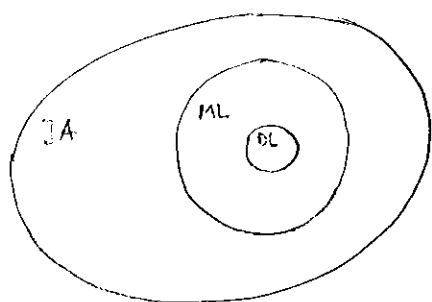


0) Intro:



- IA (en general): (rama de la computación)
 IA sin ML: estudio comportamiento inteligente en computadores
- mucho ingeniería para codificar reglas de deducción
 - conocimiento experto imprescindible
 - funciona en casos donde el problema está muy bien estructurado

ML (en general): extraer patrones desde los mismos datos. Ej supervised learning → programo/algoritmo al que le muestro ejemplos. El algoritmo adapta su funcionamiento a partir de los ejemplos con la esperanza de "generalizar"

[aprender de experiencia E una tarea T respecto de una métrica P → efectividad para resolver T medida por P mejor con experiencia E]

ML sin DL:

- gran parte de lo bueno (o malo) del algoritmo viene dado por qué tan bien se representan los datos de entrada
- aun se necesita conocimiento experto para elegir las características que aseguran una buena predicción/generalización

DL:

- aprender la representación correcta de los datos (representation learning) a medida que se resuelve la tarea en cuestión. En la misma tarea lo que guía la representación
- la representación se aprende combinando representaciones más simples hasta llegar a las más complejas (más "profundas")

Ejemplo: clasificar una frase en positiva o negativa

"Este película es excelente si tienes poco cerebro o quieres perder el tiempo."

Por ejemplo:

- Clasificación de Integran
- Traducción automática

IA sin ML

- programación + lógica + reglas de deducción

ML sin DL

- considerar palabras + cuenta las positivas/negativas + peso de palabras + largo de la frase
- buscar datos que estén etiquetados y entrenar con ellos

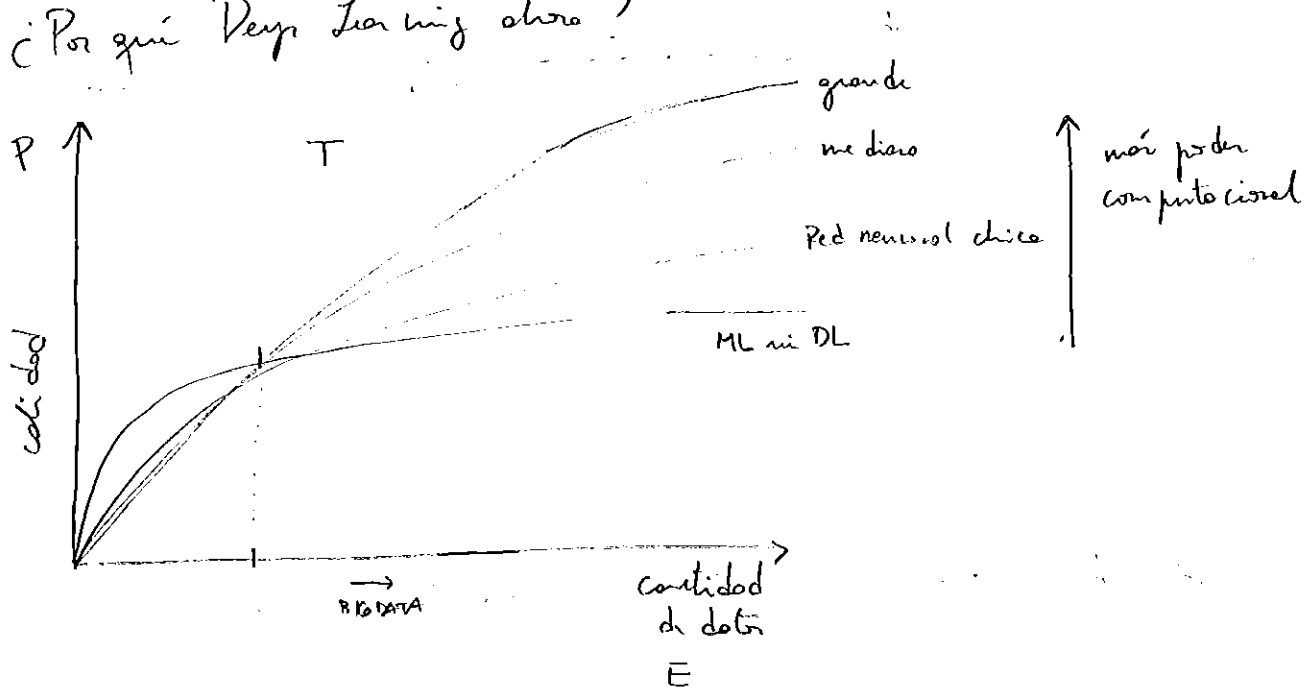
DL:

- uso la frase tal como viene, incluso carácter o carácter
- con muchos datos aprende representaciones más ricas (palabras, positivas/negativas)

DL: genera representaciones usando redes neuronales ordenadas en capas
 ↳ Sigo: anime que de tona se puede resolver como una composición jerárquica de partes.

- Necesito mucho datos para pasarle al computador ✓
- Necesito mucho tiempo de computación ✓

¿Por qué Deep Learning ahora?



Interludio: ¿por qué necesitamos profundidad?

PARIDAD: dado un conjunto de bits $x_1, \dots, x_n \rightsquigarrow (\sum x_i) \% 2$

Teo: PARIDAD no se puede computar con una cantidad polinomial de computación en un circuito de profundidad constante (PARIDAD $\notin AC^0$)

Se puede hacer con profundidad constante, leyendo $x_1 \text{ XOR } x_2 \text{ XOR } \dots \text{ XOR } x_n$ y luego llevándolo a CNF

