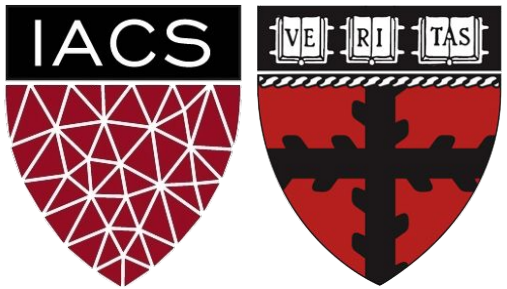
Caption This - Group BKKST

Advanced Practical Data Science, MLOps

AC295

Team Members: Al-Muataz Khalil, Ed Bayes, Stephen Knapp, Matthew Stewart, Shih-Yi Tseng



Outline

- Project Scope
- Project Workflow
- Process Flow
- Data
- Baseline Model
- Current Best Model
- App design
- Deployment

Problem Definition

The World Health Organization (WHO) estimated that 314 million people have visual impairment across the world, including 269 million who have low vision, and 45 million who are blind (Ono et al 2010). Many people with visual impairments rely on screen readers in order to access the internet through audio, and thus depend on image captions (Yesilada et al 2004). Therefore, accessibility, as well as automatic indexing and other goals, make accurate image captioning an important priority (Hossain et al 2018).

Proposed Solution

We will design, build, and deploy at-scale an application which receives an image of an everyday activity which then assigns a caption of the image contents, based on state-of-art computer vision and natural language models. Additionally, the app will provide a visualization of the image components reflected in the caption through saliency maps.

Project Scope

Proof Of Concept (POC)

- Set up CI/CD pipeline
- Store Flickr8k and COCO datasets in GCP bucket
- Conduct image feature extraction and EDA
- Test baseline model (efficientB0 net + RNN)
- Improve model architecture (CLIP + transformer)and training with full dataset
- Verify models predict labels for unseen photos

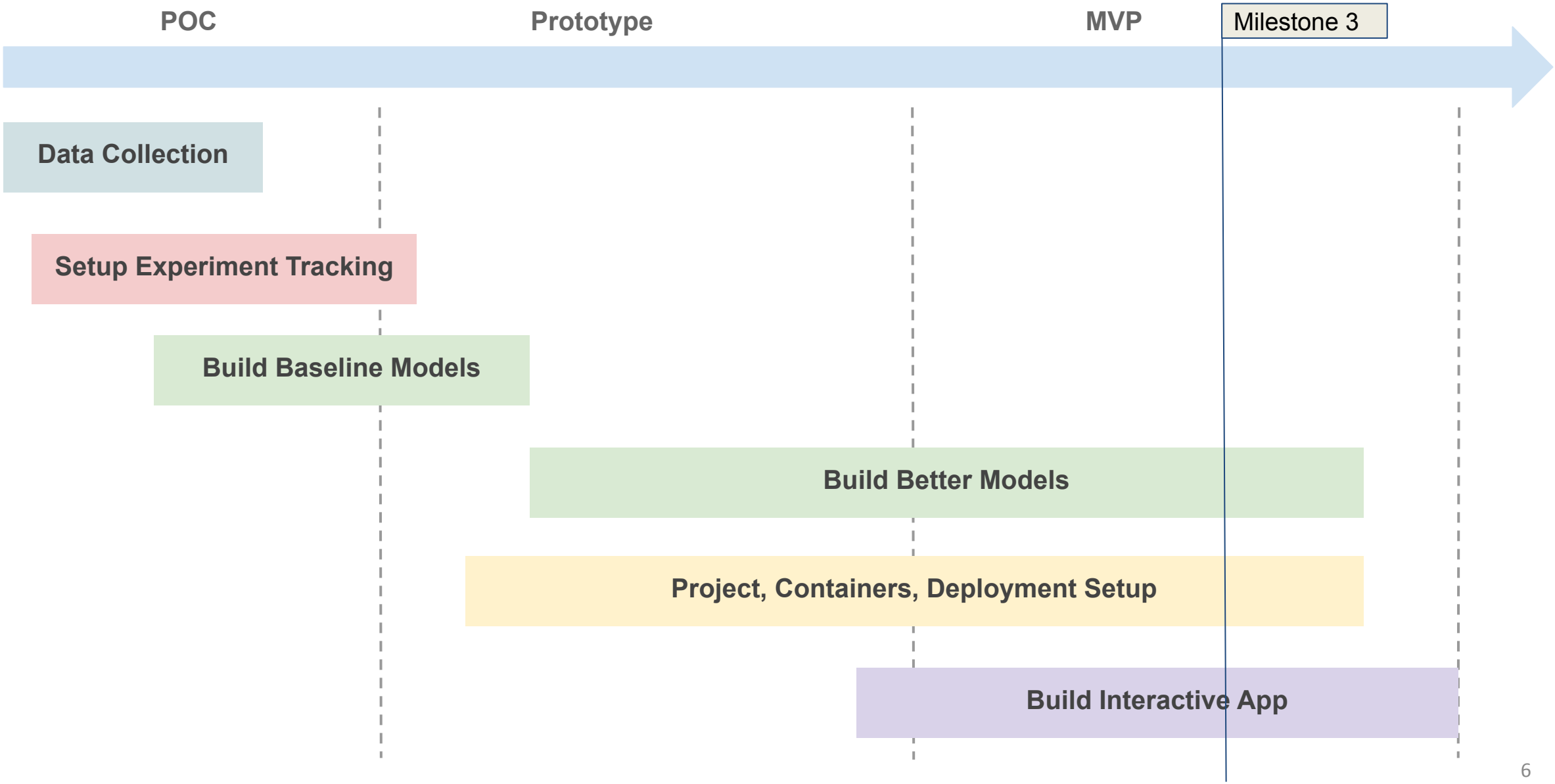
Prototype

- Create ‘looks like’ mockup of UX using figma
- Deploy one model to Fast API to service model predictions as an API

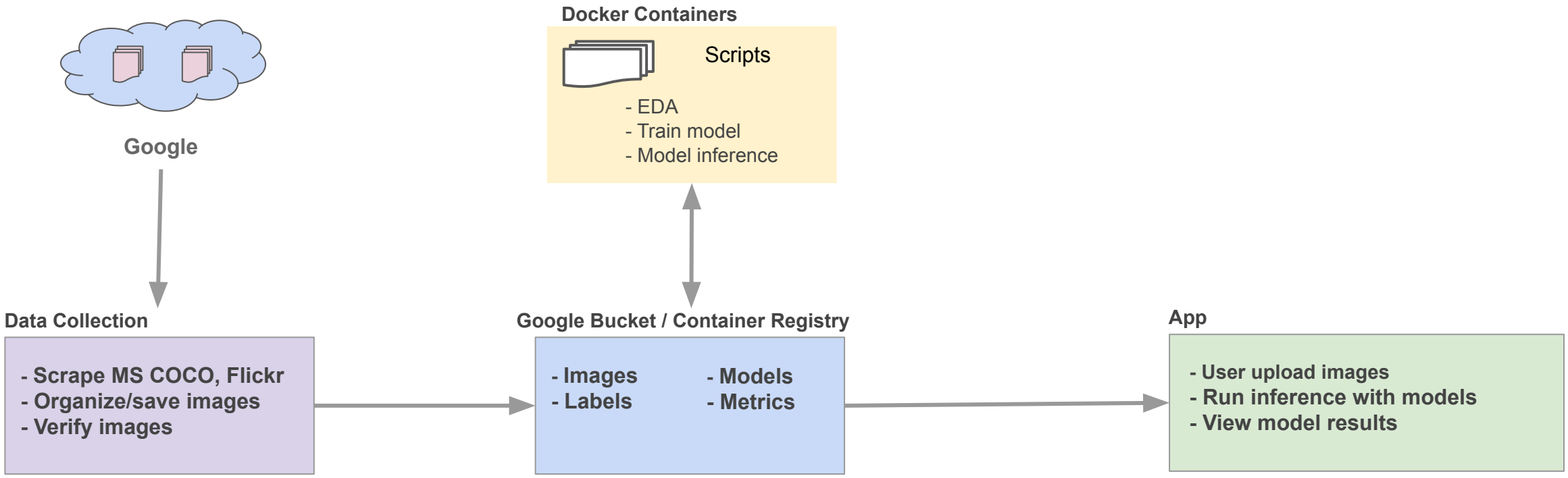
Minimum Viable Product (MVP)

- Create App that labels unseen photos
- API Server for uploading images and predicting using best model
- Deploy with Kubernetes on GCP

Project Workflow



Process Flow



Data

- Public Google bucket (link here) containing MS COCO and Flickr 8K datasets used during this project.
- **Flickr 8K:** 8,091 images from one of six categories, each with 5 corresponding image captions.
- **MS COCO (2014):** 164K images split into training (83K), validation (41K) and test (41K) sets.
- Both datasets are standardized datasets used for benchmarking and released under CC0 license (public domain).

Data Example

A brown dog in two black collars running through a grassy field .

a small brown and black dog lying down in a furry rug .

A man on the street standing by his bicycle .

Friends and family dance on a beach by their vehicles .

A man feels on top of the world on top of a large rock formation .

A dog leaps into the air to catch a ball in its mouth .

A dog leaps into the air to catch a ball in its mouth .

Some children watching fish in a pool .

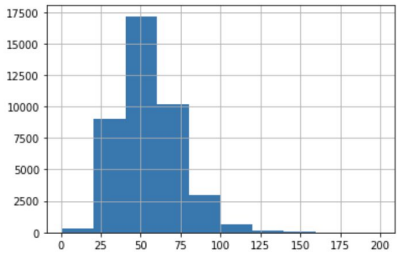
Two gray dogs jump at each other over the tall grass .

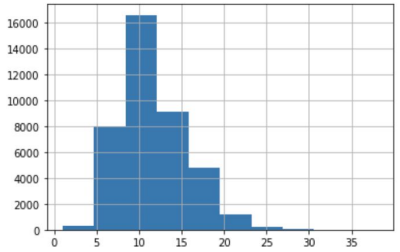
Data Details

- Flickr8K
  - Total number of images: 8091
  - Label counts: 40455
- MS-COCO
  - 2014 split: 83K training + 41K validation
  - Label counts: 616K

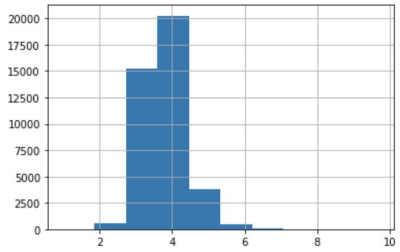
Caption Data Analysis

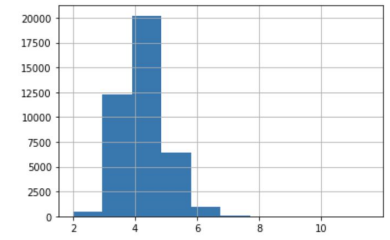
Captions are generally between 25 and 100 characters and 5 to 20 words



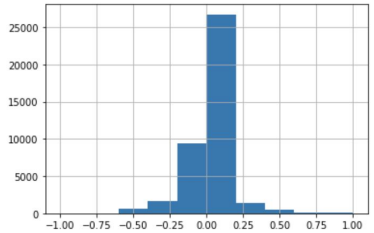


with an average word length of 3 to 4 characters (with and without stopwords)

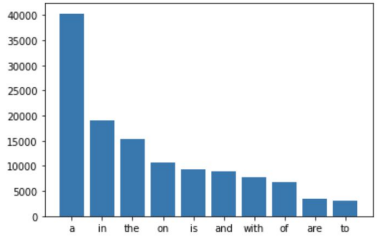




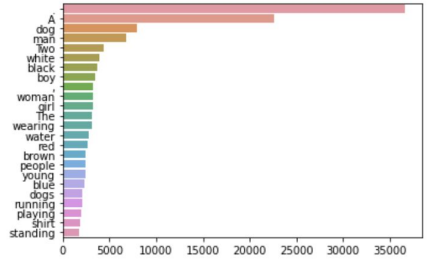
the majority of the sentiment polarity scores cluster around zero meaning most are pretty neutral



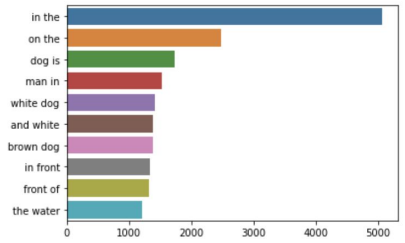
Breaking down captions into individual words top stop words are

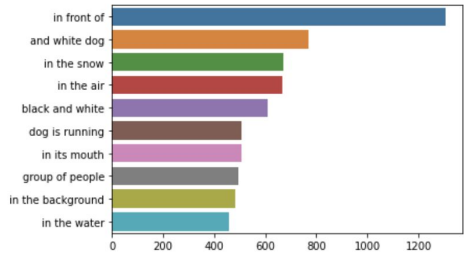


and plotting lemmatized words removing stopwords shows the top words are 'dog' and 'man'



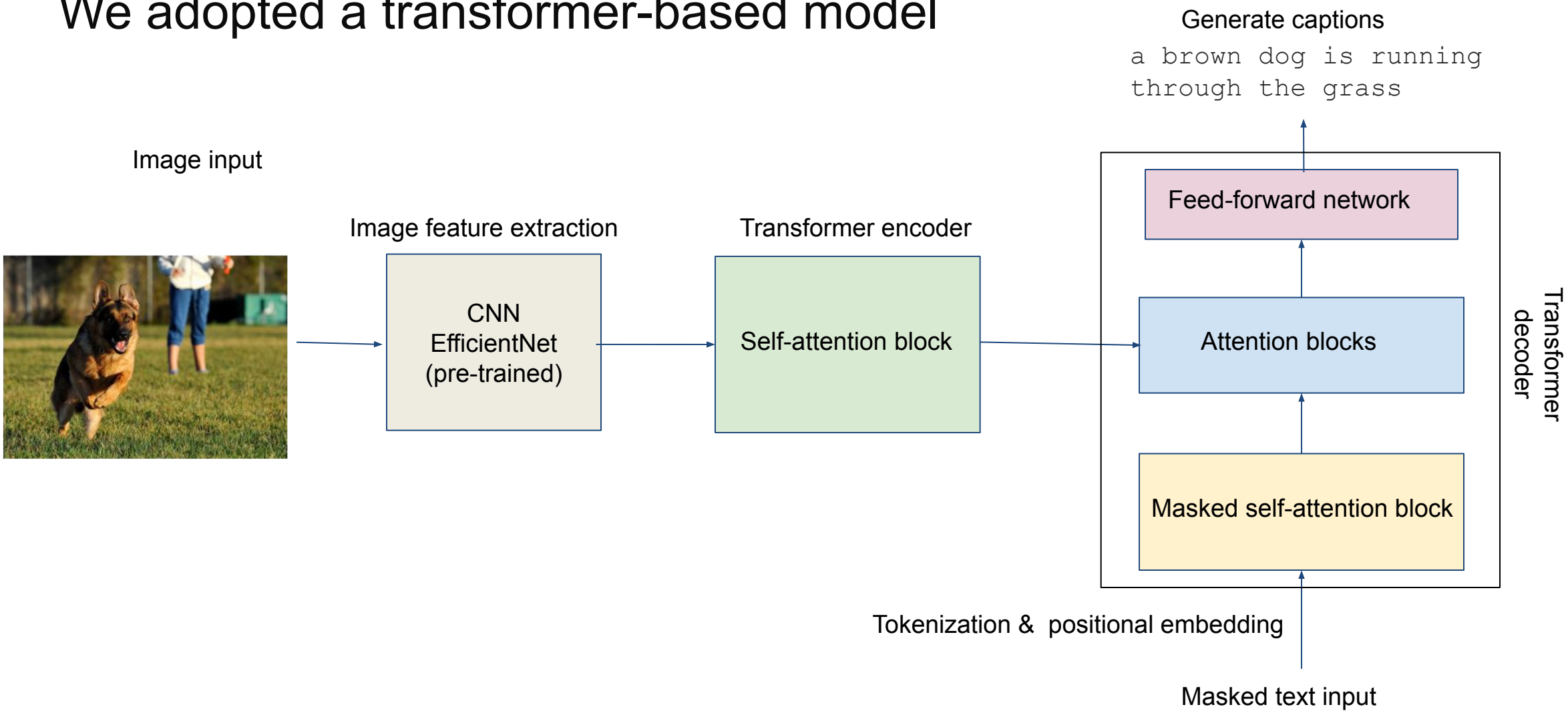
using ngram exploration, we can look at which pairs of words come up the most, and which trigrams





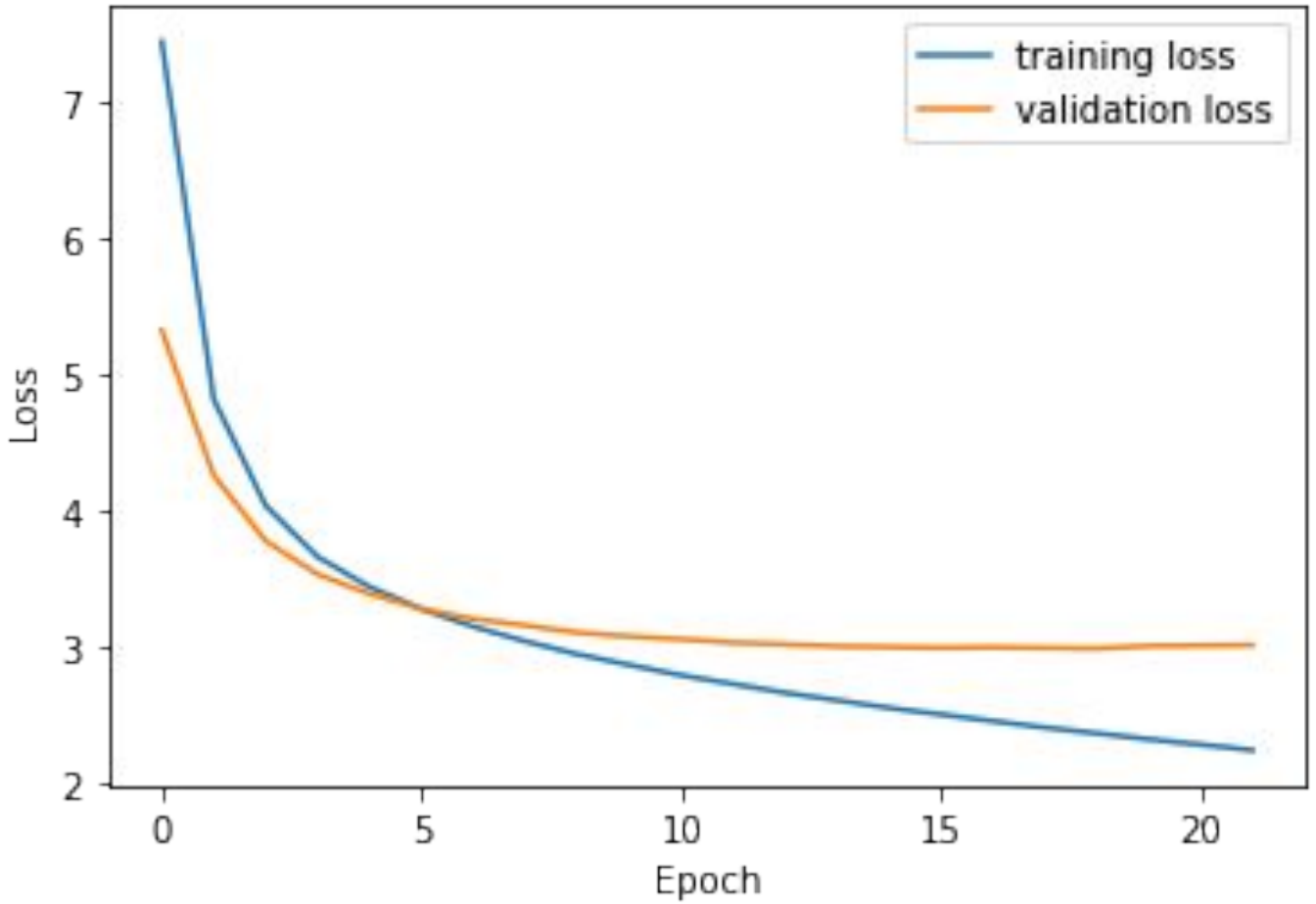
Baseline Model

We adopted a transformer-based model



Models - Training Progress

Trained on 72% of Flickr8k data, validated on 18% (10% saved as test data)



Test Results

Generate captions on test images

a black dog is running on the beach

two children playing in the snow

a little boy is playing with a toy on the grass

a soccer player is running on the field

a man is sitting on a wooden dock near a lake

a red boat is being ridden over a wave

Current Best Model

Improvement from baseline model:

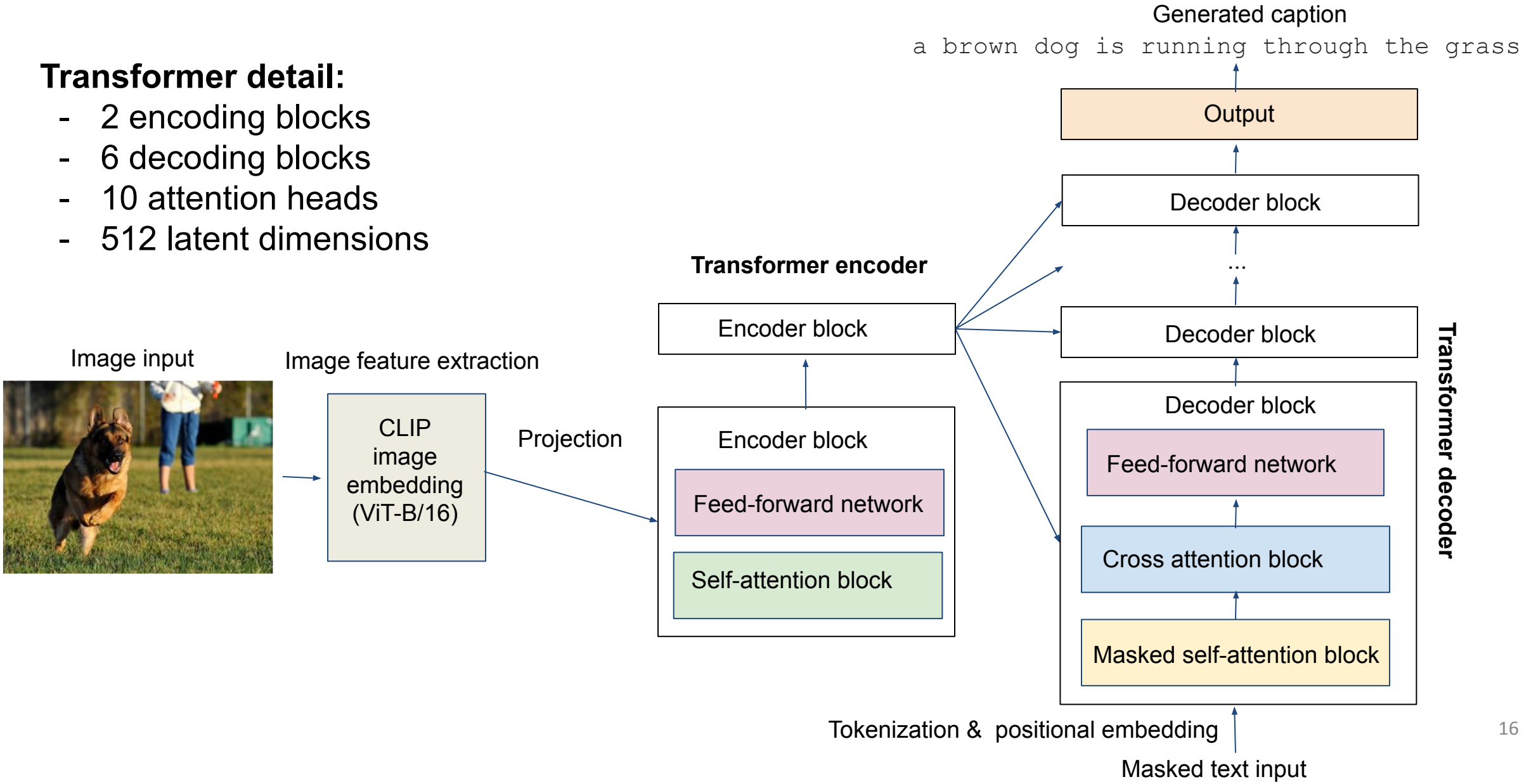
- CLIP embedding for images
  - Instead of CNN for pre-processing, we adopted OpenAI CLIP for generating image embedding (https://openai.com/blog/clip/)
  - CLIP (*Contrastive Language–Image Pre-training*): trained on minimizing contrastive loss between a large number of image-caption pairs; generate better embedding for representing details of images

- Larger transformer architecture
  - 2 encoding blocks, 6 decoding blocks, 10 attention heads for each block, 512 latent dimension

Model Architecture

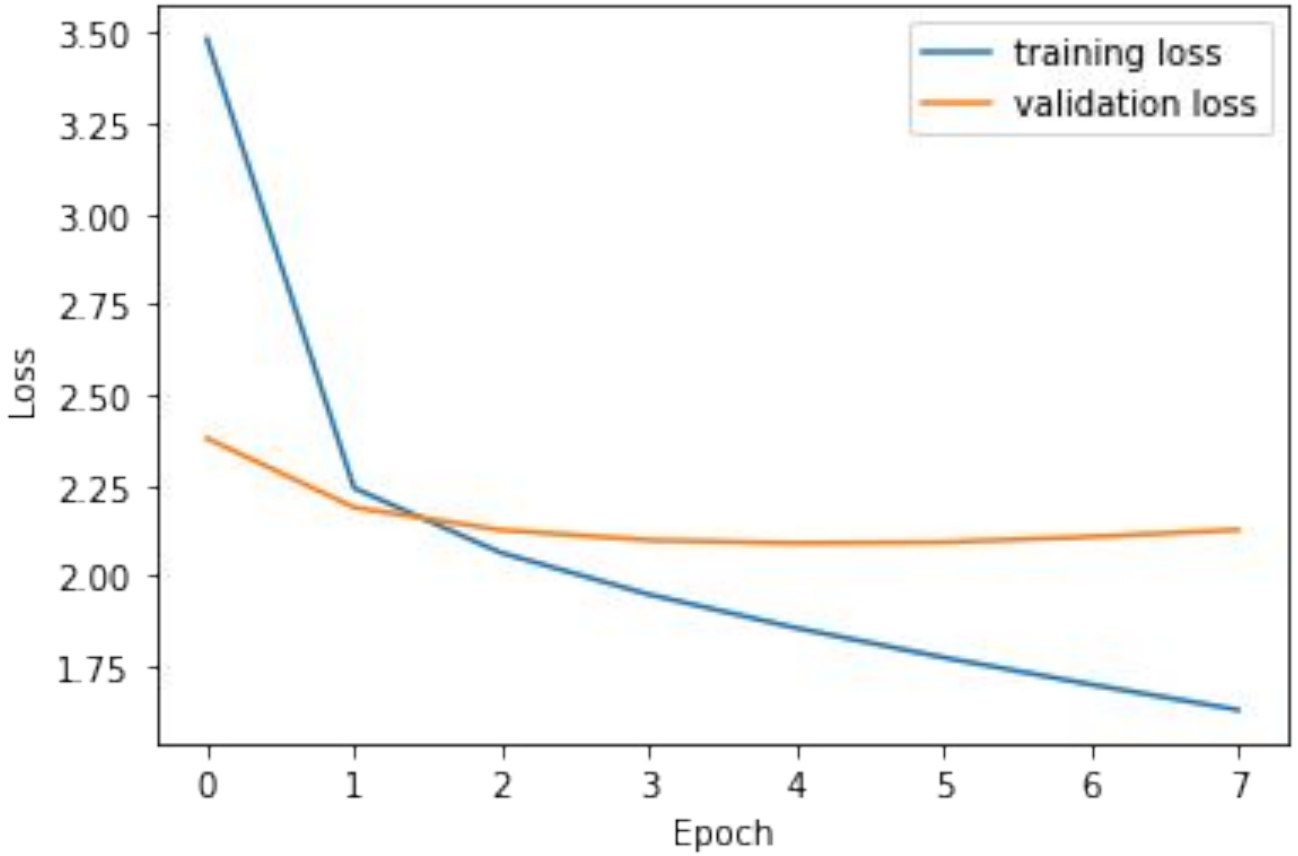
Transformer detail:

- 2 encoding blocks
- 6 decoding blocks
- 10 attention heads
- 512 latent dimensions



Training

Trained on 80% of Flickr8K + MS-COCO data (591K), validated on 10% (657K), with the rest 10% was saved as test data



Test Results - Generated Captions on Example Test Images

a cat sitting on a bed next to a stuffed animal

a dog sitting in the back seat of a car

two people in uniform are on a boat

a man in a red jacket is skiing in the snow

two giraffes standing in a field with trees in the background

a man in a kitchen preparing a sandwich

App Design

- API Server:
  - Fast API, serving the current best model

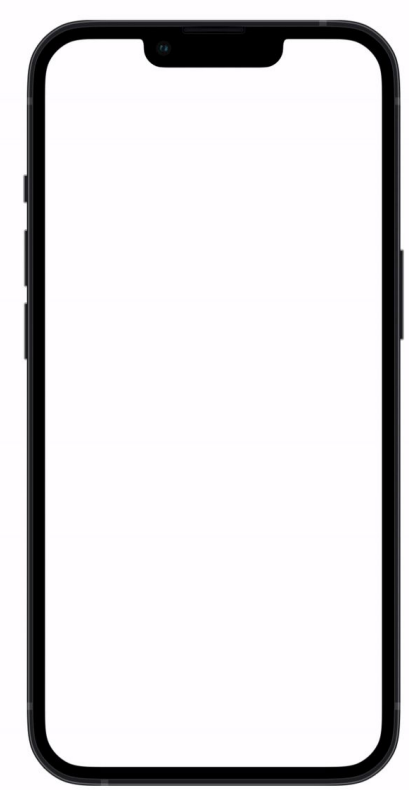
- React Frontend
  - Interacting with user for uploading images, sending image to API server, and displaying the generated caption

- NGINX
  - Bridging API server and react frontend, exposing the App to web

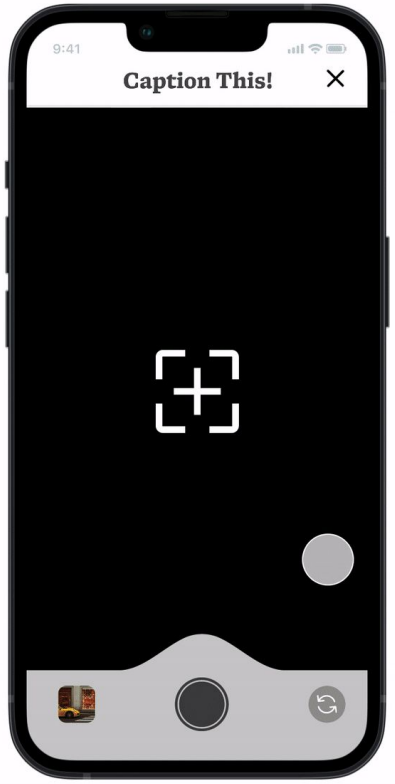
App Design (in progress)

User logs into app and clicks ‘Get Started’

App Design (in progress)

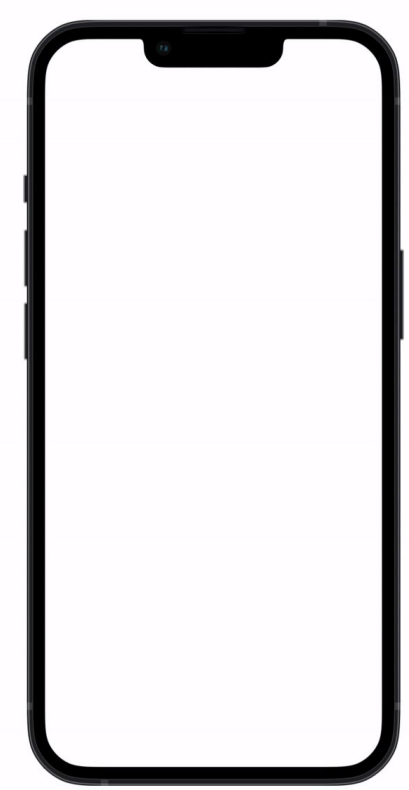


User logs into app and clicks ‘Get Started’

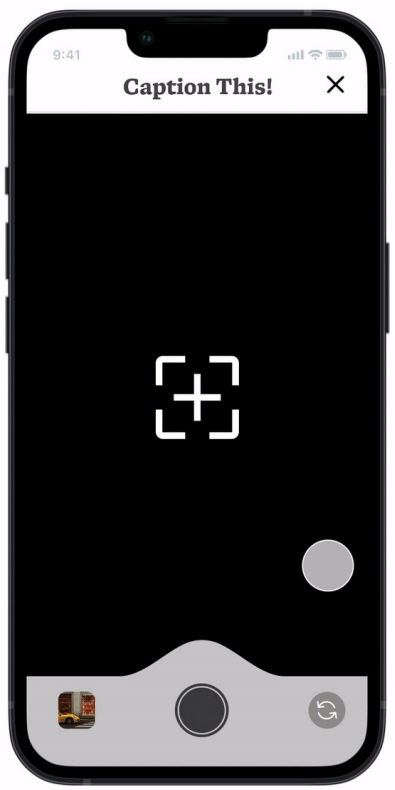


User takes photo and caption is generated

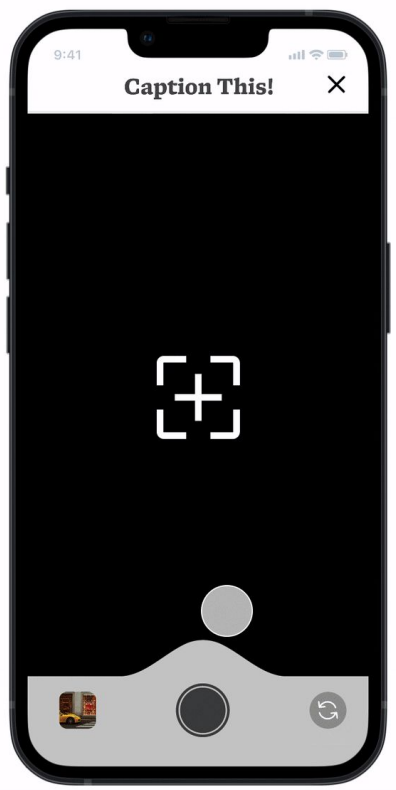
App Design (in progress)



User logs into app and clicks ‘Get Started’



User takes photo and caption is generated

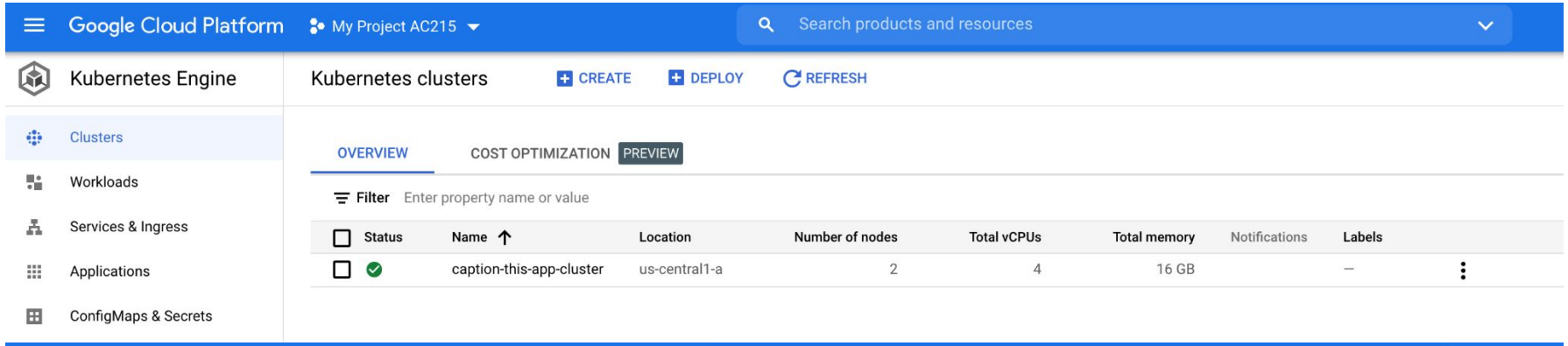


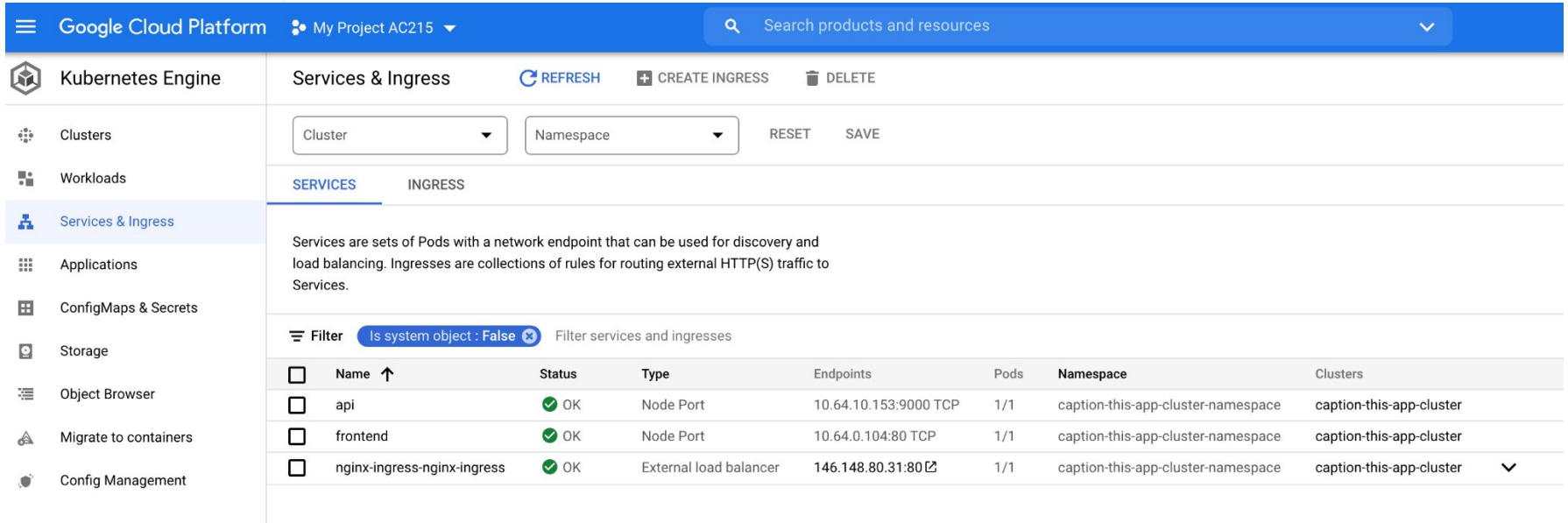
Or user uploads photo and caption is generated

Deployment

We have deployed the model-serving API, the React Frontend, and Nginx service with Kubernetes cluster using Ansible

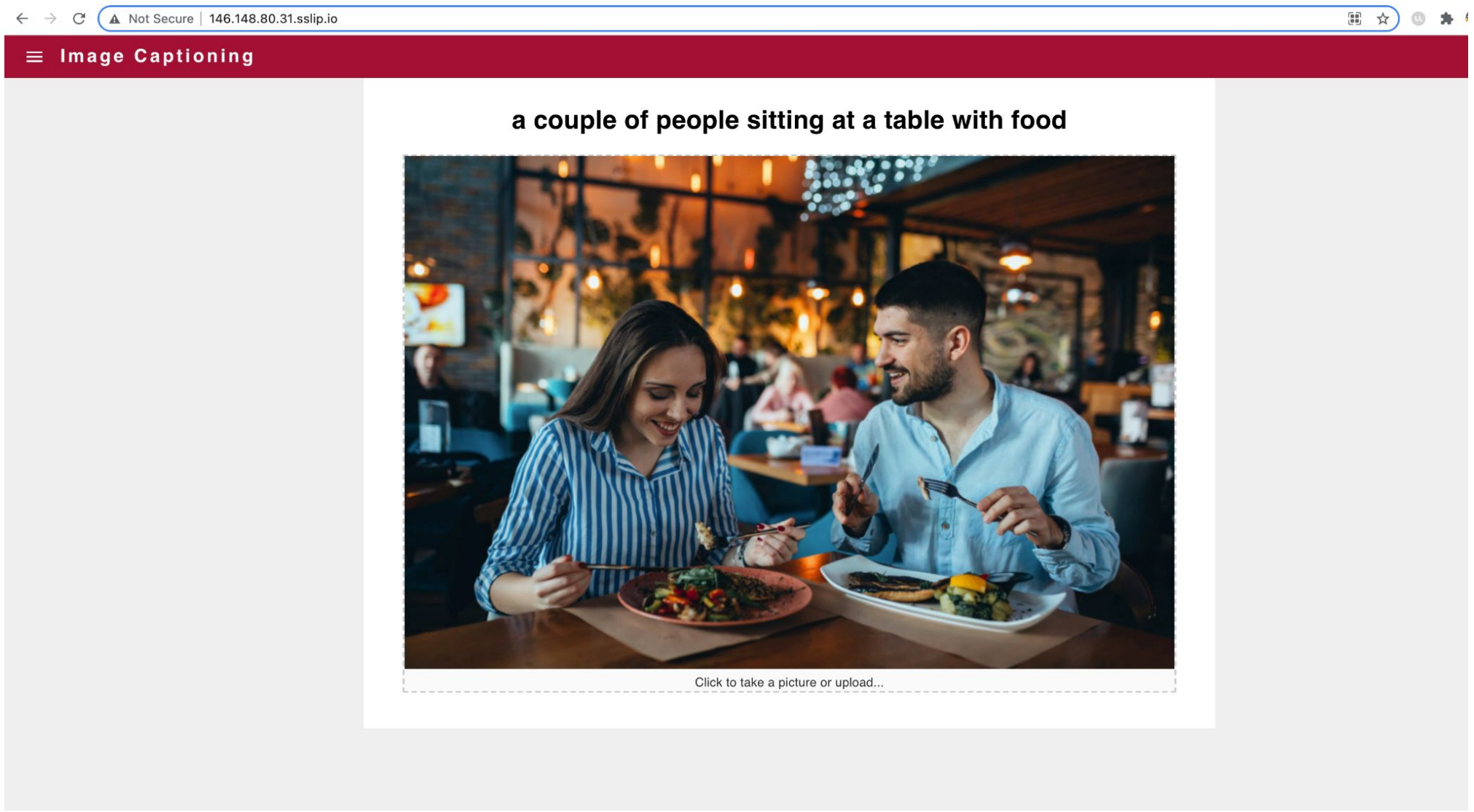
Screen shots:





Deployment

Screen shot of the deployed App on external web:

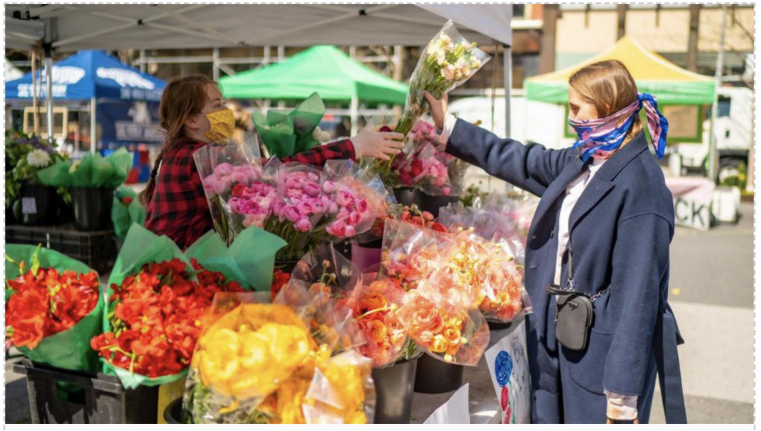


Deployment

More examples with random images downloaded from internet:

a couple of people walking across a park

a woman holding a flower in a flower shop



a group of children are sitting in a classroom

two giraffes eating from a womans hand while a child holds their hands out to the side

a plate of pasta with shrimp and vegetables

a group of people sitting at a bar

Supporting Notebooks in Repo

- Link to the repo:
  https://github.com/skgithub14/AC215_KKST
- EDA + Baseline Model:
  https://github.com/skgithub14/AC215_KKST/blob/main/submissions/milestone2_KKST/Milestone2_EDA_with_baseline_models.ipynb
- Current Best Model:
  https://github.com/skgithub14/AC215_KKST/blob/main/notebooks/Transformer_based_image_captioning_with_CLIP_embedding.ipynb