

SNP-S³: Shared Network Pre-training and Significant Semantic Strengthening for Various Video-Text Tasks

Xingning Dong^{1*}, Qingpei Guo^{1*}, Tian Gan^{2†}, Qing Wang¹,
Jianlong Wu³, Xiangyuan Ren¹, Yuan Cheng⁴, Wei Chu¹

¹Ant Group, ²Shandong University, ³Harbin Institute of Technology (Shenzhen), ⁴Fudan University
dongxingning1998@gmail.com, qingpei.gqp@antgroup.com, gantian@sdu.edu.cn,
wq176625@antgroup.com, jlwu1992@pku.edu.cn, xiangyuan.rxy@antgroup.com,
cheng-yuan@fudan.edu.cn, weichu.cw@antgroup.com

Abstract

We present a framework for learning cross-modal video representations by directly pre-training on raw data to facilitate various downstream video-text tasks. Our main contributions lie in the pre-training framework and proxy tasks. First, based on the shortcomings of two mainstream pixel-level pre-training architectures (limited applications or less efficient), we propose Shared Network Pre-training (SNP). By employing one shared BERT-type network to refine textual and cross-modal features simultaneously, SNP is lightweight and could support various downstream applications. Second, based on the intuition that people always pay attention to several “significant words” when understanding a sentence, we propose the Significant Semantic Strengthening (S³) strategy, which includes a novel masking and matching proxy task to promote the pre-training performance. Experiments conducted on three downstream video-text tasks and six datasets demonstrate that, we establish a new state-of-the-art in pixel-level video-text pre-training; we also achieve a satisfactory balance between the pre-training efficiency and the fine-tuning performance. The codebase are available at https://github.com/alipay/Ant-Multi-Modal-Framework/tree/main/prj/snps3_vtp.

1. Introduction

Owing to successful applications of pre-training methods in NLP [7, 43] and CV [5, 22], more and more researchers attempt to explore this “Pre-training & Fine-tuning” paradigm in the video-text field [25, 33], which has achieved remarkable performance gain in various downstream video understanding tasks, such as video-text re-

*Xingning Dong and Qingpei Guo contributed equally to this manuscript.

†Tian Gan is the corresponding author.

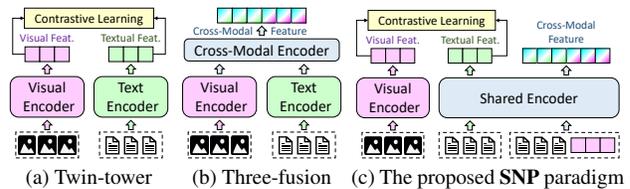


Figure 1. Comparison of mainstream pixel-level pre-training architectures: a) Twin-tower-based, b) Three-fusion-based, and c) the proposed Shared Network Pre-training (SNP) methods.

trieval [10, 38, 53], video question answering [44, 55, 59], and video reasoning [6, 15, 42, 54, 57]. There are two mainstream paradigms in current video-text pre-training methods: the feature-level paradigm and the pixel-level one.

Compared with feature-level pre-training methods [24, 31, 47] that employ off-the-shelf visual and textual features extracted by frozen models, pixel-level pre-training methods [3, 14, 21] treat raw visual pixels and text tokens as inputs, which could optimize the cross-modal learning ability in an end-to-end manner. Thus, the pixel-level paradigm tends to achieve better performance and has been widely followed. There are two mainstream pixel-level pre-training architectures, *i.e.*, twin-tower-based [3, 11, 14] in Figure 1a and three-fusion-based [13, 21, 23] in Figure 1b. Twin-tower-based models are usually lightweight and time-efficient; however, since they do not generate cross-modal video representations, their applications are limited mainly in the cross-modal retrieval task. Three-fusion-based models usually contain three separate encoders to embed visual, textual, and cross-modal features. Though they could support various downstream video understanding applications, they usually contain massive training parameters, leading to computational inefficiency and high cost of GPU memory.

Towards this end, we propose a new architecture that

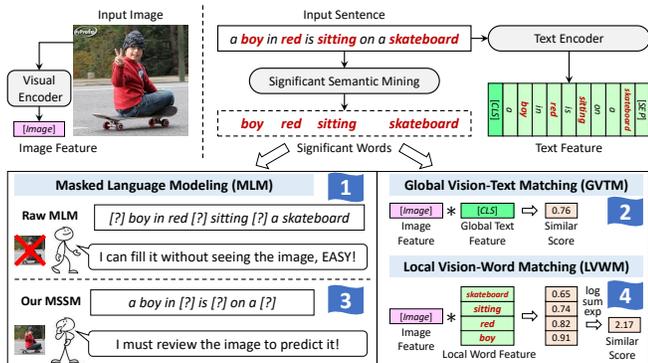


Figure 2. Comparison of two widely-employed masking and matching proxy tasks (MLM-1 and GVTM-2) and our improved version (MSSM-3 and LVWM-4).

not only supports various video-text tasks like three-fusion-based models, but also as lightweight as twin-tower-based ones. Based on the thorough investigation, we observe that: 1) the text and cross-modal encoder in conventional three-fusion-based models [13, 21, 23] are mainly BERT-type transformers; 2) the inputs and outputs of both encoders are token-type features; and 3) as CLIP4Clip [30] pointed out, it is hard to find suitable parameters to initialize the cross-modal encoder, leading to sub-optimal pre-training performance. Therefore, we propose the Shared Network Pre-training (SNP) method. As shown in Figure 1c, SNP employs a shared BERT-type network to refine textual and cross-modal features simultaneously, combining the advantages of twin-tower-based and three-fusion-based methods.

In order to promote the cross-modal interaction for better performance, current video-text pre-trained models would set several proxy tasks, where Masked Language Modeling (MLM) and Global Vision-Text Matching (GVTM) are two widely-employed ones. However, for the conventional MLM, some masking words could be easily filled according to grammar without reviewing the image. *E.g.*, given a sentence “[?] *boy in red* [?] *sitting* [?] *a skateboard*”, one can directly refer to “a”, “is” and “on”. Thus, it seems that conventional MLM could hardly benefit the cross-modal interaction. For GVTM that aims to model the cross-modal alignment, current methods usually take pair-wised visual features and global-pooling textual features (refer to the hidden state of the token [cls]) as inputs. However, global-pooling textual features focus on the sentence level, which would omit the local information of some informative semantics at the word level, leading to limited performance.

Intuitively, humans usually capture several “significant words” when understanding a sentence. *I.e.*, some words (*e.g.*, verbs and nouns) would provide significant information while others (*e.g.*, prepositions and conjunctions) only play the role of “lubricants” to make the sentence fluent and

vivid. In order to improve the cross-modal interaction, we hope pre-trained models would lay their emphasis on those significant words rather than other trivial ones. Therefore, we propose the Significant Semantic Strengthening (S^3) strategy, leading pre-trained models to automatically find and emphasize these significant semantics within the input sentence. As shown in the third and fourth parts of Figure 2, S^3 includes two novel proxy tasks for better cross-modal interaction: 1) Masked Significant Semantic Modeling (MSSM) masks out informative words to force models to resume these clozes from textual and visual information, replacing the conventional MLM. 2) Local Vision-Word Matching (LVWM) learns the cross-modal interaction at the word level, which is a complementary to existing GVTM.

Our contributions are summarized in three-folds:

- We propose Shared Network Pre-training (SNP), which is a lightweight pixel-level pre-training method and could support various downstream video-text applications.
- We propose the Significant Semantic Strengthening (S^3) strategy, including two novel proxy tasks (MSSM and LVTM), which is model-agnostic, parameter-free, and could facilitate the cross-modal interaction.
- Experiments conducted on three downstream video-text tasks and six datasets indicate the superiority of our proposed method, which establishes a new state-of-the-art in the field of video-text pre-training.

2. Related Work

Video-Text Pre-training and Fine-Tuning.

Inspired by superior performance of Transformers [12, 32, 49] and BERT [9, 18, 52], video-text pre-training has attracted increasing interest in recent years, which could be roughly divided into feature-level pre-training methods [24, 29, 47] and pixel-level ones [3, 21, 60]. Since the former approaches employ offline visual and textual features extracted from frozen models (*e.g.*, S3D [45] and DistillBert [34]), they would limit the fine-tuning performance as there remain domain gaps between pre-training datasets and frozen feature extractors. While for pixel-level video pre-training methods, they attempt to learn cross-modal representations from raw visual pixels and text tokens in an end-to-end manner, whose frameworks are mainly twin-tower-based [3, 11, 39] architectures or three-fusion-based [13, 21, 23] ones. However, both architectures have their limitations: Twin-tower-based methods could hardly support various downstream video understanding tasks like the latter. In contrast, three-fusion-based methods contain more parameters and are not as lightweight as the former. Due to the high cost of GPU memory, conventional three-fusion-based methods mainly pre-train their models

on large-scale image-text datasets, and fine-tune them on downstream video-text tasks.

In this work, we propose Shared Network Pre-training, which is a lightweight pixel-level pre-training architecture and could support various downstream video-text tasks.

Significant Element Mining in Video Pre-Training.

Recently a few video-text pre-training methods have started to consciously recognize and emphasize the “significant elements” in various proxy tasks for better fine-tuning performance. 1) For **masking** tasks, MERLOT [58] and VIOLET [13] propose the Attended Masking (AM) strategy, which optimizes conventional MLM by masking out 50% of tokens with high attention weights calculated from a language-only transformer (MERLOT) or a cross-modal encoder (VIOLET). Different from those methods, we optimize conventional MLM by explicitly leveraging the Parts-of-Speech (POS) tags within a sentence. 2) For **matching** tasks, TACo [51] proposes a token-level contrastive loss based on the maximum dot-products of visual and textual token features. However, TACo may still omit some local information as it only takes one token embedding into computation like conventional GVTM. Besides, 3) some innovative proxy tasks that leverage those significant elements have been proposed, *e.g.*, Multiple Choice Questions (MCQ) [14] first builds several questions by erasing verb/noun phrases of a sentence. It then forces the model to select right answers from several candidates.

Based on these successful explorations, we attempt to leverage the “Significant Elements” for better cross-modal interaction in both masking and matching proxy tasks.

3. Methodology

We propose **SNP-S³**, which is a pixel-level video pre-training method following the conventional protocol that first pre-trains on large-scale image-text datasets and then fine-tunes on downstream video-text tasks. Moreover, we report the results pre-trained on video-text datasets in the EXPERIMENT section for fair comparison. To achieve a satisfactory balance between the pre-training efficiency and the fine-tuning performance, we simplify the conventional framework and propose two novel proxy tasks.

3.1. Pre-training on Image-Text Datasets

Given a mini-batch (denoted as \mathcal{B}) of images $\{I_i\}_{i=1}^{|\mathcal{B}|}$ and their corresponding descriptions $\{S_i\}_{i=1}^{|\mathcal{B}|}$, pixel-level pre-training methods would first extract visual features $\{\mathbf{v}_i\}_{i=1}^{|\mathcal{B}|}$ and textual features $\{\mathbf{t}_i^{cls}\}_{i=1}^{|\mathcal{B}|}$ from raw data, and then generate cross-modal video representations $\{\mathbf{m}_i^{cls}\}_{i=1}^{|\mathcal{B}|}$ to facilitate various downstream video-text tasks (Note that twin-tower-based methods would skip this step).

3.1.1 Three-fusion-based Pre-training Architecture

As shown in Figure 1b, conventional three-fusion-based pre-training methods usually contain a visual encoder E_{vis} , a text encoder E_{txt} , and a cross-modal encoder E_{mul} . Given an image-text pair (I, S) , we first employ a BERT embedder E_B to process the sentence S into fixed-length token embeddings $\mathbf{W} = [\mathbf{w}^{cls}, \mathbf{w}^1, \mathbf{w}^2, \dots, \mathbf{w}^{N_t-1}]$, where $\mathbf{W} \in \mathbb{R}^{N_t \times d}$, N_t is the length of tokens, and d is the embedding dimension. Notably, each element in \mathbf{W} except the special tokens (*e.g.*, $[cls]$) could be treated as a word embedding. We then employ E_{vis} and E_{txt} to obtain image features $\mathbf{v} \in \mathbb{R}^{1 \times d}$ and textual features $\mathbf{T} \in \mathbb{R}^{N_t \times d}$ from raw image pixels I and token embeddings \mathbf{W} . Afterwards, we concatenate these two features into $[\mathbf{T}, \mathbf{v}]$, and feed them into E_{mul} to obtain cross-modal features $\mathbf{M} \in \mathbb{R}^{(N_t+1) \times d}$. This forward process could be formulated as follows:

$$\mathbf{M} = E_{mul}([E_{txt}(E_B(S)), E_{vis}(I)]), \quad (1)$$

where $\mathbf{M} = [\mathbf{m}^{cls}, \mathbf{m}^1, \mathbf{m}^2, \dots, \mathbf{m}^{N_t}]$ and $[\cdot, \cdot]$ denotes the concatenation operation. Notably, we treat the first $[cls]$ features \mathbf{m}^{cls} as global-pooling video representations.

3.1.2 SNP: Shared Network Pre-training

Based on the thorough investigation of twin-tower-based (Figure 1a) and three-fusion-based (Figure 1b) architectures, we aim to absorb their advantages and overcome their shortcomings for better pre-training performance. Therefore, we propose the novel Shared Network Pre-training (SNP) architecture. As illustrated in Figure 1c, we simplify the three-fusion-based paradigm by employing a shared BERT-type transformer to embed textual and cross-modal features for three reasons: 1) Unlike the visual encoder that usually incorporates pyramid structures (*e.g.*, Resnet [16] and PVT [41]), the text and cross-modal encoder are both BERT-type transformers, whose difference only lies in the number of transformer blocks; 2) The inputs of the text and cross-modal encoder are both token-type embeddings/features (\mathbf{W} and $[\mathbf{T}, \mathbf{v}]$). Besides, we hypothesize that the visual features extracted by visual encoders could be treated as high-level semantic tokens as text embeddings; And 3) it is difficult to find a suitable weight initialization for training the cross-modal encoder, which would decrease the pre-training performance as discussed in CLIP4Clip [30]. In this way, **SNP** is as lightweight as twin-tower-based models, and could support various downstream video-text tasks like three-fusion-based ones, achieving a satisfactory balance between pre-training efficiency and performance.

As illustrated in the left part of Figure 3, we denote the shared BERT-type encoder as E_{snp} , where textual features \mathbf{T} and cross-modal features \mathbf{M} can be calculated as:

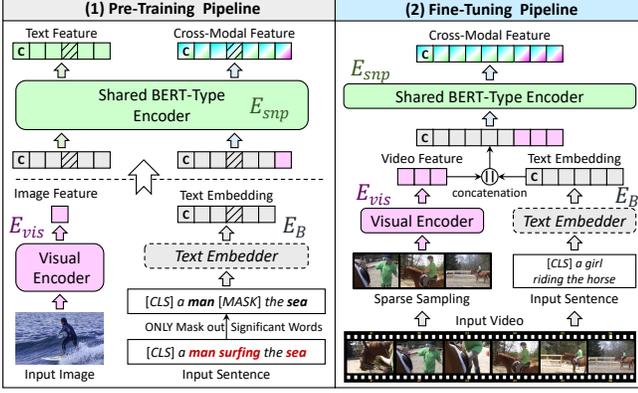


Figure 3. The framework of SNP-S^3 , which employs a shared BERT-type encoder to process textual and cross-modal features. Following the previous work, we 1) pre-train on image-text datasets, and 2) fine-tune on downstream video-text tasks. We also report the results pre-trained on video-text datasets in Section 4.3.

$$\begin{cases} \mathbf{T} = E_{snp}(E_B(S)), \\ \mathbf{M} = E_{snp}([(E_B(S)), E_{vis}(I)]). \end{cases} \quad (2)$$

3.2. Limitations of Conventional MLM and GVTM

Proxy tasks directly determine the pre-training objectives, where Masked Language Modeling (MLM) and Global Vision-Text Matching (GVTM) are two widely-employed ones. MLM first masks out a certain percentage of words in a given sentence, and then forces the model to restore these clozes according to visual and textual cues. However, some masking words like prepositions and conjunctions could be easily predicted only by grammar without reviewing the image, which may contribute little to the cross-modal interaction. GVTM aims to learn the cross-modal interaction from image features and global-pooling textual features. However, we believe that global features of a sentence and local information of some inner informative words are equally important, while GVTM only emphasizes the former but omits the latter.

Intuitively, not all words contribute equally to understanding a sentence. Specifically, some words like verbs and nouns are more significant as they provide rich information, while others only act as “lubricants” to make the pale description “somebody do something” more fluent and vivid. Based on this intuition, we aim to capture these informative words rather than trivial ones to promote the cross-modal interaction. Towards this end, we propose the Significant Semantic Strengthening (S^3) strategy, which includes a novel masking task (MSSM) and a matching one (LVWM) for better pre-training performance.

Algorithm 1: Offline Significant Semantic Mining.

Input: BERT vocabulary list L_{BERT} ,
All captions within datasets $\{Cap_i\}_{i=1}^{N_{data}}$,
Pre-defined num K^{ss} .
Output: Significant semantic vocabulary L_{spaCy} .

```

1 Set  $L_{POS} = [0] * \text{len}(L_{BERT})$ 
2 for  $i \leftarrow 1$  to  $N_{data}$  do
3    $T_{spaCy} = \text{spaCy.tokenize}(Cap_i)$ ;
4    $P_{spaCy} = \text{spaCy.POS}(Cap_i)$ ;
5   for  $j \leftarrow 1$  to  $\text{len}(T_{spaCy})$  do
6     if  $T_{spaCy}[j]$  in  $L_{BERT}$  then
7       if  $P_{spaCy}[j]$  in  $[Verb, Adjective, Noun]$ 
8         then
9            $label = \text{BERT.tolabel}(T_{spaCy}[j])$ ;
10           $L_{POS}[label] += 1$ ;
11        end
12      end
13    end
14 Set  $\text{Num}_K = \text{Get-Maximum-K}(L_{POS}, K^{ss})$ ;
15 Set  $L_{spaCy} = [0] * \text{len}(L_{BERT})$ ;
16 for  $i \leftarrow 1$  to  $\text{len}(L_{spaCy})$  do
17   if  $L_{POS}[i] \geq \text{Num}_K$  then
18      $L_{spaCy}[i] = 1$ ;
19   end
20 end
```

Algorithm 2: Online Significant Semantic Mining.

Input: BERT vocabulary list L_{BERT} ,
Significant semantic vocabulary L_{spaCy} ,
Caption Cap .
Output: Significant semantic chosen list L_{ss} .

```

1 Set  $L_{ss} = []$ ;
2  $T_{BERT} = \text{BERT.tokenize}(Cap)$ ;
3 for  $i \leftarrow 1$  to  $\text{len}(T_{BERT})$  do
4    $label = \text{BERT.tolabel}(T_{BERT}[i])$ ;
5   if  $L_{spaCy}[label] == 1$  then
6      $L_{ss} += [i]$ 
7   end
8 end
```

3.3. Significant Semantic Mining Algorithm

We simply define VERBs, NOUNs and ADJECTIVEs as significant semantics since they provide essential information for understanding a sentence. Then the question is how to distinguish these informative words from other trivial ones efficiently. We first attempt to select these significant words by an open-source NLP toolkit spaCy* during the

*Official Website of spaCy: <https://spacy.io/>

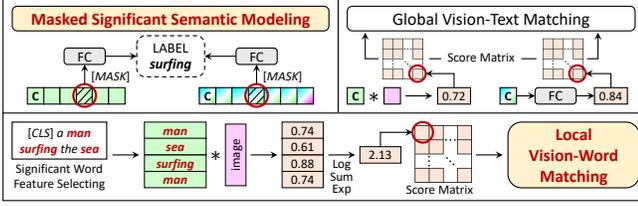


Figure 4. Three types of proxy tasks for pre-training the proposed **SNP-S³**. Notably, we propose two improved tasks marked in red (MSSM and LVWM) to facilitate the cross-modal interaction. Specifically, MSSM first masks out some significant informative words and forces models to restore these clozes, while LVWM learns the video-text alignment at the word level.

pre-training in an end-to-end manner. Unfortunately, it is infeasible for two reasons: 1) The inference time of spaCy is unaffordable, which would slow down the pre-training obviously. And 2) the dictionary of spaCy is quite different from BERT, whose results could not be processed into BERT-type token embeddings directly.

Towards this end, We design a direct and efficient mining algorithm with the open-source NLP toolkit spaCy to find those informative words, and organize them into the significant semantic chosen list L_{ss} . Specifically, we first maintain an offline significant semantic vocabulary L_{spaCy} according to the BERT vocabulary by employing spaCy to review all the captions within pre-training datasets, and then leverage this offline vocabulary L_{spaCy} to build the online significant semantic list L_{ss} according to the input sentence during the pre-training. The workflow is summarized in Algorithm 1 and Algorithm 2, where $\text{len}(\cdot)$ obtains the length of the given list, $\text{spaCy.tokenize}(\cdot)$ splits the given caption into tokens, and $\text{spaCy.POS}(\cdot)$ obtains the parts-of-speech tag of each word by employing spaCy; $\text{BERT.tolabel}(\cdot)$ translates the given BERT token into the corresponding label number and $\text{BERT.tokenize}(\cdot)$ tokenizes the given sentence according to the BERT vocabulary; $\text{Get-Maximum-K}(\cdot, K)$ aims to find the K^{th} maximum number in the given list. We choose the Top 2000 (K^{ss}) vocabs whose parts-of-speech tags are nouns, verbs, or adjectives according to their frequency to build the significant semantic vocabulary L^{spacy} . We then leverage it to obtain the significant semantic chosen list L_{ss} .

It takes about half a day to complete the offline significant semantic mining step on all pre-training corpus. Compared with the whole pre-training (usually three days), this time assumption is acceptable. Moreover, the purpose of offline mining step is to build the significant semantic vocabulary, which could be reused after the first building.

3.4. Overall Optimization Objectives

Figure 4 shows all proxy tasks we employed in our **SNP-S³**, including Masked Significant Semantic Modeling

(MSSM), Global Vision-Text Matching (GVTM), and Local Vision-Word Matching (LVWM).

MSSM is an improved version of MLM, which only masks out the informative words within the significant semantic chosen list L_{ss} . Note that all settings (e.g., the masking rate) except the chosen masked tokens remain the same as the conventional MLM protocol. We calculate MSSM twice for textual features \mathbf{T} (\mathcal{L}_1) and cross-modal features \mathbf{M} (\mathcal{L}_2), which can be formulated as follows:

$$\mathcal{L}_1 = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathcal{L}_{CE}(y^q, \mathbf{t}^q), \quad (3)$$

$$\mathcal{L}_2 = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathcal{L}_{CE}(y^q, \mathbf{m}^q), \quad (4)$$

where \mathcal{Q} denotes the masked token set, $|\cdot|$ denotes the length of a given set, y^q denotes the ground-truth token label, and \mathcal{L}_{CE} is the regular Cross-Entropy cost function.

GVTM aims to model the cross-modal interaction by employing visual features and global-pooling textual features (refer to the hidden state of the first $[cls]$ token). Following the conventional paradigm [51], we set two GVTM tasks to align visual and textual features in a parameter-free (\mathcal{L}_3) and parameter-employed (\mathcal{L}_4) way:

$$\mathcal{L}_3 = - \sum_{i=1}^{|\mathcal{B}|} \log \frac{\exp\langle \mathbf{v}_i, \mathbf{t}_i^{cls} \rangle}{\exp\langle \mathbf{v}_i, \mathbf{t}_i^{cls} \rangle + \sum_{j \neq i} \exp\langle \mathbf{v}_j, \mathbf{t}_i^{cls} \rangle}, \quad (5)$$

$$\mathcal{L}_4 = - \sum_{i=1}^{|\mathcal{B}|} \log \frac{\exp\Theta(\mathbf{m}_{i,i}^{cls})}{\exp\Theta(\mathbf{m}_{i,i}^{cls}) + \sum_{j \neq i} \exp\Theta(\mathbf{m}_{j,i}^{cls})}, \quad (6)$$

where $|\mathcal{B}|$ is the length of a mini-batch, $\langle \cdot, \cdot \rangle$ denotes the matrix multiplication operation, Θ is a Multi-Layer Perceptron (MLP), and $\mathbf{m}_{j,i}^{cls}$ is the global-pooling cross-modal features of the image-text pair (I_j, S_i) . Since $\mathbf{v}, \mathbf{t}^{cls} \in \mathbb{R}^{1*d}$, \mathcal{L}_3 counts similarity score matrices without introducing any parameters.

LVWM is a complementary to GVTM as it focuses on modeling several informative semantic features at the word level rather than the sentence level. We first build the local significant semantic feature set $\hat{\mathbf{T}} = \{\hat{\mathbf{t}}^i \mid \hat{\mathbf{t}}^i \in \mathbf{T}\}_{i=1}^{N_L}$ from textual features \mathbf{T} according to the list L_{ss} , which contains N_L significant token features by either random-sampling (if $\text{len}(L_{ss}) > N_L$) or over-sampling (if $\text{len}(L_{ss}) < N_L$). Then we calculate LVWM loss (\mathcal{L}_5) as follows:

$$\mathcal{L}_5 = - \sum_{i=1}^{|\mathcal{B}|} \log \frac{\sum_{l=1}^{N_L} \exp\langle \mathbf{v}_i, \hat{\mathbf{t}}_l^i \rangle}{\sum_{l=1}^{N_L} (\exp\langle \mathbf{v}_i, \hat{\mathbf{t}}_l^i \rangle + \sum_{j \neq i} \exp\langle \mathbf{v}_j, \hat{\mathbf{t}}_l^i \rangle)}. \quad (7)$$

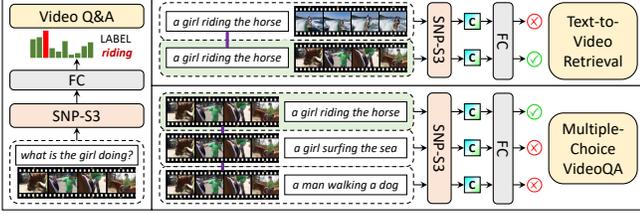


Figure 5. Details of fine-tuning the pre-trained model on three downstream video-text tasks.

Ultimately, the objective function of the proposed SNP-S^3 is the combination of Eqs. (3)-(7), which is defined as:

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3 + \mathcal{L}_4 + \mathcal{L}_5. \quad (8)$$

3.5. Pre-training on Video-Text Datasets

We pre-train our SNP-S^3 on large video-text datasets to pursue better performance. As a video could be treated as a group of images (frames) in time streams, we sparsely (and randomly) sample N_V frames from a given video (N_V is usually much smaller than the total number of frames of this video), and set their position embeddings to zero following CLIP4Clip [30]. Therefore, the video features $\mathbf{V} = [\mathbf{v}^1, \mathbf{v}^2, \dots, \mathbf{v}^{N_V}]$ could be processed by the visual encoder E_{vis} . The forward propagation step and optimization objectives (proxy tasks) remain the same as the protocol pre-trained on image-text datasets. *I.e.*, first replacing image features \mathbf{v}_i with video features $\mathbf{V} = \{\mathbf{v}_i^k\}_{k=1}^{N_V}$, and then repeating operations of Eqs. (2)-(8).

3.6. Fine-tuning on Downstream Video-Text Tasks

We fine-tune our model on three downstream video-text tasks to evaluate the performance of SNP-S^3 , the fine-tuning details are illustrated in Figure 5. Similar to the pre-training protocol in Section 3.5, we randomly sample several frames from raw videos to serve as visual inputs.

Text-to-Video Retrieval (TVR) aims to retrieve the most relevant video according to the input text query. We fine-tune our pre-trained model by reusing two Global Vision-Text Matching objectives \mathcal{L}_3 (Eq.5) and \mathcal{L}_4 (Eq.6).

Video Question Answering (VQA) aims to answer natural language questions according to the given videos. In the field of video understanding, VQA is essentially a classification task rather than a generation one. Thus, we add a classifier Θ_a on top of the global-pooling cross-modal features \mathbf{m}^{cls} in the last layer of the model. We fine-tune it by calculating the Cross-Entropy loss \mathcal{L}_{VQA} as follows:

$$\mathcal{L}_{VQA} = \mathcal{L}_{CE}(y_a, \Theta_a(\mathbf{m}^{cls})), \quad (9)$$

where y_a is the label of the ground-truth answer.

Multi-Choice Video Question Answering (MC-VQA) aims to align each video with one out of several candidate

answers. Currently, this task is only supported by MSR-VTT multi-choice test set [56] without available training data. We directly employ the best model in TVR fine-tuning and traverse this test set once during evaluation, which could be treated as a zero-shot classification task.

4. Experiments

4.1. Datasets

4.1.1 Pre-training Datasets

We pre-train our SNP-S^3 on three large-scale image-text datasets: **COCO** [26], **Visual Genome** [19] (VG), and **Conceptual Captions** [37] (CC), which contain more than 0.59M, 5.4M, and 3.1M image-text pairs, respectively.

Besides these image-text datasets, we also pre-train our SNP-S^3 on WebVid [3], a large-scale video-text dataset including 2.5M video-text pairs, to pursue better fine-tuning performance on three downstream tasks.

4.1.2 Fine-tuning Datasets

We fine-tune our pre-trained model on three downstream video-text tasks, including six corresponding datasets.

Text-to-Video Retrieval : 1) *MSR-VTT* [48] contains 10K video clips associated with 200K sentences. There are two widely-employed validation splits, one takes 7K videos for training and randomly selects 1K videos from the remaining ones for testing (7K-1K split), while the other uses 9K videos for training and the remaining 1K for testing (9K-1K split). In this paper, we report the fine-tuning results on these two splits. 2) *DiDeMo* [2] contains 10K Flickr videos associated with 40K sentences. 3) *MSVD* [4] contains 2K video clips associated with 80K descriptions.

Video-Question Answering: 4) *MSRVTT-QA* [46] is built upon MSR-VTT and contains 10K videos with 243K open-ended questions and 1.5K answer classes. 5) *MSVD-QA* [46] is built upon MSVD and contains 2K videos with 50K open-ended questions and 2.4K answer classes.

Multi-Choice Video Question Answering: 6) *MSR-VTT Multi-Choice Test Set* [56] contains 3K videos. Each video has five candidates with one correct answer.

Metrics: For TVR, we employ Recall@K (R@K) and Median Rank (M_dR) to measure the text-to-video retrieval performance. For VQA and MC-VQA, we employ Accuracy (Acc) to evaluate the answering correctness.

4.2. Experimental Settings

4.2.1 Pre-training Implementation Details

We employ three types of visual encoders, namely Resnet-50 (R50) [16], Pyramid Vision Transformer (PVT) [41], and Video Swin Transformer (VST) [28]. We initialize these three modules with the parameters pre-trained on ImageNet

Model	R@1/5/10 \uparrow	Model	R@1/5/10 \uparrow	Model	R@1/5/10 \uparrow
CE	9.9 / 29.0 / 41.2	Frozen (p)	25.5 / 54.5 / 66.1	CLIPBERT (p)	20.4 / 48.0 / 60.8
CLIPBERT (p)	22.0 / 46.8 / 59.9	VIOLET (p)	23.5 / 50.5 / 63.9	VIOLET (p)	22.8 / 51.2 / 62.0
HERO* (f)	16.8 / 43.4 / 57.7	SNP-S³-PVT	28.9 / 57.0 / 69.4	SNP-S³-PVT	26.6 / 57.3 / 69.1
CoCoBERT* (f)	22.0 / 48.3 / 61.6	MMT	24.6 / 54.0 / 67.1	CE	16.1 / 41.1 / 54.4
TACo* (f)	24.5 / 52.8 / 65.5	TACo* (f)	28.4 / 57.8 / 71.2	MCQ* (p)	37.0 / 62.2 / 73.9
VLM* (f)	28.1 / 55.5 / 67.4	Frozen* (p)	32.5 / 61.5 / 71.2	Frozen* (p)	31.0 / 59.8 / 72.4
SNP-S³-PVT	26.6 / 55.5 / 67.7	VIOLET* (p)	34.5 / 63.0 / 73.4	VIOLET* (p)	32.6 / 62.8 / 74.7
SNP-S³-VST*	31.5 / 61.3 / 73.2	SNP-S³-VST*	33.6 / 65.8 / 75.1	SNP-S³-VST*	34.2 / 64.2 / 75.9

(a) MSRVTT Retrieval (7K-1K split). (b) MSRVTT Retrieval (9K-1K split). (c) Didemo Retrieval.

Model	R@1/5/10 \uparrow	Model	Acc	Model	Acc	Model	Acc
CE	19.8 / 49.0 / 63.8	HCRN	35.6	DualVGR	39.0	JSFusion	83.4
HERO* (f)	19.2 / 47.4 / 61.8	CLIPBERT (p)	37.4	SSML* (f)	35.1	CLIPBERT (p)	88.2
SSML* (f)	20.3 / 49.0 / 63.3	SSML* (f)	35.1	CoMVT* (f)	42.6	MERLOT* (f)	90.9
CoCoBERT* (f)	21.3 / 50.0 / 63.6	JustAsk* (f)	41.5	JustAsk* (f)	46.3	VLM* (f)	91.6
Frozen* (p)	33.7 / 64.7 / 76.3	ALPRO* (p)	42.1	ALPRO* (p)	45.9	VIOLET* (p)	91.9
SNP-S³-PVT	33.1 / 64.5 / 73.7	SNP-S³-PVT	42.0	SNP-S³-PVT	46.2	SNP-S³-PVT	92.3
SNP-S³-VST*	35.1 / 70.3 / 80.9	SNP-S³-VST*	43.1	SNP-S³-VST*	47.1	SNP-S³-VST*	96.5

(d) MSVD Retrieval. (e) MSRVTT-QA. (f) MSVD-QA. (g) MSRVTT MC-VQA Set.

Table 1. Performance comparison of different methods on three downstream video-text tasks and six corresponding datasets. The superscript “*” denotes that the method is pre-trained on large-scale video datasets (*e.g.*, WebVid). p and f denote the methods belong to pixel-level pre-training and feature-level ones. The suffix “PVT” and “VST” represent the model whose visual encoder is based upon PVTv2-B2 (pre-trained on COCO+VG) and VideoSwin-B (pre-trained on CC+WebVid). Note that some methods only conduct experiments on a certain range of datasets. Thus, Baselines on different datasets may vary a lot.

No.	Model Name	Parameters Count	Losses	MSRVTT (7K-1K split)		MSVD	
				R@1/5/10 \uparrow (MdR \downarrow)	QA: Acc	R@1/5/10 \uparrow (MdR \downarrow)	QA: Acc
A1	P3E-R50	205.5M	MLM, GVTM	18.3 / 46.0 / 58.8 (7)	40.40	24.5 / 49.8 / 64.2 (6)	40.99
A2	SNP-R50	160.4M (-22.0%)		21.7 / 47.8 / 61.4 (6)	40.96	22.8 / 53.9 / 66.1 (5)	43.63
A3	P3E-PVT	206.2M		22.7 / 48.0 / 62.5 (6)	40.76	26.4 / 54.6 / 67.8 (4)	43.83
A4	SNP-PVT	161.1M (-21.9%)		25.0 / 52.3 / 64.1 (5)	41.44	28.2 / 58.7 / 70.8 (3)	44.71

Table 2. Ablation study of the conventional three-fusion-based pixel-level pre-training paradigm (P3E) and our proposed version (SNP) that includes a shared BERT-type encoder on two downstream video-text tasks (TVR and VQA) of two datasets (MSRVTT and MSVD). “R50” and “PVT” represent that the model utilizes Resnet-50 and PVTv2-B2 as the basic visual encoder.

[8]. For the shared BERT-type network that processes textual and cross-modal features, we utilize the BERT-Base version with 12 layers of transformer blocks. We initialize this BERT-type module with the parameters pre-trained on BookCorpus [61] and English Wikipedia. Note that **SNP-R50** and **SNP-PVT** are pre-trained on COCO+VG (5.6M image-text pairs), while **SNP-VST** is pre-trained on CC+WebVid (5.5M image/video-text pairs).

For hyper-parameters, we set the length of text tokens $N_t = 30$, and the dimension of the hidden state $d = 768$. For the proposed MSSM task, the masking rate is 15%. For the proposed LVMM task, we set the number of the chosen significant tokens $N_L = 3$. We sparsely sample 4 frames from raw videos when pre-training on video-text datasets.

We pre-train our **SNP-S³** by the Adam optimizer with a momentum of 0.9. The total pre-training stage lasts for 200,000 steps with a batch size of 128. The initial learning

rate is $5e-5$ and is decayed by a factor of 10 after 110,000 iterations. The whole pre-training takes about 3 days to complete on 8 NVIDIA V100 GPUs.

4.2.2 Fine-tuning Implementation Details

For all three downstream video-text tasks, the optimizer and hyper-parameters remain the same as the pre-training configuration. Following [21], we sparsely sample 16 frames from input videos for fine-tuning and testing. The total fine-tuning stage lasts for 15,000 steps. The batch size is set to 32. The initial learning rate is set to $1e-5$.

4.3. Performance Comparison

We compare our **SNP-S³** with various state-of-the-art baselines, including the following methods:

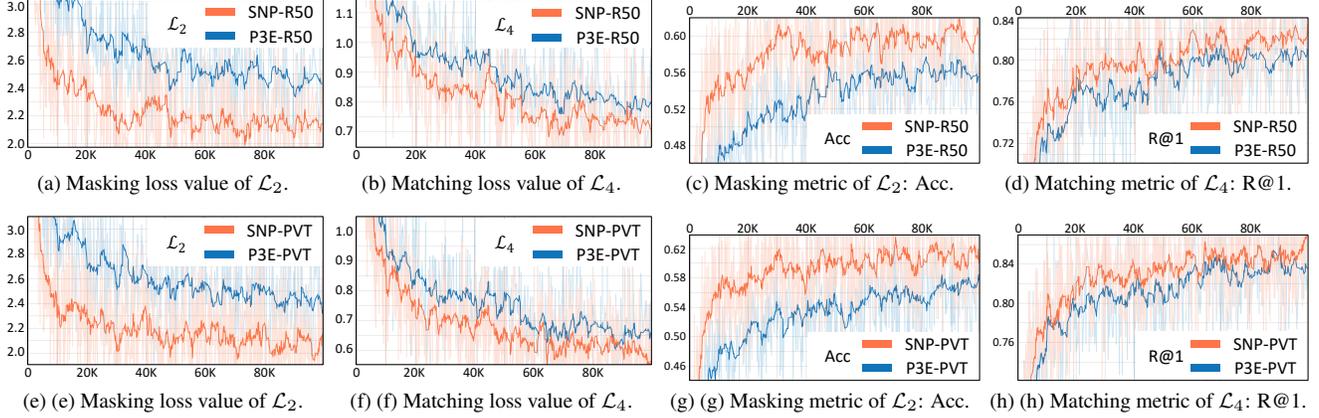


Figure 6. Visualization results of losses and evaluation performance towards masking (\mathcal{L}_2) and matching (\mathcal{L}_4) objectives in TensorBoard. The smoothing rate is 0.9, and we visualize the first 100K steps of the pre-training.

No.	Model Name	Masking Loss	Matching Loss(es)	MSRVTT (7K-1K split)		MSVD	
				R@1/5/10 (MdR)	QA: Acc	R@1/5/10 (MdR)	QA: Acc
B1		MLM	GVTM	21.7 / 47.8 / 61.4 (6)	40.96	22.8 / 53.9 / 66.1 (5)	43.63
B2	SNP	MSSM	GVTM	20.9 / 49.3 / 63.3 (6)	41.21	24.6 / 55.4 / 67.5 (4)	44.42
B3	R50	MLM	GVTM, LVWM	21.2 / 49.3 / 62.8 (6)	41.22	24.9 / 55.8 / 69.1 (4)	44.22
B4		MSSM	GVTM, LVWM	23.7 / 51.2 / 64.6 (5)	41.47	27.3 / 56.9 / 70.2 (4)	44.87
B5		MLM	GVTM	25.0 / 52.3 / 64.1 (5)	41.44	28.2 / 58.7 / 70.8 (3)	44.71
B6	SNP	MSSM	GVTM	27.2 / 53.2 / 65.7 (5)	41.74	31.0 / 59.0 / 71.8 (3)	45.69
B7	PVT	MLM	GVTM, LVWM	25.5 / 52.4 / 64.7 (5)	41.72	30.3 / 60.5 / 71.2 (3)	45.31
B8		MSSM	GVTM, LVWM	26.6 / 55.5 / 67.7 (4)	42.00	33.1 / 64.5 / 73.7 (3)	46.18
B9	SNP	MLM	GVTM	28.7 / 59.4 / 71.1 (4)	42.52	33.4 / 66.7 / 78.5 (3)	45.42
B10	VST	MSSM	GVTM, LVWM	31.5 / 61.3 / 73.2 (3)	43.09	35.1 / 70.3 / 80.9 (2)	47.15

Table 3. Ablation study of employing different combinations of masking and matching losses. The proposed S^3 strategy includes a novel masking loss (Masked Significant Semantic Modeling, MSSM) and a matching one (Local Vision-Word Matching, LVWM).

- Methods without pre-training: MMT [17], CE [27], HCRN [20], DualVGR [40], and JSFusion [56].
- Pixel-level pre-training methods: CLIPBERT (5.6M) [21], Frozen (5.5M) [3], VIOLET (185M) [13], MCQ (5.5M) [14], and ALPRO (5.5M) [23]. The number in “()” denotes the volume of video/image-text pairs within the pre-training corpus. We use 5.6M image-text pairs in our Resnet/PVT version and 5.5M image/video-text pairs in our VST version for a fair comparison.
- Feature-level pre-training methods: HERO [24], VLM [47], TACo [51], SSML [1], CoMVT [36], JustAsk [50], MERLOT [58], and CoCoBERT [31].

Table 1 presents detailed experimental results on three downstream video-text tasks (TVR, VQA, and MC-VQA) and six corresponding datasets. Note that some methods only conduct experiments on a certain range of datasets (e.g., Frozen reports its results on MSRVTT, Didemo, and MSVD in its paper). Thus, baselines on different

datasets may vary a lot. We report the results of our methods pre-trained on image-text datasets ($SNP-S^3-PVT$) and video-text datasets ($SNP-S^3-VST^*$). We have several observations as follows:

- $SNP-S^3-VST^*$ achieves the best performance among all pixel-level and feature-level video pre-training methods. We outperform Frozen [21] by 3.9%, 3.5%, and 4.6% at R@10 on the TVR tasks of MSRVTT (9K-1K), Didemo, and MSVD, respectively.
- $SNP-S^3-PVT$ outperforms other pixel-level pre-training methods pre-trained on image-text datasets by a large margin. On the MSRVTT dataset, we outperform current SOTA baselines by 7.8% (CLIPBERT) on R@10 of the 7K-1K split, 3.3% (Frozen) on R@10 of the 9K-1K split, 4.6% (CLIPBERT) on the Accuracy of VQA, and 4.1% (CLIPBERT) on the Accuracy of MC-VQA.
- $SNP-S^3-PVT$ that pre-trained on image-text datasets shows comparable performance with other methods

pre-trained on video-text datasets. *E.g.*, **SNP-S³-PVT** outperforms TACo at all metrics on the MSRVT 7K-1K split by more than 2.0%.

4.4. Ablation Study of SNP

As aforementioned, the proposed Shared Network Pre-training (**SNP**) combines the advantages of two mainstream pixel-level architectures, which is lightweight and could support various downstream video-text tasks.

Table 2 compares the fine-tuning performance on several downstream video-text tasks between the conventional three-fusion-based pixel-level paradigm (P3E) and our proposed version (SNP), while Figure 6 compares losses and evaluation performance during pre-training. Since the only difference between SNP and P3E is how to build its cross-modal encoder, where SNP shares the same parameters with the text encoder while P3E utilizes a separate one, so we only compare losses and metrics (\mathcal{L}_2 in Eq.4 and \mathcal{L}_4 in Eq.6) whose outputs are generated by the cross-modal encoder. Note that the cross-modal encoder in the P3E version is actually a three-layers BERT-base transformer blocks, whose setting follows UniVL [29]. Moreover, we find that the pre-training would hardly converge under the setting of employing the original BERT-Base encoder with 12 layers of blocks. As shown in Table 2 and Figure 6, we have several observations as follows:

- SNP is more **lightweight**. We count total trainable parameters in Table 2, both SNP-R50 and SNP-PVT reduce more than 20% of parameters compared with their P3E version (about 200M \rightarrow 160M), which verifies that SNP effectively simplifies the original model size.
- SNP is more **time-efficient** as it takes less time to reach a comparable performance. As illustrated in Figure 6, SNP needs fewer steps to achieve the same performance as P3E for both masking tasks and matching tasks.
- SNP is **easier to train and converge**. As illustrated in Figure 6, both masking and matching losses converge faster when equipped with SNP than with P3E. Besides, SNP avoids possible risks brought by an improper parameter initialization that P3E needs for training the separate cross-modal encoder. Moreover, as can be seen in Table 2 (*A1 v.s. A2, A3 v.s. A4*), SNP also shows better fine-tuning performance on several downstream video-text datasets, proving that SNP is more powerful than P3E.

For the above three observations, SNP is a lighter, faster, and stronger pre-training architecture than P3E.

Model	Chosen Num	Retrieval (7K-1K)	VideoQA
SNP-S ³ PVT	$N_L=1$	26.2 / 54.2 / 66.3	41.90
	$N_L=2$	25.5 / 53.2 / 66.8	41.76
	$N_L=3$	26.6 / 55.5 / 67.7	42.00
	$N_L=4$	26.4 / 54.7 / 67.5	41.94

Table 4. Parameter analysis of the number of the chosen significant token features (N_L). All experiments are evaluated on two tasks (Retrieval on 7K-1K split and VideoQA) of the MSRVT dataset under the backbone of SNP-S³-PVT.

4.5. Ablation Study of S³

As aforementioned, the proposed Significant Semantic Strengthening (**S³**) strategy is model-agnostic, parameter-free, and could evidently promote the fine-tuning performance. We conduct several ablation study to verify these advantages.

Our proposed **S³** strategy includes a novel masking loss (MSSM) and a matching one (LVWM). Table 3 presents the experimental results of employing different pre-training objectives. We have several observations as follows:

- Our proposed MSSM is a more **powerful** masking loss than conventional MLM. As shown in Table 3 (*B1 v.s. B2, B5 v.s. B6*), masking significant semantics rather than other trivial ones could force the model to predict these clozes according to textual and visual cues, which benefits the cross-modal interaction and further promotes the performance.
- Our proposed LVWM is an **effective complementary** matching loss to GVTM. As shown in Table 3 (*B1 v.s. B3, B5 v.s. B7*), both sentence-level global representations (the “[cls]” token) and word-level local information (token lists of some significant semantics) are beneficial for understanding a given sentence, and combining both of them could further facilitate the cross-modal interaction.
- Our proposed MSSM and LVWM are **model-agnostic**. As shown in Table 3 (*B1 v.s. B4, B5 v.s. B8, B9 v.s. B10*), both MSSM and LVWM are encoder-independent and could largely promote the fine-tuning performance.
- Notably, our proposed MSSM and LVWM are both **parameter-free** objectives. As shown in Eq.3, Eq.4, and Eq.7, both MSSM and LVWM do not introduce new parameters during computation. Therefore, they would not heavily slow down the speed of pre-training.

For the above four observations, we believe that **S³** is a model-agnostic and implementation-friendly strategy, which could efficiently promote the performance.

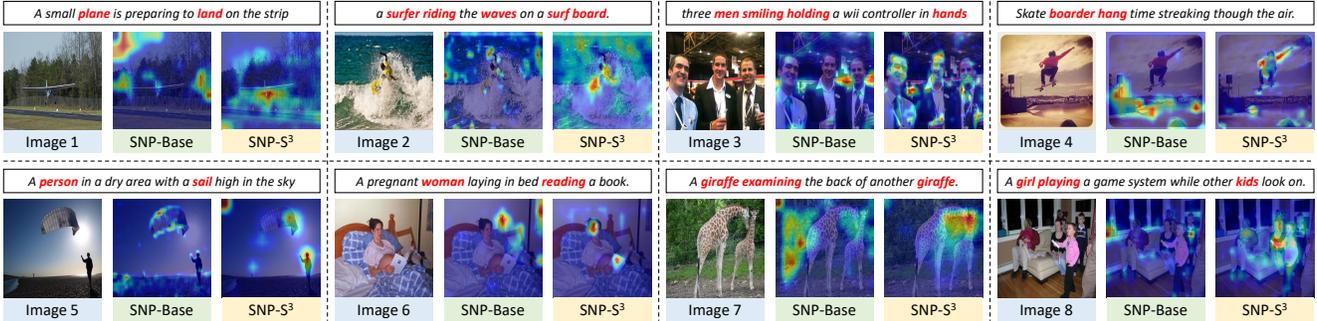


Figure 7. Qualitative analysis of the proposed Significant Semantic Strengthen (S^3) strategy between SNP-PVT and SNP- S^3 -PVT. We visualize the attention localization map of the last convolution layer in PVT by the toolkit Grad-CAM. We use red to mark the informative words emphasized by SNP- S^3 -PVT while omitted by SNP-PVT.

4.6. Parameter Analysis

We conduct the parameter analysis towards the number of chosen significant token features (N_L) of the LVWM loss. As shown in Table 4, the fine-tuning performance on the MSRVTT dataset first increases when adding more significant token features in computing the LVWM loss, and it would reach the peak at $N_L = 3$. It is probably due to the fact that the significant words (NOUNs, VERBs, and ADJECTIVEs) in one sentence are limited, so expanding the chosen number would have a performance upper bound.

4.7. Qualitative Analysis of S^3

To get an intuitive perception of the advantages of S^3 , we employ Grad-CAM [35], a widely-employed “visual explanation” toolkit, to visualize the attention location map of the last convolution layer in PVT. As shown in Figure 7, compared with SNP-PVT, the improved version SNP- S^3 -PVT tends to emphasize those informative words, thus the latter would better model the cross-modal interaction. *E.g.*, SNP-PVT wrongly lays its attention on the surroundings in Image 1, while SNP- S^3 -PVT correctly emphasizes the object “plane” and its action “land”. While in Image 2, SNP- S^3 -PVT successfully recognizes the scene “surfer-riding-waves”, while SNP-PVT fails to do so.

4.8. Performance Comparison of Related Methods

In Section 2, we have introduced some Significant Element Mining methods, including the Attended Masking (denoted as MLM-AM) strategy proposed by VIOLET [13] and the maximum token-level contrastive loss (denoted as TACo-L2) proposed by TACo [51]. To prove the superiority of our Significant Semantic Strengthening (S^3) strategy, we reproduce MLM-AM and TACo-L2 based on the SNP architecture. As illustrated in Table 5, for masking strategies, MSSM outperforms MLM-AM by 0.4/0.6/1.5 on R@1/5/10 of TVR and 0.30 on the Accuracy of VQA. While for matching strategies, LVWM outperforms TACo-L2 by 1.2/0.4/0.6 on R@1/5/10 of TVR and 0.47 on the

No.	Losses of SNP-PVT	TVR (7K-1K)	VQA
C1	MLM-AM, GVTM	26.8/52.6/64.2	41.44
C2	MSSM, GVTM	27.2/53.2/65.7	41.74
C3	MLM, GVTM, TACo-L2	24.3/52.0/64.1	41.25
C4	MLM, GVTM, LVWM	25.5/52.4/64.7	41.72

Table 5. Performance comparison of related Significant Elements Mining methods and our proposed Significant Semantic Strengthening (S^3) strategy. For masking tasks, we compare the Attended Masking (MLM-AM) in VIOLET [13] and our MSSM task. For matching tasks, we compare the maximum token-level contrastive loss (TACo-L2) in TACo [51] and our LVWM task. All the experiments are conducted under the SNP-PVT backbone and evaluated on the MSRVTT dataset.

Accuracy of VQA. One possible reason is that TACo-L2 only takes one token with the maximum similarity score into computation, which may also omit some local information compared with LVWM that employ multiple informative semantics to model the sentence at the word level. These comparisons prove that S^3 is an effective strategy to promote the pre-training performance.

5. Conclusion

In this paper, we improve conventional video-text pre-training methods from two aspects. For the pre-training architecture, we propose Shared Network Pre-training (SNP), a novel paradigm that effectively absorbs the advantages of two mainstream pixel-level models and overcomes their shortcomings. For pre-training proxy tasks, we propose the Significant Semantic Strengthening (S^3) strategy to optimize masking and matching tasks for better cross-modal interaction. In the future, we plan to employ a shared encoder to embed visual, textual, and cross-modal information from raw video data; and design a more robust Significant Semantic Mining algorithm to promote the cross-modal interaction.

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. 8
- [2] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5803–5812, 2017. 6
- [3] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1728–1738, 2021. 1, 2, 6, 8
- [4] David Chen and William Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 190–200, Portland, Oregon, USA, June 2011. 6
- [5] Xusong Chen, Chenyi Lei, Dong Liu, Guoxin Wang, Haihong Tang, Zheng-Jun Zha, and Houqiang Li. E-commerce storytelling recommendation using attentional domain-transfer network and adversarial pre-training. *IEEE Transactions on Multimedia*, 24:506–518, 2022. 1
- [6] Zhenfang Chen, Kexin Yi, Yunzhu Li, Mingyu Ding, Antonio Torralba, Joshua B Tenenbaum, and Chuang Gan. Comphy: Compositional physical reasoning of objects and events from videos. *arXiv preprint arXiv:2205.01089*, 2022. 1
- [7] Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. Pre-training with whole word masking for chinese bert. *IEEE Transactions on Audio, Speech, and Language Processing*, 29:3504–3514, 2021. 1
- [8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 7
- [9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2
- [10] Jianfeng Dong, Yabing Wang, Xianke Chen, Xiaoye Qu, Xirong Li, Yuan He, and Xun Wang. Reading-strategy inspired visual representation learning for text-to-video retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(8):5680–5694, 2022. 1
- [11] Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. Clip2video: Mastering video-text retrieval via image clip. *arXiv preprint arXiv:2106.11097*, 2021. 1, 2
- [12] Zerun Feng, Zhimin Zeng, Caili Guo, and Zheng Li. Temporal multimodal graph transformer with global-local alignment for video-text retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(3):1438–1453, 2023. 2
- [13] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. Violet: End-to-end video-language transformers with masked visual-token modeling. *arXiv preprint arXiv:2111.12681*, 2021. 1, 2, 3, 8, 10
- [14] Yuying Ge, Yixiao Ge, Xihui Liu, Dian Li, Ying Shan, Xiaohu Qie, and Ping Luo. Bridging video-text retrieval with multiple choice questions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 16167–16176, 2022. 1, 3, 8
- [15] Madeleine Grunde-McLaughlin, Ranjay Krishna, and Maneesh Agrawala. Agqa: A benchmark for compositional spatio-temporal reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11287–11297, 2021. 1
- [16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 3, 6
- [17] Gabriel Huang, Bo Pang, Zhenhai Zhu, Clara Rivera, and Radu Soricut. Multimodal pretraining for dense video captioning. *arXiv preprint arXiv:2011.11760*, 2020. 8
- [18] Zhicheng Huang, Zhaoyang Zeng, Bei Liu, Dongmei Fu, and Jianlong Fu. Pixel-bert: Aligning image pixels with text by deep multi-modal transformers. *arXiv preprint arXiv:2004.00849*, 2020. 2
- [19] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalanidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017. 6
- [20] Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. Hierarchical conditional relation networks for video question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9972–9981, 2020. 8
- [21] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7331–7341, 2021. 1, 2, 7, 8
- [22] Bowen Li, Weixia Zhang, Meng Tian, Guangtao Zhai, and Xianpei Wang. Blindly assess quality of in-the-wild videos via quality-aware pre-training and motion perception. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(9):5944–5958, 2022. 1
- [23] Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven CH Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4953–4963, 2022. 1, 2, 8
- [24] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. Hero: Hierarchical encoder for video+language omni-representation pre-training. *arXiv preprint arXiv:2005.00200*, 2020. 1, 2, 8

- [25] Yehao Li, Jiahao Fan, Yingwei Pan, Ting Yao, Weiyao Lin, and Tao Mei. Uni-eden: Universal encoder-decoder network by multi-granular vision-language pre-training. *ACM Trans. Multimedia Comput. Commun. Appl.*, 18(2), feb 2022. 1
- [26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, pages 740–755. Springer, 2014. 6
- [27] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. *arXiv preprint arXiv:1907.13487*, 2019. 8
- [28] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3202–3211, 2022. 6
- [29] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*, 2020. 2, 9
- [30] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860*, 2021. 2, 3, 6
- [31] Jianjie Luo, Yehao Li, Yingwei Pan, Ting Yao, Hongyang Chao, and Tao Mei. Coco-bert: Improving video-language pre-training with contrastive cross-modal matching and denoising. In *Proceedings of the ACM International Conference on Multimedia*, pages 5600–5608, 2021. 1, 8
- [32] Cheng Ma, Haowen Sun, Yongming Rao, Jie Zhou, and Jiwen Lu. Video saliency forecasting transformer. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(10):6850–6862, 2022. 2
- [33] Xin Man, Jie Shao, Feiyu Chen, Mingxing Zhang, and Heng Tao Shen. Tevl: Trilinear encoder for video-language representation learning. *ACM Trans. Multimedia Comput. Commun. Appl.*, feb 2023. Just Accepted. 1
- [34] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019. 2
- [35] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017. 10
- [36] Paul Hongsuck Seo, Arsha Nagrani, and Cordelia Schmid. Look before you speak: Visually contextualized utterances. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 16877–16887, 2021. 8
- [37] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 2556–2565, 2018. 6
- [38] Xue Song, Jingjing Chen, Zuxuan Wu, and Yu-Gang Jiang. Spatial-temporal graphs for cross-modal text2video retrieval. *IEEE Transactions on Multimedia*, 24:2914–2923, 2022. 1
- [39] Alex Jinpeng Wang, Yixiao Ge, Guanyu Cai, Rui Yan, Xudong Lin, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Object-aware video-language pre-training for retrieval. *arXiv preprint arXiv:2112.00656*, 2021. 2
- [40] Jianyu Wang, Bingkun Bao, and Changsheng Xu. Dualvgr: A dual-visual graph reasoning unit for video question answering. *IEEE Transactions on Multimedia*, 2021. 8
- [41] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 568–578, 2021. 3, 6
- [42] Bo Wu, Shoubin Yu, Zhenfang Chen, Joshua B Tenenbaum, and Chuang Gan. Star: A benchmark for situated reasoning in real-world videos. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. 1
- [43] Jun Wu, Tianliang Zhu, Jiahui Zhu, Tianyi Li, and Chunzhi Wang. A optimized bert for multimodal sentiment analysis. *ACM Trans. Multimedia Comput. Commun. Appl.*, 19(2s), feb 2023. 1
- [44] Yu-Chieh Wu and Jie-Chi Yang. A robust passage retrieval algorithm for video question answering. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(10):1411–1421, 2008. 1
- [45] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European Conference on Computer Vision*, pages 305–321, 2018. 2
- [46] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the ACM International Conference on Multimedia*, pages 1645–1653, 2017. 6
- [47] Hu Xu, Gargi Ghosh, Po-Yao Huang, Prahal Arora, Masoumeh Aminzadeh, Christoph Feichtenhofer, Florian Metze, and Luke Zettlemoyer. Vlm: Task-agnostic video-language model pre-training for video understanding. *arXiv preprint arXiv:2105.09996*, 2021. 1, 2, 8
- [48] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5288–5296, 2016. 6
- [49] Wanru Xu, Zhenjiang Miao, Jian Yu, Yi Tian, Lili Wan, and Qiang Ji. Bridging video and text: A two-step polishing transformer for video captioning. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(9):6293–6307, 2022. 2
- [50] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Just ask: Learning to answer questions from millions of narrated videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1686–1697, 2021. 8

- [51] Jianwei Yang, Yonatan Bisk, and Jianfeng Gao. Taco: Token-aware cascade contrastive learning for video-text alignment. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 11562–11572, 2021. [3](#), [5](#), [8](#), [10](#)
- [52] Xiaofeng Yang, Fengmao Lv, Fayao Liu, and Guosheng Lin. Self-training vision language bert with a unified conditional model. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2023. [2](#)
- [53] Xiaoshan Yang, Tianzhu Zhang, and Changsheng Xu. Text2video: An end-to-end learning framework for expressing text with videos. *IEEE Transactions on Multimedia*, 20(9):2360–2370, 2018. [1](#)
- [54] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. Clevrer: Collision events for video representation and reasoning. *arXiv preprint arXiv:1910.01442*, 2019. [1](#)
- [55] Ting Yu, Jun Yu, Zhou Yu, Qingming Huang, and Qi Tian. Long-term video question answering via multimodal hierarchical memory attentive networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(3):931–944, 2021. [1](#)
- [56] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European Conference on Computer Vision*, pages 471–487, 2018. [6](#), [8](#)
- [57] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. Merlot reserve: Neural script knowledge through vision and language and sound. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 16375–16387, 2022. [1](#)
- [58] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. Merlot: Multimodal neural script knowledge models. *Advances in Neural Information Processing Systems*, 34:23634–23651, 2021. [3](#), [8](#)
- [59] Jipeng Zhang, Jie Shao, Rui Cao, Lianli Gao, Xing Xu, and Heng Tao Shen. Action-centric relation transformer network for video question answering. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(1):63–74, 2022. [1](#)
- [60] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8746–8755, 2020. [2](#)
- [61] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 19–27, 2015. [7](#)