

Baselines and Analysis

Geerisha Jain* Madhura Deshpande* Shaurya Singh* Shrey Madeyanda*
{geerishj, mvdesHPa, shauryas, smadeyan}@andrew.cmu.edu

1 Analysis

For our analysis, we will first evaluate the performance of our chosen baseline models using intrinsic metrics to measure the performance of the models’s components on fundamental tasks. We will then proceed to analyze some failure cases for our baseline models, and identify the reasons for those failures.

1.1 Intrinsic Metrics

These metrics are meant to evaluate different aspects of the baseline models’ performance, independent of an external application or context, and will be used to get insights into the models’ strengths or weaknesses.

Not all metrics will apply to all models, and the corresponding slots in the table are left blank accordingly.

1.1.1 Object Detection Score

This intrinsic metric will be used to evaluate the efficacy of multimodal models and our unimodal VisualBert-based model in detecting and classifying objects in images. We will be measuring the **performance of the visual encoders** for this task, since accurate object detection forms the bedrock upon which which coherent and contextually relevant image captions are generated.

We have calculated the score in the following manner - we take the objects detected by our models from within an image, concatenate these identified objects into a single string, and then perform a comparison with a similarly concatenated string of object types listed in the ground truth. For the comparison, we obtained a similarity score, using Spacy’s ¹ *en_core_web_lg* vectors, which will help us capture the semantic similarity when comparing object types, rather than looking for an exact match. For instance, we do not want ”person” and ”man” to be considered as entirely incorrect.

We have run the baseline models against a subset of the MS COCO (Lin et al., 2015) validation set. The subset consists of those images which have 10 or less objects listed in the ground truth, and we based this on the following factors:

- There are over 3800 images containing with 10 objects or less listed in the COCO validation annotations, out of a total of 5000. This makes our choice a very representative sample.
- The models were trained on a subset of 25000 images from the COCO training set (obtained through coco-minitrain (Samet et al., 2020)). While this is a rep-

*Everyone Contributed Equally – Alphabetical order

¹<https://spacy.io/>

Table 1: Intrinsic Metrics for Baselines

Methods	Object Detection	Lexical Density	Mean	Variance
Multilayer Perceptron (Text only)	-	-	0.352*	$2.4e - 2^*$
VisualBERT-based model (Image only)	0.473	-	0.501	$2.3e - 4$
Vera (Liu et al., 2023) (Text only)	-	-	0.501	$1.7e - 3$
Simple CNN-LSTM model	0.462	0.492	0.498	$2.7e - 4$
CNN-LSTM with Visual Attention (Xu et al., 2016)	0.445	0.469	0.502	$1.5e - 5$
CLIP ViT-B/32 (Radford et al., 2021)	-	-	0.700	$6.3e - 2$
RN50 (He et al., 2015)	-	-	0.504	$3.8e - 5$
RoBERTa-ViT-B-32 (Liu et al., 2019)	-	-	0.507	$7.9e - 5$
CLIP Enhance-FineGrained (Zhang et al., 2023)	-	-	0.505	$2.7e - 5$

representative set, it does have a bias towards images containing 10 objects or less.

- Some of the models being evaluated are rather basic, and their performance is expected to be poor when it comes to detecting a large number of objects (for instance, 30 objects).

The scores can be seen in Table 1. We have also plotted graphs showing the variation in the object detection scores for different numbers of objects in the ground truth. The aim was to explore if the performance changes as the number of objects that need to be detected increase.

Simple CNN-LSTM This was a simple multimodal implementation, consisting of a CNN encoder (visual backbone) and LSTM decoder. The visual backbone is being evaluated here, and it is making use of the Xception architecture (Chollet, 2017), obtained from the Keras library. This has been pre-trained on the ImageNet (Deng et al., 2009) dataset, and has gone through some finetuning on the MS COCO dataset for our purposes. The score obtained by this model is significantly lower than obtained in the original paper (where it was tested on the ImageNet dataset). The following are some insights we gain from this performance:

- The Xception model should have gone through additional finetuning on the COCO dataset. While we should expect its performance to generalize, there is bound to be a greater difference in the predicted and actual object labels from different datasets without adequate finetuning. This would be even more necessary for such a simple encoder-decoder architecture. Additionally, the features learned from ImageNet may not align well with those necessary for detecting and recognizing objects in the diverse contexts presented in COCO.
- The Xception model requires the images to be resized to the dimension 299x299. This downsampling leads to a loss in information as compared to the original image, which in turn gives a degraded performance for object detection. The difference in image quality can be seen in Figures 1 and 2.
- The Xception model is designed primarily for image classification, predicting a single label for the whole image. Adapting it to object detection involves not only classifying but also precisely locating multiple objects within an image, a task for which it wasn't specifically optimized,

even though it can be used for this.

- We can see from Figure 4 that the object detection score varies from 0.44 to 0.48, which is a quite a narrow range. This would be because the captions generated by the model as rather generic. For instance, "person", "animal", and "fruit", instead of "girl", "bear", and "banana". While these have some similarity, the generic nature will result in a generally lower score. There are also some interesting instances where the model gives a very specific category. For example, for the image in Figure 2, one of the predicted object categories was "Granny-Smith", which is a type of green apple, while the ground truth Figure 1 has simply listed "apple" as its category. We also see from the graph that the score generally drops as the number of objects in the ground truth increases from 1 to 4, though the decrease is not significant. The reason for the decrease is immediately apparent because of the increased complexity of the images. However, we also observe that an increase when the number of objects goes above 4. This is likely because of object types being repeated in the COCO annotations, combined with our concatenating of the object types in a single string for comparison, resulting in the slightly increased similarity score.
- There are some cases, as in Figures 1 and 2 where the probability of the first object classification by the model is very high, while the probability of the remaining predicted objects is extremely low. As we can see in the example, this is a correct prediction by the model, but included those values with low probability has had an adverse impact on our similarity score.

CNN-LSTM with Visual Attention This architecture used in this model is the one proposed by (Xu et al., 2016), and in our implementation, makes use of VGG-19 (Simonyan and Zisserman, 2015) as its visual backbone (encoder). We see a slightly reduced score for this model (see 1), which is expected as the Xception architecture beat VGG-19 when tested on the ImageNet dataset. The plot of the similarity scores against the number of objects listed in the ground truth can be seen in Figure 5. While this model makes use of attention mechanisms, it is being used in the decoder, and hence the insights we gain from the score and graph are similar to what we saw in the previous model (Simple CNN-LSTM).

VisualBERT-based model This was a custom image-captioning model built upon the foundations of VisualBert and pretrained on the Flickr8 Dataset. The finetuned version of the visual backbone is being evaluated here. The visual backbone is a Vision Transformer (ViT) i.e. a transformer encoder model (BERT-like) pretrained on a large collection of images in a supervised fashion, namely ImageNet-21k, at a resolution of 224x224 pixels. Further, the model was fine-tuned on ImageNet (also referred to as ILSVRC2012), a dataset comprising 1 million images and 1,000 classes, at a higher resolution of 384x384.

The evaluation of the model based on the object detection task reveal a consistent performance, with an average similarity score of approximately 0.476 and a narrow range from 0.45 to 0.49 as illustrated in Figure 6. While the model demonstrates stability in detecting and classifying objects across various images, the limited range of similarity scores suggests potential challenges in accurately discerning between objects in certain contexts. This observation indicates the need for ongoing improvements to enhance the model's precision and reliability for object detection tasks.

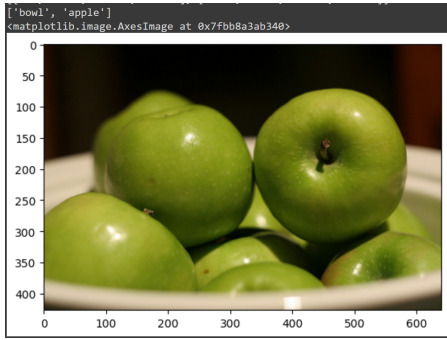


Figure 1: Original Image

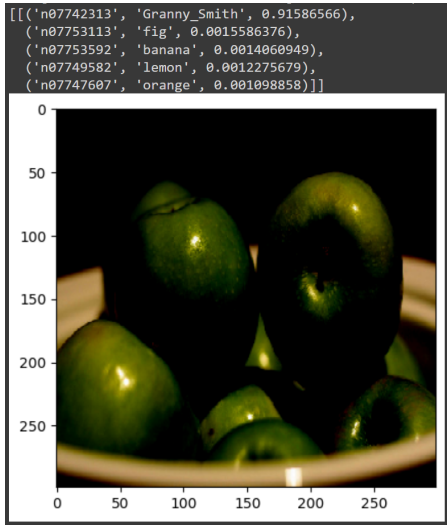


Figure 2: Resized image

This model, originally designed for image classification is bound to face challenges in object detection tasks due to its division of images into fixed-size patches. This fixed-size patch approach limits the model’s ability to accurately localize objects within images, as objects spanning multiple patches may not be effectively captured. Object localization in detection tasks requires understanding spatial relationships within objects, which the patch-based processing of ViT may struggle to capture. The lack of detailed spatial context and object boundaries within patches could lead to lower similarity scores in object detection, highlighting the need for modifications to en-

Figure 3: COCO image used for Caption Generated



hance ViT’s performance in tasks requiring precise object localization.

1.1.2 Lexical Density

Lexical density is a measure of the complexity of a text, quantifying the proportion of content words — nouns, verbs, adjectives, and adverbs — relative to the total number of words. In image captioning, it reflects the caption’s informativeness and linguistic richness, revealing the model’s capability to emulate human-like descriptive language.

In examining lexical density distribution of ground truth captions alongside those generated by a simple CNN-LSTM model and a CNN-LSTM model with visual attention, several insights can be gleaned about the efficacy and linguistic characteristics of the models in comparison to human standards.

As per Figure 7, for the ground truth captions, the distribution reveals a balanced variance with a symmetrical bell curve, peaking around a lexical density score of 0.5. This suggests that human captions tend to have a moderate and consistent level of lexical richness, without tending towards either extreme verbosity or oversimplification. The smoothness of the distribution also implies a diverse usage of language across different samples.

As illustrated in Table 1, the lexical density

Figure 4: Object Detection by Simple CNN-LSTM Model

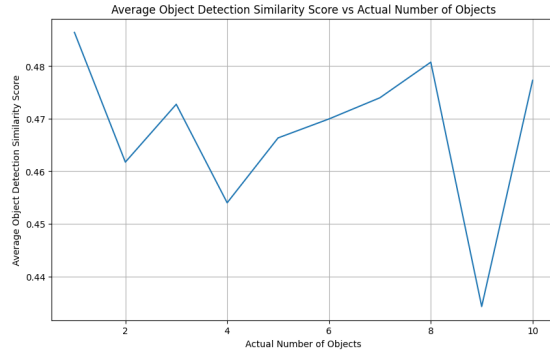


Figure 5: Object Detection by CNN-LSTM with Visual Attention

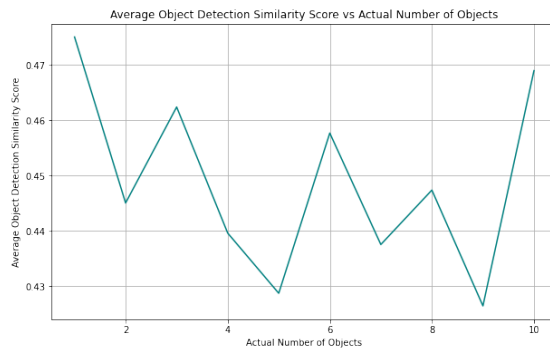
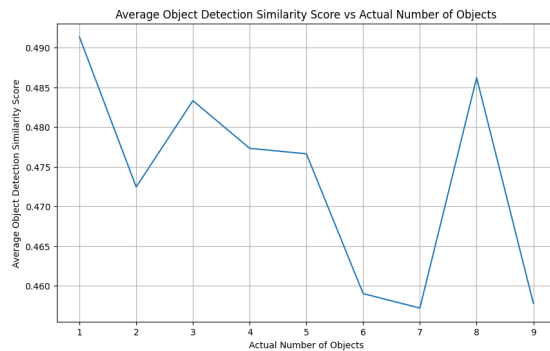


Figure 6: Object Detection by Vision Transformer



scores for the simple CNN-LSTM model are higher than those of the CNN-LSTM model with attention. However, a higher average lexical density score does not necessarily equate to a superior model performance. Therefore, we will now analyze the distributions to ascertain the implications of lexical density variations on the quality of the generated captions.

In analyzing the performance of our simple CNN-LSTM model for image captioning tasks, from Figure 9 it is observed that the model consistently generates captions with a moderate lexical density. This is evidenced by the prominent peak at the 0.5 lexical density score seen in the histogram provided. Such a distribution suggests a certain level of consistency in the model’s captioning, with a recurring use of a comparable number of content words across different images. However, this consistency in lexical density does not extend to the breadth of linguistic expression. The model’s captions lack the variability characteristic of human-generated text, as indicated by the narrow range of lexical density scores. The model’s current architectural configuration seems to limit its ability to produce a richer vocabulary and varied sentence structures, which are essential for capturing the nuanced descriptions often found in human captions.

Using an example caption generated by this model —“start of the kitchen with white walls and white walls and white walls”— for the image shown in Figure 3, illustrates a tendency towards verbosity that does not contribute to the informative content of the caption. Such repetitive verbosity points to a shortfall in the model’s language generation capabilities, emphasizing a single attribute unnecessarily without offering a comprehensive description of the image.

The CNN-LSTM model with visual attention, however, shows improvement over the simple model, with a distribution that attempts

to approximate the ground truth pattern, peaking closer to the lexical density score of the ground truth captions. Upon analyzing the lexical density histogram for our CNN-LSTM model with attention as given in Figure 8, we observe a more variable distribution than that of the simple CNN-LSTM model. The distribution is broader with several peaks, indicating the model’s capacity to produce captions with varying lexical densities. This variation suggests that the attention mechanism in the model allows for a more nuanced understanding of images, which translates into a diversity of caption complexity. However, despite this variability, the distribution lacks the smooth, bell-shaped curve seen in the ground truth data, which would indicate a well-balanced use of language across captions. The multiple peaks in the model’s histogram may point to inconsistencies in caption generation, where the model fluctuates between different levels of detail and complexity, potentially producing verbose captions in certain instances. The provided example caption, with its excessive repetition, further exemplifies the model’s inclination towards verbosity. It highlights a need to refine the model’s ability to generate concise and contextually relevant descriptions. While the attention mechanism enables the model to focus on various aspects of the image, the current training may not adequately penalize the generation of repetitive or superfluous content.

1.1.3 Similarity Score Distribution

We plot similarity score values for those models which are using similarity scores to select between positive and hard-negative captions. We believe this offers insight into the inherent ability of our models to differentiate between positive and hard-negative captions based on their similarity scores. It serves as a means to evaluate the discriminative power of the model in distinguishing between nuanced categories, thereby assessing its ability to capture subtle

differences in semantic meaning. This metric is particularly essential for tasks aimed at improving model compositionality, where the accurate representation of complex relationships and semantic nuances is crucial. Here, we are not concerned with the accuracy of the classifications, but rather it’s ”confidence”.

An important thing to note here is that we have used the terms **Category A** and **Category B** to represent the positive and hard-negative captions. A well-separated distribution with minimal overlap indicates that the model effectively captures the differences between categories, while a more blended distribution suggests potential challenges in discrimination.

For models - RN50 Figure 10, RoBERTa-ViT-B-32 Figure 11, Simple CNN-LSTM Figure 12 and CNN-LSTM with Visual Attention Figure 13 model we found that the data points in the scatter plot were mostly distributed along the **Boundary of Uncertainty**. From this we can draw the insight that the similarity scores generate by using both Category A and Category B are closer to each other, and there’s only a marginal difference that helps us to distinguish whether an output caption is more similar to Category A or to Category B. These models are not able to categorize with a high degree of confidence.

For Vera Figure 14, we can see that it is able to categorize the instances with greater confidence. However, we need to keep in mind that this is a language-only model, which assigns plausibility scores to texts, and does not use image features at all. Also, the captions provided are from SugarCrepe (Hsieh et al., 2023), and have very subtle differences between the different categories. The text-only unimodal baselines in our analysis are present only to highlight the importance of avoiding blatant discrepancies between the positive and hard-negative captions. If such discrepancies are present, the language-only models have been

shown to give a performance equal to or better than the vision-language models (Hsieh et al., 2023).

1.1.4 Probability Distribution of Selecting Category A

For our analysis, we will focus on the probabilities assigned by the models for Category A. Using this intrinsic metric, we aim to glean insights into model confidence along with its predictive confidence.

Two of our models - Multilayer Perceptron (see Figure 15) and CLIP (see Figure 19) were directly returning probability values for whether Category A is correct or Category B. However, other models were returning similarity scores. For converting these similarity scores to probability values, we apply a softmax function over them. The value that we get for Category A is then treated as the probability of choosing Category A as the correct caption.

We also find the mean and variance values for these distributions (see Table 1. These help us understand how are model is performing in average and how the confidence of the model varies over the dataset.

We see that for most of our baseline models - VisualBERT (see Figure 16), Vera (see Figure 17), Simple CNN-LSTM (see Figure 18), RN50 (see Figure 20), RoBERTA (see Figure 21) and CNN-LSTM with Visual Attention - have high uncertainty with mean of the distribution at or around 0.5. We correlate this with probability distribution graphs and see that the distributions also peak around the value of 0.5, indicating that the models are not able to discriminate as strongly between the two categories.

Since the MLP model (see Figure 15) operates in a way entirely different to other models, by classifying each instance in the test dataset as either Category A or Category B. It does so by calculating sigmoid values. Due to this

inherent difference, we see that the number of instances over which we calculate probability distribution doubles, however, the values of the mean and variance of this distribution still mean the same. For this particular baseline we see that our model is more varied in its confidence, and that it mostly predicts instances as not belonging to Category A.

For CLIP (see Figure 19) and CLIP Enhanced (see Figure 22), we see that the models mostly predicts that instances belong to Category A. However, we see that CLIP is more certain in its decision, whereas that is not the case with CLIP Enhanced.

Figure 7: Lexical density of Ground Truth Captions

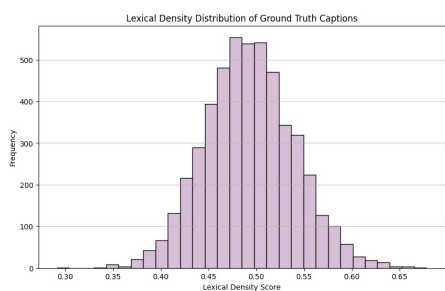


Figure 8: Lexical density of CNN-LSTM with Visual Attention

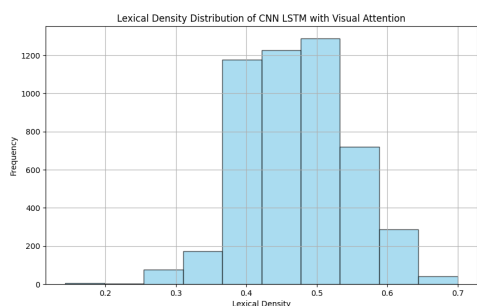


Figure 9: Lexical density of CNN-LSTM

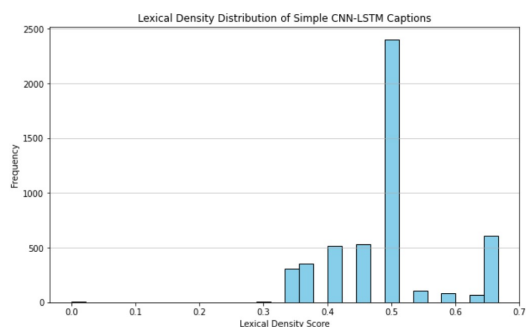


Figure 10: Similarity Scores for RN50

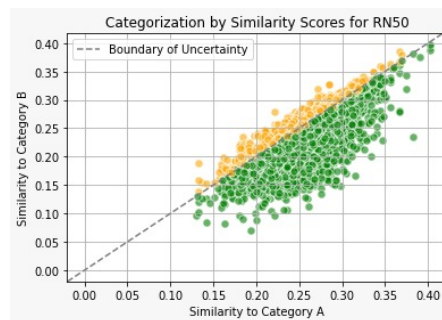


Figure 11: Similarity Scores for RoBERTa-ViT-B-32

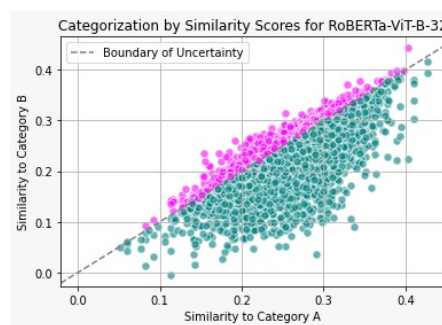


Figure 12: Similarity Scores for Simple CNN-LSTM model

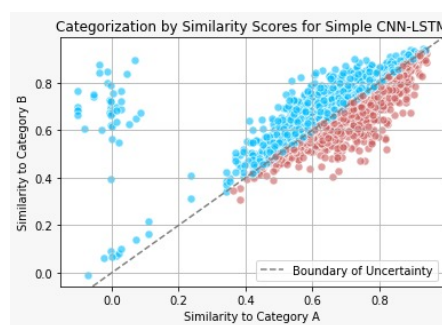


Figure 13: Similarity Scores for CNN-LSTM with Visual Attention model

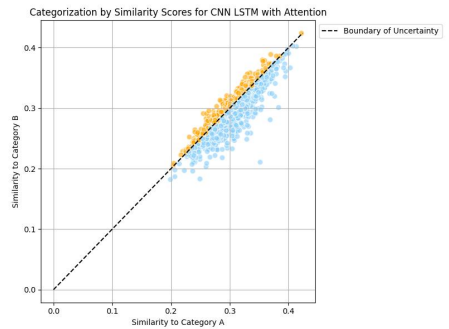


Figure 14: Similarity Scores for Vera

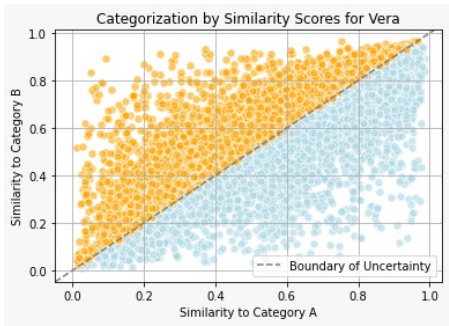


Figure 15: Probability Distribution of Selecting Category A (MLP)

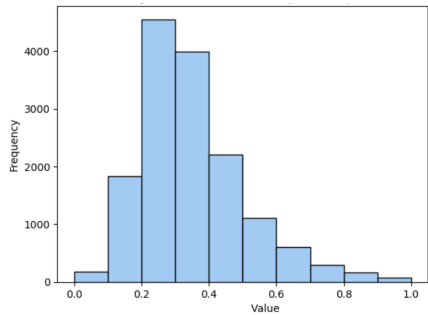


Figure 16: Probability Distribution of Selecting Category A (VisualBERT-based)

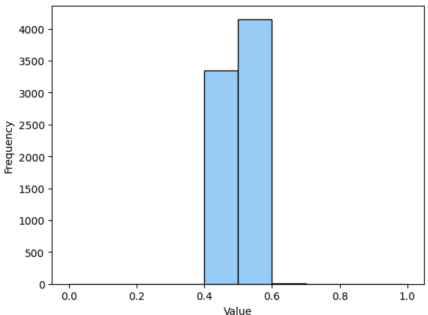


Figure 17: Probability Distribution of Selecting Category A (Vera)

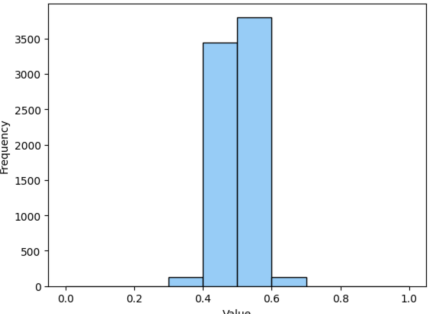


Figure 18: Probability Distribution of Selecting Category A (Simple CNN-LSTM)

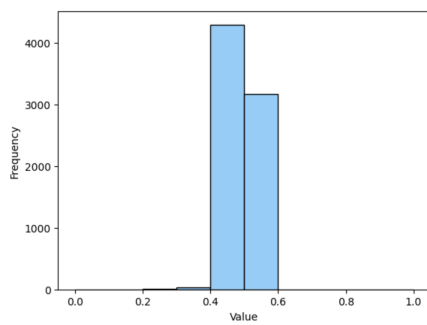


Figure 19: Probability Distribution of Selecting Category A (CLIP ViT-B/32)

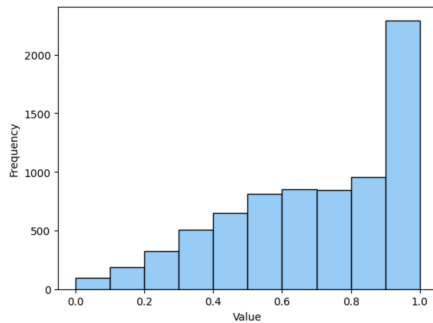


Figure 20: Probability Distribution of Selecting Category A (RN50)

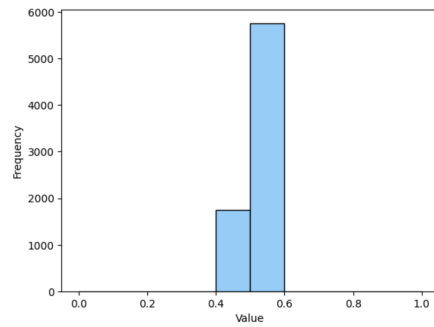


Figure 21: Probability Distribution of Selecting Category A (RoBERTa-ViT-B-32)

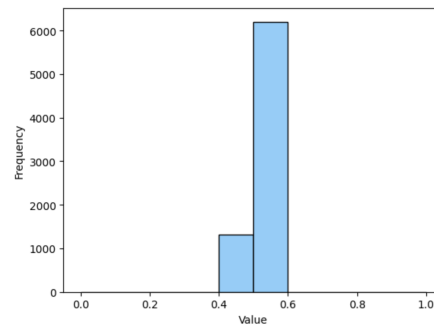
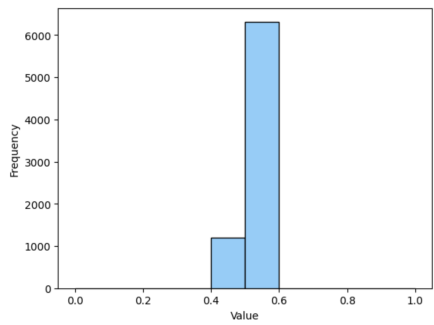


Figure 22: Probability Distribution of Selecting Category A (CLIP Enhanced-FineGrained)



1.2 Qualitative Analysis and Examples

We now proceed to evaluate the some scenarios where the baseline models gave an incorrect output. The failures are with respect to the main task i.e. measuring the compositional understanding of vision-language models. We measure this by checking the accuracy with the model selects positive captions when compared against hard-negative captions (please refer to Report 2 for a detailed description of the task).

Failure cases for all the baseline cases can be seen in Figure 23. We have provided the corresponding image, the positive and hard-negative captions from which the model is expected to choose, and the scores assigned by the models to each of those captions. Since these are failure cases, the scores for the negative captions will be higher in all the instances. We should note that some models use probability values rather than similarity scores (like the MLP model, CLIP and CLIP Enhanced-FineGrained). We have chosen to group them all under "Scores" as well, for readability.

In all the cases, we can see that there are very subtle differences between the positive and hard-negatives, and therefore the models in these cases are unable to differentiate between them with any confidence. This can also be seen in the very small difference between the positive and negative scores in most cases.

An interesting aspect, is that these low difference can be seen in the more sophisticated RN50 and RoBERTa-ViT-B-32 (for both correct and incorrect classification).




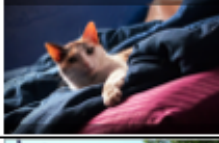



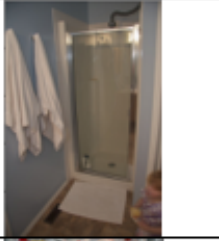

The failures of the text-only models, MLP and Vera, are expected, since they are only looking at the plausibility of the statements, and the hard-negative captions are not blatantly unreasonable.

1.3 Insights

Our analysis of the baseline models' performance has helped clarify our path forward. It is evident that we need auxiliary loss functions for the image and text encoders in our proposed model (please refer to Report 2) that will minimize the distance between attributes and their corresponding entity nouns, as well as maintain a minimum semantic distance between positive and hard-negative caption pairs. This will help avoid scenario where our model might select the correct caption with a probability hovering around chance i.e. improve its confidence.

The importance of a good visual encoder has also been emphasized by our analysis. We would need to select encoders that provide embeddings that can differentiate between additional, swapped or replaced attributes and entities. We can see this difference between encoders by looking at the higher confidence with which CLIP ViT-B/32 makes its classification as opposed to an encoder like RN50. We need to make sure that we do not just focus on the overall accuracy of our model's performance, but also it's sub-tasks.

Figure 23: Failure Cases for Baseline Models

Model	Image	Positive Caption	Negative Caption	Positive Score	Negative Score
Multilayer Perceptron (Text only)		A man in black shirt standing in field with baseball mitt.	A man in baseball shirt standing in field with black mitt.	0.446	0.554
Simple CNN-LSTM		A table topped with a cake covered in berries next to a plate of sandwiches.	A crystal table topped with a cake covered in berries next to a plate of sandwiches.	0.618	0.622
CNN-LSTM with Visual Attention		The extremely small car is parked behind the bus	The bus is parked behind the extremely small car.	0.271	0.294
RN50		A kitten on a bad with its arm stretched out toward the camera	A kitten with a hat on its head is on a bed with its arm stretched out toward the camera.	0.223	0.241
RoBERTa-ViT-B-32		Two young women are washing two motorcycles with hoses.	Two young women are washing two gilded motorcycles with hoses.	0.327	0.330
Vera		a coin meter that has paint all over it	A coin meter that is clean all over it	0.369	0.558
CLIP ViT-B/32		A tall vase with red and white tulips in water.	A tall vase with red and white tulips next to a bowl of water.	0.492	0.508
VisualBERT-based model (Image only)		Blue bathroom with two white towels hanging by the shower.	White bathroom with two blue towels hanging by the shower.	0.190	0.194
CLIP Enhance-FineGrained		Child's bed and colorful quilt surrounded by blue plastic walls.	Child's bed and monochromatic quilt surrounded by blue plastic walls.	0.447	0.552

2 Team member contributions

Shrey Madeyanda contributed to error analysis for multimodal and competitive baselines, plot generation, failure cases, and overall writing of the report.

Shaurya Singh contributed to metric analysis for one unimodal baseline, similarity score metric, probability distribution metric, plot generation and overall writing of the report.

Geerisha Jain contributed to metric analysis for one multimodal baseline, the relevant failure case, and lexical density part of the report

Madhura Deshpande contributed to metric analysis for one unimodal baseline, one competitive baseline, the relevant failure cases and the Similarity Score Distribution part of the report.

References

- François Chollet. 2017. [Xception: Deep learning with depthwise separable convolutions](#).
- Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. [Imagenet: A large-scale hierarchical image database](#). In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. [Deep residual learning for image recognition](#).
- Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Anirudha Kembhavi, and Ranjay Krishna. 2023. [Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality](#).
- Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. [Microsoft coco: Common objects in context](#).
- Jiacheng Liu, Wenya Wang, Dianzhuo Wang, Noah A. Smith, Yejin Choi, and Hannaneh Hajishirzi. 2023. [Vera: A general-purpose plausibility estimation model for commonsense statements](#).
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [Roberta: A robustly optimized bert pre-training approach](#).
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. [Learning transferable visual models from natural language supervision](#).
- Nermin Samet, Samet Hicsonmez, and Emre Akbas. 2020. [Houghnet: Integrating near and long-range evidence for bottom-up object detection](#).
- Karen Simonyan and Andrew Zisserman. 2015. [Very deep convolutional networks for large-scale image recognition](#).
- Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. 2016. [Show, attend and tell: Neural image caption generation with visual attention](#).
- Le Zhang, Rabiul Awal, and Aishwarya Agrawal. 2023. [Contrasting intra-modal and ranking cross-modal hard negatives to enhance visiolinguistic compositional understanding](#).