# Multi-Modal Image Captioning for the Visually Impaired

**Hiba Ahsan**[*]    **Nikita Bhalla**[*]    **Daivat Bhatt**[*]    **Kaivankumar Shah**[*]

{hahsan,nbhalla,dbhatt,kaivankumars}@umass.edu
University of Massachusetts Amherst

## Abstract

One of the ways blind people understand their surroundings is by clicking images and relying on descriptions generated by image captioning systems. Current work on captioning images for the visually impaired do not use the textual data present in the image when generating captions. This problem is critical as many visual scenes contain text. Moreover, up to 21% of the questions asked by blind people about the images they click pertain to the text present in them (Bigham et al., 2010). In this work, we propose altering AoANet, a state-of-the-art image captioning model, to leverage the text detected in the image as an input feature. In addition, we use a pointer-generator mechanism to copy the detected text to the caption when tokens need to be reproduced accurately. Our model outperforms AoANet on the benchmark dataset VizWiz, giving a 35% and 16.2% performance improvement on CIDEr and SPICE scores, respectively.

.

## 1 Introduction

Image Captioning as a service has helped people with visual impairments to learn about images they take and to make sense of images they encounter in digital environments. Applications such as (Tap-TapSee, 2012) allow the visually impaired to take photos of their surroundings and upload them to get descriptions of the photos. Such applications leverage a human-in-the-loop approach to generate descriptions. In order to bypass the dependency on a human, there is a need to automate the image captioning process. Unfortunately, the current state-of-the-art (SOTA) image captioning models are built using large, publicly available, crowd-sourced datasets which have been collected and created in a contrived setting. Thus, these models perform poorly on images clicked by blind people

---

[*] Equal contribution

largely because the images clicked by blind people differ dramatically from the images present in the datasets. To encourage solving this problem, Gurari et al. (2020) released the VizWiz dataset, a dataset comprising of images taken by the blind. Current work on captioning images for the blind do not use the text detected in the image when generating captions (Figures 1a and 1b show two images from the VizWiz dataset that contain text). The problem is critical as many visual scenes contain text and up to 21% of the questions asked by blind people about the images clicked by them pertain to the text present in them. This makes it more important to improvise the models to focus on objects as well as the text in the images.

With the availability of large labelled corpora, image captioning and reading scene text (OCR) have seen a steady increase in performance. However, traditional image captioning models focus only on the visual objects when generating captions and fail to recognize and reason about the text in the scene. This calls for incorporating OCR tokens into the caption generation process. The task is challenging since unlike conventional vocabulary tokens which depend on the text before them and therefore can be inferred, OCR tokens often cannot 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.

In this work, we build a model to caption images for the blind by leveraging the text detected in the images in addition to visual features. We alter AoANet, a SOTA image captioning model to consume embeddings of tokens detected in the image using Optical Character Recognition (OCR). In many cases, OCR tokens such as entity names or dates need to be reproduced exactly as they are in the caption. To aid this copying process, we employ a pointer-generator mechanism. Our contributions

53

are 1) We build an image captioning model for the blind that specifically leverages text detected in the image. 2) We use a pointer-generator mechanism when generating captions to copy the detected text when needed.



(a) **Model**: a bottle of water is on top of a table
**Ground Truth:** a clear plastic bottle of Springbourne brand spring water
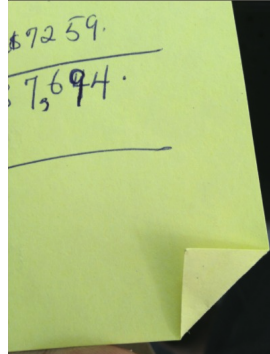
(b) **Model**: A piece of paper with text on it
**Ground Truth**: In a yellow paper written as 7259 and totally as 7694

## 2 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 (Aneja et al., 2018; Yao et al., 2018; Cornia et al., 2018). Language modeling based approaches have also been explored for image captioning (Kiros et al., 2014; Devlin et al., 2015). Apart from the architecture, image captioning approaches are also diverse in terms of the features used. Visual-based image captioning models exploit features generated from images. Multimodal image captioning approaches exploit other modes of features in addition to image-based features such as candidate captions and text detected in images (Wang et al., 2020; Hu et al., 2020).

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 (Chen et al., 2015) and Flickr30k (Plummer et al., 2015) . The task is relatively less explored. Previous works have solved the problem using human-in-the-loop approaches (Aira, 2017; BeSpecular, 2016; TapTapSee, 2012) as well as automated ones (Microsoft; Facebook). 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. (2020) created VizWiz-Captions, a dataset that consists of descriptions of images taken by people who are blind. In addition, they analyzed how the SOTA image captioning algorithms performed on this dataset. Concurrent to our work, Dognin et al. (2020) created a multi-modal transformer that consumes ResNext based visual features, object detection-based textual features and OCR-based textual features. Our work differs from this approach in the following ways: we use AoANet as our captioning model and do not account for rotation invariance during OCR detection. We use BERT to generate embeddings of the OCR tokens instead of fastText. Since we use bottom-up image feature vectors extracted using a pre-trained Faster-RCNN, we do not use object detection-based textual features. Similarly, since the Faster-RCNN is initialized with ResNet-101 pre-trained for classification, we do not explicitly use classification-based features such as those generated by ResNext.

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 (See et al., 2017; Gu et al., 2016). It has also been used in image captioning to aid learning novel objects (Yao et al., 2017; Li et al., 2019). Also, Sidorov et al. (2020) introduced an M4C model that recognizes text, relates it to its visual context, and decides what part of the text to copy or paraphrase, requiring spatial, semantic, and visual reasoning between multiple text tokens and visual entities such as objects.

## 3 Dataset

The Vizwiz Captions dataset (Gurari et al., 2020) consists of over 39,000 images originating from people who are blind that are each paired with five captions. The dataset consists of 23,431 training images, 7,750 validation images and 8,000 test images. The average length of a caption in the train set and the validation set was 11. We refer readers to the VizWiz Dataset Browser (Bhattacharya and Gurari, 2019) as well as the original paper by Gurari et al. (2020) for more details about the dataset.

## 4 Approach

We employ AoANet as our baseline model. AoANet extends the conventional attention mechanism to account for the relevance of the attention results with respect to the query. An attention mod-

ule $f_{att}(Q, K, V)$ 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 $V$.

$$a_{i,j} = f_{sim}(q_i, k_j), \alpha = \frac{e^{a_{i,j}}}{\sum_j e^{a_{i,j}}} \quad (1)$$

$$\hat{v}_i = \sum_j \alpha_{i,j} v_{i,j} \quad (2)$$

$$f_{sim}(q_i, k_j) = \text{softmax}(\frac{q_i k_j^T}{\sqrt{D}}) v_i \quad (3)$$

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:

$$i = W_q^i q + W_v^i \hat{v} + b^i \quad (4)$$

$$g = \sigma(W_q^g q + W_v^g \hat{v} + b^g) \quad (5)$$

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 $\hat{i}$.

$$\hat{i} = g \odot i \quad (6)$$

The AoA module can thus be formulated as:

$$AoA(f_{att}, Q, K, V) = \sigma(W_q^g Q + W_v^g f_{att}(Q, K, V) + b^g) \odot (W_q^i Q + W_v^i f_{att}(Q, K, V) + b^i) \quad (7)$$

The AoA module is applied to both the encoder and decoder. The model is trained by minimizing the cross-entropy loss:

$$L(\theta) = -\sum_{t=1}^{T} \log(p_\theta(y_t^* | y_{1:t-1}^*)) \quad (8)$$

where $y_{1:T}^*$ is the ground truth sequence. We refer readers to the original work (Huang et al., 2019) for more details. We altered AoANet using two approaches described next.

## 4.1 Extending Feature Set with OCR Token Embeddings

Our first extension to the model was to increase the vocabulary by incorporating OCR tokens. We use an off-the-shelf text detector available - Google Cloud Platform's vision API (Google). After extracting OCR tokens for each image using the API, we use a standard stopwords list[1] as part of necessary pre-processing. We use this API to detect text in an image and then generate an embedding for each OCR token that we detect using a pre-trained base, uncased BERT (Devlin et al., 2019) model. The image and text features are fed together into the AoANet model. We expect the BERT embeddings to help the model direct its attention towards the textual component of the image. Although we also experiment with a pointer-generator mechanism explained in Section 4.2, 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.

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. With this threshold, the total words added were $4,555$. We then put a count threshold of 2. 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 do not make sense. With this threshold, the total words added were $19,781$. A quantitative analysis of the OCR tokens detected and their frequency is shown in Figure 2.
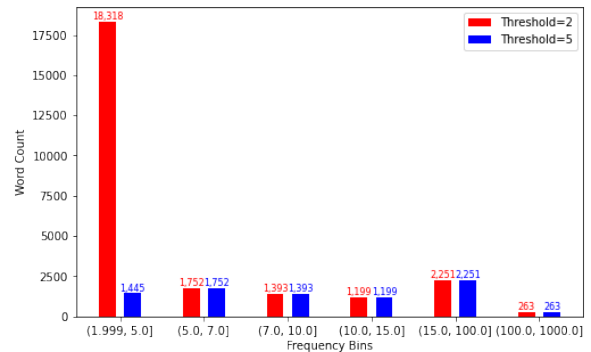


Figure 2: Word counts and frequency bins for threshold = 2 and threshold = 5

## 4.2 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 (Gu et al., 2016).

In this approach, in addition to augmenting the input feature set with OCR token embeddings, we employ the pointer-generator mechanism (See et al., 2017) to 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 probability*, $p_{gen}$, calculated at each time-step, $t$, as follows:

$$p_{gen} = \sigma(w_h^T c_t + w_s^T h_t + w_x^T x_t + b_{ptr}) \quad (9)$$

where $\sigma$ 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:

$$P(w) = p_{gen}P_{vocab}(w) + (1 - p_{gen})\sum_{i:w_i=w} a_i^t \quad (10)$$

$P_{vocab}$ is the probability of $w$ using the fixed vocabulary and $a$ is the attention distribution. 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.

## 5 Experiments

In our experiments, we alter AoANet as per the approaches described in Section 4 and compare these with the baseline model. AoANet-E refers to AoANet altered as per the approach described in Section 4.1. 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 with frequency greater than or equal to 5. (2) E2: Only OCR words that occur with frequency greater than or equal to 2. AoANet-P refers to AoANet altered as per the approach described in Section 4.2. The extended vocabulary consists of OCR words that occur with frequency greater than or equal to 2.

We use the code[2] released by the authors of AoANet to train the model. 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 [3]. We train our models on a Google Cloud VM instance with 1 Tesla K80 GPU. Like the original work, we use a Faster-RCNN (Ren et al., 2015) model pre-trained on ImageNet (Deng et al., 2009) and Visual Genome (Krishna et al., 2017) 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 Huang et al. (2019). The baseline AoANet is trained for 10 epochs while AoANet-E and AoANet-P are trained for 15 epochs.

## 6 Results

We show quantitative metrics for each of the models that we experimented with in Table 1. We show qualitative results where we compare captions generated by different models in Table 2. Note that none of the models were pre-trained on the MS-COCO dataset as Gurari et al. (2020) have done as part of their experimenting process.

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

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

56

| 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 | 19.5 | 43.1 | 12.2 | 40.8 |
| AoANet-E5 | 21.4 | 43.6 | 10.8 | 41.4 | 19.8 | 42.9 | 11.9 | 40.2 |
| AoANet-E2 | **24.3** | **46.1** | **12.9** | **54.1** | **22.3** | **45.0** | **14.1** | **53.8** |
| AoANet-P | 21.6 | 43.6 | 11.5 | 46.1 | 19.9 | 42.7 | 12.8 | 45.4 |

Table 1: Validation and Test scores for AoANet, AoANet-E5 (extended vocabulary variant with OCR frequency threshold as 5), AoANet-E2 (extended vocabulary variant with OCR frequency threshold as 2) and AoANet-P (pointer-generator variant).

OCR tokens after extending the vocabulary also perform 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 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 accumulates smaller values as discussed by Holtzman et al. (2019).

## 7 Error Analysis

Although there have been concerns about the robustness of the GCP API towards noise (Hosseini et al., 2017), we focused our attention on the model's captioning performance and on the pointer-generator mechanism. We agree that the API's performance might hinder the quality of the captions generated but we expected it to not have a large enough impact.

We first look at how the Extended variants compare with the baseline. We observe that adding text-based features to the feature set imparts useful information to the model. In 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 accommodate more OCR words helps address this problem. In image 2b, 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.

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 2b and 2f. This is referred to as repetition which is a common problem in sequence-to-sequence models (Tu et al., 2016) as well as in pointer generator networks. Coverage mechanism (Tu et al., 2016; See et al., 2017) is used to handle this and we wish to explore this in the future.

## 8 Conclusion

In this work, we propose a pointer-generator based image captioning model that deals specifically with images taken by people with visual disabilities. Our alteration of AoANet shows significant improvement on the VizWiz dataset compared to the baseline. As stated in Section 7, we would like to explore coverage mechanism in the future. Dognin et al. (2020) recently discussed their winning entry to the VizWiz Grand Challenge. In addition, Sidorov et al. (2020) introduced a model that has shown to gain significant performance improvement by using OCR tokens. We intend to compare our model with these and improve our work based on the observations made.

## 9 Acknowledgements

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

Table 2: Examples of captions generated by AoANet, AoANet-E5 (extended vocabulary variant with OCR frequency threshold as 5), AoANet-E2 (extended vocabulary variant with OCR frequency threshold as 2) and AoANet-P (pointer-generator variant) for validation set images along with their respective ground truth captions.

# References

Aira. 2017. Aira: Connecting you to real people instantly to simplify daily life.

Jyoti Aneja, Aditya Deshpande, and Alexander G Schwing. 2018. Convolutional image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5561–5570.

BeSpecular. 2016. Bespecular: Let blind people see through your eyes.

Nilavra Bhattacharya and Danna Gurari. 2019. Vizwiz dataset browser: A tool for visualizing machine learning datasets. *arXiv preprint arXiv:1912.09336*.

Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, and Tom Yeh. 2010. Vizwiz: Nearly real-time answers to visual questions. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, page 333–342, New York, NY, USA. Association for Computing Machinery.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. 2015. Microsoft COCO captions: Data collection and evaluation server. *CoRR*, abs/1504.00325.

Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. 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.

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. Ieee.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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.

Jacob Devlin, Hao Cheng, Hao Fang, Saurabh Gupta, Li Deng, Xiaodong He, Geoffrey Zweig, and Margaret Mitchell. 2015. Language models for image captioning: The quirks and what works. *arXiv preprint arXiv:1505.01809*.

Pierre Dognin, Igor Melnyk, Youssef Mroueh, Inkit Padhi, Mattia Rigotti, Jarret Ross, Yair Schiff, Richard A Young, and Brian Belgodere. 2020. Image captioning as an assistive technology: Lessons learned from vizwiz 2020 challenge. *arXiv preprint arXiv:2012.11696*.

Facebook. How does automatic alt text work on Facebook? — Facebook Help Center.

Google. Google cloud vision. (Accessed on 11/30/2020).

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor OK Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. *arXiv preprint arXiv:1603.06393*.

Danna Gurari, Yinan Zhao, Meng Zhang, and Nilavra Bhattacharya. 2020. Captioning images taken by people who are blind.

Ari Holtzman, Jan Buys, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *CoRR*, abs/1904.09751.

H. Hosseini, B. Xiao, and R. Poovendran. 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.

Ronghang Hu, Amanpreet Singh, Trevor Darrell, and Marcus Rohrbach. 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.

Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. 2019. Attention on attention for image captioning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4634–4643.

Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. 2014. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73.

Yehao Li, Ting Yao, Yingwei Pan, Hongyang Chao, and Tao Mei. 2019. Pointing novel objects in image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12497–12506.

Microsoft. Add alternative text to a shape, picture, chart, smartart graphic, or other object. (Accessed on 11/30/2020).

Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 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.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

O. Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. 2020. Textcaps: a dataset for image captioning with reading comprehension. *ArXiv*, abs/2003.12462.

TapTapSee. 2012. Taptapsee - blind and visually impaired assistive technology - powered by the cloudsight.ai image recognition api.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. *arXiv preprint arXiv:1601.04811*.

Jing Wang, Jinhui Tang, and Jiebo Luo. 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.

Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 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.

Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2018. Exploring visual relationship for image captioning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 684–699.