COLX-585: Trends in Computational Linguistics

Muhammad Abdul-Mageed

muhammad.mageed@ubc.ca

Deep Learning & NLP Lab

The University of British Columbia

Generative Deep Learning: Language Models

Table of Contents

Generating Sequences With Recurrent Neural Networks

Alex Graves
Department of Computer Science
University of Toronto
graves@cs.toronto.edu

Abstract

This paper shows how Long Short-term Memory recurrent neural networks can be used to generate complex sequences with long-range structure, simply by predicting one data point at a time. The approach is demonstrated for text (where the data are discrete) and online handwriting (where the data are real-valued). It is then extended to handwriting synthesis by allowing the network to condition its predictions on a text sequence. The resulting system is able to generate highly realistic cursive handwriting in a wide variety of styles.

RNN Architecture

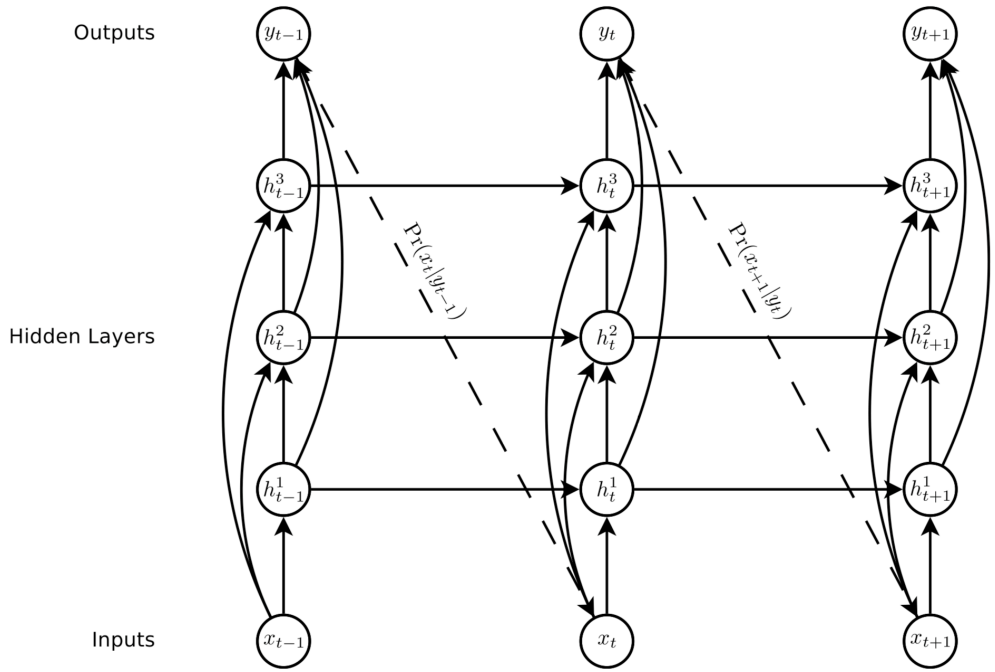

Figure 1: **Deep recurrent neural network prediction architecture.** The circles represent network layers, the solid lines represent weighted connections and the dashed lines represent predictions.

Note: the 'skip connections' from the inputs to all hidden layers, and from all hidden layers to the outputs. These mitigate 'vanishing gradients'.

Character-Level LSTM

Why Char Level?

1. Modeling words would run into data sparsity
2. For softmax-based models, high computational cost for evaluating all exponentials during training
3. Predicting one char at a time, allows the network to invent novel 'words'

Evaluation

Eval

1. **Bits-per-character (BPC)**: = avg value of $-log_2\ p(x_{i+1}|y_i)$ over the whole test set
2. **Perplexity:** Two to the power of the average number of bits per word
   - For a test set with avg. number of 5.6 chars, perplexity $\approx 2^{5.6BPC}$

Sample Generated Wikipedia Article

```
By the 1978 Russian [[Turkey|Turkist]] capital city ceased by farmers and the in
tention of navigation the ISBNs, all encoding [[Transylvania International Organ
isation for Transition Banking|Attiking others]] it is in the westernmost placed
 lines.  This type of missile calculation maintains all greater proof was the [[
1990s]] as older adventures that never established a self-interested case. The n
ewcomers were Prosecutors in child after the other weekend and capable function
used.

Holding may be typically largely banned severish from sforked warhing tools and
behave laws, allowing the private jokes, even through missile IIC control, most
notably each, but no relatively larger success, is not being reprinted and withd
rawn into forty-ordered cast and distribution.

Besides these markets (notably a son of humor).

Sometimes more or only lowed &quot;80&quot; to force a suit for http://news.bbc.
co.uk/1/sid9kcid/web/9960219.html ''[[#10:82-14]]''.
&lt;blockquote&gt;

===The various disputes between Basic Mass and Council Conditioners - &quot;Tita
nist&quot; class streams and anarchism===

Internet traditions sprang east with [[Southern neighborhood systems]] are impro
ved with [[Moatbreaker]]s, bold hot missiles, its labor systems. [[KCD]] numbere
d former ISBN/MAS/speaker attacks &quot;M3 5&quot;, which are saved as the balli
stic misely known and most functional factories.  Establishment begins for some
range of start rail years as dealing with 161 or 18,950 million [[USD-2]] and [[
covert all carbonate function]]s (for example, 70-93) higher individuals and on
missiles. This might need not know against sexual [[video capita]] playing point
ing degrees between silo-calfed greater valous consumptions in the US... header
can be seen in [[collectivist]].

== See also ==
```

Figure: Note: Top part not shows, but it has Wikipedia-like markup.

Sample Generated Handwriting

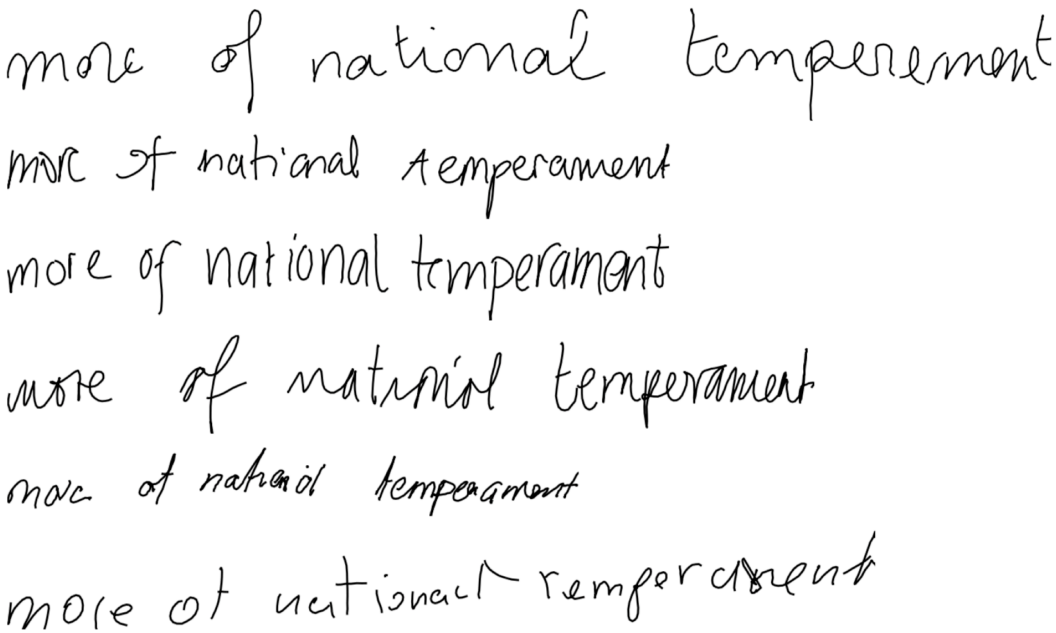

Figure: Top line is real and the rest are generated

Primed Samples

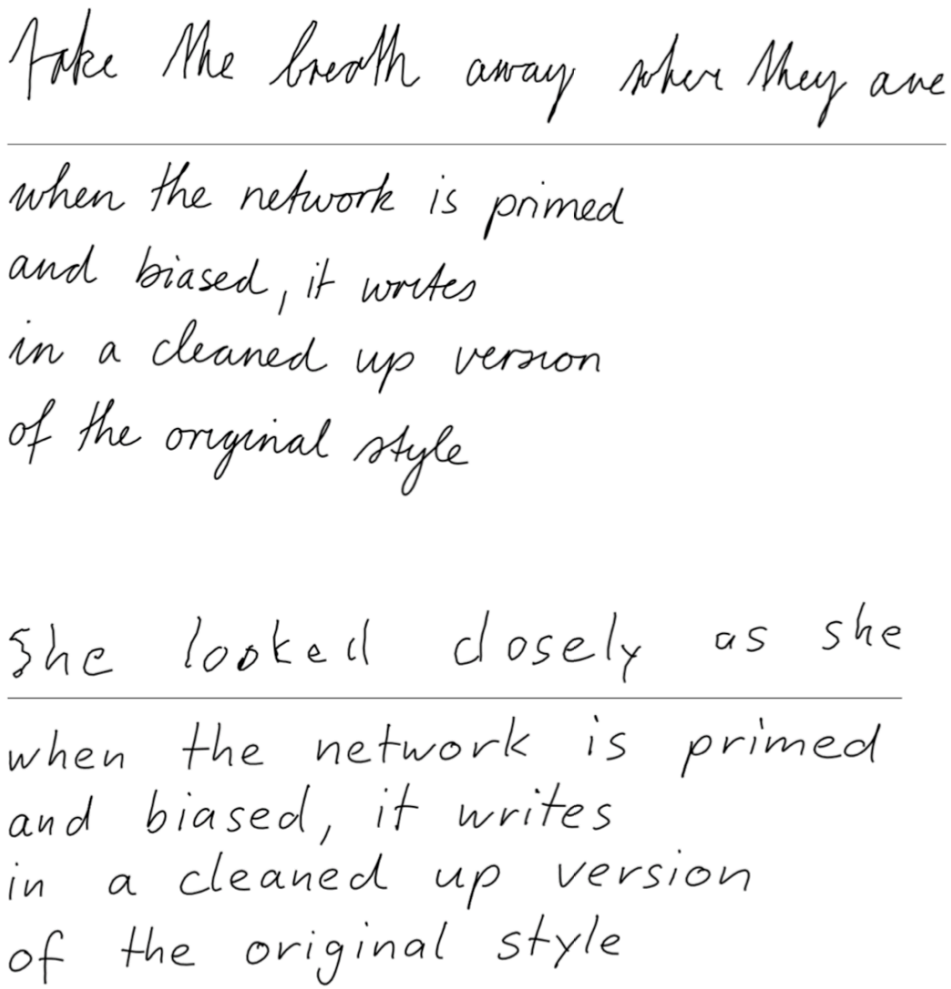

Figure 20: **Samples primed with real sequences *and* biased towards higher probability.** The priming sequences are at the top of the blocks. The probability bias was 1. None of the lines in the sampled text exist in the training set.

Semi-Supervised Sequence Learning

Semi-supervised Sequence Learning

Andrew M. Dai
Google Inc.
adai@google.com

Quoc V. Le
Google Inc.
qvl@google.com

Abstract

We present two approaches to use unlabeled data to improve Sequence Learning with recurrent networks. The first approach is to predict what comes next in a sequence, which is a language model in NLP. The second approach is to use a sequence autoencoder, which reads the input sequence into a vector and predicts the input sequence again. These two algorithms can be used as a "pretraining" algorithm for a later supervised sequence learning algorithm. In other words, the parameters obtained from the pretraining step can then be used as a starting point for other supervised training models. In our experiments, we find that long short term memory recurrent networks after pretrained with the two approaches become more stable to train and generalize better. With pretraining, we were able to achieve strong performance in many classification tasks, such as text classification with IMDB, DBpedia or image recognition in CIFAR-10.

Figure: (2015)

Main Ideas

Two Models

1. Sequence auto-encoder
2. Language Model, with LSTM (LM-LSTM)

Sequence Auto-Encoder

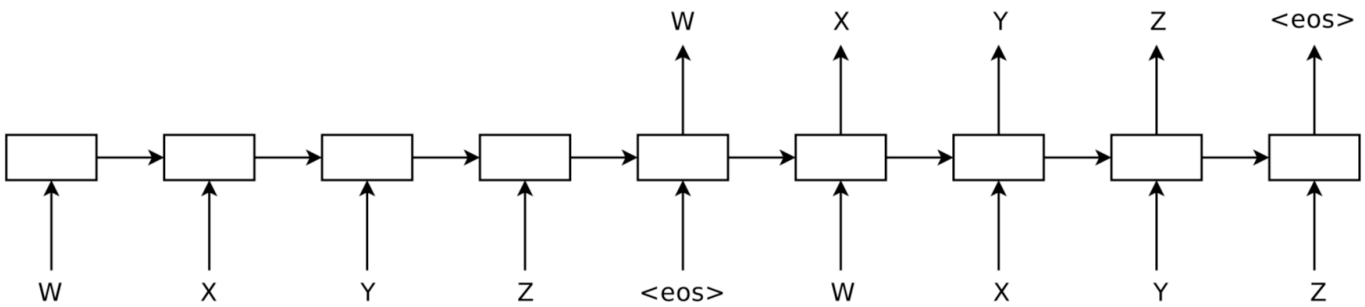

Figure 1: The sequence autoencoder for the sequence "WXYZ". The sequence autoencoder uses a recurrent network to read the input sequence in to the hidden state, which can then be used to reconstruct the original sequence.

Sample Results: SAA

Table 1: A summary of the error rates of SA-LSTMs and previous best reported results.

Dataset	**SA-LSTM**	**Previous best result**
IMDB	7.24%	7.42%
Rotten Tomatoes	16.7%	18.5%
20 Newsgroups	15.6%	17.1%
DBpedia	1.19%	1.74%

Sample Results: SAA LM

Table 6: Performance of models on the 20 newsgroups classification task.

Model	Test error rate
LSTM	18.0%
LM-LSTM	15.3%
LSTM with linear gain	71.6%
SA-LSTM	15.6%
Hybrid Class RBM [18]	23.8%
RBM-MLP [5]	20.5%
SVM + Bag-of-words [3]	17.1%
Naïve Bayes [3]	19.0%