## GÁN NHÃN DỮ LIỆU (DATA LABELING)

### Định nghĩa

* **Gán nhãn dữ liệu** là quá trình gắn nhãn (tag) hoặc gán nhãn phân lớp cho từng mẫu dữ liệu (văn bản, ảnh, âm thanh, video, v.v.) để tạo tập huấn luyện cho các mô hình học có giám sát. Nhãn cung cấp “đáp án” để thuật toán học mối quan hệ giữa đầu vào và đầu ra.

### Tầm quan trọng

* Là nền tảng cho supervised learning: mô hình chất lượng cao cần dữ liệu gán nhãn chính xác để học.
* Ảnh hưởng trực tiếp tới độ chính xác, độ tin cậy và công bằng của hệ thống AI; nhãn kém hoặc thiên lệch dẫn tới model kém hoặc có thành kiến.

### Các phương pháp gán nhãn phổ biến

* **Gán nhãn thủ công (Manual labeling)**: con người đọc/gán nhãn từng mẫu — độ chính xác cao nhưng tốn thời gian, chi phí.
* **Crowdsourcing**: chia nhỏ tác vụ cho nhiều annotator (Mechanical Turk, v.v.) — rẻ hơn nhưng cần QC/aggregation (majority vote, adjudication).
* **Semi-automated / ML-assisted labeling**: dùng mô hình sơ bộ gợi ý nhãn, con người chỉ hiệu chỉnh — tăng tốc đáng kể.
* **Active learning**: mô hình chọn những mẫu “khó” hoặc lợi ích lớn để con người gán nhãn trước, giảm số lượng mẫu cần gán.

### Các vấn đề chung khi gán nhãn

* **Không đồng nhất / hướng dẫn annotator không rõ** → kết quả không ổn định.
* **Sự mơ hồ ngôn ngữ** (không rõ ranh giới entity, nhãn chồng chéo).
* **Bias (thành kiến)** do tập dữ liệu hoặc annotator.
* **Chi phí & thời gian** khi mở rộng quy mô; cần pipeline QC (sampling, adjudication).
* **Bảo mật / quyền riêng tư** khi dữ liệu nhạy cảm (y tế, tài chính) — yêu cầu anonymization & hợp đồng NDA.

### Các nền tảng gán nhãn (mới nhất / phổ biến)

* **Labelbox** — end-to-end labeling + model-assisted workflows.
* **Label Studio** — open-source, rất linh hoạt cho text/vision/audio.
* **Amazon SageMaker Ground Truth** — tích hợp mạnh với AWS, có workforce và automation.
* **Scale AI, SuperAnnotate, Prodigy** — mỗi nền tảng có thế mạnh (scale, UI annotation, research workflow nhanh).

### Phân tích kết quả & đánh giá chất lượng nhãn

* Metric/Quy trình thường dùng:
* **Inter-annotator agreement (IAA)**: đo độ đồng nhất giữa annotator — giúp đánh giá tính rõ ràng của guideline. Nếu IAA thấp, cần sửa guideline hoặc đào tạo annotator.
* **Sampling + Adjudication**: lấy mẫu ngẫu nhiên từ dataset đã gán nhãn để kiểm tra thủ công (adjudicator quyết định nhãn “chính thức”).
* **Ảnh hưởng tới model**: xây một baseline model trên dữ liệu đã gán nhãn; dùng precision/recall/F1, confusion matrix để phát hiện các lớp bị gán sai nhiều. So sánh performance trước/sau clean nhãn để ước tính tác động của noise.
* Ảnh hưởng của label noise & cách giảm thiểu:
* Label noise có thể làm suy giảm hiệu năng model, đặc biệt với noise có liên hệ tới đặc trưng đầu vào (feature-dependent noise). Có nhiều nghiên cứu cho thấy tỉ lệ noise cao làm giảm tính khái quát hóa và một số mô hình/chiến lược (noise-robust loss, data cleaning, ensembles, pretraining-then-cleaning) có thể giảm thiểu.

## RÚT TRÍ TÊN THỰC THỂ (NAMED ENTITY RECOGNITION — NER)

### NER — khái quát

* **Named Entity Recognition** là bài toán phát hiện và phân loại các thực thể (ví dụ Person, Organization, Location, Date, Money, v.v.) trong văn bản. NER là bài toán sequence labeling (gán nhãn cho từng token/chuỗi token).

### Hướng 1 — Rule-based

* Mô tả
* Dùng **tập quy tắc** (regular expressions, mẫu ngữ pháp), **gazetteers** (danh sách tên riêng: địa danh, tên công ty), và heuristic (ví dụ: chữ in hoa, tiền tố/suffix).
* Thường phù hợp khi domain cố định, dữ liệu ít thay đổi, hoặc khi yêu cầu giải thích được (explainable).
* Ưu/nhược
* **Ưu**: dễ triển khai nhanh, giải thích rõ ràng, không cần nhiều dữ liệu gán nhãn; hiệu quả trong domain nhỏ.
* **Nhược**: kém mở rộng, dễ vỡ khi ngôn ngữ/biểu đạt thay đổi, tốn công duy trì quy tắc khi domain phức tạp.
* Ví dụ minh hoạ (Python — rule-based đơn giản)

A screenshot of a computer program

AI-generated content may be incorrect.

* *Ghi chú:* phương pháp này yêu cầu chuẩn hoá/điều chỉnh nhiều với tiếng Việt (chú ý viết hoa, dấu câu, tên địa phương viết tắt, v.v.).

### Hướng 2 — Statistical learning (classical → deep learning → Transformers)

* Mô tả & tiến hoá
* **Classical**: dựa trên đặc trưng thủ công (word shape, gazetteer). CRF từng là công nghệ tiên tiến cho NER.
* **Deep learning**: BiLSTM với embedding (word + character) — mô hình tự trích đặc trưng, đạt hiệu quả cao hơn mà ít cần feature engineering.
* **Transformer / Pretrained LM**: BERT, RoBERTa, các mô hình lớn fine-tuned cho NER — hiện tại đạt hiệu năng tốt nhất trên nhiều bộ dữ liệu; dễ fine-tune với vài chục–vài trăm ví dụ nếu dùng transfer learning. Gần đây LLMs/transformer-based models & graph-based enhacements đang được nghiên cứu rộng rãi.
* Ưu/nhược
* **Ưu**: khái quát hóa tốt, ít phụ thuộc feature engineering, hiệu năng cao (đặc biệt với pre-trained transformer).
* **Nhược**: cần dữ liệu gán nhãn (hoặc fine-tune), tài nguyên tính toán lớn hơn, giải thích kém hơn rule-based.
* Đánh giá cho NER
* **Entity-level Precision / Recall / F1** là metric chuẩn. Có thể tính micro/F1 hoặc macro/F1 tùy mục tiêu.

### Minh hoạ: Dùng spaCy (statistical / transformer) để extract NER

* spaCy là thư viện NLP phổ biến, dễ dùng cho demo và sản xuất
* Ví dụ code (Python, spaCy)

A computer screen shot of a code

AI-generated content may be incorrect.

* *Ghi chú:* để dùng cho tiếng Việt, có thể dùng model pre-trained phù hợp (nếu có) hoặc fine-tune model multilingual/transformer; spaCy cung cấp pipeline và hướng dẫn fine-tune.

### So sánh ngắn gọn (Rule-based vs Statistical)

* **Chi phí dữ liệu**: Rule-based ít cần nhãn, Statistical cần dataset gán nhãn.
* **Mức độ robust (mạnh )**: Statistical (đặc biệt transformer) mạnh mẽ hơn với biến thể ngôn ngữ chung; rule-based tốt nếu domain cố định.
* **Explainability & maintainability**: Rule-based dễ giải thích; statistical khó giải thích nhưng dễ mở rộng.