Search-Based Unsupervised Text Generation

Lili Mou

Dept. Computing Science, University of Alberta
Alberta Machine Intelligence Institute (Amii)

doublepower.mou@gmail.com
Outline

• Introduction

• General framework

• Applications
  - Paraphrasing
  - Summarization
  - Text simplification

• Conclusion & Future Work
A fading memory ...

- Of how I learned natural language processing (NLP):

\[
\text{NLP} = \text{NLU} + \text{NLG}
\]

- NLU was the main focus of NLP research.

- NLG was relatively easy, as we can generate sentences by rules, templates, etc.

- Why this may NOT be correct?

  - Rules and templates are not natural language.

  - How can we represent meaning? — Almost the same question as NLU.
A fading memory …

• Of how I learned natural language processing (NLP):

NLP = NLU + NLG
  **Understanding**  **Generation**

  - NLU was the main focus of NLP research.
  
  - NLG was relatively easy, as we can generate sentences by rules, templates, etc.

• Why this may NOT be correct?

  - Rules and templates are not natural language.
  
  - How can we represent meaning? — Almost the same question as NLU.
Why NLG is interesting?

- Industrial applications
- Machine translation
- Headline generation for news
- Grammarly: grammatical error correction

https://translate.google.com/
Why NLG is interesting?

- Industrial applications
  - Machine translation
  - Headline generation for news
  - Grammarly: grammatical error correction

- Scientific questions
  - Non-linear dynamics for long-text generation
  - Discrete “multi-modal” distribution
Supervised Text Generation

Sequence-to-sequence training

Training data = \{ (x^{(m)}, y^{(m)}) \}_{m=1}^M

known as a parallel corpus

Reference/target sentence

\begin{align*}
& y_1 & y_2 & y_3 \\
\end{align*}

Predicted sentence

\begin{align*}
& \hat{y}_1 & \hat{y}_2 & \hat{y}_3 \\
\end{align*}

Sequence-aggregated Cross-entropy loss
Unsupervised Text Generation

• Training data = \{x^{(m)}\}_{m=1}^M
  - Not even training (we did it by searching)

• Important to industrial applications
  - Startup: No data
  - Minimum viable product

• Scientific interest
  - How can AI agents go beyond NLU to NLG?
  - Unique search problems
General Framework
General Framework

• Search objective
  - Scoring function measuring text quality

• Search algorithm
  - Currently we are using stochastic local search
Scoring Function

- Search objective
  - Scoring function measuring text quality
    \[ s(y) = s_{LM}(y) \cdot s_{Semantic}(y)^{\alpha} \cdot s_{Task}(y)^{\beta} \]
- Language fluency
- Semantic coherence
- Task-specific constraints
Scoring Function

- **Search objective**
  - Scoring function measuring text quality
    \[ s(y) = s_{LM}(y) \cdot s_{Semantic}(y)^\alpha \cdot s_{Task}(y)^\beta \]

- **Language fluency**
  - Language model estimates the “probability” of a sentence
    \[ \tilde{\text{PPL}}(y) = 2^{2|y|} \left( \prod_i \frac{1}{p_{LM}(y_i|y_{<i})} \right) \left( \prod_i \frac{1}{p_{LM}(y_i|y_{>i})} \right) \]
    \[ s_{LM}(y) = \text{PPL}(y)^{-1} \]

- **Semantic coherence**

- **Task-specific constraints**
Scoring Function

• Search objective
  - Scoring function measuring text quality
    \[ s(y) = s_{LM}(y) \cdot s_{Semantic}(y)^\alpha \cdot s_{Task}(y)^\beta \]

• Language fluency

• **Semantic coherence**
  \[ s_{semantic} = \cos(e(y), e(y)) \]

• Task-specific constraints
Scoring Function

- Search objective
  - Scoring function measuring text quality
    \[ s(y) = s_{LM}(y) \cdot s_{Semantic}(y)^\alpha \cdot s_{Task}(y)^\beta \]

- Language fluency

- Semantic coherence

- Task-specific constraints
  - Paraphrasing: lexical dissimilarity with input
  - Summarization: length budget
Search Algorithm

• Observations:
  - The output closely resembles the input
  - Edits are mostly local
  - May have hard constraints

• Thus, we mainly used **local stochastic search**
Search Algorithm

(stochastic local search)

Start with $y_0$  # an initial candidate sentence

Loop within budget at step $t$:

$$y' \sim \text{Neighbor}(y_t)$$  # a new candidate in the neighbor

Either reject or accept $y'$

If accepted, $y_t = y'$, or otherwise $y_t = y_{t-1}$

Return the best scored $y^*$
Search Algorithm

Local edits for $y' \sim \text{Neighbor}(y_t)$

- General edits
  - Word deletion
  - Word insertion
  - Word replacement

- Task specific edits
  - Reordering, swap of word selection, etc.

\[
p(w_*|\cdot) = \frac{f_{\text{sim}}(x_*, x_0) \cdot f_{\text{exp}}(x_*, x_0) \cdot f_{\text{flu}}(x_*)}{Z},
\]

\[
Z = \sum_{w_* \in \mathcal{W}} f_{\text{sim}}(x_*, x_0) \cdot f_{\text{exp}}(x_*, x_0) \cdot f_{\text{flu}}(x_*),
\]

Gibbs in Metropolis
Search Algorithm

**Example:** Metropolis—Hastings sampling

Start with $y_0$  # an initial candidate sentence

Loop within your budget at step $t$:

$y' \sim \text{Neighbor}(y_t)$  # a new candidate in the neighbor

Either reject or accept $y'$

If accepted, $y_t = y'$, or otherwise $y_t = y_{t-1}$

Return the best scored $y^*$
Search Algorithm

**Example:** Simulated annealing

Start with $y_0$  # an initial candidate sentence

Loop within your budget at step $t$:

$$y' \sim \text{Neighbor}(y_t) \quad \# \text{a new candidate in the neighbor}$$

Either reject or accept $y'$

$$p(\text{accept}|x_*, x_t, T) = \min \left( 1, e^{\frac{f(x_*) - f(x_t)}{T}} \right)$$

If accepted, $y_t = y'$, or otherwise $y_t = y_{t-1}$

Return the best scored $y_*$
Search Algorithm

Example: Hill climbing

Start with $y_0$  # an initial candidate sentence

Loop within your budget at step $t$:

$$y' \sim \text{Neighbor}(y_t)$$  # a new candidate in the neighbor

Either reject or accept $y'$  

If accepted, $y_t = y'$, or otherwise $y_t = y_{t-1}$

Return the best scored $y^*$
Applications
## Paraphrase Generation

<table>
<thead>
<tr>
<th>Input</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which is the best training institute in Pune for digital marketing?</td>
<td>Which is the best digital marketing training institute in Pune?</td>
</tr>
</tbody>
</table>

Could be useful for various NLP applications

- E.g., query expansion, data augmentation
Paraphrase Generation

- Search objective
  - Fluency
  - Semantic preservation
  - Expression diversity
    - The paraphrase should be different from the input
      \[ s_{exp}(y^*, y_0) = 1 - \text{BLEU}(y^*, y_0) \]
      BLEU here measures the \( n \)-gram overlapping

- Search algorithm
- Search space
- Search neighbors
Paraphrase Generation

- Search objective
  - Fluency
  - Semantic preservation
  - Expression diversity
    - The paraphrase should be different from the input

$$s_{exp}(y^*, y_0) = 1 - \text{BLEU}(y^*, y_0)$$

BLEU here measures the $n$-gram overlapping

- Search algorithm: Simulated annealing

- Search space: the entire sentence space with $y_0 = \text{input}$

- Search neighbors
  - Generic word deletion, insertion, and replacement
  - Copying words in the input sentence
### Text Simplification

<table>
<thead>
<tr>
<th>Input</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>In 2016 alone, American developers had spent 12 billion dollars on constructing theme parks, according to a Seattle based reporter.</td>
<td>American developers had spent 12 billion dollars in 2016 alone on building theme parks.</td>
</tr>
</tbody>
</table>

**Could be useful for**

- education purposes (e.g., kids, foreigners)
- for those with dyslexia

**Key observations**

- Dropping phrases and clauses
- Phrase re-ordering
- Dictionary-guided lexicon substitution
Text Summarization

Search objective
- Language model fluency (discounted by word frequency)
- Cosine similarity
- Entity matching
- Length penalty
- Flesh Reading Ease (FRE) score [Kincaid et al., 1975]

Search operations
Text Summarization

Search objective
- Language model fluency (discounted by word frequency)
- Cosine similarity
- Entity matching
- Length penalty
- Flesh Reading Ease (FRE) score [Kincaid et al., 1975]

Search operations
- Dictionary-guided substitution (e.g., WordNet)
- Phrase removal
- Re-ordering with parse trees
Text Summarization

Key observation

- Words in the summary mostly come from the input
- If we generate the summary by selecting words, we have

  bhp billiton dropping hostile bid for rio tinto
Text Summarization

• Search objective
  – Fluency
  – Semantic preservation
  – A hard length constraint

\[
\begin{align*}
  f_{\text{LEN}}(y; s) &= \begin{cases}
    1, & \text{if } |y| = s, \\
    -\infty, & \text{otherwise}.
  \end{cases}
\end{align*}
\]

(Explicitly controlling length is not feasible in previous work)

• Search space

• Search neighbor

• Search algorithm
Text Summarization

- Search objective
  - Fluency
  - Semantic preservation
  - A hard length constraint

\[ f_{\text{LEN}}(y; s) = \begin{cases} 
1, & \text{if } |y| = s, \\
-\infty, & \text{otherwise.}
\end{cases} \]

(Explicitly controlling length is not feasible in previous work)

- Search space with only feasible solutions

\[ |\mathcal{X}| |y| \implies \left( \begin{array}{c}
|x| \\
s
\end{array} \right) \]

- Search neighbor: swap only

- Search algorithm: hill-climbing
Experimental Results
Research Questions

• General performance
• Greediness vs. Stochasticity
• Search objective vs. Measure of success
## General Performance

### Paraphrase generation

<table>
<thead>
<tr>
<th>Model</th>
<th>Quora</th>
<th>Wikianswers</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>iBLEU</td>
<td>BLEU</td>
<td>Rouge1</td>
<td>Rouge2</td>
</tr>
<tr>
<td>ResidualLSTM</td>
<td>12.67</td>
<td>17.57</td>
<td>59.22</td>
<td>32.40</td>
</tr>
<tr>
<td>VAE-SVG-eq</td>
<td>15.17</td>
<td>20.04</td>
<td>59.98</td>
<td>33.30</td>
</tr>
<tr>
<td>Pointer-generator</td>
<td>16.79</td>
<td>22.65</td>
<td>61.96</td>
<td>36.07</td>
</tr>
<tr>
<td>Transformer</td>
<td>16.25</td>
<td>21.73</td>
<td>60.25</td>
<td>33.45</td>
</tr>
<tr>
<td>Transformer+Copy</td>
<td>17.98</td>
<td>24.77</td>
<td>63.34</td>
<td>37.31</td>
</tr>
<tr>
<td>DNPG</td>
<td><strong>18.01</strong></td>
<td><strong>25.03</strong></td>
<td><strong>63.73</strong></td>
<td><strong>37.75</strong></td>
</tr>
<tr>
<td>Supervised</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pointer-generator</td>
<td>5.04</td>
<td>6.96</td>
<td>41.89</td>
<td>12.77</td>
</tr>
<tr>
<td>Transformer+Copy</td>
<td>6.17</td>
<td>8.15</td>
<td>44.89</td>
<td>14.79</td>
</tr>
<tr>
<td>Shallow fusion</td>
<td>6.04</td>
<td>7.95</td>
<td>44.87</td>
<td>14.79</td>
</tr>
<tr>
<td>MTL</td>
<td>4.90</td>
<td>6.37</td>
<td>37.64</td>
<td>11.83</td>
</tr>
<tr>
<td>MTL+Copy</td>
<td>7.22</td>
<td>9.83</td>
<td>47.08</td>
<td>19.03</td>
</tr>
<tr>
<td>DNPG</td>
<td>10.39</td>
<td>16.98</td>
<td>56.01</td>
<td>28.61</td>
</tr>
<tr>
<td>Supervised + Domain-adapted</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VAE</td>
<td>8.16</td>
<td>13.96</td>
<td>44.55</td>
<td>22.64</td>
</tr>
<tr>
<td>Lag VAE</td>
<td>8.73</td>
<td>15.52</td>
<td>49.20</td>
<td>26.07</td>
</tr>
<tr>
<td>CGMH</td>
<td>9.94</td>
<td>15.73</td>
<td>48.73</td>
<td>26.12</td>
</tr>
<tr>
<td>UPSA</td>
<td>12.03</td>
<td>18.21</td>
<td>59.51</td>
<td>32.63</td>
</tr>
</tbody>
</table>

**BLEU and ROUGE scores are automatic evaluation metrics based on references**
## General Performance

### Text Summarization

<table>
<thead>
<tr>
<th>Model</th>
<th>Data</th>
<th>Len D</th>
<th>Rouge F1</th>
<th>Len O</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>article</td>
<td>title</td>
<td>external</td>
<td>R-1</td>
</tr>
<tr>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lead-N-8</td>
<td>✓</td>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>HC_article_8</td>
<td>✓</td>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>HC_title_8</td>
<td>✓</td>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>HC_article_10</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>Wang and Lee (2018)</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>Zhou and Rush (2019)</td>
<td>✓</td>
<td>✓</td>
<td>billion</td>
<td>-</td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lead-P-50</td>
<td>✓</td>
<td></td>
<td></td>
<td>50%</td>
</tr>
<tr>
<td>Fevry and Phang (2018)</td>
<td>✓</td>
<td></td>
<td>SNLI</td>
<td>50%</td>
</tr>
<tr>
<td>Baziotis et al. (2019)</td>
<td>✓</td>
<td></td>
<td></td>
<td>50%</td>
</tr>
<tr>
<td>HC_article_50p</td>
<td>✓</td>
<td></td>
<td></td>
<td>50%</td>
</tr>
<tr>
<td>HC_title_50p</td>
<td>✓</td>
<td></td>
<td></td>
<td>50%</td>
</tr>
</tbody>
</table>

Note: Len D refers to the length of the generated summary, Rouge F1 measures the quality of the summary, and Len O refers to the length of the original text.
## General Performance

### Text Simplification

<table>
<thead>
<tr>
<th>Method</th>
<th>BLEU</th>
<th>SARI</th>
<th>Add</th>
<th>Delete</th>
<th>Keep</th>
<th>GM</th>
<th>FKGL</th>
<th>Len</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>100</td>
<td>70.13</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>83.74</td>
<td>3.20</td>
<td>12.75</td>
</tr>
</tbody>
</table>

**Baselines**

<table>
<thead>
<tr>
<th>Method</th>
<th>BLEU</th>
<th>SARI</th>
<th>Add</th>
<th>Delete</th>
<th>Keep</th>
<th>GM</th>
<th>FKGL</th>
<th>Len</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complex</td>
<td>21.30</td>
<td>2.82</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>7.75</td>
<td>8.62</td>
<td>23.06</td>
</tr>
<tr>
<td>Reduced-250</td>
<td>11.79</td>
<td>28.39</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>18.29</td>
<td>-0.23</td>
<td>14.48</td>
</tr>
</tbody>
</table>

**Supervised Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>BLEU</th>
<th>SARI</th>
<th>Add</th>
<th>Delete</th>
<th>Keep</th>
<th>GM</th>
<th>FKGL</th>
<th>Len</th>
</tr>
</thead>
<tbody>
<tr>
<td>PBMT-R</td>
<td>18.1</td>
<td>15.77</td>
<td>3.07</td>
<td>38.34</td>
<td>5.90</td>
<td>16.89</td>
<td>7.59</td>
<td>23.06</td>
</tr>
<tr>
<td>Hybrid</td>
<td>14.46</td>
<td>28.61*</td>
<td>0.95</td>
<td>78.86*</td>
<td>6.01*</td>
<td>20.34</td>
<td>4.03</td>
<td>12.41</td>
</tr>
<tr>
<td>EncDecA</td>
<td>21.68</td>
<td>24.12</td>
<td>2.73</td>
<td>62.66</td>
<td>6.98</td>
<td>22.87</td>
<td>5.11</td>
<td>16.96</td>
</tr>
<tr>
<td>Dress</td>
<td>23.2</td>
<td>27.37</td>
<td><strong>3.08</strong></td>
<td>71.61</td>
<td>7.43</td>
<td>25.2</td>
<td>4.11</td>
<td>14.2</td>
</tr>
<tr>
<td>Dress-Ls</td>
<td>24.25</td>
<td>26.63</td>
<td>3.21</td>
<td>69.28</td>
<td>7.4</td>
<td>25.41</td>
<td>4.21</td>
<td>14.37</td>
</tr>
<tr>
<td>DMass</td>
<td>11.92</td>
<td>31.06</td>
<td>1.25</td>
<td>84.12</td>
<td>7.82</td>
<td>19.24</td>
<td>3.60</td>
<td>15.07</td>
</tr>
<tr>
<td>S2S-All-FA</td>
<td>19.55</td>
<td>30.73</td>
<td>2.64</td>
<td>81.6</td>
<td><strong>7.97</strong></td>
<td>24.51</td>
<td>2.60</td>
<td>10.81</td>
</tr>
<tr>
<td>Edit-NTS</td>
<td>19.85</td>
<td>30.27*</td>
<td>2.71*</td>
<td>80.34*</td>
<td>7.76*</td>
<td>24.51</td>
<td>3.41</td>
<td>10.92</td>
</tr>
<tr>
<td>EncDecP</td>
<td>23.72</td>
<td>28.31</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>25.91</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>EntPar</td>
<td>11.14</td>
<td><strong>33.22</strong></td>
<td>2.42</td>
<td><strong>89.32</strong></td>
<td>7.92</td>
<td>19.24</td>
<td>1.34</td>
<td>7.88</td>
</tr>
</tbody>
</table>

**Unsupervised Methods (Ours)**

<table>
<thead>
<tr>
<th>Method</th>
<th>BLEU</th>
<th>SARI</th>
<th>Add</th>
<th>Delete</th>
<th>Keep</th>
<th>GM</th>
<th>FKGL</th>
<th>Len</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base</td>
<td><strong>27.22</strong></td>
<td>26.07</td>
<td>2.35</td>
<td>68.35</td>
<td>7.5</td>
<td>26.64</td>
<td>2.95</td>
<td>12.9</td>
</tr>
<tr>
<td>Base+LS</td>
<td>27.17</td>
<td>26.26</td>
<td>2.28</td>
<td>68.94</td>
<td>7.57</td>
<td><strong>26.71</strong></td>
<td>2.93</td>
<td>12.88</td>
</tr>
<tr>
<td>Base+RO</td>
<td>26.31</td>
<td>26.99</td>
<td>2.47</td>
<td>70.88</td>
<td>7.63</td>
<td>26.64</td>
<td>3.14</td>
<td>12.81</td>
</tr>
<tr>
<td>Base+LS+RO</td>
<td>26.21</td>
<td>27.11</td>
<td>2.40</td>
<td>71.26</td>
<td>7.67</td>
<td>26.66</td>
<td>3.12</td>
<td>12.81</td>
</tr>
</tbody>
</table>
## General Performance

Human evaluation on paraphrase generation

<table>
<thead>
<tr>
<th>Model</th>
<th>Relevance</th>
<th>Fluency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean Score</td>
<td>Agreement</td>
</tr>
<tr>
<td>VAE</td>
<td>2.65</td>
<td>0.41</td>
</tr>
<tr>
<td>Lag VAE</td>
<td>2.81</td>
<td>0.45</td>
</tr>
<tr>
<td>CGMH</td>
<td>3.08</td>
<td>0.36</td>
</tr>
<tr>
<td>UPSA</td>
<td><strong>3.78</strong></td>
<td>0.55</td>
</tr>
</tbody>
</table>
### General Performance

#### Examples

<table>
<thead>
<tr>
<th>Input</th>
<th>VAE</th>
<th>Lag VAE</th>
<th>CGMH</th>
<th>UPSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>where are best places for spring snowboarding in the us?</td>
<td>where are best places for running in the world? (3.33)</td>
<td>where are best places for honeymoon year near the us? (2.33)</td>
<td>Where is best store for the snowboarding in the US? (3.67)</td>
<td>Where can I find the best places in the US for snowboarding? (4.67)</td>
</tr>
<tr>
<td>how can I become good in studies?</td>
<td>how can i have a good android phone? (2.33)</td>
<td>how can i become good students? (4.33)</td>
<td>how can i become very rich in studies? (4.00)</td>
<td>how should i do to get better grades in my studies? (4.33)</td>
</tr>
<tr>
<td>what are the pluses and minuses about life as a foreigner in singapore?</td>
<td>what are the UNK and most interesting life as a foreigner in medieval greece? (2.33)</td>
<td>what are the UNK and interesting things about life as a foreigner? (2.33)</td>
<td>what are the misconception about UNK with life as a foreigner in western? (2.33)</td>
<td>what are the mistakes and pluses life as a foreigner in singapore? (2.67)</td>
</tr>
</tbody>
</table>

#### Main conclusion

- Search-based unsupervised text generation works in a variety of applications

- Surprisingly, it does yield **fluent sentences**.
Greediness vs Stochasticity

Paraphrase generation

Findings:

- Greedy search ≺ Simulated annealing
- Sampling ≺ stochastic search
Search Objective vs. Measure of Success

Experiment: summarization by word selection

- Exhaustive search does yield higher scores $s(y)$
- Exhaustive search does NOT yield higher measure of success (ROUGE)
Conclusion & Future Work
Search-based unsupervised text generation

General framework

• Search objective
  – fluency, semantic coherence, etc.
• Search space
  – word generation from the vocabulary, word selection
• Search algorithm
  – Local search with word-based edit
  – MH, SA, and hill climbing

Applications

– Paraphrasing, summarization, simplification
Future Work

Defining the search neighborhood

**Input:** What would you do *if given the* power to become invisible?

**Output:** What would you *do when you have* the power to be invisible?

Current progress

- Large edits are possibly due to the less greedy SA but are rare

Future work

- Phrase-based edit (combining discrete sampling with VAE)
- Syntax-based edit (making use of probabilistic CFG)
Future Work

Initial state of the local search

Current applications

- Paraphrasing, summarization, text simplification, grammatical error correction
- Input and desired output closely resemble each other

Future work

- Dialogue systems, machine translation, etc.
- Designing initial search state for general-purpose TextGen
- Combining retrieval-based methods
Future Work

Combining search and learning

Disadvantage of search-only approaches
- Efficiency: 1—2 seconds per sample
- Heuristically defined objective may be deterministically wrong

Future work
- MCTS (currently exploring)
- Difficulties: large branching factor, noisy reward


Acknowledgments

Lili Mou is supported by AltaML, Amii Fellow Program, and Canadian CIFAR AI Chair Program.
Q&A

Thanks for listening!