Short-form Video Classification

Midterm Presentation

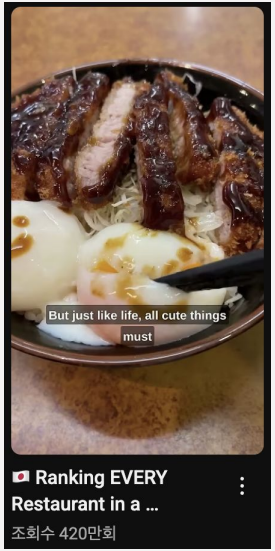
- Team 3 -

Task Definition

Cuisine-Type Classification for Short-form Video

Shorts

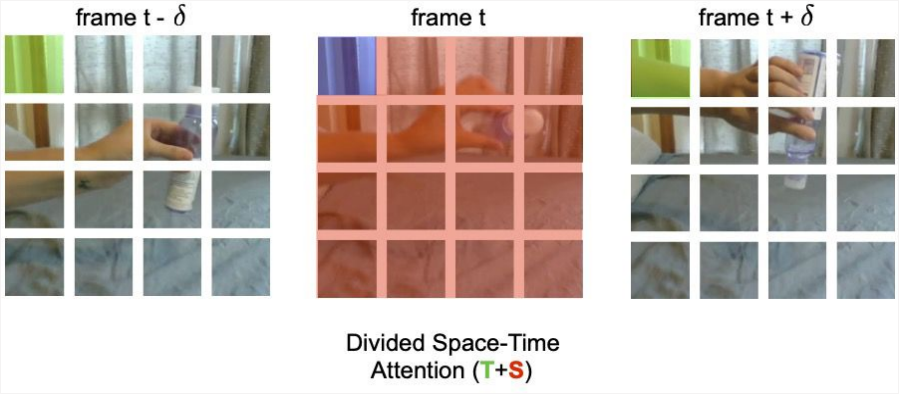
Korean?
Japanese?
Italian?
American?
…



Related Works (Video classification)

TimeSformer

- A convolution-free video classification model based entirely on self-attention
- Adopted divided space-time attention
- Achieved state-of-the-art accuracy on Kinetics-400 and-600
    - Human action classification



Model	Pretrain	K400 Training Time (hours)	K400 Acc.	Inference TFLOPs	Params
I3D 8x8 R50	ImageNet-1K	444	71.0	1.11	28.0M
I3D 8x8 R50	ImageNet-1K	1440	73.4	1.11	28.0M
SlowFast R50	ImageNet-1K	448	70.0	1.97	34.6M
SlowFast R50	ImageNet-1K	3840	75.6	1.97	34.6M
SlowFast R50	N/A	6336	76.4	1.97	34.6M
TimeSformer	ImageNet-1K	**416**	75.8	**0.59**	121.4M
TimeSformer	ImageNet-21K	**416**	**78.0**	**0.59**	121.4M

https://arxiv.org/abs/2102.05095

Related Works (Short-form Video classification)

TikGuard

- Multi label classification
  - Safe
  - Adult
  - Harmful
  - Suicide
- ~4K TikTok short-form videos
- Fine-tuned pretrained video classification models with ~6K steps

TABLE III
PERFORMANCE OF MODELS ON THE VALIDATION SET

Model	ACC	F1	Recall	Precision
TimesFormers	**0.8666**	**0.8662**	**0.8679**	**0.8662**
VideoMAE	0.7911	0.7917	0.7915	0.7911
VIVIT	0.8616	0.8624	0.8646	0.8624

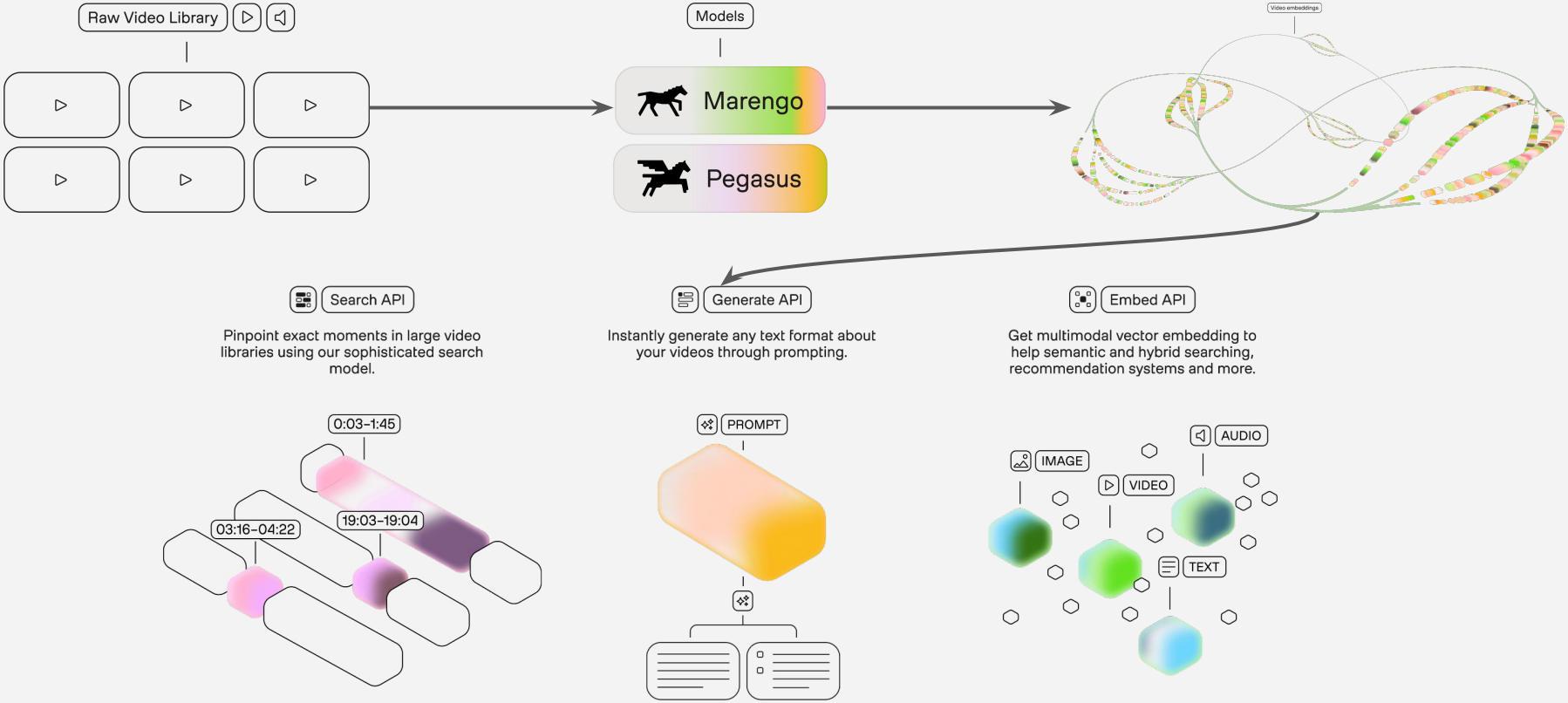
TABLE IV
PERFORMANCE OF MODELS ON THE TEST SET

Model	ACC	F1	Recall	Precision
TimesFormers	**0.8671**	**0.8668**	**0.8671**	**0.8669**
VideoMAE	0.7816	0.7802	0.7816	0.7826
VIVIT	0.8418	0.8408	0.8418	0.8467

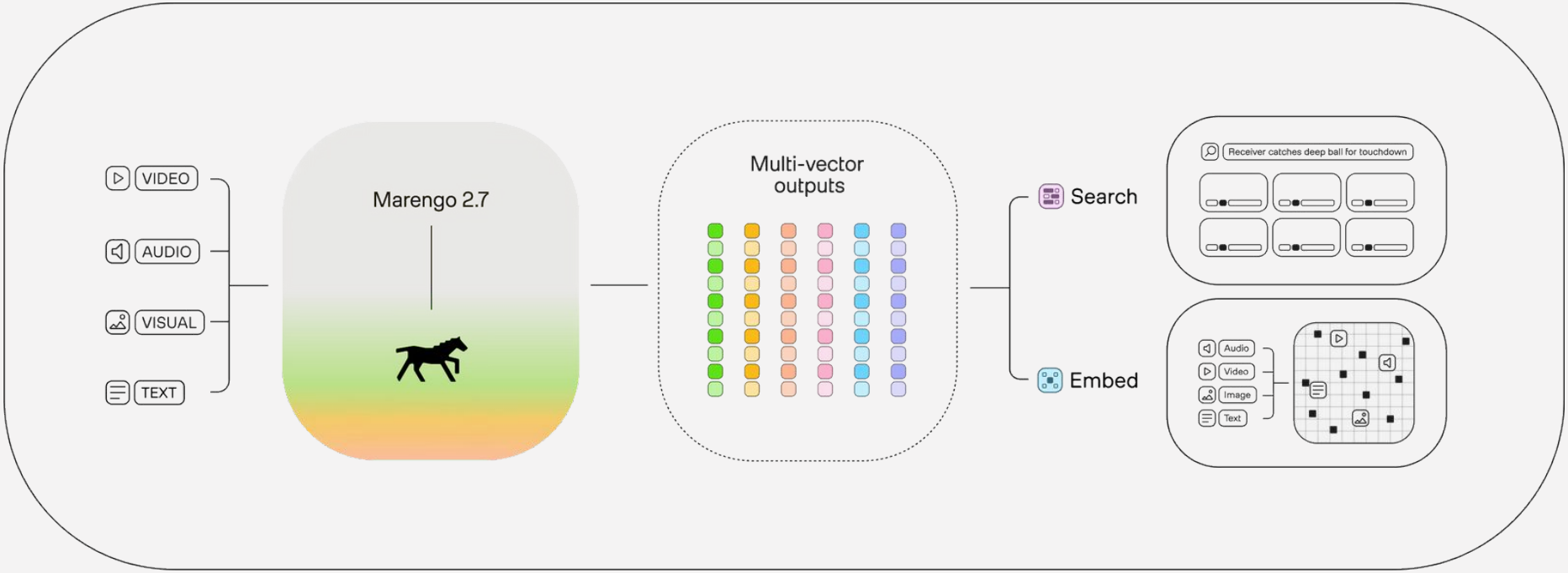
Fig. 1. Examples of each class in the TikHarm dataset.

https://arxiv.org/abs/2410.00403

Data preprocessing > TwelveLabs



Data preprocessing > TwelveLabs > Marengo 2.7



TwelveLabs > Marengo 2.7 > SOTA Performance

Model’s capabilities and performances evaluated with an extensive evaluation framework encompassing over 60 diverse datasets:

Generic visual understanding

Complex query comprehension

Small object detection

OCR interpretation

Logo recognition

Audio processing (verbal and non-verbal)

TwelveLabs > Marengo 2.7 > Quantitative Evaluation

Baseline models

Data Filtering Network-H/14-378

(Fang et al, Apple & University of Washington, 2023.09)

InternVideo2-1B

(Wang et al, OpenGVLab, 2024.08)

Google Vertex Multimodal Embedding API

(multimodalembedding@001, 2024.10)

Marengo 2.6

(Twelve Labs, 2024.03)

Evaluation Datasets

Text-to-Visual Datasets

- MSRVTT, COCO: Text-to-video/image
- Something-Something v2: Motion understanding
- TextCaps, BLIP3-OCR: OCR-focused search
- Object365-med, Mapillary-med, BDD-med: Small object retrieval (1–10% area coverage)

Text-to-Audio Datasets

- AudioCaps, Clotho: Generic audio retrieval
- GTZAN: Music genre classification via templates

Image-to-Visual Datasets

- Object365 (easy/med): Cropped-object to scene retrieval
- LaSOT: Image-to-video matching
- OpenLogo: Logo search (image-to-image)
- Ads-logo, Basketball-logo: Custom logo-in-video retrieval

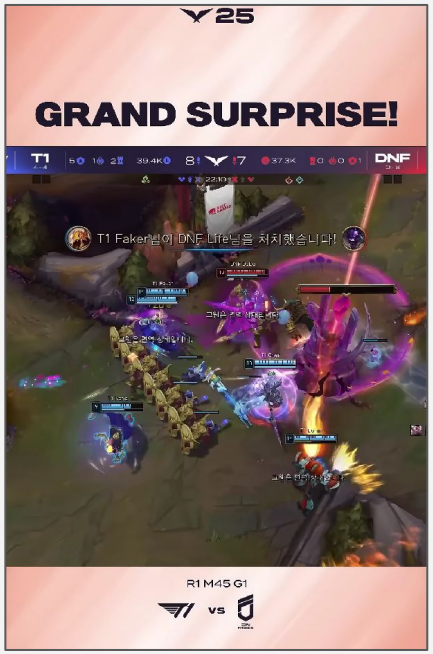
Marengo 2.7 > Quantitative Eval. > Benchmarks results

General Text to Visual Search AVG (R@1, R@5, R@10)	MSRVTT text-to-video	COCO text-to-image	Average
Apple CLIP (SOTA Image Retrieval Model)	60.0%	73.7%	66.8%
InternVideo2 (SOTA Video Retrieval Model)	65.9%	74.8%	70.3%
Google Vertex Multimodal Embedding API	59.9%	70.1%	65.0%
Marengo 2.6	66.9%	73.6%	70.2%
Marengo 2.7	72.2%	77.6%	74.9%
vs Marengo 2.6	+5.4%	+4.0%	+4.7%
vs External SOTA model	+6.4%	+2.8%	+4.6%

Text to Motion Search AVG (R@1, R@5, R@10)	SthSth-v2 text-to-video
Apple CLIP (SOTA Image Retrieval Model)	41.6%
InternVideo2 (SOTA Image Retrieval Model)	47.7%
Google Vertex Multimodal Embedding API	48.1%
Marengo 2.6	55.6%
Marengo 2.7	78.1%
vs Marengo 2.6	+22.5%
vs External SOTA model	+30.0%

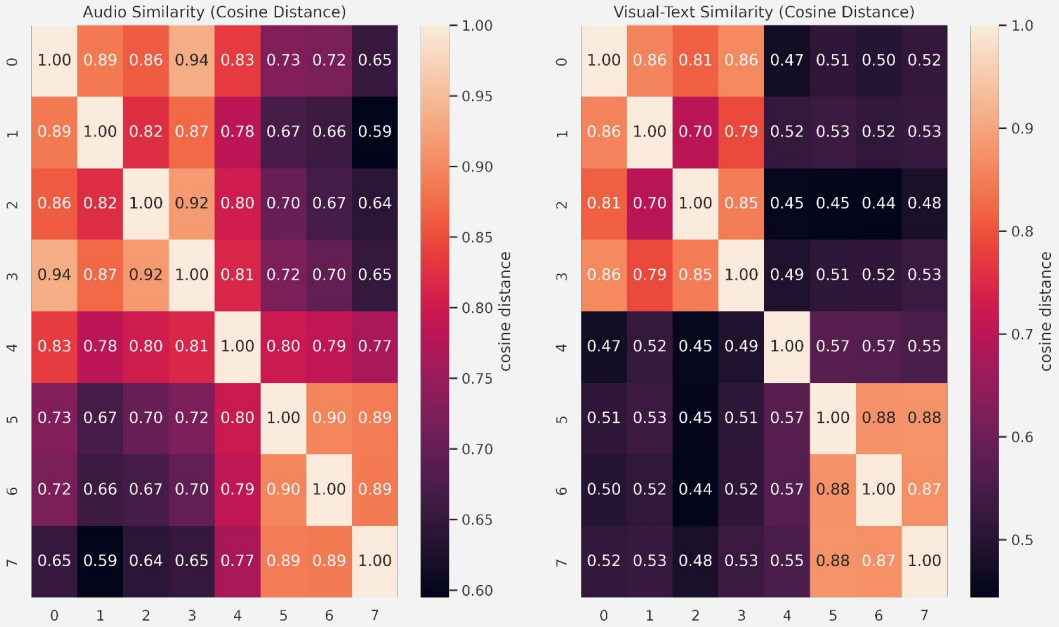
Marengo 2.7 > 1st simulation (concerts & eSports)

YouTube Shorts previews



Marengo 2.7 > 1st simulation (concerts & eSports)

Experimental results
(0-3 concerts; 4-7 eSports)

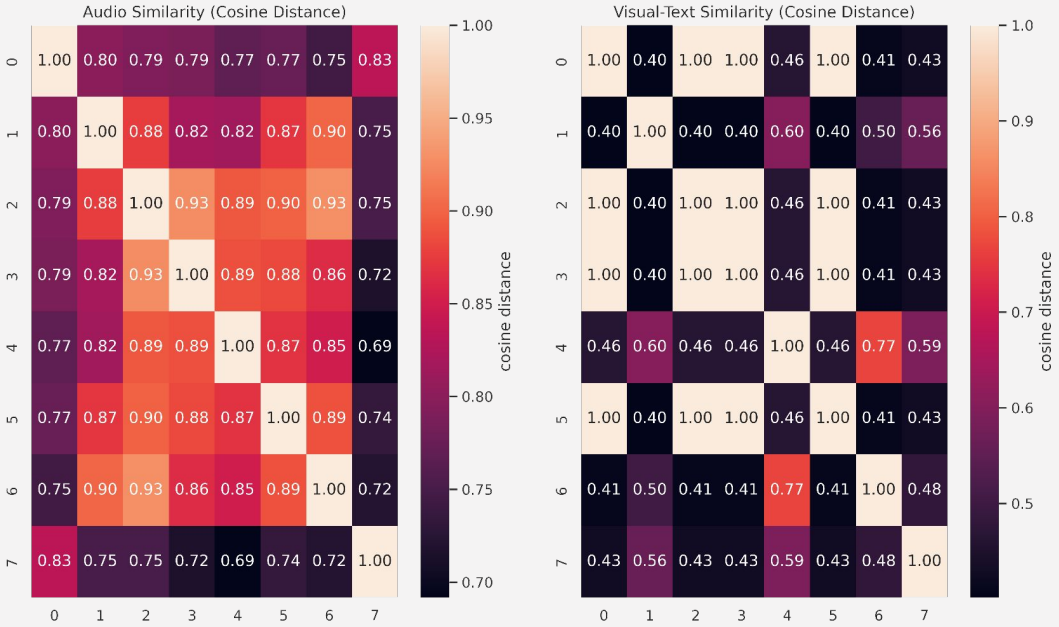


Marengo 2.7 > 2nd simulation (Korean & Italian cuisine)

YouTube Shorts previews

Marengo 2.7 > 2nd simulation (Korean & Italian cuisine)

Experimental results
(0-3 Korean; 4-7 Italian)



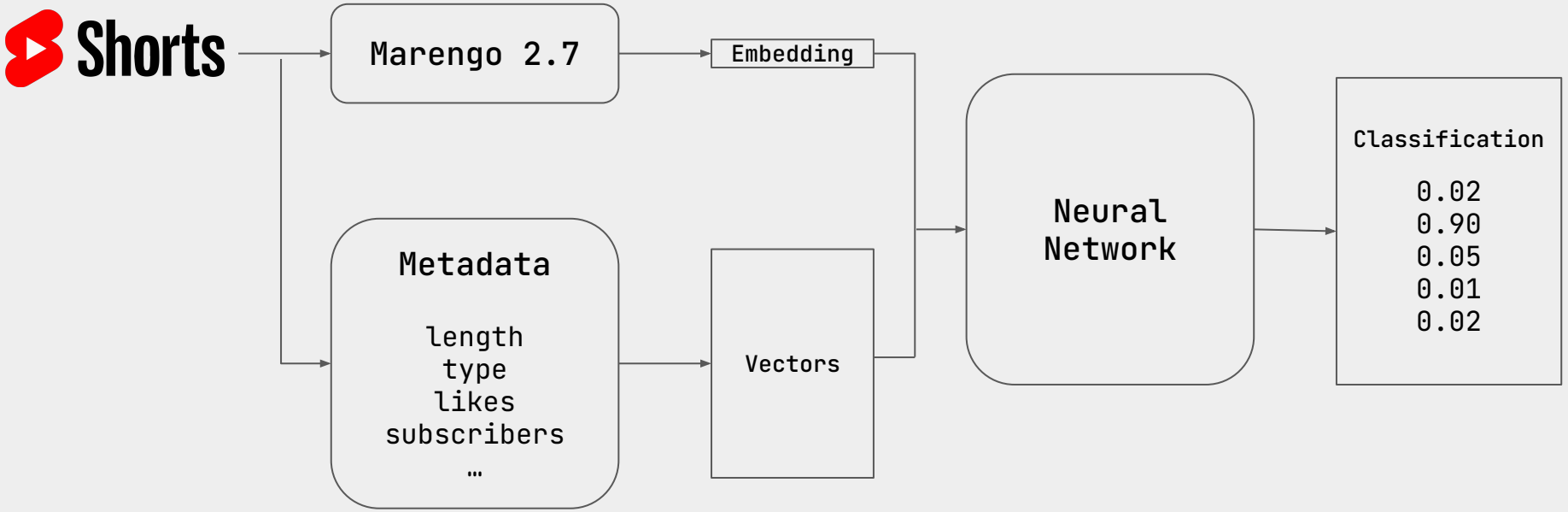
Baseline Method - Zero-shot Classification

- Methodology
  - Extract multimodal embeddings from each video using Twelve Labs Marengo 2.7
  - Apply zero-shot classification with text embeddings
    - Similarity metric: cosine distance, Euclidean distance
    - Each class represented by text embedding vector
- Limitations
  - No incorporation of auxiliary metadata (e.g., video title, length..)
  - Struggles with visually similar cuisines (e.g., ramen vs JJamppong)
  - Cannot model temporal or cross-modal interactions

Baseline Method - TikGuard Approach

- Methodology
  - Fine-tune pretrained video classification models
    - TimeSformer
    - VideoMAE
    - ViViT
- Limitations
  - Heavy computation cost
  - Large parameters to optimize

Proposed Method > Overview



Proposed Method > Details (1)

- **Task Definition**
  - Given a food-related YouTube Shorts video, predict its cuisine origin using video content and contextual signals
- **System Architecture (see diagram):**
  - Input
    - Video: YouTube Shorts containing food or cooking scenes
    - Metadata (optional): video length, view count, like ratio, uploader info..
  - Multimodal embedding extraction
    - Using Marengo 2.7, extracted embeddings

Proposed Method > Details (2)

- **Feature fusion**
  - Combine embedding vectors and metadata using:
    - Concatenation, or
    - Cross-modal attention layers for conditional modulation
- **Neural network backbone**
  - Current: Multi-layer Perceptron (MLP) with dropout regularizaiton
  - Future extension: lightweight Transformer encoder
- **Classification output**
  - Final layer: Softmax classifier over cuisine categories
  - Loss function: Cross-Entropy Loss, possibly class-balanced

Proposed Method > Next steps

1. Evaluate contribution of metadata via ablation study
2. Explore interpretability of each modality
3. Visualize semantic clusters using t-SNE on embeddings
4. Apply data augmentation methods

Github Repository Link

https://github.com/usingcolor/20251R0136COSE47400