Neural Network Language Models

Nicholas Dronen
ndronen@gmail.com

May 29, 2014
Language is unstructured
Supervised learning

\[ F(X, Y) \rightarrow (f(x) \rightarrow \hat{y}) \]

- \( X \): Training matrix
- \( Y \): Target vector
- \( x \): Feature vector
- \( \hat{y} \): Prediction
Language models (LMs)

- Language as a sequence
- Tasks
  - Training
    - Learn joint probability of word given context
  - Predicting
    - Compute probability of word and context
    - Generate sequences
LMs are unsupervised

\[ X \] Sequences of words
\[ x \] A sequence of words \( w \ w_{n-1} \ldots \)
\[ \hat{P}(x) \] Probability of \( w \) given context

\[ F(X) \rightarrow (f(x) \rightarrow \hat{P}(x)) \]
N-grams

Unigrams: N-grams with N=1.

\[ P(w_n) = \frac{C(w_n)}{\sum_w C(w)} \]
N-grams

Bigrams: N-grams with N=2.

\[ P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{\sum_w C(w_{n-1}w)} = \frac{C(w_{n-1}w_n)}{C(w_{n-1})} \]

Trigrams, quadrigrams, ....
N-gram example

this is the way the world ends
this is the way the world ends
dexter
this is the way the world ends
dexter
this is the way the bunny hops song

About 376,000,000 results (0.23 seconds)

not with a bang but a whimper
not with a bang but a whimper
not with a club the heart is broken
not with a whisper but with a bang
not with a bang damon knight

About 963,000 results (0.31 seconds)
## N-gram example

<table>
<thead>
<tr>
<th>Phrase</th>
<th>Counts (Google)</th>
</tr>
</thead>
<tbody>
<tr>
<td>This is the way the world</td>
<td>2,750,000</td>
</tr>
<tr>
<td>This is the way the world <strong>ends</strong></td>
<td>1,920,000</td>
</tr>
<tr>
<td>This is the way the world <strong>is</strong></td>
<td>537,000</td>
</tr>
<tr>
<td>This is the way the world <strong>turns</strong></td>
<td>319,000</td>
</tr>
<tr>
<td>This is the way the world <strong>works</strong></td>
<td>211,000</td>
</tr>
<tr>
<td>This is the way the world <strong>bling</strong></td>
<td>0</td>
</tr>
</tbody>
</table>
N-gram example (cont.)

\[ P(w | w_{n-1} \ldots) = P(\text{ends} \mid \text{this is the way the world}) \]

\[ = \frac{1,920,000}{2,750,000} \]

\[ = 0.69 \]
### Evaluating LMs

<table>
<thead>
<tr>
<th>Type</th>
<th>Loss function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regressor</td>
<td>$\frac{1}{N} \sum (y - \hat{y})^2$</td>
</tr>
<tr>
<td>Classifier</td>
<td>$\frac{1}{N} \sum I(y, \hat{y})$</td>
</tr>
<tr>
<td>Language model</td>
<td>$\sqrt{\sum_{i=1}^{N} \frac{1}{P(w_i</td>
</tr>
</tbody>
</table>
A problem

N-grams that don't occur in the training set cause N-gram probability to go to 0.

\[
P(w|w_{n-1} \ldots) = P(bling | this \ is \ the \ way \ the \ world) \\
= 0 / 2,750,000 \\
= 0
\]
Solution - Good-Turing

Steal from here

Number of occurrences
Frequency

0  10  20  30  40
0  900  800  700  600  500  400  300  200  100  0
Solution - Backoff

3-gram ? → 2-gram ? → 1-gram ?
Solution - Interpolation

\[ LM = \alpha_1 lm_1(w_n w_{n-1}) + \cdots + \alpha_n lm_2(w_n w_{n-1}) \]
Questions before moving to neural network (NN) LMs?
NN LMs are different

\( X \) Sequences of text

\( x \) Sequence of text \( w \ w_{n-1} \ldots \)

\( \hat{P}(x) \) Probability of \( w \) given context

\( W \) Matrix of word representations

\( \mathbf{F}(X) \rightarrow (f(x) \rightarrow \hat{P}(x), W) \)
Early neural network LM

A Neural Probabilistic Language Model, Bengio et al, 2003
Recurrent neural network based language model, Mikolov et al, 2010
Recurrent neural network LM

<table>
<thead>
<tr>
<th>Model</th>
<th># words</th>
<th>Perplexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>KN5</td>
<td>200K</td>
<td>336</td>
</tr>
<tr>
<td>KN5 + RNN</td>
<td>200K</td>
<td>271</td>
</tr>
<tr>
<td>KN5</td>
<td>1M</td>
<td>287</td>
</tr>
<tr>
<td>KN5 + RNN</td>
<td>1M</td>
<td>225</td>
</tr>
<tr>
<td>KN5</td>
<td>6.4M</td>
<td>221</td>
</tr>
<tr>
<td>KN5 + RNN</td>
<td>6.4M</td>
<td>156</td>
</tr>
</tbody>
</table>

Performance on WSJ dev set.
Learning representations

Linguistic Regularities in Continuous Space Word Representations, Mikolov et al, 2013
Properties of learned representations

Linguistic Regularities in Continuous Space Word Representations, Mikolov et al, 2013
A fast neural network LM

Efficient Estimation of Word Representations in Vector Space, Mikolov et al, 2013

Also try this online demo.
Questions?
Feel free to contact me at
ndronen@gmail.com