

---

# ADL x MLDS 2017 Fall

## HW1 - Sequence Labeling

2017/10/02  
adlxmls@gmail.com

---

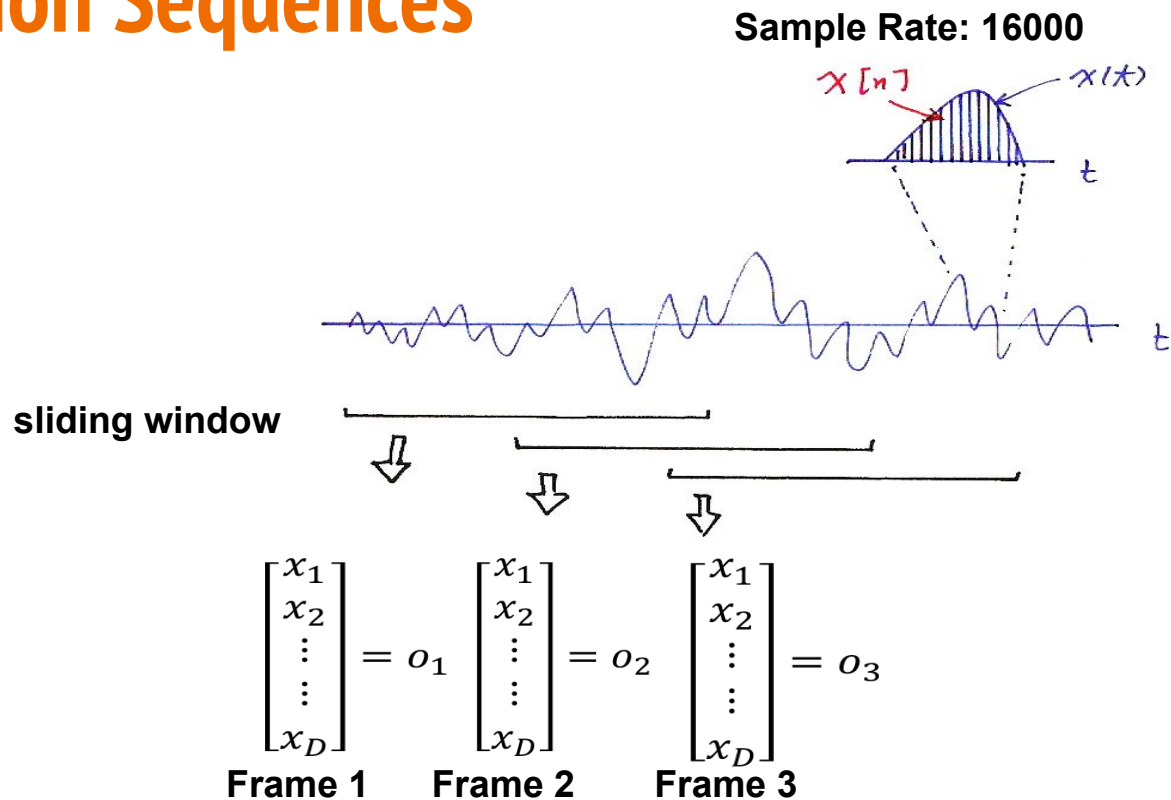
# Outline

- Task Description
  - Phone sequence labeling
  - TIMIT Dataset and Data Format
- Recurrent Neural Networks & Convolutional Neural Networks
- Kaggle
- Grading
- Format and Submission Rules

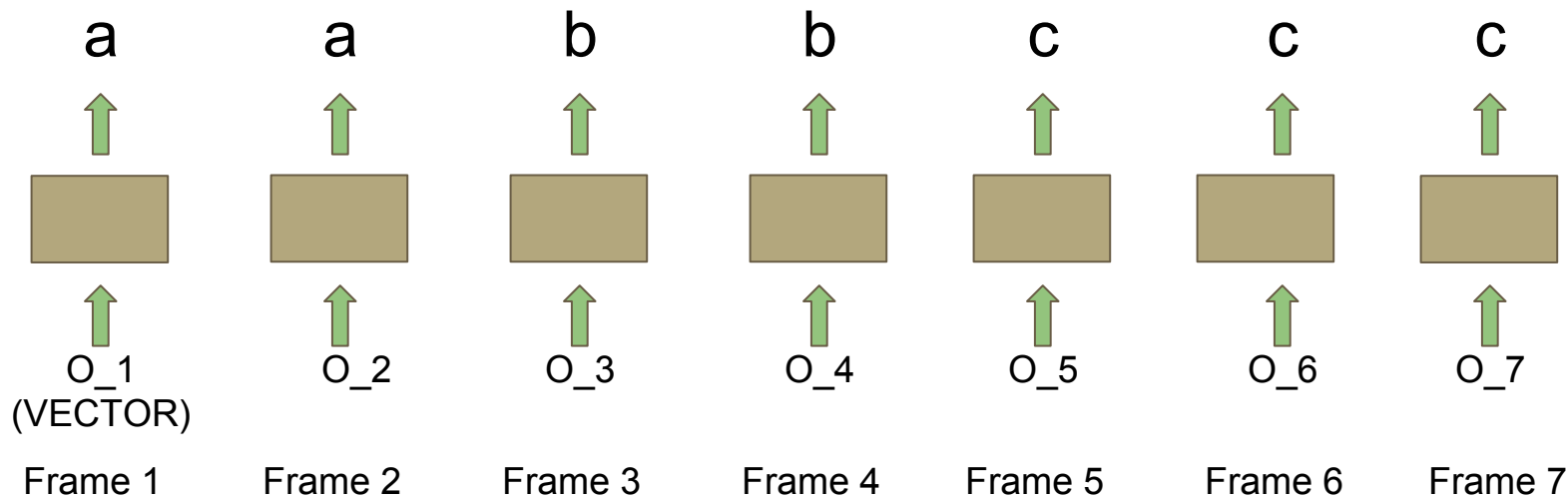
# Speech Recognition

- In speech processing
  - Each word consist of syllables
  - Each syllables consist of phone
  - “青色” → “青(く | ㇿ)色(ゑ ㇿ、)” → ”く ” (syllables)  
青:TSI --I -N (phone)  
色:S--@ (phone)
- Each time frame, with an observance (**vector**) mapped to a phone.

# Observation Sequences



# Framewise Prediction



# Phone Prediction

- What really matters in speech recognition is the **final phone sequence**, not the **framewise alignment**.
- That is, the final evaluation in this homework is based on the **phone sequence**.
- You have to trim the frame-level sequence into phone sequence.

# Trimming on Framewise Sequence (1/2)

- Remove consecutive duplicate labels

Framewise prediction: {a, a, b, b, c, c, c}

Phone prediction : {a, b, c}

You need to report result in phone sequence

## Trimming on Framewise Sequence (2/2)

- Remove only leading and tailing silence

Framewise prediction: {<sil>, <sil>, a, a, b, <sil>, c, c, <sil>}

Phone prediction : {a, b, <sil>, c}

You need to report result in phone sequence



# Dataset (1/2)

- **TIMIT**(Texas Instrument and **M**assachusetts **I**nstitute of **T**echnology)
- Well-transcribed speech of American English speakers of different sexes and dialects.
- Designed for the development and evaluation of ASR systems.
- Features
  - MFCC: 39 dim
  - FBank: 69 dim

# Dataset (2/2)

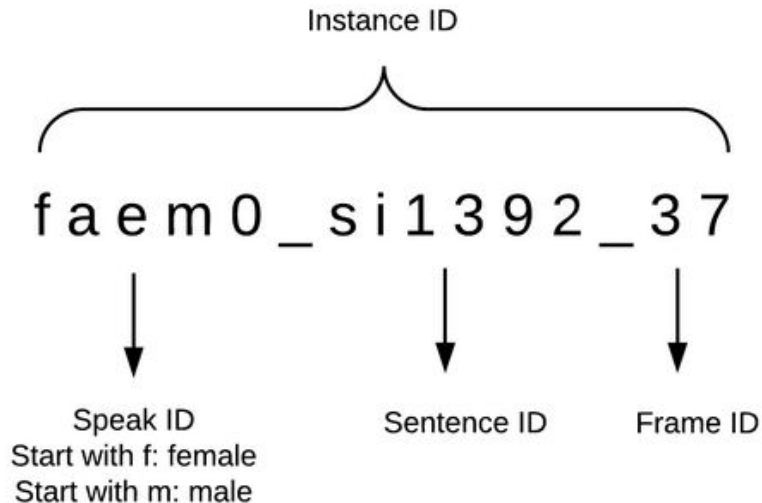
Each instance consist of 3 parts: Speaker ID, Sentence ID, Frame ID

Ex:

Speaker ID: faem0

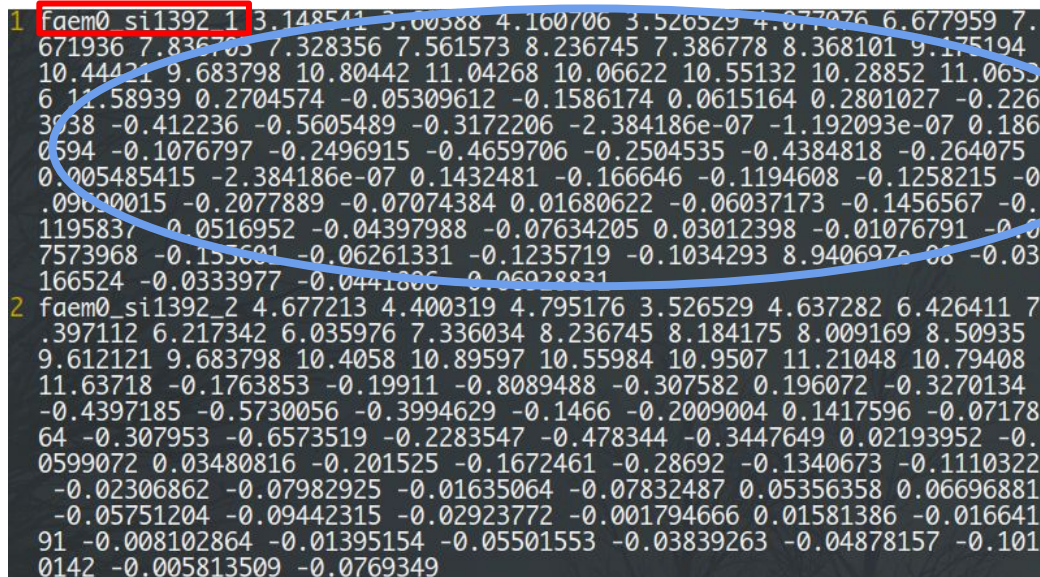
Sentence ID: si1392

Frame ID: 37



# Data Format (1/3)

- WAV file: Speaker-Sentence\_ID + .wav → Check by your ears
- ARK file: Instance ID + features



```
1 faem0_si1392_1 3.148541 3.60388 4.160706 3.526529 4.077076 6.677959 7.671936 7.836705 7.328356 7.561573 8.236745 7.386778 8.368101 9.175194 10.44421 9.683798 10.80442 11.04268 10.06622 10.55132 10.28852 11.0653 6 11.58939 0.2704574 -0.05309612 -0.1586174 0.0615164 0.2801027 -0.226 3938 -0.412236 -0.5605489 -0.3172206 -2.384186e-07 -1.192093e-07 0.186 0594 -0.1076797 -0.2496915 -0.4659706 -0.2504535 -0.4384818 -0.264075 0.005485415 -2.384186e-07 0.1432481 -0.166646 -0.1194608 -0.1258215 -0.09690015 -0.2077889 -0.07074384 0.01680622 -0.06037173 -0.1456567 -0.1195837 0.0516952 -0.04397988 -0.07634205 0.03012398 -0.01076791 -0.07573968 -0.155601 -0.06261331 -0.1235719 -0.1034293 8.940697e-08 -0.03166524 -0.0333977 -0.0441806 0.06928831
2 faem0_si1392_2 4.677213 4.400319 4.795176 3.526529 4.637282 6.426411 7.397112 6.217342 6.035976 7.336034 8.236745 8.184175 8.009169 8.50935 9.612121 9.683798 10.4058 10.89597 10.55984 10.9507 11.21048 10.79408 11.63718 -0.1763853 -0.19911 -0.8089488 -0.307582 0.196072 -0.3270134 -0.4397185 -0.5730056 -0.3994629 -0.1466 -0.2009004 0.1417596 -0.07178 64 -0.307953 -0.6573519 -0.2283547 -0.478344 -0.3447649 0.02193952 -0.0599072 0.03480816 -0.201525 -0.1672461 -0.28692 -0.1340673 -0.1110322 -0.02306862 -0.07982925 -0.01635064 -0.07832487 0.05356358 0.06696881 -0.05751204 -0.09442315 -0.02923772 -0.001794666 0.01581386 -0.016641 91 -0.008102864 -0.01395154 -0.05501553 -0.03839263 -0.04878157 -0.101 0142 -0.005813509 -0.0769349
```

# Data Format (2/3)

- LAB file: Instance ID + , + label
- 48 phones
- Map them to 39 phones by yourselfs

```
1 maeb0_si1411_1,sil
2 maeb0_si1411_2,sil
3 maeb0_si1411_3,sil
4 maeb0_si1411_4,sil
5 maeb0_si1411_5,sil
6 maeb0_si1411_6,sil
7 maeb0_si1411_7,sil
8 maeb0_si1411_8,sil
9 maeb0_si1411_9,sil
10 maeb0_si1411_10,sil
11 maeb0_si1411_11,r
12 maeb0_si1411_12,r
13 maeb0_si1411_13,r
14 maeb0_si1411_14,r
15 maeb0_si1411_15,r
16 maeb0_si1411_16,r
17 maeb0_si1411_17,r
18 maeb0_si1411_18,r
19 maeb0_si1411_19,ix
20 maeb0_si1411_20,ix
21 maeb0_si1411_21,ix
22 maeb0_si1411_22,ix
```

# Data Format (3/3)

- MAP file: 2 mapping

(1) 48 phones - 39 phones

(2) 48 phones - 48 English characters

Delimiter: '\t'

aa	aa
ae	ae
ah	ah
ao	aa
aw	aw
ax	ah
ay	ay
b	b
ch	ch
cl	sil

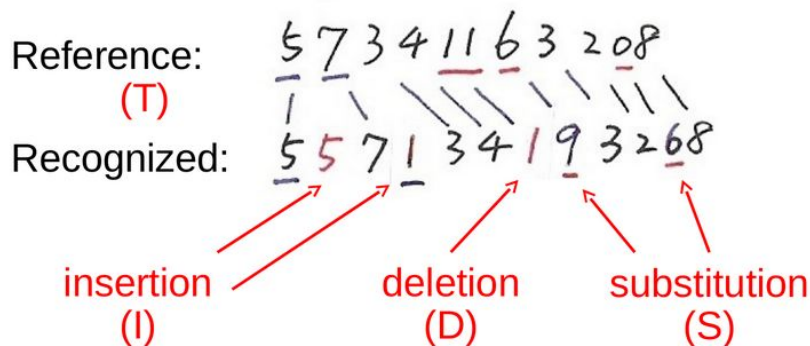
MAP (1)

aa	0	a
ae	1	b
ah	2	c
ao	3	d
aw	4	e
ax	5	f

MAP (2)

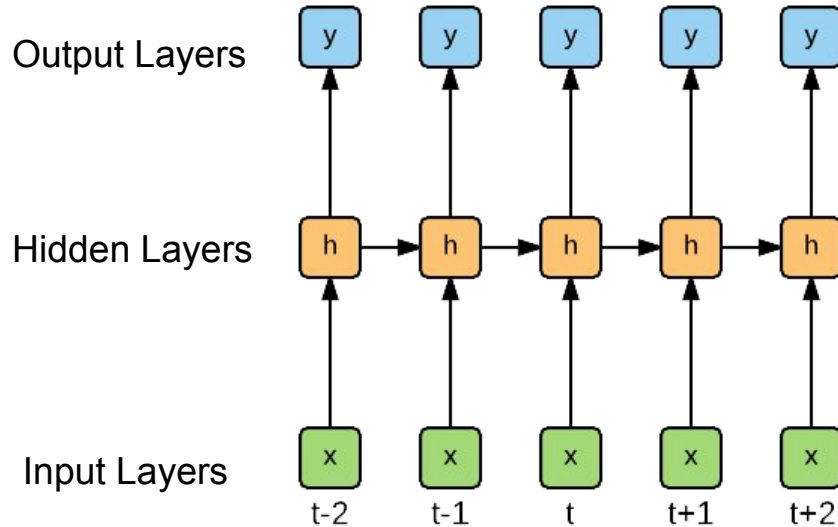
# Evaluation

- **Average Phone Sequence Edit Distance**
  - Compare your trimmed phone sequence with correct ones
- **Edit Distance = Insertion + Deletion + Substitution**
- Consider the following case, edit distance =  $I + D + S = 2 + 1 + 2 = 5$



# Recurrent Neural Networks

# RNN - Unfolded View

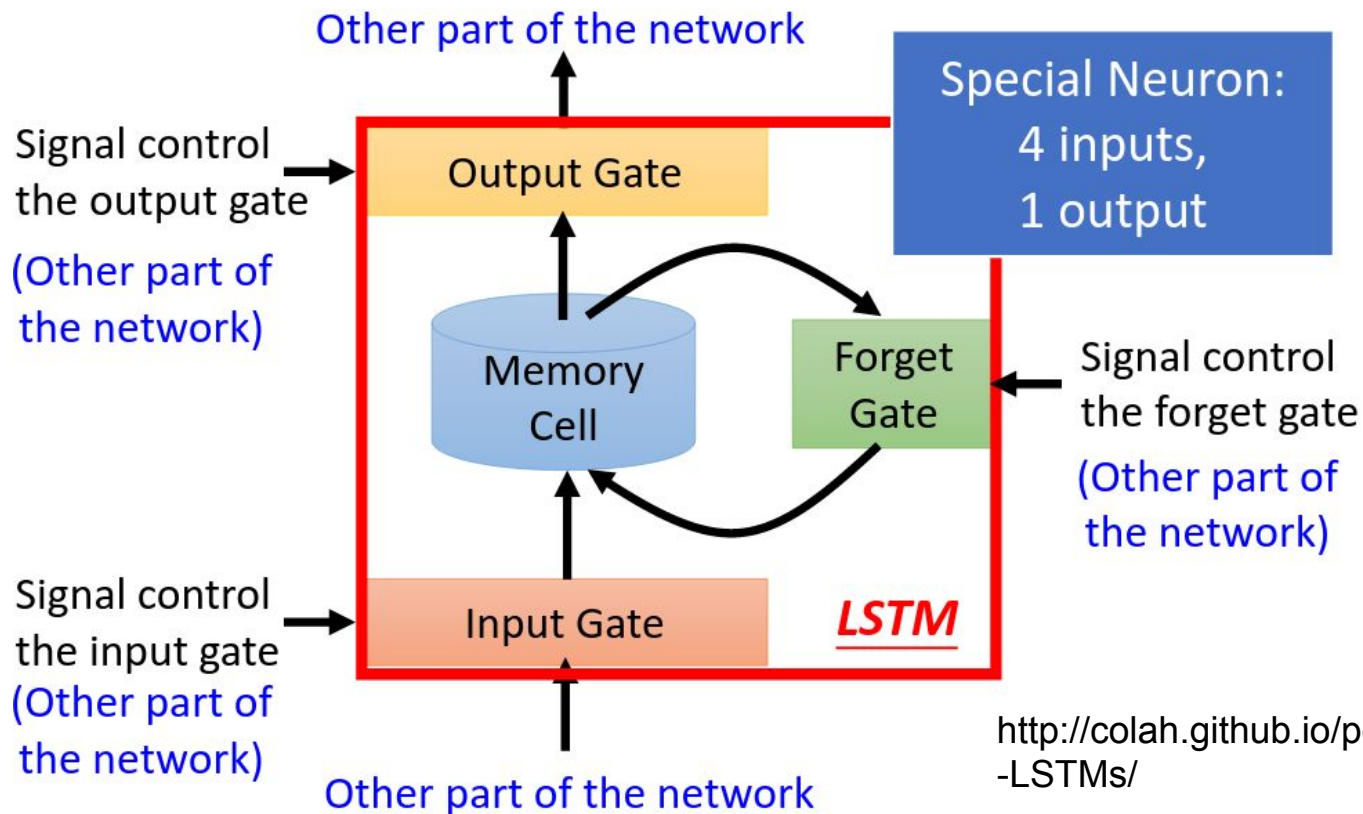




# RNN

- With Hidden Layer (Memory Layer), RNN can learn more long-term information.
  - **Sequential Information.**
- With **LSTM** gated-extension, the RNN can learn longer and longer.

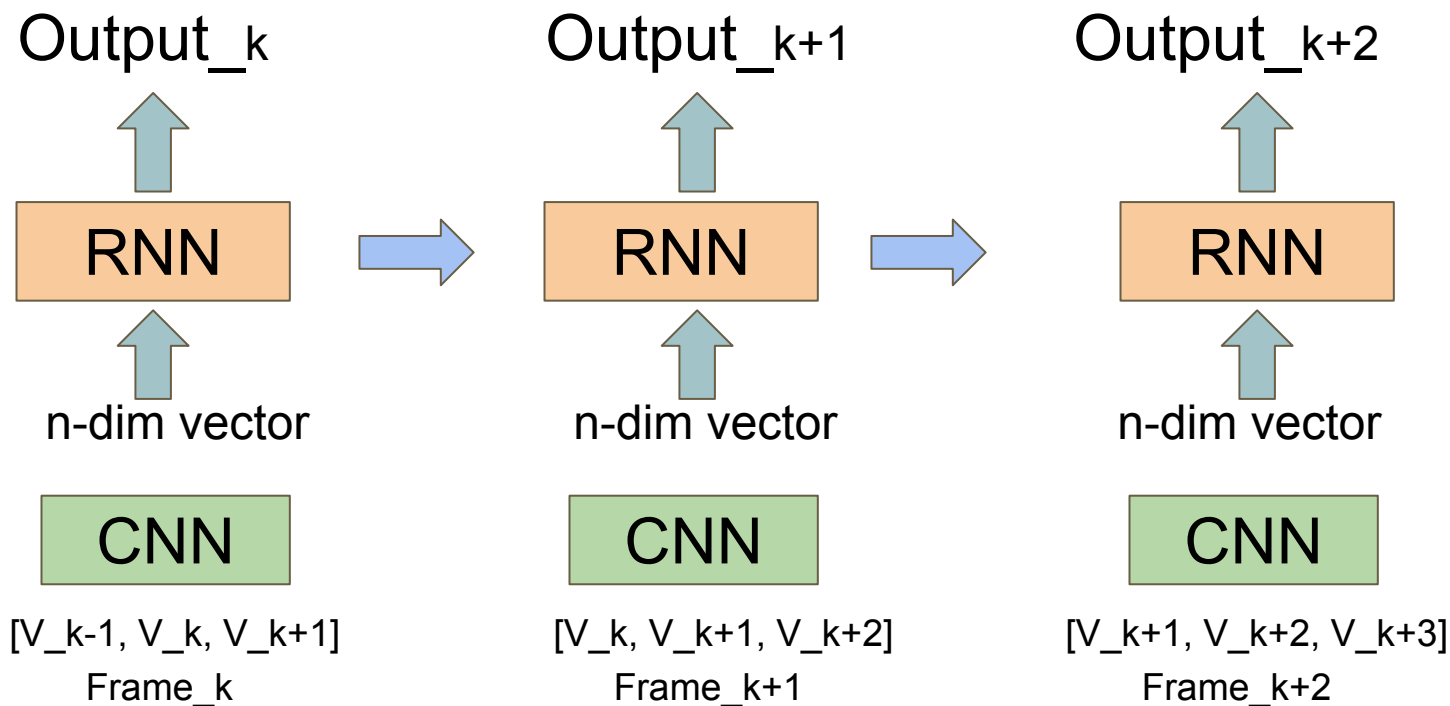
# Long Short-term Memory (LSTM)



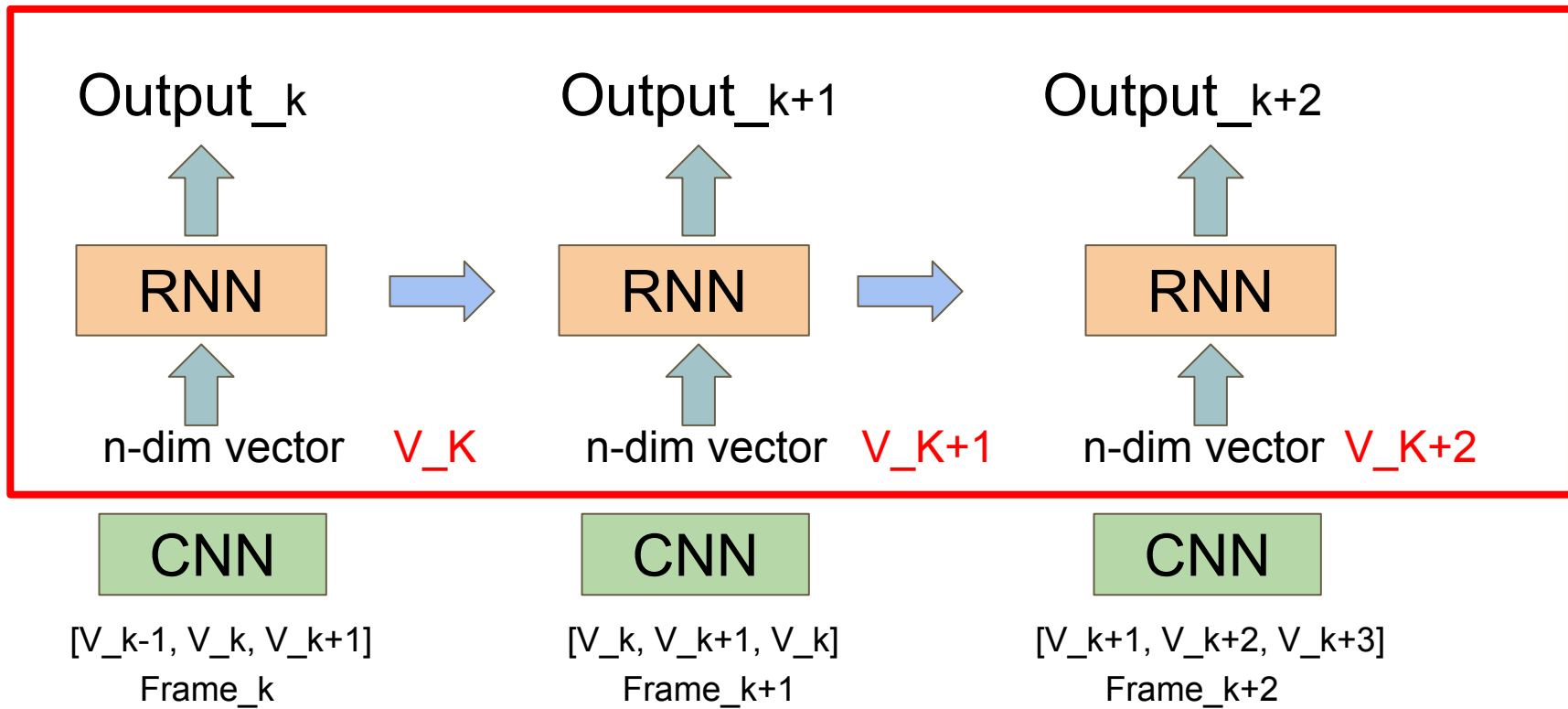
<http://colah.github.io/posts/2015-08-Understanding-LSTMs/>

# Convolutional Neural Network

# Jointly train RNN with CNN

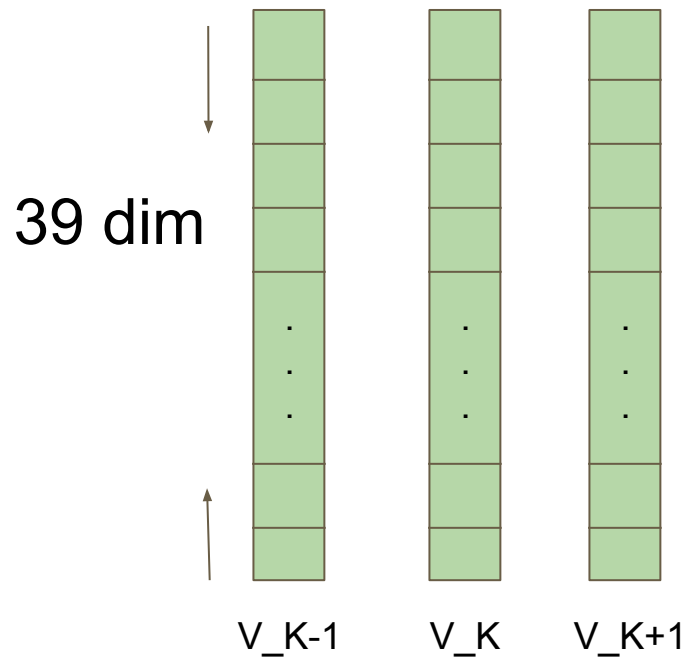


# Jointly train RNN with CNN



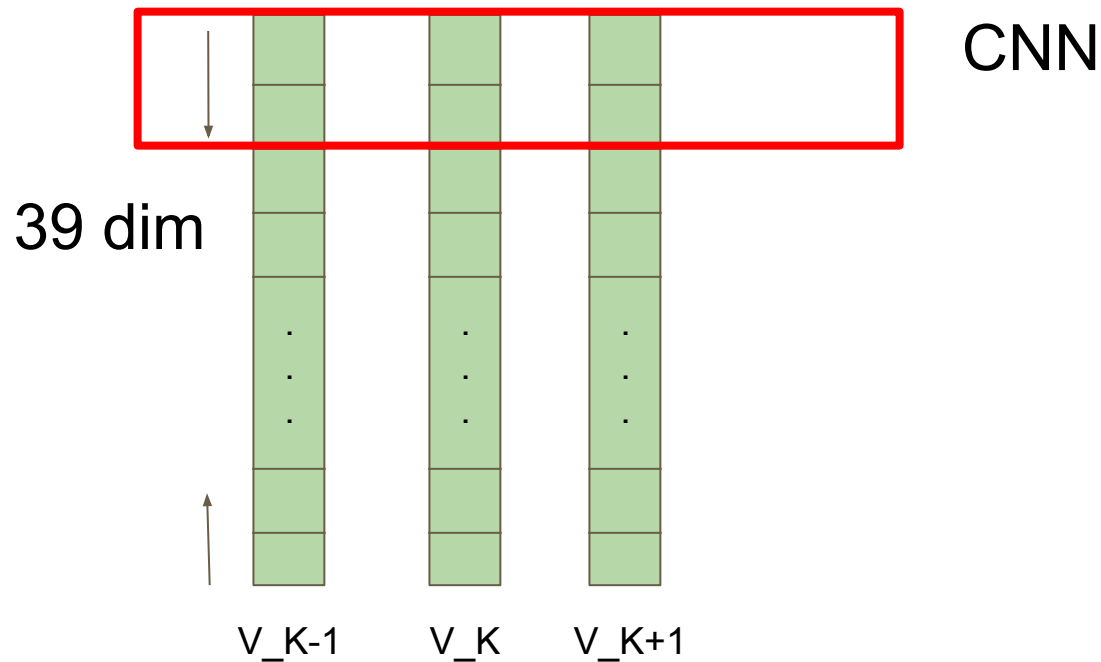
# CNN on acoustic features

Take feature MFCC for example:



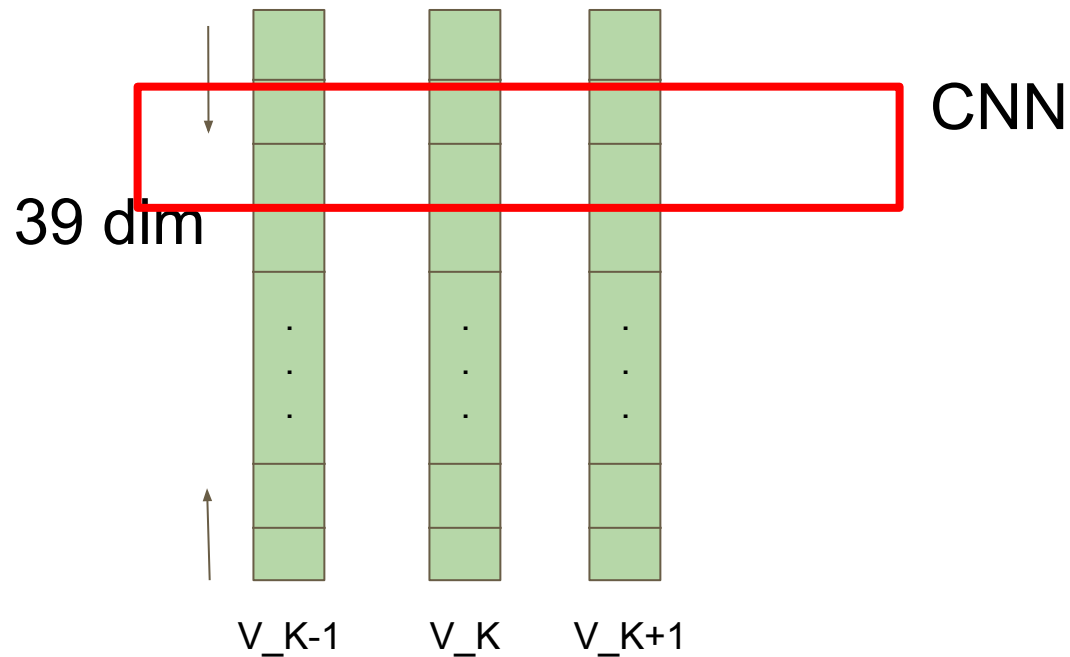
# CNN on acoustic features

Take feature MFCC for example:



# CNN on acoustic features

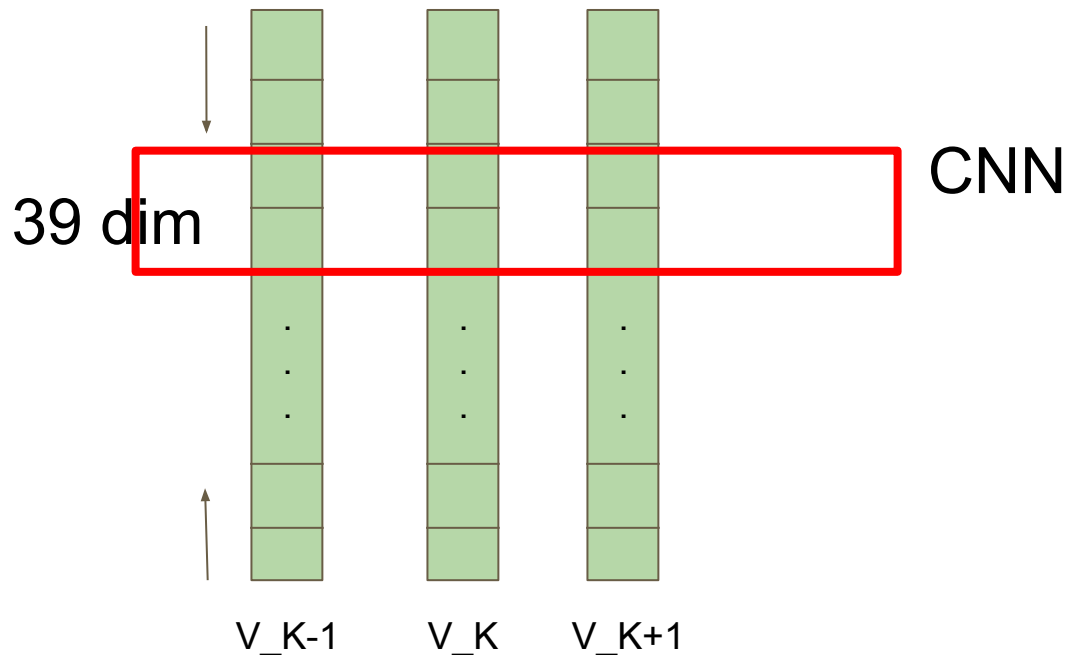
Take feature MFCC for example:





# CNN on acoustic features

Take feature MFCC for example:



Try different  
experiment settings  
and write down your  
observation in the  
report !

# CNN Lectures

- Machine Learning 2016 Fall

<https://www.youtube.com/watch?v=FrKWiRv254g>

- Tóth, László. "Convolutional Deep Maxout Networks for Phone Recognition", Interspeech, 2014

# HW Rules

# HW Rules

- Please write shell script to run your code.
- There should be `hw1_rnn.sh`, `hw1_cnn.sh`, `hw1_best.sh`
- Please follow the script usage below:
  - `./hw1_rnn.sh $1 $2`
  - `./hw1_cnn.sh $1 $2`
  - `./hw1_best.sh $1 $2`
  - \$1: the data directory, \$2: output filename
- Ex: `./hw1_best.sh myData/ best.csv`

# HW Rules

- Please implement RNN-based to fulfill the task
- Please also implement CNN+RNN-based to fulfill the task
- Please use python with version  $\geq 3.5$
- Please do not use extra dataset
- Allowed package includes:
  - PyTorch v0.2.0
  - tensorflow r1.3
  - Keras 2.0.7
  - MXNet 0.11.0
  - CNTK 2.2

# Kaggle

# Kaggle (1/3)

- Kaggle: <https://www.kaggle.com/t/0d61e84f89594f998b12d999fa4b4d5f>
- competition will started at **2017/10/5 12:00 (GMT+8)**.
- Please create **ONE** account using your school mail (Ex: NTU)
- For students taking this class, your title on leaderboard should start with **your student ID**
  - Ex: b03xxxxxx\_SamIsTheBest
- At most **5** submissions per day
- Individual task, do not team up!
- **No score counted if**
  - you create more accounts to get more submissions == cheating!
  - your title does not conform to the naming rules



# Kaggle (2/3)

- Testing set is divided into two sets: **public** and **private**
- Your performance on leaderboard during the competition is based on the **public set**
- After deadline, the **private set** will be evaluated
- Remember to choose **2 submissions** for the final evaluation before deadline, otherwise Kaggle will select for you
- Please do not attempt to fit the public set



# Kaggle (3/3)

- Submission format: a **.csv** file with then content as below
- Remember to map the framewise output to 39 phones
- Remember to map phones to English letter
- Remember to trim <sil>
- With header row: "id,phone\_sequence"
- Instance ID + , + predicted phone sequence

```
id,phone_sequence
fadg0_si1279,HrLAJarDeBLMrDcLMwU
fadg0_si1909,vbLAFKnLhyUwJmrBJLAWLSyLAWKr
fadg0_si649,lwLJctryJvrCaBgLHwDLKyDwLHJywDLHrLHwJnDryJLABLMtrBwLABsmrQI
fadg0_sx109,SJKyJBnLMwDJLHIyDyCrFLABSaDwDJLMYLAJBc
fadg0_sx19,vnBFDwDnDyUJFQsrLABwDatwDLhyJBcDLJwByD
fadg0_sx199,lynDyKnBLkrwDyKwmwLhaIymyLAJLhcIKrLMwLAWLksLzwJLAJwUyJwJ
```

# Grading

# Grading Policy

- I. Baseline (6%)
- II. Ranking (8%)
- III. Report (4%)
- IV. Bonus (2%).
- V. Notice

# Grading Policy -- Baseline&Ranking

- Pass the public baseline (3%)
- Pass the private baseline (3%)
- **Ranking (8%)** For those passing the private baseline, your score will be linearly grade, rounded to the 2nd decimal place
  - Ex: if 100 people pass the baseline, you will get 6 points if you're at 25th place.
- We will run your code to make sure your leaderboard performance is aligned with your submission

# Grading Policy -- Report(4%)

- Do not exceed **4** pages and **written in Chinese**
- Model description (2%)
  - RNN (1%)
  - RNN+CNN (1%)
- How to improve your performance (1%)
  - Write down the method that makes you outstanding
  - Describe the model or technique (0.5%)
  - Why do you use it (0.5%)
- Experimental results and settings (1%)
  - Compare and analyze the results between RNN and CNN (0.5%)
  - Compare and analyze the results with other models (0.5%)
    - other models can be variant of basic RNN, like LSTM, or some novel ideas you use

# Grading Policy -- Bonus(2%)

- TAs will select about 5 persons, according to both **creativity** and **performance** (top 10%) for introducing your model during the class
- If you are chosen, **you have to present** in order to get the bonus

# Grading Policy -- Notice

- 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
- You can still upload your result on Kaggle, although it won't be counted in your grade
- You get 0% 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

# Submission Rules



# Submission Rules

- Please refer to this [link](#) **first**.
- Create hw1 directory under ADLxMLDS2017
- Under hw1, there should be:
  - report.pdf
  - **your\_rnn\_model, your\_cnn\_model, your\_best\_model**
  - hw1\_rnn.sh // should run your RNN model
  - hw1\_cnn.sh // should run your CNN+RNN model
  - hw1\_best.sh // should run your best-performed model
  - model\_rnn.py, model\_cnn.py, model\_best.py and other necessary files
  - \*In model\_rnn.py, model\_cnn.py and model\_best.py should include your training codes.
- **Please do not upload TIMIT dataset to Github**
- If your model are too big for github, upload to a cloud space and **write it in your script to download the model**
- Your script should be done **within 10 mins (include preprocessing)** excluding model donwloading

# Deadline

1. Kaggle deadline: **2017/10/28 12:00 (GMT+8)**
2. Github code & report deadline: **2017/10/28 23:59 (GMT+8)**

# FAQ

# Q1: 使用的lib 限制

A:

除了拿來training的lib有限制以外，其他lib在使用的時候只要沒有使用外部的dataset都是可以的。並且記得在report中註明使用的lib名稱以及版本。

Ex: sklearn的train, test, split沒有用到助教的其他data, 所以可以使用。

## Q2: 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。

## Q3: Dataset在哪裡下載？

A:

Dataset可以從Kaggle上下載。

## Q4: 執行的時候助教要怎麼知道我是使用哪一種feature?

A:

在助教的電腦上, data directory結構如右所示。

而助教在測試的時候, 我們argv只會輸入"data/"。所以

同學必須要自己設定好你們需要的檔案路徑讓助教output

正確的答案。

```
data/  
----fbank/  
-----test.ark  
-----train.ark  
----label/  
-----train.lab  
----mfcc/  
-----test.ark  
-----train.ark  
----phones/  
-----48_39.map  
----48phone_char.map
```

**Q5:** Training label和feature的instance\_ID順序不一樣, 是要自己去對齊嗎?

A:

是的! 這部分要麻煩同學自己去對齊!



## Q6: 哪些檔案可以上傳到github呢？

A:

任何你們需要的檔案，只要這個檔案不是拿來作弊的 (Ex: 外部的dataset)，就可以上傳！

Ex: 自己建立的phone2phone, phone2idx這類的dictionary也是可以上傳的！

# Q7: 可以上傳前處理的data嗎？

A:

不行。前處理的時間包含在testing的10min之內。

## Q8: 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。

# FAQ

- If you have other questions,
  - please contact TAs via [adlxmlds@gmail.com](mailto:adlxmlds@gmail.com)
  - post your questions on [facebook group](#)
  - go to TA office hours
    - 王昱翔 Mon 16:00-17:30 (電二531)
    - 樊恩宇 Fri 10:30-12:00 (明達526)
    - 古志文 Fri 14:30-16:00 (德田524)