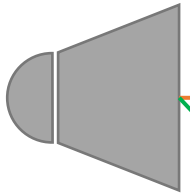
CONFIDENCE ESTIMATION FOR ATTENTION-BASED SEQUENCE-TO-SEQUENCE MODELS FOR SPEECH RECOGNITION

Qiujia Li[1*], *David Qiu*[2], *Yu Zhang*[2], *Bo Li*[2], *Yanzhang He*[2],
Philip C. Woodland[1], *Liangliang Cao*[2], *Trevor Strohman*[2]

[1] University of Cambridge, UK, [2] Google LLC, USA

[1]{ql264,pcw}@eng.cam.ac.uk, [2]{qdavid,ngyuzh,boboli,yanzhanghe,llcao,strohman}@google.com

[1. 소 개]



[ASR system]

Markob model

lattice decoding을 할 때 word posteriors로부터 confidence score를 얻을 수 있음

attention-based seq2seq

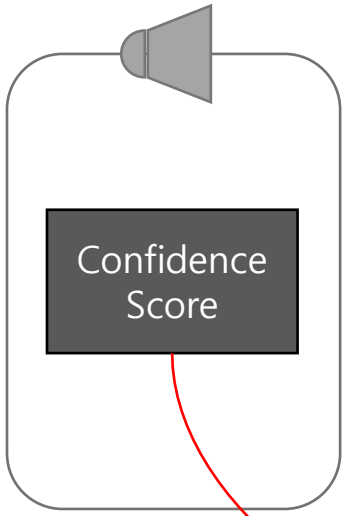
직접 word posteriors를 계산하기 힘들어
대안책으로 decoder softmax probability를 model confidence로 사용

* 일반적으로 사용되는 정규화 방법이 softmax 기반 confidence score에 어떤 영향을 미치는가

* end-to-end model의 overconfident behaviour 소개

* CEM 소개

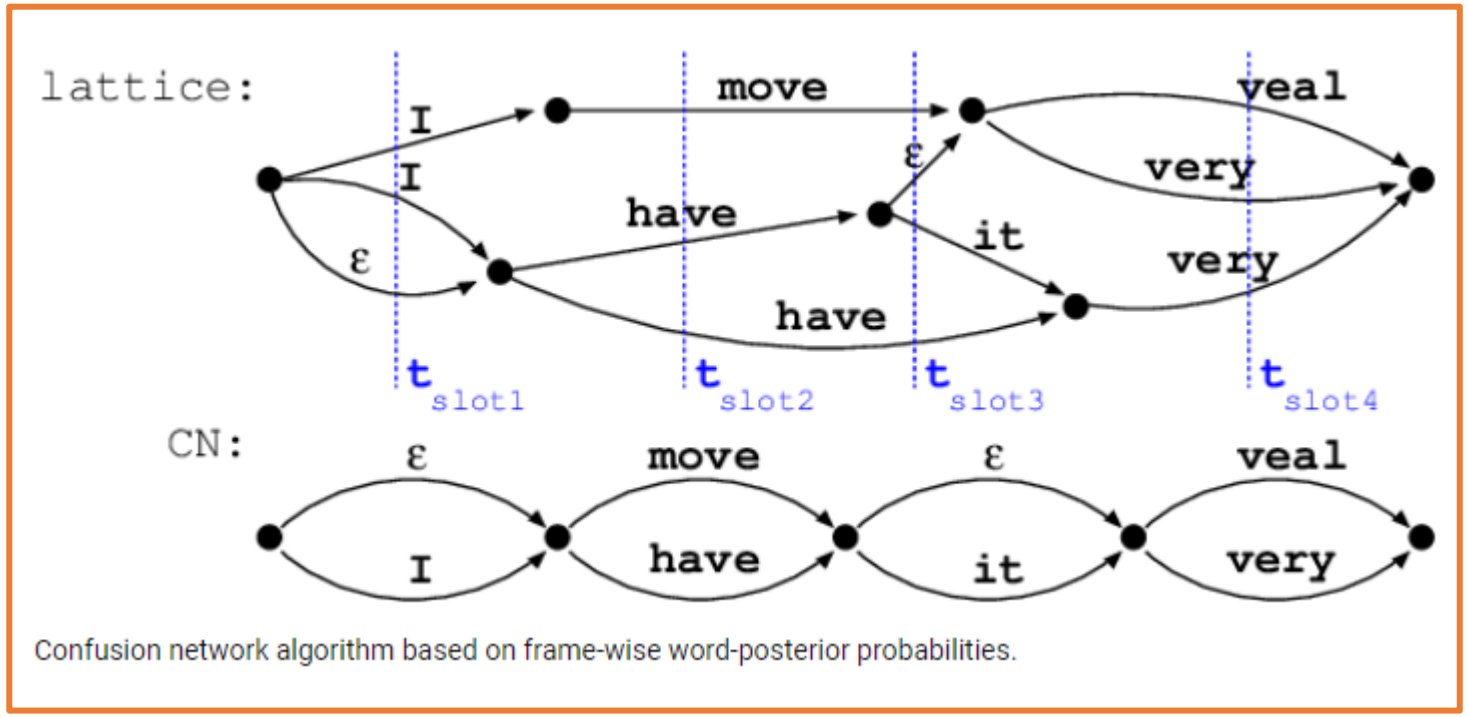
[2. 서 론]



e.g) semi-supervised, active learning 과정에서
높은 confident hypothesis 공간을 가진 utteranc가 ASR performance 향상을 위해 선택됨

e.g) 인공지능 대화 시스템에서 “잘 듣지 못했어요“, “이해하지 못했어요“ 등의 반응을 하도록 작동함

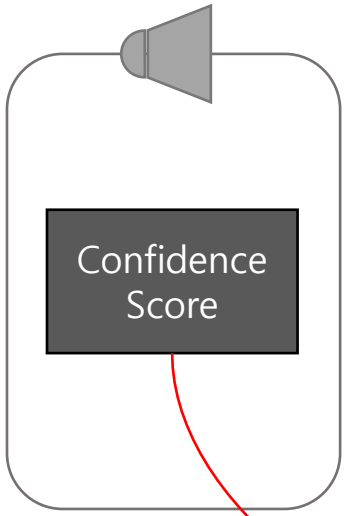
HMM-based System



Confusion network algorithm based on frame-wise word-posterior probabilities.

hypothesis space의 간결한 표현인 lattice/Confusion Network로부터 word posterior probability를 계산함으로써 얻을 수 있다

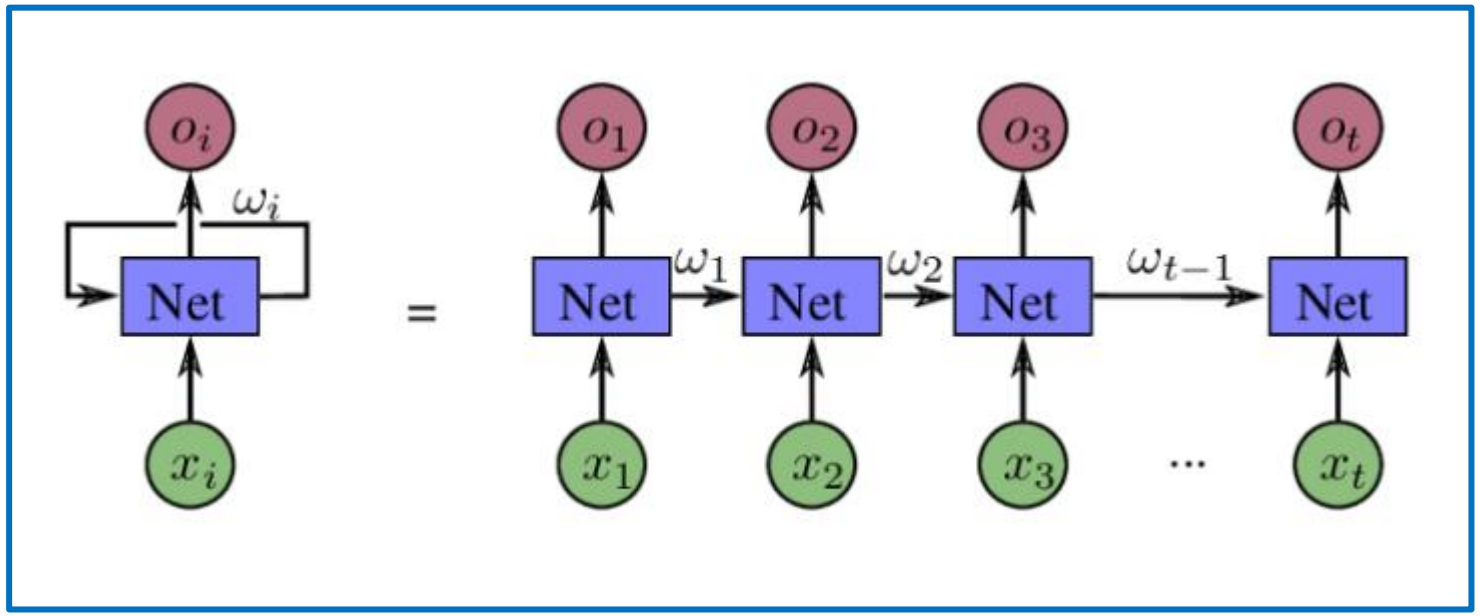
[2. 서 론]



e.g) semi-supervised, active learning 과정에서
높은 confident hypothesis 공간을 가진 utteranc가 ASR performance 향상을 위해 선택됨

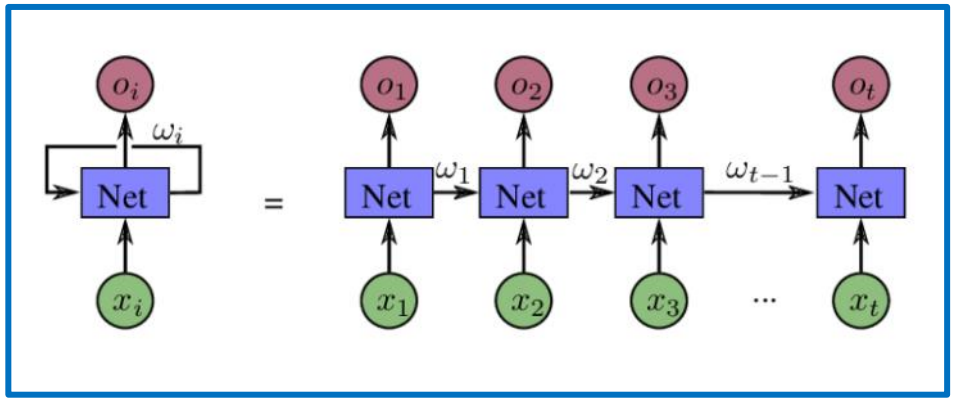
e.g) 인공지능 대화 시스템에서 “잘 듣지 못했어요“, “이해하지 못했어요“ 등의 반응을 하도록 작동함

attention-based seq2seq using auto-regressive-decoder



성능은 좋지만, 특정 decoder 구조, 또는 heuristic한 방법을 사용하지 않으면 word posterior probability를 계산하기 어렵다 (비용 多)

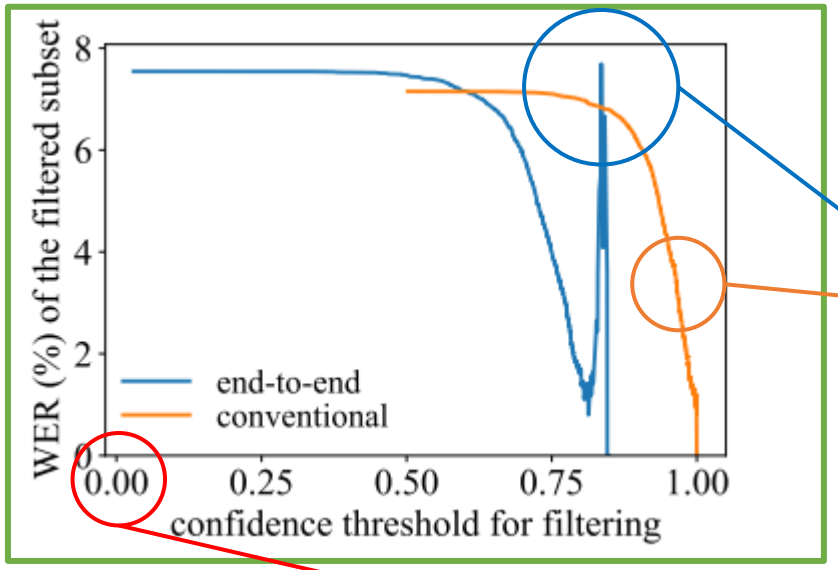
[2. 서 론]



[Greedy한 접근법]

decoder의 각 토큰에 대해서 decode의 각 step마다 softmax probability를 취하면 confidence score를 얻을 수 있지 않을까?

[2. 서 론]



x 축 : confidence score threshold (이 threshol을 넘는 score를 가진 utterance가 선택됨)

y 축 : Word Error Rate of filtered subset

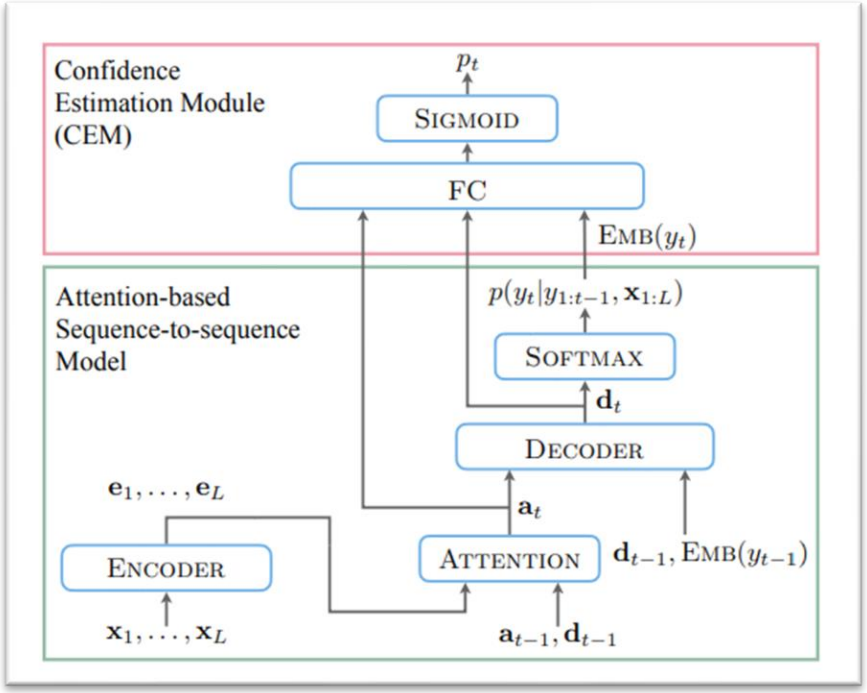
HMM의 경우 threshold값이 증가할수록 WER이 단조롭게 감소하는 모습

threshold=0, 즉 모든 utterance가 통과될 경우 HMM과 attention-based seq2seq의 성능은 비슷

threshold값의 상승이 항상 WER의 감소로 이어지지 않는다!

⇒즉, softmax probabilities는 overconfident 하다!

[3. C E M – 용어 정리]



$x_{1:L}$: Input Utterance

$e_{1:L} = Encoder(x_{1:L})$: Encoder의 Output

$a_t = Attention(a_{t-1}, d_{t-1}, e_{1:L})$: t번째 decoding step의 attention 값

$d_t = Decoder(a_t, d_{t-1}, EMB_{(y_{t-1})})$: t번째 decoding step의 Decoder output

y : label token (grapheme(자소) / word piece)

$EMB_{(y_t)}$: current label token embedding

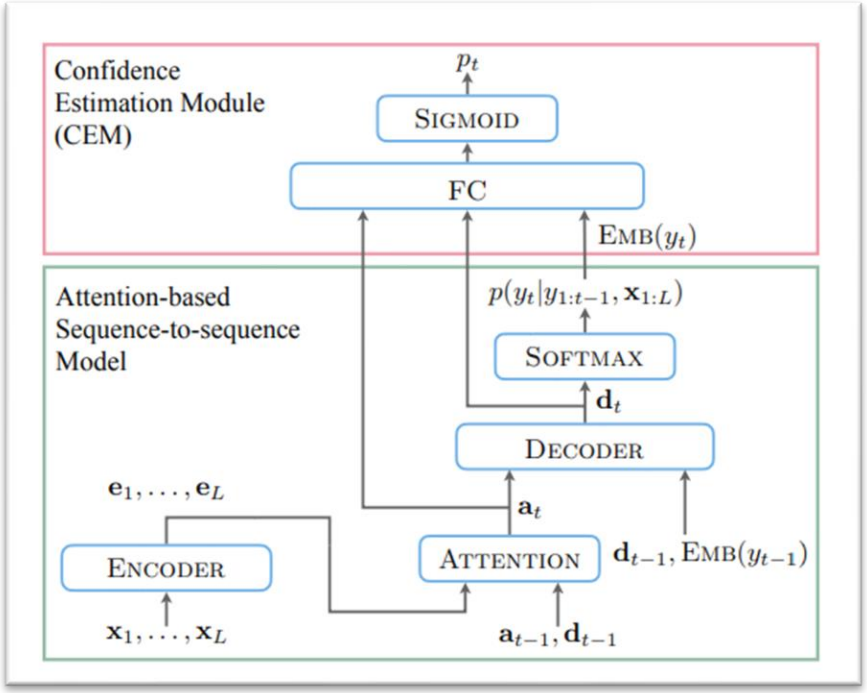
[Confidence Score] : 주어진 Token(알파벳, 음소 등의 단위)이 정답일 확률 (recognizer가 output token이 정답에 확실하다고 생각할수록 1에 가까움)

[trade – off] : attention-based seq2seq의 모델이 깊고 커질수록 성능은 올라가지만,
output으로 내놓은 sequence의 calibration behaviour(보정 작업)의 투명도는 낮아진다

[proposition] : 모델의 성능은 유지하면서 높은 퀄리티의 confidence score을 얻는 방법인 CEM을 제안한다

[CEM] : Confidence Estimation Module

[3. C E M – 용어 정리]



$x_{1:L}$: Input Utterance(입으로 낸 음성)

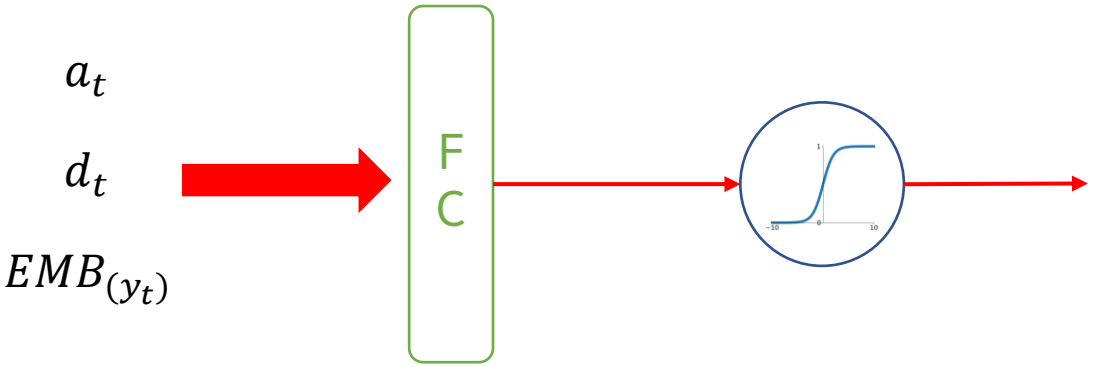
$e_{1:L} = Encoder(x_{1:L})$: Encoder의 Output

$a_t = Attention(a_{t-1}, d_{t-1}, e_{1:L})$: t번째 decoding step의 attention 값

$d_t = Decoder(a_t, d_{t-1}, EMB_{(y_{t-1})})$: t번째 decoding step의 Decoder output

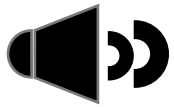
y : label token (grapheme(자소) / word piece)

$EMB_{(y_t)}$: current label token embedding



p_t : t 번째 token에 대한 confidence score

[3. C E M – 훈련 방법]



well-trained
LAS Model

hypothesis_sequence_0

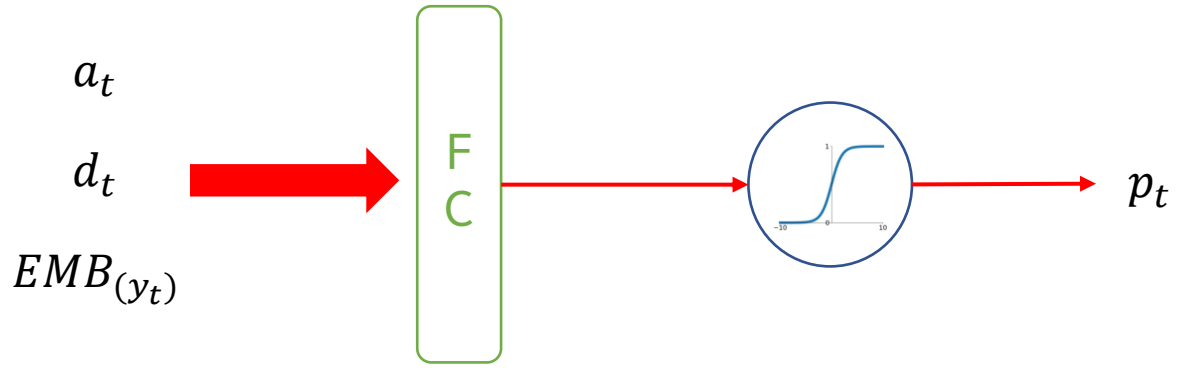
hypothesis_sequence_1

hypothesis_sequence_2

...

hypothesis_sequence_n-1

n-best hypotheses



각 hypothesis sequence와 ground truth sequence간의 edit distance 계산해 alignment 얻기

alignment에서 올바른 token은 1로,
(정답에 대해서)바뀌거나 (정답에는 없는) 새로 삽입된 토큰에 대해서는 0으로 지정된다면
이를 confidence score로 사용할 수 있다!

1. ground_truth_i = “A B C D”
2. hypothesis_sequence_i = “A C C D”
3. binary_target_sequence c_i = [1, 0, 1, 1]

이렇게 얻은 c 를 target으로 해서 CEM의 output인 p와의
binary cross entropy를 minimize 하도록 CEM을 훈련한다
(이 때 훈련 중에는 attention-based seq2seq 모델의 파라미터는 고정)

$$\mathcal{L}(\mathbf{c}, \mathbf{p}) = -\frac{1}{T}\sum_{t=1}^{T}\Big(c_t \log(p_t) + (1 - c_t)\log(1 - p_t)\Big).$$

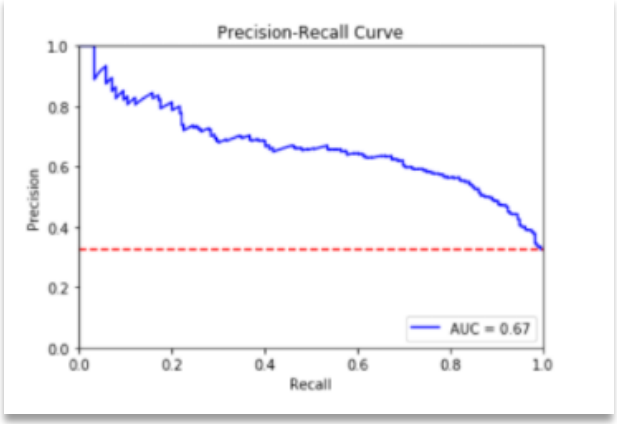
$$Total\_Loss = \sum_{k=0}^{n-1} L_k(c, p)$$

[4. 실 험 세 팅 - metrics]

각 hypothesis와 ground truth 사이의 alignment - Levenshtein Distance -

i \ j		0	1	2	3	4	5	6	7	8
		ϵ	C	A	T	G	A	C	T	G
0	ϵ	0	0	0	0	0	0	0	0	0
1	T	1	1	1	0	1	1	1	0	1
2	A	2	2	1	1	1	1	2	1	1
3	C	3	2	2	2	2	2	1	2	2
4	T	4	3	3	2	3	3	2	1	2
5	G	5	4	4	3	2	3	3	2	1

keyword spotting을 위한 임계값 작동 표현 - P-R Curve의 AUC(area under the curve) -



$$\text{precision}(\tilde{p}) = \frac{\text{TP}(\tilde{p})}{\text{TP}(\tilde{p}) + \text{FP}(\tilde{p})}, \quad \text{recall}(\tilde{p}) = \frac{\text{TP}(\tilde{p})}{\text{TP}(\tilde{p}) + \text{FN}(\tilde{p})}. \tag{8}$$

일반적으로 임계값이 증가하면 FP가 감소, FN이 증가 = 높은 precision, 낮은 recall
둘 사이의 trade-off는 왼쪽 상단 모서리에서부터 오른쪽 하단 모서리로 이어지는 하양 곡선을 만듦.

AUC(max = 1)는 confidence estimator의 quality를 측정하는 데에 사용.
두 confidence estimator가 동일한 AUC 값을 가지더라도 다른 NCE값을 가질 수 있음

confidence score의 quality 측정 - normalized cross-entropy -

$$\text{NCE}(\mathbf{c}, \mathbf{p}) = \frac{H(\mathbf{c}) - H(\mathbf{c}, \mathbf{p})}{H(\mathbf{c})}$$

- p : 각 토큰에 대해 구한 confidence socre array
- c : 각 토큰의 confidence score target array
- H(c) : target sequenc의 entropy
- H(c, p) : target과 confidence score사이의 binary cross-entropy
- confidence score가 인식된 단어가 얼마나 정확할 확률에 가까운지 측정

[4. 실 험 세 팅 – baseline]

L A S

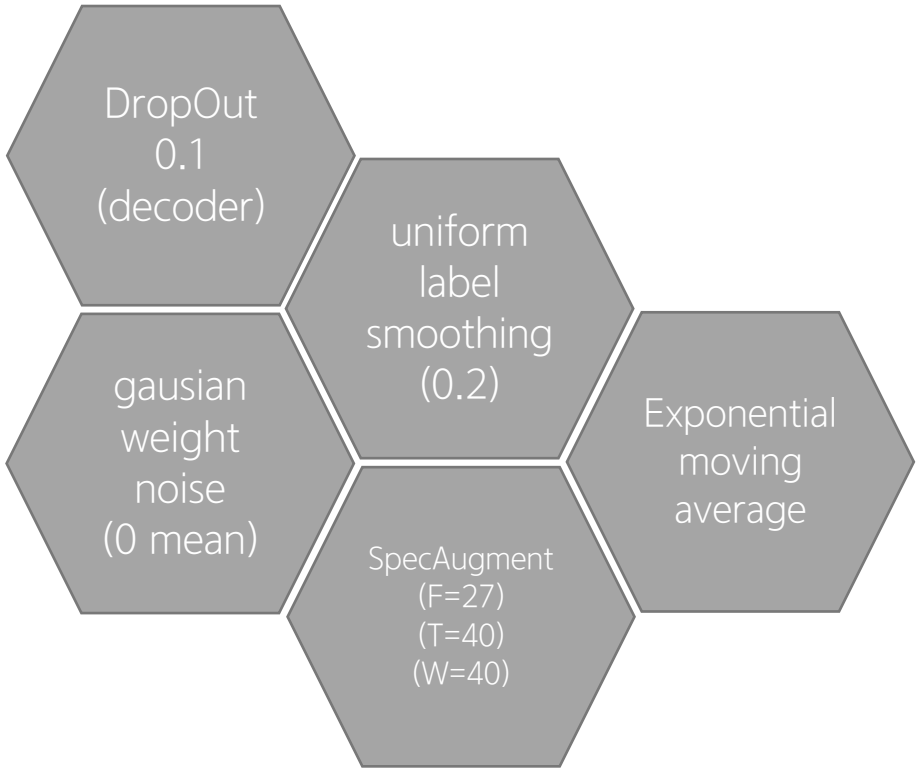
Adam
(0.001)

Encoder

- Conv Layer with max pool (s=2)
- Conv Layer with max pool (s=2)
- bi-LSTM (1024 units)
- bi-LSTM (1024 units)
- bi-LSTM (1024 units)
- bi-LSTM (1024 units)

Decoder

- uni-LSTM (1024 units)
- uni-LSTM (1024 units)



https://kaen2891.tistory.com/74

Lingvo Toolkit 오픈소스를 이용해 train한 LAS model

[5. 실 험 결 과 - Softmax Probabilities]

large model일수록 강력한 정규화 작업이 "state-of-the-art"급의 성능을 얻을 수 있게 해준다

- Augmenting Input Feature : SpecAugment
- 모델 가중치 manipulating : dropout, EMA, weight noise
- output target modifying : label smoothing

[각 regularisation tech를 제거했을 때의 WER, AUC, NCE value]

	WER ↓	AUC ↑	NCE ↑
baseline	7.5/21.6	0.976/0.912	-0.195/ 0.131
– dropout	7.8/22.0	0.977/0.916	-0.204/ 0.130
– EMA	8.2/24.8	0.974/0.903	-0.189/ 0.120
– label smoothing	10.6/24.6	0.985/0.950	0.106/-0.131
– weight noise	12.9/25.8	0.978/0.925	-0.459/-0.012
– SpecAugment	10.8/34.3	0.952/0.911	0.012/ 0.160

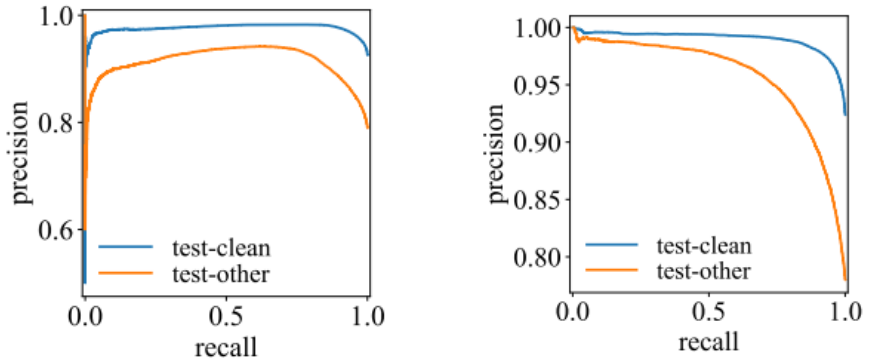
* WER은 예상대로 정규화 작업이 빠질 때마다 값이 안 좋아짐
* AUC와 NCE가 동일하게 변하는 추세를 보이지는 않음
* Augmenting Input Feature와 Model Weight Manipulating이 적어도 one metric에 대해서는 confidence performance를 올려주는 듯
  (disentanglement 적용한 관련 추가 실험 필요)
* softmax probabilites는 confidence score로 사용할 수는 있지만, (특히 output target이 modifed되는 경우의) regularisation techniques에 강하게 영향을 받는다

[5. 실 험 결 과 - CEM]

[각 regularisation tech를 제거했을 때의 WER, AUC, NCE value]

		AUC ↑	NCE ↑ (w/o PWLM)	NCE ↑ (w/ PWLM)
token	softmax	0.976/0.912	-0.195/0.131	0.166/0.172
	CEM	**0.990/0.958**	0.189/0.019	**0.344/0.275**
word	softmax	0.981/0.927	-0.180/0.139	0.269/0.195
	CEM	**0.990/0.962**	0.192/0.039	**0.350/0.270**

Table 2: Comparison of confidence scores between using softmax probabilities and using the CEM on the baseline model. The first row corresponds to the baseline in Table 1.

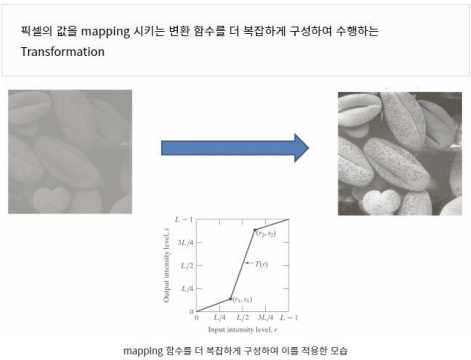


(a) softmax (b) CEM

Fig. 3: Precision-recall curves for token-level confidence scores on LibriSpeech test-clean and test-other sets.

* CEM이 훈련되는 동안에 negative sample에 대해서 WER을 더 증가시키기 위해 더 강력한 SpecAugment(T=50)이 적용됨
* 각 utterance에 대해 8-best hypotheses가 바로 생성되고 target과 aligned된다
* CEM이 추가되면서 기존 LAS에 비해 0.4%가 증가된 파라미터를 가짐

* confidence score가 token/workd accuracy와 더 잘 match되게 하기 위해
  PWLMs(Piece-wise Linear Mappings)를 dev-clean/dev-other에 대해 측정하였고 test-clean/test-other에 적용됨



[Intensity Transformation]
https://juyoungit.tistory.com/187

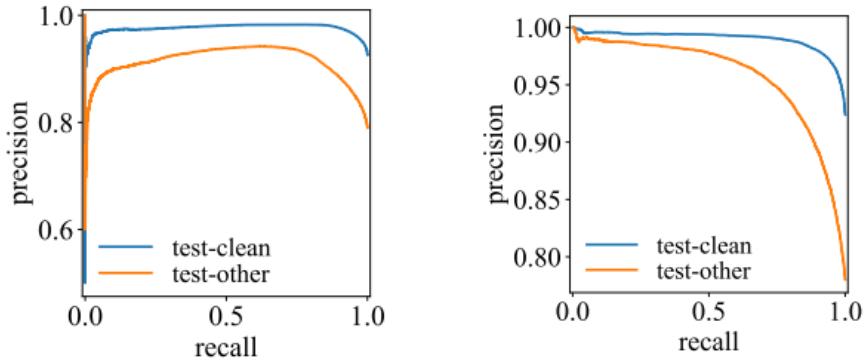
* PWLM은 monotonic(단조롭기) 때문에 confidence score의 relative order는 변경되지 않고, AUC가 유지되는 동안 NCE가 향상될 수 있음

[5. 실 험 결 과 - CEM]

[각 regularisation tech를 제거했을 때의 WER, AUC, NCE value]

		AUC ↑	NCE ↑ (w/o PWLM)	NCE ↑ (w/ PWLM)
token	softmax	0.976/0.912	-0.195/0.131	0.166/0.172
	CEM	**0.990/0.958**	0.189/0.019	**0.344/0.275**
word	softmax	0.981/0.927	-0.180/0.139	0.269/0.195
	CEM	**0.990/0.962**	0.192/0.039	**0.350/0.270**

Table 2: Comparison of confidence scores between using softmax probabilities and using the CEM on the baseline model. The first row corresponds to the baseline in Table 1.



(a) softmax (b) CEM

Fig. 3: Precision-recall curves for token-level confidence scores on LibriSpeech test-clean and test-other sets.

* AUC는 token/world level에 대해서 동시에 향상됨을 확인
* softmax 와 달리, NCE 값은 CEM을 사용했을 때 항상 positive
* PWLM을 적용하니, CEM은 더 높은 NCE value를 가지게 되었음

* softmax 그래프에서는 몇몇 incorrect token에 대해 overconfident한 모습을 보임
* CEM은 그에 반해 적절한 trade-off가 일어났음을 볼 수 있으며
  동일한 metrics에 대해서 더 적절한 confidence estimator임을 알 수 있음

[5. 실 험 결 과 - languate model과 fusion했을 때의 성능]

		WER ↓	AUC ↑	NCE ↑
baseline	softmax	7.5/21.6	0.981/0.927	0.269/0.195
	CEM		**0.990/0.962**	**0.350/0.270**
+ LM	softmax	6.8/19.8	0.981/0.928	0.103/0.109
	CEM		**0.991/0.966**	**0.337/0.263**

Table 3: ASR and word-level confidence performance for models with and without RNNLM shallow fusion (with PWLM).

* confidence estimation을 위한 LM info를 얻기 위해, CEM의 input에 current token에 대한 LM probability가 추가되었다

* WER의 경우 약 8~9%의 성능 향상이 있었지만, AUC에서는 눈에 띌만한 성능 향상이 없었음

* LM이 추가되었을 때 CEM의 confidence estimation의 quality가 상승하긴 함

[5. 실 험 결 과 - 최 종 분 석]

		WER ↓	AUC ↑	NCE ↑
baseline	softmax	18.7	0.935	0.230
	CEM		**0.970**	**0.280**
+ LM	softmax	17.7	0.933	0.159
	CEM		**0.965**	**0.266**

Table 4: ASR and confidence performance on WSJ eval92. PWLM is estimated on LibriSpeech dev-other set. LM used is trained on LibriSpeech LM corpus as in Sec. 4.3.

* CEM 훈련 때 사용한 데이터와 ASR 훈련 때 사용한 데이터가 같으므로 train-set에 대해서는 more confident

* train set과 test set의 다른 distribution은 더 많은 augmentation과 dev set에서 측정한 PWLMs을 적용하면 완화시킬 수 있음

* 하지만 여전히 다른 distribution을 가지는 domain의 데이터에 대해서는 얼마나 잘 일치할지는 모르는 일

[5. 실 험 결 과 - 최 종 분 석]

[Confidence Score로 filtering한 utterances에 대한 WER 그래프]

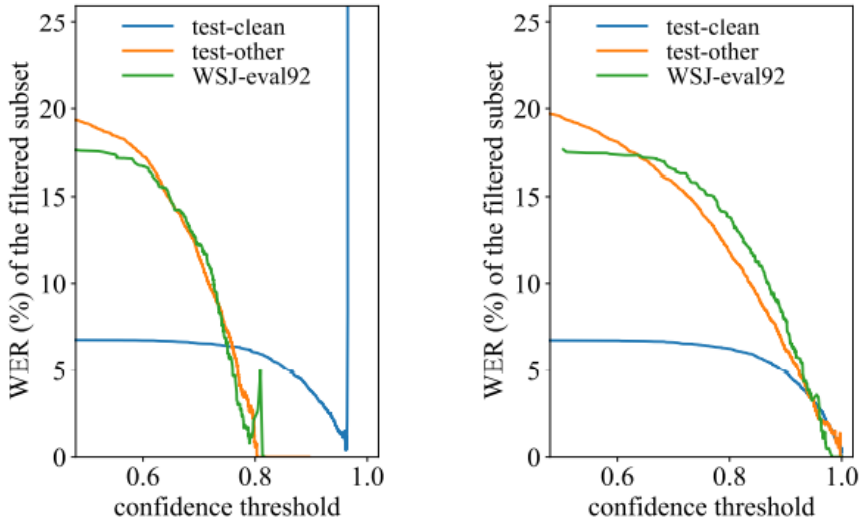


Fig. 4: WERs of filtered utterances w.r.t. confidence thresholds for softmax and CEM with LM shallow fusion.

* Confidence Score는 semi-supervised learning에서 ASR performance를 향상시키기 위해 unlabelled data를 선택하는 것에 사용됨
* 먼저 speech recogniser가 제한된 data로 훈련되고, 이후에 모델의 훈련에 noisy label로서 사용될 수 있는 unlabelled 된 data를 transcribe함.
* 하지만 이러한 automatic transcription은 model에 손상을 줄 수 있음 => confidence score를 사용해 semi-supervised training에 사용할 데이터를 filtering

* 만약 confidence score가 WER과 강력하게 연관되었다면, higher threshold는 낮은 WER의 subset도 잘 걸러낼 수 있을 것
* (b) 그래프에서 알 수 있듯이, CEM으로 측정된 confidence score는 WER과 더욱 잘 match되고 있음을 알 수 있음