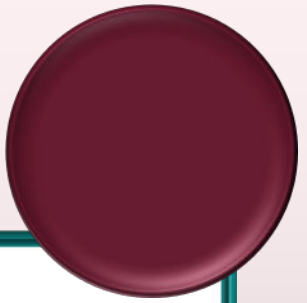


An Introduction of Analyzing Unstructured Data in Education



Learning Objectives

1

Learners will be able to define and identify unstructured data commonly encountered in educational contexts.

2

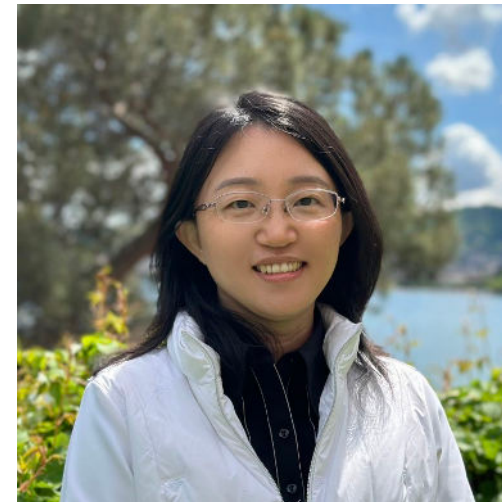
Learners will be able to describe the methods and tools used to preprocess, represent, analyze, and evaluate unstructured data.

3

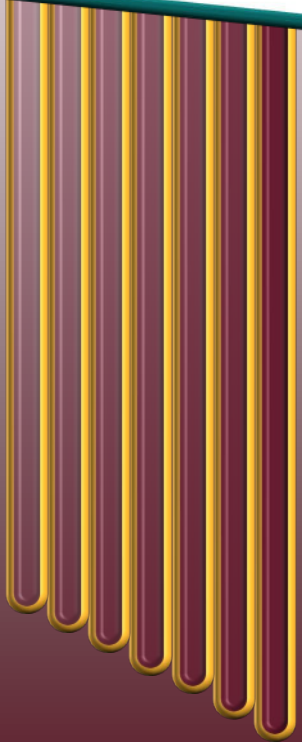
Learners will be able to explain key considerations of data privacy, sensitivity, and access when working with unstructured educational data.

About the Author

- Mo Zhang:
 - Researcher with 15+ years of experience
 - PhD in Educational Psychology
 - Currently Principal Research Scientist at Educational Testing Service
- Mengxiao Zhu
 - Extensive multidisciplinary research background
 - PhD in Industrial Engineering
 - Currently Professor at University of Science and Technology of China

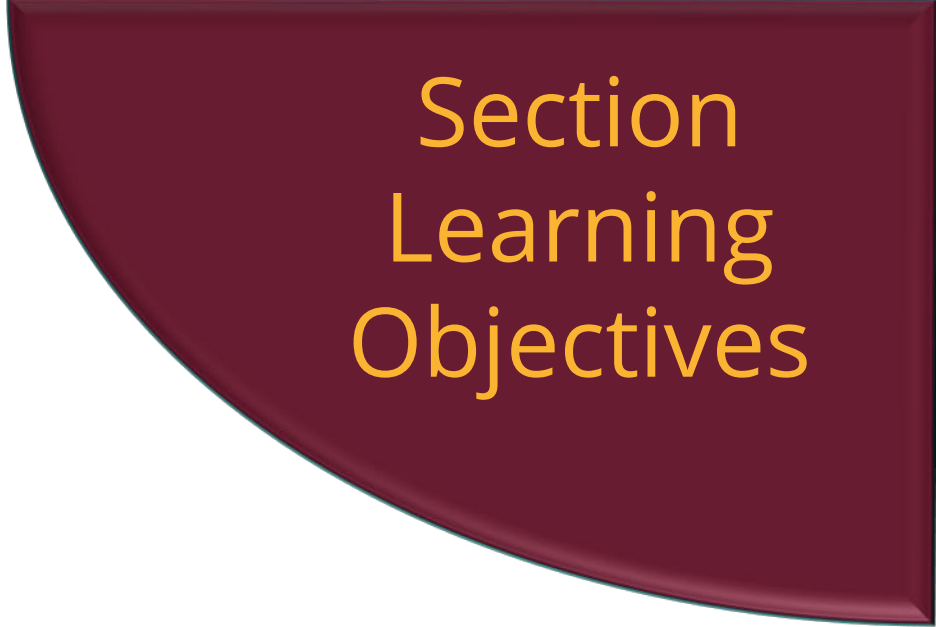


Section 1: Introduction

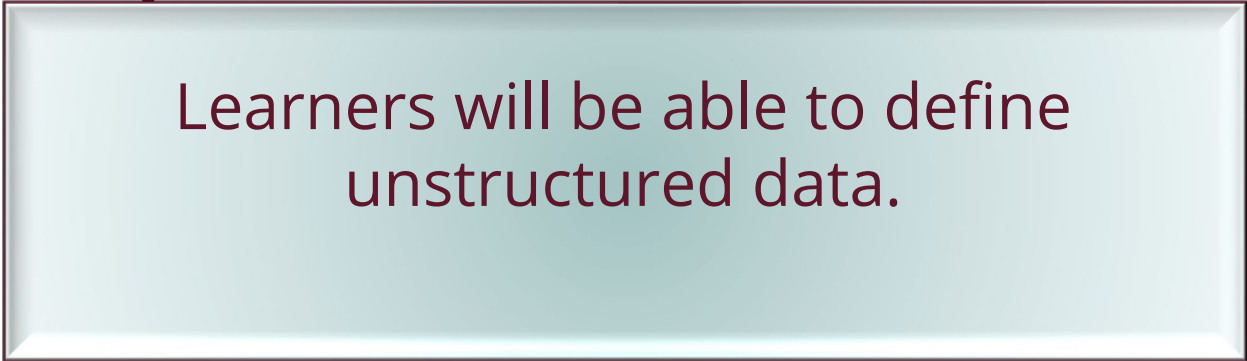




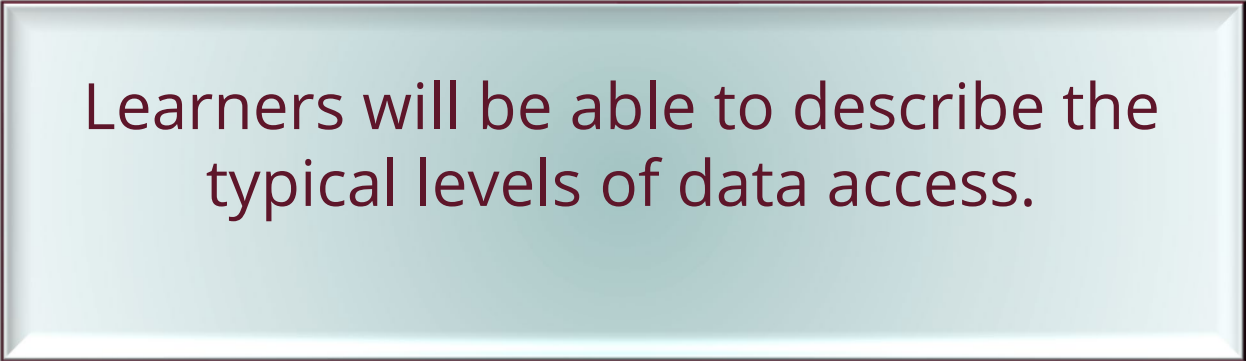
Section1: Introduction



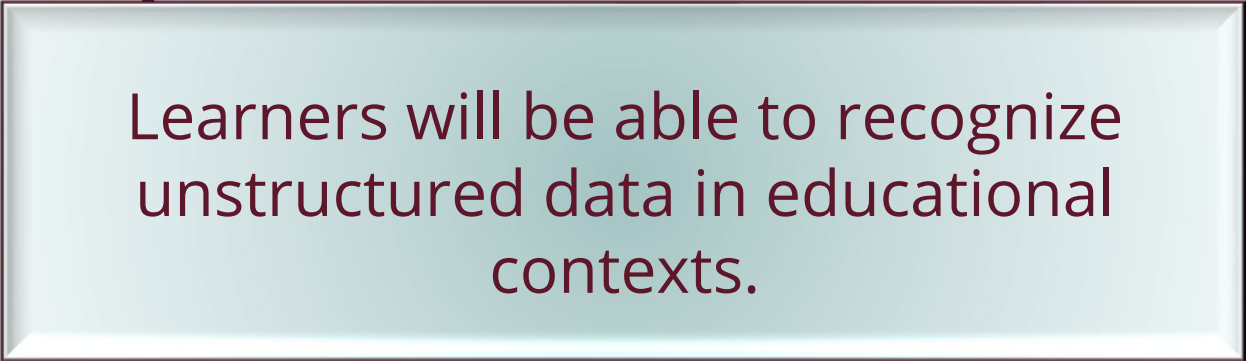
Section Learning Objectives



Learners will be able to define
unstructured data.



Learners will be able to describe the
typical levels of data access.



Learners will be able to recognize
unstructured data in educational
contexts.

What is Unstructured Data?

- Data without a predefined schema or fixed fields, where meaning only emerges from neighboring data points.
- **Common modalities:** text, images (e.g., JPG), audio (e.g., WAV), video (e.g., MP4), document scans (such as PDFs), social media posts, process logs and click streams.
- Unstructured data will require pre-processing and representation before measurement model can be applied.



A Simple Example of Structured Data

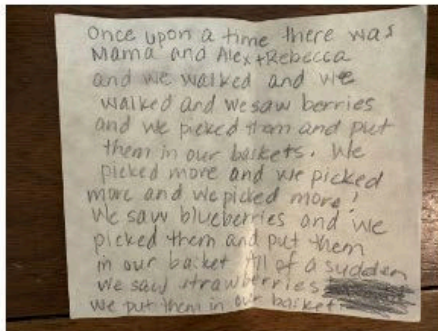
An Item Bank Database:

Item ID	domain	Difficulty b	Discrimination a	format
A11	Algebra	-0.42	1.13	MC
A12	Geometry	0.88	0.97	CR
A13	Algebra	0.15	1.42	MC
A14	Statistics	-1.05	0.88	MC

- Each cell carries meaning on its own
- Support sorting, filtering, averaging... various data operations.

Characteristics of Unstructured Data

- **No pre-defined schema:** no fixed rows, columns, or fields.
- **High dimensionality:** even a short essay, image, or audio clip contains thousands to millions of raw values.
- **Meaning emerges from context:** the value of any single unit is meaningless in isolation.



Text



Audio



Image



Video

Context-Dependent Meaning

<i>Text</i> one letter → no inference; a sentence → a message	<i>Image</i> One pixel → uninformative; Thousands of pixels → a shape
<i>Audio</i> one amplitude → silence; many amplitudes → a phoneme	<i>Clickstream</i> one click → uninformative; a click sequence → a strategy

Therefore, making inferences from unstructured data will require (sophisticated) models that capture sequence, structure, or spatial relations of the data.

Structured vs. Semi-Structured vs. Unstructured

Aspect	Structured	Semi-Structured	Unstructured
Common formats	Tabular, tensor, time series, graphs	JSON, XML, YAML	Text, audio, video, images
Single value	Stands alone	Stands alone via key	Context-dependent
Common storage	CSV, Excel, SQL (relational database)	NoSQL, JSON files	Data lakes, file systems
Schema	Fixed, predefined	Flexible, self-describing	None

A JSON example:

```
{  
  "student_id": 10421,  
  "grade": 8,  
  "gender": "female",  
  "scores": {"math": 78, "reading": 85},  
  "absent_dates": ["2026-04-21", "2026-04-22"],  
  "free_lunch": true  
}
```

Distinction at *Cell* Level

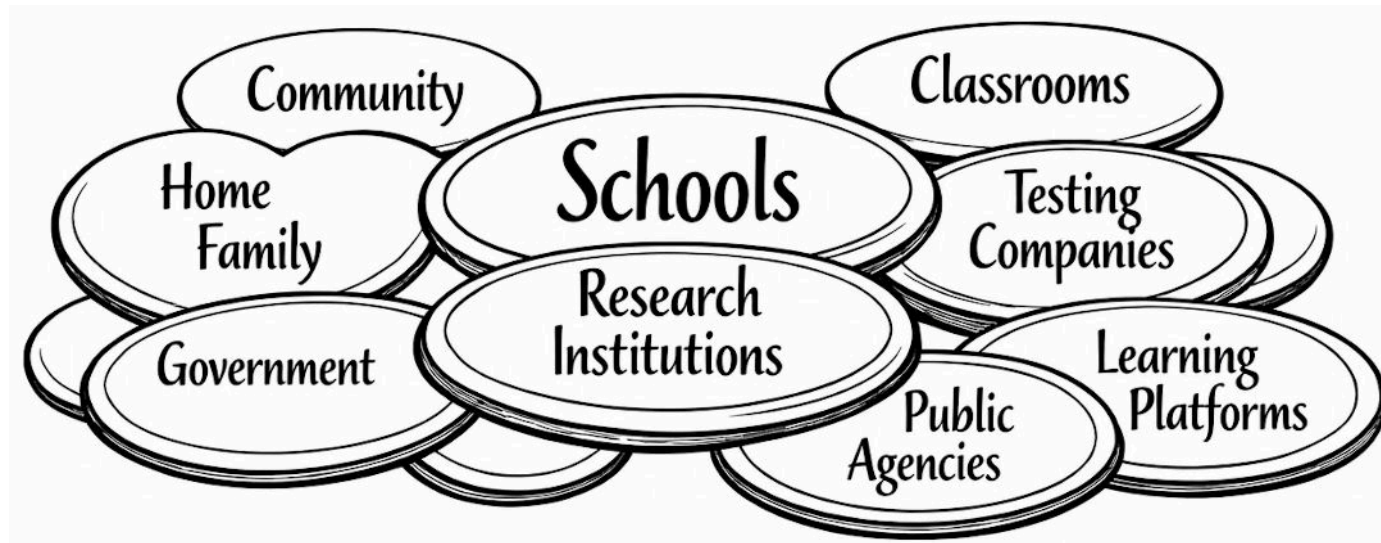
A structured table can carry unstructured content, or value, inside individual cells.

student_id	prompt_id	score	essay_text
s021	P03	4	"Although the author argues that social media harms teenagers, the evidence she provides is mostly anecdotal..."
s022	P03	5	"Social media platforms have transformed how young people communicate, but the research is more nuanced..."
s023	P03	3	"I think social media is bad because..."

- The first three columns are **structured** (sortable, arithmetic operation)
- The `essay_text` column is **unstructured**

Why Unstructured Data Matters in Education?

- Data collection is becoming ubiquitous, and un-intrusive
 - GenAI produces *and* consumes unstructured data
- Providing authentic evidence for teaching, learning, and assessment, capturing a broad range of skills
- Informing policy, public service, and decision-making



Where Unstructured Data Exist? (1/2)

- **Schools and classrooms**

- School: administrative records, policies, incident reports, course materials, parent correspondence, social media accounts
- Teacher: lesson plans, lecture notes and recordings, classroom photos, observation videos
- Student: homework in a variety of formats, discussion-board posts, projects, performance recordings

- **Home and family**

- Informal art projects, observation journals, learning artifacts
- Chat logs with AI tutors
- Wearable and home devices (smart speakers, study apps)
- Photos and videos

- **Government and public agencies**

- Large-scale assessments: NAEP, PISA, TIMSS, PIRLS
- Administrative records, policy documents, public surveys

Where Unstructured Data Exist? (2/2)

- **Testing companies and LMS platforms**

- Operational data (e.g., ETS GRE, NBME, LSAC, Coursera, Khan Academy, Google Classroom): writing samples, video and audio recordings, art portfolios, clinical notes
- Participation in discussion forums

- **Community and informal settings**

- Museums, libraries, national parks: visitor interactions, program artifacts
- Tutoring services and after-school programs: session recordings, written feedback
- Online communities: study-group chats, education-focused social media content

- **Research and academic institutions**

- Lab and field studies: eye-tracking, think-aloud transcripts, classroom-video corpora
- Open research corpora

Levels of Access

Source	Typical Access	Typical Governance
Schools & classrooms	Restricted	Restricted/IRB ¹ approval; FERPA ²
Home & family	Highly restricted	Parental control; COPPA ³
Government agencies	Restricted/Open	Restricted-use licenses
Testing companies	Proprietary	Client contracts
LMS platforms	Proprietary	ToS ⁴ ; client contracts
Community settings	Variable	Often unregulated
Research institutions	Restricted/Open	IRB approval

¹IRB: Institutional Review Board

²FERPA: Family Educational Rights and Privacy Act (U.S., 1974)

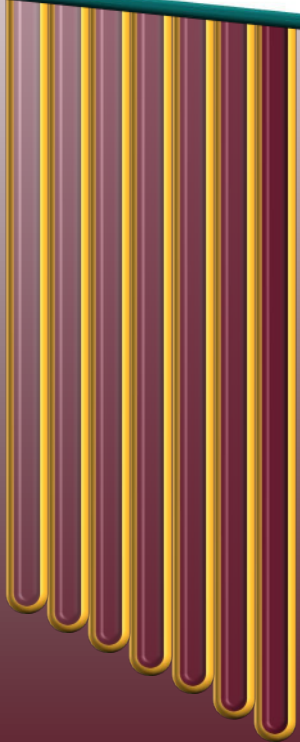
³COPPA: Children's Online Privacy Protection Act (U.S., 1998)

⁴ToS: Terms of Service

What's Covered in Sections 2 to 4

	Sections 2 and 3 (Preprocesssing, Representation)	Section 4 (Analysis, Making inferences)
Focus	Is the data ready?	What can I learn from it?
Inputs	Raw text, audio, images, logs, etc.	Clean data, tokens, features, embeddings
Outputs	Tokens, features, vectors, tensors	Scores, classes, clusters, predictions

Section 2: Treating and Preprocessing Unstructured Data



Section Learning Objectives

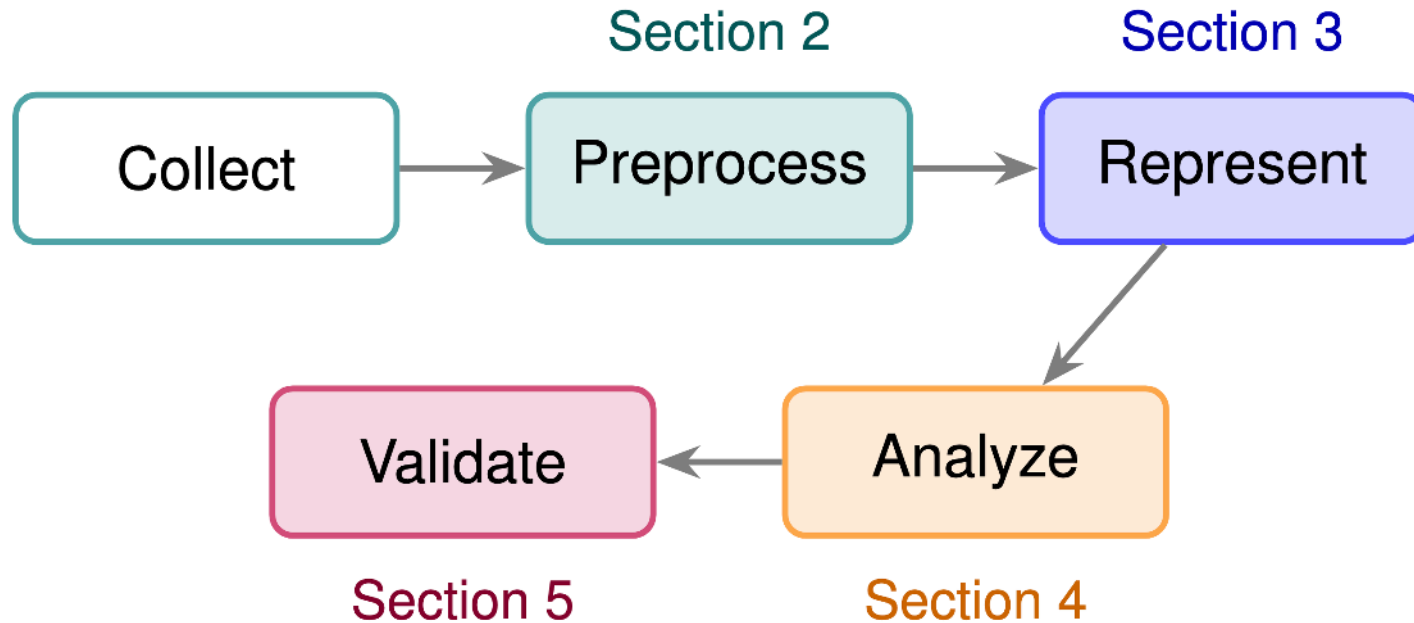
Section 2: Treating and Preprocessing Unstructured Data

Learners will be able to describe how raw unstructured data is cleaned and prepared for modeling.

Learners will be able to identify the main preprocessing steps and explain the purpose of each.

Learners will be able to recognize how preprocessing differs across modalities.

The General Analysis Pipeline



- In Section 2, we will cover **Methods and tools** to preprocess unstructured data, and to generate numeric data representation
- Then, in Section 3, we will cover **considerations for choosing the “right” representation**, including aspects such as tradeoffs in interpretability, cost, and downstream model fit

Sections 2-3 Roadmap

Section 2. Preprocessing

- Text
- Handwriting, drawing, scans
- Audio
- Video
- Process log
- Multimodal and sensor data

Preprocessing: cleaning and converting raw unstructured data into consistent, model-ready units (tokens, segments, signals etc.).

Section 3. Representation

- Static embeddings
- Contextual embeddings
- LLM embeddings
- Image/video features
- Audio features
- Sequence embeddings

Representation: encoding the cleaned data as numeric vectors that models can compute on.

Preprocessing: Raw Texts

Raw XML input:

```
"I'm Running\n quickly to school!! \& I\u2019m late.<br>"
```

Option	Purpose	Example Outcome
Cleaning	Remove markup and encoding artifacts, keep the words.	"I'm Running quickly to school!! & I'm late."
Normalization	Standardize surface form (e.g., lowercase).	"i am running quickly to school!! & i am late."
Tokenization	Split text into discrete units (e.g., words, punctuation, or subwords).	["i", "am", "running", "quickly", "to", "school", "!!", "&", "i", "am", "late", "."]
Filtering	Drop uninformative tokens (e.g., stopwords, punctuation).	["running", "quickly", "school", "late"]
Lemmatization	Reduce each word to its dictionary base form (lemma).	["run", "quickly", "school", "late"]
Sentence segmentation	Divide the text into individual sentences.	[S1: "I'm running quickly to school!", S2: "I'm late."]

Preprocessing Decisions Matter

- **Clean only what you need to.** Aggressive cleaning can erase construct-relevant information.
- **Match preprocessing to the model.** Modern transformer models expect raw, lightly cleaned text; over-processing can hurt downstream performance.
- **Tokenizer** (i.e., a component that splits raw text into the discrete units or tokens) choice **matters**:

	"i am running quickly to school"
Word-level	["i", "am", "running", "quickly", "to", "school"]
Subword	["i", "am", "run", "##ning", "quick", "##ly", "to", "school"]
Character-level	["i", " ", "a", "m", " ", "r", "u", "n", "n", "i", "n", "g", " ", "q", "u", "i", "c", "k", "l", "y", "..."]

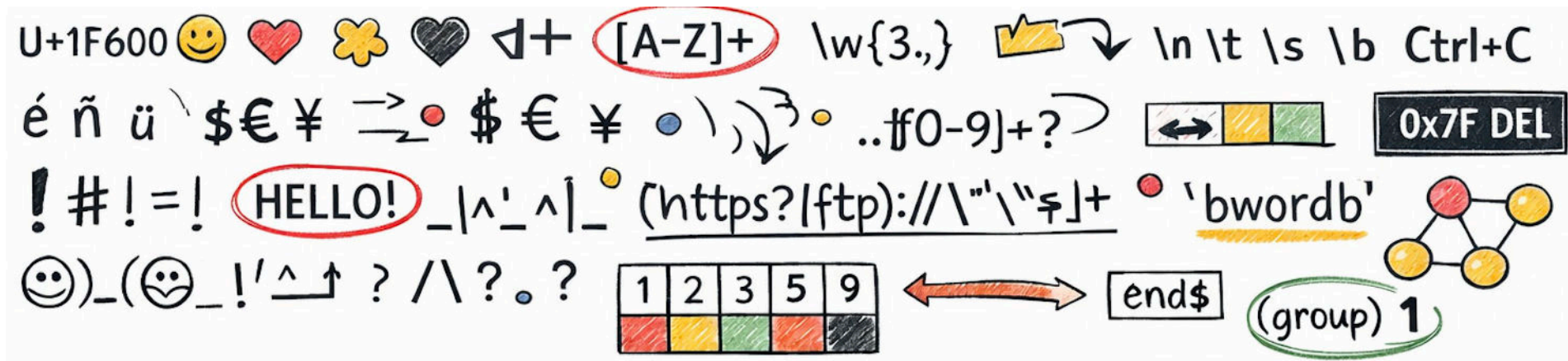
Some Common Tools for Text Processing (1/2)

- **General-purpose Natural Language Processing (NLP) libraries**
 - spaCy: production-ready pipelines: tokenization, lemmatization, POS tagging, sentence segmentation
 - NLTK: includes many tokenizers, stemmers, corpora
 - Stanza (Stanford NLP): multilingual neural NLP pipelines
- **Tokenizers for transformer / LLM models**
 - HuggingFace Tokenizers: BPE, WordPiece, SentencePiece
 - tiktoken: OpenAI's tokenizer for GPT-3.5/4/4o/nano
 - SentencePiece: tokenizer used by LLaMA, mBART

Some Common Tools for Text Processing (2/2)

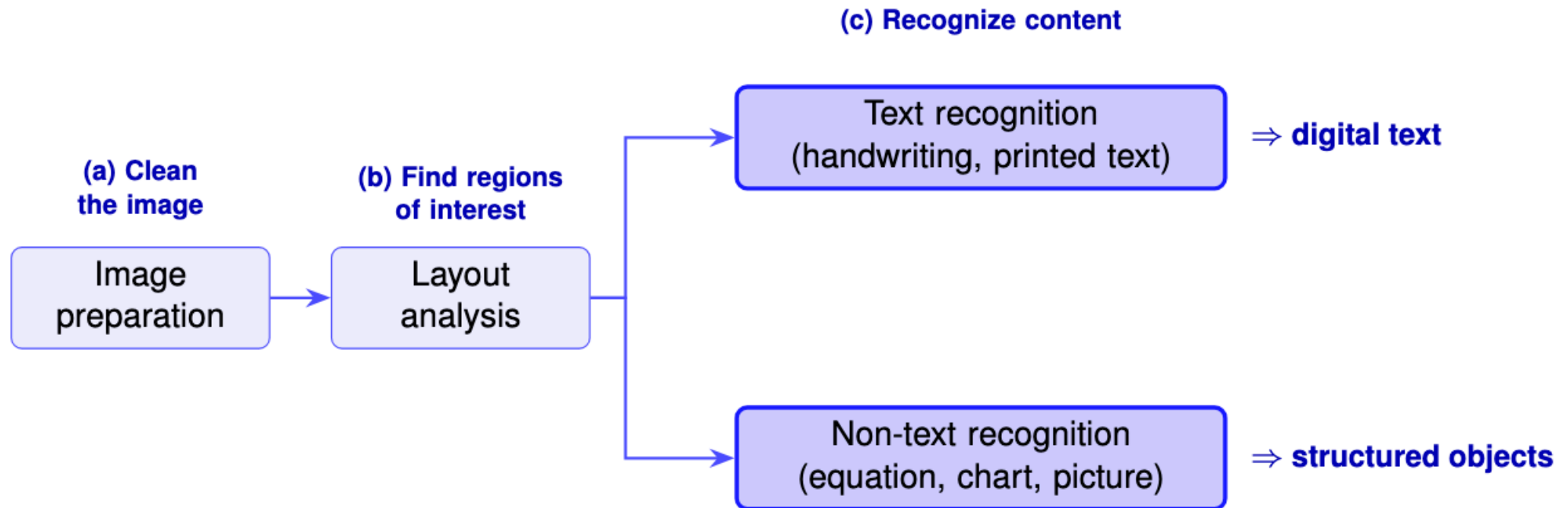
- **Utility tools**

- regex (built-in R, Python, JavaScript etc.): pattern-based cleaning and tokenization
- ftfy: “fixes text for you”, Unicode repair
- Unidecode: transliterate Unicode to ASCII



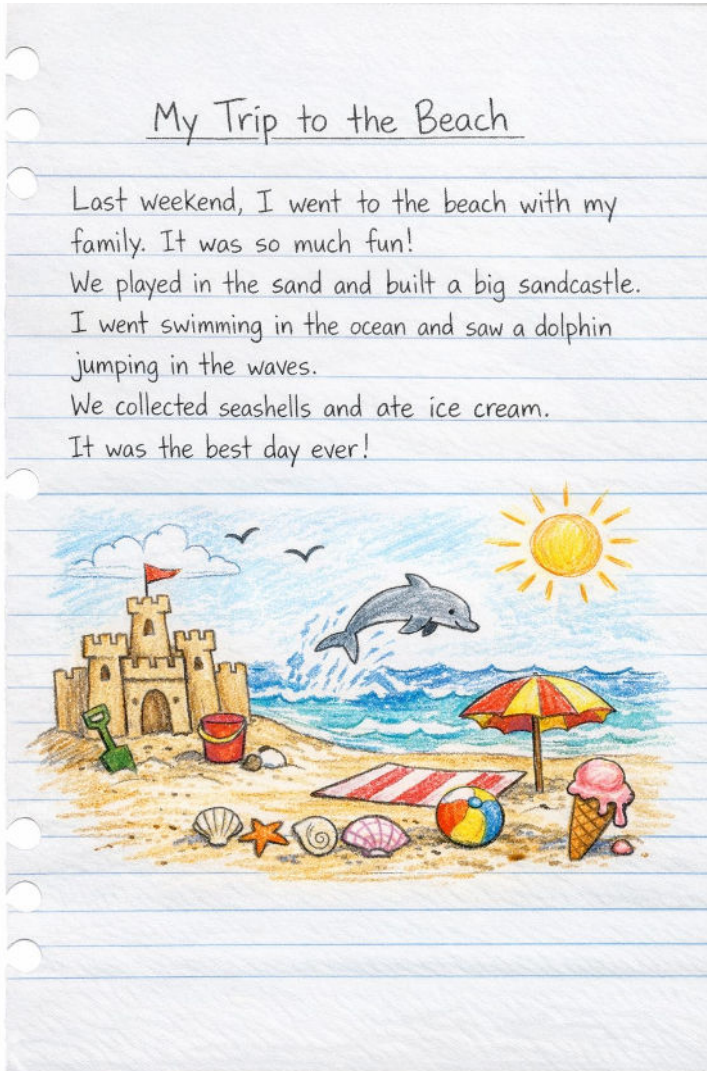
Preprocessing: Handwriting, Drawing, Scans

- Artifacts like scanned essays, answer sheets, lab notes, student drawings, and lecture notes etc. are captured as **images**
- How to get from pixels to content?

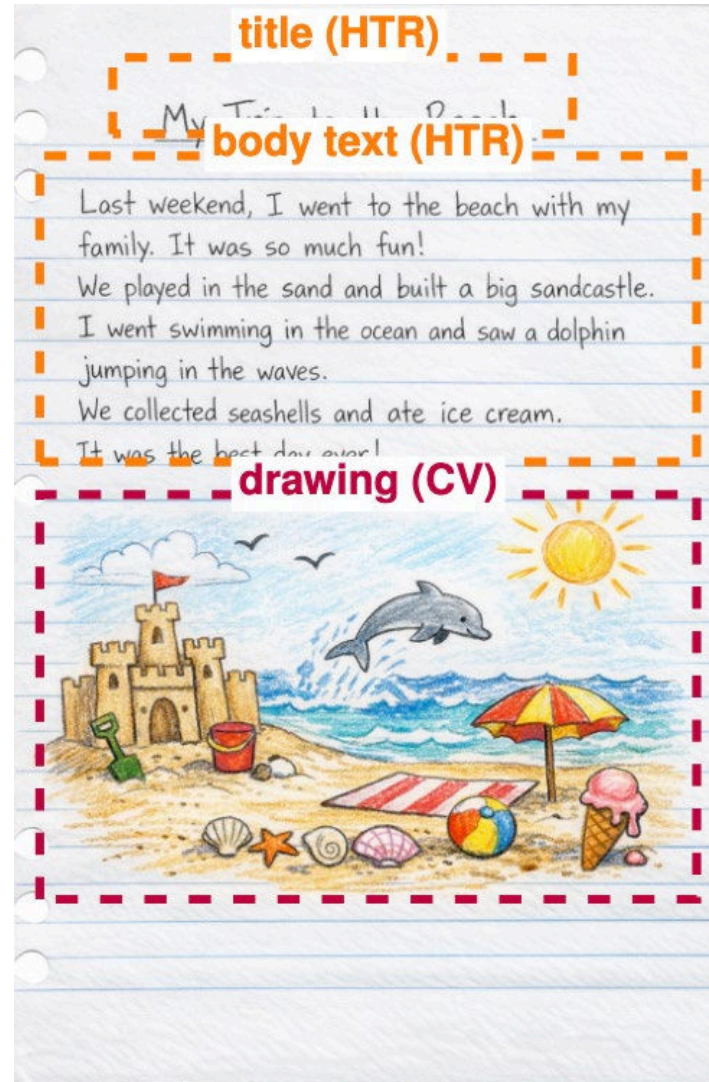


From Pixels to Recognized Content

1. Raw scan



2. Layout analysis



3. Digitized output

HTR¹ ⇒ text

My Trip to the Beach
Last weekend, I went to
the beach...

CV² ⇒ objects

```
{  
  sandcastle, sun,  
  dolphin, towel,  
  seashells,...  
}
```

1. HTR: Handwritten Text Recognition

2. CV: Computer Vision

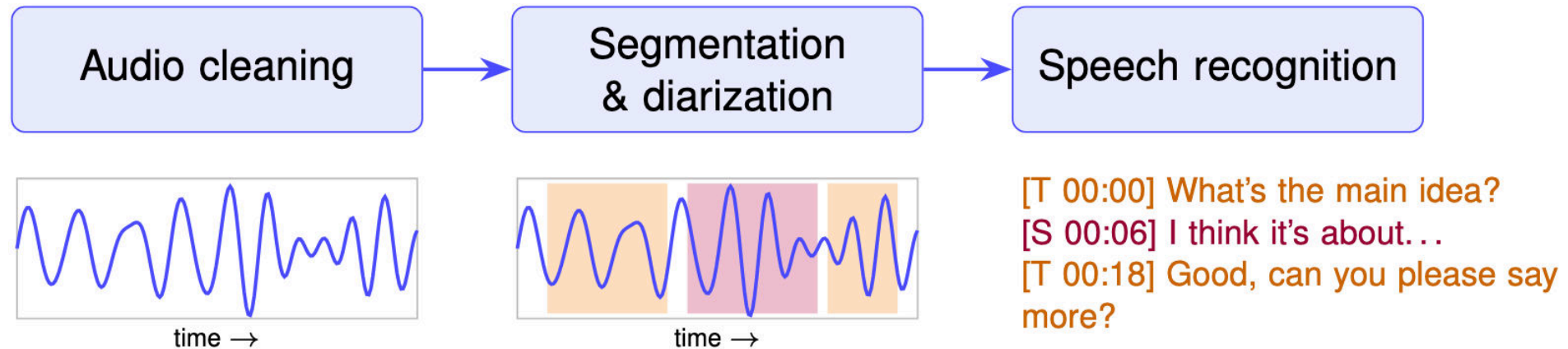
Task and Tools for Image Processing

Task	Purpose	Example Tools
Image preparation	de-skew (rotate page), de-noise (remove specks), normalize contrast, etc.	OpenCV (general-purpose image processing library); Pillow (Python toolkit)
Layout analysis	detect regions	LayoutParser (pretrained models for document layout); Adobe PDF Extract ; Mistral OCR ¹
Text recognition	Recognize printed or handwritten text	Tesseract : open-source OCR; Google Vision , Amazon Textract
Non-text recognition	Recognize equations, diagrams, objects, etc.	Mathpix , OpenAI CLIP (image classification etc.)

¹OCR: Optical Character Recognition

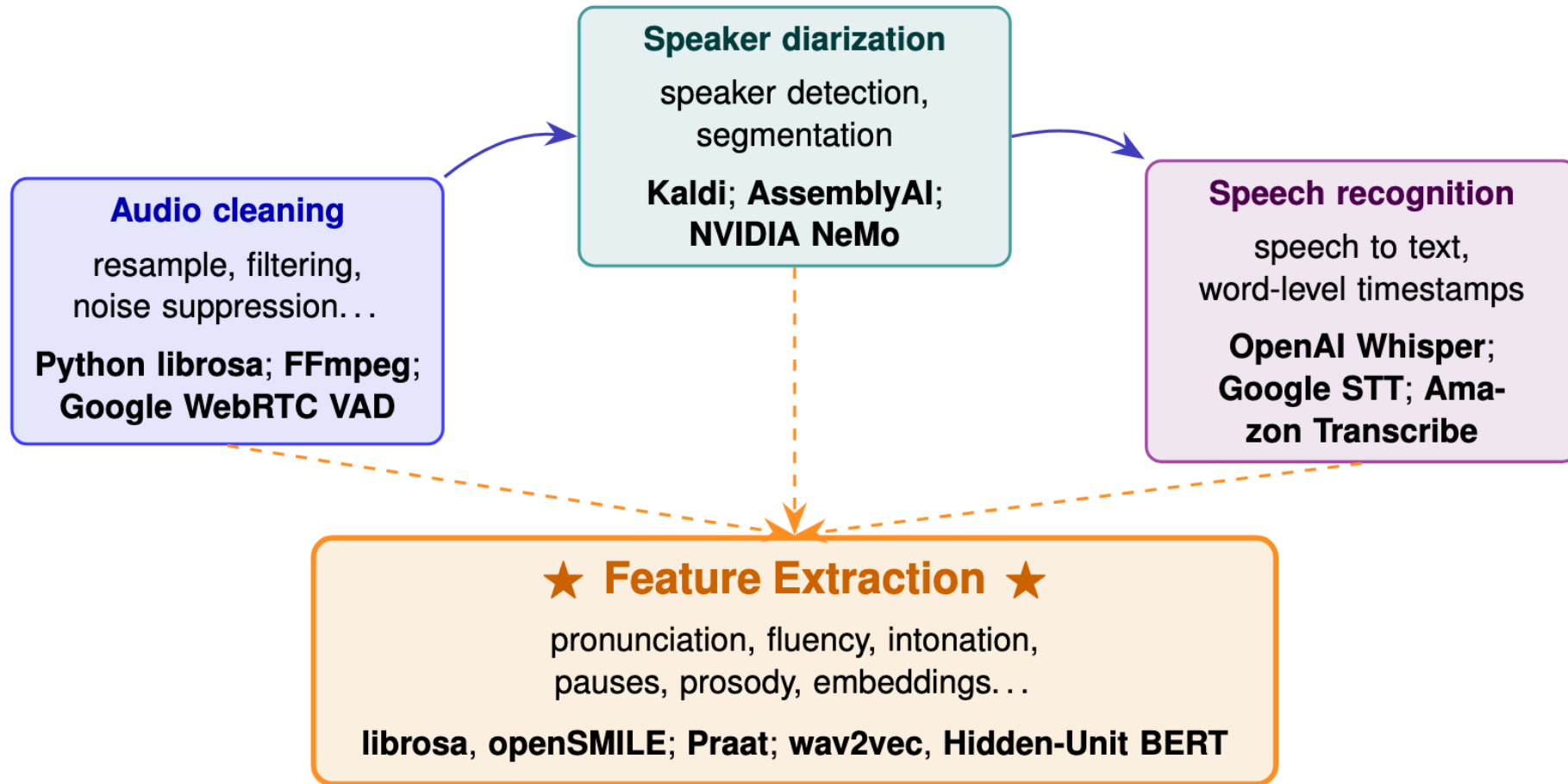
Preprocessing: Audio

Read-aloud responses, oral interviews, classroom recordings, lecture audio, etc.



- Audio carries signal beyond text.
- Automatic Speech Recognizer (ASR) quality is measured in word error rate (WER).
- Partitioning of an audio recording by *who* is speaking *when* (aka., diarization) may be unreliable in certain context.
- Acoustic models are trained on particular voices and recording conditions, so their accuracy can drop for accents, dialects, non-native or child speakers, and noisy audio.
- Voice is biometric.

Task and Tools for Audio Preprocessing



Preprocessing: Video

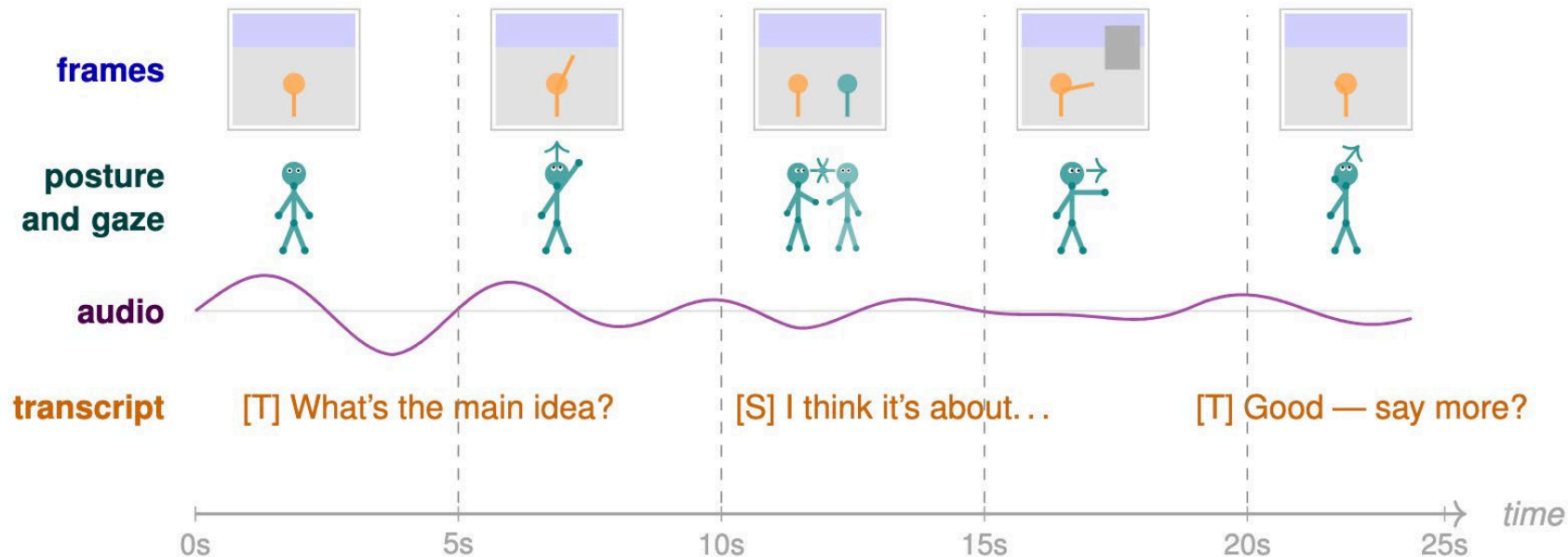
- Video is **multimodal**: streams of frames, audio, temporal
- Data source: classroom observations, oral presentations, test-taking recordings...



Reference: Bovik, A. B. (2009). The Essential Guide to Video Processing. Academic Press, Inc.

Preprocessing: Video (cont.d)

Preprocessing will extract and align signals across axes:

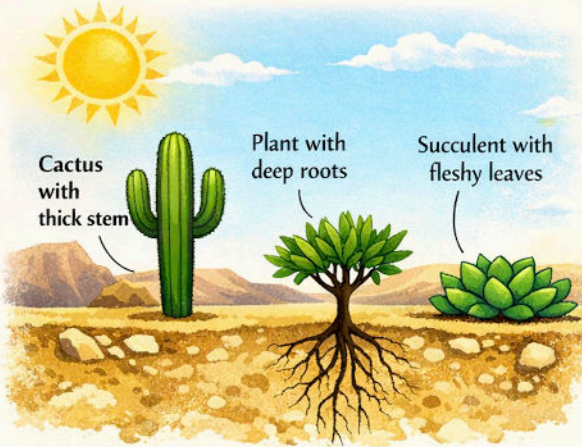


Some tools: **Google's MediaPipe**, **NVIDIA NeMo**, **Video-LLaVA**, **OpenPose** (motion detection), **Ultralytics YOLO** (object and motion detection)

Preprocessing: Process Logs

Often known as response process data, but can be more broadly captured, such as:

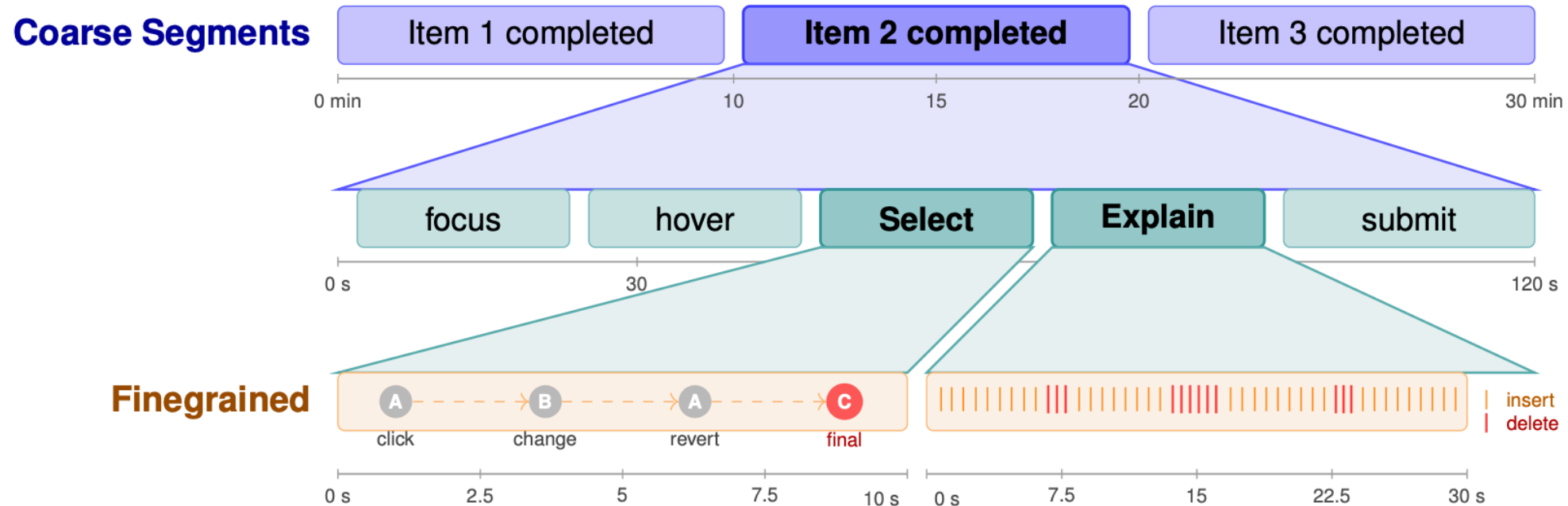
- action and navigation sequence, click stream, keystroke log, answer-change and revisit history, tool use...

Plant Adaptations in the Desert	Question 1
 The diagram illustrates three types of desert plants and their adaptations. On the left is a cactus labeled 'Cactus with thick stem'. In the center is a tree labeled 'Plant with deep roots' showing its root system extending deep into the ground. On the right is a succulent labeled 'Succulent with fleshy leaves'. The background shows a desert landscape with a sun and mountains.	A. Which adaptation helps plants survive in the desert? ☒ Storing water in thick stems. ☐ Having large, broad leaves; ☐ Growing in shady forests. B. Explanation: Correct. Storing water in thick stems helps desert plants survive by conserving water in harsh, dry conditions. Your Response:

Reference and additional reading: Ercikan, K., & Pellegrino, J. W. (Eds.). (2017). Validation of score meaning for the next generation of assessments: The use of response processes. Routledge.

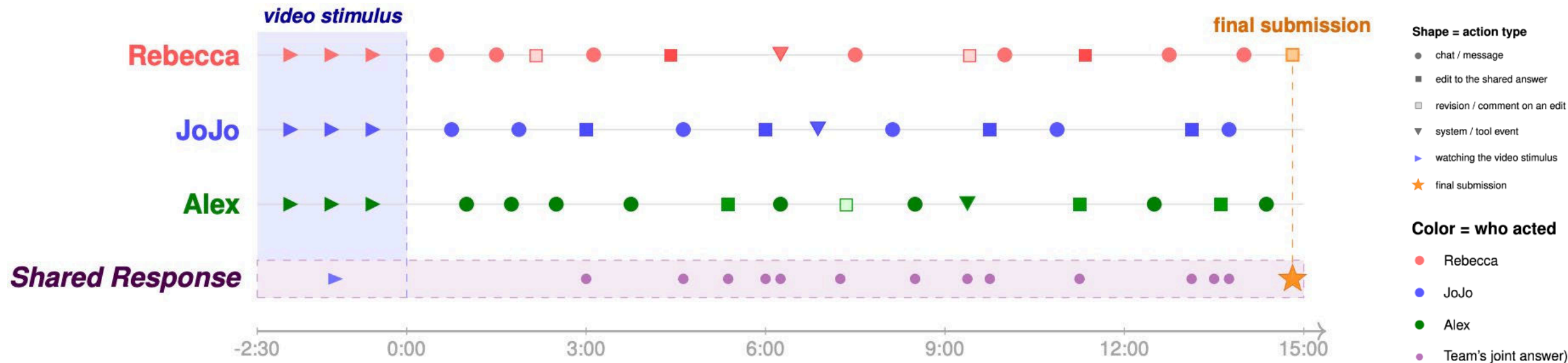
Data Granularity

- The right *data granularity* depends on the task, the item type, and the question to be answered.
- *Data granularity* is the level of detail and/or time resolution at which events are recorded and analyzed, ranging from coarse (whole items or sessions) to fine (individual clicks or keystrokes).



Process Log Example: Collaborative Problem Solving (CPS)

- Three students working together on a response in a shared online workspace (simulated data).
- The raw log is messy: a mix of chat turns, clicks, edits, and video-watching events, from multiple actors with no fixed schema and no scoring rubric attached.



Note. **Shared Response** lane tracks the team's one jointly-authored answer, e.g., the collaborative workspace that all members edit together, separate from each member's individual actions.

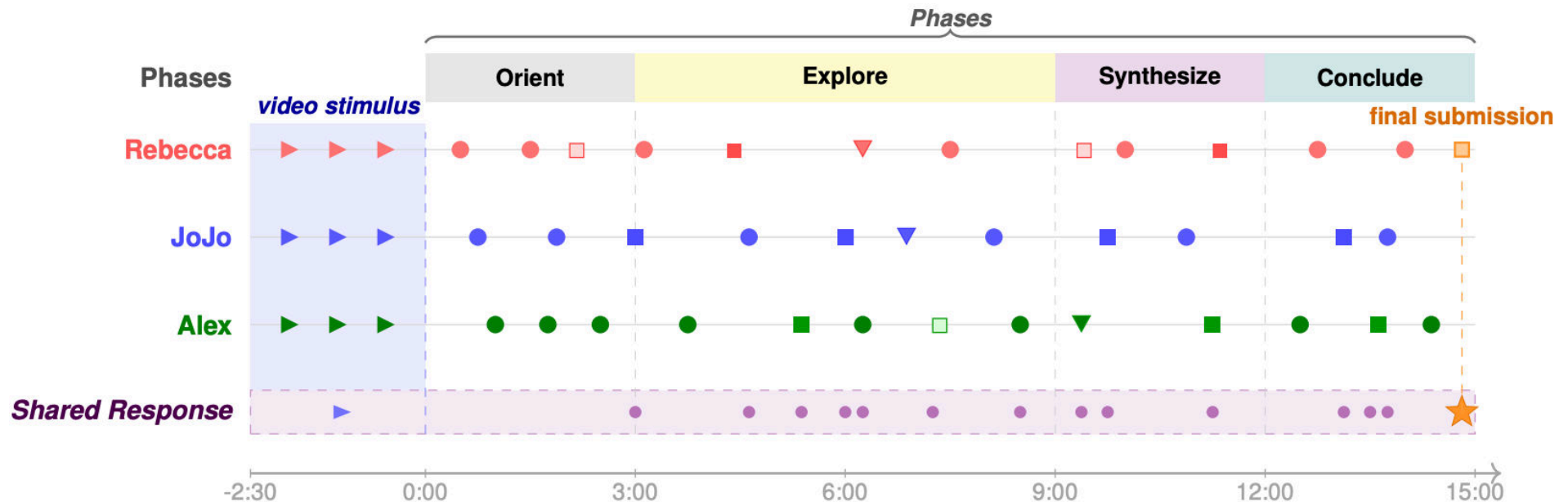
Methods & Tools for Analyzing Process Logs

- Process-log analysis overlaps with text, audio, and video processing.
- What makes it distinctive is that each action carries a time stamp, specifically enabling **sequence mining** and modeling of the order in which actions occur and **network analysis** of interactions among participants.

Method	Selected Models	Selected Tools
Statistical models	Markov models; time-series; latent class analysis	Python hmmlearn ; R markovchain , seqHMM
Network analysis	Social network analysis; Epistemic network analysis	Python NetworkX , igraph ; R sna , rENA
Sequence mining	Sequence pattern mining; alpha / heuristic / inductive miners; conformance checking	Python PM4Py , PrefixSpan-py ; R bupaR
Machine learning	clustering; classification; regression; topic modeling	Python scikit-learn
Deep learning & embedding models	LSTM; Transformer sequence encoders; contextual embeddings	PyTorch , TensorFlow/Keras , HuggingFace transformers , sentence-transformers , spaCy
LLM-based methods	zero-/few-shot tagging; RAG	OpenAI APIs; Qwen-3

Information Slicing

One often needs to slice the information in the logs into smaller, meaningful pieces to meet a targeted analytic goal.



Mechanical process

- total time
- response latency
- phase transition

Social aspect

- turn taking
- co-action

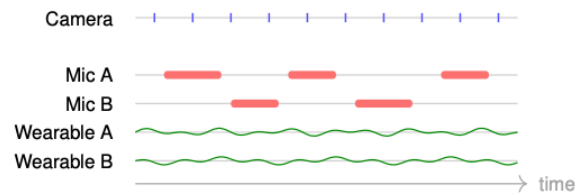
Cognitive aspect

- content structure
- idea uptake
- topic shift

Multimodality in Education

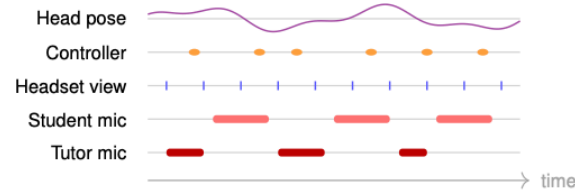
- Multiple streams open up new constructs.
- Devices (e.g., wearables, tablets, VR/AR headsets) have become more accessible.
- Foundational AI models and cheaper sensors have lowered the cost of data analytics.

Classroom discussion



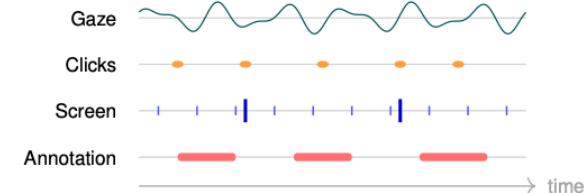
Sync video, mic, and wearable to each student.

VR/AR science lesson



Sync head pose to what's rendered and who is speaking.

Online reading

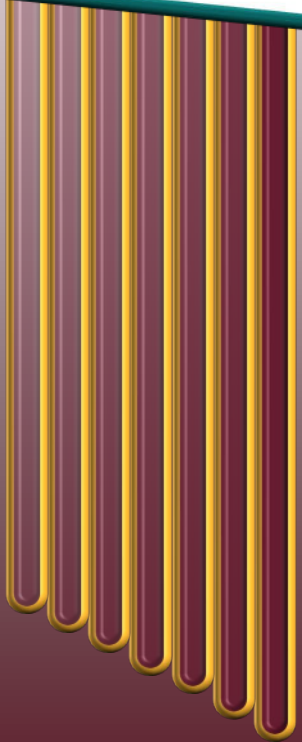


Sync gaze to the on-screen page, actions, and comments.

Preprocessing: Multimodal and Sensor Data

- Single-modality preprocessing (text, audio, video, logs) cleans one stream at a time.
- Multimodal preprocessing starts where the single-modality pipelines end with 2+ streams that need to be related
 - **Different clocks.** Each device has its own reference time.
 - **Different sampling rates.** 250Hz gaze vs. 1Hz heart rate.
 - **Different identities.** The face in the video, the voice in the mic, and the wearable should link to the same student.
 - **Different semantic units.** A fixation, an utterance, a key press, and a heart-rate etc. require meaningful alignment.
- Fusion choices depend on analytic goals.

Section 3: Representing Unstructured Data



Section Learning Objectives

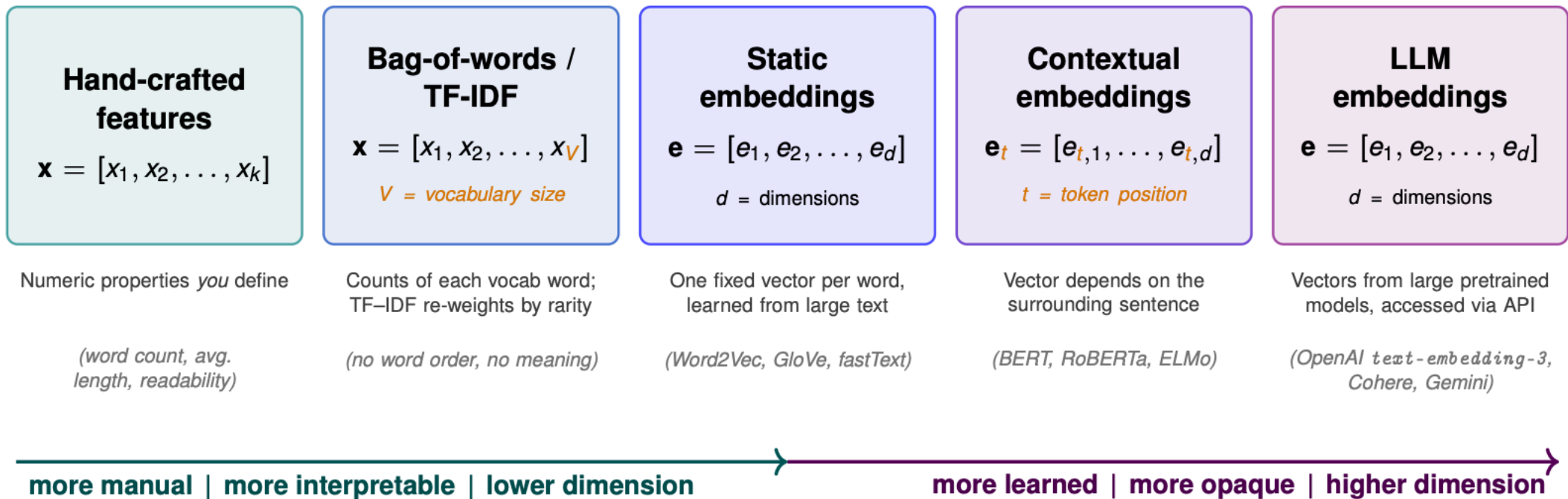
Section 3: Representing Unstructured Data

Learners will be able to distinguish different ways of representing unstructured data as numeric vectors.

Learners will know the considerations when choosing an embedding model.

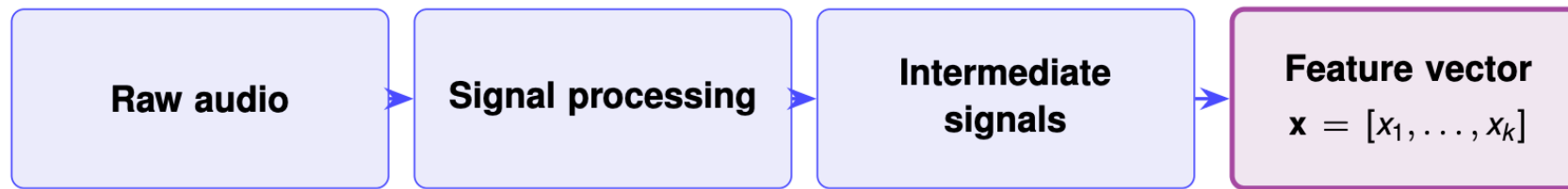
Representing Unstructured Data

- Turn cleaned text, images, audio, or sequences into **numeric vectors** that downstream models can consume.
- Take *text* as example:



Hand-Crafted Features

- Hand-crafted features can be extracted from intermediate signals.
- Use **audio** as an example.



Signal processing: set of methods for cleaning, transforming, and extracting features from time- or space-varying data, turning the raw signal into model-ready representations.

Construct	Intermediate signal	Example features
Fluency	Word & silence timings	speaking rate, mean pause duration
Pronunciation	acoustic-model posteriors	mean acoustic likelihood
Prosody & rhythm	Pitch & intensity contours	pitch variation, durational variability
Vocabulary & grammar	Automatic Speech Recognizer (ASR) transcript	type-token ratio, grammatical-error rate

Bag-of-Words

- Represent a document as a vector of word counts over a fixed vocabulary
- Word order and meaning are discarded.

D1: "students read books" **D2:** "students write essays" **D3:** "students read essays" **D4:** "teachers write books"

	students	read	books	write	essays	teachers
D1	1	1	1	0	0	0
D2	1	0	0	1	1	0
D3	1	1	0	0	1	0
D4	0	0	1	1	0	1

Each document $d \mapsto \mathbf{x}_d \in \mathbb{R}^V$, where V is the vocabulary size.

- Pros: simple, fast, interpretable; strong baseline for classification.
- Cons: high-dimensional and sparse; ignores word order and syntax; treats morphological variants ("essay"/"essays") and synonyms ("essay"/"paper") as unrelated.

TF-IDF: Re-weighting by Rarity

Raw counts over-value frequent words. **TF-IDF** down-weights words that appear in many documents and up-weights distinctive ones.

$$tfidf(t, d) = tf(t, d) \times \log \frac{N}{df(t)}$$

Notation:

- t : a term (word)
- d : a document
- $tf(t, d)$: count of t in d
- N : total number of documents
- $df(t)$: number of documents containing t

Intuition (from the 4 documents):

- “students” is in 3 of 4 docs → low weight (not distinctive)
- “teachers” is in only 1 doc → high weight (distinctive)

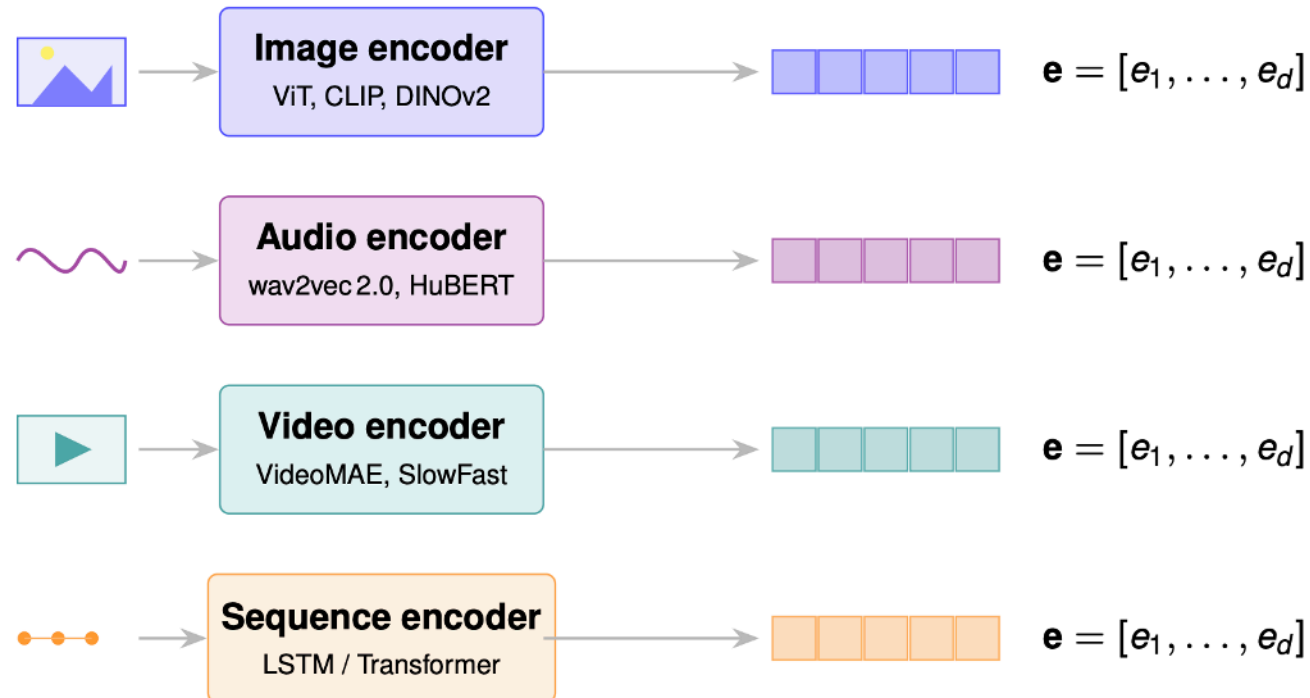
Text Embeddings: Static → Contextual → LLM

Family	Main Idea	Example Tools
Static	One vector per word, regardless of context.	Word2Vec, GloVe, fastText (also via {spaCy} static-vector models)
Contextual	The vector for a word depends on the surrounding sentence. Same word, different meaning different vector.	ELMo, BERT, RoBERTa, DeBERTa; HuggingFace transformers, sentence-transformers
LLM-based	Embeddings produced by very large pretrained models; often the strongest off-the-shelf semantic similarity.	OpenAI text-embedding-3, Cohere embed-v4, Gemini text-embedding, Qwen3-Embedding

- **Typical dimensions:** static ~50-300 (GloVe 50/100/200/300, Word2Vec 300); contextual ~768-1024 (BERT-base 768, BERT-large 1024); LLM ~1024-4096 (OpenAI `text-embedding-3-small` 1536, `-large` 3072)
- **Tradeoff:** richer representation → higher cost, larger storage.

Embeddings Beyond Text

- Any modality can be encoded into a numeric vector that downstream models consume, just like text embeddings.
- Each single-modality embedding can be used on its own to model scores, make predictions, cluster, or detect anomalies.



A Shared Embedding Space for Multimodal Data

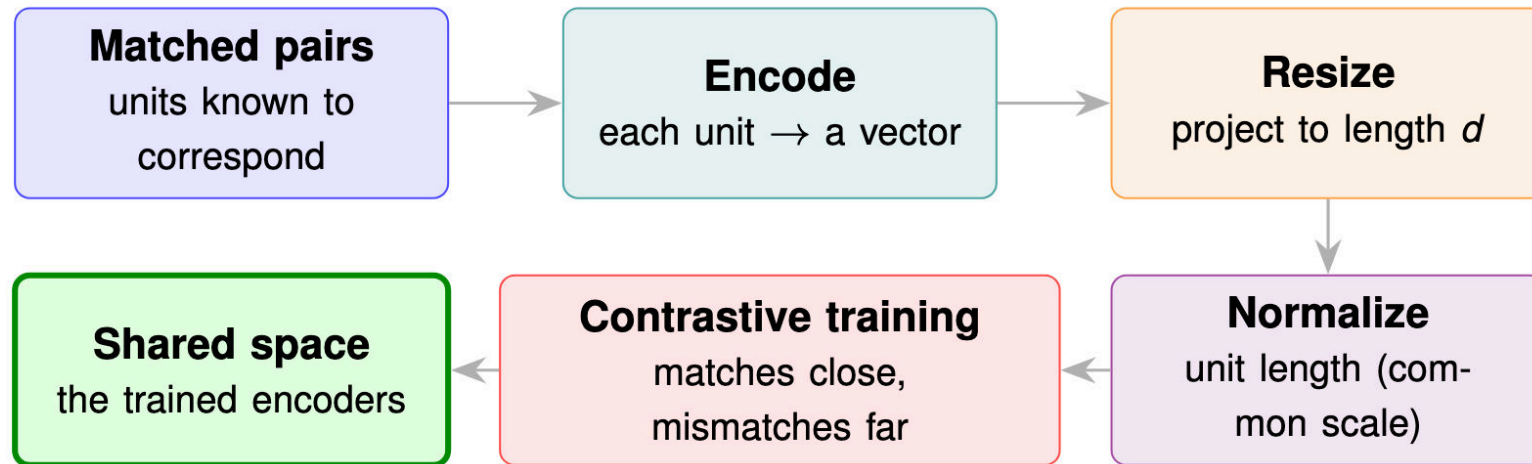
A shared embedding space places vectors from different modalities into one common space, where nearby vectors mean similar things.

Why build one?

- **Cross-modal search & retrieval:** query in text, find matching drawings, video, or audio.
- **Link streams from the same event:** relate transcript $\leftrightarrow$ audio $\leftrightarrow$ on-screen action.
- **One model, many inputs:** a single downstream model works regardless of modality.
- **Compare units with no shared surface form:** a drawing and a sentence can be close if they mean the same thing.

Building a Shared Space: One Common Approach

- A shared space can be built in several ways. The dominant approach today is contrastive learning (e.g., CLIP, trained on ~400M image-caption pairs).
 - Contrastive learning: a training strategy that pulls matching pairs together in a shared embedding space and pushes non-matching pairs apart.



Encoders (models):

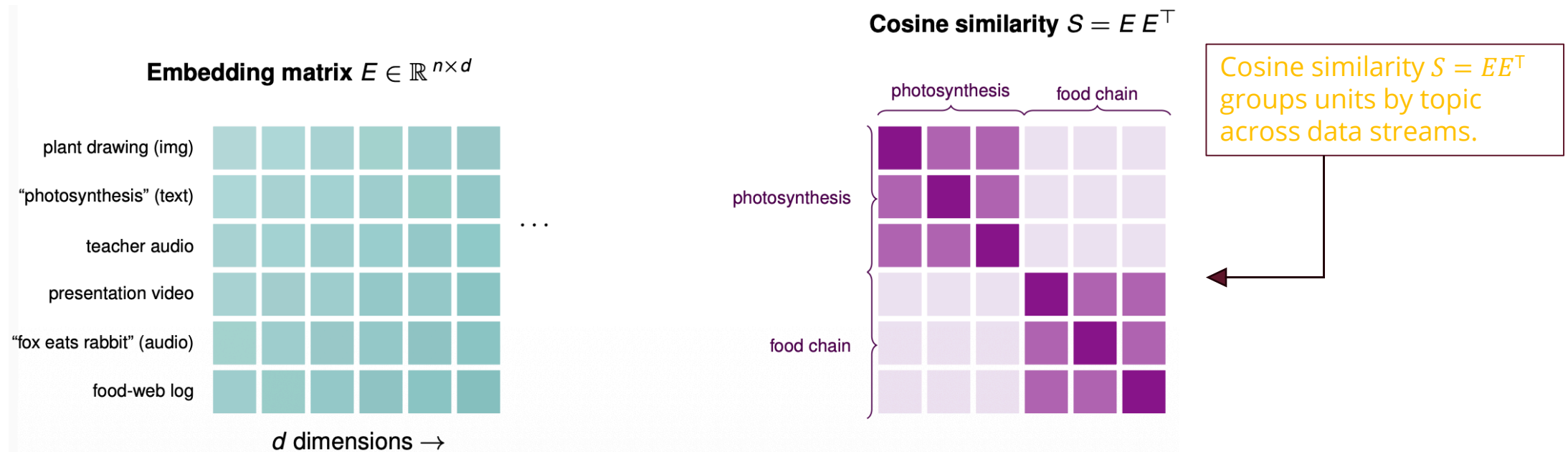
- OpenAI's **CLIP**: image, text
- **ImageBind**: different modalities in one space

Open-source tools:

- HuggingFace **transformers**
- similarity search: **FAISS** (Facebook's AI Similarity search)

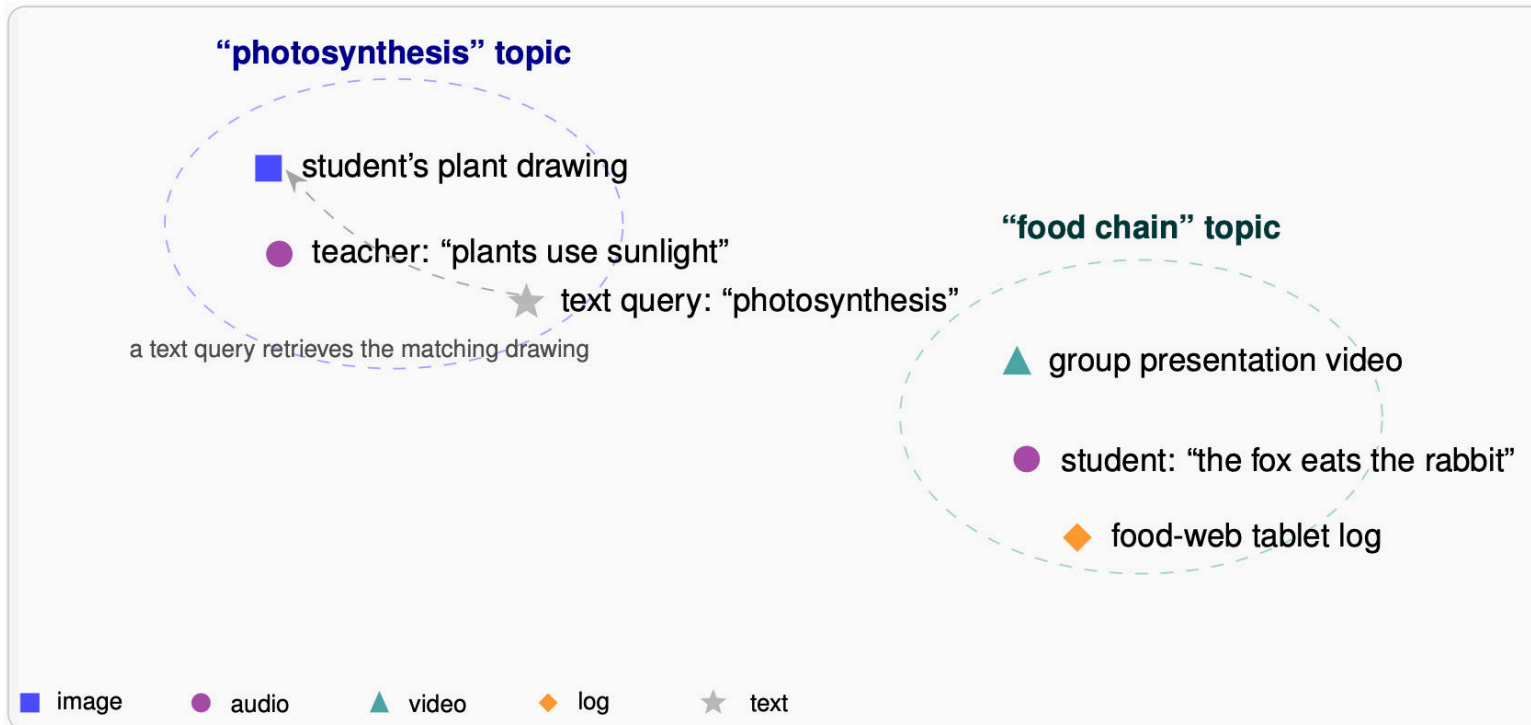
Embedding Output for Downstream Analysis

- Run through an encoder, every item, regardless of modality, becomes a normalized d -dimensional vector in the same space.
- Stacked, they form an $n \times d$ table E , the input to Section 4 (e.g., clustering, retrieval, and scoring on these vectors).



A Hypothetical Classroom Scenario

A science lesson on *ecosystems*, recorded with a classroom **camera**, group **microphones**, students' **drawings & logs**, the lesson **transcript**, and the **lecture notes**.



Organize mixed evidence: units cluster by topic, not file type; a portfolio of evidence groups itself.

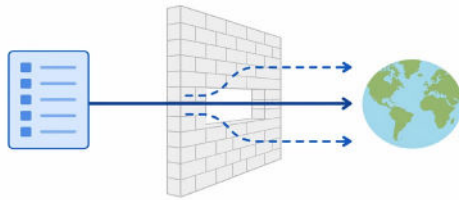
Spot gaps & links: see which topics lack student evidence, or connect a struggling student's audio to the relevant lesson note.

Common Factors for Choosing an Embedding Model

- **Dimension & cost.** Larger vectors can capture more nuance but cost more to store, search, and serve, **and** they do not always improve task performance.
- **Pretraining domain.** A model trained on web text may underperform on handwriting, audio, or other data types; favor in-domain or fine-tuned models when possible.
- **Local vs. API.** Open-source models (e.g., BERT) run on your own hardware; hosted APIs are convenient but add cost, vendor dependency, and data governance concerns.

Cautionary Notes on Using Embeddings

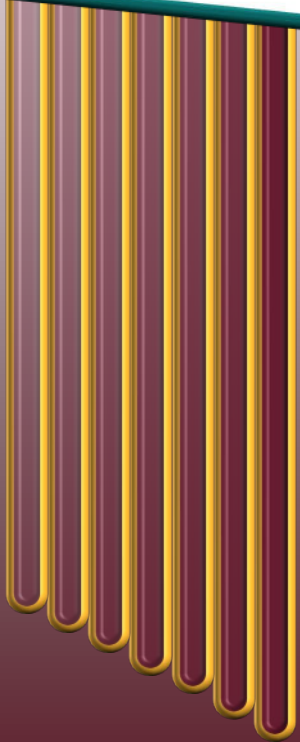
- **Record the model version.** Vendor updates can silently shift downstream scores.
- **Watch for data leakage.** Pretraining data may overlap with your evaluation set, inflating results; be cautious when reporting performance.
- **Fairness and bias.** Embeddings inherit biases from pretraining data.
- **Reproducibility.** Store the full pipeline with the data, so that results can be regenerated.



812406, 7259134, 62017, 930542, 205781, 742398, 5829105, 9134620
6712839, 309871, 2045876, 195023, 8764251, 430028, 998214, 2174036
112908, 8845120, 452771, 6109033, 223144, 7710590, 318567, 5098876
6731045, 809237, 7345612, 998301, 4057298, 665100, 123876, 7765432
945210, 5531892, 110876, 254003, 9156324, 778214, 607891, 4006327
3216547, 700238, 8910542, 125987, 5086431, 889201, 312765, 2590386
230981, 9408632, 187236, 6745098, 336107, 721549, 6032981, 994210
1158073, 6629401, 403692, 8507194, 905332, 4462189, 720563, 135884
5723068, 209867, 6173589, 331275, 8046127, 993702, 148593, 6647321
778491, 1162540, 5028791, 948661, 2330175, 8607432, 719805, 306244
4635817, 187904, 5501283, 297661, 6804510, 722036, 9402058, 6621173
501388, 9587372, 221645, 7342190, 880120, 4198763, 104568, 6057328
... 207845, 8120673, 149839, 6205789, 990341, 4146752, 8812096



Section 4: Analyzing Unstructured Data



Section Learning Objectives

Section 4: Analyzing Unstructured Data

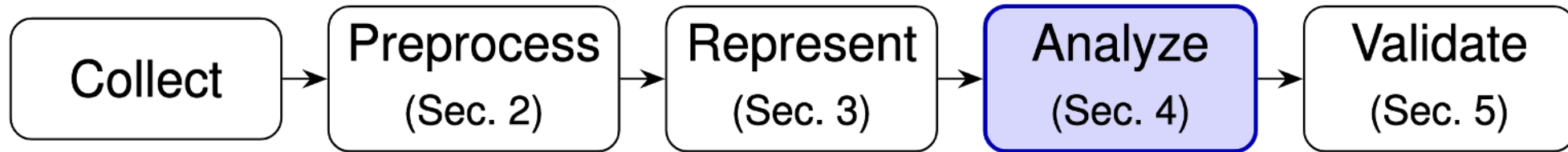
Learners will be able to explain how data representations are turned into inferences.

Learners will be able to align the analysis approach with its purpose.

Learners will be able to identify the main families of analysis and know the options.

From Representations to Inferences

In Section 4, we ask: what can we learn from those data representations?

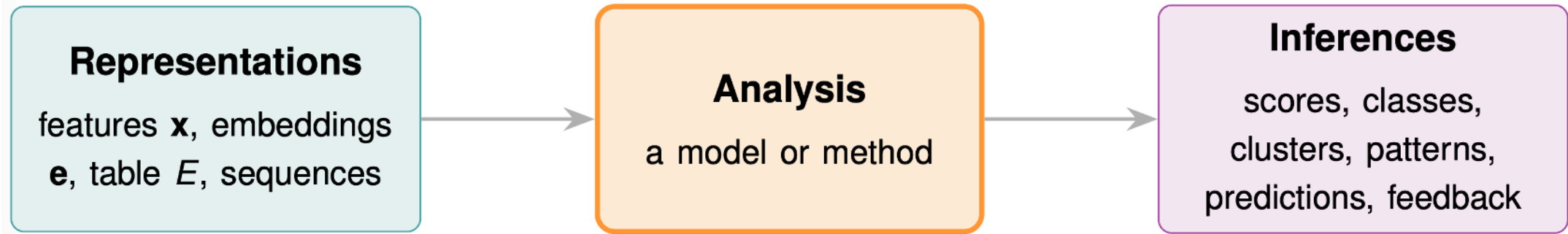


The choice of analysis depends on the question, the labels you have, and the construct of measurement.

References:

- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction*. Springer Science & Business Media.
- Cook, L., & Pitoniak, M. (2025). *Educational measurement* (5th Edition). URL: <https://ncme.org/wp-content/uploads/2026/01/Educational-Measurement-Fifth-Edition.pdf>
- Agresti, A., & Finlay, B. (2008). *Statistical Methods for the Social Sciences* (4th Edition). Prentice Hall.

What Analyses Produce



- **Descriptive:** summarize what is in the data (e.g., clusters, topics, process maps).
- **Predictive:** assign labels or scores (e.g., classification, scoring).
- **Relational:** characterize structure and interaction (e.g., sequences, networks).

Analysis Families

A. Supervised (labels observed)

- Classification (SVM, RF)
- Regression & scoring (ridge, GBM)
- Learner modeling (DKT, GLMM)

B. Unsupervised (no labels)

- Clustering (k-means, nearest neighbor)
- Topic modeling (LDA, BERTopic)
- Anomaly detection (Isolation Forest)

Questions Crosscutting A and B

Is the target observed or latent?: directly measured, or an unobserved construct estimated by a latent-variable or measurement model (e.g., IRT, CDM, SEM).

What shape is the data?: *tabular*, *sequential* (process mining, Markov model), or *relational* (networks: SNA / ENA).

What tool implements it?: classical ML · DL · LLMs (zero-shot = unsupervised; fine-tuned = supervised).

Note. *Self-supervised learning* (labels derived from the data) is a third learning signal. It trained the Section 3 representations and today's LLMs, then feeds into A or B.

Analysis Purpose: Families (by label)

Family	Typical question it answers	Examples / output
A. Supervised	Can I train a model to assign a known label or score?	Labels, scores, ability estimates
B. Unsupervised	What structure or themes exist with no labels?	Clusters, topics, outliers

Crosscutting Models: With vs. Without Labels

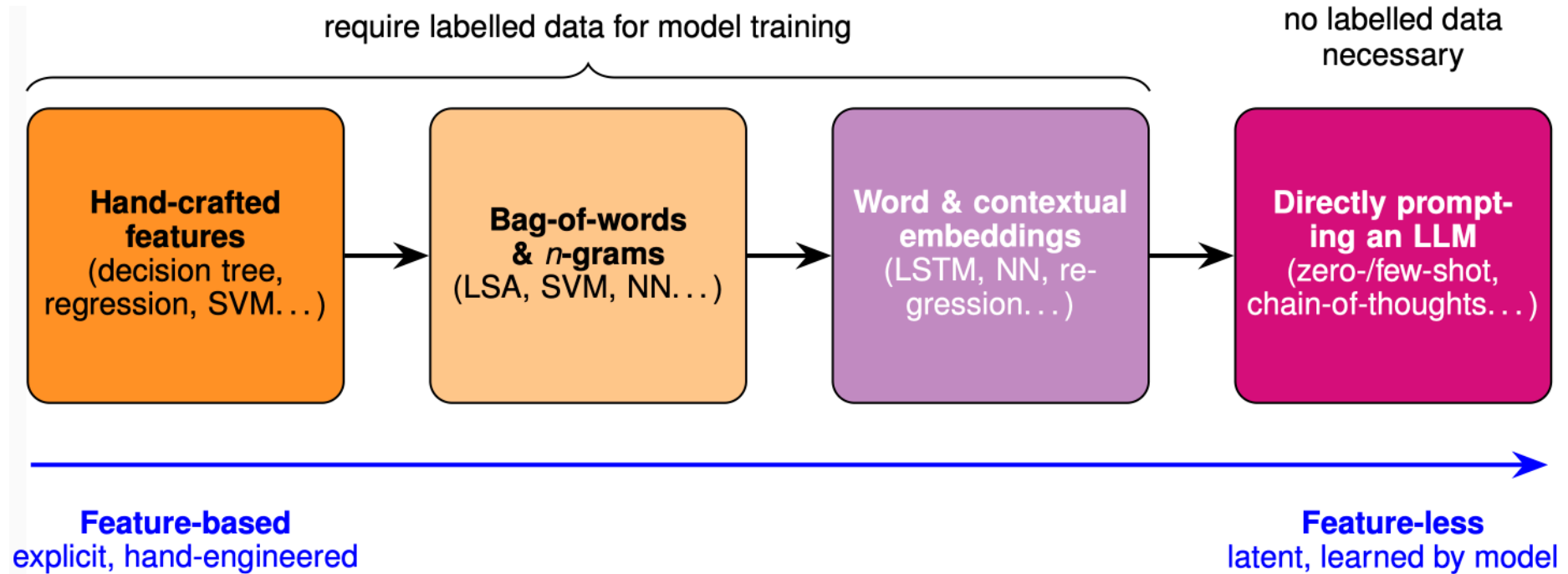
Model / method	Without labels (unsupervised) Examples	With labels (supervised) Examples
Process & sequence mining	Discover the process map from action logs	Use sequence features to classify teams as high- vs. low-performing
Markov chain	Uncover latent behavioral states in click-streams	Estimate transitions among human-coded phases; state prediction
Network analysis	Describe interaction structure via centrality and density	Use network metrics to predict rated collaboration quality
LLMs	Zero-/few-shot tagging; embed responses and cluster	Fine-tune on human-scored essays to predict rubric scores
Latent-variable / measurement model	Cognitive Diagnostic Models for profiling	Score new responses with known item parameters; DIF across known groups

Illustration with Two Applications

- Common applications of unstructured-data analysis in education, each studied with a wide range of methods.
- Each one can be tackled with **many different techniques and models**, depending on whether human-provided labels are available, all operating on the vector representations x_i built in Section 2.

	Without labels (unsupervised)	With labels (supervised)
AI scoring & feedback	Cluster responses, model topics, or prompt an LLM zero-/few-shot for scores and rationales	Fit a model on human-scored responses to predict rubric scores and flag strengths/weaknesses
Teamwork skill coding & analysis	Discover interaction processes, latent states, and network patterns from the unlabeled log	Train a classifier to code chat turns/actions into skill categories and predict team success

From Hand-Crafted Features to LLM-Based Scoring



- **Left end:** AI scores are a function of a set of *well-defined, hand-crafted features*.
- **Right end:** AI scores are a function of *latent relations inside the text response*, inferred by LLMs.

AI Scoring: Supervised Model

- Each response i is represented by a vector,
 $x_i = (x_{i1}, \dots, x_{id})^T \in \mathbb{R}^d$ (TF-IDF features or an LLM embedding).
- A *model* maps the representation to a predicted score,

$$\hat{y}_i = f_{\theta}(x_i),$$

where θ are the model parameters, y_i is the human-assigned score (e.g., a rubric score), and $\hat{y}_i$ is the predicted score.

- Estimate θ by minimizing a *loss* over n human-scored responses, e.g., the mean squared error (MSE):

$$\hat{\theta} = \arg \min_{\theta} \frac{1}{n} \sum_{i=1}^n (y_i - f_{\theta}(x_i))^2.$$

AI Scoring: Using an LLM

Supervised: fine-tuned

- Fine-tune on human-scored pairs $\{(text_i, y_i)\}$
- Trainable parameters θ predict the score:

$$\hat{y}_i = f_{\theta}(text_i)$$

- Estimate θ by minimizing a loss (e.g., MSE):

$$\hat{\theta} = \arg \min_{\theta} \frac{1}{n} \sum_{i=1}^n (y_i - f_{\theta}(text_i))^2$$

Unsupervised: zero-/few-shot

- Prompt a LLM (fixed parameters θ_0 with nothing given or only the rubric:
$$\hat{y}_i = f_{\theta_0}(rubric, text_i)$$
- No loss is minimized: θ_0 stays fixed.
- Few-shot further adds k example pairs (text + its correct score, so:

$$\hat{y}_i = f_{\theta_0}(rubric, (text_1, y_1), \dots (text_k, y_k), text_i)$$

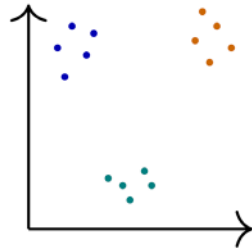
Note. Unlike the fixed representation x_i of the previous slide, the LLM builds its own internal representation of $text_i$.

AI Scoring: Unsupervised Beyond a Score

Suppose we only have the representation $x_i \in \mathbb{R}^d$ (and the raw $text_i$), but no human scores y_i to learn from.

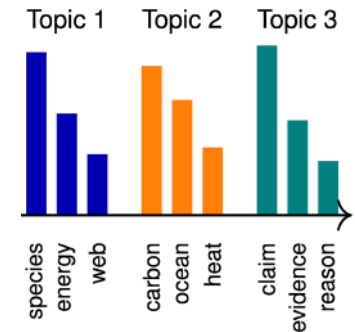
Clustering

K-means: $\min \sum_i \|x_i - \mu_{c(i)}\|^2$,
where c_i is cluster of i , $\mu_{c(i)}$ is its centroid.



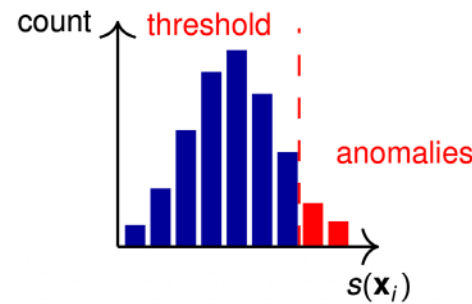
Topic modeling

bar = $P(\text{word} | \text{topic})$;
mixture $\pi_i \in \delta^{K-1}$ (LDA), where δ^{K-1} is simplex over K topics.



Anomaly detection

anomaly measure $s(x_i)$, where $s(\cdot)$ is an anomaly index
Off-task/copied (Isolation Forest, LOF)



Rationales and critiques

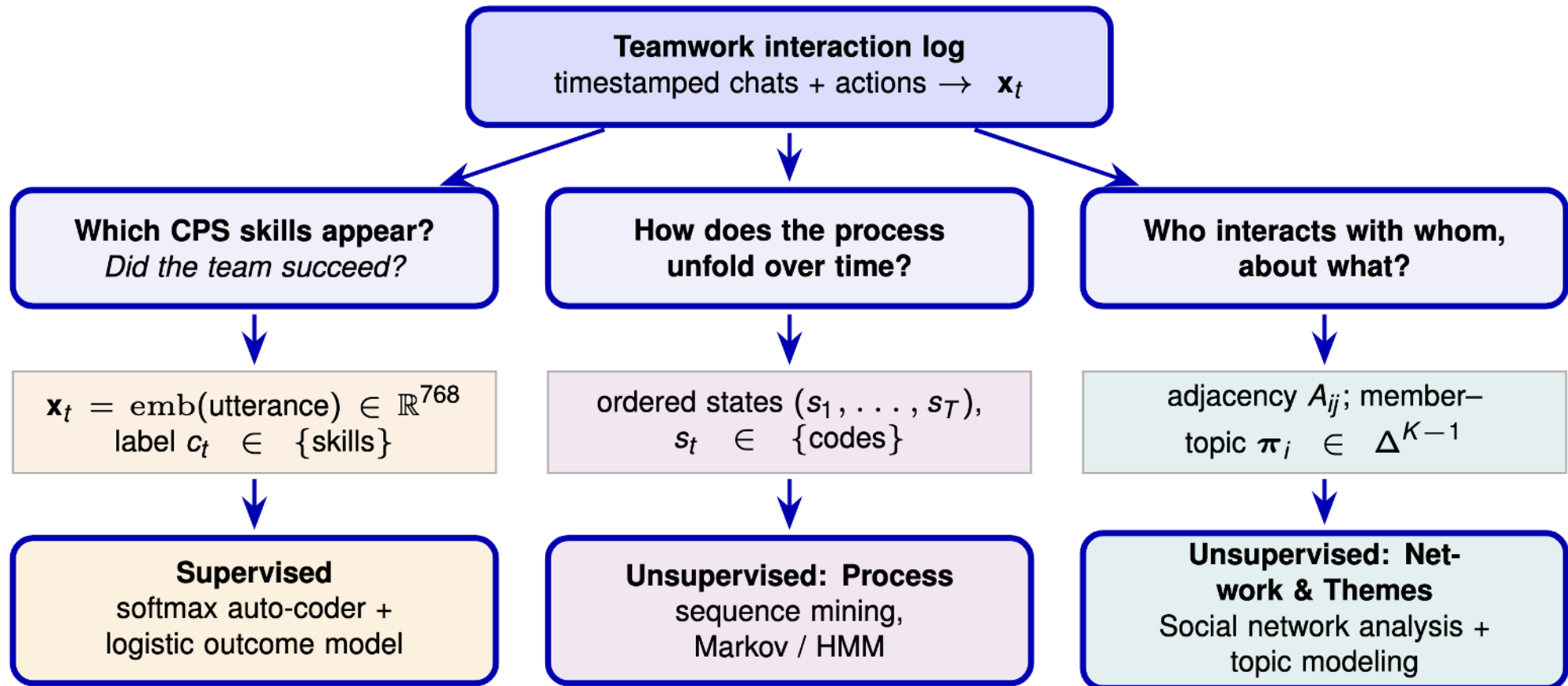
Prompt LLM to generate rationale and critiques of a response: $critique_i = f_{\theta_0}(\text{rubric}, text_i)$, where θ_0 is frozen (pretrained) LLM weights

"Clear thesis, but the evidence is thin.
Add a source and tie it back to your claim."

Reference: Johnson, M., et al. (June 2026). Modeling construct-relevant processes in automated essay scoring using LLM-generated rationales. Presentation at the 2026 Modern Modeling Methods Conference, New York, NY.

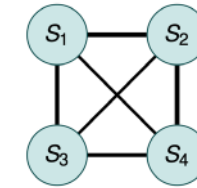
Analyzing Teamwork From Timestamped Chats & Actions

Assume each chat turn and task action is encoded as a representation $x_t \in \mathbb{R}^d$ (Section 3)

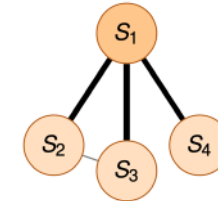


Networks, Clusters & Topics Analysis

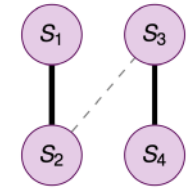
- Social network analysis:** adjacency A_{ij} = # exchanges from student i to j (n = team size). Example measures:
- Degree centrality $\deg(i) = \sum_j A_{ij}$ (central vs. peripheral).
 - Density = $\frac{\sum_{i \neq j} 1(A_{ij} > 0)}{n(n-1)}$ ($1(\cdot) = 1$ when tied)



Team 1 • Balanced



Team 2 • Leader-driven

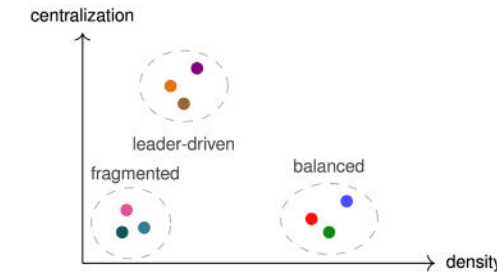


Team 3 • Fragmented

edge thickness $\propto A_{ij}$

Clustering: each team $\rightarrow$ feature vector g_t (density, centralization). K-means group them:

$$\min \sum_t \|g_t - \mu_{c(t)}\|^2.$$



each colored point = one team

Topic modeling: each team's chat d is a mixture of K topics, θ_d , each topic k is a set of characteristic words, ϕ_k .

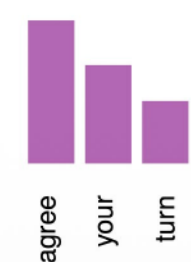
Planning



Evidence



Coordination

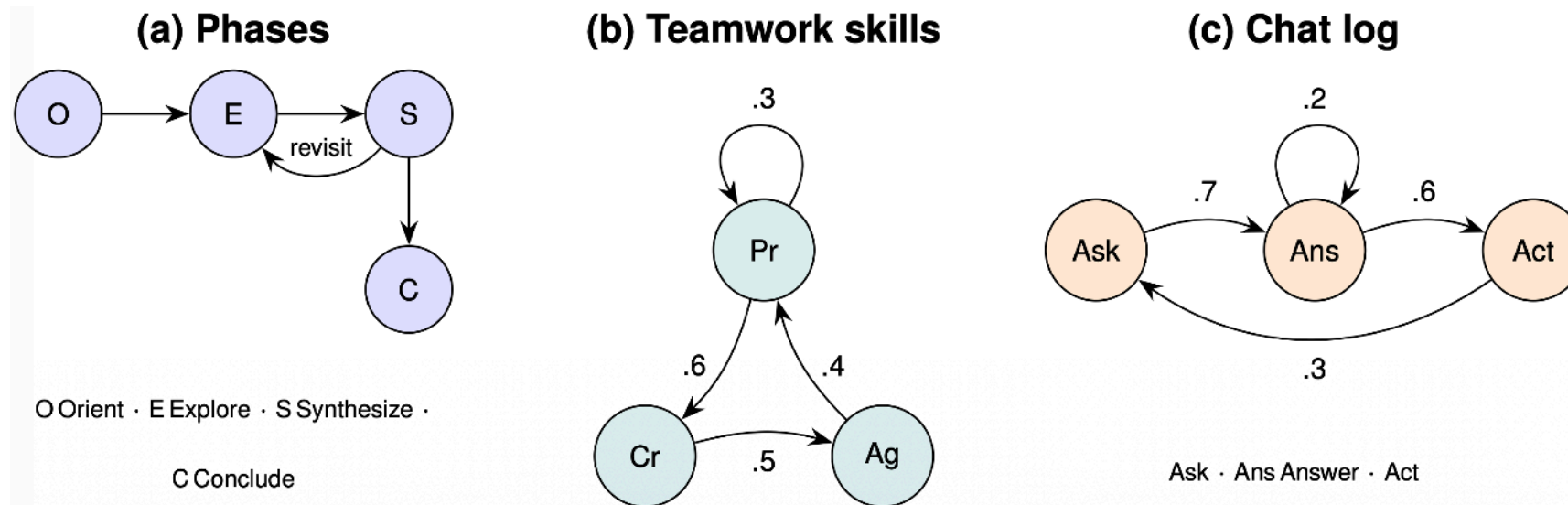


Unsupervised Process Modeling

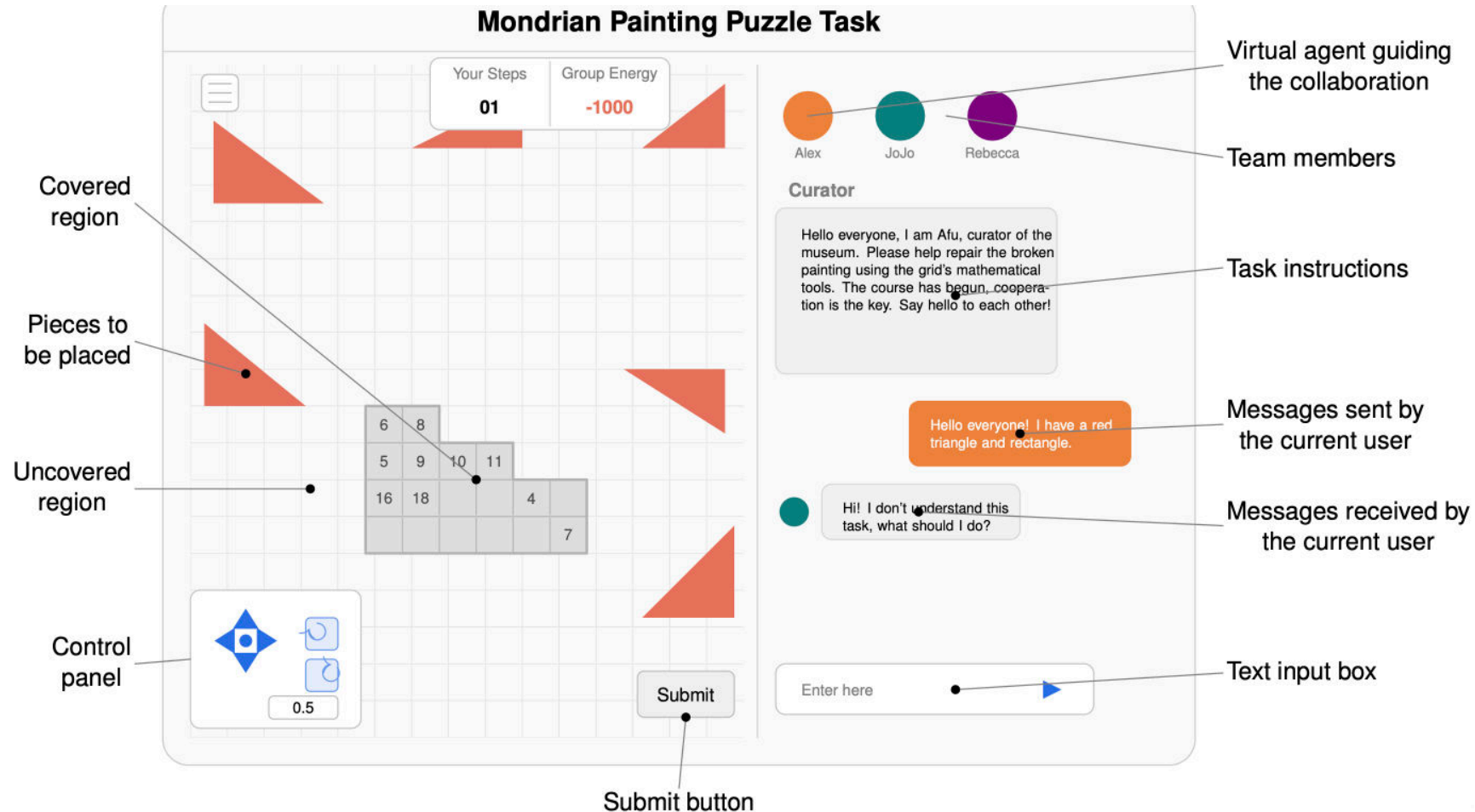
With a sequence $s_1, s_2, \dots$ of states (s_t state at step t):

- **Markov chain:** transition probability $T_{ij} = \Pr(s_{t+1} = j | s_t = i)$, estimated from counts $\hat{T}_{ij} = n_{ij} / \sum_k n_{ik}$, where $n_{ik} = \#$ of transitions $i \rightarrow k$ and the denominator sums over all next states k .
- **Hidden Markov model:** each observed state s_t is emitted by a hidden state z_t ; parameters $(\pi, \mathbf{T}, \mathbf{B})$ (initial-state π , transition $\mathbf{T}$, emission $\mathbf{B}$) are fitted.

Three examples (edge is T_{ij})



Teamwork Skill Coding and Analysis



The task area shows an incomplete Mondrian painting made of shaded regions (with clear internal divisions) and blank regions (unknown structure), surrounded by movable triangular and rectangular pieces. Because "Mondrian" is an information-asymmetric task, each collaborator sees the overall target shape and their own shaded structure, but not the others' pieces or shaded regions, so they must pool everyone's information to finish. A control panel at the lower left lets players rotate and move pieces, while a chat panel on the right supports real-time communication.

Reference: Xu, Q., Ge, Z., Zhu, M., Wang, J., & Zhang, M. (2024). Designing an Online Jigsaw Puzzle Task to Assess Collaborative Problem-Solving Skills. In Proceedings of the 2024 IEEE Integrated STEM Education Conference (ISEC), 1–8. <https://doi.org/10.1109/ISEC61299.2024.10664914>

A Real Data Set and Scripts

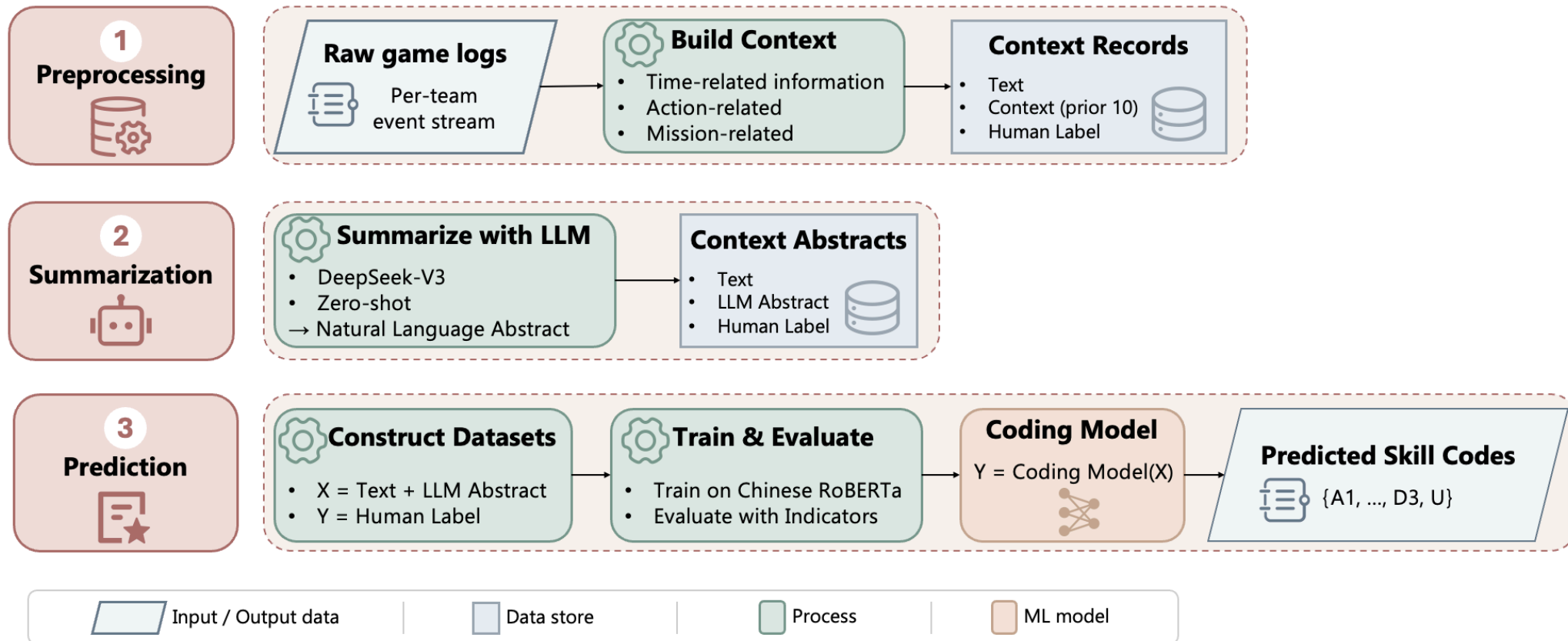
- Raw logs include: user names, timestamps, action on the puzzle, free-text chat utterances interleaved with in-game actions and mission events.
- Human annotated on teamwork skills

	A	B	C	D	E	F	G	H
1	class	user_name	timestamp	puzzle_action	progress	user_move	chat_pos	skill
1890	304	李翌晨	2023-09-06 16:18:12.0	To [A] move [X] steps and [B] get to the [C] place	第二关	X	-0.5	
1891	304	李翌晨	2023-09-06 16:18:12.0	To [A] move [X] steps and [B] get to the [C] place	第二关	X	-0.5	
1892	304	李翌晨	2023-09-06 16:18:12.0	To [A] move [X] steps and [B] get to the [C] place	第二关	X	-0.5	
1893	304	李翌晨	2023-09-06 16:18:12.0	To [A] move [X] steps and [B] get to the [C] place	第二关	X	-0.5	
1894	304	李翌晨	2023-09-06 16:18:12.0	To [A] move [X] steps and [B] get to the [C] place	第二关	Y	-0.5	
1895	304	李翌晨	2023-09-06 16:18:12.0	To [A] move [X] steps and [B] get to the [C] place	第二关	Y	-0.5	
1896	304	李翌晨	2023-09-06 16:18:12.0	To [A] move [X] steps and [B] get to the [C] place	第二关	Y	-0.5	
1897	304	李翌晨	2023-09-06 16:18:12.0	To [A] move [X] steps and [B] get to the [C] place	第二关	Y	-0.5	
1898	304	李翌晨	2023-09-06 16:18:12.0	To [A] move [X] steps and [B] get to the [C] place	第二关	Y	-0.5	
1899	304	李翌晨	2023-09-06 16:18:21.0	[A] send [B]		公良儒	干啥	U
1900	304	巫文轩	2023-09-06 16:18:31.0	[A] send [B]		庖成	g不要和c放一起	C1
1901	304	巫文轩	2023-09-06 16:18:46.0	[A] send [B]		庖成	两个e	C1
1902	304	巫文轩	2023-09-06 16:18:52.0	[A] send [B]		庖成	together	C1
1903	304	李翌晨	2023-09-06 16:18:58.0	To [A] move [X] steps and [B] get to the [C] place	第二关	Y	-0.5	
1904	304	李翌晨	2023-09-06 16:18:58.0	To [A] move [X] steps and [B] get to the [C] place	第二关	Y	-0.5	
1905	304	李翌晨	2023-09-06 16:18:58.0	Change select from [A] to [B]	第二关	17	15	
1906	304	李翌晨	2023-09-06 16:18:58.0	Unselect [A]	第二关	15		
1907	304	李翌晨	2023-09-06 16:18:58.0	Select [A]	第二关	15		
1908	304	李翌晨	2023-09-06 16:18:58.0	Unselect [A]	第二关	15		
1909	304	李翌晨	2023-09-06 16:18:58.0	Select [A]	第二关	15		
1910	304	李翌晨	2023-09-06 16:18:58.0	Unselect [A]	第二关	15		
1911	304	李翌晨	2023-09-06 16:18:59.0	Select [A]	第二关	15		
1912	304	李翌晨	2023-09-06 16:18:59.0	Unselect [A]	第二关	15		
1913	304	李翌晨	2023-09-06 16:18:59.0	Select [A]	第二关	15		
1914	304	李翌晨	2023-09-06 16:18:59.0	To [A] move [X] steps and [B] get to the [C] place	第二关	X	0.5	

Sample data and scripts: <https://github.com/zestgroup/cps-coding-sna>

Teamwork Skill Coding Demo

Purpose: automatically code each behavior into skills, turning raw interaction streams into interpretable teamwork measures at scale.

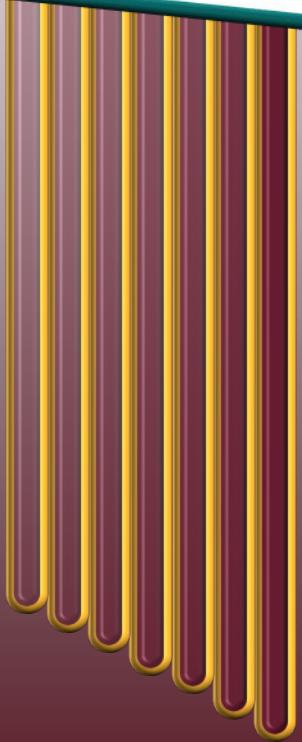


Reference: Zhao, H., Feng, L., Zhang, M., & Zhu, M. (2026). From Long-Term Behavioral Logic to LLM-Assisted Situational Abstraction: Automatic Coding of Collaborative Problem Solving. In Proceedings of the 2026 ICAIE.

Choosing Analysis Approach(es)

	Approach	Tools & Packages
Descriptive	Clustering & topic modeling Anomaly & similarity	scikit-learn, gensim, BERTopic PyOD, sentence-transformers
Predictive	Scoring & labeling Sentiment prediction	scikit-learn, XGBoost, HuggingFace transformers, VADER
Relational	Sequence & process Network analysis	PM4Py, bupaR, hmmlearn, seqHMM NetworkX, igraph, rENA
Crosscutting	Speech & vision Generative AI (zero-shot to fine-tuned)	Whisper, Tesseract, vision transformers OpenAI, DeepSeek, HuggingFace

Section 5: Psychometric Considerations



Section Learning Objectives

Section 5: Psychometric Considerations

Learners will be able to frame validity as an argument and identify methods for detecting bias.

Learners will be able to identify key practical considerations for reporting and governance.

Learners will be able to describe how to assess the reliability of LLM-based measurement.

Why Unstructured Data Raises New Quality Questions

1. The label may be constructed: “truth” is coded by humans from open content, so coding reliability becomes its own measurement problem.
2. The scorer may be generative: LLM outputs vary across prompts, seeds, and model versions, so the scorer itself is a source of error.
3. The signal may be latent: thousands of features (tokens, audio frames, image regions, log events) entangled with surface form.
4. The mistakes may cascade: Errors and biases propagate through the pipeline.

The Label Is Constructed: Coding Reliability

For unstructured data, “truth” is often coded by humans reading / watching / listening to open content. Coding quality is the ceiling an inference model is compared against.

Quantify coder agreement

- **Inter-coder reliability:**
Krippendorff's $\alpha = 1 - \frac{D_o}{D_e}$, where D_o is observed disagreement and D_e is expected disagreement
- **Cohen's κ** (two coders) / **Fleiss' κ** (many).
- **Adjudication:** a third rater, consensus rater, or an expert to resolve disagreement.
- **Codebook quality:** ambiguous categories (i.e., a noisy target) can yield low α .

Metric by code type

- **Ordinal:** Quadratically weighted kappa, exact and adjacent %.
- **Nominal:** macro- $F1$ (harmonic mean of precision and recall), Cohen's or Fleiss's κ (chance-corrected agreement), confusion matrix (a true-vs-predicted count table).
- **Spans** (e.g., evidence highlighting, a step in a proof): token / span-overlap $F1$ score.

References:

Krippendorff, K. (2018). Content analysis: An introduction to its methodology (4th ed.). SAGE Publications.
Gwet, K. L. (2014). Handbook of inter-rater reliability: The definitive guide to measuring the extent of agreement among raters (5th Edition). AgreeStat Analytics.

Reliability of LLM-Based Measurement (1/3)

- A generative scorer is **non-deterministic and prompt-dependent**, so classical “test-retest” becomes **re-run** and **re-prompt** reliability.
- One way to think about it is to treat run, prompt, and order as facets (in Generalizability Theory) and decompose the score variance:

$$\sigma_{total}^2 = \sigma_{person}^2 + \sigma_{run}^2 + \sigma_{prompt}^2 + \sigma_{\varepsilon}^2$$

1. **Run-to-run.** The same input, called repeatedly, returns different scores; quantify how much of the spread is per-call noise vs. true person differences.

Estimate: take R draws per response and fit a one-way random-effects

ANOVA: $\hat{\sigma}_{run}^2 = MS_{within}$, $\hat{\sigma}_{person}^2 = (MS_{between} - MS_{within})/R$.

Reportables: σ_{run}/CV and $ICC = \frac{\sigma_{person}^2}{\sigma_{person}^2 + \sigma_{run}^2}$

Reliability of LLM-Based Measurement (2/3)

- 2. Prompt sensitivity.** Reword the prompt (including rubric and instructions) while keeping the meaning fixed; a trustworthy measure should be stable under semantically equivalent paraphrases

Estimate: over T meaning-preserving paraphrases estimate σ_{prompt}^2 prompt (or pairwise QWK across variants).

- 3. Self-consistency.** The flip side of run-to-run: rather than just measuring the per-call noise σ_{run}^2 , *exploit* replication: draw m outputs $s^{(1)}, \dots, s^{(m)}$ on the same input and aggregate them into $\hat{s}$ to obtain a more stable score.

Reliability of LLM-Based Measurement (3/3)

Not every threat is a variance source; some are systematic biases. For instance:

- **Order effects.** In LLM-as-judge setups the model can favor whichever option is presented first; detect it by swapping the order.
- **Central tendency.** The model may score systematically high or low, or cluster ratings near the scale midpoint.

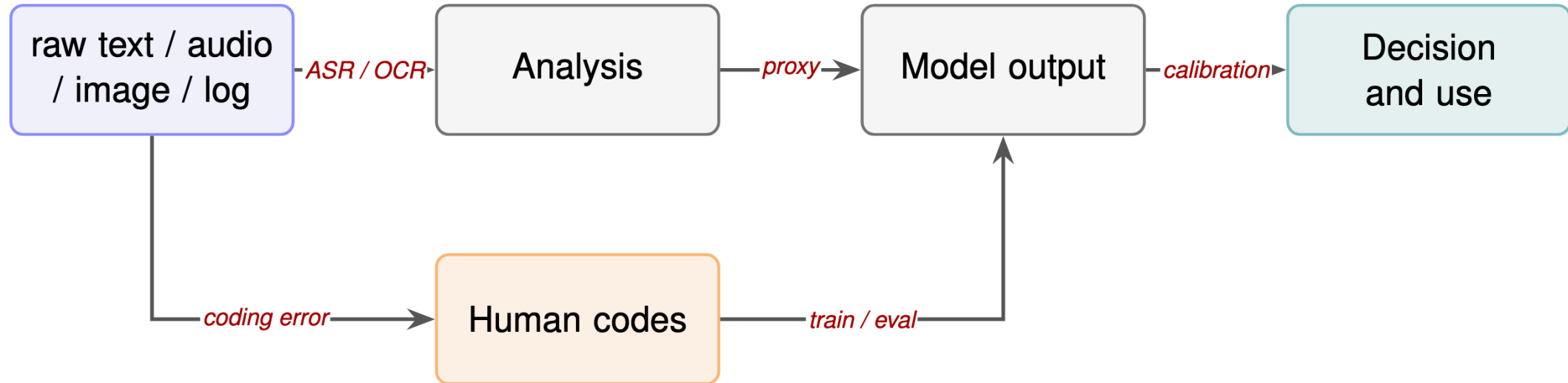
Reliability and bias are distinct: shrink *variance* by averaging replications $E(\rho^2)$, but remove bias by design (such as randomize order, balance positions, and audit against human labels.)

Does the Signal Carry the Construct?

Representation Validity: Validity starts from representation (text embeddings, audio features, image patches, log sequences...).

- **Probing**: can a simple classifier recover the construct from the representation? Can it *also* recover an unwanted attribute (author L1, speaker gender, school)?
- **Ablation**: drop length / keywords / a modality / a feature family; if performance barely moves, the construct wasn't doing the work.
 - **Multimodal alignment**: in classroom video, are speech, gesture, and gaze all contributing, or is one channel carrying everything?

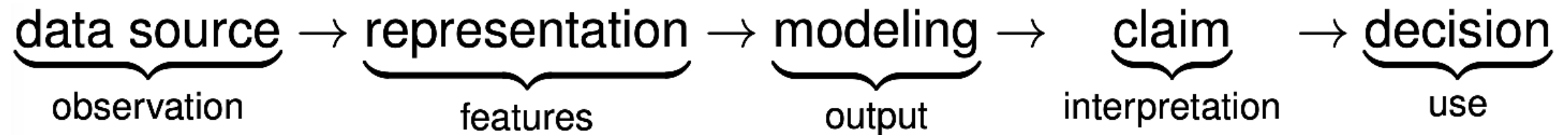
When Errors Cascade



ASR / OCR: transcription & recognition mistakes when digitizing audio/images; **coding error:** unreliable human labels; **train / eval:** noisy labels propagate into the fitted model; **proxy:** the model keys on surface form (length, volume) not the construct; **calibration:** scores mis-scaled or biased before they drive a decision.

Validity as an Argument

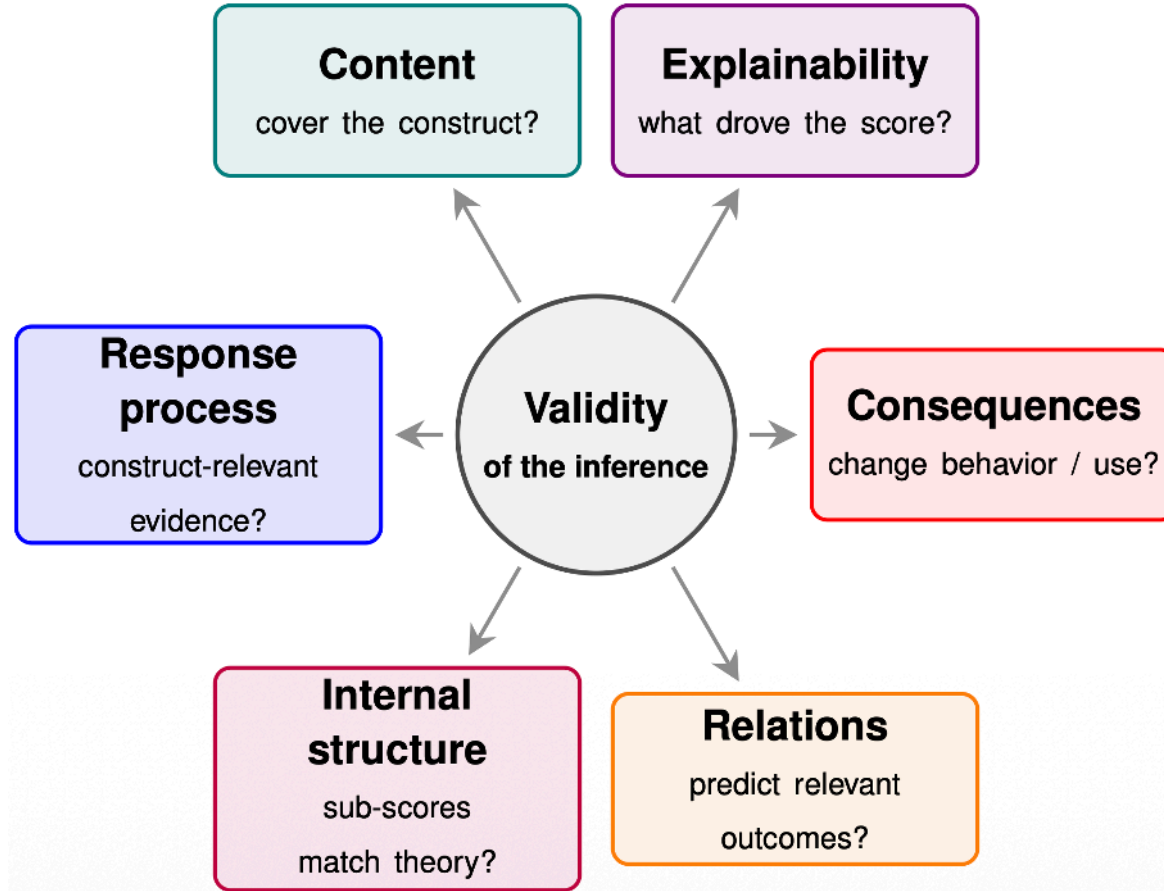
Validity concerns the inference, not the model.



- For example: A high quadratically-weighted kappa (QWK) value supports the modeling link only but not interpretation or use.
- Evidence should be gathered and challenged at each link.

Validity Source

Validity is built from multiple sources of evidence.



Key References

- *Standards for educational and psychological testing* (AERA, APA, NCME, 2014)
- *Validating the interpretations and uses of test scores* (by M. Kane, JEM, 2013)
- *Responsible AI for measurement and learning: Principles and practices* (By M. Johnson, ETS, 2025)

The Proxy Trap: Surface Form vs. Construct

- The validity threat: the model scores a **surface proxy** that correlates with the construct in training but does not actually represent it.
- Counterfactual edits, adversarial inputs, and feature ablations help expose whether construct or proxy drives the output.

Data	Proxy	What it should measure
Essays	length, sophisticated words, shell language	Argument & evidence quality
Spoken responses	speaking rate, loudness	Fluency & content
Code	passes the test suite	correct reasoning, not memorized
Handwritten math	neatness, legibility	valid solution steps
Forum posts	post count, emoji, length	substantive contribution
Tutoring logs	clicks, time-on-page	Strategy & learning

Source of Bias in Unstructured Data

Scores derived from unstructured student data carry identity-linked surface cues, so bias can enter easily and threaten the fairness of the inference. For examples :

- **Text:** dialect, register, code switching, second-language-learner status confounded with “error” features.
- **Speech:** ASR word-error rate and scoring quality degraded for accents, dialects, non-native speakers.
- **Vision:** handwriting, face, gesture models vary by skin tone, script, lighting, camera quality etc.
- **Access:** device, bandwidth, assistive-tech differences shape the captured signal.

Bias Detection

Let M be the mode (predicted) score, T the human criterion rating or true score, and G the student group:

- **Subgroup performance gaps:** examine model performance *within* each $G = g$.
- **Differential Item Functioning (DIF):** at *matched proficiency* T , test whether the score depends on group, $E[M|T, G]$ varies with G (Mantel-Haenszel or logistic-regression DIF).
- **Differential prediction:** regressing an external criterion on the score, test whether $E[T|M, G]$ varies with G .
- **Measurement invariance:** multi-group CFA.
- **Label bias:** human rating bias exists in the criterion T and propagates into training.

Conflicting Criteria

- There is no single metric of “fairness.”
- “Fairness” is not one property but several competing statistical requirements
- It can be operationally defined differently.

References:

Dorans, N. J., & Cook, L. L. (Eds.). (2016). Fairness in Educational Assessment and Measurement (1st ed.). Routledge.

<https://doi.org/10.4324/9781315774527>

Bennett, R. E. (2026): Fairness and Postsecondary Admissions Testing: A Selective History, Applied Measurement in Education.

DOI:10.1080/08957347.2026.2674562

Fairness Definitions

Let M be the predicted score (or decision), T the target or true score, and G the student group:

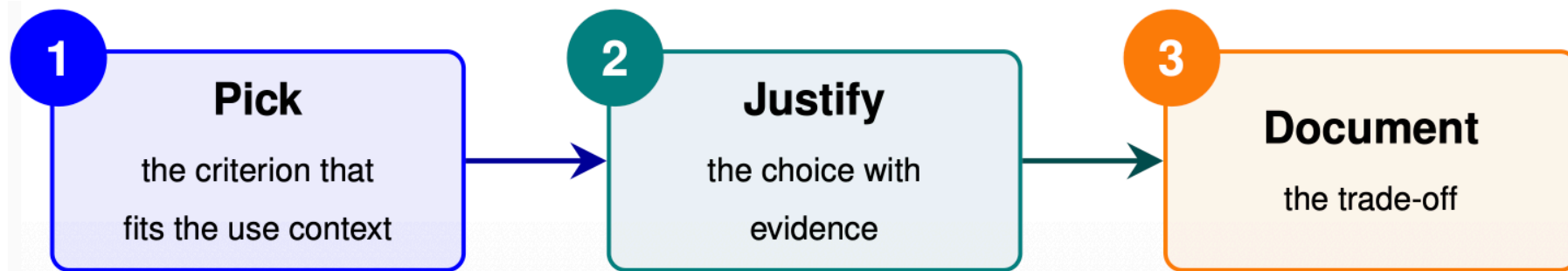
- **Independence** (demographic parity), $M \perp\!\!\!\perp G$: Predicted score M is independent of subgroups G .
 - $P(M = 1|G = g)$ equal across g .
- **Sufficiency** (calibration within groups), $T \perp\!\!\!\perp G \mid M$: The target T is conditionally independent of subgroups G given the predicted score M .
 - $P(T = 1|M = m, G = g) = m \forall g$.
- **Separation** (equalized odds / equal error rates), $M \perp\!\!\!\perp G \mid T$: Predicted score M is conditionally independent of subgroups G given the target T .
 - $P(M = 1|T = t, G = g)$ equal across g for $t \in \{0,1\}$.

Practical Implication

- These fairness criteria cannot be simultaneously satisfied.

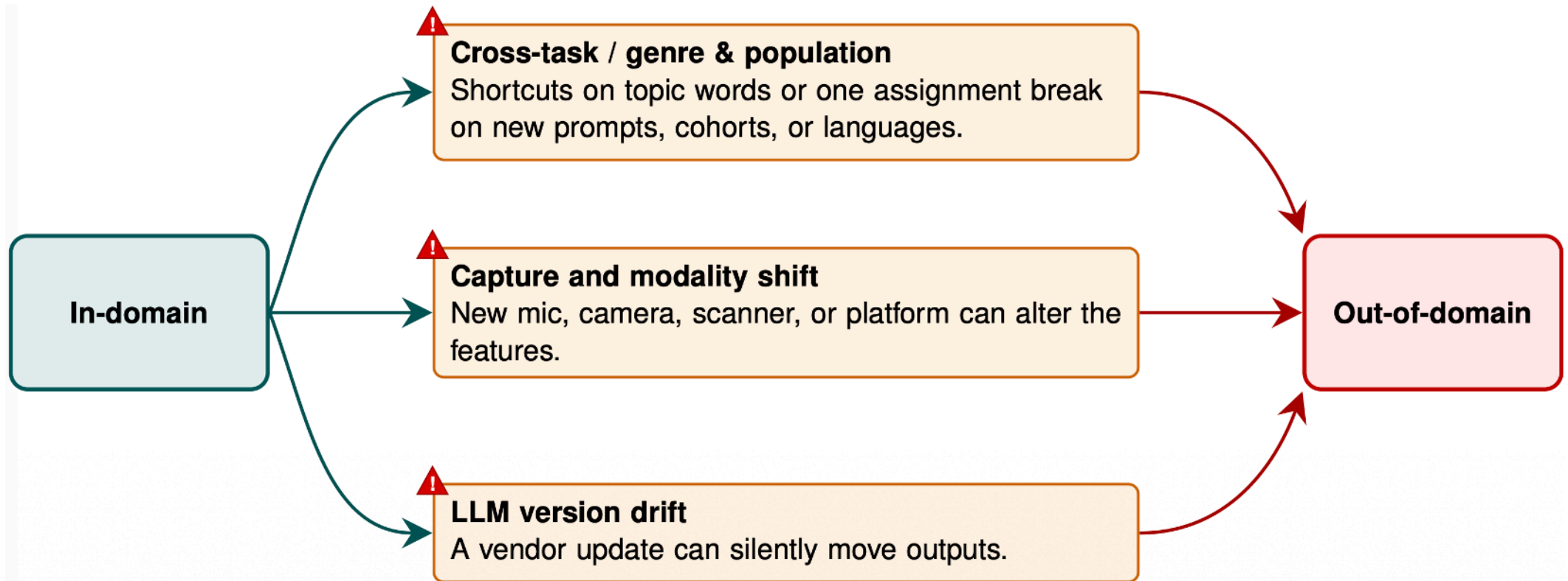
$$P(T = 1|G = g) \neq P(T = 1|G = g')$$

- They are all satisfied together only in trivial cases
 - Perfect prediction, or
 - Equal base rates across groups.



Test Generalization, Don't Assume It

- A measure derived from unstructured data will not necessarily hold up outside the conditions it was built in.

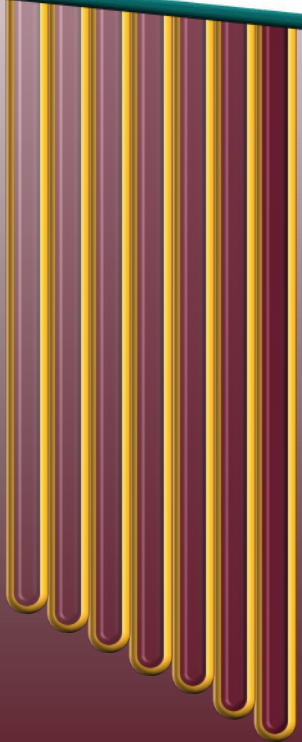


Reporting & Governance

Keep every judgment made in the pipeline auditable, and bound what it is allowed to claim according to evidence.

- **Model & data cards:** document intended use, target populations, subgroup performance, and known proxy and failure modes.
- **Human-in-the-loop:** route low-confidence / atypical cases (off-topic essays, heavy-accent audio, messy handwriting, novel code) to human review.
- **Provenance & reproducibility:** store raw data, capture data collection settings, codebook version, prompts to LLMs, model & package versions, seeds, decoding settings.

Section 6: Privacy, Consent, and Governance



Section Learning Objectives

Section 6: Privacy, Consent, and Governance

Learners will be able to explain what makes unstructured data sensitive.

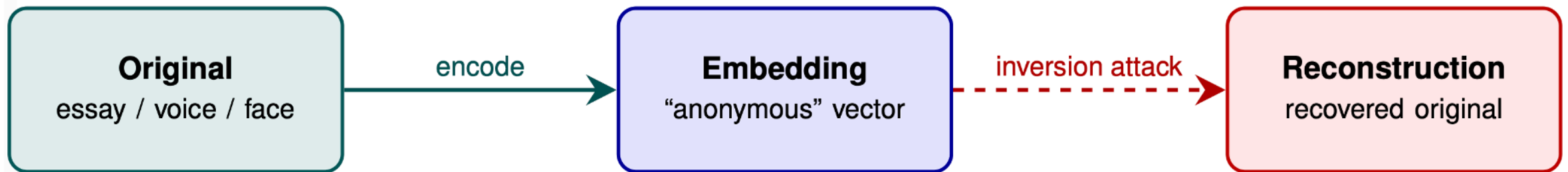
Learners will be able to describe the legal and ethical landscape.

Learners will be able to describe what responsible use entails.

Privacy Is a Measurement-Quality Issue

Unstructured data is often far more *identifying* and *sensitive* than a scored item response.

- A scored response reveals little; but the raw essay or voice clip can reveal identity, health, background, and affect.
- Even “anonymous” vectors are not safe
 - Techniques like **inversion attacks** can reconstruct the original text, voice, or face from its embedding.



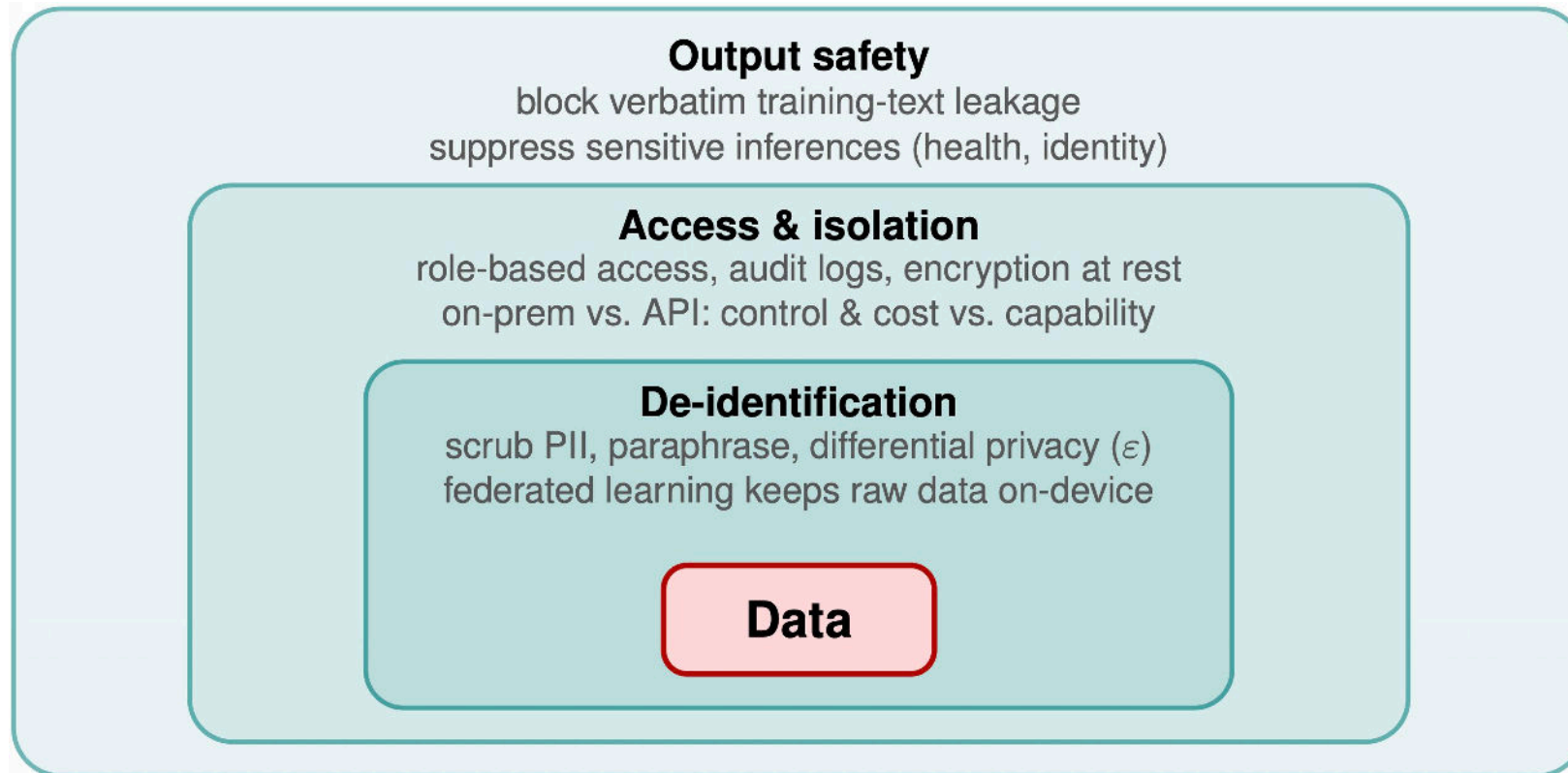
What Makes Unstructured Data Sensitive

The signal that powers a measure is often the very signal that identifies a person.

- **Re-identification by modality:** audio carries voice prints; video shows face & gesture; behavioral patterns are biometric.
- **Quasi-identifiers:** even though no field names the student, essay topic + classroom + date can uniquely pick one out.
- **Special groups:** minors (FERPA, COPPA), disability accommodations, and mental-health or safety signals surfacing in free text demand heightened protection.

Defense in Depth: Technical Safeguards

- No layer is sufficient alone.
- Combine *minimization*, *access control*, and *formal privacy* in proportion to the stakes.



Legal and Ethical Landscape

- **Regulatory frameworks:** include FERPA, COPPA, GDPR, state laws (CA CCPA, IL BIPA), and IRB review.
- **Third-party APIs:** know where data goes, such as retention, sub-processors, and whether training opt-out is available.

Consent is Foundation

- ✓ **Informative:** Use plain-language, age-appropriate, layered notice of what is collected and why.
- ✓ **Specific:** Consent to one is not consent to all. Opt-in for each secondary use (model training, research, sharing...).
- ✓ **Revocable:** offer a true withdrawal path, with downstream deletion that actually propagates.

Responsible Use

- Unstructured responses are richer and vulnerable to identification.
- Governing them well is part of measurement quality.
- Handling the raw data deserves the same care we give to validity and fairness.

Data Discipline

Govern the raw data.

- Purpose limitation: no scope creep
- Explicit retention schedules
- Limit copies & access

Accountability

Document data treatment and analysis.

- Audit results for measurement and psychometric quality
- Be clear what it does and doesn't measure

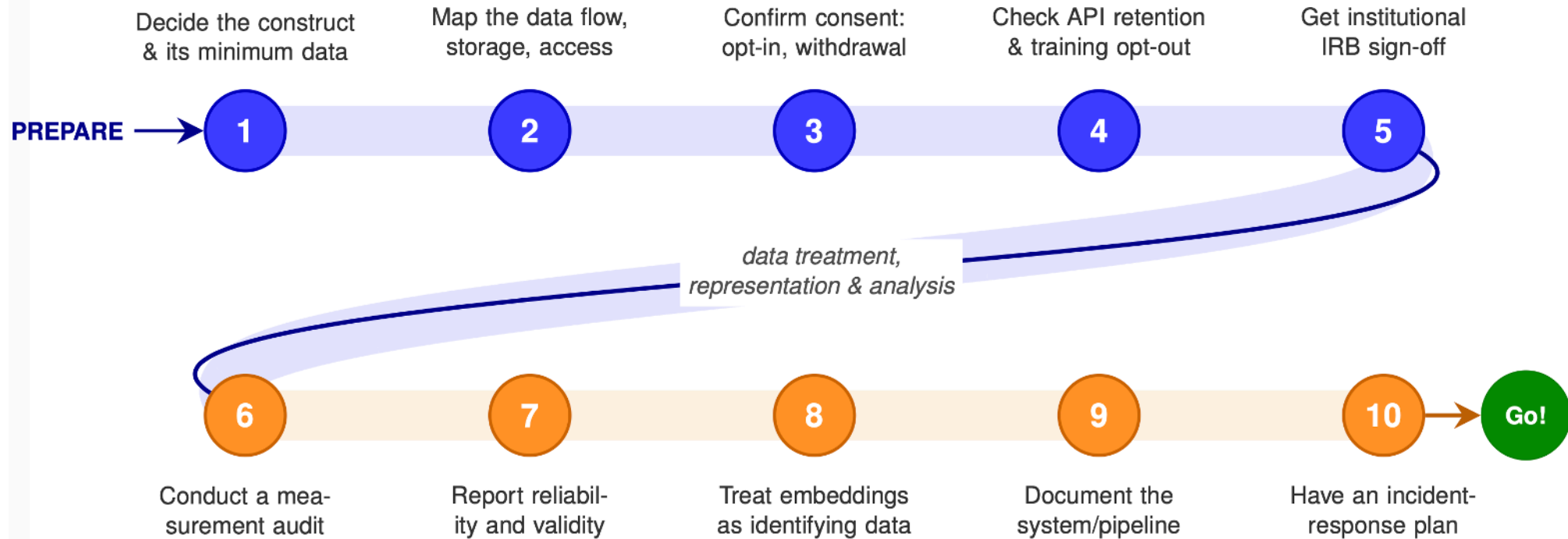
Incident Response

Plan for when it fails.

- Detect → Notify → Remediate → Re-validate

A Quick Governance Checklist

BEFORE YOU START



BEFORE YOU PUBLISH or DEPLOY

Acknowledgements

- Thank you for joining us on this journey.
- We thank the NCME ITEMS editor, Dr. Stella Kim, as well as reviewers Dr. Danhui Zhang (Beijing Normal University), and Dr. Michael Fauss and Dr. Daniel McCaffrey (Educational Testing Service), for their feedback and suggestions.
- We thank Feng Li, Han Zhao, and Zongting Ge from University of Science and Technology of China for their assistance in setting up the repository.