Jakovác Antal
Department of Computational Sciences

System 1 and System 2 in artificial intelligence

MT Kurbucz
P Pósfay
A Telcs
TS Biró

GPU days, Budapest, May 15-16 2023

Introduction

OpenAI ChatGPT4.0

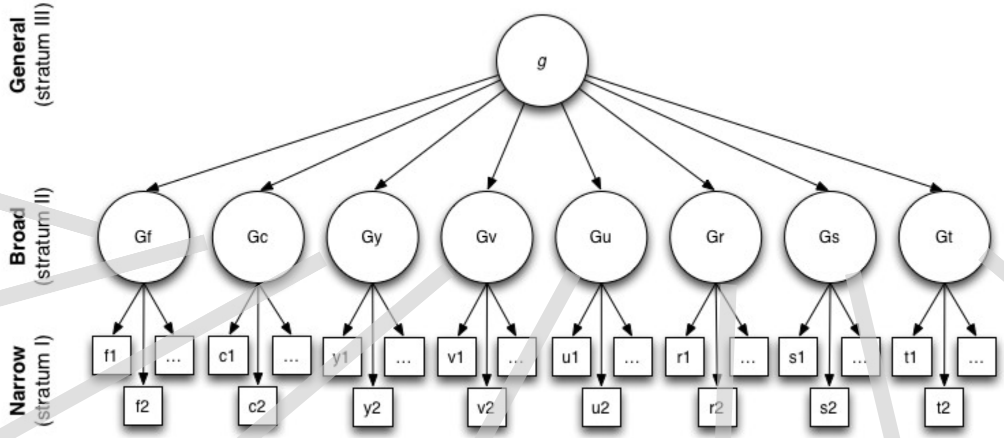
Recent impressive developments in AI

- Text generation: chatGPT, autoGPT, bing AI, bard AI, etc.
- Image generation: midjourney, thispersondeosnotexists, Dall-E, Dreamstudio, gencraft
- AI doomsday?
- Intelligent and useful tools (chatGPT: text generation, check, summary, programming)
- Heuristic, sometimes stupid, improvising, “lying”
- **Why do they work so well, and why do they fail so stupidly?**

Introduction

Human intelligence is not a monolithic entity:

Cattell–Horn–Carroll theory



Fluid intelligence: reasoning, analysis

Crystallized intelligence: use acquired knowledge

Memory and learning

Visual perception

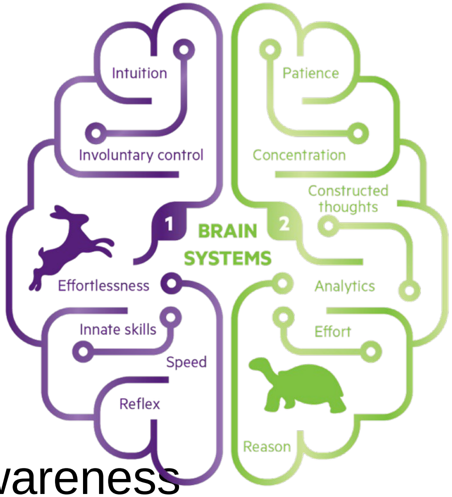
Auditory perception

Retrieval ability

Cognitive speediness

processing speed

Modes of human thinking



Categorization in cognitive psychology: *(Daniel Kahneman)*

- **System 1**:
  - fast, automatic, and intuitive, instinctive mode without conscious awareness
  - more prone to biases and errors due to its reliance on heuristics and automatic processing.
  - appropriate for fast response
- **System 2**:
  - slower, conscious, deliberate mode for evaluating information and decision making
  - controlled, more accurate, but can be slower and more effortful
  - appropriate for structured thinking

Modes of human thinking

In human thinking there are no extreme unbalances:

- *We use all parts of intelligence*
- *We use both System1 and System2*

unlike in AI models

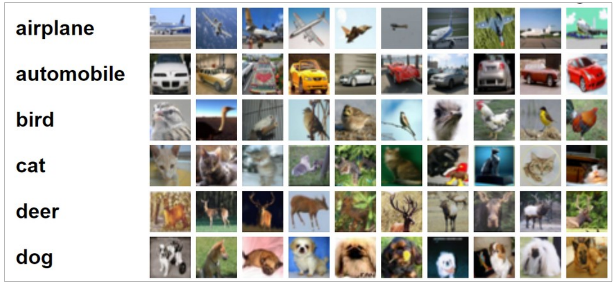
- We tend to think that all parts of IQ are present (cf. ELIZA, chatGPT → doomers)
- The performance of AI strongly **depends on the task** we want to solve with them
  - Turing's intelligence definition → deceive observers
  - Classification task → main stream applications, CNN, VAE, GAN → **System 1**
  - Data representation task → representation learning → **System 2**

Classification task

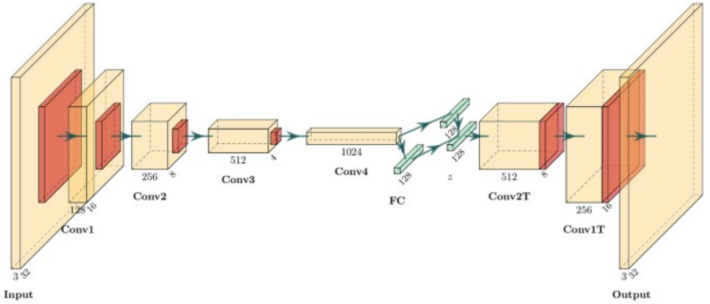
- **Task**: classification → performs classification tasks like humans
- Mathematical background: Bayesian analysis, we shall assess the probability of belonging to a given class
- Typically use loss functions, global parameter fitting (backpropagation)
- paradigms: supervised learning (annotation)

Classification task

- Most successful present-day AI primarily work in this way
  - Classifiers: MNIST, CIFAR, dog breeds, birdsong, faces, …
  - NLP models: classification of the next word/phrase
  - Recognition and generation

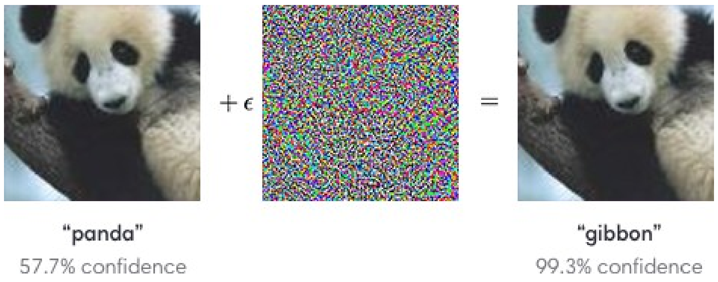


- Technology:
  - Deep Neural Networks (image recognition: CNN, VGG-16, AlexNet, ResNet, Inceptionv3, …)
  - Transformers (NLP, attention, GPT, BERT, XLNET, …)
  - GAN, VAE
  - …



Classification task

- **Advantages**:
  - Very fast, effective
  - Good interpolation properties
- **Disadvantages** (apart from technical ones)
  - Slow training: needs a lot of data and uses a large amount of parameters
  - No control over the mistakes (c.f. adversarial attacks)
  - Input → output is a continuous function, can not train with very unbalanced data (e.g. can not have a class “no cat images”)
  - Specific → *catastrophic forgetting*: classification outputs are interdependent
- **Corresponds to System1 thinking**



Relevance based intelligence

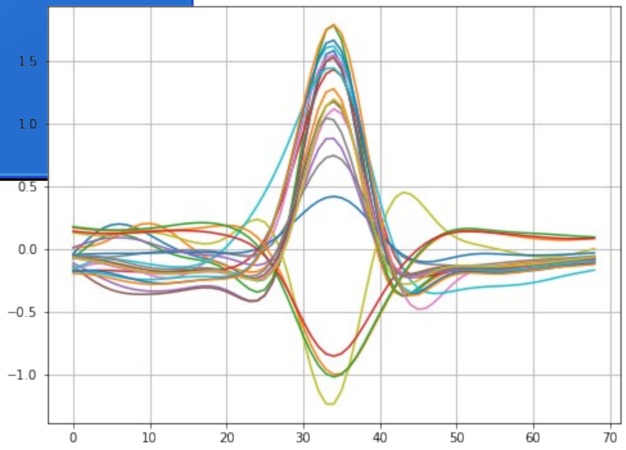
Task: data representation *(c.f. representation learning)*

- **Data driven**: we present those data that are assumed to have a common property
- **Context dependent**: the same data can be characterize differently
- **Method**: separate irrelevant and relevant features (*cognitive science: relevance realization*)
  - *irrelevant features* do not change the class
  - *relevant features* are constant above classes → **“laws”**
  - *Manifestation: law-based feature transformation* (LLT)
- Effectiveness of realization → entropy
  (TS Biró, AJ, Universe 8 (1), 53; AJ, A Telcs, Entropy 24 (9), 1313)

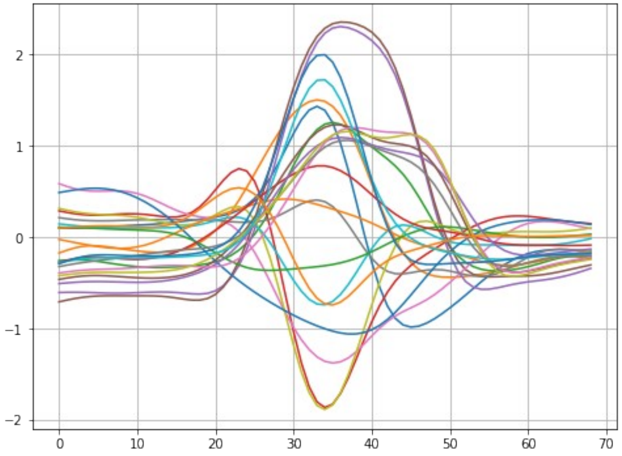
Relevance based intelligence

- **Advantages**:
  - Control over mistakes: several laws with AND relation
  - Can be used with unbalanced data: intersection of sets belonging to different laws
  - No forgetting: laws separate class elements from not class elements, no interdependence between laws
  - Fast training: needs fewer data and less parameters than training
- **Disadvantages** (apart from technical ones)
  - Can be slow for a lot of laws (parallelization necessary)
  - Scalability? → needs further studies

Application: ECG analysis



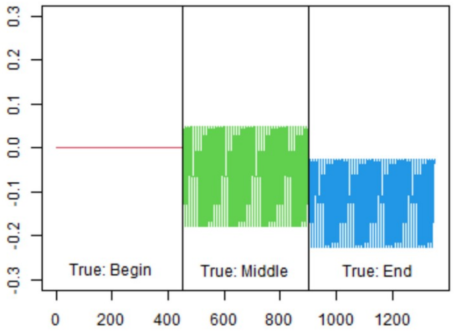
- Goal: classify heart beats into normal and ectopic
- ECG signal: cleaning, standardizing
- Mehtod: prepare test, validation and training sets
  - Find linear laws for the QRS complex (11 leg embedding, universal laws)
  - Train a classifier on the results (KNN, RF, SVM)
  - Results depend on several factors, best result SVM: 94.3% (close to state-of-art results)
  - More data could help to improve accuracy
- Can be used in a non-annotated dataset (self annotation)



Application: AReM database

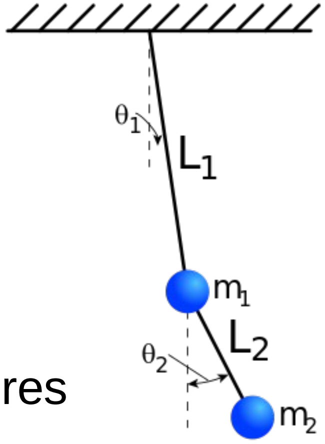
MT Kurbucz, P Pósfay, AJ, Scientific Reports 12 (1), 18026

- Activity Recognition system based on Multisensor data fusion (AReM) Data Set
  - 7 motion classes (bending, lying, cycling,etc.)
  - 3 sensor data → 6 features (mean and variance)
  - 88 time series (instances), 480 values in each
- Mehtod: LLT (Linear Law based feature Transformation)
  - Determine the laws for each instances and channels in the training sets
  - Apply them to the test series, take temporal average/variance → features
  - Train a classifier on the results (KNN, DT, SVM)
  - KNN provides error-free classification

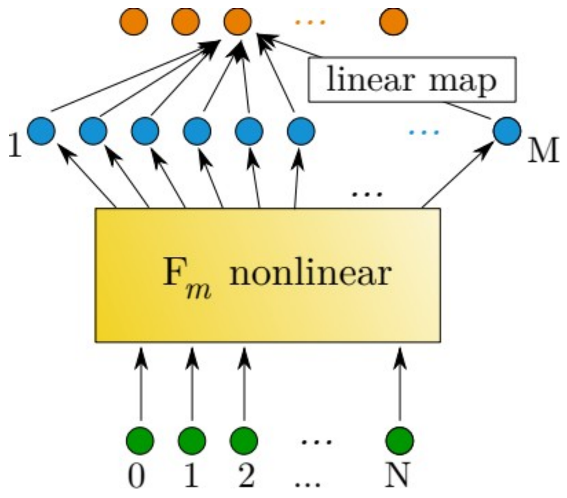


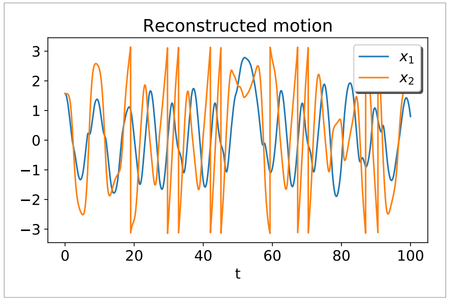
Nonlinear laws

AJ, MT Kurbucz, P Pósfay, New Journal of Physics 24 (7), 073021



- Generalization: input are not directly the embedded data, but pre-trained features
- F_m can be represented by (deep) neural network
- Extreme learning: the exact form of F_m does not matter
- Reconstruction of mechanical motions: 3-leg embedding (discrete Newton-equations)
- Chaoticity, stability → recursion to reconstruct motion

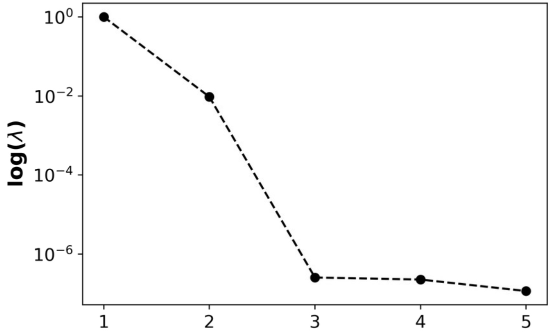


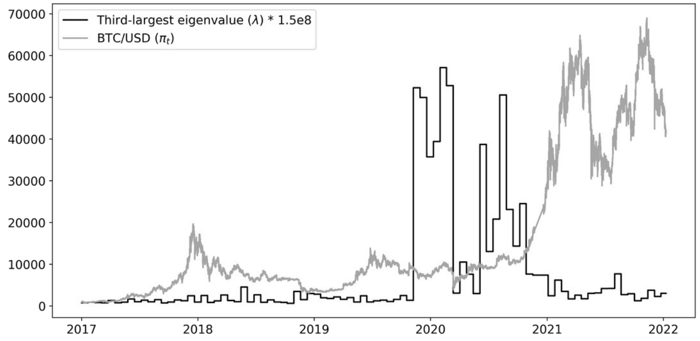


Stochastic processes

MT Kurbucz, P Pósfay, A Jakovác, arXiv preprint arXiv:2201.09790

- Markov chains: stochastic process where $P^{(n+1)}(x)=\sum_y T^{(n)}_{x,y} P^{(n)}(y) \Rightarrow P^{(n+1)}=T\,P^{(n)}$
- In equilibrium (steady state) no *n* dependence, for equilibrium distribution: $P=T\,P$
- 2-variable correlation functions: $\langle f(x_n,x_{n+k})\rangle=\mathrm{Tr}(F\,T^k)$ where $F_{xy}=f(x,y)P(x)$
- These satisfy linear laws: $\sum_k \langle f(x_n,x_{n+k})\rangle w_k=0$ if $\sum_k w_k T^k=0$ characteristic polynomial
- Dimensionality of the Markov process can be determined from the laws





Conclusions

The question/task we want to solve determines the possible answers

- **Turing's intelligence definition**: programs deceiving humans
- **Classification task**
  - Probabilistic systems, specific tasks
  - Method of development: training
  - Slow training, fast operation → System 1
- **Representation task**
  - Structured systems, generic tasks, context
  - Method of development: finding relevant features, laws
  - Fast learning, slower operation → System 2

The end