EMNLP 2021 Presentation

Self-Supervised Quality Estimation for Machine Translation

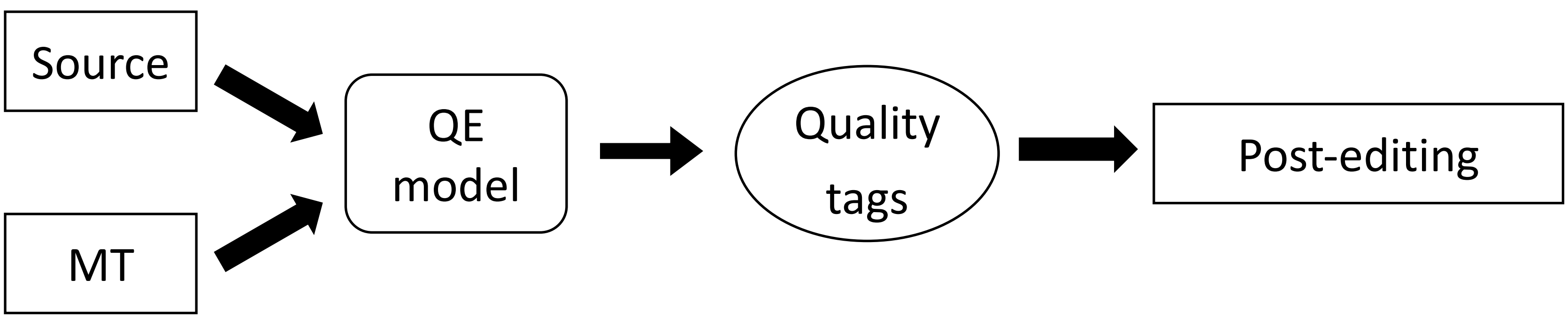
Yuanhang Zheng

Tsinghua University

Background

- Quality estimation (QE) for machine translation (MT) aims to evaluate the quality of machine-translated sentences without references.

- QE can reduce human efforts in post-editing (Specia, 2011).



Background

- QE data with human-annotated quality labels are difficult to obtain in practice.
- Thus, various studies have explored unsupervised QE.

Number of sentences in the WMT 2018 QE training data

En-De	De-En	En-Lv	En-Cs
49,715	25,963	24,187	40,254

Number of sentences in the WMT 2020 QE training data

En-De	En-Zh	Ro-En	Et-En	Si-En	Ne-En	Ru-En
7,000	7,000	7,000	7,000	7,000	7,000	7,000

Previous Work and Challenges

- Comparison of advantages and disadvantages of previous unsupervised QE methods (Popović, 2012; Etchegoyhen et al., 2018; Zhang et al., 2020; Zhou et al., 2020; Fomicheva et al., 2020; Tuan et al., 2021)

Method	Advantages	Disadvantages
Feature-based	Simple and effective	Limited to sentence-level
Synthetic data-based	Suitable for both sentence- and word-level	Affected by noise Complex

Task Description

- QE aims to predict the quality scores of the machine-translated sentences (for sentence-level) or detect the erroneous words in the target sentences (for word-level) without using references.

- The labels are generated by comparing the target sentences with their post-editions using the TER tool (Snover et al., 2005).

- For word-level QE, each target word is annotated with “OK” or “BAD”, where “OK” denotes correct and “BAD” denotes erroneous.

Source	我 喜欢 音乐 。
Target	I like songs .
Post-Edition	I like music .
Word-Level QE	OK OK BAD OK

Task Description

- For sentence-level QE, target sentences are annotated with HTER scores, which measure the percentage of human edits to correct the target sentences:

$$\mathrm{HTER} = \frac{\text{number of edits}}{\text{number of words in the post}-\text{edition}}$$

- Sentence-level scores are calculated based on the word-level errors in the target sentences, and thus they can be approximately regarded as a summary of word-level tags.

Source	我 喜欢 音乐 。
Target	I like songs .
Post-Edition	I like music .
Word-Level QE	OK OK BAD OK
Sentence-Level QE	0.25

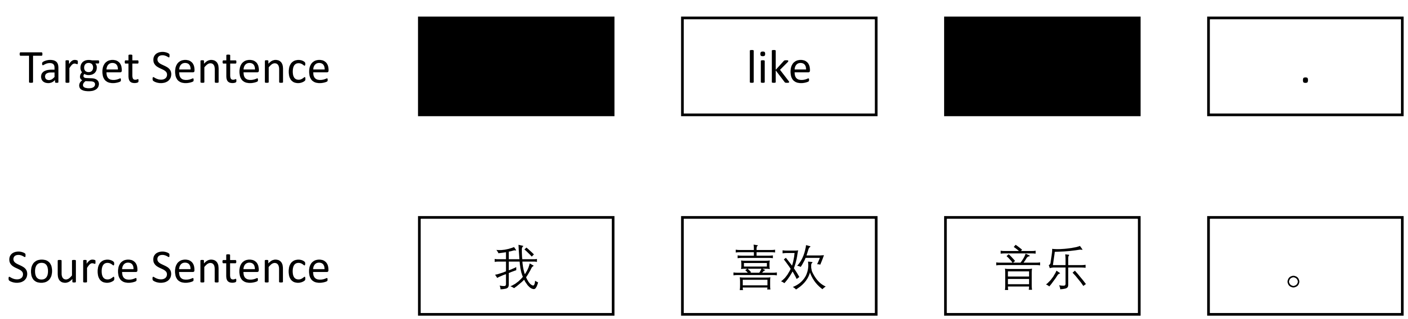
Methodology

- We mask some target words and use the source sentence and the remaining target words to recover the masked words.
- A target word is correct if it can be successfully recovered, otherwise it tends to be erroneous.
- We obtain sentence-level scores by summarizing word-level predictions.

Target Sentence	I	like	songs	.
Source Sentence	我	喜欢	音乐	。

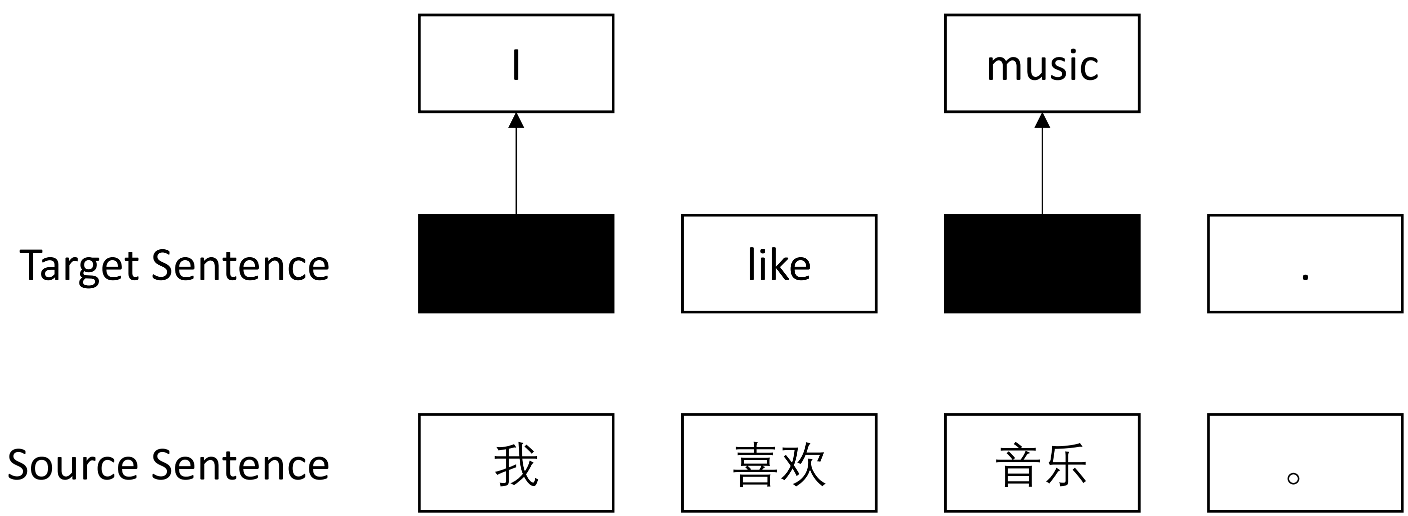
Methodology

- We mask some target words and use the source sentence and the remaining target words to recover the masked words.
- A target word is correct if it can be successfully recovered, otherwise it tends to be erroneous.
- We obtain sentence-level scores by summarizing word-level predictions.



Methodology

- We mask some target words and use the source sentence and the remaining target words to recover the masked words.
- A target word is correct if it can be successfully recovered, otherwise it tends to be erroneous.
- We obtain sentence-level scores by summarizing word-level predictions.

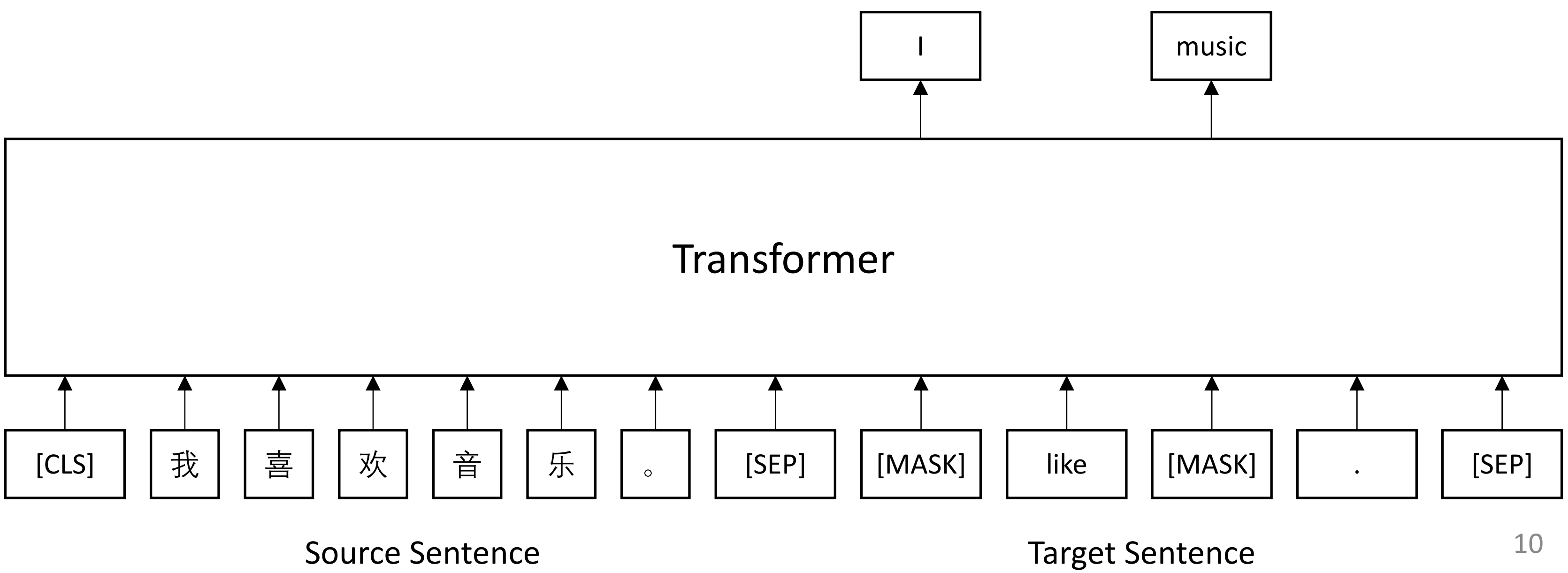


Model Architecture

- Our method is based on the multilingual BERT (Devlin et al., 2019).

- The input is the concatenation of the source sentence and the partially masked target sentence.

- We use a Transformer encoder to recover the masked target words.



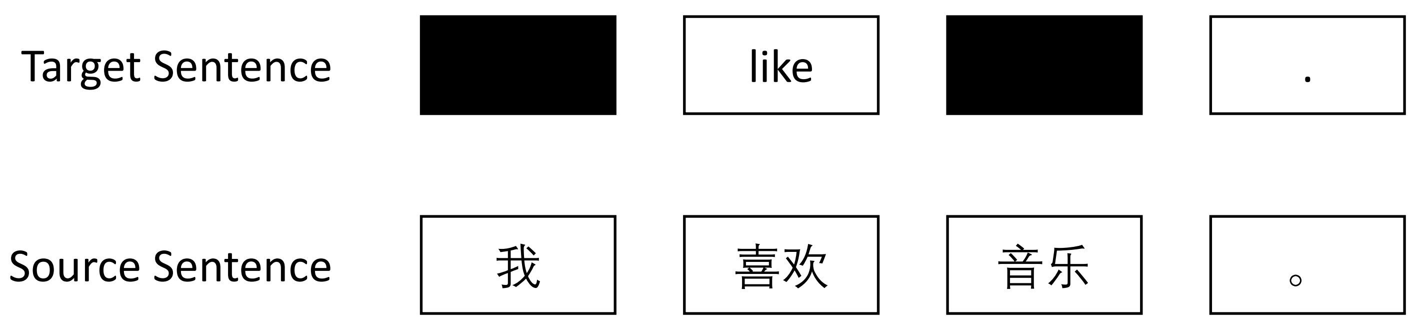
Training Process

- The model is trained on authentic parallel corpora.
- During training, we mask some words in the target sentence, and the model is required to recover the masked words.

Target Sentence	I	like	music	.
Source Sentence	我	喜欢	音乐	。

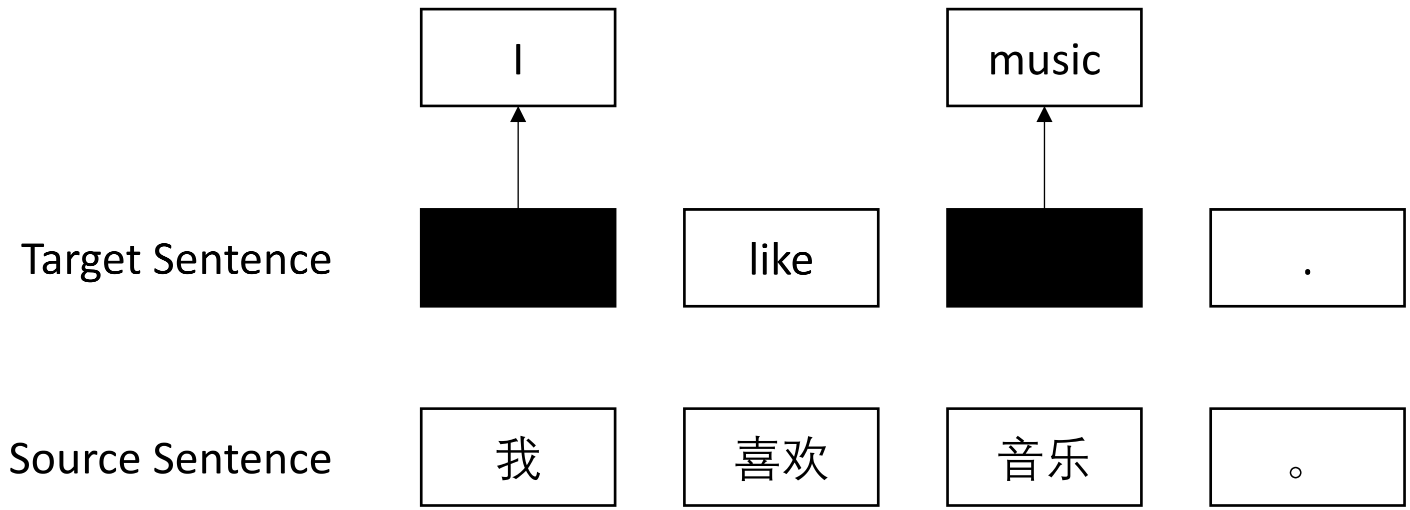
Training Process

- The model is trained on authentic parallel corpora.
- During training, we mask some words in the target sentence, and the model is required to recover the masked words.



Training Process

- The model is trained on authentic parallel corpora.
- During training, we mask some words in the target sentence, and the model is required to recover the masked words.



Inference Process

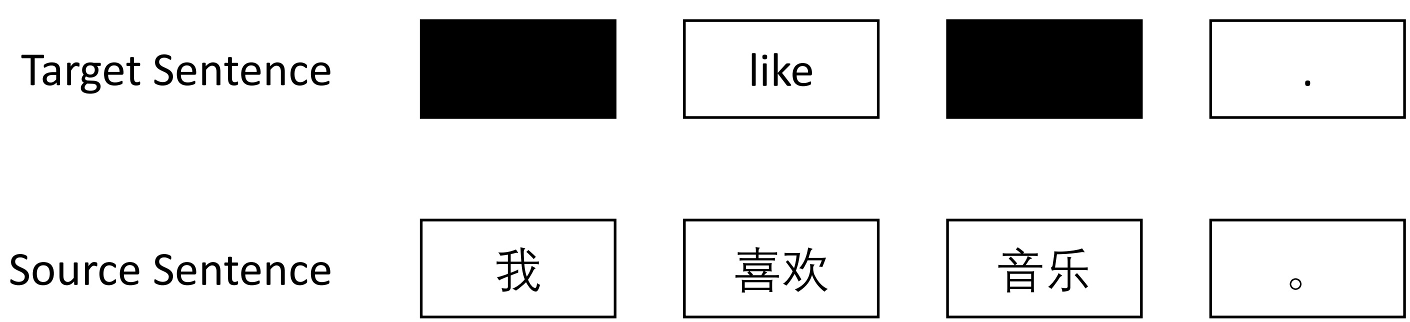
- During inference, we detect erroneous target words using the probability of successful recovery.

- For sentence-level QE, we calculate the quality score by averaging the quality scores over all target words.

Target Sentence	I	like	songs	.
Source Sentence	我	喜欢	音乐	。

Inference Process

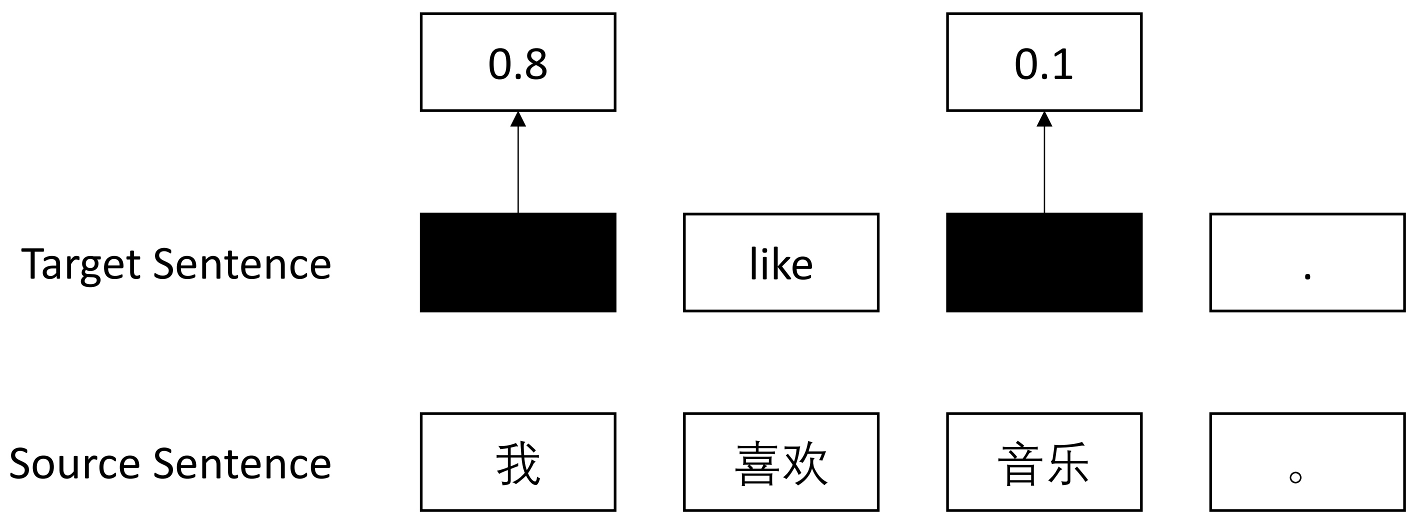
- During inference, we detect erroneous target words using the probability of successful recovery.
- For sentence-level QE, we calculate the quality score by averaging the quality scores over all target words.



Inference Process

- During inference, we detect erroneous target words using the probability of successful recovery.

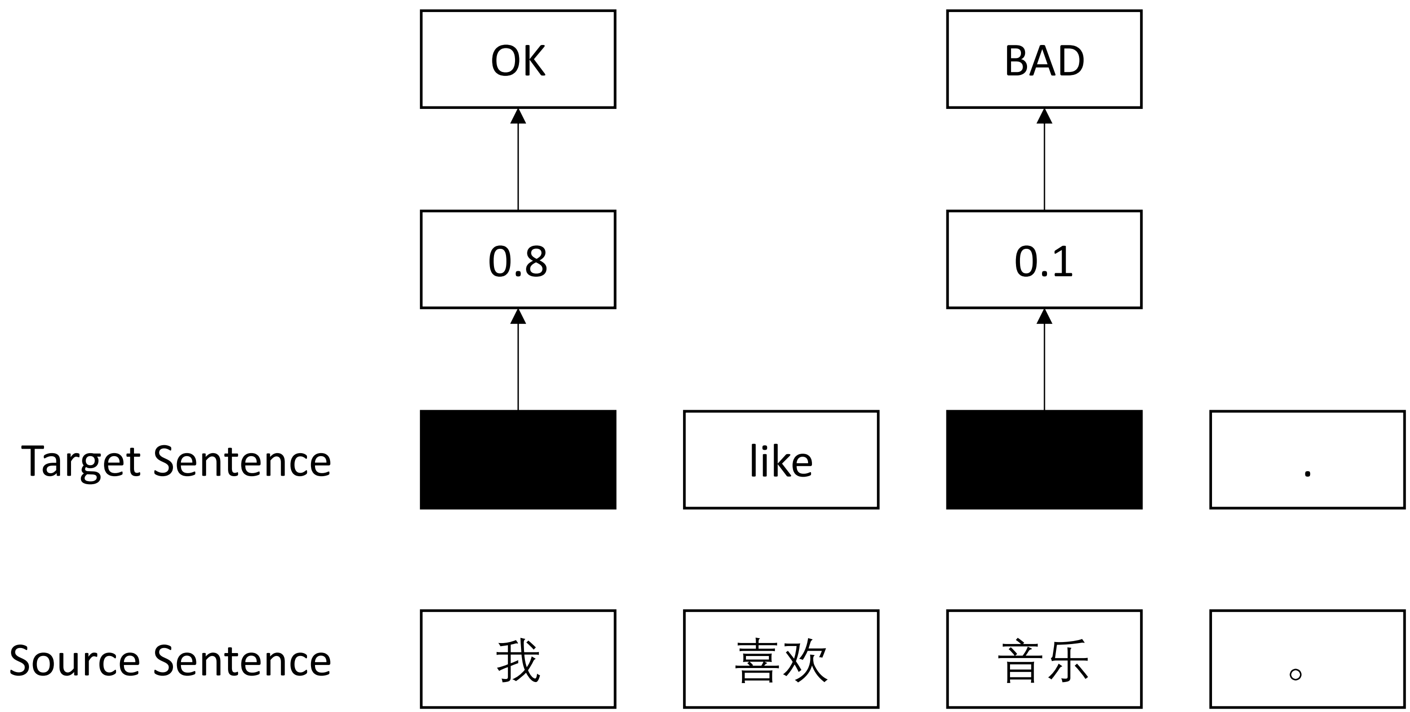
- For sentence-level QE, we calculate the quality score by averaging the quality scores over all target words.



Inference Process

- During inference, we detect erroneous target words using the probability of successful recovery.

- For sentence-level QE, we calculate the quality score by averaging the quality scores over all target words.



Inference Process

- To further improve the model's performance, we utilize Monte-Carlo (MC) Dropout (Gal and Ghahramani, 2016), which can extract model uncertainty, and is proven conducive to the performance of unsupervised QE models (Fomicheva et al., 2020).

Algorithm 1 Calculating quality scores with Monte Carlo Dropout

Input: source sentence $\mathbf{x}$, target sentence $\hat{\mathbf{y}} = (\hat{y}_1, \cdots, \hat{y}_T)$, number of samples for each target token N, number of estimations N', model parameter $\boldsymbol{\theta}$

Output: quality scores of all target tokens $score(\hat{y}_1), \cdots, score(\hat{y}_T)$

for $n \leftarrow 1$ to N' **do**
    $\hat{\mathbf{y}}_m^{(n)} \leftarrow \emptyset$
for $t \leftarrow 1$ to T **do**
    $score(\hat{y}_t) \leftarrow 0$
    Randomly sample N integers $n_1, n_2, \cdots, n_N$ from $[1, N']$
    for $i \leftarrow 1$ to N **do**
        $\hat{\mathbf{y}}_m^{(n_i)} \leftarrow \hat{\mathbf{y}}_m^{(n_i)} \cup \{\hat{y}_t\}$
for $n \leftarrow 1$ to N' **do**
    $\hat{\mathbf{y}}_o^{(n)} \leftarrow \hat{\mathbf{y}} \backslash \hat{\mathbf{y}}_m^{(n)}$
    Sample a model $\hat{\boldsymbol{\theta}}_n$ from $\boldsymbol{\theta}$ using dropout
    Calculate $P(\hat{y}_t|\mathbf{x}, \hat{\mathbf{y}}_o^{(n)}; \hat{\boldsymbol{\theta}}_n)$ for all $\hat{y}_t \in \hat{\mathbf{y}}_m^{(n)}$ using the model $\hat{\boldsymbol{\theta}}_n$
    for each $\hat{y}_t \in \hat{\mathbf{y}}_m^{(n)}$ **do**
        $score(\hat{y}_t) \leftarrow score(\hat{y}_t) + P(\hat{y}_t|\mathbf{x}, \hat{\mathbf{y}}_o^{(n)}; \hat{\boldsymbol{\theta}}_n) \ / \ N$
return $score(\hat{y}_1), \cdots, score(\hat{y}_T)$

Main Results

- Comparison with SyntheticQE (Tuan et al., 2021)

Method	En-De				En-Ru			
	Sentence-Level		Word-Level		Sentence-Level		Word-Level	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test
Results of Supervised Models								
Supervised	0.473	0.507	0.366	0.396	0.495	0.517	0.410	0.448
Results of Single Unsupervised Models								
SyntheticQE-MT	0.478	0.425	0.349	0.338	0.201	0.233	0.263	0.265
SyntheticQE-MLM	0.386	0.368	0.318	0.309	0.204	0.284	0.181	0.208
Ours	**0.504**	**0.463**	**0.381**	**0.383**	**0.242**	**0.435**	**0.318**	**0.338**
Results of Ensemble Unsupervised Models								
SyntheticQE-MT Ensemble	0.488	0.428	0.360	0.339	0.212	0.246	0.274	0.297
SyntheticQE-MLM Ensemble	0.407	0.379	0.318	0.307	0.210	0.299	0.185	0.216
SyntheticQE-MT+MLM	0.508	0.460	0.373	0.362	0.247	0.317	0.262	0.286
Ours Ensemble	**0.518**	**0.462**	**0.395**	**0.385**	**0.248**	**0.453**	**0.318**	**0.359**

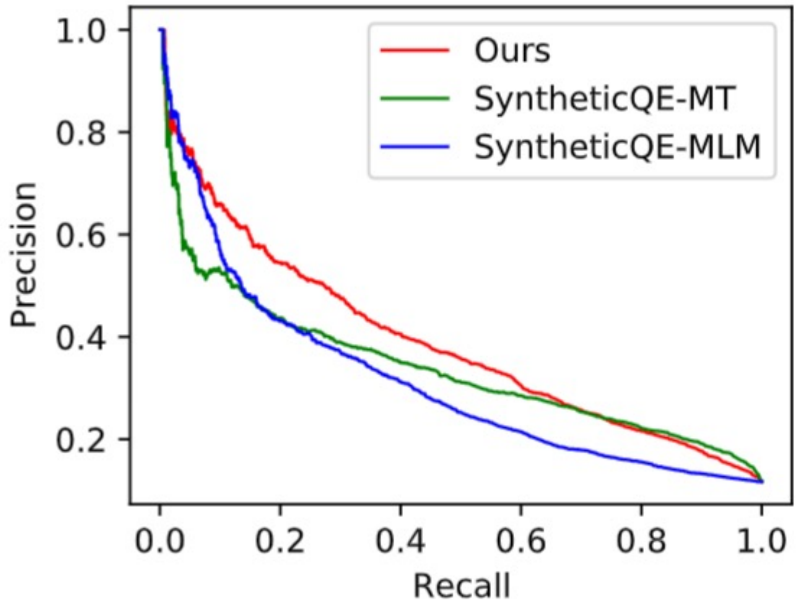
Main Results

- Comparison with feature-based unsupervised QE methods

Method	En-Lv		En-De	En-Ru
	SMT	NMT	NMT	NMT
uMQE (Etchegoyhen et al., 2018)	0.385	0.550	0.375	0.243
BERTScore (Zhang et al., 2020)	0.176	0.221	-0.101	0.093
BERTScore++ (Zhou et al., 2020)	0.213	0.155	-0.073	0.069
NMT-QE (Fomicheva et al., 2020)	0.540	0.580	0.452	0.372
Ours	**0.560**	**0.590**	**0.463**	**0.435**

Analysis

- Precision-Recall Curve
  - Precision of SyntheticQE-MT is relatively low when recall < 0.2.
  - Precision of SyntheticQE-MLM is relatively low when recall > 0.2.
  - Our method obtains relatively high precision whenever the recall is low or high.



Analysis

- In SyntheticQE-MT, the target side of the synthetic data is produced by MT models.

- More words may be labeled with “BAD” in synthetic data since references are less similar to machine-translated sentences than post-editions (Snover et al., 2005).

Source	昨天 我 吃 了 一 个 蛋糕 。
Target	Yesterday I ate a cakes .
Reference	I ate a cake yesterday .
Synthetic Labels	BAD OK OK OK BAD OK
Post-Edition	Yesterday I ate a cake .
Authentic Labels	OK OK OK OK BAD OK

Analysis

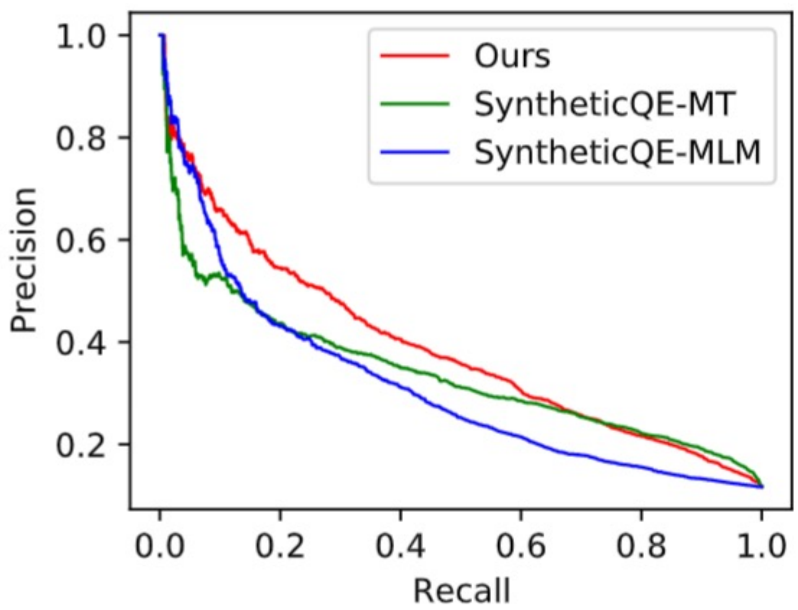
- In SyntheticQE-MLM, the target side of the synthetic data is produced by MLMs.

- Sentences rewritten by MLM usually contain catastrophic errors, which rarely appear in machine-translated sentences (Tuan et al., 2021).

Source	我 喜欢 音乐 。
Reference	I like music .
Masked Reference	I like [MASK] .
Synthetic Target	I like reading .

Analysis

- Our self-supervised QE method does not rely on synthetic data.
- Our method is not affected by the noise and achieves better results whenever the recall is low or high.



Analysis

- Case study (erroneous word “Schnappschüsse” is corrected to “Schnappschüssen” in the post-edition)

Source	switch between the snapshots to find the settings you like best .
Target & Golden	wechseln Sie zwischen den Schnappschüsse , um die gewünschten Einstellungen zu finden .
SyntheticQE-MT	wechseln Sie zwischen den Schnappschüsse , um die gewünschten Einstellungen zu finden .
SyntheticQE-MLM	wechseln Sie zwischen den Schnappschüsse , um die gewünschten Einstellungen zu finden .
Ours	wechseln Sie zwischen den Schnappschüsse , um die gewünschten Einstellungen zu finden .

Conclusion and Future Work

- In this work, we propose a self-supervised QE method.
- The central idea is to perform QE by recovering masked target words.
- This method is easy to implement and is not affected by noisy synthetic data.
- Experimental results show that our method outperforms previous unsupervised methods.
- In the future, we plan to extend our method to phrase- and document-level tasks.

Thanks for your Listening!