Computer Vision: Fall 2022 — Lecture 17

Dr. Karthik Mohan

Univ. of Washington, Seattle

December 1, 2022

References

Generic ML/DL

1. Good Book for Machine Learning Concepts
2. Deep Learning Reference

CNN

1. Convolutional Neural Networks for Visual Recognition
2. Convolutional Neural Net Tutorial
3. CNN Transfer Learning
4. PyTorch Transfer Learning Tutorial

CNN Publication References

CNN surveys

1. Convolutional Neural Networks: A comprehensive survey, 2019
2. A survey of Convolutional Neural Networks: Analysis, Applications, and Prospects, 2021

CNN Archs

1. GoogLeNet
2. Top models on ImageNet
3. ResNet ILSVRC paper

Object Detection and Image Captioning References

Object Detection

1. A survey of modern deep learning based object detection methods
2. YOLO Survey
3. YOLO Original Paper

Image Captioning

1. From Show to Tell: A survey on Deep Learning-based Image Captioning
2. A survey of image captioning models
3. StyleNet

Today

1. Image Captioning Models Recap
2. StyleNet - Image Captioning with Style
3. Final Mini-Project

Next Topic: Image Captioning Models

COCO Data Set

MS-COCO

Woman on a horse jumping over a pole jump.

A glass bowl contains peeled tangerines and cut strawberries.

COCO Data Set

COCO

Woman on a horse jumping over a pole jump.

A glass bowl contains peeled tangerines and cut strawberries.

COCO

mini-Project

word cloud

Why Image Captioning?

1. Virtual Assistants

Why Image Captioning?

1. Virtual Assistants
2. Visually impaired assistance

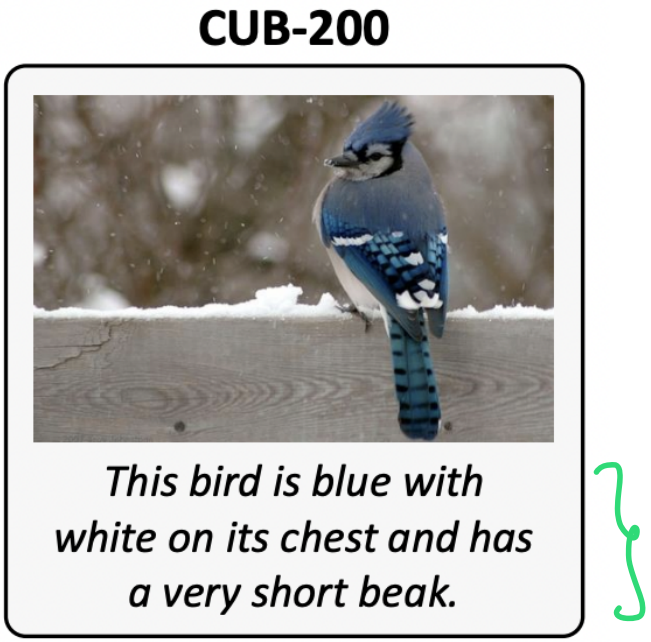
Why Image Captioning?

1. Virtual Assistants
2. Visually impaired assistance
3. Robotics

Why Image Captioning?

1. Virtual Assistants
2. Visually impaired assistance
3. Robotics
4. Any other use case?

CUB-200 Data Set

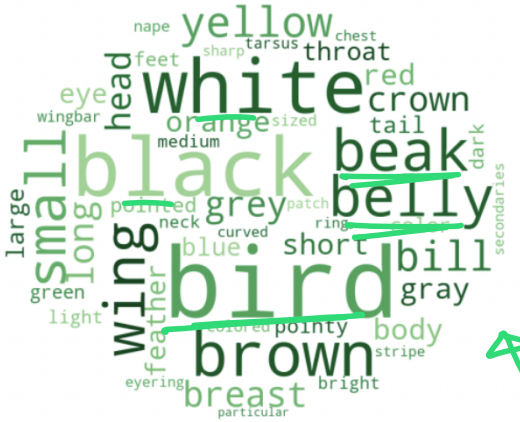


CUB-200 Data Set

CUB-200

This bird is blue with white on its chest and has a very short beak.

CUB-200

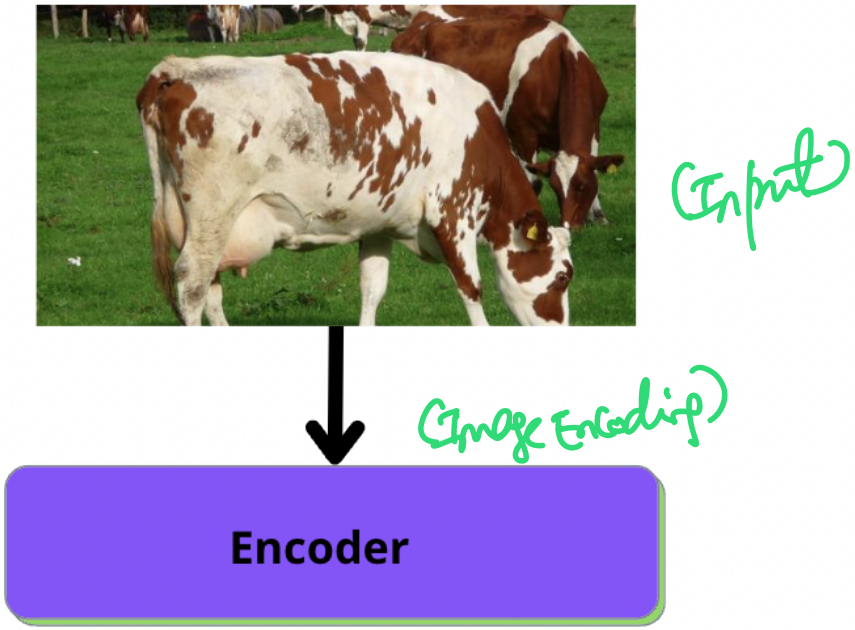


Word cloud

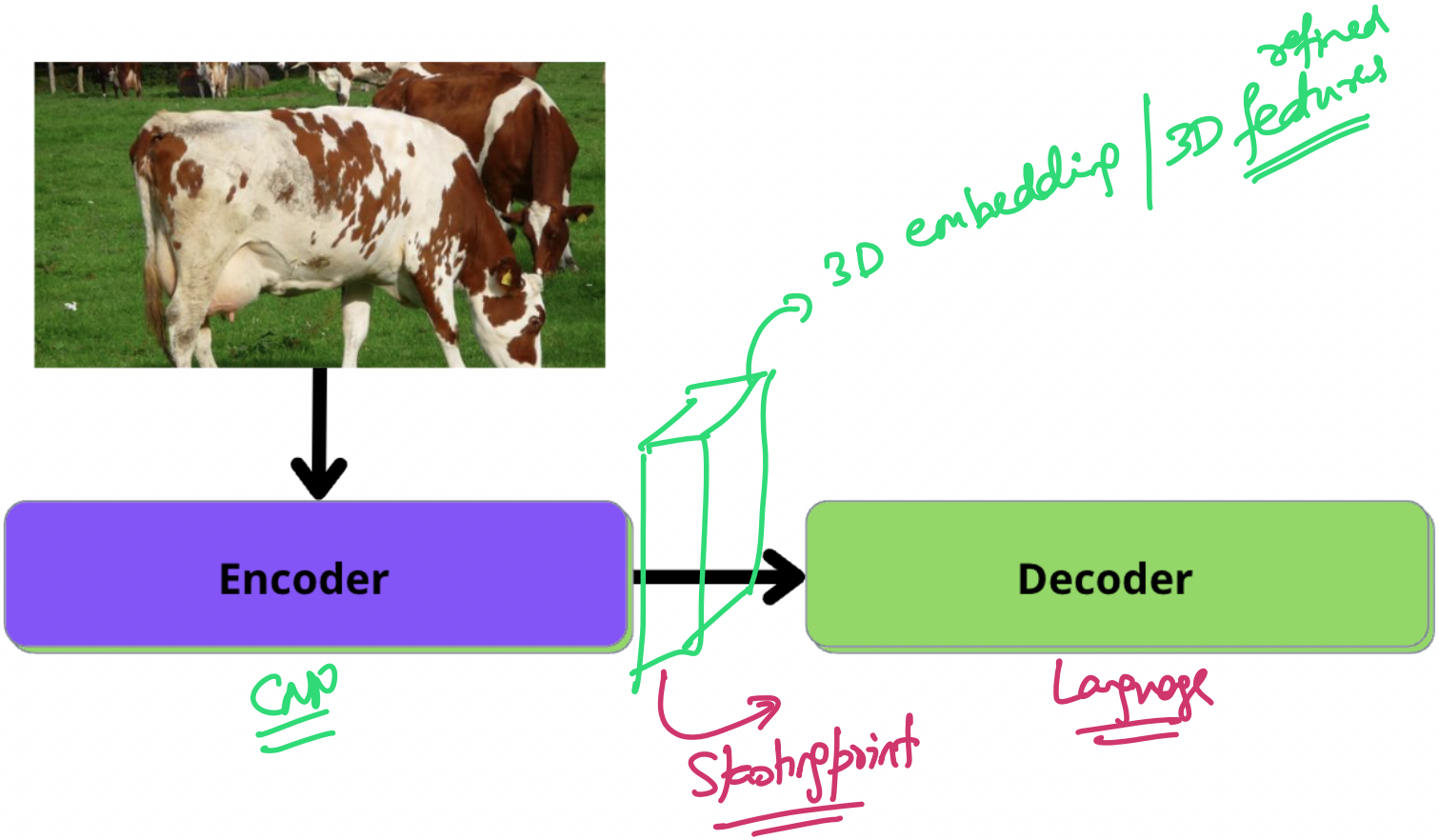
Encoder-Decoder Model for Image Captioning

Encoder-Decoder Model for Image Captioning

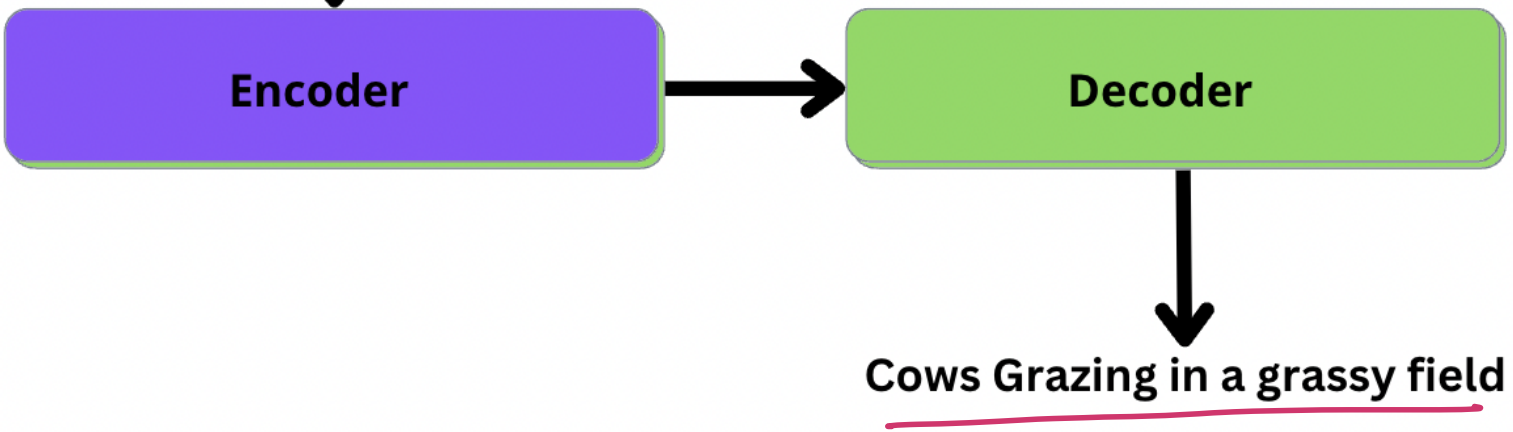
Encoder-Decoder Model for Image Captioning



Encoder-Decoder Model for Image Captioning



Encoder-Decoder Model for Image Captioning



Encoder-Decoder Model for Image Captioning

Image Encoder (Typically CNN)

- VGG
- InceptionV3
- ResNet
- ResNeXt

Language Decoder (Typically RNN/LSTM)

(Transformers/BERT models)

Cows Grazing in a grassy field

Language Model (LM)

(Masked LM)

1. Cows Grazing on a field

Input

next word pred.

2. Masked LM: Cows Grazing in a [MASK] field

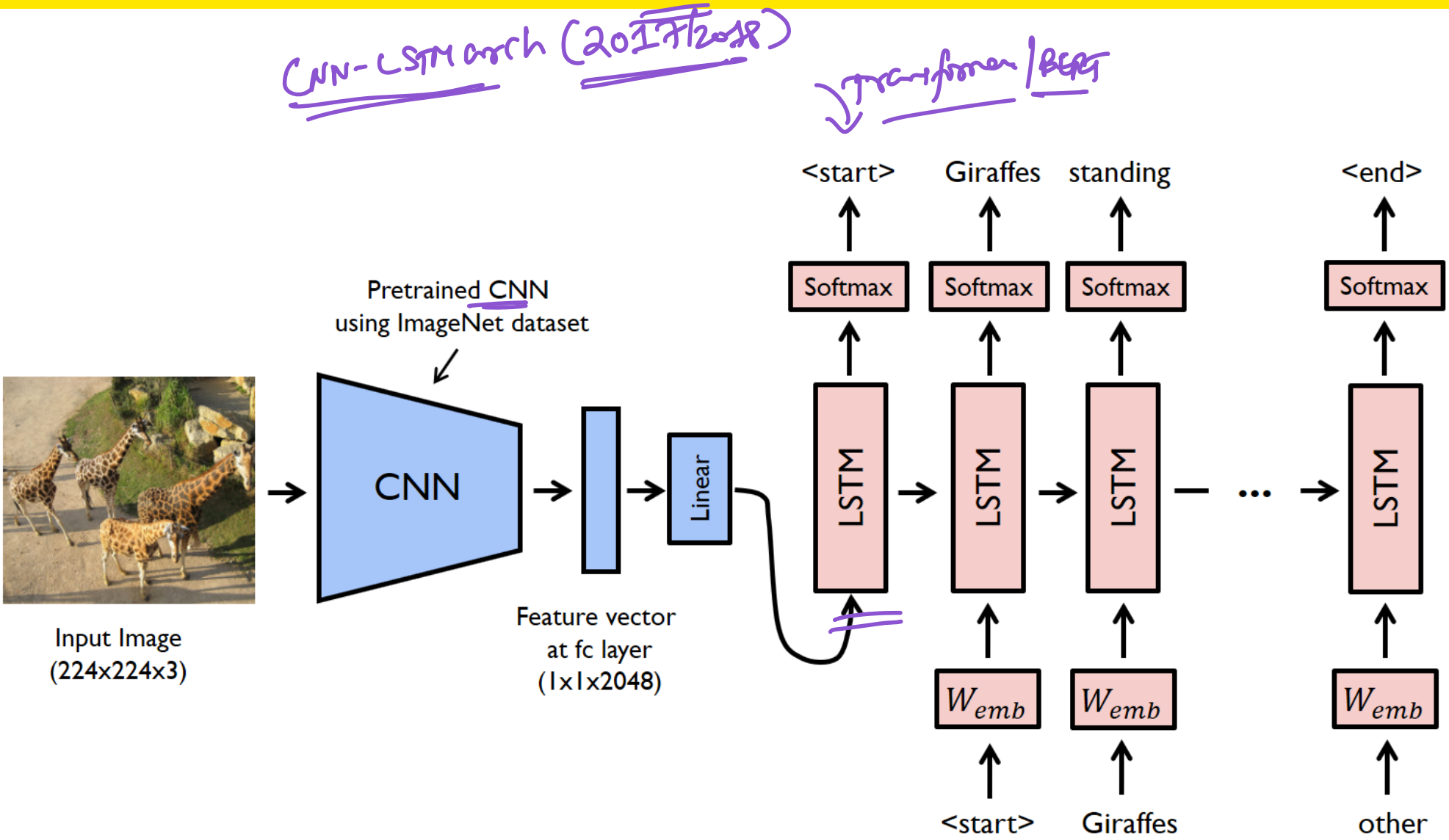
Encoder-Decoder Model for Image Captioning

Image Encoder (Typically CNN Pre-trained + Fine-tuned)

Language Decoder (Typically RNN/LSTM)

Cows Grazing in a grassy field

Encoder-Decoder Architecture

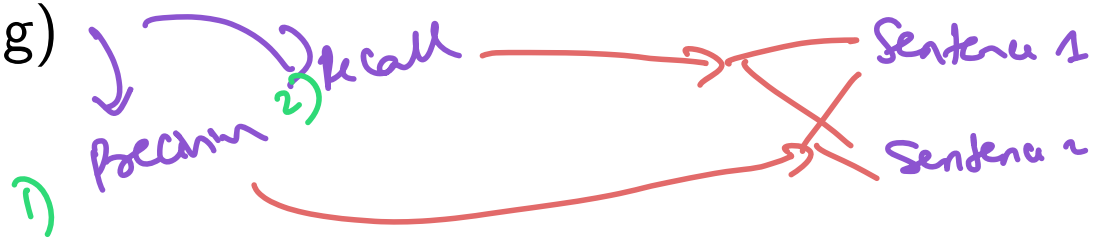


Metrics for Image Captioning

1. BLEU (Bilingual evaluation understudy)

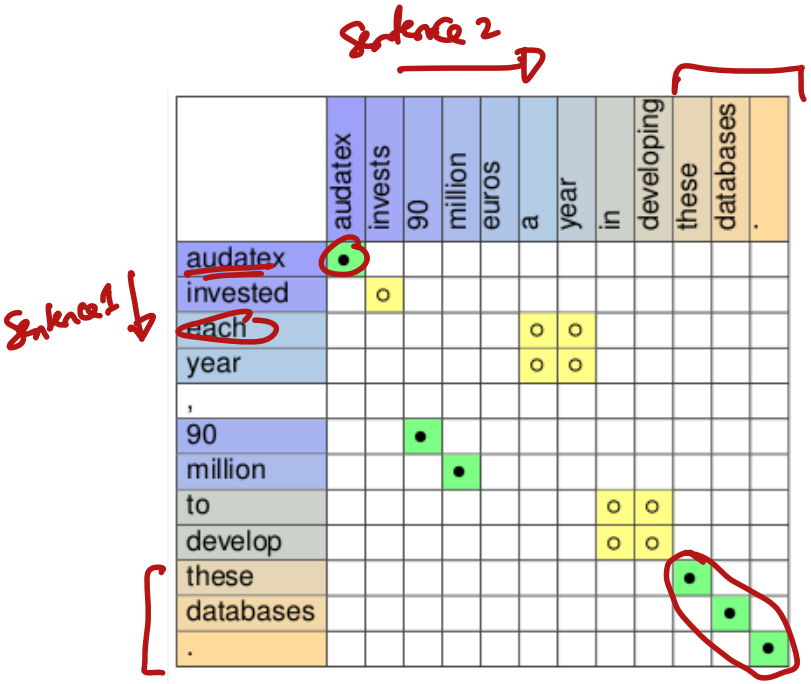
Metrics for Image Captioning

1. BLEU (Bilingual evaluation understudy)
2. **METEOR** (Metric for Evaluation of Translation with Explicit Ordering)



3) F-Score
4) Ordering:-
5) Penalty = $0.5 \times (\text{frag})^3$
6) METEOR = F-Score $\times (1 - p)$

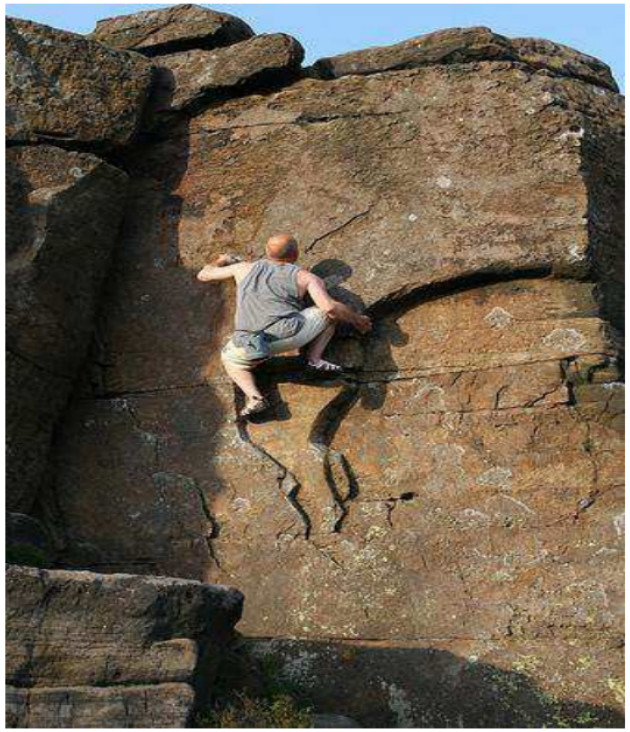
METEOR Metric



Segment 2437

P:	0.807
R:	0.764
Frag:	0.490
Score:	0.393

StyleNet - Image Captioning in Style!



CaptionBot: A man on a rocky hillside next to a stone wall.

Romantic: A man uses rock climbing to conquer the high.

Humorous: A man is climbing the rock like a lizard.

CaptionBot: A dog runs in the grass.

Romantic: A dog runs through the grass to meet his lover.

Humorous: A dog runs through the grass in search of the missing bones.

Why Style Matters?

1. Greater engagement with chat bots

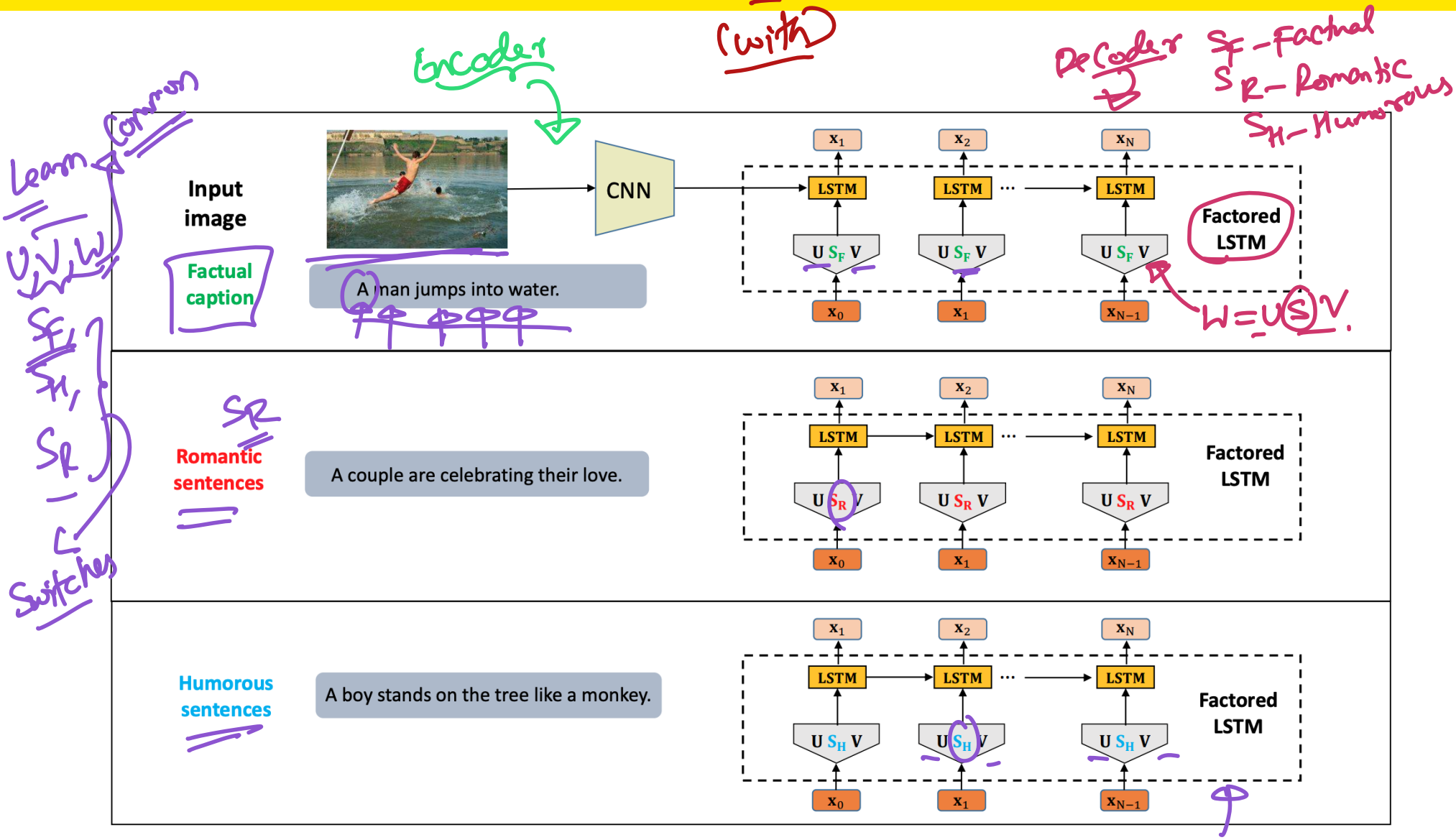
Why Style Matters?

1. Greater engagement with chat bots (Siri)
2. Better captioning for social media!

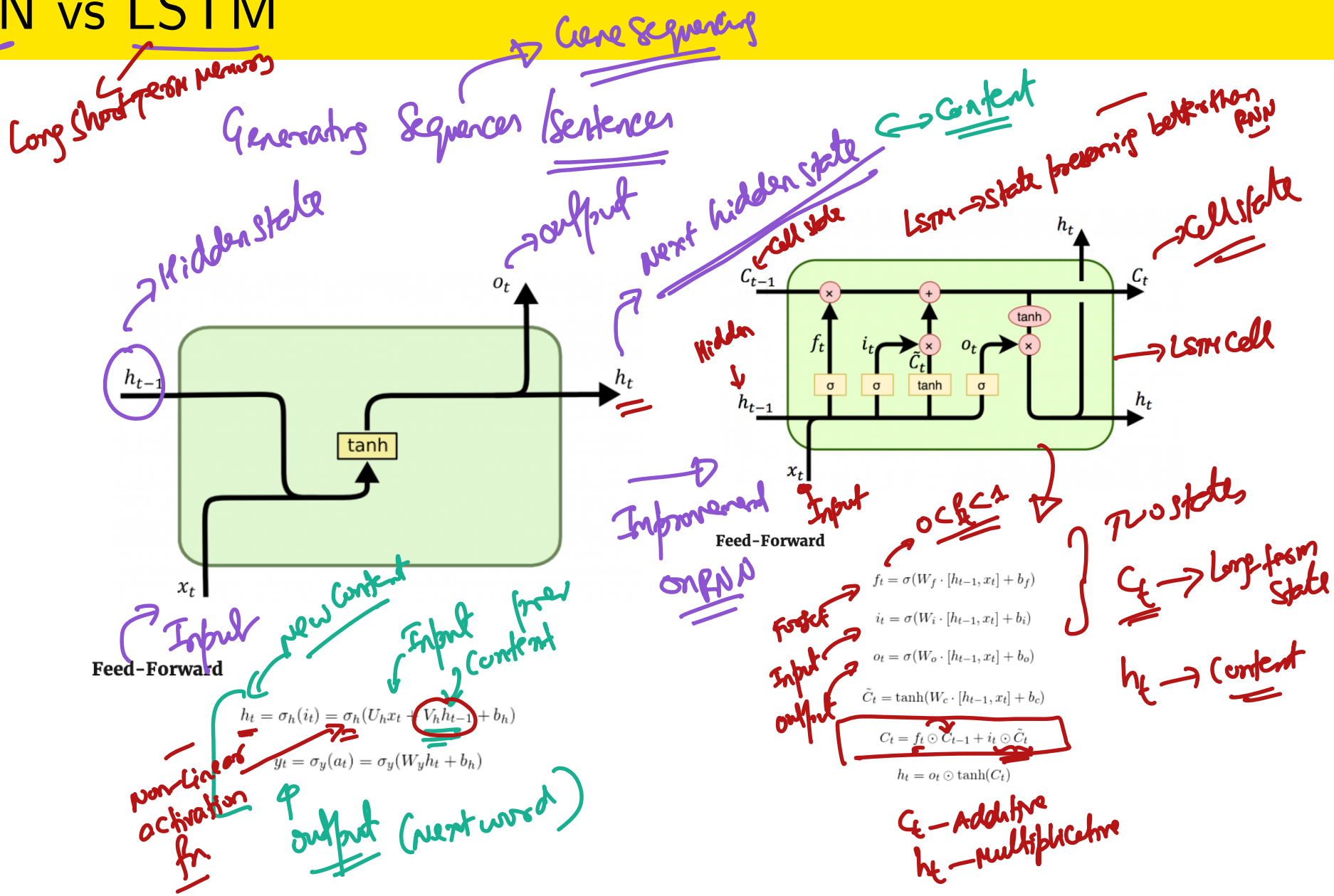
Why Style Matters?

1. Greater engagement with chat bots
2. Better captioning for social media!
3. Personalizing captions!

StyleNet - Image Captioning in Style!



RNN vs LSTM


Long Short Term Memory
Generating Sequences/Sentences
Scene Sequencing
Hidden State
Output
Next hidden state
Context
LSTM → State preserving better than RNN
Cell state
Cell State
Hidden
LSTM Cell
Improvement on RNN
Input
Feed-Forward
New Context
Input
Context from
Non-linear activation fn
Output (next word)
$h_t = \sigma_h(i_t) = \sigma_h(U_h x_t + V_h h_{t-1} + b_h)$
$y_t = \sigma_y(a_t) = \sigma_y(W_y h_t + b_h)$
$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$
$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$
$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$
$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$
$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$
$h_t = o_t \odot \tanh(C_t)$
$0 < f_t < 1$
Forget
Input
Output
Two states
C_t → Long-term State
h_t → Content
C_t – Additive
h_t – multiplicative

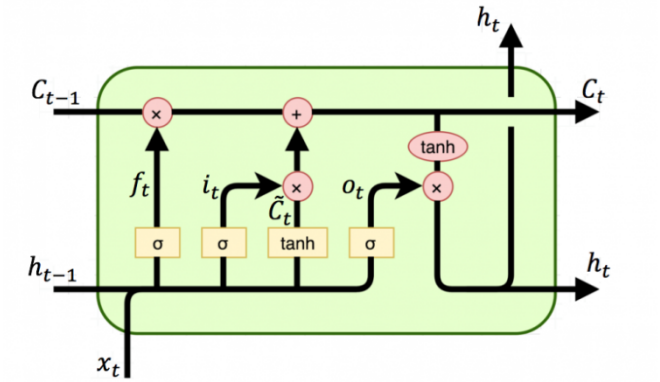
ICE #1

Popular DL interview question

What issues in RNN does LSTM fix? (Pick all that apply!)

1. Exploding Gradients ✓
2. Vanishing Gradients ✓
3. Making the output more sequential and recursive
4. Linear Model

LSTM module



Feed-Forward

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$$

$$h_t = o_t \odot \tanh(C_t)$$

$0.01 \times 0.0001 = 10^{-6}$

$0.01 + 0.0001 = 0.01 = 10^{-2}$

$$\boldsymbol{i}_t = \text{sigmoid}(\boldsymbol{W}_{ix}\boldsymbol{x}_t + \boldsymbol{W}_{ih}\boldsymbol{h}_{t-1}) \quad (1)$$

$$\boldsymbol{f}_t = \text{sigmoid}(\boldsymbol{W}_{fx}\boldsymbol{x}_t + \boldsymbol{W}_{fh}\boldsymbol{h}_{t-1}) \quad (2)$$

$$\boldsymbol{o}_t = \text{sigmoid}(\boldsymbol{W}_{ox}\boldsymbol{x}_t + \boldsymbol{W}_{oh}\boldsymbol{h}_{t-1}) \quad (3)$$

$$\tilde{\boldsymbol{c}}_t = \tanh(\boldsymbol{W}_{cx}\boldsymbol{x}_t + \boldsymbol{W}_{ch}\boldsymbol{h}_{t-1}) \quad (4)$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \tilde{\boldsymbol{c}}_t \quad (5)$$

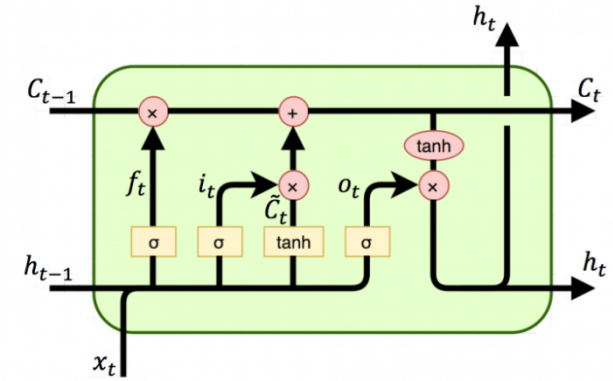
$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot \boldsymbol{c}_t \quad (6)$$

$$\boldsymbol{p}_{t+1} = \text{Softmax}(\boldsymbol{C}\boldsymbol{h}_t) \quad (7)$$

Additive Cell State

1) Preserves content over longer sequences
2) Doesn't have issues with expl/vanishing grads

Regular LSTM vs StyleNet LSTM



Feed-Forward

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$$

$$h_t = o_t \odot \tanh(C_t)$$

S matrix capture styles

Wix

StyleNet

Regular

$$\boldsymbol{i}_t = \text{sigmoid}(\boldsymbol{W}_{ix}\boldsymbol{x}_t + \boldsymbol{W}_{ih}\boldsymbol{h}_{t-1}) \quad (1)$$

$$\boldsymbol{f}_t = \text{sigmoid}(\boldsymbol{W}_{fx}\boldsymbol{x}_t + \boldsymbol{W}_{fh}\boldsymbol{h}_{t-1}) \quad (2)$$

$$\boldsymbol{o}_t = \text{sigmoid}(\boldsymbol{W}_{ox}\boldsymbol{x}_t + \boldsymbol{W}_{oh}\boldsymbol{h}_{t-1}) \quad (3)$$

$$\tilde{\boldsymbol{c}}_t = \tanh(\boldsymbol{W}_{cx}\boldsymbol{x}_t + \boldsymbol{W}_{ch}\boldsymbol{h}_{t-1}) \quad (4)$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \tilde{\boldsymbol{c}}_t \quad (5)$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot \boldsymbol{c}_t \quad (6)$$

$$\boldsymbol{p}_{t+1} = \text{Softmax}(\boldsymbol{C}\boldsymbol{h}_t) \quad (7)$$

$$\boldsymbol{i}_t = \text{sigmoid}(\boldsymbol{U}_{ix}\boldsymbol{S}_{ix}\boldsymbol{V}_{ix}\boldsymbol{x}_t + \boldsymbol{W}_{ih}\boldsymbol{h}_{t-1}) \quad (9)$$

$$\boldsymbol{f}_t = \text{sigmoid}(\boldsymbol{U}_{fx}\boldsymbol{S}_{fx}\boldsymbol{V}_{fx}\boldsymbol{x}_t + \boldsymbol{W}_{fh}\boldsymbol{h}_{t-1}) \quad (10)$$

$$\boldsymbol{o}_t = \text{sigmoid}(\boldsymbol{U}_{ox}\boldsymbol{S}_{ox}\boldsymbol{V}_{ox}\boldsymbol{x}_t + \boldsymbol{W}_{oh}\boldsymbol{h}_{t-1}) \quad (11)$$

$$\tilde{\boldsymbol{c}}_t = \tanh(\boldsymbol{U}_{cx}\boldsymbol{S}_{cx}\boldsymbol{V}_{cx}\boldsymbol{x}_t + \boldsymbol{W}_{ch}\boldsymbol{h}_{t-1}) \quad (12)$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \tilde{\boldsymbol{c}}_t \quad (13)$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot \boldsymbol{c}_t \quad (14)$$

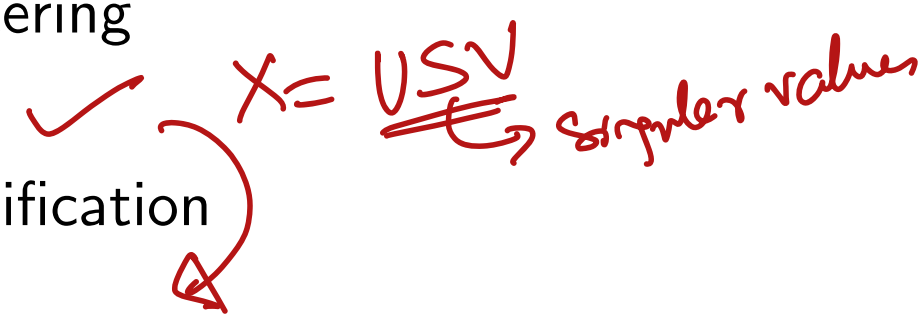
$$\boldsymbol{p}_{t+1} = \text{Softmax}(\boldsymbol{C}\boldsymbol{h}_t) \quad (15)$$

ICE #2

Multi-factor LSTM

What concept we saw in this course, does the factorization process in the multi-factor LSTM remind you of?

1. Clustering
2. SVD
3. Classification
4. PCA



Training StyleNet

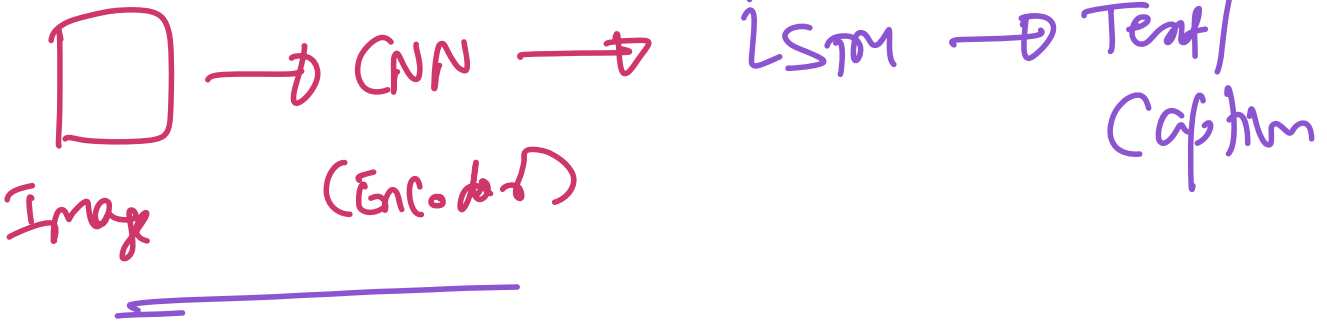
Multi-Task Learning

1. **Main Training:** Train StyleNet with "factual" factor on standard Image-Caption DataSet

Training StyleNet

Multi-Task Learning

1. **Main Training:** Train StyleNet with "factual" factor on standard Image-Caption DataSet
2. **Fine-tune Style:** Train only the Style Factors of the LSTMs for romance and humor on romance and humor texts



ICE #3

StyleNet Training

Why can't StyleNet be trained in one shot simultaneously on (Image, Caption) pairs of different styles (humorous, romantic, factual)?

1. It is possible and easy to train in one shot
2. It's hard to find data for all styles, making one shot not tractable
3. This is reminiscent of transfer learning, which has performed well in practice
4. May not be scalable to do one shot once the number of styles grow

StyleNet - Image Captioning in Style!

F: A football player in a red uniform is kicking the ball .

R: A soccer player in a red jersey is trying to win the game .

H: A football player runs toward the ball but ignore his teammates.

F: A boy jumps into a pool.

R: A boy is jumping into a pool , enjoying the happiness of childhood.

H: A boy jumps into a swimming pool to get rid of mosquitoes.

F: A group of people are standing on a beach.

R: A group of people stand on the beach , enjoying the beauty of nature.

H: A group of people are standing in front of a lake looking for pokemon go.

StyleNet - Image Captioning in Style!

F: A snowboarder in the air .

R: A man is doing a trick on a skateboard to show his courage .

H: A man is jumping on a snowboard to reach outer space .

F: A brown dog and a black dog play in the snow.

R: Two dogs in love are playing together in the snow.

H: A brown dog and a black dog are fighting for a bone.

F: A man riding a dirt bike on a dirt track .

R: A man rides a bicycle on a track , speed to finish the line.

H: A man is riding a bike on a track to avoid being late for dating.

StyleNet - Image Captioning in Style!

Standard: A man is playing guitar.
Romantic: A man practices the guitar, dream of being a rock star.
Humorous: A man is playing guitar but runs way.

StyleNet - Image Captioning in Style!

Romantic References							
Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE	CIDEr	METEOR
CaptionBot [46]	40.4	20.2	12.7	7.6	0.36	0.26	0.133
NIC [50]	42.0	21.4	12.5	7.8	0.36	0.28	0.134
Fine-tuned	43.2	21.6	12.7	7.6	0.34	0.24	0.139
Multi-task [31]	44.1	23.7	14.3	9.5	0.36	0.29	0.145
StyleNet (F)	41.2	21.4	12.1	7.7	0.36	0.24	0.135
StyleNet (R)	**46.1**	**24.8**	**15.2**	**10.4**	**0.38**	**0.31**	**0.154**
Humorous References							
Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE	CIDEr	METEOR
CaptionBot [46]	43.4	21.4	12.2	7.1	0.35	0.21	0.134
NIC [50]	43.1	22.8	13.2	7.9	0.36	0.23	0.136
Fine-tuned	43.0	20.7	12.9	7.8	0.34	0.19	0.128
Multi-task [31]	47.1	23.9	13.9	8.8	0.37	0.25	0.148
StyleNet (F)	42.9	22.3	12.9	7.7	0.36	0.23	0.135
StyleNet (H)	**48.7**	**25.4**	**14.6**	**10.1**	**0.38**	**0.27**	**0.152**

Table 1. Compared image caption results with baseline approaches on the FlickrStyle10K dataset.

Mini-Project 2 Guidelines

- **Image captioning:** On MSCOCO data set. Given an image - print an appropriate caption for it.
- **Deliverables:** You have to submit a Jupyter/IPython notebook file and report as part of your submission. You can use the template notebook given and add your solutions to it.
- **Team Work:** You can work in a team of 2. Pick your team mate for this project - When you make your report submission, you are expected to breakdown the contribution of each team member. Ensure that both team members get to work and test the Neural Network models.
- **Report:** The report should be in pdf format and have all images, plots and metrics added in it. Feel free to use either latex or word for creating it. You are required to answer all of the conceptual questions in the write up below, and show your learnings and insights.

Submission Guidelines (contd)

- You may discuss/brainstorm ideas to solve the assignment with peers - However, your submission should be your own and show your code implementation.
- Add your BLEU and METEOR scores to the google sheet given below for a fun peer learning experience!
- **Submissions:** You need to finally submit the report, train.py, eval.py, model.py and the completed EEP596_Mini_Project_2_template.ipynb file.

Dataset Description

The MS COCO'14 dataset has 123,287 images, with each image having 5 captions. This makes the dataset fairly huge and robust enough for the problem statement Automatic Image Captioning. Some of the examples of the dataset can be seen in the Figure 1. The template notebook uses Andrej Karpathy's train test split to create training, validation and test datasets.

Dataset Description

a man standing next to a zebra *in a field*

a man is standing next to a zebra

a group of people riding on the back of *an elephant*

a group of people riding **horses** down a **dirt road**

a *bike* parked next to a parking meter

a **parking meter** sitting next to a parking meter

a cat sitting on *top* of a car

a cat sitting on the **hood** of a car

a red fire hydrant sitting *in the snow*

a red fire hydrant on the **side of the road**

a plane is flying in the *blue* sky

a plane flying in the sky **with a sky background**

a small dog sitting in a *kitchen sink*

a small dog sitting on a **table** in a kitchen

a desk with a *laptop computer* and a keyboard

a desk with a **computer monitor** and keyboard

Problem Overview

Automatic Image Captioning is a project focused on generating a single line description of an image which is grammatically correct. As this project attempts to combine algorithms from both Computer Vision and Natural Language Processing, the task is quite complex and requires expertise in the fields of visual feature extraction and machine translation.

Image Captioning Deliverables

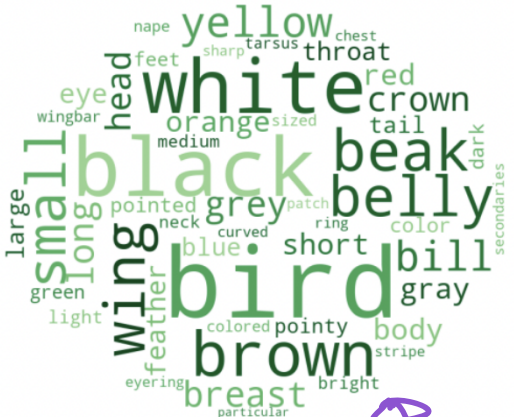
1. **Understanding the data:** For any 5 random images from the training dataset, show the images along with all the captions given for them. How many images are in the training and validation sets respectively? Describe the format in which dataset_coco.json saves the captions (the key value pairs of the dictionary). (10+5+10 points)
2. Mention any **preprocessing** of dataset required. (5 points)
3. Do **literature review** on recent models being used for image captioning. Explain in brief the deep learning models used in them. Also mention the paper being referenced. You can use one paper from the references below, and search for the other on. (40 points)
4. Go through the Data_Loader.py class and describe in a paragraph what specifically does it do? (15 points)
5. Build **word cloud** images (one each) for the training and validation datasets, in which high frequency words shall be more bolder. Do the two word clouds look similar or different and how? (20 points)

CUB-200 Data Set

CUB-200

This bird is blue with white on its chest and has a very short beak.

CUB-200



Word cloud

Image Captioning Deliverables (contd)

6. **Baseline Model:** Create a Encoder Decoder model in which the Encoder is a pretrained AlexNet model, whose output is given to the predefined LSTM-based Decoder model. Make changes in the model.py file. (15 points)
7. **Advanced Models:** Create a Encoder Decoder model in which the Encoder is a pretrained RESNET model, whose output is given to the predefined LSTM-based Decoder model. Make changes in the model.py file. (Note: If you are not able to complete the run in Google Colab, you can try training with a fraction of the training dataset.) (10 points)
8. **Understanding the model:** With the help of a block diagram explain the architecture of the encoder-decoder model. How is the encoder connected to the decoder? (10 points)
9. **Train** both the models constructed using train.py. (50 points)

Image Captioning Deliverables (contd)

10. Briefly explain how the **METEOR** metric is calculated and show the calculation of fragmentation value and METEOR score for a prediction from your model one of the test examples. (10 points)
11. **Metrics on Validation:** As an initial step towards understanding the performance of your models, compute and report the BLEU and METEOR scores on the validation data set. (HINT: nltk.translate.bleu_score.corpus_bleu and nltk.translate.meteor_score libraries in python) In a table, add the metrics and the time taken to train the model and predict caption for 1 image to the table. (20 points).
12. **Final evaluation:** Final evaluation of your model performance will be based on a "held-out" evaluation data set - the details of which will be shared in a few days (40 points).

Image Captioning Deliverables (contd)

12. **Interpretability** - Show examples of images (False Positives and False Negatives) for which the caption generated is incorrect. .....? (10 points)

Mini-Project Deliverable Deadlines

For full grade on Mini-project, and to maximize your learning from this project - Please meet all the deadlines on the deliverables below:

1. **First Milestone:** Submit the deliverables connected to understanding the data and building a baseline model by **Sunday, December 11th**

Mini-Project Deliverable Deadlines

For full grade on Mini-project, and to maximize your learning from this project - Please meet all the deadlines on the deliverables below:

1. **First Milestone:** Submit the deliverables connected to understanding the data and building a baseline model by **Sunday, December 11th**
2. **Team Mini-Project Presentation (8 minutes per team):** Your team can make a presentation in our Thursday class slot on **Thursday, December 15th**

Mini-Project Deliverable Deadlines

For full grade on Mini-project, and to maximize your learning from this project - Please meet all the deadlines on the deliverables below:

1. **First Milestone:** Submit the deliverables connected to understanding the data and building a baseline model by **Sunday, December 11th**
2. **Team Mini-Project Presentation (8 minutes per team):** Your team can make a presentation in our Thursday class slot on **Thursday, December 15th**
3. **Final Submission:** Final submission of all deliverables by **Friday, December 16th**

Breakout for Takeaways!

Discuss Takeaways (5 mins)

From today's lecture in your zoom group