Towards Multimodal Vision-Language Models Generating Non-Generic Text

Wes Robbins
Montana State University
100 Culbertson Hall
Bozeman, MT 59717
wesley.robbins@students.montana.edu

Zanyar Zohourianshahzadi & Jugal Kalita
University of Colorado, Colorado Springs
1420 Austin Bluffs Pkwy
Colorado Springs, CO 80918
{zzohouri,jkalita}@uccs.edu

Abstract
Vision-language models can assess visual context in an image and generate descriptive text. While generated text may be accurate and syntactically correct, it is often overly general. Recent work has used optical character recognition to supplement visual information with text extracted from an image. In many cases, using text in the image improves the specificity and usefulness of generated text. In this work, we contend that vision-language models can benefit from additional sets of tokens that can be extracted from an image, but are ignored by current models. We modify previous multimodal frameworks to accept relevant information from any number of auxiliary classifiers. In particular, we focus on person names as an additional set of tokens and create a novel image-caption dataset to facilitate this task. The dataset, Politicians and Athletes in Captions (PAC), consists of captioned images of well-known people in context. By fine-tuning pretrained models with this dataset, we demonstrate a model that can naturally integrate facial recognition tokens into generated text by training on only a few samples. For the PAC dataset, we provide an in-depth discussion about collection, analysis, and fairness. Finally, we present baseline benchmark scores on PAC.

1 Introduction
Vision-language models combine deep learning techniques from computer vision and natural language processing to assimilate visual and textual understanding. Models in this domain can demonstrate visual and linguistic knowledge by performing tasks such as vision question answering (VQA) and image captioning. There are many applications of these tasks, including aiding the visually impaired by providing scene information and screen reading (Morris et al. 2018).

To perform a vision-language task a model needs to understand visual context and natural language, and operate in a shared embedding space between the two. To transcribe visual information to text, an encoder-decoder architecture is trained to learn the necessary shared embeddings. Approaches in the literature have improved performance by pre-training models for both visual context and language understanding (Chen et al. 2020; Lu et al. 2019a; Su et al. 2019; Li et al. 2020; Tan and Bansal 2019). These frameworks have yielded accurate and semantically appropriate VQAs or captions. However, the text generated from these are general and overlook clues that allow for richer text generation with improved contextualization.

Recent work has used optical character recognition (OCR) in order to incorporate text that appears in images (Zhu et al. 2021; Gao et al. 2020b; Maffa et al. 2021; Hu et al. 2020; Kant et al. 2020; Wang et al. 2021; Han, Huang, and Han 2020; Liu et al. 2020). In many cases, this significantly enhances the usefulness of the generated text (Hu et al. 2020). Such frameworks include OCR as an additional input modality. This results in three modalities for VQA (image, question, and OCR) and two modalities for image captioning (image and OCR). Attention based mechanisms have been successful in integrating these multiple modalities for vision-language tasks.

While using OCR allows enhancement of some generated text, specific information that exists in human-level description may also come from additional sources. Without proper nouns or other specific vocabulary, the generated text is at risk of being awkwardly general, demonstrating a lack of shared knowledge that is expected throughout society. Figure 1 show two images where using specific terms is critical to human like descriptions.

A parallel field of study is entity-aware image captioning. The focus of this task is to extract relevant information from an image-article pair to generate a caption. While models in this domain can generate captions with non-generic text, they rely on an article for specific vocabulary rather than strictly on the image content.

In this work, we propose a method for entity-aware text generation that can be based solely on image content. We propose generalizing the OCR input modality to accept helpful tokens from any number of auxiliary classifiers. This framework allows a model to leverage easily available sophisticated libraries for tasks like face recognition and OCR extraction. We refer to all tokens from upstream sources, including OCR tokens, as special tokens.

We focus on person names as an example special token. To facilitate this task we create a novel image-caption dataset, Politicians and Athletes in Captions (PAC), which includes person names in captions in addition to relevant scene-text found on signs or labels. PAC has 1,572 images and three captions per image. A full discussion on the dataset is provided in Section 4.
By training on PAC in addition to other image-caption datasets, we create a model that can naturally integrate person names into captions. The same model still performs well on more general image captioning tasks. We show this can be learn by training on a limited number of samples. Evaluation of the methods is available in Section 5.

In summary, this paper makes the following contributions:

1. Proposes special tokens as a framework to incorporate tokens from external sources into generated text.
2. Releases PAC image-captioning dataset and baseline results.
3. Demonstrates a model that integrates OCR and facial recognition into image captioning.

2 Related Work

The encoder-decoder architecture divides the image captioning task into two parts. The encoder acts as feature extractor and the decoder handles word generation. Before the advent of attention mechanism, deep learning models for image captioning used CNN encoders for feature extraction from the whole input image as a whole (Kiros, Salakhutdinov, and Zemel 2014; Karpathy, Joulin, and Fei-Fei 2014; Vinyals et al. 2015).

Attention mechanism was first introduced in encoder-decoder architecture for neural machine translation (Bahdanau, Cho, and Bengio 2015). Enabling the model to perform better translation due to its ability to focus on the relevant parts of the input for generating the output at each time step. The seminal image captioning model, Show, Attend and Tell (Xu et al. 2015), applied attention mechanism on input visual features and previously generated word (during inference) at each time step for textual caption word generation. Soft and hard visual attention were the first types of attention mechanisms used for image captioning.

The majority of current state-of-the-art methods for image captioning and visual question answering benefit from the bottom-up and top-down attention mechanism (Anderson et al. 2018). Bottom-up and top-down attention was introduced in the context of encoder-decoder architecture (Sutskever, Vinyals, and Le 2014) for image captioning and visual question answering.

Bottom-up attention acts as a hard attention mechanism and leverages an object detector, Faster R-CNN (Ren et al. 2015) for detecting the most important regions in the image (Anderson et al. 2018). Top-down attention acts as a soft attention mechanism as it performs a soft modulation over the set of input visual features from object detection regions. Similar to how bottom-up attention was adopted for feature extraction in OCR (Hu et al. 2020) we adopt this mechanism for feature extraction in facial recognition. Rather than Faster R-CNN, we use RFBNet (Liu, Huang, and Wang 2018) facial region detection. For facial feature extraction we use ArcFace (Deng et al. 2019) pre-trained on MegaFace dataset (Kemelmacher-Shlizerman et al. 2016).

Several techniques have been proposed to handle OCR tokens in vision-language tasks. The M4C algorithm uses an indiscriminate attention layer followed by a dynamic pointer (Hu et al. 2020). The SS-Baseline model uses individual attention blocks for each input modality followed by a single fusion encoding layer (Zhu et al. 2021). Several approaches have been proposed to better handle spatial information about OCR tokens (Gao et al. 2020b; 2020a; Wang et al. 2021; Kant et al. 2020; Han, Huang, and Han 2020). The MMR method was proposed to utilize spacial information of objects and scene-text via a graph structure (Mafla et al. 2021). TextOCR was introduced as an end-to-end method for identifying OCR tokens (Singh et al. 2021). TAP was introduced as a method to integrate OCR tokens into pre-training. We adopt M4C as a base model for our work. Modifications are enumerated in Section 3.

Entity aware image captioning focuses on creating captions from image-article pairs (Biten et al. 2019a; Tran, Mathews, and Xie 2020; Lu et al. 2018). Our work is distinct in that it focuses on generating captions strictly from visual information rather than articles. More similar to our work, Zhao et al. uses an upstream vision classifier as input to a
captioning model. They introduce a multi-gated decoder for handling input from external classifiers (Zhao et al. 2019). Comparatively we use general OCR and facial recognition classifiers rather than a web entity recognizer as an upstream classifier. Our approach is different from Zhao et al. in that we use bottom-up and top-down attention rather than a stand alone CNN for object detection, use a common embedding space rather than a gated decoder for handling multi-modal inputs, and use rich representations (see Section 3.2) rather than only textual information for handling tokens from upstream classifiers.

MS-COCO (Lin et al. 2014) is a large dataset for common objects in context used for image captioning. Similar to MS-COCO, Flickr30k (Young et al. 2014) is another common dataset used for image captioning. For MS-COCO, Karpathy’s split (Karpathy and Fei-Fei 2015) is used as a common benchmark for image captioning. Google’s conceptual captions (Sharma et al. 2018) is a vast dataset used for pre-training multitasking vision-language models and fine-tuning them on other vision-language downstream tasks (Lu et al. 2019b; 2020).

To facilitate use of optical character recognition in the Vision-Language domain, several datasets have been released, including ST-VQA (Biten et al. 2019b) for scene text visual question answering and TextCaps (Sidorov et al. 2020) for image captioning with reading comprehension. Along with the introduction of TextCaps dataset, the M4C model (Hu et al. 2020) originally used for visual question answering was adopted for image captioning. We modify the M4C model via adding another modality of information that includes bottom-up facial recognition features.

3 Special Tokens

We propose special tokens as a placeholder for extracted textual information that is identified as present in an image by an upstream source and then subsequently used by a vision-language model. In this approach there are two modalities that hold information about an image. The first modality is generic visual features which are responsible for informing the model of general context. The second modality, special tokens, is responsible for informing the model of specific terms that are relevant to the image. The first modality is represented by visual feature vectors where as the special token modality consists of visual feature vectors and textual feature vectors. Special tokens are additionally made available for direct copy into generated text which allows for zero-shot inclusion of words not seen prior. This structure has been successful on OCR vision-language datasets. The key hypothesis of this paper is that a model can further learn to differentiate types of tokens with in the special token modality and subsequently use each token type appropriately. Passing all tokens from upstream sources through the same modality allows the model to accept any tokens from any number of sources at inference time. A multi-modal transformer is used such that all special tokens can attend to each other during training and inference. This is the substrate for which a model can not only learn to differentiate tokens, but also learn relationships between tokens and what effect the presence of each token should have on generated text.

In this approach, OCR tokens are placed under the umbrella of special tokens and received along side tokens from other upstream sources. In this work, we demonstrate adding facial recognition tokens received from a face recognition module. We focus our experimentation on learning to integrate facial recognition tokens by training on the PAC dataset. However, any set of words that can be identified by some module can conceivably be a set of special tokens.

3.1 Motivation

The goal of special tokens is to integrate vocabulary tokens from external sources into generated text. An opposing approach would be to use an enhanced object detector for all detection and correspondingly extend the model vocabulary to include all desired vocabulary tokens. The special tokens approach can be motivated based on several following observations.

1) Different architectures perform better on different tasks. Several tasks, such as OCR detection and facial recognition, benefit from specialized approaches that differ from traditional object detection. In OCR, detection finds characters individually rather than classifying words. In facial recognition, a regression model is trained to output face embeddings which are subsequently compared to embeddings of known individuals. Even on standard classification tasks, significant research is put into fine-tuning architectures to get state of the art results on dataset benchmarks. This work can be leveraged by utilizing the fine-grained classifiers as upstream sources.

2) The space complexity of all possible vocabulary tokens is intractably large. By appending special tokens to the vocabulary at inference time the captioning model’s vocabulary is prevented from inflating.

3) Using non-generic terms does not always increase the complexity of the caption. For example in Figure 1, the names ‘General Motors’ and ‘Barack Obama’ are substitutions for what could have been generic terms such as building or person. If a captioning model can generate a caption such as ‘A person giving a speech’, it does not need a significant increase in contextual understanding to generate the caption ‘Barack Obama giving a speech’. Rather, the model just needs to know to use ‘Barack Obama’. The special token approach takes advantage of this and allows the model to learn these correlations in few iterations and on few samples. At the same time, the special token approach does not limit the model from learning more nuanced correlations between tokens and context assuming rich enough data is provided.

4) Not all applications will expect the same special tokens. A captioning model that supplements screen reading for the visually impaired deployed in one area of the world may desire different sets or subsets of tokens than the same model deployed in a different part of the world. In this sense special tokens are highly practical. The same model can be transferred between applications simply by adding or detracting the external sources used for generating tokens.

3.2 Rich Representations

Utilizing the same representation for all special tokens allows the model to accept tokens from any external source.
The representation encodes several types of information about the token allowing the model to learn how different components of a token effect its use. We adopt the the visual and textual feature representation from Hu et al. and add a token source feature (Hu et al. 2020). A formal description of the special representation is described below and a visual representations is provided in Figure 2.

Special tokens are represented by a feature vector $x_i^{spec}$, where $i = 1...N$. $x_i^{spec}$ incorporates visual features, textual features, and a classifier feature. The visual features include a bounding box $x_i^{b}$ and a feature vector from an object detector $x_i^{f}$. Following previous work we use a pretrained Faster-RCNN with a ResNet backbone to generate $x_i^{f}$s from the RoI created by the tokens bounding box. The textual features are a fastText encoding $x_i^{t}$ and a pyramidal hierarchy of characters (PHOC) encoding $x_i^{p}$. The classifier feature $x_i^{c}$ is a one-hot encoding of sources used for generating special tokens, $x_i^{fr}, x_i^{ft},$ and $x_i^{p}$ are concatenated together and projected into a tuned encoding dimensionality $d$ by a learned linear transformation $W_1$. Additionally, $x_i^{b}$ and $x_i^{c}$ are projected into $d$ by separate learned linear transformations $W_2$ and $W_3$. Layer normalization $LN$ is applied to the three $d$ dimensional vectors. $x_i^{spec}$ is a result of element wise addition of these three vectors after layer normalization as shown below:

$$x_i^{spec} = LN(W_1(x_i^{fr}+x_i^{ft}+x_i^{p}))+LN(W_2x_i^{b})+LN(W_3x_i^{c})$$

(1)

### 3.3 Adopting M4C

We utilize the multimodal mesh copy module (M4C) introduced it Hu et al. in order to copy special tokens into generated text (Hu et al. 2020). This method has been directly adopted by many subsequent OCR Vision-Language papers for copying scene text into generated text. We generalize the input from OCR tokens to special tokens, but make no change to the copying mechanism. Here we formalize the differences between our captioning model and the M4C Captioning model.

The input modalities into the M4C captioning model are object features $\{x_{1}^{obj}, ..., x_{M}^{obj}\}$ for $M$ objects and OCR tokens $\{x_{1}^{ocr}, ..., x_{N}^{ocr}\}$ for $N$ OCR tokens. We generalize OCR tokens to special tokens such that the inputs are $\{x_{1}^{obj}, ..., x_{M}^{obj}\}$ and $\{x_{1}^{spec}, ..., x_{N}^{spec}\}$ for $N$ special tokens.

M4C captioner predicts fixed vocab scores $\{y_{1,t}, ..., y_{K,t}\}$ where $K$ is a fixed vocab size and $t$ is the decoding step and ocr vocab scores $\{y_{1,t}^{ocr}, ..., y_{N,t}^{ocr}\}$ where $N$ is the number of ocr tokens. The selected word at each time step $w_t = \text{argmax}(y_{N,t}^{ocr})$ where $y_{N,t}^{ocr} = [y_{1,t}^{ocr}; y_{N,t}^{ocr}]$. We substitute $y_{t}^{spec} = [y_{1,t}^{spec}; y_{N,t}^{spec}]$, where $N$ is the number of special tokens, for $y_{t}^{ocr}$ such that $y_{N,t}^{t} = [y_{1,t}^{ocr}; y_{t}^{spec}]$.

### 4 PAC Dataset

With this paper we release the Politicians and Athletes in Captions (PAC) dataset. PAC is image-caption dataset consisting of images well-known individuals in context. PAC has 1,572 images and three captions per image. Samples from PAC can be seen in Figure 4.

We create PAC with the goal of studying the use of non-generic vocabulary in image captioning. The non-generic terms emphasized in PAC are person names and OCR tokens. The PAC dataset offers the following technical challenges:

1. Correctly identifying people in a variety of settings.
2. Reasoning about the effect of the presence of the individual. If a known person is in a scene, the description of the scene often based on the known person.
3. Naturally integration of a name into a generated caption.

Recent work has shown the ability of pretrained vision-language models to adapt to new domains with limited samples (Tsimploukelli et al. 2021). PAC can be used in the sense to test a pre-trained models ability to learn problems 2 and 3.

### 4.1 Collection

Images were collected from the Creative Commons image database which are made available under the CC licence. In the collection of the dataset, 62 public figures were searched for. We selected images from the first 100 returned, filtering out duplicate images and images without visible faces.

Annotators were instructed to provide a caption of the image including the name of the individual in which was searched for when collecting the image. Other famous individuals who happened to appear in the image may also be mentioned in the captions. Additionally, annotators were instructed to use scene-text if it improved the quality of the caption. These annotation instructions differ those for caption collection of previous datasets. For example, in the collection of MS-COCO captions, annotators were instructed
to not use proper nouns (Chen et al. 2015) and annotators for TextCaps were instructed to always use text in the scene (Sidorov et al. 2020). 658 images were captioned by college students and 914 were outsourced to Ground Truth. Captions were scanned for grammar and spelling errors.

4.2 Analysis

PAC includes images 1,572 images with 3 captions each. All images include at least one famous politician or athlete. Overlap exists in several images. 62 different individual are in the dataset for an average of 25.2 images per person. 23 of the individuals are athletes while 39 are athletes.

Each corresponding caption includes the name of at least one person name in the image. 66.1% of the data has scene text that is recognized by Google Cloud OCR. 35.9% of use scene text exactly as recognized by Google Cloud OCR in one of the captions. Comparatively, 96.9% of TextCaps images have OCR and 81.3% of captions use at least one OCR token. In 96.3% of the images a face RoI is detected by the RFB net, the face detector we use throughout this work.

(Sidorov et al. 2020)

PAC captions have a lower average word count than other image-caption datasets at 8.35 words per caption. Comparatively, TextCaps has 12.4 words per caption, Conceptual Captions has 9.7, and MS-COCO has 10.5 (Sidorov et al. 2020). A number of images in PAC are close-up photos of individuals. Often, a precise caption for such an image contains few words (e.g. ‘Joe Biden wearing a suit’) which explains the lower average. The names of individuals range between one and five words. Names account for 25.7% of the words in PAC captions.

4.3 Fairness

The fact that machine learning model learn bias from datasets has been well documented in the literature (Belkin et al. 2019). When the dataset includes information about people, discriminatory biases may be encoded in the data (Zliobaite 2015). A model can learn to correlate any characteristic in the data to an outcome which can result discriminatory model behavior. In PAC, the human characteris-

5 Experiments

We compare results on PAC with and without special tokens. We test several configurations of combining TextCaps and PAC for training and highlight the best results in Table 2.

5.1 Implementation Details

We build our implementation on top of the MMF library (Singh et al. 2020). For detecting regions in the image with faces we use RFB net (Liu, Huang, and Wang 2018). For facial recognition we use ArcFace (Deng et al. 2019).
Table 1: Training on PAC with and without special tokens.

<table>
<thead>
<tr>
<th>#</th>
<th>Model</th>
<th>Training</th>
<th>B-4</th>
<th>M</th>
<th>R</th>
<th>C</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>M4C</td>
<td>TextCaps→PAC</td>
<td>2.1</td>
<td>6.4</td>
<td>14.3</td>
<td>24.6</td>
<td>4.3</td>
</tr>
<tr>
<td>2</td>
<td>M4C+ST</td>
<td>TextCaps→PAC</td>
<td>14.5</td>
<td>22.4</td>
<td>42.0</td>
<td>156.8</td>
<td>30.3</td>
</tr>
</tbody>
</table>

ST: Special Tokens; B-4: BLEU-4; M: METEOR; R: ROUGE; C: CIDEr; S: SPICE

Figure 4: Captions generated for PAC test set images. Green words indicate tokens from the face recognition module and blue words indicate tokens from the OCR module. Corresponding metrics found in Table 1.

Using ArcFace we extract facial embedding for all individuals in the dataset. At inference, a face token if is extracted if the $l_2$ distance between the new embeddings and the pre-calculated embedding of a known individual is less than a threshold $T$. First names and last names become separate tokens with copied visual features. Face RoIs were found to be present 96.3% of images by RFB net. Approximately 80% of the predictions made by ArcFace on the face RoIs were correct. To reduce noise in the training set, we manually update the face tokens to be present in all images and further assure by the face token is the correct name by referencing the name that was searched for to get the respective image. All reported quantitative and qualitative test set results use unmodified tokens generated from the aforementioned facial recognition models.

We use Google Cloud OCR for extracting OCR tokens and set a limit at $N = 50$ OCR tokens. Following previous work, we use a pretrained faster RCNN with a ResNet-101 backbone to propose RoIs and extract features for each region. A limit is set at $M = 100$ object features. PAC is broken up into the same 80-20 train-test split for all experiments.

5.2 Benchmarks

In Table 1 we compare results on PAC with vanilla M4C and M4C with special tokens (M4C+ST). Both models are trained on TextCaps for 12,000 iterations and subsequently on PAC for 1,300 iterations. M4C+ST sees 200-700% increases across metrics on the PAC test set. Without access to name tokens the vanilla M4C model has a small chance of using the correct name only if the name happens to be in model vocabulary. If the name is not in model vocabulary there is no chance. Corresponding qualitative examples are provided in Figure 4. In the samples, M4C+ST has appropriately used the face token in the caption. The right two images are samples where M4C+ST switched between model vocabulary, face tokens, and OCR tokens. This demonstrates the model has learned to differentiate separate special token types.

In Table 2, we report scores after training on several different combinations of PAC and TextCaps. The best captioning model is the one that performs well on PAC while still performing while on previous datasets. For this reason, scores are reported for PAC and TextCaps for all training combinations. The model trained on a ratio of TextCaps 8:1 PAC (Table 2 line 4) scores the highest in this regard.

6 Conclusion

Text generated by vision-language models often lacks specific terms that would be present in human level descriptions or answers. The special token approach can be used to introduce non-generic information to a vision-language model and consequently improve generated text. The special token approach accepts information from any number of upstream sources. The Politicians and Athletes in Captions dataset consists of image-caption pairs with well-known individuals. By using the special token approach and the PAC dataset, we train a model to integrate person names into
Table 2: PAC Baselines using Special Tokens and M4C Architecture. In the training column, a → between datasets indicates one dataset was trained on before the other whereas a comma in between datasets indicates they were trained on simultaneously.

<table>
<thead>
<tr>
<th>#</th>
<th>Tokens Training</th>
<th>Test</th>
<th>Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>B-4</td>
</tr>
<tr>
<td>1 a.</td>
<td>Special TextCaps</td>
<td>PAC</td>
<td>0.7</td>
</tr>
<tr>
<td>b.</td>
<td></td>
<td>TextCaps</td>
<td>22.9</td>
</tr>
<tr>
<td>2 a.</td>
<td>Special PAC</td>
<td>PAC</td>
<td>14.9</td>
</tr>
<tr>
<td>b.</td>
<td></td>
<td>TextCaps</td>
<td>2.0</td>
</tr>
<tr>
<td>3 a.</td>
<td>Special TextCaps→PAC</td>
<td>PAC</td>
<td>14.5</td>
</tr>
<tr>
<td>b.</td>
<td></td>
<td>TextCaps</td>
<td>20.7</td>
</tr>
<tr>
<td>4 a.</td>
<td>Special PAC,TextCaps</td>
<td>PAC</td>
<td>13.6</td>
</tr>
<tr>
<td>b.</td>
<td></td>
<td>TextCaps</td>
<td>22.1</td>
</tr>
<tr>
<td>5 a.</td>
<td>Special TextCaps→PAC,TextCaps</td>
<td>PAC</td>
<td>6.0</td>
</tr>
<tr>
<td>b.</td>
<td></td>
<td>TextCaps</td>
<td>23.2</td>
</tr>
</tbody>
</table>

B-4: BLEU-4; M: METEOR; R: ROUGUE; C: CIDEr; S: SPICE

text. This paper works towards vision-language models that generate human-like non-generic text, but comes far from solving the problem. Possible improvements to the proposed method include inclusion of more external sources, integrating open-domain knowledge with special tokens, or other architecture improvements.

Acknowledgement

The work reported in this paper is supported by the National Science Foundation under Grant No. 2050919. Any opinions, findings and conclusions or recommendations expressed in this work are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References


