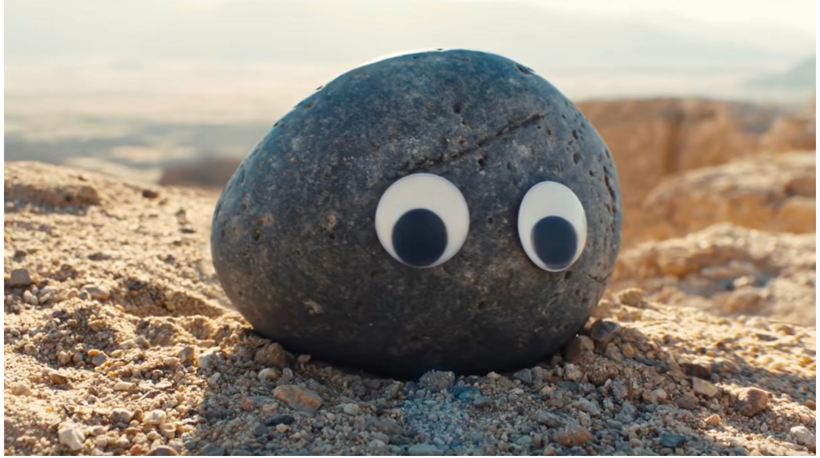
Image Caption

Final project for MSDS 631 Introduction of Deep Learning

Karsten Cao & Chenjia Guo

Image Captioning



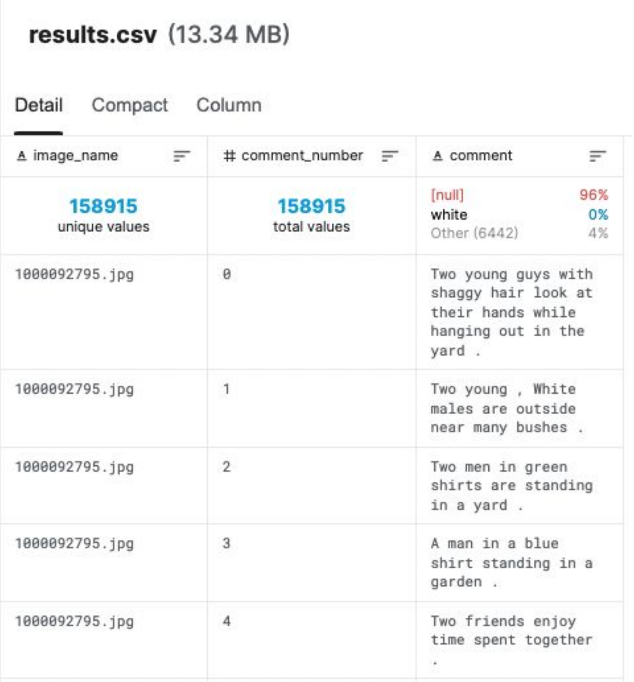
Caption Goal:

A rock with googly eyes

Results:

<SOS>

Data Source: Flickr Image Dataset | Kaggle

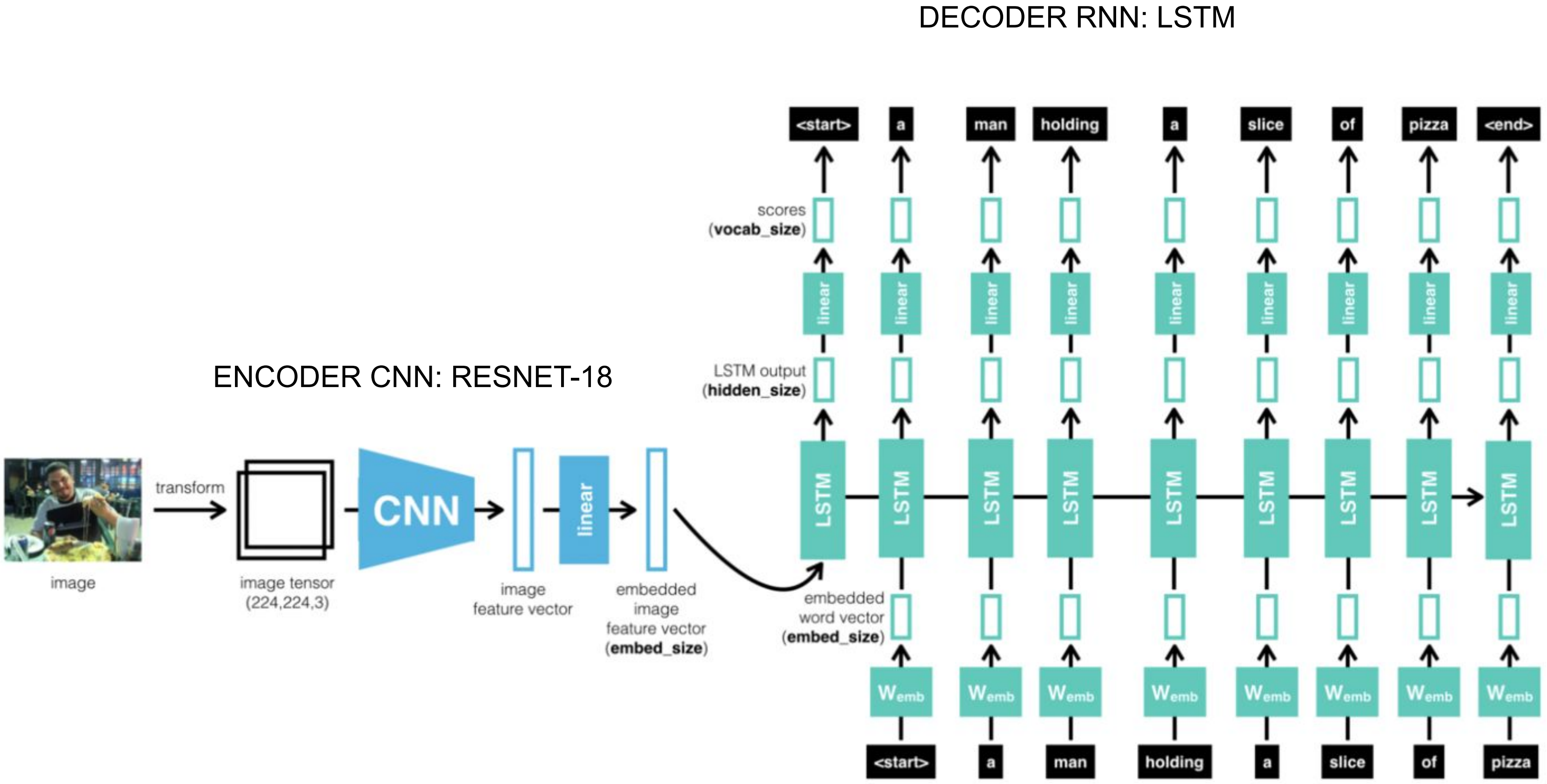


results.csv (13.34 MB)

image_name	comment_number	comment
158915 unique values	158915 total values	[null] 96% white 0% Other (6442) 4%
1000092795.jpg	0	Two young guys with shaggy hair look at their hands while hanging out in the yard .
1000092795.jpg	1	Two young , White males are outside near many bushes .
1000092795.jpg	2	Two men in green shirts are standing in a yard .
1000092795.jpg	3	A man in a blue shirt standing in a garden .
1000092795.jpg	4	Two friends enjoy time spent together .

https://www.kaggle.com/datasets/hsankesara/flickr-image-dataset

The CNN-RNN Architecture



https://medium.com/@deepeshrishu09/automatic-image-captioning-with-pytorch-cf576c98d319

Applied Techniques

Token Padding:

- Allows for batch training
- Freedom of caption length (unnecessary)
- Negatively affects the loss function

Drop <EOS>:

- The model will naturally decide where to place their own stop
- Doing so made sentences significantly shorter.

Tokenizing/Lemmatizing:

- Trade off between topic versus clarity. Removing stopwords retained topical information but lost all potential for coherent captions.

Different Encoder:

- Due to other issues, there was not much of a difference between Resnet18 and Inception.

More Epochs:

- Generated texts found different heuristics.

CNN output RNN input

- Different papers different methods. Some input it in a cell state, some in the hidden, and some as input.
- Not enough time to discern differences

Data Augmentation

- Helped with the noun identification

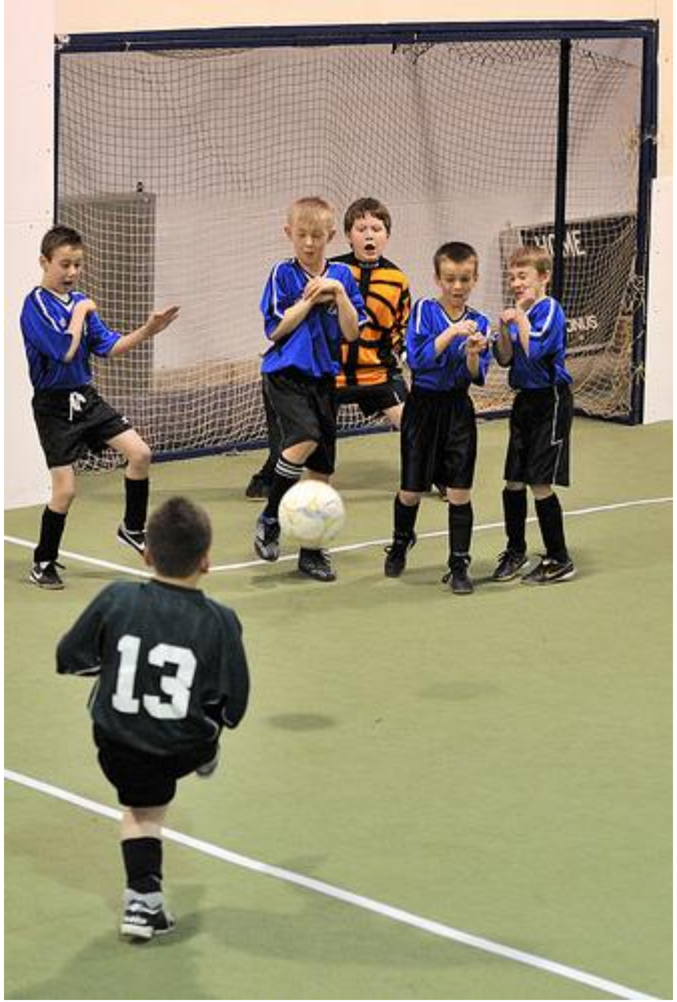
Results and Performance

10 epochs with <EOS>

“<SOS> shirt <UNK> shirt shirt shirt

with a ball <EOS>”

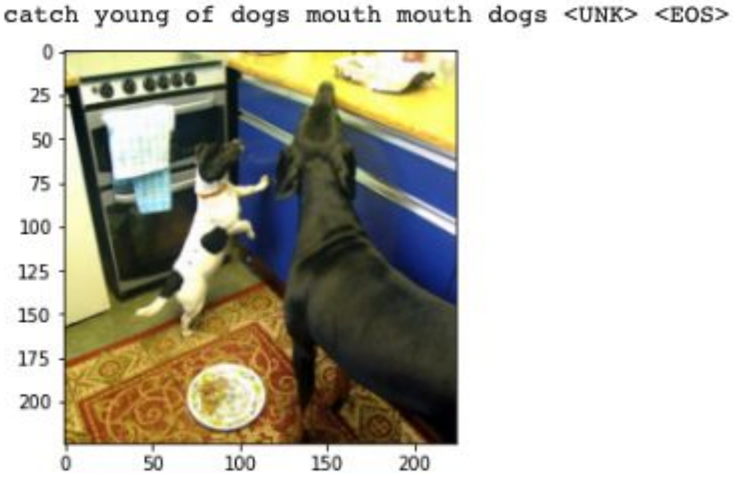
Average log loss: 3.10



Results and Performance

Drop <EOS>, padding, 5 epochs

Average Log Loss: 3.332



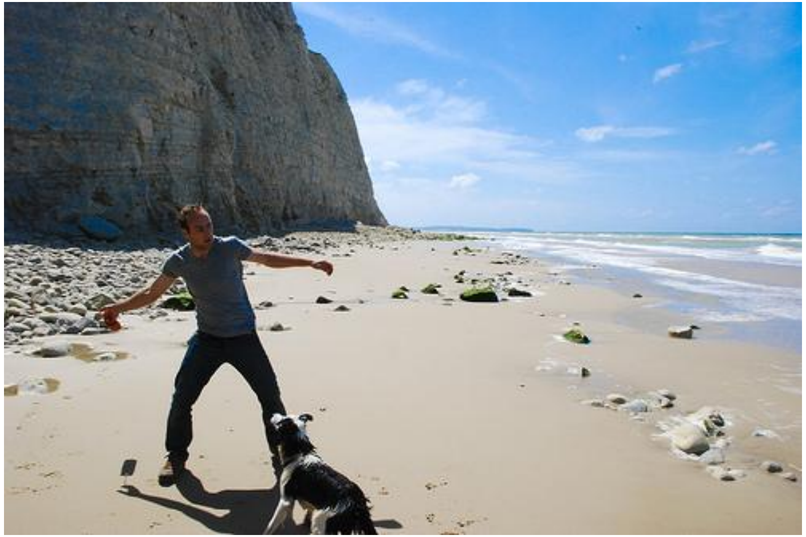
```
catch young of dogs mouth mouth dogs <UNK> <EOS>
```

Results and Performance

Can identify nouns, cannot work with grammar.

Answer: <SOS> a man playing fetch with his dog on a beach <EOS>

Predicted: <SOS> a man is boy with a dog that is running in the off boy <EOS>



Issues

- Putting the models together
- Loss function (CE) was too simple
- Research groups normally took days to train their models
- Scope limitation, vocabulary and image recognition

Future Possibilities

- **Increase LSTM layering**
- **Use subword tokenizers**
- **Create more layers after the pretrained CNN**
- Add in attention
- Train for longer
- Use different loss functions, for example BLEU or use embedding comparisons.

Summary

- Image to Caption contains two hard problems: Image Recognition and Text Generation
- Solution: CNN to RNN architecture
- Explosion of parameters to tune and methods to use

Reference Paper/Work

Kaggle Notebook:
https://www.kaggle.com/code/shourabhpayal/cnn-lstm-pytorch-image-captioning/notebook
https://www.kaggle.com/code/sauravmaheshkar/neural-image-captioning

https://github.com/sauravraghuvanshi/Udacity-Computer-Vision-Nanodegree-Program/blob/master/project_2_image_captioning_project

Multi-Modal Methods: Image Captioning (From Translation to Attention):
https://blog.mlreview.com/multi-modal-methods-image-captioning-from-translation-to-attention-895b6444256e
https://medium.com/@deepeshrishu09/automatic-image-captioning-with-pytorch-cf576c98d319