Clase 146 — CLIP, SigLIP: multimodal embeddings (visión + texto)

Parte: 2 — Deep Learning · Fuente: Radford et al. (2021) CLIP + Zhai et al. (2023) SigLIP.

Duración estimada: 80 min.

Clase 146 — CLIP, SigLIP: multimodal embeddings (visión + texto)

> *Parte: 2 — Deep Learning · Fuente: Radford et al. (2021) CLIP + Zhai et al. (2023) SigLIP. Duración estimada: 80 min.*

Objetivo

Conocer CLIP (OpenAI 2021) y su evolución SigLIP (Google 2023) — los foundation models que mapean imágenes y texto al mismo espacio vectorial, entrenados con contrastive learning sobre 400M-4B pares. Aplicaciones: zero-shot classification, image search por texto, content moderation, embeddings para RAG multimodal.

Resultados de aprendizaje

Al finalizar, el estudiante podrá:

- Cargar CLIP/SigLIP desde HuggingFace: CLIPModel.from_pretrained('openai/clip-vit-base-patch32').
- Calcular embeddings de imágenes y de texto; cosine similarity entre ambos.
- Implementar zero-shot classification: predecir clase con la mayor similaridad a "a photo of a [class]".
- Hacer image retrieval por texto sobre un corpus de imágenes.
- Diferenciar CLIP (softmax contrastive) de SigLIP (sigmoid pairwise, mejor escalabilidad).

Temas

- Contrastive Language-Image Pre-training: matchear pares correctos, separar incorrectos.
- Arquitectura: image encoder (ViT) + text encoder (Transformer).
- Cosine similarity como métrica.
- Zero-shot vs few-shot.
- Variantes modernas: SigLIP (sigmoid loss), EVA-CLIP, OpenCLIP, Apple AIM.

Definiciones y características

- CLIP: dual encoder, contrastive softmax.
- SigLIP: dual encoder, sigmoid pairwise — escala mejor, no necesita global batch.
- Embedding space: cosine similarity para comparar.
- Zero-shot: clasificar sin training en la tarea, solo con prompt textual.
- Variantes: ViT-B/32 (rápido), ViT-L/14 (mejor), ViT-H/14 (top).

Dataset / recursos

- Imágenes propias o tfds.load('cats_vs_dogs').
- HuggingFace: openai/clip-vit-base-patch32, google/siglip-base-patch16-224.
- Librerías: transformers, torch, PIL.

Ejercicios

1. CLIP setup: cargar processor + model. Embed una imagen y un texto. Cosine similarity.
2. Zero-shot classification: dada una imagen, comparar contra ["a photo of a cat", "a photo of a dog"]. Predict argmax.
3. Image search: corpus de 100 imágenes; query texto "a sunset over the ocean" → top-5 más similares.
4. SigLIP: misma tarea, comparar accuracy.
5. Fine-tune ligero: con dataset chico custom, fine-tunear CLIP con LoRA para un dominio específico.

Homework verificable

Buscador de imágenes por texto:

1. 200 imágenes diversas.
2. Embed todas con CLIP ViT-B/32.
3. UI simple (notebook): input texto → top-5 imágenes.
4. Probar 10 queries; reportar precisión subjetiva.

Criterio de aceptación: queries claras devuelven imágenes relevantes en top-3 con frecuencia alta.

Errores comunes

Síntoma / mensaje	Causa y cómo arreglar
Similitud baja para imagen y texto que par	Prompt mal estructurado. Fix: "a photo of
Lento en CPU	Modelos grandes. Fix: ViT-B/32 (chico) o G
OOM con ViT-L/14	Fix: ViT-B/32 o batch=1.
Embeddings no normalizados → cosine raro	Fix: normalizar con outputs.image_embeds /
Imágenes en formato wrong	CLIP espera RGB 224×224. Fix: usar el proc

Preguntas frecuentes

CLIP o SigLIP?

SigLIP entrena más rápido y escala mejor; calidad similar. En 2026, SigLIP / SigLIP-2 es default moderno.

¿En LLM multimodal (LLaVA)?

CLIP/SigLIP es el image encoder. La LLM (Llama) recibe los embeddings.

Open vocabulary detection con CLIP?

OWL-ViT, Grounding DINO usan CLIP como base. Hacen detección con prompts texto.

CLIP en producción de búsqueda?

Sí — Pinterest, Adobe, Shopify. Vector DB (Qdrant, Pinecone) con embeddings CLIP.

Train CLIP desde cero?

400M+ pares necesarios. Open-source: OpenCLIP, MetaCLIP — alternativas abiertas con datasets públicos.

Referencias

- Radford et al. (2021), CLIP, ICML.
- Zhai et al. (2023), Sigmoid Loss for Language Image Pre-Training (SigLIP), ICCV.
- OpenCLIP.
- LAION-5B — dataset abierto.

Siguiente clase

Clase 147 — Whisper: ASR, transcripción, traducción de audio

Apéndice: notebook (primer bloque)

CLIP entrena texto + imagen para mapear ambos al mismo espacio (cosine similarity). Habilita zero-shot classification y image search.

```
USE_ST = False
try:
    from sentence_transformers import SentenceTransformer
    USE_ST = True
    print('sentence_transformers disponible')
except Exception as e:
    print('ST no disponible. Fallback embeddings random tagged. Motivo:', type(e).__name__)

import numpy as np
import matplotlib.pyplot as plt
np.random.seed(42)
```

Archivos complementarios

- notebook.ipynb