Investigating Methods to Improve Language Model Integration for Attention-based Encoder-Decoder ASR Models

Mohammad Zeineldeen\textsuperscript{1,2}, Aleksandr Glushko\textsuperscript{1}, Wilfried Michel\textsuperscript{1,2}, Albert Zeyer\textsuperscript{1,2}, Ralf Schlüter\textsuperscript{1,2}, Hermann Ney\textsuperscript{1,2}

RWTH Aachen University\textsuperscript{1}, AppTek GmbH\textsuperscript{2}
Interspeech 2021, Brno
Introduction

- Attention encoder-decoder (AED) models benefit from external language model integration
Introduction

- Attention encoder-decoder (AED) models benefit from external language model integration
- **Problem**: AED models learn an implicit **internal language model** (ILM) from the training data
Introduction

- Attention encoder-decoder (AED) models benefit from external language model integration
- **Problem:** AED models learn an implicit **internal language model** (ILM) from the training data
- How to compute the ILM probability for prior correction during recognition for better performance?
Internal Language Model Estimation

During recognition, the search algorithm searches for the best word sequence $w_1^N$ that maximizes:

$$\hat{w}_1^N = \arg\max_{N,w_1^N} \{ \log P(w_1^N|x_T^1) \}$$

The posterior probability can be defined as:

$$P(w_1^N|x_T^1) \propto P_{AED}(w_1^N|x_T^1) \cdot P_{ILM}(w_1^N)$$

The ILM is defined as:

$$P_{ILM}(w_1^N) = \sum_{T,x_T^1} P_{AED}(w_1^N|x_T^1) \cdot P(x_T^1)$$

However, the summation is intractable.

We propose different novel methods to estimate the ILM for AED models.
Internal Language Model Estimation

During recognition, the search algorithm searches for the best word sequence $w_1^N$ that maximizes:

$$\hat{w}_1^N = \arg \max_{N,w_1^N} \{ \log P(w_1^N | x_1^T) \}$$

The posterior probability can be defined as:

$$P(w_1^N | x_1^T) \propto P_{AED}(w_1^N | x_1^T) \cdot P_{LM}^{\lambda_1}(w_1^N)$$

However, the summation is intractable.

We propose different novel methods to estimate the ILM for AED models.
Internal Language Model Estimation

During recognition, the search algorithm searches for the best word sequence $w_1^N$ that maximizes:

$$\hat{w}_1^N = \arg \max_{N,w_1^N} \{ \log P(w_1^N|x_T^1) \}$$

The posterior probability can be defined as:

$$P(w_1^N|x_T^1) \propto P_{AED}(w_1^N|x_T^1) \cdot P_{LM}^{\lambda_1}(w_1^N) \cdot P_{ILM}^{-\lambda_2}(w_1^N)$$

However, the summation is intractable.
Internal Language Model Estimation

During recognition, the search algorithm searches for the best word sequence $w_1^N$ that maximizes:

$$\hat{w}_1^N = \arg \max_{N,w_1^N} \{ \log P(w_1^N|x_1^T) \}$$

The posterior probability can be defined as:

$$P(w_1^N|x_1^T) \propto P_{AED}(w_1^N|x_1^T) \cdot P_{LM}^{\lambda_1}(w_1^N) \cdot P_{ILM}^{-\lambda_2}(w_1^N)$$

The ILM is defined as:

$$P_{ILM}(w_1^N) = \sum_{T,x_1^T} P_{AED}(w_1^N|x_1^T) \cdot P(x_1^T)$$
Internal Language Model Estimation

During recognition, the search algorithm searches for the best word sequence $w_1^N$ that maximizes:

$$\hat{w}_1^N = \arg \max_{N,w_1^N} \{ \log P(w_1^N|x_1^T) \}$$

The posterior probability can be defined as:

$$P(w_1^N|x_1^T) \propto P_{AED}(w_1^N|x_1^T) \cdot P_{LM}^{\lambda_1}(w_1^N) \cdot P_{ILM}^{-\lambda_2}(w_1^N)$$

The ILM is defined as:

$$P_{ILM}(w_1^N) = \sum_{T,x_1^T} P_{AED}(w_1^N|x_1^T) \cdot P(x_1^T)$$

However, the summation is intractable.
Internal Language Model Estimation

During recognition, the search algorithm searches for the best word sequence \( w_1^N \) that maximizes:

\[
\hat{w}_1^N = \arg \max_{N,w_1^N} \{ \log P(w_1^N|x_1^T) \}
\]

The posterior probability can be defined as:

\[
P(w_1^N|x_1^T) \propto P_{AED}(w_1^N|x_1^T) \cdot P_{LM}^{\lambda_1}(w_1^N) \cdot P_{-\lambda_2}^{ILM}(w_1^N)
\]

The ILM is defined as:

\[
P_{ILM}(w_1^N) = \sum_{T,x_1^T} P_{AED}(w_1^N|x_1^T) \cdot P(x_1^T)
\]

However, the summation is intractable.

→ We propose different novel methods to estimate the ILM for AED models
Approaches

- ILM estimation methods can be classified as:
  1. Model-agnostic methods (e.g., Density Ratio [McDermott & Sak 19])
  2. Model-specific methods [Variani & Rybach 20, Meng & Parthasarathy 20]

- We argue that using encoder bias can be helpful and this is more consistent with training.

- This work focuses on model-specific estimation methods by replacing attention context vector with either static or trained context vectors.
Approaches

- ILM estimation methods can be classified as:
  1. Model-agnostic methods (e.g. Density Ratio [McDermott & Sak 19])
  2. Model-specific methods [Variani & Rybach 20, Meng & Parthasarathy 20]

- We argue that using encoder bias can be helpful and this is more consistent with training.

- This work focuses on model-specific estimation methods by replacing attention context vector with either static or trained context vectors.
**Approaches**

- ILM estimation methods can be classified as:
  1. Model-agnostic methods (e.g., Density Ratio [McDermott & Sak+ 19])
  2. Model-specific methods [Variani & Rybach+ 20, Meng & Parthasarathy+ 20]
Approaches

- ILM estimation methods can be classified as:
  1. Model-agnostic methods (e.g. Density Ratio [McDermott & Sak+ 19])
  2. Model-specific methods [Variani & Rybach+ 20, Meng & Parthasarathy+ 20]

- We argue that using encoder bias can be helpful and this is more consistent with training
Approaches

- ILM estimation methods can be classified as:
  1. Model-agnostic methods (e.g. Density Ratio [McDermott & Sak+ 19])
  2. Model-specific methods [Variani & Rybach+ 20, Meng & Parthasarathy+ 20]

- We argue that using encoder bias can be helpful and this is more consistent with training
- This work focuses on model-specific estimation methods by replacing attention context vector with either static or trained context vectors
Attention Encoder-Decoder Model
Static Context Vector Estimation

- Static vector $\rightarrow$ position independent
- Replace original context vector $c_i$ by $\hat{c}$:
  - Zero vector (all elements are zero)
  - Average of all encoder states over train data
  - Average of all context vectors over train data
Trained Context Vector Estimation

- Training Steps
  1. **Freeze** all the parameters of AED model
  2. Add Linear and Mini-LSTM **trainable** layers
  3. Retrain the AED model for few epochs
- **Minimizes** directly the **perplexity**
- Trained only on transcription
ILM Suppression

- Limited Context Decoder
  - Replace the LSTM in the decoder with feed-forward layers
  - Less effective ILM

- Train AED together with LM via sequence training or local log-linear combination
  [Michel & Schlüter+ 20]
  - ASR model relies on the LM for language modeling and focuses on acoustic modeling
## Results on Switchboard 300h

<table>
<thead>
<tr>
<th>Method</th>
<th>WER [%]</th>
<th>Hub5'01</th>
<th>RT03</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td></td>
<td>13.4</td>
<td>16.3</td>
</tr>
<tr>
<td>Shallow Fusion</td>
<td></td>
<td>13.0</td>
<td>15.7</td>
</tr>
<tr>
<td>Density Ratio</td>
<td></td>
<td>12.7</td>
<td>15.3</td>
</tr>
<tr>
<td>zero</td>
<td></td>
<td>12.9</td>
<td>15.6</td>
</tr>
<tr>
<td>$E_D[h]$</td>
<td></td>
<td>12.3</td>
<td>15.0</td>
</tr>
<tr>
<td>$E_D[c]$</td>
<td></td>
<td>12.4</td>
<td>14.9</td>
</tr>
<tr>
<td>$E_x[h]$</td>
<td></td>
<td>12.6</td>
<td>15.2</td>
</tr>
<tr>
<td>Mini-LSTM</td>
<td><strong>12.2</strong></td>
<td><strong>14.8</strong></td>
<td></td>
</tr>
</tbody>
</table>

- ILM estimation by replacing attention context vector by:
  - `zero`: zero vector
  - $E_D[h]$: average of encoder states over train data
  - $E_D[c]$: average of context vectors over train data
  - $E_x[h]$: average encoder states during recognition
  - Mini-LSTM: trained context vector

- Achieved **6% relative improvement** in terms of WER compared to Shallow Fusion
## Results on LibriSpeech 960h

<table>
<thead>
<tr>
<th>Method</th>
<th>WER [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>dev-other</td>
</tr>
<tr>
<td>None</td>
<td>10.37</td>
</tr>
<tr>
<td>Shallow Fusion</td>
<td>6.80</td>
</tr>
<tr>
<td>Density Ratio</td>
<td>6.68</td>
</tr>
<tr>
<td>train w. LM</td>
<td>6.19</td>
</tr>
<tr>
<td>zero</td>
<td>6.43</td>
</tr>
<tr>
<td>$E_D[h]$</td>
<td>6.19</td>
</tr>
<tr>
<td>$E_D[c]$</td>
<td>6.19</td>
</tr>
<tr>
<td>$E_x[h]$</td>
<td>6.34</td>
</tr>
<tr>
<td>Mini-LSTM</td>
<td><strong>5.76</strong></td>
</tr>
</tbody>
</table>

- **train w. LM:** train AED model with LM to suppress ILM
- **ILM estimation** by replacing attention context vector by:
  - zero: zero vector
  - $E_D[h]$: average of encoder states over train data
  - $E_D[c]$: average of context vectors over train data
  - $E_x[h]$: average encoder states during recognition
- **Mini-LSTM:** trained context vector

Achieved **15% and 16% relative improvement** in terms of WER compared to Shallow Fusion
Cross-domain Evaluation

- ASR model trained on LibriSpeech 960h dataset
- Evaluated on TED-LIUM-V2 [Rousseau & Delégilse+ 14] dev and test datastes

<table>
<thead>
<tr>
<th>Method</th>
<th>WER [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TLv2-dev</td>
</tr>
<tr>
<td>None</td>
<td>22.0</td>
</tr>
<tr>
<td>Shallow Fusion</td>
<td>18.5</td>
</tr>
<tr>
<td>Density Ratio</td>
<td>16.6</td>
</tr>
<tr>
<td>zero</td>
<td>17.3</td>
</tr>
<tr>
<td>$\mathbb{E}_D[h]$</td>
<td>16.7</td>
</tr>
<tr>
<td>$\mathbb{E}_D[c]$</td>
<td>16.8</td>
</tr>
<tr>
<td>$\mathbb{E}_x[h]$</td>
<td>16.7</td>
</tr>
<tr>
<td>Mini-LSTM</td>
<td>16.1</td>
</tr>
</tbody>
</table>

- ILM estimation by replacing attention context vector by:
  - zero: zero vector
  - $\mathbb{E}_D[h]$: average of encoder states over train data
  - $\mathbb{E}_D[c]$: average of context vectors over train data
  - $\mathbb{E}_x[h]$: average encoder states during recognition
  - Mini-LSTM: trained context vector
Limited Context Decoder - Switchboard 300h

<table>
<thead>
<tr>
<th>Method</th>
<th>WER [%]</th>
<th>Hub5’01</th>
<th>RT03</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td></td>
<td>14.0</td>
<td>16.8</td>
</tr>
<tr>
<td>SF</td>
<td>13.2</td>
<td>15.6</td>
<td></td>
</tr>
<tr>
<td>DR</td>
<td>13.2</td>
<td>15.6</td>
<td></td>
</tr>
<tr>
<td>zero</td>
<td>12.6</td>
<td>15.0</td>
<td></td>
</tr>
<tr>
<td>$E_D[h]$</td>
<td>12.4</td>
<td>14.8</td>
<td></td>
</tr>
<tr>
<td>$E_D[c]$</td>
<td></td>
<td></td>
<td>14.9</td>
</tr>
<tr>
<td>$E_x[h]$</td>
<td>12.5</td>
<td></td>
<td>14.9</td>
</tr>
<tr>
<td>Mini-LSTM</td>
<td>12.6</td>
<td></td>
<td>14.9</td>
</tr>
</tbody>
</table>

- ILM estimation by replacing attention context vector by:
  - zero: zero vector
  - $E_D[h]$: average of encoder states over train data
  - $E_D[c]$: average of context vectors over train data
  - $E_x[h]$: average encoder states during recognition
  - Mini-LSTM: trained context vector

- 1-layer FF decoder with context size 3
- **Average-based static** estimation methods perform better
Conclusions

- Subtracting the internal language model (ILM) during recognition gives significant improvements in terms of WER.

- We proposed a novel method to train the attention context vector for ILM estimation which outperforms other methods.

- We achieved 6% relative improvement in terms of WER on Switchboard 300h test sets as well as 15%-16% on LibriSpeech test sets.

- Feed-forward or limited context decoder AED model can achieve comparable results to a recurrent decoder on Switchboard 300h task with ILM subtraction.

- This work shows the importance of considering ILM subtraction in order to achieve better results.
Thank you for your attention

Any questions?
References

A density ratio approach to language model fusion in end-to-end automatic speech recognition.

Internal language model estimation for domain-adaptive end-to-end speech recognition.

Early Stage LM Integration Using Local and Global Log-Linear Combination.

Enhancing the TED-LIUM corpus with selected data for language modeling and more TED talks.
References
