

VideoBERT

A Joint Model for Video and Language Representation Learning

Pascale Walters

CS 886
University of Waterloo

March 9, 2020

Motivation

- Good performance in computer vision tasks tends to require huge amounts of labelled data
- Online platforms, such as YouTube, have huge amounts of unlabelled video data
- Can we get high level representations of videos without the need for manual annotation?

Motivation

- Humans can describe objects and events in a video at a high level with language
 - Natural source of self-supervision
- Use BERT to learn a joint linguistic and visual representation
 - $p(x, y)$, x is a series of visual words and y is a series of spoken words
- Applications: text-to-video prediction, dense video captioning, long-range forecasting

BERT

- BERT (bidirectional encoding representations from transformers) gets representations of words from text by looking to the left and right
- It is pre-trained in an unsupervised manner with a large corpus of text and fine-tuned for specific tasks
- Pre-training is done in two steps:
 - 1 Masked language model
 - 2 Next sentence prediction

VideoBERT

- Extend BERT to videos by generating “visual words” with hierarchical vector quantization
 - Can still use pretrained models and existing implementations
 - The model focuses on high level semantics and long-range temporal dynamics

Combining Visual and Linguistic Sentences

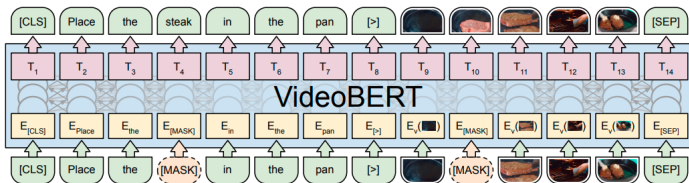


Figure: VideoBERT token prediction (*cloze* task)

In VideoBERT, visual and linguistic sentences are combined with a special token

[CLS] orange chicken with [MASK] sauce >]

v01 [MASK] v08 v72 [SEP]

Combining Visual and Linguistic Sentences

- Replace the next sentence prediction task from BERT with a linguistic-visual alignment task
 - Predict whether linguistic sentence is temporally aligned with visual sentence

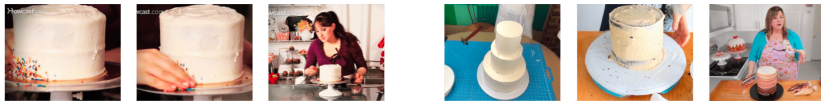
Dataset

- Instructional videos: spoken words are more likely to refer to the visual content
- Cooking videos: existing annotated dataset for evaluation
 - YouCook II: 2K videos, 176 hours
- Training dataset collected from YouTube
 - 312K videos, 23,186 hours
 - Text extracted with YouTube's automatic speech recognition toolkit

Video Tokenization

- Just like with BERT, a tokenized representation of the input video is required
- Extract representations for 30-frame clips with S3D pretrained on Kinetics dataset
- Tokenize the visual features with hierarchical k -means clustering
- Text is tokenized with WordPieces (as with BERT)
- To break the video into sentences,
 - Start and end timestamps of a text sentence, or
 - 16 tokens

Video Tokenization



"but in the meantime, you're just kind of moving around your cake board and you can keep reusing make sure you're working on a clean service so you can just get these all out of your way but it's just a really fun thing to do especially for a birthday party."



"apply a little bit of butter on one side and place a portion of the stuffing and spread evenly cover with another slice of the bread and apply some more butter on top since we're gonna grill the sandwiches."

Figure: Examples of frame and sentence pairs. The images on the left are frames extracted from the video and the images on the right are visual centroids.

Downstream Tasks

VideoBERT proposes two downstream tasks:

- 1 Zero-shot classification
- 2 Feature extraction for video captioning

Zero-Shot Classification

- Zero-shot: test model with different data or labels that were used for training
- Want to compute $p(y | x)$, x is the sequence of visual tokens, y is a fixed sentence (extract verb and noun):

now let me show you how to [MASK] the [MASK].

- Model trained on YouTube dataset is tested on the YouCook II dataset
 - VideoBERT achieves a comparable top-5 accuracy to a fully supervised classifier trained on YouCook II

Zero-Shot Classification

Method	Supervision	verb top-1 (%)	verb top-5 (%)	object top-1 (%)	object top-5 (%)
S3D [34]	yes	16.1	46.9	13.2	30.9
BERT (language prior)	no	0.0	0.0	0.0	0.0
VideoBERT (language prior)	no	0.4	6.9	7.7	15.3
VideoBERT (cross modal)	no	3.2	43.3	13.1	33.7

Figure: Performance of the VideoBERT model on the zero-shot classification task tested on the YouCook II dataset.

Zero-Shot Classification



Top verbs: make, assemble, prepare
Top nouns: pizza, sauce, pasta



Top verbs: make, do, pour
Top nouns: cocktail, drink, glass



Top verbs: make, prepare, bake
Top nouns: cake, crust, dough

Figure: Examples of VideoBERT zero-shot classification.

Transfer Learning for Video Captioning

- VideoBERT can be used to extract features from video input:

now let's [MASK] the [MASK] to the [MASK],

and then [MASK] the [MASK].

- Use the output as input for a supervised downstream model
 - Concatenate average of video tokens and masked out text tokens
 - Model is a transformer encoder-decoder that maps video segments to captions
- Best performance is achieved on cross-modal pretraining

Transfer Learning for Video Captioning

Method	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr
Zhou <i>et al.</i> [39]	7.53	3.84	11.55	27.44	0.38
S3D [34]	6.12	3.24	9.52	26.09	0.31
VideoBERT (video only)	6.33	3.81	10.81	27.14	0.47
VideoBERT	6.80	4.04	11.01	27.50	0.49
VideoBERT + S3D	7.59	4.33	11.94	28.80	0.55

Figure: Performance on the video captioning task on the YouCook II dataset. All models are based on that of Zhou *et al.*, but with varying inputs.

Transfer Learning for Video Captioning



GT: add some chopped basil leaves into it

VideoBERT: chop the basil and add to the bowl

S3D: cut the tomatoes into thin slices



GT: cut the top off of a french loaf

VideoBERT: cut the bread into thin slices

S3D: place the bread on the pan



GT: cut yu choy into diagonally medium pieces

VideoBERT: chop the cabbage

S3D: cut the roll into thin slices



GT: remove the calamari and set it on paper towel

VideoBERT: fry the squid in the pan

S3D: add the noodles to the pot



Figure: Examples of VideoBERT captioning.

Effects of Dataset Size

- Is there an effect on the size of the training dataset on the performance of the model?
- Performance increases with increased training dataset size with no appearance of saturation

Method	Data size	verb top-1 (%)	verb top-5 (%)	object top-1 (%)	object top-5 (%)
VideoBERT	10K	0.4	15.5	2.9	17.8
VideoBERT	50K	1.1	15.7	8.7	27.3
VideoBERT	100K	2.9	24.5	11.2	30.6
VideoBERT	300K	3.2	43.3	13.1	33.7

Figure: Performance of VideoBERT on the zero-shot classification task with increasing dataset size.

Limitations of Vector Quantization

- The vector quantization in VideoBERT can lose fine-grained details that may be critical for downstream tasks
- The authors propose another method for encoding video sequences for training bidirectional transformer models

Contrastive Bidirectional Transformer

- CBT doesn't need a pre-trained visual encoder
- Use softmax version of noise contrastive estimation loss in BERT, since images and videos have real-valued vectors as inputs instead of a fixed discrete vocabulary
- Better at utilizing long temporal context

Contrastive Bidirectional Transformer

- Try to maximize mutual information between the video and language sequences at a sequence level, rather than at a frame level

CBT Loss Function

$$L_{\text{cbt}} = w_{\text{bert}}L_{\text{bert}} + w_{\text{visual}}L_{\text{visual}} + w_{\text{cross}}L_{\text{cross}} \quad (1)$$

w_{bert} is set to 0, since a pretrained BERT is used

w_{visual} is set to 1

w_{cross} is either 1 or 0

CBT Loss Function

Softmax version of the noise contrastive estimation

$$L_{visual} = -E_{x \sim \mathcal{D}} \sum_t \log \text{NCE}(x_t | x_{-t}) \quad (2)$$

$$\text{NCE}(x_t | x_{-t}) = \frac{\exp(e_t^T \hat{e}_t)}{\exp(e_t^T \hat{e}_t) + \sum_{j \in \text{neg}(t)} \exp(e_j^T \hat{e}_t)} \quad (3)$$





where $e_t = f_{\text{enc}}(x_t)$ is the output of the S3D model applied to a small window around frame t , $\hat{e}_t = g_{\text{context}}(e_{-t})$ is the output of the visual transformer, and $\text{neg}(t)$ is a set of negative examples.

Performance on Video Captioning

Method	BLEU-4	METEOR	ROUGE-L	CIDEr
Zhou et al. (2018c)	4.38	11.55	27.44	0.38
S3D	3.24	9.52	26.09	0.31
VideoBERT	4.33	11.94	28.80	0.55
CBT	5.12 (± 0.02)	12.97 (± 0.05)	30.44 (± 0.08)	0.64 (± 0.00)

Figure: The CBT method performs better than VideoBERT on video captioning and achieves state-of-the-art on the YouCookII dataset. The only difference between the methods is the video encoder.

References

-  Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
-  Sun, C., Baradel, F., Murphy, K. & Schmid, C. *Learning Video Representations using Contrastive Bidirectional Transformer*. 2019. [arXiv: 1906.05743 \[cs.LG\]](https://arxiv.org/abs/1906.05743).
-  Sun, C., Myers, A., Vondrick, C., Murphy, K. & Schmid, C. *Videobert: A joint model for video and language representation learning*. in *Proceedings of the IEEE International Conference on Computer Vision* (2019), 7464–7473.
-  Zhou, L., Zhou, Y., Corso, J. J., Socher, R. & Xiong, C. *End-to-End Dense Video Captioning with Masked Transformer*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), 8739–8748.

Thank You

Video Captioning

- End-to-end video captioning
- The encoder encodes the video frames (features) into the proper representation. The proposal decoder then decodes this representation with different anchors to form event proposals, i.e., start and end time of the event, and a confidence score. The captioning decoder then decodes the proposal specific representation using a masking network, which converts the event proposal into a differentiable mask.

Dense Video Captioning

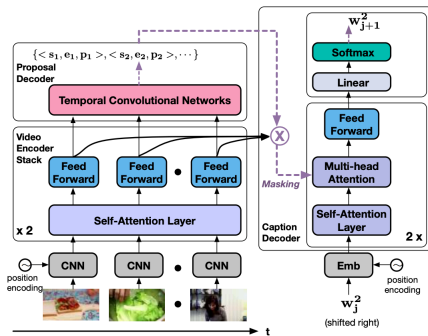


Figure 1. Dense video captioning is to localize (temporal) events from a video, which are then described with natural language sentences. We leverage temporal convolutional networks and self-attention mechanisms for precise event proposal generation and captioning.

Figure: Dense video captioning.

S3D Feature Extraction

- Separable 3D CNN (S3D) is a network that finds an optimal balance between complexity and accuracy for video classification

S3D Feature Extraction

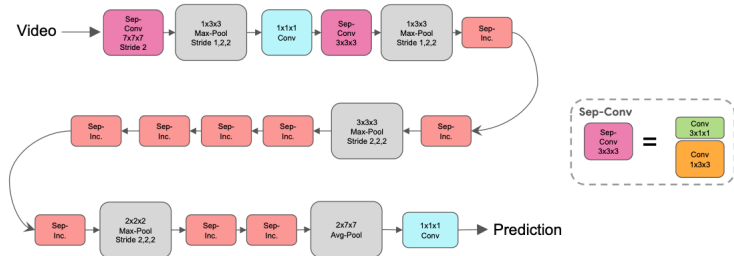


Figure: S3D architecture.