Video Future Frame Prediction of Bouncing Numerals

Teaching Assistant: Jungbeom Lee, Eunji Kim
email: ml.class.snu@gmail.com

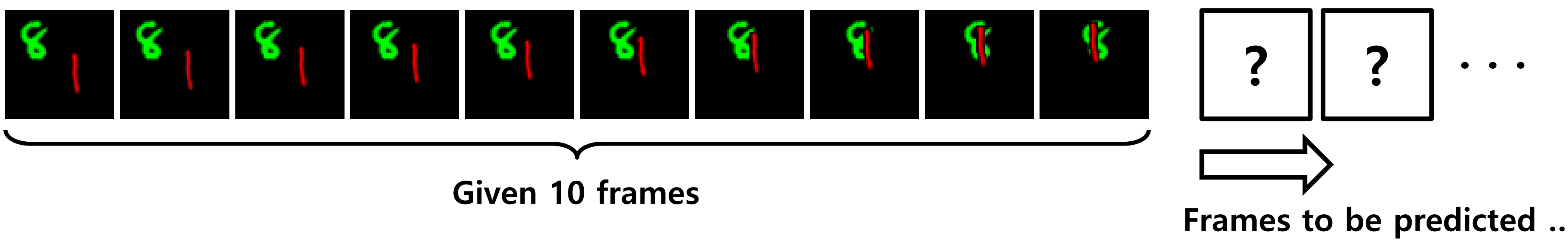
Final Project

Machine Learning Fundamentals and Applications in ECE

M2608.001300_001, 2019 Spring

Introduction

- Video Frame Prediction
  - Given some frames, what would the next several frames look like?



  - Our goal is to predict next upcoming 10 frames given past 10 frames
  - We should consider both **spatial** and **temporal** information

Dataset

- Bouncing Numerals Dataset (not a public dataset)
  - 10,000 **training** sequences with 20 frames each.
  - 500 **validation** sequences with 20 frames each.
  - 500 **test** sequences with **10 frames** each.
    - You should predict the following 10 frames of each sequence.

- Data Utilization
  - Train your network **only using** training sequences
  - Find hyper-parameters, network architecture, … with validation sequences
  - The ground truth of test sequences will not be available publicly.
    - You can check your score and current rank on the leaderboard of Kaggle

Evaluation (1/2)

- Evaluation Metric
  - (Main Metric) Mean Absolute Error (MAE)
    - MAE = $\frac{1}{w \times h \times c} \sum_{w,h,c} |x_{w,h,c} - \hat{x}_{w,h,c}|$
      - x : ground-truth frame, $\hat{x}$: predicted frame
    - The MAE is averaged over predicted 10 frames
  - (Additional Metric) Structural Similarity (SSIM)
    - $\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$
    - SSIM will not be calculated by Kaggle.
    - You can compute SSIM with “pyssim” module on python

Evaluation (2/2)

- Sign up Kaggle, and come to project competition page
  - https://www.kaggle.com/t/2b41fd37f29f434988c88984bfbfa816

- You should also upload the prediction results of test sequences to ETL.
  - The format of the files **SHOULD** be as follows.
    - test_predicted/sequencexxx/framesxx.png
      - For example, the first predicted frame of sequence 123 should be test_predicted/sequence123/frames00.png
      - As a result, there must be 500 folders for each sequence, and each folder must have 10 images, from frames00 through frames09.

File description

- Unzip the given file
  - Unzip 2019Spring_ML_final.zip
- 2019Spring_ML_final/Data
  - Contains train, validation, and test sequences
  - Note that test sequences contain only **10 frames**
- 2019Spring_ML_final/data_utils.py
  - It provides train, validation, and test data loading functions
  - You can change those codes freely
- 2019Spring_ML_final/make_submission_file.py

Evaluation

- Make submission csv file for Kaggle
  – To submit your results to Kaggle, you need to create a csv file.
    - Run make_submission_file.py
      – Do not forget to change "root_dir" and "save_dir"
        » root_dir : path to test_predicted folder
        » save_dir : path where the csv file for submission will be saved
  – Do not change make_submission_file.py
    - If you have any questions or concerns, please let TA know.

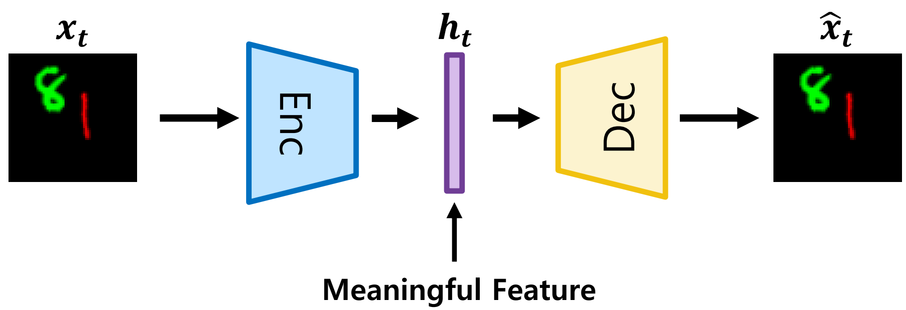
submission.csv

2584	0
2585	0
2586	45
2587	0
2588	0
2589	133
2590	0
2591	0
2592	180
2593	0
2594	0
2595	177
2596	0
2597	0
2598	162
2599	0
2600	0
2601	130
2602	0
2603	0
2604	108
2605	0
2606	0
2607	98
2608	0

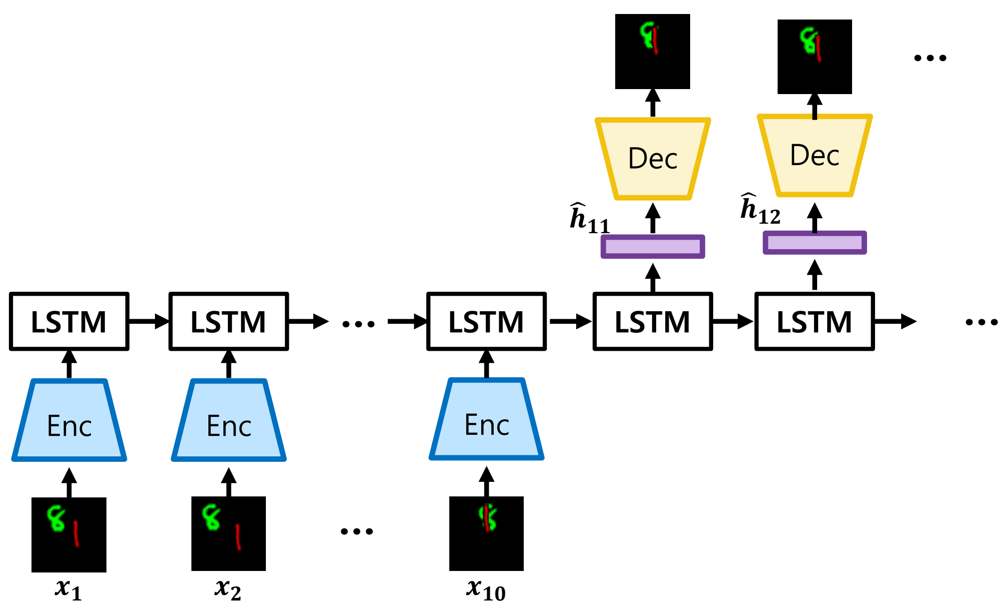
⋮

Hint (1/2)

- How was the baseline created?
  - Step1 : Train an autoencoder



  - Step2 : Each encoded feature is fed into LSTM sequentially



Loss functions

Step1: $\|x_t - \hat{x}_t\|_2^2$

Step2: $\sum_{t=11}^{20} \|h_t - \hat{h}_t\|_2^2$

Hint (2/2)

- Hidden Markov Model (HMM) can replace LSTM
  - HMM will soon be covered in class.

Three basic problems for HMM

problem 1:

given observation sequence $O = O_1 O_2 \cdots O_T$ and model $\lambda = (A, B, \pi)$, how do we **efficiently compute** $P(O \mid \lambda)$, the probability of the observation sequence, given the model?

problem 2:

given observation sequence $O = O_1 O_2 \cdots O_T$ and model λ, how do we **choose a corresponding state sequence** $Q = q_1 q_2 \cdots q_T$ which is optimal in some meaningful sense?

problem 3:

how do we **adjust model parameters** $\lambda = (A, B, \pi)$ to maximize $P(O \mid \lambda)$?

Timeline

- Timeline **(시간, 장소 반드시 확인)**
  - 5/20: Project Out
  - 5/22: Project 소개 (7pm, 모랩)
  - 6/5: Project Q&A (by Jungbeom Lee, 7pm, 301동 102호)
  - 6/10: Project Q&A (by Eunji Kim), Final Q&A (by all TAs), 기존 수업시간
  - 6/12: Final Exam (3:30 pm, TBD)
  - **6/14: Final Exam Claim (TBD)
  - ** 6/17: Project Due (ETL에 file 업로드 꼭)

- TA schedule
  - Jungbeom Lee attends the conferences in the US from June 7 to June 23.
    - Quick answers to your project questions may be difficult.
    - Eunji Kim can give you a quick answer, but if not, Jungbeom Lee may be able to answer in **12:00 PM ~ 2:00 PM Korea time** (online).