A Study of Contrastive Language-Image Pretraining (CLIP) for Image-Text Matching

Raja Sai Nandhan, Komal Krishna Mogilipalepu

Abstract

The need to bridge the gap between vision and language gives rise to solving the fundamental task of image-text matching. The key challenge here is extracting the features of image and text and measuring the similarity between the two. Recent works, Negative Aware Attention Framework (NAAF) focus on the importance of *negative attention* to measure the similarity and Contrastive Language-Image Pre-training (CLIP) focus on learning qualitative embeddings of image and text from 400 million image-text pairs. This work emphasizes extracting features and measuring the similarity by using Contrastive Language-Image Pre-training (CLIP) for image-text matching. We compare the CLIP-based image-text matching with the significant work Negative Aware Attention Framework (NAAF) for image-text matching. This work discusses the performance of CLIP for image-text matching on Flickr30K test dataset. From our analysis, we found that CLIP was able to reach the generalization capability of NAAF on unseen Flickr30K test data.

1 Introduction

Image-Text matching is a fundamental task in Computer Vision (CV) and Natural Language Processing (NLP). The goal is to search for relevant images given a caption query or relevant texts for a given image query.

There are numerous applications of Image-Text matching in the industry. Popular among them is search experience in e-commerce websites. Given a search query for an item, it should return a list of relevant items matching the text description. It enhances the user experience of the site by quickly returning the more relevant product without the need for an exhaustive search by the user. Another application would be image filtering on the internet. For example, when the user inputs the query in Table 1, the model should return Figure 1 as a potential search result.

There exist two paradigms for image-text matching.

- Global level matching: semantic correspondence between complete image and text. Popular examples include CLIP
- Local level matching: semantic correspondence between fine-grained sections in image and salient words in captions. Examples include NAAF.

On a clear blue day, a man had used a climbing device to shinny up what appeared to be a coconut palm tree.

Table 1: Given Caption

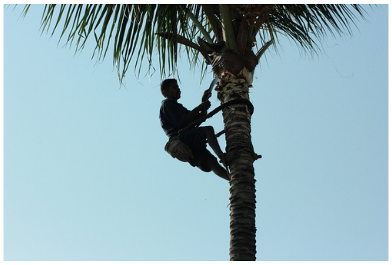

Figure 1: Retrieved Image

2 Related Work

2.1 Negative Aware Attention Framework (NAAF)

NAAF is a framework, that uses local-level matching. Unlike other similar models that ignore mismatched portions, NAAF accounts for both positive and negative matched sections in the image-text matching and computes similarity. As can be seen in Figure 2, other similar models only consider matched regions and completely ignore any mismatched regions by using ReLU. NAAF considers both matched and mismatched regions and computes the overall similarity score.

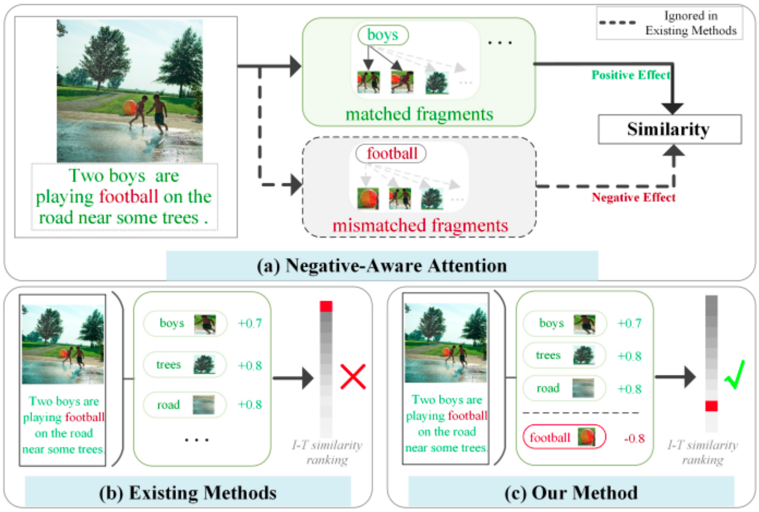

Figure 2: NAAF

2.2 Contrastive Langauge-Image Pretraining (CLIP)

CLIP is a neural network from OpenAI that effectively learns visual concepts from natural language supervision. Similar to previous GPT models such as GPT-2 and GPT-3, it can be applied to any classification benchmark by merely providing names of classes to be recognized. CLIP is mainly built on *natural language supervision*, *multimodal learning*, and *zero-shot transfer*. The model is trained on unfiltered, highly varied and noisy dataset of 400 million images. Traditional models need to be re-optimized on benchmark performance whenever data distribution is changed. CLIP is created to solve this problem and generalize on unseen data without the need to re-optimize.

3 Methodology

3.1 Image-Text Matching

Image-Text matching bridges the semantic gap between two different modalities image and text. The key challenge here is to accurately learn the semantic correspondence between texts and images to measure their similarity. The process here is first, we have to encode the image and text features to a common embedding space and compare those two vectors. Several researchers used pre-trained Faster R-CNN and bidirectional GRU/BERT models to get image and text embeddings respectively. Several papers focussed on attention mechanisms to measure the similarity of image and text after getting the embeddings. Negative-Aware Attention Framework (NAAF) considers negative attention including positive ones to measure the similarity between the segments of an image and the words in a text instead of making zero.

1 (Recal@K). *Given a similarity between an image and all the available sentences, a percentage of ground truth sentences available from the retrieved top K similarities.*

3.2 NAAF

Figure 3 outlines the modules in NAAF. In NAAF, there are two stages: Feature Extraction and Negative-aware Attention. In Feature extraction, we first extract important features from sections of both image and captions with Bi-directional GRU and encode them in vectors. They are input to negative-aware attention stage that measures image-text similarity, using both positive and negative effects. The negative-aware attention stage consists of two phases: 1) Discriminative Mismatch Mining, 2) Neg-Pos Branch Matching. The goal of Discriminative Mismatch Mining is maximize mismatched fragments, by minimizing the overlapping error between the matched and mismatched fragments. The goal of Neg-Pos branch matching is to calculate effects of both positive matches and negative mismatches to compute overall similarity with negative and positive attention branches and finally sum up the similarities to get the overall similarity. NAAF is one of the best methods which gives a significantly good Recal rate compared to the existing ones.

3.3 Contrastive Language-Image Pretraining (CLIP)

The Contrastive Language Image Pre-training (CLIP) has two steps to predict the best sentence for the given image. First, we have two encoders one for image and one for text. Both encoders encode the image and text information. To compare the image and text we need to project the encoded image and text vectors to a common embedding space. Next, we use the cosine similarity of the preprocessed image and text to measure the relevance between them. This is a supervised approach, we have a set of N image text pairs and the idea is to maximize the cosine similarity between the preprocessed image and text pair

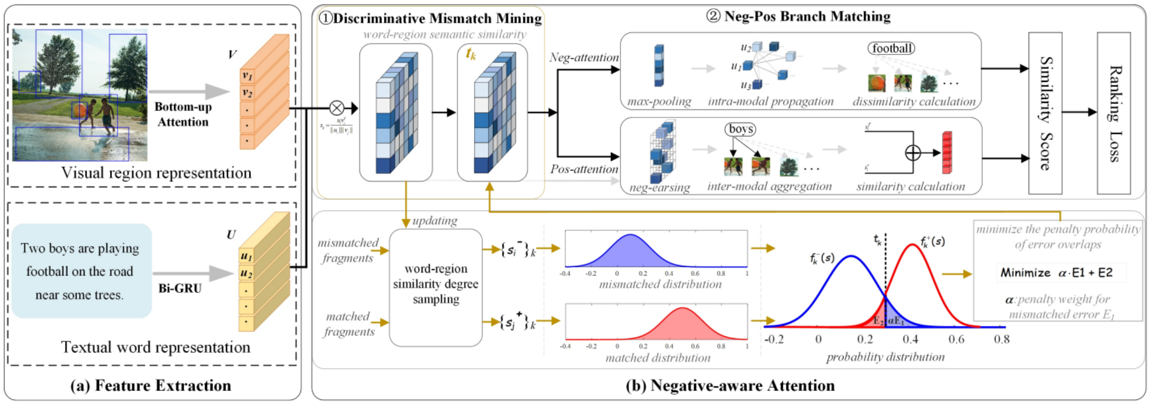

Figure 3: Overview of Negative-Aware Attention Framework (NAAF)

and minimize the cosine similarity between the unpaired ones. This multimodal way of training extracts important features from images and text resulting in the best image and text encoder architectures. While testing we give an image and several text sentences to the pre-trained CLIP. We measure the similarity between the image with each text sentence. We match the text which has the highest similarity score to the given image.

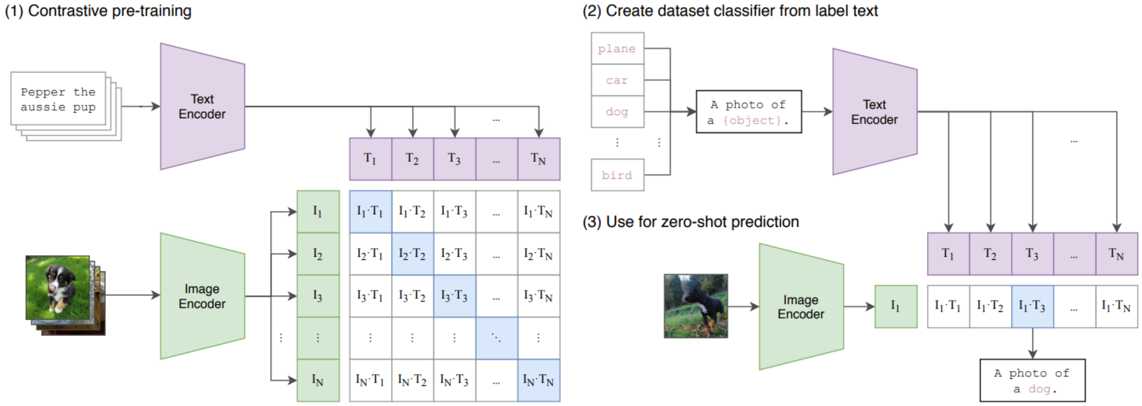

Figure 4: Summary of CLIP model.

3.4 Image-Text matching through Contrastive Language-Image Pretraining (Ours)

We are interested in image text matching using the pre-trained encoders of CLIP instead of Faster R-CNN and GRU/BERT to extract the best features and accurately learn the semantics of image and text. Given an image and sentences CLIP gives the probability for each matching. We use those probabilities as a measure of similarity. We assume the datasets are the ground truth in our analysis even though there might exist the best sentence for the given image in the other image-sentences pair. We evaluate the CLIP model's ability in differentiating the given sentences or retrieving the best-matched sentence by manually giving the sentences which are in a closer approximation to the given image. The CLIP representation in matching the text and image gives promising results and includes the Negative Attention part inside it. So, we compare the NAAF and CLIP-based image-text matching in this work.

3.5 Flickr30K dataset

The Flickr30k dataset is one of the standard benchmarks for Image-Text matching. The dataset contains 31,000 images and 155,000 captions (five for an image). In general 29,000 images of the dataset is used for training, 1000 images for validating and rest for testing the model. We evaluate the performance of CLIP model on the Flickr30K test dataset which is a completely new distribution to which CLIP is actually trained. Since CLIP is trained on 400 million images, we expect that it will generalize well on new data distributions and match the performance with respective benchmark models.

3.6 Data Sampling

The CLIP model is computationally expensive even for inference. To inference one sample, it took ≈ 4 seconds on our machine. We decided to get a subset of Flickr30K test data through random sampling without replacement. From the 1000 test samples of Flickr30K dataset, we sampled 10 images and their corresponding 50 captions and evaluated different recall rates (Recall@1, Recall@5, Recall@10). We repeated this process for 50 iterations to get the average rates and standard deviation of recall rates.

4 Experimental Results

All the experiments are run on Ubuntu 20.04 Machine. We evaluated the CLIP model with NAAF as baseline.

In our preliminary evaluation, we sampled two images (A, B, and C) and their corresponding labels from Flickr30K dataset and feed them to the CLIP model along with labels. We didn't use any baseline and evaluated CLIP based on human intuition.

For image A, we passed additional similar labels to the model to test the ability of CLIP. From Table 1, the label "Three men are watching the man in the cherry picker" has the highest probability. Even though all the labels are identical and appropriate, CLIP suggests the best possible label that contains more information about features. In addition, CLIP is able to differentiate between two labels "2 men/A man" standing inside a cherry picker, and gives more importance to "A man" since image A says the same. Similarly, a man standing inside a "cherry picker/utility truck" are differentiated significantly by giving higher importance to the cherry picker as the image contains a cherry picker. By observing the image, "3 men are watching the man in the cherry picker", got the highest probability, which is exactly the main content of the image.

Now we choose two images B and C of similar context as shown in Figure. 5 and their corresponding labels as mentioned in Table. 3 from the Flickr30K dataset. We pick the two best possible labels of image B and the three labels of image C which are close to image B labels. We then feed image B along with the selected 5 labels to the model. Table. 4

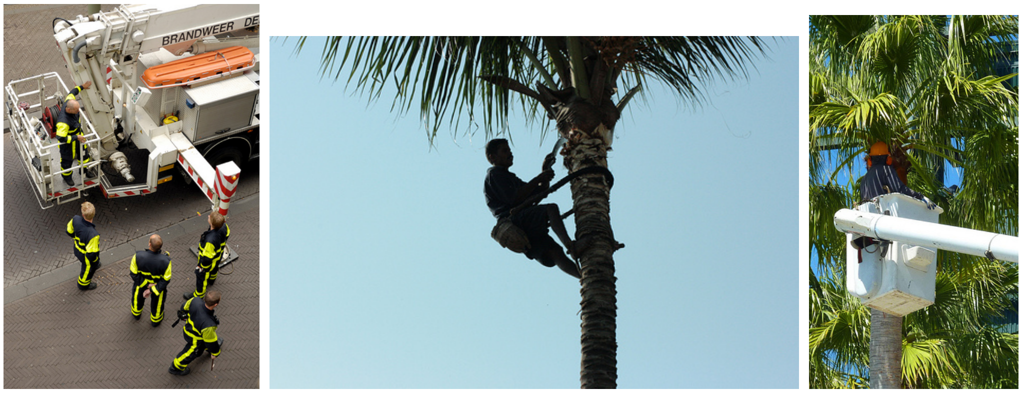

Figure 5: Three input Images A, B, and C for CLIP model. (from left to right)

Labels for Image A	**Probability**
5 men dressed in yellow and black suits are standing by a truck.	0.0003228294
Three men are watching the man in the cherry picker.	0.43304285
The four men are standing around the utility truck.	0.1624944
A man is standing inside a cherry picker.	0.27854863
A group of firemen standing.	0.038099032
2 men are standing inside a cherry picker.	0.054065842
A man standing inside a utility truck	0.03342639

Table 2: Output of CLIP for Image A.

gives the output of CLIP for image B. It shows that "On a clear blue day, a man had used a climbing device to shinny up what appeared to be a coconut palm tree." got more weightage compared to the other labels because the content of the image says exactly the same. With these observations, we believe that CLIP learns the accurate representations of images and text which resemble obtaining the highest similarity scores for the matched ones. To generalize this idea we show a detailed analysis in the final version of our work.

From the table 5 CLIP has better performance on par with the NAAF on average recall. While the NAAF has slightly better R@1 and R@10, however, CLIP has better R@5. Even though CLIP uses global-level semantic correspondence, it is performing on par with NAAF. Since CLIP is trained on 400 million natural Image-Text pairs, it is generalized well on the different datasets i.e. flickr30k.

Our final trails took ≈ 45 seconds per iteration and 10 minutes in total

5 Conclusion

In general, Image-Text matching should work in both directions, i.e, given image, it should predict a relevant caption, and for a given caption, it should select a relevant image. But in this study, we only considered the former direction. In future work, we plan to extend the study of the CLIP to the later one as well and model to another distribution, the MSCOCO dataset, and observe its generalizability power. In addition, we would like to combine the

Labels for Image B	Labels for Image C
Adult man climbing a huge palm tree.	Person in a cherry-picker working on a palm tree.
A young man is high up on a palm tree and is pruning it with a large knife.	A tree service employee is wearing safety equipment while up in a lift to trim some branches.
On a clear blue day, a man had used a climbing device to shinny up what appeared to be a coconut palm tree.	A worker is on a cherry picker in a palm tree.
A dark guy climbing a tree to get some coconuts down.	A man is in a bucket, looking at a tree.
a man is busy trimming a palm tree.	A man on a lift is inspecting a tree.

Table 3: Corresponding Labels of Figure A and B from Flickr30k dataset.

Label	Probability
On a clear blue day, a man had used a climbing device to shinny up what appeared to be a coconut palm tree.	0.97513956
Adult man climbing huge palm tree.	0.013960304
A tree service employee is wearing safety equipment while up in a lift to trim some branches.	0.0012578001
Person in a cherry-picker working on a palm tree.	0.0036812804
A worker is on a cherry picker in a palm tree.	0.0059610447

Table 4: Output of Figure C from CLIP

	Recall@1	Recall@5	Recall@10	Average Recall
CLIP	9.4	26.8	45.4	27.2
NAAF	10.4	25.6	45.6	27.2

Table 5: Comparison of Recal rate for CLIP and NAAF

features of CLIP and NAAF because they will contain both global and local features.

References

[1] CLIP - Connecting Text and Images. *https://openai.com/research/clip*

[2] K. Zhang, Z. Mao, Q. Wang and Y. Zhang, ”Negative-Aware Attention Framework for Image-Text Matching,” 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 2022, pp. 15640-15649, doi: 10.1109/CVPR52688.2022.01521.