Supplementary: Preserving Modality Structure Improves Multi-Modal Learning

We organize this supplement as follows. We present additional results in Sec. 1, followed by the design choices for Multi-SK in Sec. 2.1, discussion on the computational complexity of Multi-SK in Sec. 2.2 and pseudo-code for the Multi-SK in Sec. 3. Further, we present more qualitative results in Sec. 4 and additional experimental setup details in Sec. 5.

1. Results

1.1. Zero-Shot Classification

We further evaluate the effectiveness of our method on zero-shot action classification task on HMDB [7] and UCF-101 [14] datasets. Following [3, 13], we average the representations from video and audio modalities and use that as the representation for evaluation. The HMDB dataset consists of realistic videos from various sources, including movies and web videos. The dataset is composed of $6,849$ video clips from 51 action categories, with each category containing at least 101 clips. The UCF-101 dataset consists of over $13k$ clips from 101 action classes. To compute accuracy, we perform k-means clustering on top of the features and use the Hungarian matching algorithm [8] to find a one-to-one mapping for each cluster to ground-truth classes and report performance. For these experiments, we test on the full HMDB and UCF-101 datasets. We report the results in Tab. 1. We notice that our proposed method outperforms the baseline on both datasets, especially on the HMDB dataset it achieves 2.1% improvement.

Method	Train Mod.	Train Dataset	Visual BB	HMDB **Acc↑**	UCF-101 **Acc↑**
EAO [13]	tva	HT100M	R152 + RX101	35.4	64.0
Ours	tva	HT100M	R152 + RX101	**37.5**	**64.8**

Table 1: Zero-shot classification on HMDB/UCF-101

1.2. Zero-shot Retrieval MSVD

We further evaluate our approach on MSVD dataset using R152+RX101 features and compare with EAO [13] in Tab. 2). Our approach outperforms the previous SoTA method EAO.

Method	**R@5↑**	**R@10↑**	**MedR↓**	**MeanR↓**
EAO [13]	50.4	64	5	25.9
Ours	**51.6**	**66.6**	5	**24.9**

Table 2: Zero-shot Retrieval results on MSVD.

Method	Visual BB	Textual BB	MSR-VTT **R@5↑**	**R@10↑**	**MedR↓**	**MeanR↓**
EAO [13]	CLIP	word2vec	33.8	43.4	14	72.3
Ours	CLIP	word2vec	**35.1**	**45.1**	**13**	**67.2**
EAO [13]	CLIP	CLIP	36.7	49.1	10	66.7
Ours	CLIP	CLIP	**39.7**	**49.3**	10	**63.3**

Table 3: Zero-shot Retrieval results on **MSR-VTT** with CLIP backbones. BB=Backbone.

1.3. Zero-shot Retrieval with CLIP features

We further evaluate our approach with stronger visual and text backbones *i.e*. CLIP backbones [11]. We use the CLIP model pre-trained on the large WebImageText WIT dataset. To be particular, we use the ViT-B/32 model to extract a single 512-dimensional feature per second for video and a single 512-dimensional feature per word for text. For both modalities, we adapt features after projection into the multi-modal embedding space. We report the performance of zero-shot text-to-video retrieval in Tab. 3. First, we evaluate using CLIP as visual backbone and word2vec as text backbone and compare with EAO [13]. Our approach improves baseline on all the metrics as shown in Tab. 3 with $1.3\%, 1.7\%$ gain for $R@5, R@10$ respectively. Next, we evaluate with CLIP as both visual and textual backbone and show gains on all the metrics, specifically a 3% gain on $R@5$. It can be observed that using CLIP backbone features improves the overall performance compared to R152 + RX101 and word2vec backbones. Overall, these results demonstrate that our approach can even be applied to multi-modal pretrained features where preserving the modality-specific semantic structure can further improve the performance.

1.4. Text-to-Video Only Model

For comparison with text-video only models, we also train our approach on text and video data and compare with

Method	Visual BB	MSR-VTT R@1↑	R@5↑	R@10↑	MedR↓
ActBERT [15]	Res3D+Faster R-CNN	8.6	23.4	33.1	36
HT100M [9]	R152 + RX101	7.5	21.2	29.6	38
NoiseEstim. [1]	R152 + RX101	8.4	22.0	30.4	36
EAO [13]	R152 + RX101	9.6	26.1	36.1	23
Ours	R152 + RX101	**11.4**	**26.6**	**36.3**	**22**

Table 4: Zero-shot Retrieval results on **MSR-VTT** for *Text-Video only* model. For fair comparison, we compare with models trained on HT100M and frozen backbones. BB=Backbone.

state-of-the-art in Tab. 4. Our approach achieves 1.8% improvement in $R@1$ on MSR-VTT dataset. These results further validates the effectiveness of our proposed method in solving zero-shot cross-modal retrieval tasks.

1.5. Analysis for Loss Coefficients

We provide an analysis for loss coefficients λ_{nce} and λ_{sspc} (defined in Sec 3.3 of main text) in Tab. 5. The first two rows demonstrate that increasing the value of λ_{nce} leads to better performance compared to increasing the value of λ_{sspc}. However, the last row demonstrates that our default setup with equal loss weights leads to the best performance.

	λ_{nce}	λ_{sspc}	MSR-VTT R@5↑	R@10↑	MedR↓	MeanR↓
	2.0	1.0	23.9	32.1	25.5	94
	1.0	2.0	22.8	33.1	28.5	94.9
Ours	1.0	1.0	**25.1**	**34.5**	26	**91.8**

Table 5: Analysis for Loss coefficients. Zero-shot Retrieval results on **MSR-VTT** and **YouCook2** datasets.

Cross-modal (CM) **and modality-specific** (MS) SSPC loss coefficient ablation: To empirically demonstrate that preserving MS structure helps in generalization, we conduct an experiment to further analyze the effect of CM and MS SSPC loss and present our results in Tab. 6. Removing either of the SSPC losses leads to a noticeable performance drop. Particularly, the last row demonstrates a $2-3.5\%$ drop in performance when the modality-specific SSPC loss is removed. This empirically validates that preserving modality-specific structure improves cross-modal representation.

	CM	MS	MSR-VTT R@10↑	YouCook2 R@10↑
Ours	1	1	**34.5**	**50.1**
	0	1	31.8	48.3
	1	0	32.4	46.7

Table 6: Effect of Cross-Modal (CM) and Modality-Specific (MS) losses. Zero Shot Retrieval on MSR-VTT and YouCook2 datasets.

2. Multi-Assignment Sinkhorn-Knopp

In this section, first we discuss the design choices for generating the similarity matrix in Sec. 2.1 and then discuss about computational complexity of the proposed Multi-SK in Sec. 2.2

2.1. Design Choices for generating 3D Similarity Matrix

The primary objective of Multi-SK is to select the top K' anchors. As discussed in main text, we generate the 3D similarity matrix, $\mathbf{S}'$, in such a way that there is a pre-defined rank between the channels, which can be utilized for selecting the top K' anchors. In the main text, we describe our default setup (Approach 1) for generating the 3D similarity matrix, $\mathbf{S}'$, where we use a damping factor μ to differentiate between the channels. However, this approach does not provide a fine ranking amongst the top K' channels. To this end, we experiment with another approach (Approach 2) for generating $\mathbf{S}'$ which enables fine ranking between the top K' channels. We generate monotonically decreasing weights for the channels. A simple formulation for generating such continuous channels is $\mathbf{S'}_i = (\eta + (1-\eta)(1-i/K))\mathbf{S}$, where η is the hyperparameter to control the range of weights across the channels.

For our problem, we are only interested in selecting the top K' anchors and not concerned about obtaining a fine ranking of anchors. Hence, we employ approach 1. For analysis, we compare the design choices *i.e.* approach 1 vs 2 of the $3D$ similarity matrix and report results in Tab. 7. For this, we train a model with a $3D$ similarity matrix using approach 2 and set the value of η to 0.1. We observe that our approach slightly outperforms approach 2 validating our hypothesis that for our tasks having a fine ranking between top K' channels is not necessary.

Strategy	MSR-VTT R@5↑	R@10↑	MedR↓	MeanR↓
Approach 2	25.1	34.7	27	97.3
Approach 1 (Ours)	25.1	34.5	26	91.8

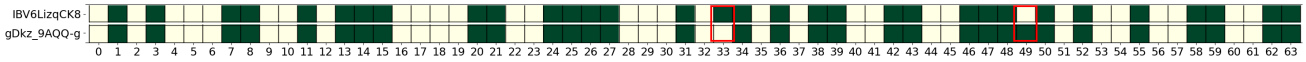
Table 7: Effect of $3D$ similarity matrix generation for zero-shot retrieval task on **MSR-VTT** dataset.

2.2. Computational Complexity for Multi-SK

For the proposed Multi-SK algorithm, in addition to the row-wise and column-wise operation in SK, we per-

(a) IBV6LizqCK8 (Computers and Electronics)

(b) gDkz_9AQQ-g (Computers and Electronics)

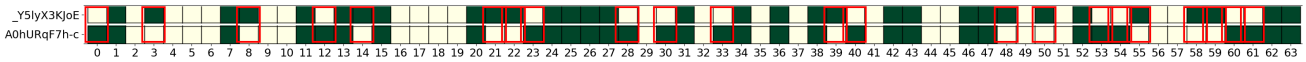


(c) Video Anchor Assignments

Figure 1: Anchor assignments for samples within a category that visually look different. As the samples look different (as shown in (a) & (b)) the anchor assignment (c) is also slightly different demonstrating the effectiveness of our approach to capture variance across samples within a category.Green cell → Anchor assigned, Yellow → Anchor not assigned. Difference in anchor assignments indicated in red.

(a) A0hURqF7h-c (Personal Care and Style)

(b) _Y5lyX3KJoE (Finance and Business)



(c) Video anchors

Figure 2: Anchor assignments for samples across categories that visually look different. The samples look very different (as shown in (a) & (b)) and therefore the anchor assignments are also very different as shown in (c). Green cell → Anchor assigned, Yellow → Anchor not assigned. Difference in anchor assignments indicated in red.

form an additional operation along the depth dimension for each iteration (shown as v, u, ch respectively in the pseudo-code below), thus computational complexity does not scale quadratically with input. Let N, K represent the number of samples and anchors respectively. The overall computation time of Multi-SK is $O(2 \times N \times K + K^2) \approx O(N \times K)$ as $N \gg K$; thus keeping the overall amortized time complexity *similar* to vanilla SK.

3. Pseudo Code: Multi-Assignment Sinkhorn-Knopp

Here, we present the pseudo-code for the proposed Multi-SK algorithm.

```
# Multi-Assignment Sinkhorn-Knopp
def multi_sk(scores, eps, niters=10):
  Q = exp(scores / eps)
  Q /= sum(Q)
  K, N, K = Q.shape
  for _ in range(niters):
    # Row normalization
    v = 1 / sum(Q, dim=2, keepdim=True)
    Q *= v
    # Column normalization to enforce
    # equal partition constraint N/K
    u = (N/K) / sum(Q, dim=1, keepdim=True)
    Q *=  u
    # Depth normalization for
    # unique anchors per sample
    ch = 1 / sum(Q, dim=0, keepdim=True)
    Q *= ch
  return (Q/sum(Q, dim=2, keepdim=True))
```

4. Qualitative Results

In this section, we present a fine-grained visual analysis of the learned anchors in Sec. 4.1 and also show the distribution of samples across HT100M categories based on anchor assignments in Sec. 4.2, followed by qualitative retrieval results in Sec. 4.4.

4.1. Fine-grained Anchor Analysis.

We show fine-grained analysis of the learned anchors below. We compare the anchor assignments for the following scenarios to demonstrate the effectiveness of anchors.

- Similar categories (Figure 2 at Main Text)
- Within a category (Figure 1)
- Different categories (Figure 2)
- Confusing samples (Figure 3)

For the purpose of this analysis, we visualize the anchor assignments as binary assignments. However, during training we use soft anchor assignments.

In Figure 2 (Main text), we compare the anchor assignments for samples from *similar categories*. It can be seen that the videos are visually similar even though they belong to different categories and the anchors learned are presented in Figure 2 (Main text). We notice that the anchor assignments for these two examples are able to capture the sample similarity. To be particular, all the assigned anchors except for anchors 39 and 54 (as highlighted) are the same for these two visually similar examples. This further validates our claim that our proposed method can assign semantically meaningful anchors without any explicit supervision.

Next, we analyze the anchors for samples *within a category* as shown in Figure 1. Here, we observe that as the samples vary within a particular category, so do the corresponding anchors. This validates the flexibility of our multi-anchor based sample representation in modeling the intra-class variance.

In Figure 2, we compare the anchor assignments for videos from *different categories* and it can be seen that the anchor assignments are very different as expected. This shows that our anchor modeling does not collapse to a fixed assignment rather it can model different classes differently.

Finally, in Figure 3, we compare anchor assignments for *confusing samples*. Here, we show one such example where the actual label is *Car and other Vehicles* while the video contains frames similar to *kitchen items* as shown in Figure 3. We analyse both visual and textual anchors for this sample. It can be seen that the visual anchors (Figure 3d) for *confusing car* sample (Figure 3a) is closer to visual anchors for *food* sample (Figure 3c) while the text anchors of *confusing car* sample is closer to the actual car sample (Figure 3e). This shows that the textual anchors can represent the car concept which is missing in the visual data. This demonstrates the effectiveness of our approach as is it able to capture both visual similarity and textual similarity, allowing flexibility across modalities.

4.2. Distribution of samples across categories in HT100M

We compute the distribution of samples per category based on anchor assignments to analyze the relationship of anchors to semantic categories. In Figure. 4, we show the distribution w.r.t single anchor and a pair of anchors. As shown in Figure 4 (a), even a single anchor assignment is grouping similar categories reasonably and 4(b) when filtered based on pair of anchors, it is able to capture distinctive aspects with the distribution specializing towards specific categories. This demonstrates that utilizing multiple anchors for modeling relationship between samples helps to capture both shared and unique aspects of the data.

Anchor	Top Categories
0	Relationships, Youth, Family life
22	Finance & Business, Education
30	Food & Entertaining
56	Cars, Computers & Electronics

Table 8: Anchor Analysis: Top video categories for each anchor.

4.3. Qualitative Anchor Analysis

Here, we provide additional anchor analysis. In the above section, we show the distribution of samples per category based on anchor assignments. In Tab. 8, we show top video categories for different anchors on HT100M dataset.

4.4. Qualitative Retrieval Results

We present more qualitative Text-to-Video Zero-shot Retrieval results of our approach on both datasets in Fig. 5

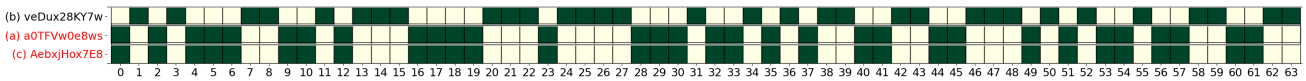
5. Experimental Setup: Additional Details

Following previous works [9, 12, 3, 13], as visual backbone, we use a combination of ResNet-152 [6], pretrained on Imagenet [4] and compute one 2D-feature (2048 dim) per second, as well as ResNeXt101 [5] pretrained on Kinetics [2] to get 1.5 3D-feature (2048 dim) per second. We temporally upsample 2D-features with nearest neighbors to have the same number of features as 3D-features and concatenate them to obtain 4096-dimensional vectors. As the text backbone, we use GoogleNews pretrained Word2vec model [10] with 300-dimensional embedding per word. These backbones are *frozen* and not finetuned during training. Following [3, 13], we use a trainable CNN with residual layers as an audio backbone and adapt the last two residual blocks to extract 1.5 4096-dimensional features per second.

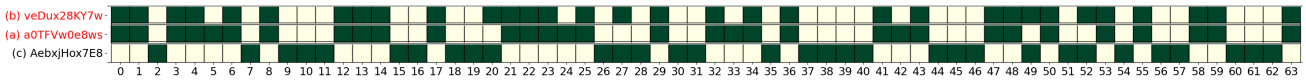
(a) `a0TFVw0e8ws` (Cars and other Vehicles)

(b) `veDux28KY7w` (Cars and other Vehicles)

(c) `AebxjHox7E` (Food and Entertaining)

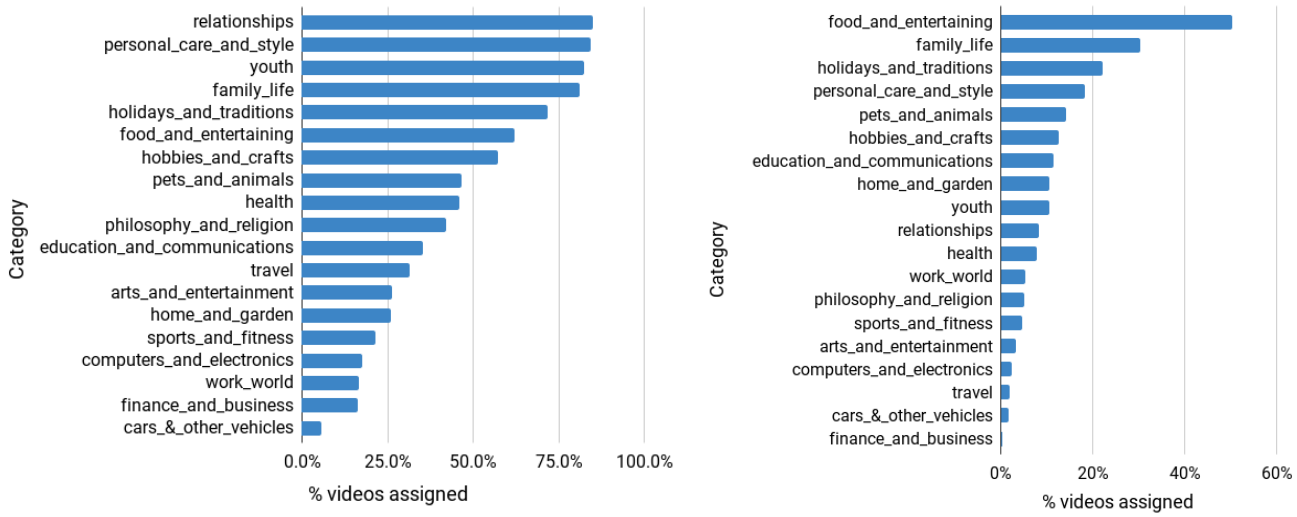


(d) Video Anchor Assignments



(e) Text Anchor Assignments

Figure 3: Video and Text anchor assignments for confusing samples. Here (a) and (b) $\rightarrow$ samples from category Cars & Other Vehicles and (c) $\rightarrow$ Food & Entertaining category in *HT100M* dataset. (a), (c) look visually similar hence the video anchor assignments are similar. Interestingly, the text anchor assignments for (a), (b) are similar as our method is able to capture the concept `car` from text which is missing in the video. Green cell $\rightarrow$ Anchor assigned, Yellow $\rightarrow$ Anchor not assigned. YouTube IDs for videos with similar anchor assignments highlighted in red.



(a) (b)

Figure 4: Distribution of samples across categories in HT100M w.r.t anchor assignments. We show % videos per category filtered based on (a) a single anchor assignment (b) a pair of anchor assignments. Even a single anchor can model relationships between categories reasonably and similar categories are grouped together, and for a combination of 2 anchors the distribution gets skewed towards a single category and models the relationships better.

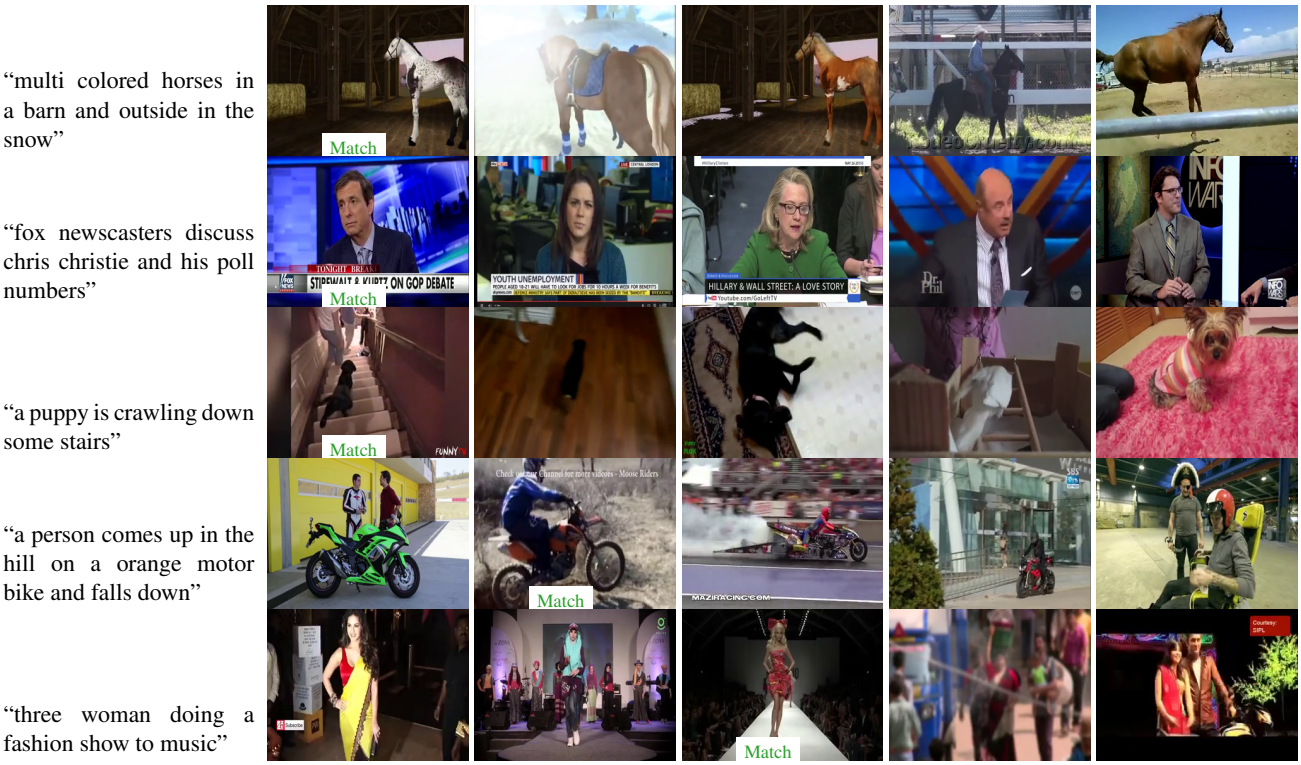


Figure 5: Examples of Zero-Shot Text-to-Video Retrieval on **MSR-VTT** dataset. Each row consists of Textual Query, and top-5 retrieved videos for our method. *Match* indicates correct video for the query.

References

[1] Elad Amrani, Rami Ben-Ari, Daniel Rotman, and Alex Bronstein. Noise estimation using density estimation for self-supervised multimodal learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6644–6652, 2021. 2

[2] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 4

[3] Brian Chen, Andrew Rouditchenko, Kevin Duarte, Hilde Kuehne, Samuel Thomas, Angie Boggust, Rameswar Panda, Brian Kingsbury, Rogerio Feris, David Harwath, et al. Multimodal clustering networks for self-supervised learning from unlabeled videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8012–8021, 2021. 1, 4

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 4

[5] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555, 2018. 4

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. cvpr. 2016. *arXiv preprint arXiv:1512.03385*, 2016. 4

[7] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International conference on computer vision*, pages 2556–2563. IEEE, 2011. 1

[8] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 1

[9] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2630–2640, 2019. 2, 4

[10] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. 4

[11] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*. PMLR, 2021. 1

[12] Andrew Rouditchenko, Angie Boggust, David Harwath, Brian Chen, Dhiraj Joshi, Samuel Thomas, Kartik Audhkhasi, Hilde Kuehne, Rameswar Panda, Rogerio Feris, et al. Avlnet: Learning audio-visual language representations from instructional videos. *arXiv preprint arXiv:2006.09199*, 2020. 4

[13] Nina Shvetsova, Brian Chen, Andrew Rouditchenko, Samuel Thomas, Brian Kingsbury, Rogerio S Feris, David Harwath, James Glass, and Hilde Kuehne. Everything at once-multimodal fusion transformer for video retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20020–20029, 2022. 1, 2, 4

[14] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 1

[15] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020. 2