Neighbors Are Not Strangers: Improving Non-autoregressive Translation under Low-frequency Lexical Constraints

Chun Zeng*¹, Jiangjie Chen*¹, Tianyi Zhuang¹, Rui Xu¹, Hao Yang², Ying Qin², Shimin Tao², Yanghua Xiao¹
Non-autoregressive Translation

- **Autoregressive Translation (AT)**
  - Autoregressive decoding: $p(y_t | x, y_{<t})$
  - $O(n)$, $n =$ target length

- **Non-autoregressive Translation (NAT)**
  - Independent decoding: $p(y_t | x)$
  - $O(1)$: Decode in parallel (Faster!)

| Models                        | WMT14 En→De | WMT14 De→En | WMT16 En→Ro | WMT16 Ro→En | IWSLT16 En→De | Latency / Speedup |
|-------------------------------|-------------|-------------|-------------|-------------|---------------|-------------------|
| NAT                           | 17.35       | 20.62       | 26.22       | 27.83       | 25.20         | 39 ms            | 15.6×            |
| NAT (+FT)                     | 17.69       | 21.47       | 27.29       | 29.06       | 26.52         | 39 ms            | 15.6×            |
| NAT (+FT + NPD $s = 10$)     | 18.66       | 22.41       | 29.02       | 30.76       | 27.44         | 79 ms            | 7.68×            |
| NAT (+FT + NPD $s = 100$)    | 19.17       | 23.20       | 29.79       | 31.44       | 28.16         | 257 ms           | 2.36×            |
| Autoregressive ($b = 1$)     | 22.71       | 26.39       | 31.35       | 31.03       | 28.89         | 408 ms           | 1.49×            |
| Autoregressive ($b = 4$)     | 23.45       | 27.02       | 31.91       | 31.76       | 29.70         | 607 ms           | 1.00×            |

[1] Non-autoregressive neural machine translation (Gu et al., 2018)
Constrained NAT: Iterative Editing-based NAT

- **Iterative NAT**: trade-off of speed and performance
  - Conditioned on previous iteration
Constrained NAT: Iterative Editing-based NAT

- **Iterative NAT**: trade-off of speed and performance
  - Conditioned on previous iteration
- **Iterative editing for constrained NAT**
  - e.g. (Constrained) Levenshtein Transformer (LevT)

[2] Levenshtein Transformer (Gu et al., 2019)
Constrained NAT:
Iterative Editing-based NAT

- **Iterative NAT**: trade-off of speed and performance
  - Conditioned on previous iteration
- **Iterative editing for constrained NAT**
  - e.g. (Constrained) Levenshtein Transformer (LevT)
  - Forced *non-deletion* of constraint words as initial sequence to be edited

[2] Levenshtein Transformer (Gu et al., 2019)
[7] Lexically Constrained Neural Machine Translation with Levenshtein Transformer (Susanto et al., 2020)
Low-frequency Word Problem in Constrained NAT

- **Pre-defined terminologies** as lexical constraints to ensure the correct translation of terms
- Low-frequency constraints: *geschrien*

| Source                  | Target                               |
|-------------------------|--------------------------------------|
| Travellers *screamed* and children cried . | Reisende hätten *geschrien* und Kinder geweint . |
| 1.8K 24 2.8M 30.0K 122 | 944 9.9K 13 2.6M 20.1K 13          |

| Terminology Constraints |
|-------------------------|
| *scream* → *geschrien*  |
Low-frequency Word Problem in Constrained NAT

• *Pre-defined terminologies* as lexical constraints to ensure the correct translation of terms

• Low-frequency constraints: *geschrien*

| Source | Travellers *screamed* and children cried. |
|--------|------------------------------------------|
|        | 1.8K 24 2.8M 30.0K 122                  |

| Target | Reisende hätten *geschrien* und Kinder geweint. |
|--------|-----------------------------------------------|
|        | 944 9.9K 13 2.6M 20.1K 13                    |

**Terminology Constraints**

`scream` $\rightarrow$ *geschrien*

**Unconstrained translation**

Reisende *schrien* und Kinder rieen. $\Rightarrow$ *wrong term*
Low-frequency Word Problem in Constrained NAT

- **Pre-defined terminologies** as lexical constraints to ensure the correct translation of terms
- Low-frequency constraints: *geschrien*

| Source             | Target                          | Terminology Constraints | Hard constrained translation |
|--------------------|--------------------------------|--------------------------|------------------------------|
| Travellers *screamed* and children cried. | Reisende hätten *geschrien* und Kinder geweint. | scream → *geschrien*         | Reisende *geschrien*. ⇒ *incomplete sentence* |

**Hard Constraint**
Given constraint must appear in the translation.
Low-frequency Word Problem in Constrained NAT

- **Pre-defined terminologies** as lexical constraints to ensure the correct translation of terms
- Low-frequency constraints: *geschrien*

| Source | Travellers *screamed* and children cried. |
|--------|------------------------------------------|
|        | 1.8K  24  2.8M  30.0K  122              |

| Target | Reisende hätten *geschrien* und Kinder geweint. |
|--------|------------------------------------------|
|        | 944  9.9K  13  2.6M  20.1K  13          |

**Terminology Constraints**

- *scream* → *geschrien*

**Soft constrained translation**

- Reisende *rien*.  ⇒ *incomplete sentence & wrong term*

**Soft Constraint**

Allow constraints to be changed.
Motivating Study: Self-Constrained Translation

- Constrained NAT models seem to suffer from low-frequency constraint issues.

- **Self-constrained Translation**: Using different words in a sentence as constraints.

---

**Sort Words**
Based on Frequency

**Divide Words**
Into buckets by frequency order

**Sample a Word**
From each bucket as lexical constraints
Motivating Study: Self-Constrained Translation

Same target for different self-constraints
Motivating Study: Self-Constrained Translation

Drop#1
• Mostly unknown tokens (i.e., <UNK>) in the bucket 2.

Drop#2
• Low-frequency tokens as constraints lead to severe performance drop. 😞
The *Trade-off* in Constrained NAT

- **Easy to Translate the Constraint Itself:**
  - The model *does not have to translate rare constraints* as they are set as an *initial sequence*.

![Diagram](image-url)
The *Trade-off* In Constrained NAT

- **Easy to Translate the Constraint Itself:**
  - The model *does not have to translate rare constraints* as they are set as an *initial sequence*

- **Hard to Recognize its Neighbors:**
  - The model *has a hard time translating the context of the rare constraints*
The Trade-off In Constrained NAT

- **Easy to Translate the Constraint Itself:**
  - The model does not have to translate rare constraints as they are set as an *initial sequence*

- **Hard to Recognize its Neighbors:**
  - The model has a hard time translating the context of the rare constraints

| Source                     | 1.8K | 24 | 2.8M | 30.0K | 122 |
|----------------------------|------|----|------|-------|-----|
| Travellers screamed and children cried. |      |    |      |       |     |

| Target                     | 944  | 9.9K | 13   | 2.6M  | 20.1K| 13  |
|----------------------------|------|------|------|-------|------|-----|
| Reisende hätten geschrien und Kinder geweint. |      |      |      |       |      |     |

| Terminology Constraints    |      |      |      |       |      |     |
|----------------------------|      |      |      |       |      |     |
| scream → geschrien         |      |      |      |       |      |     |

| Unconstrained translation  |      |      |      |       |      |     |
|----------------------------|      |      |      |       |      |     |
| Reisende *schrien* und Kinder rieen. | ⇒ wrong term |

| Soft constrained translation |      |      |      |       |      |     |
|------------------------------|      |      |      |       |      |     |
| Reisende *rien*. ⇒ incomplete sentence & wrong term |

| Hard constrained translation |      |      |      |       |      |     |
|------------------------------|      |      |      |       |      |     |
| Reisende *geschrien*. ⇒ incomplete sentence |
The Trade-off In Constrained NAT

- **Easy to Translate the Constraint Itself:**
  - The model does not have to translate rare constraints as they are set as an initial sequence.

- **Hard to Recognize its Neighbors:**
  - The model has a hard time translating the context of the rare constraints.

| Source | Travellers screamed and children cried . |
|--------|------------------------------------------|
| Target | Reisende hätten geschrien und Kinder geweint . |
| Terminology Constraints | scream → geschrien |

| Unconstrained translation | Reisende *schrien* und Kinder rieen.  ⇒ wrong term |
|---------------------------|--------------------------------------------------|
| Soft constrained translation | Reisende *rien*.  ⇒ incomplete sentence & wrong term |
| Hard constrained translation | Reisende *geschrien*.  ⇒ incomplete sentence |

- **Drop #2**

![Graph showing BLEU scores with different self-constraints.](image)
Motivation: Neighbors Are Not Strangers

1. Know your neighbors.
   - Constraints are strangers (rare), but neighbors are not.
   - Prompting the alignment information between target-side constraint tokens and source tokens

2. Train to preserve constraints.
   - Bridge the gap between training and constrained decoding.

| Source | Travellers screamed and children cried. |
|--------|----------------------------------------|
|        | 1.8K 24 2.8M 30.0K 122                |

| Target | Reisende hätten geschrien und Kinder geweint. |
|--------|----------------------------------------------|
|        | 944 9.9K 13 2.6M 20.1K 13                  |

Terminology Constraints
scream → geschrien
Our Proposal

• A plug-in algorithm for lexically constrained NATs, i.e., **Aligned Constrained Training (ACT)**

• ACT is designed based on two major ideas:
  • *Constrained Training (CT)*: bridging the discrepancy between training and constrained inference
  • *Alignment Prompting*: helping the model understand the context of the constraints

\[ \text{ACT} = \text{CT} + \text{Alignment Prompting} \]
Training LevT: Imitation Learning

• Learn to Insert: \( y' \rightarrow y^* \)
  – Random deletion is applied for ground-truth \( y^* \) to get the incomplete sentences \( y' \)

• Learn to Delete: \( y'' \rightarrow y^* \)
  – Let model(\( \theta \)) insert from \( y' \) to \( y'' \)

[2] Levenshtein Transformer (Gu et al., 2019)
Discrepancy between Training and Inference

- Random deletion training in iterative NATs

- The model does not learn to
  - Preserve fixed tokens
  - Organize the translation around the tokens.
(1) Constrained Training

- Disallow deletion during building data samples for imitation learning

- Build pseudo terminology constraints
  - Sample 1-3 words (more tokens) from reference as the *pre-defined constraints* for training

![Diagram showing constrained training process]
(2) Alignment Prompting

**Alignment Embedding**

**Positional Embedding**

**Token Embedding**

**Constraint Alignment**

**Constraints Sampling** (Only at training-time)

e.g. TF-IDF distribution

$C_{src} = \{(x_1, x_2), (x_3)\}$

$C_{tgt} = \{(y_1), (y_4)\}$
(2) Alignment Prompting

1. Get constraints (during training or inference)

Constraints Sampling
(Only at training-time)

e.g. TF-IDF distribution
(2) Alignment Prompting

2. Build alignment with external alignment tools. e.g. GIZA++
(2) Alignment Prompting

3. Build alignment embedding for source tokens

Alignment Embedding

Token Embedding

Constraint Alignment

Constraints Sampling (Only at training-time)

e.g. TF-IDF distribution
(2) Alignment Prompting

4. Prompt the alignment information to the model
Experimental Setup

• **Training Set**
  • WMT14 (En-De)

• **Test Sets**
  • General domain (news)
    • WMT14-WIKT
    • WMT14-IATE
    • WMT17-WIKT
  • Specific domain
    • OPUS-EMEA (medical)
    • OPUS-JRC (legal)

• **Evaluation**
  • BLEU
  • Term Usage Rate

| Dataset       | # Sent. | Avg. Len. of Con. | Avg. Con. Freq. |
|---------------|---------|-------------------|-----------------|
| WMT14-WIKT    | 454     | 1.15              | 25,724.73       |
| WMT17-IATE    | 414     | 1.09              | 3,685.42        |
| WMT17-WIKT    | 728     | 1.22              | 26,252.70       |
| OPUS-EMEA     | 2,996   | 1.95              | 2,187.63        |
| OPUS-JRC      | 2,984   | 1.99              | 3,725.71        |
Main Results

| Models | WMT17-IATE  | WMT17-WIKT  | WMT14-WIKT  | Latency |
|--------|-------------|-------------|-------------|---------|
|        | Term% | BLEU | Term% | BLEU | Term% | BLEU | (ms) |
| **Reported results in previous work** | | | | | | |
| Transformer (Vaswani et al., 2017)† | 79.65 | 29.58 | 79.75 | 30.80 | 76.77 | 31.75 | 244.5 |
| DBA (Post and Vilar, 2018) | 82.00 | 25.30 | 99.50 | 25.80 | - | - | 434.4 |
| Train-by-rep (Dinu et al., 2019) | 94.50 | 26.00 | 93.40 | 26.30 | - | - | - |
| LevT (Gu et al., 2019)† | 80.31 | 28.97 | 81.11 | 30.24 | 80.23 | 29.86 | 92.0 |
| w/ soft constraint (Susanto et al., 2020) | 93.81 | 29.73 | 93.44 | 30.82 | 94.43 | 29.93 | - |
| w/ hard constraint (Susanto et al., 2020) | 100.00 | 30.13 | 100.00 | 31.20 | 100.00 | 30.49 | - |
| EDITOR (Xu and Carpuat, 2021)† | 83.00 | 27.90 | 83.50 | 28.80 | - | - | 121.7 |
| w/ soft constraint | 97.10 | 28.80 | 96.80 | 29.30 | - | - | - |
| w/ hard constraint | 100.00 | 28.90 | 99.80 | 29.30 | - | - | 134.1 |
| **Our implementation** | | | | | | |
| LevT† | 78.32 | 29.80 | 80.20 | 30.75 | 79.53 | 29.95 | 71.9 |
| + constrained training (CT)† | 78.76 | 29.46 | 80.77 | 30.82 | 79.13 | 30.24 | 78.6 |
| + aligned constrained training (ACT)† | 79.43 | 29.57 | 80.20 | 30.63 | 77.17 | 30.35 | 77.0 |
| LevT w/ soft constraint | 94.25 | 30.11 | 93.78 | 30.92 | 94.88 | 30.38 | 79.5 |
| + constrained training (CT) | 96.24 | 30.19 | 96.61 | 30.96 | 97.44 | 31.01 | 75.4 |
| + aligned constrained training (ACT) | 96.90 | 30.56 | 97.62 | 31.06 | 98.82 | 31.08 | 76.3 |
| LevT w/ hard constraint | 100.00 | 30.31 | 100.00 | 30.65 | 100.00 | 30.49 | 82.7 |
| + constrained training (CT) | 100.00 | 30.31 | 100.00 | 30.99 | 100.00 | 31.01 | 78.1 |
| + aligned constrained training (ACT) | 100.00 | 30.68 | 100.00 | 31.18 | 100.00 | 31.11 | 77.0 |
Ablation for CT and ACT: Term Usage Rate

1. Term usage rate increases mainly because of CT, and can be further improved by Alignment Prompting.

WMT17-IATE

WMT14-WIKT

WMT17-WIKT
Ablation for CT and ACT: BLEU

2. Translation quality (BLEU) increases due to the additional hard alignment of ACT over CT

WMT17-IATE  WMT14-WIKT  WMT17-WIKT
• Even greater performance gain
  - LevT would have a hard time recognizing them as constraints.
  - LevT + ACT knows the context ("neighbors") of the rare constraint ("strangers") and insert the translated context around the lexical constraints

| Model              | OPUS-EMEA |          | OPUS-JRC |          |
|--------------------|-----------|----------|----------|----------|
|                    | Term%     | BLEU     | Term%    | BLEU     |
| LevT†              | 52.40     | 27.90    | 55.39    | 30.24    |
| + ACT†             | 53.41     | 28.30    | 55.35    | 31.01    |
| LevT w/ soft       | 83.37     | 30.35    | 84.32    | 32.53    |
| + ACT              | 92.09     | 32.02    | 91.94    | 33.70    |
| LevT w/ hard       | 100.00    | 30.77    | 100.00   | 30.08    |
| + ACT              | 100.00    | 32.30    | 100.00   | 34.09    |
Self-Constrained Translation Revisited
Self-Constrained Translation Revisited

- ACT successfully breaks the drop with better understanding of the provided contextual information.

![Graph](image)

(a) Sorting self-constraints by frequency.
Self-Constrained Translation Revisited

- **What if the self-constraints are sorted based on TF-IDF?**
  - Very similar trends

![Graph showing BLEU scores for different conditions](image)

(b) Sorting self-constraints by TF-IDF.
How does ACT perform under different kinds of lexical constraints?

(1) Are improvements by ACT robust against constraints of different frequencies?
How does ACT perform under different kinds of lexical constraints?

(1) Are improvements by ACT robust against constraints of different frequencies?

| Model            | WMT14-WIKT       | WMT17-IATE       | WMT17-WIKT       |
|------------------|------------------|------------------|------------------|
|                  | ALL  | HIGH | MED.  | LOW  | ALL  | HIGH | MED.  | LOW  | ALL  | HIGH | MED.  | LOW  |
| LevT†            | 29.95 | 30.46 | 28.03 | 31.49 | 29.80 | 30.08 | 29.72 | 29.45 | 30.75 | 30.96 | 29.09 | 32.16 |
| + ACT†           | 30.35 | 30.68 | 28.00 | 32.54 | 29.57 | 29.63 | 29.57 | 29.20 | 30.63 | 30.35 | 29.11 | 32.46 |
| LevT w/ soft     | 30.38 | 30.37 | 28.50 | 32.19 | 30.11 | 29.25 | 30.67 | 30.04 | 30.92 | 30.70 | 29.58 | 32.23 |
| + ACT            | 31.08 | 30.48 | 29.18 | 33.85 | 30.56 | 29.93 | 31.05 | 30.36 | 31.06 | 30.72 | 29.53 | 32.73 |
| LevT w/ hard     | 30.49 | 30.50 | 28.67 | 31.99 | 30.31 | 29.46 | 30.66 | 30.37 | 30.65 | 30.28 | 29.44 | 32.00 |
| + ACT            | 31.11 | 30.23 | 29.32 | 33.85 | 30.68 | 29.97 | 31.18 | 30.67 | 31.18 | 30.58 | 29.71 | 32.90 |

Table 6: Ablation results of terminology-constrained En→De translation tasks w.r.t. word frequency of terms.

- LevT benefits mostly from ACT in the scenarios of lower frequency terms for three datasets.
How does ACT perform under different kinds of lexical constraints?

(2) Are improvements by ACT robust against constraints of different numbers?
How does ACT perform under different kinds of lexical constraints?

(2) Are improvements by ACT robust against constraints of different numbers?

- The translation quality ostensibly becomes better for LevT with or without ACT.
- ACT consistently brings extra improvements.
Limitations
Limitations

• For unconstrained translation, ACT does not bring much performance gain.
Limitations

• For unconstrained translation, ACT does not bring much performance gain.

• We do not propose a new paradigm for constrained NAT (editing-based iterative NATs).
Limitations

• For unconstrained translation, ACT does not bring much performance gain.

• We do not propose a new paradigm for constrained NAT (editing-based iterative NATs).

• We call for new paradigms for constrained NAT! Perhaps even one-pass NAT!
Takeaways

• Neighbors are not strangers: prompting constrained NATs with alignment information alleviates low-frequency constraints problem.

• We propose a plug-in algorithm (ACT) to improve lexically constrained NAT, especially under low-frequency constraints.

• Further analyses show that the findings are consistent over constraints varied from frequency, TF-IDF, and numbers.
More About ACT

https://github.com/sted-byte/ACT4NAT

jjchen19@fudan.edu.cn

https://jiangjiechen.github.io