



# Mesterséges Intelligencia és gépi tanulás

## Mi az a deep learning?



Kertész Gábor

[kertesz.gabor@sztaki.hu](mailto:kertesz.gabor@sztaki.hu)

# Strong AI vs Weak AI

- ▶ Fontos látni, hogy a mai állapotok szerint általános, tudattal rendelkező, emberi intellektust imitáló gépi intelligencia nem létezik
- ▶ A korunk kiemelkedő eredményei e területen egy-egy jól definiált, konkrét probléma megoldására készített gépi tanulás alapú eszközt jelentenek
- ▶ Azaz, a gyakorlatban a mesterséges intelligencia - a science-fiction irodalomhoz mérten - még gyerekcipőben jár

# Jelen



Source Sequence

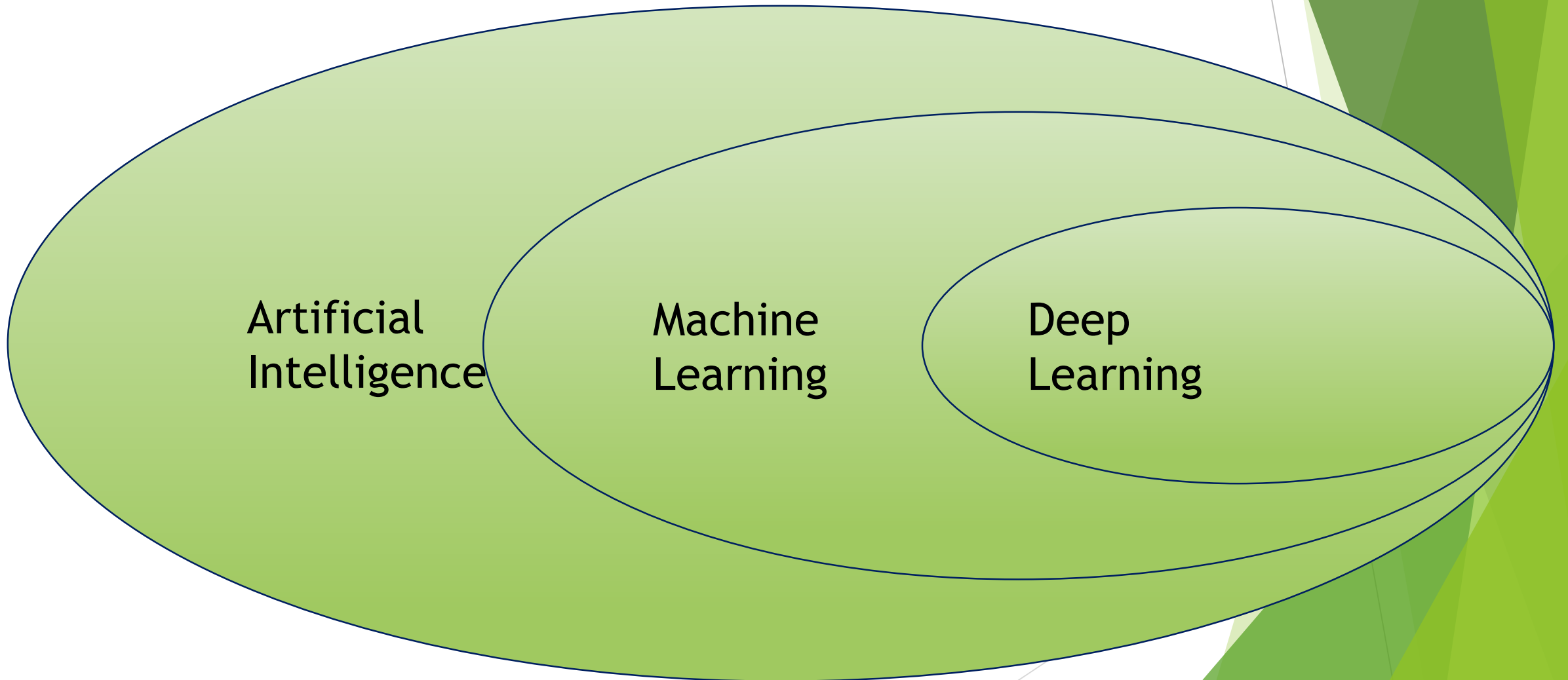
Our Reenactment  
(Full Head)

Averbuch-Elor et al. 2017



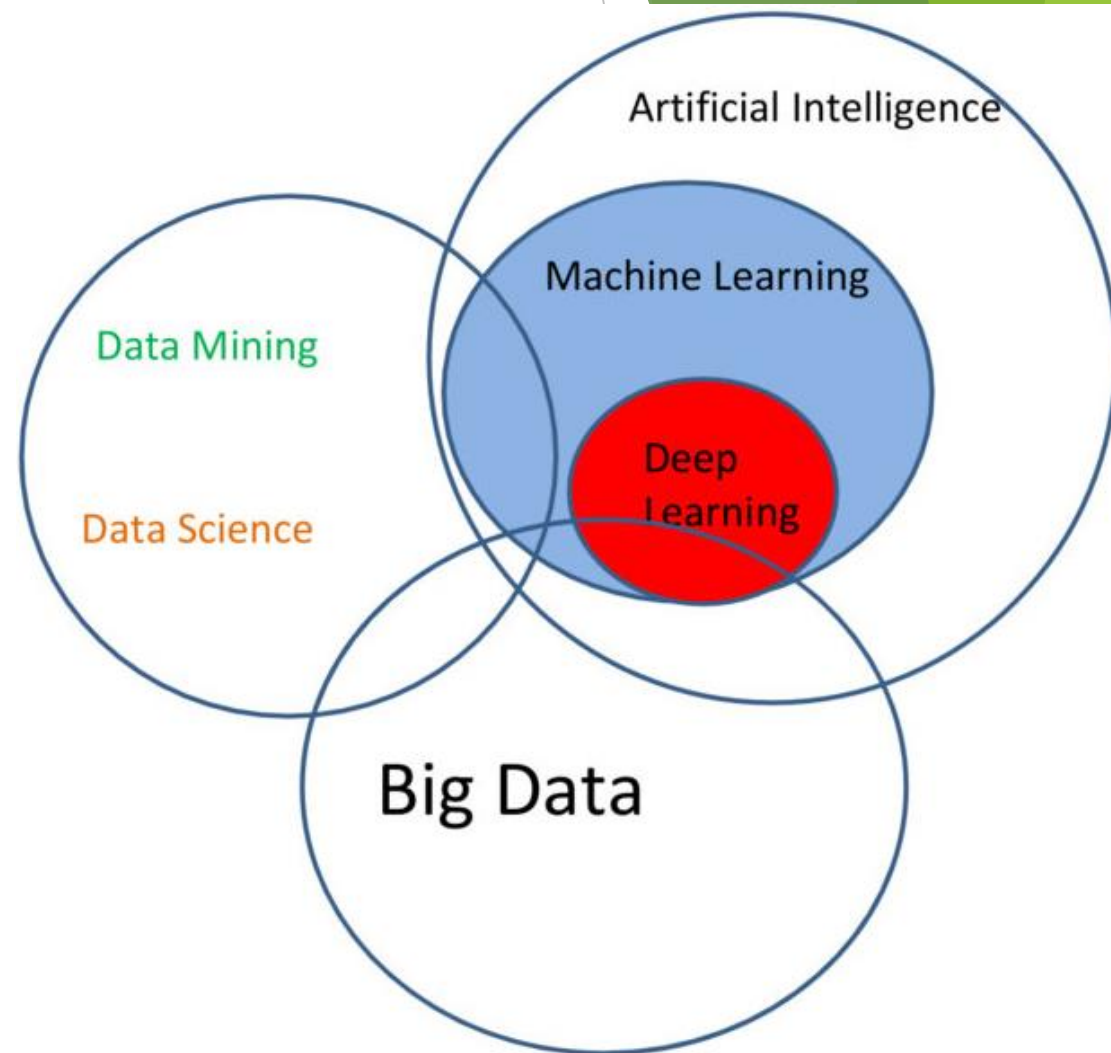
# Terminológia

Mély neurális hálózatokon alapuló gépi tanulási algoritmusok  
A magasabb rétegek egyszerű, a mélyebb rétegek összetett jellemzők detektálására alkalmasak



# Adatok

- ▶ Data Science, Big Data
  - ▶ Nagy mennyiségű adat tárolása, kezelése, feldolgozása
- ▶ Nagy számításigényű feladat
  - ▶ Felhők, szuperszámítógépek szükségesek



# Gépi tanulás

Jellemzően két nagy csoportra szokás bontani:

- ▶ Felügyelt tanulás
- ▶ Felügyelet nélküli tanulás

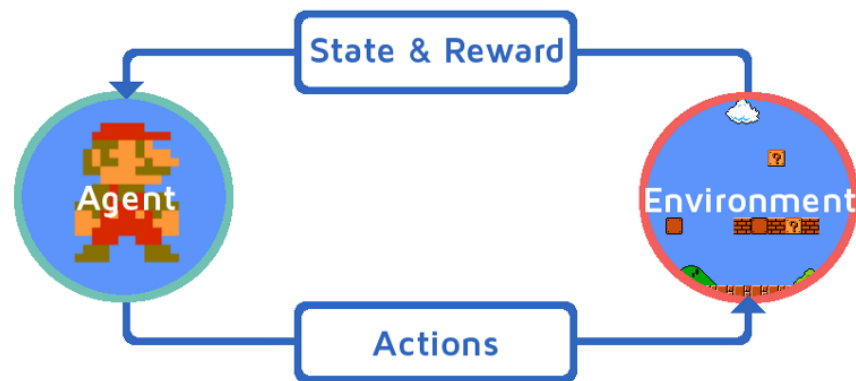
Felügyelt tanulás esetén rendelkezésre állnak olyan tanítóminták, ahol **ismert az elvárt kimenet** is, így tanításkor mérhető a pontosság.

Felügyelet nélküli tanítás esetén **címkezetlen adatok** közötti összefüggések felismerése a cél, amely jellemzően csoportosítást, klaszterezést jelent.

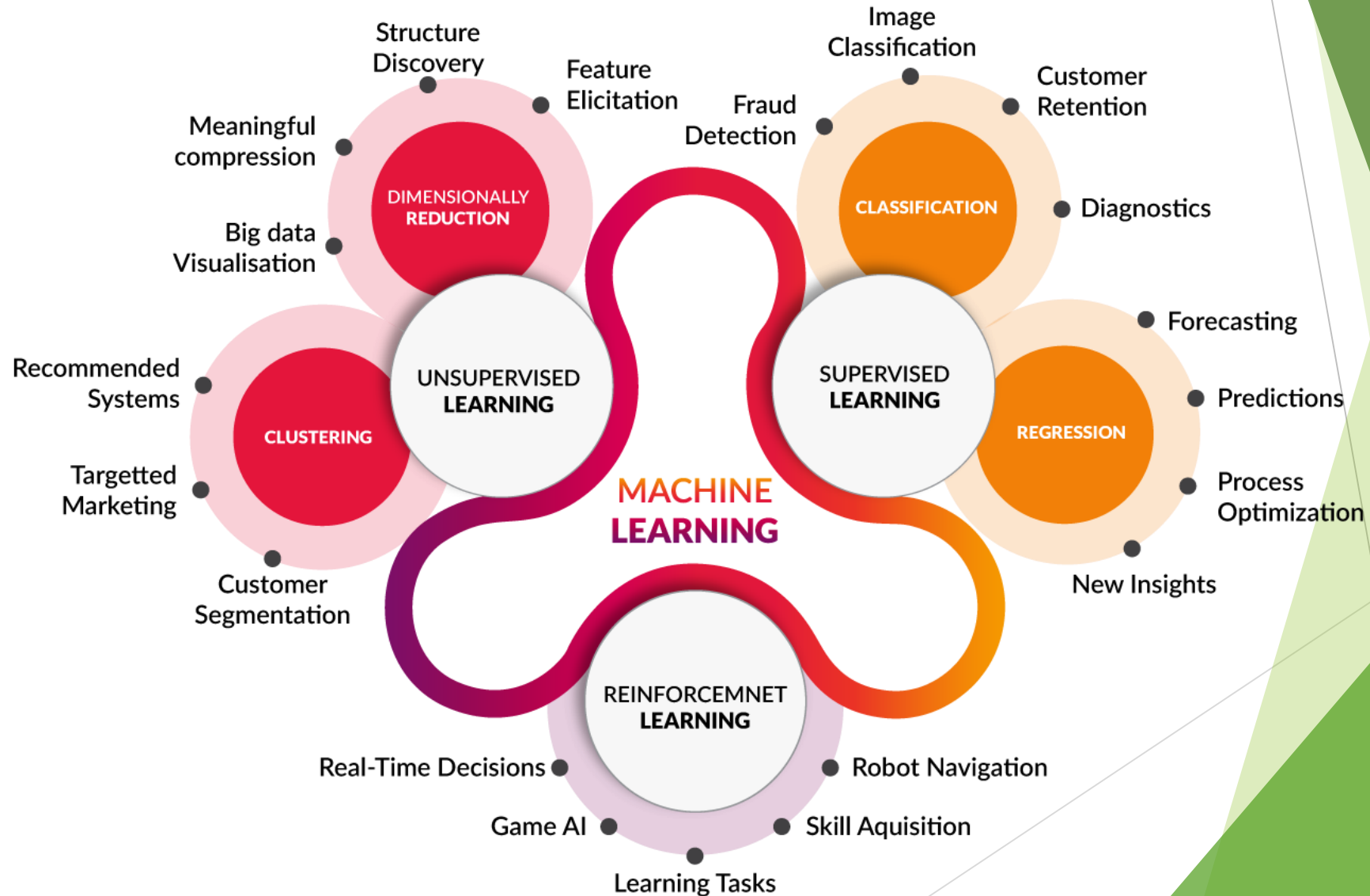
# Megerősítéssel tanulás

A gépi tanulás egy különleges kategóriája a megerősítéssel tanulás: ebben az esetben a feladat egy ügynök **dinamikus környezetben** való döntéshozatalának optimalizálása a környezettől kapott jutalmak maximalizálására.

Tipikus példák ilyen esetekre a különböző játékokat automatikusan játszó mesterséges ügynökök, vagy épp a Go-világbajnokot legyőző számítógépes program.



# Gépi tanulás





# A mesterséges neuron

A mesterséges neuronok a biológiai neuron által ihletett egyszerű, bemenettel és kimenettel rendelkező feldolgozó egységek

A neuron az egyes bemenetekhez súlyértékeket tárol, amelyekből súlyozott összeget kalkulál:

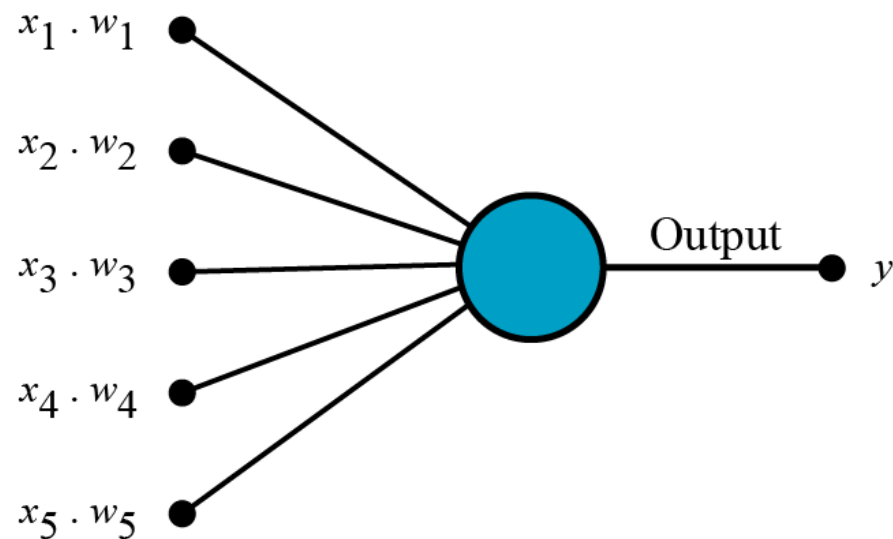
$$y = \sum_{i=1}^N w_i x_i$$

Ahol  $y$  a kimenet,  $x_i$  az  $N$  darab bemeneti paraméter  $i$ -edik eleme,  $w_i$  pedig az ehhez tartozó súlyérték.

# A mesterséges neuron

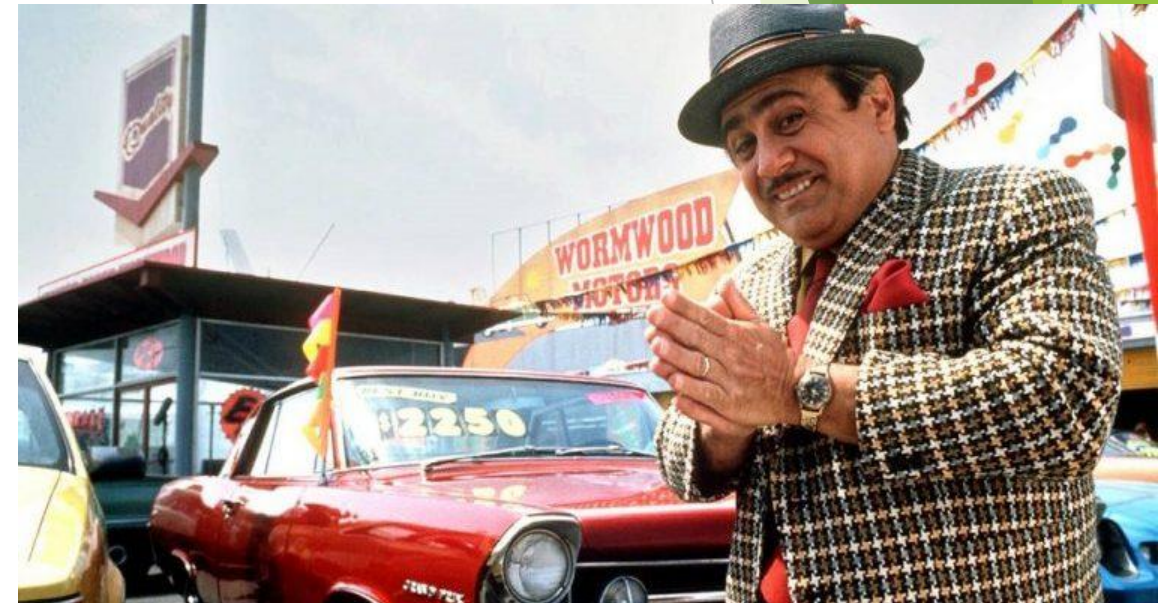
$$y = \sum_{i=1}^N w_i x_i$$

A neuron tanítása a  $w_i$  paraméterek olyasfajta módosítását jelenti, hogy a kimeneti  $y$  érték hibája minimális legyen.



# Egy gyors példa - az autókereskedő

- ▶ Egy autókereskedő egy használt autó árát annak különböző paraméterei alapján képes megbecsülni, mint például:
  - ▶ Típus
  - ▶ Szín
  - ▶ Évjárat
  - ▶ Hengerűrtartalom
  - ▶ Futott kilométerek száma
  - ▶ Ajtók száma
  - ▶ Állapot
  - ▶ stb



# Egy gyors példa - az autók kereskedő

- ▶ A kereskedő a fejében valójában egy modellt épít fel: a különböző paraméterek különböző súlyokkal vannak számításba véve, ezek szorzatösszege alapján kerül megbecslésre az ár

Hogyan tanulta meg ezt a kereskedő?

- ▶ Több példát látott, amelyeknél ismert a jármű ára is: ezeket címkézett tanítómintáknak nevezzük
- ▶ A tanítóminták alapján felmérte a különböző paraméterek fontosságát, és a **kapcsolódó súlyértékeket** ennek megfelelően **változtatta**

# Egy gyors példa - az autókereskedő

- ▶ Hogyan lehet a súlyokat finomhangolni?
- ▶ A tanítóminták (összesen  $M$  darab) és az aktuális modell alapján kiszámolható a hiba:

$$\text{Hiba} = \sum_{j=1}^M |\text{Várt érték}_j - \text{Becsült érték}_j|$$

- ▶ Alkalmazható az itt látható abszolút hiba (MAE - mean average error) helyett a négyzetes hiba is (MSE - mean squared error)
- ▶ A cél ennek a **hibának a minimalizálása** a súlyértékek módosításán keresztül. Ez persze megoldható random korrekciókkal is, de léteznek ennél szofisztikáltabb optimalizációs eljárások is

# Egy gyors példa - az autókereskedő

Ebben az esetben egyetlen neuron segítségével működtethető a modell

Mi a gond ezzel a megközelítéssel?

- ▶ A modell bár egyszerű összefüggéseket felfedez (pl. kevés futott kilométerrel a piros színű 2 hónapos Ferrari ára magas), de a komplexebb kapcsolatok felfedezésére alkalmatlan
  - ▶ Példa: XOR kapcsolat a bemeneti paraméterek között

# Felügyelt gépi tanulás

Felügyelt gépi tanulás esetén a megoldandó feladatok a kimeneti érték típusa szerint jellemzően kétfélék lehetnek:

- ▶ **Regresszió**

Ebben az esetben a **kimenet számszerű vagy folytonos**, mint az autókereskedő árbecslése esetén

- ▶ **Klasszifikáció**

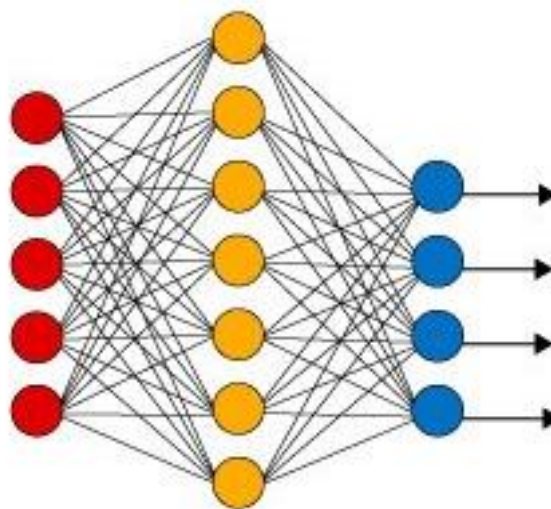
Amennyiben a **kimenet kategorikus**, akkor annak megbecslésére van szükség hogy egy adott elem a paramétereit alapján mely osztályba sorolható.

Az osztályok számától függően megkülönböztethetünk bináris vagy többosztályú klasszifikációt.

# A neurális hálózatokról dióhéjban

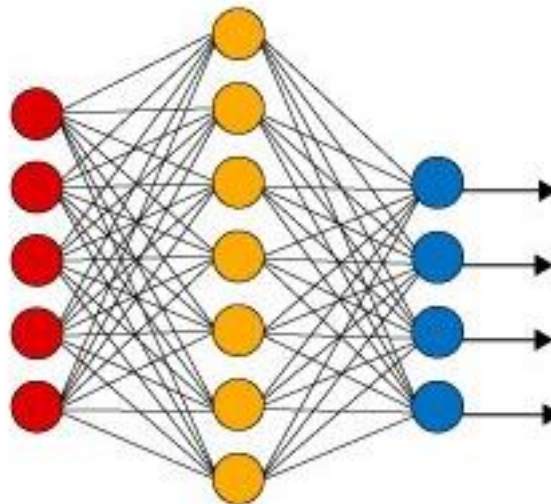
Neurális hálózatoknál tradicionális felépítés a többrétegű előrecsatolt hálózat, ahol több neuron **rétegek**be rendezve kerül összekapcsolásra.

Ezen hálózatok esetén az egyes rétegek neuronjainak kimenetei a következő réteg minden bemenetére lesznek rákötve, kezdve a bemeneti paraméterek számával azonos méretű bemeneti réteggel.





# A neurális hálózatokról dióhéjban



Érdemes megjegyezni, hogy ebben az esetben a súlyparaméterek száma jelentősen növekszik: a sárga színnel ábrázolt úgynevezett rejtett réteg 7 neuronból áll, amelyek mindegyike 5 bemeneti paramétert kap, amely így összesen 35 tanítható paramétert jelent.

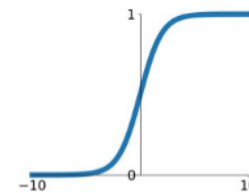
A rétegekben egy további, eltolás (bias) nevű, tanítható paramétert is rendelnek az egyes bemenetekhez.

# A neurális hálózatokról dióhéjban

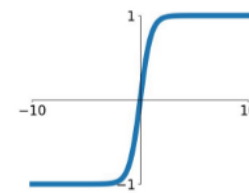
$$A_j = \varphi \left( \sum_{i=1}^N w_i x_i + b \right)$$

Ahol  $A_j$  jelöli a  $j$ . neuron aktivációját,  $\varphi$  az aktivációs függvényt,  $w_i, x_i$  az  $i$ . súly és bemenet értékeket,  $b$  pedig az eltolást.

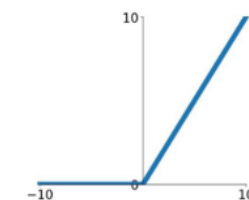
- ▶ Bár a paraméterszám növekszik, é **Sigmoid** rétegű hálózatok egyértelműen jól kinyerésében  $\sigma(x) = \frac{1}{1+e^{-x}}$
- ▶ A mélyebb rétegek összetettebb jeleket **tanh**  $\tanh(x)$
- ▶ Ha megnöveljük a rétegszámot, akkor **ReLU**  $\max(0, x)$



de a több  
abb jellemzők



ii  
ális



**ReLU**  
 $\max(0, x)$

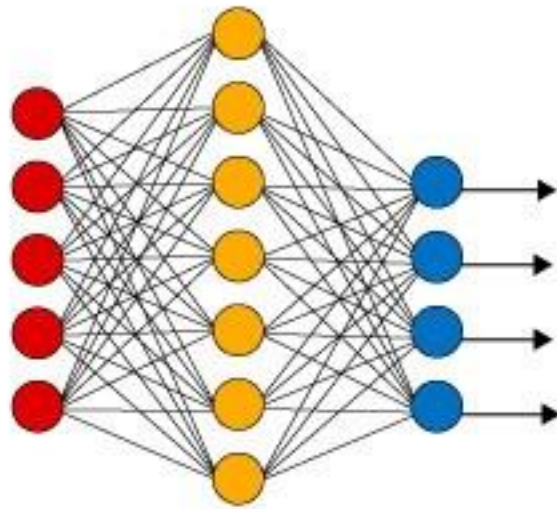


# Mély Neurális Hálózatok

- ▶ A mély hálózat több rétegből áll, a rétegekben nagy számban vannak tanítható paraméterek
- ▶ A tanításhoz nagy számú tanítóminta szükséges (*Big Data*)
  - ▶ Ha nincs elégséges számú tanítóminta, akkor lehet hogy a tanítási adatokra túlságosan illeszkedni fog a modell, és így nem általánosít majd jól (overfitting)
- ▶ A tanítás folyamata így memóriaigényes, és a számítási komplexitás magas
- ▶ **A feladat azonban nagyszerűen párhuzamosítható multiprocesszoros környezetben, kifejezetten GPU gyorsítókkal**

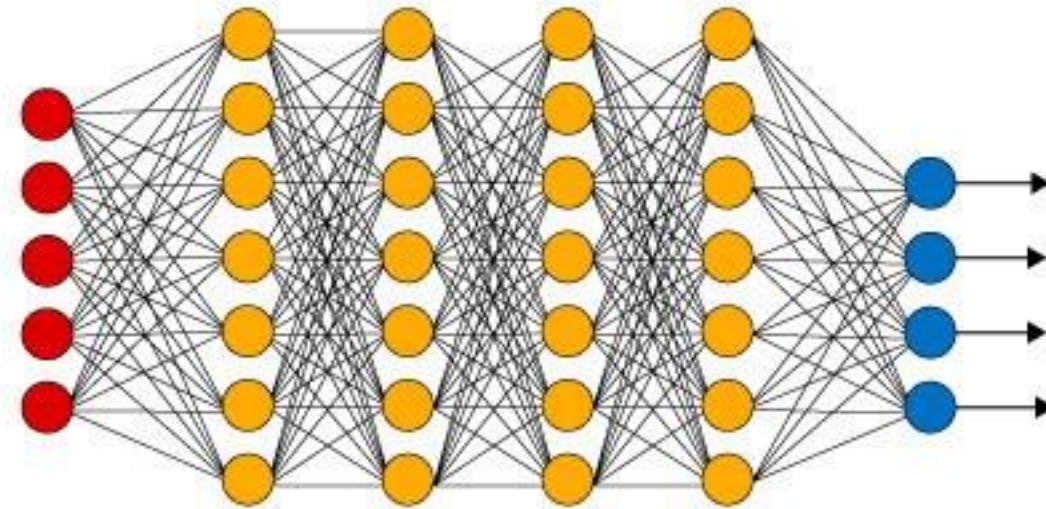
# Mély Neurális Hálózatok

## Simple Neural Network



● Input Layer

## Deep Learning Neural Network



● Hidden Layer

● Output Layer

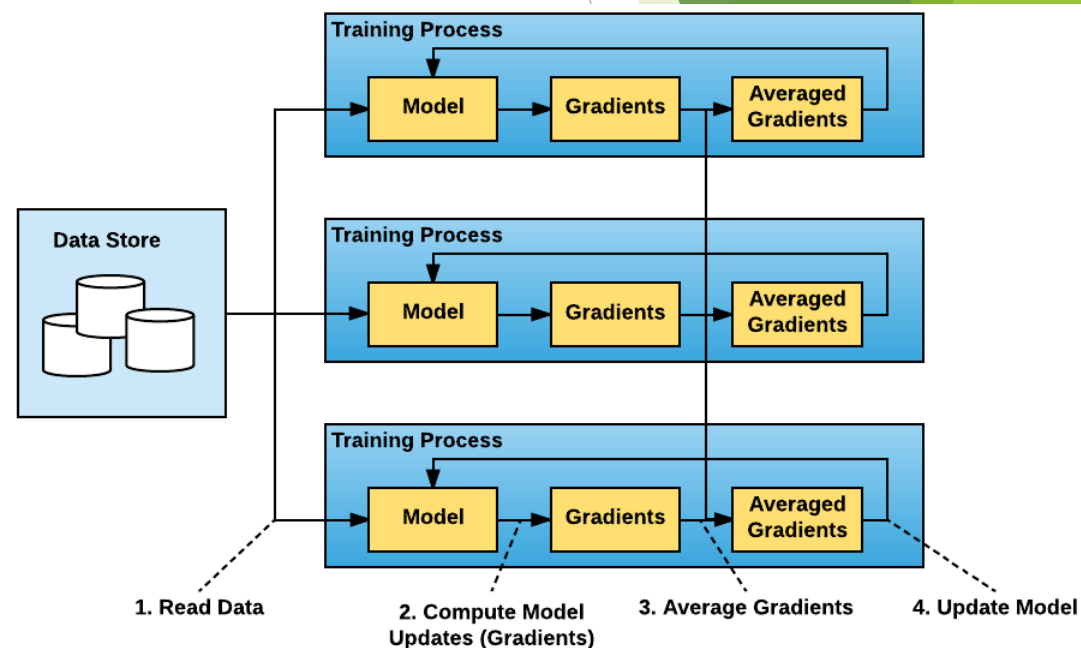
# A mély tanulás számítási komplexitása

- ▶ A mini batcheken alapuló backpropagation algoritmust lehetséges adatpárhuzamosan implementálni
  - ▶ Erre nagyszerű eszköz a grafikus gyorsító, amely több ezer műveletvégző egységgel rendelkezik
  - ▶ A 2010-es évek elején jelentek meg az első GPGPU-implementációk
- ▶ A hardvergyártók esetenként már dedikáltan multiprocesszoros feldolgozásra terveznek eszközöket
  - ▶ Nem grafikai renderelésre, hanem kutatáshoz készülnek
  - ▶ Ezen költséges eszközök beszerzése helyett felhő alapú, igény szerint történő felhasználásuk a logikus



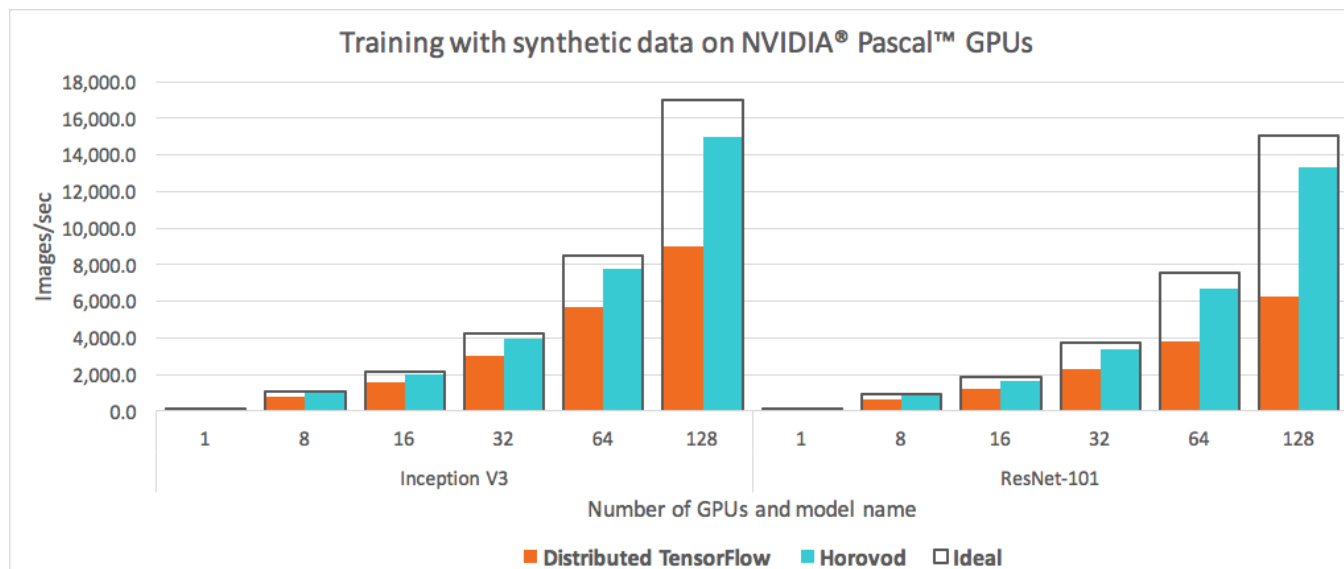
# Elosztott feldolgozás

- ▶ A fizikai korlátok elérésekor érdemes elgondolkodni a horizontális skálázáson, azaz a feladat részfeladatokra bontásán és elosztott feldolgozásán
- ▶ A backpropagation algoritmus esetén sajnos ez nem triviális
- ▶ Az adatdekompozíció alapuló megközelítés népszerű, és könnyen implementálható
  - ▶ A modellről másolatok készülnek klaszter minden elemére
  - ▶ Minden állomáson lezajlik a tanítás egy-egy kötegni tanítómintán
  - ▶ A kiszámolt gradiensok megosztásra kerülnek egymással, a súlyparaméterek együtt kerülnek módosításra



# Elosztott mély tanulás

- ▶ Az elosztott mély tanulás sarokpontja a kommunikáció
  - ▶ A gradiensek cseréje és a paraméterek módosítása költséges is lehet
  - ▶ Nagyobb, mélyebb modell több paramétert jelent, amely nagyobb átviteli költséget eredményez
- ▶ Léteznek megoldások, amelyek magas hatékonyságot ígérnek a skálázáshoz
- ▶ Mély tanulási klasztert nem érdemes saját eszközökből kialakítani, az idő jelentős részében kihasználatlan lenne - éppen erre jó a felhő



# Alkalmazások



- ▶ Ajánlórendszerek
  - ▶ Anomália detektálás
- ▶ Gépi látás
  - ▶ Képi adatok feldolgozása
  - ▶ Többosztályú klasszifikáció
- ▶ Természetes szöveg feldolgozás
  - ▶ Beszéd felismerés
  - ▶ Szöveg generálás
- ▶ Tartalomgenerálás

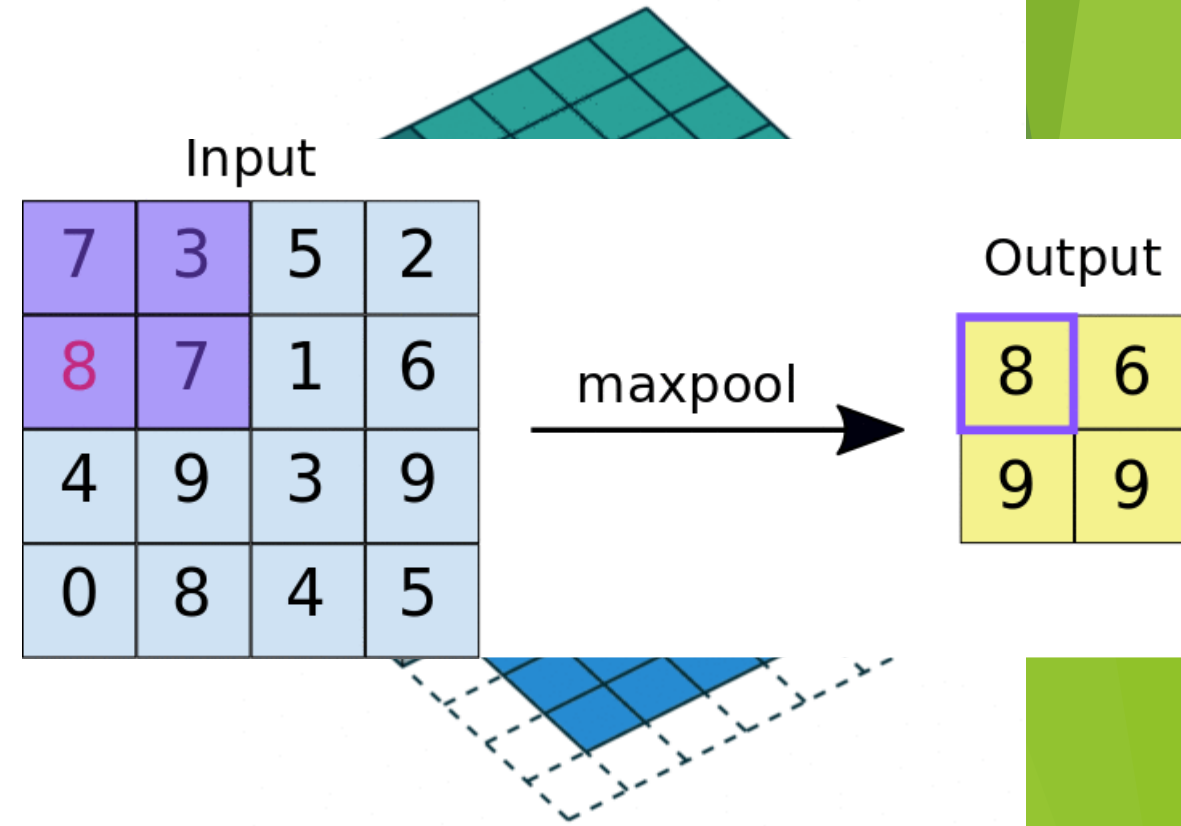




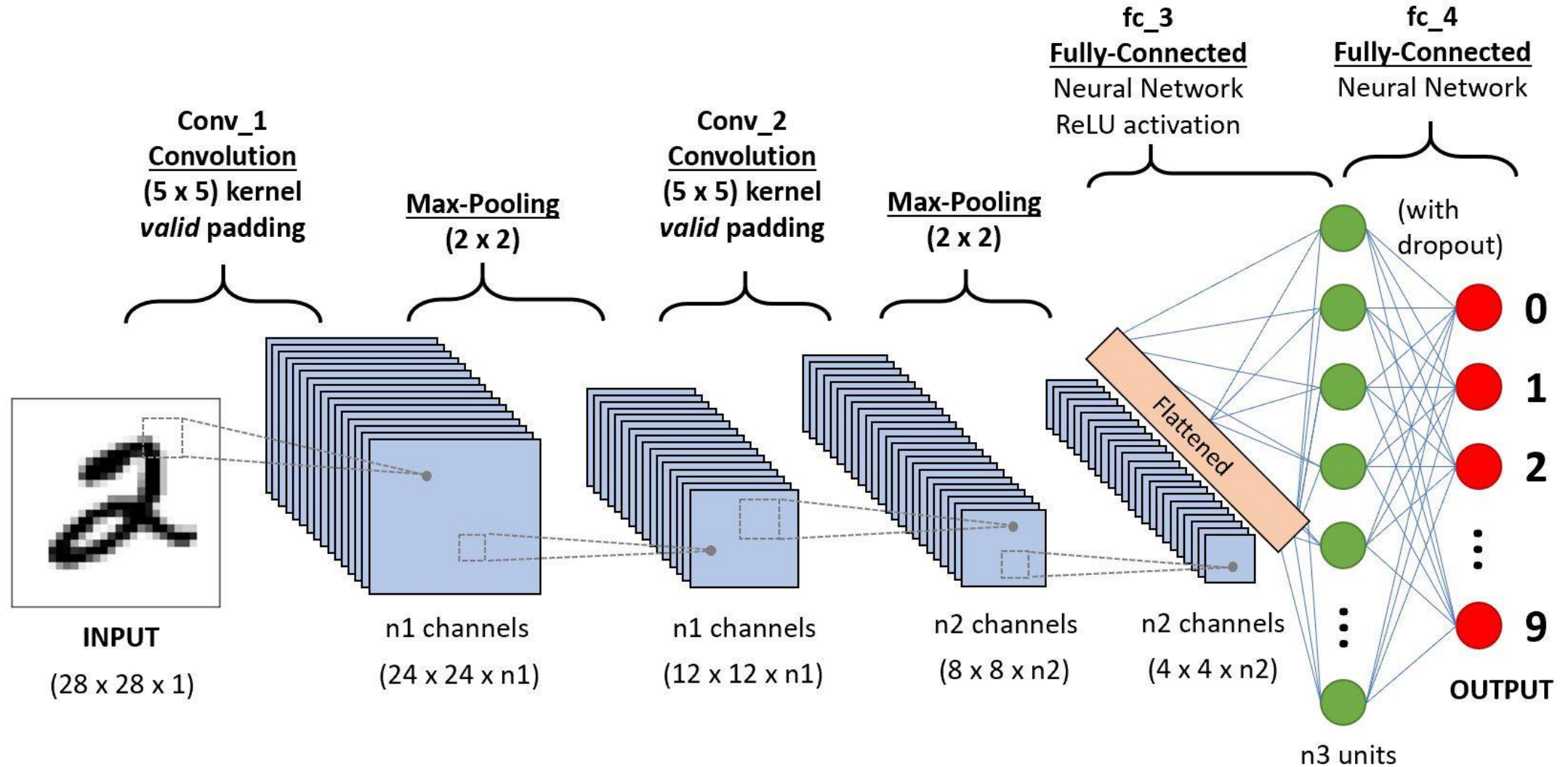
# Konvolúciós Neurális Hálózatok

- ▶ Yann LeCun és tsai, 1989
- ▶ A szomszédos pixelek értékei között kapcsolat van
- ▶ Konvolúciós kernel
  - ▶ Csúszóablakon alapuló megoldás

- Pooling ablak
  - A reprezentáció csökkentésére
  - MaxPooling, AvgPooling, ...

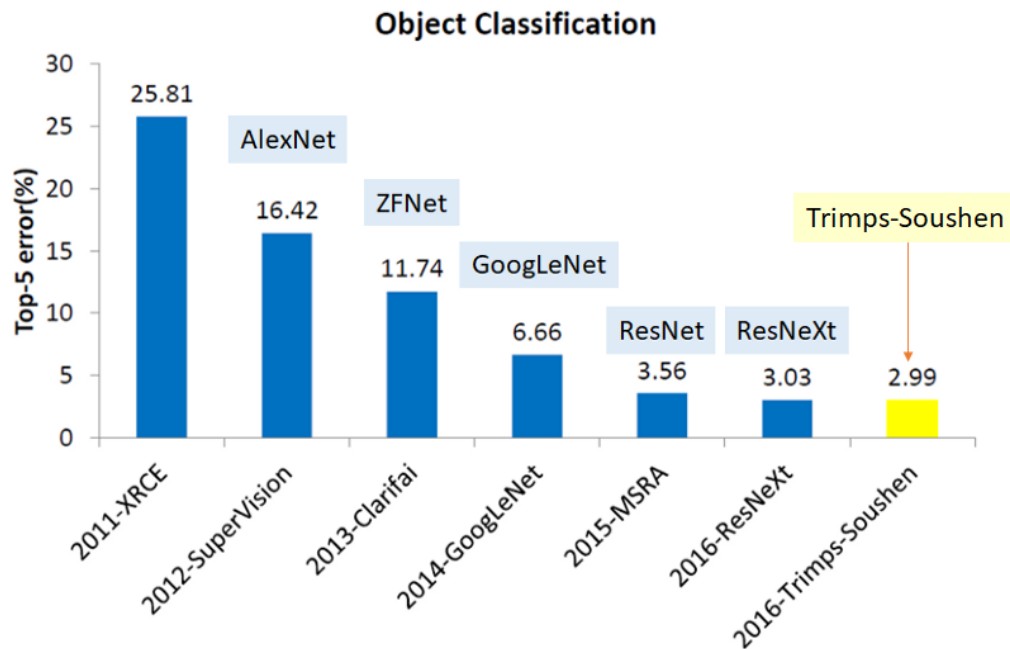


# Konvolúciós hálózatok felépítése



# ImageNet challenge

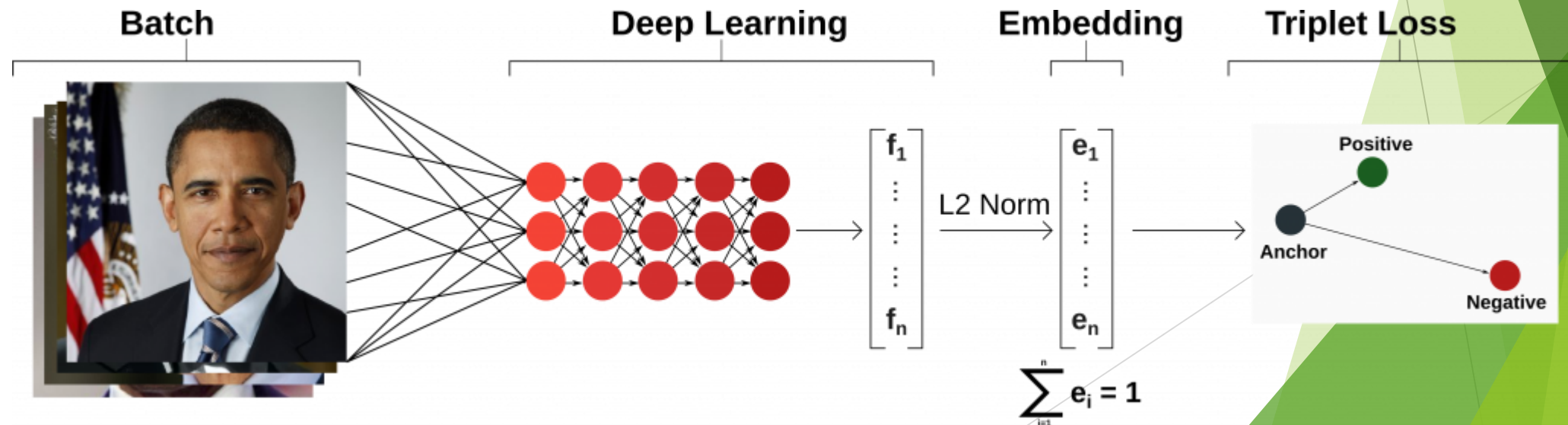
- ▶ ImageNet: 14 millió címkézett képből álló adatbázis
- ▶ ImageNet Large Scale Visual Recognition Challenge
- ▶ 1000 képkategória



2011: 25%  
 2012: 16%  
 2014: <10%  
 2017: <5%

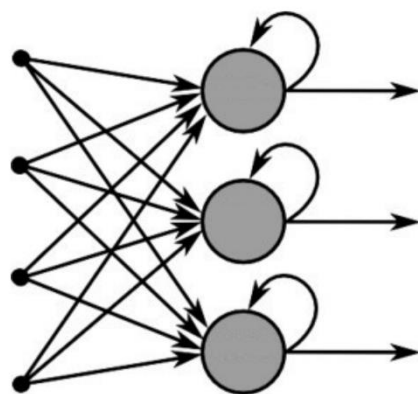
# People identification

- ▶ Arcfelismerés „kevés” tanítóminta alapján
  - ▶ One-shot learning
- ▶ Alacsonyabb dimenzionalitású vektorok összehasonlítása
- ▶ A jellemzők meghatározása és a leképezés gépi tanuláson alapul

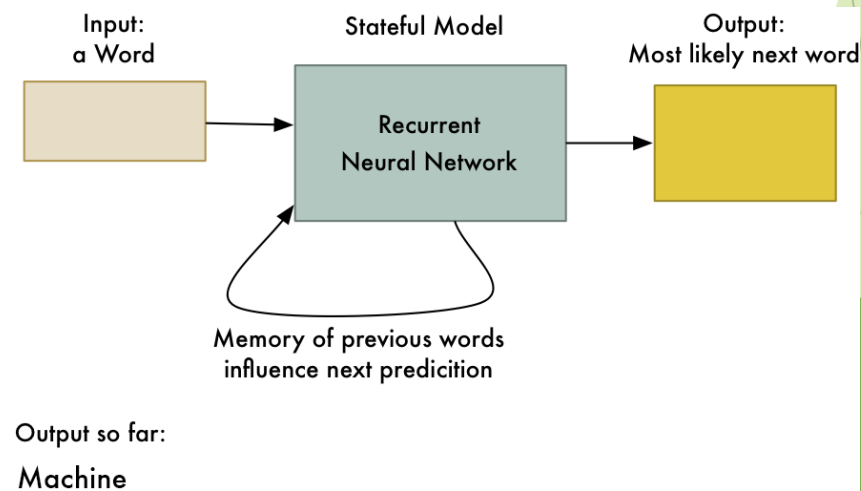


# Rekurrens Neurális Hálózatok

- ▶ Idősoros adatokon végzett feldolgozás
- ▶ Visszacsatolás a hálózatban a korábbi állapot tárolására

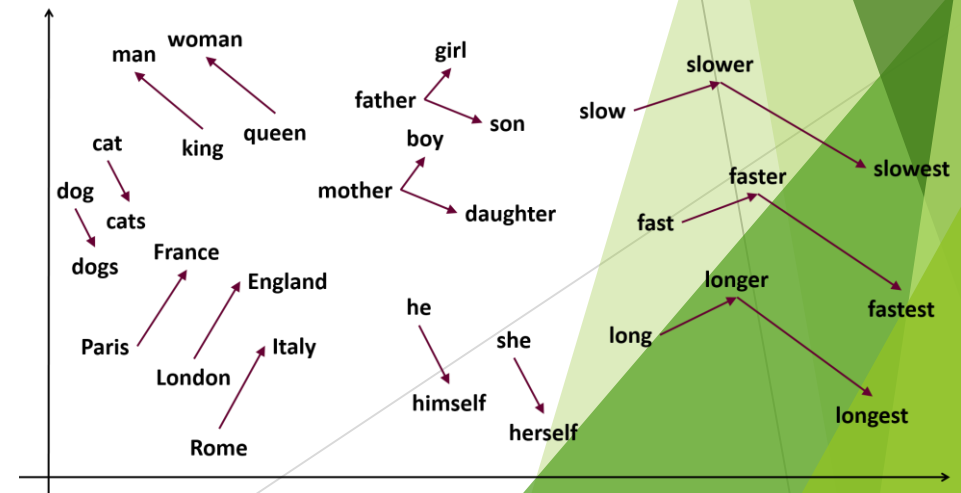
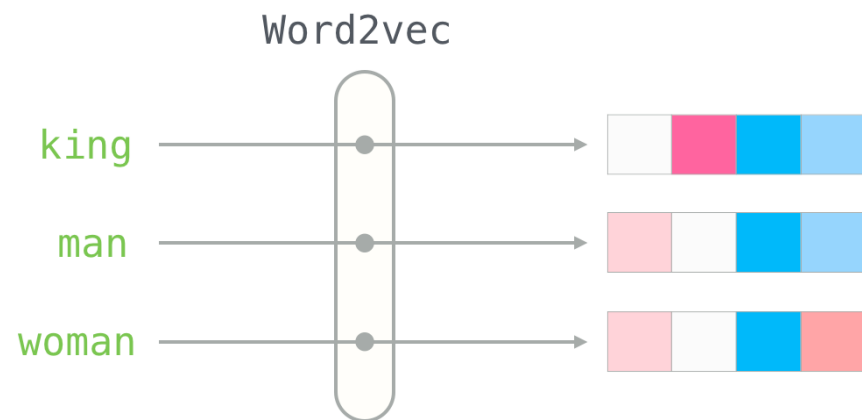


Recurrent Neural Network



# Word2Vec, text recognition

- ▶ Hogyan működnek a chatbotok?
- ▶ A neurális hálózatok bemeneteként továbbra is számérték adható meg
- ▶ A szavak vektorokká alakítása az előfordulásuk és környezetük alapján



# OpenAI text generation

- ▶ GPT-3
- ▶ A világhálón elérhető szövegek alapján végzett „önálló” tanulást
- ▶ Az információkat kigyűjtötte és egy összetett állapotterben tárolja

William Shakespeare ¶

**Transformer AI poetry: Poetry classics as reimaged and rewritten by an artificial intelligence.**

**“Sonnet 18”**

**William Shakespeare**

**Shall I compare thee to a summer’s day?**

**Thou art more lovely and more temperate:**

**Rough winds do shake the darling buds of May,**

**And summer’s lease hath all too short a date;**

A winter’s day, when beams of sun are few,

And skies are grey with clouds intemperate,

Is to thy beauty somehow sordid grown.

Her lovely-lovely eye is quite shut up,

Like windows pale before a sunny shower.

The earth that’s cold and moist, like to my mistress,

When once that she unbrac’d, her lovely jewel,

Grows as both ugly and uncourtly.

Poor soul, the center of my motley host,

See, now they request thy sovereign presence;

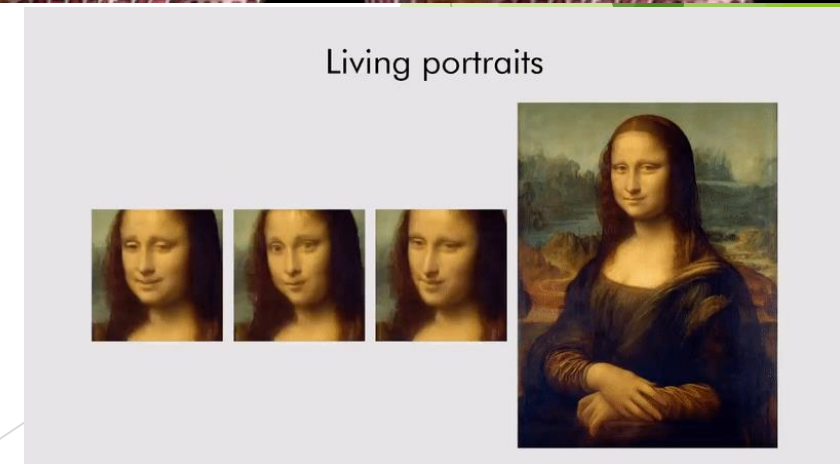
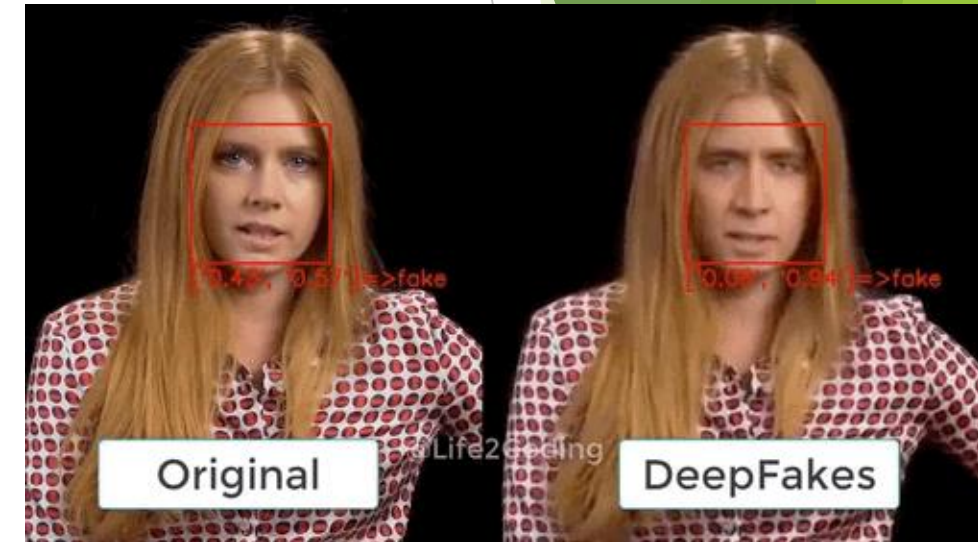
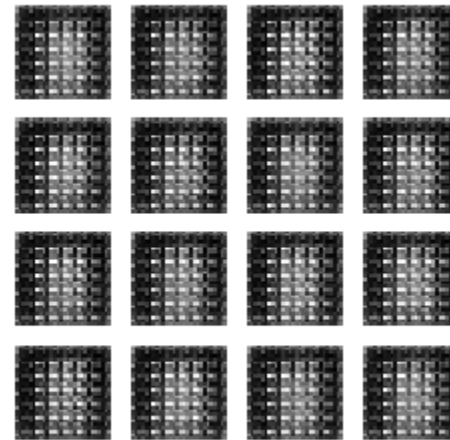
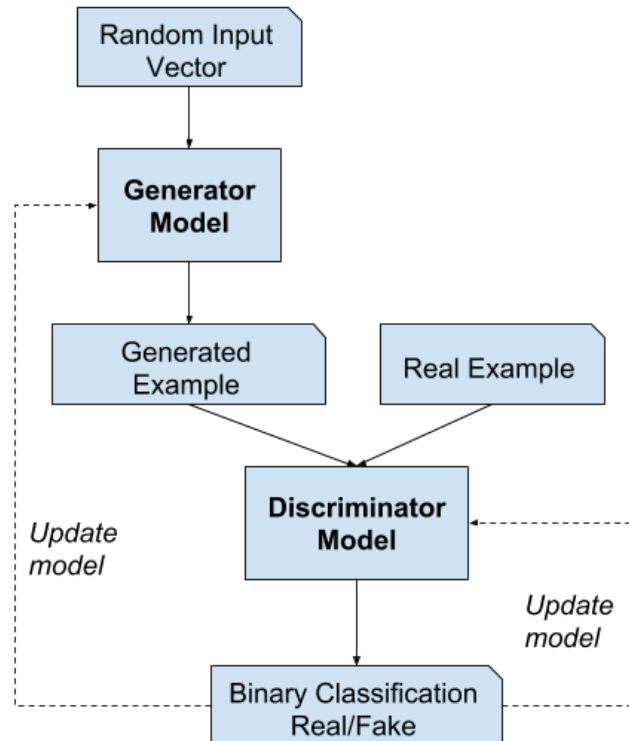
The din of merry hearts hath brought me thus

To greet thee, thou unsoil’d and saint-like beauty.

Brown, Tom B., et al. “Language models are few-shot learners.” *arXiv preprint arXiv:2005.14165* (2020).

# Generative Adversarial Networks

- ▶ Ian Goodfellow, Yoshua Bengio et al
- ▶ Generator vs Discriminator







„Any sufficiently advanced technology is indistinguishable from magic.”

Arthur C. Clarke, 1973

**Köszönöm a figyelmet!**