A rat named Magawa is being honored with one of the highest awards in the animal world. He has potentially saved numerous lives for clearing landmines from fields in Cambodia.

A rat in Cambodia has a very special job. He helps make things safer for people. The rat's name is Magawa. He received a gold medal for his work. It is the highest award in the animal world. He has possibly saved many lives.

Example credit: USA Today and Newsela (2020/08/11).
What is so difficult ....

with text simplification?
What is so difficult?

Existing datasets:

- Simple Wikipedia (large but noisy)

https://simple.wikipedia.org/
What is so difficult?

Existing datasets:

- Simple Wikipedia (large but noisy)
- Newsela (high-quality, small, hard to align)

https://newsela.com/about/resources/research/
What is so difficult?

Existing datasets:
- Simple Wikipedia (large but noisy)
- Newsela (high-quality, small, hard to align)

Supervised seq2seq is limited.

Do we really need data?
What if we approach Text Simplification in an **unsupervised** way:

1. Define a “text simplification” reward

2. Train a pre-trained language model to optimize the reward.
Talk Outline

1. Reward Design
2. Optimization
3. Evaluation
1

Reward
Adapting the Summary Loop to the domain of Text Simplification

Objective: The generated text should contain the same information as the original text.
Key Idea: Adapting the Coverage model from the Summary Loop (ACL 2020) to text simplification.
Keep it Simple: Fluency

**Objective:** The generated text should be grammatical and be in fluent English.
Keep it Simple: Fluency

Fluency Component 1: Pretrained-Language Model

\[
\text{GPT2Score("A rat in Cambodia has a very special job..."')} = 0.98
\]

\[
\text{GPT2Score("landmines honor Cambodia Magawa rat saving..."')} = 0.01
\]

\[\uparrow\]

*Keyword soup gets low fluency score*

The language model checks that word sequences *feel like* sentences.
Fluency Component 1: Pretrained-Language Model

**Problem:** the fluency model is static (it does not change during training). The generator can learn *common patterns* to artificially score high on the fluency metric.
**Keep it Simple: Fluency**

**Fluency Component 1:** Pretrained-Language Model

**Fluency Component 2:** Dynamic Discriminator (Adversarial)

“Real” Corpus Sentences  
*(label = 1)*

Generated Sentences  
*(label = 0)*

Discriminator

Score(text) = P(Y=real|text)

Train dynamically  
(every 2,000 generated samples)

Details in paper
Keep it Simple: Simplicity

**Objective:** The generated text should be *simpler* than the original text, both *syntactically* and *lexically.*
Keep it Simple: Simplicity

**Syntactic Simplicity:** $S_{\text{Score}}$

We use the standard Flesch-Kincaid Grade Level (FKGL)

FKGL(“A rat named Magawa is being honored...”) = 9.5
FKGL(“A rat in Cambodia has a very special job...”) = 4.1

During training: target a fixed amount of drop (e.g., 2 grade levels). Ramp score based on how close the model gets to the target.

Keep it Simple: Simplicity

Syntactic Simplicity: $S_{Score}$

Lexical Simplicity: $L_{Score}$

Words added in the generated text should be more common than words removed from the original text.

Words Removed

<table>
<thead>
<tr>
<th>Word</th>
<th>Commonness</th>
</tr>
</thead>
<tbody>
<tr>
<td>commonness(honored)</td>
<td>4.0</td>
</tr>
<tr>
<td>commonness(potentially)</td>
<td>4.3</td>
</tr>
<tr>
<td>commonness(numerous)</td>
<td>4.7</td>
</tr>
<tr>
<td>commonness(landmines)</td>
<td>2.7</td>
</tr>
</tbody>
</table>

Words Added

<table>
<thead>
<tr>
<th>Word</th>
<th>Commonness</th>
</tr>
</thead>
<tbody>
<tr>
<td>commonness(very)</td>
<td>6.0</td>
</tr>
<tr>
<td>commonness(possibly)</td>
<td>4.7</td>
</tr>
<tr>
<td>commonness(many)</td>
<td>5.9</td>
</tr>
<tr>
<td>commonness(things)</td>
<td>5.7</td>
</tr>
</tbody>
</table>
2

Optimization
Combining Reward Components

1. Salience
   - Coverage

2. Fluency
   - Language Model
   - Discriminator

3. Simplicity
   - Lexical
   - Syntactic

Each of the 5 components is normalized between [0,1].

The total score is the product of component scores.
Self-Critical Sequence Training Recap

1. Generate 2 candidates (1 argmax and 1 sampled)
2. Compute each candidate’s total reward
3. Train model to increase likelihood of highest-reward candidate

\[ L = (\hat{R} - R^S) \sum_{i=0}^{N} \log p(w_i^S | w_1^S ... w_{i-1}^S, P) \]

Self-Critical Sequence Training Recap

1. Generate 2 candidates (1 argmax and 1 sampled)
2. Compute each candidate’s total reward
3. Train model to increase likelihood of highest-reward candidate

\[ L = (\hat{R} - R^S) \sum_{i=0}^{N} \log p(w_i^S | w_1^S ... w_{i-1}^S, P) \]

If both candidates have similar scores, loss is close to zero, no learning in the sample. In practice can happen with >30% of samples.

Contribution: k-SCST modification

1. Generate k candidates (all sampled, for example k=6)
2. Compute each candidate’s total reward
3. Train model to increase likelihood of candidates with higher than average total reward

As k increases, likelihood of good performing candidates increases.
k–SCST modification

Training the KiS model, each configuration run with 6 runs. Increasing k leads to faster, more stable training.
3 Evaluation
Automatic Results

*Keep it Simple* achieves state-of-the-art on news simplification (even outperforming supervised models)

<table>
<thead>
<tr>
<th>Method</th>
<th>SARI</th>
<th>% Lexile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>-</td>
<td>79</td>
</tr>
<tr>
<td>Finetune Baseline</td>
<td>0.470</td>
<td>52</td>
</tr>
<tr>
<td>ACCESS (Martin et al. 2020)</td>
<td>0.666</td>
<td>63</td>
</tr>
<tr>
<td>ACCESS 90</td>
<td>0.674</td>
<td>64</td>
</tr>
<tr>
<td>Unsup. NTS (Surya et al. 2019)</td>
<td>0.677</td>
<td>57</td>
</tr>
<tr>
<td>Keep It Simple (ours)</td>
<td>0.709</td>
<td>72</td>
</tr>
</tbody>
</table>

**SARI** is an n-gram overlap measure with hand-written simplified references.

**% Lexile** is the percentage of generated texts that achieve higher readability than the input, according to the Lexile measure (gold standard).
Human Evaluation of Simplification

Designing human evaluation for text simplification.

What does success mean for simplification?
Human Evaluation of Simplification

Designing human evaluation for text simplification.

**Hypothesis 1: Increasing Accessibility**
Can a broader audience understand the simplified text than the original text?
Human Evaluation of Simplification

Designing human evaluation for text simplification.

**Hypothesis 1: Increasing Accessibility**
Can a broader audience understand the simplified text than the original text?

⚠️ *Hypothesis 1 would require a study with a large and diverse population.*
Human Evaluation of Simplification

Designing human evaluation for text simplification.

Hypothesis 2: Decrease Cognitive Load
Does the simplified text lead to a similar level of understanding in less time?
Designing human evaluation for text simplification.

Hypothesis 2: Decrease Cognitive Load
Does the simplified text lead to a similar level of understanding in less time?

We design a Human Comprehension Study to investigate Hypothesis 2.
Human Comprehension Study

Original
Short News Article (~200 words)
Human Comprehension Study

Original
Short News Article (~200 words)

1. Who manages…. ?   A/ B/ C/ D/
2. Why is…. ?         A/ B/ C/
3. How did … ?         A/ B/ C/
4. What is … ?         A/ B/ C/ D/
5. When did … ?        A/ B/

Generate 5 comprehension questions.
### Human Comprehension Study

#### 4 candidate simplifications (1 human, 3 systems)

<table>
<thead>
<tr>
<th>Original</th>
<th>Newsela</th>
<th>ACCESS</th>
<th>Finetune</th>
<th>KiS</th>
</tr>
</thead>
</table>

| 1. Who manages…. ? | A/ B/ C/ D/ |
| 2. Why is…. ? | A/ B/ C/ |
| 3. How did … ? | A/ B/ C/ |
| 4. What is … ? | A/ B/ C/ D/ |
| 5. When did … ? | A/ B/ |

Generate 4 candidate simplifications
Human Comprehension Study

Participants are randomly assigned to a text version.
Human Comprehension Study

1. Who manages…. ?  A/ B/ C/ D/
2. Why is…. ?  A/ B/ C/
3. How did … ?  A/ B/ C/
4. What is … ?  A/ B/ C/ D/
5. When did … ?  A/ B/

Participants re-submit questionnaire until correct
Human Comprehension Study

1. Who manages…?   A/ B/ C/ D/
2. Why is…?   A/ B/ C/
3. How did …?   A/ B/ C/
4. What is …?   A/ B/ C/ D/
5. When did …?   A/ B/

90 participants; 4 documents; 244 total submissions
Human Comprehension Study

Hypothesis 2: Confirmed. Simplified text leads to faster completion.

<table>
<thead>
<tr>
<th>Method</th>
<th>Completion Time (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Text</td>
<td>174.0</td>
</tr>
<tr>
<td>Newsela References</td>
<td>163.3</td>
</tr>
<tr>
<td>ACCESS (Martin et al. 2020)</td>
<td>188.5</td>
</tr>
<tr>
<td>Finetune Baseline</td>
<td>161.0</td>
</tr>
<tr>
<td>Keep It Simple (ours)</td>
<td>142.6 *</td>
</tr>
</tbody>
</table>

3 / 4 methods lead to drop in completion time
KiS stat. significant drop (p < 0.05) compared to original
Thanks

See you at the Q&A!

Keep It Simple: Unsupervised Text Simplification
Laban, Schnabel, Bennet, Hearst

Code on Github:
https://github.com/tingofurro/keep_it_simple

Contact:
phillab@berkeley.edu