Descriptive Image Captioning

Using Deep Learning to generate captions for images

Index

- Project Aim
- Approach
  - Dense Captioning
  - Combining Geometric, Textual and Image Features
  - Encoder-decoder model
- Results
- Conclusion and Future Work
- References

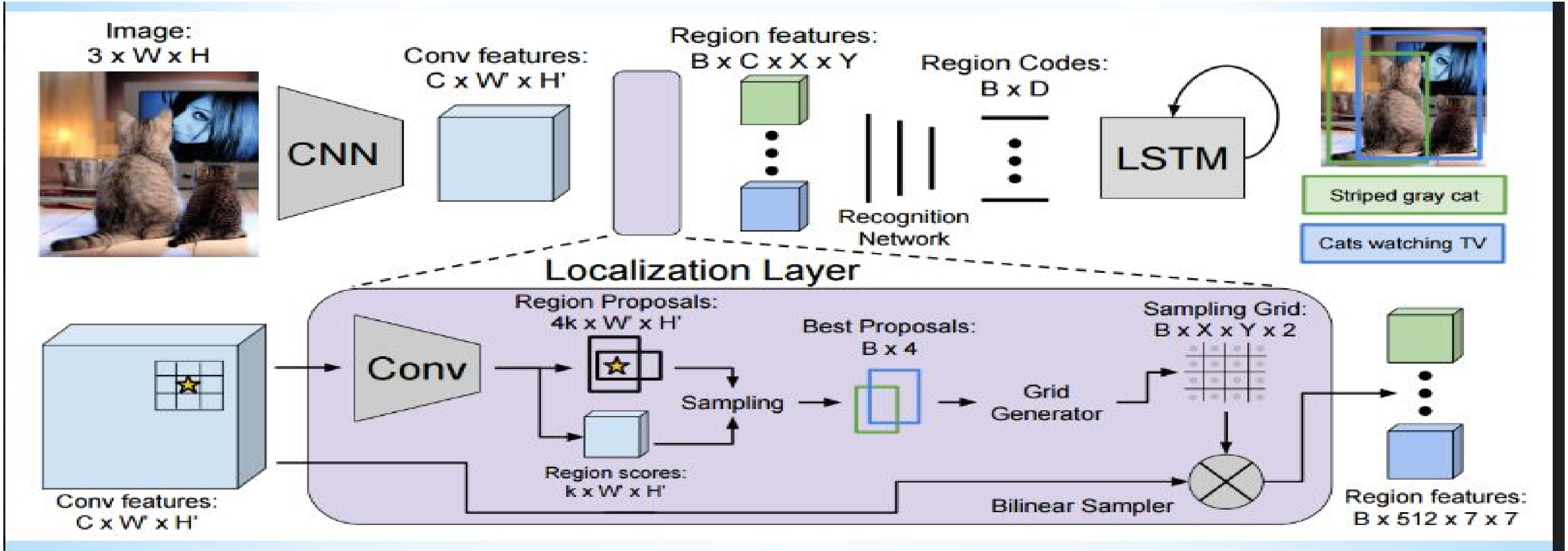
Project Aim

- Image Captioning aims to fill up the gap between visual and language interpretations which shall find wide applications among robots and humans as well.
- The notable work in this field was achieved by Dense Captioning which produces small captions for every region proposal.
- The aim of this project was to semantically combine the incomplete captions to generate a set of sentences describing the image in detail.

Approach

- We replicated the results of '**DenseCap: Fully Convolutional Localization Networks for Dense Captioning'** achieving results close enough as in the paper.
- We implemented the paper '**Combining Geometric, Textual and Visual Features for Predicting Prepositions in Image Descriptions**' to predict prepositions between two connecting nouns.
- We refined the captions by reducing the region proposals and predicted the preposition joining the incomplete captions.
- The captions and the prepositions were then passed through an encoder-decoder model trained on phrases and sentences to generate meaningful descriptions.

Dense Captioning



Convolutional Network ⇨ Localization Layer

⇩

Recognition Model ⇨ Language Model

Convolutional Network

- VGG-16 with 13 layers of 3x3 convolutions (stride 1 and pad 1) and 5 layers of 2x2 max pooling (stride 2 no padding)
- Assume we start with an input image of shape 224 and depth is 3 (RGB)
- Initial 2 layers of convolutions and max pooling -

- No of filters are increased and max pooling reduces the width and height. Thus, the tensor of features is of shape CxW’xH’ where C=512, H’=H/16=14, W’=W/16=14.
- Output Image – Set of uniformly sampled image locations can be seen.

A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Table 2: **Number of parameters** (in millions).

Network	A,A-LRN	B	C	D	E
Number of parameters	133	133	134	138	144

VGG- 16 layer information

Softmax Function

$$P(y=j|\mathbf{x}) = \frac{e^{\mathbf{x}^\mathsf{T}\mathbf{w}_j}}{\sum_{k=1}^{K} e^{\mathbf{x}^\mathsf{T}\mathbf{w}_k}}$$

The function is normally used to highlight the largest values and suppress values which are significantly below the maximum value.

Localization Layer

- It essentially identifies spatial regions of interest and smoothly extracts a fixed sized representation from each region.
- Input : Tensor of activations of size C x H’ x W’
- Internally selects B regions and returns three output tensors
  - Region Coordinates : Matrix Bx4 giving bounding box coordinates for each output region
  - Region Scores : Vector of length B with confidence score of each region.
  - Region Features :Tensor of shape BxCxXxY giving features for output regions
- Project each point in W’xH’ grid of input back into WxH image plane
- Consider k anchor boxes of different sizes centred at this projected point. For each k box, a confidence score and four coordinates are predicted.

Localisation Layer

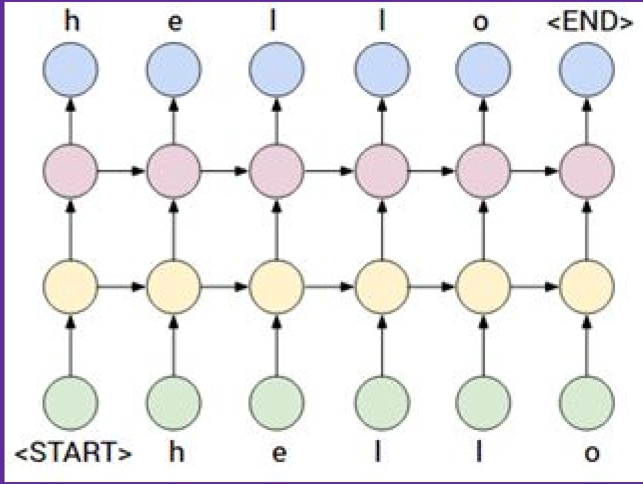
- **Computation:**
  Input feature Map → 3x3 conv with 256 filters → ReLU (Source of non-linearity) →1x1 with 5k filters →W'xH'x5k
- **Box Regression**: With coordinates, width and height of the center predicts scalars to normalise offsets and log space transforms to output region has center and shape.
- **Box Sampling:** Too many region proposals imposes the need to subsample them.
- **Training time :** A minibatch B=256 boxes with atmost B/2 positive regions and rest negatives.
- **Test time :** As sample B=300 of most confident proposals is used.
- **Bilinear Interpolation-**Bilinear sampling grid (B x X x Y x 2) is a linear function of the proposal coordinates

Recognition Network

- Features from each region are flattened into a vector.
- It is then passed through 2 Fully Connected Layers, each with ReLu (source of non-linearity) and regularized using Dropout.
- Each region produces a code of dimension D=4096 that compactly encodes its visual appearance.
- Codes for all positive regions is collected and put in matrix BxD which is passed to RNN
- The network also refines the confidence and position of each proposed region.

Language Model

- Training sequence of tokens $s_1, .., s_t$ is fed to RNN with + 2 word vectors $x_{-1}, x_0 \ldots x_t$ where x_{-1}=CNN(I) & x_0 is start token.
- RNN computes a sequence of hidden states h_t and output vector y_t using formula $h_t, y_t = f(h_{t-1}, x_t)$
- Output vector size is V+1 where V is the token vocabulary and '1' is END token.



Loss Functions

- The model predicts positions and confidences of sampled regions twice: in the localization layer and again in the recognition network.
- Binary logistic losses are used for the confidences trained on sampled positive and negative regions.

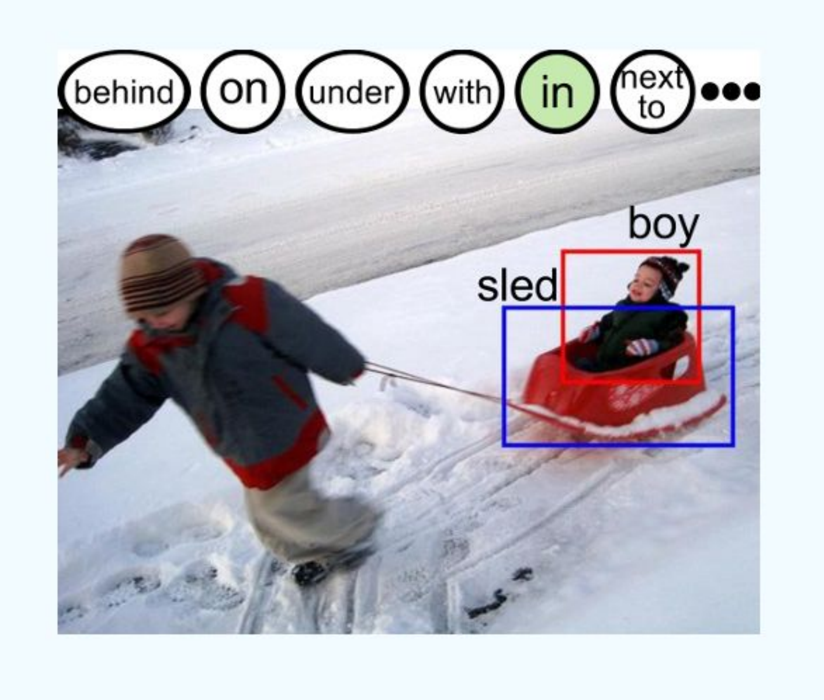
$$L\Big(y, f(\mathbf{x})\Big) \quad = \quad \log\Big(1 + \exp(-yf(\mathbf{x})\Big)$$

- For box regression, a smooth L1 loss is used. There is a term in the loss function-cross-entropy term at every time-step of the language model.
- Normalization of all loss functions by the batch size and sequence length in the RNN is carried out.

Combining Geometric, Textual and Image Features

- The output obtained from DenseCap contained overlapping bounding boxes per image with brief captions. We refined the boxes obtained to reduce unnecessary overlapping.
- The captions obtained per bounding box was then passed through Stanford Dependency Parser to obtain root as the landmark and trajectory.
- The landmark and trajectory were encoded using Word2Vec and the bounding boxes were used to obtain 11 features like percentage of overlap, Intersection over union and so on.
- This 611 sized vector was used to train the Multilogistic regression which then predicted the preposition between the landmark and trajectory.

Example



Encoder- Decoder Model

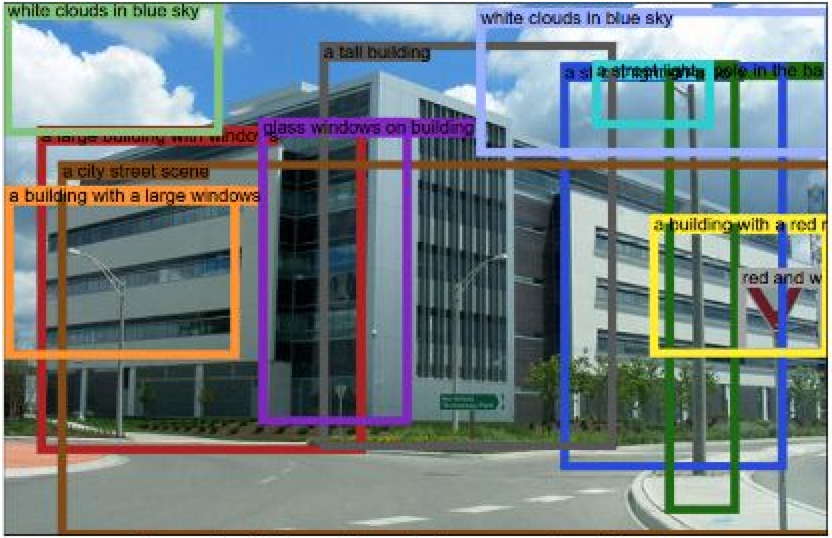
- The encoder-decoder model is used for language translation. We use the model and train it with phrases and sentences to make the model learn english grammar.
- The model consists of LSTM to make use of the sequence information present in language.
- The predicted prepositions with the incomplete captions are then fed into the trained model to generate meaningful sentences.
- The captions are hence combined to produce generative captions.
- We have made a dataset with key words and prepositions extracted using stanford parser on the Wiki dump(after cleaning) to train the model.

Encoder- Decoder Model

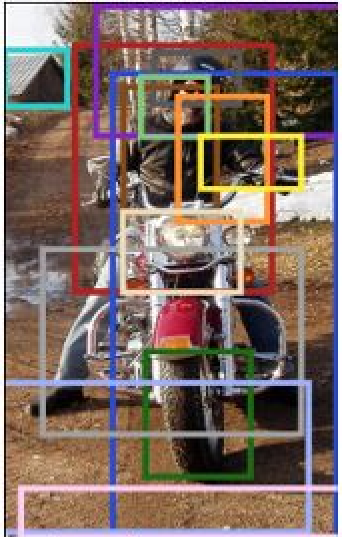
- A neural machine translation system is a neural network that directly models the conditional probability p(y|x) of translating a source sentence, x1, . . . , xn, to a target sentence, y1, . . . , ym.
- Basic form of NMT consists of two components:
  - an encoder which computes a representation for each source sentence
  - a decoder which generates one target word at a time and hence decomposes the conditional probability as:

$$\log p(y|x) = \sum_{j=1}^{m} \log p\left(y_j | y_{<j}, \boldsymbol{s}\right)$$

Results



a large building with windows. a street light on a pole. a tall building. a tall pole in the background. a city street scene. glass windows on building. red and white sign. white clouds in blue sky. white clouds in blue sky. a street light. a building with a large windows. a building with a red roof. a building with a large windows. a stop sign on a pole. a tree in the distance. a tall pole. a small green bush.

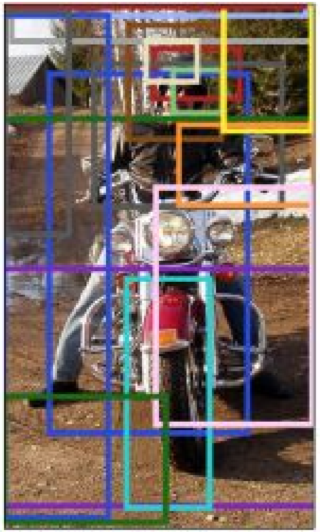


man on a motorcycle. a man on a motorcycle. black helmet on the man. front wheel of motorcycle. man wearing black jacket. trees behind the fence. a red motorcycle. man wearing sunglasses. dirt road. a small house in the background. man wearing black jacket. a bike rack. headlight on the front of the motorcycle. dirt on the ground. the bike is black. dirt on the ground. back wheel of motorcycle.

Results of pre-trained model on Visual genome dataset

Results

a building with a roof. a building with a roof. red and white sign. a building with a roof. a white building with a red and white sign. a building with a roof. a sign on the street. a tall street light. a tall building. a white building. a tall pole. a tall pole. the sky is cloudy. a window on the building. a tall pole. a tree in the background. a tall green tree. a white building.

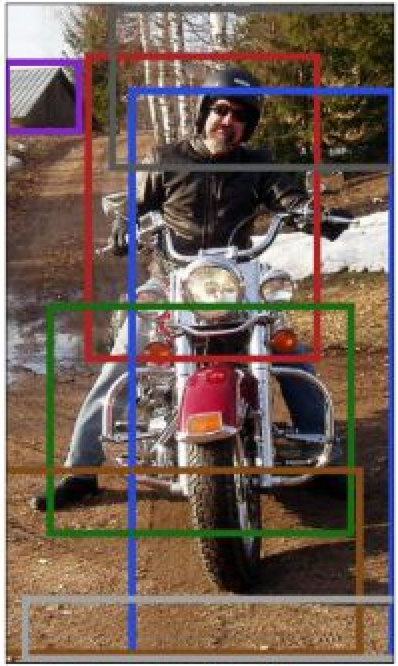


the helmet is black. a man riding a motorcycle. a man wearing a black helmet. trees behind the trees. a man wearing a helmet. the motorcycle is black. the tree is green. the man is wearing a hat. the tree is green. the front wheel of a motorcycle. a black and white bench. trees in the background. the helmet is black. a motorcycle parked on the road. the tree is green. a large brown dirt. a large rock. the ground is brown.

Results of self-trained model on Visual genome dataset

Results

a large building with windows. a street light on a pole. a tall building. a tall pole in the background. a city street scene. glass windows on building. white clouds in blue sky.



man on a motorcycle. a man on a motorcycle. trees behind the fence. a red motorcycle. dirt road. a small house in the background. dirt on the ground.

Results after reducing the overlapping of boxes

Results

a large building with windows	a city street scene	building	city	in
a large building with windows	glass windows on building	building	windows	in
a large building with windows	white clouds in blue sky	building	clouds	with
white clouds in blue sky	a tall pole in the background	clouds	pole	in
white clouds in blue sky	a red brick sidewalk	clouds	brick	in
a building in the background	a street light on a pole	building	street	in
man on a motorcycle	a man on a motorcycle	man	man	in
man on a motorcycle	a small wooden house	man	house	with
man on a motorcycle	a dirt road	man	road	in
man on a motorcycle	a concrete sidewalk	man	sidewalk	in
man on a motorcycle	dirt on the ground	man	dirt	in
a small house in the background	dirt on the ground	house	dirt	in
a tree with no leaves	green leaves on trees	tree	leaves	on
a small wooden house	roof of a building	house	roof	with
a dirt road	a red motorcycle	road	motorcycle	on
a dirt road	a tree in the background	road	tree	with
a red brick building	a red fence	brick	fence	with
a red brick building	power lines above the train	brick	lines	with
a red brick building	power lines above the train	brick	lines	with

Prediction of prepositions for every pair of captions for an image

Results

A Minneapolis large building with graffiti on the side and a city street scene .
A mustachioed large building with graffiti on the windows and glass windows in front of a building .
A mustachioed large building with graffiti on the windows and blue clouds in the background .
A businesswoman in front of a large building that has graffiti painted on it , and a building in the background , and a large building in the background .
A businesswoman in front of a large building that has graffiti written on the windows and a building in the background and a large building in the background .
A woodworker in a large building with graffiti on the ground in front of the building , and the one in the background is in the background .
A motorbiker is standing on a city street in front of a store that has graffiti on it and a tall building in the background .
White clouds , one in blue against the sky , and one in the foreground is holding a numeral tall pole and the one in the background .

Results from encoder-decoder model

Conclusion and Future Work

- The mean Average Precision value achieved for the self trained model of DenseCap is 3.48 as against 5 reported in the paper.
- The prepositions were predicted with an accuracy of 67% (70% reported in paper)
- The present encoder decoder has been modified from a Machine Translation Model to sentence generation from phrases. The dataset used for now is not apt as the phrases do not contain prepositions.
- We propose to use a better dataset with phrases and prepositions to generate sentences.

Papers Referred

- Johnson, Justin and Karpathy, Andrej and Fei-Fei, Li DenseCap: Fully Convolutional Localization Networks for Dense Captioning Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,2016
- Ramisa, Arnau and Wang, Josiah and Lu, Ying and Dellandrea, Emmanuel and Moreno-Noguer, Francesc and Gaizauskas, Robert. Combining Geometric, Textual and Visual Features for Predicting Prepositions in Image Descriptions.EMNLP 2015.
- Minh-Thang Luong Hieu Pham Christopher D. Manning.Effective Approaches to Attention-based Neural Machine Translation.EMNLP 2015.

Datasets Used

- Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik, Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models, IJCV, 2016
- Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations Ranjay Krishna, Yuke Zhu, Oliver Groth,Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li Jia-Li, David Ayman Shamma, Michael Bernstein, Li FeiFei.
- Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60

Thank You