AN ANALYSIS OF INCORPORATING AN EXTERNAL LANGUAGE MODEL INTO A SEQUENCE - TO - SEQUENCE MODEL

Spring Semester Capstone Study

TEAM Kai.Lib

발표자 : 원철황

2020.04.08 (WED)

ABSTRACT

- This leads to the use of shallow fusion with an external language model at inference time
  - 추측의 시간에 외부 언어모델을 이용해 shallow fusion을 진행함

- Every timestep at Beam Search
  - 빔서치의 매 스텝에서 shallow fusion을 진행하여 prediction을 수행함
  - 우리 코드에 적용하려면 beamsearch 코드 역시 수정해야 할 필요성 대두

ABSTRACT

Attention-based sequence-to-sequence models for automatic speech recognition jointly train an acoustic model, language model, and alignment mechanism. Thus, the language model component is only trained on transcribed audio-text pairs. This leads to the use of *shallow fusion* with an external language model at inference time. Shallow fusion refers to log-linear interpolation with a separately trained language model at each step of the beam search. In this work, we investigate the behavior of shallow fusion across a range of conditions: different types of language models, different decoding units, and different tasks. On Google Voice Search, we demonstrate that the use of shallow fusion with an neural LM with wordpieces yields a 9.1% relative word error rate reduction (WERR) over our competitive attention-based sequence-to-sequence model, obviating the need for second-pass rescoring.

본 논문에서 수행한 바는 세 가지이다.

(1) Different types of language models
: n-gram vs RNNs
(2) Different decoding units
: Graphemes vs Wordpiece
(3) Different tasks
: WSJ vs Google Voice Search

1. INTRODUCTION

- LAS model (Listen, Attend and Spell)
  - Encoder : Acoustic Model in ASR model
  - Decoder : Language Model in Natural Language Processing
  - Attention mechanism : For Alignment

1. INTRODUCTION

We propose that one reason for the performance degradation could be that the LAS decoder, which replaces the LM component in a traditional ASR system, is trained only on transcribed audio-text pairs, which is about 15 million utterances for the Google Voice Search task [5]. In comparison, state-of-the-art LMs are typically trained on a billion words or more [6]. This raises the question of whether the LAS decoder can learn a strong enough LM from the training transcripts. In particular, we posit that in a task like Google Voice Search, which has a very long tail of queries, the training transcripts may not sufficiently expose the LAS decoder to rare words and phrases.

- 본 논문에서는 기존 ASR 모델에 대한 LAS 모델의 성능 개선을 위한 방법을 제안
- 해당 논문은 해당 모델의 성능 저하는 LAS 모델의 decoder 라 제안함
- 원인은 LM 학습 요소가 오직 audio-text pair에 불과하기 때문
- State-of-the-art LMs : Use a billion words

1. INTRODUCTION

- Use LMs with Attention-based models (n-gram)
  - [1]: LAS에 의해 생긴 n-best 추측에 n-gram LMs Model 사용
  - [2]: 위의 개념을 Beamsearch decoding step으로 확장한 논문
  - [7]: 해당 논문에서 Shallow Fusion을 제안
  - [8]: Coverage Penalty 라는 개념을 써서 Shallow Fusion을 보완

    이들은 모두 LM과의 결합으로 성능을 높이기 위해 작은 Domain의 WSJ를 사용하였다. 이때 사용된 모든 LMs는 n-gram 이었으며, Bidirectional 이 적용되었다.

- Use LMs with Attention-based models (RNNs)
  - [7, 9, 10]: 언제 어떻게 외부 LM을 '학습' 에 사용하는가에 대한 방법을 거론함.
  - RNN-LM 적용

- Summary
  - 이들 중 누구도 RNNs 과 n-gram 모델을 사용했을 때 성능을 비교하지 않음.
  - 본 논문에서는 성능 개선을 확인하기 위해 WSJ 이라는 비교적 작은 corpus 이용해 모델간 성능 비교
  - 본 논문에서는 Domain, Decoding Unit과 더불어 어떤 LM의 성능이 좋은지를 확인함
  - 또한 이를 Google Voice Research로 넓혀 실험을 진행함

1. INTRODUCTION

- First Goal of this work
  - Small Corpus : WSJ
  - Different sub-word type : Graphemes or Wordpieces
  - Different type of LMs : n-gram or RNNs
  - 위의 조건들로 Shallow Fusion을 진행해 [8] 번 논문의 연구를 확장함

- Second Goal of this work
  - Large Corpus and Vocabulary : English Voice Searc
  - Different sub-word type : Graphemes or Wordpieces
  - Different type of LMs : n-gram or RNNs
  - 위의 조건들로 Shallow Fusion을 진행하고 기존 Base Line에 대한 성능 확인

2. SHALLOW FUSION WITH LAS MODELS

2.1 Listen, attend, and spell

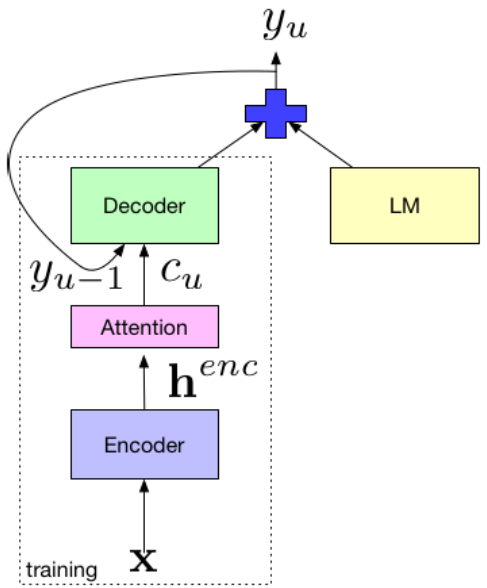


Fig. 1: The dotted line box shows the basic LAS model, including an encoder, attention, and decoder. In shallow fusion, an external LM is incorporated via log-linear interpolation.

문제

- Encoder가 Decoder에 Feed 하는 구조
- 이 때문에 내부 LM은 오직 audio-text PAIR 에만 영향

해결

- 학습하는 Text 양을 방대하게 증가
- 외부 LM을 사용

2. SHALLOW FUSION WITH LAS MODELS

2.2 Integrating a language model

Typical Decoding in seq2seq model

$$\mathbf{y}^* = \arg\max_y \log p(y|x)$$

Shallow Fusion criterion

$$\mathbf{y}^* = \arg\max_y \log p(y|x) + \lambda \log p_{LM}(y) + \gamma c(x, y)$$

$\lambda \; and \; \gamma$

- Tuened by dev set

$\mathrm{c}(\mathrm{x}, \mathrm{y}) = \sum_{\mathrm{j}} \log(\min(\sum_{\mathrm{i}} \alpha_{\mathrm{ij}}, 0.5))$

- Coverage Penalty
- Input frames 가 attention weigh에 얼마나 cover 되었는지에 대한 정도 측정
- Output y의 Attention 수가 많을 수록 비교적 높은 확률 값을 얻음
- “Babble” 현상 막아줌

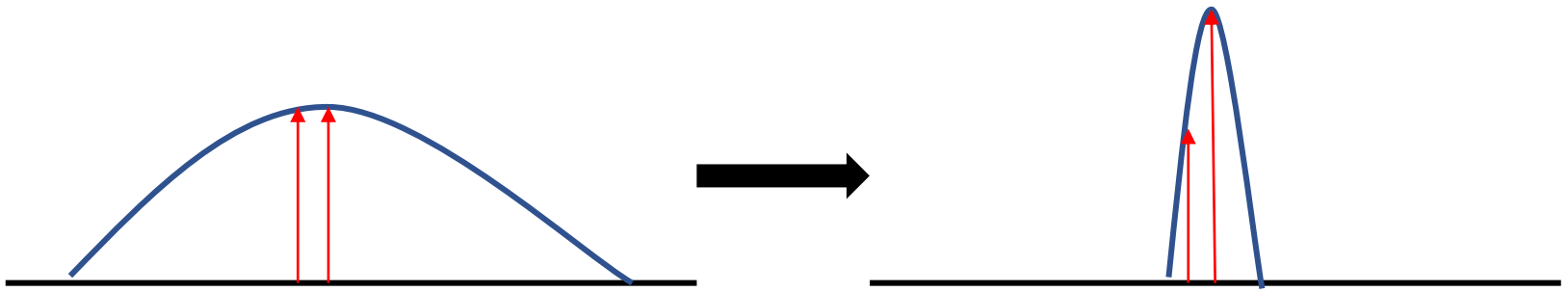
※ Toward better decoding and language model integration in seq2seq models

Towards better decoding and language model integration in sequence to sequence models

Jan Chorowski, Navdeep Jaitly

Google Brain
Google Inc.
Mountain View, CA 94043, USA
jan.chorowski@cs.uni.wroc.pl,ndjaitly@google.com

- seq2seq discriminative training (using attention mechanism)
  - Overfitting Probability
  - High Confidence of Models with Low Loss
  - Sharp Distribution caused by High Confidence reduce Beam Search Diversity
  - 동일 조건이라도 각 beam의 추론 값의 현저한 차이가 발생함



※ Toward better decoding and language model integration in seq2seq models

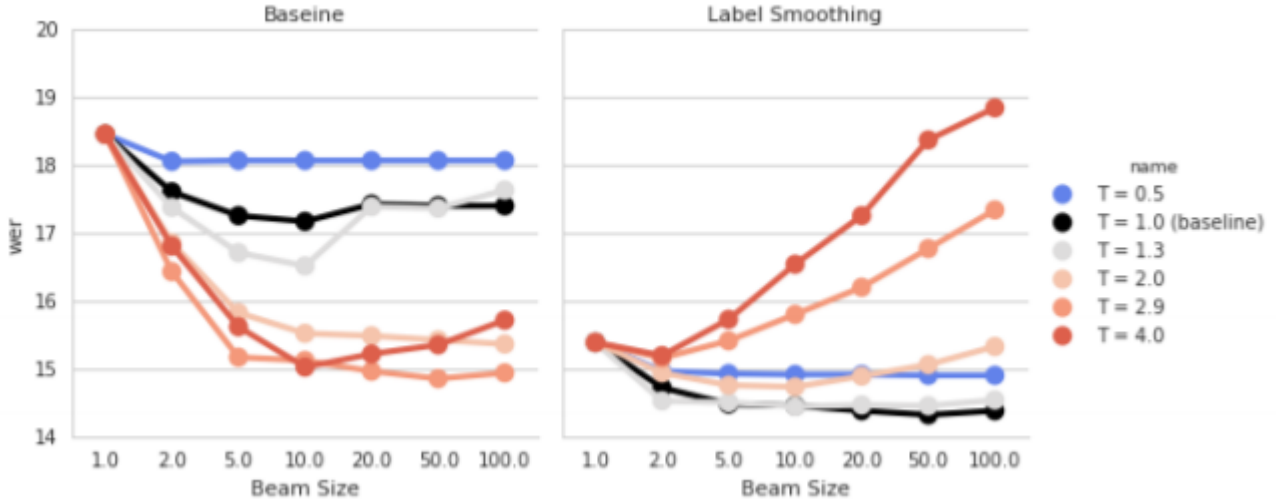
3.1. Impact of Model Overconfidence

Model confidence is promoted by the the cross-entropy training criterion. For the baseline network the training loss (7) is minimized when the model concentrates all of its output distribution on the correct ground-truth character. This leads to very peaked probability distributions, effectively preventing the model from indicating sensible alternatives to a given character, such as its homophones. Moreover, overconfidence can harm learning the deeper layers of the network. The derivative of the loss backpropagated through the SoftMax function to the logit corresponding to character c equals $[\boldsymbol{y}_i = c] - p(\boldsymbol{y}_i|\boldsymbol{y}_{<i}, \boldsymbol{x})$, which approaches 0 as the network's output becomes concentrated on the correct character. Therefore whenever the spelling RNN makes a good prediction, very little training signal is propagated through the attention mechanism to the listener.

Model overconfidence can have two consequences. First, next-step character predictions may have low accuracy due to overfitting. Second, overconfidence may impact the ability of beam search to find good solutions and to recover from errors.

We first investigate the impact of confidence on beam search by varying the temperature of the SoftMax function. Without retraining the model, we change the character probability distribution to depend on a *temperature* hyperparameter T:

$$p(y_i) = \frac{\exp(l_i/T)}{\sum_j \exp(l_j/T)}. \qquad (10)$$



- Temperature가 높아질 수록 확률 분포 값이 Uniform 해짐
- Preventing High Confidence Model
- Baseline Model에서 Temperature가 높아질 수록 WER 감소
- 반면 Label Smoothing 을 진행한 모델은 Temperature의 증가가 영향을 끼치지 않음
- 오히려 Beam = 1 일 때, WER이 내려간 것으로 보아 성능 개선 효과
- 즉, Label Smoothing을 통해 Overconfidence, WER 개선

※ Toward better decoding and language model integration in seq2seq models

seq2seq failure mode : incomplete transcript

Table 1: *Example of model failure on validation '4k0c030n'*

Transcript	LM cost $\log p(y)$	Model cost $\log p(y\|x)$
"chase is nigeria's registrar and the society is an independent organization hired to count votes"	-108.5	-34.5
"in the society is an independent organization hired to count votes"	-64.6	-19.9
"chase is nigeria's registrar"	-40.6	-31.2
"chase's nature is register"	-37.8	-20.3
""	-3.5	-12.5

"완전하지 않은 문장에서 더 낮은 Loss 계산"

Coverage Penalty 를 사용해 방지

3. EXPLORING SHALLOW FUSION

3.1. Tasks: WSJ vs. Google Voice Search

This work investigates the impact of shallow fusion on two different tasks. This is because we hypothesize that there are several task-specific properties that can affect the relative gain afforded by an external LM:

- *Size of training corpus*, because on a large training corpus the LAS decoder will itself be a very strong LM.
- *Size of vocabulary*, as some of the benefit of an external LM may simply be exposure to unseen words and phrases.
- *Availability of LM training data*, since the LM training data must come from the same domain as the task

- Size of corpus
  - Large training corpus를 가지고 있다면 LAS decoder 자체가 좋은 성능의 LM 역할이 가능함
- Size of Vocabulary
  - 외부 unseen word에 대한 경험이 정확도와 유연한 문장 생성
- Availability
  - LM의 데이터는 발화 주제에 관한 Domain으로부터 오기 때문에 연관성이 높고 유용성이 높음

3. EXPLORING SHALLOW FUSION

Like graphemes, wordpieces have the advantage that there are no out-of-vocabulary terms because any word can be decomposed into wordpieces. (All graphemes are included in the wordpiece vocabulary.) But wordpieces have the additional benefit that they effectively capture more context per decoding step than graphemes. This reduces the length of dependencies that must be learned by an LM.

For example, the phrase "the company announced today" consists of 27 graphemes, which means that a grapheme-level LM (LM-G) would require 27 decoding steps to output the full phrase; but a wordpiece-level LM (LM-WP) might compose this phrase as, for example `the _com pany _announc ed _today` which would require only 5 steps to output.

As a result, we expect that LM-WP can achieve lower (word-level) perplexity than LM-G, which could make it more effective in shallow fusion.

Wordpieces

- 통계기반
- **Perplexity** 감소

VS

Graphemes

- 알파벳 + 띄어쓰기
- **Decoding Step** 증가

3. EXPLORING SHALLOW FUSION

before but which are spelled phonetically. Though the techniques of Bayesian interpolation and incorporating dictionary constraints currently apply only the n-gram models, we posit that analogous methods should be possible for RNN LMs, and identify these as areas for future work.

“RNN LMs is Future”

N-gram Bayesian interpolation, Dictionary constraints 등의 기술을 적용할 수 있어도 **RNN LM**을 사용

5. RESULT

5.1. Comparing LMs for shallow fusion

We begin by comparing three types of LMs in the context of shallow fusion with the LAS grapheme model LAS-G on the WSJ task: (1) an RNN LM trained on graphemes (RNN-G), (2) a 20-gram LM trained on graphemes (20-GRAM-G), and (3) a 3-gram LM trained on words and composed with a speller (3-GRAM-W).

Comparing these, we see that 3-GRAM-W barely outperforms 20-GRAM-G. This shows that, given the same amount of context, having word constraints and an implicit dictionary has only a slight benefit. RNN-G, however, outperforms both of the n-gram LMs, suggesting while the word constraints may help, they are insufficient to make up the gap between RNN LMs and n-gram LMs. One opportunity for future work would be incorporating word constraints into RNN-G.

System	Dev	Test
LAS-G	13.0	10.3
LAS-G + 20-GRAM-G	10.3	7.7
LAS-G + 3-GRAM-W	10.0	7.6
LAS-G + RNN-G	9.3	6.9

Table 1: WER of LAS-G fused with various LMs. While word constraints do help the n-gram LM, RNN-G performs even better.

LAS model with Graphems integrated with RNN with Graphems shows best performance

5. RESULT

5.2. Extending shallow fusion to wordpiece models

Next, we perform a comparison for LAS-WP. Since we have shown that word constraints are helpful for sub-word-level n-gram LMs, we limit our comparison to just two LMs: (1) an RNN LM trained on wordpieces (RNN-WP), and (2) a 3-gram LM trained on words and composed with a speller (3-GRAM-W).

As Table 2 shows, we see the same trend on LAS-WP, with RNN-WP significantly better than 3-GRAM-W. However, it should be noted that the baseline LAS-WP is worse than LAS-G. This is likely due to the small amount of data being insufficient to train the large number of additional parameters: we found that the larger we made the wordpiece vocabulary, the worse the model became. As a result of this difference, the LM results for LAS-WP are not directly comparable to the LM results for LAS-G. The main observation we make is that the RNN performs best in both cases, with the relative improvement being roughly consistent for both graphemes and wordpieces.

System	Dev	Test
LAS-WP	15.7	12.3
LAS-WP + 3-GRAM-W	12.9	9.3
LAS-WP + RNN-WP	11.5	8.2

Table 2: WER of LAS-WP combined with various LMs on WSJ. RNN-WP again performs best.

LAS model with Wordpieces integrated with RNN with Wordpieces shows best performance

5. RESULT

5.3. Scaling up to Voice Search

We now turn to the Voice Search task. First, since we have an abundance of training data, we see in the first two lines of Table 3 that the wordpiece model (LAS-WP) is now comparable with the grapheme model (LAS-G). Thus our analysis here is limited to LAS-WP.

Thus, as with WSJ, we see that RNN-WP more effectively encodes the LM information compared to the n-gram model. In addition, RNN-WP is 1.5% the size of PRODLM2, and also enjoys the additional benefit of not having out-of-vocabulary words since it is trained on wordpieces. Note that both PRODLM1 and PRODLM2 are interpolated across several data-source-specific LMs, while RNN-WP uses ad hoc mixing weights for the various data sources. Investigating a more principled method of mixing the data sources for RNN-WP is an opportunity for future work.

System	Dev	Test	LM size
LAS-G	9.5	7.7	0GB
LAS-WP	9.2	7.7	0GB
LAS-WP + PRODLM1	8.8	7.4	2GB
LAS-WP + PRODLM2	8.7	7.2	80GB
LAS-WP + RNN-WP	8.4	7.0	1.1GB
LAS-WP + RNN-WP + PRODLM2	8.4	7.0	81.1GB

Table 3: WER of shallow fusion of LAS with production n-gram LMs and an RNN LM. The RNN LM captures all the benefits of PRODLM2 in a compact form.

LAS model with Wordpieces in Large Scale integrated with RNN with Wordpieces shows best performance

[1] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, "Listen, attend and spell," *CoRR*, vol. abs/1508.01211, 2015.

[2] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-End Attention-based Large Vocabulary Speech Recognition," in *Proc. ICASSP*, 2016.

[3] A. Graves, "Sequence transduction with recurrent neural networks," *CoRR*, vol. abs/1211.3711, 2012.

[4] A. Graves, S. Fernandez, F. Gomez, and J. Schmidhuber, "Connectionist Temporal Classification: Labeling Unsegmented Seuqnece Data with Recurrent Neural Networks," in *Proc. ICML*, 2006.

[5] R. Prabhavalkar, K. Rao, B. Li, L. Johnson, and N. Jaitly, "A Comparison of Sequence-to-sequence Models for Speech Recognition," in *Proc. Interspeech*, 2017.

[6] R. Jozefowicz, O. Vinyals, M. Schuster, N. Shazeer, and Y. Wu, "Exploring the limits of language modeling," *CoRR*, vol. abs/1602.02410, 2016.

[7] C. Gulcehre, O. Firat, K. Xu, K. Cho, L. Barrault, H. Lin, F. Bougares, H. Schwenk, and Y.Bengio, "On using monolingual corpora in neural machine translation," *CoRR*, vol. abs/1503.03535, 2015.

[8] J. K. Chorowski and N. Jaitly, "Towards Better Decoding and Language Model Integration in Sequence to Sequence Models," in *Proc. Interspeech*, 2017.

[9] T. Hori, S. Watanabe, Y. Zhang, and W. Chan, "Advances in joint ctc-attention based end-to-end speech recognition with a deep cnn encoder and rnn-lm," 2017.

[10] A. Sriram, H. Jun, S. Satheesh, and A. Coates, "Cold fusion: Training seq2seq models together with language models," *CoRR*, vol. abs/1708.06426, 2017.

[11] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov 1997.

[12] Z. Tu, Z. Lu, Y. Liu, X. Liu, and H. Li, "Modeling coverage for neural machine translation," *CoRR*, vol. abs/1601.04811, 2016.

[13] Y. Wu, M. Schuster, and et. al., "Google's neural machine translation system: Bridging the gap between human and machine translation," *CoRR*, vol. abs/1609.08144, 2016.

[14] M. Schuster and K. Nakajima, "Japanese and Korean voice search," *2012 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2012.

[15] Improving neural machine translation models with monolingual data, "R. sennrich and b. haddow and a. birch," *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2015.

[16] T.Mikolov, M. Karafiat, L. Burget, J. Cernocky, and S. Khudanbur, "Recurrent neural network based language model," *Proc. Interspeech*, 2010.

[17] M. Mohri, F. Pereira, and M. Riley, "Weighted finite state transducers in speech recognition," vol. 16, pp. 69–88, 2002.

[18] C. Allauzen, M. Riley, J. Schalkwyk, W. Skut, and M. Mohri, "Openfst: A general and efficient weighted finite-state transducer library," pp. 11–23, 2007.

[19] C. Allauzen and M. Riley, "Bayesian language model interpolation for mobile speech input," *Proc. Interspeech*, 2011.

[20] M. Schuster and K. K. Paliwal, "Bidirectional LSTM Networks for Improved Phoneme Classification and Recognition," *Artificial Neural Networks: Formal Models and Their Applications-ICANN*, pp. 799–804, 2005.

[21] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, J. Silovsky P. Schwarz, G. Stemmer, , and K. Vesely, "The kaldi speech recognition toolkit," 2011.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," *CoRR*, vol. abs/1706.03762, 2017.

[23] H. Sak, A. Senior, K. Rao, and F. Beaufays, "Fast and Accurate Recurrent Neural Network Acoustic Models for Speech Recognition," in *Proc. Interspeech*, 2015.

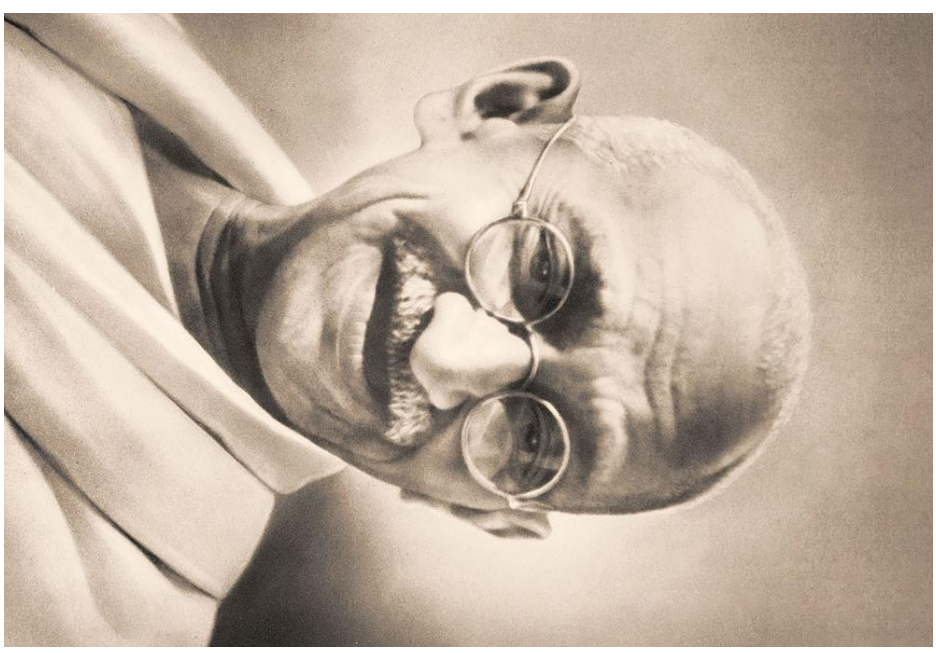
[24] G. Pundak and T. N. Sainath, "Lower Frame Rate Neural Network Acoustic Models," in *Proc. Interspeech*, 2016.

[25] M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," Available online: http://download.tensorflow.org/paper/whitepaper2015.pdf, 2015.

[26] K. Rao, R. Prabhavalkar, and H. Sak, "Exploring Architectures, Data and Units for Streaming End-to-End Speech Recognition with RNN-Transducer," in *Proc. ASRU*, 2017.

8. REFERENCES

참고문헌 목록



이 분은 마하트마 간디