Less is More: Recreating a simple model to answer VQA

Rana Mohamed Barakat, Mohab Mohamed Ali

***Abstract*—Most architectures used for multi modial problems such as visual question answering rely on transformers or other costly methods for training and evaluation. In this paper, we present a simple solution that utilizes pre trained models to achieve significant accuracies with a simple model that doesn't rely on long training time or expensive hardware based on an architecture used by the University of the Bundeswehr Munich[1].**

I. Problem

Visual question answering is where an image question pair is provided and the model outputs an answer to the question with the context of the image. This is usually to help the vision impaired and is a great step towards a more accessible world powered by ai. This task can be achieved in many ways, with one of the most popular ones being transformers considering their recent advancements and magnitude of applications. However, transformers suffer from large complexity and long training times. We address this by utilizing openAI's pretrained CLIP[2] model for image and text encoding and a simple classifier to find a suitable answer to the question.

II. Dataset

We used VizWiz's VQA dataset that has a few flaws. The dataset was composed of crowd sourced answers for image question pairs that were provided by the vision impaired and thus the answers are not always correct and are not always consistent. These flaws must be addressed in preprocessing to be able to produce good results.

III. Vocab selection

For our vocabulary, we tried a few strategies including selecting the answer that most corresponds to a caption of the image using CLIP's ViT-B/32, however that produced too many classes which resulted in a high error rate in the validation set. We also tried the first most common answer, and most common with draws resolved by the previous selection method. However, in the end we adapted the strategy used by the original authors of this architecture, in which we selected the most common answer per image/question pair and draws are resolved by selecting the answer that most resembles the rest of the answers using pair wise Levenshtein distance which proved to produce the best results and we ended up with a vocab size of 4690.

IV. Input and Output

After our vocab set was created, we now tackle the input and output shapes. We started by augmenting the image by rotating 90 and -90, encoding each image, then finding the mean. That would be our image representation. We then concatenated the image encoding with the question encoding all of which was done by CLIP with freezing the model to minimize train and evaluation time. The output on the other hand was a label binarized set created by sklearn's LabelBinarizer that took all 4690 classes and returned them as one-hot-encoded vectors. We decided to use this encoding as opposed to label encoding to make sure not to add any bias to the training set. The model also outputs the answer type which we use as an auxiliary loss to help improve training.

V. Model

In training, we used cross-entropy loss with both outputs and Adam optimizer with a learning rate of 1e-4 and wight decay of 1e-5 to avoid overfitting. The model is a simple classifier where the input first goes through a fully connected layer that reduces dimensionality, then the output is normalized and goes through a high dropout layer. The output is then passed to a fully connected layer

that maps directly to the vocab size.

Answer type gate: The reduced dimensionality output is also passed to another fully connected layer that outputs the answer type as a one-hot-encoded vector then passes through another fully connected layer that maps it back to the vocab size. This is multiplied by the answers produced before to provide the final output. This answer type gate helps the model identify the answer by adjusting the classification based on the predicted answer types.

TABLE I
VALIDATION ACCURACIES

	Answer acc	Answer type acc	Answerability acc
ViT-L/14	%50.46	%76.17	%83.20
RN50x64	%50.87	%77.88 %	%83.01
ViT-L/14@336px	%50.68	%76.05	%83.39
RN50x16	%49.57	%76.17	%82.70
Ensemble	%51.47	%77.94	%83.64

VI. RESULTS

With our architecture and vocab, we achieved a final accuracy of %51.6 in answers, %76.2 in answer types, and %82.60 in answerability on the test set. Training time took 4 minutes for 70 epochs per model on Kaggle's P100 which was sufficient for convergence. The final model is an ensemble of features extracted by encoding the image/question pairs using each of the following clip models: ViT-L/14, RN50x64, ViT-L/14@336px, RN50x16. Each model is trained indepentently then upon evaluation each model takes its respective encoing as input and the mean of all outputs is the final prediction for both answers and answer types.

VII. CONCLUSION

This problem, along many other multi modial problems can be solved with simpler models that don't require the enormous time and complexity requirements that transformers impose. We have shown that encoding the images and questions with pretrained models can produce significant results. This is a step towards sompler more powerful models that are accessible and easy to implement.

REFERENCES

[1] F. Deuser, K. Habel, P. J. Rösch, and N. Oswald, "Less is more: Linear layers on clip features as powerful vizwiz model," *arXiv preprint arXiv:2206.05281*, 2022.

[2] A. Radford, J. W. Kim, C. Hallacy, *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*, PMLR, 2021, pp. 8748–8763.