CS230

Using CLIP on the Charades Video Dataset for Visual Understanding (Computer Vision)

Eva Prakash, Josh Singh, Isaiah Turner
Department of Computer Science
Stanford University
eprakash@stanford.edu, jsingh5@stanford.edu, isaiah42@stanford.edu

Abstract

Captioning videos is important for making arbitrary videos more accessible to the hearing and vision impaired, but often the task is too labor-intensive for humans. In this work, we aim to finetune a pretrained CLIP model [2] on the Charades video dataset to match captions to videos. Our baseline model uniformly sampled frames from each video and matched a caption to each one. We found a training accuracy of 164/26957 and a test accuracy of 0/11949 for the baseline model. We improved upon our baseline model by creating a novel video encoder to pass in a 3 x 3 grid image per video into CLIP. Using a ResNet50-PCA-K-Means-SIFAR pipeline, each 3 x 3 grid image is comprised of 9 selected key frames for a particular video. This 3 x 3 grid image allows CLIP to learn the most important information from each video all at once instead of looking at randomly sampled frame images. We found a training accuracy of 34/1000 and test accuracy of 1/500 for this improved model.

1 Introduction

In the realm of computer vision, video understanding has applications in accurately describing objects, actions, and relationships in videos. In particular, image-text pairing when matching captions to videos split into frames has shown that more efficient, complex, and accurate image contextual representations can be learned through natural language supervision [2]. Specifically, it has major benefits in helping the deaf and blind communities understand videos, flagging inappropriate videos, and describing medical images [1].

Our project aims to evaluate the CLIP (Contrastive Language–Image Pre-training) [2] model, which uses a neural network to pair images with corresponding text descriptions, on video captioning for the Charades video dataset. [3] The CLIP model encodes an image and caption, then uses the encodings to match the image to the caption. CLIP has been used for zero-shot transfer, natural language supervision, and multimodal learning. The Charades dataset contains action annotations, object labels, and text descriptions for each of its approximately 10,000 videos. Charades is of particular interest to us because of its collection of daily dynamic scenes, which display a vast range of everyday objects and actions.

Our algorithm encodes information from the Charades dataset and matches video frame images with accurate text descriptions. The input to our model is a video from Charades and its corresponding caption. Our baseline model takes frames at set intervals from each video and then uses the image-captioning functionality of CLIP to match each frame image to one of the video captions in the Charades dataset. Our final model learns 9 key frame images per video with ResNet and K-means

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

clustering and places the images into a 3 x 3 grid to provide relevant video information to CLIP all at once. A grid of key frames is intended to improve video captioning, since individual frames generally cannot encapsulate the entire video caption. We used 1000 videos in our training set and 500 videos in our test set for both our baseline and final models. The baseline model achieved a training accuracy of 164/26957 and a test accuracy of 0/11949, and our final model improved upon the baseline by achieving a training accuracy of 34/1000 and a test accuracy of 1/500.

2 Related Work

We referenced previous work to aid in working with the CLIP model for video captioning using image-text pairing. In terms of fine-tuning CLIP on a novel dataset, we found that the Hugging Face/Google conference finetuned the image-captioning ability of CLIP with RCISD remote sensing (satellite) images. While their baseline CLIP model achieved 57.2% accuracy on the test set, they were able to finetune CLIP on their data and eventually achieve 88.3% accuracy [4]. We used this paper as inspiration for motivating our task of similarly fine-tuning CLIP on the Charades dataset. However, we did have to keep in mind that natural language supervision for image representation is still a fairly new realm, and vast, varied datasets are difficult to accurately caption. As an example, Li et al. was only able to achieve 11.5% accuracy on image-text pairing for ImageNet using a transformer-based model [6].

We explored a range of papers in deciding how to construct our video encoder. Carreira and Zisserman created the I3D network based on the 2D ConvNet inflation and expanded filters and pooling kernels to 3 dimensions, allowing the model to learn spatial-temporal features and leverage successful ImageNet architecture on video data. I3D pretrains on the data-heavy, complex Kinetics Human Action Video Dataset and outperforms the previously best models [14]. In contrast to our model, I3D does not preserve realism in its video embedding, so it would be difficult to pass these embeddings into CLIP for image-text pairing. Fan et al. takes a different approach to video understanding and captioning by turning the problem into an image recognition task in their SIFAR paper. They focus on choosing uniformly sampled frames from a video to create a super image that they pass into a transformer-based model for recognition. A key difference between our model and Fan et al.'s work is that we select key frames to put into our super image using our novel video encoder instead of simply using random frames. Jadon and Jasim's work trains ResNet16 without its final loss layer on ImageNet to get image embeddings and clusters these images using K-Means and Gaussian Mixture Models to group them based on shared conceptual characteristics [9]. We were inspired by this approach to choose the most key images to create our super image to pass into our model.

A pitfall of video understanding is the lack of multilabel tasks [12] and lack of natural language supervision [2]. Our modifications and applications to the CLIP model address both of these concerns. While to our knowledge, CLIP has not yet been used to evaluate videos or Charades in particular, it has been seen to be successful in other visual understanding tasks.

3 Dataset and Features

We are using the Charades dataset, which was created by the Allen Institute for AI. [3] Charades is available for public download on AI2's website.The dataset contains 9,848 videos of various indoor activities. Each video is accompanied by temporal action annotations, object class labels, and text descriptions of the video. The videos were recorded by participants in the Amazon Mechanical Turk program, so the resolutions and frame rates vary. However, we downloaded the data uniformly scaled down to 480p. By default, the dataset has the following train/test set split:

Train example count: 7,985 videos with total of 66.2 hours of runtime
Test example count: 1,863 videos with a total of 14.8 hours of runtime

To gather results for our models, due to the storage limitations of AWS, we used 1000 randomly sampled videos for training and 500 randomly sampled videos for testing within each subject category. We wanted to ensure each of the 16 video scene types (e.g. "garage", "kitchen," "bedroom," etc.) were present in each set, so we took 65 randomly sampled videos from each scene type except for "other" for the training set and 32 randomly sampled videos from each scene type except for "other" for the test set. To reach 1000 and 500 videos respectively, we randomly sampled the necessary

numbers of "other" videos (i.e. 25 and 20 videos for the training and test sets respectively). For the baseline model, we sampled every 50th frame from each video. Each video has the following information attached to it: an id, scene description, verified status, script, objects present, description, actions, and length. The video description is of particular interest, since we use the descriptions as captions for our images. Figure A in the appendix is a example of the raw annotations for a video. We also provide Figure B in the appendix that shows part of a video and the annotated information.

4 Approach

4.1 Baseline Model

Our model takes in a Charades video and its corresponding caption, intending to split the video into a series of representative frames. Our baseline model follows the naive approach of uniformly sampling frames from each video. We tuned the hyperparameter of "every Nth frame" to reach the number of $N = 50$, which was the most representative of each video without generating too many frames per video. We then perform the task of image-captioning using CLIP. We input the video frames and the caption of each video in order to have CLIP calculate the probability of each of the provided Charades captions across all videos matching to a particular frame image. We used the ViT-B/32 Vision Transformer [2] backbone for the image encoder and the Transformer [2] backbone for the text encoder. We finetuned CLIP on our data to utilize transfer learning from the pretrained CLIP model.

The image and text encoders of CLIP maximize the cosine similarity of the image and text embeddings of the correct image-caption pairs and minimize the cosine similarity of the embeddings of the incorrect image-caption pairs. CLIP uses cross-entropy loss to minimize mismatched images and captions and the softmax function to output the probability of each provided caption belonging to a certain image. The CLIP loss calculation is shown in the set of equations below, in which I is the matrix of image embeddings, T is the matrix of text embeddings, t is the learned temperature parameter, $CELoss$ is cross-entropy loss, and $Labels$ are the ground-truth image-text pairings [2]:

$$Logits = e^t(I \times T^T) \tag{1}$$

$$ImageLoss, TextLoss = CELoss(Logits, Labels) \tag{2}$$

$$Loss = \frac{ImageLoss + TextLoss}{2} \tag{3}$$

CLIP also makes use of the Adam optimizer to take advantage of the benefits of a moving average of squared gradients, momentum, and bias correction.

4.2 Final Model

Figure 1: Grid image generated by our final model for a video with the caption "A person is eating and then tidying up their mess, next wiping a mirror with a towel in a dining room."

The novel video encoder we built as part of our final model was constructed with inspiration from Fan et al. [13] and Jadon and Jasim [9]. In order to best select frame images from each video, we wanted to be able to isolate representative key frames. Our algorithm passes the frame images used to run our baseline model for each video through ResNet50. ResNet50 [10] is a high-performing deep residual neural network trained on ImageNet for the task of image classification. By removing the last pooling and fully-connected layer from ResNet50, we were able to obtain an image embedding of shape (7, 7, 2048) for each frame in order to conceptually represent them in the ResNet50 feature space.

We then flattened each image embedding into a vector of shape (100352,) and performed dimensionality reduction over the set of all flattened image embeddings using Principal Component Analysis (PCA). PCA performs dimensionality reduction on data by removing interrelated components and retaining the uncorrelated principal components. We used PCA in order for the following K-Means algorithm to calculate more meaningful clusters that were not overloaded by a high number of data dimensions.

Following PCA, we performed K-means clustering to partition the image embeddings into 9 clusters by minimizing the sum of squares between each embedding and the centroids of each cluster. The K-Means algorithm clusters data points around a given number of centroids by minimizing the following objective function J, where k is the number of clusters, n is the number of datapoints, $x_i^{(j)}$ is the i^{th} data point assigned to cluster j, and c_j is the centroid for cluster j:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} ||x_i^{(j)} - c_j||^2 \tag{4}$$

In essence, K-Means aims to minimize the squared error within each cluster. The frames with image embeddings closest to each centroid were used as the 9 key frames for each video, given the assumption that a centroid is what conceptually grounds a particular cluster. The Charades captions tend to contain multiple activities and objects, so a single image frame alone cannot likely match a video caption exactly. As a result, we followed the SIFAR method of Fan et al. [13] and placed the key frames into a 3 x 3 grid image in order to fit all the relevant video information into one image. Figure 1 displays the 3 x 3 grid image of a randomly chosen video after being passed through our video encoder. These grid images are then fed into CLIP with the corresponding video captions.

5 Experiments, Results, and Discussion

In our hyperparameter tuning, we found that Adam β_1 = 0.9, Adam β_2 = 0.98, Adam ϵ = $1e^{-6}$, and Adam weight decay = 0.2 worked best, as seen in the CLIP paper. We found that learning rate = $5e^{-5}$ was slow enough to not overshoot the global optimum but fast enough to avoid computational inefficiency. We used 15 epochs and tuned our batch size to 50 to be representative of the dataset while allowing for computational efficiency.

As seen in Figure 2, our baseline model achieved an accuracy of 164/26957 on our training dataset and 0/11949 on our test set. Our final model achieved an accuracy of 34/1000 on our training dataset and 1/500 on our test set. We saw that our final model improved upon our baseline, as it was able to encode all key video information into a single image, whereas none of the separate frame images used in our baseline model could likely display all the conceptual information of the corresponding video caption.

Model	Training Accuracy	Testing Accuracy
Baseline	164/26957	0/11949
Final	34/1000	1/500

Figure 2: Baseline and Final Model Accuracies

When we observed the average correct and overall caption lengths for both our training and test sets, we found that the average correct caption lengths were much shorter than the average overall

caption lengths. This seems to indicate that CLIP has a more difficult time matching longer captions to the correct videos, since there is a larger amount of information encoded in longer captions. Based on Figure 4, we found that in our training set, our final model was best able to correctly match "entryway", "pantry", and "basement" video scenes to their corresponding captions.

Test/Training	Average Correct Caption Length	Average Overall Caption Length
Training	78.18	111.299
Test	75	114.022

Figure 3: Average Character Length of Captions Overall and for Correct Matches for both Test and Training Data for Final Model

Scene	Training Accuracy	Scene	Training Accuracy
Home Office / Study	1/65	Kitchen	3/65
Garage	0/65	Bedroom	0/65
Entryway	5/65	Bathroom	2/65
Closet / Walk-in closet / Spear closet	2/65	Hallway	1/65
Laundry room	2/65	Living room	3/65
Recreation room / Man cave	3/65	Stairs	2/65
Dining room	1/65	Basement	4/65
Pantry	4/65	Other	1/25

Figure 4: Per Scene Training Set Accuracies for Final Model

While we observed our final model to improve upon our baseline, both models achieved relatively poor performance on our data. We suspect this is due to the length and complexity of the Charades captions. Even for videos that display the same high-level scene, the details of each caption can make such videos still appear very different, which makes generalizing to the test set from the training set difficult. As discussed previously, it is also more difficult to understand and encode the larger amount of information in longer captions in order to match these captions to the right videos. The Charades dataset is also a large dataset of videos with no bounds or restrictions on the type of content. As described by Li et al. [6], natural language supervision on large visual datasets with a wide variety of captions and content is prone to low accuracies due to data complexity. In terms of future work, our main course of action would be to run our final model and novel video encoder on a video dataset that is more narrow in content and contains shorter captions. This way, we would be able to work with more granular, bounded data.

6 Conclusion and Future Work

By using ResNet50 and K-Means to detect key frames in a video and organize them into a single grid image, we were able to observe accuracy gains from our baseline model when training and testing CLIP on the Charades video dataset. There is much room for improvement in the future. Aside from running our model on a different video dataset, we could try creating 2 x 2 grid images instead of 3 x 3 grid images, which may aid in allowing CLIP to zero in on an even narrower set of frames so as to not be potentially overwhelmed by 9 key frames. We were also limited by the constraints of AWS, so we could attempt further hyperparameter tuning and increase the number of epochs of training in the future. We could also try adapting I3D to work with K-Means instead of ResNet50 in our video encoder.

7 Contributions and Code

We all contributed equally to preprocessing our dataset, construction of our models, finetuning CLIP, and calculating evaluation metrics. Our code can be found at https://github.com/evaprakash/CLIP-Charades

References

[1] Haque, A., Milstein, A. Fei-Fei, L. Illuminating the dark spaces of healthcare with ambient intelligence. Nature 585, 193–202 (2020). https://doi.org/10.1038/s41586-020-2669-y

[2] Radford et al. Learning Transferable Visual Models From Natural Language Supervision. arXiV (2021). https://doi.org/10.48550/arxiv.2103.00020

[3] Allen Institute for AI (2016) Charades. ECCV (2016).

[4] Hugging Face (2021) Fine tuning CLIP with Remote Sensing (Satellite) images and captions. https://huggingface.co/blog/fine-tune-clip-rsicd

[5] Wortsman et al. Robust fine-tuning of zero-shot models. arXiV (2021)

[6] Li et al. Supervision Exists Everywhere: A Data Efficient Contrastive Language-Image Pre-training Paradigm. arXiV (2022). https://arxiv.org/pdf/2110.05208.pdf

[7] Zhong et al. RegionCLIP: Region-based Language-Image Pretraining. DeepAI (2021). https://deepai.org/publication/regionclip-region-based-language-image-pretraining

[8] Mu et al. SLIP: Self-supervision meets Language-Image Pre-training. arXiV (2021). https://arxiv.org/abs/2112.12750

[9] Jadon S. & Jasim, M. Unsupervised video summarization framework using keyframe extraction and video skimming. arXiV (2020). https://arxiv.org/pdf/1910.04792.pdf.

[10] He et al. Deep Residual Learning for Image Recognition. arXiV (2015). https://arxiv.org/abs/1512.03385

[11] Brownlee J. (2017) https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/

[12] Diba et al. Large Scale Holistic Video Understanding. arXiV (2020). https://arxiv.org/abs/1904.11451v3

[13] Fan et al. CAN AN IMAGE CLASSIFIER SUFFICE FOR ACTION RECOGNITION? ICLR (2022). https://openreview.net/pdf?id=qhkFX-HLuHV

[14] Carriera J. & Zisserman A. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. arXiV (2018). https://arxiv.org/pdf/1705.07750.pdf

[15] Li, A., Jabri, A., Joulin, A., and van der Maaten, L. Learning visual n-grams from web data. In Proceedings of the IEEE International Conference on Computer Vision, pp. 4183–4192, 2017.

8 Appendix

Figure A:

```
id: YSKX3
scene: Bedroom
verified: Yes
script: A person fixes the bed then throws pillow on it.
objects: bed;blanket;mattress;pillow
description: A person looks under a mattress and pats the
   bed.  This person picks up a pillow, and throws it on
   the bed.;A person is in a bedroom.  The person is
   fixing the bed. After the person cleans up his bed, the
    person leaves.
actions: c077 12.10 18.00;c079 11.80 17.30;c080 13.00
   18.00;c076 11.80 17.50;c075 5.40 14.10
length: 16.62
```

Figure B: