

# PROJECT-2

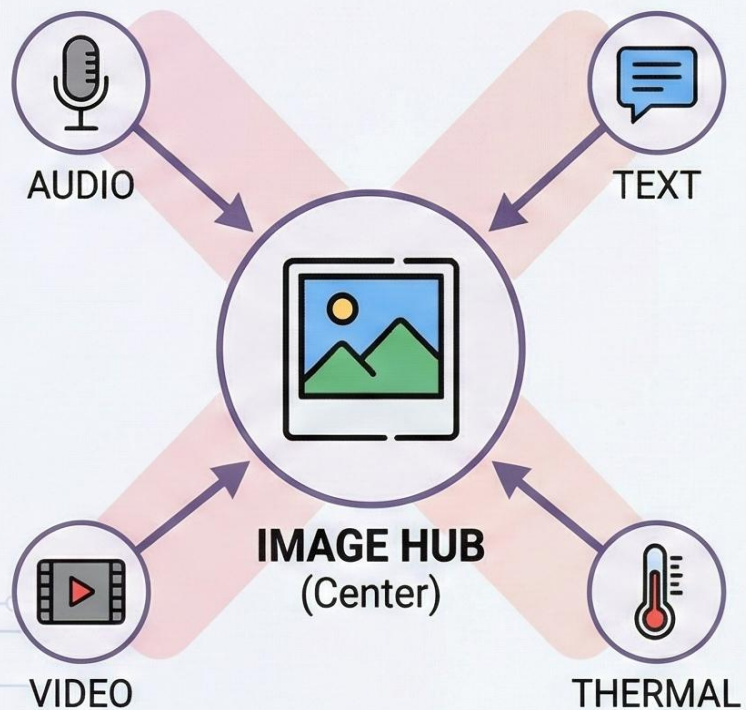
# UNIFIED MULTIMODAL LEARNING USING CONTRASTIVE EMBEDDINGS - UNIBIND

---

**PRESENTERS : AJMAL ABBAS, ASWIN SANKAR,  
PRATHESWARAN HARIHARAN ,VISHNU SELVARAJ**

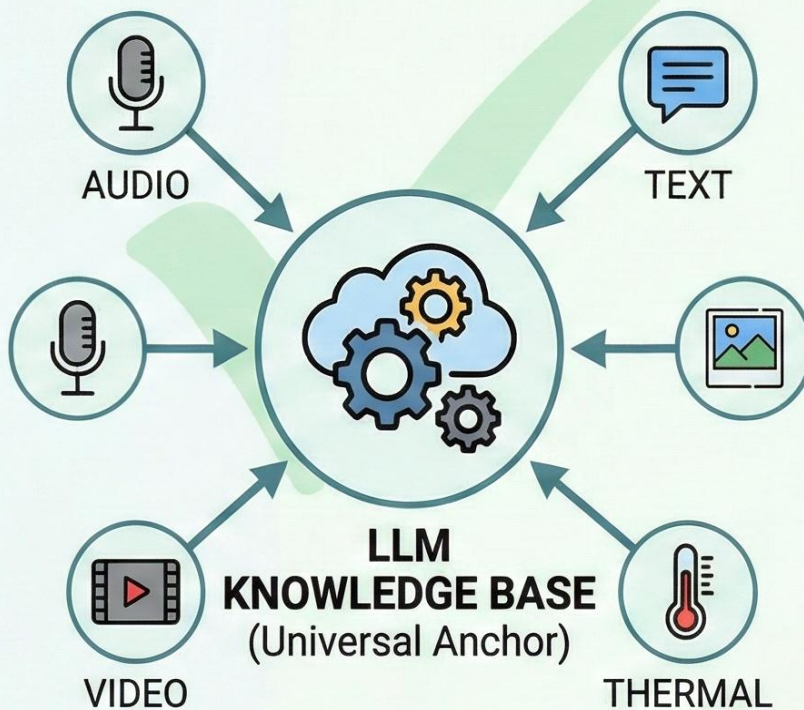
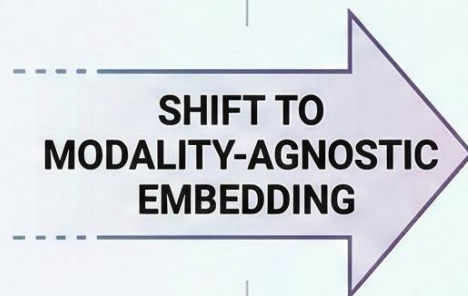
# UniBind: Moving Beyond Image-Centric Multimodal Learning

## OLDER VERSIONS (e.g., ImageBind)

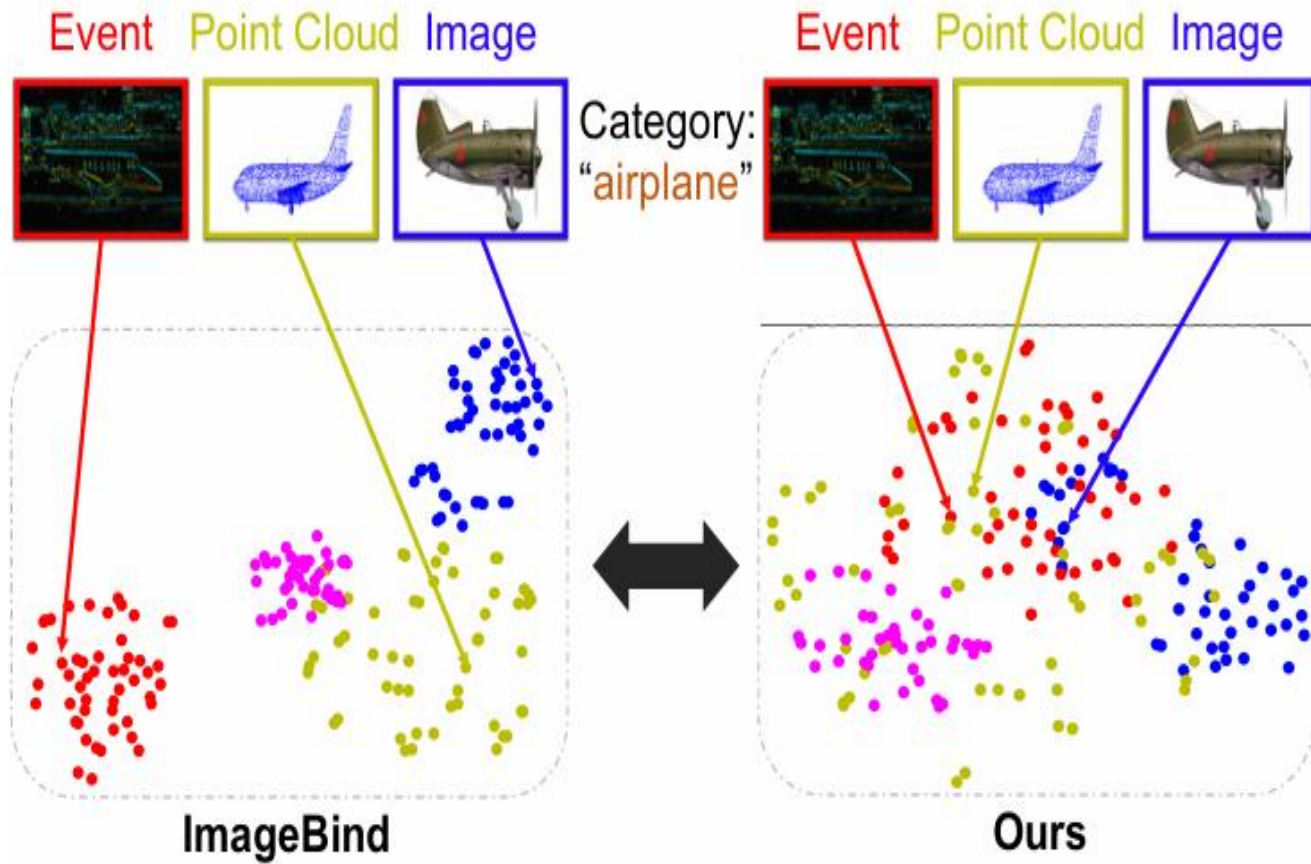


**Image-Centered:** All modalities align to the image representation.  
(Biased Space)

## UNIBIND CONCEPT (Proposed)



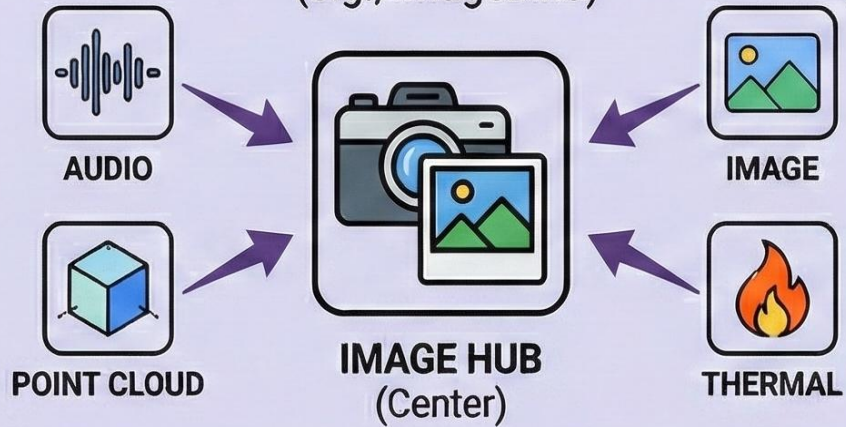
**Language-Centered:** All modalities, including images, align to a shared semantic space.  
(Balanced & Unified)



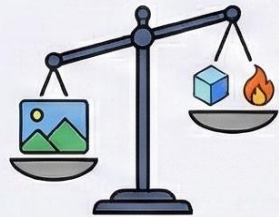
# UNIBIND: A Balanced Approach to Multimodal Learning

## (Solving Image-Centric Bias)

### THE OLD WAY: Image-Centric & Biased (e.g., ImageBind)



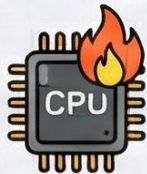
**Image-Centered:**  
All modalities align to the  
image representation.  
Creates a **Biased Space**.



#### Weak Semantics

Airplane, Dog

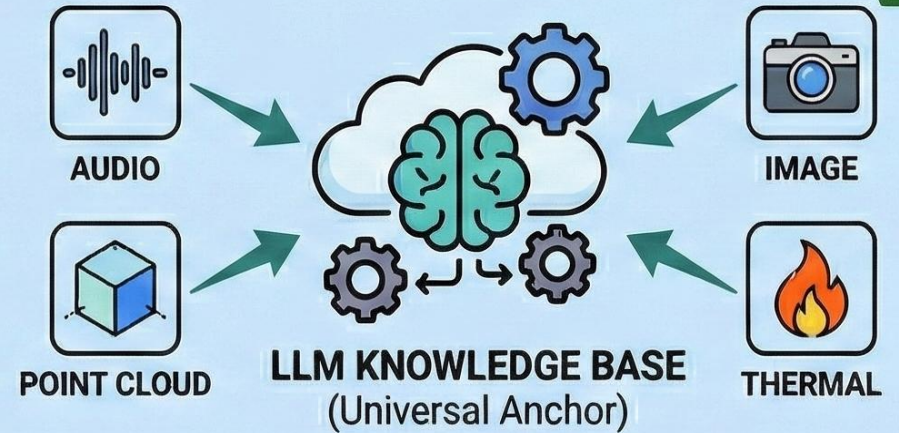
#### High Training Cost



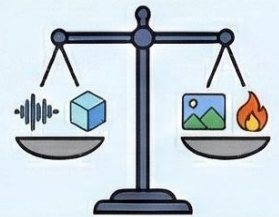
Full fine-tuning

SHIFT TO  
MODALITY-AGNOSTIC  
& EFFICIENT

### THE UNIBIND SOLUTION: Language-Centered & Balanced



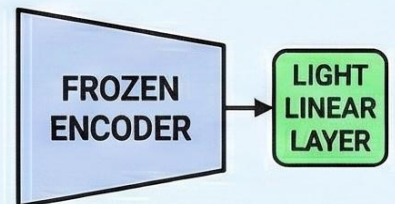
**Language-Centered:**  
All modalities align to a shared  
semantic space.  
Achieves **Balanced & Unified**  
Representation.



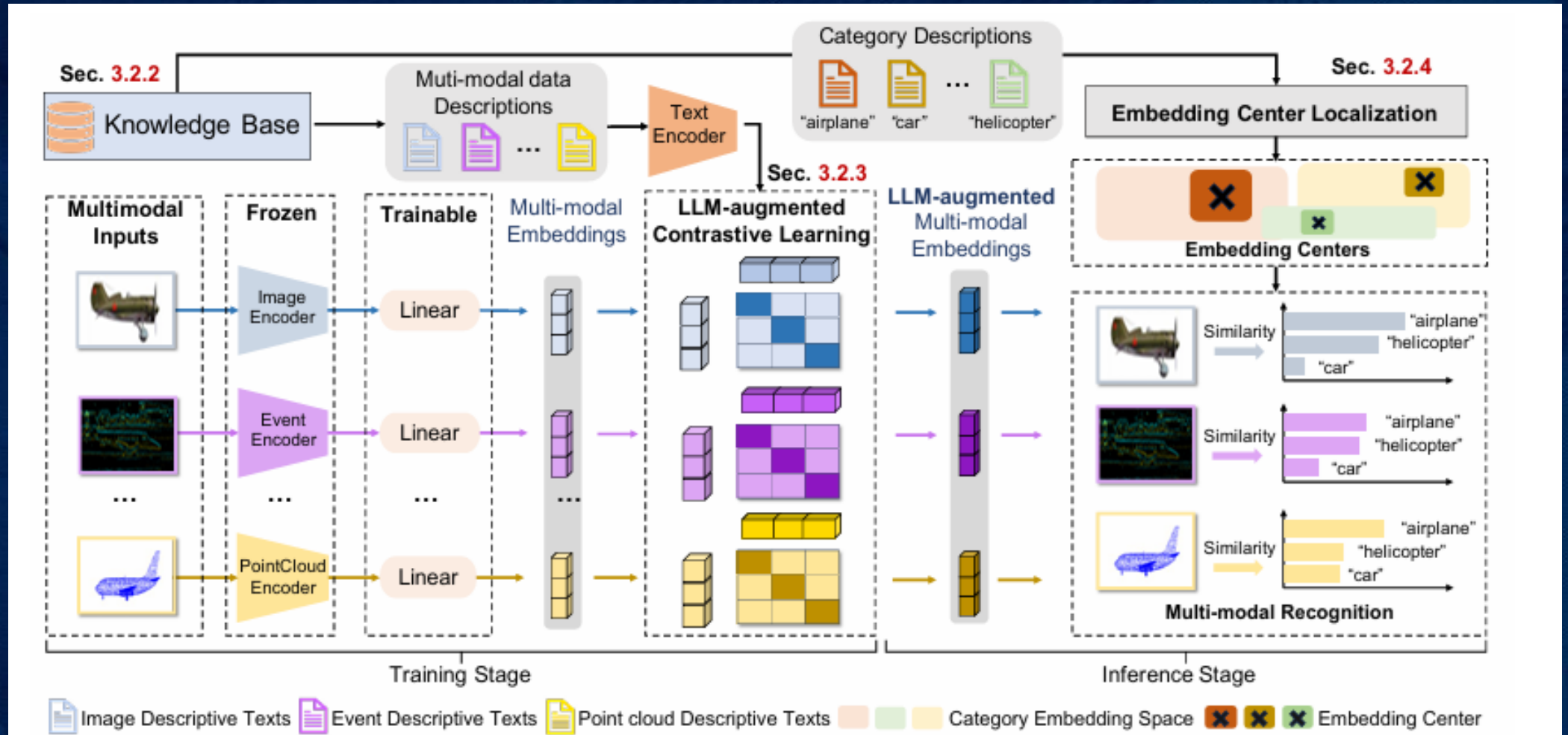
#### Rich Semantics

A large commercial  
aircraft with wings and  
engines, flying in the  
sky.

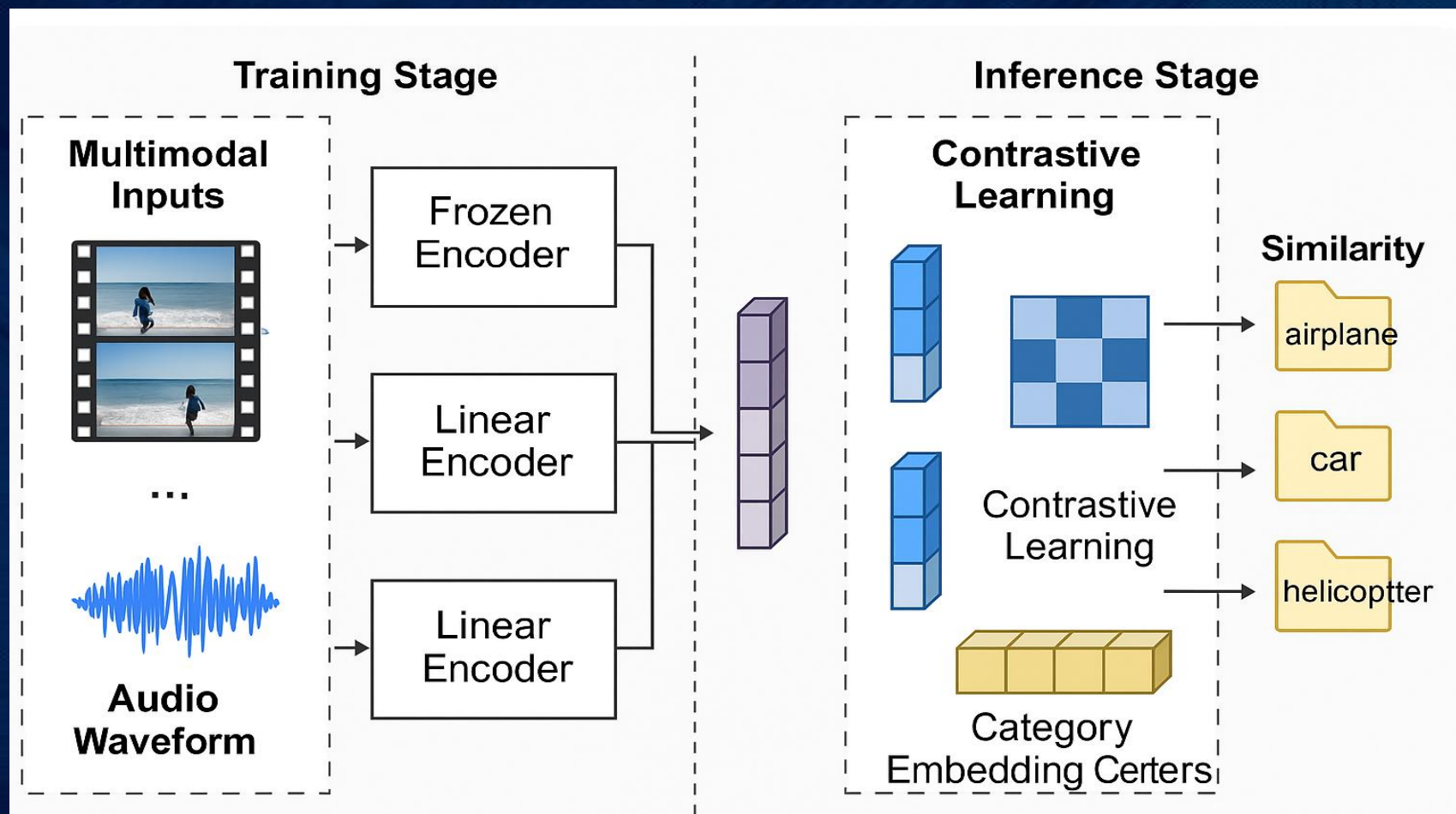
#### Efficient Training



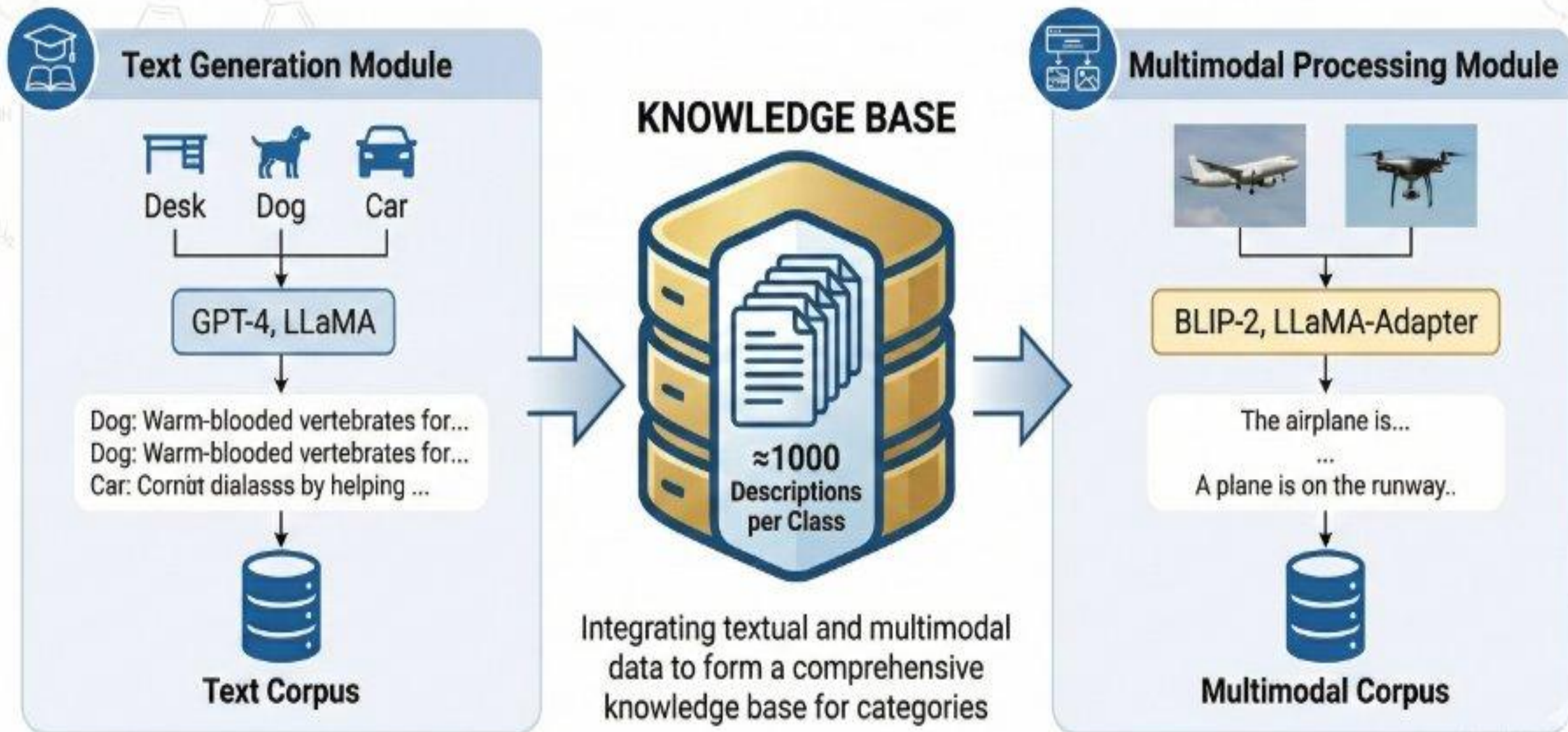
# BASE ARCHITECTURE



# PROPOSED ARCHITECTURE

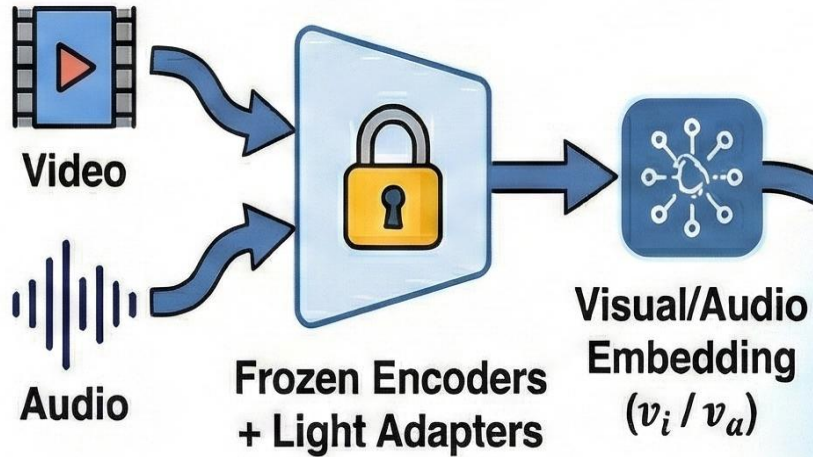


# STAGE 1: KNOWLEDGE BASE CONSTRUCTION



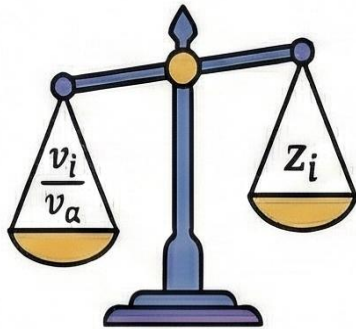
# STAGE 2: UNIFIED REPRESENTATION LEARNING

## VISUAL/AUDIO PIPELINE



## UNIFIED, MODALITY-AGNOSTIC REPRESENTATION SPACE

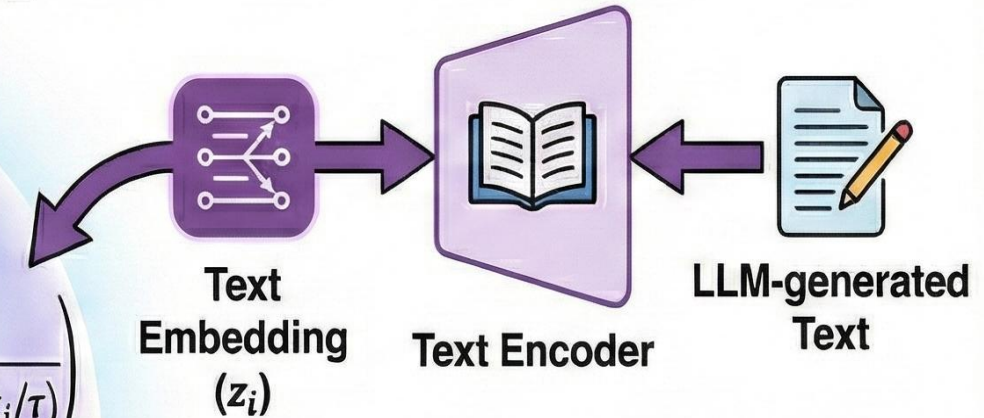
$$L = -\log \left( \frac{\exp(v_i^T z_i / \tau)}{\exp(v_i^T z_i / \tau) + \sum_{j \neq i} \exp(v_j^T z_j / \tau)} \right)$$



Contrastive Loss with Text Embedding  
= Balanced Alignment

Builds a Unified, Modality-Agnostic  
Representation Space

## TEXT PIPELINE



Symbol	Meaning
$\frac{v_i}{v_a}$	Visual/Audio Embedding
$z_i$	Text Embedding (negative)
$\tau$	Temperature Parameter

# STAGE 3: EMBEDDING CENTER LOCALIZATION (ECL)

## EMBEDDING CENTER CREATION

LLM-generated  
Text Embeddings  
( $z_1, \dots, z_k, \dots, z_{50}$ )



Selection

Embedding Center ( $E_C$ )  
Top 50 similar to "An  
audio/Video of a [class]"



Embedding  
Center ( $E_C$ )

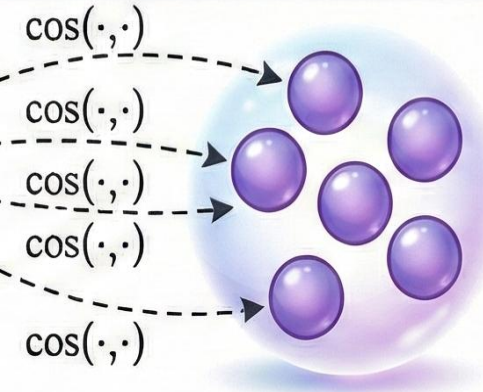
Built from top 50 LLM-generated text embeddings

## PREDICTION PROCESS

$$S(M_i, E_C^j) = \max\{\cos(F_m(M_i), z_1^C, \dots, z_{50}^C)\}$$



Modality  
Embedding  
( $F_m(M_i)$ )

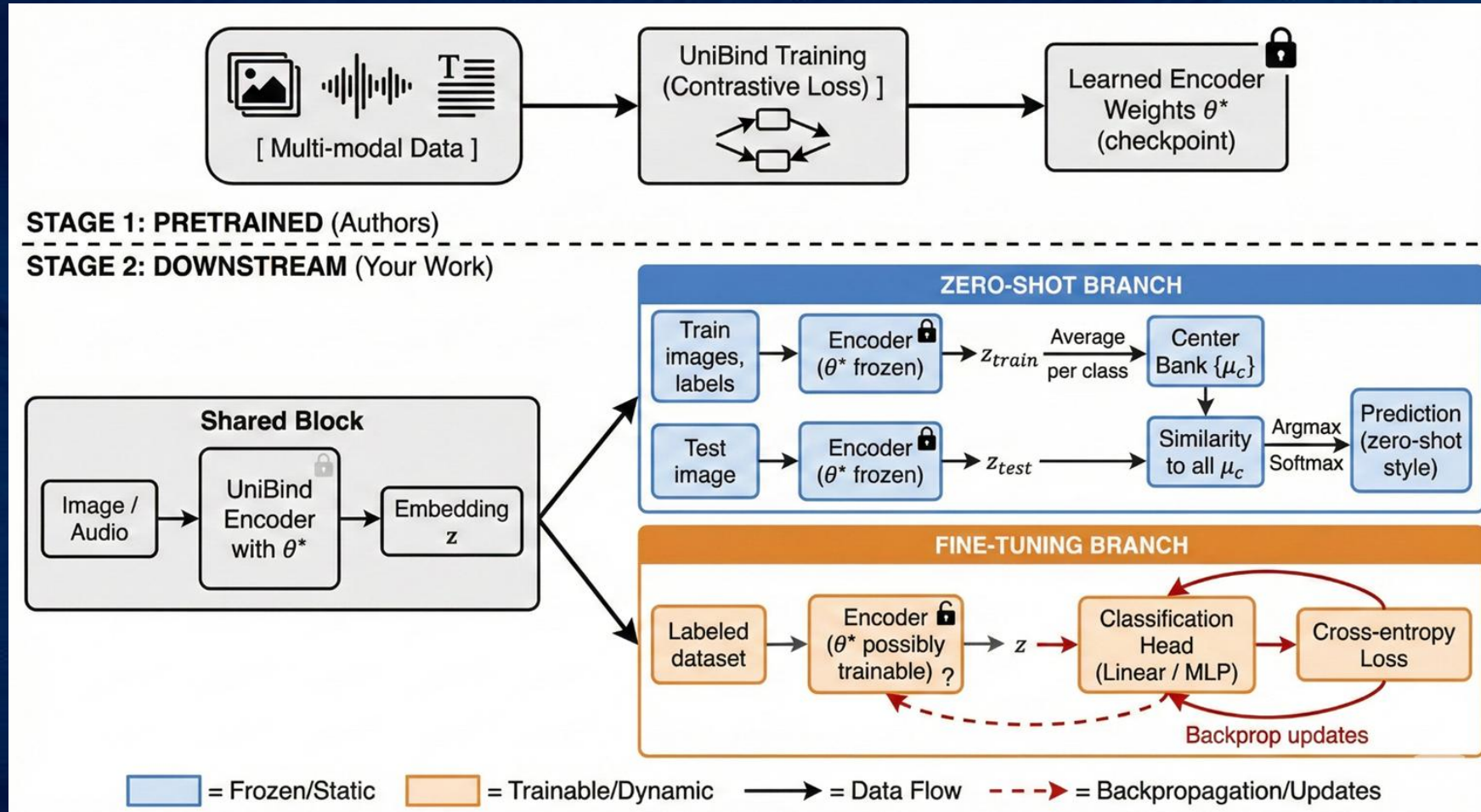


Prediction

Prediction = modality embedding closest to centre (cosine similarity)

Symbol	Intuition	Symbol	Intuition
$M_i$	e.g., one image of an airplane	$z_{kj}^C$	represents one LLM-generated description (e.g., "a jet taking off")
$F_m(M_i)$	gives an embedding vector in the unified space	$\cos(\cdot, \cdot)$	measures how close they are semantically
$E_C^j$	built from top 50 LLM-generated text embeddings	$\max\{\cdot\}$	pick the one most semantically aligned

# CONSTRUCTION



# COMPARATIVE ANALYSIS OF EXPERIMENTAL RESULTS: AUTHOR VS. REPLICATION

## AUTHOR'S REPORTED RESULTS (Source 1)

Method	Audio Accuracy	Video Accuracy
LLM Generated Prompts	69%	56%
UNIBIND	80%	71%

Data as reported in original study.

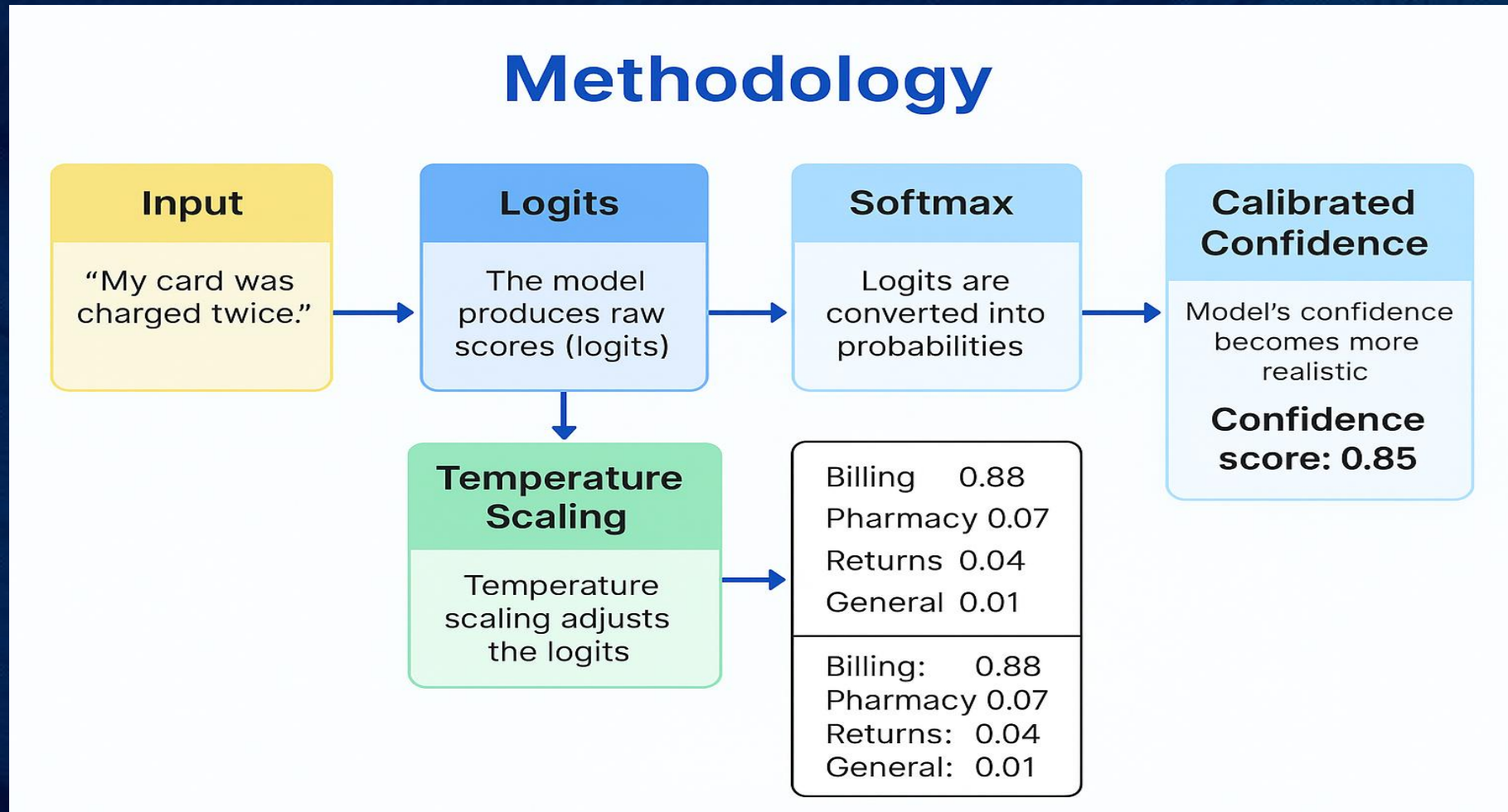
## MY EXPERIMENTAL RESULTS (REPLICATION) (Source 2)

Method	Image Accuracy	Audio Accuracy	Video Accuracy	Event Accuracy
LLM Generated Prompts	79%	-	34%	51%
UNIBIND	83%	80%	40%	59%

Data from independent replication experiment.

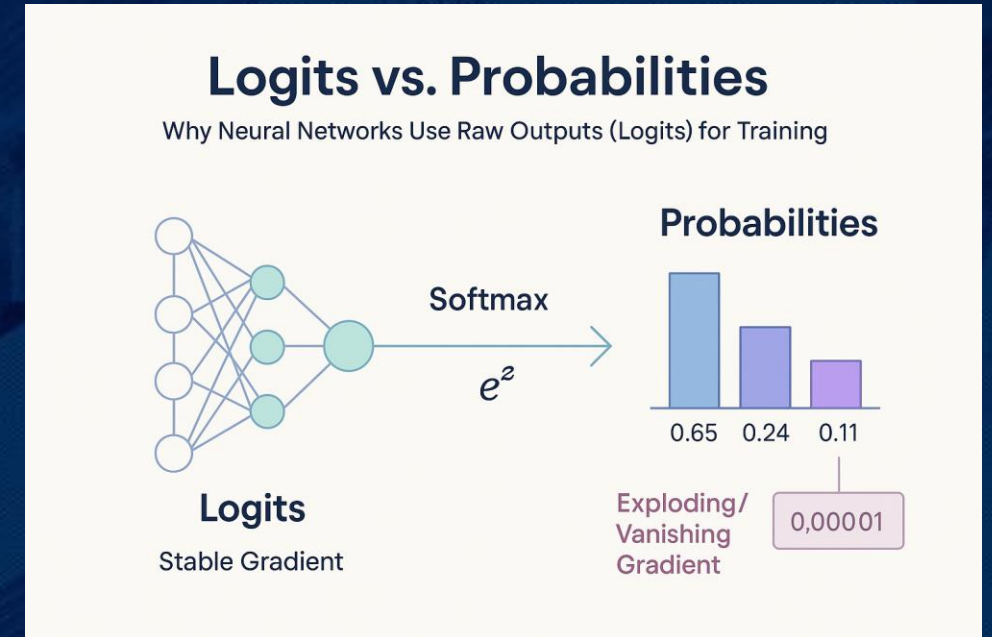
**SUMMARY:** UNIBIND consistently shows higher accuracy than LLM Generated Prompts across both original and replicated experiments. While Audio Accuracy for UNIBIND is replicated exactly (80%), Video Accuracy shows a significant discrepancy between the author’s results (71%) and the replication (40%). Image and Event accuracies were also evaluated in the replication.

# TEXT DATA CONFIDENCE METHODOLOGY



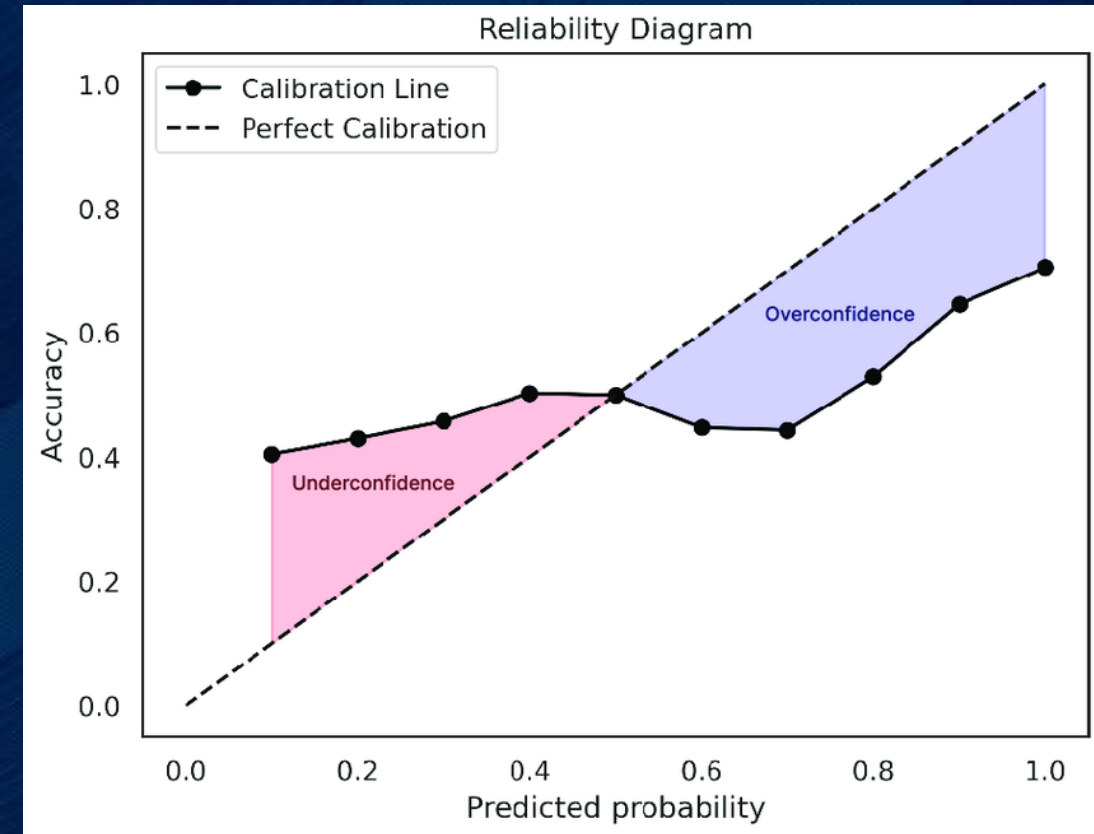
# FROM LOGITS TO CONFIDENCE

- The model reads an input message and produces logits (raw scores)
- Logits are converted into probabilities using softmax
- The highest probability = model confidence score
- Before calibration -> model is usually confident
- After Temperature scaling -> confidence becomes more realistic



# HOW WE CHECK IF CONFIDENCE SCORES ARE TRUSTWORTHY

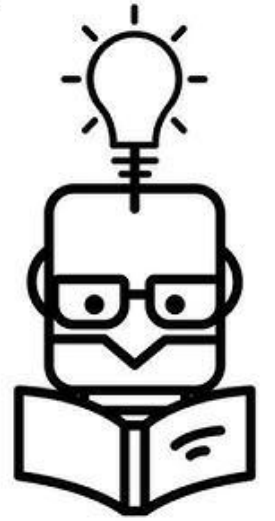
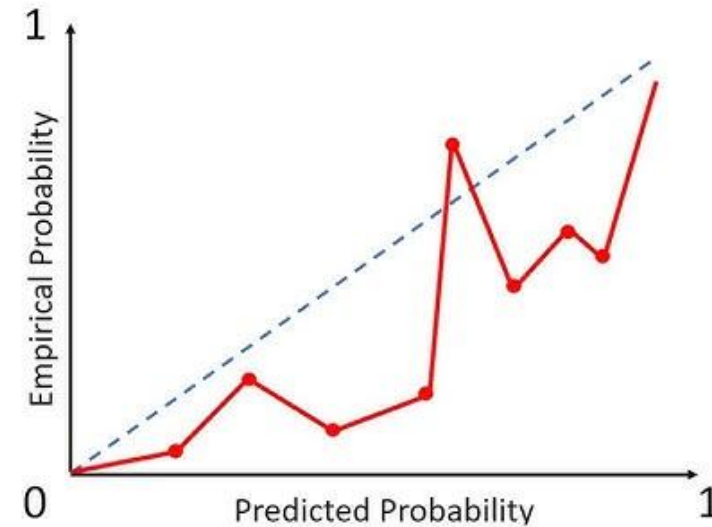
- Even if a model produces probabilities, they may not be reliable.
- **Calibration** checks if “confidence = accuracy”.
- Two main tools:
  - 1. **ECE (Expected Calibration Error)**
  - 2. **Reliability Diagram**



## 1. ECE (Expected Calibration Error)

- Measures how far the model's confidence is from the true accuracy.
- **Low ECE** → **good calibration**
- **High ECE** → **overconfident or underconfident model.**

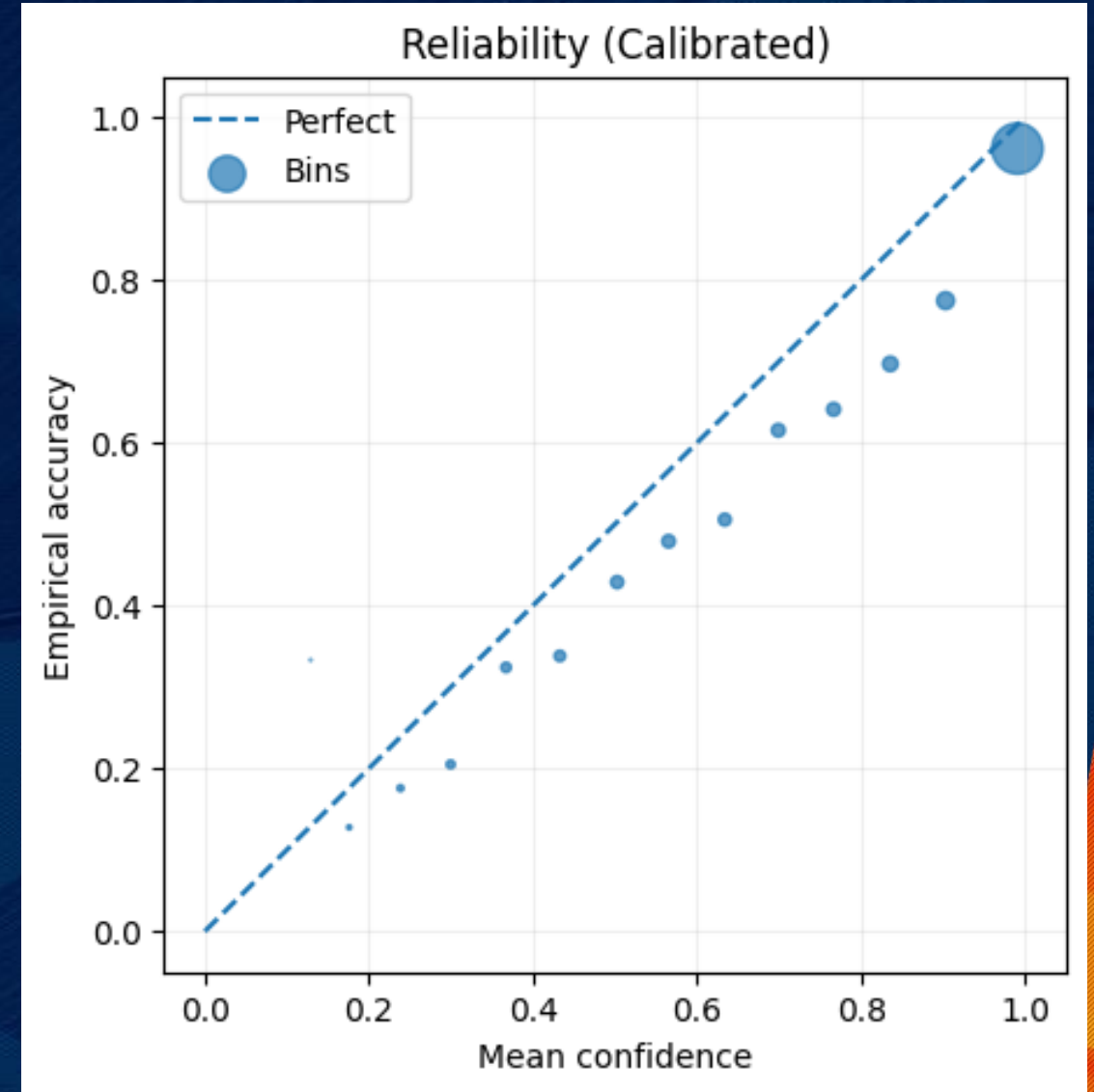
### Expected Calibration Error



$$ECE = \sum_{i=1}^M \frac{|B_i|}{N} |acc(B_i) - conf(B_i)|$$

## 2. Reliability Diagram

- X-axis: Model's predicted confidence
- Y-axis: Actual accuracy
- Points close to diagonal → good calibration
- Points below → model is **overconfident**
- Points above → **underconfident**



# BEFORE VS AFTER TEMPERATURE SCALING + FINAL CONFIDENCE SCORE

## Uncalibrated Model

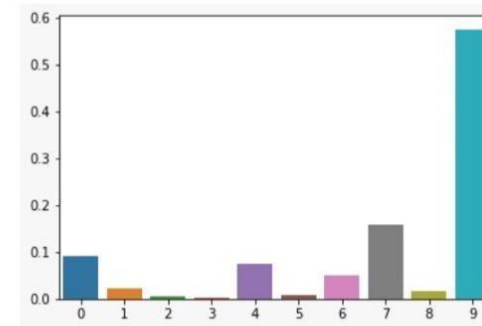
- Tends to be **overconfident**
- Example: Says “95% sure” but correct only 80% of the time
- Reliability diagram points lie **below diagonal**

## After Temperature Scaling

- Confidence becomes more honest
- Example: Says “85% sure” and correct ~85%
- Reliability diagram points move closer to diagonal
- ECE reduces → better calibration

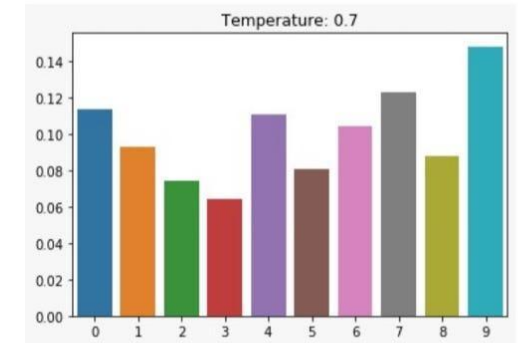
SOFTMAX WITHOUT TEMPERATURE ( $T=1$ )

$$\frac{e^{z_i}}{\sum_j e^{z_j}}$$



SOFTMAX WITH TEMPERATURE

$$\frac{e^{z_i/T}}{\sum_j e^{z_j/T}}$$

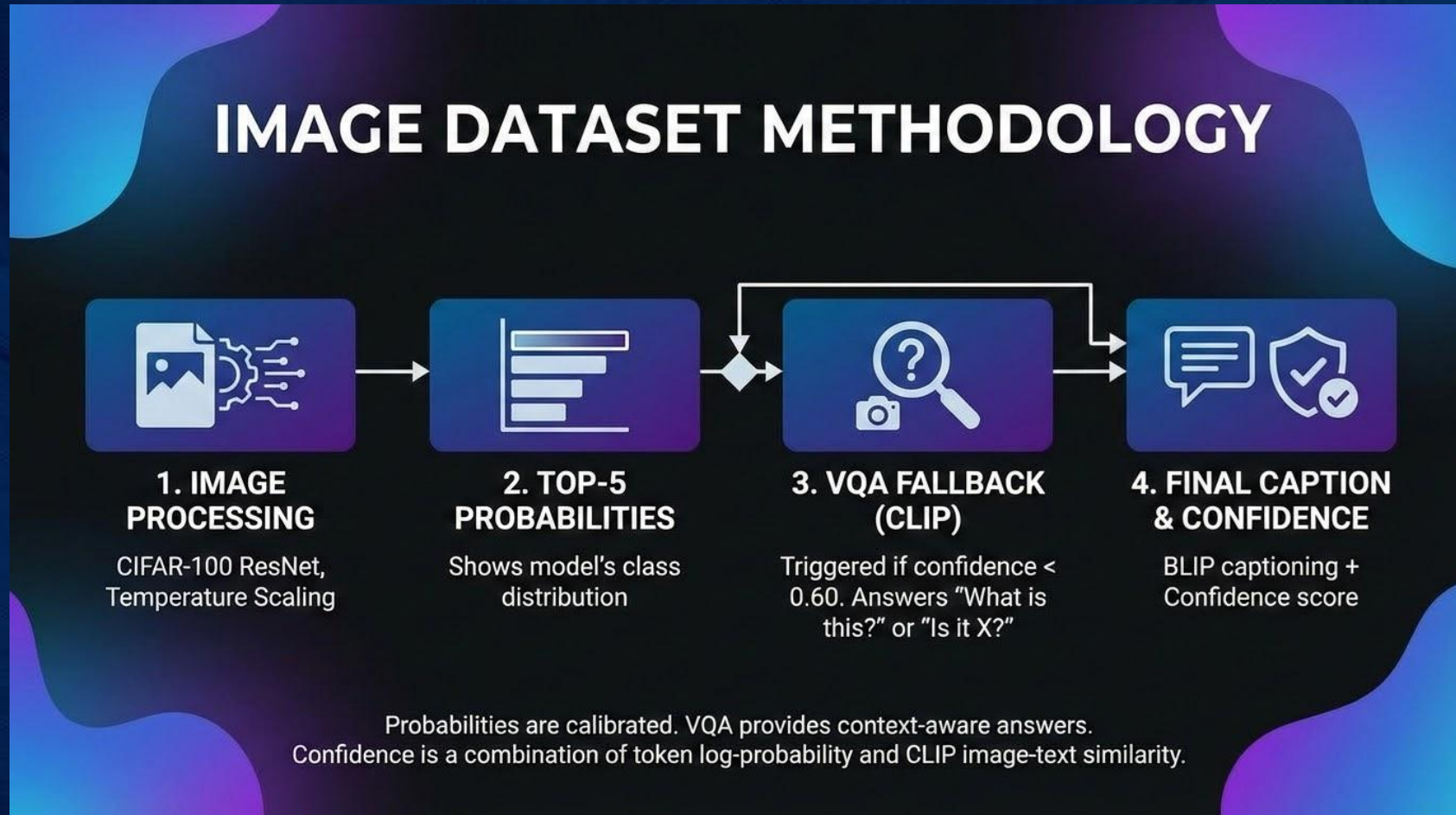


LESS ENTROPY

INCREASE IN ENTROPY  
WITH INCREASE IN  $T$

MORE ENTROPY

# IMAGE DATASET METHODOLOGY

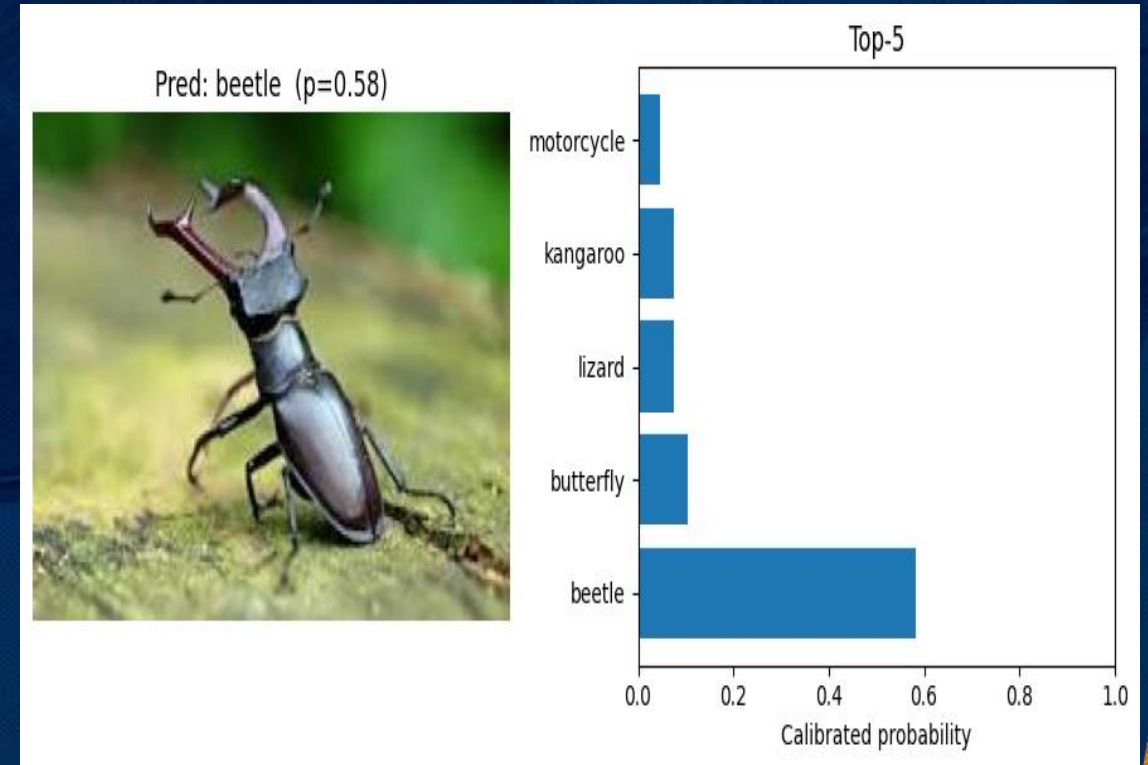


# HOW THE IMAGE IS PROCESSED

- Input image is passed through CIFAR-100 ResNet model.
- The model outputs logits (raw scores)
- Logits → Temperature Scaling → calibrated logits
- Softmax → Final probabilities Top-1 = predicted label Also compute entropy (uncertainty)

# TOP-5 PROBABILITIES

- Sorted probabilities from calibrated softmax
- Top-5 labels with highest probability
- Shows model's distribution over possible classes



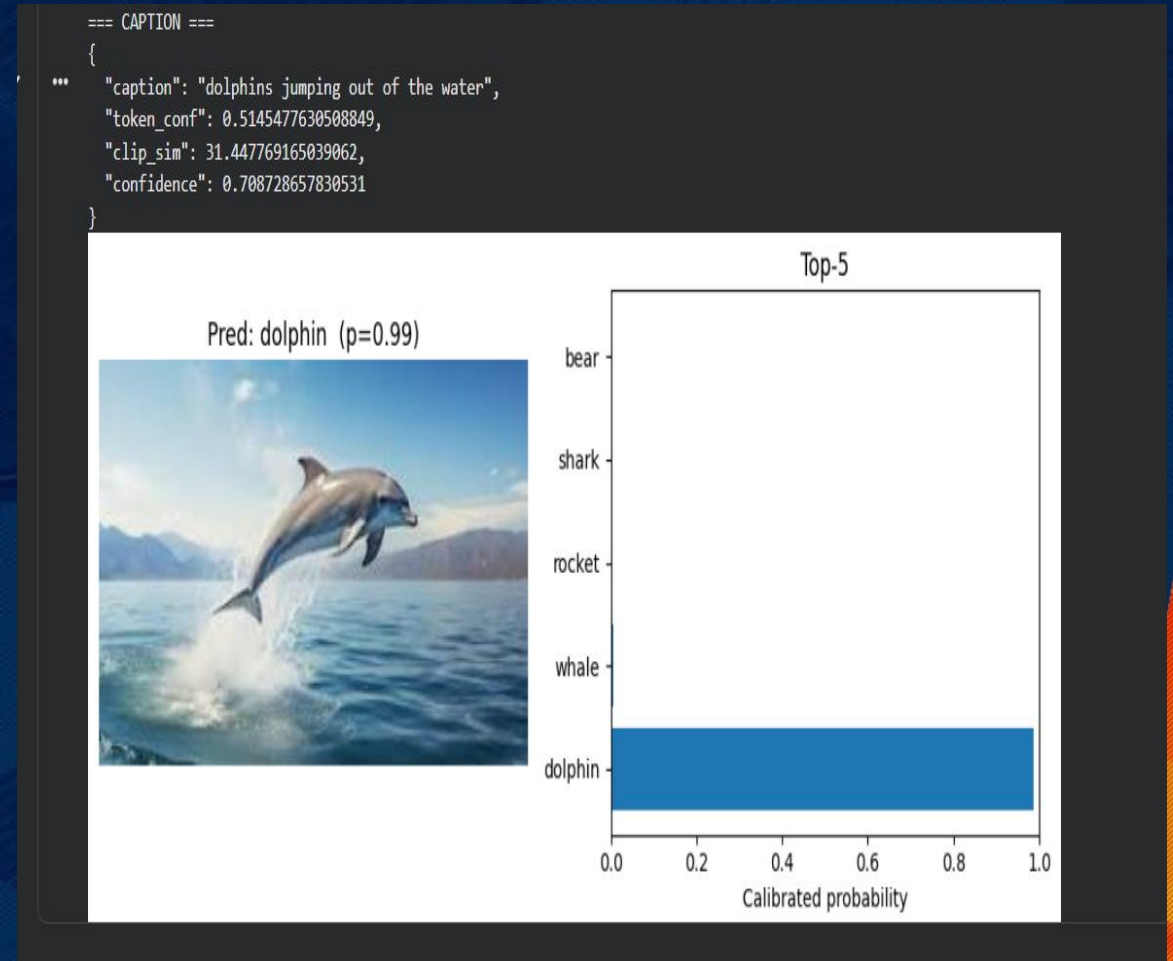
# VQA FALLBACK USING CLIP

- If CIFAR model confidence  $< 0.60$   
→ fallback mode
- CLIP zero-shot classification steps in
- VQA answers two types of questions:
- What is this? → CLIP picks best labels it
- X? → CLIP chooses yes/no

```
=== VQA: What is this? ===  
{  
  "answer": "insect",  
  "confidence": 0.9911876916885376,  
  "mode": "what-is-this [clip_zero_shot]"  
}  
  
=== VQA: Yes/No ===  
{  
  "answer": "yes",  
  "confidence": 0.7896366715431213,  
  "target": "beetle",  
  "mode": "yesno [clip_zero_shot]"  
}
```

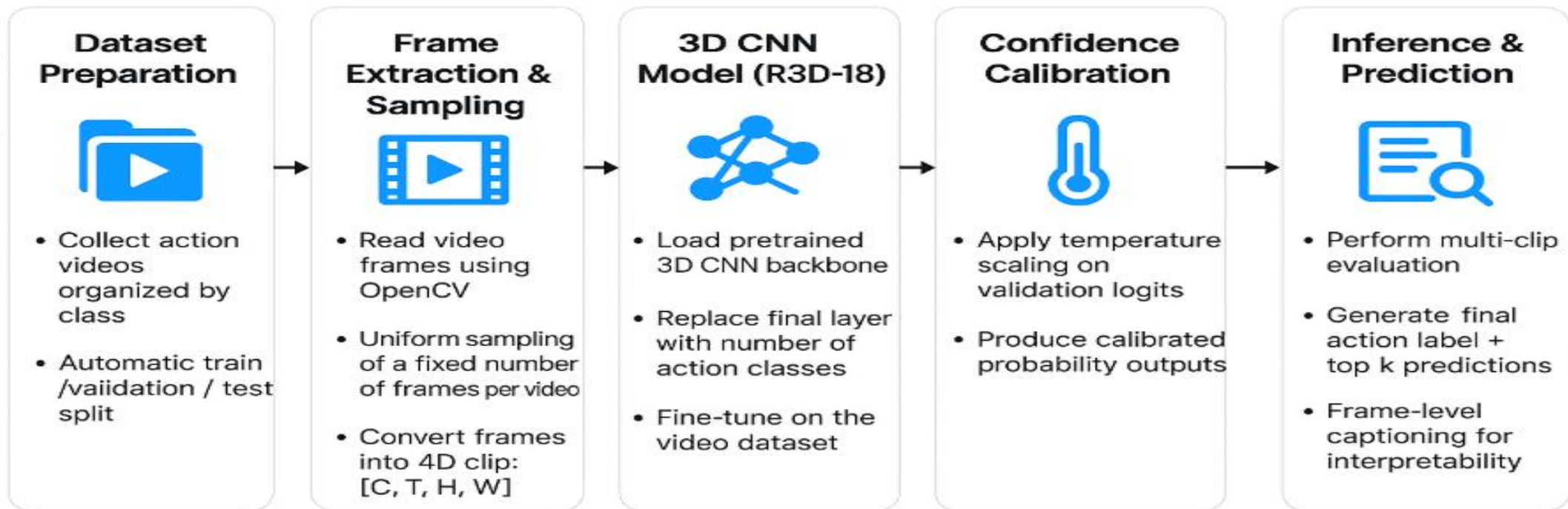
# IMAGE CAPTIONING + IMAGE CONFIDENCE

- BLIP captioning: generates text “dolphin is jumping out of the water”
- Confidence = combination of:  
Token log-probability (how confident model is in its own words)  
CLIP image-text similarity score
- Final caption confidence  $\approx 0.68$



# Video Dataset Methodology

## Methodology



# Frame Extraction & Sampling

## Frame Handling

- Videos loaded using OpenCV in RGB format
- All frames extracted from each video
- Uniform sampling is used to select a fixed number of frames ( $T = 16$ )
- Temporal jitter added during training for clip variation.
- Frames stacked into a clip tensor: [Channels  $\times$  Time  $\times$  Height  $\times$  Width]

# Pre-processing Pipeline

## Clip Pre-processing Steps

- **Resize frames to target dimensions**
- **Apply random or center cropping**
- **Normalize pixel values using pretrained mean and std**
- **The custom PyTorch Dataset class manages:**
  - **Frame extraction**
  - **Sampling**
  - **Augmentation**
  - **Conversion to model-ready tensors**

# Model Input & Confidence Estimation

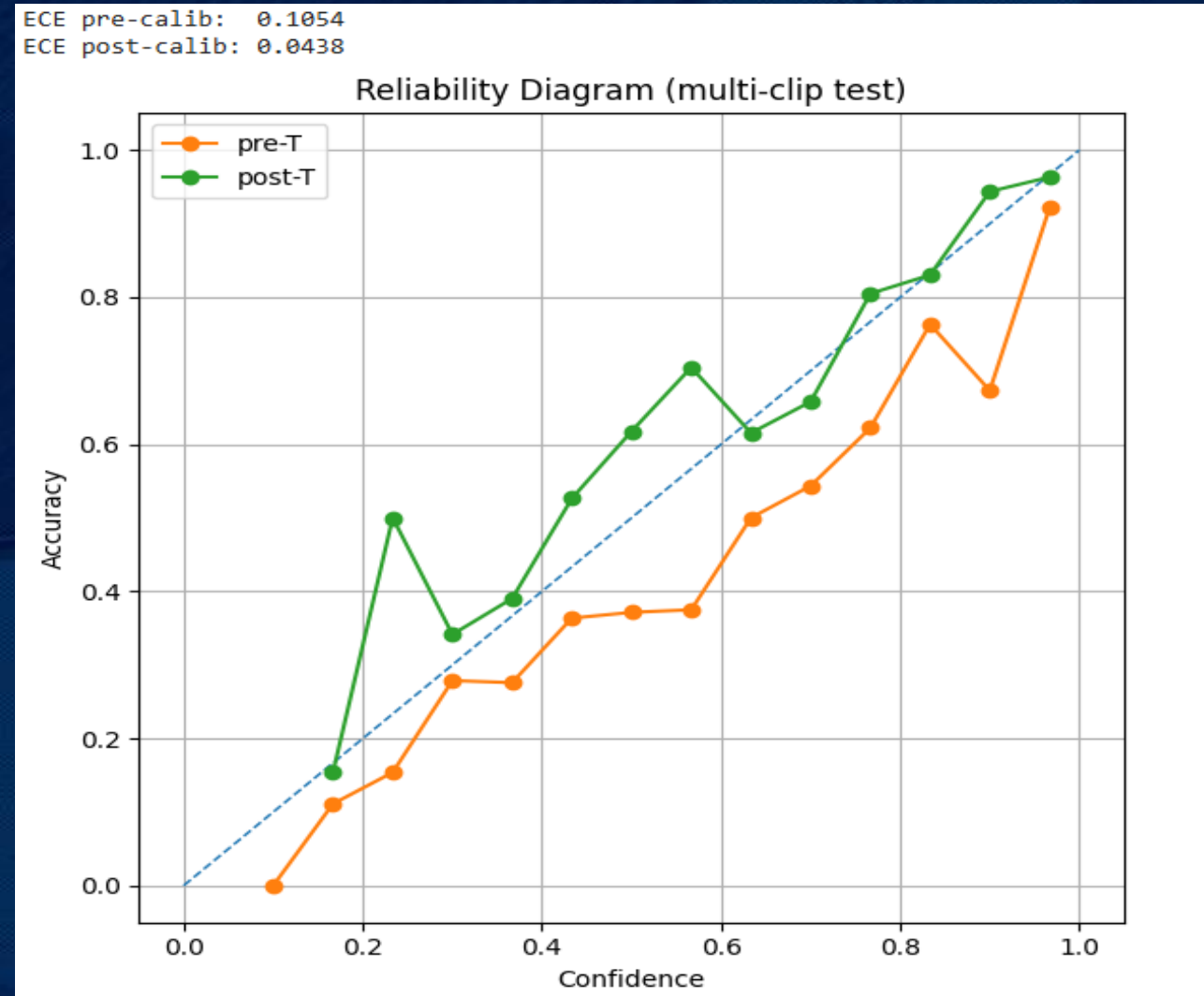
## Model Input & Evaluation

- Processed video clips fed into R3D-18 3D CNN for feature extraction
- Multi-clip evaluation performed for stable predictions
- Temperature scaling applied to logits for confidence calibration
- Final outputs include:
  - Predicted action label
  - Top-k class probabilities
  - Calibrated confidence scores
  - Uncertainty measures (entropy / confidence interval)

Epoch 01	loss=3.6351	val_acc=5.17%
Epoch 02	loss=3.3715	val_acc=15.09%
Epoch 03	loss=3.0647	val_acc=18.97%
Epoch 04	loss=2.7755	val_acc=29.02%
Epoch 05	loss=2.4782	val_acc=36.06%
Epoch 06	loss=2.2265	val_acc=41.24%
Epoch 07	loss=1.9373	val_acc=51.58%
Epoch 08	loss=1.6656	val_acc=46.55%
Epoch 09	loss=1.4223	val_acc=58.91%
Epoch 10	loss=1.2266	val_acc=61.21%
Epoch 11	loss=1.0317	val_acc=64.66%
Epoch 12	loss=0.8643	val_acc=68.10%
Epoch 13	loss=0.7155	val_acc=70.98%
Epoch 14	loss=0.6051	val_acc=69.40%
Epoch 15	loss=0.4952	val_acc=71.70%
Epoch 16	loss=0.4112	val_acc=72.27%
Epoch 17	loss=0.3996	val_acc=76.87%
Epoch 18	loss=0.3526	val_acc=75.57%
Epoch 19	loss=0.2817	val_acc=76.87%
Epoch 20	loss=0.2341	val_acc=77.44%

# Reliability Diagram

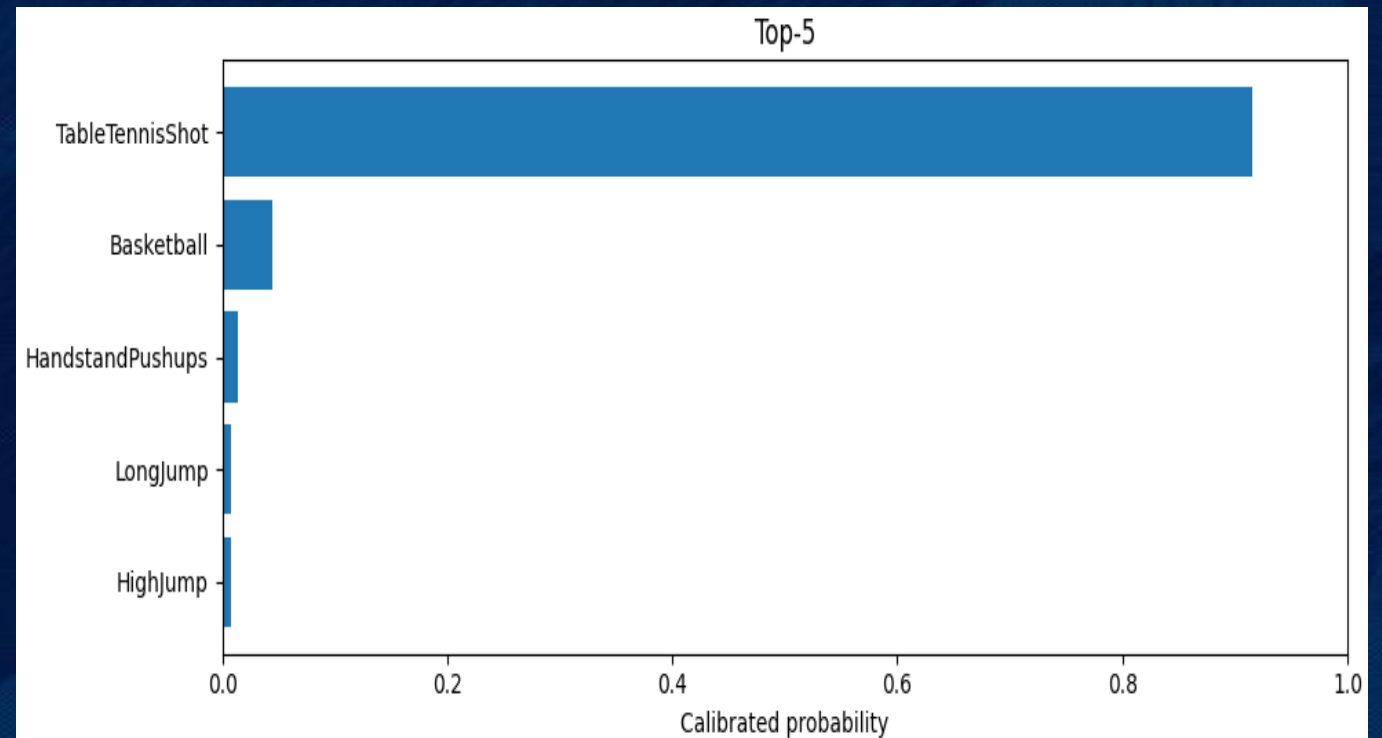
- Shows how well the model's confidence matches its actual accuracy.
- Orange curve (pre-T): model is overconfident before calibration.
- Green curve (post-T): confidence becomes more accurate and closer to the ideal diagonal line.
- Temperature scaling significantly improves calibration.
- ECE (Expected Calibration Error) drops from 0.1054  $\rightarrow$  0.0438, indicating more trustworthy confidence scores.



# CONFIDENCE SCORE

- The calibrated confidence score (0.9152) indicates strong certainty in the predicted action (TableTennisShot).
- The Yes/No VQA result returns “yes” with the same confidence, confirming consistency between classification and validation.
- The JSON output on the right shows the model’s final action prediction, confidence score, and verification response, demonstrating how the calibrated system interprets a real video.

```
=== VQA: What action is this? ===  
{  
  "answer": "TableTennisShot",  
  "confidence": 0.915228545665741,  
  "mode": "what-action [multiclip+temp-scale]"  
}  
  
=== VQA: Yes/No ===  
{  
  "answer": "yes",  
  "confidence": 0.915228545665741,  
  "target": "TableTennisShot",  
  "mode": "yesno [multiclip+temp-scale]"  
}
```



```
=== CAPTION ===  
{  
  "caption": "Two men playing a game of Tabletennis on a blue court ",  
  "model": "nlpconnect/vit-gpt2-image-captioning",  
  "note": "Caption generated from the representative video frame."  
}
```

# CHALLENGES

- Computational cost
- UniBind Environment setup
- Generate the embeddings for Audio Modality

# LIMITATION

- Depends on quality of LLM descriptions
- Needs more robust alignment for noisy; modalities
- Prompt sensitivity
- No Dynamic Adaption
- Limitation reasoning Ability
- How Confidence is the model Answers are?

# FUTURE ENHANCEMENTS

- Interpretability : Investigate the specific feature alignments to understand the basis of similarity detection between modalities.
- Multi Class Expansion : Extended the current architecture to support multi-class prediction, enabling the identification of multiple distinct concepts within a single input.