Sampling and Stochastic Search for Sentence Generation

Lili Mou
doublepower.mou@gmail.com
lili-mou.github.io
RNN Generation

The book is interesting

<SOS> The book is interesting <EOS>
RNN Generation

Question: Can we generate a sentence right-to-left?
Issues with Single Directional Generation

- Information bottleneck
- Error cumulation
  - Due to sampling or incompetency of the RNN

The book is interesting
movie
...
Generation by Local Changes

• Suppose we have a blueprint

The book is interesting <EOS>
Generation by Local Changes

• Suppose we have a blueprint

The book is interesting <EOS>

This
Generation by Local Changes

• Suppose we have a blueprint

The book is interesting <EOS>

This book is quite interesting <EOS>
Generation by Local Changes

• Suppose we have a blueprint

  The book is interesting <EOS>

  This book is interesting <EOS>

  quite fascinating
Applications

• Paraphrase generation
  - “Sample” a sentence with similar semantics but different wordings

• Summarization
  - “Sample” a sentence with similar semantics

• Grammatical error correction
  - “Sample” a more likely sentence with the same semantics
Sampling Methods
Independent Sampling

• Sampling from CDF
  - Probabilistic density function (PDF)
    \[ \Pr[a \leq x \leq b] = \int_{a}^{b} f(x) \, dx \]
  - Cumulative density function (CDF)
    \[ F(x) = \int_{-\infty}^{x} f(u) \, du = \Pr[u \leq x] \]
  - Sampling procedure
    \[ u \sim U[0,1]; \quad x = \text{CDF}^{-1}(u) \]

• Problems
  - CDF not analytic
  - Especially, the conditional CDF in multivariate cases
Independent Sampling

- Reject Sampling
  - To sample from $p(x) = \frac{1}{Z} \tilde{p}(x)$
  - We instead sample $x \sim q(x)$
  - Accept the sample $x$ with probability $\frac{\tilde{p}(x)}{k \cdot q(x)}$
    where $k$ is a constant s.t. $kq(x) \geq \tilde{p}(x), \forall x$
  - Reject $x$ w.p. $1 - \frac{\tilde{p}(x)}{k \cdot q(x)}$

- Many other sampling methods
Dependent Sampling

- Problem: Sample from $p(x)$

- MCMC sampling
  - Start from an arbitrary initial sample $x^{(0)}$
  - Sample $x^{(1)} \sim p(x^{(1)} | x^{(0)})$, $x^{(2)} \sim p(x^{(2)} | x^{(1)})$, ... 
  - Hope $p(x^{(n)}) \rightarrow p(x)$ as $n \rightarrow \infty$
Markov Chain

- States: \( S = \{s_1, s_2, \ldots\} \)

- Initial distribution \( \pi^{(0)} \)

- Transition probability: \( \mathcal{T}_{i \rightarrow j} = p(x^{(t+1)} = s_j | x^{(t)} = s_i) \)
  - \( x^{(t+1)} \) is independent of \( x^{(t-1)} \), given \( x^{(t)} \)
  - \( \mathcal{T}_{i \rightarrow j} \) works for all time steps \( t \)

- **Thm**: Starting from an arbitrary initial distribution, a Markov Chain converges to a **unique** stationary distribution (under mild assumptions).
Markov Chain Monte Carlo

- Problem: Sample from $p(x)$

- MCMC sampling
  - Start from an arbitrary initial sample $x^{(0)}$
  - Sample $x^{(1)} \sim p(x^{(1)} | x^{(0)})$, $x^{(2)} \sim p(x^{(2)} | x^{(1)})$, ... 
  - Hope $p(x^{(n)}) \to p(x)$ as $n \to \infty$
Markov Chain Monte Carlo

- Problem: Sample from \( p(x) \)

- MCMC sampling
  - Start from an arbitrary initial sample \( x^{(0)} \)
  - Sample \( x^{(1)} \sim p(x^{(1)} | x^{(0)}) \), \( x^{(2)} \sim p(x^{(2)} | x^{(1)}) \), \( \ldots \)
    by following a carefully designed Markov chain
  - Hope \( p(x^{(n)}) \rightarrow p(x) \) as \( n \rightarrow \infty \)

Guaranteed that
Metropolis—Hastings Sampler

• **Input**
  - An arbitrary desired distribution $p(x)$

• **Output**
  - An unbiased sample $x \sim p(x)$

• **Algorithm**
  - Start from an arbitrary initial state $x^{(0)}$
  - For every step $t$
    - Propose a new state $x' \sim g(x' | x^{(t)})$
    - Accept $x'$ w.p. $A(x' | x) = \min \left\{ 1, \frac{p(x')g(x^{(t)} | x')}{p(x)g(x' | x^{(t)})} \right\}$, i.e., $x^{(t)} = x'$
    - Reject $x'$ otherwise, i.e., $x^{(t+1)} = x^{(t)}$
  - Return $x^{(t)}$ with a large $t$
Proof Sketch

- Detailed balance property $\Rightarrow$ Stationary distribution

If

$$\forall x, y, \quad \pi(x) \cdot \mathcal{T}_{x \rightarrow y} = \pi(y) \cdot \mathcal{T}_{y \rightarrow x}$$

Then

$\pi(x)$ is a stationary distribution

Because

$$\forall x, \quad \pi(x) = \sum_y \pi(y) \mathcal{T}_{y \rightarrow x} = \sum_y \pi(x) \mathcal{T}_{x \rightarrow y} = \pi(x)$$
Proof Sketch (Cont.)

- MH Sampler satisfies detailed balance

\[ \forall x, y, \quad p(x) \cdot \mathcal{T}_{x \rightarrow y} = p(x) \cdot g(y | x) \cdot \min \left\{ 1, \frac{p(y)g(x | y)}{p(x)g(y | x)} \right\} \quad (1) \]

\[ p(y) \cdot \mathcal{T}_{y \rightarrow x} = p(y) \cdot g(x | y) \cdot \min \left\{ 1, \frac{p(x)g(y | x)}{p(y)g(x | y)} \right\} \quad (2) \]

- W.L.O.G., we assume \( p(x)g(y | x) \geq p(y)g(x | y) \)

\( (1) = p(y) \cdot g(x | y) \)

\( (2) = p(y) \cdot g(x | y) \)
Gibbs Sampler

- Suppose \( \mathbf{x} = (x_1, x_2, \cdots, x_{i-1}, x_i, x_{i+1}, \cdots, x_n) \)

- If the proposal distribution is \( x'_i \sim p(x_i | \mathbf{x}_{-i}) \)

- Then, the acceptance rate is \( A(x'|x) = \min \left\{ 1, \frac{p(x')g(x | x')}{p(x)g(x'|x)} \right\} \)

  - Notice that \( x' = (x_1, x_2, \cdots, x_{i-1}, x'_i, x_{i+1}, \cdots, x_n) \)

  - Thus, \( \frac{p(x')g(x | x')}{p(x)g(x'|x)} = \frac{p(x_{-i})p(x'_i | x_{-i}) \cdot p(x_i | x_{-i})}{p(x_{-i})p(x_i | x_{-i}) \cdot p(x'_i | x_{-i})} = 1 \)

=> Gibbs step is a special case of an MH step, with AC rate = 1
Applying MH to Sentence Generation
MH Components

- State: Every sentence

- Target distribution: Depend on the task

- Proposal distribution
  - Task agnostic, or task specific

- Compute acceptance rate
  - We can’t do anything here
Target distribution

- General formula
  - \( p(x) \propto p_{LM}(x) \cdot s_1(x) \cdots s_n(x) \)
  - \( s_i(x) \): scoring functions specific to the task
Target distribution

- General formula
  \[ p(x) \propto p_{LM}(x) \cdot s_1(x) \cdots s_n(x) \]
  - \( s_i(x) \): scoring functions specific to the task

- Keywords-to-sentence generation
  \[ s(x) = \begin{cases} 
  1, & \text{if keywords in } x \\
  0, & \text{otherwise} 
\end{cases} \]

- Paraphrase generation/Grammatical error correction
  \[ s(x) = \text{sim}_{\text{semantic}}(x, x_0) + \text{diff}_{\text{word}}(x, x_0) \]
Proposal Distribution

• Replace

\[ g_{\text{replace}}(x'|x) = \pi(w^*_m = w^c|x_{-m}) = \]
\[ \frac{\pi(w_1, \ldots, w_{m-1}, w^c, w_{m+1}, \ldots, w_n)}{\sum_{w \in \mathcal{V}} \pi(w_1, \ldots, w_{m-1}, w, w_{m+1}, \ldots, w_n)} \]

• Delete

• Insert

- Also sample from posterior
### Examples: Keywords-to-Sentence

<table>
<thead>
<tr>
<th>Keyword(s)</th>
<th>Generated Sentences</th>
</tr>
</thead>
<tbody>
<tr>
<td>friends</td>
<td>My good <strong>friends</strong> were in danger .</td>
</tr>
<tr>
<td>project</td>
<td>The first <strong>project</strong> of the scheme .</td>
</tr>
<tr>
<td>have, trip</td>
<td>But many people <strong>have</strong> never made the <strong>trip</strong> .</td>
</tr>
<tr>
<td>lottery, scholarships</td>
<td>But the <strong>lottery</strong> has provided <strong>scholarships</strong> .</td>
</tr>
<tr>
<td>decision, build, home</td>
<td>The <strong>decision</strong> is to <strong>build</strong> a new <strong>home</strong> .</td>
</tr>
<tr>
<td>attempt, copy, painting, denounced</td>
<td>The first <strong>attempt</strong> to <strong>copy</strong> the <strong>painting</strong> was <strong>denounced</strong> .</td>
</tr>
</tbody>
</table>
## Examples: Paraphrase Generation

<table>
<thead>
<tr>
<th>Model</th>
<th>BLEU-ref</th>
<th>BLEU-ori</th>
<th>NLL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin Sentence</td>
<td>30.49</td>
<td>100.00</td>
<td>7.73</td>
</tr>
<tr>
<td>VAE-SVG (100k)</td>
<td>22.50</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VAE-SVG-eq (100k)</td>
<td>22.90</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VAE-SVG (50k)</td>
<td>17.10</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VAE-SVG-eq (50k)</td>
<td>17.40</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Seq2seq (100k)</td>
<td>22.79</td>
<td>33.83</td>
<td>6.37</td>
</tr>
<tr>
<td>Seq2seq (50k)</td>
<td>20.18</td>
<td>27.59</td>
<td>6.71</td>
</tr>
<tr>
<td>Seq2seq (20k)</td>
<td>16.77</td>
<td>22.44</td>
<td>6.67</td>
</tr>
<tr>
<td>VAE (unsupervised)</td>
<td>9.25</td>
<td>27.23</td>
<td>7.74</td>
</tr>
<tr>
<td>CGMH w/o matching</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>w/ KW</td>
<td>18.85</td>
<td>50.28</td>
<td>7.52</td>
</tr>
<tr>
<td>w/ KW + WVA</td>
<td>20.17</td>
<td>53.15</td>
<td>7.57</td>
</tr>
<tr>
<td>w/ KW + WVM</td>
<td>20.41</td>
<td>53.64</td>
<td>7.57</td>
</tr>
<tr>
<td>w/ KW + ST</td>
<td>20.89</td>
<td>54.96</td>
<td>7.46</td>
</tr>
<tr>
<td>w/ KW + ST</td>
<td>20.70</td>
<td>54.50</td>
<td>7.78</td>
</tr>
</tbody>
</table>

### Type | Examples
--- | ---
Ori | what’s the best plan to lose weight
Ref | what is a good diet to lose weight
Gen | what’s the best way to slim down quickly
Ori | how should i control my emotion
Ref | how do i control anger and impulsive emotions
Gen | how do i control my anger
Ori | why do my dogs love to eat tuna fish
Ref | why do my dogs love eating tuna fish
Gen | why do some dogs like to eat raw tuna and raw fish
Examples: Paraphrase Generation

<table>
<thead>
<tr>
<th>Model</th>
<th>#parallel data</th>
<th>GLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMU</td>
<td>2.3M</td>
<td>44.85</td>
</tr>
<tr>
<td>CAMB-14</td>
<td>155k</td>
<td>46.04</td>
</tr>
<tr>
<td>MLE</td>
<td>720k</td>
<td>52.75</td>
</tr>
<tr>
<td>NRL</td>
<td>720k</td>
<td>53.98</td>
</tr>
<tr>
<td>CGMH</td>
<td>0</td>
<td>45.5</td>
</tr>
</tbody>
</table>

Ori | Even if **we are failed**, We have to try to get a **new things**.
Ref | Even if we all failed, we have to try to get new things.
Gen | Even if we are failing, We have to try to get some new things.

Ori | In the world **oil price very high** right now.
Ref | In today ’s world, oil prices are very high right now.
Gen | In the world, oil prices are very high right now.
Figure 3: Overlap rates of CGMH and VAE for each word position of sentences.
Figure 2: Generation quality with corrupted initial states. At each situation, 0/5%/10%/100% of the words in initial sentences are randomly replaced with other words.
Summary (Take-Home Msg)

- MCMC: Dependent sampling with a Markov chain
  - Gibbs sampler: posterior sampling
  - MH sampler: propose-and-reject sampling
- MH sentence generation
  - The framework is the same
  - Need to design target and proposal distributions
  - Various applications: paraphrase generation, grammatical error correction, keywords-to-sentence generation, etc.

(A blueprint is needed for MH generation)
Thank you!

Q&A