LSTM: A Search Space Odyssey

Authors: Klaus Greff, Rupesh K. Srivastava, Jan Koutník, Bas R. Steunebrink, Jürgen Schmidhuber
Presenter: Sidhartha Satapathy
Scientific contributions of the paper:

- The paper aims at evaluating different elements of the most popular LSTM architecture.
- The paper shows the performance of various variants of the vanilla LSTM by making a single change which allows us to isolate the effect of each of these changes on the performance of the architecture.
- The paper also provide insights gained about hyperparameters and their interaction.
IAM Online Handwriting Database: The IAM Handwriting Database contains forms of handwritten English text which can be used to train and test handwritten text recognizers and to perform writer identification and verification experiments.
In mid-April Anglésy moved his family and entourage from Rome to Naples, there to await the arrival of
Each sequence or line in this case is made up of frames and the task at hand is to classify each of these frames into one of the 82 characters.

Here are the output characters:
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789 !"#$\'()*+,-./[\]:;? And the empty symbol.

The performance in this case is the character error rate.
Dataset 2: TIMIT

- TIMIT Speech corpus: **TIMIT** is a corpus of phonemically and lexically transcribed speech of American English speakers of different sexes and dialects.
Our experiments focus on the frame-wise classification task for this dataset, where the objective is to classify each audio-frame as one of 61 phones.

The performance in this case is the classification error rate.
<table>
<thead>
<tr>
<th>Phone Label</th>
<th>Example</th>
<th>Phone Label</th>
<th>Example</th>
<th>Phone Label</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>iy</td>
<td>22</td>
<td>ch</td>
<td>43</td>
<td>en</td>
</tr>
<tr>
<td>2</td>
<td>ih</td>
<td>23</td>
<td>b</td>
<td>44</td>
<td>eng</td>
</tr>
<tr>
<td>3</td>
<td>eh</td>
<td>24</td>
<td>d</td>
<td>45</td>
<td>l</td>
</tr>
<tr>
<td>4</td>
<td>ey</td>
<td>25</td>
<td>g</td>
<td>46</td>
<td>r</td>
</tr>
<tr>
<td>5</td>
<td>ae</td>
<td>26</td>
<td>p</td>
<td>47</td>
<td>w</td>
</tr>
<tr>
<td>6</td>
<td>aa</td>
<td>27</td>
<td>t</td>
<td>48</td>
<td>y</td>
</tr>
<tr>
<td>7</td>
<td>aw</td>
<td>28</td>
<td>k</td>
<td>49</td>
<td>hh</td>
</tr>
<tr>
<td>8</td>
<td>ay</td>
<td>29</td>
<td>dx</td>
<td>50</td>
<td>hv</td>
</tr>
<tr>
<td>9</td>
<td>ah</td>
<td>30</td>
<td>s</td>
<td>51</td>
<td>el</td>
</tr>
<tr>
<td>10</td>
<td>ao</td>
<td>31</td>
<td>sh</td>
<td>52</td>
<td>bcl</td>
</tr>
<tr>
<td>11</td>
<td>oy</td>
<td>32</td>
<td>z</td>
<td>53</td>
<td>dcl</td>
</tr>
<tr>
<td>12</td>
<td>ow</td>
<td>33</td>
<td>zh</td>
<td>54</td>
<td>gcl</td>
</tr>
<tr>
<td>13</td>
<td>uh</td>
<td>34</td>
<td>f</td>
<td>55</td>
<td>pcl</td>
</tr>
<tr>
<td>14</td>
<td>uw</td>
<td>35</td>
<td>th</td>
<td>56</td>
<td>tcl</td>
</tr>
<tr>
<td>15</td>
<td>ux</td>
<td>36</td>
<td>v</td>
<td>57</td>
<td>kcl</td>
</tr>
<tr>
<td>16</td>
<td>er</td>
<td>37</td>
<td>dh</td>
<td>58</td>
<td>q</td>
</tr>
<tr>
<td>17</td>
<td>ax</td>
<td>38</td>
<td>m</td>
<td>59</td>
<td>pau</td>
</tr>
<tr>
<td>18</td>
<td>ix</td>
<td>39</td>
<td>n</td>
<td>60</td>
<td>epi</td>
</tr>
<tr>
<td>19</td>
<td>axr</td>
<td>40</td>
<td>ng</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>ax-h</td>
<td>41</td>
<td>em</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>jh</td>
<td>42</td>
<td>nx</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2. 61 TIMIT original phone set.
Dataset 3: JSB Chorales

- JSB Chorales: JSB Chorales is a collection of 382 four part harmonized chorales by J. S. Bach, the networks where trained to do next-step prediction.
\[ \bar{z}^t = W_z x^t + R_z y^{t-1} + b_z \]
\[ z^t = g(\bar{z}^t) \]
\[ \bar{i}^t = W_i x^t + R_i y^{t-1} + p_i \odot c^{t-1} + b_i \]
\[ i^t = \sigma(\bar{i}^t) \]
\[ \bar{f}^t = W_f x^t + R_f y^{t-1} + p_f \odot c^{t-1} + b_f \]
\[ f^t = \sigma(\bar{f}^t) \]
\[ c^t = z^t \odot i^t + c^{t-1} \odot f^t \]
\[ \bar{o}^t = W_o x^t + R_o y^{t-1} + p_o \odot c^t + b_o \]
\[ o^t = \sigma(\bar{o}^t) \]
\[ y^t = h(c^t) \odot o^t \]

block input

input gate

forget gate

cell

output gate

block output
Variants of the LSTM Block:

- NIG: No Input Gate
- NFG: No Forget Gate
- NOG: No Output Gate
- NIAF: No Input Activation Function
- NOAF: No Output Activation Function
- CIFG: Coupled Input and Forget Gate
- NP: No Peepholes
- FGR: Full Gate Recurrence
NIG: No Input Gate

\[ \tilde{z}^t = W_z x^t + R_z y^{t-1} + b_z \]
\[ z^t = g(\tilde{z}^t) \]
\[ \tilde{i}^t = W_i x^t + R_i y^{t-1} + p_i \sigma c^{t-1} + b_i \]
\[ i^t = \sigma(\tilde{i}^t) \]
\[ \tilde{f}^t = W_f x^t + R_f y^{t-1} + p_f \sigma c^{t-1} + b_f \]
\[ f^t = \sigma(\tilde{f}^t) \]
\[ c^t = z^t \sigma i^t + c^{t-1} \sigma f^t \]
\[ \tilde{o}^t = W_o x^t + R_o y^{t-1} + p_o \sigma c^t + b_o \]
\[ o^t = \sigma(\tilde{o}^t) \]
\[ y^t = h(c^t) \sigma o^t \]
NFG: No Forget Gate

$$z^t = W_z x^t + R_z y^{t-1} + b_z$$

$$z^t = g(z^t)$$  \(\text{block input}\)

$$\bar{i}^t = W_i x^t + R_i y^{t-1} + p_i \odot c^{t-1} + b_i$$

$$i^t = \sigma(\bar{i}^t)$$  \(\text{input gate}\)

$$\bar{f}^t = W_f x^t + R_f y^{t-1} + p_f \odot c^{t-1} + b_f$$

$$f^t = \sigma(\bar{f}^t)$$  \(\text{forget gate}\)

$$c^t = z^t \odot i^t + c^{t-1} \odot f^t$$  \(\text{cell}\)

$$\bar{o}^t = W_o x^t + R_o y^{t-1} + p_o \odot c^t + b_o$$

$$o^t = \sigma(\bar{o}^t)$$  \(\text{output gate}\)

$$y^t = h(c^t) \odot o^t$$  \(\text{block output}\)
NOG: No Output Gate: $o^t = 1$

\[
\begin{align*}
\bar{z}^t &= W_z x^t + R_z y^{t-1} + b_z \\
z^t &= g(\bar{z}^t) \\
\bar{i}^t &= W_i x^t + R_i y^{t-1} + p_i \odot c^{t-1} + b_i \\
i^t &= \sigma(\bar{i}^t) \\
\bar{f}^t &= W_f x^t + R_f y^{t-1} + p_f \odot c^{t-1} + b_f \\
f^t &= \sigma(\bar{f}^t) \\
c^t &= z^t \odot i^t + c^{t-1} \odot f^t \\
o^t &= W_o x^t + R_o y^{t-1} + p_o \odot c^t + b_o \\
o^t &= \sigma(o^t) \\
y^t &= h(c^t) \odot o^t
\end{align*}
\]
NIAF: No Input Activation Function

\[
\begin{align*}
\bar{z}^t &= W_z x_t + R_z y^{t-1} + b_z \\
z^t &= g(\bar{z}^t) \\
\bar{i}^t &= W_i x_t + R_i y^{t-1} + p_i \odot c^{t-1} + b_i \\
i^t &= \sigma(\bar{i}^t) \\
\bar{f}^t &= W_f x_t + R_f y^{t-1} + p_f \odot c^{t-1} + b_f \\
f^t &= \sigma(\bar{f}^t) \\
c^t &= z^t \odot i^t + c^{t-1} \odot f^t \\
\bar{o}^t &= W_o x_t + R_o y^{t-1} + p_o \odot c^t + b_o \\
o^t &= \sigma(\bar{o}^t) \\
y^t &= h(c^t) \odot o^t
\end{align*}
\]
NOAF: No Output Activation Function

\[ h(x) = x \]

\[
\begin{align*}
\tilde{z}^t &= W_z x^t + R_z y^{t-1} + b_z \\
z^t &= g(\tilde{z}^t) \\
i^t &= W_i x^t + R_i y^{t-1} + p_i \odot c^{t-1} + b_i \\
i^t &= \sigma(i^t) \\
\bar{f}^t &= W_f x^t + R_f y^{t-1} + p_f \odot c^{t-1} + b_f \\
f^t &= \sigma(\bar{f}^t) \\
c^t &= \tilde{z}^t \odot i^t + c^{t-1} \odot f^t \\
\bar{o}^t &= W_o x^t + R_o y^{t-1} + p_o \odot c^t + b_o \\
o^t &= \sigma(\bar{o}^t) \\
y^t &= h(c^t) \odot o^t
\end{align*}
\]
CIFG: Coupled Input and Forget Gate

\[ f^t = 1 - i^t \]

\[
\bar{z}^t = W_z x^t + R_z y^{t-1} + b_z
\]

\[
z^t = g(\bar{z}^t) \quad \text{block input}
\]

\[
\bar{i}^t = W_i x^t + R_i y^{t-1} + p_i \circ c^{t-1} + b_i
\]

\[
i^t = \sigma(\bar{i}^t) \quad \text{input gate}
\]

\[
\bar{f}^t = W_f x^t + R_f y^{t-1} + p_f \circ c^{t-1} + b_f
\]

\[
f^t = \sigma(\bar{f}^t) \quad \text{forget gate}
\]

\[
c^t = z^t \circ i^t + c^{t-1} \circ f^t \quad \text{cell}
\]

\[
\tilde{o}^t = W_o x^t + R_o y^{t-1} + p_o \circ c^t + b_o
\]

\[
o^t = \sigma(\tilde{o}^t) \quad \text{output gate}
\]

\[
y^t = h(c^t) \circ o^t \quad \text{block output}\]
No Peepholes

NP: No Peepholes:

\[
\begin{align*}
\bar{i}^t &= W_i x^t + R_i y^{t-1} + b_i \\
\bar{f}^t &= W_f x^t + R_f y^{t-1} + b_f \\
\bar{o}^t &= W_o x^t + R_o y^{t-1} + b_o
\end{align*}
\]
NP: No Peepholes

\[
\begin{align*}
\bar{z}^t &= W_z x^t + R_z y^{t-1} + b_z \\
z^t &= g(\bar{z}^t) \\
i^t &= W_i x^t + R_i y^{t-1} + p_i \odot c^{t-1} + b_i \\
i^t &= \sigma(i^t) \\
\bar{f}^t &= W_f x^t + R_f y^{t-1} + p_f \odot c^{t-1} + b_f \\
f^t &= \sigma(\bar{f}^t) \\
c^t &= z^t \odot i^t + c^{t-1} \odot f^t \\
o^t &= W_o x^t + R_o y^{t-1} + p_o \odot c^t + b_o \\
o^t &= \sigma(o^t) \\
y^t &= h(c^t) \odot o^t
\end{align*}
\]
FGR: Full Gate Recurrence:

\[
\begin{align*}
\bar{i}^t &= W_{ix} x^t + R_{iy} y^{t-1} + p_i \circ c^{t-1} + b_i \\
&\quad + R_{ii} i^{t-1} + R_{fi} f^{t-1} + R_{oi} o^{t-1} \\
\bar{f}^t &= W_{fx} x^t + R_{fy} y^{t-1} + p_f \circ c^{t-1} + b_f \\
&\quad + R_{if} i^{t-1} + R_{ff} f^{t-1} + R_{of} o^{t-1} \\
\bar{o}^t &= W_{ox} x^t + R_{oy} y^{t-1} + p_o \circ c^{t-1} + b_o \\
&\quad + R_{io} i^{t-1} + R_{fo} f^{t-1} + R_{oo} o^{t-1}
\end{align*}
\]
FGR: Full Gate Recurrence

\[ z^t = W_z x^t + R_z y^{t-1} + b_z \]
\[ z^t = g(\bar{z}^t) \]  
**block input**

\[ \tilde{i}^t = W_i x^t + R_i y^{t-1} + p_i \odot c^{t-1} + b_i \]
\[ i^t = \sigma(\tilde{i}^t) \]  
**input gate**

\[ \tilde{f}^t = W_f x^t + R_f y^{t-1} + p_f \odot c^{t-1} + b_f \]
\[ f^t = \sigma(\tilde{f}^t) \]  
**forget gate**

\[ c^t = z^t \odot i^t + c^{t-1} \odot f^t \]
**cell**

\[ \bar{o}^t = W_o x^t + R_o y^{t-1} + p_o \odot c^t + b_o \]
\[ o^t = \sigma(\bar{o}^t) \]  
**output gate**

\[ y^t = h(c^t) \odot o^t \]  
**block output**
Hyperparameter Search

While there are other methods to efficiently search for good hyperparameters, this paper uses random search has several advantages for our setting:

- it is easy to implement
- trivial to parallelize
- covers the search space more uniformly, thereby improving the follow-up analysis of hyperparameter importance.
The paper shows 27 random searches (one for each combination of the nine variants and three datasets). Each random search encompasses 200 trials for a total of 5400 trials of randomly sampling the hyperparameters.
The hyperparameters and ranges are:

- hidden layer size: log-uniform samples from $[20; 200]$
- learning rate: log-uniform samples from $[10^{-6}; 10^{-2}]$
- momentum: $1 - \text{log-uniform samples from } [0.01; 1.0]$
- standard deviation of Gaussian input noise: uniform samples from $[0; 1]$. 
### Results and Discussions:

<table>
<thead>
<tr>
<th>Datasets</th>
<th>State of the art:</th>
<th>Best result:</th>
</tr>
</thead>
<tbody>
<tr>
<td>IAM Online</td>
<td>26.9% (Best LSTM Result)</td>
<td>9.26%</td>
</tr>
<tr>
<td>TIMIT</td>
<td>26.9%</td>
<td>29.6%</td>
</tr>
<tr>
<td>JSB Chorales</td>
<td>-5.56</td>
<td>-8.38</td>
</tr>
</tbody>
</table>
Hyperparameter Analysis:

- Learning Rate: It is the most important hyperparameter and accounts for 67% of the variance on the test set performance.
- We observe there is a sweet-spot at the higher end of learning rate, where the performance is good and the training time is small.
Hyperparameter Analysis:

- Hidden Layer Size: Not surprisingly the hidden layer size is an important hyperparameter affecting the LSTM network performance. As expected, larger networks perform better.
- It can also be seen in the figure that the required training time increases with the network size.
Hyperparameter Analysis:

- Input Noise: Additive Gaussian noise on the inputs, a traditional regularizer for neural networks, has been used for LSTM as well. However, we find that not only does it almost always hurt performance, it also slightly increases training times. The only exception is TIMIT, where a small dip in error for the range of \([0:2; 0:5]\) is observed.
Conclusion:

- We conclude that the most commonly used LSTM architecture (vanilla LSTM) performs reasonably well on various datasets.
- None of the eight investigated modifications significantly improves performance. However, certain modifications such as coupling the input and forget gates or removing peephole connections, simplified LSTMs in our experiments without significantly decreasing performance.
● The forget gate and the output activation function are the most critical components of the LSTM block. Removing any of them significantly impairs performance.

● The learning rate (range: log-uniform samples from $[10^{-6}; 10^{-2}]$) is the most crucial hyperparameter, followed by the hidden layer size (range: log-uniform samples from $[20; 200]$).

● The analysis of hyperparameter interactions revealed no apparent structure.
THANK YOU