

Aprendizaje profundo y sus usos en las políticas públicas

Objetivos del curso

- Entender las arquitecturas centrales del aprendizaje profundo (CNNs, RNNs, Transformers, etc.).
 - Implementar modelos de aprendizaje profundo de extremo a extremo usando **PyTorch**.
 - Evaluar de manera crítica modelos de aprendizaje profundo (desempeño, equidad, impacto social).
 - Aplicar aprendizaje profundo a problemas reales (análisis de texto, imágenes, series temporales, etc.).
 - Diseñar sistemas robustos, transparentes y explicables — más allá de “cajas negras”.
-

Prerrequisitos (qué deberías saber antes)

- **Programación en Python:** estructuras, funciones, clases y OOP básica.
 - **ML básico:** funciones de pérdida, optimización (GD/SGD), evaluación y métricas.
 - **Flujos de trabajo de ciencia de datos:** limpieza, split train/val/test, versionado de código y datos.
-

Roadmap (resumen semanal)

- **Semanas 1–4 — Fundamentos**
 - Repaso ML básico
 - Redes neuronales superficiales y profundas
 - Entrenamiento (optimización, inicialización, performance)
 - Regularización
 - **Semanas 5–7 — CNNs y RNNs**
 - **Semanas 8–10 — Transformers**
 - **Semana 11 — Por qué funciona DL / Fairness / Ética / preguntas abiertas**
 - **Semana 12 — Presentaciones**
-

Evaluación

- **8 tareas** (preguntas teóricas + ejercicios en Python) — **60%**.
 - **Nota:** NO está permitido usar LLMs para resolver las tareas. Sí se permite consultar documentación oficial, libros, StackOverflow y trabajar con compañeros.
- **Presentación de paper (en parejas)** — **40%**.
 - Papers asignados aleatoriamente. Formato y rúbrica en guía aparte (MD).

- **Para las presentaciones** se permite usar LLMs **para entender** o preparar ideas, **pero es obligatorio** entregar la conversación/registro (chat) junto con los entregables.
-

Papers a presentar (lista)

1. AlexNet — Krizhevsky, Sutskever & Hinton (2012)
 2. ResNet — He et al. (2015)
 3. Batch Normalization — Ioffe & Szegedy (2015)
 4. Attention is All You Need (Transformers) — Vaswani et al. (2017)
 5. BERT — Devlin et al. (2018)
 6. Variational Autoencoders (VAE) — Kingma & Welling (2013)
 7. GANs — Goodfellow et al. (2014)
 8. Double Descent — Belkin et al. (2019) / Nakkiran et al. (2019)
-

Libros y lecturas recomendadas

- Prince — *Unifying Deep Learning* (texto base del curso)
 - Goodfellow, Bengio & Courville — *Deep Learning*
 - Zhang et al. — *Dive into Deep Learning (d2l)*
 - Chollet — *Deep Learning with Python*
-

Papers de DL en políticas públicas (opcional / lectura recomendada)

Medición de pobreza con imágenes satelitales - Jean et al. (2016) — *Combining satellite imagery and machine learning to predict poverty* (Science)

- Huang, Hsiang & Gonzalez-Navarro (2021) — *Using Satellite Imagery and Deep Learning to Evaluate the Impact of Anti-Poverty Programs* (NBER)

- Babenko et al. (2017) — *Poverty Mapping Using Convolutional Neural Networks* (arXiv)

Análisis de textos legislativos - Korn & Newman (2020) — *A Deep Learning Model to Predict Congressional Roll Call Votes From Legislative Texts*

- Chalkidis et al. (2019) — *Large-Scale Multi-Label Text Classification on EU Legislation*

- Wei et al. (2019) — *Empirical Study of Deep Learning for Text Classification in Legal Document Review*

Notas prácticas (reproducibilidad y ejecución)

- **Notebooks:** todos los notebooks de clase estarán en **PyTorch** y contendrán instrucciones reproducibles: semilla fija, detección de device (CPU/GPU/MPS), y opciones para checkpoints.

- **Si no tienes GPU:** diseña experimentos pequeños (submuestras, redes reducidas) que permitan ilustrar los fenómenos estudiados.