Improving Type-Driven Multi-Turn Corrections through Distillation

Yanguang Wang, Yaoxiang Wang, Hongyao Tu, Zijun Min, Yi Luo
School of Informatics, Xiamen University, China
31520231154319, 31520231154320, 31520231154315, 31520231154305, 23020231154215

Abstract

Grammatical Error Correction (GEC) aims to automatically detect and correct grammatical errors. In this paper, we address the challenge of enhancing type-driven multi-turn error correction systems using a novel approach of distillation. Prior methods often struggled with the integration of raw and pseudo data, preventing models from gradually learning error correction, and neglected the interdependencies between various error types. In contrast, subsequent work introduced the concept of intermediate sentences to train models progressively in error correction and demonstrated that rectifying one error type could enhance the model's ability to predict other error types. This insight has inspired our work, where we harness distillation to allow models that correct one type of error to transmit their knowledge of other error types to other models. In this study, we propose an approach to improve type-driven multi-turn correction by distilling knowledge. We simultaneously train multiple teacher models and a student model, all sharing the same model structure. The student model is trained on raw data, while the teacher models are trained on data corrected for specific error types based on the original data. These teacher models excel at addressing different error types, which significantly narrows the prediction gap between the student and teacher models. Our contributions encompass introducing distillation in the domain of grammatical error correction and substantiating its effectiveness through comprehensive qualitative and quantitative experiments.

Introuction

Grammatical Error Correction (GEC) aims at automatically detecting and correcting grammatical (and other related) errors in a text. It attracts much attention due to its practical applications in writing assistant(Napoles, Sakaguchi, and Tetreault 2017), speech recognition systems (Karat et al. 1999; Wang et al. 2020) etc. Inspired by the success of neural machine translation (NMT), some models adopt the same paradigm, namely NMT-based models. They have been quite successful, especially with data augmentation approach (Boyd 2018; Ge, Wei, and Zhou 2018a; Takahashi, Katsumata, and Komachi 2020). However, these models have been blamed for their inefficiency during inference (Chen et al. 2020; Sun et al. 2021). To tackle this issue, many researchers resort to the sequence-to-label (Seq2Label) formulation, achieving comparable or better performance with efficiency (Malmi et al. 2019; Stahlberg and Kumar 2020; Omelianchuk et al. 2020a).

Despite their success, both NMT-based and Seq2Label models are trained by one-iteration learning, while correcting errors for multiple iterations during inference.As a consequence, they suffer from exposure bias and exhibit performance degrade (Ge, Wei, and Zhou 2018a; Parnow, Li, and Zhao 2021). To deal with this issue, (Ge, Wei, and Zhou 2018a) propose to generate fluency-boost pseudo instances as additional training data. Besides, (Parnow, Li, and Zhao 2021) dynamically augment training data by introducing the predicted sentences with high error probabilities.However, they simply mixed raw data with pseudo data, and the model was unable to gradually learn to correct errors. Moreover, they did not consider the interdependencies between different types of errors. Therefore, (Lai et al. 2022) introduced intermediate sentences to train the model to gradually correct errors, and also demonstrated that after correcting one type of error, the model's ability to predict other types of errors improved. This insight inspired us, as we believe that distillation can effectively leverage this point, enabling a model that first corrects one type of error to pass on better knowledge of other types of errors to other models.

In this paper, we propose to enhance type-driven multi-turn correction through distillation. Specifically, we simultaneously train multiple teacher models and one student model with identical model structures. In terms of data, the student model uses raw data, while the teacher models use data corrected for a specific type of error based on the original data. In other words, they are trained to correct other types of errors. According to previous research, teacher data tends to learn better when addressing other types of errors, thus bridging the gap between the student and teacher model's predictions in this aspect. Overall, our contributions are as follows:

- We introduce a method to improve type-driven multi-turn correction systems using distillation, to the best of our knowledge, for the first time in the field of grammatical error correction.

- Through comprehensive qualitative and quantitative experiments, we explore the effectiveness of distillation in enhancing type-driven multi-turn correction systems and validate our ideas.

Related Work

In the domain of Grammatical Error Correction (GEC), two primary model categories have emerged, namely, Transformer-Dominant Neural Machine Translation (NMT)-Based Models and Seq2Label Models. Furthermore, the works involving Seq2Label Models can be subcategorized into two distinct approaches: those focused on single-turn grammatical error correction and those dedicated to multiple-turn grammatical error correction.

Transformer-Dominant NMT-Based Models

The first category encompasses Transformer-dominant Neural Machine Translation (NMT)-based models that approach GEC as a machine translation task. Noteworthy contributions within this category include works by (Boyd 2018) and (Grundkiewicz, Junczys-Dowmunt, and Heafield 2019). These models take erroneous sentences as input and generate corrected sentences token by token.

Seq2Label Models

single-turn grammatical error correction The second category of models is led by GECToR-based Seq2Label models, with contributions from (Malmi et al. 2019) and (Omelianchuk et al. 2020b). Seq2Label models, particularly GECToR, have demonstrated enhanced efficiency and effectiveness in grammatical error correction. They leverage pretrained language models as encoders to learn word-level representations and employ softmax-based classifiers to predict editing-action labels.

GEC models may not always achieve complete sentence corrections with a single iteration of inference. To overcome this limitation, researchers have explored data augmentation techniques commonly used in the broader field of Natural Language Processing (NLP). (Ge, Wei, and Zhou 2018b) introduced an iterative inference approach and proposed a fluency boost learning method. They pair predicted less fluent sentences with their reference sentences during training, thereby creating new erroneous-reference sentence pairs. The objective is to enhance the model's fluency correction capabilities. Additionally, (Parnow, Li, and Zhao 2021) developed a confidence-based method to address the mismatches between training and inference in Seq2Label models. Their approach involves creating additional training data by pairing low-confidence sentences with reference sentences. This strategy aims to improve the correction accuracy of Seq2Label models. Nonetheless, it is important to note that these two methodologies simply amalgamate pseudo data with the original dataset in a one-iteration learning approach, overlooking the interdependencies between different types of errors.

Multiple Turns Grammatical Error Correction Diverging from the conventional one-iteration approaches, we adopt an iterative method as proposed by (Lai et al. 2022). This approach is engineered to enhance the model's awareness of gradual corrections, thereby fostering its adaptability and progressive improvement. Moreover, traditional multiple turns GEC often neglect the existing interdependence among diverse error types. To address this, our model incorporates the multiple turns GEC strategy to boost performance by discerning and leveraging the interrelatedness of these errors.

Knowledge Distillation

Knowledge distillation is a widely-used technique in machine learning that involves training a smaller, simpler model (referred to as the student) to emulate the larger, more intricate model (the teacher). The objective is to enable the student model to attain a performance level comparable to that of the teacher model, albeit with a lower computational cost. In the sphere of Grammatical Error Correction (GEC), distillation techniques have demonstrated promising outcomes as (Fathullah, Gales, and Malinin 2021). Their research introduced Ensemble Distillation (EnD) and Ensemble Distribution Distillation (EnDD), novel methods that consolidate the ensemble into a singular model. Moreover, (Xia et al. 2022) employed knowledge distillation to compress model parameters, thereby enhancing the model's resilience against attacks. Building on these prior studies, our research applies distillation techniques to train a grammatical error correction model. Our goal is to harness the power of knowledge distillation to devise a robust and efficient model for GEC.

Approach

In this section, we introduce our approach in detail. Our approach involves training multiple teacher models and a student model concurrently. All models will share the same architectural framework to ensure consistency in learning.The teacher models will be trained on pseudo data corrected for specific error types, while the student model will be trained on raw data.

Problem Formulation

In the realm of Grammatical Error Correction (GEC), our approach incorporates token-level transformations and seq2label processes to enhance the accuracy of corrections. This methodology is pivotal in addressing common grammatical errors, such as spelling mistakes, noun number errors, subject-verb agreement issues, and verb form errors.

Token-level Transformations: We define custom token-level transformations $T(x_i)$, which are applied to source tokens $(x_1, x_2, ..., x_N)$ to recover the target text. These transformations are designed to increase the coverage of grammatical error corrections, especially for a limited output vocabulary. Our system includes basic transformations such as KEEP, DELETE, APPEND, and REPLACE, along with specialized g-transformations for tasks like case changing, token merging, and splitting. Additionally, grammatical properties for tokens are encoded in specific transformations like NOUN NUMBER and VERB FORM.

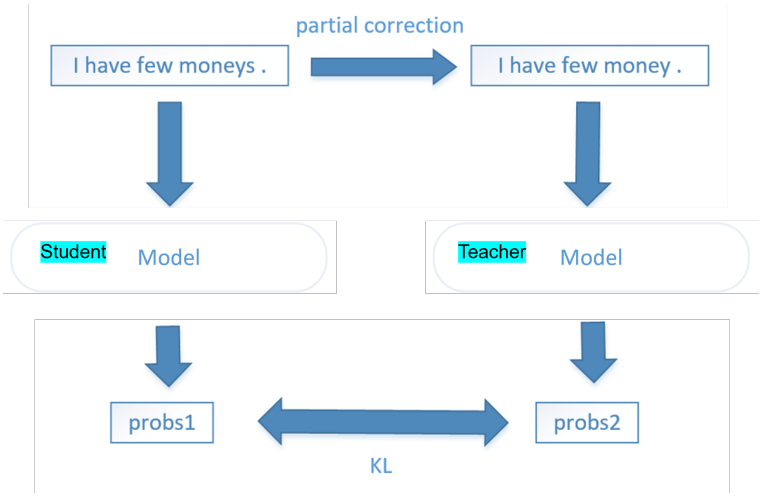

Figure 1: The overall process example of our distillation method. Both the student model and the teacher model use the same model structure, but the parameters are not shared. The teacher model is trained using modified data; the student model is trained using unmodified data, and subsequently, the KL loss is calculated based on the probability distributions output by both models

Seq2label Process: The seq2label process begins by detecting minimal spans of tokens that define the differences between source tokens $(x_1, x_2, ..., x_N)$ and target tokens $(y_1, y_2, ..., y_M)$. For each source token x_i, we search for the best-fitting subsequence of target tokens $\Upsilon_i = (y_{j1}, ..., y_{j2})$ by minimizing a modified Levenshtein distance. This step ensures that each source token is associated with the most appropriate sequence of target tokens. The final step involves selecting one transformation for each source token, with a preference for non-KEEP tags in cases of multiple transformations.

Three Stage Training

Our model employs a comprehensive three-stage training process, leveraging powerful language models such as RoBERTa and XLNet. This training regimen is designed to progressively refine the model's ability to detect and correct grammatical errors in texts. The stages of training are as follows:

1. **Pre-training on Synthetic Errorful Sentences:** In the initial stage, the model undergoes pre-training on a large dataset comprising synthetic errorful sentences. In this stage, approximately 9 million parallel sentences are used. These sentences are synthetically generated to include a wide range of grammatical errors. This extensive pre-training serves as a foundation for the model, equipping it with a preliminary understanding of common grammatical mistakes and their corrections.
2. **Fine-tuning on Errorful-only Sentences:** The second stage involves fine-tuning the model exclusively on errorful sentences. This stage aims to focus the model's learning on the nuances of various grammatical errors without the interference of correct sentences. By doing so, the model develops a more refined ability to identify and correct errors, enhancing its accuracy and efficiency in error correction.
3. **Fine-tuning on a Subset of Errorful and Error-free Sentences:** In the final stage, the model undergoes further fine-tuning on a mixed dataset of both errorful and error-free sentences. This stage introduces the model to a more realistic linguistic environment, where it learns to distinguish between correct and incorrect usage in a more balanced context. This stage is crucial for fine-tuning the model's decision-making process, ensuring it does not overcorrect and maintains the integrity of the original text when no errors are present.

Partially Corrected Distillation

During training stage 2 and stage 3, we employ distillation to enhance performance. The teacher models are trained on data corrected for specific errors. We postulate that through this process, the teacher models acquire additional knowledge beneficial for correcting other errors. This knowledge is transferred to the student models using Kullback-Leibler (KL) divergence. In this transfer, we only align certain output aspects of the teacher and student models, specifically where the teacher's training data remains unmodified. This ensures that the student model learns only the supplementary knowledge related to correcting other errors during distillation. If alignment was done across all aspects, the corrected error outputs in the teacher model's training data would adversely impact the student model's error correction capabilities.

Our Kullback-Leibler loss formula is as follows:

$$D_{KL}(T \parallel S) = \sum_{x \in \mathcal{X}} T(x) \log \left(\frac{T(x)}{S(x)} \right) \quad (1)$$

When calculating the loss, we use bidirectional KL loss for distillation:

Dataset	# sentences	% errorful sentences	Training stage
PIE-synthetic	9,000,000	100.0%	1
Lang-8	947,344	52.5%	2
NUCLE	56,958	38.0%	2
FCE	34,490	62.4%	2
W&I+LOCNESS	34,304	67.3%	2, 3

Table 1: Training datasets. Training stage 1 is pretraining on synthetic data. Training stages 2 and 3 are for fine-tuning.

Model	Pre-trained	BEA-2019(test)			CoNLL-2014(test)		
		Prec.	Rec.	F0.5	Prec.	Rec.	F0.5
GECToR	RoBERTa	77.2	55.1	71.5	72.1	42.0	63.0
	XLNet	79.2	53.9	72.4	77.5	40.1	65.3
GECToR(REPLACE-first)	RoBERTa	81.27	50.67	72.51	77.36	40.35	65.37
	XLNet	81.33	51.55	72.91	77.83	41.82	66.40
GECToR-D(APPEND+DELETE)	RoBERTa	79.85	51.53	72.94(+0.03)	75.39	41.57	65.44(+0.07)
	XLNet	81.14	50.83	72.92(+0.01)	77.08	42.03	66.66(+0.26)
GECToR-D(RAPLACE+DELETE)	RoBERTa	79.39	52.25	72.95(+0.04)	75.70	39.85	65.46(+0.09)
	XLNet	82.35	49.52	72.95(+0.04)	77.05	42.03	66.53(+0.13)
GECToR-D(APPEND+RAPLACE)	RoBERTa	80.31	51.14	72.98(+0.07)	76.77	40.95	65.59(+0.22)
	XLNet	81.89	50.55	**73.04**(+0.13)	78.18	42.67	**67.02**(+0.62)

Table 2: Results of our quantitative experiments. GECToR and GECToR(REPLACE-first) are baselines. GECToR-D(XX+XX) are our methods

$$D_{loss} = \sum_{i=1}^{k} \frac{D_{KL}(T_i \parallel S) + D_{KL}(S \parallel T_i)}{2} \qquad (2)$$

where T denotes the teacher model, S represents the student model, and $k = 2$, as we employ two teacher models.

Experiment

Experiment Setup

As mentioned in *Approach*, our method trains multiple teacher models as well as a student model simultaneously. To ensure the consistency of the learning process, all models follow the same architecture. The teacher model is trained on data corrected for a specific error type, while the student model is trained on initial data without any corrections. Multiple teacher models and student models are optimized through Kullback-Leibler Divergence. In view of the powerful capabilities and wide application of pre-trained language models, we use XLNet and RoBERTa as our backbone models respectively.

XLNet is an advanced language model that redefines how contextual information is captured in text. It innovatively combines autoregressive and autoencoding techniques, utilizing permutation language modeling to predict the next word by considering all possible word permutations in a sentence. With a Transformer-XL architecture, XLNet effectively captures bidirectional context, demonstrating superior performance in various natural language processing tasks.

RoBERTa is an enhanced version of the BERT model, focusing on robust pretraining approaches to refine language representation. By optimizing hyperparameters, leveraging larger training datasets, and employing dynamic masking strategies during pretraining, RoBERTa outperforms BERT in understanding complex language structures and nuances. This model has achieved state-of-the-art results across diverse NLP benchmarks, showcasing its prowess in handling contextual information and providing more robust language representations.

Datasets

To make a fairer comparison, we use the same training data and initialization parameters as (Omelianchuk et al. 2020b) and evaluate the model on BEA-2019(W&I+LOCNESS) dev, test set, and CoNLL-2014 test set. **Table 1** describes the dataset details used in the different training stages of training. For pretraining stage 1, we use 9M parallel sentences with synthetically generated grammatical errors (Awasthi et al. 2019). In fine-tuning phases 2 and 3, we used the following datasets: National University of Singapore Corpus of Learner English (NUCLE), Lang-8 Corpus of Learner English (Lang-8), FCE dataset, the publicly available part of the Cambridge Learner Corpus, and Write & Improve + LOCNESS Corpus.

Metric

We use precision, recall, and harmonic mean F0.5 as evaluation metric.

Precision measures the accuracy of the positive predictions made by the model. It is the ratio of true positive predictions to the total number of positive predictions (true positives + false positives).

Recall (also known as sensitivity or true positive rate) measures the model's ability to correctly identify all positive instances. It is the ratio of true positive predictions to the total number of actual positives (true positives + false negatives).

F0.5 score is the harmonic mean of precision and recall, giving more weight to precision. It combines both metrics into a single value, where an F0.5 score closer to 1 indicates better precision and recall balance.

Baseline

Given that our proposed improved method is inspired by (Lai et al. 2022), and to the best of our knowledge, (Lai et al. 2022), (Omelianchuk et al. 2020b) is the current state-of-the-art method, we choose (Lai et al. 2022), (Omelianchuk et al. 2020b) as our baseline.

GECToR In (Omelianchuk et al. 2020b), the authors introduce a straightforward and effective GEC (Grammar Error Correction) sequence tagger utilizing a Transformer encoder. Their system undergoes pre-training on synthetic data followed by two stages of fine-tuning: initially on error-containing corpora and subsequently on a blend of corpora containing errors alongside error-free parallel data. Custom token-level transformations are devised to align input tokens with target corrections.

GECToR(REPLACE-fitst) In (Lai et al. 2022), additional training instances are constructed from each training instance, incorporating the correction of specific types of errors. These additional instances, along with the original ones, are used to train the model successively. This method trains the model to progressively correct errors and leverage the interdependence between different error types for improved performance.We choose the most advanced experimental results in (Lai et al. 2022) as another baseline, that is, the method of correcting Replace errors first and then correcting other types of errors.

Main Results and Analysis

Table 2 shows our experimental results. For the BEA-2019 dataset, the baseline GECToR models show reasonable performance, with XLNet slightly outperforming RoBERTa in terms of precision, recall, and the F0.5 score. The GECToR-D models demonstrate a marked improvement across all metrics when using XLNet, with the most significant gain observed in precision. Notably, the GECToR-D (APPEND+REPLACE) model achieves the highest F0.5 score when using XLNet as the pre-trained model, indicating that the combination of append and replace operations is particularly effective for this dataset. The gains in the F0.5 score suggest a better balance of precision and recall, skewed towards precision.

Turning to the CoNLL-2014 dataset, the baseline performances are again surpassed by the proposed models. Here, GECToR-D (APPEND+REPLACE) with XLNet attains the highest precision, recall, and F0.5 score among all tested configurations, showing a substantial increase, especially in the F0.5 score. This model's superior performance underscores the efficacy of the combined append and replace strategies for this specific testing set.

The improvement of our methods over the baseline could be attributed to several factors: **(1) Operation Combination:** The integration of different operations like append, replace, and delete likely provides a more nuanced approach to correcting various types of errors, leading to higher precision and recall. **(2) Error Type Sensitivity:** Some GEC errors are better corrected with specific operations. For instance, the APPEND operation might be more suitable for missing word errors, while REPLACE is ideal for substituting incorrect words. The combined strategies cater to a broader range of error types. **(3) Model Synergy:** The pre-trained models, RoBERTa and XLNet, may have inherent strengths that are better leveraged by the proposed methods. For instance, XLNet's permutation-based training might be more adept at handling the order-sensitive nature of append and replace operations. **(4) Contextual Understanding:** The proposed methods may enhance the contextual understanding of sentences, allowing for more accurate corrections that consider the broader textual environment, which is crucial for GEC tasks.

In conclusion, our methods effectively utilize the strengths of pre-trained language models, employing a tailored combination of operations that align with the complexities of grammar error correction. These techniques seem to enhance both the detection (recall) and the correction (precision) of grammatical errors, as reflected in the improved F0.5 scores.

Conclusion

In this paper, we introduce a novel approach to enhance type-driven multi-turn error correction systems using a distillation-based method. Our method simultaneously trains multiple teacher models, each specializing in correcting specific error types, along with a student model on raw data. The distillation process allows the knowledge gained by the teacher models in correcting one type of error to be transmitted to the student model, narrowing the prediction gap.

Through comprehensive experiments on BEA-2019 and CoNLL-2014 datasets, we demonstrate the effectiveness of our approach. The results show significant improvements in precision, recall, and F0.5 score compared to the baseline models.

Our work not only introduces distillation to the domain of grammatical error correction but also provides insights into the interdependence between different error types. The results suggest that a model trained to correct one type of error can transfer knowledge beneficial for correcting other types of errors.

In future work, we aim to explore additional strategies for error correction and investigate the generalization capabilities of the proposed method across diverse datasets. Additionally, we plan to analyze the impact of hyperparameter tuning and model architecture variations on the overall performance. Overall, our work opens avenues for further research in the intersection of grammatical error correction, distillation, and multi-turn error correction systems.

References

[Awasthi et al. 2019] Awasthi, A.; Sarawagi, S.; Goyal, R.; Ghosh, S.; and Piratla, V. 2019. Parallel iterative edit models for local sequence transduction. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, 4259–4269. Association for Computational Linguistics.

[Boyd 2018] Boyd, A. 2018. Using wikipedia edits in low resource grammatical error correction. In Xu, W.; Ritter, A.; Baldwin, T.; and Rahimi, A., eds., *Proceedings of the 4th Workshop on Noisy User-generated Text, NUT@EMNLP 2018, Brussels, Belgium, November 1, 2018*, 79–84. Association for Computational Linguistics.

[Chen et al. 2020] Chen, M.; Ge, T.; Zhang, X.; Wei, F.; and Zhou, M. 2020. Improving the efficiency of grammatical error correction with erroneous span detection and correction. *Cornell University - arXiv,Cornell University - arXiv*.

[Fathullah, Gales, and Malinin 2021] Fathullah, Y.; Gales, M. J.; and Malinin, A. 2021. Ensemble distillation approaches for grammatical error correction. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2745–2749. IEEE.

[Ge, Wei, and Zhou 2018a] Ge, T.; Wei, F.; and Zhou, M. 2018a. Fluency boost learning and inference for neural grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

[Ge, Wei, and Zhou 2018b] Ge, T.; Wei, F.; and Zhou, M. 2018b. Fluency boost learning and inference for neural grammatical error correction. In Gurevych, I., and Miyao, Y., eds., *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, 1055–1065. Association for Computational Linguistics.

[Grundkiewicz, Junczys-Dowmunt, and Heafield 2019] Grundkiewicz, R.; Junczys-Dowmunt, M.; and Heafield, K. 2019. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In Yannakoudakis, H.; Kochmar, E.; Leacock, C.; Madnani, N.; Pilán, I.; and Zesch, T., eds., *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, BEA@ACL 2019, Florence, Italy, August 2, 2019*, 252–263. Association for Computational Linguistics.

[Karat et al. 1999] Karat, C.-M.; Halverson, C.; Horn, D.; and Karat, J. 1999. Patterns of entry and correction in large vocabulary continuous speech recognition systems. In *Proceedings of the SIGCHI conference on Human factors in computing systems the CHI is the limit - CHI '99*.

[Lai et al. 2022] Lai, S.; Zhou, Q.; Zeng, J.; Li, Z.; Li, C.; Cao, Y.; and Su, J. 2022. Type-driven multi-turn corrections for grammatical error correction. *arXiv preprint arXiv:2203.09136*.

[Malmi et al. 2019] Malmi, E.; Krause, S.; Rothe, S.; Mirylenka, D.; and Severyn, A. 2019. Encode, tag, realize: High-precision text editing. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, 5053–5064. Association for Computational Linguistics.

[Napoles, Sakaguchi, and Tetreault 2017] Napoles, C.; Sakaguchi, K.; and Tetreault, J. 2017. Jfleg: A fluency corpus and benchmark for grammatical error correction. *Cornell University - arXiv,Cornell University - arXiv*.

[Omelianchuk et al. 2020a] Omelianchuk, K.; Atrasevych, V.; Chernodub, A.; and Skurzhanskyi, O. 2020a. Gector – grammatical error correction: Tag, not rewrite. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*.

[Omelianchuk et al. 2020b] Omelianchuk, K.; Atrasevych, V.; Chernodub, A. N.; and Skurzhanskyi, O. 2020b. Gector - grammatical error correction: Tag, not rewrite. In Burstein, J.; Kochmar, E.; Leacock, C.; Madnani, N.; Pilán, I.; Yannakoudakis, H.; and Zesch, T., eds., *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications, BEA@ACL 2020, Online, July 10, 2020*, 163–170. Association for Computational Linguistics.

[Parnow, Li, and Zhao 2021] Parnow, K.; Li, Z.; and Zhao, H. 2021. Grammatical error correction as gan-like sequence labeling. In Zong, C.; Xia, F.; Li, W.; and Navigli, R., eds., *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, 3284–3290. Association for Computational Linguistics.

[Stahlberg and Kumar 2020] Stahlberg, F., and Kumar, S. 2020. Seq2edits: Sequence transduction using span-level edit operations. In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, 5147–5159. Association for Computational Linguistics.

[Sun et al. 2021] Sun, X.; Ge, T.; Wei, F.; and Wang, H. 2021. Instantaneous grammatical error correction with shallow aggressive decoding. *Cornell University - arXiv,Cornell University - arXiv*.

[Takahashi, Katsumata, and Komachi 2020] Takahashi, Y.; Katsumata, S.; and Komachi, M. 2020. Grammatical error correction using pseudo learner corpus considering learner's error tendency. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*.

[Wang et al. 2020] Wang, H.; Dong, S.; Liu, Y.; Logan, J.; Agrawal, A. K.; and Liu, Y. 2020. Asr error correction with augmented transformer for entity retrieval. In *Interspeech 2020*.

[Xia et al. 2022] Xia, P.; Zhou, Y.; Zhang, Z.; Tang, Z.; and Li, J. 2022. Chinese grammatical error correction based on knowledge distillation. *arXiv preprint arXiv:2208.00351*.