Video-Text Representation Learning via Differentiable Weak Temporal Alignment (Supplement)

Dohwan Ko[1] Joonmyung Choi[1] Juyeon Ko[1] Shinyeong Noh[1]
Kyoung-Woon On[2] Eun-Sol Kim[3] Hyunwoo J. Kim[1*]

[1]Department of Computer Science and Engineering, Korea University
[2]Kakao Brain [3] Department of Computer Science, Hanyang University
{ikodoh, pizard, juyon98, dneirfi, hyunwoojkim}@korea.ac.kr
{kloud.ohn}@kakaobrain.com {eunsolkim}@hanyang.ac.kr

We first introduce experimental setup and describe additional experiments including ablation studies and qualitative analysis. Then, we provide further discussions about negative societal impacts, limitations, and future directions.

A. Experimental Setup

A.1. Datasets

Action Recognition. We evaluate our learned visual representations on the action recognition task with HMDB51 [5] and UCF101 [9] datasets. We use only video representations for this task and evaluate in the same protocol as [6]. The HMDB51 contains 7K videos from 51 human action categories collected from movies and open sources. The UCF101 contains 13K videos divided into 101 realistic action categories collected from YouTube.

Video and Text Retrieval. We evaluate on the video-to-text retrieval and text-to-video retrieval tasks with two widely used benchmarks: YouCook2 [18] and MSR-VTT [13]. The YouCook2 provides 2,000 instructional long untrimmed videos for 89 recipes collected from YouTube. The MSR-VTT contains 200K clip-sentence pairs with 20 natural sentences per clip for video understanding. We report performance using the recall at K (R@K) metric (K=1,5,10).

Action Step Localization. CrossTask is used to evaluate localization performance, which measures the number of correct step assignments. Following the evaluation protocol of [20], we report the performance using CrossTask average recall (CTR) metric. The CrossTask dataset provides 4.7K instructional videos, collected for 83 tasks that are divided into 18 primary and 65 related tasks.

A.2. Implementation Details

Backbone Model. We use S3D [12] for a visual encoder f and a fully connected layer with pretrained word2vec [8] embeddings for a text encoder g. We train S3D from scratch. We randomly sample 8 consecutive clip-caption pairs per one video (*i.e.*, $n = m = 8$). Each clip consists of 16 frames of 5 fps (3.2 seconds) and the size of each frame is 224×224. For the word embedding, we use the word2vec pretrained by Google News with the dimension of 300. The output dimensions of representations for both clips and captions are 512. For the results of MIL-NCE, we report the performance of their official github since our code is based on it.

Hyperparameters. We use a smoothing parameter $\gamma = 0.1$ at the soft-min and a temperature $\tau = 0.02$ in Eq. (12) of the main paper. As a distance metric, we use a shifted cosine distance by -1, *i.e.*, $\delta_{i,j} = -\frac{f(x_i)^\top g(y_j)}{\|f(x_i)\| \cdot \|g(y_j)\|}$.

Optimization. We use the ADAM [3] optimizer with a cosine annealing and train our model for 300 epochs. The warm-up strategy is adopted with the learning rate from 10^{-5} to 10^{-3} during first 100K steps before the cosine annealing. For ablation studies, we use 10% of the HowTo100M dataset and train 100 epochs.

B. Additional Experiments

B.1. Fine-Tuning on Downstream Tasks

In the main paper, we conducted a zero-shot learning setting to evaluate only the quality of learned representations. In this section, we also fine-tune our backbone model to various downstream tasks to evaluate the adaptability of learned representations.

Text-to-Video Retrieval. We fine-tune the backbone model on the text-to-video retrieval task.

Table 1 and 2 show the results of the fine-tuned models on the YouCook2 and MSR-VTT datasets. VT-TWINS is superior to or on par with strong transformer-based [11] baselines (*e.g.*, COOT, ActBERT, and TACo) even though we do not use a transformer-based cross-modal encoder. It

*is the corresponding author.

Method	Backbone	R@1	R@5	R@10	MedR
Miech *et al*. [7]	R3D-101 + w2v	8.2	24.5	35.3	24
ActBERT [19]	R3D-101 + BERT	9.6	26.7	38.0	19
VideoAsMT [4]	-	11.6	-	43.9	-
COOT [1]	I3D + Transformer	16.7	40.2	52.3	9
TACo [14]	S3D + BERT	16.1	40.3	52.2	9
VT-TWINS	S3D + w2v	**17.2**	**43.8**	**57.2**	**7**

Table 1. **Text-to-Video Retrieval on YouCook2.**

Method	Backbone	R@1	R@5	R@10	MedR
C+LSTM+SA [10]	VGG-19	4.2	12.9	19.9	55
SNUVL [16]	R3D-152 + LSTM	3.5	15.9	23.8	44
Kaufman *et al*. [2]	-	4.7	16.6	24.1	41
CT-SAN [17]	R3D-152 + LSTM	4.4	16.6	22.3	35
JSFusion [15]	R3D-152 + GloVe	10.2	31.2	43.2	13
Miech *et al*. [7]	R3D-152 + w2v	14.9	40.2	52.8	**9**
VideoAsMT [4]	-	14.7	-	52.8	-
ActBERT [19]	R3D-101 + BERT	16.3	**42.8**	**56.9**	10
VT-TWINS	S3D + w2v	**19.4**	40	52.5	**9**

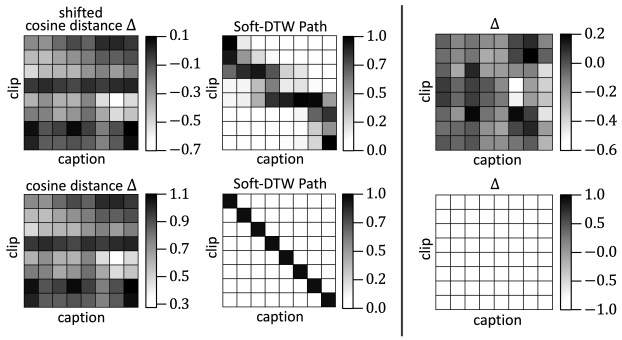
Table 2. **Text-to-Video Retrieval on MSRVTT.**

shows that our proposed alignment algorithm helps to learn the powerful representations of video and text.

B.2. Ablation Studies and Qualitative Analysis

Shifted Cosine Distance. As mentioned in Section 5.1.2 of the main paper, we use the shifted cosine distance instead of the original cosine distance. When using the original cosine distance, the range of the distance is [0, 2], *i.e*., positive values. Therefore, the DTW tends to find a trivial path, *e.g*., a diagonal path, since it passes the minimum number of pairs. On the other hand, there are both negative and positive values if using shifted cosine distance because it is in the range of [-1, 1]. These negative values encourage the DTW path to visit more pairs since the more negative valued pairs make the total cost decrease. Figure 1a shows the different DTW paths, using cosine distance (the bottom row) and shifted cosine distance (the top row). The DTW path with the original cosine distance tends to form a trivial diagonal path regardless of the values of the pairwise distance matrix. In Table 3, (1) and (2) show that the alignment path obtained by shifted cosine distance (**S**) improves the performance by a large margin compared to the original cosine distance (**C**).

Feature Collapsing. As aforementioned, minimizing the DTW loss alone (*i.e*., without negative pairs) causes feature collapsing. The DTW with negative pairs, which can be interpreted as the contrastive learning scheme, avoids feature collapsing by repelling the negative pairs. In Figure 1b, the top one shows the pairwise distance with a contrastive



(a) Cosine Distance. (b) Feature Collapsing.

Figure 1. **Results of Shifted Cosine Distance and Feature Collapsing.** Δ is a pairwise distance matrix. The Soft-DTW path matrix is the gradient matrices M defined in Section 4.1 of the main paper.

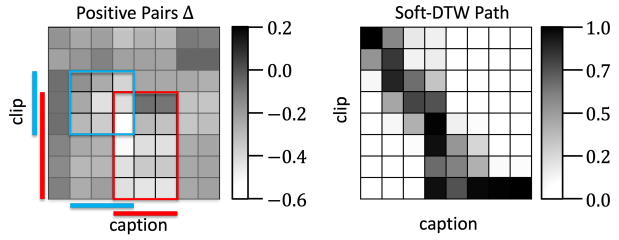
learning scheme and the bottom one shows the pairwise distance without contrastive learning, *i.e*., just minimizing the DTW loss. Without negative pairs, all the embeddings of clips and captions are converged to a single point (*i.e*., feature collapsing) so that the distances of all pairs are about -1. (3) in Table 3 demonstrates that, with feature collapsing, the model turns to be incapable of any tasks compared to (1) learned with contrastive learning scheme.

Smoothing Parameter γ. We also experiment with various values of γ from the soft-min function to find an optimal value. As mentioned in Section 3.1 of the main paper, larger γ makes the Soft-DTW path take into account the cost of suboptimal paths. (1), (4), and (5) in Table 3 show the results of various γ and it shows the best performance at $\gamma = 0.1$. The results demonstrate that considering proper suboptimal paths with the optimal path helps to learn representations.

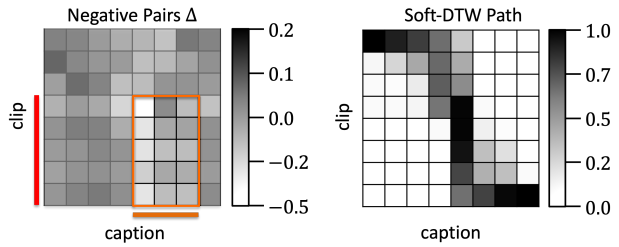
Hard Negative Mining. As mentioned in Section 4.4 of the main paper, VT-TWINS also implicitly mines the hard negatives by applying InfoNCE loss to the S2DTW. Figure 3 is an example of positive and negative clip-caption pairs of a particular series of clips. Figure 2 shows the pairwise distance and the Soft-DTW path of Figure 3, *i.e*., Figure 2a shows the positive pairs (left) of Figure 3 and Figure 2b shows the negative pairs (right) of Figure 2b. There are several alignments between positive clip-caption pairs. However, some negative captions are also aligned with the clips, like the ones including 'sliced apple' and 'apple juice' from the negative captions. VT-TWINS automatically aligns the clips with the negative captions and implicitly mines the hard negative pairs, *i.e*., the clips including 'sliced apple' repels the captions including 'apple juice' more than the other ones.

	CL	CD	γ	HMDB	UCF	YC2	MV	CT
(1)	✔	**S**	0.1	42	72.1	12.5	17.4	28.2
(2)	✔	**C**	0.1	38.6	68	6.8	9.7	22.5
(3)	-	**S**	0.1	5.7	7.9	0	0.3	11.9
(4)	✔	**S**	0.01	33.1	62.1	10.4	13.8	21.8
(5)	✔	**S**	1	33.7	57.8	9.8	12.7	24

Table 3. **Ablation Studies.** We report accuracy on the **HMDB**[1] and **UCF**[2], R@10 on the YouCook2[3](**YC2**) and MSR-VTT[4](**MV**), and CTR on the CrossTask (**CT**) to evaluate the contribution of the followings: contrastive learning scheme (**CL**), cosine distance (**CD**), and smoothing parameter (γ). (1) is our proposed model, VT-TWINS. For **CD**, we evaluate the following strategies: **S**: shifted cosine distance (ours), and **C**: original cosine distance.



(a) Positive Pairs.



(b) Negative Pairs.

Figure 2. **Soft-DTW Path of Positive Pairs and Negative Pairs.** The red box and blue box of (a) are the aligned pairs between positive clip-caption pairs. The orange box of (b) is the aligned pairs between negative clip-caption pairs.

C. Further Discussions

C.1. Negative Societal Impacts

This paper introduces a multi-modal self-supervised representation learning algorithm using a large-scale video dataset, HowTo100M[5], whose capacity is about 13TB. Training the HowTo100M requires a lot of GPUs or TPUs and they emit CO_2 which is the main cause of environmental pollution including global warming.

C.2. Limitations and Future Directions

We propose a multi-modal self-supervised representation learning algorithm between video and text. Applying our framework to other modalities and extending it beyond two modalities (*e.g.*, audio) can be interesting. In addition, applying it to the transformer-based [11] encoder as a backbone model is also promising. For example, our proposed framework can be applied to the cross-modal transformer-based encoder, and it will learn more powerful representations in multi-modal settings. Also, as mentioned above, a lot of computing resources and time are wasted to train our model. We need to lighten our model so that it is used in various fields and domains. These problems are left for future works.

[1] Licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) License.

[2] Copyright ©2011 CRCV.

[3] Copyright ©2018 MichiganCOG. Licensed under MIT License

[4] Copyright ©2021 Microsoft.

[5] Copyright ©Inria.

References

[1] Simon Ging, Mohammadreza Zolfaghari, Hamed Pirsiavash, and Thomas Brox. Coot: Cooperative hierarchical transformer for video-text representation learning. In *NeurIPS*, 2020. 2

[2] Dotan Kaufman, Gil Levi, Tal Hassner, and Lior Wolf. Temporal tessellation: A unified approach for video analysis. In *ICCV*, 2017. 2

[3] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2014. 1

[4] Bruno Korbar, Fabio Petroni, Rohit Girdhar, and Lorenzo Torresani. Video understanding as machine translation, 2020. 2

[5] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *ICCV*, 2011. 1

[6] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, 2020. 1

[7] 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 *ICCV*, 2019. 2

[8] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *ICLR workshop*, 2013. 1

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

[10] Atousa Torabi, Niket Tandon, and Leonid Sigal. Learning language-visual embedding for movie understanding with natural-language, 2016. 2

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 1, 3

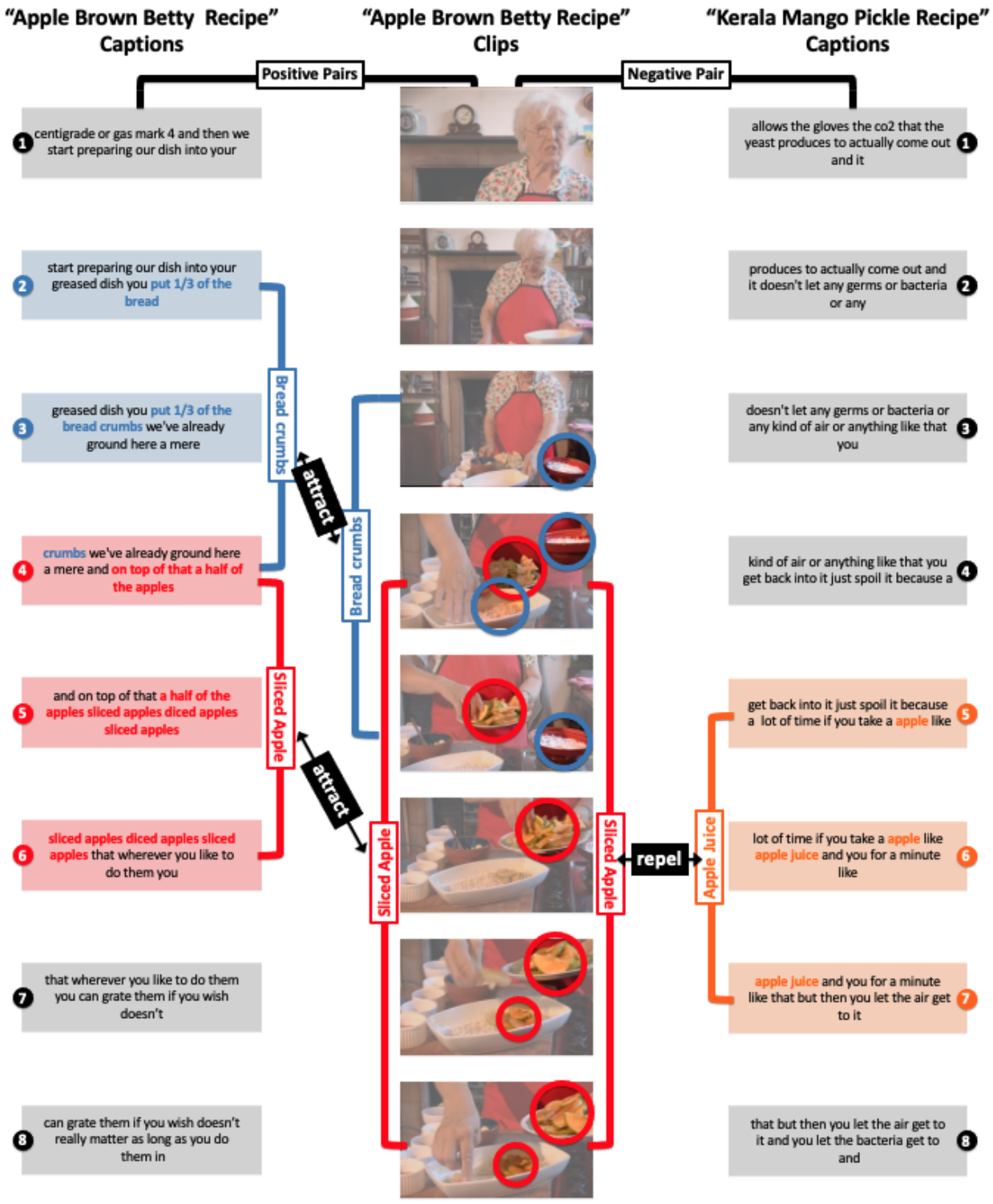


Figure 3. **Comparison between Positive Pairs and Hard Negative Pairs.** The left captions and the right captions are positive pairs and negative pairs of the given center clips, respectively. The alignments between positive pairs are attracted each other and the alignments between negative pairs are repelled each other.

[12] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *ECCV*, 2018. 1

[13] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016. 1

[14] Jianwei Yang, Yonatan Bisk, and Jianfeng Gao. Taco: Token-aware cascade contrastive learning for video-text alignment. In *ICCV*, 2021. 2

[15] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *ECCV*, 2018. 2

[16] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. Video captioning and retrieval models with semantic attention, 2016. 2

[17] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. End-to-end concept word detection for video captioning, retrieval, and question answering. In *CVPR*, 2017. 2

[18] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. 1

[19] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, 2020. 2

[20] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *CVPR*, 2019. 1