VideoBERT
A Joint Model for Video and Language Representation Learning

Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid

Seminar on Current Works in Computer Vision

Presenter: Anna Khatyreva
Advisor: Mohammadreza Zolfaghari
Presentation Overview

• Introduction

• Historical overview

• BERT model

• VideoBERT model

• Summary
Introduction

Motivation
Understand the contents and dynamics of video

Goal
Discover high-level semantic features that correspond to actions and events that unfold over longer time scales

Approach
Learn a bidirectional joint distributions over sequences of visual and linguistic tokens

Cut the *lettuce* into pieces
Introduction: Tasks

Source: Sun et al. [2019]
Introduction: Tasks

Source: Sun et al. [2019]
Historical Overview

Bidirectional Transformer

<table>
<thead>
<tr>
<th>Left-to-right Transformer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concatenation of left-to-right and right-to-left LSTMs</td>
</tr>
</tbody>
</table>

Source: Devlin et al. [2019]
**Goal:** Design a model to pretrain deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers

**Approach:** Multi-layer bidirectional Transformer encoder
BERT: Architecture

Architectures

- $\text{BERT}_{\text{BASE}}$: Number of Transformers layers = 12
  Number of self-attention heads = 12
- $\text{BERT}_{\text{LARGE}}$: Number of Transformers layers = 24
  Number of self-attention heads = 16

Datasets

- Wikipedia (2.5B words)
- BookCorpus (800M words)

Steps

1. Train a language model on a large unlabeled text corpus
2. Fine-tune this large model to specific NLP tasks
## BERT: Input Representation

<table>
<thead>
<tr>
<th>Input</th>
<th>[CLS]</th>
<th>my</th>
<th>dog</th>
<th>is</th>
<th>cute</th>
<th>[SEP]</th>
<th>he</th>
<th>likes</th>
<th>play</th>
<th>#.#ing</th>
<th>[SEP]</th>
</tr>
</thead>
</table>

### Token Embeddings

<table>
<thead>
<tr>
<th>Token Embeddings</th>
<th>$E_{[CLS]}$</th>
<th>$E_{my}$</th>
<th>$E_{dog}$</th>
<th>$E_{is}$</th>
<th>$E_{cute}$</th>
<th>$E_{[SEP]}$</th>
<th>$E_{he}$</th>
<th>$E_{likes}$</th>
<th>$E_{play}$</th>
<th>$E_{#.#ing}$</th>
<th>$E_{[SEP]}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

### Segment Embeddings

<table>
<thead>
<tr>
<th>Segment Embeddings</th>
<th>$E_{A}$</th>
<th>$E_{A}$</th>
<th>$E_{A}$</th>
<th>$E_{A}$</th>
<th>$E_{A}$</th>
<th>$E_{B}$</th>
<th>$E_{B}$</th>
<th>$E_{B}$</th>
<th>$E_{B}$</th>
<th>$E_{B}$</th>
<th>$E_{B}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

### Position Embeddings

<table>
<thead>
<tr>
<th>Position Embeddings</th>
<th>$E_{0}$</th>
<th>$E_{1}$</th>
<th>$E_{2}$</th>
<th>$E_{3}$</th>
<th>$E_{4}$</th>
<th>$E_{5}$</th>
<th>$E_{6}$</th>
<th>$E_{7}$</th>
<th>$E_{8}$</th>
<th>$E_{9}$</th>
<th>$E_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Source: Devlin et al. [2019]
BERT: Masked Language Model

Problem

Bidirectional conditioning would allow each word to trivially predict the target word in a multi-layer context.

Solution

Mask out 15% of all WordPiece tokens in each sentence at random.

Input:

the man went to [MASK] store

[MASK] — token that replaces “masked” words
BERT: Next Sentence Prediction

[CLS] – special classification token
[SEP] – special token to separate to different sentences
[MASK] – token that replaces “masked” words
Label = \{IsNext, NotNext\} – shows if the two sentences are consistent

**Input:**

[CLS] the man went to [MASK] store [SEP] he bought a gallon [MASK] milk [SEP]
Label = IsNext

**Input:**

[CLS] the man [MASK] to the store [SEP] penguin [MASK] are flight ## less birds [SEP]
Label = NotNext
BERT: Fine-tuning

Sentence Pair Classification Tasks

Single Sentence Classification Tasks

Source: Devlin et al. [2019]
BERT: Fine-tuning

Question Answering Tasks

Single Sentence Tagging Tasks

Source: Devlin et al. [2019]
**Goal:** Find a way to model the relationships between the visual model and the linguistic domain

**Approach:**

1. Use an automatic speech recognition (ASR) system to convert speech into text
2. Apply vector quantization (VQ) to low-level spatio-temporal visual features derived from pretrained video classification models
3. Use BERT model for learning joint distributions over sequences of discrete tokens
VideoBERT: Datasets

Training

⇒ Focus on cooking videos
⇒ Use videos from YouTube
⇒ Use YouTube’s ASR toolkit provided by the YouTube Data API to obtain text from videos

Result: 120K videos

Evaluation

⇒ Use YouCook II dataset

Result: 2K videos
**Input:**

[CLS] orange chicken with [MASK] sauce [>] v01 [MASK] v08 v72 [SEP]

[CLS] — special classification token
[SEP] — special token to separate to different sentences
[MASK] — token that replaces “masked” words
[>] — special token to combine text and video sentences
v01, v08, v72 — visual tokens
Illustration of VideoBERT in the context of a cloze task

Source: Sun et al. [2019]
[CLS] — the first token of a sequence
[SEP] — special token to separate to different sentences
[MASK] — token that replaces “masked” words
\( c = \{0, 1\} \) — shows if the two sentences are consistent
\( v01, v08, v72 \) — visual tokens
[>] — special token to combine text and video sentences


Class label: \( c = 1 \)
Problem

[CLS] token is a noisy indicator, since the speaker may be referring to something that is not visually present

Solution

- Concatenate neighboring sentences into a single long sentence
- Randomly pick a subsampling rate of 1 to 5 steps for the video tokens
VideoBERT: Video and Language Preprocessing

Visual words

1. Sample frames at 20 fps and create clips from 30-frame non-overlapping windows over video

2. For each 30-frame clip, apply S3D network to obtain feature vector

3. Tokenize the features using hierarchical $k$-means

The number of hierarchy levels $d = 4$
The number of clusters per level $k = 12$
$12^4 = 20736$ clusters in total
“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”

Source: Sun et al. [2019]
Utilize YouTube’s ASR toolkit provided by the YouTube Data API to retrieve timestamped speech information.

1. Tokenize the text obtained from the video into WordPieces.

Visual sentences

1. According to the ASR sentence

2. If ASR sentence is not available, use 16 tokens as a segment.
VideoBERT: Zero-shot action classification

Action 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

Source: Sun et al. [2019]
### Action classification performance on YouCook II dataset

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

Source: Sun et al. [2019]
**Idea**

Use VideoBERT as a feature extractor

**Approach**

- Append the video tokens to a template sentence
- Extract the feature for the video tokens and the masked out text tokens

"now let’s [MASK] the [MASK] to the [MASK], and then [MASK] the [MASK]"
# VideoBERT: Transfer Learning

## Video captioning performance on YouCook II

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

Source: Sun et al. [2019]
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

Source: Sun et al. [2019]
VideoBERT: Tasks

Text-to-video generation

Retrieved centroid

Retrieved centroid

Cut the *steak* into pieces

Cut the *carrot* into pieces

Source: Sun et al. [2019]
VideoBERT: Tasks

Text-to-video generation

Put the *cookies* into oven

Put the *pizza* into oven

Source: Sun *et al.* [2019]
**Video-to-text generation**

**Original**

**Centroids**

**Original**

**Centroids**

**VideoBERT: Tasks**

**Video-to-text generation**

**ASR:**

“This is what happens when you play with dough thinking of yourselves as a kitten who happens to look like Ed Sheeran”

**Top verbs:** make, shape, roll

**Top nouns:** dough, filling, chicken

**ASR:**

“I highly recommend that you use a whole sheet just because when you make the smaller sushi rolls, they the fixings tend to fall out”

**Top verbs:** roll, make, cut

**Top nouns:** fish, salmon, dough

Source: Sun et al. [2019]
VideoBERT: Tasks

Video-to-video generation

Source: Sun et al. [2019]
The impact of the training dataset size on the action classification performance on YouCook II dataset

<table>
<thead>
<tr>
<th>Method</th>
<th>Data size</th>
<th>verb top-1 (%)</th>
<th>verb top-5 (%)</th>
<th>object top-1 (%)</th>
<th>object top-5 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>VideoBERT</td>
<td>10K</td>
<td>0.4</td>
<td>15.5</td>
<td>2.9</td>
<td>17.8</td>
</tr>
<tr>
<td>VideoBERT</td>
<td>50K</td>
<td>1.1</td>
<td>15.7</td>
<td>8.7</td>
<td>27.3</td>
</tr>
<tr>
<td>VideoBERT</td>
<td>100K</td>
<td>2.9</td>
<td>24.5</td>
<td>11.2</td>
<td>30.6</td>
</tr>
<tr>
<td>VideoBERT</td>
<td>300K</td>
<td>3.2</td>
<td>43.3</td>
<td>13.1</td>
<td>33.7</td>
</tr>
</tbody>
</table>

Source: Sun et al. [2019]
Summary

- VideoBERT can be used in numerous tasks, including action classification and video captioning
- This paper adapts the BERT model to learn a joint visual-linguistic representation for video
- The model is able to learn high level semantic representations and outperforms the state-of-the-art for video captioning on the YouCook II dataset
- The model can be used directly for open-vocabulary classification
- The performance of the model grows monotonically with the size of training set