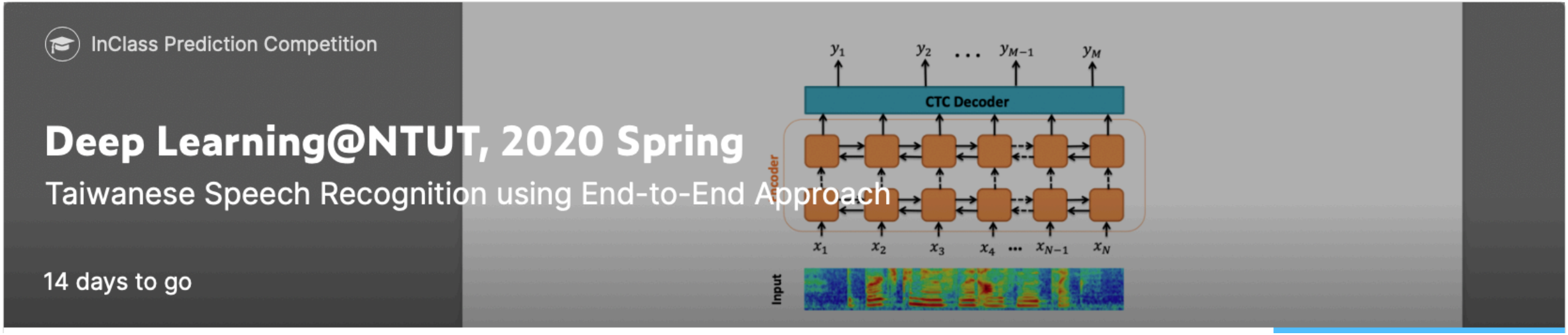
Kaggle inClass Competition - End-to-End Taiwanese Speech Recognition

Connectionist Temporal Classification

Yuan-Fu Liao

National Taipei University of Technology



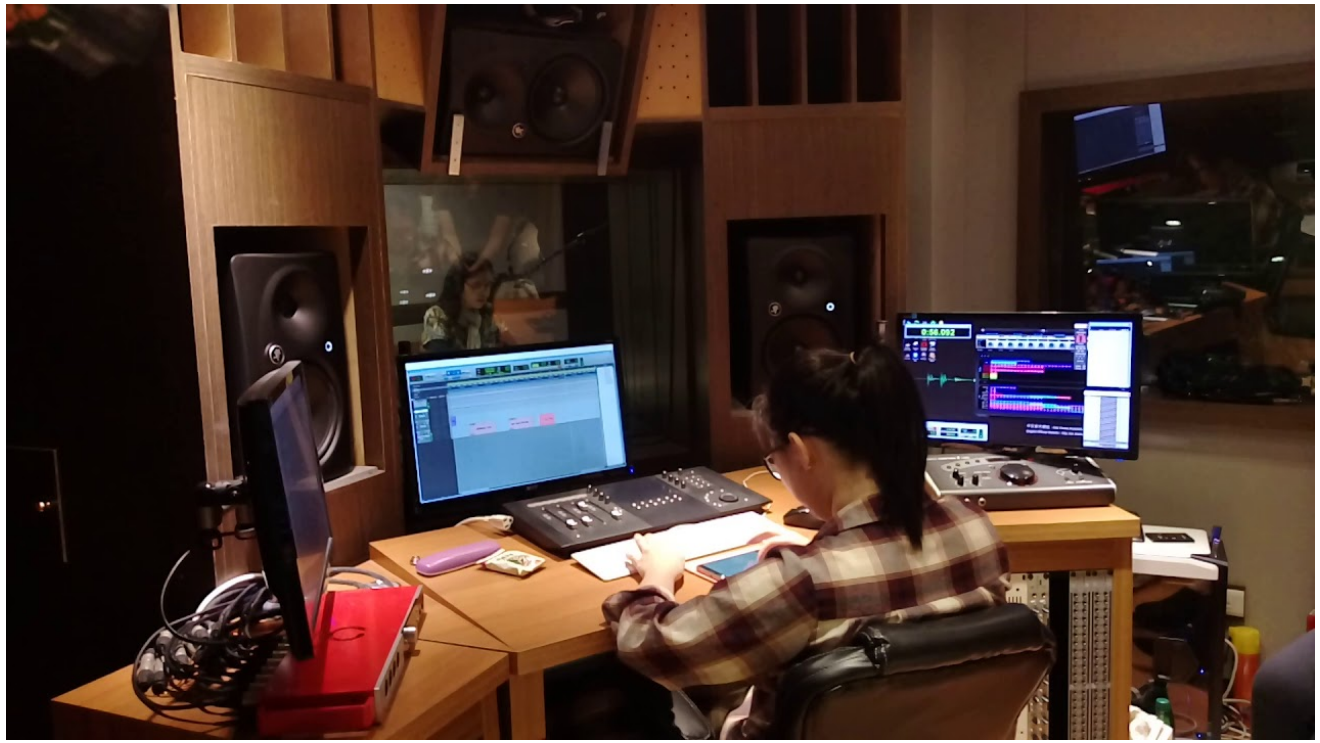
- End-to-End Taiwanese Speech Recognition using Connectionist Temporal Classification
  https://www.kaggle.com/t/7e182024e8ce4b1fb7fb067967c84a97

- Metric
  - Levenshtein Distance = Character error rate in %
- Word error rate (WER) = (D + S + I) / N × 100%
  - N - total number of labels （總詞數）
  - D - deletion errors （刪除錯誤）
  - S - substitution errors （替換錯誤）
  - I - Insertion errors （插入錯誤）

實驗資料

- 台灣媠聲 2.0
  - https://suisiann-dataset.ithuan.tw

https://youtu.be/GJvtWyuizyA



Data Format

- lexicon.txt
- train/*.wav
- Test/*.wav
- train.csv
- test.csv
- sample.csv

台羅拼音	子音	母音
Khau	kh	au
Kheh	kh	eh
Khe	kh	e
Khennh	kh	ennh
……		

```
id,text
1,li2 be7 e5 mih8 kiann7 lan5 lan5 san1 san1 long2 be7
tsiau5 tsng5
2,suah4 ka7 li2 tim3 tioh8
3,kiu3 lang5 ooh4
……
```

```
id,text
1,a1 e2 i3 o4 u5
2,a1 e2 i3 o4 u5
3,a1 e2 i3 o4 u5
……
```

Reference CTC Code

- Tensorflow CTC Speech Recognition
  - https://github.com/philipperemy/tensorflow-ctc-speech-recognition

  - Install

```
git clone https://github.com/philipperemy/tensorflow-ctc-speech-recognition.git ctc-speech

cd ctc-speech
pip3 install -r requirements.txt
```

  - Prepare data

```
wget https://www.dropbox.com/s/xecprghgwbbuk3m/vctk-pc225.tar.gz
tar xvzf vctk-pc225.tar.gz
python generate_audio_cache.py --audio_dir vctk-p225
```

  - training

```
python3 ctc_tensorflow_example.py
```

Reference Data Structure

- vctk-p225
  - txt
    - p225
      - p225_001.txt
      - p225_002.txt
      - …
  - wav48
    - p225
      - p225_001.wav
      - p225_002.wav
      - …

```
Please call Stella.
Ask her to bring these things with her from the store.
Six spoons of fresh snow peas, five thick slabs of blue cheese, and maybe a sna
We also need a small plastic snake and a big toy frog for the kids.
She can scoop these things into three red bags, and we will go meet her Wednesd
When the sunlight strikes raindrops in the air, they act as a prism and form a
The rainbow is a division of white light into many beautiful colors.
These take the shape of a long round arch, with its path high above, and its tw
There is , according to legend, a boiling pot of gold at one end.
```

```
Input File     : 'p225_001.wav'
Channels       : 1
Sample Rate    : 48000
Precision      : 16-bit
Duration       : 00:00:02.05 = 98473 samples ~ 153.864 CDDA sectors
File Size      : 197k
Bit Rate       : 768k
Sample Encoding: 16-bit Signed Integer PCM
```

- 請依樣畫葫蘆，換成台語語料！

Reference LSTM

目前只有一層LSTM ……

請嘗試不同網路架構！

看什麼樣的網路效果較好？

```
graph = tf.Graph()
with graph.as_default():
    inputs = tf.placeholder(tf.float32, [None, None, num_features], name='inputs')
    targets = tf.sparse_placeholder(tf.int32, name='targets')
    seq_len = tf.placeholder(tf.int32, [None], name='seq_len')

    cell = tf.contrib.rnn.LSTMCell(num_hidden, state_is_tuple=True)
    stack = tf.contrib.rnn.MultiRNNCell([cell], state_is_tuple=True)
    outputs, _ = tf.nn.dynamic_rnn(stack, inputs, seq_len, dtype=tf.float32)

    shape = tf.shape(inputs)
    batch_s, max_time_steps = shape[0], shape[1]

    outputs = tf.reshape(outputs, [-1, num_hidden])

    W = tf.Variable(tf.truncated_normal([num_hidden, num_classes], stddev=0.1))
    b = tf.Variable(tf.constant(0., shape=[num_classes]))

    logits = tf.matmul(outputs, W) + b
    logits = tf.reshape(logits, [batch_s, -1, num_classes])
    logits = tf.transpose(logits, (1, 0, 2))
    loss = tf.nn.ctc_loss(targets, logits, seq_len)
    cost = tf.reduce_mean(loss)

    optimizer = tf.train.AdamOptimizer(learning_rate=5e-4).minimize(cost)
    decoded, log_prob = tf.nn.ctc_greedy_decoder(logits, seq_len)
    ler = tf.reduce_mean(tf.edit_distance(tf.cast(decoded[0], tf.int32), targets))
```

注意

- 此參考程式沒有寫存檔，也沒有測試部分，請自己加上，例如：

```
saver = tf.train.Saver(max_to_keep=None)
if not os.path.exists(modelpath):
      os.mkdir(modelpath)
......
if ((curr_epoch+1)%1==0):
      save_path = saver.save(session, modelpath+'/'+modelname,
global_step=curr_epoch+1)
      print("save model to ", save_path)
```

- 此參考程式沒有寫測試部分，請自己加上，可以參考validation部分的解碼程式，例如：

```
val_feed = {inputs: val_inputs, targets: val_targets, seq_len: val_seq_len}
d = session.run(decoded[0], feed_dict=val_feed)
decode_batch(d, val_original, phase='validation')
```

Reference Experiments

- Experimental Settings
  - LSTM with 256 cells
  - one speaker: p225
  - test set: 15 shortest utterances
  - training set: rest utterances
  - batch size 16

Reference Experimental Results

```
Epoch 2723/3000, train_cost = 1.108, train_ler = 0
- Original (training) : but the commission is on a
- Decoded  (training) : but the commission is on a
- Original (training) : this action reflects a slu
- Decoded  (training) : this action reflects a slu
- Original (training) : they had to learn to work
- Decoded  (training) : they had to learn to work
- Original (training) : it depends on the internal
- Decoded  (training) : it depends on the internal
- Original (training) : irvine said his company wa
- Decoded  (training) : irvine said his company wa
- Original (training) : the pain was almost too mu
- Decoded  (training) : the pain was almost too mu
- Original (training) : this is a very common type
- Decoded  (training) : this is a very common type
- Original (training) : in fact he should never ha
- Decoded  (training) : in fact he should never ha
- Original (training) : saddam is not the only exa
- Decoded  (training) : saddam is nat the only exa
- Original (training) : so did she meet him
- Decoded  (training) : so did she meet him
- Original (validation) : it is a court case
- Decoded  (validation) : it is a cot ase
```