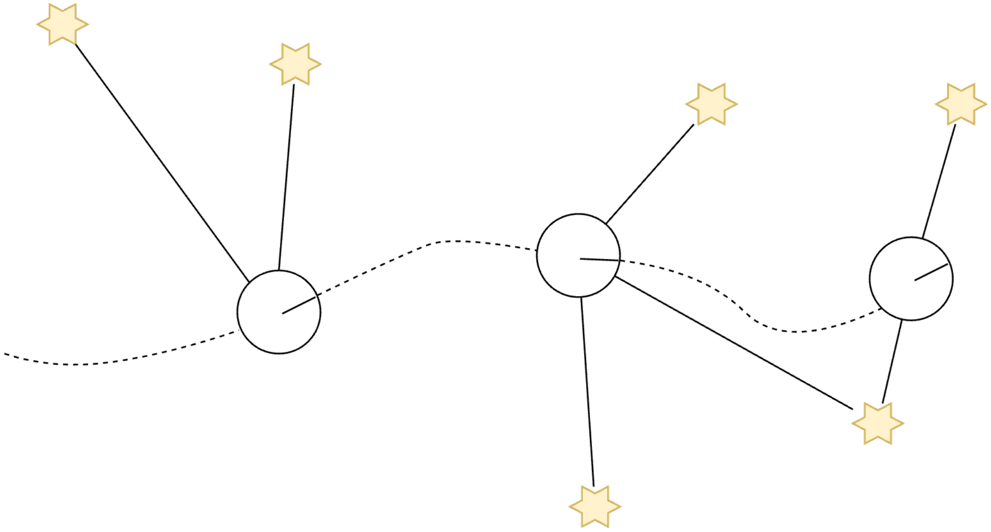
If a map were given → estimating location is easy

If pose were given → estimating a map is easy

Joint estimation is more difficult: Simultaneous localization & mapping (SLAM)

Core concepts:

- Localization & Mapping
- State estimation
- SLAM approaches:
  - Extended Kalman Filters
  - Particle Filters
  - Graph-based SLAM



Content

Network Science

Network Science uses networks to describe and analyze complex systems. A complex system exhibits emergent behaviour:

- Self—organized (no central organizing authority)
- evolving as a function of time
- adaptive to environment

Networks appear in various disciplines:

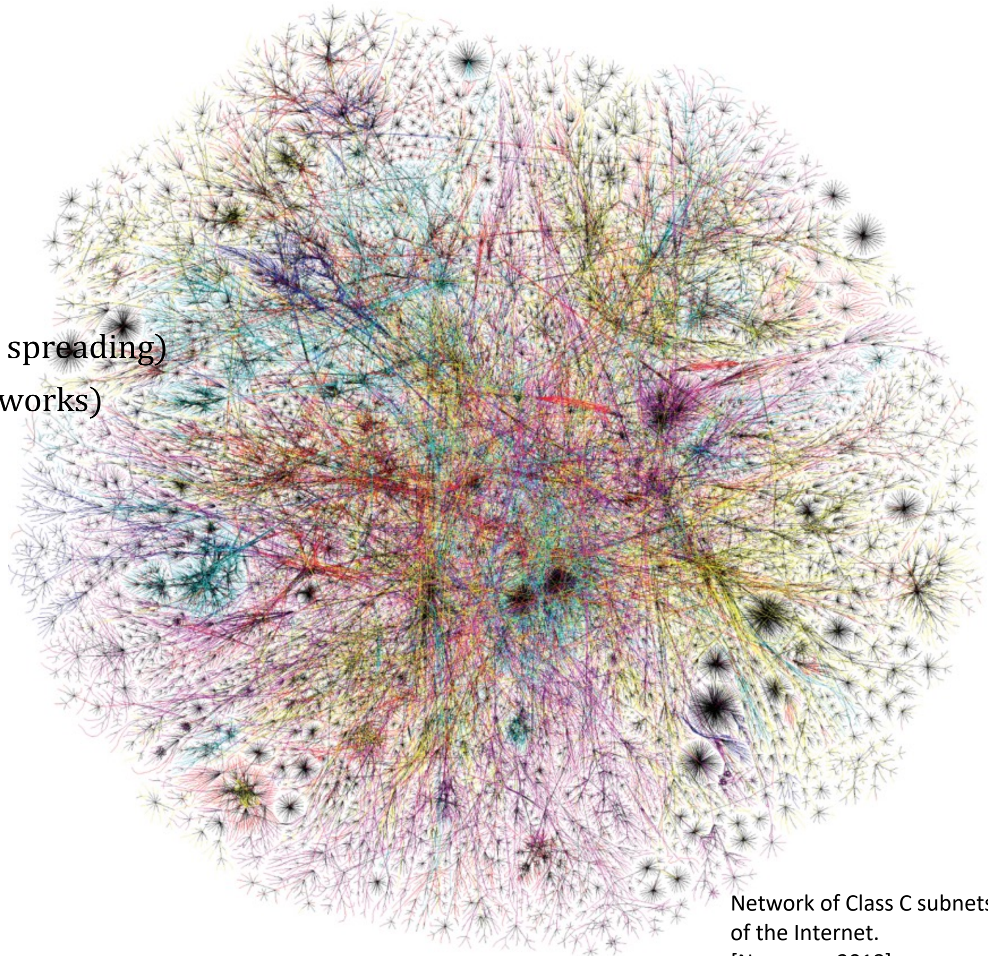
- Economics (e.g. world trade, stock market)
- Biology (e.g. human brain, populations & epidemic spreading)
- Social Sciences (e.g. social networks, affiliation networks)
- Technology (e.g. web, power grids, transportation)

Network characteristics:

- Constituents of networks
- Measures and Metrics
- Network models

Python

```
> networkx
```



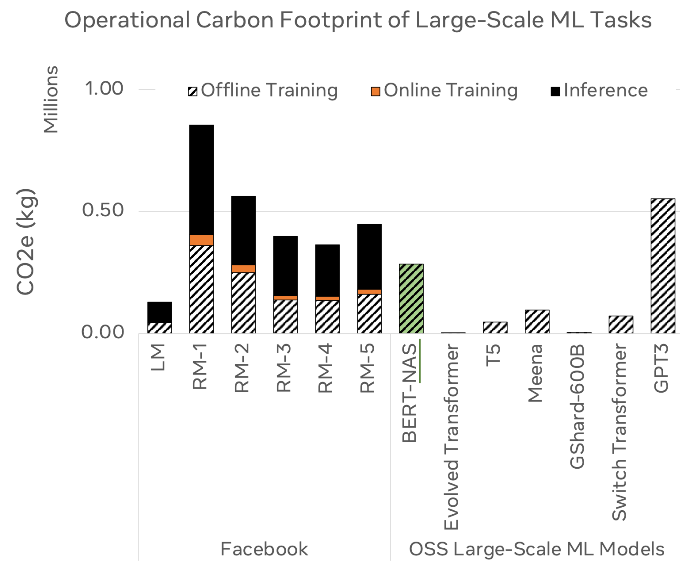
Network of Class C subnets of the Internet. [Newman, 2018]

Content

Risks & Challenges

- Energy consumption & Carbon footprint

Average per capita CO2 emission in Germany (2022): **10,8 t**
Umweltbundesamt



[Wu et al., 2022]

- Feedback loops
  - amplify bias
  - compromise fairness

- Privacy
  - Algorithms are developed under the assumption that data is fully accessible at one place.
  - In reality, data is distributed across multiple source without the intent to share publicly.
  - ➢ Privacy-preserving analysis (e.g. sharing of central statistics, homomorphic encryption)

- Security
  - Sensitive domains (e.g. face identification, autonomous driving) require models to be both accurate <u>and</u> robust against malicious behaviour
  - Guarding against adversarial attacks is an active research area.

Carefully arranged stickers make the system classify a STOP sign as a SPEED LIMIT sign.
[Eykholt et al., 2018]

References

- Wu, Carole-Jean and Raghavendra, Ramya and Gupta, Udit and Acun, Bilge and Ardalani, Newsha and Maeng, Kiwan and Chang, Gloria and Behram, Fiona Aga and Huang, James and Bai, Charles and Gschwind, Michael and Gupta, Anurag and Ott, Myle and Melnikov, Anastasia and Candido, Salvatore and Brooks, David and Chauhan, Geeta and Lee, Benjamin and Lee, Hsien-Hsin S. and Akyildiz, Bugra and Balandat, Maximilian and Spisak, Joe and Jain, Ravi and Rabbat, Mike and Hazelwood, Kim. *Sustainable AI: Environmental Implications, Challenges and Opportunities. arXiv. 2021.*
- Eykholt, Kevin and Evtimov, Ivan and Fernandes, Earlence and Li, Bo and Rahmati, Amir and Xiao, Chaowei and Prakash, Atul and Kohno, Tadayoshi and Song, Dawn. *Robust Physical-World Attacks on Deep Learning Models*. arXiv. 2017.
- Newman, Mark. *Networks*. 2nd Edition. Oxford University Press. 2018.