Directed Automatic Speech Transcription Error Correction Using Bidirectional LSTM

Da Zheng, Zhehuai Chen, Yue Wu, Kai Yu
Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering
SpeechLab, Department of Computer Science and Engineering, Brain Science and Technology Research Center
Shanghai Jiao Tong University, Shanghai, China

Background

There are two types of error correction mechanisms for automatic speech transcription, the first type is machine-only correction, the second imports human factors.

In our method, the only thing the user needs to do is to direct where the transcriber goes wrong. After that machine will automatically give correction there. Although this way would not cover every error case, which means users have to manually correct errors if the system failed to correct, our method is lightweight enough to seamlessly get integrated into existing speech transcription systems.

Directed Error Correction Paradigm

Confusion Network Pruning

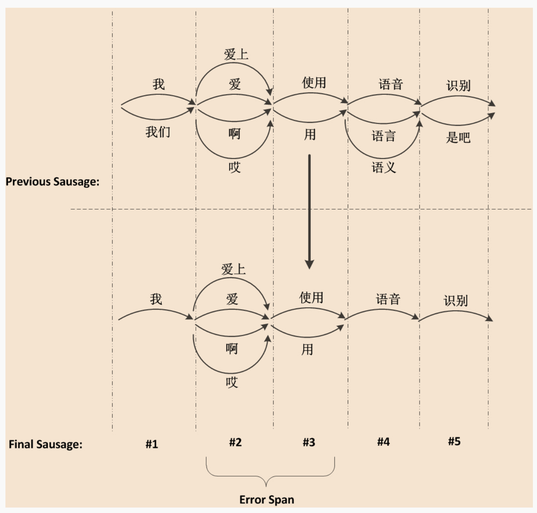

Figure 1: *A pruning example of confusion network*

- Conduct WFST-based decoding on utterance and keep the lattice.
- Confusion network (CN) is generated from the lattice.
- Prune the CN of untouched area and leave the candidate there unique.

BLSTM Rescoring

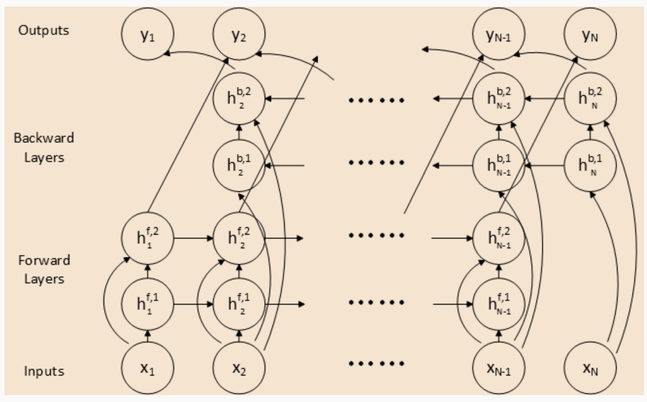

Figure 2: *BLSTM structure for prediction*

- Calculate LM scores on all paths of pruned CN.
- Mix LM scores with CN posteriors.
- Use the path with the largest score(differs from the original one)

In the structure we use, output $\boldsymbol{y}_t$ is determined by

$$P(\boldsymbol{y}_t|\{\boldsymbol{x}_d\}_{d\neq t}) = \sigma\left(\boldsymbol{W}_y^f \boldsymbol{h}_{t-1}^f + \boldsymbol{W}_y^b \boldsymbol{h}_{t+1}^b + \boldsymbol{b}_y\right) \quad (1)$$

The language model score is calculated by

$$\text{lmscore}(\boldsymbol{\omega}) = \sum_{i=1}^{N} \log P(\boldsymbol{y}_t|\{\omega_j\}_{j\neq i}) \quad (2)$$

ω represents a word sequence $\{\omega_1, \omega_2 \cdots \omega_N\}$.

The language model score is calculated by

$$\text{score}(\boldsymbol{\omega}) = \text{lmscore}(\boldsymbol{\omega}) + \lambda P_{CN}(\boldsymbol{\omega}) \quad (3)$$

where $P_{CN}(\boldsymbol{\omega})$ represents CN posterior.

Since BLSTM language model is not able to calculate sequence probability using the chain rule. We assume the relative quantity of

$$\prod_{i=1}^{N} P(\omega_i|\{\omega_j\}_{j\neq i}) \quad (4)$$

somewhat reflect the quality of the sequence $\boldsymbol{\omega}$. We use inversed normalized version of (4) as *pseudo perplexity* to evaluate the model

Experiment Setup

Dataset	Language	Train	Validation	Test
PTB	English	930K	740K	820K
SMS30M	Chinese	5.6M	165K	112K

Language	Utterances	1-pass decoder	1-pass AM	1-pass LM
Chinese	656	Real-time WFST based	DNN-HMM (5000 h)	3-gram (100M words)

Table 1: Text d*atasets/Error Correction dataset used in experiments*

Experiments

Training Language Model

A 3-gram model, 2 LSTM and 2 BLSTM language models were firstly trained on the PTB dataset.

Dataset	Hidden Size	Train PP	Valid PP	Test PP
3-gram	-	23	172	168
LSTM	200	141	112	134
LSTM	1500, 1500	70	83	105
BLSTM	200	69	45	52
BLSTM	1500, 1500	47	54	51

Table 2: *(Pseudo) perplexity of LSTM and BLSTM on PTB*

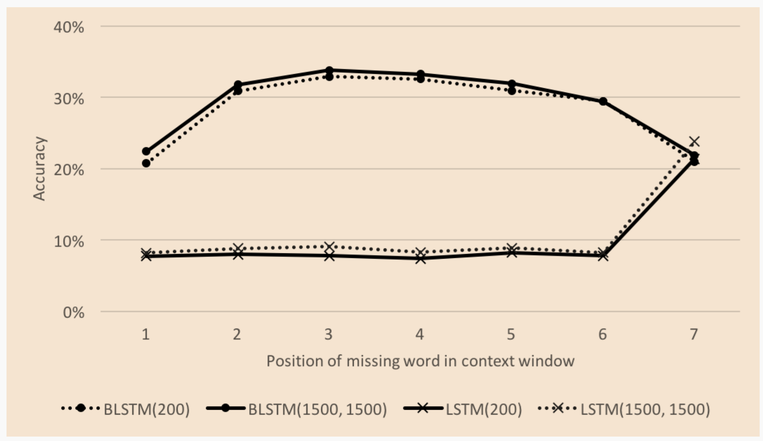

Figure 3: *Prediction accuracies of different models at each position within a context window*

Table 2:
Deeper and larger model ✓

Figure 3:
BLSTM models ✓
LSTM behaves poorly on position 1-6.

Sentence Level Word Prediction

Task: predicting a missing word.

- We compared our model's prediction power with humans'.
- Some human prediction examples are shown on the right.

会 在 我 不 开心 的 时候 陪 我
会 在 我 不 开心 的 时候 安慰 我 ✗

我 最 喜欢 的 是 容嬷嬷
我 最不 喜欢 的 是 容嬷嬷 ✗

看似 简单 的 事情
看似 简单 的 事情 ✓

Model	Cases	Accuracy
BLSTM	111803	**21.77%**
3-gram(simple)	71804	14.93%
3-gram(hybrid)*	111803	16.82%
human	500	18.8%

Table 3: *Sentence level predicting accuracy*

*hybrid uses 3-gram **x**ab to predict the first missing word, a**x**b to predict the second and penultimate word, abx to predict the last word, ab**x***x**cd to predict the rest words

Error Correction

Two types of correction strategies are compared in table 4.
Second: use the second best alternatives in pruned CN.
BLSTM: uses our trained BLSTM model to rescore pruned confusion networks.

Model	Cases	Corrected
Second	656	35.82%
BLSTM	656	**39.63%**

Table 4: *Comparison of BLSTM based correction and second best correction*

Conclusion

Our correction method focuses on getting the correction result on user directed area. Experiments show that with the combination of pruned sausages and BLSTM language model, better corrections can be retrieved.