Meow-Omni 1: A Quad-Modal Model for Feline Future-Behavior Prediction

Jucheng Hu[*]
The Chinese University of Hong Kong
jchu@link.cuhk.edu.hk

Zhangquan Chen[*]
Tsinghua University
czq23@mails.tsinghua.edu.cn

Yulin Chen
University College London
stephen.chen.22@ucl.ac.uk

Chengjie Hong
University College London
zcabong@ucl.ac.uk

Liang Zhou
University College London
zcablz0@ucl.ac.uk

Tairan Wang
University College Londony
tairan.wang.22@ucl.ac.uk

Sifei Li
University College London
zcabsl0@ucl.ac.uk

Giulio Zhu
University College London
zcabgzh@ucl.ac.uk

Feng Zhou
University College London
zcabfzh@ucl.ac.uk

Yiheng Zeng
University College London
leo.zeng.22@ucl.ac.uk

Suorong Yang[†]
Nanjing University
sryang@smail.nju.edu.cn

Dongzhan Zhou[†]
Shanghai Artificial Intelligence Laboratory
zhoudongzhan@pjlab.org.cn

Abstract

Predicting an animal's future behavior from multimodal observations is an important challenge in computational ethology because visually or acoustically similar states can precede different actions. Existing Multimodal Large Language Models (MLLMs) do not natively process high-frequency physiological time-series data, limiting their ability to use wearable-sensor context alongside video and audio. We introduce **Meow-Omni 1**, the first open-source quad-modal MLLM for feline future-behavior prediction that combines video, audio, physiological time series, and text. We adapt a MiniCPM-o backbone with a scientific time-series encoder and projector, and train it with a Next-Behavior Prediction objective on the **Meow-10K** dataset. On **MeowBench**, an expert-verified benchmark containing 586 label-matched multimodal examples, Meow-Omni 1 achieves 71.16% top-1 accuracy, exceeding the zero-shot Qwen3.5-Omni-Plus baseline by 4.27 points, although that comparison combines domain adaptation, architecture, scale, and TS-representation effects. Expert verification attained Fleiss' $\kappa = 0.803$, supporting the reproducibility of the benchmark-screening process. We release the model, training framework, Meow-10K, and evaluation protocol as resources for multimodal animal-behavior research. Code available at: https://github.com/smgjch/Meow-Omni-1.

[*]Equal contribution.
[†]Corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

1 Introduction

The interpretation and prediction of animal behavior are central to veterinary medicine, wildlife conservation, and ethological research [1]. Yet an observable signal can be ambiguous: a feline purr, for example, is associated with contentment [2], but may also occur during pain or respiratory distress [3]. We refer to this observational ambiguity as *semantic aliasing* [4]. Relying on a single modality, such as visual cues or vocalisations alone, inevitably fails to resolve these critical ambiguities.

We introduce **Meow-Omni 1**, a Multimodal Large Language Model (MLLM) for feline future-behavior prediction that combines text, video, audio, and biological time-series signals. The work is motivated by three considerations. First, next-behavior prediction provides a measurable target for computational ethology: the model uses a current observation window to predict a subsequent labeled action or state. Second, an LLM backbone allows heterogeneous observations and natural-language prompts to be handled in a shared sequence model. Third, physiological time series can provide complementary evidence when posture, motion, or vocalization alone is ambiguous.

Technically, Meow-Omni 1 builds on MiniCPM-o 4.5 [5] and incorporates a time-series encoder adapted from Intern-S1 Pro [6, 7]. A learned projection layer maps time-series features into the language model's embedding space, where they are interleaved with visual, acoustic, and textual tokens.

We operationalize the task through Next-Behavior Prediction (NBP): given multimodal history $\mathcal{H}_t = \{V_{1:t}, A_{1:t}, B_{1:t}\}$, the model predicts a discrete future-behavior label y_{t+1}. These labels span actions and postural states. They can serve as observable proxies in downstream studies of latent state, but successful prediction does not establish that the animal subjectively intended the labeled outcome. This distinction governs the claims and terminology throughout the paper.

Our contributions are fourfold. We present an open quad-modal MLLM implementation that processes video, audio, physiological time series, and text in a unified sequence. We curate Meow-10K, a multi-source training dataset of 10,831 feline samples with varying modality combinations. We introduce MeowBench, a 586-example, expert-screened benchmark for feline future-behavior prediction, and report its annotation protocol and inter-annotator agreement. Finally, we provide a controlled same-backbone native-TS versus text-TS comparison, together with modality-masking, paired-bootstrap, per-class, and uncertainty analyses that characterize both the benefits and limitations of physiological input. The model, data resources, training pipeline, and evaluation protocol are released to support follow-up work on naturally synchronized recordings and stronger controlled evaluations.

2 Related Work

2.1 Traditional Methods in Animal Behaviour Interpretation

The computational study of animal behaviour has evolved from manual ethogram coding to automated, single-modality pipelines. These approaches fall broadly into three categories: acoustic analysis, postural recognition, and physiological monitoring.

Acoustic Analysis Significant effort has been devoted to the automatic classification of animal vocalizations. AVES [8] leveraged massive unlabeled audio datasets to train a HuBERT-based [9] self-supervised model, which was subsequently fine-tuned on downstream tasks from the BEANS benchmark [10]. This approach achieved performance comparable to fully supervised models, demonstrating strong transferability of learned audio representations. Building on this, Perch 2.0 [11] emerged as a bioacoustics foundation model, delivering state-of-the-art results on marine mammal and underwater audio tasks.

Beyond universal models, species-specific studies have targeted gibbons [12], birds [13], and dolphins [14]. Specifically for felines, Ntalampiras [15] developed a framework that identifies contextual states such as waiting for food, isolation, or being brushed from cat meows, using advanced signal processing and pattern recognition techniques.

Visual Analysis Recent work has focused on developing universal visual representations that generalize across diverse animal taxa. Sun et al. [16] first demonstrated that general-purpose video foundation models can serve as effective feature extractors for animal behavior analysis, transferring well to unseen species. Extending this, the Universal Action Space (UAS) [17] enables cross-species behavior analysis without requiring backbone fine-tuning. For few-shot adaptation, UniAP [18]

employs prompt-based learning to support cross-species perception in tasks like pose estimation and segmentation. In postural analysis, SuperAnimal [19] provides pretrained foundation models, drastically reducing the need for labeled data.

Focusing on specific species, SIPEC [20] offers an integrated pipeline for segmentation in socially interacting primates and mice. As for felines, vision-based affect detection has centred on facial landmarks. The Feline Grimace Scale has been digitized using DeepLabCut and CNNs to detect pain indicators. For instance, Feighelstein et al. [21] employed a multi-view setup to classify feline pain with high sensitivity. However, these systems often degrade in low-light or occluded conditions. In such scenarios, audio or biometric modalities can provide critical complementary signals.

Temporal Modeling Time-series (TS) data in animal behavior research typically originate from raw wearable sensor streams (e.g., accelerometers) and kinematic trajectories derived from pose estimation. Across species, these sequential signals are commonly mapped to behavioral states using classical classifiers [22, 23]. With the maturation of deep learning, One-dimensional Convolutional Neural Networks (1D-CNNs) and Long Short-Term Memory (LSTM) networks are widely used to process both raw sensor windows and keypoint trajectories [24]. For multi-sensor fusion, hybrid 1D-CNN/LSTM pipelines remain dominant [25–27].

In the feline domain, TS modeling remains largely limited to discrete physical actions. Recent work by [28] applied 1D-CNN and LSTM architectures to recognize simple behaviours like walking, grooming, and eating in cats using IMU data.

2.2 The Rise of MLLMs

While traditional deep learning excels at isolated pattern recognition, unified MLLMs offer a flexible interface for combining heterogeneous observations and language-based tasks. The field has therefore moved from modular late-fusion pipelines toward end-to-end multimodal models.

Leading Foundation Models Recent advancements have yielded highly capable Vision-Language Models (VLMs) and Omni-modal models. Proprietary architectures, such as Claude Opus 4.7 [29] and Gemini 3.1 Pro [30], alongside open-source leaders like Qwen3.5-397B-A17B [31] and Qwen3-Omni-30B-A3B [32], demonstrate remarkable capabilities in integrating text, high-resolution vision, and audio. These models achieve instant, human-like sensory integration. However, their architectural focus remains predominantly constrained to human-centric modalities. They excel in linguistic speech understanding while treating non-linguistic signals as background noise.

Scientific and TS Multimodality Extending MLLMs into non-linguistic and numerical domains represents a crucial new frontier. Notably, models such as Intern-S1-Pro [6] have pioneered the integration of continuous TS modalities natively into the LLM framework. Unlike conventional models that treat numerical data as flat text, Intern-S1-Pro employs specialized temporal encoders to process raw, high-frequency scientific sensor readings and physiological data, grounding continuous 1D signals within a semantic embedding space.

Existing Constraints in Computational Ethology Despite the immense power of these foundation models, they face three critical limitations when applied to animal behavior:

1. **Modality Blindness and Integration:** While Intern-S1-Pro supports TS data, and models like Gemini or Qwen excel in audio-visual streams, no existing general-purpose MLLM natively co-embeds high-frequency biological TS data (e.g., IMU/ECG) *alongside* both audio and video. Current architectures treat physiological readings as disparate streams.
2. **The Symbol Grounding Gap:** Current models lack the sensory-motor grounding to correlate a real-world physiological spike (e.g., a 50Hz acceleration burst) with a specific behavioral micro-expression or non-linguistic feline vocalization.
3. **Domain Bias:** Trained overwhelmingly on human data, these models frequently misinterpret feline communication such as subtle frequency shifts, pupil dilation, and nuanced tail positioning, as generic "animal noise."

2.3 Benchmarks and Evaluation Vacuum

Evaluation frameworks for animal intelligence are remarkably sparse. While general benchmarks like Animal-Bench [33] and MammalNet [34] exist, they focus predominantly on species recognition or basic action labeling (e.g., “running”). There is currently no standardized framework for *intent*

inference (distinguishing "defensive aggression" from a "playful swipe"). Meow-Omni 1 addresses this by introducing MeowBench, the first benchmark specifically designed to test cross-modal reasoning between biometrics, vision, and audio.

3 Problem Formulation

3.1 Observable Prediction Target

Let the observation history up to time t be $\mathcal{H}_t = \{V_{1:t}, A_{1:t}, B_{1:t}\}$, where V, A, and B denote video, audio, and biometric time-series data. Meow-Omni 1 predicts a discrete future-behavior label $y_{t+1} \in \mathbb{Y}$ from this history:

$$\hat{y}_{t+1} = \arg\max_{y \in \mathbb{Y}} P(y \mid \mathcal{H}_t; \theta). \tag{1}$$

The label set includes overt actions and postural states. Accordingly, the task is future-behavior prediction rather than direct inference of a subjective cognitive state. In near-free-choice recordings, a future action can be used as an operational proxy for an antecedent behavioral tendency, but the mapping is neither one-to-one nor sufficient to establish intention.

3.2 Learning Objective

The model is trained with the standard conditional language-model objective over the supervised response sequence r associated with each multimodal input:

$$\mathcal{L}(\theta) = -\sum_{(\mathcal{H}_t, r) \in \mathcal{D}} \log P(r \mid \text{Encoder}(\mathcal{H}_t); \theta). \tag{2}$$

For NBP examples, the response contains the future-behavior label and its natural-language description. Physiological time series are included as potentially complementary evidence. The objective itself is not causal, and our experiments test predictive performance and sensitivity to modal disagreement rather than causal identification of latent intent.

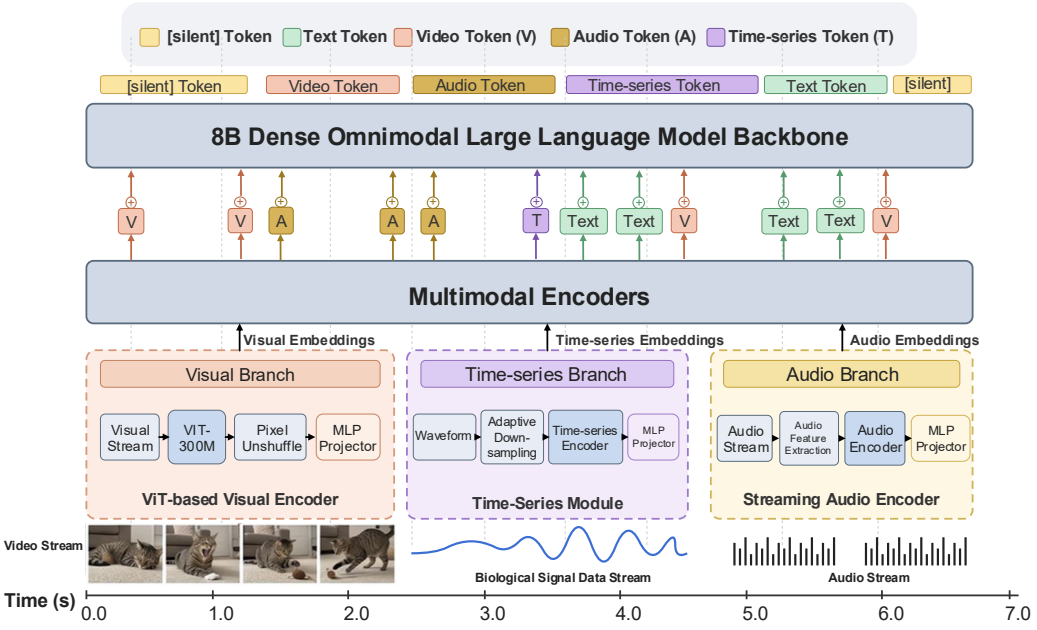


Figure 1: Meow-Omni 1 architecture. Video and audio use MiniCPM-o encoders, physiological time series use a frozen Intern-S1-Pro encoder and learned projector, and the resulting embeddings are interleaved with text tokens in the language backbone.

4 Methods

The development of Meow-Omni 1 follows a multi-stage approach encompassing specialized model surgery, a novel temporal labeling strategy, and a rigorous alignment-specialization training pipeline.

4.1 Architecture: Model Adaptation and Tokenizer Expansion

We build Meow-Omni 1 upon the MiniCPM-o backbone, performing a series of architectural "transplantations" to accommodate feline-specific biometrics. The overall architecture is illustrated in Figure 1.

1) Vocabulary Expansion. We extend the MiniCPM tokenizer with three unique control tokens: `<|ts_start|>`, `<|ts_unit|>`, and `<|ts_end|>`. To handle varying lengths of time-series (TS) data, a `MeowOmniProcessor` dynamically expands the `<|ts_unit|>` placeholder to match the number of hidden states produced by the encoder.

2) Backbone Adaptation. We implement the `MeowOmni1PreTrainedModel` class with a customized `MeowOmni1Config`. The configuration integrates a dedicated TS encoder and a linear projector adapted from Intern-S1-Pro. We resize the base LLM's embedding layer for the added TS tokens and inject the projected TS states into the transformer input sequence.

3) Multimodal Forward Pass. The primary class, `MeowOmni1ForCausalLM`, customizes the forward pass to accept N-dimensional TS tensors alongside standard vision and language inputs. TS embeddings are projected into the LLM's hidden dimension and interleaved with visual and linguistic embeddings, forming a unified multimodal context. The model is designed to process variable-length sequences that may contain any subset of the available modalities; absent modalities are simply omitted from the input sequence.

4.2 Training Pipeline: Alignment and Specialization

The training process is divided into two distinct phases to ensure modal stability.

Stage 1: Projector Alignment. We pre-train the TS projector using 383,853 labeled TS samples (described in Appendix A). During this stage, the LLM backbone and the TS encoder are frozen; only the projector is updated to map biometric features into the linguistic latent space. An early-stopping strategy with 1-epoch patience is employed, yielding an optimal alignment checkpoint after 2 epochs, which we denote as *Meow-Omni 1 Aligned*.

Stage 2: Multimodal Specialization. We fine-tune using the Meow-10K dataset, which comprises 10,831 high-quality samples with varying modality combinations (A/V/TS, A/V, V/TS, A/TS, single-modality, etc.). All encoders and projectors are frozen, and only the LLM backbone is updated. Missing modalities are absent from the input sequence (no placeholder tokens are used), allowing the model to learn to reason from any subset of the available streams. Using the same 1-epoch patience, the final *Meow-Omni 1* model is obtained after 1 epoch of fine-tuning.

4.3 Dataset Generation

The Meow-10K training set is assembled from multiple pipelines; full details are provided in Appendix A. Each sample ultimately carries a natural-language query-response pair and, where applicable, a label from a unified 26-class intention taxonomy (listed in Appendix E). The main labeling strategies are summarized below:

- **Time-Series (TS) Data.** We employ a Next-Behavior Prediction (NBP) labeling strategy: for a fixed-length accelerometer window, the target is a future behavior label from the source dataset. Intermediate transient movements and invalid labels are filtered to retain stable prediction targets.
- **Video-Only Clips.** A VLM-based pipeline detects action onsets and generates natural-language action captions. Each caption is subsequently mapped to one of the 26 future-behavior classes by an LLM; the mapping is verified by the same expert group that curated MeowBench.
- **Audio-Only Clips.** Standalone audio recordings are captioned with rich behavioural descriptions generated by a text-only LLM expansion of the original short captions. No pre-assigned future-behavior label is given; the model is trained to produce a descriptive caption from the audio signal.
- **Synchronized Audio-Video Pairs.** Clips derived from AudioSet inherit their temporal alignment from the source videos. A VLM jointly analyzes audio and video to produce an audio-focused caption and a probability distribution over the 26 future-behavior classes, which serves as the supervision target.

- **Synthetic Quad-Modal Samples.** A small number of samples combine video-audio pairs with TS data that share the same future-behavior label, following the same protocol as MeowBench (described next). These samples are verified by the expert reviewers and are included to teach the model joint tri-modal reasoning.

After balancing modality representation (by subsampling the abundant TS-only data to 2,000 samples), the final Meow-10K dataset contains 10,831 training samples.

4.4 MeowBench: Label-Matched Synthesis and Evaluation

MeowBench is a held-out suite for evaluating cross-modal future-behavior prediction. Because no naturally synchronized quad-modal feline dataset was available, candidate examples were constructed by pairing a synchronized video-audio clip with a time-series sample from a different session that carried the same canonical behavior label. The benchmark therefore tests prediction on expert-screened, label-matched combinations rather than naturally co-occurring physiology and behavior; this limitation is stated explicitly in Section 8.

Eight professional feline ethologists, each with at least five years of clinical or research experience, were organized into three groups of sizes 3, 3, and 2. Each expert first rated assigned candidates independently as *Plausible*, *Uncertain*, or *Implausible*. Cases with disagreement or any non-Plausible rating were then discussed, and a candidate was retained only after unanimous group acceptance. Of 645 initial candidates, 586 were retained and 59 were rejected. Before consensus, unanimous three-category agreement was 588/645 (91.2%). Binarizing ratings as Plausible versus Not-Plausible yielded Fleiss' $\kappa = 0.803$ (SE $= 0.042$, 95% CI $[0.721, 0.885]$, $z = 19.20$, $p < 0.0001$). The main rejection reasons were behavior-signal mismatch, temporal inconsistency, and recordings in which the animal was visibly constrained.

Each retained candidate was converted into a four-option multiple-choice question (MCQ), with the canonical future-behavior label as the answer and three distractors sampled from the label inventory. We use MCQ evaluation because it provides a model-agnostic interface for API-only and open-weight systems, avoids adding model-specific classification heads, and yields a non-degenerate difficulty range across heterogeneous baselines. It is not a substitute for full-label-space diagnosis; therefore, we additionally report per-class results and common confusions in Appendix C, and release the labels in a form that supports open-ended classification.

5 Experiments

In this section, we describe the experimental setup, baseline models, and modality-masking analyses used to evaluate future-behavior prediction from quad-modal inputs.

5.1 Baselines for Comparison

Given that Meow-Omni 1 processes text, video, audio, and biometrics, we compare it with available unimodal and omni-modal systems to provide reference points rather than an architecture-controlled leaderboard.

1. **Unimodal Baselines:**
   - *Vision:* We use **Qwen3.5-122B-A10B** as a strong zero-shot baseline for visual future-behavior prediction.
   - *Audio:* We compare against the specialized framework by [15], which employs signal processing and pattern recognition for context-aware cat vocalization classification.
   - *TS:* We benchmark against the feline behavior recognition model proposed by [35], which uses deep learning for IMU-based behavior classification.
2. **Bi-modal Baselines:** We evaluate **Qwen3.5-Omni-Plus**, a leading omni-modal foundation model, using various two-modal combinations (e.g., Video + Audio) from our MeowBench suite.
3. **Full-Input Comparison:** Qwen3.5-Omni-Plus accepts video and audio natively and receives TS as text, whereas Meow-Omni 1 encodes TS natively; this is a system-level rather than controlled comparison.

Detailed descriptions of model architectures, preprocessing pipelines, and training hyperparameters for each modality are provided in Appendix B. All external LLM/MLLM baselines are evaluated zero-shot or inference-only; they are not fine-tuned on Meow-10K. Qwen3.5-Omni-Plus is proprietary and API-only, so fine-tuning it was not feasible. Its time-series input is a structured textual summary rather than raw sensor data. These differences make its comparison with Meow-Omni 1 a system-level comparison rather than a controlled estimate of native TS encoding. To provide the requested control, we additionally train **MiniCPM-o-textTS**: it uses the same MiniCPM-o backbone, Meow-10K data, and Stage 2 hyperparameters as Meow-Omni 1, but removes the TS encoder and projector and prepends the same structured textual TS summary used for Qwen3.5-Omni-Plus. Thus, MiniCPM-o-textTS and Meow-Omni 1 differ in how TS information is represented while holding the backbone, domain data, and fine-tuning recipe fixed. The modality configurations of Meow-Omni 1 use inference-time masking of the same final multimodal checkpoint; they are not separately trained unimodal or bimodal models.

5.2 Evaluation Metrics

All models are evaluated on the MeowBench MCQ suite (586 expert-verified samples). We report top-1 accuracy for future-behavior prediction. To further examine the role of the time-series modality, we additionally report a test-set paired-bootstrap 95% confidence interval for the paired V+A versus V+A+TS comparison.

5.3 Uncertainty Quantification via Temperature Sampling

While standard evaluation metrics capture the model's accuracy on congruent data, real-world veterinary and ethological applications demand robust Uncertainty Quantification (UQ). Traditional LLM softmax outputs are frequently miscalibrated and entangle aleatoric (data) and epistemic (model) uncertainty. To evaluate Meow-Omni 1's ability to recognize inherently ambiguous or contradictory multimodal signals, we employ a sampling-based predictive entropy approach.

5.3.1 Conflict Dataset

We design a controlled inference experiment to probe sensitivity to modal disagreement. The congruent group contains 50 MeowBench instances for which video, audio, and TS share the same future-behavior label. The conflict group contains 50 synthesized examples in which the video-audio pair and the injected TS sample carry different behavior labels. These deliberately mismatched examples test response to contradiction; they are not observations of naturally occurring hidden distress.

5.3.2 Predictive Entropy Formulation

Rather than relying on a single deterministic decoding pass, we force the model to reveal its internal uncertainty by sampling from its output distribution multiple times. For each instance in both the Control and Conflict groups, we execute $N = 10$ independent stochastic forward passes using a generation temperature of $T = 0.7$.

Let C represent the set of unique future-behavior categories generated across the N samples for a multimodal sequence $\mathcal{H}_t$. We calculate the empirical probability $\hat{p}(c)$ of predicting behavior category c. Predictive Shannon entropy is then:

$$H(\mathcal{H}_t) = -\sum_{c \in C} \hat{p}(c) \log \hat{p}(c) \tag{3}$$

A low entropy score ($H \approx 0$) indicates high confidence and modal agreement. Conversely, a high entropy score indicates that the model has detected the injected ambiguity, correctly distributing its probability mass across the competing modalities (aleatoric uncertainty) or yielding a uniform distribution due to out-of-distribution inputs (epistemic uncertainty).

5.4 Ablation Study: Modality Masking

To characterize reliance on each input stream, we apply **inference-time modality masking** to the same final Meow-Omni 1 checkpoint. The reported unimodal and bimodal configurations therefore do not correspond to separately fine-tuned models. Under unimodal masking, two sensory streams are omitted; under bimodal masking, one stream is omitted; and the full condition retains video, audio, and TS together with the text prompt.

Table 1: Comparison on MeowBench. External Qwen baselines are zero-shot; MiniCPM-o-textTS is fine-tuned with the same backbone, data, and hyperparameters as Meow-Omni 1. "Text" and "Native" denote the TS representation.

Model	Vision	Audio	TS	Accuracy (%)
Acoustic SOTA [15]		✓		36.86%
Video SOTA (Qwen3.5-122B)	✓			61.95%
TS SOTA [35]			✓	48.98%
Qwen3.5-Omni-Plus (V+A)	✓	✓		65.36%
Qwen3.5-Omni-Plus (V+TS)	✓		✓	66.21%
Qwen3.5-Omni-Plus (TS+A)		✓	✓	42.15%
Qwen3.5-Omni-Plus (V+A+TS)	✓	✓	✓	66.89%
MiniCPM-o-textTS (controlled)	✓	✓	Text	68.09%
Meow-Omni 1 (Ours)	✓	✓	Native	**71.16%**

6 Results

In this section, we present a quantitative evaluation of Meow-Omni 1. We first compare our architecture against SOTA baselines, followed by a comprehensive ablation study and an analysis of the model's uncertainty under signal conflict.

6.1 Main Results and Controlled TS-Representation Comparison

As shown in Table 1, Meow-Omni 1 achieves 71.16% top-1 accuracy. The controlled MiniCPM-o-textTS baseline reaches 68.09% under the same backbone, Meow-10K training data, and Stage 2 hyperparameters. Because the two systems differ only in whether TS is represented by the native encoder-projector path or by a structured textual summary, the 3.07-percentage-point advantage provides direct evidence that preserving the temporal signal through native encoding is more effective than compressing it into summary statistics under this experimental setup.

Meow-Omni 1 also exceeds the zero-shot Qwen3.5-Omni-Plus V+A+TS baseline (66.89%) by 4.27 points. Unlike the MiniCPM-o-textTS control, this cross-model difference combines domain-specific fine-tuning, architecture, model scale, and TS-representation effects and therefore remains a system-level comparison.

Table 2: Inference-time modality masking of the same final Meow-Omni 1 checkpoint.

Modality Configuration	Vision	Audio	TS	Accuracy (%)
Unimodal (Audio)		✓		51.88
Unimodal (Vision)	✓			69.97
Unimodal (TS)			✓	55.63
Bi-modal (V+A)	✓	✓		68.43
Bi-modal (V+TS)	✓		✓	70.82
Bi-modal (TS+A)		✓	✓	54.95
Full (V+A+TS)	✓	✓	✓	**71.16**

6.2 Modality-Masking and Paired Error Analysis

Table 2 shows that vision is the strongest single modality. Adding TS to vision increases accuracy from 69.97% to 70.82%, a gain of 0.85 percentage points. Relative to V+A, adding TS raises accuracy from 68.43% to 71.16%, a gain of 2.73 points. A paired bootstrap over test-set predictions produced a 95% confidence interval of $[0.17, 5.98]$ percentage points for the V+A versus V+A+TS difference.

The paired prediction overlap provides a more direct account of this change. Adding TS to V+A corrected 51 previously incorrect predictions and introduced 35 new errors, a net improvement of 16 examples (2.73% of the 586-example benchmark). The largest directional gains occurred in `inactive_lying.resting` (+14.81 points; $n = 27$), `maintenance_nutrition.eating` (+12.50 points; $n = 16$), and `inactive_lying.crouch` (+9.26 points). For the first two classes, the 95% bootstrap intervals were $[0.00, 37.04]$ and $[0.00, 31.25]$ points, with permutation p-values of 0.070 and 0.503, respectively. These class-specific effects are therefore directional rather than statistically conclusive at the available sample sizes.

Table 3: Classes with the largest directional gains after adding TS to V+A.

Future-Behavior Class	Accuracy Gain (pp)
`inactive_lying.resting`	+14.81
`maintenance_nutrition.eating`	+12.50
`inactive_lying.crouch`	+9.26

Table 4: Predictive entropy under congruent and deliberately mismatched modalities.

Model	Congruent H (bits)	Conflict H (bits)	ΔH (bits)
Qwen3.5-Omni-Plus	0.99	1.56	+0.57
Meow-Omni 1	1.28	3.15	+1.87

The relatively small global audio increment is consistent with the supervision and task composition. Standalone audio examples are trained with descriptive captions rather than NBP class labels, and many benchmark classes are primarily postural or locomotor. Qwen3.5-Omni-Plus shows a similarly modest change from V+TS (66.21%) to V+A+TS (66.89%), suggesting that limited global audio gain is not unique to the Meow-Omni projector. A conditional vocalization-subset analysis is not reported here because the benchmark does not provide audited subset labels; we therefore do not claim a larger conditional audio effect.

6.3 Response to Synthetic Modal Conflict

Table 4 reports entropy on 50 congruent and 50 deliberately mismatched examples, with $N = 10$ samples at temperature $T = 0.7$. Meow-Omni 1's entropy increases by 1.87 bits, compared with 0.57 bits for Qwen3.5-Omni-Plus receiving TS as a textual summary; the shift is 3.28 times larger. This result is descriptive evidence that the model's predictions are more sensitive to injected modal contradiction when TS is encoded natively. It does not test naturally occurring hidden distress, establish calibration, or demonstrate clinical safety.

7 Discussion

The results establish Meow-Omni 1 as an effective future-behavior predictor on MeowBench while also delimiting what the current evidence can support. Vision carries most of the predictive signal, and physiological TS provides complementary information under inference-time masking. More importantly, the same-backbone control distinguishes TS availability from TS representation: Meow-Omni 1 reaches 71.16%, compared with 68.09% for MiniCPM-o-textTS, a 3.07-point advantage when the backbone, domain data, and fine-tuning recipe are held constant. The separate 4.27-point advantage over Qwen3.5-Omni-Plus remains a system-level comparison because the models differ in domain adaptation, architecture, scale, and TS representation.

7.1 Controlled Evidence for Native TS Encoding

The MiniCPM-o-textTS experiment provides a controlled same-backbone comparison: it holds the backbone, Meow-10K data, and fine-tuning hyperparameters fixed while varying only the representation of TS information. Its 68.09% accuracy is higher than the zero-shot Qwen3.5-Omni-Plus result, showing that domain-specific fine-tuning accounts for part of the earlier cross-model gap. Meow-Omni 1 nevertheless improves by a further 3.07 points to 71.16%, indicating that access to TS information as summary text is not equivalent to retaining the temporal signal through the native encoder-projector pathway.

7.2 Targeted Contribution of Physiological Time Series

Within the final multimodal checkpoint, the direct V-to-V+TS change is 0.85 points, whereas the paired V+A-to-V+A+TS change is 2.73 points with a bootstrap interval excluding zero. The overlap analysis further shows that corrections are concentrated in several behaviorally ambiguous, low-motion classes, but the class-specific tests are underpowered and do not individually reach conventional significance thresholds. We therefore describe TS as a targeted source of complementary evidence rather than a universal accuracy booster or proof that semantic aliasing has been resolved.

7.3 Interpreting Modal Conflict

The entropy experiment demonstrates a response to deliberately mismatched inputs. The larger entropy shift of Meow-Omni 1 relative to the text-summary baseline indicates stronger sensitivity to cross-modal contradiction, but the experiment does not represent naturally occurring occult pain or establish that entropy is a calibrated clinical alarm. Evaluation on naturally synchronized physiological and behavioral recordings, together with explicit calibration and deferral metrics, is necessary before clinical interpretation.

8 Limitations and Future Work

MeowBench is constructed from cross-session, label-matched combinations rather than naturally synchronized quad-modal observations. Expert screening and substantial agreement support the reproducibility of the plausibility assessment, but they cannot recreate the joint distribution of physiology, audio, and behavior in a single animal at a single time. Naturally synchronized collection is therefore the highest-priority extension.

The target labels are observable future behaviors and postures, not direct measurements of subjective intention. NBP may be useful as a proxy for antecedent behavioral tendency in near-free-choice settings, but this proxy can fail for long-horizon or inhibited behaviors.

The modality-masking study evaluates one multimodally trained checkpoint and therefore does not substitute for independently trained modality-specific models. The MiniCPM-o-textTS experiment addresses the principal representation confound by holding the backbone, data, and fine-tuning recipe fixed, but only its point accuracy is available in the current experimental record. Paired predictions, confidence intervals, and multi-seed variability are still needed to quantify the uncertainty of the 3.07-point native-TS advantage. Likewise, the reported per-class effects are based on small supports and require a larger, class-balanced benchmark.

External MLLM baselines are zero-shot and differ substantially in scale and accessibility. Qwen3.5-Omni-Plus is proprietary and receives TS only as a textual summary. Table 1 should therefore be read as a comparison of available systems under a common evaluation interface, not as an architecture-only comparison. The MCQ protocol improves comparability but narrows the output space; open-ended and full-label-space evaluation should accompany future benchmark releases.

Finally, predictive entropy under synthetic conflict is not a validated safety mechanism. Before veterinary deployment, the model would require calibration on naturally occurring cases, prospective evaluation, explicit abstention criteria, and expert oversight. Cross-species generalization also remains untested.

9 Conclusion

We introduced Meow-Omni 1, an open quad-modal MLLM that combines video, audio, physiological time series, and text for feline future-behavior prediction. On the 586-example MeowBench, the model reaches 71.16% accuracy. The controlled MiniCPM-o-textTS baseline reaches 68.09% with the same backbone, training data, and hyperparameters, providing a more direct 3.07-point estimate of the benefit of native rather than text-summary TS representation in this setup. Inference-time masking and paired analysis further show that vision provides the dominant signal, while TS adds a modest overall benefit and directionally larger gains in selected ambiguous classes. The expert-verification process attains substantial agreement, although the benchmark's cross-session construction and the lack of uncertainty estimates for the textTS control limit broader conclusions.

The primary contribution is therefore an open, reproducible resource and integration framework for multimodal animal-behavior research. The released model, training pipeline, Meow-10K dataset, controlled textTS baseline, and evaluation protocol provide a basis for future studies using naturally synchronized recordings, larger class-balanced evaluations, multi-seed controlled comparisons, and clinically validated uncertainty measures.

References

[1] Paul E. Rose and Lisa M. Riley. Conducting behavioural research in the zoo: A guide to ten important methods, concepts and theories. *Journal of Zoological and Botanical Gardens*, 2 (3):421–444, 2021. ISSN 2673-5636. doi: 10.3390/jzbg2030031. URL https://www.mdpi.com/2673-5636/2/3/31.

[2] Chloé Tavernier, Sohail Ahmed, Katherine Houpt, and Seong Chan Yeon. Feline vocal communication. *Journal of Veterinary Science*, 21, 01 2020. doi: 10.4142/jvs.2020.21.e18.

[3] Isabella Merola and Daniel S Mills. Systematic review of the behavioural assessment of pain in cats. *Journal of Feline Medicine and Surgery*, 18(2):60–76, Feb 2016. ISSN 1532-2750. doi: 10.1177/1098612X15578725. URL https://doi.org/10.1177/1098612X15578725.

[4] Andrea M. Green and Dora E. Angelaki. Multisensory integration: resolving sensory ambiguities to build novel representations. *Current Opinion in Neurobiology*, 20(3):353–360, Jun 2010. ISSN 1873-6882. doi: 10.1016/j.conb.2010.04.009.

[5] OpenBMB Team. MiniCPM-o 4.5: A next-generation omni-modal large language model. https://huggingface.co/openbmb/MiniCPM-o-4_5, 2025. Accessed: 2026-04-19.

[6] Yicheng Zou, Dongsheng Zhu, Lin Zhu, Tong Zhu, Yunhua Zhou, Peiheng Zhou, Xinyu Zhou, Dongzhan Zhou, Zhiwang Zhou, Yuhao Zhou, Bowen Zhou, Zhanping Zhong, Zhijie Zhong, Haiteng Zhao, Penghao Zhao, Xiaomeng Zhao, Zhiyuan Zhao, Yechen Zhang, Jin Zhang, Wenwei Zhang, Hongjie Zhang, Zhuo Zhang, Wenlong Zhang, Bo Zhang, Chao Zhang, Chen Zhang, Yuhang Zang, Fei Yuan, Jiakang Yuan, Jiashuo Yu, Jinhui Yin, Haochen Ye, Qian Yao, Bowen Yang, Danni Yang, Kaichen Yang, Ziang Yan, Jun Xu, Yicheng Xu, Wanghan Xu, Xuenan Xu, Chao Xu, Ruiliang Xu, Shuhao Xing, Long Xing, Xinchen Xie, Ling-I Wu, Zijian Wu, Zhenyu Wu, Lijun Wu, Yue Wu, Jianyu Wu, Wen Wu, Fan Wu, Xilin Wei, Qi Wei, Bingli Wang, Rui Wang, Ziyi Wang, Zun Wang, Yi Wang, Haomin Wang, Yizhou Wang, Lintao Wang, Yiheng Wang, Longjiang Wang, Bin Wang, Jian Tong, Zhongbo Tian, Huanze Tang, Chen Tang, Shixiang Tang, Yu Sun, Qiushi Sun, Xuerui Su, Qisheng Su, Chenlin Su, Demin Song, Jin Shi, Fukai Shang, Yuchen Ren, Pengli Ren, Xiaoye Qu, Yuan Qu, Jiantao Qiu, Yu Qiao, Runyu Peng, Tianshuo Peng, Jiahui Peng, Qizhi Pei, Zhuoshi Pan, Linke Ouyang, Wenchang Ning, Yichuan Ma, Zerun Ma, Ningsheng Ma, Runyuan Ma, Chengqi Lyu, Haijun Lv, Han Lv, Lindong Lu, Kuikun Liu, Jiangning Liu, Yuhong Liu, Kai Liu, Hongwei Liu, Zhoumianze Liu, Mengjie Liu, Ziyu Liu, Wenran Liu, Yang Liu, Liwei Liu, Kaiwen Liu, Junyao Lin, Junming Lin, Tianyang Lin, Dahua Lin, Jianze Liang, Linyang Li, Peiji Li, Zonglin Li, Zehao Li, Pengze Li, Guoyan Li, Lingkai Kong, Linglin Jing, Zhenjiang Jin, Feifei Jiang, Qian Jiang, Junhao Huang, Zixian Huang, Haian Huang, Zhouqi Hua, Han Hu, Linfeng Hou, Yinan He, Conghui He, Tianyao He, Xu Guo, Qipeng Guo, Aijia Guo, Yuzhe Gu, Lixin Gu, Jingyang Gong, Qiming Ge, Jiaye Ge, Songyang Gao, Jianfei Gao, Xinyu Fang, Caihua fan, Yue Fan, Yanhui Duan, Zichen Ding, Shengyuan Ding, Xuanlang Dai, Erfei Cui, Ganqu Cui, Pei Chu, Tao Chu, Guangran Cheng, Yu Cheng, Kai Chen, Yongkang Chen, Chiyu Chen, Guanzhou Chen, Qiaosheng Chen, Sitao Chen, Xin Chen, Haojiong Chen, Yicheng Chen, Weihan Cao, Yuhang Cao, Qinglong Cao, and Lei Bai. Intern-s1-pro: Scientific multimodal foundation model at trillion scale, 2026. URL https://arxiv.org/abs/2603.25040.

[7] Lei Bai, Zhongrui Cai, Maosong Cao, Weihan Cao, Chiyu Chen, Haojiong Chen, Kai Chen, Pengcheng Chen, Ying Chen, Yongkang Chen, Yu Cheng, Yu Cheng, Pei Chu, Tao Chu, Erfei Cui, Ganqu Cui, Long Cui, Ziyun Cui, Nianchen Deng, Ning Ding, Nanqin Dong, Peijie Dong, Shihan Dou, Sinan Du, Haodong Duan, Caihua Fan, Ben Gao, Changjiang Gao, Jianfei Gao, Songyang Gao, Yang Gao, Zhangwei Gao, Jiaye Ge, Qiming Ge, Lixin Gu, Yuzhe Gu, Aijia Guo, Qipeng Guo, Xu Guo, Conghui He, Junjun He, Yili Hong, Siyuan Hou, Caiyu Hu, Hanglei Hu, Jucheng Hu, Ming Hu, Zhouqi Hua, Haian Huang, Junhao Huang, Xu Huang, Zixian Huang, Zhe Jiang, Lingkai Kong, Linyang Li, Peiji Li, Pengze Li, Shuaibin Li, Tianbin Li, Wei Li, Yuqiang Li, Dahua Lin, Junyao Lin, Tianyi Lin, Zhishan Lin, Hongwei Liu, Jiangning Liu, Jiyao Liu, Junnan Liu, Kai Liu, Kaiwen Liu, Kuikun Liu, Shichun Liu, Shudong Liu, Wei Liu, Xinyao Liu, Yuhong Liu, Zhan Liu, Yinquan Lu, Haijun Lv, Hongxia Lv, Huijie Lv, Qidang Lv, Ying Lv, Chengqi Lyu, Chenglong Ma, Jianpeng Ma, Ren Ma, Runmin Ma, Runyuan Ma, Xinzhu Ma, Yichuan Ma, Zihan Ma, Sixuan Mi, Junzhi Ning, Wenchang Ning,

Xinle Pang, Jiahui Peng, Runyu Peng, Yu Qiao, Jiantao Qiu, Xiaoye Qu, Yuan Qu, Yuchen Ren, Fukai Shang, Wenqi Shao, Junhao Shen, Shuaike Shen, Chunfeng Song, Demin Song, Diping Song, Chenlin Su, Weijie Su, Weigao Sun, Yu Sun, Qian Tan, Cheng Tang, Huanze Tang, Kexian Tang, Shixiang Tang, Jian Tong, Aoran Wang, Bin Wang, Dong Wang, Lintao Wang, Rui Wang, Weiyun Wang, Wenhai Wang, Yi Wang, Ziyi Wang, Ling-I Wu, Wen Wu, Yue Wu, Zijian Wu, Linchen Xiao, Shuhao Xing, Chao Xu, Huihui Xu, Jun Xu, Ruiliang Xu, Wanghan Xu, GanLin Yang, Yuming Yang, Haochen Ye, Jin Ye, Shenglong Ye, Jia Yu, Jiashuo Yu, Jing Yu, Fei Yuan, Bo Zhang, Chao Zhang, Chen Zhang, Hongjie Zhang, Jin Zhang, Qiaosheng Zhang, Qiuyinzhe Zhang, Songyang Zhang, Taolin Zhang, Wenlong Zhang, Wenwei Zhang, Yechen Zhang, Ziyang Zhang, Haiteng Zhao, Qian Zhao, Xiangyu Zhao, Xiangyu Zhao, Bowen Zhou, Dongzhan Zhou, Peiheng Zhou, Yuhao Zhou, Yunhua Zhou, Dongsheng Zhu, Lin Zhu, and Yicheng Zou. Intern-s1: A scientific multimodal foundation model, 2025. URL https://arxiv.org/abs/2508.15763.

[8] Masato Hagiwara. Aves: Animal vocalization encoder based on self-supervision, 2022. URL https://arxiv.org/abs/2210.14493.

[9] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units, 2021. URL https://arxiv.org/abs/2106.07447.

[10] Masato Hagiwara, Benjamin Hoffman, Jen-Yu Liu, Maddie Cusimano, Felix Effenberger, and Katie Zacarian. Beans: The benchmark of animal sounds, 2022. URL https://arxiv.org/abs/2210.12300.

[11] Andrea Burns, Lauren Harrell, Bart van Merriënboer, Vincent Dumoulin, Jenny Hamer, and Tom Denton. Perch 2.0 transfers 'whale' to underwater tasks, 2025. URL https://arxiv.org/abs/2512.03219.

[12] Jules Cauzinille, Benoît Favre, Ricard Marxer, Dena Clink, Abdul Ahmad, and Arnaud Rey. Investigating self-supervised speech models' ability to classify animal vocalizations: The case of gibbon's vocal signatures. pages 132–136, 09 2024. doi: 10.21437/Interspeech.2024-1096.

[13] Lukas Rauch, René Heinrich, Ilyass Moummad, Alexis Joly, Bernhard Sick, and Christoph Scholz. Can masked autoencoders also listen to birds?, 2025. URL https://arxiv.org/abs/2504.12880.

[14] Chiara Semenzin, Faadil Mustun, Roberto Dessi, Alexis Emanuelli, Pierre Orhan, Gonzalo G. de Polavieja, Yair Lakretz, and German Sumbre. Dolph2vec: Self-supervised representations of dolphin vocalizations, 2026. URL https://openreview.net/forum?id=QGAFX5kcR5.

[15] Stavros Ntalampiras, Luca Andrea Ludovico, Giorgio Presti, Emanuela Prato Previde, Monica Battini, Simona Cannas, Clara Palestrini, and Silvana Mattiello. Automatic classification of cat vocalizations emitted in different contexts. *Animals*, 9(8):543, 2019. URL https://doi.org/10.3390/ani9080543.

[16] Jennifer Sun, Hao Zhou, Long Zhao, Liangzhe Yuan, Bryan Seybold, David Hendon, Florian Schroff, David Ross, Hartwig Adam, Bo Hu, and Ting Liu. Video foundation models for animal behavior analysis, 07 2024.

[17] Hung-Shuo Chang, Yue-Cheng Yang, Yu-Hsi Chen, Wei-Hsin Chen, Chien-Yao Wang, James C. Liao, Chien-Chang Chen, Hen-Hsen Huang, and Hong-Yuan Mark Liao. A universal action space for general behavior analysis, 2026. URL https://arxiv.org/abs/2602.09518.

[18] Meiqi Sun, Zhonghan Zhao, Wenhao Chai, Hanjun Luo, Shidong Cao, Yanting Zhang, Jenq-Neng Hwang, and Gaoang Wang. Uniap: Towards universal animal perception in vision via few-shot learning, 2023. URL https://arxiv.org/abs/2308.09953.

[19] Shaokai Ye, Anastasiia Filippova, Jessy Lauer, Steffen Schneider, Maxime Vidal, Tian Qiu, Alexander Mathis, and Mackenzie Weygandt Mathis. Superanimal pretrained pose estimation models for behavioral analysis. *Nature Communications*, 15(1):5165, Jun 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-48792-2.

[20] Markus Marks, Jin Qiuhan, Oliver Sturman, Lukas von Ziegler, Sepp Kollmorgen, Wolfger von der Behrens, Valerio Mante, Johannes Bohacek, and Mehmet Fatih Yanik. Deep-learning based identification, tracking, pose estimation, and behavior classification of interacting primates and mice in complex environments. *Nature Machine Intelligence*, 4(4):331–340, Apr 2022. ISSN 2522-5839. doi: 10.1038/s42256-022-00477-5.

[21] Marcelo Feighelstein, Lea Henze, Sebastian Meller, Ilan Shimshoni, Ben Hermoni, Michael Berko, Friederike Twele, Alexandra Schütter, Nora Dorn, Sabine Kästner, Lauren Finka, Stelio P. L. Luna, Daniel S. Mills, Holger A. Volk, and Anna Zamansky. Explainable automated pain recognition in cats. *Scientific Reports*, 13(1):8973, Jun 2023. ISSN 2045-2322. doi: 10.1038/s41598-023-35846-6.

[22] Cheng Fang, Tiemin Zhang, Haikun Zheng, Junduan Huang, and Kaixuan Cuan. Pose estimation and behavior classification of broiler chickens based on deep neural networks. *Computers and Electronics in Agriculture*, 180:105863, 2021. ISSN 0168-1699. doi: https://doi.org/10.1016/j.compag.2020.105863. URL https://www.sciencedirect.com/science/article/pii/S0168169920305159.

[23] Dema Saleh, Moemen Ahmed, Mai Zaafan, Yasmine Farouk, and Ayman Atia. A pharmacology toolkit for animal pose estimation, tracking and analysis. In *2023 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, pages 1–7, 2023. doi: 10.1109/MIUCC58832.2023.10278344.

[24] Edoardo Fazzari, Fabio Carrara, Fabrizio Falchi, Cesare Stefanini, and Donato Romano. Using ai to decode the behavioral responses of an insect to chemical stimuli: towards machine-animal computational technologies. *International Journal of Machine Learning and Cybernetics*, 15 (5):1985–1994, May 2024. ISSN 1868-808X. doi: 10.1007/s13042-023-02009-y.

[25] Reza Arablouei, Liang Wang, Lachlan Currie, Jodan Yates, Flavio A.P. Alvarenga, and Greg J. Bishop-Hurley. Animal behavior classification via deep learning on embedded systems. *Computers and Electronics in Agriculture*, 207:107707, 2023. ISSN 0168-1699. doi: https://doi.org/10.1016/j.compag.2023.107707. URL https://www.sciencedirect.com/science/article/pii/S0168169923000959.

[26] Thai-Ha Dang, Ngoc-Hai Dang, Viet-Thang Tran, and Wan-Young Chung. A lorawan-based smart sensor tag for cow behavior monitoring. In *2022 IEEE Sensors*, pages 1–4, 2022. doi: 10.1109/SENSORS52175.2022.9967209.

[27] Zhixin Pan, Huihui Chen, Weizhao Zhong, Aiguo Wang, and Chundi Zheng. A cnn-based animal behavior recognition algorithm for wearable devices. *IEEE Sensors Journal*, 23(5): 5156–5164, 2023. doi: 10.1109/JSEN.2023.3239015.

[28] Md Ariful Islam Mozumder, Tagne Poupi Theodore Armand, Rashadul Islam Sumon, Shah Muhammad Imtiyaj Uddin, and Hee-Cheol Kim. Automated pipeline for robust cat activity detection based on deep learning and wearable sensor data. *Sensors*, 24(23), 2024. ISSN 1424-8220. doi: 10.3390/s24237436. URL https://www.mdpi.com/1424-8220/24/23/7436.

[29] Anthropic. Introducing claude opus 4.7. https://www.anthropic.com/news/claude-opus-4-7, apr 2026. Large language model. API identifier: claude-opus-4-7. Knowledge cutoff: January 2026.

[30] Google DeepMind. Gemini 3.1 pro model card. https://deepmind.google/models/model-cards/gemini-3-1-pro/, feb 2026. Large language model. Multimodal reasoning model with 1M token context window. API identifier: gemini-3.1-pro. Knowledge cutoff: 2026.

[31] Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

[32] Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, Baosong Yang, Bin Zhang, Ziyang Ma, Xipin Wei, Shuai Bai, Keqin Chen, Xuejing Liu, Peng Wang, Mingkun Yang, Dayiheng Liu,

Xingzhang Ren, Bo Zheng, Rui Men, Fan Zhou, Bowen Yu, Jianxin Yang, Le Yu, Jingren Zhou, and Junyang Lin. Qwen3-omni technical report, 2025. URL https://arxiv.org/abs/2509.17765.

[33] Yinuo Jing, Ruxu Zhang, Kongming Liang, Yongxiang Li, Zhongjiang He, Zhanyu Ma, and Jun Guo. Animal-bench: Benchmarking multimodal video models for animal-centric video understanding. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=DexM7d1H6e.

[34] Jun Chen, Ming Hu, Darren J. Coker, Michael L. Berumen, Blair Costelloe, Sara Beery, Anna Rohrbach, and Mohamed Elhoseiny. Mammalnet: A large-scale video benchmark for mammal recognition and behavior understanding, 2023. URL https://arxiv.org/abs/2306.00576.

[35] Guanyu Chen, Yoshinari Takegawa, Kohei Matsumura, Hiroki Watanabe, and Keiji Hirata. Cat and dog behavior recognition method using deep learning approach based on inertial measurement unit sensor data. *Sensors and Materials*, 37(3(3)):1073–1098, Mar 2025. ISSN 0914-4935. doi: 10.18494/SAM5359. Published March 28, 2025.

[36] Michelle Smit, Seer J Ikurior, Rene A Corner-Thomas, Christopher J Andrews, Ina Draganova, and David G Thomas. The use of triaxial accelerometers and machine learning algorithms for behavioural identification in domestic cats (felis catus): A validation study. *Sensors*, 23(16): 7165, 2023.

[37] Carolyn E Dunford, Nikki J Marks, Rory P Wilson, and D Michael Scantlebury. Identifying animal behaviours from accelerometers: Improving predictive accuracy of machine learning by refining the variables selected, data frequency, and sample duration. *Ecology and Evolution*, 14 (5):e11380, 2024.

[38] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1728–1738, 2021.

[39] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[40] Silvio Giancola and Bernard Ghanem. Temporally-aware feature pooling for action spotting in soccer broadcasts. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4490–4499, 2021.

[41] Tsung-Ming Tai, Sofia Casarin, Andrea Pilzer, Werner Nutt, and Oswald Lanz. Action-guided attention for video action anticipation. *arXiv preprint arXiv:2603.01743*, 2026.

[42] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 776–780. IEEE, 2017.

[43] Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer. *arXiv preprint arXiv:2104.01778*, 2021.

[44] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. URL https://arxiv.org/abs/2502.13923.

[45] Eduardo Fonseca, Jordi Pons, Xavier Favory, Frederic Font, Dmitry Bogdanov, Andres Ferraro, Sergio Oramas, Alastair Porter, and Xavier Serra. Freesound datasets: A platform for the creation of open audio datasets. In *ISMIR*, pages 486–493, 2017.

[46] Tao Jiang. cat_class. Hugging Face Dataset, 2026. URL https://huggingface.co/datasets/taozi555/cat_class. Accessed: 2026-04-20.

[47] Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, et al. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*, 2025.

Appendix

A Data Processing Pipeline

A.1 Bio Dataset Preprocessing

Cat behavioral biosignal data are scarce, particularly datasets that combine high temporal resolution with behavior annotations. We use two accelerometer datasets from peer-reviewed studies [36, 37]. Their source labels include 26 behavior names; the observed inventory is listed in Appendix E.

In terms of signal processing, the original high-frequency accelerometer data (e.g., 30 Hz or 60 Hz) are aggregated into second-level signals by averaging the measurements within each second. This approach reduces high-frequency noise (a common practice in animal behaviour analysis) and mitigates discrepancies in sampling rates across different studies, improving data comparability.

We then construct a task-specific dataset. Invalid or ambiguous labels (e.g., "other" categories) and transient movements that do not provide a stable prediction target are excluded. The cleaned time series for each cat are segmented into fixed-length windows (5, 7, 10, and 15 seconds). The behavior label at a future offset (1, 2, 3, or 5 seconds) is used as the target, yielding the NBP task. No window crosses individuals or temporal discontinuities.

Finally, for each sample we generate a user query and a corresponding response via prompt-based generation (see Appendix D) rather than fixed templates. Three large language models (DeepSeek, GPT, and Gemini) are employed to produce diverse natural-language formulations. The resulting 383,853 labeled TS samples are used for projector alignment in Stage 1 (Section 4.2); a random subset of 2,000 samples is included in Meow-10K to prevent over-representation of the TS modality.

A.2 Video Dataset Preprocessing

To extract high-quality, action-centric clips from raw, unannotated cat videos taken from an open-source dataset [38], we employ an automated Vision-Language Model (VLM) based preprocessing pipeline. The pipeline uses *Qwen2-VL-72B-Instruct* [39] to perform zero-shot frame analysis and consists of four primary stages:

- **Global Action Detection:** The raw video is sampled into coarse frames (up to 16 frames at approximately 1.5 s intervals). The VLM analyzes these frames to detect significant changes in body posture or position. A gap rescan (approximately 0.5 s intervals) is applied to temporal gaps exceeding 2.0 s to avoid missing precursor actions.
- **Precise Temporal Localization:** For each detected anchor, a dense sequence of frames is extracted within a 4.0 s window at a fine interval of 0.20 s. The VLM compares consecutive pairs to pinpoint the exact onset frame t_{anchor} where the action begins.
- **Verification:** A three-frame verification step compares the subject's state several seconds before the suspected action with frames taken precisely at and shortly after the onset, confirming a definitive state change.
- **Clip Generation and Annotation:** Upon verification, a clip is extracted with a fixed observation window $T_{\text{obs}} = 6.0\,\text{s}$. Following asymmetric windowing practices in action anticipation [40, 41], the temporal window is heavily biased towards the pre-action phase (spanning from $t_{\text{anchor}} - 0.85 \times T_{\text{obs}}$ to $t_{\text{anchor}} + 0.15 \times T_{\text{obs}}$). Finally, the VLM generates a sequence caption and a specific action label for the cropped segment.

Each VLM-generated action label is subsequently mapped to one of the 26 future-behavior classes by a secondary LLM (Qwen3.5-35B). This mapping was reviewed and validated by the same expert groups that curated MeowBench, ensuring consistency with the biological ground truth. All video clips processed in this way are video-only (no accompanying audio track); synchronized audio-video pairs are handled through the separate AudioSet pipeline described below.

A.3 Audio Dataset Preprocessing

Audio data originate from two independent sources and are processed through distinct pipelines.

AudioSet-Derived Synchronized A/V Clips. The first source is AudioSet [42], a large-scale human-labelled audio event dataset from which we use video clips containing cat vocalisations. Since

these clips carry both video and audio streams, excerpt boundaries are inherited directly from the video preprocessing stage, ensuring temporal alignment by construction.

Each extracted audio track is first verified for cat sound presence using an Audio Spectrogram Transformer (AST) classifier fine-tuned on AudioSet [43]. The classifier scores each clip against cat-related AudioSet labels (*cat*, *meow*, *purr*, *caterwaul*). Clips scoring above 0.10 are retained; clips below 0.03 are discarded. Marginal clips (0.03–0.10) undergo stationary noise reduction with a suppression factor of 0.85, after which the classifier is re-applied; a clip is retained only if the post-denoising score meets the 0.03 threshold.

Verified clips are then passed to Qwen2.5-VL-7B [44], which jointly analyses the video and audio to produce an audio-focused caption and an action label. The model is explicitly instructed to use visual context only as interpretive reference, grounding the final description and classification in the acoustic content. Simultaneously, the model assigns a score from 1 to 10 to each of the 26 future-behavior classes; scores are normalized to form a probability distribution used as the training target for these samples.

Standalone Audio Clips. The second source consists of standalone audio recordings from Freesound [45] and the huggingface/taozi555/cat_class dataset [46]. These carry short human-written captions but no associated video. No sound verification step is required; the original captions serve as the starting point.

Processing is carried out via a text-only pipeline using Qwen3.5-35B-A3B-GPTQ-Int4 [31], served locally or through an OpenAI-compatible API. The original caption is expanded into a behavior-relevant semantic description grounded in the source text, with uncertain inferences phrased tentatively. No future-behavior label is assigned; the expanded caption itself serves as the training target. Thus, standalone audio contributes descriptive supervision rather than direct NBP classification supervision.

A.4 Meow-10K Dataset Assembly

All samples produced by the preceding pipelines are consolidated into the Meow-10K training set. To prevent the abundant TS-only data from overwhelming the other modalities, we randomly subsample 2,000 sequences from the full 383,853 TS pool. The remaining components comprise:

- Video-only clips from the Bain dataset (Section A.2);
- Naturally synchronized audio-video pairs from AudioSet (Section A.3);
- Standalone audio captions from Freesound and cat_class (Section A.3);
- A small set of expert-verified synthetic quad-modal samples, constructed following the MeowBench protocol (Section 4.4), where a synchronized A/V pair is matched with a TS sample sharing the same future-behavior label and the combination is verified by one of the expert groups.

The final Meow-10K dataset contains 10,831 samples, each consisting of a natural-language query-response pair and, where applicable, a future-behavior label from the source-label inventory.

B Baselines Detailed Discussion

B.1 Video Baseline

We implemented a SOTA video baseline using the Qwen3.5-122B-A10B large vision-language model. The evaluation was conducted in a zero-shot, audio-blind setting with the model's internal "thinking" module disabled to strictly isolate and assess native visual-temporal reasoning capabilities. The visual preprocessing pipeline included uniform temporal sampling of 8 frames per input clip, which were provided as sequential inputs to the model. The baseline was evaluated on MeowBench to benchmark the model's visual understanding against ground-truth annotations. During evaluation, each model response was processed through a regular-expression-based parsing logic to extract the predicted multiple-choice letter (A, B, C, or D), which was then mapped to the corresponding benchmark answer. Accuracy was employed as the primary evaluation metric to ensure parity with the audio baseline.

B.2 Biosignal Baseline

The biosignal baseline estimates how much future-behavior information can be extracted from TS alone. We based this baseline on the cat and dog IMU behaviour recognition model proposed in

[35], and adapted it to our benchmark format. Since the original work was designed for a different data setting, our implementation should be regarded as a paper-style reproduction rather than a full reproduction of the original experiment. In particular, we retained the core temporal modelling idea of combining convolutional feature extraction with recurrent sequence modelling, while simplifying the architecture to fit the structure of our benchmark data.

The final baseline architecture follows the main design philosophy of [35]. First, the $(5, 3)$ input tensor is transposed to channel-first format and passed through two 1D convolution layers. The first convolution layer uses 32 channels, followed by ReLU activation, max pooling, and dropout. The second convolution layer increases the feature dimension to 64 channels, followed again by ReLU and dropout. The output sequence is then transposed back to time-major format and fed into a single-layer LSTM with hidden size 64. The hidden state of the last time step is batch-normalized and passed to a two-layer fully connected classifier with dropout, producing the final logits over the 10 future-behavior classes. This design preserves the paper's idea of combining local temporal pattern extraction and sequential modelling, while remaining lightweight enough for our short benchmark sequences. Figure 2 shows the architecture of the baseline; the main difference from the original is that we omit the manually extracted feature branch (right-hand side of the original figure).

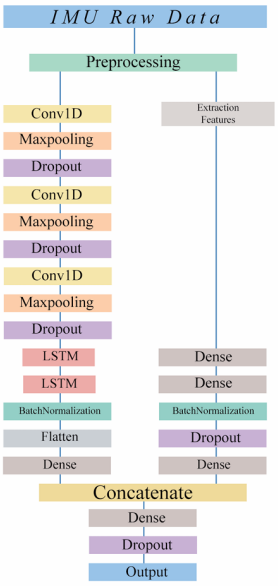


Figure 2: Baseline for biosignal modality, adapted from [35].

For training, the benchmark data were split into train, validation, and test sets with a ratio of approximately 70:10:20 using stratified sampling over the future-behavior labels. The model was optimized with Adam using a learning rate of 10^{-4} and a batch size of 64. To avoid selecting the final model based on the test set, we adopted early stopping using validation accuracy only. The checkpoint with the best validation accuracy was saved, and the final test accuracy was computed once using this checkpoint after training stopped. In addition to the classification accuracy on the future-behavior labels, we also evaluated the model in the original multiple-choice benchmark format by mapping the predicted future-behavior class back to the corresponding answer option (A/B/C/D) for each sample. Raw prediction outputs, including predicted labels and logits, were stored for later benchmark analysis and potential recalculation if problematic items were removed.

This baseline provides a unimodal estimate of how much future-behavior information can be extracted from short acceleration signals and connects prior wearable-sensor behavior recognition to the MeowBench evaluation format.

B.3 Audio Baseline

We implemented three top-performance audio baselines from Ntalampiras et al. [15]: a support vector machine (SVM), a directed acyclic graph hidden Markov model (DAG-HMM), and a class-specific hidden Markov model. All methods used the same acoustic preprocessing pipeline, including silence

removal, mel-frequency cepstral coefficient (MFCC)-based features, and features describing temporal variation in the signal. The baselines were trained on our 22-class audio training set and evaluated on MeowBench in the audio-only setting.

MeowBench contains 26 answer-option labels, whereas our audio-only training set covers 22 classes that correspond to audible behaviours. The remaining eight labels are primarily pose-, motion-, scene-, or sensor-oriented categories, which fall outside the audio-only supervision space; for those eight classes, audio-based prediction was not attempted. During evaluation, each model predicted one class label from the 22-class set for an input clip; when the gold label belonged to the 22 audible classes, the predicted label was mapped to the corresponding answer option in the multiple-choice benchmark. Accuracy over the class-matched subset was used as the main evaluation metric.

B.4 Omni Baseline

We evaluate **Qwen3.5-Omni-Plus** [47], accessed via the DashScope API, as our omni-modal baseline. Qwen3.5-Omni-Plus is a proprietary instruction-following variant built upon the Qwen3-Omni architecture [47], a Thinker–Talker Mixture-of-Experts end-to-end multimodal model that natively processes text, images, audio, and video within a single unified model. We use Qwen3.5-Omni-Plus rather than the open-weight Qwen3-Omni directly for two practical reasons: the open-weight Qwen3-Omni release does not provide a publicly accessible instruction-following API, and its Captioner variant does not support video input, making both unsuitable for our quad-modal evaluation format. Qwen3-Omni achieves open-source SOTA performance on 32 of 36 audio and audio-visual benchmarks, surpassing closed-source systems including Gemini 2.5 Pro and GPT-4o-Transcribe [47], establishing it as a strong zero-shot upper-bound comparator.

Prompting strategy The model receives video and audio as native modality inputs. As Qwen3.5-Omni-Plus does not support raw TS as a modality, bio-signal data is injected as a structured textual summary computed from the accelerometer `.npy` array, reporting array shape, mean absolute magnitude, average temporal change, and per-channel mean, standard deviation, minimum, and maximum across the first eight channels. This summary is prepended to the MeowBench MCQ question. The system prompt instructs the model to treat all three inputs as co-equal evidence and respond with a single letter on the first line followed by exactly two explanatory sentences. Each MCQ is formulated as a *structured decision problem*, where the model must select one option from four candidates (A/B/C/D), each corresponding to a semantically distinct hypothesis about the underlying behaviour or cross-modal relationship.

Ablation conditions To isolate each modality's contribution, we evaluate the same backbone under four input configurations: *Video + Audio*, *Video + Bio*, *Audio + Bio*, and *Video + Audio + Bio*. All conditions use greedy decoding (temperature $= 0$), ensuring that performance differences are attributable solely to modality availability rather than model or sampling variation.

C Benchmark Reliability and Error Analysis

C.1 Expert Screening Protocol

The 645 candidate tuples were divided approximately equally among three expert groups (sizes 3, 3, and 2). Each rater independently assigned one of three plausibility ratings before discussion. Unanimous three-category agreement was observed for 588 candidates (91.2%). For the binary distinction Plausible versus Not-Plausible, Fleiss' κ was 0.803 (SE $= 0.042$, 95% CI $[0.721, 0.885]$, $z = 19.20$, $p < 0.0001$). Consensus discussion retained 586 examples and rejected 59. Rejections primarily reflected an obvious behavior-signal mismatch, temporal inconsistency between a motion burst and the observed clip, or visible environmental constraint. These statistics describe reproducibility of the plausibility screen; they do not make cross-session tuples equivalent to naturally synchronized recordings.

C.2 Per-Class Performance and Common Confusions

Table 6 reports the class-level results available in the experimental record. The unequal supports show why aggregate accuracy can be dominated by visually distinctive, high-support classes. Table 7 lists the five most frequent reported confusions. Several involve low-motion postures or behaviorally dissimilar distractors, motivating class-balanced evaluation and full-label-space diagnostics in future benchmark versions.

Table 5: Summary of MeowBench expert screening.

Quantity	Value
Initial candidates	645
Retained after consensus	586
Rejected after consensus	59
Unanimous pre-consensus ratings	588/645 (91.2%)
Fleiss' κ (binary)	0.803
95% CI for κ	$[0.721, 0.885]$

Table 6: Reported per-class accuracy for Meow-Omni 1 with V+A+TS.

Class	Support	Correct	Accuracy (%)
`active_walking`	145	130	89.7
`inactive_lying.down`	151	113	74.8
`inactive_sitting.stationary`	42	28	66.7
`active_rubbing`	6	4	66.7
`active_playfight.fighting`	10	5	50.0
`active_jumping.vertical`	11	5	45.5

D Prompt Design for Query Generation

We design four types of prompts to generate natural language queries for the TS prediction task. Here, A denotes the length of the input time window, and B denotes the prediction horizon (i.e., B seconds after the window).

(1) Prediction with window length A

> Generate 300 different natural language queries asking to predict the cat's behavior after a time window of {A} seconds.
>
> Each query must include "<|ts_start|><|ts_unit|><|ts_end|>".
>
> Example (keep "{A}" unchanged):
> "Given a {A}-second window <|ts_start|><|ts_unit|><|ts_end|>, predict the behavior."
>
> Encourage diverse wording; output a Python list of strings.

(2) Prediction with delay B

> Generate 300 different natural language queries asking to predict the cat's behavior {B} seconds after a given TS segment.
>
> Each query must include "<|ts_start|><|ts_unit|><|ts_end|>".
>
> Example (keep "{B}" unchanged):
> "Given <|ts_start|><|ts_unit|><|ts_end|>, predict the behavior after {B} seconds."
>
> Encourage diverse wording; output a Python list of strings.

(3) Prediction with window A and delay B

Table 7: Most frequent reported ground-truth-to-prediction confusions.

Ground Truth	Prediction	Count
inactive_sitting.stationary	maintenance_grooming	10
inactive_sitting.stationary	active_playfight.fighting	9
inactive_lying.down	active_climbing	8
active_walking	maintenance_littering.defecating	7
inactive_lying.down	inactive_standing.stationary	7

Generate 300 different natural language queries asking to predict the cat's behavior {B} seconds after a time window of {A} seconds.

Each query must include "<|ts_start|><|ts_unit|><|ts_end|>".

Example (keep "{A}" and "{B}" unchanged):
"Given a {A}-second window <|ts_start|><|ts_unit|><|ts_end|>, predict the behavior after {B} seconds."

Encourage diverse wording; output a Python list of strings.

(4) Basic prediction without temporal parameters

Generate 300 different natural language queries asking to predict the cat's behavior from a given TS segment.

Each query must include "<|ts_start|><|ts_unit|><|ts_end|>".
Example:
"Given <|ts_start|><|ts_unit|><|ts_end|>, predict the cat's behavior."

Encourage diverse wording; output a Python list of strings.

(5) Response generation from feature summaries

Generate 300 different natural language assistant responses for a cat behavior prediction task.

You are given:
- a feature summary describing the acceleration pattern of one behavior class: {features}
- the target behavior label: {label}

Write one-sentence responses that infer the behavior from the described signal pattern. The response should explain the prediction briefly rather than only repeating the label.

Example:
"The recurring peaks in the acceleration signal confirm the cat is likely walking forward."

Encourage diverse wording; output a Python list of strings.

E Feature Construction

The feature summaries provided in the prompt are constructed based on the statistical characteristics of the acceleration data for each behaviour class. For each class, we analyse aggregated signals and derive representative patterns in terms of variance, temporal dynamics, and overall motion structure. These patterns are then expressed in natural language and used as the {features} input to guide response generation.

The behaviour-specific feature descriptions used in this work are as follows:

- active_climbing: shows sustained multidirectional movement, with noticeable variation on all three axes and relatively frequent changes over time.
- active_jumping.horizontal: exhibits strong short-term fluctuations across all axes, with acceleration centered near zero and clear burst-like motion.
- active_jumping.vertical: is characterized by large vertical excursions and rapid second-to-second changes, indicating repeated impulsive movement.
- active_playfight.fighting: displays irregular and active motion with substantial frame-to-frame changes, especially along the X and Z axes.
- active_playfight.playing: contains only a single aggregated sample, so it appears as a fixed acceleration state with no observable temporal variation.
- active_rubbing: shows moderate multidirectional variation with a wider spread on Z and relatively smooth second-to-second change.
- active_trotting: presents rhythmic movement with moderate spread on all axes and a stable pattern of repeated variation over time.
- active_walking: shows regular periodic motion with moderate dispersion and smooth temporal transitions in acceleration.
- inactive_lying.crouch: has a stable posture-like pattern with very small second-to-second changes and relatively fixed acceleration orientation.
- inactive_lying.down: shows low temporal variation with a consistent offset in acceleration, reflecting a mostly still posture.
- inactive_lying.resting: is highly stable, with tightly clustered values on all axes and minimal change over time.
- inactive_sitting.down: displays moderate spread in acceleration with noticeable but not abrupt temporal changes.
- inactive_sitting.stationary: maintains a very stable pattern over time, with clear orientation offsets but very small second-to-second variation.
- inactive_sitting.up: shows a relatively steady posture with moderate spread and gentle temporal fluctuations.
- inactive_standing.stationary: has a largely stable acceleration pattern with posture-dependent offsets and limited temporal movement.
- inactive_standing.up: shows marked short-term movement with substantial second-to-second change despite being labeled as a standing transition state.
- maintenance_grooming: exhibits moderate multidirectional spread with smooth temporal evolution and no strong single-axis dominance.
- maintenance_littering.defecating: shows a fairly steady pattern with limited temporal change and a consistent downward Z tendency.
- maintenance_littering.digging: displays repeated controlled movement with moderate spread and steady changes across all three axes.
- maintenance_littering.none: shows a compact acceleration pattern with mild fluctuations and a stable overall orientation.
- maintenance_littering.urinating: has a relatively stable pattern with moderate spread and modest temporal change, especially on Z.
- maintenance_nutrition.eating: shows a steady acceleration orientation with limited second-to-second variation and moderate axis spread.
- maintenance_scratching: exhibits active multidirectional movement with noticeable spread and frequent short-term changes.

- `maintenance_shake.body`: shows strong whole-body oscillatory motion with large spread and rapid temporal variation, especially on Z.
- `maintenance_shake.head`: displays sharp and frequent short-term changes, with pronounced oscillation and strong variability across axes.
- `other_social.allogrooming`: exhibits moderate controlled movement with a stable Y component and gentle temporal variation.

For samples that do not conform well to the characteristic patterns of their assigned class, or that correspond to transitional states where the behaviour is about to change, we apply separate matching strategies to generate appropriate responses.

F Reproducibility Details

We provide all information necessary to reproduce the main experimental results of Meow-Omni 1. The model architecture, training pipeline, and evaluation protocol are fully described in the main paper; here we consolidate the exact implementation details.

Hardware and Software Environment

All experiments were conducted on a single computing node equipped with **8× NVIDIA H200 80 GB GPUs** (NVLink interconnected).

Model Architecture

Meow-Omni 1 is based on the MiniCPM-o 4.5 backbone. We expand the tokenizer with three special tokens <|ts_start|>, <|ts_unit|>, and <|ts_end|> and integrate a time-series encoder adapted from Intern-S1 Pro. A linear projector maps the TS encoder outputs to the LLM's hidden dimension. During the forward pass, TS embeddings are interleaved with visual tokens and text tokens in a unified multimodal sequence. The TS encoder itself is frozen during all training stages.

Training Pipeline

Training proceeds in two stages.

Stage 1: Projector Alignment

- **Data:** 383,853 time-series samples (window lengths 5, 7, 10, 15 s) paired with behaviour labels.
- **Trainable parameters:** Only the TS projector; the LLM backbone and TS encoder are frozen.
- **Optimization:** AdamW, learning rate 10^{-4}, weight decay 0.0, batch size 1 per device, gradient accumulation steps 2.
- **Learning rate schedule:** Cosine annealing with linear warmup for 3% of total steps; maximum number of epochs 10.
- **Early stopping:** Validation accuracy monitored; patience 1 epoch. The best checkpoint (epoch 2) is saved as *Meow-Omni 1 Aligned*.
- **Loss:** Cross-entropy on the predicted next behaviour class.

Stage 2: Multimodal Specialization

- **Data:** 10,831 high-quality quad-modal samples from the Meow-10K dataset, including partial-modality combinations (V+A, V+TS, TS+A, V+A+TS).
- **Trainable parameters:** LLM backbone only; all encoders and projectors frozen.
- **Optimization:** AdamW as above, learning rate 2×10^{-5}, weight decay 0.0, batch size 1 per device, gradient accumulation steps 2.
- **Learning rate schedule:** Cosine decay with 3% warmup; maximum epochs 5.
- **Early stopping:** Validation accuracy used; patience 1 epoch. The final *Meow-Omni 1* checkpoint was taken at epoch 1.

- **Loss:** Next-token prediction loss over the full multimodal sequence, including special tokens.

Evaluation Procedure

All MeowBench results are obtained with a single deterministic greedy decoding pass (temperature $T = 0$; top-p disabled). The predicted future-behavior label is mapped back to the multiple-choice option (A/B/C/D) via a regex that extracts the first uppercase letter appearing after the model's answer prefix. In addition, because the backbone model has weak instruction-following ability, a response is also considered correct if it fully matches the correct answer and does not fully match any incorrect answer—i.e., if the model made the correct choice but included an explanation (even though it was instructed to output only the correct answer as a capital letter).

G Potential Negative Impacts

Meow-Omni 1 predicts feline future behavior from multimodal sensor data. The technology could be misused for continuous behavioral surveillance, and erroneous predictions could lead to inappropriate veterinary decisions if used without expert oversight. We therefore require human review for high-stakes use and do not treat the synthetic-conflict entropy result as a validated safety or deferral mechanism. Any future deployment should establish calibration, abstention criteria, privacy protections, and prospective clinical validation.

H Licenses of Existing Assets

All third-party datasets, pre-trained models, and code packages used in this work are publicly released under permissive licenses (CC-BY 4.0 or equivalent). The specific license for each asset is noted in its original publication and, where applicable, in the asset metadata. No copyrighted material has been used without permission.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction explicitly state the paper's contributions (Meow-Omni 1 architecture, MeowBench, tri-modal fusion) and scope (resolving semantic aliasing in feline intent recognition). These claims are directly supported by the experimental results (e.g., 71.16% accuracy), with no overstatement.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations are discussed in a dedicated "Limitations and Future Work" section, covering the Next-Behaviour Prediction proxy assumption, the synthetic nature of the conflict dataset, the model's passive operation, and the open question of cross-species generalisation.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: paper does not include theoretical results.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper fully discloses the model architecture, the complete training pipeline, and the exact training command with all hyperparameters in Appendix F. This provides sufficient detail to reproduce the main experimental results, independent of code release.significance tests.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The code and the dataset will be fully open-sourced after paper acceptance.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

Justification: The paper specifies all training and evaluation details in Appendix F, model stages, and pre-processing pipelines. These are sufficient to understand and interpret the reported results.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: The paper reports only point estimates (accuracy, macro-F1) without error bars, confidence intervals, or statistical significance tests.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: All experiments (training, evaluation, and ablation studies) were conducted on a single node with 8×NVIDIA H200 80 GB GPUs. This information is included in the appendix and provides the compute context required for reproduction.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research uses only publicly available datasets with proper citations, involves no human subjects, and does not raise ethical concerns that would violate the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: A dedicated in Section "Potential Negative Impacts" and "Discussion", both positive applications (veterinary diagnostics, conservation) and potential negative societal impacts (misuse for surveillance, misinterpretation risks) are discussesed, along with suggested mitigation strategies.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.

- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release any models or datasets that carry a high risk of misuse, so no safeguards are relevant.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All existing datasets and models used in the paper are properly cited in the references. Their licenses are explicitly stated to be CC-BY 4.0 in the appendix, and the terms of use are respected.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The Meow-10K dataset and the Meow-Omni 1 model are documented in the paper and appendices, including detailed preprocessing pipelines, model architecture, and training stages.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The work does not involve crowdsourcing or human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: No human subjects were involved, so IRB approval is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The core reasoning component of Meow-Omni 1 is a large language model (the MiniCPM-o backbone). Its architecture, integration with the TS encoder, and training are described in detail in Section "Methods". This usage is fundamental to the methodology, and the paper fully documents how the LLM is adapted for multimodal intent inference.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.