Multi-Modal Image Captioning

Hiba Ahsan hahsan@umass.edu
Nikita Bhalla nbhalla@umass.edu
Daivat Bhatt dbhatt@umass.edu
Kaivan Shah kaivankumars@umass.edu

1 Problem Statement

Image Captioning is now a common feature available on many social media platforms like Facebook and productivity platforms like Microsoft Power Point. This service has helped people who are blind to learn about images they take and also to make sense of images they encounter in digital environments. Unfortunately the current state of the art models are built using large, publically available, crowd-sourced data sets which has been collected and created in a contrived setting. Thus these state of the art models perform poorly on images clicked by blind people which are part of the VizWiz dataset. Gurari et al. (17) have suggested that 21% of questions visually-impaired people asked about an image were related to the text in it. This makes it more important to improvise the SOTA models using datasets which focuses on objects as well as text in the images to make a coherent sense.

With the availability of large labelled corpora, progress in image captioning has seen a steady increase in performance and quality and reading scene text (OCR) has matured. However, while OCR only focuses on written text, state-of-the-art image captioning models focus only on the visual objects when generating captions and fail to recognize and reason about the text in the scene. Incorporating OCR tokens into a sentence is a challenging task, as unlike conventional vocabulary tokens which depend on the text before them and therefore can be inferred, OCR tokens often can not be predicted from the context and therefore represent independent entities. Predicting a token from vocabulary and selecting an OCR token from the scene are two rather different tasks which have to be seamlessly combined to tackle this task.

Through our research we intended to leverage OCR and integrate it with the SOTA image captioning models to get better performance on images with text in them.

2 What you proposed vs. what you accomplished

- ~~Collect and preprocess VizWiz Captions Dataset~~
- ~~Generate OCR tokens and word embeddings for all images~~
- ~~Build and train vanilla AoANet and BUTD on VizWiz dataset and examine its performance~~
- ~~Build and train extended vocabulary model and inspect results~~
- ~~Improve the extended vocabulary model to perform better than baseline~~
- ~~Implement pointer generator network for both AoANet and BUTD models~~
- ~~Improve pointer generator network to beat baseline performance~~:
- *Compare to the current M4C Implementation which uses OCR tokens for image captioning*: We had also proposed to compare our proposed models to the M4C captioner introduced in Sidorov et al. (27) but because of the large amount of time consumed in the back and forth with the MMF community, we priortized our efforts for other tasks.

3 Related work

Automated image captioning has seen a significant amount of recent work. The task is typically handled using an encoder-decoder framework; image-related features are fed to the encoder and the decoder generates the caption (8),

(33), (11). Language modeling based approaches have also been explored for image captioning (21), (14). Apart from the architecture, image captioning approaches are also diverse in terms of the features used. Visual-based image captioning exploit features generated from images. Multi-modal image captioning approaches exploit other modes of features in addition to image-based features such as candidate captions and text detected in images (31), (19).

The task we address deals with captioning images specifically for the blind. This is different from traditional image captioning due to the authenticity of the dataset compared to popular, synthetic ones such as MS-COCO (10) and Flickr30k (24) . The task is relatively less explored. Previous works have solved the problem using human-in-the-loop approaches (Air), (BeS), (TapToSee) as well as automated ones (MS), (FB). A particular challenge in this area has been the lack of an authentic dataset of photos taken by the blind. To address the issue, Gurari et al. (17) created VizWiz-Captions, a dataset that consists of descriptions about images taken by people who are blind.

We explored copy mechanism in our work to aid copying over OCR tokens from the image to the caption. Copy mechanism has been typically employed in textual sequence-to-sequence learning for tasks such as summarization (26), (16). It has also been used in image captioning to aid learning novel objects (32), (23).

Also in Sidorov et al. (27), M4C model has shown to gain significant performance improvement over baseline by using OCR tokens.

4 Dataset

The Vizwiz Captions dataset (17) consists of over 39,000 images originating from people who are blind that are each paired with five captions. Our motive was to understand what information needs to be encompassed into a caption so that it is relevant and helpful to people with visual impairments. The authors explain that popular datasets are inferior in building models that provide descriptive and adequate information from captions. Moreover, the images are in these datasets are 'ideal' in that they are of a superior quality than on average. Most user-clicked images are blurry.

As an example, we see in Figure 1 an image that exists in the VizWiz Captions dataset. We can see the quality of the image being grossly out of the image-quality distribution of datasets such as Flickr30k (34) or COCO Captions (10). However, we routinely find ourselves clicking blurry images such as tehse. As difficult it is for any human to come up with a descriptive caption for images like this, the aspect most important here is to appreciate the groundedness of this dataset, and the need to build more robust models. This image is from the training set. The five captions associated with this image are:

- A bottle of pancake syrup is on the table.
- Log Cabin syrup bottle in the foreground of a brightly sunlit kitchen and indistinguishable items in the background.
- Quality issues are too severe to recognize visual content.
- A bottle of syrup, some is missing in a kitchen with a window.
- Quality issues are too severe to recognize visual content.

We can see the variability in the kind of captions that are generated. Not only do they differ in the amount of information contained (rightly so, because knowing the brand of the syrup requires an existing knowledge base that might not be available to all annotators), but also in the very existence of a caption. Some annotators were not able to come up with a reasonably detailed caption.

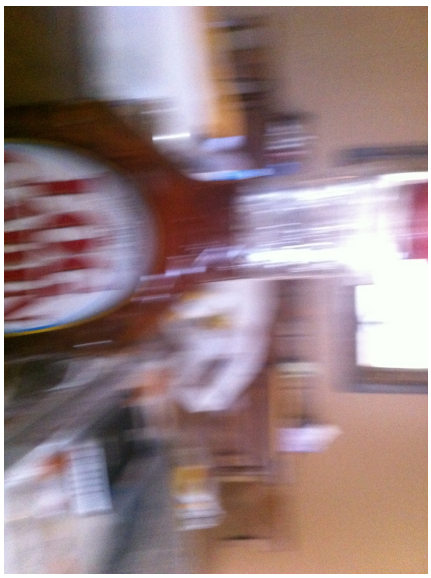

Figure 1: An image labelled as Low Quality Image (LQI)

Moreover, we can see in Figure 3 as to how the main object of the image captioning (i.e. the can)

is hidden from the view. This image is from the training set. The five captions associated with this image are:

- A person holding a food can with the rear of the label facing forward.
- A White man's hand holding a Vitamin Bottle.
- canned food held by a man's fingers with the thumb visible
- imagine how you would describe this image on the phone to a friend.
- the photographer's hand holding a round bottle with a black lid.

In the above list of captions, we find some degree of commonality in the details described in the captions. However, most of the detailing is tangential largely due to the absence of the subject of the image - the can.

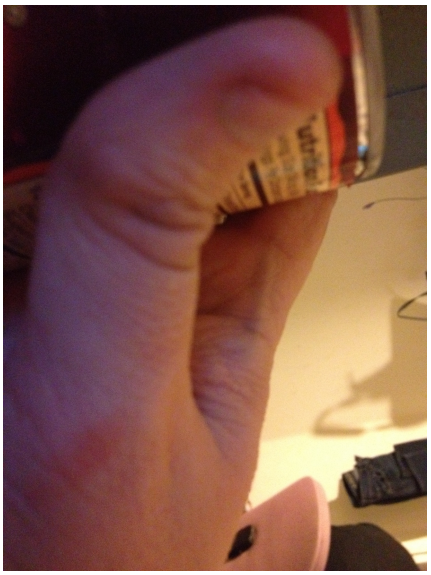

Figure 2: An image labelled as Insufficient Visual Evidence (IVE)

The total training images present in this dataset are 24909. The average length of a caption in the training set was 11 words. Note that each image has 5 captions associated with it.
The total validation images are 5601. The average length of a caption in the validation set was also 11 words.
The test set has 8000 images.

We performed an OCR detection task to implement our pointer-generator network. The average OCR tokens detected were about 4. The median was 5. Note that the OCR detection, because its a part of the caption generator pipeline, was also performed on the test set. The distribution of OCR tokens is shown below:

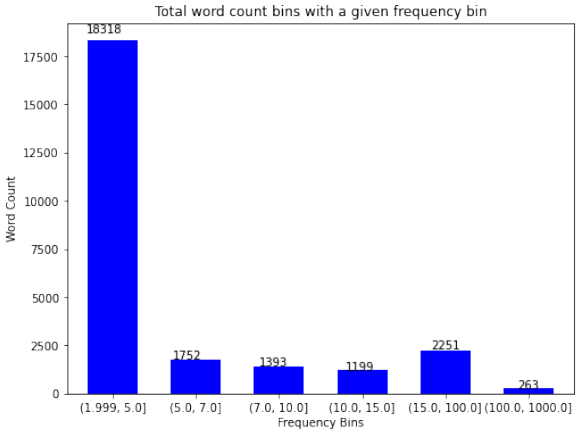

Figure 3: Word counts and frequency bins of all OCR tokens

On the VizWiz Dataset Browser (9), we put multiple filters to capture the issues with the dataset. When the filter 'LQI - Low quality image', we find that there are a total of 9,908 images that match the description. Note that this includes all images from the Training and Validation set.
When the filter 'IVE - Insufficient visual evidence - answer not present in the image' is applied, we get a total of 12,118 images.
This means that almost a significant portion of the images used for training *or* validation contain some form of impediments in generating a caption.
The dataset is available here and contains all the necessary files required to run the code.

4.1 Data preprocessing

For the OCR tokens detected in each image, we used the English stopwords list compiled here. This list of OCRs was then appended to the vocabulary of the model.

5 Baselines

Our project has two baseline models: Attention on Attention (AoA) (20), Bottom-Up Top-Down Attention for Image Captioning(BUTD) (7). They are explained below.

5.1 Attention on Attention for Image Captioning (AoANet)

The output of a sequence-to-sequence learning model with an attention mechanism conditions on the attention result. However, there are cases when the attention result may not be useful to the decoder either because the attention module didn't do well or because there genuinely is no worthwhile information in the candidate vectors that could be useful to the decoder. To address these issue, the authors of AoANet propose extending the conventional attention mechanism to account for the relevance of the attention results with respect to the query. They do this by adding another attention.

An attention module $f_{att}(Q, K, V)$ operates on queries Q, keys K and values V. It measures the similarities between Q and K and using the similarity scores to compute a weighted average over V.

$$a_{i,j} = f_{sim}(q_i, k_j), \alpha = \frac{e^{a_{i,j}}}{\sum_j e^{a_{i,j}}}$$

$$\hat{v}_i = \sum_j \alpha_{i,j} v_{i,j}$$

$$f_{sim}(q_i, k_j) = \text{softmax}(\frac{q_i k_j^T}{\sqrt{D}}) v_i$$

where $q_i \in Q$ is the i_{th} query, $k_j \in K$ and $v_j \in V$ are the j_{th} key/value pair, f_{sim} is the similarity function, D is the dimension of q_i and $\hat{v}_i$ is the attended vector for query q_i.

The AoANet model introduces a module AoA which measures the relevance between the attention result and the query. The AoA module generates an "information vector", i and an "attention gate" g, both of which are obtained via separate linear transformations, conditioned on the attention result and the query:

$$i = W_q^i q + W_v^i \hat{v} + b^i \quad (1)$$

$$g = \sigma(W_q^g q + W_v^g \hat{v} + b^g) \quad (2)$$

where $W_q^i, W_v^i, b^i, W_q^g, W_v^g, b^g$ are parameters. AoA module then adds another attention by applying the attention gate to the information vector to obtain the attended information $\hat{i}$.

$$\hat{i} = g \odot i \quad (3)$$

The AoA module can thus be formulated as:

$$AoA(f_{att}, Q, K, V) = \sigma(W_q^g Q + W_v^g f_{att}(Q, K, V) + b^g) \odot (W_q^i Q + W_v^i f_{att}(Q, K, V) + b^i) \quad (4)$$

The AoA module is applied to both the encoder and decoder. In the encoder, traditionally, a set of features $A = a_1, a_2, ..., a_k$ is extracted from a CNN or R-CNN based network and directly fed to the decoder. We refine the encoder with the AoA module, denoted as AoA^E:

$$A' = \text{LayerNorm}(A + AoA^E(f_{att}, W^{Q_e} A, W^{K_e} A, W^{V_e} A)) \quad (5)$$

where $W^{Q_e}, W^{K_e}, W^{V_e} \in R^{D \times D}$ are linear transformation matrices and LayerNorm refers to layer normalization. We used multi-head attention with 8 heads for f_{att}.

The decoder generates a caption y with the refined feature vectors A'. A context vector c_t is computed to attain the conditional probabilities over the vocabulary:

$$P_{vocab} = P(y_t | y_{1:t-1}) = \text{softmax}(W_p c_t) \quad (6)$$

where $W_p \in R^{D \times |\Sigma|}$ is the weight parameter to be learnt and $|\Sigma|$ is the size of the vocabulary.

The LSTM in the decoder is fed the embedding of the input word at the current time step, and a visual vector $(\bar{a} + c_{t-1})$, where $\bar{a}$ is the mean pooling of A' and c_{t-1} is the context vector in the previous time step.

$$x_t = [W_e \Pi_t, \bar{a} + c_{t-1}] \quad (7)$$

$$h_t, m_t = \text{LSTM}(x_t, h_{t-1}, m_{t-1}) \quad (8)$$

where $W_e \in R^{E \times |\Sigma|}$ is the word embedding matrix and Π_t is the one-hot encoding of the input word w_t. c_t is obtained by adding the AoA module to the decoder, denoted as AoA^D:

$$c_t = AoA^D(f_{att}, W^{Q_d}[h_t], W^{K_d} A', W^{V_d} A') \quad (9)$$

where $W^{Q_d}, W^{K_d}, W^{V_d} \in R^{D \times D}$, $h_t, m_t \in R^D$ are the hidden states of LSTM and h_t serves as the attention query.

The model is trained by minimizing the cross-entropy loss:

$$L(\theta) = -\sum_{t=1}^{T} \log(p_\theta(y_t^* | y_{1:t-1}^*)) \quad (10)$$

where $y_{1:T}^*$ is the ground truth sequence.

5.2 Bottom-Up Top-Down Attention for Image Captioning

The encoder in the baseline BUTD model uses Visual Attention to extract all objects and salient regions from the image with no context of the task. Faster R-CNN pretrained on ImageNet (12) and Visual Genome (22) is used for this task.

Given the attention candidates and task context, the decoder calculates the weights of the candidates.It consists of two LSTMs layers used as the top down visual attention model and the language model respectively.

The first LSTM layer is the top down attention LSTM. The input vector to this layer at each time step is a combination of mean pooled image features v, word embeddings of the previously generated word π and previous output of the Language LSTM. These inputs provides the model with the overall context of image, partial caption output generated so far and the maximim content regarding the state of language LSTM.

$$x_t^1 = [h_{t-1}^2, v, W_e \pi_t] \tag{11}$$

where $W_e \in R^{E \times |vocab|}$ is a word embedding matrix for a vocabulary $vocab$, and π_t is one-hot encoding of the input word at timestep t.

$$h_t = LSTM(x_t, h_{t-1}) \tag{12}$$

where x_t is the LSTM input vector and h_t is the LSTM output vector.

After generating the output h_t^1, normalized attention weight $\alpha_{i,t}$ is calculated for each image feature.

$$a_{i,t} = w_a^T tanh(W_{va} v_i + W_{ha} h_t^1)$$

$$\alpha_t = softmax(a_t) \tag{13}$$

where $W_{va} \in R^{HXV}$, $W_{ha} \in R^{HXM}$ and $w_a \in R^H$ are learned parameters. A convex combination of input features serves as the input for the second layer:

$$\hat{v}_t = \sum_{i=1}^{K} \alpha_{i,t} v_i \tag{14}$$

The input to the language model LSTM consists of the attended image feature V, concatenated with the output of the attention LSTM.

$$x_t^2 = [\hat{v}_t, h_t^1]$$

AT each time step t, using the conditional distribution, possible output word is concluded:

$$p(y_t | y_{1:t-1}) = softmax(W_p h_t^2 + b_p)$$

$W_p \in R^{\Sigma XM}$ and b_p are learned weights and biases.

The distribution over the complete output sequence is calculated as follow:

$$p(y_{1:T}) = \prod_{t=1}^{T} p(y_t | y_{1:t-1})$$

The model is trained by minimizing the cross entropy loss where the ground truth sequence is $y_{1:T}^*$

$$L_{XE}(\theta) = -\sum_{t=1}^{T} \log(p_\theta(y_t^* | y_{1:t-1}^*)) \tag{15}$$

The vanilla BUTD model (without text-based features) will serve as the baseline.

6 Approach

Both the approaches defined below were implemented on both the Attention on Attention net and the Bottom-Up Top-Down Attention. We talk about our experiments, results and errors in subsequent sections.

6.1 Extending Feature Set with OCR Token Embeddings

Our first extension to the models was to increase the vocabulary available to the model, make necessary changes so that these changes are propagated through the network and the model is able to access these words and observe the results. For starters, we use an off-the-shelf text detector available on Google Cloud Platform's vision API (GGL). After obtaining the text for each image, we use a standard stopwords list as part of necessary preprocessing. Note that we would like the model to have as few candidate words for a timestep to improve its confidence and performance. Moreover, we expect certain words to be already present in the vocabulary when building it using the ground truth captions. Using the 'cleaned' text for each image, we generate a vector for each word using a pretrained base, uncased BERT (13) model. We build a matrix using these vectors and write it to file —one matrix for each

image. These vectors, appended to the visual features, will be the input to our model. These visual features are generated by the AoANet or the BUTD model as explained in 5.1 and 5.2. Note that the vectors are generated for all the words - the preprocessing is done later.
We expect these BERT vectors to help the model direct its attention towards the textual component of the image. Although we also experiment with a pointer-generator network explained in 6.2, we wanted to leverage the model's inbuilt attention mechanism that currently performs as a state of the art model, and guide it towards using these OCR tokens.
For BUTD model, there are two LSTM layers. Apart from increasing the feature size to hold both visual and textual features to be attended to in layer 2 of BUTD model, we also had to change the input to the first layer. As shown in 11, the input takes in the whole context of visual features as an average of all image features, we changed the feature vector to also contain the average of textual features as well. The equation remains the same but the dimensionality of the input vector v increases as now it will hold the context of both visual and textual features.
Our thinking was motivated by the fact that this seemed to be a natural progression when moving from the state of the art Image Captioning model that utilized attention to networks that make use of input information not previously available to improve performance. As described in 6.2, a pointer-generator network can help in generating captions to include proper nouns or OOV words. Similarly, to have a better understanding of the sequence-to-sequence network as well as the attention mechanisms utilized by these models, we decided to experiment with extending the feature set.
The total words in the vocabulary without any extension are 7799.

Once the OCR tokens were detected, we conducted two different experiments with varying sizes of thresholds. We first put a count threshold of 5 i.e. we only add words to the vocabulary which occur 5 or more times. We expect this to severely limit the model's access to OCR tokens. With this threshold, the total words added were 4555. A quantitative analysis of the word counts is shown in Figure 4. The graph shows all OCR tokens detected with the given threshold. Note that it is possible for there to be some overlap between detected OCR tokens and words already existing in the vocabulary.

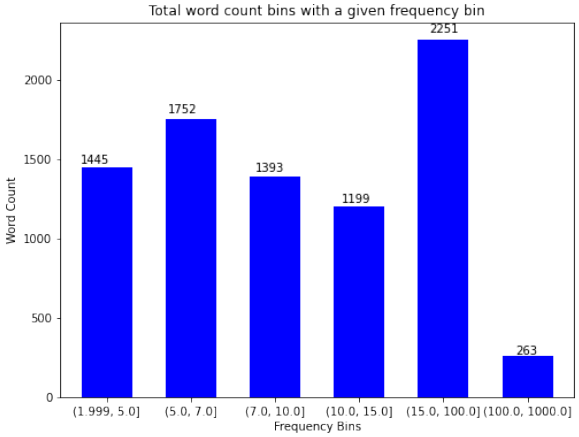

Figure 4: Word counts and frequency bins for threshold = 5

We then put a count threshold of 2 i.e. we only add words to the vocabulary which occur twice or more times. With such a low threshold, we expect a lot of noise to be present in the OCR tokens vocabulary - half-detected text, words in a different language, or words that don't make sense. As part of an experimental setup, we wanted to know how much of a contextual signal can propogate through the model with so much noise as well as relevance mixed together. With this threshold, the total words added were 19781. A quantitative analysis of the word counts is shown in Figure 5. The graph shows all OCR tokens detected with the given threshold. Note that it is possible for there to be some overlap between detected OCR tokens and words already existing in the vocabulary.

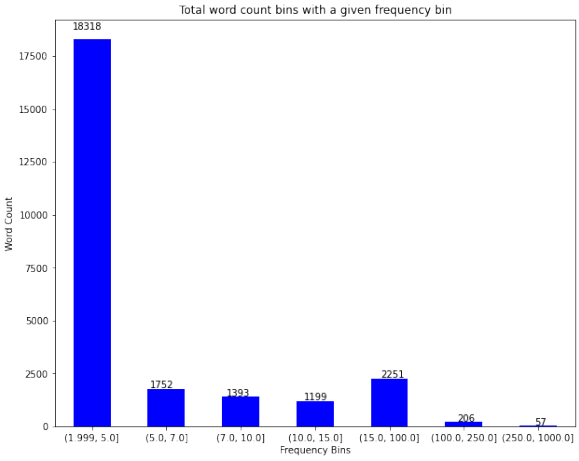

Figure 5: Word counts and frequency bins for threshold = 2

The BERT vectors were generated on Google Colab using a NVIDIA Tesla P4 GPU.

6.2 Copying OCR Tokens via Pointing

In sequence-to-sequence learning, there is often a need to copy certain segments from the input sequence to the output sequence as they are. This can be useful when sub-sequences such as entity names or dates are involved. Instead of heavily relying on meaning, creating an explicit channel to aid copying of such sub-sequences has been shown to be effective (16).

In this approach, in addition to augmenting the input feature set with OCR token embeddings, we employ the pointer mechanism (26) to help copy OCR tokens to the caption when needed. The decoder then becomes a hybrid that is able to copy OCR tokens via pointing as well as generate words from the fixed vocabulary. A soft-switch is used to choose between the two modes. The switching is dictated by *generation of probability*, p_{gen}, calculated at each time-step, t, as follows:

$$p_{gen} = \sigma(w_h^T c_t + w_s^T h_t + w_x^T x_t + b_{ptr}) \quad (16)$$

where σ is the sigmoid function and w_h, w_s, w_x and b_{ptr} are learnable parameters. c_t is the context vector, h_t is the decoder hidden state and x_t is the input embedding at time t in the decoder. At each step, p_{gen} determines whether a word has to be generated using the fixed vocabulary or to copy an OCR token using the attention distribution at time t. Let *extended vocabulary* denote a union of the fixed vocabulary and the OCR words. The probability distribution over the *extended vocabulary* is given as:

$$P(w) = p_{gen} P_{vocab}(w) + (1 - p_{gen}) \sum_{i:w_i=w} a_i^t \quad (17)$$

P_{vocab} is the probability of w using the fixed vocabulary. If w does not appear in the fixed vocabulary, then P_{vocab} is zero. If w is not an OCR word, then $\sum_{i:w_i=w} a_i^t$ is zero.

7 Experiments

In our experiments, we alter AoANet and BUTD as per the approaches described in section 6 and compare these with the baseline models.

BUTD-E and AoANet-E refer to BUTD and AoANet altered as per the approach described in section 6.1 respectively. To observe the impact of the number of OCR words added to the extended vocabulary, we train two Extended variants: (1) E5: Only OCR words that occur at least 5 times in the images are added to the vocabulary. (2) E2: Only OCR words that occur at least 2 times in the images are added to the vocabulary.

BUTD-P and AoANet-P refer to BUTD and AoANet altered as per the approach described in section 6.2 respectively.

We use the code[1] released by the authors of AoANet to train the model. BUTD was implemented as a baseline in the repository and we leverage that to train our BUTD models. We cloned the repository and made changes to extend the feature set and the vocabulary using OCR tokens as well as to incorporate the copy mechanism during decoding [2]. We train our models on a Google Cloud VM instance with 1 Tesla K80 GPU. We also trained models available on gypsum clusters at the University of Massachusetts, Amherst and used an NVIDIA GeForce RTX 2080 Ti GPU.

For both AoANet and BUTD, we use a pre-trained Faster-RCNN (25) model on ImageNet (12) and Visual Genome (22) to extract bottom-up feature vectors of images. The OCR token embeddings are extracted using a pre-trained base, uncased BERT model.

The AoANet models are trained using the Adam optimizer and a learning rate of $2e-5$ annealed by 0.8 every 3 epochs as recommended in (20). The baseline AoANet is trained for 10 epochs while AoANet-E and AoANet-P are trained for 15 epochs.

The BUTD models are trained using the Adam optimizer and a learning rate of $2e-5$. The baseline BUTD and BUTD-E are trained for 10 epochs while BUTD-P is trained for 20 epochs because of more parameters.

8 Results

We show quantitative metrics for each of the models that we experimented in Table 1. We show qualitative results where we compare captions generated by different models in Table 2 and 3. Note that none of the models were also pre-trained on the MS-COCO dataset as Gurari et al. (17) have done as part of their experimenting process.

[1] https://github.com/husthuaan/AoANet
[2] https://github.com/hiba008/AlteredAoA/

Model	Validation Scores				Test Scores			
	BLEU-4	ROUGE_L	SPICE	CIDEr	BLEU-4	ROUGE_L	SPICE	CIDEr
AoANet	21.4	43.8	11.1	40.0	19.52	43.12	12.19	40.76
AoANet-E5	21.41	43.62	10.8	41.4	19.77	42.86	11.88	40.19
AoANet-E2	24.3	46.1	12.9	54.10	22.29	44.96	14.08	53.84
AoANet-P	21.6	43.6	11.5	46.1	19.91	42.73	12.85	45.43
BUTD	19.9	42.6	10.1	35.1	18.52	42.3	11.6	36.5
BUTD-E5	20.6	42.9	10.38	38.1	18.69	42.03	11.48	37.82
BUTD-E2	20.35	42.63	10.37	36.95	18.52	41.77	11.36	36.99
BUTD-P	19.1	40.4	9.5	31.3	17.9	40.5	10.9	34.9

Table 1: Metric Scores

We think that CIDEr (30) and SPICE (6) scores are relevant because of the properties of the ground truth captions that they encode. CIDEr scores use a term-frequency model that counts how many times each *n-gram* occurs. In ground truth captions, we find the usage of words detected by the annotators in the image. Naturally, it is important for any captioning model to include them. SPICE, meanwhile, encodes important image properties while calculating the score - something extremely essential for the task at hand. Ideally, we would want the visual model to encode object-level and scene-level information and use it for decoding. Moreover, our additions to specifically focus on OCR tokens aligns with the goal of the task as well as how SPICE looks at image captioning tasks. Particularly, CIDEr and other metrics other than SPICE are prone to n-gram overlap. However, *n-gram overlap is neither necessary nor sufficient for two sentences to convey the same meaning* (15). SPICE takes this main idea as a motivation to turn these representations into graph-based semantic structure. Such graphs, which encode relationships, objects and other attributes, can be fruitful to have a reliable metric for systems such as image captioning.

We compare different models and find that merely extending the vocabulary helps to improve model performance on the dataset. We see that the AoAnet-E5 matches the validation scores for AoANet but we see an improvement in the CIDEr score. Moreover, we see a massive improvement in validation and test CIDEr scores for AoANet-E2. Similarly, we see a gain in the other metrics too. This goes to show that the BERT embeddings generated for each OCR token for the images do provide an important context to the task of generating captions. Moreover, we see the AoANet-P scores, where we use pointer-generator to copy OCR tokens by extending the vocabulary also perform significantly better than our baseline AoANet model. This goes to show that an OCR copy mechanism is an essential task in generating image captions. Intuitively, it makes sense because we would expect to humans to use these words while generating lengthy captions ourselves.

We also feel that *top-k* sampling is a worthwhile direction of thought especially when we would like some variety in the captions. Beam-search is prone to preferring shorter captions, as the probability values for longer captions accumulate and accumulate smaller values.

Similar to the AoAnet setup, the results presented for BUTD reveal that just using the extended vocabulary and giving the model a chance to focus on these textual features make the model work better than the vanilla model. For both BUTD-E5 and BUTD-E2, across all the metrics we see an improvement for both validation and test set. For BUTD-P model, however under-performs as compared to BUTD baseline. The OCR copy mechanism should have helped but due to lack of attention in BUTD, the model rarely performed copying of an OCR token using the attention distribution and when it did it might have copied the incorrect OCR token explaining the drop in scores. However, as compared to the AoA model where we have a linear model and therefore more parameters, the decoder for the BUTD model is an LSTM. We feel that the extending of vocabulary has provided a large candidate-space for the model, such that a confident prediction is rarely possible. Although, this means that the Pointer-Generator network should ideally beat the baseline model, it doesn't. We feel that this has more

to do with the BUTD model preferring to use its visual features while decoding. The best performance is the extended-vocabulary model, which goes to show that it is placing a substantial weight in its visual input.

9 Error analysis

Although there have been concerns about the robustness of the GCP API towards noise (18), we intended to focus our attention on the pointer generator aspect as well as the analysis of the model with respect to its captioning performance. We agree that the API's performance might hinder the quality of the captions generated, we expected it to not have a large enough impact. That said, there were plenty of extraneous and partly-recognized words in the OCR vocabulary. Thus, there was a considerable amount of noise in our extended vocabulary itself. However, as the results and the following discussions will show, our models with a lower count-threshold performed better.

We first look at how the Extended variants compare with the baseline. We observe that adding text features to the feature set imparts useful information to the model. In image 2a, AoANet perceives the card as a box of food. Addition of text features enables AoANet-E5 to perceive it as a box with black text. While not entirely correct, it is an improvement over the baseline. The alteration also encourages it to be more specific. When the model is unable to find the token that entails specificity, it resorts to producing UNK. Extending the vocabulary to accommmodate more OCR words helps address this problem. In image 3b, baseline AoANet is unable to recognize that the bottle is a supplements bottle. AoANet-E5 attempts to be specific but since 'dietary' and 'supplement' are not present in the extended vocabulary, it outputs UNK. AoANet-E2 outputs a much better caption. We see a similar pattern in 2c.

In BUTD, for a number of images, performance by vanilla and two extension models are similar similar to what we see in 3a. But there is a comparable performance like AoA in terms of images where BUTD-E2 and BUTD-E5 performs better. When the model is unable to find the token that entails specificity, it resorts to producing UNK. For example in image 3b, BUTD-E5 model is trying to be specific about what the container holds but ends up with UNK because of not being able to figure out the exact word —something that could be attributed to a low confidence. In 3c and 3e, we see an improvement with BUTD-E2 and BUTD-E5 model respectively, where they are being more specific about the colors. In 3d the captions by BUTD-E2 and BUTD-E5 both are more cognizant of the label on the tshirt as compared to vanilla model. Some of the best performance can be seen in images like 3f where the extension models are predicting good information as compared to vanilla.

We now look at how the Pointer variant performs compared to the baseline and the Extended variant. Incorporating copy mechanism helps the Pointer variant in copying over OCR tokens to the caption. AoANet-P is able to copy over 'oats' and 'almonds' in 2d and the token 'rewards' in 2e. But the model is prone to copying tokens multiple times as seen in images 3b and 2f. This is referred to as repetition which is a common problem in sequence-to-sequence models (29) as well as in pointer generator networks. Coverage mechanism (29), (26) is used to handle this and we wish to explore this in the future.

In BUTD-P, the pointer variant is compared to the baseline and extended variant. We think BUTD-P prefers the visual features over OCR tokens during caption generation. The model is unable to find a token that requires specificity and instead resorts to predicting UNK. For example in image 3c model should ideally pick the brand or the phrase 'red label' but ends up predicting UNK. While in images like 3f and 3e the model could have been specific about the object in the scene but prefers visual features and ends up predicting similar captions just like the baseline model. Due to the combination of predicting UNK and focussing on visual features, the captions are similar or worse compared to the baseline and extended variants.

Image	Captions
 (a)	**AoANet**: the back of a box of food that is yellow **AoANet-E5**: the back of a yellow box with black text **AoANet-E2**: the back of a card with a barcode on it **AoANet-P**: the back of a UNK UNK card **GT1**: The back of an EBT card that is placed on a black surface. **GT2**: The back of a California EBT debit card. **GT3**: A yellow EBT card on a dark fabric surface. **GT4**: The backside of a beige EBT card with a magnetic strip. **GT5**: back of yellow Quest card with black text on it and a white empty signature box
 (b)	**AoANet**:a person is holding a bottle of seasoning **AoANet-E5**: a person is holding a bottle of UNK **AoANet-E2**: a person is holding a bottle of dietary supplement **AoANet-P**: a person is holding a bottle of super tablets tablets tablets tablets tablets tablets **GT1**: A bottle of Nature's Blend Vitamin D3 2000 IU with 100 tablets. **GT2**: bottle of Nature's Blend brand vitamin D3 tablets, 100 count, 2000 IU per tab **GT3**: A hand is holding a container of vitamin D. **GT4**: Someone is holding a black bottle with a yellow lid. **GT5**: A person's hand holds a bottle of Vitamin D3 tablets.
 (c)	**AoANet**: a a green bottle with a green and white label **AoANet-E5**: a green bottle of UNK UNK UNK UNK **AoANet-E2**: a bottle of body lotion is on a table **AoANet-P**: a bottle of vanilla lotion is sitting on a table **GT1**: A container of vanilla bean body lotion is on a white table. **GT2**: A bottle of body lotion sits on top of a white table **GT3**: a plastic bottle of vanilla bean body lotion from bath and body works **GT4**: A bottle of body lotion that says Noel on it sitting on a table with a phone behind it and other items around it. **GT5**: A body lotion bottle is on top of table with several papers behind it and a set of keys in the background.
 (d)	**AoANet**: a box of frozen dinner is on top of a table **AoANet-E5**: a box of UNK 's UNK brand UNK UNK **AoANet-E2**: a box of granola granola granola granola bars **AoANet-P**: a box of oats 's almond almond bars **GT1**: A box of nature valley roasted almond crunchy bars is on a table. **GT2**: A box of granola bars sitting on a floral cloth near a wooden object. **GT3**: A granola bar box sits on a table cloth with other items. **GT4**: Green box with roasted almond granola bar place tablecloth with flower prints. **GT5**: A package of granola bars is lying on top of a table.
 (e)	**AoANet**: a hand holding a box of chocolate 's brand **AoANet-E5**: a person is holding a package of food **AoANet-E2**: a hand holding a card with a number on it **AoANet-P**: a person is holding a box of rewards card **GT1**: Appears to be a picture of a reward card **GT2**: A plastic card that says speedy rewards membership card. **GT3**: A Speedy Rewards membership card with a large gold star displayed on it. **GT4**: a human hold some cards like credit cards and reward cards **GT5**: Rewards membership card from the Speedway chain of stores.
 (f)	**AoANet**: a bottle of water is on top of a table **AoANet-E5**: a bottle of water is on top of a table **AoANet-E2**: a bottle of vanilla vanilla coffee mate creamer **AoANet-P**: a bottle of vanilla vanilla vanilla vanilla vanilla **GT1**: A bottle of coffee creamer has a plastic flip top cap that can also be twisted off. **GT2**: A blue bottle of coffee creamer is sitting on a counter top next to a black cup. **GT3**: A container of Coffee Mate French Vanilla showing part of the front and part of the back. **GT4**: A bottle of French vanilla coffee creamer sits in front of a mug on the table. **GT5**: A bottle of creamer is on top of a table.

Table 2: Validation Image Captions for AoANet

Image	Captions
 (a)	**BUTD:** a can of campbell 's soup on a counter **BUTD-E2:** a can of campbell 's soup on a counter top **BUTD-E5:** a can of campbell 's soup on a counter **BUTD-P:** a can of campbell 's soup on a counter **GT1**: a can of Campbell's cream of celery condensed soup on a counter top **GT2**: A red Campbell's cream soup on a white background. **GT3**: A container keep in the table shown in the picture. **GT4**: A can of Campbell's soup laying on its side **GT5**: some sort of Campbell's noodle soup that can be used to eat
 (b)	**BUTD:** a person is holding a bottle of wine **BUTD-E2:** a person is holding up a bottle of wine **BUTD-E5:** a person is holding a bottle of UNK **BUTD-P:** a person is holding a bottle of UNK **GT1**: A bottle of Nature's Blend Vitamin D3 2000 IU with 100 tablets. **GT2**: bottle of Nature's Blend brand vitamin D3 tablets, 100 count, 2000 IU per tab **GT3**: A hand is holding a container of vitamin D. **GT4**: Someone is holding a black bottle with a yellow lid. **GT5**: A person's hand holds a bottle of Vitamin D3 tablets.
 (c)	**BUTD:** a clear plastic bottle of water is on a table **BUTD-E2:** a clear plastic bottle of water with a red label **BUTD-E5:** a clear plastic bottle of water is on a table **BUTD-P:** a clear plastic bottle of UNK UNK UNK **GT1**: an empty Arrowhead spring water bottle on a wood table **GT2**: An empty plastic bottle of Arrowhead spring water. **GT3**: Recyclable Plastic Bottle of Arrowhead Mountain Spring water **GT4**: A plastic bottle of Arrowhead spring water is partially empty. **GT5**: A bottle of Arrowhead spring water that has been drunk
 (d)	**BUTD:** a person is holding up a blue shirt **BUTD-E2:** a person is wearing a blue shirt with a blue and white shirt **BUTD-E5:** a person wearing a blue shirt with a blue and white label **BUTD-P:** a person is wearing a blue shirt with a blue and white **GT1**: A person wearing a blue shirt with a logo patch. **GT2**: A blue fabric polo shirt is worn by someone. **GT3**: Quality issues are too severe to recognize visual content. **GT4**: Someone is wearing a blue shirt with a white logo. **GT5**: A light blue polo shirt with a logo stamped on the upper right side of the shirt.
 (e)	**BUTD:** a bottle of lotion is on top of a table **BUTD-E2:** a tube of lotion is on top of a table **BUTD-E5:** a yellow tube of lotion is on top of a table **BUTD-P:** a bottle of lotion is on top of a table **GT1**: "Front side of a yellow bottle on top of a fabric surface **GT2**: A yellow tube of personal use lotion with a pop top. **GT3**: A golden bottle of lotion is on top of the chair. **GT4**: A gold colored tube has lotion in it **GT5**: A small golden yellow bottle of lotion against fabric
 (f)	**BUTD:** a can of food is on top of a table **BUTD-E2:** a green can of green beans on a counter **BUTD-E5:** a can of green beans is on a table **BUTD-P:** a can of food is on top of a table **GT1**: A can of vegetables displaying the nutrition facts. **GT2**: A can of green beans is on the counter. **GT3**: a can good with the back label nutritional facts. **GT4**: Back side of a metal can of food, unable to determine what the product is. **GT5**: The metal can is wrapped in a green label and has the nutrition facts displayed."

Table 3: Validation Image Captions for BUTD

10 Contributions of group members

- Hiba: Code change to pass text features in addition to image features to the encoder, copy mechanism changes in the decoder.
- Nikita: Code changes for BUTD model to use textual features in its two LSTM layers. Worked on using m4c captioner model for VizWiz. Worked on training BUTD Vanilla, BUTD-E2, BUTD-E5 on Vizwiz.
- Daivat: Detect OCR tokens for all images, generate BERT embeddings for all the words detected after preprocessing them and removing stopwords as well as duplicates. Code changes to implement beam search for size ≥ 2.
- Kaivan: Code changes for BUTD pointer generation/copy mechanism model. Code to do analysis on the captions of the validation images.

11 Conclusion

We suggest a pointer-generated oriented Image Captioning model would be more suited when dealing with a specific demographic - people with visual disabilities. Because the task at hand requires a sufficient amount of information contained in the captions, we believe this line-of-thought is extremely relevant. Although there were issues with the model learning to copy tokens once the vocabulary had been extended, we believe that an implementation of coverage as well as cleaning the extended vocabulary can have an impact on the results. It must be noted that as compared to a lot of other NLP or CV models, the amount of data available here was quite low, and a qualitative analysis in Table 2 and Table 3 show encouraging signs. As stated in Section 9, we would like to explore Coverage mechanism in the future. Also, there are other aspects of image captioning that are worth exploring - object detection, counting and segmentation. With a pointer network, this entire system can improve the quality of the captions on the VizWiz dataset. Some other experimentation can also be made by varying the loss function, such as using a multi-label sigmoid loss over softmax loss. Future work also entails some hyperparameter training which we could not perform because of lack of GCP credits and compute power.

References

[MS] Add alternative text to a shape, picture, chart, smartart graphic, or other object.

[Air] Aira.

[BeS] Bespecular.

[GGL] Google vision. https://cloud.google.com/vision. (Accessed on 11/30/2020).

[FB] How does automatic alt text work on facebook? — facebook help center.

[6] Anderson, P., Fernando, B., Johnson, M., and Gould, S. (2016). Spice: Semantic propositional image caption evaluation. In *ECCV*.

[7] Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., and Zhang, L. (2018). Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[8] Aneja, J., Deshpande, A., and Schwing, A. G. (2018). Convolutional image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5561–5570.

[9] Bhattacharya, N. and Gurari, D. (2019). Vizwiz dataset browser: A tool for visualizing machine learning datasets. *arXiv preprint arXiv:1912.09336*.

[10] Chen, X., Fang, H., Lin, T., Vedantam, R., Gupta, S., Dollár, P., and Zitnick, C. L. (2015). Microsoft COCO captions: Data collection and evaluation server. *CoRR*, abs/1504.00325.

[11] Cornia, M., Baraldi, L., Serra, G., and Cucchiara, R. (2018). Paying more attention to saliency: Image captioning with saliency and context attention. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(2):1–21.

[12] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

[13] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

[14] Devlin, J., Cheng, H., Fang, H., Gupta, S., Deng, L., He, X., Zweig, G., and Mitchell, M. (2015). Language models for image captioning: The quirks and what works. *arXiv preprint arXiv:1505.01809*.

[15] Giménez, J. and Màrquez, L. (2007). Linguistic features for automatic evaluation of heterogenous mt systems. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 256–264.

[16] Gu, J., Lu, Z., Li, H., and Li, V. O. (2016). Incorporating copying mechanism in sequence-to-sequence learning. *arXiv preprint arXiv:1603.06393*.

[17] Gurari, D., Zhao, Y., Zhang, M., and Bhattacharya, N. (2020). Captioning images taken by people who are blind.

[18] Hosseini, H., Xiao, B., and Poovendran, R. (2017). Google's cloud vision api is not robust to noise. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 101–105.

[19] Hu, R., Singh, A., Darrell, T., and Rohrbach, M. (2020). Iterative answer prediction with pointer-augmented multimodal transformers for textvqa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9992–10002.

[20] Huang, L., Wang, W., Chen, J., and Wei, X.-Y. (2019). Attention on attention for image captioning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4634–4643.

[21] Kiros, R., Salakhutdinov, R., and Zemel, R. S. (2014). Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

[22] Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., et al. (2017). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73.

[23] Li, Y., Yao, T., Pan, Y., Chao, H., and Mei, T. (2019). Pointing novel objects in image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12497–12506.

[24] Plummer, B. A., Wang, L., Cervantes, C. M., Caicedo, J. C., Hockenmaier, J., and Lazebnik, S. (2015). Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649.

[25] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.

[26] See, A., Liu, P. J., and Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

[27] Sidorov, O., Hu, R., Rohrbach, M., and Singh, A. (2020). Textcaps: a dataset for image captioning with reading comprehension. *ArXiv*, abs/2003.12462.

[TapToSee] TapToSee. Taptosee - blind and visually impaired assistive technology - powered by the cloudsight.ai image recognition api.

[29] Tu, Z., Lu, Z., Liu, Y., Liu, X., and Li, H. (2016). Modeling coverage for neural machine translation. *arXiv preprint arXiv:1601.04811*.

[30] Vedantam, R., Lawrence Zitnick, C., and Parikh, D. (2015). Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[31] Wang, J., Tang, J., and Luo, J. (2020). Multimodal attention with image text spatial relationship for ocr-based image captioning. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4337–4345.

[32] Yao, T., Pan, Y., Li, Y., and Mei, T. (2017). Incorporating copying mechanism in image captioning for learning novel objects. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6580–6588.

[33] Yao, T., Pan, Y., Li, Y., and Mei, T. (2018). Exploring visual relationship for image captioning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 684–699.

[34] Young, P., Lai, A., Hodosh, M., and Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.