Comparing Image Captioning Models

Nisarg Patel, Trevor Powers, James McKee

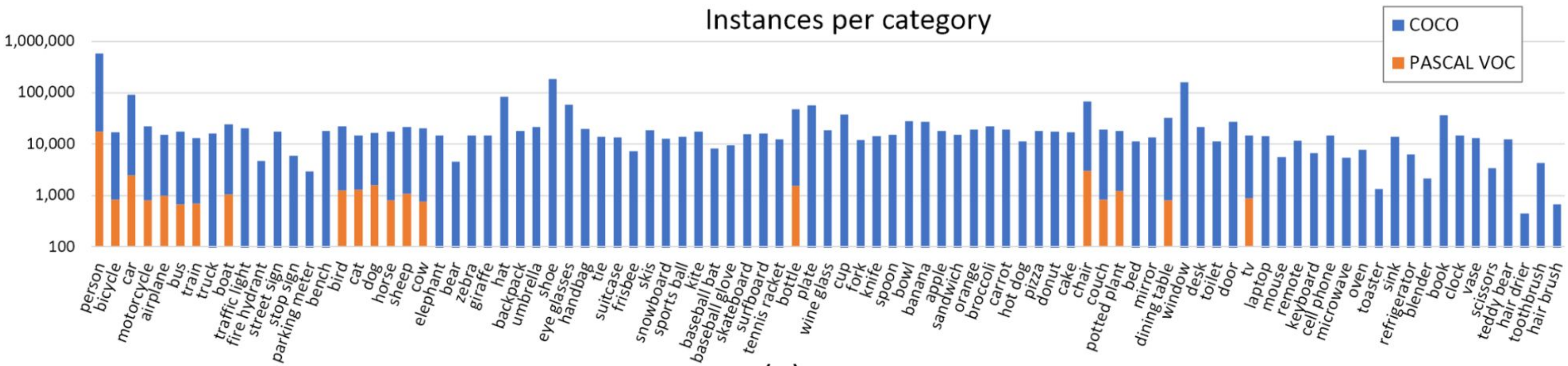
Image Captioning

- Problem: Given an image, a machine learning model must output a sequence of text describing what is happening
- Uses: Wide range of applications - virtual assistants, technology for the disabled, image database quieres
- ML Tasks: Computer Vision and Natural Language Processing

Meta

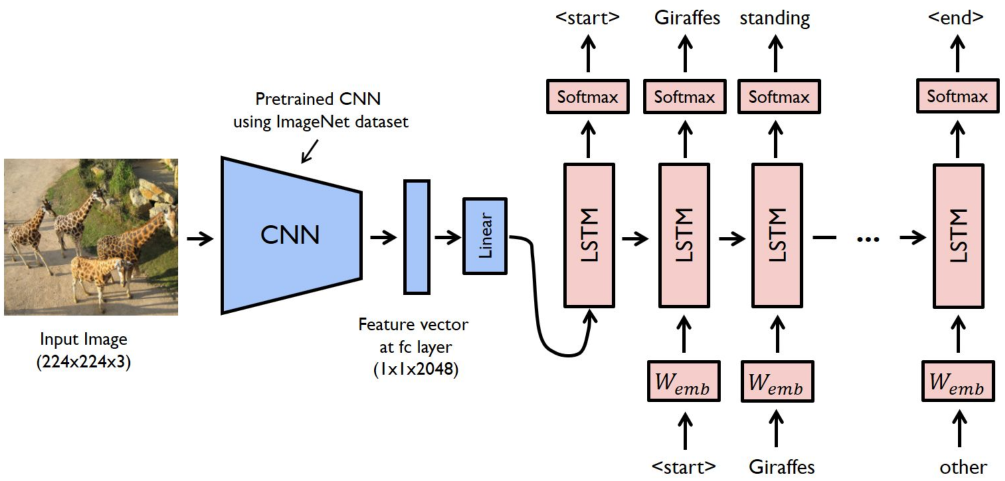
The Dataset

- Microsoft COCO Dataset - ‘Common Objects in Context’
- Largest Image Captioning Dataset Easily Accessible, significantly surpasses other major datasets (see below)
- Contains 5 captions per image
- We use 80k training images, 20k validation images



Encoding - Decoding

- Image Captioning Consists of Two ML Tasks
  - Computer Vision
  - Natural Language Processing
- Encoder: Uses a CNN to extract features from the input image, relies on computer vision
- Decoder: Uses a model to take in extracted image features and output an image caption, relies on natural language processing



The Models

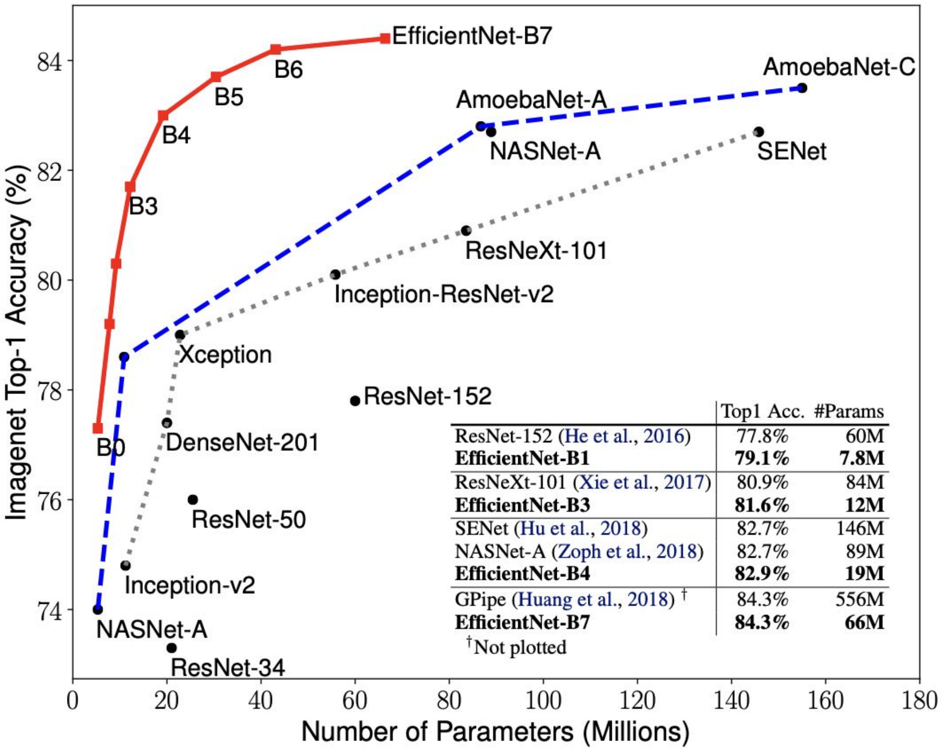
This project makes use of the following models:

- Encoder
    - CNN: EfficientNetB0
- Decoders:
    - RNN: Basic Decoder
    - LSTM: More Complex Decoder
    - Transformer: State-of-the-Art Decoder

EfficientNetB0

- Top performing, low computational expense CNN
- Uses compound model scaling to select model parameters used in this project

$$
\begin{aligned}
\text{depth: } & d = \alpha^{\phi} \\
\text{width: } & w = \beta^{\phi} \\
\text{resolution: } & r = \gamma^{\phi} \\
\text{s.t. } & \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \\
& \alpha \geq 1, \beta \geq 1, \gamma \geq 1
\end{aligned}
$$

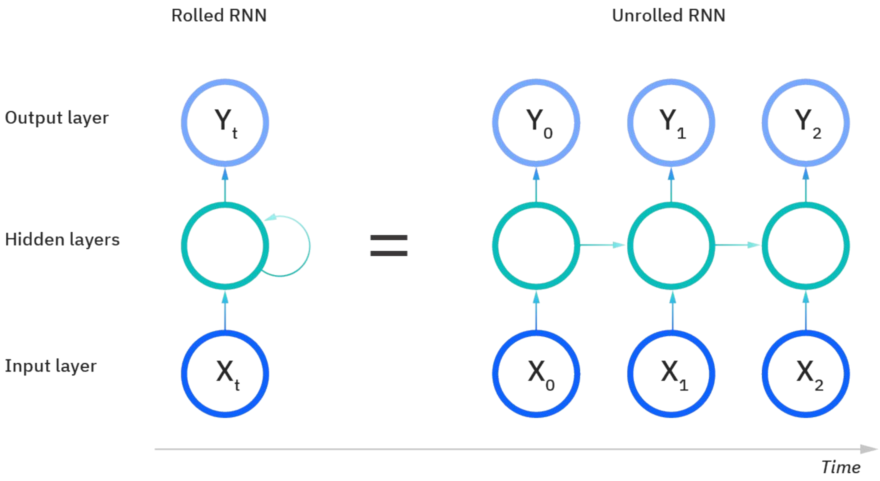


	Top1 Acc.	#Params
ResNet-152 (He et al., 2016)	77.8%	60M
EfficientNet-B1	**79.1%**	**7.8M**
ResNeXt-101 (Xie et al., 2017)	80.9%	84M
EfficientNet-B3	**81.6%**	**12M**
SENet (Hu et al., 2018)	82.7%	146M
NASNet-A (Zoph et al., 2018)	82.7%	89M
EfficientNet-B4	**82.9%**	**19M**
GPipe (Huang et al., 2018) †	84.3%	556M
EfficientNet-B7	**84.3%**	**66M**

†Not plotted

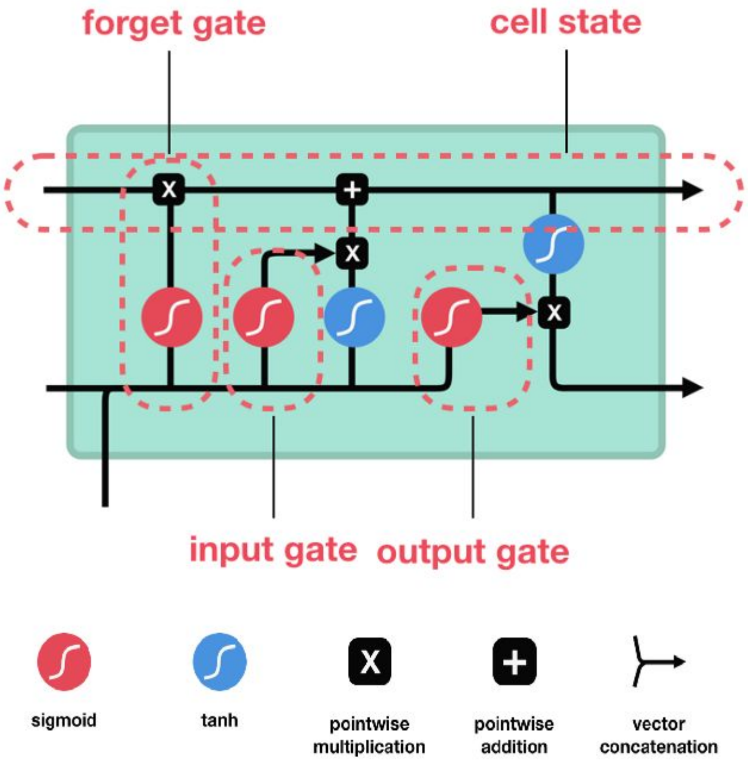
Decoder 1: RNN

- Sequences are input one-at-a-time
- Feeds output of each hidden layer in next layer
- Learns short-term dependencies
- Limited



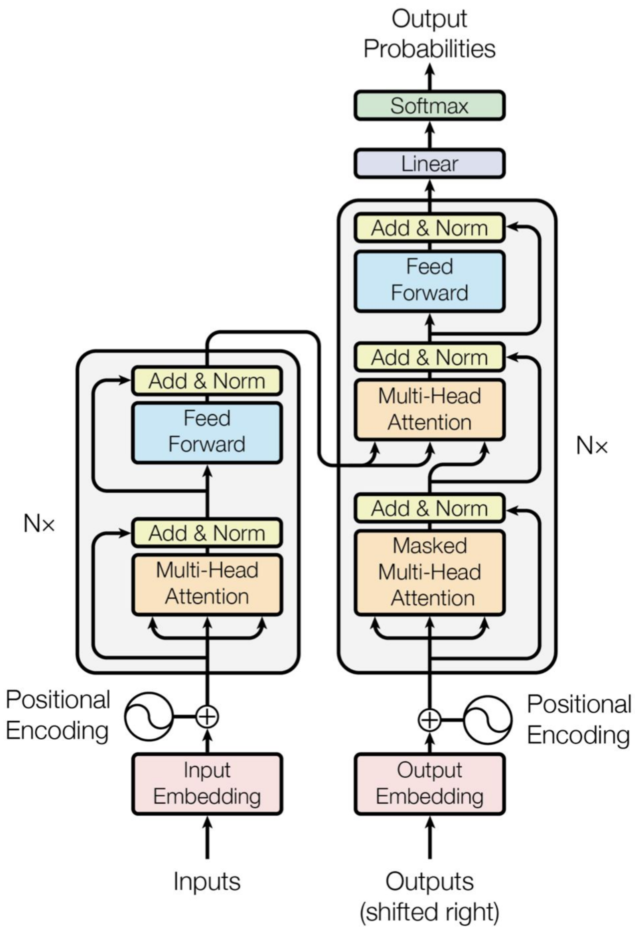
Decoder 2: RNN-LSTM

- Long-Short Term Memory
- Capable of learning long-term dependencies which solving the vanishing gradient problem.
- Flow of information in cell state is regulated by 3 gates.
- Input is the sequence of words in a caption.
- Output is the probabilities of next word in the sequence.
- Around 25 training instances for each caption (Slow to learn).



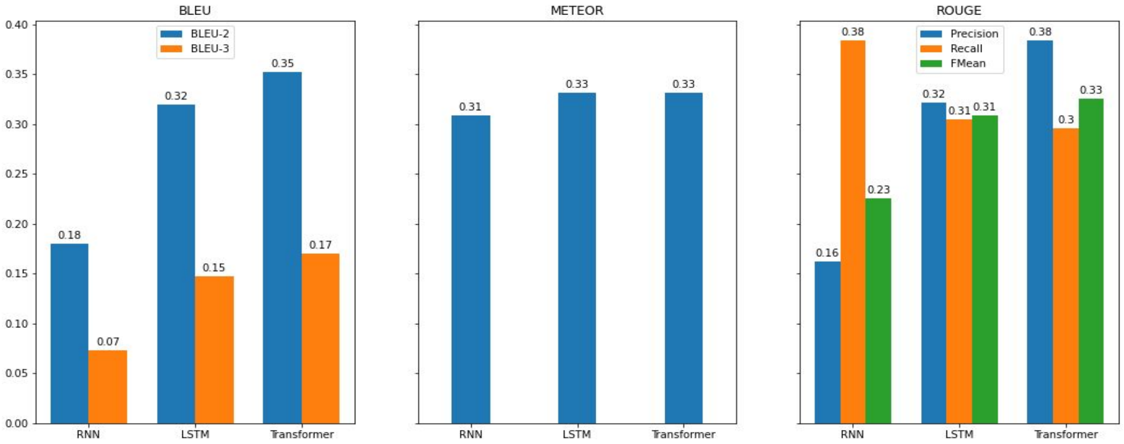
Decoder 3: Transformer

- Google AI Transformer Model considered state of the art for many NLP tasks (BERT etc.)
- We use the Keras.io implementation of the transformer designed by Google AI
- Transformer is does not rely on sequence input order, unlike RNNs (expect faster training and higher accuracy)



The Results

- Clearly superior performance by the LSTM and Transformer models
- Interesting recall by the RNN model, likely caused by lengthy predictions
- Much better precision by LSTM and Transformer
- Transformer has a moderate edge over LSTM
- Notable disparity in training times



Example Captions

REF: - A very large clock tower on the side of a church.
- An old building with a clock at the top of it.
- The tall clock tower is built into the corner edge of the building.
- An old building has a clock tower with a weather vane.
- An old building's clock tower is being displayed.

RNN: ben clock tower with a clock on it and a clock tower in the background of a building with a clock tower in the background
LSTM: a large building with a clock tower in the middle of it
Transformer: a clock tower is shown in the middle of the street

REF: - A polar bear sitting on some rocks.
- A large white polar bear sitting on top of a rocky ground.
- A polar bear sitting on a stone in his exhibit.
- A polar bear sits in the sun and dries off.
- A polar bear sitting on some ice by a fence.

RNN: polar bear standing by a pool of water with its paws on its paws on a persons legs of its paws on a shoe on
LSTM: a polar bear is laying down on a rock
Transformer: a polar bear is sitting on a rock

Example Captions

REF: - a wireless computer mouse on a wooden table
- A cat tail next to a mouse
- A mouse kept just besides a cat's tail.
- A cat with it's paw next to a computer mouse on a wooden table.
- A computer mouse with a cat's paw next to it.

RNN: deserts on a plate on a table with a knife on it and a cup of coffee on it and a cup of coffee on
LSTM: a person holding a cell phone in their hand
Transformer: a close up of a mouse on a table

REF: - A photo of two people skiing on the snow.
- A man skiing on snow besides a child.
- A picture of a couple people skiing in the snow.
- Two people in the distance skiing against the horizon.
- Two skiers in the distance under a cloudy sky.

RNN: people are flying kites on a beach near the ocean with a man on the beach and a kite in the background and a man
LSTM: a man is flying a kite in the sky
Transformer: a person on a snowboard is jumping over the ocean

Future Work

- Explore Other Encoder Options
  - Other Efficient Net Versions
  - Other CNNs
- Explore More Transformer Attention Heads
- Consider other validation metrics
  - SPICE
  - CIDEr
- Combine more datasets (Flicker8k, etc.)

References

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation, 2016. URL https://arxiv.org/abs/1607.08822.

Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In Jade Goldstein, Alon Lavie, Chin-Yew Lin, and Clare R. Voss, editors, *IEEvaluation@ACL*, pages 65–72. Association for Computational Linguistics, 2005. URL http://dblp.uni-trier.de/db/conf/acl/ieevaluation2005.html#BanerjeeL05.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

Keras.io. Keras documentation: Image captioning. URL https://keras.io/examples/vision/image_captioning/#model-training.

Chin-Yew Lin. Rouge: a package for automatic evaluation of summaries. 2004.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014. URL https://arxiv.org/abs/1405.0312.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. 2019. doi: 10.48550/ARXIV.1905.11946. URL https://arxiv.org/abs/1905.11946.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017. URL https://arxiv.org/abs/1706.03762.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4566–4575, 2015. doi: 10.1109/CVPR.2015.7299087.