Artificial Intelligence
Advanced Topics in AI & ML

Introduction. Course logistics and syllabus. Deep Learning and Neural Nets

Aleksandr Petiushko

ML Research

SOFIA UNIVERSITY

AP

Content

Content

Content

Content

Content

Intro

About the lecturer[1]

- Aleksandr Petiushko, PhD in theoretical CS (2016)
- Lecturer in Lomonosov MSU / MIPT for Machine Learning, Computer Vision, Deep Learning Theory, Python for an ML Researcher since 2019
- Former Huawei Chief Scientist (Scientific Expert), AIRI Director of Key Research Programs (Leading Scientific Researcher), Nuro Head of ML Research
- Currently at Gatik, leading the AI Research

[1] Homepage: `https://petiushko.info/`

Intro

Time to introduce yourselves: what are your hobbies, motivation in ML, etc.: please go into “**Module 1 Students Introduction**” thread

Sofia Plagiarism Policy

- It covers parts "*sourced from AI*"
  - Please read the "**Sofia Plagiarism Policy**" thread
  - **First offense**: students need to rewrite assignment
  - **Second offense**: students fail the course
  - **Third offense**: students re to be withdrawn from their program

Note about xGPTx

- It can produce very plausible answers in 90% of cases

Note about xGPTx

- It can produce very plausible answers in 90% of cases
- The caveats are the following:

Note about xGPTx

- It can produce very plausible answers in 90% of cases
- The caveats are the following:
  - It can really hallucinate some things which are just untrue

Note about xGPTx

- It can produce very plausible answers in 90% of cases
- The caveats are the following:
  - It can really hallucinate some things which are just untrue
  - It can produce very different information in comparison to the source used to ask question (e.g., book chapter)

Note about discussions

- Discussion answers like “I agree because of bla-bla-bla” won’t be graded — they do not provide any value

Note about discussions

- Discussion answers like "I agree because of bla-bla-bla" won't be graded — they do not provide any value
- Only the answers with some non-trivial arguments that:
  - either contradict the initial post,
  - or add some non-obvious missing things to the initial message

  will be considered as graded ones

Course logistics

- Course grading will be done based on attendance, assignments, discussions, (optional: mini research problem) and the final exam.

Course logistics

- Course grading will be done based on attendance, assignments, discussions, (optional: mini research problem) and the final exam.
- Contribution:
  - 50%: attendance, assignments, discussions
  - 50%: final exam

Course logistics

- Course grading will be done based on attendance, assignments, discussions, (optional: mini research problem) and the final exam.
- Contribution:
  - 50%: attendance, assignments, discussions
  - 50%: final exam
- Preliminary grading scale:

Grade	Percent accumulated
A	90-100 %
B	75-89 %
C	60-74 %

Note about optional MRP

- MRP = **M**ini **R**esearch **P**roblem

Note about optional MRP

- MRP = **M**ini **R**esearch **P**roblem
- Topics: Please refer to **Module 0**

Note about optional MRP

- MRP = **M**ini **R**esearch **P**roblem
- Topics: Please refer to **Module 0**
- MRP to be done in form of a pdf presentation that should be shared via student's github

Note about optional MRP

- MRP = **M**ini **R**esearch **P**roblem
- Topics: Please refer to **Module 0**
- MRP to be done in form of a pdf presentation that should be shared via student's github
- It should contain:
  - tl;dr: motivation, main idea, and results briefly (1 slide)

Note about optional MRP

- MRP = **M**ini **R**esearch **P**roblem
- Topics: Please refer to **Module 0**
- MRP to be done in form of a pdf presentation that should be shared via student's github
- It should contain:
  - tl;dr: motivation, main idea, and results briefly (1 slide)
  - Literature review on the related methods

Note about optional MRP

- MRP = **M**ini **R**esearch **P**roblem
- Topics: Please refer to **Module 0**
- MRP to be done in form of a pdf presentation that should be shared via student's github
- It should contain:
  - tl;dr: motivation, main idea, and results briefly (1 slide)
  - Literature review on the related methods
  - Approach used (with all relevant links used)

Note about optional MRP

- MRP = **M**ini **R**esearch **P**roblem
- Topics: Please refer to **Module 0**
- MRP to be done in form of a pdf presentation that should be shared via student's github
- It should contain:
  - tl;dr: motivation, main idea, and results briefly (1 slide)
  - Literature review on the related methods
  - Approach used (with all relevant links used)
  - Youtube link to the pre-recorded demonstration of the built system

Note about optional MRP

- MRP = **M**ini **R**esearch **P**roblem
- Topics: Please refer to **Module 0**
- MRP to be done in form of a pdf presentation that should be shared via student's github
- It should contain:
  - tl;dr: motivation, main idea, and results briefly (1 slide)
  - Literature review on the related methods
  - Approach used (with all relevant links used)
  - Youtube link to the pre-recorded demonstration of the built system
- Presentations to be presented by the author during the last onground session
  - Duration: apprx 10 min

Late Submission Policy

Late submission deduction percent: **15% every day**;

Late Submission Policy

Late submission deduction percent: **15% every day**;

- It means that if you're **7 days late** than no need to submit: you'll get **0 score** anyway.

Makeup Policy

Missing onground class:

- Student's grade is dropped by 10%;

Makeup Policy

Missing onground class:

- Student's grade is dropped by 10%;

Missing Exam:

- A student gets 0 for the Exam;

Makeup Policy

Missing onground class:

- Student's grade is dropped by 10%;

Missing Exam:

- A student gets 0 for the Exam;

Missing Assignment in time (1 week, please refer to the *Late Submission Policy*):

- No makeups, i.e. 0 for the Assignment;

Makeup Policy

Missing onground class:

- Student's grade is dropped by 10%;

Missing Exam:

- A student gets 0 for the Exam;

Missing Assignment in time (1 week, please refer to the *Late Submission Policy*):

- No makeups, i.e. 0 for the Assignment;

Unless:

- A Student has a serious medical condition, and this condition is validated by a hospital or licensed California physician (in English)
- The Student contacts in time our chair (Donna Dulo) and describes the situation and provides all the needed proofs
- The Student notifies in time Professor about the situation with the confirmation from our chair (Donna Dulo)

Github

- Course page: `https://github.com/fatheral/sofia-aiml-2024`
- The professor's lectures will be uploaded there

What is Artificial Intelligence?

Natural Intelligence (human)

- Able to perceive the information, analyze it, make decisions based on this analysis

What is Artificial Intelligence?

Natural Intelligence (human)

- Able to perceive the information, analyze it, make decisions based on this analysis

Artificial Intelligence

- (Strong) The same as natural intelligence, but computer is instead of human

What is Artificial Intelligence?

Natural Intelligence (human)

- Able to perceive the information, analyze it, make decisions based on this analysis

Artificial Intelligence

- (Strong) The same as natural intelligence, but computer is instead of human
- (**Weak**) Algorithm which is able to train using the input data in order to do tasks afterward — instead of human

What is Machine Learning

In 1959 Arthur Samuel introduced the term “machine learning” into scientific use.

General definition

Machine Learning — the process leading computers to gain ability to show the behavior that wasn't explicitly programmed.

What is Machine Learning

In 1959 Arthur Samuel introduced the term “machine learning” into scientific use.

General definition

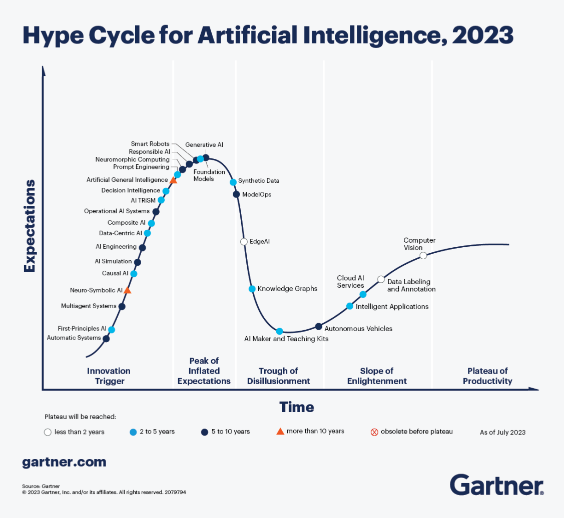
Machine Learning — the process leading computers to gain ability to show the behavior that wasn’t explicitly programmed.

In 1997 Tom M. Mitchell introduced more formal definition of a machine learning algorithm.

Formal definition

A **computer program** is said **to learn** from examples E for some set of problems T and a quality metric P if its performance on problems from T, as measured by P, is improved by using examples E.

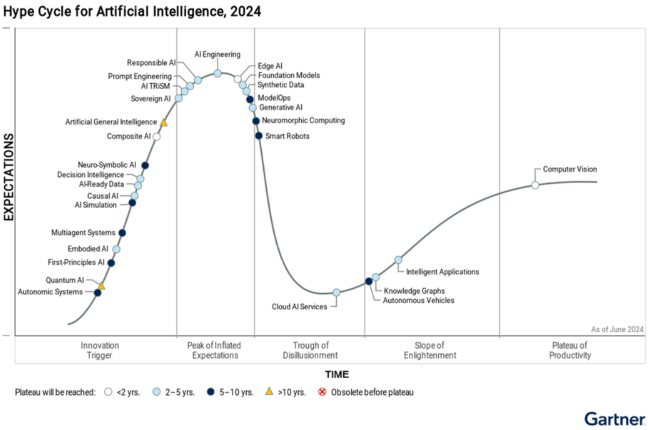
Broad concepts: AI Hype Cycle 2023[2]



[2] www.gartner.com

Broad concepts: AI Hype Cycle 2024[3]

Figure 1: Hype Cycle for Artificial Intelligence, 2024



[3] www.jaggaer.com

Course content

- Deep Learning and Neural Nets
- Generative AI: Generative Adversarial Networks
- Generative AI: Diffusion
- Transformers: encoders and decoders
- LLMs: BERT, GPT
- Applications: Computer Vision
- Applications: Speech Recognition
- Multi-tasking
- Multi-modality
- Interpretability and Explainability
- Embodied AI: Self-Driving
- AI Ethics
- Robust ML

Deep Learning[4] and Neural Nets

- Neural Net (NN): a (usually!) non-linear function mapping a (usually) multi-dimensional input to some output (which can be of the same dimension, or a bigger/smaller one)

[4] Deep Learning Classical Book

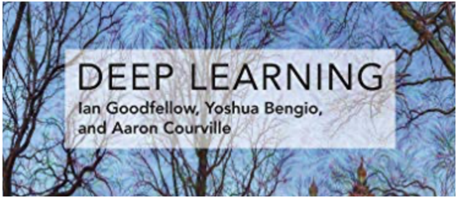
Deep Learning[4] and Neural Nets

- Neural Net (NN): a (usually!) non-linear function mapping a (usually) multi-dimensional input to some output (which can be of the same dimension, or a bigger/smaller one)
- Most common NN atomic operations: addition, multiplication, scalar non-linearity, aggregation/normalization

[4] Deep Learning Classical Book

Deep Learning[4] and Neural Nets

- Neural Net (NN): a (usually!) non-linear function mapping a (usually) multi-dimensional input to some output (which can be of the same dimension, or a bigger/smaller one)
- Most common NN atomic operations: addition, multiplication, scalar non-linearity, aggregation/normalization
- Deep Learning: a NN consisting of more 2 layers of atomic operations (that's why deep) and the corresponding procedure of the training ("learning") it weights using back propagation process



[4] Deep Learning Classical Book

Feature engineering

- Initially the Machine Learning machinery was targeted at **feature engineering** (or extraction, or discovery)

Feature engineering

- Initially the Machine Learning machinery was targeted at **feature engineering** (or extraction, or discovery)
  - Mapping f from arbitrary object $o \in O$ to its unified feature space representation $x \in X \subseteq \mathbb{R}^D$: $f : O \to X$

Feature engineering

- Initially the Machine Learning machinery was targeted at **feature engineering** (or extraction, or discovery)
  - Mapping f from arbitrary object $o \in O$ to its unified feature space representation $x \in X \subseteq \mathbb{R}^D$: $f : O \to X$
- Suppose we have m objects $\Rightarrow$ can construct training dataset X^m

Feature engineering

- Initially the Machine Learning machinery was targeted at **feature engineering** (or extraction, or discovery)
  - Mapping f from arbitrary object $o \in O$ to its unified feature space representation $x \in X \subseteq \mathbb{R}^D$: $f : O \to X$
- Suppose we have m objects $\Rightarrow$ can construct training dataset X^m
- On top of that representation (usually a *linear*) a classifier is learned (e.g., SVM): $a(w, X^m) = \operatorname{sign} g(x, w) = \operatorname{sign}\langle x, w\rangle$

Representation Learning

- Main goal: to have objects representations that are rich and descriptive:
  - Representations of objects from the same class are better *to be close* (according to some metric, e.g., Euclidean) in the feature space

Representation Learning

- Main goal: to have objects representations that are rich and descriptive:
  - Representations of objects from the same class are better *to be close* (according to some metric, e.g., Euclidean) in the feature space
  - Representation of objects from different classes are better to *lie far* from each other (according to some metric)

Representation Learning

- Main goal: to have objects representations that are rich and descriptive:
  - Representations of objects from the same class are better *to be close* (according to some metric, e.g., Euclidean) in the feature space
  - Representation of objects from different classes are better to *lie far* from each other (according to some metric)
- Sometimes the initial features $x \in X$ are not suited enough for this task

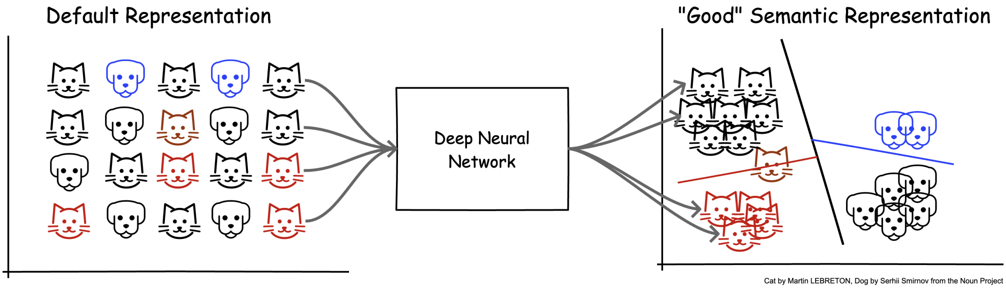
Representation Learning

- Main goal: to have objects representations that are rich and descriptive:
    - Representations of objects from the same class are better *to be close* (according to some metric, e.g., Euclidean) in the feature space
    - Representation of objects from different classes are better to *lie far* from each other (according to some metric)
- Sometimes the initial features $x \in X$ are not suited enough for this task
- For this purpose, additional mapping h from X into some additional representation space $R \subseteq \mathbb{R}^N$: $h : X \to R$

Representation Learning

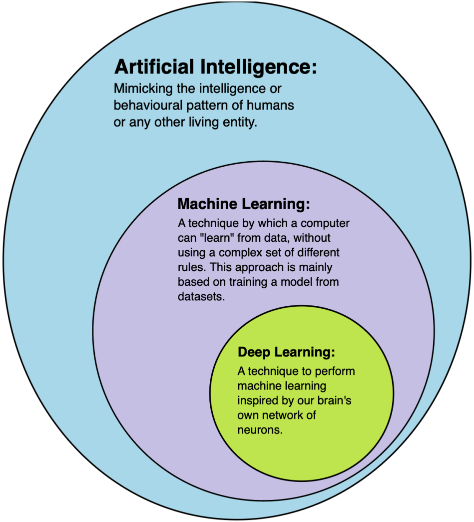
- Main goal: to have objects representations that are rich and descriptive:
  - Representations of objects from the same class are better *to be close* (according to some metric, e.g., Euclidean) in the feature space
  - Representation of objects from different classes are better to *lie far* from each other (according to some metric)
- Sometimes the initial features $x \in X$ are not suited enough for this task
- For this purpose, additional mapping h from X into some additional representation space $R \subseteq \mathbb{R}^N$: $h : X \to R$
- The learning of this additional mapping h is called **Representation Learning**

Representation spaces illustration[5]



[5] Image credit: blog.fastforwardlabs.com

AI vs ML vs DL[6]



[6] Wiki

Deep Learning History I[7]

1943	McCulloch & Pitts	introduced MCP Model, which is considered as the ancestor of Artificial Neural Model.
1949	Donald Hebb	considered as the father of neural networks, introduced Hebbian Learning Rule, which lays the foundation of modern neural network.
1958	Frank Rosenblatt	introduced the first perceptron, which highly resembles modern perceptron.
1974	Paul Werbos	introduced Backpropagation
1980	Teuvo Kohonen	introduced Self Organizing Map
	Kunihiko Fukushima	introduced Neocogitron, which inspired Convolutional Neural Network
1982	John Hopfield	introduced Hopfield Network
1985	Hilton & Sejnowski	introduced Boltzmann Machine
1986	Paul Smolensky	introduced Harmonium, which is later known as Restricted Boltzmann Machine
	Michael I. Jordan	defined and introduced Recurrent Neural Network

[7] On the Origin of Deep Learning

Deep Learning History II[8]

1990	Yann LeCun	introduced LeNet, showed the possibility of deep neural networks in practice
1997	Schuster & Paliwal	introduced Bidirectional Recurrent Neural Network
	Hochreiter & Schmidhuber	introduced LSTM, solved the problem of vanishing gradient in recurrent neural networks
2006	Geoffrey Hinton	introduced Deep Belief Networks, also introduced layer-wise pretraining technique, opened current deep learning era.
2009	Salakhutdinov & Hinton	introduced Deep Boltzmann Machines

- 2011: AlexNet — the first neural net winning the ImageNet challenge
- 2017: Invention of Transformer, the main architecture of LLM
- 2022: Invention of ChatGPT

[8] On the Origin of Deep Learning

More details on DL History and NN Architectures

Please read two links below:

- Deep Learning in a Nutshell: Core Concepts
- Deep Learning in a Nutshell: History and Training

Takeaway notes

1. Please go through all the materials of **Module 0**

Takeaway notes

1. Please go through all the materials of **Module 0**
2. Please introduce yourself and complete the **Assignment 1**

Takeaway notes

1. Please go through all the materials of **Module 0**
2. Please introduce yourself and complete the **Assignment 1**
3. Deep Learning is responsible for the most of the AI success today!

Takeaway notes

1. Please go through all the materials of **Module 0**
2. Please introduce yourself and complete the **Assignment 1**
3. Deep Learning is responsible for the most of the AI success today!
4. Let's get our journey started!

Thank you!