Artionyms and Machine Learning: Auto naming of the paintings

Supervisors: Alexander Semenov, Dmitry Grigoriev, Valeria Kolycheva
Student: Anna Altynova

2021.02

CEBA lab, Saint-Petersburg State University

Table of contents

Task

Our goal is to find a model, which can generate a proper name for a given painting, such that generated name would represent contents of the painting and its idea.

Methods

To accomplish such a task and evaluate results we use:

- WikiArt dataset
- Neural network with Transformer architecture
- Neural network with Attention mechanism
- Neural network with simple encoding-decoding architecture
- BLEU metric

Dataset

The Wikiart paintings dataset[1]:

- more than 80,000 fine-art paintings
- more than 1,000 artists
- fifteen century to modern times
- 27 different styles
- 45 different genres

WikiArt

	Author	Name	Year	Style	Filename
count	78587	78587	46517.000000	78587	78587
unique	1105	64711	NaN	27	77298
top	vincent van gogh	self portrait	NaN	Impressionism	viktor-vasnetsov_crucified-christ-1896.jpg
freq	1889	493	NaN	12987	2
mean	NaN	NaN	1867.140508	NaN	NaN
std	NaN	NaN	122.398838	NaN	NaN
min	NaN	NaN	1029.000000	NaN	NaN
25%	NaN	NaN	1874.000000	NaN	NaN
50%	NaN	NaN	1902.000000	NaN	NaN
75%	NaN	NaN	1926.000000	NaN	NaN
max	NaN	NaN	2012.000000	NaN	NaN

Models

We compare performances of three different types of neural networks based on encoder-decoder architecture, which are highly usable in image captioning:

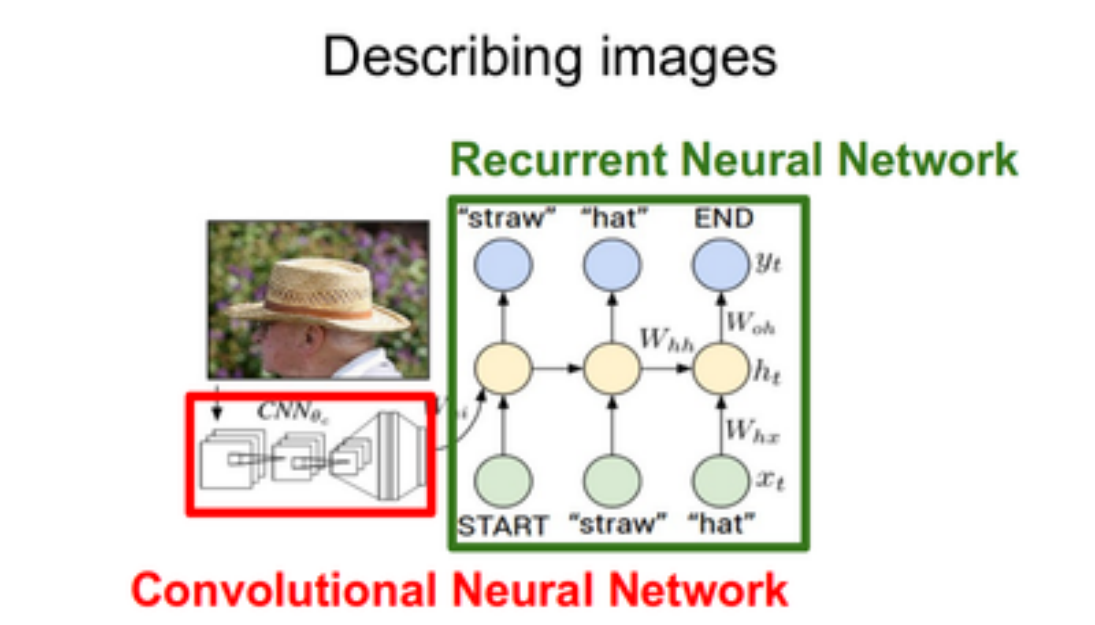
- Neural Image Caption(NIC) network with simple encoding-decoding architecture
- network with Attention mechanism
- state-of-the-art network with Transformer architecture

NIC

This architecture was proposed in a paper titled “Show and Tell: A Neural Image Caption Generator” by Google in 2015[2].
NIC contains:

1. vision CNN that extracts the features and nuances out of a given image
2. language generating RNN that translates the features and objects given by our image based model to a natural sentence

NIC


Describing images
Recurrent Neural Network
"straw"
"hat"
END
y_t
W_{oh}
W_{hh}
h_t
W_{hx}
x_t
START
"straw"
"hat"
CNN_{θ_c}
Convolutional Neural Network

NIC

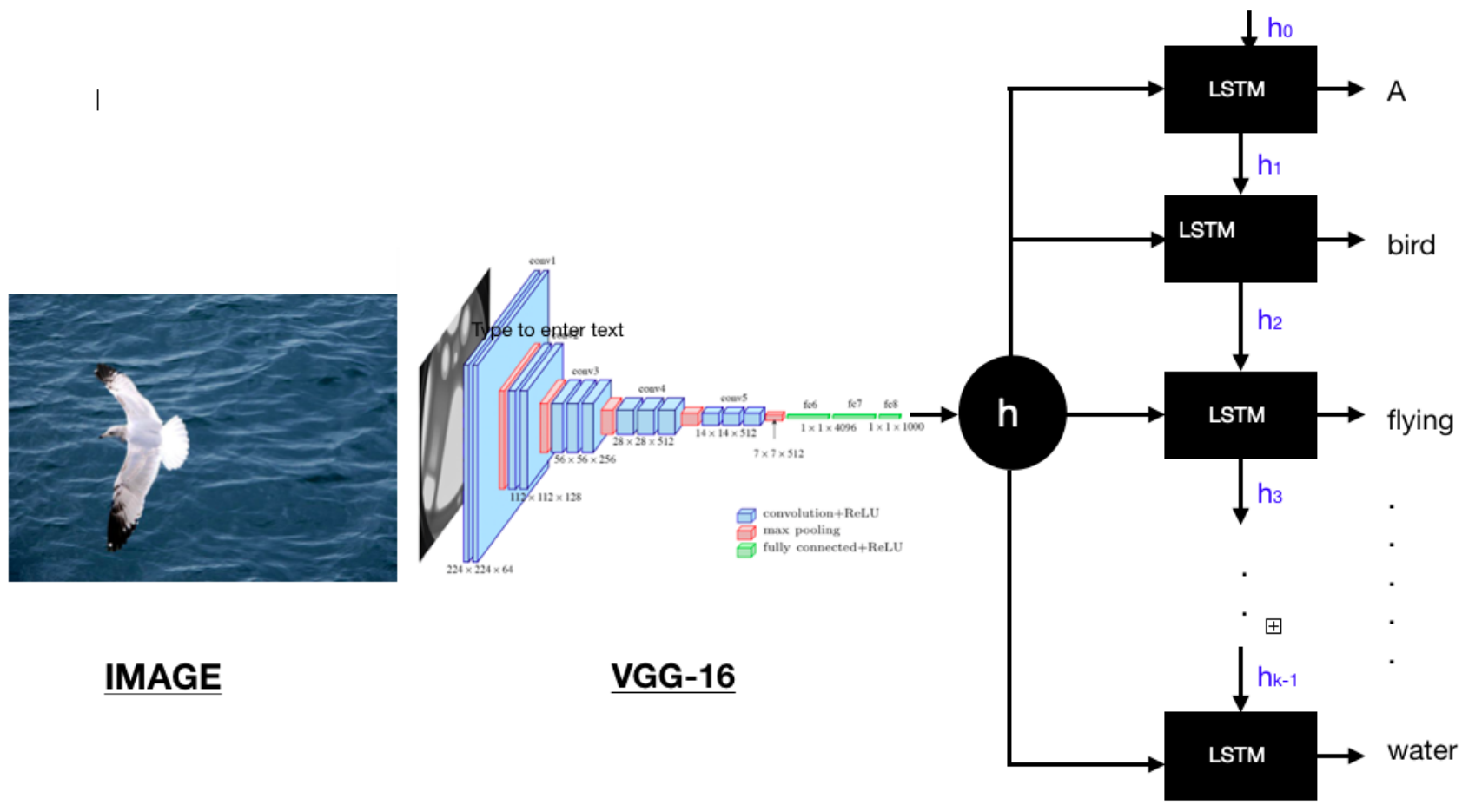
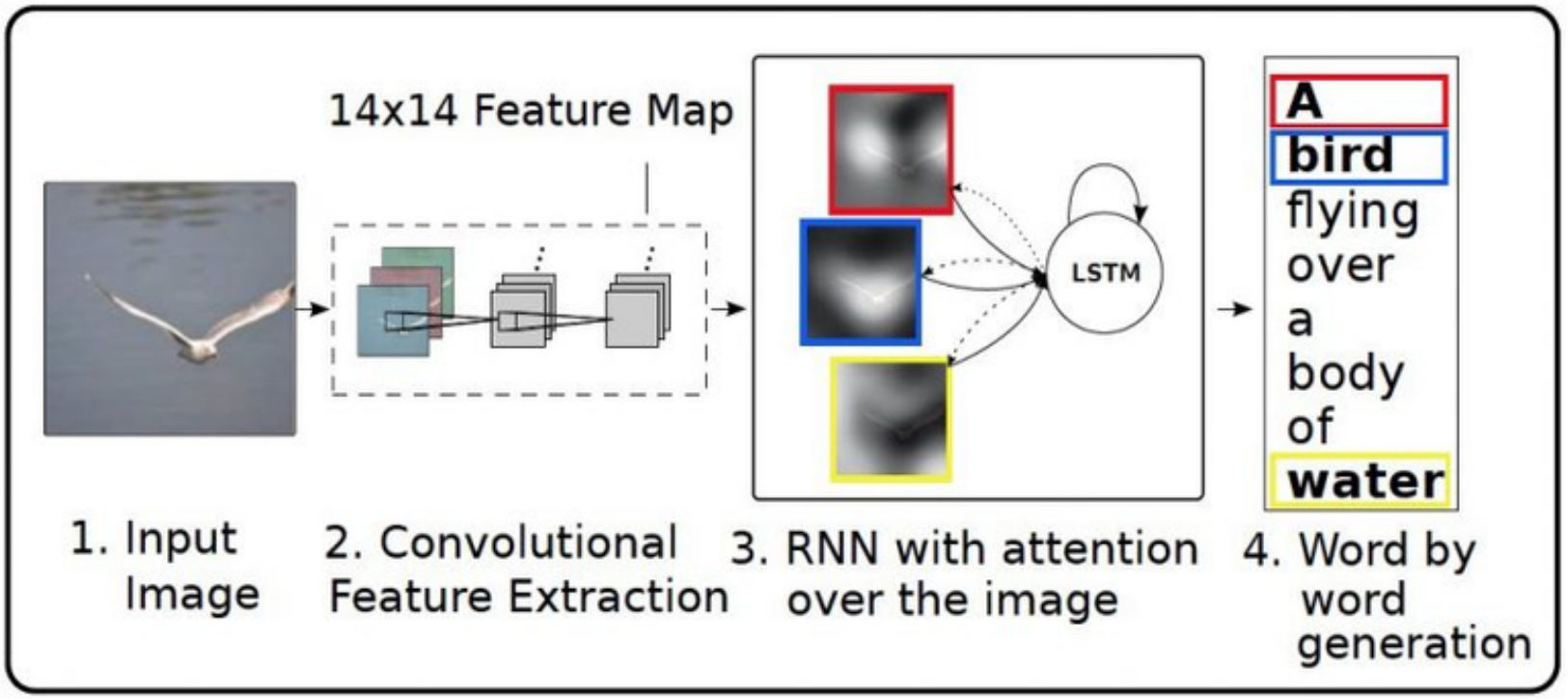


Figure 1: classic NIC

Attention

This architecture was proposed in a paper titled “Show, Attend and Tell: Neural Image Caption Generation with Visual Attention” in 2016[3].
What’s different from NIC:
The image is first divided into n parts, and we compute an image representation of each part. When the RNN is generating a new word, the attention mechanism is focusing on the relevant part of the image, so the decoder only uses specific parts of the image.

Attention


14x14 Feature Map
LSTM
A
bird
flying
over
a
body
of
water
1. Input Image
2. Convolutional Feature Extraction
3. RNN with attention over the image
4. Word by word generation

Attention

So model with attention has three main parts:

- a CNN that extracts features from images
- an attention mechanism that weights the image features
- RNN that generates captions to describe the weighted image features

Attention

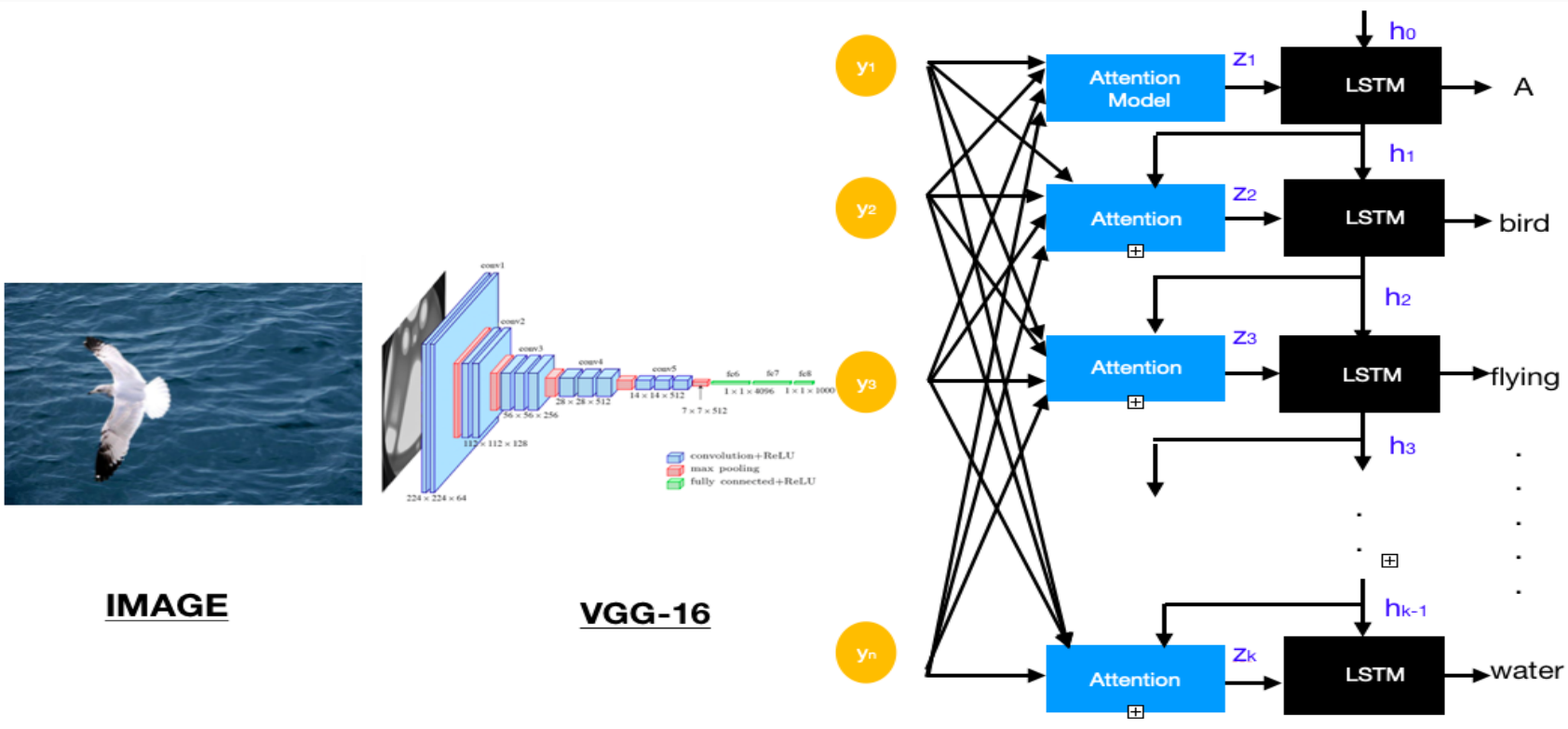


IMAGE
VGG-16
convolution+ReLU
max pooling
fully connected+ReLU
y1
y2
y3
yn
Attention Model
Attention
Attention
Attention
z1
z2
z3
zk
h0
h1
h2
h3
hk-1
LSTM
LSTM
LSTM
LSTM
A
bird
flying
water

Transformers

Discussed models perform well, however, dealing with RNN in decoder has two limitations:

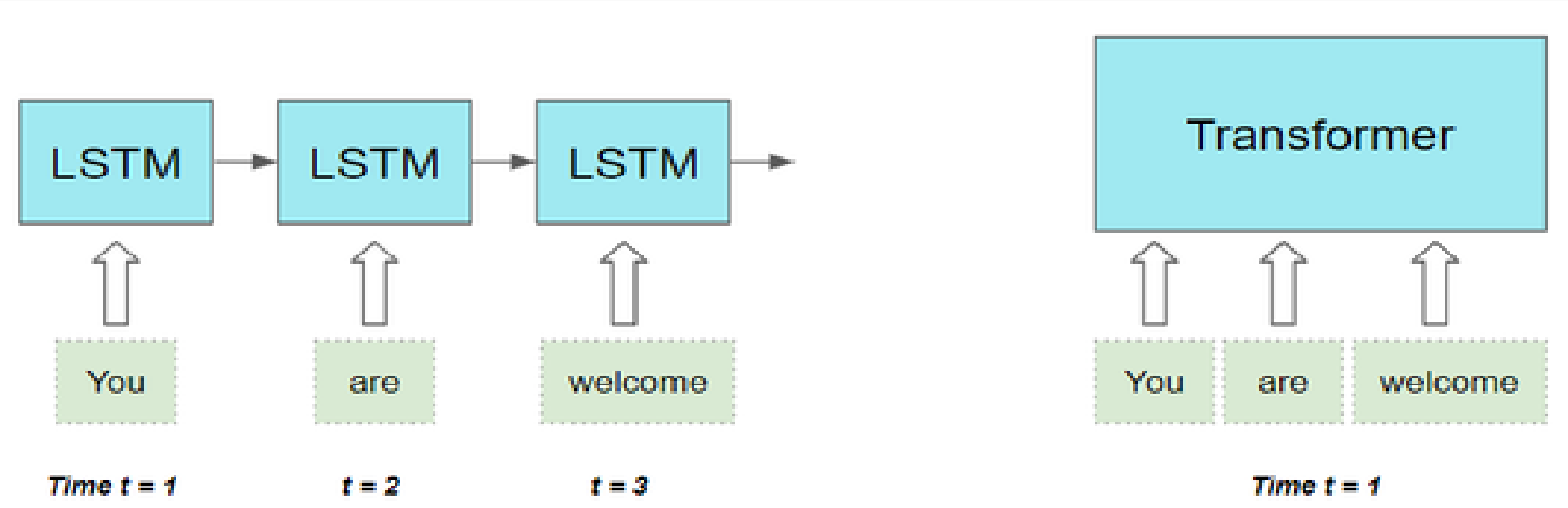
1. It is challenging to deal with long-range dependencies between words that are spread far apart in a long sentence.
2. They process the input sequence sequentially one word at a time, which means that it cannot do the computation for time-step t until it has completed the computation for time-step t - 1. This slows down training and inference.

Transformers

The Transformer architecture addresses both of these limitations.It was first introduced in the paper "Attention is all you need" in 2017[4] and was quickly established as the leading architecture for most text data applications. It got rid of RNNs altogether and relied exclusively on the benefits of Attention.

Transformers

Transformers process all the words in the sequence in parallel, thus greatly speeding up computation.

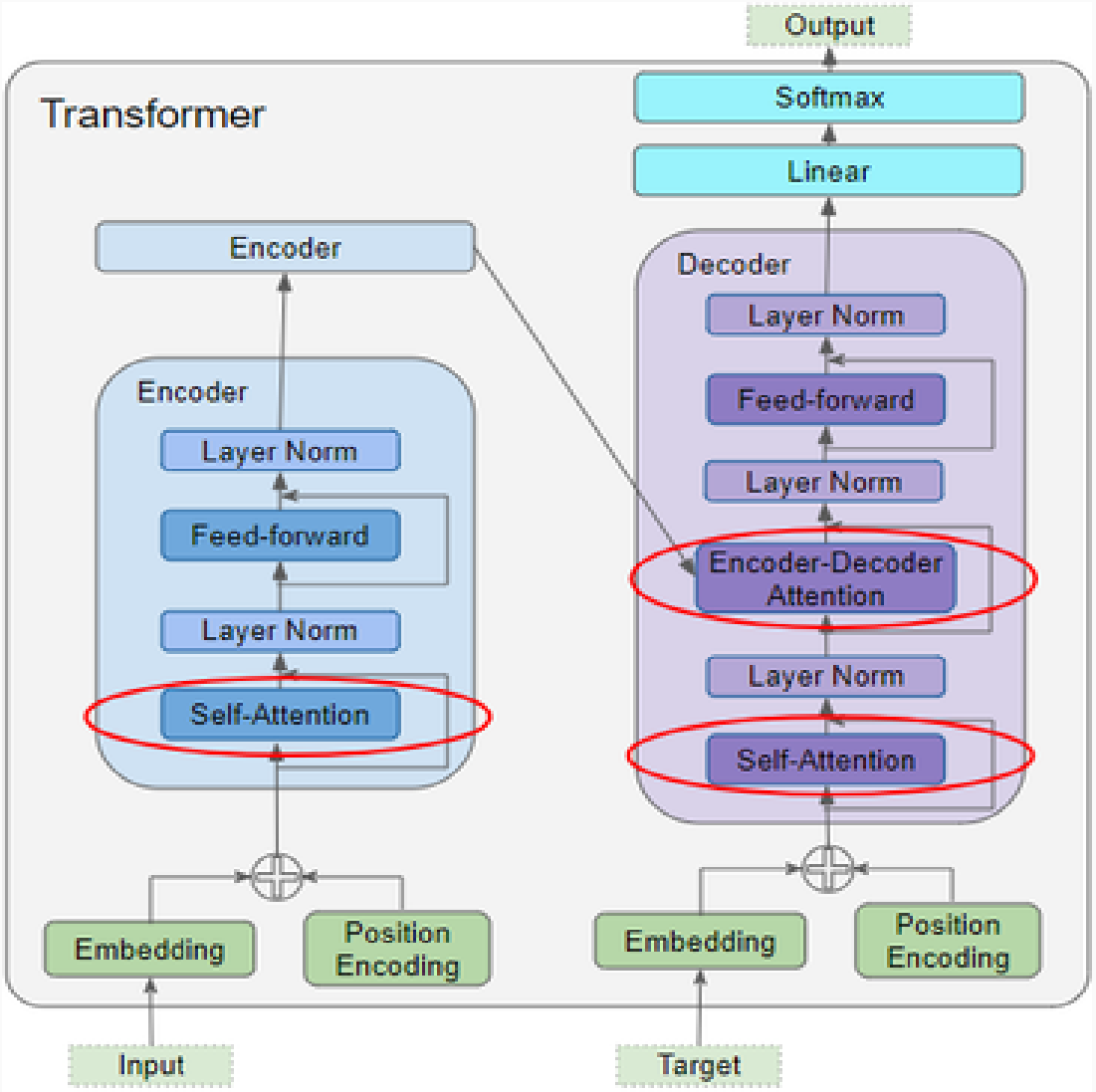


Transformers

Attention is used in the Transformer in three places:

1. Self-attention in the Encoder — the input sequence pays attention to itself
2. Self-attention in the Decoder — the target sequence pays attention to itself
3. Encoder-Decoder-attention in the Decoder — the target sequence pays attention to the input sequence

Transformers



Metric

The Bilingual Evaluation Understudy Score(2002, IBM)[5], or BLEU, is a metric for evaluating a generated sentence to a reference sentence. BLEU was one of the first metrics to claim a high correlation with human judgements of quality.

Metric

The approach works by counting matching n-grams in the candidate sentence to n-grams in the reference text, where 1-gram or unigram would be each token and a bigram comparison would be each word pair. It is common to report the BLEU-1 to BLEU-4 scores when describing the skill of a text generation system.

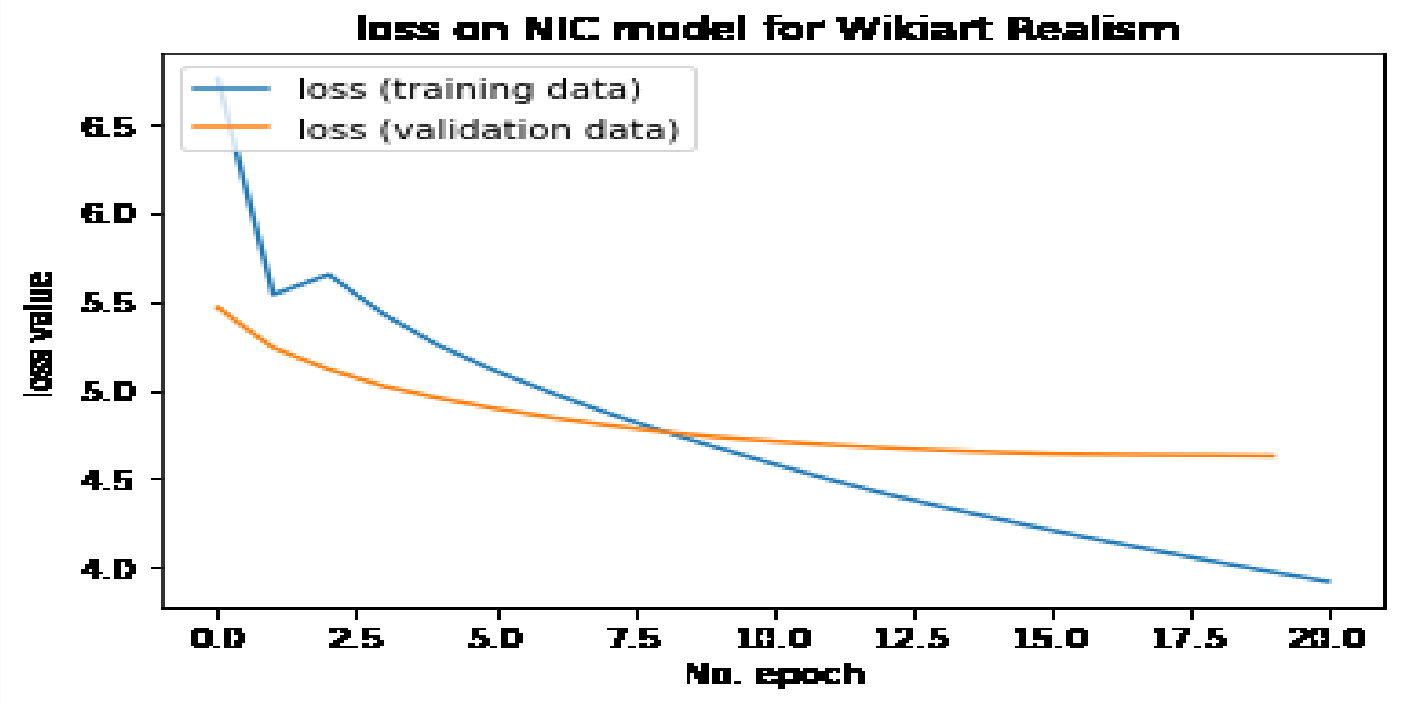
What have we done

Up to this point, we've been training our models on a subset of wikiart dataset, which includes only paintings made in Realism style(since it's much faster). This subset contains 10634 images, that is usually enough for the image captioning task.

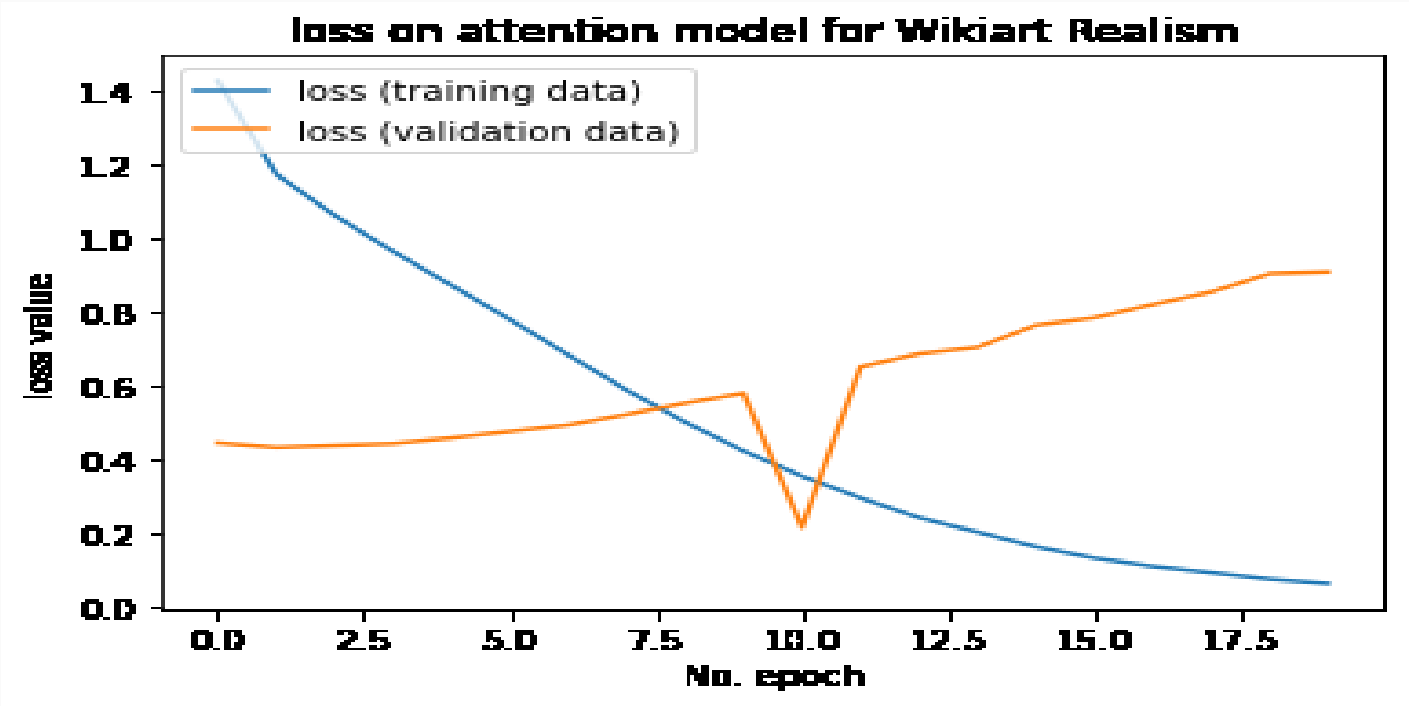
Training process

Our training, validation, ant test sets include around 6000, 2000 and 2600 images respectively. We use categorical crossentropy as loss function and Adam optimizer.

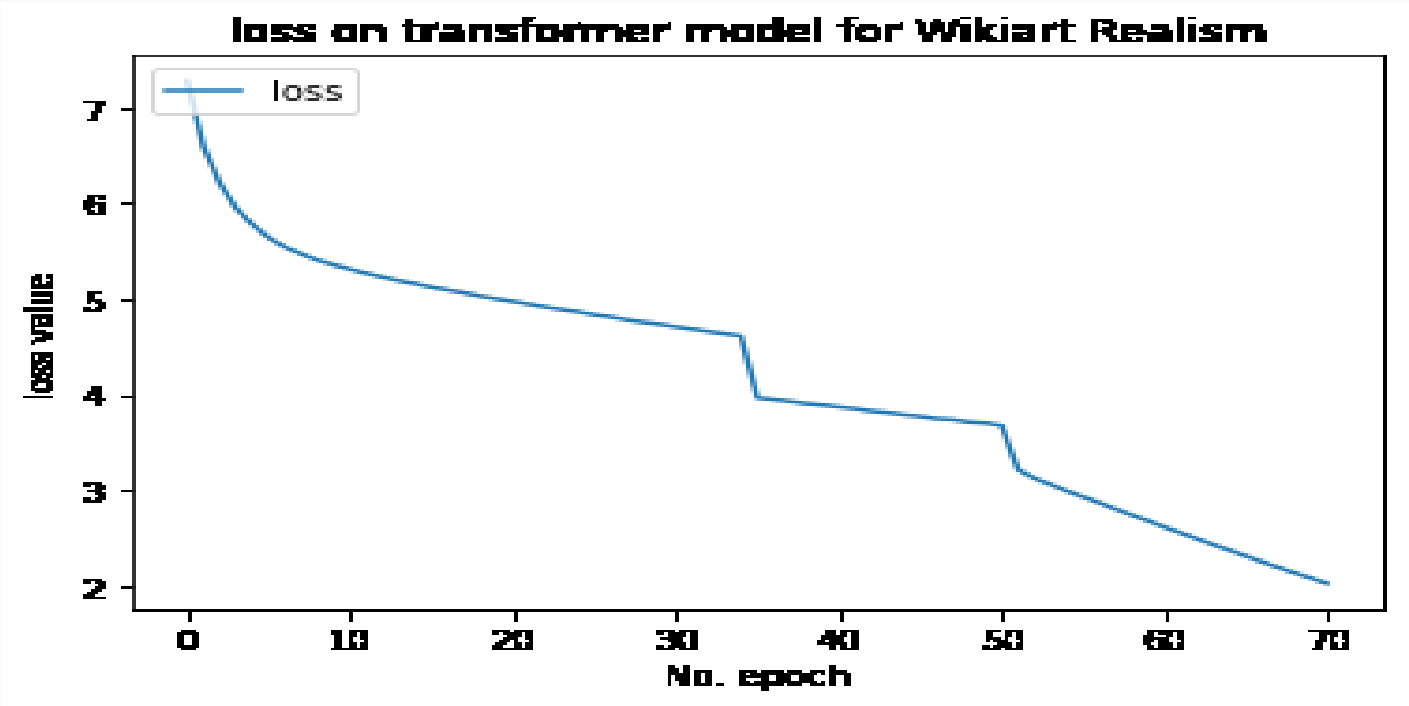
Graphs


loss on NIC model for Wikiart Realism
loss (training data)
loss (validation data)
loss value
6.5
6.0
5.5
5.0
4.5
4.0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
No. epoch

Graphs


loss on attention model for Wikiart Realism
loss (training data)
loss (validation data)
loss value
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
No. epoch

Graphs


loss on transformer model for Wikiart Realism
loss
loss value
7
6
5
4
3
2
0
10
20
30
40
50
60
70
No. epoch

Results

Model: Encoder + Decoder	Bleu-1	Bleu-2	Bleu-3	Bleu-4
NIC: VGG16 + LSTM	0.106	0.143	0.175	0.196
NIC + Bahdanau Attention: InceptionResNetV2 + GRU	0.09	0.126	0.162	0.185
Multihead Transformer	0.067	0.11	0.143	0.164

Table 1: Comparisons

Examples

Painting	Result
	Original: peasant woman digging Prediction1: a woman and a basket Bleu-1: 0.66 Prediction2: girl carrying on a basket digging Bleu-1 : 0.63 Prediction3: peasant woman with a cap Bleu-1: 0.67
	Original: portrait of dr edward anthony spitzka Prediction1: head of the artist Bleu-1: 0.42 Prediction2: priest of grain Bleu-1: 0.27 Prediction3: the portrait of v p amosov Bleu-1: 0.16

Examples

Painting	Result
	Original:the beach at trouville at low tide Prediction1: landscape in the river Bleu-1: 0.27 Prediction2: road in the chestnut figures Bleu-1: 0.44 Prediction3: old man on a basket of the river Bleu-1: 0.32
	Original: an arcadian Prediction1: autumn Bleu-1: 0 Prediction2: fontainebleau Bleu-1: 0 Prediction3: in auvergne life with a basket of puy Bleu-1: 0

Examples

Painting	Result
	Original: pharmacist Prediction1: woman Bleu-1: 0 Prediction2: a man carrying peat and handkerchief Bleu-1: 0 Prediction3: landscape Bleu-1: 0
	Original: bridge astrakhan Prediction1: view at the danube Bleu-1: 0 Prediction2: view of the boat Bleu-1: 0 Prediction3: head of a woman Bleu-1: 0

Boundaries

1. As it turns out, Bleu
score does not correlate well with quality
of generated captions. For instance,
generated name "Village soldier"
for Vincent van Gogh's "Orphans"
seems fine, but its bleu score is zero.

Figure 2: "Orphans"

Boundaries

2. Actual names in training set can cause an unexpected effect in some cases. Many of them contain proper names, e.g. cities, first and second names. This leads to generation of caption "Pskov church" for Vasily Vereshchagin's "The three main gods in a chingacheling buddhist monastery in sikkim".

Figure 3: "The three main gods in a chingacheling buddhist monastery in sikkim"

Next Steps

1. Clear data from proper names or substitute them for something neutral
2. Train models on the whole dataset
3. Consider another metrics or do expert check on generated captions

Literature

1. WikiArt
   https://www.wikiart.org/
2. Show and Tell: A Neural Image Caption Generator
   https://arxiv.org/pdf/1411.4555.pdf
3. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention
   https://arxiv.org/abs/1502.03044
4. Attention is all you need
   https://arxiv.org/abs/1706.03762
5. BLEU
   https://www.aclweb.org/anthology/P02-1040.pdf