On the Efficacy of Text-Based Input Modalities for Action Anticipation

Apoorva Beedu[1] , **Karan Samel**[1] , **Irfan Essa**[1,2]
[1]Georgia Institute of Technology
[2]Google Research
{abeedu3, ksamel, irfan}@gatech.edu

Abstract

Although the task of anticipating future actions is highly uncertain, information from additional modalities help to narrow down plausible action choices. Each modality provides different environmental context for the model to learn from. While previous multi-modal methods leverage information from modalities such as video and audio, we primarily explore how text inputs for actions and objects can also enable more accurate action anticipation. Therefore, we propose a Multi-modal Anticipative Transformer (MAT), an attention-based video transformer architecture that jointly learns from multi-modal features and text captions. We train our model in two-stages, where the model first learns to predict actions in the video clip by aligning with captions, and during the second stage, we fine-tune the model to predict future actions. Compared to existing methods, MAT has the advantage of learning additional environmental context from two kinds of text inputs: action descriptions during the pre-training stage, and the text inputs for detected objects and actions during modality feature fusion. Through extensive experiments, we evaluate the effectiveness of the pre-training stage, and show that our model outperforms previous methods on all datasets. In addition, we examine the impact of object and action information obtained via text and perform extensive ablations. We evaluate the performance on on three datasets: EpicKitchens-100, EpicKitchens-55 and EGTEA GAZE+; and show that text descriptions do indeed aid in more effective action anticipation.

1 Introduction

Suppose you go to a cafe and order a coffee and you see your barista steaming milk, can you predict what they might do next? Action anticipation is the task of predicting future actions, using visual cues and data from other modalities such as audio, sensor data, etc. from current and prior observations, as displayed in Figure 1. Predicting future actions is important for many Artificial Intelligence (AI) applications such as autonomous driving [Jain *et al.*, 2015; Rasouli *et al.*, 2020], assistive robotics [Petković *et al.*, 2019; Koppula and Saxena, 2015; Korbar *et al.*, 2018], augmented reality, etc. For instance, a kitchen robot can preemptively chop onions and hand them over if it can anticipate that the recipe requires chopped onions to be added next.

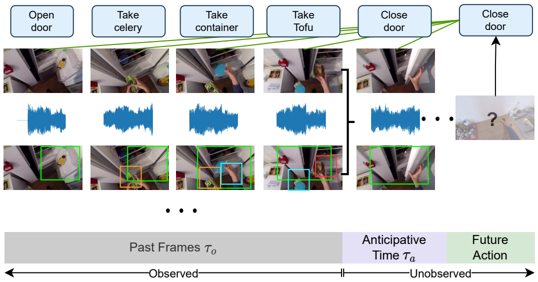


Figure 1: **Anticipating actions** τ_a seconds after observing information for τ_o seconds using multiple modalities.

This task, although seemingly straightforward for humans, is a challenging task for deep networks due to uncertainty of predicting the future, and large variability in the actions the models have to learn. Models not only have to detect the action happening at the observed time, but also fuse information from all available modalities to anticipate future actions. To formalize this task, challenging large-scale datasets such as Epic-Kitchens and Ego-4D [Damen *et al.*, 2018; Damen *et al.*, 2020; Grauman *et al.*, 2022; Stein and McKenna, 2013] have been developed, along with accompanying benchmarks, challenges, and leaderboards for state-of-the-art methods, for action anticipation, action recognition, action detection etc.

Typically, short- and long-term action anticipation involves extracting frame level features from videos and aggregating them using either temporal recurrent [Furnari and Farinella, 2019; Furnari and Farinella, 2020] and attention modules [Girdhar and Grauman, 2021; Zhong *et al.*, 2023], or directly extracting video level features using attention [Girdhar and Grauman, 2021; Wu *et al.*, 2022; Roy and Fernando, 2023; Roy and Fernando, 2021]. Although previous frames can be automatically 'attended' to, action anticipation using only videos–a single modality–still remains challenging,

Under review

and the availability of additional and complementary modalities is typically advantageous [Girdhar and Grauman, 2021; Zhong *et al.*, 2023]. For example, if an assistive robot is performing a task with camera pointed away from the person, and the person falls, the robot should still be able to react in a timely manner *if* audio is being used as an additional modality. Such scenarios demonstrate that information from multiple modalities can aid in many tasks, including action anticipation. Accordingly, recent works [Zhong *et al.*, 2023; Girdhar and Grauman, 2021; Thakur *et al.*, 2023] have shown that action anticipation greatly benefits from multi-modal training, as well as knowledge from other visual and audio cues such as active object detection, self-supervised future feature prediction, speech recognition, and hand-object contact information, typically by using modality specific encoders. Further, contrastive pre-training has also been employed for multi-modal setups, by aligning with text, e.g., CLIP [Radford *et al.*, 2021] and other foundation models [Girdhar *et al.*, 2022; Girdhar *et al.*, 2023; Wu *et al.*, 2022; Ni *et al.*, 2022]. In contrast, we examine the necessity for training such *modality-specific encoders*, and, instead aim to determine *if natural language descriptions are effective for action anticipation.* Therefore, we leverage language models to generate features by encoding object and actions in text, in lieu of relying on traditional feature extracting methods. Further, we also study which modalities are more beneficial and inspect how the accuracy of action recognition for the observed frames affects anticipation.

In this paper, we present 'Multi-Modal Anticipative Transformer (MAT)' that utilizes contrastive pre-training for generated captions from actions for videos. As Large Language Models (LLMs) are adept in giving long descriptions when prompted, we utilize this feature to generate long descriptions for actions that often involve additional knowledge on the environment and objects, e.g., kitchen vs living room, utensils used etc., allowing MAT to encode this knowledge aiding in action anticipation. We employ a two-stage network that first learns to recognize actions by contrasting fused features from multiple modalities against text, and subsequently predicting future actions. The first stage utilizes a CLIP-like framework which contrasts visual embeddings fused from multiple modalities against text embeddings, enabling our model to learn more descriptions for actions generated through GPT models. In the second stage, these fused features are then used to anticipate actions by training a classifier.

In summary, our contributions are:

- We propose a novel training protocol for predictive video modeling by contrasting modality features against action descriptions generated using LLMs.
- We propose and analyze using modalities in text format (actions and objects as text) for predictive modeling, and show that text based inputs generate strong features.
- We perform extensive analysis and ablations for different design choices, pre-training protocols and modalities used for our approach.
- We also conduct in-depth analysis on the affects of accurate action prediction for the observed frames.

2 Related Work

Action Anticipation is the task of predicting future actions after certain time units in a given video clip. Although the task has been explored extensively for third-person videos [Abu Farha *et al.*, 2018; Gao *et al.*, 2017; Huang and Kitani, 2014; Jain *et al.*, 2016].

The release of large-scale egocentric datasets and challenges like Epic-Kitchen [Damen *et al.*, 2018; Damen *et al.*, 2020], Ego-4D [Grauman *et al.*, 2022] have fast tracked the development for first-person scenarios as well. To model the temporal progression of past actions, [Furnari and Farinella, 2020] used a rolling-unrolling-based LSTM network to anticipate actions, such that rolling LSTMs account for the observed video frames, while unrolling LSTMs accounted for the anticipation. [Sener *et al.*, 2020; Sener *et al.*, 2021] made use of long-range past information by building a multi-scale temporal aggregating framework. In addition to gathering strong visual features, recent methods have used other visual cues like modeling the environment [Nagarajan *et al.*, 2020] or hand-object contact and activity modeling [Dessalene *et al.*, 2021]. More recently, the use of vision transformers [Dosovitskiy *et al.*, 2021] has also been explored. While, AVT [Girdhar and Grauman, 2021] proposes causal modeling of video frames, and using self-supervision to learn the future frame features, MeMViT [Wu *et al.*, 2022] perform multi-scale representation of frame features by hierarchically attending the previously cached "memories". AFFT [Zhong *et al.*, 2023] proposes a fusion method to effective fuse features from multiple modalities and extend AVT for action anticipation. [Roy and Fernando, 2023], AntGPT [Zhao *et al.*, 2023] and leverages the goal information to reduce the uncertainty in future predictions. AntGPT [Zhao *et al.*, 2023] trains Large Language Models (LLM) to infer goals and model temporal dynamics. In contrast, we use pretrained LLMs to generate additional contextual cues about the actions, and create additional text based modalities from objects and actions.

Language Image Pre-training Training images jointly with natural language text (e.g., captions) has been established as an effective pre-training method for zero-shot learning, open vocabulary testing, and as well as classification tasks. CLIP [Radford *et al.*, 2021], ALIGN [Jia *et al.*, 2021], FLorence [Yuan *et al.*, 2021], X-CLIP [Ma *et al.*, 2022] have shown that training on large-scale image-text pairs using contrastive learning exhibits impressive performance for zero-shot prediction. OWL-ViT [Minderer *et al.*,] uses a CLIP-based contrastive approach to transfer image-level pre-training to open vocabulary object detection. Similarly, CoCa [Yu *et al.*, 2022] is not only trained on the contrastive loss, but also leverages generative modeling via the captioning loss. Flamingo [Alayrac *et al.*, 2022] on the other hand interleaves visual data with text and produces free-form text as output, demonstrating effective performance on several downstream tasks. Such natural language supervision also aids in video representation learning. For instance, [Bertasius and Torresani, 2020] used a visual detector to map every object instance in the video frame into its contextualized word representation obtained from narration. Building on these works, we propose a CLIP-like architecture that learns from

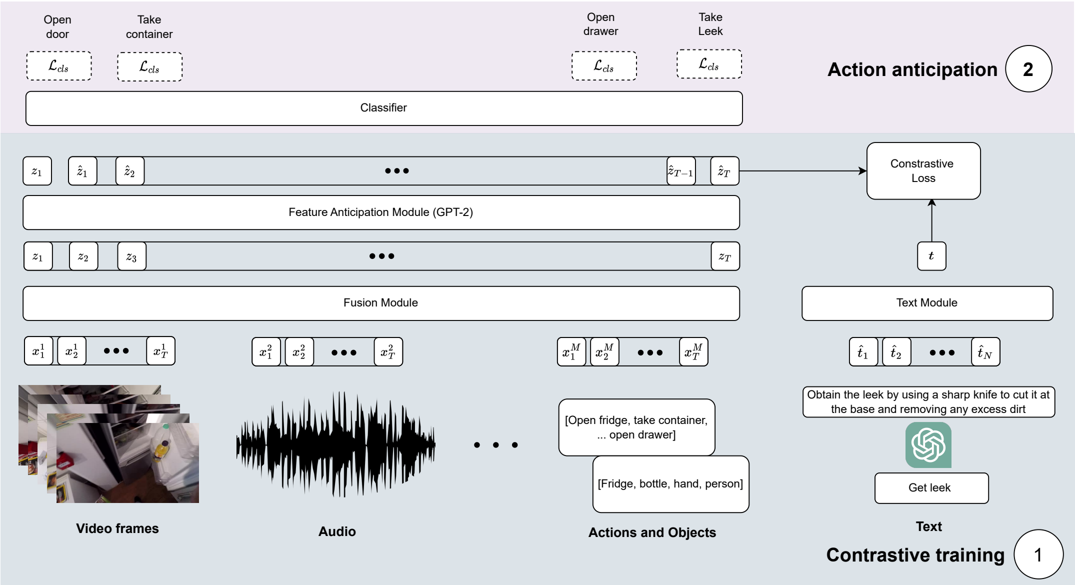


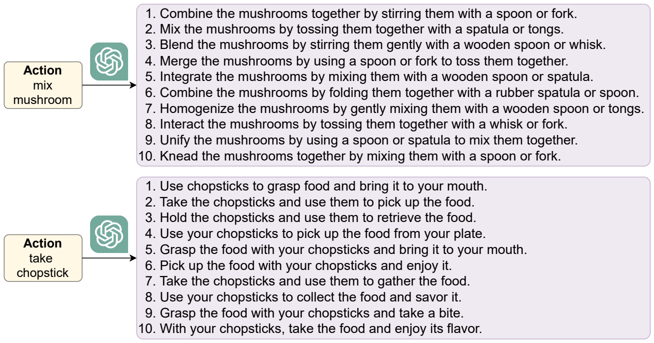
Figure 2: Overview of our architecture: we split the training into two stages. In the first stage, contrastive training, we fuse embeddings from different modalities using a Fusion Module $\mathcal{F}$, followed by an anticipation module $\mathcal{B}$. The output is contrasted against the action descriptions generated from ChatGPT. The second stage involves action anticipation, i.e., utilizing a linear layer to predict actions.

generated captions. We also investigate the effectiveness of using descriptions and simple prompts during training.

Multi-modal training Typically, modalities used for action anticipation include RGB images, optical flow, object information, IMU, and audio [Girdhar and Grauman, 2021; Furnari and Farinella, 2019; Sener *et al.*, 2020; Wu *et al.*, 2021; Zatsarynna *et al.*, 2021; Zhong *et al.*, 2023]. Features from each modality are simply averaged, either weighted [Girdhar and Grauman, 2021] or unweighted [Furnari and Farinella, 2019], or an Multi-Layer Perceptron (MLP) is used [Kazakos *et al.*, 2019]. Recently, multi-head cross attention is being employed to attend over different modalities [Zhong *et al.*, 2023; Liu *et al.*, 2021]. However, training modality specific encoders can be computationally expensive. Instead, we explore the usage of text based inputs as modalities ie objects and actions detected in text form in lieu of visual features. To this end, we propose an architecture which contrasts fused features from different modalities including text from actions and objects detected in the video, with descriptions generated from action labels.

3 Methodology

Our model architecture (illustrated in Figure 2) comprises two stages: pre-training and fine-tuning. During pre-training, the model consists of M modality specific feature extractors $\mathcal{B}_m, m \in \{1, \ldots, M\}$, and a fusion model $\mathcal{F}$. The fine-tuning stage has an additional classifier that predicts the action, and the model is trained end-to-end. We follow [Zhong *et al.*, 2023] for the fusion module and a variation of the GPT2 model used in [Girdhar and Grauman, 2021] for feature anticipation to predict $\hat{z}_{i+1} = \mathcal{D}(z_i), i \in \{1, \ldots, T\}$. In what follows, we detail the two stages, along with the implementation details.



1. Combine the mushrooms together by stirring them with a spoon or fork.
2. Mix the mushrooms by tossing them together with a spatula or tongs.
3. Blend the mushrooms by stirring them gently with a wooden spoon or whisk.
4. Merge the mushrooms by using a spoon or fork to toss them together.
5. Integrate the mushrooms by mixing them with a wooden spoon or spatula.
6. Combine the mushrooms by folding them together with a rubber spatula or spoon.
7. Homogenize the mushrooms by gently mixing them with a wooden spoon or tongs.
8. Interact the mushrooms by tossing them together with a whisk or fork.
9. Unify the mushrooms by using a spoon or spatula to mix them together.
10. Knead the mushrooms together by mixing them with a spoon or fork.

1. Use chopsticks to grasp food and bring it to your mouth.
2. Take the chopsticks and use them to pick up the food.
3. Hold the chopsticks and use them to retrieve the food.
4. Use your chopsticks to pick up the food from your plate.
5. Grasp the food with your chopsticks and bring it to your mouth.
6. Pick up the food with your chopsticks and enjoy it.
7. Take the chopsticks and use them to gather the food.
8. Use your chopsticks to collect the food and savor it.
9. Grasp the food with your chopsticks and take a bite.
10. With your chopsticks, take the food and enjoy its flavor.

Figure 3: Descriptions generated using ChatGPT API for actions in the EPIC-Kitchen dataset. The descriptions generated add more contextual cues for the model to learn from. For instance, for the action *take chopsticks*, the description is already alluding to the future action of "picking up food" or "eating" taking place. During training, for every action, we randomly select one description.

3.1 Action Anticipation

As illustrated in Figure 1, for each action segment starting at τ_s, the goal is to recognize the action using τ_o length of video segment τ_a units before it, *i.e.* from $\tau_s - (\tau_a + \tau_o)$ to $\tau_s - \tau_a$.

The anticipation time τ_a is usually fixed for every dataset, while the observation time τ_o can be varied. We extract T temporally sequential inputs for M modalities and denote it as x_i^m, $i \in \{1, \ldots, T\}$ and $m \in \{1, \ldots, M\}$. For all our training protocols, all modality feature are extracted from pretrained models.

3.2 Pre-training Network

We employ a CLIP-like [Radford *et al.*, 2021] architecture, where the embeddings from different modalities (e.g., images and audio) are contrasted against text embeddings, which comprise augmented action classes (detailed below). In what follows, we detail the encoders for the modalities, and the losses used for training.

Visual Representation: Given a video segment *V* consisting of T frames, the backbone network B extracts features for each frame. Following [Zhong *et al.*, 2023], we use the Swin transformer features extracted with Omnivore [Girdhar *et al.*, 2022], that was trained for action recognition.

Text Representation: In our framework, we use several forms of text data. For action classes against which other modality features are contrasted, we use a pre-trained CLIP-based text encoder and finetune it on our datasets. However, to diversify the text inputs, we use GPT3.5 [Brown *et al.*, 2020] from the OpenAI API to convert the classes into sentences using the prompt "Describe <xyz> action in 1 sentence in 10 different ways", and randomly select one response during training. We provide examples in Figure 3 and a discussion in Section 4.1. When additional modalities like objects and actions are used, we use pre-trained CLIP based Text Encoder [Radford *et al.*, 2021].

For objects detected from a pre-trained FasterRCNN model [Furnari and Farinella, 2020], we obtain text features by using the phrase "A video containing the following objects: <list of objects>", with the aforementioned text encoder. Similarly for actions, the prompt used was "A video containing the following actions: <list of action>". As some of the datasets do not have dense action annotations, when action is not available, we use the "no action" tag. During both training and testing, we use ground-truth action labels. However, we analyze the impact of the action recognition accuracy on action anticipation performance in Section 4.3.

Cross Modal Fusion: In order to fuse information from multiple modalities x_i^m, we follow [Zhong *et al.*, 2023] and use self-attention fuser (SA-Fuser) blocks. It applies L consecutive Transformer encoders at each time step with dimentionality of d and k attention heads and contains a learnable token x^{Λ}. The final output is the mean of all learnable tokens.

Pre-training: The Fused embeddings are passed through a variation of the GPT-2 [Radford *et al.*, 2019] module to predict the future features

$$\hat{\mathbf{z}}_1, \dots \hat{\mathbf{z}}_T = \mathcal{D}(\mathbf{z}_1, \dots, \mathbf{z}_T) \tag{1}$$

where $\hat{\mathbf{z}}_t$ is the predicted feature corresponding to the frame $\mathbf{z}_t$ after attending to all the frames $\mathbf{z}_1, \dots, \mathbf{z}_{t-1}$. We refer the reader to [Girdhar and Grauman, 2021] for more details.

Features at z_T, which have attended over all the frames, are then trained to align with the text embeddings via contrastive learning.

For a batch size B with C classes, the loss is defined as:

$$\begin{aligned} \mathcal{L}_{v2t} &= \frac{1}{B} \sum_{i=1}^{B} log \frac{exp(s(v_i, t_i))}{\sum_{j=1}^{B} exp(s(v_i, t_j))} \\ \mathcal{L}_{t2v} &= \frac{1}{B} \sum_{i=1}^{B} log \frac{exp(s(v_i, t_i))}{\sum_{j=1}^{B} exp(s(v_j, t_i))} \\ \mathcal{L}_{cross} &= \mathcal{L}_{v2t} + \mathcal{L}_{t2v} \end{aligned} \tag{2}$$

Following AVT [Girdhar and Grauman, 2021], we also utilize a self-supervised feature loss $\mathcal{L}_{feat}$ in addition to the contrastive loss. Therefore, our final loss function is $\mathcal{L} = \mathcal{L}_{cross} + \mathcal{L}_{feat}$, where $\mathcal{L}_{feat}$ is defined as mean squared error between $\hat{\mathbf{z}}_t$ and $\mathbf{z}_{t+1}$, which matches the future features predicted with the true features in a self-supervised manner.

Dataset	τ_a	Modalities	Metrics
EGTEA+	0.5s	RGB, Flow	Top-1, cm Top-1
Ek55	1.0s	RGB, Obj, Flow, Audio, Objects (text), Actions(text)	Top-1, Top-5
EK100	1.0s	RGB, Obj, Flow, Audio, Objects (text), Actions(text)	Recall@5

Table 1: Modalities and metrics used for different datasets.

3.3 Fine-tuning Network

During the fine-tuning stage, we perform training for the action anticipation task. We use the features obtained from the feature anticipation module, $\hat{\mathbf{z}_T}$, in conjunction with a linear layer, and train with the cross entropy loss $\mathcal{L}_{cls}$.

3.4 Implementation details

We process the input videos similar to [Girdhar and Grauman, 2021], and sample 16 frames at 1 Fps for most experiments, by setting $\tau_o = 16s$. We pre-process the video by randomly scaling between 248 and 280px, and take crops with 224px at training time.

We use the Swin Transformer from Omnivore [Girdhar *et al.*, 2022] for the video features, and use GPT-2 for temporal learning. We use pre-trained CLIP based text encoder, processor, and tokenizer, provided by [Wolf *et al.*, 2020]. During pre-training, the encoded features are projected to 1024 dimensions. In the fine-tuning stage, the fused features are passed through a linear layer. We train our model with SGD+momentum using learning rate $= 1e^{-3}$ and weight decay $= 1e^{-6}$ for 50 epochs, for both pre-training and fine-tuning. Further, we use a cosine annealing learning rate schedule with a warmup for 20 epochs, and the training is performed on Nvidia A40 GPUs. We use a mix of augmentations such as jittering, brightness, saturation, contrast and hue and random flipping during training.

For the optical flow and object features, we use the official RULSTM [Furnari and Farinella, 2020] repository, and for audio, we use features provided by [Zhong *et al.*, 2023]. Following [Kazakos *et al.*, 2019], 1.28s of audio was extracted, converted to single channel, and resampled at 24kHz, for obtaining the log-spectrogram representations (STFT of window length 10ms, 256 band frequency). This was then fed to TSN [Wang *et al.*, 2016] and trained for action recognition.

4 Experiments

To demonstrate the effectiveness of our pre-training protocol, we empirically evaluate our method on benchmarks covering both first and third person views. We detail the datasets and metrics used below, followed by our experimental results and discussions.

4.1 Experimental setup

Datasets and metrics: We evaluate on three popular action anticipation datasets: *(i) Epic-Kitchens 100 (EK100)* [Damen *et al.*, 2020], which is a large egocentric video dataset with 700 long unscripted videos of cooking activities totaling 100

Model	Top-1			Class mean acc		
	Verb	Noun	Act.	Verb	Noun	Act.
I3D-Res50 [Carreira and Zisserman, 2017]	48.0	42.1	34.8	31.3	30.0	23.2
FHOI [Liu *et al.*, 2020]	49.0	45.5	36.6	32.5	32.7	25.3
AVT(TSN) [Girdhar and Grauman, 2021]	51.7	50.3	39.8	41.2	41.4	28.3
AFFT* [Zhong *et al.*, 2023]	52.1	50.7	41.4	38.4	43.7	31.8
Ours (w/ flow)	**53.0**	**50.8**	**41.7**	**42.2**	**45.0**	**32.9**

Table 2: **EGTEA Gaze+** Model performance for Split=1 at $\tau_a = 0.5s$. * indicates that the model was retrained. **Bolded** values indicate highest score.

hours. The dataset consists of 90K segments, and has 3807 action classes, 97 verbs and 300 nouns. We report the class-mean Recall@5 for actions, verbs and nouns; *(ii) EpicKitchens 55 (EK55)* [Damen *et al.*, 2018] is an earlier version of Epic-Kitchens 100. For comparison to existing approaches, we report the test accuracy on this dataset as well. EK55 has about 39K segments, and 2513 action classes, 124 verbs and 351 noun classes. For EK55, we report Top-1 and Top-5 for actions, verbs and nouns. We use the standard train and val splits to report performance. *(iii) EGTEA Gaze+* [Li *et al.*, 2018], an egocentric dataset containing about 10K segments, and 19 verbs, 51 nouns and 106 unique actions. Following [Girdhar and Grauman, 2021], we report the performance on the first split of the dataset at τ_a = 0.5s. We report the Top-1 and class-mean(cm) Top-1 accuracies for actions, nouns and verb.

Modalities: We summarize the different modalities used in Table 1. We use pre-trained TSN features provided by the official repositories [Furnari and Farinella, 2020; Zhong *et al.*, 2023] for object features, audio, and flow. For objects, we use the FasterRCNN model trained on Epic-Kitchen 55 dataset [Furnari and Farinella, 2020], and use a threshold of 0.15 to pick the top 5 objects for every image. In Section 4.3, we further evaluate if the objects in text form are more beneficial to the learning than object features extracted from fasterrcnn models. For actions, we use the ground truth labels provided by the dataset during training and evaluation. We evaluate the impact of action recognition accuracy and discuss the results in Section 4.3.

Baselines: In addition to comparing our method to its variants containing different modalities, we also evaluate against the state-of-the-art for action anticipation, including, RULSTM [Furnari and Farinella, 2020], AVT [Girdhar and Grauman, 2021], ActionBanks [Sener *et al.*, 2020], AFFT [Zhong *et al.*, 2023], and MeMViT [Wu *et al.*, 2022]. RULSTM [Furnari and Farinella, 2020] leverages a 'rolling' LSTM to encoder the past and an 'unrolling' LSTM to predict the future. ActionBanks [Sener *et al.*, 2020] improves over RULSTM by carefully leveraging long-term action blocks and non-local blocks. AVT [Girdhar and Grauman, 2021] uses an attention-based video modelling architecture that attends to previous frames to anticipate the future. MeMViT [Wu *et al.*, 2022], on the other hand, processes videos online by using cache "memory", through which the model learns to refer prior context for long-term anticipation. AFFT [Zhong *et al.*, 2023] improves on AVT by using multiple modalities, and using self-attention modules to fuse the features together. For fair comparison, as we re-train the AFFT model on our local environment setup, there is a small discrepancy in performance relative to the published paper. As the goal of this paper is to demonstrate the effectiveness of learning from text embeddings, we do not compare against other state-of-the-art methods that have a completely different architecture like [Roy and Fernando, 2021; Roy and Fernando, 2023].

ChatGPT generated action descriptions: We provide examples of the descriptions generated using the ChatGPT API (with GPT3.5 Turbo) on the action classes in Figure 3 for EpicKitchens datasets. We see that in the descriptions, there are generally mentions of other objects that are used when the source action takes place. For the action "take chopsticks", the descriptions provided by the ChatGPT API are very informative, as most often, chopsticks are used to grab or gather food. Similarly, for the "mix mushroom" action, most often there is an involvement of a tong, spoon or a spatula. However, during the generation, there were classes like "take finger:lady", which the API wasn't able to recognize as the vegetable "lady finger", in which case we manually generate the descriptions by altering the action. Similarly, as action classes were generated for all verb noun pairs, the API was unable to generate descriptions for actions like "take TV", "consume garbage", and "stab hand" due to ethical reasons. In such cases, we added descriptions like "Do not eat garbage" and "Do not stab hand" to avoid the model from learning it as a class. We believe further care needs to be taken to clean out such descriptions.

4.2 Comparison against baselines

EGTEA+ In Table 2, we compare our results on split 1 (as in [Liu *et al.*, 2020]) at $\tau_a = 0.5s$. In addition to the RGB data, we use the flow data provided by [Furnari and Farinella, 2020]. Similar to AFFT, we use the pre-trained TSN features for RGB inputs. For EGTEA+ dataset, in order to evaluate the effectiveness of contrastive pre-training, we do not generate action descriptions using ChatGPT, instead use a simple prompt - "This is a video with $< xyz >$ action". Although our model was trained with this simple prompt, we see an improvement of 4% in class mean accuracy for verbs, and over 1% for nouns and actions, outperforming all baselines. We also note that the results for AFFT were obtained by using the official code on our local environment.

Epic-Kitchen In Table 3 and Table 4, we compare the performance of our method to state-of-the-art methods for EK55 and EK100 datasets. For EK55, we generate the results for the AFFT model using the authors' code. We first compare the performance when all layers save the classifier are kept frozen during the second stage of training (rows highlighted in green) against baselines. By using GPT generated descriptions during pre-training, we get an improved performance of 1% on Top-5 and 0.6% on Top-1 metrics for actions and nouns. When all layers are fine-tuned, we outperform all methods for noun and action classification in Top-1 metrics, and perform comparably on Top-5 for actions. Therefore, we observe that when GPT generated captions are used during pre-training, the model is able to learn stronger features about actions and objects (nouns) in the scene as the generated descriptions often contain more details about what objects are

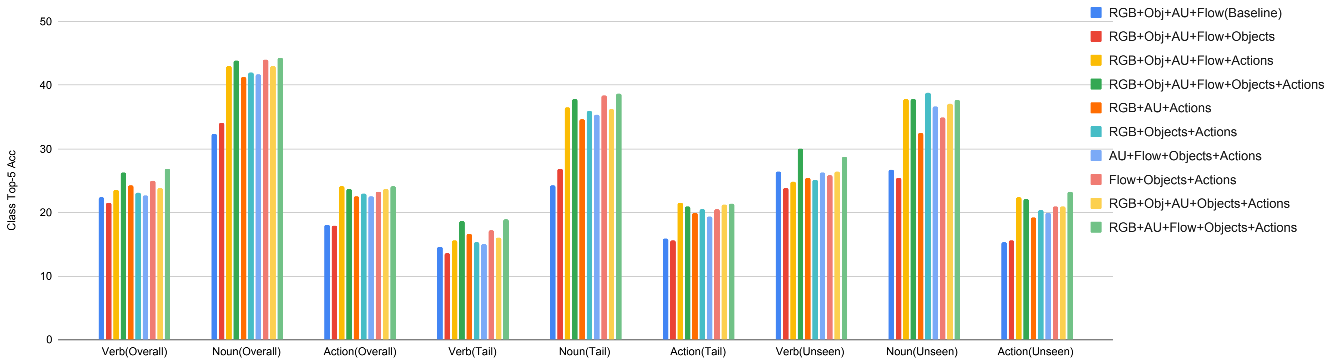


Figure 4: Recall@5 for verb, noun and actions on the EK100 dataset for different modality combinations. The first bar is the baseline (i.e., AFFT) using (RGB+Obj+Flow+Audio) modalities. Objects and actions are used as input by converting them to text through – "A video containing the objects/actions <xyz>", and embeddings from the text encoder are used in the fusion module.

Method	Verb		Noun		Action	
	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
RULSTM	32.4	79.6	23.5	51.8	15.3	35.3
ActionBanks	**35.8**	80.0	23.4	52.8	15.1	35.6
AVT+	32.5	79.9	24.4	54	16.6	37.6
AFFT	34.9	78.7	**26.2**	53.9	17.0	34.3
Ours (w/o gpt)	32.8	78.7	23.7	51.2	14.3	31.4
Ours (w gpt)	32.8	78.9	23.1	52.3	14.9	32.6
Ours	35.1	**80.7**	25.7	**55.3**	16	36.5
Ours*					**19**	**38.6**

Table 3: **EK55** val sets. Comparison of state-of-the-art method on the validation set of Ek55 using all the modalities. * indicates that additional action and objects modalities in the text form were used. w/o gpt indicates that the model was not pre-trained using GPT descriptions. Rows in green have fozen layers, and only the final classifier layer is trained during the second stage of the training, while the rest of the methods are fine-tuned.

Method	Overall			Unseen			Tail		
	Verb	Noun	Action	Verb	Noun	Action	Verb	Noun	Action
RULSTM	27.8	30.8	14.0	28.8	27.2	14.2	19.8	22.0	11.1
TempAgg	23.2	31.4	14.7	28	26.2	14.5	14.5	22.5	11.8
AVT+	28.2	32.0	15.9	29.5	23.9	11.9	21.2	25.8	14.1
MeMViT	**32.3**	37.0	17.7	28.6	27.4	15.2	**25.3**	31.0	15.5
AFFT	22.4	32.4	18.1	26.5	26.8	15.3	14.6	24.3	15.9
Ours	21.9	32	18.4	25.8	26.2	16.2	14.0	24.0	16.0
Ours*	26.3	**43.9**	**23.7**	**30.0**	**37.9**	**22.1**	18.6	**37.9**	**21.0**

Table 4: **EK100** val set performance: comparison of state-of-the-art method on the validation set of EK100 using all the modalities. MeMViT uses only RGB data, while the rest use multiple modalities. * indicates that additional action and object modalities in the text form were used.

used for the said action to take place, where it occurs etc. When we fine-tune the entire network, we observe clear performance improvements for the Top-5 metric for all classes compared to AFFT, whereas the performance is comparable to other baselines. It is worth noting that, similar to AFFT, our model was trained on pre-extracted features.

For EK100, we compare our two-stage network against single-stage methods, as well as using action and object information via text in Table 4. We seen an absolute 1% improvement for the tail classes for actions, and 0.3% overall. In addition, we also see a substantial improvement of 5% for actions when text inputs are used. MeMVit, trained only on RGB data, but trained for long term modeling consistently performs better than our method, AVT+ and AFFT for verb classification. This can be expected as our model is designed similar to AVT (backbone and Future prediction) and AFFT (fusion model). We also hypothesize that training MeMViT with these additional modalities will only help improve the performance.

4.3 Ablations and Analysis

Impact of different modalities: In Figure 4, we explore the contributions of various modalities to performance. For all the experiments, we use the objects provided by [Furnari and Farinella, 2020], and ground truth labels for actions. We see that when objects detected by FasterRCNN are used in addition to the baseline modalities, we see an improvement in overall noun predictions, and action prediction for unseen data. However, we see a significant improvement when the action labels are used. Using actions as one of the 'modalities', we ablate over different combinations of the remaining modalities, and see that the performance improvement remains consistent. Particularly, when only flow and text for actions and objects are used, we still see an overall improvement of 5% for action prediction. Adding RGB features only improves the accuracy by a small percentage, indicating that when text inputs are present, the model gravitates towards learning from text more than the other modality features. We also observe that audio is another modality that improves action anticipation. However, adding the FasterRcnn objects does not improve performance by significant margins.

To understand the impact of using object information as an additional modality, we examine the detected objects and the actions in Table 5. We see that for rows 1 and 3, the object required for the action prediction is not detected by the FasterRcnn model with high probability. For rows 2 and 4, while the object was detected, presence of other objects make the action prediction challenging. On the other hand, actions (which are often defined as a verb-noun pair) give more information about the objects being interacted and the actions in the observed frames. Therefore, while detecting objects accurately is essential and makes one part of the action (<verb,noun>), it is also vital that an active hand-object

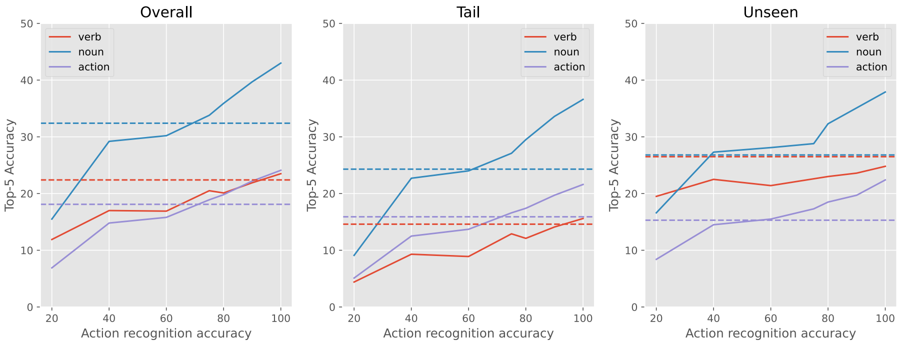


Figure 5: Impact of action recognition accuracy on the prediction of verbs, nouns and actions for EK100. Values in dashed lines are the corresponding results from the AFFT baseline.

	FasterRCNN objects	Actions	Future Action
1	‘sponge, tap’, ‘sponge, tap’,‘sponge, tap’,‘sponge, tap’, ‘sponge, tap’, ‘sponge, tap’,‘sponge, tap’, ‘sponge, tap’,‘sponge, tap’,‘sponge, tap’	‘wash plate’, ‘wash plate’, ‘no action’, ‘wash plate’, ‘wash plate’, ‘wash plate’, ‘wash plate’, ‘wash plate’, ‘insert plate’, ‘insert plate’, ‘wash sponge’,‘wash sponge’, ‘wash sponge’, ‘wash sponge’,‘wash sponge’, ‘wash sponge’	Wash cloth
2	‘bin, spoon’, ‘bin’, ‘knife, ’, ‘knife, ’, ‘bin’, ‘bin’, ‘bag’, ‘bag’, ‘bag’, ‘bin, bag’, ‘bin, bag’,‘bin, bag’, ‘bin, bag’, ‘bin, bag’, ‘bin, bag’, ‘bin, bag’	‘wrap bag’, ‘wrap bag’, ‘wrap bag’, ‘wrap bag’,‘wrap bag’, ‘wrap bag’, ‘wrap bag’, ‘wrap bag’,‘wrap bag’, ‘wrap bag’, ‘wrap bag’, ‘wrap bag’,‘wrap bag’, ‘wrap bag’, ‘wrap bag’	Tie Bag
3	‘cupboard’, ‘cupboard’, ‘cupboard’, ‘cupboard’, ‘cupboard’, ‘pan,cupboard’,‘pan,cupboard’, ‘cupboard’, ‘cupboard’, ‘cupboard, lid’, ‘pan,cupboard’, ‘pan,cupboard’,‘pan,cupboard’, ‘pan,cupboard’, ‘pan,cupboard’, ‘pan, ’	‘take plate’, ‘take plate’, ‘take plate’, ‘take plate’, ‘take plate’, ‘no action’, ‘open cupboard’, ‘no action’, ‘insert plate’, ‘insert plate’, ‘no action’, ‘no action’, ‘take cup’,‘no action’, ‘open cupboard’, ‘insert cup’	Put-into Cup
4	‘bowl,spoon, tap, knife’, ‘bowl,spoon, tap, knife’, ‘bowl,spoon, tap, knife’, ‘bowl, spoon, cup, tap, knife’, ‘bowl, spoon, tap, knife’, ‘bowl, spoon, tap, knife’, ‘bowl, spoon, cup, knife, bottle’, ‘bowl, cup, tap, knife, lid’, ‘bowl,knife, tap’, ‘bowl, spoon, tap, knife, lid’, ‘bowl, spoon, tap, knife’, ‘bowl,knife, tap’, ‘bowl, spoon, tap, knife’, ‘bowl, spoon, tap, knife, sponge’, ‘knife,tap, spoon’	‘wash cup’, ‘wash cup’, ‘no action’, ‘wash spoon’, ‘wash spoon’, ‘put spoon’, ‘wash cup’, ‘wash cup’, ‘wash cup’,‘wash cup’, ‘wash cup’, ‘wash cup’, ‘wash cup’, ‘no action’, ‘no action’, ’ turn-off tap’	Turn-off tap

Table 5: Per frame objects and actions detected in a video clip in EPICKitchens-100 dataset. The objects are detected using FasterRCNN trained on EK55 dataset. We set a threshold of 0.15, and select only top 5 objects per frame. Actions described here are the ground truth annotations. When actions are not detected, a ‘no action” label is used instead.

interaction be detected.

Effect of actions: In Figure 5, we evaluate how accurate action recognition impacts the performance for action anticipation. During evaluation, we vary the % age ground-truth action labels used by randomly sampling actions for every frame. When 20% actions are predicted, it implies that 80% of the times actions were randomly sampled (i.e., they are incorrect). We notice that as the action recognition accuracy increases, the noun predictions also increase drastically. Adding recognized actions as an additional modality starts to aid in performance when the accuracy of action recognition exceeds 70%. We believe that since the text inputs provide stronger features to the model (as seen from modality ablations), having incorrect actions confuses and deteriorates the performance. However, for unseen classes, an action recognition accuracy of 55% results in performance increase. Overall, we observe that with accurate action and object recognition systems, inputs in the text format can greatly improve prediction performances, without having to train modality specific encoders.

5 Conclusion and Future Work

In this work, we presented Multi-Modal Anticipative Transformer(MAT) – a contrastive learning method that learns contextual information from descriptions generated from actions. The goal of the paper was to evaluate the impact of text based input modalities for action anticipation. We first demonstrated this through a contrastive learning based pre-training protocol, where the model is trained to align the fused modality features with the descriptions generated by LLMs. Secondly, we used the actions and objects detected in the video as text inputs, and fused the text embeddings to other modalities, and thereby observed a significant improvement in the accuracy. We further analyzed the effect of different modalities on performance, and also the impact of the accuracy of action recognition. While we trained to align the modality features with action descriptions from ChatGPT through contrastive learning, learning with small batch sizes tend to be sub-optimal. To this end, we would like to utilize VLMs to prompt videos and learn the actions, objects, context etc., and to utilize this towards anticipating actions. In the future, we would like to utilize a pre-training stage similar to ImageBind [Girdhar *et al.*, 2023], which learns from multiple modalities and datasets. However, initial experiments showed that this is challenging with smaller batch sizes and existing computational constraints, requiring careful handling of data and model architectures.

References

[Abu Farha *et al.*, 2018] Yazan Abu Farha, Alexander Richard, and Juergen Gall. When will you do what?-anticipating temporal occurrences of activities. In *CVPR*, 2018.

[Alayrac *et al.*, 2022] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.

[Bertasius and Torresani, 2020] Gedas Bertasius and Lorenzo Torresani. Cobe: Contextualized object embeddings from narrated instructional video. In *NeurIPS*, 2020.

[Brown *et al.*, 2020] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020.

[Carreira and Zisserman, 2017] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

[Damen *et al.*, 2018] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *ECCV*, pages 720–736, 2018.

[Damen *et al.*, 2020] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision. *arXiv preprint arXiv:2006.13256*, 2020.

[Dessalene *et al.*, 2021] Eadom Dessalene, Michael Maynord, Chinmaya Devaraj, Cornelia Fermuller, and Yiannis Aloimonos. Forecasting action through contact representations from first-person video. *TPAMI*, 2021.

[Dosovitskiy *et al.*, 2021] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[Furnari and Farinella, 2019] Antonino Furnari and Giovanni Maria Farinella. What would you expect? anticipating egocentric actions with rolling-unrolling lstms and modality attention. In *ICCV*, 2019.

[Furnari and Farinella, 2020] Antonino Furnari and Giovanni Maria Farinella. Rolling-unrolling lstms for action anticipation from first-person video. *TPAMI*, 2020.

[Gao *et al.*, 2017] Jiyang Gao, Zhenheng Yang, and Ram Nevatia. Red: Reinforced encoder-decoder networks for action anticipation. In *BMVC*, 2017.

[Girdhar and Grauman, 2021] Rohit Girdhar and Kristen Grauman. Anticipative video transformer @ epic-kitchens action anticipation challenge 2021. In *CVPR Workshop*, 2021.

[Girdhar *et al.*, 2022] Rohit Girdhar, Mannat Singh, Nikhila Ravi, Laurens van der Maaten, Armand Joulin, and Ishan Misra. Omnivore: A single model for many visual modalities. In *CVPR*, 2022.

[Girdhar *et al.*, 2023] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15180–15190, 2023.

[Grauman *et al.*, 2022] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022.

[Huang and Kitani, 2014] De-An Huang and Kris M Kitani. Action-reaction: Forecasting the dynamics of human interaction. In *ECCV*, 2014.

[Jain *et al.*, 2015] Ashesh Jain, Hema S Koppula, Bharad Raghavan, Shane Soh, and Ashutosh Saxena. Car that knows before you do: Anticipating maneuvers via learning temporal driving models. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3182–3190, 2015.

[Jain *et al.*, 2016] Ashesh Jain, Avi Singh, Hema S Koppula, Shane Soh, and Ashutosh Saxena. Recurrent neural networks for driver activity anticipation via sensory-fusion architecture. In *ICRA*, 2016.

[Jia *et al.*, 2021] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.

[Kazakos *et al.*, 2019] Evangelos Kazakos, Arsha Nagrani, Andrew Zisserman, and Dima Damen. Epic-fusion: Audio-visual temporal binding for egocentric action recognition. In *ICCV*, 2019.

[Koppula and Saxena, 2015] Hema S Koppula and Ashutosh Saxena. Anticipating human activities using object affordances for reactive robotic response. *TPAMI*, 2015.

[Korbar *et al.*, 2018] Bruno Korbar, Du Tran, and Lorenzo Torresani. Co-operative learning of audio and video models from self-supervised synchronization. In *NeurIPS*, 2018.

[Li *et al.*, 2018] Yin Li, Miao Liu, and James M Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In *Proceedings of the European conference on computer vision (ECCV)*, pages 619–635, 2018.

[Liu *et al.*, 2020] Miao Liu, Siyu Tang, Yin Li, and James Rehg. Forecasting human object interaction: Joint prediction of motor attention and actions in first person video. In *ECCV*, 2020.

[Liu *et al.*, 2021] Huidong Liu, Shaoyuan Xu, Jinmiao Fu, Yang Liu, Ning Xie, Chien-Chih Wang, Bryan Wang, and Yi Sun. Cma-clip: Cross-modality attention clip for image-text classification. *arXiv preprint arXiv:2112.03562*, 2021.

[Ma *et al.*, 2022] Yiwei Ma, Guohai Xu, Xiaoshuai Sun, Ming Yan, Ji Zhang, and Rongrong Ji. X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 638–647, 2022.

[Minderer *et al.*,] M Minderer, A Gritsenko, A Stone, M Neumann, D Weissenborn, A Dosovitskiy, A Mahendran, A Arnab, M Dehghani, Z Shen, et al. Simple open-vocabulary object detection with vision transformers. arxiv 2022. *arXiv preprint arXiv:2205.06230*.

[Nagarajan *et al.*, 2020] Tushar Nagarajan, Yanghao Li, Christoph Feichtenhofer, and Kristen Grauman. Ego-topo: Environment affor-dances from egocentric video. In *CVPR*, 2020.

[Ni *et al.*, 2022] Bolin Ni, Houwen Peng, Minghao Chen, Songyang Zhang, Gaofeng Meng, Jianlong Fu, Shiming Xiang, and Haibin Ling. Expanding language-image pretrained models for general video recognition, 2022.

[Petković *et al.*, 2019] Tomislav Petković, David Puljiz, Ivan Marković, and Björn Hein. Human intention estimation based on hidden markov model motion validation for safe flexible robotized warehouses. *Robotics and Computer-Integrated Manufacturing*, 57:182–196, 2019.

[Radford *et al.*, 2019] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[Radford *et al.*, 2021] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[Rasouli *et al.*, 2020] Amir Rasouli, Iuliia Kotseruba, and John K Tsotsos. Pedestrian action anticipation using contextual feature fusion in stacked rnns. *arXiv preprint arXiv:2005.06582*, 2020.

[Roy and Fernando, 2021] Debaditya Roy and Basura Fernando. Action anticipation using pairwise human-object interactions and transformers. *IEEE Transactions on Image Processing*, 30:8116–8129, 2021.

[Roy and Fernando, 2023] Debaditya Roy and Basura Fernando. Predicting the next action by modeling the abstract goal, 2023.

[Sener *et al.*, 2020] Fadime Sener, Dipika Singhania, and Angela Yao. Temporal aggregate representations for long-range video understanding. In *ECCV*, 2020.

[Sener *et al.*, 2021] Fadime Sener, Dibyadip Chatterjee, and Angela Yao. Technical report: Temporal aggregate representations. *arXiv preprint arXiv:2106.03152*, 2021.

[Stein and McKenna, 2013] Sebastian Stein and Stephen J McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *UbiComp*, 2013.

[Thakur *et al.*, 2023] Sanket Thakur, Cigdem Beyan, Pietro Morerio, Vittorio Murino, and Alessio Del Bue. Enhancing next active object-based egocentric action anticipation with guided attention. *arXiv preprint arXiv:2305.12953*, 2023.

[Wang *et al.*, 2016] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016.

[Wolf *et al.*, 2020] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45, 2020.

[Wu *et al.*, 2021] Yu Wu, Linchao Zhu, Xiaohan Wang, Yi Yang, and Fei Wu. Learning to anticipate egocentric actions by imagination. *TIP*, 2021.

[Wu *et al.*, 2022] Chao-Yuan Wu, Yanghao Li, Karttikeya Mangalam, Haoqi Fan, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In *CVPR*, 2022.

[Yu *et al.*, 2022] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

[Yuan *et al.*, 2021] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

[Zatsarynna *et al.*, 2021] Olga Zatsarynna, Yazan Abu Farha, and Juergen Gall. Multi-modal temporal convolutional network for anticipating actions in egocentric videos. In *CVPR Workshop*, 2021.

[Zhao *et al.*, 2023] Qi Zhao, Ce Zhang, Shijie Wang, Changcheng Fu, Nakul Agarwal, Kwonjoon Lee, and Chen Sun. Antgpt: Can large language models help long-term action anticipation from videos? *arXiv preprint arXiv:2307.16368*, 2023.

[Zhong *et al.*, 2023] Zeyun Zhong, David Schneider, Michael Voit, Rainer Stiefelhagen, and Jürgen Beyerer. Anticipative feature fusion transformer for multi-modal action anticipation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6068–6077, 2023.