Verbs in Action: Improving verb understanding in video-language models

Liliane Momeni$^1$ Mathilde Caron$^2$ Arsha Nagrani$^2$ Andrew Zisserman$^1$ Cordelia Schmid$^2$

$^1$Visual Geometry Group, University of Oxford, UK $^2$Google Research

Abstract

Understanding verbs is crucial to modelling how people and objects interact with each other and the environment through space and time. Recently, state-of-the-art video-language models based on CLIP have been shown to have limited verb understanding and to rely extensively on nouns, restricting their performance in real-world video applications that require action and temporal understanding. In this work, we improve verb understanding for CLIP-based video-language models by proposing a new Verb-Focused Contrastive (VFC) framework. This consists of two main components: (1) leveraging pre-trained large language models (LLMs) to create hard negatives for cross-modal contrastive learning, together with a calibration strategy to balance the occurrence of concepts in positive and negative pairs; and (2) enforcing a fine-grained, verb phrase alignment loss. Our method achieves state-of-the-art results for zero-shot performance on three downstream tasks that focus on verb understanding, including video-text matching, video-question-answering and video classification; while maintaining performance on noun-focused settings. To the best of our knowledge, this is the first work which proposes a method to alleviate the verb understanding problem, and does not simply highlight it. Our code is publicly available at [16]: scenic/projects/verbs_in_action.

1. Introduction

Large-scale visual-language models (VLMs) such as CLIP [58] have shown strong performance on multiple video-language tasks such as text-to-video retrieval [44], video question-answering, and open-set action recognition [42]. These models perform surprisingly well on these tasks in a zero-shot setting, despite being trained only on image-language pairs (with no access to temporal data), even outperforming strong video-specific models [5, 87].

A recently highlighted and well-documented problem with such models, however, is their strong noun or object bias, as evidenced by their lower performance in distinguishing between verbs in natural language descriptions [31, 53, 93]. This was first studied in images alone by the SVO-Probes benchmark [31], which shows that image-language models struggle to distinguish between different verbs, and often rely on the nouns instead. This problem persists with video-language models that inherit these VLMs, even after they are fine-tuned on video-text datasets [62, 85]. For example, Park et al. [53] similarly propose evaluation sets with hard verb negatives, and show that CLIP-based models, even when fine-tuned on video datasets, have difficulties discriminating verbs in a multi-choice setting where the context remains unchanged. Yükselkogon et al. [93] further highlight limitations of vision-language models at understanding attribute, relationship,
and order information. This deficiency in verb understanding limits the model’s applicability for real-world tasks. Verbs encapsulate how people and objects interact with each other, and the environment, via actions in space and time.

We believe that there are two probable causes for this deficiency, even after fine-tuning on video-text data: (i) existing visual-text datasets have a strong bias towards single-frame concepts such as objects and backgrounds as well as static actions [9, 37, 67]. Models are hence less incentivized to understand dynamics and temporal actions [67], biasing them towards noun understanding; and (ii) the limitations of the cross-modal contrastive pretraining objective used by most current vision-language models [93]. In contrastive learning, the model is trained to distinguish correct video-caption pairs from incorrect ones. Since it is unlikely that existing datasets contain many examples with captions of similar context but different verbs, the task can be solved by taking little verb information into account. This relates to shortcut learning in deep neural networks [27].

In an attempt to mitigate this problem, we propose a novel training framework for tackling the task of verb understanding in vision-language models. Our framework, called Verb-Focused Contrastive pretraining (VFC), consists of two novel technical modifications to the contrastive learning framework. We first introduce a method to automatically generate negative sentences for training where only the verb has changed, keeping the context the same. This is done using LLMs [23, 59], in an automatic and scalable manner. Note that we generate hard negative captions, unlike works that simply mine hard negatives from an existing paired dataset [57], or change the order of words [93]. For example, given the caption ‘two brown horses eating grass’, we generate the negative caption ‘two brown horses running on the grass’ (see Fig. 1). While this improves performance on some downstream tasks, we find that introducing concepts simply in negative examples can also lead to an imbalance in the contrastive objective, favouring certain concepts in the feature space. To solve this, we propose a simple but effective calibration strategy to balance the occurrence of verbs in both positive and negative captions.

Secondly, inspired by recent works on grounding concepts in vision-language learning [10, 35], we also introduce a verb phrase loss that explicitly isolates the verb from a caption for more focused training. For example, we extract the verb phrase ‘eating grass’ from the caption ‘two brown horses eating grass’ (see Fig. 1). We find that this helps particularly for zero-shot performance on downstream tasks that do not use long sentences in their evaluation [28]. Verb phrases are also extracted from sentences using LLMs.

We then train a CLIP-based model [44] on a video-language dataset with this novel training framework. We show that a single model trained in this way transfers well to diverse downstream tasks that focus particularly on verb understanding, including three video benchmarks (multiple choice video-text matching on MSR-VTT [85], video question answering on Next-QA [82], action recognition on Kinetics [11]) and one image benchmark (SVO-probes [31]), achieving state-of-the-art performance compared to previous works in zero-shot settings (and often with fine-tuning as well); while maintaining performance on noun-focused settings. On Kinetics, we also introduce a verb split of the data which specifically highlights classes that are challenging to distinguish without fine-grained verb understanding (‘brushing hair’ vs ‘curling hair’) and show that our model particularly improves performance on this split.

2. Related works

LLMs for video-text tasks. LLMs have been used for various vision applications, for example to initialise vision-text models [12, 45, 66]. Recent works further use frozen LLMs via prompting for tackling vision-language tasks [3, 24, 70, 76, 89, 91, 95]. LLMs have also been used in creative ways to obtain better supervision for training for various tasks [41, 64, 88, 94, 97]. For example, [88] use LLMs to generate question-answer pairs from transcribed video narrations, while [94] use LLMs to rephrase questions into sentences. [41] use LLMs to match noisy speech transcriptions to step descriptions of procedural activities. [51] train BERT [17] to predict action labels from transcribed speech segments and use this to scale up training data for action classification. [97] use pretrained LLMs conditioned on video to create automatic narrations. Recent works [64, 97] also show the benefits of using LLMs to paraphrase captions for data augmentation for video-language pretraining. [39] use LLMs to generate negative captions by manipulating event structures. Our work differs to [39] in that we focus specifically on verb negatives, and videos instead of images. Most closely related to our work, [53] construct a test set for verb understanding by leveraging T5 [59] and highlight the poor performance of current video-language models. Our work is substantially different: (i) we automatically construct hard negative captions for training (not testing), (ii) we compare the use of different LLMs, (iii) we show that training with such negative captions can improve verb understanding on various verb-focused benchmarks.

Hard negatives for contrastive pretraining. Hard negatives have been used to improve performance in metric representation learning and contrastive learning [30, 34, 80]. Recent works mine hard negatives from an existing paired dataset [57, 84, 90]. In comparison, in our work, we generate hard negative captions and propose a careful calibration mechanism for training effectively with such unpaired data. We also verify here the benefit of the HardNeg-NCE loss [57] when training with generated hard negative captions. [93] construct hard negative captions by shuffling words from the original caption to improve order and com-
positionality understanding. Our work differs by (i) focusing specifically on verb reasoning, as opposed to object-attribute relationships, (ii) using LLMs to construct hard verb text negatives as opposed to perturbing the word order, (iii) focusing on video-language models.

Learning from parts-of-speech in video. Recent works use parts-of-speech (PoS) tags for video understanding [25, 28, 63, 79, 86]. [79] learn multi-label verb-only representations, while other works focus on learning adverb representations [21, 22]. [2] use verb-noun pairs for unsupervised learning with instructional videos, while [25] leverage such pairs to generate data augmentations in the feature space. Other works exploit PoS for fine-grained or hierarchical alignment between video and text [14, 96]. [78] learn a separate multi-modal embedding space for each PoS tag and then combine these embeddings for fine-grained action retrieval. [14] construct a hierarchical semantic graph and use graph reasoning for local-global alignments. Most closely related to our work, [90] use a PoS based token contrastive loss. Our work differs in that: (i) we apply a verb phrase contrastive loss, as opposed to separate verb and noun losses; (ii) we extract verb phrases using a LLM and show this performs better than PoS tagging with NLTK [8] (Tab. 5); (iii) we evaluate our methods on verb-focused downstream tasks. Similarly to [28], we find that training with verb phrase supervision helps for zero-shot performance on tasks with shorter sentences.

Temporal understanding in videos. A long term goal in computer vision is temporal understanding in videos [11, 18, 29, 65, 68, 81, 98]. However, current training and test datasets have a strong visual bias towards objects and backgrounds as well as static actions [32, 67], with some works [9, 37] demonstrating strong results with a single frame. Despite these challenges, many recent works in video-only self-supervised learning propose pretext tasks for improving temporal modelling [1, 6, 7, 15, 19, 36, 40, 47, 54, 56, 60, 72, 73, 77, 92]. Unlike these works that use only video, [10, 69] focus on fine-grained temporal video-text alignment via localization of text sub-tokens. [4] also leverage before/after relations in captions to create artificial training samples for video-text. Differently to these works (which create augmented video negatives or positives), we approach the problem of improving verb understanding in video-language models from the language side, by leveraging the strong generalization capabilities of LLMs.

3. Method

Our goal is to adapt large-scale vision-language pretrained models (such as CLIP) to understand verbs. We aim to do this without requiring such models to be retrained from scratch, but by simply fine-tuning them on a video-language dataset. However, given the pitfalls with using the standard video-text contrastive setup [58] on existing video-language datasets, we propose a new framework which we call Verb-Focused Contrastive pretraining (VFC). It consists of two components, both using the power of LLMs: (i) a novel calibrated hard negative training method where we train with synthetic verb-focused hard negative captions, and (ii) an additional verb phrase loss where videos are contrasted against isolated verb phrases as opposed to the entire caption. Note that a ‘verb phrase’ can be a single verb or verb-noun pair depending on the caption (see Fig. 1).

3.1. Preliminaries

Large Language Models (LLMs) are generative text models with impressive capacities, in particular for few-shot or prompt-based learning [23]. In our work, we design prompts to instruct a LLM to (i) create verb-focused hard negative captions and (ii) isolate verb phrases from the captions of a dataset. LLMs allow scalability and generalisation, and as we show in the ablations (see Tab. 2 and Tab. 5), are preferable to manual or rule based methods (eg. NLTK [8]). In particular, we use PaLM [23], a state-of-art autoregressive model, throughout this paper. However, our framework is agnostic to this choice and other LLMs can be used instead (see Tab. 2).

Video-language contrastive pretraining works by learning to distinguish between aligned and non-aligned video-text pairs. Given a dataset of $N$ pairs $\{(V_i, T_i)\}_{i \in N}$ with video $V_i$ and caption text $T_i$, we extract normalised feature representations $v_i$ and $t_i$ by using a video encoder $f$ and text encoder $g$: we have $v_i = f(V_i)$ and $t_i = g(T_i)$. We use the InfoNCE loss [71] to make aligned (‘positive’) pairs close in feature space and all other pairwise combinations in the batch further apart [58]. We optimize for video-to-text $L^{v2t}$ and text-to-video $L^{t2v}$ alignments:

$$L^{t2v}_i = -t_i^T v_i / \sigma + \log \sum_{j=1}^B \exp(t_i^T v_j / \sigma) \quad (1)$$

where $B$ is the batch size and $\sigma$ a temperature parameter controlling the sharpness of the distribution. $L^{v2t}$ is obtained by inverting $v$ and $t$ in Eq. 1.

Architecture: adapting image-text models to videos. We leverage CLIP [58] for video-language tasks following the CLIP4CLIP ‘seqTrans’ protocol [44]. Both single-modal encoders (video $f$ and text $g$) are initialized with CLIP weights, with four additional temporal frame aggregation transformer blocks stacked on top of the image encoder (see Sec. C.2 of the appendix for more details). Our approach is agnostic to model architecture and so any state-of-the-art video-language architecture could be potentially used.

3.2. Verb-Focused Contrastive Pretraining (VFC)

We describe both our calibrated hard negative training (Sec. 3.2.1) and the proposed verb phrase loss (Sec. 3.2.2).
In regular contrastive learning, given a video-caption pair, other captions in the batch are simply pushed further in the feature space. Since it is unlikely that existing datasets contain many examples with captions of similar context but different verbs, the task can be solved by paying little attention to verbs. Instead, our goal is to encourage the video-language model to focus on verb reasoning. We do so by tasking a LLM to generate hard negative captions where only the verb(s) change. Second, we train with these additional negative captions. We find that naive training with additional data leads to imbalances affecting the resulting video-text feature space. We propose a simple but effective calibration mechanism to solve this.

### 3.2.1 Calibrated Hard Negative training

In regular contrastive learning, given a video-caption pair, other captions in the batch are simply pushed further in the feature space. Since it is unlikely that existing datasets contain many examples with captions of similar context but different verbs, the task can be solved by paying little attention to verbs. Instead, our goal is to encourage the video-language model to focus on verb reasoning. We do so by tasking a LLM to generate hard negative captions where only the verb(s) change. Second, we train with these additional negative captions. We find that naive training with additional data leads to imbalances affecting the resulting video-text feature space. We propose a simple but effective calibration mechanism to solve this.

#### Generating verb-focused hard negatives with PaLM.

Given a caption $T_i$, we task PaLM to replace the verbs with other verbs that convey a different action, but still form a linguistically and semantically viable sentence (which may not be guaranteed with random verb replacements – see qualitative examples in Sec. B.4 of the appendix). For example, in the caption ‘a man washes his face’, the verb ‘washes’ should not be replaced with ‘jumps’ or ‘plays’.

The generated caption is then a negative match for the corresponding video $V_i$ (albeit a hard negative, as the nouns and context remain the same). We experiment with different handcrafted prompts, and find our best performing prompt to be the following: *In this task, you are given an input sentence. Your job is to tell me 10 output sentences with a different meaning by only changing the action verbs*. We also add four input-output pair examples to the prompt, which increases the quality of PaLM’s predictions (see Sec. A.3.2 of the appendix). We use one PaLM forward pass per caption $T_i$ to generate ten verb-focused hard negatives for that caption (qualitative examples of the generated captions can be seen in Fig. 2). During training, we randomly sample $N^\text{hard}$ generated captions for each pair $(V_i, T_j)$ in the mini-batch, which we denote $(T^\text{hard}_k)^{k \in [1, N^\text{hard}]}$. Importantly, note that a $T^\text{hard}_k$ is a new generated text caption, or an unpaired data sample, meaning that it does not come with a corresponding matching (‘positive’) video.

#### Calibration.

Interestingly, we observe that naively adding in negative captions into training with a contrastive loss leads to harmful feature space distortions, as some concepts are only seen in negative captions but never in positives. This is observed by careful analysis of downstream performance (see study in Tab. 3 and Tab. 4). We hence next describe a calibration mechanism to avoid such distortions: we first denote the vocabulary of all verb phrases in the original and generated captions as $\Omega$. For each verb phrase $\omega$ (or ‘concept’) in $\Omega$, we use $S_\omega$ to represent the number of times it appears in the captions of the original dataset and $G_\omega$ for the number of times it appears in the PaLM-generated captions. We then derive equations for $R_\omega$ (see Tab. 1), which we define as the ratio of the number of times a verb phrase $\omega$ is used as a negative versus as a positive during training, for different choices of the video-to-text contrastive loss (note $L_T^{(\omega)}$ is unchanged).

#### Contrastive training with paired data (Baseline).

We first note that the ratio $R_\omega$ is independent of the verb phrase $\omega$ in regular contrastive learning (paired data only). It simply depends on the batch size $B$, as $S_\omega$ is cancelled from both the numerator and denominator. This means that the number of times a concept is used as a positive versus negative sample is the same regardless of the considered verb phrase. This naturally balances training, and is a great property of the contrastive framework.

#### Adding generated unpaired negative captions (HN).

However, when training with unpaired captions, this ratio is proportional to $G_\omega/S_\omega$ and therefore becomes dependent on the considered verb phrase $\omega$. This can have significant consequences for the video-text feature representations. The model can learn to either ignore or always predict
some concepts based on the average concept occurrences in positive or negative pairs during training.

**Hard negatives with calibration (Calibrated HN).** In order to make $R_{\omega}$ as $\omega$-agnostic as possible, we introduce an ensemble of two techniques which we refer to as ‘calibration’. First, we ignore the hard negative captions from the other elements of the batch (see row 3 in Tab. 1), which allows us to mitigate the influence of $G_{\omega}/S_{\omega}$ by not amplifying it by the batch size $B$ (equal to 256). Second, we filter the generated PaLM captions to have $G_{\omega} \approx S_{\omega}$. In practice, we discard some generations so that the number of times a verb phrase appears in the set of kept generations is equal to the number of times it is originally present in the dataset. We denote our video-to-text loss (text-to-video is unchanged) as $L_{\text{VFC}}$ for calibrated hard negative training.

**Video mining.** An alternative to avoid imbalances due to the addition of negative captions would be to avoid training with unpaired data at all, by mining a matching video $V_{i_k}^{\text{hard}}$ for each generated caption $T_i^{\text{hard}}$. We attempt this via CLIP-based text-to-video retrieval in a large video database but found that finding a video matching a detailed, long caption is challenging, as such a precise video may not exist in a given corpus (see Sec. A.3.1 in the appendix for examples).

### 3.2.2 The verb phrase loss

In order to further encourage our model to focus on verbs, we introduce a contrastive ‘verb phrase’ loss. We use PaLM to extract the verb phrase $T_i^{\text{verb}}$ in a caption $T_i$ with the following prompt: ‘**In this task, you are given an input sentence. Your job is to output the action verb phrases.**’ While multiple parts-of-speech (PoS) tagging tools exist, we use a LLM for the following reasons: (i) we would like to isolate verb phrases, which may correspond to single verbs or verb-noun pairs depending on the caption, (ii) LLMs deal better with ambiguous cases (see qualitative examples in Sec. B.5 of the appendix). We show the benefits experimentally via an ablation in Tab. 5. During training, we minimize the following loss:

$$L_i^{\text{verb-phrase}} = -v_i^T t_i^{\text{verb}} / \sigma + \sum_{j=1}^B \exp(v_i^T t_j^{\text{verb}} / \sigma)$$

where the negative verb phrase representations $t_j^{\text{verb}}$ simply come from other captions in the batch. Note that we do not require the calibration mechanism described in Section 3.2.1 since all verb phrases $T_i^{\text{verb}}$ have a positive video match $V_i$ (i.e. the video aligned with $T_i$).

Overall, our verb-focused contrastive (VFC) pretraining optimizes the sum of three objectives:

$$L_{\text{VFC}} = \frac{1}{B} \sum_{i=1}^B \left( \lambda_1 L_i^{2r} + \lambda_2 L_i^{\text{CHN}} + \lambda_3 L_i^{\text{verb-phrase}} \right)$$

with parameters $\lambda_1$, $\lambda_2$ and $\lambda_3$ weighting the contribution of the different terms. We learn the parameters of $f$ and $g$ via back-propagation.

### 3.3 Implementation details

**Spoken Moments in Time (SMiT) pretraining dataset.** The SMiT [49] training set consists of 481K pairs of 3-seconds video clips with corresponding captions. It is a subset of Moments in Time (MiT) [48]. Our work falls under the umbrella of transfer learning: we pretrain on SMiT and then use the resulting features to solve different downstream tasks in a zero-shot or fine-tuned manner. Pretraining is either done as in regular contrastive learning (‘baseline’) or with our VFC framework. We find that the baseline already performs competitively on our benchmarks, despite the relatively small size of SMiT compared to other datasets such as HowTo100M [46], due to the quality and diversity of the manually annotated captions. We encourage the community to consider SMiT as a powerful pretraining dataset.

**PaLM.** We use PaLM-540B [23] with beam size 4, output sequence length 512, and temperature of 0.7. The negative captions are generated in an autogressive way and are therefore of arbitrary length. We post-process them by removing text after any newline character and by filtering out candidates which contain the same verbs as the original caption.

**Training details.** Most hyper-parameters follow CLIP4CLIP [44]. We initialise our model with CLIP ViT/B-32 and train with VFC for 100 epochs with a batch size of 256, base learning rate of 1e-7, weight decay of 1e-2, temperature of 5e-3 and weights $\lambda_1 = 2$, $\lambda_2 = 1$, $\lambda_3 = 1$ which we empirically find to work well in our experiments. Indeed, this balances the video-to-text and text-to-video loss terms. We also normalise each loss term by its value obtained from a random uniform prediction in order to have all loss terms in the same range (loss always equal to 1 for a random uniform prediction). We sample 32 frames
per video at 25fps, with a 2 frame stride. See Sec. C in the appendix for further implementation details and extensive evaluation protocols.

4. Experiments

We curate a suite of benchmarks from existing works to evaluate verb understanding which we present in Sec. 4.1. Then we ablate various components of our VFC framework in Sec. 4.2. Finally, we demonstrate improved performance on our diverse set of downstream tasks in Sec. 4.3, and compare to the state of the art.

4.1. Verb-Focused Benchmarks

MSR-VTT multiple choice (MC) is a benchmark of 10K videos of length 10−30 secs. We evaluate on the standard 3k split and on VerbH from [53]. In this setting, the task is to associate each video to the right caption among five choices. While the four wrong captions are randomly chosen from other videos in the standard 3k split, one of them is replaced by a hard verb negative in VerbH [53].

Video question answering on NEXT-QA The train (resp. val) split contains 3870 (resp. 570) videos with 32K (resp. 5k) questions. There are three types of questions: causal (C), temporal (T) and descriptive (D). We consider the standard setting as well as ATPhard [9], a subset automatically constructed with questions that are non-trivially solved with a single frame. ATPhard is designed to be a better benchmark for the model’s true causal and temporal understanding which we believe is strongly related to verb reasoning.

Kinetics-400 is a video classification dataset with 400 human action classes. We report top-1, top-5 and their average classification accuracy. We follow [58] to evaluate classification in an open-set, zero-shot manner. This benchmark allows to assess transfer ability to action classification, which requires strong verb understanding (given actions are usually described with verb phrases).

SVO-probes dataset is a benchmark specifically designed to measure progress in verb understanding of image-text models [31]. It contains image-caption pairs with 421 different verbs. We simply replicate the image multiple times unless otherwise specified.

4.2. Ablation Study

In this section, we analyze our different design choices. We report results when transferring the models on two of our benchmarks: MSR-VTT multi-choice verb split (‘VerbH’) and Kinetics-400 video classification (‘K-400’). We chose these two benchmarks as they have very different properties: the first involves captions, while the second involves action labels. We note that Nhard = 1 for all ablations unless otherwise specified.

Hard negative captions generation. In Tab. 2, we ablate the technique used to obtain additional negative captions: we compare two LLMs (T5 [59] and PaLM [23]) and two non LLM-based methods: (i) ‘random verb’: we replace verbs by random verbs from the UPenn XTag\footnote{https://www.cis.upenn.edu/xtag/} verb corpus and (ii) ‘antonym verb’: we replace verbs with their antonyms, using the NLTK [8] package. We see in Tab. 2
that ‘random verb’ and ‘antonym verb’ already give moderate performance gains on VerbH compared to the baseline. However, using LLM-based generations improves the results by a large margin compared to the non LLM-based methods. This is likely due to the fact that (i) random or antonym replacements often create non semantically or linguistically plausible negative captions; (ii) some verbs do not have antonyms in NLTK (see qualitative examples in Sec. B.4 of the appendix). Finally, we see in Tab. 2 that T5 generations work very well in our framework too, which demonstrates that our framework is LLM-agnostic and can be extended to other LLMs. We observe that the best performance is achieved using PaLM, with a substantial gain over the baseline on MSR multi-choice (+8.1%) and a moderate gain on Kinetics (+0.2%).

**Hard negative captions: the importance of calibration.** We demonstrate the effect of the calibration mechanism described in Section 3.2.1 for training with unpaired captions. Tab. 3 shows the performance of hard negative training with ('w/') versus without ('w/o') calibration. First, we observe that the performance boost on MSR-VTT compared to the baseline is slightly stronger without calibration than with calibration. We believe this is because calibrating the PaLM generations reduces their number. However, we see that training with hard negatives without calibration deteriorates a lot the performance on Kinetics (−2.0% compared to the baseline). We hypothesize that this is due to some verb phrases being seen only as repulsive in the video-text feature space, while others are seen equally as attractive and repulsive. We illustrate this in Tab. 4 by showing the confusion matrix for a subset of the Kinetics classes, along with the ratio $R_\omega$ (defined in Sec. 3.2.1) for each verb phrase. Intuitively, $R_\omega$ measures the ‘attraction’ (if low) and ‘repulsion’ (if high) of a verb phrase $\omega$. The confusion matrix in Tab. 4 shows that the verb phrase ‘brushing hair’ becomes an attraction point in the absence of calibration. Indeed, the number of times the verb phrase ‘brushing hair’ is repulsive versus attractive is low ($R_{\text{brushing hair}} \approx 12$) compared to the other concepts such as for example ‘curling hair’ ($R_{\text{curling hair}} \approx 78$): we have $R_{\text{brushing hair}} << R_{\text{curling hair}}$. Hence, predictions for ‘brushing hair’ become dominant. This actually improves the performance for that class but deteriorates the performance on all the other classes related to ‘hair’. We see in Tab. 4 that our calibration mechanism alleviates this effect by making the ratio $R_\omega$ independent of $\omega$ as in regular contrastive learning. Calibration allows us to improve performance over the baselines on both tasks with a single model.

**Generating positive versus negative captions.** In Tab. 5 (left), we investigate the impact of generating positive captions instead of negatives with PaLM. In this case, positives correspond to sentences where the verb in the original caption is changed to a synonym verb, but the remaining context is unchanged: PaLM therefore acts as a data augmentation generator for text (similar to [64, 97]). Details about the positive caption generation implementation are in Sec. C.5 of the appendix. We observe that using positive captions has a negative impact on the performance in our benchmarks, possibly because with positive captions the model becomes more invariant to different verbs.

**Verb phrase loss.** In Tab. 5 (right), we explore two alternatives for verb phrase extraction used in the verb phrase loss: (i) using human-annotated action labels for clips from the Moments in Time (MiT) dataset (these are available as MiT data inherits from MiT [48]) and (ii) using a rule-based method (NLTK [8]) to isolate verbs. We observe in Tab. 5 that using PaLM to extract verb phrases from the caption outperforms both, probably because it extracts more fine-grained action information. Qualitative analysis of the verb phrases is shown in Sec. B.5 of the appendix.

**Combining calibrated hard negatives and verb phrase loss.** We show in Tab. 6 the complementarity between our two contributions: the calibrated hard negative training and the verb phrase loss. The former greatly improves performance on tasks requiring complex language understanding such as MC VerbH. On the other hand, the verb phrase loss improves transfer to video classification by focusing particularly on the action label in the sentence. We see in Tab. 6 that combining both approaches during training results in a single model with excellent performance on both MSR-VTT MC and Kinetics zero-shot transfer. Indeed, compared to the baseline, VFC pretraining achieves 9.2% relative improvement on MSR-VTT MC and 5.2% relative improvement on Kinetics.

**Number of hard negative captions.** In Tab. 7, we experiment with increasing the maximum number of hard negative captions $N_{\text{hard}}$ sampled per video in the batch. We find that setting this to 5 increases the performance on VerbH while...
sess exactly the task we are trying to solve). We even out- focused pretraining transfers well to the MSR-VTT multi- evaluation protocols.

We observe that increasing the maximum number of hard negative captions sampled per video increases the performance on Verb$_H$. We use $N_{\text{hard}} = 5$ in the remaining of the paper.

<table>
<thead>
<tr>
<th>Method</th>
<th>$N_{\text{hard}}$</th>
<th>Verb$_H$</th>
<th>K-400</th>
</tr>
</thead>
<tbody>
<tr>
<td>VFC (Ours)</td>
<td>1</td>
<td>76.3</td>
<td>58.5</td>
</tr>
<tr>
<td>VFC (Ours)</td>
<td>3</td>
<td>77.8</td>
<td>58.5</td>
</tr>
<tr>
<td>VFC (Ours)</td>
<td>5</td>
<td>78.3</td>
<td>58.5</td>
</tr>
</tbody>
</table>

Table 7. Maximum number of hard negative captions. We observe that increasing the maximum number of hard negative captions sampled per video increases the performance on Verb$_H$. We use $N_{\text{hard}} = 5$ in the remaining of the paper.

We therefore adopt HardNeg-NCE from [57, 61], which is denoted as HardNeg-NCE. With this objective, difficult negative pairs (with higher similarity) are emphasised, and easier pairs are ignored. We use $\alpha = 1$ and $\beta = 0.1$ in the equations from [57]. We note that we only adapt $L_{i}^{2\text{ev}}$ and $L_{i}^{\text{CSW}}$ with HardNeg-NCE. Adapting $L_{i}^{\text{verb-phase}}$ does not bring further improvements, so we omit this for simplicity. We observe in Tab. 8 that VFC is complementary to existing hard negative frameworks: using HardNeg-NCE instead of the standard NCE loss achieves the highest performance. We observe a large boost on Verb$_H$ [53], a benchmark that specifically involves hard negatives. We therefore adopt HardNeg-NCE in the remaining of this paper.

4.3. Comparisons to the State of the Art

We compare our VFC features to the state of the art on a diverse set of tasks requiring verb understanding. Note that we use the same model across different tasks, which is non-trivial in itself as the tasks cover a wide range of domains and evaluation protocols.

**MSR-VTT MC results.** We see in Tab. 9 that our verb-focused pretraining transfers well to the MSR-VTT multi-choice task, especially on the hard verb split (curated to assess exactly the task we are trying to solve). We even out-

<table>
<thead>
<tr>
<th>Method</th>
<th>Contrastive loss</th>
<th>Verb$_H$</th>
<th>K-400</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>NCE</td>
<td>69.9</td>
<td>55.6</td>
</tr>
<tr>
<td>Baseline</td>
<td>HardNeg-NCE</td>
<td>72.0</td>
<td>56.4</td>
</tr>
<tr>
<td>VFC (Ours)</td>
<td>NCE</td>
<td>78.3</td>
<td>58.5</td>
</tr>
<tr>
<td>VFC (Ours)</td>
<td>HardNeg-NCE</td>
<td>80.5</td>
<td>58.8</td>
</tr>
</tbody>
</table>

Table 8. Complementarity with other hard negative mining methods. We observe that using the HardNeg-NCE loss, instead of standard NCE, gives the highest performance. We use HardNeg-NCE from now on. We note that for VFC we use $N_{\text{hard}} = 5$.

Complementarity with other hard negative mining methods. We investigate whether our VFC framework is complementary to existing approaches for hard negatives with the contrastive learning framework. Specifically, we reimplement the hard negative noise contrastive multimodal alignment loss from [57, 61], which is denoted as HardNeg-NCE. With this objective, difficult negative pairs (with higher similarity) are emphasised, and easier pairs are ignored. We use $\alpha = 1$ and $\beta = 0.1$ in the equations from [57]. We note that we only adapt $L_{i}^{2\text{ev}}$ and $L_{i}^{\text{CSW}}$ with HardNeg-NCE. Adapting $L_{i}^{\text{verb-phase}}$ does not bring further improvements, so we omit this for simplicity. We observe in Tab. 8 that VFC is complementary to existing hard negative frameworks: using HardNeg-NCE instead of the standard NCE loss achieves the highest performance. We observe a large boost on Verb$_H$ [53], a benchmark that specifically involves hard negatives. We therefore adopt HardNeg-NCE in the remaining of this paper.

We see in Tab. 9 that our verb-focused pretraining transfers well to the MSR-VTT multi-choice task, especially on the hard verb split (curated to assess exactly the task we are trying to solve). We even out-

<table>
<thead>
<tr>
<th>Model</th>
<th># params.</th>
<th>3k val.</th>
<th>Verb$_H$ [53]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ZERO-SHOT</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VideoCLIP [84]</td>
<td>–</td>
<td>73.9</td>
<td>–</td>
</tr>
<tr>
<td>CLIP [58]</td>
<td>151M</td>
<td>91.1</td>
<td>64.1</td>
</tr>
<tr>
<td>InternVideo [75]</td>
<td>$\approx$ 460M</td>
<td>93.4</td>
<td>–</td>
</tr>
<tr>
<td>VFC (Ours)</td>
<td>164M</td>
<td>95.1</td>
<td>80.5</td>
</tr>
<tr>
<td><strong>FINE-TUNED</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ClipBERT [38]</td>
<td>–</td>
<td>88.2</td>
<td>–</td>
</tr>
<tr>
<td>MMT [26]</td>
<td>–</td>
<td>92.4</td>
<td>71.3</td>
</tr>
<tr>
<td>VideoCLIP [84]</td>
<td>–</td>
<td>92.1</td>
<td>–</td>
</tr>
<tr>
<td>CLIP-straight [55]</td>
<td>151M</td>
<td>94.1</td>
<td>65.1</td>
</tr>
<tr>
<td>MMT [26] (CLIP features)</td>
<td>–</td>
<td>95.0</td>
<td>71.4</td>
</tr>
<tr>
<td>C4CL-mP [53]</td>
<td>151M</td>
<td>96.2</td>
<td>73.7</td>
</tr>
<tr>
<td>VFC (Ours)</td>
<td>164M</td>
<td>96.2</td>
<td>85.2</td>
</tr>
</tbody>
</table>

Table 9. Multi-choice MSR-VTT. We report accuracy on the 3k val and on the verb-focused Verb$_H$ [53] splits. While VFC improves the performance on both splits in a zero-shot setting, the gap with previous works is especially important on Verb$_H$ [53], a split measuring verb understanding. When available, we add model parameter counts.

<table>
<thead>
<tr>
<th>Model</th>
<th>all D T C</th>
<th>ATP$_{\text{hard}}$ [9]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ZERO-SHOT</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLIP [58]</td>
<td>43.9</td>
<td>57.0</td>
</tr>
<tr>
<td>VFC (Ours)</td>
<td>51.5</td>
<td>64.1</td>
</tr>
<tr>
<td><strong>FINE-TUNED</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HGA† [33]</td>
<td>49.7</td>
<td>59.3</td>
</tr>
<tr>
<td>ATP [9]</td>
<td>49.2</td>
<td>58.9</td>
</tr>
<tr>
<td>Temp[ATP] [9]</td>
<td>51.5</td>
<td>65.0</td>
</tr>
<tr>
<td>TAATP† [83]</td>
<td>54.3</td>
<td>66.8</td>
</tr>
<tr>
<td>VGT [83]</td>
<td>55.0</td>
<td>64.1</td>
</tr>
<tr>
<td>VFC (Ours)</td>
<td>58.6</td>
<td>72.8</td>
</tr>
</tbody>
</table>

Table 10. NEXT-QA video question answering. We report accuracy. We consider either ‘all’ questions or only causal (‘C’), temporal (‘T’) or descriptive (‘D’) questions. We also use ATP$_{\text{hard}}$ split [9]. VFC improves performance for both zero-shot and fine-tuning. †Temp[ATP]+ATP. ‡ Uses additional motion features.

perform concurrent InternVideo [75] while using a significantly smaller setting both in terms of architecture (InternVideo uses $2.8\times$ more parameters and $12.4\times$ more flops) and pretraining dataset size (they use $24\times$ more data). We also note that our method does not degrade performance on other standard object-based tasks, such as text-to-video retrieval on MSR-VTT (results compared to the state of the art are shown in Sec. A.2 of the appendix).

NEXT-QA results. We show in Tab. 10 that our verb-focused pretraining gives a significant boost in both the standard and ATP$_{\text{hard}}$ setting introduced by [9]. We highlight the improved performance for the descriptive (and therefore more noun-focused) setting. To the best of our knowledge, we are the first work to report zero-shot results
The evaluation is then run on a random sample of Kinetics-400, but within Kinetics-600 are used for evaluation. Specifically, the subset of categories which are outside Kinetics-600 in Tab.

Zero-shot Kinetics-600 results. We evaluate our model on Kinetics-600 in Tab. 12 and follow the protocol in [13, 52]. Specifically, the subset of categories which are outside Kinetics-400, but within Kinetics-600 are used for evaluation. The evaluation is then run on a random sample of 160 categories from this subset. The final performance is averaged over three iterations. We observe that by evaluating our model in a zero-shot setting, we surpass the performance of works [13, 52] which fine-tune on Kinetics-400.

### Table 13. Zero-shot Kinetics-verb

<table>
<thead>
<tr>
<th>Model</th>
<th>all</th>
<th>Kinetics-verb</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>55.6</td>
<td>52.1</td>
</tr>
<tr>
<td>VFC (Ours)</td>
<td>58.8 (+3.2)</td>
<td>57.1 (+5.0)</td>
</tr>
</tbody>
</table>

### Table 14. Verb understanding on SVO-probes [31]

<table>
<thead>
<tr>
<th>Model</th>
<th>AP</th>
<th>APverb</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLIP [58]</td>
<td>48.3</td>
<td>52.3</td>
</tr>
<tr>
<td>No-MRM-MMT [31]†</td>
<td>51.5</td>
<td>53.1</td>
</tr>
<tr>
<td>Baseline (Ours)</td>
<td>60.2</td>
<td>61.9</td>
</tr>
<tr>
<td>VFC (Ours)</td>
<td>61.8</td>
<td>64.6</td>
</tr>
</tbody>
</table>

### 5. Conclusion

Video-language models based on CLIP have been shown to have limited verb understanding, relying extensively on nouns. We attempt to alleviate this problem with two technical contributions on the contrastive learning framework: first, we leverage LLMs to automatically generate hard negative captions focused on verbs; second, we introduce a verb phrase alignment loss. We validate our verb-focused pretraining by showing improved performance on a suite of benchmarks, chosen in particular to assess verb understanding. Our framework is general and could be employed for other video-language tasks, and further readily scales with the rapid progress in language modelling.

**Acknowledgements.** We would like to thank Ahmet Iscen, Anurag Arnab, Paul Hongsuck Seo, Antoine Yang, Shyamal Buch, Alex Salcianu for their precious help and discussions. We also thank Sagar Vaze for his invaluable support.
References


[32] De-An Huang, Vignesh Ramanathan, Dhruv Mahajan, Lorenzo Torresani, Manohar Paluri, Li Fei-Fei, and


[53] Jae Sung Park, Sheng Shen, Ali Farhadi, Trevor Darrell, Yejin Choi, and Anna Rohrbach. Exposing the limits of video-text models through contrast sets. In *ACL*, 2021. 1, 2, 6, 8, 3, 4, 5, 10, 13


[62] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Chris Pal, Hugo Larochelle, Aaron Courville, and
Bernt Schiele. Movie description. JVC, 2017. 1


[77] Dongli Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In CVPR, 2018. 3


[81] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiqing He, Philipp Krähenbühl, and Ross Girshick. Long-Term Feature Banks for Detailed Video Understanding. In CVPR, 2019. 3


[85] 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, 2

[86] Ran Xu, Caiming Xiong, Wei Chen, and Jason J. Corso. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In AAAI, 2015. 3

[87] Shen Yan, Xuehan Xiong, Anurag Arnab, Zhichao Lu, Mi Zhang, Chen Sun, and Cordelia Schmid. Multiview transformers for video recognition. In CVPR, 2022. 1


[93] Mert Yuksegkonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In ICLR, 2023. 1, 2


[95] Andy Zeng, Maria Attarian, Brian Ichtier, Krzysztof Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aweek Purohit, Michael Wray and Dima Damen. Learning visual actions. In Neurips, 2019. 3

[96] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and
