
MLDS HW2-1

TAs
ntu.mldsta@gmail.com

Outline

- ❖ **Timeline**
- ❖ **Task Descriptions**
- ❖ **Q&A**

Timeline

Two Parts in HW2

- (2-1) Video caption generation
 - Sequence-to-sequence model
 - Training Tips
- (2-2) Chat-bot

Schedule

- 3/30:
 - Release HW2-1
- 4/13:
 - Release HW2-2
- 4/27:
 - Midterm
 - HW1 上台分享
- 5/4:
 - All HW2 due (including HW2-1, HW2-2)

Task Descriptions

HW2-1: Video caption generation

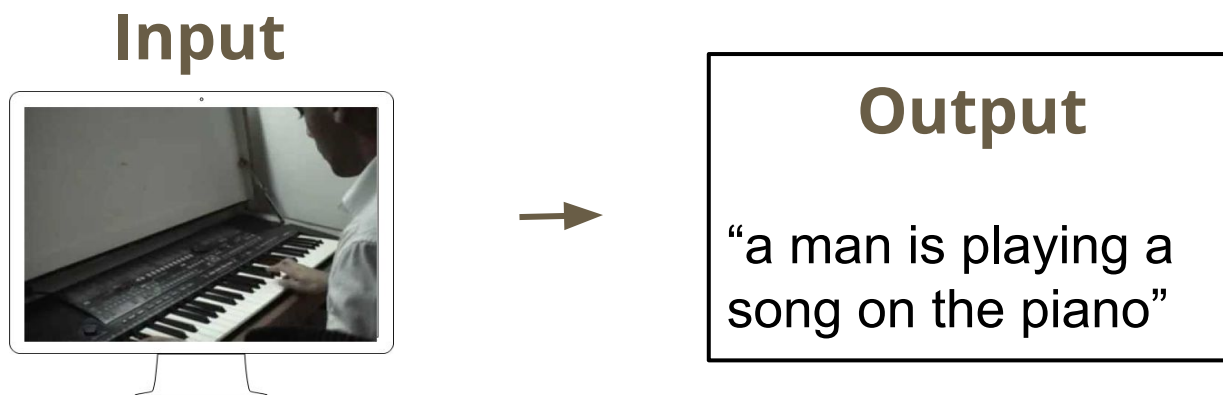
- Introduction
- Sequence-to-sequence model
- Training Tips
 - Attention
 - Schedule Sampling
 - Beamsearch
- How to reach the baseline ?

HW2-1: Video caption generation

- Introduction
- Sequence-to-sequence model
- Training Tips
 - Attention
 - Schedule Sampling
 - Beamsearch
- How to reach the baseline ?

HW2-1 Introduction

- Video Caption Generation
 - a. Input : A short video
 - b. Output: The corresponding caption that depicts the video

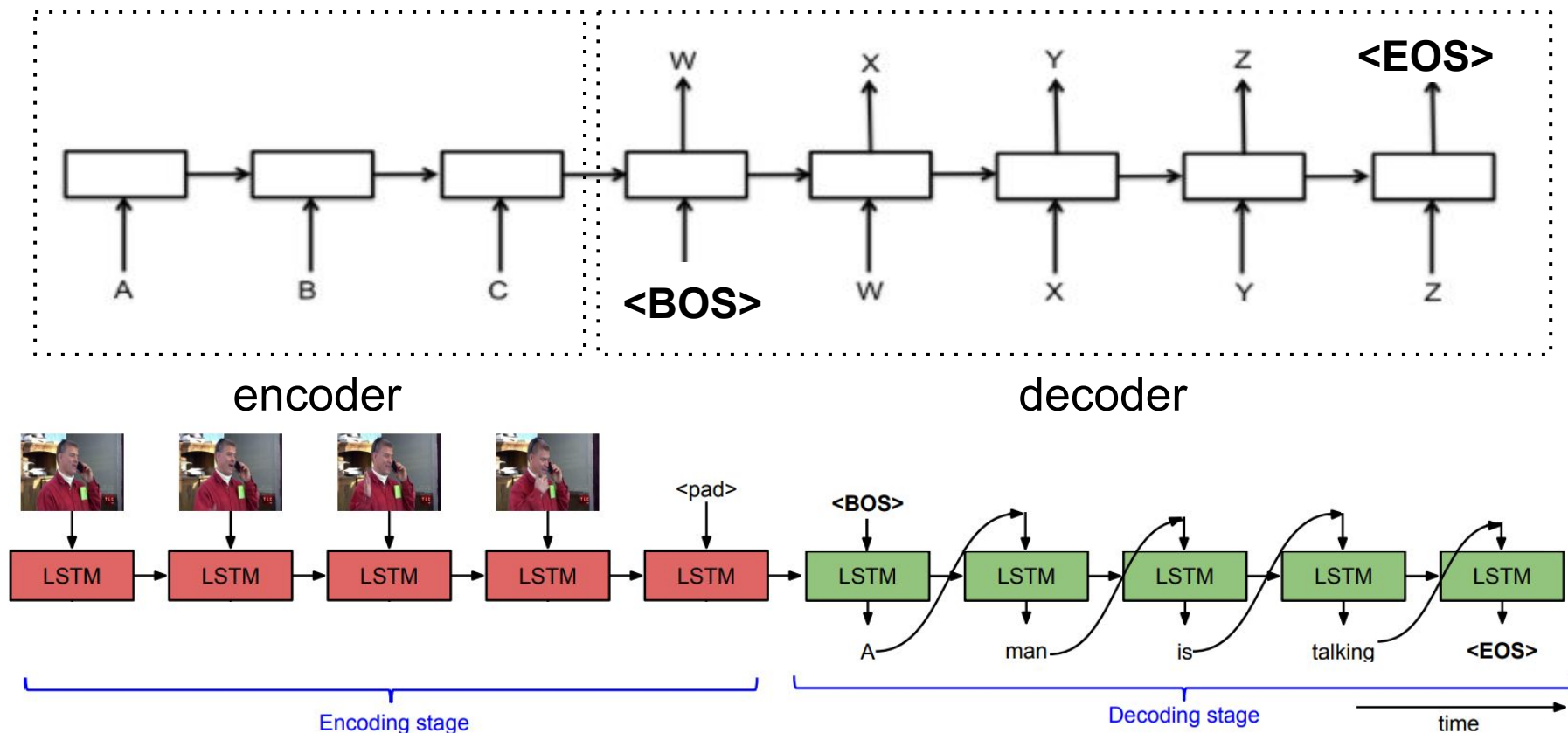


- There are several difficulties including:
 - a. Different attributes of video (object, action)
 - b. Variable length of I/O

(In this task, video features will be provided)

HW2-1 Sequence-to-sequence ^{1/5}

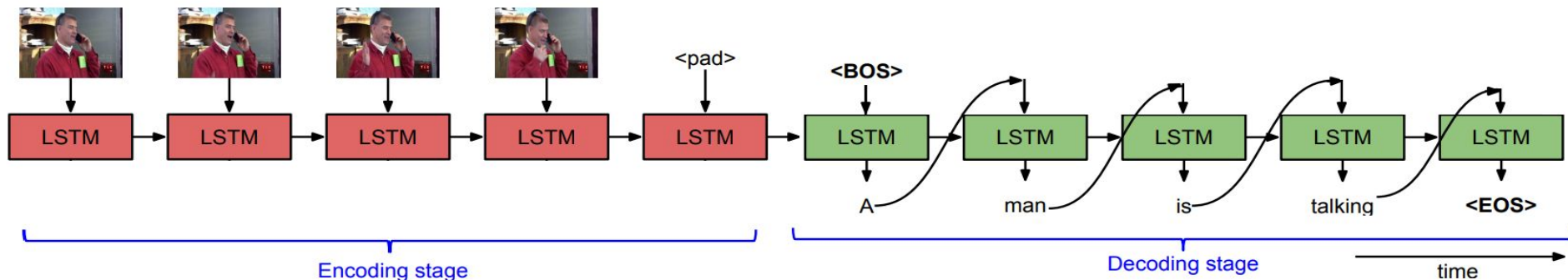
- Two recurrent neural networks (RNNs)
 - an encoder that processes the input
 - a decoder that generates the output



HW2-1 Sequence-to-sequence 2/5

- **Data preprocess:**

- Dictionary - most frequently word or min count
- other tokens: <PAD>, <BOS>, <EOS>, <UNK>
 - <PAD> : Pad the sentences to the same length
 - <BOS> : Begin of sentence, a sign to generate the output sentence.
 - <EOS> : End of sentence, a sign of the end of the output sentence.
 - <UNK> : Use this token when the word isn't in the dictionary or just ignore the unknown word.



HW2-1 Sequence-to-sequence ^{3/5}

- **Text Input:**

reference

- One-hot Vector encoding

(1-to-N coding, N is the size of the vocabulary in dictionary)

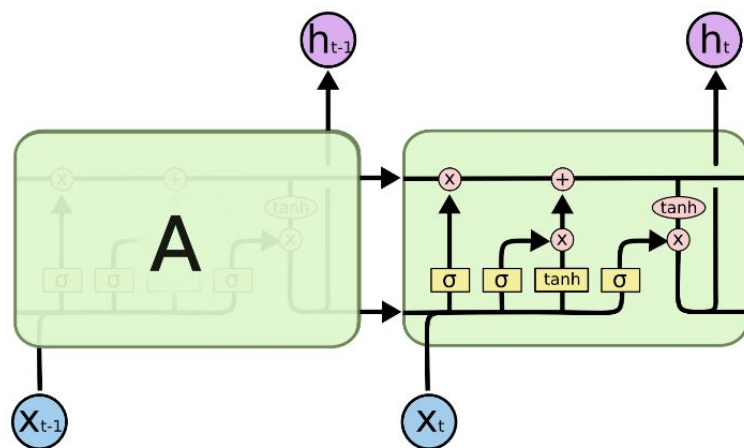
- e.g.

- neural = $[0, 0, 0, \dots, 1, 0, 0, \dots, 0, 0, 0]$

- network = $[0, 0, 0, \dots, 0, 0, 1, \dots, 0, 0, 0]$

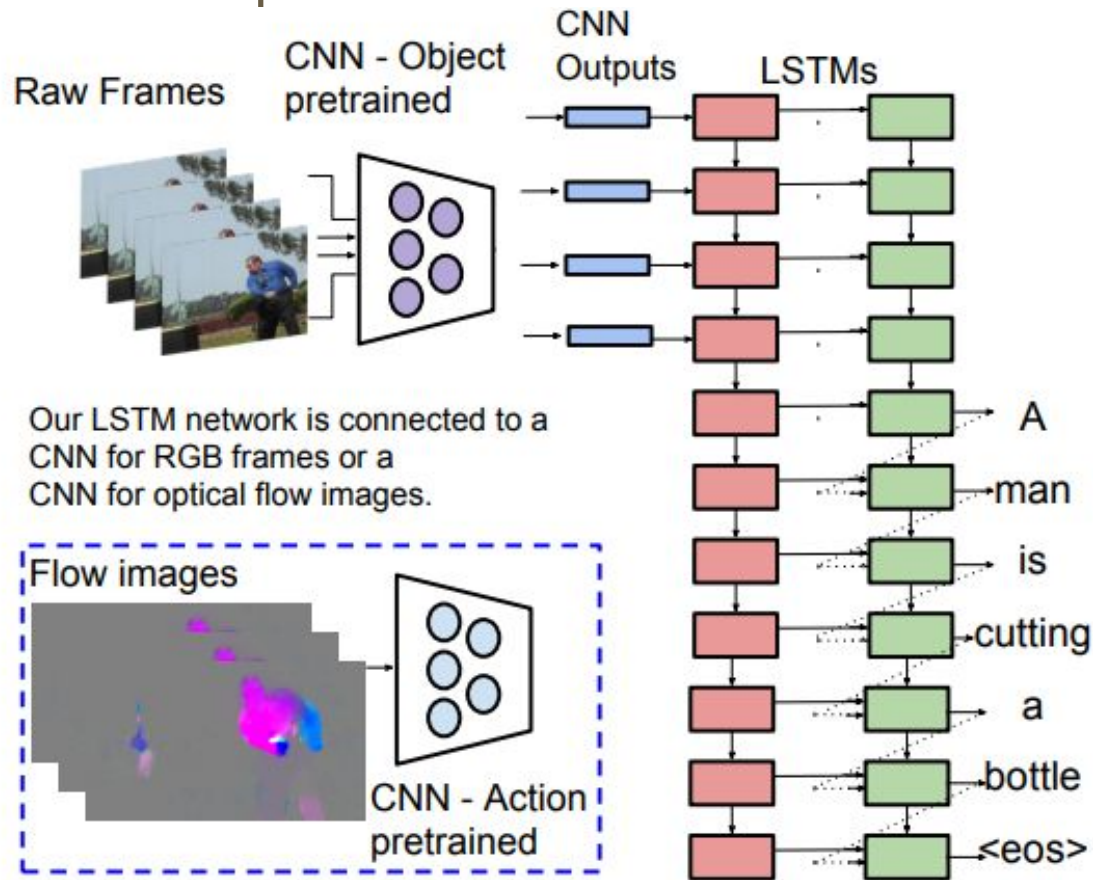
- **LSTM unit:**

cell output than project to a vocabulary-size vector



HW2-1 Sequence-to-sequence - S2VT ^{4/5}

- Sequence-to-Sequence Based Model: S2VT

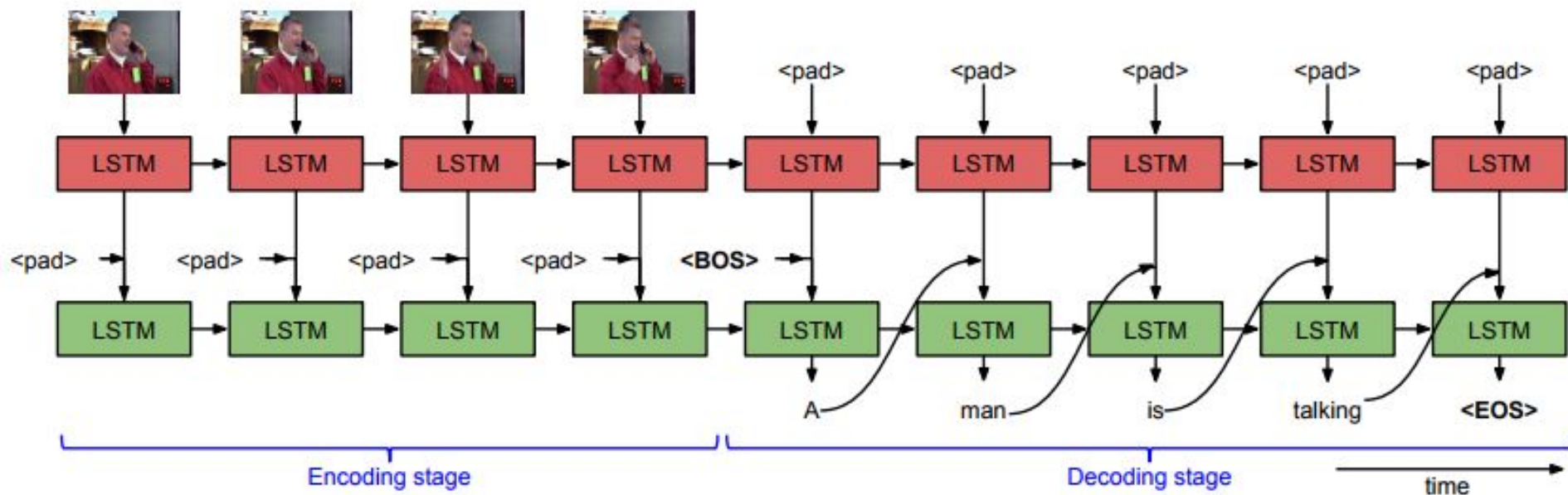


Refer to the following paper for detailed info:

<http://www.cs.utexas.edu/users/ml/papers/venugopalan.iccv15.pdf>

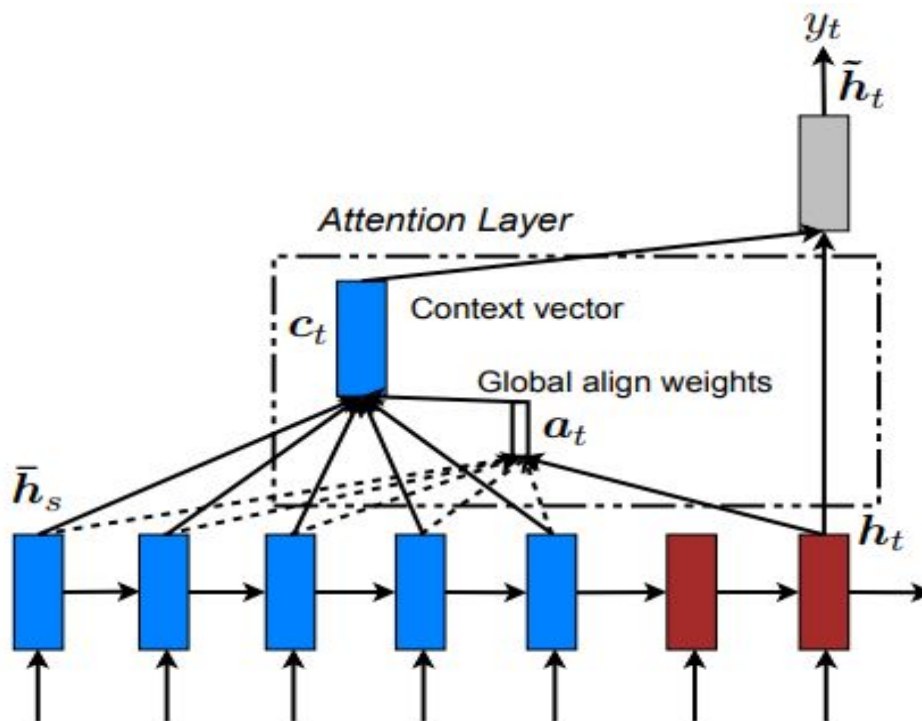
HW2-1 Sequence-to-sequence - S2VT ^{5/5}

- Sequence-to-Sequence Based Model: S2VT
 - Two layer LSTM structure



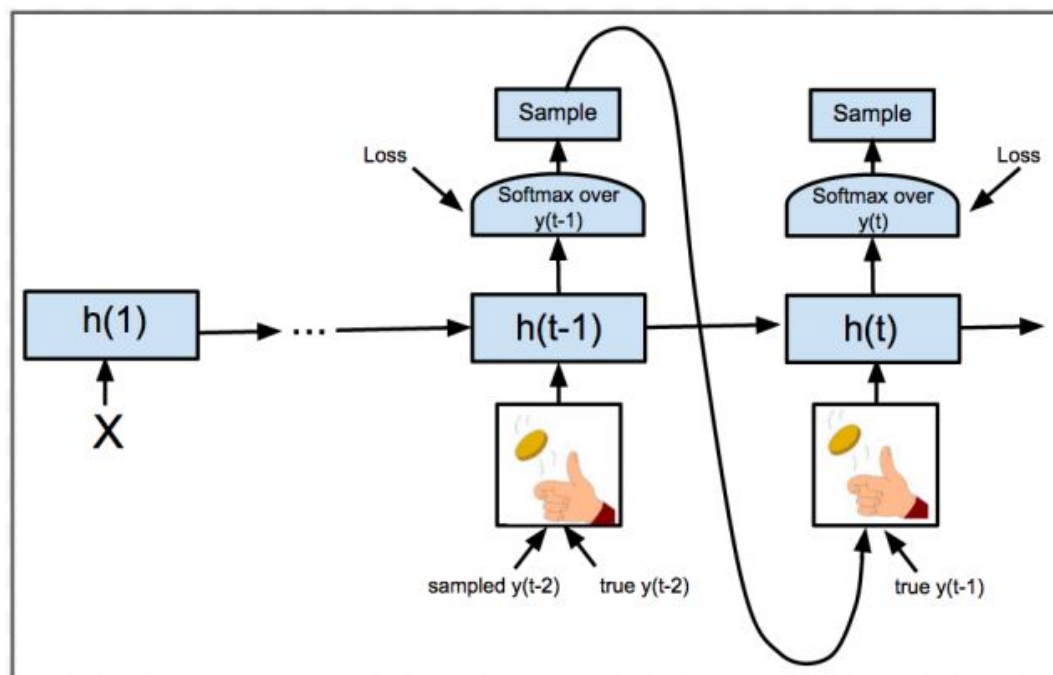
HW2-1 Training Tips - Attention ^{1/3}

- Attention on encoder hidden states :
 - Allow model to peek at different sections of inputs at each decoding time step



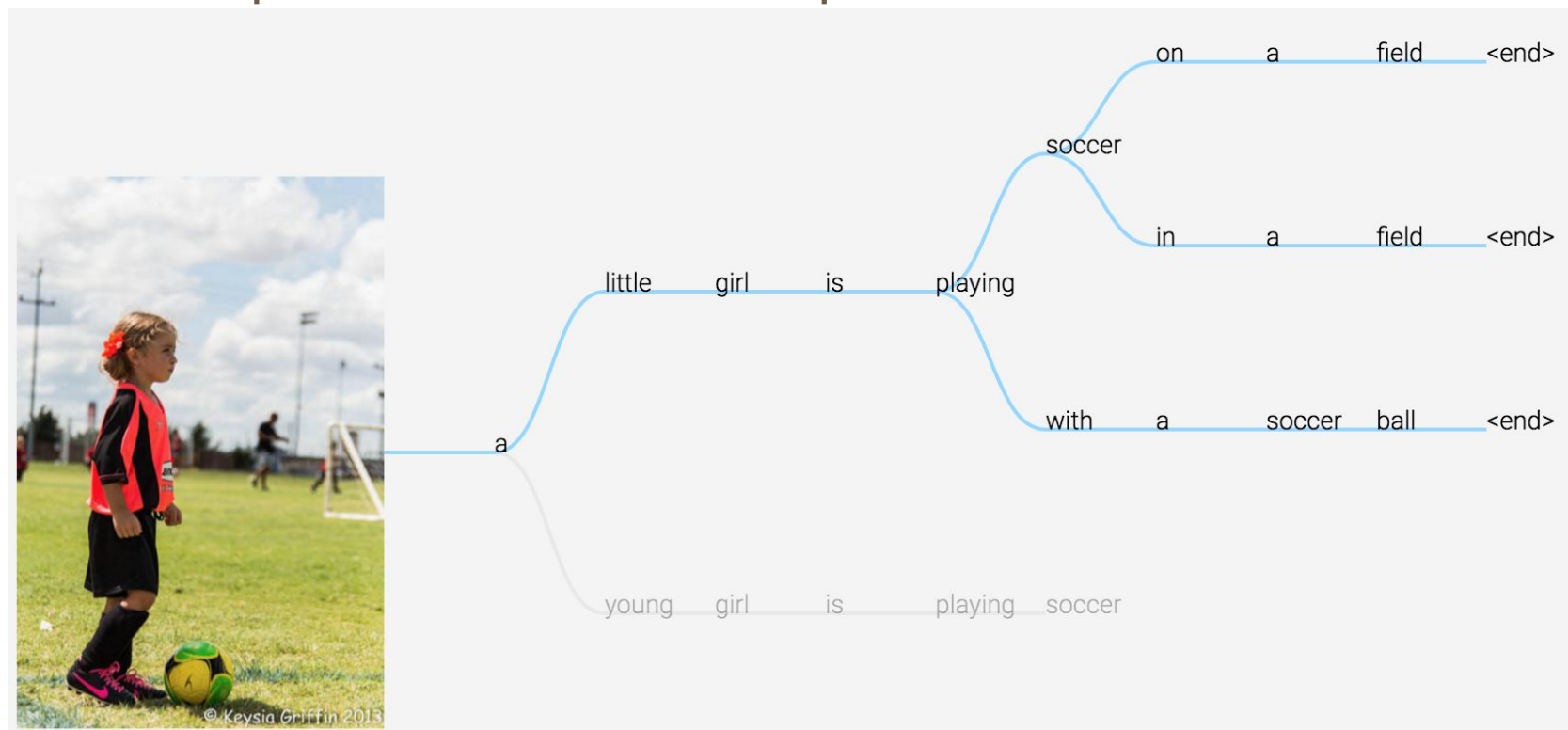
HW2-1 Training Tips - Schedule Sampling ^{2/3}

- Schedule Sampling:
 - To solve “exposure bias” problem,
When training, we feed (groundtruth) or (last time step’s output) as input at odds



HW2-1 Training Tips - Beam search ^{3/3}

- Beam search:
 - keep a fixed number of paths



Demo: <http://dbs.cloudcv.org/captioning>

HW2-1 How to reach the baseline ? ^{1/2}

- Evaluation: BLEU@1

- Precision = correct words / candidate length

- $$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}$$

where c = candidate length, r = reference length

- BLEU@1 = BP * Precision

- e.g.:

Ground Truth : *a man is mowing a lawn*

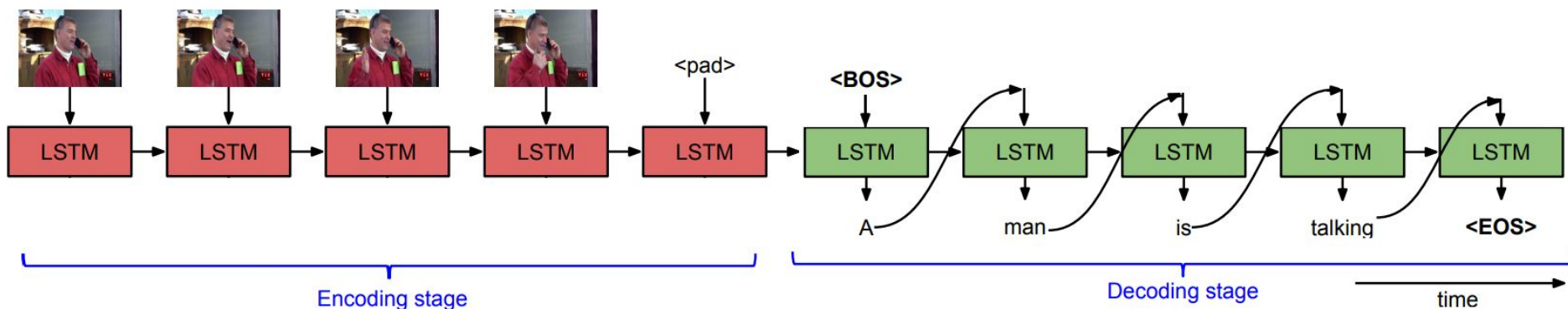
Prediction : *a man is riding a man on a woman is riding a motorcycle*

BLEU: $1 * 4/13 = 0.308$

- [paper](#)

HW2-1 How to reach the baseline ? ^{2/2}

- **Baseline: BLEU@1 = 0.65 (Captions Avg.)**
- baseline model:



- Training Epoch = 200
- LSTM dimension = 256
- Learning rate = 0.001
- vocab size = min count > 3
- AdamOptimizer
- Training time = 72 mins, using 960 TX

Data & format

- Dataset:
 - MSVD
 - 1450 videos for training
 - 100 videos for testing
- Format:
 - [Download](#) MLDS_hw2_1_data.tar.gz (4/9 update)

```
MLDS_hw2_1_data
├── testing_data
│   ├── feat
│   └── video
├── training_data
│   ├── feat
│   └── video
├── bleu_eval.py
├── sample_output_testset.txt
├── testing_id.txt
├── testing_label.json
├── training_id.txt
└── training_label.json
```

Submission & Rules

- Please implement **one seq-to-seq model** (or it's variant) to fulfill the task
- Extra dataset is allowed to use.
- Allow package:
 - python 3.6
 - **TensorFlow r1.6 ONLY** (CUDA 9.0)
 - PyTorch 0.3 / torchvision
 - Keras 2.0.7 (TensorFlow backend only)
 - MXNet 1.1.0, CNTK 2.4
 - matplotlib, Python Standard Library
 - If you want to use other packages, please ask TAs for permission first!
 - **new allowed package: pandas, tensorlayer, gensim**

Submission & Rules

- Deadline : **2018/5/4 23:59 (GMT+8)**
- Upload **code** and **report** of HW2-1, HW2-2 to Github in **different** directory.
- For HW2-1 :
 - Your github must have directory **hw2/hw2_1/**, and there should be:
(1) report.pdf **(2) your_seq2seq_model** **(3) hw2_seq2seq.sh**
(4) model_seq2seq.py (*training code should include*)
 - If your model are too big for github, upload to a cloud space and **write it in your script to download the model.**
 - Please write shell script “**hw2_seq2seq.sh**” to run your code and follow the script usage below:
 - `./hw2_seq2seq.sh $1 $2`
 - \$1: the data directory, \$2: test data output filename (format: .txt)
 - Example `./hw2_seq2seq.sh hw2_1_data/ sample_output_testset.txt`Your script should be done within **10 mins** excluding model downloading.
 - **Please do not upload any dataset to Github (include external dataset).**

Grading Policy

- HW2-1 : 15%
 - Baseline (4%):
 - BLEU@1 = **0.6** (Captions Avg.) (4/16 修改)
 - TAs review (4%):
 - Grammar score (2%)
 - Relative score (2%)
 - Report (7%)
- HW2-2 : 10%
- 分工表 : 0.5%
- 上台分享 : 1%
- 上台分享前三名 : 1%

Grading Policy - Report (7%)

- Do not exceed 4 pages and written in Chinese.
- Model description (3%)
 - Describe your seq2seq model
- How to improve your performance (3%)
(e.g. Attention, Schedule Sampling, Beamsearch...)
 - Write down the method that makes you outstanding (1%)
 - Why do you use it (1%)
 - Analysis and compare your model without the method. (1%)
- Experimental results and settings (1%)
 - parameter tuning, schedual sampling ... etc
- README : please specify library and the corresponding version in README

Grading Policy - NOTICE

- Late submission (link):
 - Please fill the late submission form first only if you will submit HW late.
 - Please push your code before you fill the form
 - There will be 25% penalty per day for late submission, so you get 0% after four days
- Bug:
 - You will get 0% in Baseline and TAs review if the required script has bug.
 - If the error is due to the format issue, please come to fix the bug at the announced time, or you will get 10% penalty afterwards.

Q&A

ntu.mldsta@gmail.com

Q1: 請問助教會跑training的程式嗎？

A: 不會。我們所規定的十分鐘只包含testing。除非我們認為有必要就會請你們來跑training的code。

Q2: 有推薦上傳model的平台嗎？

A: dropbox, google drive都是大家常用的平台。不過推薦大家可以使用gitlab, 操作方法與github類似, 但是可以上傳大容量的檔案。

p.s. github 單一檔案上傳上限為100MB, 若超過50MB則會出現警告, 但依舊能上傳。也可參考網路上的教學 ([ref](#))。

Q3: test set 的答案怎麼一起給了？

A: 因為沒有Kaggle, 方便大家validation 和測準確率, 因此也給大家testset 的答案。

Q4: data 裡的feature是怎麼抽的呢？

A: pretrain在ILSVRC的VGG19。

80*4096維的feature, 是指每個影片抽80個frame, 每個frame有4096維feature。

Q5: Average bleu score 是怎麼算的呢？

A: 對於每個影片，你的答案會對他的所有的字幕算bleu score。
將所有影片的分數取平均後，就是你的總bleu score。

p.s. 詳細演算法請見 `bleu_eval.py`