ECNU

SCHOOL OF DATA SCIENCE & ENGINEERING
数据科学与工程学院

An LLM-Enhanced Adversarial Editing System for Lexical Simplification

Keren Tan[1], Kangyang Luo[1], Yunshi Lan[1*], Zheng Yuan[2], Jinlong Shu[1]

[1]School of Data Science & Engineering, East China Normal University, Shanghai, China

[2]Department of Informatics, King's College London, U.K.

LREC-COLING 2024

2024-05

Introduction

- ## Lexical Simplification.

Complex Sentence

A committee of the institute appoints the ***laureates*** for the Nobel Prize in Physiology or Medicine.

Text Simplification (TS)

The Nobel Prize in Physiology or Medicine laureates are chosen by a committee at the institute.

Lexical Simplification (LS)

A committee of the institute appoints the ***honorees*** for the Nobel Prize in Physiology or Medicine.

- **Text Simplification (TS)** is the process of simplifying a sentence while retaining its semantics as much as possible.
- As a special category of TS tasks, **Lexical Simplification (LS)** restricts the simplification at the lexical level via replacing complex words with alternative simpler words, thus minimizing the revision to the original sentences.

Introduction

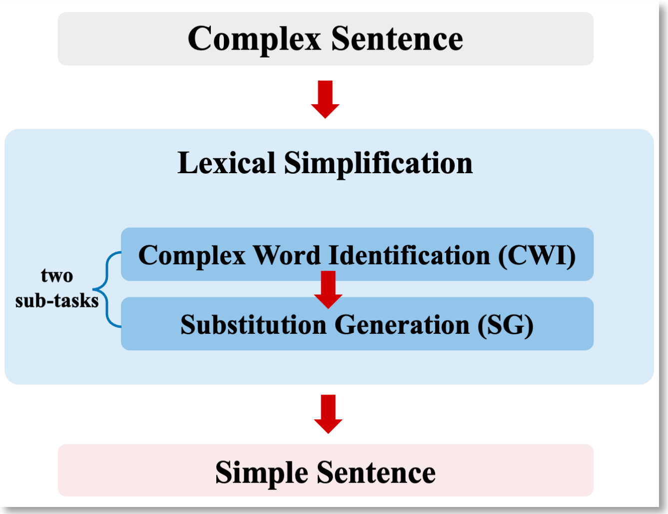
- ## Existing Methods and Challenges.

The existing two-stage approaches have a heavy reliance on the annotation of CWI and SG sub-tasks, thereby impairing their applications.

As such, we aim to develop an LS system without parallel corpora in this work, but we are also confronted with the following challenges:

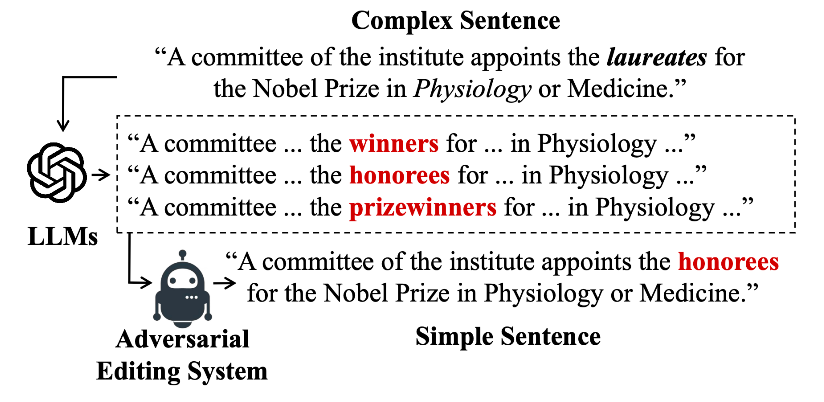
(1) In the **absence of annotated data**, the above-mentioned supervised training approaches are inapplicable, making it considerably challenging to ensure the accuracy of simplification.

(2) Constructing the previous two-stage system for LS tasks **without parallel corpora** is problematic, as in such scenarios, models struggle to learn the transformation from complex sentences to simplified ones.



Introduction

- ## Motivations.



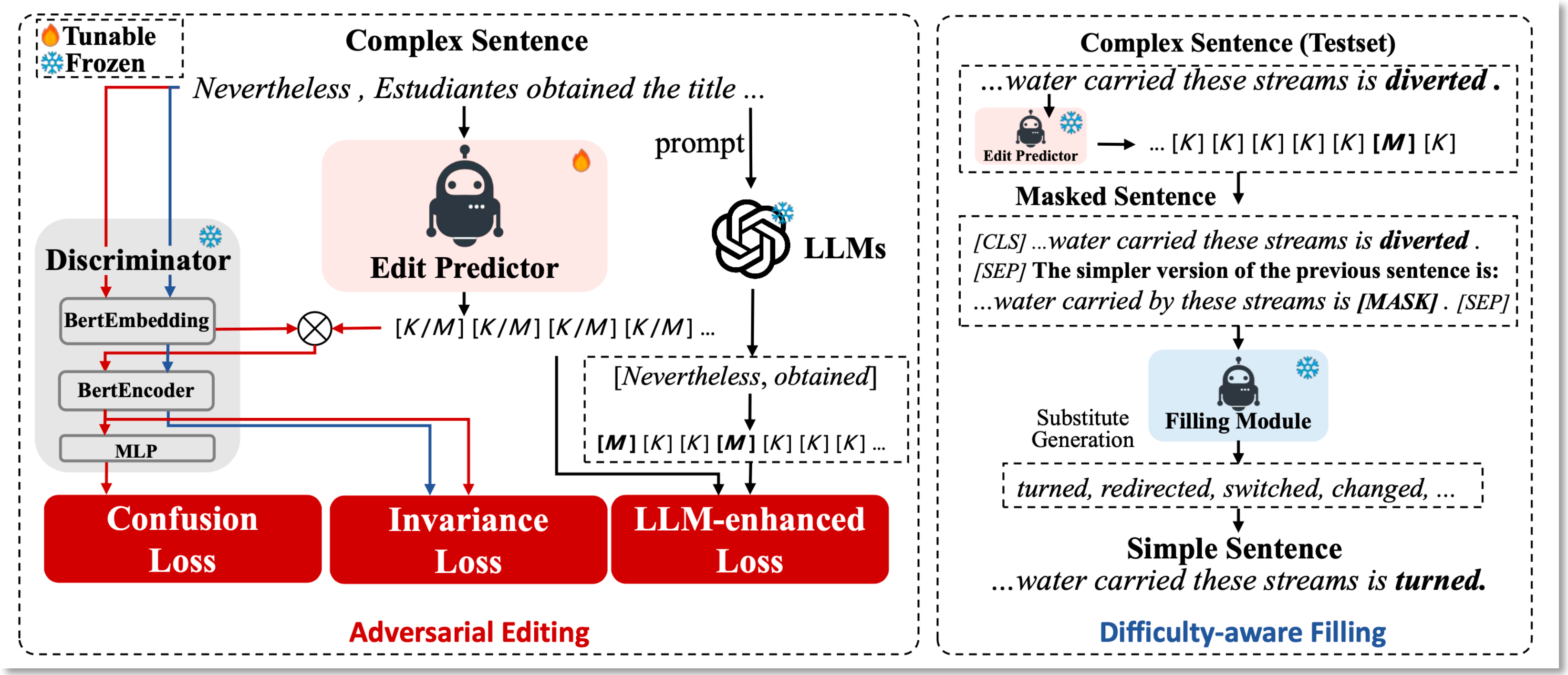
- We develop an **Adversarial Editing System** to conduct lexical edits to the original sentence with the help of non-parallel corpora, where complex words are masked by the editing system, and the substitutions are generated via a cloze model following the two-stage approaches.

Nonetheless, *striking a balance between semantic preservation and simplification degree* remains a challenging endeavor.

The motivation of our LLM-enhanced Adversarial Editing System, that is, *distilling the knowledge from LLMs to our small-size Adversarial Editing System*.

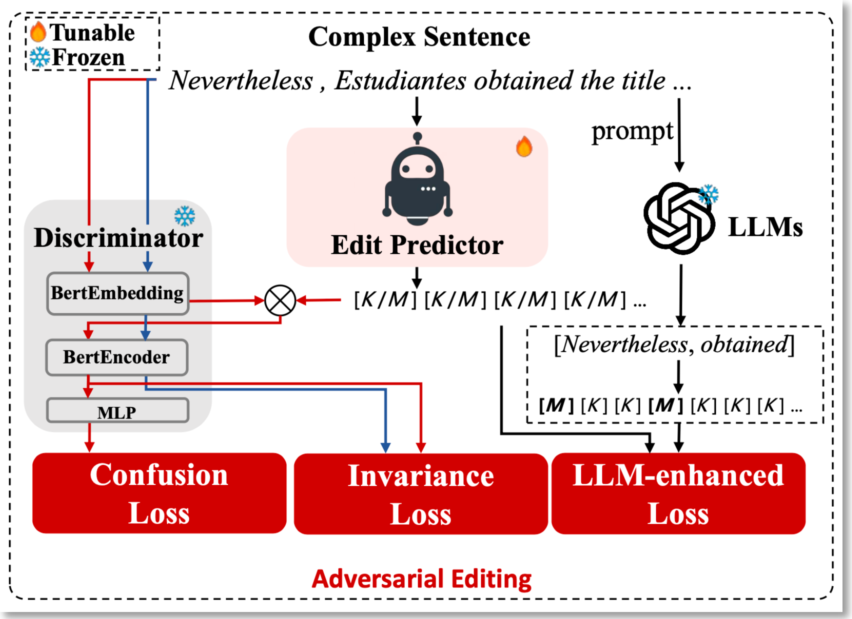
Method

- LAE-LS (LLM-Enhanced Adversarial Editing System for Lexical Simplification) .



Method

- ## Adversarial Editing.



The traditional loss for adversarial generation is not applicable in our framework due to the following two issues:

(1) It is not feasible to include the raw output of the Edit Predictor for adversarial training as "K" and "M" cannot be directly encoded by the discriminator.

(2) It is vital to control the predicted edits and maintain the syntax for lexical simplification. However, existing methods usually ignore this and lead to unexpected changes to the original sentences.

Method

- **Training.**

Confusion Loss

$$\mathbf{w}_l = \mathbf{w}_l^{tok} \cdot p_l^K + \mathbf{w}_l^{typ} + \mathbf{w}_l^{pos} (l \in [L]),$$
$$\mathbf{H}^{conf} = \mathrm{BERT}(\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_L),$$
$$P_D = \mathrm{Classifier}(\mathbf{h}_1^{conf}),$$

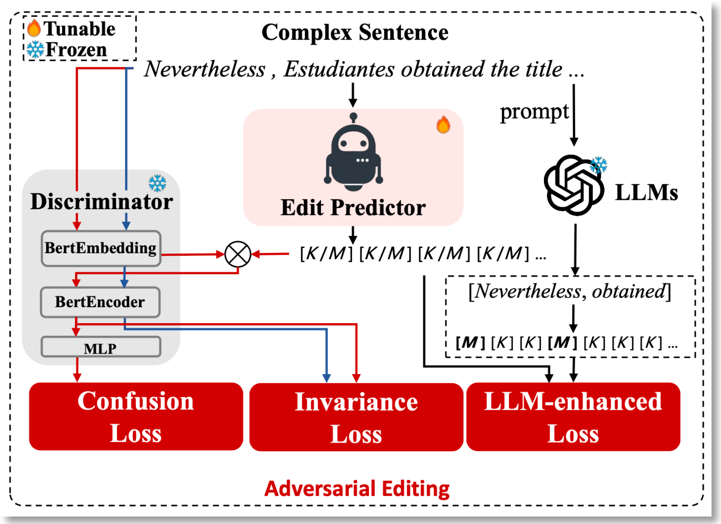
$$\mathcal{L}_G^{conf} = (P_D - \alpha)^2.$$

Invariance Loss

$$\mathcal{L}_G^{inv} = 1 - cos(\mathbf{h}_1, \mathbf{h}_1^{conf}).$$

LLM-enhanced Loss

$$\mathcal{L}_G^{LLM} = -\frac{1}{L} \sum_{w_l \in X_i} [\log_{P_G}(g_l^* | w_l)],$$

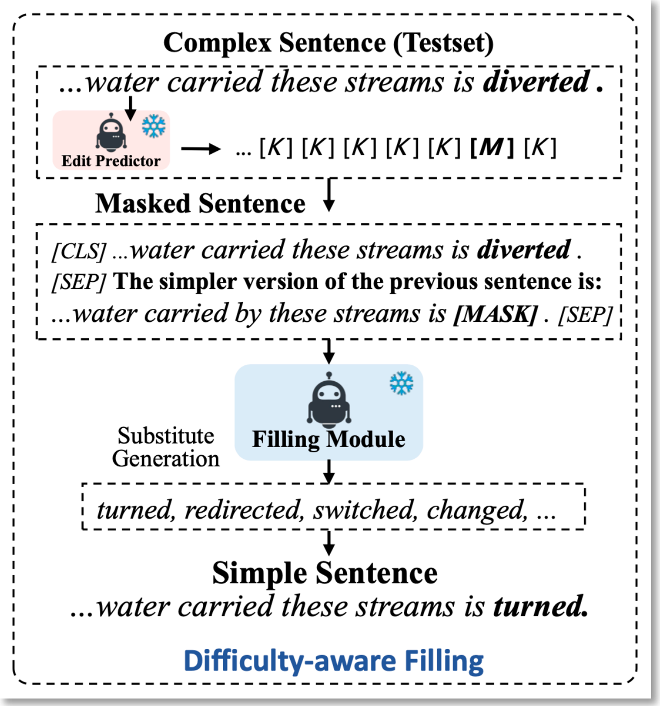


$$\mathcal{L}_G = \lambda_1 \mathcal{L}_G^{conf} + \lambda_2 \mathcal{L}_G^{inv} + \lambda_3 \mathcal{L}_G^{LLM}$$

LLMs have the risk of over-editing, taking their outputs as the supervision signals play the effect of *distilling high-quality knowledge to the small-size models*, which can effectively restrain the over-fitting issue.

Method

- **Difficulty-Aware Filling.**



Difficulty-aware Filling Prompt:

```
[CLS] Original sentence [SEP] The
↪ simpler version of the previous
↪ sentence is:  Masked sentence [SEP]
```

Example:

```
[CLS] much of the water carried these
↪ streams is diverted . [SEP] The
↪ simpler version of the previous
↪ sentence is: much of the water
↪ carried by these streams is [MASK] .
↪ [SEP]
```

- Remarkably, unlike the previous filling model, the Difficulty-aware Filling module, which is a cloze model, not only considers **original sentences as clues** but also maintains an **awareness of producing simpler words**.

Experiments

- **Comparison with Baselines.**

Methods	LexMTurk			BenchLS			NNSeval		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Complex Word Identification									
Character	0.122	0.780	0.211	0.111	0.755	0.194	0.105	0.716	0.183
Syllable	0.140	0.606	0.228	0.117	0.526	0.191	0.100	0.456	0.163
Vowel	0.132	0.764	0.226	0.117	0.727	0.201	0.108	0.678	0.186
Frequency	0.078	0.632	0.139	0.072	0.623	0.129	0.054	0.456	0.096
Attention	0.064	0.512	0.114	0.062	0.448	0.109	0.058	0.435	0.103
LSBert	**0.136**	0.795	0.231	**0.136**	0.788	**0.231**	0.121	0.707	0.207
LAE-LS (ours)	0.135	**0.810**	**0.232**	0.128	**0.813**	0.221	**0.126**	**0.824**	**0.218**
Substitute Generation									
Paetzold-CA	0.177	0.140	0.156	0.180	0.252	0.210	0.118	0.161	0.136
Paetzold-NE	0.310	0.142	0.195	0.270	0.209	0.236	0.186	0.136	0.157
REC-LS	0.151	0.154	0.152	0.129	0.246	0.170	0.103	0.155	0.124
LSBert	0.306	0.238	0.268	0.244	0.331	0.281	0.194	0.260	0.222
BART	0.192	0.183	0.188	0.196	0.178	0.192	-	-	-
SimpleBART	0.287	**0.282**	0.285	**0.280**	0.276	0.278	-	-	-
LAE-LS (ours)	**0.340**	0.264	**0.297**	0.262	**0.355**	**0.301**	**0.202**	**0.269**	**0.231**
Lexical Simplification									
Character-LSBert	0.080	0.540	0.139	0.061	0.434	0.107	0.044	0.318	0.078
Syllable-LSBert	0.090	0.410	0.148	0.064	0.299	0.105	0.042	0.201	0.069
Vowel-LSBert	0.087	0.528	0.149	0.063	0.412	0.110	0.045	0.293	0.077
Frequency-LSBert	0.047	0.440	0.085	0.036	0.364	0.066	0.023	0.226	0.042
Attention-LSBert	0.039	0.350	0.070	0.031	0.243	0.054	0.020	0.167	0.036
LSBert	**0.097**	0.564	0.166	0.075	0.454	0.129	0.056	0.335	0.095
LAE-LS (ours)	**0.097**	**0.582**	**0.167**	**0.077**	**0.489**	**0.133**	**0.058**	**0.381**	**0.101**

Table 1: CWI, SG and LS evaluations on three benchmark datasets.

LAE-LS consistently outperforms baselines across three benchmark LS datasets.

(1) For the **CWI task**, LAE-LS achieves the best results on the LexMTurk and NNSeval datasets and demonstrates competitive performance on the BenchLS dataset.

(2) In the **SG task**, our method outperforms all baselines on the three datasets.

(3) Regarding the **LS task**, our method consistently outperforms the baselines when we integrate CWI and SG together.

Experiments

- ## Comparison with LLMs.

	Size	F1-CWI	F1-SG	F1-LS
ChatGLM2	6B	0.027	0.250	0.048
llama2	13B	0.115	0.264	0.085
GPT-3.5-turbo	175B	0.221	0.296	**0.200**
LAE-LS (ours)	220M	**0.232**	**0.297**	0.167

Table 2: Comparison with various LLMs on LexMTurk Datasets in term of parameter size and F1.

LAE-LS, which has *a smaller parameter size*, can achieve *competitive results* comparing with the powerful LLMs.

- ## Ablation Study.

	F1-CWI	F1-SG	F1-LS
LAE-LS (baseline)	**0.232**	**0.297**	**0.167**
w/o LLM-enhanced Loss	0.094	0.297	0.066
w/o Confusion Loss	0.078	0.297	0.135
w/o Invariance Loss	0.089	0.297	0.153
w/o Difficulty-aware Filling	0.232	0.268	0.162

Table 3: Ablation Study of LS on LexMTurk Datasets w.r.t. F1.

It is evident that removing any of these *loss functions leads to performance drop*, suggesting that they are vital for the training of Edit Predictor.

Experiments

- ## Case Study.

Methods	Sentence
Sent (1)	Triangles ... be **classified** according to their internal angles, measured here in degrees.
Candidates	{called, labeled, divided, coded, defined, listed, **categorized**, named, organized, described...}
LSBert	**squares** ... be **categorized** according to their **external triangles**, here in **metric**.
GPT-3.5-turbo	Triangles ... be **categorized** according to their **inner corners**, **calculated** here in **units**.
LAE-LS (ours)	Triangles ... be **categorized** according to their internal angles, measured here in degrees.
Sent (2)	Stone floor tiles tend to ... ceramic tiles and somewhat more **prone** to breakage...
Candidates	{liable, easier, probable, subject, **susceptible**, disposed, **likely**, inclined, vulnerable, apt...}
LSBert	Stone floor tiles tend to ... **porcelain** tiles and somewhat more **susceptible** to **cracking**...
GPT-3.5-turbo	Stone floor tiles tend to ... **clay** tiles and somewhat more prone to **damage**...
LAE-LS (ours)	Stone floor tiles tend to ... ceramic tiles and somewhat more **likely** to breakage...

Table 4: Case study of LS on LexMTurk Datasets. Complex words are highlighted in bold. Candidates indicate the list of annotated simple words for the corresponding complex words in the dataset. Differences between generated and original sentences are in bold.

LAE-LS preserves the semantic information of the original sentence and leads to a more desirable simplification.

Conclusion

In this paper, we propose an LLM-enhanced Adversarial Editing System to address the lexical simplification task without parallel corpora, which consists of an Adversarial Editing module and a Difficulty-aware Filling module.

- **Adversarial Editing module** is guided by **a confusion loss** and **an invariance loss** to make lexical edits with a consideration of semantic preservation and simplified ratio. Meanwhile, we craft **an LLM-enhanced loss** to distill knowledge from LLMs, thus further augmenting the Adversarial Editing module.
- From that, the **Difficulty-aware Filling module** combines the original sentences and lexical edits to mask complex words within sentences and fill in the masked positions with simpler words.
- The **extensive experimental results** on three LS datasets demonstrate that our method is effective. That is, our method not only **advances lexical simplification in the absence of parallel corpora** but also showcases the potential for **leveraging the capabilities of large language models** to enhance the simplification process.

SCHOOL OF DATA SCIENCE & ENGINEERING
数据科学与工程学院

THANKS

Keren Tan[1], Kangyang Luo[1], Yunshi Lan[1*], Zheng Yuan[2], Jinlong Shu[1]

[1]School of Data Science & Engineering, East China Normal University, Shanghai, China

[2]Department of Informatics, King’ s College London, U.K.