Self-Supervised Learning

Self-Supervised Learning

1. Big Picture

- In **Supervised Learning**, you need **labeled data** (e.g., images + their labels like "cat," "dog").
- In **Unsupervised Learning**, you have **no labels**, just raw data, and you try to find patterns.
- **Self-Supervised Learning** is **in between**:

  → It **creates its own labels** from the raw data.

  → The model learns from the **structure of the data itself**, without needing manual labels.

---

2. How It Works

- The model sets up a **pretext task**: a fake or proxy problem it can solve using only the input data.
- Solving this task forces the model to learn useful **representations/features**.
- Later, those learned features can be **fine-tuned** for actual tasks (like classification, detection, NLP tasks).

---

3. Examples of Pretext Tasks

🔷 **In NLP (Natural Language Processing):**

- **Masked Language Modeling (MLM):** Hide some words in a sentence and train the model to predict them.
  - Example: "The cat sat on the ___" → Model learns "mat."
- This is how **BERT** was trained.

🔷 **In Computer Vision:**

- **Image Inpainting:** Hide a patch of an image and predict the missing part.
- **Rotation Prediction:** Rotate an image randomly (0°, 90°, 180°, 270°) and make the model predict the angle.
- **Contrastive Learning (SimCLR, BYOL):** Show two augmented views of the same image and force the model to recognize them as the same.

---

4. Why It's Useful

- **Less Labeling Effort:** No need for expensive human-annotated data.
- **Scales Easily:** You can use tons of unlabeled data (text, images, audio).
- **Better Features:** The model learns **general, transferable features** useful across many tasks.
- **State-of-the-art:** Most modern foundation models (e.g., GPT, BERT, CLIP, SimCLR, DINO) rely on SSL.

---

5. Analogy (ELI5)

Think of it like a **puzzle book**:

- You don't need a teacher giving you the answer.
- The puzzle itself forces you to **think and learn patterns**.
- Later, the skills you learned (logic, reasoning, pattern recognition) can be applied to real-world problems.

✅ **In summary:**

Self-Supervised Learning = Using the data itself to **generate supervision signals**, training models without manual labels, and producing strong general-purpose representations.

The Core Problem

Training deep learning models usually requires **huge labeled datasets**.

- In **supervised learning**, you need millions of labeled examples (e.g., ImageNet, medical images, speech transcriptions).
- But **labels are expensive and time-consuming** to get:
  - Doctors' labeling X-rays → costly
  - Humans annotating billions of images/texts → unrealistic
  - Some domains (e.g., rare diseases, satellite data) → very few labels exist

Meanwhile, the world has **abundant raw, unlabeled data**:

- billions of images,
- hours of video/audio,
- massive text corpora.

👉 **Problem:** How do we make use of this massive, unlabeled data efficiently without depending on costly human labeling?

What SSL Does

SSL **turns the problem of "no labels" into "fake labels"** created automatically from the data itself.

- Instead of asking humans, the model **creates supervised tasks** (pretext tasks).
- By solving them, the model **learns useful representations/features** from raw data.

Why This Matters

- **Reduces dependence on labels** → scales up learning.
- **Learns general features** → transferable to many downstream tasks.
- **Bridges the gap** between:
  - **Unsupervised learning** (no guidance at all, just clustering)
  - **Supervised learning** (full manual guidance with labels).

Example

Imagine you want to train a language model:

- Supervised way → Need millions of labeled "input → output" pairs (like translations, summaries).
- SSL way → Just take raw text from the internet, mask a few words ("I went to the ___") and ask the model to predict them.
- The model **teaches itself** language patterns → later, you can fine-tune it for tasks like question answering or sentiment analysis.

✅ **In short:**

Self-Supervised Learning is trying to solve the problem of the **scarcity of labeled data** and **costly human annotation**, while still enabling models to learn powerful, general-purpose features from the **abundant unlabeled data** we already have.

Unsupervised vs. Self-Supervised Learning

Aspect	**Unsupervised Learning**	**Self-Supervised Learning**
Input	Only raw, unlabeled data	Only raw, unlabeled data
Labels	No labels at all, no artificial labels created	Labels are **automatically generated from data** (pretext tasks)
Goal	Discover hidden structure or grouping in the data	Learn **representations/features** useful for later tasks
Output	Patterns, clusters, compressed data, embeddings	A model trained with useful features that can be **fine-tuned** for supervised tasks
Main Question	*"What structure exists in this unlabeled data?"*	*"Can I invent a supervised task from this unlabeled data to learn good features?"*

🔑 Intuition

- **Unsupervised learning** is like exploring a room full of objects without instructions:

→ You group similar things (clustering), or summarize the room in fewer dimensions (PCA).

- **Self-supervised learning** is like giving yourself puzzles in that room:

  → “Cover half the puzzle and guess the missing piece,” or “Rotate this photo and figure out the angle.”

  → By solving these puzzles, you learn to understand the room better.

✅ Core Difference

- **Unsupervised:** *No supervision at all* → purely structure discovery.
- **Self-Supervised:** *Creates its own supervision from raw data* → representation learning.

👉 So you can think of **Self-Supervised Learning as a special subclass of Unsupervised Learning** that makes unlabeled data act like labeled data.

Techniques of Self-Supervised Learning (SSL)

🔷 1. Pretext Task–Based Methods

These create an **artificial supervised task** from unlabeled data. The model learns by solving these tasks.

In NLP (text)

- **Masked Language Modeling (MLM):** Mask words and predict them. (e.g., BERT)
  - “The cat sat on the ___” → predict "mat."
- **Next Sentence Prediction (NSP):** Predict whether one sentence follows another. (BERT pre-training)
- **Autoregressive Prediction:** Predict the next word in a sequence. (GPT)

In Vision (images)

- **Image Inpainting:** Hide part of the image and predict the missing region.
- **Colorization:** Convert grayscale → color.
- **Rotation Prediction:** Rotate an image by {0°, 90°, 180°, 270°} and ask the model to predict the rotation.
- **Jigsaw Puzzle:** Shuffle image patches and predict the correct order.

🔷 2. Contrastive Learning Methods

Instead of predicting missing parts, these learn **representations by comparing data**.

- **Core idea:**
    - Generate two different views of the same input (through augmentations).
    - Bring their embeddings **closer** in latent space.
    - Push embeddings of different inputs **apart**.

Popular Methods

- **SimCLR (Simple Contrastive Learning of Representations):**
    - Uses data augmentations (crop, color distortions, flip).
    - Positive pair = two views of same image; Negative pair = different images.
- **MoCo (Momentum Contrast):**
    - Uses a memory bank (queue) to store negative examples for stable training.
- **BYOL (Bootstrap Your Own Latent):**
    - Removes explicit negatives!
    - Uses two networks (online & target) that bootstrap each other.
- **SimSiam:**

    - Even simpler: no negative pairs, only positive pairs with the stop-gradient trick.

🔷 3. Generative SSL

Here, the model learns by **reconstructing data**.

- **Autoencoders:** Encode → decode → reconstruct original input.
- **Variational Autoencoders (VAEs):** Learn probabilistic latent variables.
- **Masked Autoencoders (MAE):** Mask large parts of image, train a transformer to reconstruct (very effective in vision).
- **GPT-style Transformers:** Predict next token (autoregressive generation).

🔷 4. Cross-Modal SSL

Leverage relationships between **different modalities** (text, image, audio).

- **CLIP (OpenAI):**
    - Train on image + text pairs.
    - Learn to align vision embeddings with language embeddings.
- **Video-Audio Models:** Predict if a sound matches a video.

✅ Summary

SSL techniques can be grouped into:

1. **Pretext tasks** (masking, rotation, jigsaw).
2. **Contrastive learning** (SimCLR, BYOL, MoCo).
3. **Generative methods** (Autoencoders, GPT, MAE).
4. **Cross-modal methods** (CLIP).

👉 The choice depends on domain:

- Text → masking/next-word prediction.
- Vision → contrastive or masked autoencoders.

- Multi-modal → contrastive alignment like CLIP.
