# Curso de IA y modelos de lenguaje con PyTorch: De las Redes Neuronales a GPT

Instituto Politécnico Nacional
Centro de Investigación en Computación
Departamento de Diplomados y Extensión Profesional

PLAN

# DE ESTUDIO

## DURACIÓN: 40 horas

**OBJETIVO**

Al terminar el curso el estudiante podrá:

- Comprender la matemática detrás del Deep Learning.
- Implementar redes neuronales en PyTorch.
- Entrenar y entender modelos de IA propios.
- Construir un pequeño modelo de lenguaje tipo GPT.

**PERFIL DE LOS PARTICIPANTES**

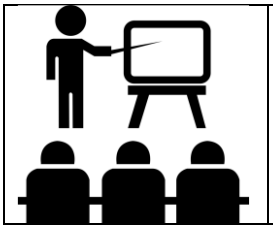
El participante que aspire a tomar el curso deberá poseer conocimientos de:

- Buena familiaridad con programación Python
- Álgebra básica (se repasa lo necesario)

**DIRIGIDO A:**

Estudiantes y profesionistas con conocimientos de Python que quieran iniciarse en la IA moderna, aprendiendo a construir redes neuronales, Transformers y modelos tipo GPT con PyTorch.

Centro de Investigación en Computación
Departamento de Diplomados y Extensión Profesional



# CONTENIDO
# DEL CURSO

## Unidad 1. Fundamentos matemáticos y Machine Learning

- Introducción al Machine Learning
  - ¿Qué es IA?
  - Machine Learning vs Deep Learning
  - Aprendizaje supervisado
  - Tipos de datos
  - Entrenamiento, validación y prueba
  - Métricas de evaluación

- Álgebra lineal necesaria
  - Multiplicación matricial
  - Producto punto
  - Normas, identidad y transpuesta
  - Inversa (intuición)
  - Cambio de dimensiones e interpretación geométrica

- Cálculo necesario
  - Funciones
  - Derivadas y gradientes
  - Regla de la cadena
  - Jacobiano (intuición)
  - Hessiano (intuición)

- Probabilidad
  - Distribuciones
  - Esperanza y varianza
  - Normal y Bernoulli
  - Información, entropía y Cross Entropy
  - KL Divergence (intuición)

## Unidad 2. PyTorch

- Numpy para Deep Learning
- Pandas para Deep Learning

- Fundamentos de Pytorch
  - Tensor Library
  - Autograd
  - GPU
  - Broadcasting

- Pytorch para Deep Learning
  - Datasets y Dataloaders
  - nn.Module
  - Capas principales en Deep Learning
  - Ciclo de entrenamiento
  - Gestión de modelos

## Unidad 3. Redes neuronales desde cero

- Neurona artificial
  - Perceptrón
  - Capas
  - Funcionamiento
  - El problema del perceptrón

- Perceptrón multicapa (MLP)
  - Conversión y manejo de datos
  - Transformaciones por capa
  - Funciones de activación
  - Profundidad y conocimiento jerarquizado
  - Generalización y aproximación de funciones

- Entrenamiento de MLP: Backpropagation
  - Derivación completa
  - Regla de la cadena
  - Funciones de pérdida
  - Gradientes
  - Actualización de pesos

- Laboratorio: Implementación desde cero
  - Implementar una red neuronal usando únicamente operaciones de PyTorch.

## Unidad 4. Optimización y entrenamiento

- Descenso de gradiente
  - Batch
  - Mini Batch
  - Stochastic
- Optimizadores
  - Gradient Descent
  - Momentum
  - Nesterov
  - Adam
  - AdamW
- Actualización de hiperparámetros
  - Inicialización
  - Learning Rate
  - Warmup
  - Step
  - Transfer Learning
- Overfitting
  - Bias
  - Variance
  - Diagnóstico
- Laboratorio
  - Entrenar una red neuronal completa.
  - Visualizar curvas de entrenamiento.

## Unidad 5. Técnicas de aprendizaje

- Batch y Layer Normalization
  - Motivación
  - Funcionamiento
  - Entrenamiento
  - Inferencia
- Regularización
  - L1
  - L2
  - Weight Decay
  - Dropout
  - Early Stopping
- Data Augmentation
  - Motivación
  - Funcionamiento

- Conexiones residuales
    - Problema de degradación
    - Vanishing Gradient
    - Residual Learning
    - Identity Mapping
- Arquitecturas profundas
    - MLP modernos
    - PreNorm
    - PostNorm
- Laboratorio
    - sin regularización
    - con Dropout
    - con BatchNorm

## Unidad 6. Transformers

- Motivación
    - Fundamentos de Redes Neuronales Recurrentes (RNN)
    - ¿Por qué abandonar las RNN?
    - Problema de dependencias largas
    - Paralelización
- Atención
    - Idea intuitiva
    - Transformaciones
    - Producto punto
    - Scaled Dot Product Attention
    - Matrices de atención
- Self-Attention
    - Cálculo paso a paso
    - Máscaras
    - Multi Head Attention
    - Concatenación
    - Proyección
- Positional Encoding
    - Necesidad
    - Sinusoidal
    - Embeddings aprendidos
- Arquitectura Transformer
    - Encoder
    - Decoder
- Laboratorio
    - Implementar Self Attention desde cero.

## Unidad 7. Modelos de lenguaje

- Tokenización
  - Word
  - Subword
  - BPE
  - SentencePiece
  - Tiktoken
- Embeddings
  - Word2Vec
  - Embeddings aprendidos
  - Posicionales
- Objetivos de entrenamiento
  - Language Modeling
  - Next Token Prediction
  - Masked Language Modeling
  - Teacher Forcing

- Arquitecturas: GPT y BERT
  - Arquitecturas
  - Entrenamiento
  - Inferencia
- Decodificación
  - Greedy
  - Temperature
  - Top-k
  - Top-p
  - Beam Search
- Laboratorio
  - Construir un GPT pequeño.
  - Entrenarlo sobre un texto pequeño.
  - Generar texto.

## Unidad 8. Modelos modernos y Fine Tuning

- Escalamiento
  - Scaling Laws
  - Chinchilla
  - Parámetros
  - Tokens
  - Compute
- Fine Tuning
  - Congelar capas
  - Transfer Learning
- PEFT
  - LoRA
  - QLoRA
  - Adapters
  - Prompt Tuning
- Cuantización
  - INT8
  - 4 bits
  - ¿Por qué funciona?
- Modelos abiertos
  - Llama
  - Mistral
  - Qwen
  - Gemma
  - DeepSeek

# CONTACTO

https://www.cic.ipn.mx/views/links/courses.php

**TELEFONOS: 55 57296000 ext. 56605, 56510 y 56644**

diplomados@cic.ipn.mx

cursos@cic.ipn.mx

**CAPACITACIONCIC**
**CURSOS Y DIPLOMADOS CIC**

**DIRECCIÓN: Av. Juan de Dios Bátiz, esq. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, Alcaldía Gustavo A. Madero, C.P. 07738, CDMX**