TGEA: An Error-Annotated Dataset and Benchmark Tasks for Text Generation from Pretrained Language Models

Jie He[†*], **Bo Peng**[§*], **Yi Liao**[§], **Qun Liu**[§] and **Deyi Xiong**[†]
[†] College of Intelligence and Computing, Tianjin University, Tianjin, China
[§] Huawei Noah's Ark Lab, Hong Kong, China
{jieh, dyxiong}@tju.edu.cn,
{peng.bo2, liaoyi9, qun.liu}@huawei.com

Abstract

In order to deeply understand the capability of pretrained language models in text generation and conduct a diagnostic evaluation, we propose TGEA[1], an error-annotated dataset with multiple benchmark tasks for text generation from pretrained language models (PLMs). We use carefully selected prompt words to guide GPT-2 to generate candidate sentences, from which we select 47K for error annotation. Crowdsourced workers manually check each of these sentences and detect 12k erroneous sentences. We create an error taxonomy to cover 24 types of errors occurring in these erroneous sentences according to the nature of errors with respect to linguistics and knowledge (e.g., common sense). For each erroneous span in PLM-generated sentences, we also detect another span that is closely associated with it. Each error is hence manually labeled with comprehensive annotations, including the span of the error, the associated span, minimal correction to the error, the type of the error, and rationale behind the error. Apart from the fully annotated dataset, we also present a detailed description of the data collection procedure, statistics and analysis of the dataset. This is the first dataset with comprehensive annotations for PLM-generated texts, which facilitates the diagnostic evaluation of PLM-based text generation. Furthermore, we use TGEA as a benchmark dataset and propose a series of automatic diagnosis tasks, including error detection, error type classification, associated span detection, error rationale generation, to further promote future study on the automatic error detection and correction on texts generated by pretrained language models.

1 Introduction

Pretrained language models (Devlin et al., 2019; Liu et al., 2019; Raffel et al., 2020; Brown et al., 2020), which are trained on a huge amount of data via self-supervised learning, have made remarkable progress on both natural language understanding (NLU) (Wang et al., 2018, 2019) and natural language generation (NLG) (Liu and Lapata, 2019; Weng et al., 2020; Cao et al., 2020).

On several NLU datasets, PLM-based neural models have gradually achieved human-level performance in terms of automatic evaluation metrics (e.g., accuracy, F_1) (He et al., 2020; Zhang et al., 2021). In order to deeply understand and analyze the capability of PLMs on NLU, a variety of more challenging NLU datasets have been proposed (Warstadt et al., 2020; Cui et al., 2020a; Jain et al., 2020; Talmor et al., 2020). These datasets can be used not only to obtain knowledge on how PLM-based models work and what they learn, but also to define new NLU tasks and to serve as a benchmark for future progress. For example, evaluating and analyzing PLM-based models on learning document structures with a carefully created benchmark test suite (Chen et al., 2019), helps to develop new methods to enhance the capability of these models on discourse modeling (Iter et al., 2020). Knowing the weakness of current PLM-based models in commonsense reasoning (Zhou et al., 2020) has inspired people to develop various reasoning datasets (Cui et al., 2020a; Zhang et al., 2020b).

On the other hand, state-of-the-art PLMs are able to generate texts that are even not distinguishable from human-written texts by human evaluators (Radford et al., 2019; Brown et al., 2020). This makes us curious about the capability of PLMs on text generation. Are they really reaching human-level performance on text generation? In contrast to the studies of PLMs on NLU, research on the

[*] Equal Contributions.
[1] The The dataset is available at https://download.mindspore.cn/dataset/TGEA/.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 6012–6025
August 1–6, 2021.

capability of PLMs on NLG is quite limited, especially in dataset building and diagnostic evaluation of text generation errors.

In this paper, in order to recognize the perimeter of text generation capability of PLMs, we propose TGEA, an error-annotated dataset with multiple benchmark tasks for text generation from pretrained language models. The original raw data are collected from texts generated by a Chinese GPT-2 model. The entire data collection and annotation procedure is visualized in Figure 1. The goals and contributions of building TGEA are as follows.

- TGEA, to the best of our knowledge, is the first dataset built on machine-generated texts from state-of-the-art pretrained language models with rich annotations. The key interest of this dataset is detecting and annotating text generation errors from PLMs. Therefore it is different from conventional text generation datasets (e.g., Multi-News (Fabbri et al., 2019), TextCaps (Sidorov et al., 2020)) that are constructed to train models to learn text generation (e.g., generating texts from images or long documents). It is also different from grammatical error correction (GEC) datasets (Zhao et al., 2018; Flachs et al., 2020) that are built from human-written texts usually by second language learners.
- TGEA provides rich semantic information for text generation errors, including error types, associated text spans, error corrections and rationals behind errors, as shown in Figure 1. Marking text spans that are closely related to erroneous words allows us to detect long-distance dependencies of errors or reasoning chains related to errors. Rationales behind errors directly explain why errors are annotated. All these error-centered manual annotations not only increase the interpretability of our dataset, but also facilitate a comprehensive diagnostic evaluation of pretrained language models on text generation.
- We created an error taxonomy for TGEA, which covers 24 error types in a two-level hierarchy. With this error taxonomy, we not only obtain a high agreement on manual error annotation but also recognize the strengths and weaknesses of GPT-2 on text generation by estimating a distribution over these 24 error types. Comparing our dataset with GEC datasets, we find that humans and GPT-2 have a very different error distribution, especially on errors related to commonsense reasoning.
- TGEA not only exhibits text generation errors from pretrained language models, but also can serve as a dataset to train various models to automatically detect and correct these errors, like GEC datasets for training models to automatically correct human errors. We define 5 benchmark tasks over our dataset, i.e., erroneous sentence detection, erroneous span and associated span detection, error type classification, error correction and error rationale generation. For all these tasks, we provide experimental results using state-of-the-art models as baselines.

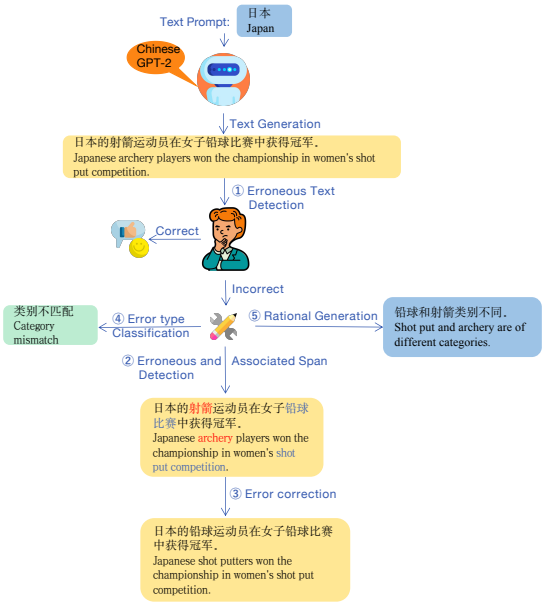

Figure 1: The different stages of the annotation process for each machine-generated text according to the prompt in TGEA. Better viewed in color.

2 Related Work

Our work is related to GEC datasets in error annotation and correction (machine vs. human errors). It is also partially related to commonsense reasoning datasets that have been proposed recently in that our dataset includes commonsense reasoning errors and rationales behind these errors. Our dataset is not related to conventional text generation datasets (Vougiouklis et al., 2017; Wiseman et al., 2017; Parikh et al., 2020) for training text generation models. A comprehensive comparison to GEC datasets and commonsense reasoning datasets is shown in Table 1.

Dataset	Task	Commonsense Reasoning	Rationales	Machine-Generated Texts	Domain	#Sentences	Language
FCE	GEC	✖	✖	✖	Essay	34K	EN
AESW	GEC	✖	✖	✖	Journal articles	1.2M	EN
JFLEG	GEC	✖	✖	✖	TOFEL Exam	1,511	EN
CMEG	GEC	✖	✖	✖	Web doc/Essay	8K	EN
CWEB	GEC	✖	✖	✖	Web doc	13K	EN
CGEC	GEC	✖	✖	✖	Essay	0.71M	ZH
WSC	Coreference Resolution	✓	✖	✖	Open	273	EN
HellaSwag	Plausible Inference	✓	✖	✓	WikiHow articles	70K	EN
Social IQA	Question Answering	✓	✖	✖	Social situations	38K	EN
CosmosQA	Reading comprehension	✓	✖	✖	Narratives	35K	EN
PIQA	Plausible Inference	✓	✖	✖	Physical situations	21K	EN
Abductive NLI	Plausible Inference	✓	✖	✖	ROCStories	200K	EN
WinoWhy	Reason Explanation	✓	✓	✓	Open	2,865	EN
TGEA (ours)	Multiple tasks	✓	✓	✓	Open	47K	ZH

Table 1: Comparison between our dataset and other datasets.

2.1 Grammatical Error Correction Datasets

FCE (Yannakoudakis et al., 2011) is an early large-scale English grammatical error correction dataset, where raw texts are produced by English learners taking the First Certificate in English exams. AESW (Daudaravicius et al., 2016) is a GEC dataset from a professional editing company. In addition to common grammatical errors, AESW covers style issues as it contains texts mainly from scholarly papers. JFLEG (Napoles et al., 2017) is a GEC dataset built from TOFEL Exams, which does not force annotators to make minimal edits, preferring holistic fluency rewrites. CMEG (Napoles et al., 2019) is different from general grammatical error correction datasets with texts from second language learners. It uses articles or blogs (e.g., Wiki, Yahoo)) written by native English speakers to explore grammatical error phenomena in different domains. CWEB (Flachs et al., 2020) also uses website texts in English, such as blogs. The difference between CWEB and CMEG is that the percentage of erroneous tokens in the former is smaller than the latter as the purpose of CWEB is to study grammatical error correction in low error density domains. CGEC (Zhao et al., 2018) is a large-scale Chinese grammatical error correction dataset, derived from wrong sentences written by Chinese learners in the process of learning Chinese as a second language.

In addition to the difference in text sources (i.e., human-written vs. machine-generated), other significant differences between our dataset and existing GEC datasets are that our dataset contains commonsense reasoning errors and provides associated text span annotations and rationales for errors, as shown in Table 1.

2.2 Commonsense Datasets

A variety of commonsense datasets have been proposed. Roemmele et al. (2011) introduce COPA that focuses on commonsense causal reasoning. Levesque et al. (2012) present Winograd Scheme Challenge (WSC), a dataset testing commonsense reasoning in the form of anaphora resolution. Winogrande, a larger version of WSC, is introduced by Sakaguchi et al. (2020), which contains $\sim 44,000$ examples. Winowhy (Zhang et al., 2020a) asks annotators to provide reasons for their decisions to WSC. In this aspect, the differences of our dataset from Winowhy are twofold. First, we provide reasons for errors rather than correct decisions to anaphora. Second, we provide reasons for all text generation errors, rather than only errors related to commonsense reasoning.

In addition to COPA and WSC-style datasets, many large crowdsourced datasets have been also proposed recently. CommonsenseQA (Talmor et al., 2019), a commonsense question answering dataset, has been constructed from ConceptNet. HellaSwag (Zellers et al., 2019b) and Abductive NLI (Bhagavatula et al., 2020) evaluate commonsense reasoning in the form of natural language inference. CosmosQA (Huang et al., 2019) is a dataset with multi-choice questions that require commonsense reading comprehension.

Beyond datasets for evaluating commonsense reasoning, there are other datasets providing commonsense knowledge. PIQA (Bisk et al., 2020) focuses on physical commonsense knowledge while SocialIQA (Sap et al., 2019) on social commonsense knowledge.

Commonsense datasets in multiple languages or languages other than English have also been created recently. XCOPA (Ponti et al., 2020) is a multilingual dataset for causal commonsense reasoning in 11 typologically different languages. Chinese

Level-1 Error Type	Example
Inappropriate combination	医生当即将刘莉的手术[囊肿]切除，并建议患者住院观察。 The doctor removed Liu Li's surgery [tumor] and suggested that the patient be hospitalized for observation.
Missing	在这里,有众多新闻记者和游客参加 [活动]。 Here, many journalists and tourists are taking part in [activities].
Redundancy	一些企业减员增效[]增效,使得企业利润增长了10%以上。 Some enterprises have reduced staff and increased efficiency[] increased efficiency, making their profits increase by more than 10%.
Discourse Error	他说自己最喜欢安阳的乡间小路，是最美的山峦 [路]。 He said that he likes the country roads in Anyang best, and it is the most beautiful mountain [road].
Commonsense Error	在国际市场上，如果信用等级越高 [低]，投资者在投资时就越不会太放心。 In the international market, the higher [lower] the credit rating, the less reassured investors are.

Table 2: Examples of level-1 error types in TGEA. Underwaved words are erroneous words while underlined words are associated words. Words in "[]" are corrections to erroneous words.

commonsense datasets, such as Mandarinograd (Bernard and Han, 2020) consisting of 154 Chinese Winograd scheme examples and CLUEWSC2020 (Xu et al., 2020) containing 1838 Winograd scheme examples, have been proposed.

In the aspect of commonsense reasoning, our dataset is different from the mentioned commonsense datasets in that we detect and annotate errors in machine-generated texts, which violates common sense, rather than creating examples to examine the commonsense reasoning ability of machines.

3 Dataset Creation

3.1 Error Taxonomy

Before crowdsourced workers manually annotate errors in machine-generated texts, we need to create an error taxonomy for such error coding. Three principles are used to guide the design of the error taxonomy: coverage, exclusiveness and easiness. The coverage rule requires that the error system can cover almost all different types of errors in machine-generated texts. The exclusiveness requirement indicates that each error type is not overlapping with other error types in the taxonomy. The final easiness principle means that the error coding system is easy to be used by annotators. With these three principles and aid from a linguist, we created an error taxonomy in a two-level hierarchy, which was revised in our pre-annotation stage.

The first level of the error taxonomy includes 5 error types.

- *Inappropriate combination.* This type of errors suggests that two words/phrases are syntactically or lexically inappropriately combined in a sentence. Such errors include not only lexical collocation errors but also long-distance syntactic constituency combination errors (e.g., inappropriate subject-object combination). This error type is similar to "replacing" error in some GEC datasets (e.g., CWEB (Flachs et al., 2020)) as one element of an inappropriate combination should be usually replaced with other expressions. As we want to find text spans associated with erroneous words/phrases, we term this error type as "inappropriate combination". We further divide this error type into five subtypes at the second level.
- *Missing.* Grammatical constituencies or words are missing. 5 subtypes are defined under this error type.
- *Redundancy.* Words or phrases are unnecessary. 5 subtypes are also defined.
- *Discourse Error.* This error type is defined for inter-sentential cohesion/coherence errors (e.g., coreference errors, incorrect discourse connectives).
- *Commonsense Error.* This error code is for errors related to commonsense reasoning. We divide this error type into 8 subtypes according to the type of commonsense knowledge type required (e.g., time, spatial, number).

All other errors that cannot be categorized into the aforementioned error types are grouped into "Other". Table 2 displays examples for the above defined error types. 24 error subtypes are displayed in Figure 2 and examples of these subtypes are shown in Appendix.

3.2 Machine-Generated Text Collection

Raw texts in our dataset are collected from a pre-trained Chinese GPT-2 (NEZHA-Gen)[2], which generates texts according to a system prompt. NEZHA-Gen has 12 layers and 12 attention heads and is trained on Chinese Wikipedia and news data (see Appendix for more details on the hyperparameters of NEZHA-Gen). As it is easy for NEZHA-Gen to generate high-quality texts with high-frequency prompt words, we create a list of prompt words according to their frequency to guarantee that there are sufficient erroneous sentences in collected raw texts. By doing so, we have found that GPT has a better chance to generate wrong sentences with such prompts. Specifically, we have randomly sampled 2M sentences from the data used to train NEZHA-Gen. The sampled sentences are then word-segmented and POS-tagged by Baidu LAC tool[3] (Jiao et al., 2018). We then select and sort nouns in a descending order according to their frequencies in the sampled corpus. Nouns ranking in the range of top [40%, 60%] are selected as prompts.

We further filter out noisy texts from texts generated with these selected prompts. Noisy texts are either texts containing no more than 15 characters or texts where Chinese characters account for less 70% of all characters.

3.3 Error Annotation

There are 5 stages in error annotation, as shown in Figure 1. We introduce each of them in this subsection.

(1) **Erroneous text detection.** Texts generated by NEZHA-Gen with prompt words are present to annotators one by one. The first stage of annotation is hence to detect erroneous texts for subsequent annotations. Corresponding tags are annotated for texts being manually checked.

(2) **Erroneous and associated span detection.** The next task for annotators is to detect erroneous and associated text spans in detected erroneous texts. For erroneous span detection, as a text may contain several spans that can be edited or the text can be corrected in different ways, which span should be regarded as erroneous is closely related to the way that we correct the text. Therefore, the basic principle that guides the annotation of erroneous spans is also the rule that we use for error correction: making minimal edits, which is also used in GEC datasets (Flachs et al., 2020; Napoles et al., 2017). In addition to the minimal edit principle, we also provide the following specific rules for annotators:

- If annotators feel that a text is ambiguous and that it is difficult to correct the text, the text can be discarded without any further annotations.
- If there are several spans that can be edited, the first erroneous span is preferred to be edited.
- If the number of errors to be corrected in a text is larger than 4, the text is removed.

Following these rules, annotators have removed 4,291 texts, which account for only 8.36% of all detected erroneous texts in the first stage.

In addition to erroneous span annotation, unlike GEC datasets (Daudaravicius et al., 2016; Zhao et al., 2018), we also detect a text span that is closely related to the already detected erroneous span with respect to the error, and term this span as "associated span". In Table 2, we show examples with annotated erroneous and associated text spans. For an inappropriate combination, the associated span is usually a span that should not co-occur with the erroneous span.

(3) **Error correction.** After detecting erroneous spans in a given text, annotators are required to make corrections following the minimal edit principle. Annotators are also required to use common words for error correction to make the corrected text as fluent as possible.

(4) **Error type classification.** Once annotators detect both erroneous and associated spans as well as provide corrections, they are becoming quite aware of these errors. Hence, we now ask them to categorize the annotated errors into error types defined in our error taxonomy. First, they select the primary type from the level-1 error types. Then, if there are level-2 error subtypes, annotators continue to select a subtype. We observe that errors annotated with "other" only account for 5.70%, suggesting that our error taxonomy has good coverage.

(5) **Rationale generation.** Partially inspired by previous datasets that provide explanations together with corresponding annotations, e.g., e-SNLI (Camburu et al., 2018), Winowhy (Zhang et al., 2020a)

[2] github.com/huawei-noah/Pretrained-Language-Model/tree/master/NEZHA-Gen-TensorFlow

[3] github.com/baidu/lac

Task	IAA (%)	Kappa (%)
Erroneous text detection	87.5	62.1
Erroneous and associated span detection	51.2	–
Error type classification	73.3	55.7

Table 3: Inter-annotator agreement results.

and R4C (Inoue et al., 2020), we ask annotators to give a reason for each error to justify their annotations. To the best of our knowledge, no GEC datasets provide explanations for error corrections. We believe that annotated rationales can be used to improve the interpretability of neural models trained on our dataset.

3.4 Annotation Quality Control

In order to ensure the quality of error annotations, we have adopted a very strict quality control protocol during annotation. First, we train two reviewers with 1K machine-generated texts. The annotation consistency of the two reviewers on the 1K texts is very high, with an average IAA of 92.3% and Cohen's Kappa (McHugh, 2012) of 82.6% across the annotation tasks (1), (2) and (4). For the texts annotated by the two reviewers, we have conducted an evaluation. The average accuracy of all tasks is 96.3% and 97.4% respectively.

Second, 200 candidate workers participate in a pre-annotation stage. The two reviewers will review annotations from these participants to distinguish whether the annotation is correct or not. Only participants who have reached an accuracy of >90% in every tasks can join in the next stage. As a result, 20 participants have passed the training in the pre-annotation stage. We then divide them into two groups and ask them to annotate the same 500 texts. The inter-annotator IAA and Cohen's Kappa are shown in Table 3, which suggests that the 20 annotators are ready for final annotation.

Third, in order to further ensure annotation quality, we have carried out iterative verification and amendment. The two reviewers will review each annotated text. If they found the annotation is wrong, the unqualified data will be returned for amendment until they are qualified.

Following this strict quality control protocol, we complete the annotation on 47K selected machine-generated texts. We randomly sample 1K annotated texts. The average accuracy over the three tasks (i.e., (1), (2) and (4)) is 89.6%, 88.5%, 84.3% respectively.

	Train	Dev	Test	All
#text	37,646	4,706	4,706	47,058
w/ 0 error	27,906	3,488	3,488	34,882
w/ 1 error	8,413	1,055	1,052	10,520
w/ 2 error	1,169	141	149	1,459
w/ 3 error	141	18	15	174
w/ 4 error	17	4	2	23
Tokens	966,765	120,889	121,065	1,208,719
Vocab	44,598	16,899	16,745	48,547
Avg. tokens	25.68	25.69	25.73	25.68
Avg. t.err	2.92	3.09	2.95	2.94
Avg. t.assoc	4.30	4.39	3.89	4.27
Avg. d.e-a	6.99	7.29	7.10	7.03
Avg. t.rationale	8.74	8.72	8.75	8.74

Table 4: Data statistics of TGEA. Avg.t.err/Avg.t.assoc: the average number of tokens in erroneous/associated text spans. Avg.t.rationale: the average number of tokens in rationales. Avg.d.e-a: the average distance between a erroneous span and its associated span.

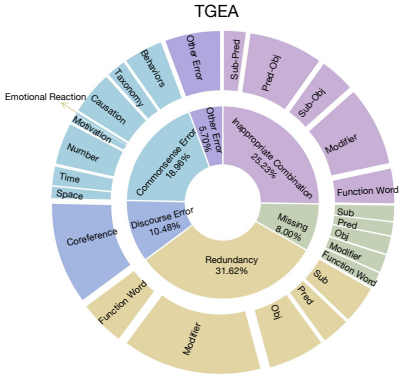

Figure 2: Distribution over the level-1 and level-2 error types in TGEA.

4 Dataset Analysis

4.1 Dataset Statistics

Overall statistics. We reshuffle all annotated texts and divide them into the training/dev/test sets with a proportion of 8:1:1. As shown in Table 4, the training set contains 27,096 correct texts and 9,740 erroneous texts. Both the development and test set contain 4,706 texts, among which 1,218 texts are erroneous. Not surprisingly, most erroneous texts contain only one error.

After Chinese word segmentation via Jieba[4], there are 1,208,719 tokens in total. On average, there are 25.68 tokens in each text.

Annotation statistics. As shown in Table 4, each erroneous text span contains 2.94 tokens while each associated span is composed of 4.27 tokens. The average distance from an erroneous text span to its associated span is 7.03 tokens, which is about 1/3 of the average text length.

[4] github.com/fxsjy/jieba

4.2 Error Type Distribution

We further show the percentages of both level-1 and level-2 error types in Figure 2. We observe that only 5.7% cases cannot be categorized into our defined error types. The inappropriate combination, missing and redundancy error, which are the main error types in GEC datasets, account for 64.85% in our dataset. In addition to these errors, we see 18.96% commonsense errors and 10.48% discourse errors, which are usually not very common in GEC datasets. However, these two types of errors with high percentages in our dataset suggest that pretrained language models can be further improved on both commonsense reasoning and discourse modeling.

5 TGEA as a Benchmark

We use our dataset as a benchmark and propose 5 tasks that are defined for errors in texts generated by PLMs. We provide baseline results for these tasks in this section.

We employ three BERT-style Chinese PLMs as baselines in our experiments, namely BERT-wwm-ext, RoBERTa-wwm-ext-large developed by Cui et al. (2020b) [5] and ALBERT-Chinese-large[6]. For notational simiplicity, we denote them as $\mathrm{BERT}_{\mathrm{zh}}$, $\mathrm{RoBERTa}_{\mathrm{zh}}$ and $\mathrm{ALBERT}_{\mathrm{zh}}$ respectively. Please refer to the Appendix for the model hyperparameter settings of each task.

5.1 Erroneous Text Detection

Task definition. This is a text classification task to judge whether a given text is erroneous. In order to avoid data imbalance, we use the same number of correct and erroneous texts for training.

Model. The three Chinese PLMs are used with standard text-classification fine-tuning.

Results. All models perform just $<$14% better than chance (random guessing), as shown in Table 5. We also provide human performance on this task. The best model $\mathrm{RoBERTa}_{\mathrm{zh}}$ is worse than human performance by 26 points. This suggests that automatically detecting erroneous texts generated by pretrained language models is very challenging even in the balanced classification scenario.

[5] github.com/ymcui/Chinese-BERT-wwm
[6] huggingface.co/voidful/albert_chinese_large

5.2 Erroneous Span and Associated Span Detection

Task definition. We define the detection of the two types of spans as a joint task as they are closely related to each other. The joint task is similar to named entity recognition (NER) (a sequence labeling task) and it requires to recognize the erroneous and associated text spans simultaneously. NER-style word-level tags are hence annotated for each erroneous text.

Model. The three Chinese PLMs with NER-like fine-tuning are evaluated for this task. Since this is a 3-class token classification task, we report class-F_1 on erroneous and associated span. The class-F_1 on class X is calculated like a normal F_1 for a binary classification task, by treating the target class X as the positive class and all other classes as negative.

Results. As shown in Table 5, all models are very poor in this task, indicating the difficulty of automatically detecting erroneous and associated spans. However, we have found that models can benefit much from the joint detection over the detection of a single type of span (either erroneous or associated span). Our preliminary experiments on the detection of only erroneous span show that the best model can only achieve 26.42% erroneous class-F_1 on the test set, while the joint task achieves 27.66% erroneous class-F_1 on the test set.

5.3 Error Type Classification

Task definition. Again this is a text classification task. We only perform classification over level-1 error types in the form of 5-way classification.

Model. We use models similar to the first task.

Results. The overall accuracy and Macro-F_1 (shown in Table 5) are very low. However, we find some error types are easier than others. The accuracy on the classification of redundancy errors is 53.91%, the highest among all error types.

5.4 Error Correction

Task definition. This task is the same as GEC, which transforms an erroneous text into a correct sequence.

Model. we use the state-of-the-art BERT-GEC model (Kaneko et al., 2020) as the baseline for this task, which is an encoder-decoder model using representations learned by PLMs as additional inputs. Following Wang et al. (2020), we feed representations learned by $\mathrm{BERT}_{\mathrm{zh}}$ and $\mathrm{RoBERTa}_{\mathrm{zh}}$ into

Task	Model	Dev			Test		
		Accuracy (%)			Accuracy (%)		
Erroneous text detection	Random	50.00			50.00		
	$\text{ALBERT}_{\text{zh}}$	63.59			63.30		
	BERT_{zh}	65.15			64.94		
	$\text{RoBERTa}_{\text{zh}}$	66.67			66.79		
	Human	92.35			93.57		
		Erroneous class-F_1 (%)		Associated class-F_1 (%)	Erroneous class-F_1 (%)		Associated class-F_1 (%)
Erroneous and associated span detection	Random	01.71		04.23	01.74		04.22
	$\text{ALBERT}_{\text{zh}}$	27.36		27.44	28.10		26.24
	BERT_{zh}	27.85		26.93	27.66		25.30
	$\text{RoBERTa}_{\text{zh}}$	28.17		27.08	27.75		27.12
		Accuracy (%)		Macro-F_1 (%)	Accuracy (%)		Macro-F_1 (%)
Error type classification	Random	24.25		20.00	24.25		20.00
	$\text{ALBERT}_{\text{zh}}$	34.76		21.04	34.38		20.56
	BERT_{zh}	44.35		33.01	41.31		31.05
	$\text{RoBERTa}_{\text{zh}}$	44.44		36.10	44.16		37.20
		P (%)	R (%)	$\text{F}_{0.5}$ (%)	P (%)	R (%)	$\text{F}_{0.5}$ (%)
Error correction	BERT_{zh}-GEC	0.62	6.49	0.76	0.60	6.30	0.74
	$\text{RoBERTa}_{\text{zh}}$-GEC	0.78	4.07	0.93	0.82	4.15	0.98
		BLEU	Rouge-L	BERT_Score	BLEU	Rouge-L	BERT_Score
Rationale generation	NEZHA-Gen	0.06%	9.17%	56.58%	0.06%	9.02%	56.17%

Table 5: Performance of benchmark models on the development and test set.

the BERT-GEC model.

Results. We report precision, recall and $\text{F}_{0.5}$ scores using the official Max-Match tool (Dahlmeier and Ng, 2012). As shown in Table 5, the best $\text{RoBERTa}_{\text{zh}}$-GEC model achieves a very low $\text{F}_{0.5}$ of 0.93% and 0.98% on the development and test set respectively. We speculate that the reasons for this are twofold. First, comparing with GEC data on human-written texts, our dataset is relatively small. Second, our dataset contains error types that are very different from those in previous GEC datasets (Zhao et al., 2018; Flachs et al., 2020). Punctuation, spelling and other word-character-level errors, which are easy to be corrected, are rare in TGEA although they are quite common in GEC datasets. In contrast, TGEA contains more complicated errors that can only be corrected with knowledge of common sense, long-distance or inter-sentential dependencies, etc.

5.5 Rationale Generation

Task definition. This is a text generation task that directly generate an explanation with respect to text generation errors from an erroneous text.

Model. We use NEZHA-Gen as the baseline for this task. We restructure annotated texts in our dataset in the form of $\{T, \text{这句话错误的原因是：}, R\}$ ($\{T$, The reason behind the errors in this sentence is:, $R\}$), where T is an erroneous sentence, while R is the error rational provided by annotators. We then fine-tune NEZHA-Gen on the reformatted training set and evaluate the fine-tuned model on the reformatted development and test set. We report BLEU (Papineni et al., 2002), Rouge-L (Lin, 2004) and BERT_Score (Zhang et al., 2020c).

Results. It can be expected that results in these metrics will be very low due to the high difficulty of this task. We analyze generated texts from the baseline and find that generated rationales are usually much longer than reference rationales provided by human annotators. This could result in the low BLEU score since long hypotheses are penalized in BLEU computation. We also experiment zero-shot generation on the test set. The results are $\{\text{BLEU} = 0.04\%, \text{Rouge-L} = 6.83\%, \text{BERT\_Score} = 54.27\%\}$, indicating that fine-tuning on the annotated training set can improve this task. We suggest that this generation task could be reformulated as a multi-choice question answering task by providing alternative rationales as distractors, similar to VCR (Zellers et al., 2019a). We leave this to our future work.

6 Discussion

Since we use machine-generated texts for error annotation, hyperparameters of models (e.g., sampling strategies, model size), model types (e.g., GPT-2, GPT-3 or other PLMs for text generation), and genres of texts used to train PLMs, etc., all

have impacts on generated texts and hence on error types and error distribution.

A straightforward way to mitigate this issue is to collect raw texts from multiple models with different hyperparameters, neural architectures and text genres. This will lead to an expanded dataset with a much larger number of instances to be manually annotated, which is expensive and time-consuming. Yet another issue with this is that it may result in a bunch of data due to inconsistency across different models and difficulty in setting the proportion of each data source.

Instead, we focus on consistently annotating errors for texts generated from a single source. In order to make TGEA as general and representative as possible, we use GPT-2 that is not only currently state of the art in text generation but also easily available. We also adopt standard and widely-used hyperparameters (see Appendix for more details) for NEZHA-Gen to generate texts.

Additionally, we use a random sampling strategy with top $k = 30$. For setting k, we have analyzed 500 examples with different values of k, and found that adjusting k has a reasonable impact on the percentage of redundancy errors. Except for the extreme case of $k = 1$, the types of errors and the distribution of them do not change significantly. Take commonsense errors as an example, which is the biggest difference from human-written texts. When k varies in a range of $\{5, 10, 20, 30, 50\}$, the percentage of commonsense errors is $18.6\% \pm 5.8\%$. Redundancy errors account for $>95\%$ when $k = 1$ (while commonsense errors account for 0.8%), but sharply drop to 37.4% as $k = 5$, and the form of repetition changes from same-word repetition to a mixed repetition of "synonymous/same-word", suggesting that a simple repetition penalty may not be sufficient to deal with semantic redundancy. When $k \in \{10, 20, 30, 50\}$, the percentage of redundancy errors is very close to the result reported in Figure 2. When $k > 30$, many generated sentences are completely incomprehensible. A larger k will also reduce the generation efficiency. Therefore, we chose a sampling strategy of $k = 30$, which is the trade-off between text quality and generation efficiency.

7 Conclusions

In this paper, we have presented TGEA, the first dataset with a variety of manual annotations on errors occurring texts generated by pretrained language models. For each erroneous text generated by a Chinese GPT-2 model, our crowdsourced annotators detect erroneous text spans with their associated text spans and provide error types defined in a two-level hierarchical taxonomy as well as rationales behind detected errors. We elaborate the 5 annotation stages for building TGEA with a strict annotation quality control protocol. We also report baseline results of the 5 benchmark tasks on TGEA. The low results suggest that our dataset is a challenging testbed for future work on automatic detection of erroneous spans and types as well as producing error corrections and rationales for texts generated by PLMs. TGEA is featured with wide error type coverage, rich semantic annotation and functional diversity, which can not only be used for deep diagnostic analysis on the text generation capability of pretrained language models, but also facilitate and promote the research of automatic and interpretable error correction for PLM-generated texts.

Acknowledgments

The present research was supported by Huawei. We would like to thank the anonymous reviewers for their insightful comments. We also want to thank MindSpore[7] for the partial suppoort of this work, which is a new deep learning computing framework. The corresponding author is Deyi Xiong (dyxiong@tju.edu.cn).

References

Timothée Bernard and Ting Han. 2020. Mandarinograd: A chinese collection of winograd schemas. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 21–26. European Language Resources Association.

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Scott Wen-tau Yi, and Yejin Choi. 2020. Abductive commonsense reasoning. *International Conference on Learning Representations*.

Yonatan Bisk, Rowan Zellers, Ronan LeBras, Jianfeng Gao, and Yejin Choi. 2020. PIQA: Reasoning about physical commonsense in natural language. In *AAAI*, pages 7432–7439.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry,

[7] 2020. MindSpore. https://www.mindspore.cn/

Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1876–1900. Curran Associates, Inc.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 9539–9549. Curran Associates, Inc.

Yu Cao, Wei Bi, Meng Fang, and Dacheng Tao. 2020. Pretrained language models for dialogue generation with multiple input sources. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 909–917, Online. Association for Computational Linguistics.

Mingda Chen, Zewei Chu, and Kevin Gimpel. 2019. Evaluation benchmarks and learning criteria for discourse-aware sentence representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 649–662, Hong Kong, China. Association for Computational Linguistics.

Leyang Cui, Yu Wu, Shujie Liu, Yue Zhang, and Ming Zhou. 2020a. MuTual: A dataset for multi-turn dialogue reasoning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1406–1416, Online. Association for Computational Linguistics.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020b. Revisiting pre-trained models for Chinese natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 657–668, Online. Association for Computational Linguistics.

Daniel Dahlmeier and Hwee Tou Ng. 2012. Better evaluation for grammatical error correction. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 568–572, Montréal, Canada. Association for Computational Linguistics.

Vidas Daudaravicius, Rafael E. Banchs, Elena Volodina, and Courtney Napoles. 2016. A report on the automatic evaluation of scientific writing shared task. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 53–62, San Diego, CA. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy. Association for Computational Linguistics.

Simon Flachs, Ophélie Lacroix, Helen Yannakoudakis, Marek Rei, and Anders Søgaard. 2020. Grammatical error correction in low error density domains: A new benchmark and analyses. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8467–8478, Online. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos QA: Machine reading comprehension with contextual commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2391–2401, Hong Kong, China. Association for Computational Linguistics.

Naoya Inoue, Pontus Stenetorp, and Kentaro Inui. 2020. R4C: A benchmark for evaluating RC systems to get the right answer for the right reason. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6740–6750, Online. Association for Computational Linguistics.

Dan Iter, Kelvin Guu, Larry Lansing, and Dan Jurafsky. 2020. Pretraining with contrastive sentence objectives improves discourse performance of language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4859–4870, Online. Association for Computational Linguistics.

Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. 2020. SciREX: A challenge dataset for document-level information extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7506–7516, Online. Association for Computational Linguistics.

Zhenyu Jiao, Shuqi Sun, and Ke Sun. 2018. Chinese lexical analysis with deep bi-gru-crf network. *arXiv preprint arXiv:1807.01882*.

Masahiro Kaneko, Masato Mita, Shun Kiyono, Jun Suzuki, and Kentaro Inui. 2020. Encoder-decoder models can benefit from pre-trained masked language models in grammatical error correction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4248–4254, Online. Association for Computational Linguistics.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'12, page 552–561. AAAI Press.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Mary McHugh. 2012. Interrater reliability: The kappa statistic. *Biochemia medica : časopis Hrvatskoga društva medicinskih biokemičara / HDMB*, 22:276–82.

Courtney Napoles, Maria Nadejde, and Joel Tetreault. 2019. Enabling robust grammatical error correction in new domains: Datasets, metrics, and analyses. *Transactions of the Association for Computational Linguistics*, 7(0):551–566.

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2017. JFLEG: A fluency corpus and benchmark for grammatical error correction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 229–234, Valencia, Spain. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A controlled table-to-text generation dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1173–1186, Online. Association for Computational Linguistics.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376, Online. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *AAAI spring symposium: logical formalizations of commonsense reasoning*, pages 90–95.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. WinoGrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8732–8740. Issue: 05.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4463–4473, Hong Kong, China. Association for Computational Linguistics.

Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. 2020. Textcaps: a dataset for image captioning with reading comprehension. *CoRR*, abs/2003.12462.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2020. olmpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*, 8:743–758.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158. Association for Computational Linguistics.

Pavlos Vougiouklis, Hady ElSahar, Lucie-Aimée Kaffee, Christophe Gravier, Frédérique Laforest, Jonathon S. Hare, and Elena Simperl. 2017. Neural wikipedian: Generating textual summaries from knowledge base triples. *CoRR*, abs/1711.00155.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32, pages 3266–3280. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Hongfei Wang, Michiki Kurosawa, Satoru Katsumata, and Mamoru Komachi. 2020. Chinese grammatical correction using BERT-based pre-trained model. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 163–168, Suzhou, China. Association for Computational Linguistics.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Rongxiang Weng, Heng Yu, Shujian Huang, Shanbo Cheng, and Weihua Luo. 2020. Acquiring knowledge from pre-trained model to neural machine translation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9266–9273.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.

Liang Xu, Hai Hu, Xuanwei Zhang, Lu Li, Chenjie Cao, Yudong Li, Yechen Xu, Kai Sun, Dian Yu, Cong Yu, Yin Tian, Qianqian Dong, Weitang Liu, Bo Shi, Yiming Cui, Junyi Li, Jun Zeng, Rongzhao Wang, Weijian Xie, Yanting Li, Yina Patterson, Zuoyu Tian, Yiwen Zhang, He Zhou, Shaoweihua Liu, Zhe Zhao, Qipeng Zhao, Cong Yue, Xinrui Zhang, Zhengliang Yang, Kyle Richardson, and Zhenzhong Lan. 2020. CLUE: A Chinese language understanding evaluation benchmark. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4762–4772, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019a. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6720–6731.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019b. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Hongming Zhang, Xinran Zhao, and Yangqiu Song. 2020a. WinoWhy: A deep diagnosis of essential commonsense knowledge for answering Winograd schema challenge. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5736–5745, Online. Association for Computational Linguistics.

Li Zhang, Qing Lyu, and Chris Callison-Burch. 2020b. Reasoning about goals, steps, and temporal ordering with WikiHow. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4630–4639, Online. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020c. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2021. Retrospective reader for machine reading comprehension. In *The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI-21)*.

Yuanyuan Zhao, Nan Jiang, Weiwei Sun, and Xiaojun Wan. 2018. *Overview of the NLPCC 2018 Shared Task: Grammatical Error Correction: 7th CCF International Conference, NLPCC 2018, Hohhot, China, August 26–30, 2018, Proceedings, Part II*, pages 439–445.

Xuhui Zhou, Yue Zhang, Leyang Cui, and Dandan Huang. 2020. Evaluating commonsense in pre-trained language models. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9733–9740. AAAI Press.

A Appendix

A.1 NEZHA-Gen Hyperparameters

Table 1 show the configuration of the generative model (NEZHA-Gen).

Model	NEZHA-Gen
hidden_size	768
num_hidden_layers	12
num_attention_heads	12
intermediate_size	3072
hidden_act	gelu
hidden_dropout_prob	0.1
attention_probs_dropout_prob	0.1
max_position_embeddings	512
type_vocab_size	16
initializer_range	0.02

Table 1: Configuration of NEZHA-Gen.

A.2 Training Setting

Table 2, 3, 4, 5, 6 show the training settings of the baseline models for each task. In these tables, $\mathrm{ALBERT_{zh}}$, $\mathrm{BERT_{zh}}$, $\mathrm{RoBERTa_{zh}}$ represent ALBERT-chinese, RoBerta-wwm-ext and RoBerta-wwm-ext respectively.

Model	$\mathrm{ALBERT_{zh}}$	$\mathrm{BERT_{zh}}$	$\mathrm{RoBERTa_{zh}}$
Model size	large	base	large
Learning rate		2×10^{-5}	
Batch size		8	
Optimizer		Adam	
Adam β_1		0.9	
Adam β_2		0.98	
Adam ϵ		1×10^{-8}	
Max epochs		50	
Loss function		cross-entropy	
Dropout		0.1	

Table 2: Training details for the Erroneous Text Detection task.

Model	$\mathrm{ALBERT_{zh}}$	$\mathrm{BERT_{zh}}$	$\mathrm{RoBERTa_{zh}}$
Model size	base	base	base
Learning rate		2×10^{-5}	
Batch size		32	
Optimizer		Adam	
Adam β_1		0.9	
Adam β_2		0.999	
Adam ϵ		1×10^{-6}	
Max epochs		5	
Loss function		cross-entropy	
Dropout		0.1	

Table 3: Training details for the Erroneous and Associated Span Detection task.

Model	$\mathrm{ALBERT_{zh}}$	$\mathrm{BERT_{zh}}$	$\mathrm{RoBERTa_{zh}}$
Model size	large	base	large
Learning rate		2×10^{-5}	
Batch size		8	
Optimizer		Adam	
Adam β_1		0.9	
Adam β_2		0.98	
Adam ϵ		1×10^{-8}	
Max epochs		50	
Loss function		cross-entropy	
Dropout		0.1	

Table 4: Training details for the Error Type Classification task.

	$\mathrm{BERT_{zh}}$-GEC	$\mathrm{RoBERTa_{zh}}$-GEC
Model	BERT-wwm-ext	RoBERTa-wwm-ext-large
Architecture	Transformer (big)	
Learning rate	3×10^{-5}	
Batch size	16	
Optimizer	Adam	
Adam β_1	0.9	
Adam β_2	0.98	
Adam ϵ	1×10^{-8}	
Max epochs	50	
Loss function	label smoothed cross-entropy ($\epsilon_{ls} = 0.1$)	
Dropout	0.3	

Table 5: Training details for the Error Correction task.

Model	NEZHA-Gen
Learning rate	5×10^{-5}
Batch size	4
Optimizer	Adam
Adam β_1	0.9
Adam β_2	0.999
Adam ϵ	1×10^{-6}
Max epochs	3
Dropout	0.1

Table 6: Training details for the Rationale Generation task.

A.3 Examples of Level-2 Error Types

Table 7 shows examples of level-2 error types in TGEA.

Level-1 Error Type	Level-2 Error Type	Example
Inappropriate Combination	Subject-Predicate	目前,该市的小说 [话剧]《我是党员、我的团员》、《我是小老头》、《小小老师》、《小小一个农家娃》正在上演。 At present, the city's novels [drama] *I am a Party member and This is My League Member*, *Little Old Man Like Me*, *Little Teacher*, *A Little Farm Boy* are on stage.
	Predicate-Object	由我主持，我要带大家去感受一下大赛主题设置的感受 [氛围]。 As a host, I will take you to experience the feel [atmosphere] shown from the theme of the competition.
	Subject-Object	女足的队员 [任务] 就是一个球，能够把球踢好，就是她们最大的资本。 The players [task] of women's football team is a ball, and playing the ball well is their biggest capitals.
	Modifier	另一方面，煤炭企业面临着煤矿安全的矛盾 [问题]。 On the other hand, coal enterprises are facing the contradiction [problem] of coal mine safety.
	Function Word	因此，我对 [因为]自身的过错作出了自己应当承担的责任。 Therefore, to [because of] my own fault, I took my own responsibility.
Misssing	Subject	当他回到车间时，_[车间]已经有了明显的变化。 When he returned to the workshop, _ [the place] had been a marked change
	Predicate	这时候我们一开始就有机会扳平比分，但是我们没有_[抓住]机会。 We had a chance to equalise at the beginning, but we didn't _ [caught] chance.
	Object	一、坚持解放思想,转变观念,推进社会主义物质文明和精神_[文明]。 1. Persisting in emancipating the mind, changing ideas and promoting socialist material civilization and spiritual_ [civilization].
	Modifier	在国内成立水牛研究中心，有利于增强_[水牛对]自然条件和人工环境的适应能力。 The establishment of Buffalo Research Center in China is conducive to enhance the adaptability _ [of buffalo] to natural conditions and artificial environment.
	Function Word	他的儿子_[在]上一届奥运会夺得冠军，并且获得当年世界锦标杯赛金牌。 His son won champion_ [in] the last Olympic Games and won the gold medal in the World Championship Cup that year.
Redundancy	Subject	但一些外资银行，尤其是外资银行[]，对我国民营经济的发展还有不少误解或偏见。 However, some foreign banks, especially foreign banks[], still have many misunderstandings or prejudices about the development of China's private economy.
	Predicate	这也是所有关心[]关心孩子成长的人的共同心声。 This is also the common voice of all those who care about[] care about children's growth
	Object	同时，学校也开展丰富多彩、有益于学生的社会实践活动、社会实践[]，丰富他们的课余生活。 At the same time, the school also carries out colorful and beneficial social practice activities, social practice[] to enrich their after-school life.
	Modifier	它们的皮毛很有光泽,可以用肉眼很难[]看出来。 Their fur is so shiny that we can see with naked eyes hardly[].
	Function Word	他是被迫进入位于市中心的一个警察局的，随后[]他被带到警察局，并遭到了手铐和警犬的威吓。 He was forced into a police station in the center of the city, then[] he was taken to the police station, where he was intimidated by handcuffs and police dogs.
Discourse Error	Coreference	在婚姻变得更为不好的时候，对她来说这是痛苦的。但是当她[它]发生变化时，她必须做出调整。 It was painful for her when the marriage got worse. But when she [it] changed, she had to adjust.
Commonsense Error	Space	他说,中美两国是近邻 [朋友],关系很好,中美合作富有创造性。 He said that China and the United States are close neighbors [friends] with good relations and creative cooperation.
	Time	国庆 [元旦]假期期间，各大汽车经销商将会以怎么样的姿态迎接新的一年? During the National Day [New Year's Day] holiday, how will major auto dealers greet the new year?
	Number	而在4月份，中国石化、招商银行、万科、上海汽车、g长安和g天威成为了最活跃的5 [6]只股票。 In April, Sinopec, China Merchants Bank, Vanke, SAIC, G Changan and G Tianwei became the most active 5 [6] stocks.
	Motivation	近日，李老的胃疼难忍，为治疗病情已连续工作 [休息]两天了，而且病情非常严重，他一躺就是几天。 Recently, Lao Li's stomach ache is unbearable. He has been working [resting] for two consecutive days to treat his illness, and his illness is very serious. He has been lying down for several days.
	Emotional Reactions	对于学校为了保障广大师生员工的安全，采取这些措施，我们深感遗憾[欣慰]。 We are very sorry [pleased] that the school has taken these measures to ensure the safety of students, teachers, and other staff.
	Causation	据悉，由于身价低廉[高昂]，子淇在国内是很少有人请得到的大牌艺人之一。 It is reported that Ziqi is one of the few famous artists that are difficult to invite in China because of his low [high] value.
	Taxonomy	酱 [花生] 油是植物油中的一种，食用后可以对皮肤有非常好的润泽效果。 Soy sauce [Peanut Oil] is a kind of vegetable oil, which has a very good moisturizing effect on the skin after eating.
	Behaviors	一位中国官员表示：我们将在近期和俄罗斯、中国 [法国] 等国合作进一步推广这一系列行动，以此来缓解人们对恐怖主义威胁的忧虑。 In the near future, we will work with Russia, China [France] and other countries to further promote this series of actions to ease people's concerns about the threat of terrorism, a Chinese official said.

Table 7: Examples of level-2 error types in TGEA. Underwaved words are erroneous words while underlined words are associated words. Words in "[]" are corrections to erroneous words.