Active Learning Block Seminar

Ahmad Dawar Hakimi

04.08.2025

Seminar Overview

Day	Morning (9:00-12:00)	Afternoon (13:00-16:00)
Day 1	Outline, Motivation Foundations, Query Strategies	Coding - Query Strategies
Day 2	Deep Active Learning Practical Considerations	Coding - Deep Active Learning
Day 3	Active Learning in the Era of LLMs	Coding - LLMs in AL Cycle
Day 4	Prepare Pitch	Pitches + Q&A

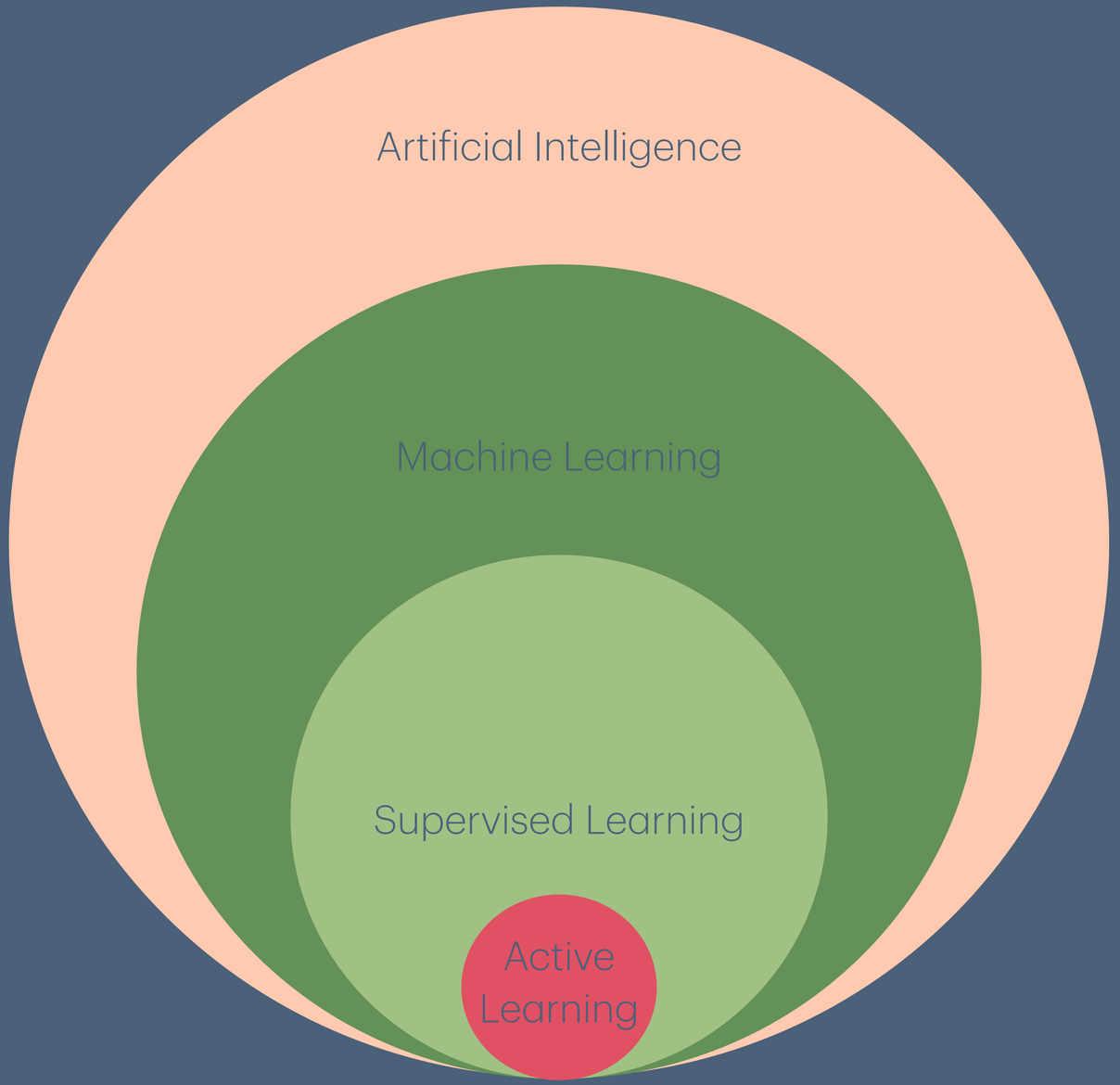
Examination

- Can be used as WP2 or WP3
- Pitch on the last day of the Block Seminar (10+5)
- Option 1:
  - 10 Page Paper (25000 Characters) about a Research Question about Active Learning
- Option 2:
  - Coding Project + Short Report (2-4) Pages
- **Submission Deadline:** End of the semester break.

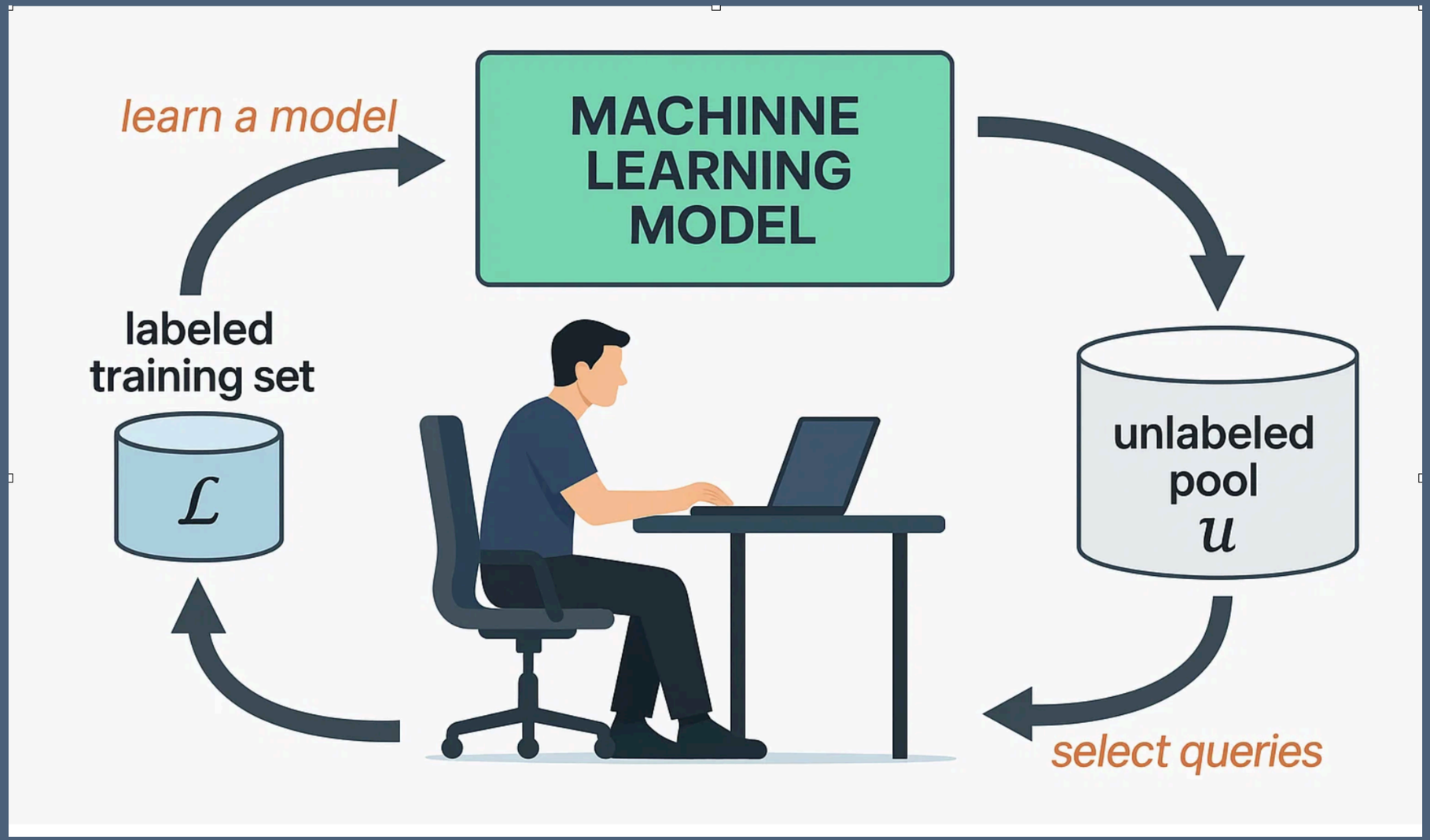
Day 1 - Active Learning Foundation

- Learning Objectives
  - What is Active Learning?
  - Why Active Learning?
  - AL Paradigms
  - Active Learning Cycle
  - Query Strategies

Active Learning



What is Active Learning?



Why Active Learning?

- **High Annotation Costs:**
  - Labeling data often demands domain experts or paid crowdworkers, straining both budget and schedule.
- **Diminishing Returns:**
  - Performance gains plateau as more labeled data are added, leading to wasted effort on uninformative examples.
- **Improved Data Efficiency:**
  - By selecting only the most informative instances, Active Learning accelerates model improvement with far fewer annotations.
- **Real-world Bottlenecks:**
  - Low-resource languages, niche domains (e.g. medical, legal), and rapidly evolving tasks often lack the large labeled corpora that traditional learning requires.
- **Budget & Time Constraints:**
  - Adaptive querying maximizes the impact of every labeling dollar and cuts down annotation turnaround.
- **Scalability:**
  - Facilitates efficient use of annotation platforms, human-in-the-loop pipelines, and expert oversight.

Annotation Exercise - Sentiment Analysis

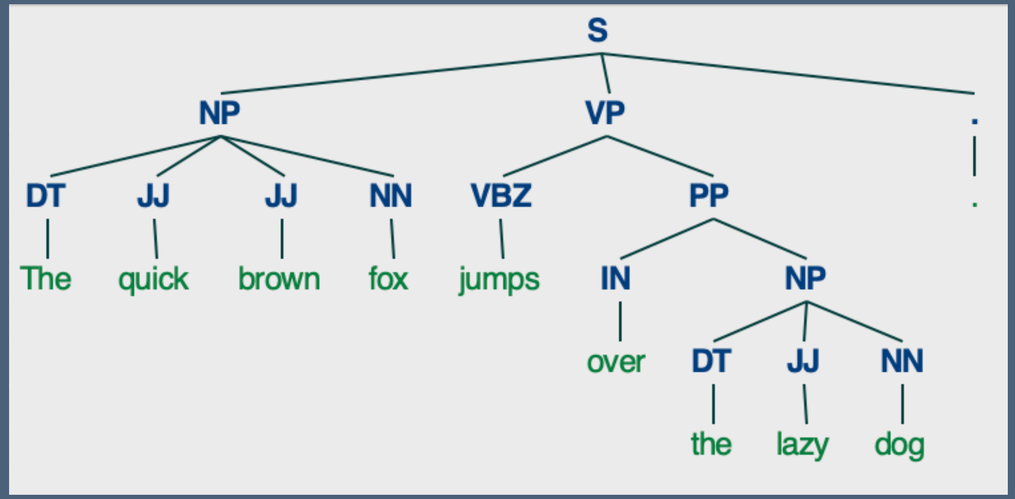
For each sentence below, decide whether the overall sentiment is Positive, Neutral, or Negative.

I love this movie	-> Positive
Not bad at all	-> Positive
Could be better, could be worse	-> Neutral
I’m not unhappy with the results	-> Positive
The plot twist was sick	-> Positive

Annotation Exercise -Part of Speech

For each sentence tag the part of speech tags from the Penn Treebank

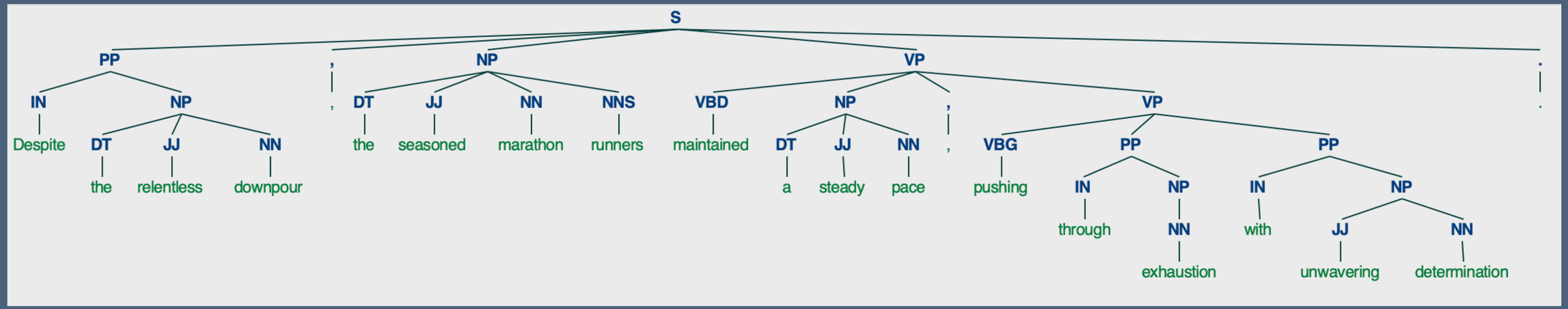
The quick brown fox jumps over the lazy dog.



Tag legend (Penn Treebank):
- DT = determiner
- JJ = adjective
- NN = noun, singular
- NNS = noun, plural
- VBZ = verb, 3rd-person singular present
- VBD = verb, past tense
- VBG = verb, gerund/present participle
- IN = preposition/subordinating conjunction
- , = comma
- . = sentence-final period

Annotation Exercise -Part of Speech

Despite the relentless downpour, the seasoned marathon runners maintained a steady pace, pushing through exhaustion with unwavering determination.



Tag legend (Penn Treebank):
- DT = determiner
- JJ = adjective
- NN = noun, singular
- NNS = noun, plural
- VBZ = verb, 3rd-person singular present
- VBD = verb, past tense
- VBG = verb, gerund/present participle
- IN = preposition/subordinating conjunction
- , = comma
- . = sentence-final period

Annotation Exercise - Named Entity Recognition

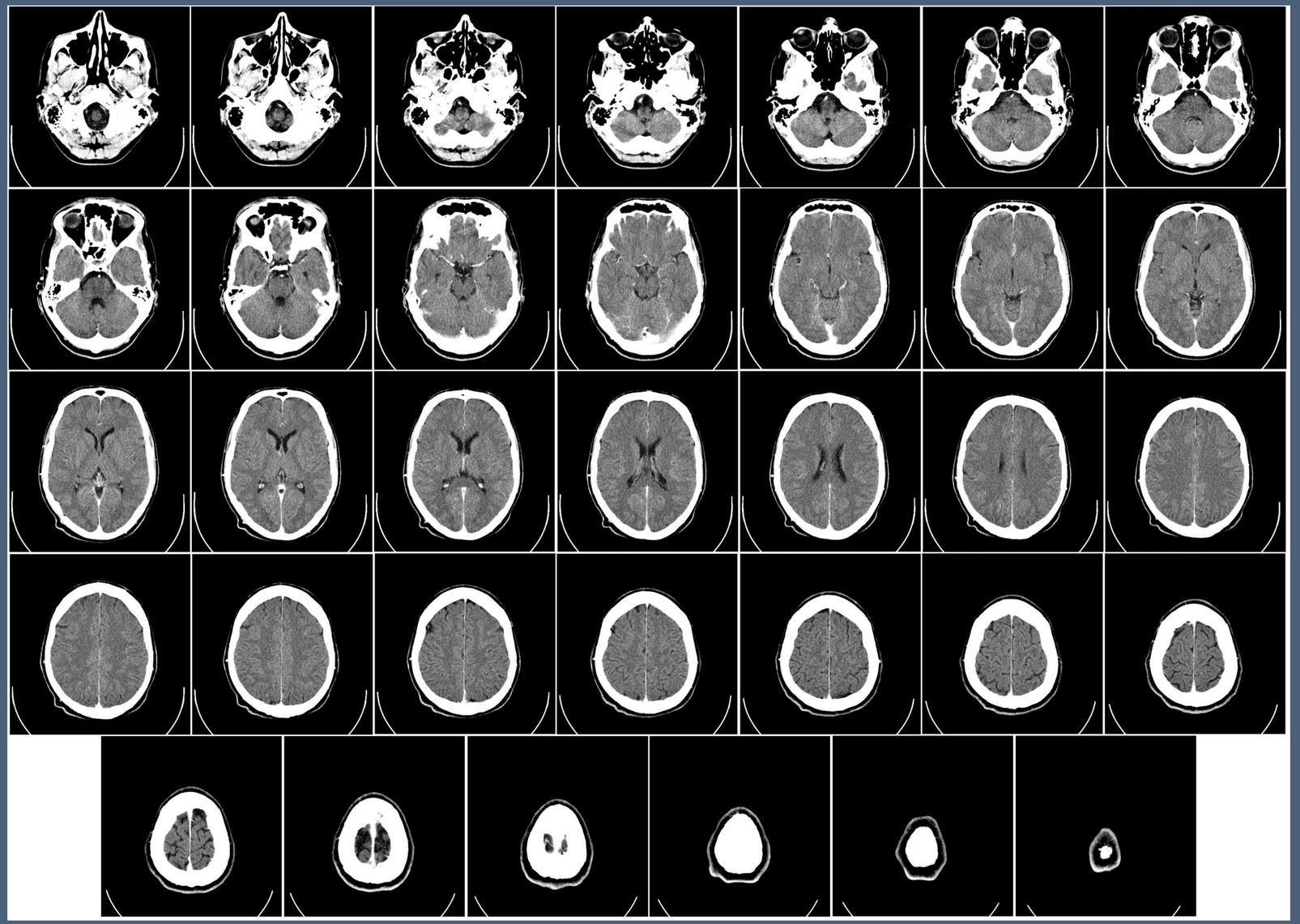
Person			Date	GPE
Barack Obama	gave a speech in		2008	in Berlin

Annotation Exercise - Named Entity Recognition

The BRICS (ORG) summit, held in Johannesburg (GPE), South Africa (GPE) from August 22–24, 2024 (Date (Span)), saw participation from leaders of Brazil (GPE), Russia (GPE), India (GPE), China (GPE), and South Africa (GPE).

Which examples to annotate?

Which examples to annotate?



What to annotate?

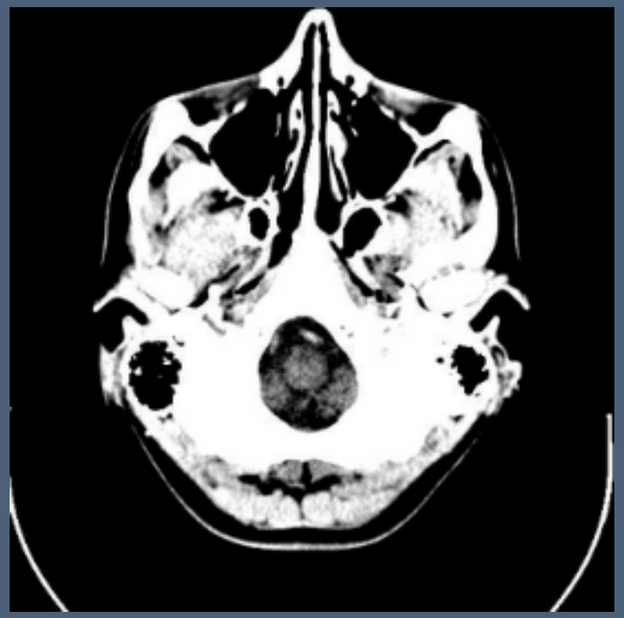
What to annotate?

What to annotate?

What to annotate?

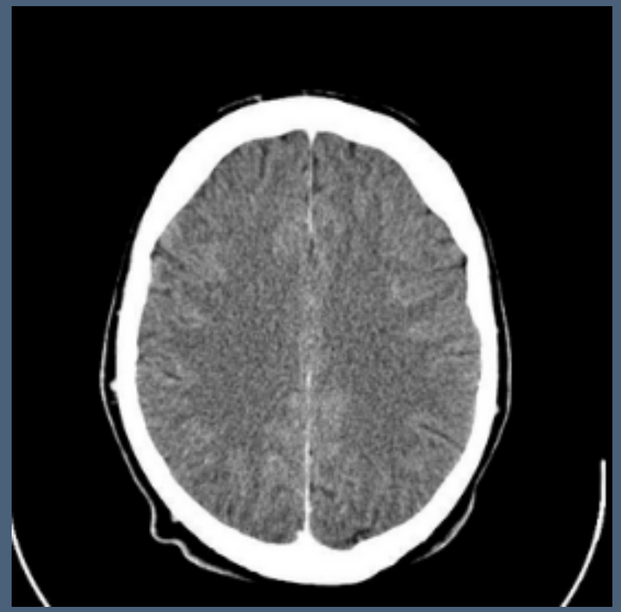
What to annotate?

What to annotate?



What to annotate?

What to annotate?



Which AL Paradigms did you observe?

AL Paradigms

Membership Query Synthesis

model generates a query de novo

Stream-based Sampling

Instance space or input distribution

Sample an instance

model decides to query or discard

Pool-based Sampling

Sample a large pool of instances

model selects the best query

Membership Query

- Create new, synthetic instances tailored to the problem space
- Query the oracle for labels on generated samples
- Pros:
  + Unlimited exploration: Not constrained by an existing data pool
  + Class balance: Can generate minority-class or rare examples to mitigate imbalance
  + LLM-enabled generation: Modern large language models can produce high-quality, contextually rich samples
- Cons:
  - Annotation noise: Synthetic samples may be unrealistic or confusing to annotators
  - Oracle refusal: Human oracles might reject nonsensical or ambiguous queries
  - Quality control: Requires filtering or human-in-the-loop validation to avoid degrading model performance

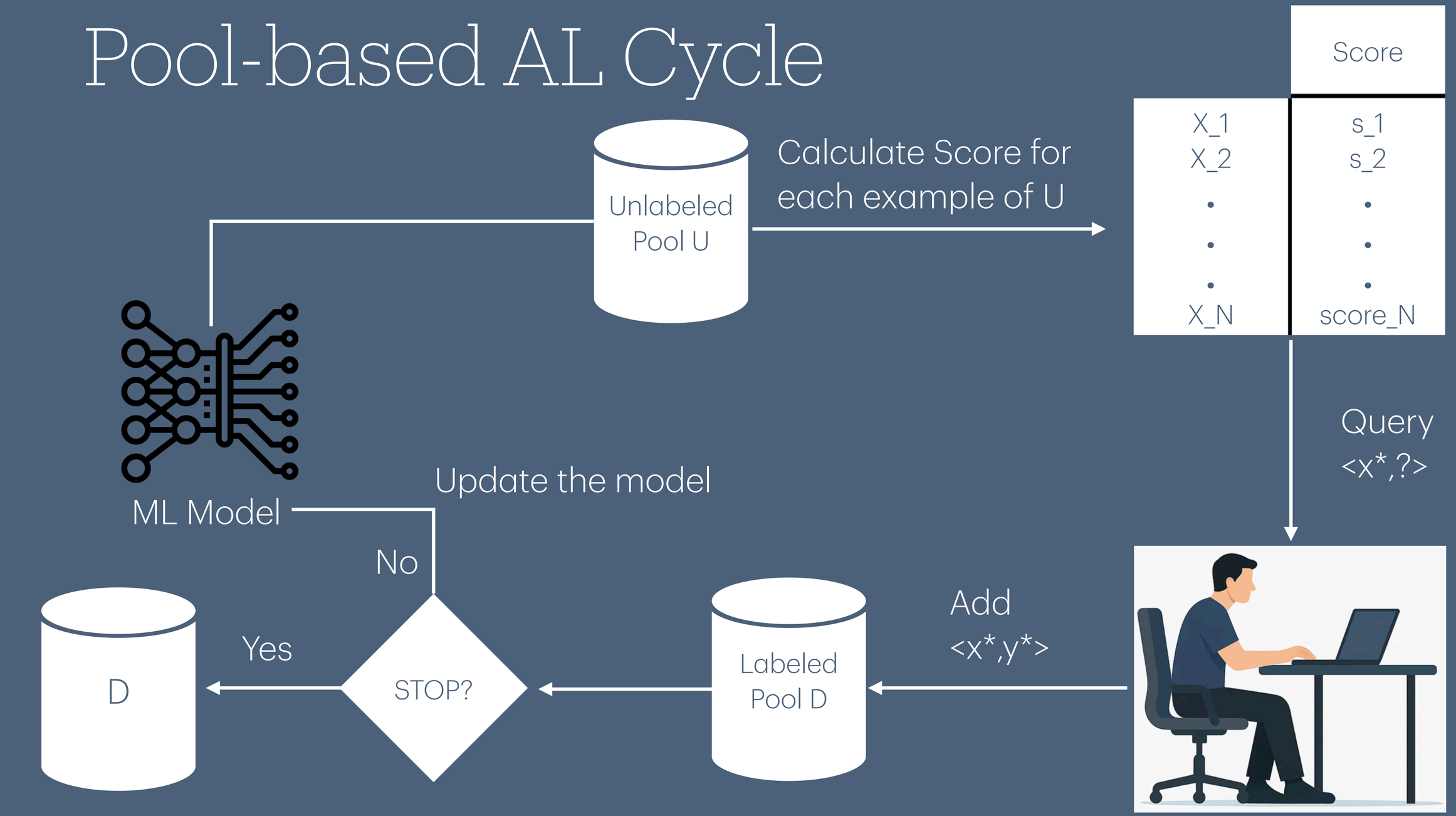
Stream-based Sampling

- Unlabeled instances arrive **one at a time**
- The learner decides **immediately** whether to query its label or discard it.
- Pros:
  + Low memory footprint: Process one instance at a time—no large pool needed.
  + Realistic queries: Always sample actual data, avoiding synthetic artifacts.
  + Adaptive to data rarity: Can capture rare events as they appear in the stream.
  + Immediate feedback loop: Model updates incrementally, supporting online learning.
- Cons:
  - Data drift vulnerability: Fixed decision rules may fail when the underlying distribution shifts.
  - One-shot decision: Once an instance is skipped, it cannot be revisited—even if later deemed highly informative.
  - Threshold tuning: Choosing the right uncertainty threshold or sampling probability often needs careful calibration.

Pool-based Sampling

- The learner has access to a large pool $\mathcal{U}$ of unlabeled data. At each iteration, it selects one or more of the most “informative” examples from $\mathcal{U}$ to query the oracle for labels, then updates the model.
- Pros:
  + Global optimality: You can compare all candidates before querying.
  + Batch flexibility: Efficient for annotation workflows (parallel labeling).
  + Revisitability: Unqueried examples remain available for future rounds.
- Cons:
  - Memory & compute: Must store and often scan the entire pool.
  - Scalability: Large pools can make selection expensive.
  - Redundancy risk: Without diversity controls, may pick near-duplicates.
  - Outlier danger: Pure uncertainty methods can focus on noise.

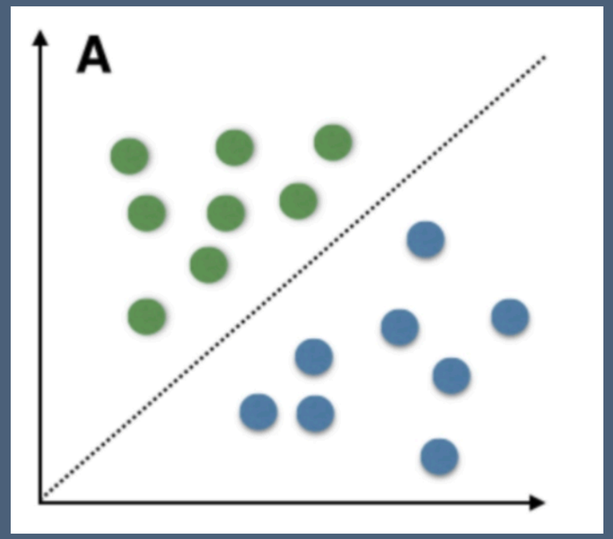
Pool-based AL Cycle



Classification

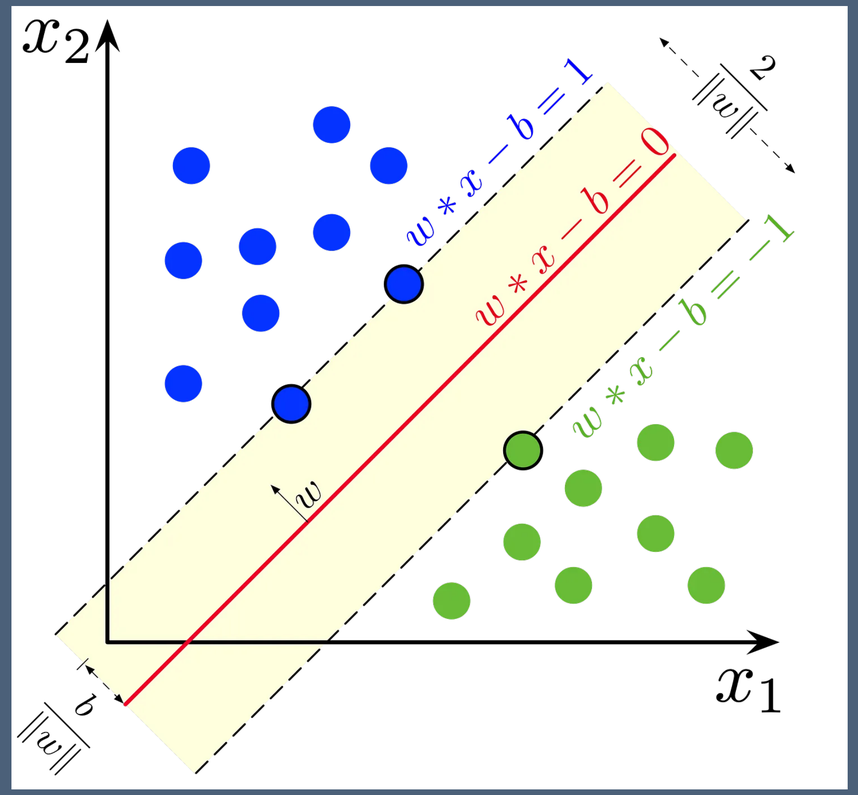
What is classification?

- GOAL: Previously unseen records should be assigned a class from a given set of classe as accurately as possible
- Approach:
  - Given a collection of records (training set)
  - each record contains a set of attributes
  - one of the attributes is the class attribute (label) that should be predicted
  - Learn a model for the class attribute as a function of the values of other attribute
- Variants:
  - Binary classification (e.g. fraud/no fraud or true/false)
  - Multi-class classification (e.g. low, medium, high)
  - Multi-label classification (more than one class per record, e.g. user interests)

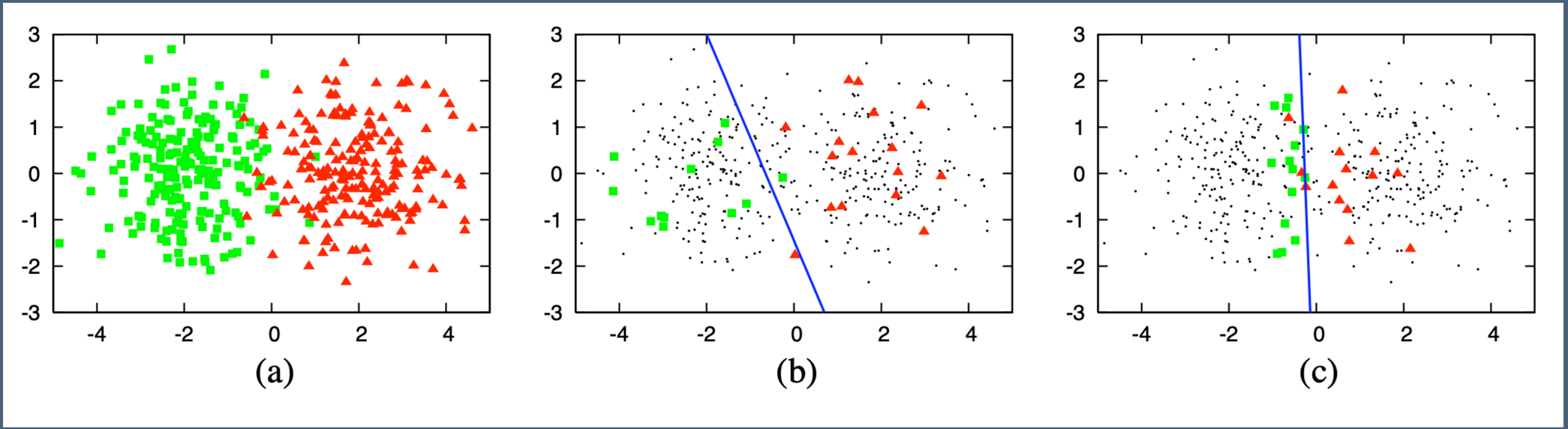


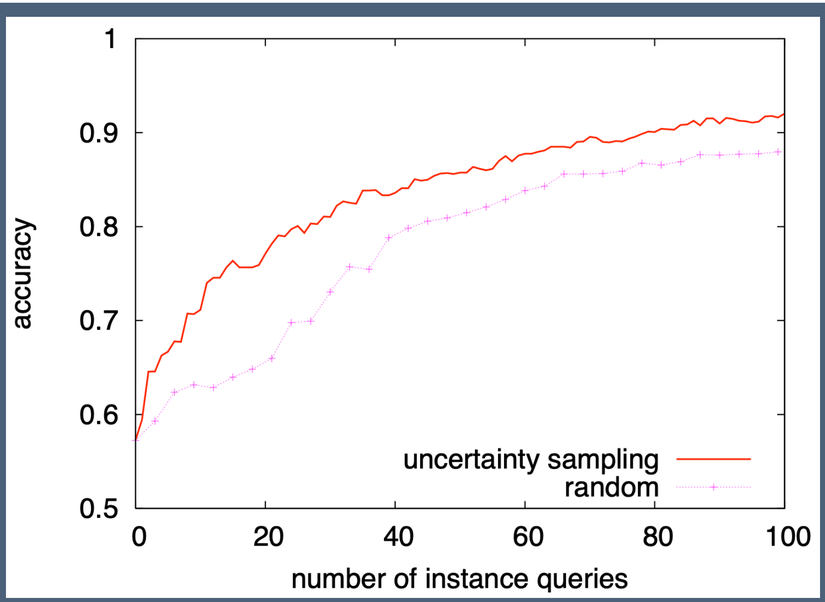
Support Vector Machines [Vapnik, 1964]

- Goal: Widest “Gap” Between Classes
- Decision Boundry & Margins:
  - The hyperplane is defined by $w^{\mathsf{T}}x - b = 0$, where w is a weight vector perpendicular to the plane and b is an offset
- Only points on the boundary or margin determine the hyperplane
- Different Loss and Kernel Methods as extensions



Why Active Learning?





Evalution Metrics

- Accuracy

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

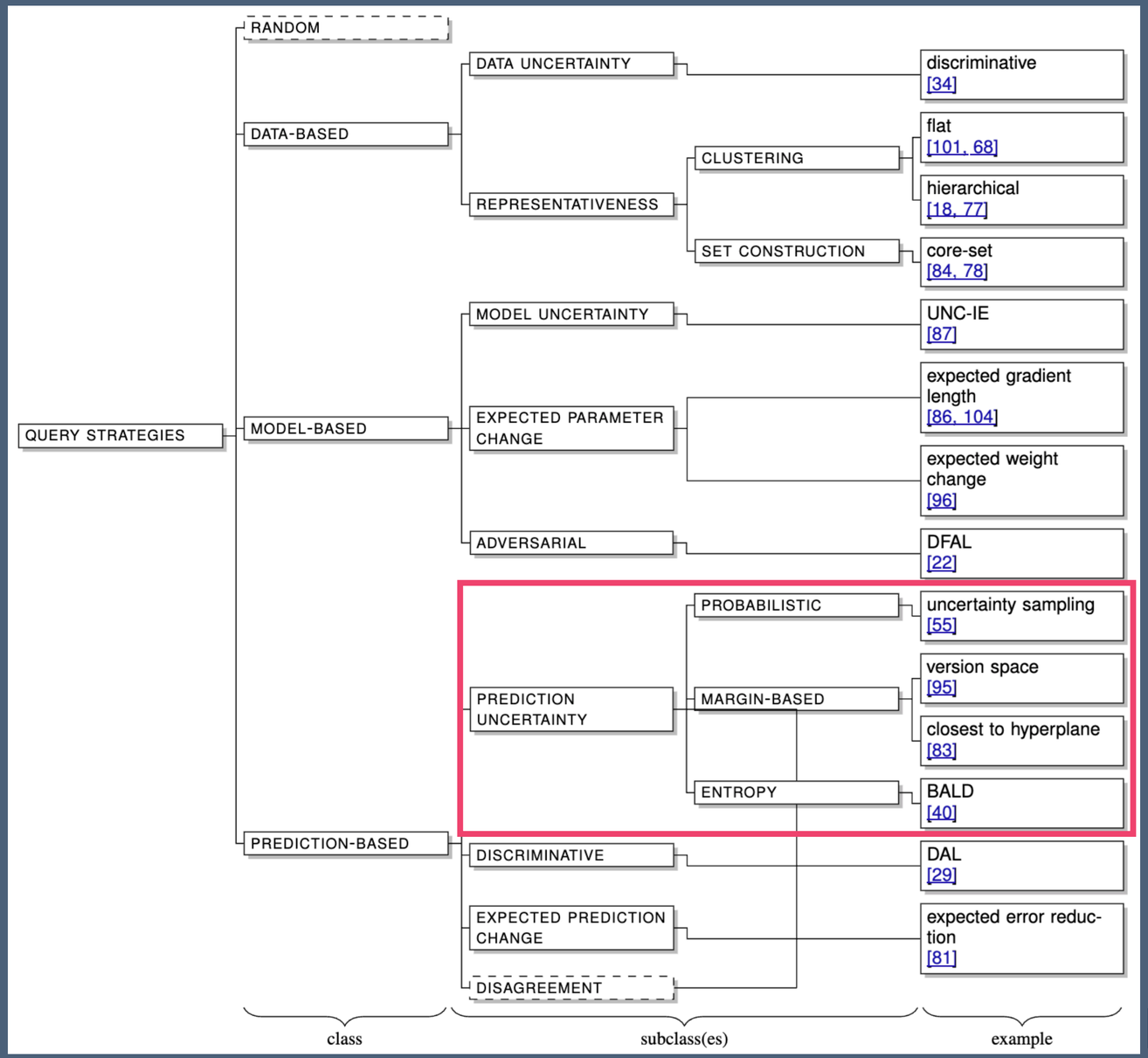
- F1

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F}_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Query Strategy Frameworks

Query Strategy Overview



[Schroeder et Niekler, 2020]

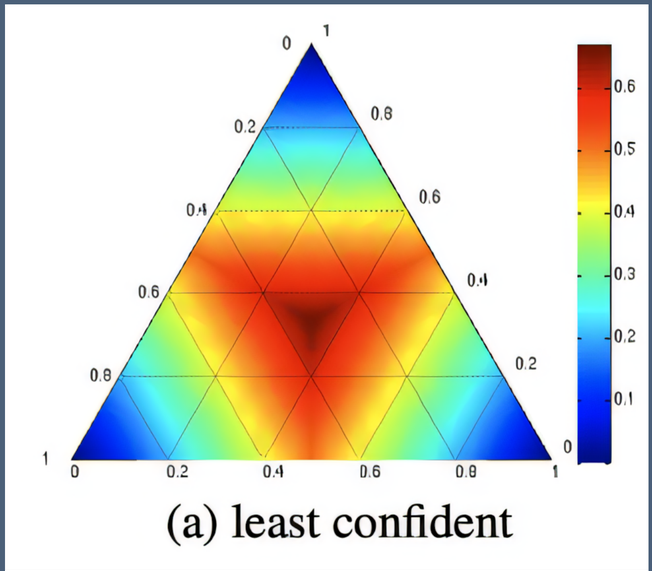
Uncertainty Sampling

- Query instances for which the current model is maximally uncertain
- Pros:
  + Simple to implement for any probabilistic classifier.
  + Often yields steep initial gains in accuracy over random sampling.
  + Works “out of the box” with logistic models, neural nets, etc.
- Cons:
  - Can focus on outliers or noise (model may be uncertain on anomalies).
  - Ignores global data distribution—risk of redundant queries.
  - Requires well-calibrated probability estimates.

Least Confident

- Query instances for which the current model is maximally uncertain

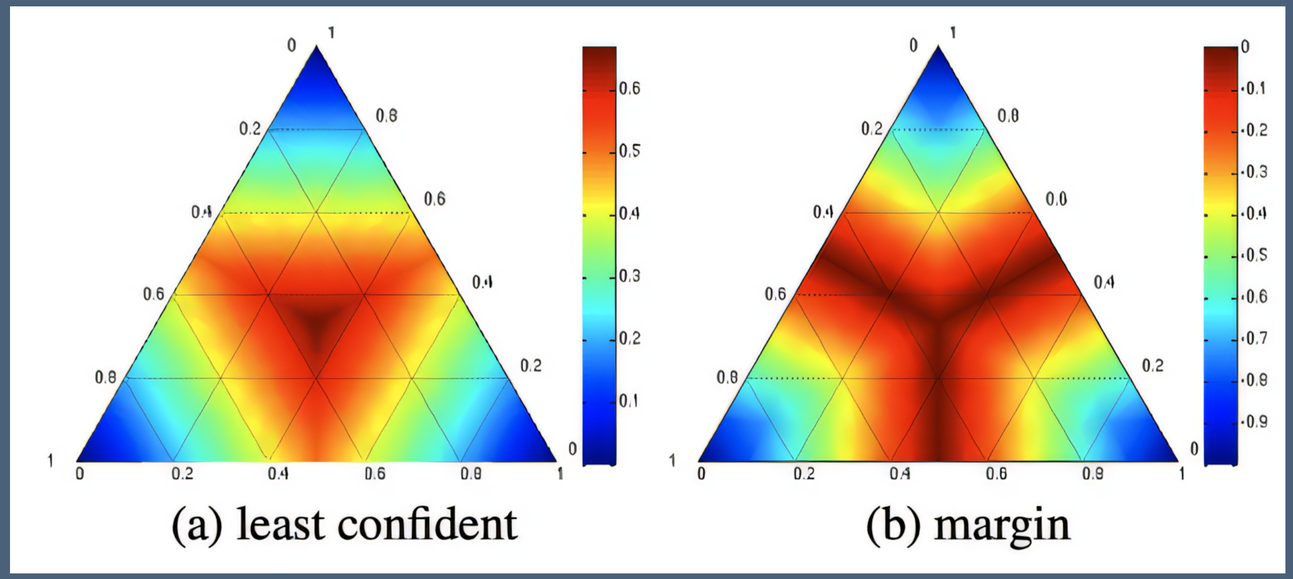
$$x^*_{\mathrm{LC}} = \arg\max_{x \in U} \Big(1 - P_\theta(\hat{y} \mid x) \Big) \quad \text{where} \quad \hat{y} = \arg\max_{y} P_\theta(y \mid x)$$



Margin Sampling

- **Core Idea:** Select the point whose top-2 class posteriors are *closest*

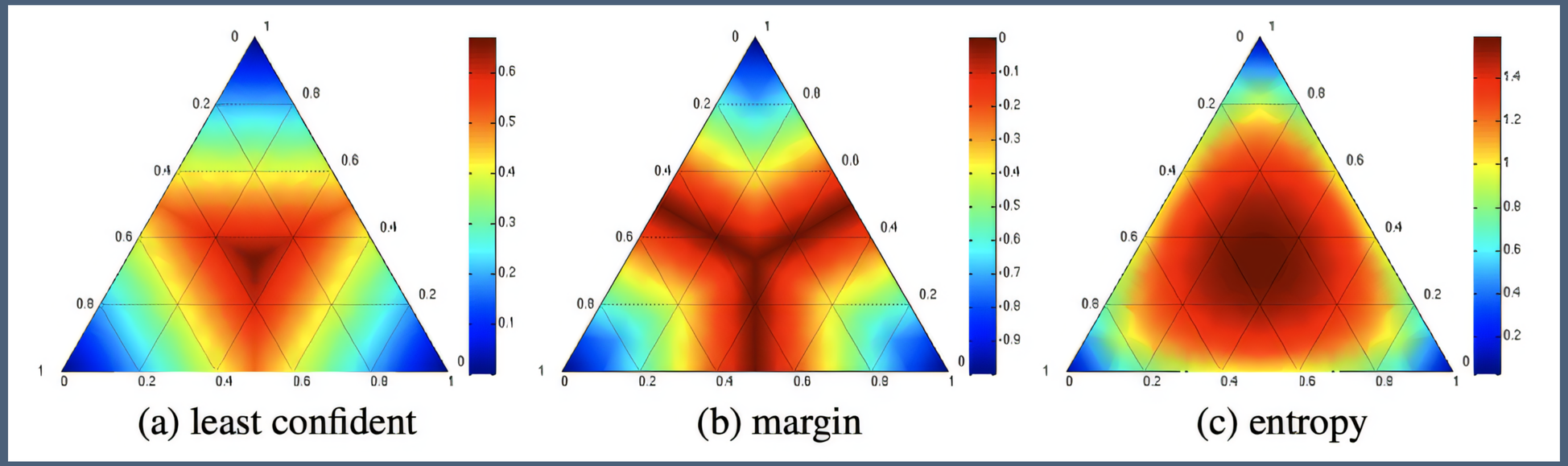
$$x_M^* = \arg\min_{x \in U} \Big(P_\theta(y_1 \mid x) - P_\theta(y_2 \mid x) \Big)$$



Entropy

- **Core Idea:** Query the points with highest posterior entropy, i.e. maximum overall uncertainty across all classes.

$$x_H^* = \arg\max_{x \in U} \left(- \sum_i P_\theta(y_i \mid x) \log P_\theta(y_i \mid x) \right)$$



Uncertainty Sampling - Example

Example	$P(A)$	$P(B)$	$P(C)$	**Least-Confident Score** $1 - \max P$
x_1	0.40	0.40	0.20	
x_2	0.90	0.05	0.05	
x_3	0.33	0.33	0.34	
x_4	0.60	0.30	0.10	

$$x^*_{\text{LC}} = \arg\max_{x \in U} \Big(1 - P_\theta(\hat{y} \mid x)\Big) \quad \text{where} \quad \hat{y} = \arg\max_y P_\theta(y \mid x)$$

Uncertainty Sampling - Example

Example	$P(A)$	$P(B)$	$P(C)$	Least-Confident Score $1 - \max P$	Margin Score $P_1 - P_2$
x_1	0.40	0.40	0.20	1 – 0.40 = 0.60	
x_2	0.90	0.05	0.05	1 – 0.90 = 0.10	
x_3	0.33	0.33	0.34	1 – 0.34 = 0.66	
x_4	0.60	0.30	0.10	1 – 0.60 = 0.40	

$$x^*_{\text{LC}} = \arg\max_{x \in U} \left(1 - P_\theta(\hat{y} \mid x)\right) \quad \text{where} \quad \hat{y} = \arg\max_y P_\theta(y \mid x)$$

$$x^*_M = \arg\min_{x \in U} \left(P_\theta(y_1 \mid x) - P_\theta(y_2 \mid x)\right)$$

Uncertainty Sampling - Example

Example	$P(A)$	$P(B)$	$P(C)$	Least-Confident Score $1 - \max P$	Margin Score $P_1 - P_2$	Entropy $-\sum p \log p$
x_1	0.40	0.40	0.20	1 – 0.40 = 0.60	0.40 – 0.40 = 0.00	
x_2	0.90	0.05	0.05	1 – 0.90 = 0.10	0.90 – 0.05 = 0.85	
x_3	0.33	0.33	0.34	1 – 0.34 = 0.66	0.34 – 0.33 = 0.01	
x_4	0.60	0.30	0.10	1 – 0.60 = 0.40	0.60 – 0.30 = 0.30	

$$x^*_{\text{LC}} = \arg\max_{x \in U} \Big(1 - P_\theta(\hat{y} \mid x)\Big) \quad \text{where} \quad \hat{y} = \arg\max_y P_\theta(y \mid x)$$

$$x^*_M = \arg\min_{x \in U} \Big(P_\theta(y_1 \mid x) - P_\theta(y_2 \mid x)\Big)$$

$$x^*_H = \arg\max_{x \in U} \Big(-\sum_i P_\theta(y_i \mid x) \log P_\theta(y_i \mid x)\Big)$$

Uncertainty Sampling - Example

Example	$P(A)$	$P(B)$	$P(C)$	Least-Confident Score $1 - \max P$	Margin Score $P_1 - P_2$	Entropy $-\sum p \log p$
x_1	0.40	0.40	0.20	1 – 0.40 = 0.60	0.40 – 0.40 = 0.00	≈1.52
x_2	0.90	0.05	0.05	1 – 0.90 = 0.10	0.90 – 0.05 = 0.85	≈0.35
x_3	0.33	0.33	0.34	1 – 0.34 = 0.66	0.34 – 0.33 = 0.01	≈1.59
x_4	0.60	0.30	0.10	1 – 0.60 = 0.40	0.60 – 0.30 = 0.30	≈1.30

$$x^*_{\text{LC}} = \arg\max_{x \in U} \Big(1 - P_\theta(\hat{y} \mid x)\Big) \quad \text{where} \quad \hat{y} = \arg\max_y P_\theta(y \mid x)$$

$$x^*_M = \arg\min_{x \in U} \Big(P_\theta(y_1 \mid x) - P_\theta(y_2 \mid x)\Big)$$

$$x^*_H = \arg\max_{x \in U} \Big(-\sum_i P_\theta(y_i \mid x) \log P_\theta(y_i \mid x)\Big)$$

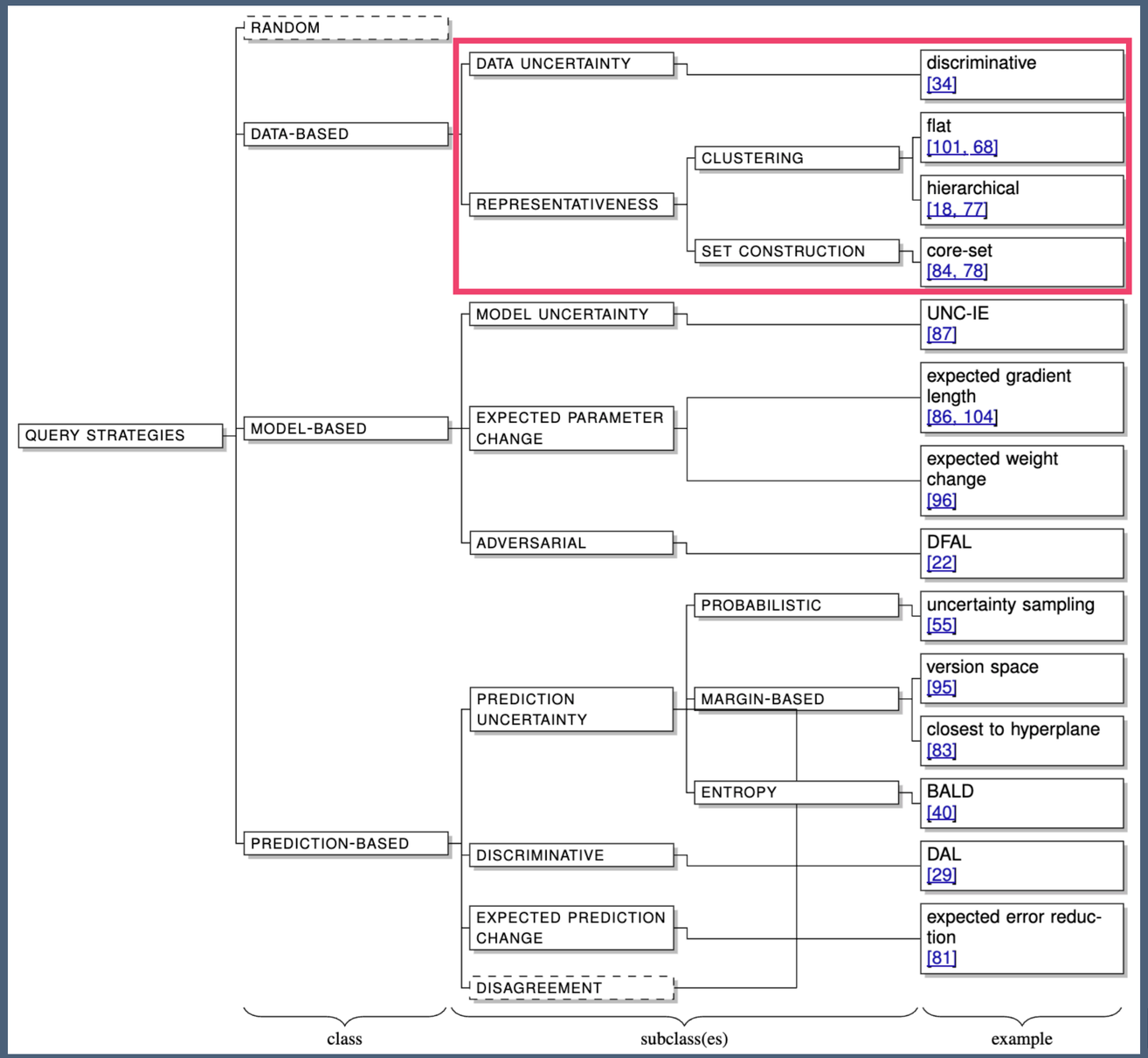
Query-By-Committee

- **Core Idea:** Maintain a committee of diverse models $C = \{\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(C)}\}$; query points with maximal inter-model disagreement.
- Committee Generation:
  - Bootstrap sampling of labeled set
  - Different random initializations / hyperparameters

$$H_{\text{vote}}(x) = -\sum_{y \in \mathcal{Y}} \underbrace{\left(\frac{1}{C} V(y \mid x)\right)}_{p_y} \log\left(\frac{1}{C} V(y \mid x)\right)$$

$$V(y \mid x) \;=\; \sum_{c=1}^{C} \left[\theta^{(c)}(x) = y\right].$$

Query Strategy Overview

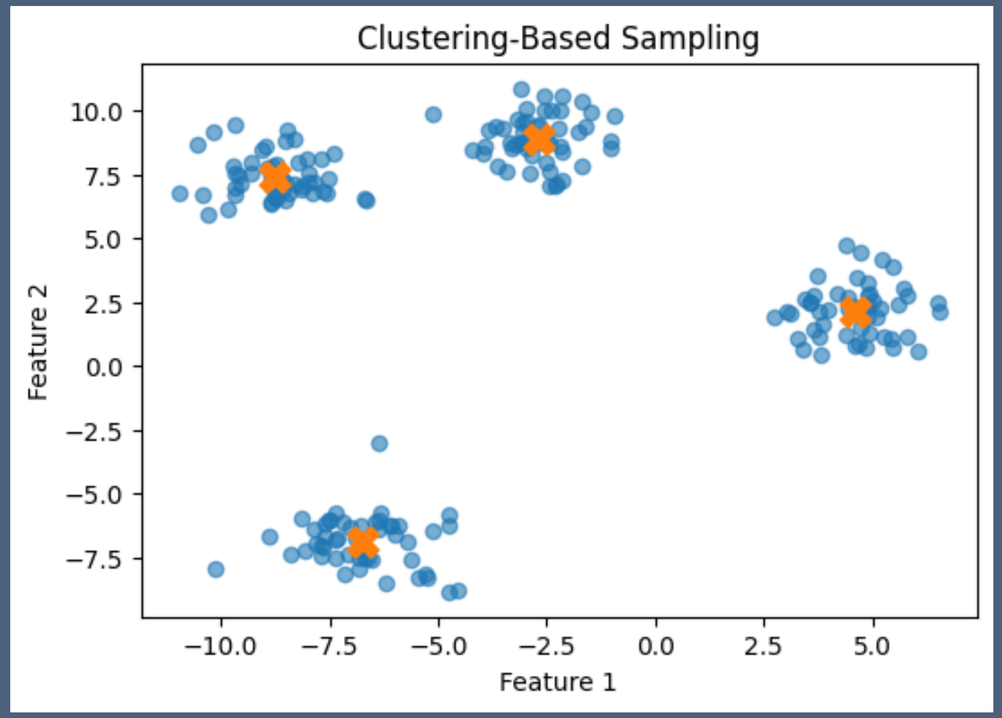


[Schroeder et Niekler, 2020]

Cluster-based Sampling

1. Embed your data into a feature space (e.g. sentence embeddings)
2. Run a clustering algorithm (K-means, Agglomerative) to partition the pool into K groups
3. Select one representative per cluster (the centroid or nearest neighbor) for labeling
4. Iteration: After labeling, you may recompute clusters on the remaining unlabeled pool

- Pros:
  + Ensures coverage of all major “regions” in your data
  + Simple to implement with standard libraries
- Cons:
  - Sensitivity to the choice of K
  - Small, rare “islands” may be missed if they form tiny clusters



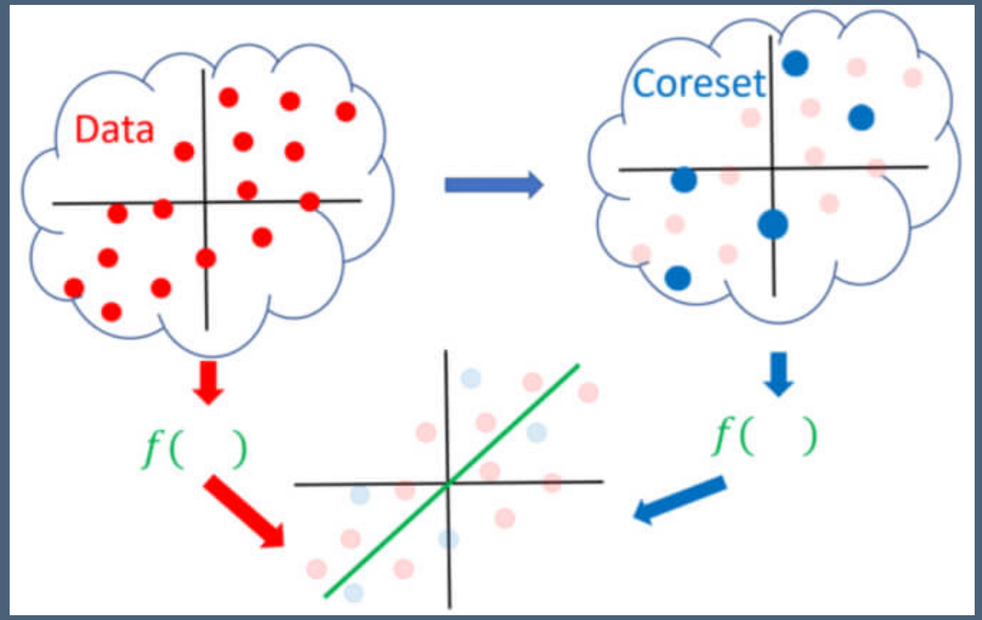
Core-Set Clustering

- Selects a small subset S of size K that “covers” the entire unlabeled pool by minimizing the maximum distance any point has to its nearest chosen member:

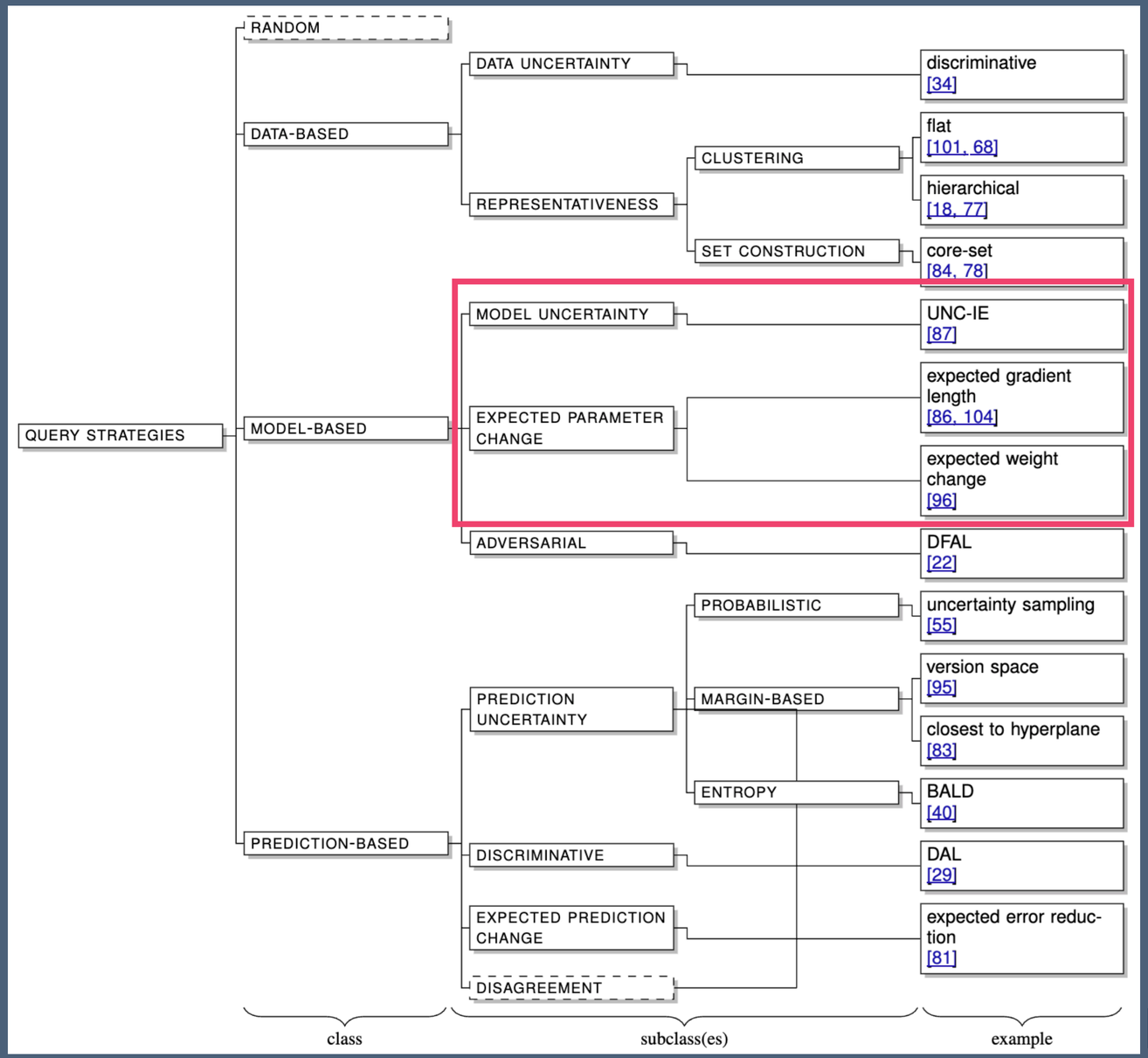
$$\min_{S \subseteq \mathcal{U},\ |S|=K} \ \max_{x \in \mathcal{U}} \left[\min_{s \in S} d(x, s) \right]$$

Greedy Farthest-First Algorithm

1. **Initialize**
   - Pick one seed (e.g. the point farthest from the mean).
2. **Repeat** until $|S| = K$:
   - For each $x \notin S$, compute $d_{\min}(x) = \min_{s \in S} d(x, s)$.
   - Select $x^* = \arg\max d_{\min}(x)$ and add to S.



Query Strategy Overview



[Schroeder et Niekler, 2020]

What have we learned today? (1/2)

- Definition & Goal:
  Active Learning is an iterative, model-driven approach that chooses which unlabeled examples to annotate in order to maximize performance with minimal labeling effort.

- Why Active Learning?
  – Cuts annotation costs & time by focusing on the most informative samples
  – Avoids wasted effort on redundant or unhelpful data
  – Essential for low-resource languages, specialized domains, and fast-moving applications

- Core AL Paradigms:

  - Membership Query Synthesis – generate and label synthetic instances

  - Stream-Based Sampling – decide on each example as it arrives

  - Pool-Based Sampling – select the top K from a static unlabeled pool

What have we learned today? (2/2)

- The Active Learning Cycle:

  1. Train initial model on small seed set

  2. Score unlabeled data with a query strategy

  3. Query labels for top examples

  4. Add to training set & retrain

  5. Repeat until budget is exhausted

- Query Strategies Introduced:
  – Uncertainty Sampling: Least-Confident, Margin, Entropy
  – Disagreement Sampling: Query-by-Committee, BALD
  – Data-Based Sampling: Clustering, Core-Set

Practical Day 1

Coding Session

- **GOAL**: Build and compare an SVM classifier using multiple Active-Learning strategies.
- 1. Dataset Selection
  - Browse Hugging Face for a suitable binary sentiment dataset (e.g., SST-2, IMDb) (start with binary, compare against multi-class)
- 2. Exploratory Analysis
  - Inspect class balance, common tokens, and ambiguous examples (think about what which parts could be interesting)
- 3. Model Setup
  - Vectorize text with TF–IDF; train an initial SVM on a small labeled seed set.
- 4. Implement Query Strategies
  - Random sampling
  - Uncertainty sampling (Least-Confident, Margin, Entropy)
  - Query-by-Committee
  - Cluster-based selection
  - Core-Set (farthest-first)
- 5. Active-Learning Loop
  - Iteratively query labels, retrain the SVM, and record test performance.
- 6. Evaluation
  - Plot accuracy (and F_1) versus the number of labeled examples for each strategy.