สถาบันไอทีจีเนียส
IT GENIUS INSTITUTE

COURSE ID
**WDC-94**
AI / DEEP LEARNING

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

# Deep Learning with Python

## สร้าง ฝึก และนำโมเดล Deep Learning ไปใช้ด้วย PyTorch และ Keras 3

- 18 ชั่วโมง
- 3 วัน
- พื้นฐาน Python และ ML เบื้องต้น
- ระดับ Advanced




**ระยะเวลา: 3 วัน**
18 ชั่วโมง (3 วัน วันละ 6 ชม.) · 09:00-16:00 น.

**รูปแบบ: Classroom / Hybrid**
Onsite หรือ Online ผ่าน Microsoft Teams · Public / In-house

**รายละเอียดหลักสูตร**
www.itgenius.co.th

Deep Learning คือกลไกเบื้องหลังระบบ AI ที่ใช้อยู่ทุกวัน ทั้งการจำแนกภาพ ตรวจจับวัตถุจากกล้อง วิเคราะห์ความรู้สึกจากข้อความ ไปจนถึงโมเดลภาษาขนาดใหญ่ แต่เครื่องมือเปลี่ยนเร็วมาก โค้ดแบบ TensorFlow 1 และ Keras 2 ที่เคยสอนกันใช้ไม่ได้แล้ว ไลบรารีหลักอย่าง Hugging Face Transformers ก็หันมาใช้ PyTorch เป็นหลัก นักพัฒนาที่อยากสร้างโมเดลเองจึงต้องเข้าใจทั้งหลักการและเครื่องมือชุดปัจจุบัน

หลักสูตรนี้พาผู้เรียนสร้างโมเดล Deep Learning ด้วย Python โดยใช้ PyTorch 2 เป็นแกนหลักร่วมกับ Keras 3 เริ่มจากหลักการ Neural Network, Backpropagation และ Training Loop ที่เขียนเอง แล้วต่อด้วยเทคนิคฝึกโมเดล CNN สำหรับงานภาพ Transfer Learning การตรวจจับวัตถุด้วย Ultralytics YOLO26 และ Autoencoder จากนั้นเข้าสู่ข้อมูลลำดับด้วย LSTM และ GRU สถาปัตยกรรม Transformer การ Fine-tune โมเดลวิเคราะห์ความรู้สึกภาษาไทยด้วย Hugging Face และการ Export โมเดลเป็น ONNX ปิดท้ายด้วย Capstone ที่สร้างโมเดลพร้อมหน้าเดโมของตนเอง (อบรม 3 วัน วันละ 6 ชั่วโมง รวม 18 ชั่วโมง ระดับ Advanced)

เครื่องมือ: Python · PyTorch · Keras · Hugging Face · GPU / Colab

ราคาค่าอบรม · PUBLIC / IN-HOUSE

**13,500** บาท / ท่าน

สมัครผ่านเว็บไซต์ ลด 10%

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม Lab Guide, Notebook และชุดข้อมูลตัวอย่างครบทุก Lab และ Certificate · รับจัด In-house ขอใบเสนอราคาได้

เพิ่มเพื่อน LINE


02-570-8449 · 088-807-9770 · @itgenius · www.itgenius.co.th

## 01 วัตถุประสงค์ Objectives

- เข้าใจหลักการ Neural Network, Loss Function, Backpropagation และ Optimizer
- ใช้ Tensor, Autograd, Dataset และ DataLoader ของ PyTorch และเขียน Training Loop เองได้
- สร้างและฝึกโมเดลด้วย Keras 3 พร้อมเข้าใจการเลือก Backend
- สร้าง CNN จำแนกภาพและใช้ Transfer Learning จากโมเดลสำเร็จรูป
- ฝึกโมเดลตรวจจับวัตถุด้วย YOLO26 กับชุดข้อมูลของตนเอง
- ใช้ LSTM, GRU และ Transformer กับข้อมูลลำดับและข้อความ
- Fine-tune โมเดลภาษาสำหรับวิเคราะห์ความรู้สึกด้วย Hugging Face Transformers
- Export โมเดลเป็น ONNX และทำหน้าเดโมให้ผู้อื่นทดลองใช้

## 02 กลุ่มเป้าหมาย Target Audience

- นักพัฒนาและ Web Developer ที่มีพื้นฐาน Python และต้องการพัฒนาระบบ AI ด้วย Deep Learning
- Data Scientist และ ML Engineer ที่ต้องการอัปเดตจาก TensorFlow 1 หรือ Keras 2 สู่เครื่องมือปัจจุบัน
- ผู้ที่ผ่านหลักสูตร Python for Machine Learning หรือมีพื้นฐาน Machine Learning มาแล้ว
- อาจารย์และนักศึกษาที่ทำโปรเจกต์ด้านภาพ ข้อความ หรือข้อมูลอนุกรมเวลา

## 03 จุดเด่นของหลักสูตร Highlights

- **ปรับปรุงใหม่ปี 2026** PyTorch 2, Keras 3, YOLO26 และ Transformers
- **เข้าใจถึงแกน** เขียน Training Loop เองก่อนใช้เครื่องมือสำเร็จรูป
- **ครบทั้งภาพและข้อความ** CNN, YOLO, LSTM และ Transformer
- **ใช้กับภาษาไทยได้จริง** Fine-tune โมเดลวิเคราะห์ความรู้สึกภาษาไทย
- **นำไปใช้งานต่อได้** Export ONNX และหน้าเดโมด้วย Gradio

## 04 ความรู้พื้นฐาน Prerequisites

- เขียน Python ได้คล่อง รวมถึง Function, Class และการใช้ NumPy กับ pandas เบื้องต้น
- เข้าใจพื้นฐาน Machine Learning เช่น Train/Test Split, Overfitting และการวัดผลโมเดล
- คณิตศาสตร์ระดับมัธยมปลาย เรื่องเมทริกซ์และอนุพันธ์จะอธิบายเท่าที่จำเป็นในชั้นเรียน
- บัญชี Google สำหรับใช้ Google Colab และบัญชี Hugging Face แบบใช้ฟรี

## 05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

| รายการ | รายละเอียด |
|---|---|
| **อุปกรณ์** | Notebook เชื่อมต่ออินเทอร์เน็ต ไม่จำเป็นต้องมี GPU ในเครื่อง |
| **GPU** | Google Colab แบบใช้ฟรี หรือ Cloud GPU ที่ผู้เรียนชำระเอง |
| **เฟรมเวิร์ก** | Python 3.12 ขึ้นไป, PyTorch 2, torchvision และ Keras 3 |
| **ไลบรารีเสริม** | Ultralytics YOLO26, Hugging Face Transformers, Datasets, ONNX Runtime และ Gradio |

**หมายเหตุ:** ปรับปรุงใหม่ปี 2026: เปลี่ยนจาก TensorFlow แบบเดิมมาใช้ PyTorch 2 ร่วมกับ Keras 3 อัปเดตการตรวจจับวัตถุเป็น YOLO26 และย้ายงาน BERT มาใช้ Hugging Face Transformers ที่เน้น PyTorch เพิ่ม Transformer, Fine-tuning และการ Export เป็น ONNX ส่วน GAN สรุปเป็นภาพรวม Generative Model Lab ใช้ Colab แบบใช้ฟรีได้แต่โควตา GPU ไม่แน่นอน ผู้ที่ต้องการ GPU ต่อเนื่องใช้ Colab แบบเสียเงินหรือ Cloud GPU ที่ชำระเอง YOLO ของ Ultralytics ใช้สัญญาอนุญาต AGPL-3.0 การใช้เชิงพาณิชย์ต้องตรวจเงื่อนไขหรือซื้อ Enterprise License

## 06 ภาพรวมการเรียน 3 วัน Big Picture

| วัน | หัวข้อหลัก |
|---|---|
| **Day 1** Section 1-5 | เตรียม GPU บน Colab, เข้าใจ Neural Network และ Backpropagation, ใช้ Tensor และ Autograd ของ PyTorch, เขียน Training Loop เอง แล้วสร้างโมเดลแบบเร็วด้วย Keras 3 |
| **Day 2** Section 6-10 | เทคนิคฝึกโมเดลให้ดีและเร็ว, สร้าง CNN จำแนกภาพ, Transfer Learning จากโมเดลสำเร็จรูป, ตรวจจับวัตถุด้วย YOLO26 และ Autoencoder กับภาพรวม Generative Model |
| **Day 3** Section 11-15 | พยากรณ์ข้อมูลลำดับด้วย LSTM และ GRU, เข้าใจ Transformer และ Attention, Fine-tune โมเดลวิเคราะห์ความรู้สึกภาษาไทยด้วย Hugging Face, Export เป็น ONNX และ Capstone |

DAY 1

# Neural Network, PyTorch และ Keras 3

เนื้อหาการอบรม · Section 1-5

รวม
**6 ชั่วโมง**

## 1 Lab: เตรียมสภาพแวดล้อม Deep Learning

- ภาพรวมเครื่องมือ Deep Learning ปี 2026: PyTorch, Keras 3 และ Hugging Face
- ใช้ Google Colab กับ GPU และตรวจว่าโค้ดรันบน CUDA จริง
- ติดตั้งบนเครื่องตนเองด้วย uv และทางเลือก Apple Silicon (MPS)
- จัดโครงสร้างโปรเจกต์และบันทึกเวอร์ชันไลบรารีเพื่อทำซ้ำผลได้

## 2 จาก Machine Learning สู่ Neural Network

- Neuron, Weight, Bias และ Activation Function อย่าง ReLU และ Softmax
- Loss Function สำหรับ Regression และ Classification
- Backpropagation และ Gradient Descent ทำงานอย่างไร
- Optimizer อย่าง SGD, Adam และ AdamW และบทบาทของ Learning Rate
- ภาพรวม CNN, RNN, Transformer และ Foundation Model ในปัจจุบัน

## 3 Lab: Tensor และ Autograd ใน PyTorch

- สร้างและจัดการ Tensor, Shape และการย้ายข้อมูลไป GPU
- Autograd คำนวณ Gradient ให้อัตโนมัติอย่างไร
- สร้างโมเดลด้วย nn.Module และชั้นพื้นฐานอย่าง Linear
- เตรียมข้อมูลด้วย Dataset และ DataLoader แบบ Batch

## 4 Lab: เขียน Training Loop ด้วย PyTorch

- ลำดับ Forward, Loss, Backward และ Optimizer Step
- แยกชุด Train และ Validation พร้อมติดตาม Loss และ Accuracy
- สร้าง MLP สำหรับงาน Regression และ Classification
- บันทึกและโหลดโมเดลด้วย state_dict และ Checkpoint
- อ่านกราฟการฝึกเพื่อดู Overfitting และ Underfitting

## 5 Lab: สร้างโมเดลเร็วขึ้นด้วย Keras 3

- Keras 3 แบบหลาย Backend เลือกรันบน PyTorch, JAX หรือ TensorFlow
- Sequential API และ Functional API สำหรับโมเดลหลายอินพุต
- compile, fit และ Callback อย่าง EarlyStopping และ ModelCheckpoint
- บันทึกโมเดลเป็นไฟล์ .keras และข้อแตกต่างจาก Keras 2
- เทียบโค้ด Keras กับ Training Loop ที่เขียนเองด้วย PyTorch

DAY 2

# Computer Vision และ Generative Model

เนื้อหาการอบรม · Section 6-10

รวม
**6 ชั่วโมง**

## 6 เทคนิคฝึกโมเดลให้ดีและเร็ว

- ลด Overfitting ด้วย Dropout, Weight Decay และ Data Augmentation
- Batch Normalization และการกำหนดค่าเริ่มต้นของ Weight
- Learning Rate Schedule และ Early Stopping
- Mixed Precision ด้วย torch.autocast และเร่งความเร็วด้วย torch.compile
- ติดตามการทดลองและเปรียบเทียบผลอย่างเป็นระบบ

## 7 Lab: CNN สำหรับจำแนกภาพ

- Convolution, Filter, Stride, Padding และ Pooling
- สร้าง CNN จำแนกภาพสินค้าและภาพทั่วไปตั้งแต่ศูนย์
- ดูภาพ Feature Map เพื่อเข้าใจสิ่งที่แต่ละชั้นเรียนรู้
- วัดผลด้วย Confusion Matrix และวิเคราะห์ภาพที่ทายผิด

## 8 Lab: Transfer Learning และ Fine-tuning

- โหลดโมเดลที่ฝึกมาแล้วจาก torchvision พร้อม Weight สำเร็จรูป
- ตรึงชั้นต้น เปลี่ยนชั้นสุดท้าย แล้วฝึกกับคลาสของเราเอง
- ปลดตรึงบางชั้นเพื่อ Fine-tune ด้วย Learning Rate ต่ำ
- ใช้โมเดลสำเร็จรูปจาก KerasHub เป็นทางเลือก
- Lab: สร้างตัวจำแนกภาพจากชุดข้อมูลขนาดเล็กของตนเอง

## 9 Lab: ตรวจจับวัตถุด้วย YOLO26

- หลักการ Object Detection, Bounding Box, IoU และ mAP
- ใช้โมเดล YOLO26 ที่ฝึกมาแล้วกับภาพและวิดีโอด้วย Ultralytics
- เตรียมชุดข้อมูลแบบ YOLO Format และติด Label ภาพ
- ฝึกโมเดลกับข้อมูลของตนเองและอ่านผลการวัด
- Export โมเดลและข้อควรรู้เรื่องสัญญาอนุญาต AGPL-3.0

## 10 Autoencoder และ Generative Model

- Autoencoder สำหรับบีบอัดข้อมูลและลบสัญญาณรบกวนในภาพ
- ตรวจจับความผิดปกติจากค่า Reconstruction Error
- แนวคิด VAE และ GAN แบบเข้าใจง่าย
- ภาพรวม Diffusion Model ที่อยู่เบื้องหลังเครื่องมือสร้างภาพ
- Lab: Autoencoder ตรวจจับภาพสินค้าที่ผิดปกติ

DAY 3

## ข้อมูลลำดับ, Transformer และการนำไปใช้

เนื้อหาการอบรม · Section 11-15

รวม
**6 ชั่วโมง**

### 11 Lab: RNN, LSTM และ GRU กับข้อมูลลำดับ

- ข้อมูลลำดับและการจัดรูปเป็น Window สำหรับฝึกโมเดล
- RNN และปัญหา Vanishing Gradient
- LSTM และ GRU แก้ปัญหาความจำระยะยาวอย่างไร
- พยากรณ์ข้อมูลอนุกรมเวลา เช่น ยอดขายรายวัน พร้อมวัดผลกับ Baseline

### 12 Transformer และกลไก Attention

- ข้อจำกัดของ RNN ที่นำไปสู่ Attention
- Self-Attention, Multi-Head Attention และ Positional Encoding
- Encoder แบบ BERT เทียบกับ Decoder แบบ GPT
- Tokenization, Embedding และการตัดคำของภาษาไทย
- Lab: สร้าง Transformer Block ขนาดเล็กด้วย PyTorch

### 13 Lab: วิเคราะห์ความรู้สึกด้วย Hugging Face

- ใช้ pipeline วิเคราะห์ความรู้สึกจากโมเดลบน Hugging Face Hub
- เตรียมข้อมูลรีวิวภาษาไทยด้วยไลบรารี Datasets และ Tokenizer
- Fine-tune โมเดลภาษาไทยอย่าง WangchanBERTa ด้วย Trainer
- วัดผลด้วย F1-score และวิเคราะห์ประโยคที่โมเดลทายผิด
- เมื่อไรควร Fine-tune โมเดลเล็ก และเมื่อไรใช้ LLM ผ่าน API

### 14 Lab: นำโมเดลไปใช้งาน

- Export โมเดล PyTorch และ Keras เป็น ONNX แล้วรันด้วย ONNX Runtime
- ลดขนาดและเร่งความเร็วด้วย Quantization เบื้องต้น
- สร้างหน้าเดโมด้วย Gradio และแชร์ผ่าน Hugging Face Spaces
- ข้อควรระวังเรื่อง Bias ข้อมูลส่วนบุคคล PDPA และลิขสิทธิ์ของข้อมูลฝึก

### 15 Workshop: Capstone Project

- เลือกโจทย์ภาพ ข้อความ หรืออนุกรมเวลา และกำหนดตัวชี้วัด
- เลือกสถาปัตยกรรมหรือโมเดลสำเร็จรูปที่เหมาะกับข้อมูล
- ฝึก ปรับจูน และเปรียบเทียบกับ Baseline
- Export โมเดล ทำหน้าเดโม และนำเสนอผลงานพร้อมข้อจำกัด

## 07 เรียนจบแล้วได้อะไร Outcomes

- **เข้าใจหลักการถึงแกน** Backpropagation, Optimizer และ Training Loop
- **ใช้ PyTorch และ Keras 3** สร้าง ฝึก และบันทึกโมเดลได้เอง
- **งานภาพครบวงจร** CNN, Transfer Learning และ YOLO26
- **งานข้อความและลำดับ** LSTM, GRU และ Transformer
- **Fine-tune ภาษาไทยได้** Hugging Face Transformers และ Trainer
- **พร้อมนำไปใช้** ONNX, Quantization และหน้าเดโม Gradio

## 08 สิ่งที่ผู้เรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมฉบับสมบูรณ์ครบทั้ง 3 วัน และ Lab Guide ทีละขั้นตอน
- Jupyter Notebook ที่รันบน Google Colab ได้ครบทุก Lab พร้อมโค้ด PyTorch และ Keras
- ชุดข้อมูลตัวอย่าง เทมเพลตชุดข้อมูล YOLO และเทมเพลต Capstone
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

## 09 ข้อมูลผู้สอน Instructor

**อ.สนิทวงศ์ กมลภากรณ์** ผู้สอนและนักพัฒนาที่สร้างโมเดล Deep Learning สำหรับงานภาพ ข้อความ และข้อมูลอนุกรมเวลาในงานจริง จุดเด่นคือการอธิบายสิ่งที่เกิดขึ้นภายในโมเดลด้วยภาพและโค้ดที่เขียนเองก่อนใช้เครื่องมือสำเร็จรูป และพาผู้เรียนวัดผลเทียบกับ Baseline ทุกครั้ง เพื่อให้กลับไปเลือกสถาปัตยกรรม ฝึก และนำโมเดลไปใช้ในงานของตนเองได้อย่างมั่นใจ

ราคาค่าอบรม 13,500 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

**13,500 บาท** / ท่าน (ยังไม่รวม VAT 7%)

สมัครผ่านเว็บไซต์ ลด 10%

02-570-8449
088-807-9770
@itgenius
contact@itgenius.co.th

เพิ่มเพื่อน LINE