Building Sequence-to-Sequence Document Revision Models from Matched and Multiple Partially-Matched Datasets

Anonymous ACL submission

Abstract

This paper defines the document revision task and proposes a novel modeling method that can utilize not only a matched dataset but also multiple partially-matched datasets. In the document revision task, we aim to simultaneously consider multiple perspectives for writing supports. To this end, it is important not only to correct grammatical errors but also to improve readability and perspicuity, through means such as conjunction insertion and sentence reordering. However, it is difficult to prepare enough the matched dataset for the document revision task since this task has to consider multiple perspectives simultaneously. To mitigate this problem, our idea is to utilize not only a limited matched dataset but also various partially-matched datasets that handles individual perspectives, e.g., correcting grammatical errors or inserting conjunctions. Since suitable partially-matched datasets have either been published or can easily be made, we expect to prepare a large amount of these partially-matched datasets. To effectively utilize these multiple datasets, our proposed modeling method incorporates "on-off" switches into sequence-to-sequence modeling to distinguish the matched datasets and individual partially-matched datasets. Experiments using our created document revision datasets demonstrate the effectiveness of the proposed method.

1 Introduction

With the advance of natural language processing technology using deep learning, applications for writing support systems have been developed (Tsai et al., 2020; Ito et al., 2020). Such writing support systems often implement a grammatical error correction task that correct errors such as typos and mistakes in inflected verbs forms (Rothe et al., 2021). To advance writing support, it is important not only to correct grammatical errors but also to improve readability and perspicuity. For example, when we manually perform document revision, we attempt not only to correct grammatical errors but also to split a long sentence into sentences to improve the readability and the perspicuity. In addition, we also consider the relationships between sentences, such as reordering to obtain a consistent order and conjunction insertion. Accordingly, this paper defines a document revision task that simultaneously considers these multiple perspectives for writing support.

In natural language processing area, the document revision task has been studied by breaking it down into partial tasks. The most common partial task is grammatical error correction, and various methods have been proposed to model this task (Sawai et al., 2013; Mizumoto and Matsumoto, 2016; Junczys-Dowmunt and Grundkiewicz, 2016). In recent studies, the sequence-to-sequence (seq2seq) modeling methods has achieved high performance with the advance of deep learning (Yuan and Briscoe, 2016; Junczys-Dowmunt et al., 2018; Rothe et al., 2021). In addition, other famous partial tasks are the sentence ordering (Yin et al., 2019) or discourse relation classification (Liu et al., 2016; Dai and Huang, 2018). Most of these tasks have also been studied with the seq2seq modeling (Wang and Wan, 2019). On the other hand, there are few studies that address multiple perspectives in the document revision task. Lin et al. (2021) addressed the sentence ordering and sentence paraphrasing tasks, and Ihori et al. (2020) addressed multiple perspectives for spoken-to-written style conversion such as style unification, disfluency deletion, punctuation restoration at the same time. However, to the best our knowledge, the document revision task that comprehensively handle multiple perspectives has not well examined. Therefore, we aim to model such document revision task using the promising seq2seq modeling.

There are two difficulties in building the document revision seq2seq models.

- The first difficulty is that the document revi-

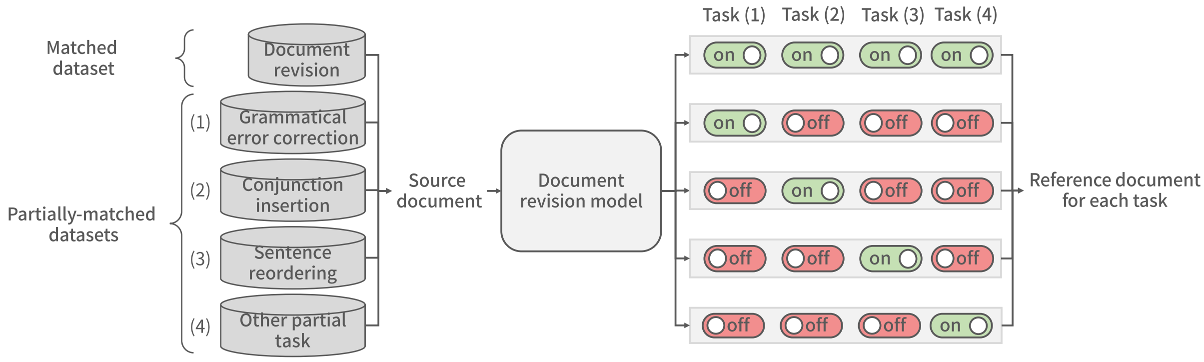

Figure 1: Document revision model using both a dataset for main task and datasets for partial tasks.

sion model has to handle multiple perspectives simultaneously. Although seq2seq models can address any problems that convert a source sequence into a target sequence, handling multiple perspectives is considered as a difficult task.

- The second difficulty is that improving the readability and perspicuity requires precisely handling long-range contexts of multiple sentences. While the conventional grammatical error correction tasks take contexts within a sentence into consideration, our document revision task must handle a set of sentences, i.e., document-level information.

These two difficulties induce us to prepare a lot of training datasets so as to robustly model the document revision task; however, it is difficult to prepare enough matched training data because these two difficulties also affect the data creation cost.

Our key idea to mitigate this problem is to utilize not only a limited matched dataset but also various partially-matched datasets that handle individual perspectives for building the document revision models. The partially-matched datasets can be regarded as datasets for the partial tasks. There are several existing datasets for grammatical error correction (Dahlmeier et al., 2013; Tajiri et al., 2012) and sentence ordering (Chen et al., 2016; Huang et al., 2016). In addition, datasets can be generated heuristically for noisy sentence deletion and conjunction insertion tasks. For example, for the conjunction insertion task, we can construct paired data by deleting and restoring conjunctions from existing documents. We expect that these partially-matched datasets will be effective for improving our document revision task. The important issue is how we exactly utilize both a limited matched dataset and various partially-matched datasets for building a document revision model.

In this paper, we propose a novel modeling method that simultaneously utilize both a matched dataset and multiple partially-matched datasets. In the proposed method, we incorporate multiple "on-off" switches into seq2seq modeling so as to distinguish the matched datasets and individual partially-matched datasets. Figure 1 shows an example of how the proposed method uses multiple switches. It is implemented by using switching tokens, which were previously proposed by (Ihori et al., 2021b). The switching tokens have the role of switching the "on" or "off" state for each task. By introducing the switching tokens into the seq2seq modeling, the main document revision task and each partial task can be explicitly distinguished within one modeling. We expect that our proposed modeling method effectively improves the main document revision task by appropriately leveraging knowledge from partially-matched datasets. Furthermore, our proposed method can be combined with self-supervised pre-training, which is the most successful approach in recent modeling methods (Kenton and Toutanova, 2019). In this approach, unpaired text datasets are used for building a base model in a pre-training phase and the model is fine-tuned by paired datasets. In natural language generation tasks using seq2seq models, several successful self-supervised pre-training methods had been proposed (Song et al., 2019; Ihori et al., 2021a). We expect that our proposed method can be effectively applied after performing the self-supervised pre-training.

For evaluation, we newly construct a Japanese document revision dataset (see Sec. 3). In our experiments, we used the new dataset as the matched dataset, and grammatical error correction and conjunction insertion datasets as the partially-matched datasets. Our experimental results demonstrate that our proposed modeling method effectively improves the document revision performance by using not only the matched dataset but also the partially-matched datasets.

Our main contributions are as follows:

- We define a document revision task that simultaneously considers multiple perspectives for writing support, and specify the relationship between our document revision task and conventional related tasks.
- We create a novel dataset for a Japanese document revision task and detail how we create it.
- We present a novel modeling method that can utilize not only a matched dataset but also multiple partially matched datasets, and show the effectiveness of the proposed method in our experiments.

2 Related Work

The partial tasks that compose a document revision task have been studied as individual tasks. The most typical task is the grammatical error correction task, which corrects the errors in an input text by deleting, inserting, and replacing words. Many studies on this task focused on sentence-level errors, and they performed error correction by using a seq2seq model to achieve high performance (Yuan and Briscoe, 2016; Junczys-Dowmunt et al., 2018; Rothe et al., 2021). In addition, recent studies have introduced the seq2seq pre-training to utilize a large amount of unpaired data to improve the performance with a limited amount of paired data (Lewis et al., 2020; Song et al., 2019; Ihori et al., 2021a). Thus, in this work, we investigated the combination of such pre-training methods and our proposal. For the grammatical error correction task, synthetic training data generation is also introduced as another way to deal with paired-data scarcity (Grundkiewicz et al., 2019; Kiyono et al., 2020; Rothe et al., 2021). For the document revision task, however, it is difficult to generate synthetic data because the task involves multiple partial tasks such as grammatical error correction, sentence reordering, and conjunction insertion.

In addition, certain tasks handle multiple sentences, such as a discourse relation classification task (Liu et al., 2016; Dai and Huang, 2018) and a sentence reordering task (Wang and Wan, 2019). In the discourse relation classification task, the model predicts the relation class (e.g., contrast and causality) of two arguments. In this work, we adopted a conjunction insertion task that is similar to the discourse relation classification task but directly completes conjunctions according to the relationship between sentences. Sentence ordering is another task that considers the document-level coherence where an input set of sentences are re-arranged into a logically consistent order. For this task, seq2seq models like pointer-network were mainly used (Cui et al., 2018). In a recent work, graph network is also introduced and achieved high performance (Yin et al., 2019). Although these studies improved readability in terms of sentence order, they did not cover other aspects of the document revision.

There are few studies to cover multiple aspects of document revision at the same time. Lin et al. (2021) proposed document-level paraphrase generation task that simultaneously performs the sentence reordering and sentence rewriting tasks. In this study, a pseudo dataset for document-level paraphrase generation task was created and the task was performed with a specific model architecture. To perform multiple tasks, the task-specific model architecture and matched dataset were needed. Thus, it is difficult to add a new task for document-level paraphrase generation task.

3 Dataset for Document Revision Task

3.1 Dataset construction

In this paper, we present a new dataset for Japanese document revision task. The dataset contains paired data consisting of source and reference documents in Japanese. The source documents were written by Japanese crowd workers. Also, the reference documents were revised by Japanese two labelers. Each document contained multiple Japanese sentences to enable the consideration of contextual information. Below, we explain the details of creating source and reference documents.

Source documents: To make the source documents, we employed crowd workers and they wrote essays consisted of a single paragraph document in Japanese. The documents have an essay-style structure, because Japanese schools teach how to write essays; thus, we expected that many of the workers could write the essays at the same level. Specifically, we employed 161 workers whose na-

(1)	Correct the following mistakes. typos, punctuation, kanji, syntax and grammatical errors, spoken-style text, and redundant expressions
(2)	Split long sentences containing more than 60 characters.
(3)	Unify words with different expressions that have the same meaning.
(4)	If there is no subject, restore the subject by using words that have already been mentioned.
(5)	Change the sentence order if it is not appropriate.
(6)	Delete sentences that describe unrelated topics.
(7)	Insert correct conjunctions for the relationships between sentences.

Table 1: Guidelines for document revision.

		# of documents	# of sentences
Training	Input	5,000	26,477
	Output	5,000	28,158
Validation	Input	554	2,922
	Output	554	3,128
Test	Input	1,121	6,054
	Output	2,242	12, 831

Table 2: Details of the dataset for document revision.

tive language was Japanese. First, we showed the workers 48 possible themes, and they individually selected 1-15 themes. The 48 themes were chosen by the crowdsourcing company from actual themes that were used for exam essays in Japan. Next, the workers wrote single paragraph documents, each of which contained 200-300 characters and four or more sentences. These multiple sentences are needed to conduct the revision by considering the relationship between sentences. Each worker wrote 1-15 documents per person, and took up to 15 minutes to write each document. Although the workers were asked to be careful about typos, they were not asked to compose the essay perfectly.

Reference documents: To revise the source documents, we employed two labelers whose native language was Japanese. One labeler was licensed as a Japanese language teacher, while the other labeler received guidance of revision for a document. In the document revision task, we should handle multiple perspectives to improve the readability of a document. Thus, we asked them to follow the revision guidelines listed in Table 1, to ensure that the labelers can consider revising from the multiple perspectives. Table 1 shows the guidelines for document revision. In the table, (1) shows the error correction task and (2-7) shows the other tasks for improving the readability and the perspicuity. Since it is difficult to clearly define the readability and the perspicuity, we told labelers specific examples of each task. For example, for (2), it is possible to divide the sentences according to the number of characters, and for (7), we represented the list of conjunctions that shows their kinds and roles, and asked them to select from this list. We expected that the labelers would be able to revise documents with equivalent quality by following the guidelines. Note that they do not necessarily have to consider all the perspectives simultaneously, but only made these revisions if there were any mistakes or unnatural points.

3.2 Details

Table 2 lists that the details of the resulting dataset for document revision task. The dataset is divided into a training set, validation set and test set. The training and validation sets have one reference document, while the test set has two reference documents for each source document. Figure 2 shows an example from the dataset. As this example demonstrates, the dataset was created while considering multiple perspectives simultaneously. For example, typo correction, too-long sentence splitting, and conjunctions insertion tasks are performed at the same as shown in Table 1. To the best of our knowledge, this is the first dataset to address such multiple perspectives of the document revision task.

4 Document Revision Models

4.1 Strategy

To build document revision model, we utilize a matched dataset for document revision task (created in chapter 3) and multiple partially-matched datasets. In this paper, the document revision task is referred to as the main task and tasks that handle each perspective in the main task are referred to as the partial tasks. The partially-matched datasets can be regarded as datasets for the partial tasks.

Our strategy is to incorporate multiple “on-off” switches into seq2seq modeling to distinguish the matched datasets and individual partially-matched datasets. It is implemented by using switching tokens (Ihori et al., 2021b). A switching token represents the “on” state (the target task) or “off” state (not the target task) for each perspective. By introducing the switching tokens into the seq2seq modeling, the main document revision task and each partial task can be explicitly distinguished within one modeling.

Source	ソーシャルメディアの発達により、私たちは好きな情報を簡単に得られるようになった。その一方で、日々膨大な情報にさらされることの弊害も出てきている。ソーシャルメディアでは、自身の趣味嗜好にあ沿った情報を得ようとすることが多い。これまでは新聞やテレビが情報源だったので全員が同じ情報に触れて対等に対話していたが、現在は無意識に自分に近しい意見を全体の情報として捉えてしまう人もおり、情報が偏る。ソーシャルメディアは一見情報の宝庫に見えるが、視点を変えれば物事を都合よくとらえる為のツールなのかもしれない。
Reference	ソーシャルメディアの発達で、私たちは好きな情報を簡単に得られるようになった。その一方で、日々膨大な情報にさらされる弊害も出てきている。また、ソーシャルメディアでは、自身の趣味嗜好に沿った情報を得ようとすることが多い。これまでは新聞やテレビが情報源だったので、全員が同じ情報に触れて対等に対話していた。しかし、現在は、無意識に自分に近しい意見を全体の情報として捉える人もおり、情報が偏る。そのため、ソーシャルメディアは一見情報の宝庫に見えるが、視点を変えれば、物事を都合よく捉える為のツールなのかもしれない。
Translation	The development of social media has made it easier to get information. On the other hand, there can be difficulties in handling vast amounts of information. Also, in most cases, we only use social media to access our favorite types and sources of information. Previously, many people got the same information from newspapers and television, and thus, they could talk on an equal footing. Now, however, some people unknowingly treat their closely held opinions as complete information, so their information is biased. Therefore, social media seems to be a treasure trove of information, but it may also be a tool for maintaining biased information.

Figure 2: Example from the document revision task dataset.

Figure 3 shows an example of our strategy using switching tokens. In this example, we use grammatical error correction (GEC) dataset, conjunction insertion (CI) dataset, and the main task dataset to build the document revision model. In this case, we use six switching tokens [gec_on], [ci_on], [other_on], [gec_off], [ci_off], and [other_off]. Here, we specify the "other" token because the main task handle other perspectives that are not considered in the grammatical error correction and conjunction insertion tasks, as listed in Table 1. The seq2seq models are split into an encoder network and a decoder network. These switching tokens are utilized for inputs of the decoder network as given contexts. In a training phase, we use all datasets for building a seq2seq model while distinguishing each task using above switching tokens. In an inference phase, we expect to perform the main task by feeding [gec_on], [ci_on], and [other_on]. Note that we can also perform the grammatical error correction or conjunction insertion by feeding appropriate switching tokens.

4.2 Proposed modeling method

In this paper, we propose a novel modeling method that simultaneously utilize both a matched dataset and multiple partially-matched datasets. In the proposed method, we incorporate multiple "on-off" switches into seq2seq modeling so as to distinguish the matched datasets and individual partially-matched datasets.

Modeling: We define the source document as $\boldsymbol{X} = \{x_1, \cdots, x_m, \cdots, x_M\}$ and the reference document as $\boldsymbol{Y} = \{y_1, \cdots, y_n, \cdots, y_N\}$, where M and N are the numbers of tokens in source and reference documents, respectively. x_m and y_n are tokens which include not only characters or words but also punctuation marks. Note that X and Y involves multiples sentences.

Our proposed document revision model predicts the generation probabilities of a reference document $\boldsymbol{Y}$ given a source document $\boldsymbol{X}$ and switching tokens $s_{1:T} = \{s_1, \cdots, s_t, \cdots, s_T\}$, where T is the number of "on-off" switches. The generation probability of $\boldsymbol{Y}$ is defined as

$$P(\boldsymbol{Y}|\boldsymbol{X}, s_{1:T}; \boldsymbol{\Theta}) \tag{1}$$

$$= \prod_{n=1}^{N} P(y_n|y_{1:n-1}, \boldsymbol{X}, s_{1:T}; \boldsymbol{\Theta}),$$

where $\boldsymbol{\Theta}$ represents the trainable parameters. s_t is the t-th switching token represented as

$$s_t \in \{[t\text{−th } \texttt{task\_on}], [t\text{−th } \texttt{task\_off}]\}. \tag{2}$$

In this paper, we use Transformer pointer-generator networks (Deaton, 2019) for this modeling. Transformer pointer-generator networks are effective for monolingual translation tasks because they contain a copy mechanism that copies tokens from a source text to help generate infrequent tokens. Note that our method does not change the architecture of a transformer pointer-generator network, but merely adds switching tokens to the model input.

Pre-training: In this paper, we use a MAsked Pointer-Generator Network (MAPGN) (Ihori et al., 2021a) because it is a suitable pre-training method for pointer-generator networks. In MAPGN, the pointer-generator network is pre-trained by predicting a sentence fragment $y_{a:b}$ giving a masked sequence $\boldsymbol{Y}_{/a:b}$ and. Here, $\boldsymbol{Y}_{/a:b}$ denotes a fragment in which positions a to b are masked, and

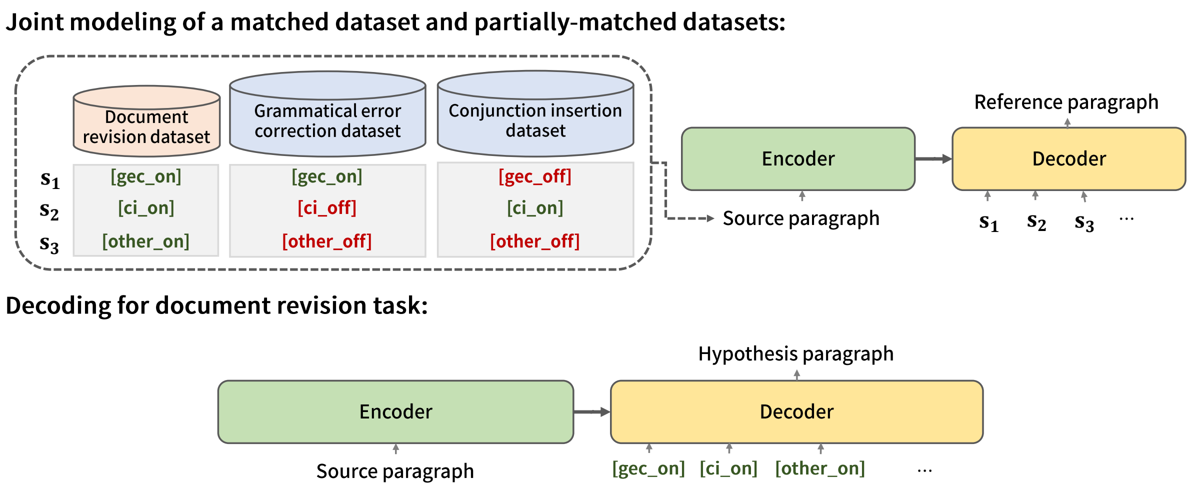

Figure 3: Example of joint modeling based on switching-token.

$y_{a:b}$ denotes a sentence fragment of $\boldsymbol{Y}$ from a to b. The model parameter set can be optimized from unpaired dataset $\mathcal{D}^{\text{u}}$. The training loss function $\mathcal{L}$ is defined as

$$\mathcal{L} = -\sum_{(\boldsymbol{Y})\in\mathcal{D}^{\text{u}}} \log P(y_{a:b}|y_{a-1}, \boldsymbol{Y}_{/a:b}; \boldsymbol{\Theta}), \tag{3}$$

$$= -\sum_{(\boldsymbol{Y})\in\mathcal{D}^{\text{u}}} \sum_{t=a}^{b} \log P(y_t|y_{a-1:t-1}, \boldsymbol{Y}_{/a:b}; \boldsymbol{\Theta}).$$

Note that all switching tokens have to be included in the vocabulary in the pre-training.

Fine-tuning: In our proposed method, the matched dataset $\mathcal{D}^{\text{m}}$, and multiple partially-matched datasets $\{\mathcal{D}_1^{\text{pm}}, \cdots, \mathcal{D}_t^{\text{pm}}, \cdots, \mathcal{D}_T^{\text{pm}}\}$ are trained jointly in a single model. The training loss function $\mathcal{L}$ is defined as

$$\mathcal{L} = \mathcal{L}^{\text{m}} + \sum_{t=1}^{T} \mathcal{L}_t^{\text{pm}}, \tag{4}$$

where $\mathcal{L}^{\text{m}}$ is the loss function against the main task and it is computed from

$$\mathcal{L}^{\text{m}} = -\sum_{(\boldsymbol{X},\boldsymbol{Y})\in\mathcal{D}^{\text{m}}} \log P(\boldsymbol{Y}|\boldsymbol{X}, \hat{s}_{1:T}; \boldsymbol{\Theta}), \tag{5}$$

where $\hat{s}_{1:T} = \{\hat{s}_1, \cdots, \hat{s}_T\}$ are switching tokens and $\hat{s}_t$ is represented as

$$s_t = [t\text{−th } \texttt{task\_on}]. \tag{6}$$

$\mathcal{L}_t^{\text{pm}}$ is the loss function against the t-th partial task and it is computed from

$$\mathcal{L}_t^{\text{pm}} = -\sum_{(\boldsymbol{X},\boldsymbol{Y})\in\mathcal{D}_t^{\text{pm}}} \log P(\boldsymbol{Y}|\boldsymbol{X}, \bar{s}_{1:T}; \boldsymbol{\Theta}), \tag{7}$$

where $\bar{s}_{1:T} = \{\bar{s}_1, \cdots, \bar{s}_T\}$ are switching tokens and $\bar{s}_{t'}$ is represented as

$$\bar{s}_{t'} = \begin{cases} [t'\text{−th } \texttt{task\_on}] & \text{if } t' = t, \\ [t'\text{−th } \texttt{task\_off}] & \text{otherwise.} \end{cases} \tag{8}$$

Decoding: The decoding problem using switching tokens is defined as

$$\hat{\boldsymbol{Y}} = \arg\max_{\boldsymbol{Y}} P(\boldsymbol{Y}|\boldsymbol{X}, s_{1:T}; \boldsymbol{\Theta}). \tag{9}$$

The model can perform the document revision task or each partial task according to the given switching tokens.

5 Experiments

We experimentally evaluated the effectiveness of the proposed modeling method that can utilize both matched and multiple partially-matched datasets.

5.1 Dataset

For preparing the partially-matched datasets, we adopted the grammatical error correction task (gec) and the conjunction insertion task (ci) as partial tasks. Accordingly, we used three datasets: document revision dataset described in section 3, a Japanese grammatical error correction dataset (Tanaka et al., 2020), and a conjunction insertion dataset. The Japanese grammatical error correction dataset was obtained from revision history on Wikipedia. It contained four categories of Japanese typos: erroneous substitution, deletion, insertion, and kanji-conversion. The conjunction insertion dataset was constructed based on Japanese Wiki-40B dataset (Guo et al., 2020), which is a high

		# of documents	# of sentences
Training	a).	5,000	26,477
	b).	-	506,786
	c).	90,000	533,422
Validation	a).	554	2,922
	b).	-	8,542
	c).	10,000	59,396
Test	a).	1,121	6,054
	b).	-	8,542
	c).	1,000	6,026
Switchs	a).	[gec_on][ci_on][other_on]	
	b).	[gec_on][ci_off][other_off]	
	c).	[gec_off][ci_on][other_off]	

a. Document revision dataset
b. Japanese grammatical error correction dataset
c. Conjunction insertion dataset

Table 3: Details of document revision task datasets

quality processed Wikipedia dataset. To construct this dataset, first, we divided the Wiki-40B dataset into single paragraph documents and selected the documents that contained conjunctions. Next, we deleted the conjunctions from each document, and we used the resulting and original documents as paired data.

For unpaired data which is used for self-supervised pre-training, we prepared 880k single paragraph documents from Wiki-40B dataset that were not used in the conjunction insertion dataset. The details of these datasets are listed in Table 3, where "Switch" refers to switching tokens. We use six switching tokens [gec_on], [ci_on], [other_on], [gec_off], [ci_off], and [other_off] for training and decoding. In decoding, we can also perform the grammatical error correction or conjunction insertion tasks by feeding appropriate switching tokens. Thus, we use test set for each partial task to evaluate each partial task performance. For example, when the model performs the grammatical error correction task in the decoding, the switching tokens [gec_on], [ci_off], and [other_off] are given in the decoder. Moreover, we compare each partial task performance using the joint modeling with a individual model performance. Note that the the number of documents corresponds to the number of sentences in the Japanese grammatical error correction dataset because the dataset is consisted not of documents but of single sentence.

5.2 Setup

For evaluation purposes, we constructed 11 Transformer-based pointer-generator networks. (1) a document revise model, (2) (1) with pre-training, and (3) a grammatical error correction model, (4) a conjunction insertion model, (5) a modeling of the document revision and grammatical error correction datasets, (6) (5) with switching tokens, (7) a modeling of the document revision and conjunction insertion datasets, (8) (7) with switching tokens, (9) a modeling of all three datasets, (10) (9) with switching tokens, (11) (10) with pre-training. (1), (3), and (4) are trained using only each task dataset. We use the unpaired data for pre-training in these models. Note that the Transformer-based pointer-generator network architecture is the same in all of these models.

As for the model details, we used the following configurations. The encoder had a 4-layer transformer encoder block with 512 units, while the decoder had a 2-layer transformer decoder block with 512 units. The output unit size (corresponding to the number of tokens in the pre-training data) was set to 12,773. To train the Transformer pointer-generator networks, we used the RAdam optimizer (Liu et al., 2019) and label smoothing (Lukasik et al., 2020) with a smoothing parameter of 0.1. We set the mini-batch size to 32 documents and the dropout rate in each Transformer block to 0.1. All trainable parameters were initialized randomly, and we used characters as tokens. The pre-training and fine-tuning were the same setups. For decoding, we used the beam search algorithm with a beam size of 4.

For evaluation, we calculated automatic evaluation scores in terms of two metrics: GLEU (Napoles et al., 2015), and $F_{0.5}$. Specifically, we calculated these metrics for characters and used 4-grams for GLEU. $F_{0.5}$ score is calculated using the characters in the generated documents. In addition, we also calculated the F1 score for conjunction insertion, denoted as C-F1, to evaluate the performance of conjunction insertion task. Note that multiple conjunctions can have the same meaning (e.g., "but", and "however"). We thus evaluated whether the system could insert conjunctions with the correct meaning.

5.3 Results

Table 4 shows the results of 11 Transformer pointer-generator networks. In the table, the models of (1)-(11) are described in section 5.2, and (6), (8), (10), and (11) are our proposals. The columns "Switch" and "Pre-train" indicate whether the proposed switching tokens and the pre-training are introduced or not, respectively. The row "Source"

	Dataset	Switch	Pre-train	Document revision			GEC		CI		
				GLEU	$F_{0.5}$	C-F1	GLEU	$F_{0.5}$	GLEU	$F_{0.5}$	C-F1
Source	-	-	-	0.886	0	0	-	-	-	-	-
(1)	a	w/o	w/o	0.857	0.198	0.193	-	-	-	-	-
(2)		w/o	w/	0.884	0.321	0.211	-	-	-	-	-
(3)	b	w/o	w/o	-	-	-	**0.943**	**0.635**	-	-	-
(4)	c	w/o	w/o	-	-	-	-	-	0.964	0.198	0.230
(5)	a + b	w/o	w/o	0.863	0.189	0.164	-	-	-	-	-
(6)		w/	w/o	0.887	0.278	0.163	-	-	-	-	-
(7)	a + c	w/o	w/o	0.881	0.155	0.101	-	-	-	-	-
(8)		w/	w/o	0.888	0.234	0.214	-	-	-	-	-
(9)	a + b + c	w/o	w/o	0.883	0.236	0.205	0.932	0.613	0.966	0.207	0.222
(10)		w/	w/o	0.889	0.282	0.270	**0.943**	0.630	**0.967**	**0.239**	**0.263**
(11)		w/	w/	**0.892**	**0.333**	**0.274**	-	-	-	-	-

a. Document revision dataset b. Japanese grammatical error correction dataset c. Conjunction insertion dataset

Table 4: Results of document revision, grammatical error correction (GEC), and conjunction insertion (CI) tasks.

indicate the results for source documents in the document revision task dataset.

First, we describe the results of the document revision task. The scores of the task with the switching tokens were higher than those without the switching tokens as shown in lines (5) v.s. (6), (7) v.s. (8), and (9) v.s. (10) in the table. In addition, the scores with switching tokens of lines (6), (8), and (10) were higher than the score of the system trained only with the main task data (1). Among the system (6), (8) and (10), the system trained with all three datasets (10) performed the best. These results indicate that the switching tokens are effective for the joint modeling, and the more partial tasks we use, the better the performance of the main task is. In addition, when we compare the results of lines (10) with (11), the results with pre-training outperformed those without pre-training. This indicate that our proposed method can be effectively applied after performing the self-supervised pre-training.

Next, we focus on the results of the performance of each partial task. The switching-token-based joint modeling can perform each partial task by feeding appropriate switching tokens. Thus, we compare the results of a model using each task dataset individually with using a matched dataset and each task dataset simultaneously. In grammatical error correction, the performance of individual modeling and switching-token-based joint modeling were not significantly different. On the other hand, the performance of joint modeling without switching tokens under-performed that of the individual modeling. For conjunction insertion task, the results of joint modeling outperformed that of individual modeling. Also, the results of joint modeling with switching tokens outperformed those without switching tokens. Therefore, these results indicated that switching-token-based joint modeling can improve the performance of the main task without impairing the performance of each task.

6 Conclusion

In this paper, we examined the document revision task with a novel modeling method that can that can utilize not only a matched dataset but also multiple partially-matched datasets. In our document revision task, we revise document descriptions by considering not only to correct grammatical errors but also to improve readability and perspicuity. In our proposed modeling method, we incorporate multiple "on-off" switches into seq2seq modeling so as to distinguish the matched datasets and individual partially matched datasets. The key strength is to effectively improve main document revision task by appropriately leveraging knowledge from partially-matched dataset. The experimental results using our created Japanese document revision dataset demonstrated that our proposed modeling method can improves the document revision performance by utilizing datasets for the grammatical error correction task and the conjunction insertion task. In addition, our proposed method can be effectively applied after performing the self-supervised pre-training. In our future work, we will develop a model architecture that is suitable for handling much longer documents.

References

Xinchi Chen, Xipeng Qiu, and Xuanjing Huang. 2016. Neural sentence ordering. *arXiv preprint arXiv:1607.06952*.

Baiyun Cui, Yingming Li, Ming Chen, and Zhongfei

Zhang. 2018. Deep attentive sentence ordering network. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4340–4349.

Daniel Dahlmeier, Hwee Tou Ng, and Siew Mei Wu. 2013. Building a large annotated corpus of learner english: The nus corpus of learner english. In *Proceedings of the eighth workshop on innovative use of NLP for building educational applications*, pages 22–31.

Zeyu Dai and Ruihong Huang. 2018. Improving implicit discourse relation classification by modeling inter-dependencies of discourse units in a paragraph. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 141–151.

John Deaton. 2019. Transformers and pointer-generator networks for abstractive summarization.

Roman Grundkiewicz, Marcin Junczys-Dowmuntz, and Kenneth Heafield. 2019. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In *Proc. Workshop on Innovative Use of NLP for Building Educational Applications (BEA Workshop)*, pages 252–263.

Mandy Guo, Zihang Dai, Denny Vrandečić, and Rami Al-Rfou. 2020. Wiki-40b: Multilingual language model dataset. In *Proc. Language Resources and Evaluation Conference (LREC)*, pages 2440–2452.

Ting-Hao Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, et al. 2016. Visual storytelling. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics (ACL)*, pages 1233–1239.

Mana Ihori, Naoki Makishima, Tomohiro Tanaka, Akihiko Takashima, Shota Orihashi, and Ryo Masumura. 2021a. Mapgn: Masked pointer-generator network for sequence-to-sequence pre-training. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7563–7567.

Mana Ihori, Naoki Makishima, Tomohiro Tanaka, Akihiko Takashima, Shota Orihashi, and Ryo Masumura. 2021b. Zero-Shot Joint Modeling of Multiple Spoken-Text-Style Conversion Tasks Using Switching Tokens. In *Proc. International Speech Communication Association (Interspeech)*, pages 776–780.

Mana Ihori, Akihiko Takashima, and Ryo Masumura. 2020. Parallel corpus for Japanese spoken-to-written style conversion. In *Proc. Language Resources and Evaluation Conference (LREC)*, pages 6346–6353.

Takumi Ito, Tatsuki Kuribayashi, Masatoshi Hidaka, Jun Suzuki, and Kentaro Inui. 2020. Langsmith: An interactive academic text revision system. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 216–226.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2016. Phrase-based machine translation is state-of-the-art for automatic grammatical error correction. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1546–1556.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Shubha Guha, and Kenneth Heafield. 2018. Approaching neural grammatical error correction as a low-resource machine translation task. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 595–606.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proc. Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 4171–4186.

Shun Kiyono, Jun Suzuki, Masato Mita, Tomoya Mizumoto, and Kentaro Inui. 2020. An empirical study of incorporating pseudo data into grammatical error correction. In *Proc. Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1236–1242.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7871–7880.

Zhe Lin, Yitao Cai, and Xiaojun Wan. 2021. Towards document-level paraphrase generation with sentence rewriting and reordering. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1033–1044.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2019. On the variance of the adaptive learning rate and beyond. In *Proc. International Conference on Learning Representations (ICLR)*.

Yang Liu, Sujian Li, Xiaodong Zhang, and Zhifang Sui. 2016. Implicit discourse relation classification via multi-task neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. 2020. Does label smoothing mitigate label noise? In *Proc. the International Conference on Machine Learning (ICML)*, pages 6448–6458.

Tomoya Mizumoto and Yuji Matsumoto. 2016. Discriminative reranking for grammatical error correction with statistical machine translation. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 1133–1138.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2015. Ground truth for grammatical error correction metrics. In *Proc. Annual Meeting on Association for Computational Linguistics (ACL)*, pages 588–593.

Sascha Rothe, Sebastian Krause, Jonathan Mallinson, Eric Malmi, and Aliaksei Severyn. 2021. A simple recipe for multilingual grammatical error correction. In *Proc. Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 702–707.

Yu Sawai, Mamoru Komachi, and Yuji Matsumoto. 2013. A learner corpus-based approach to verb suggestion for ESL. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 708–713.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. In *Proc. International Conference on Machine Learning (ICML)*, pages 5926–5936.

Toshikazu Tajiri, Mamoru Komachi, and Yuji Matsumoto. 2012. Tense and aspect error correction for ESL learners using global context. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 198–202.

Yu Tanaka, Yugo Murawaki, Daisuke Kawahara, and Sadao Kurohashi. 2020. Building a Japanese typo dataset from Wikipedia's revision history. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 230–236.

Chung-Ting Tsai, Jhih-Jie Chen, Ching-Yu Yang, and Jason S. Chang. 2020. LinggleWrite: a coaching system for essay writing. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 127–133.

Tianming Wang and Xiaojun Wan. 2019. Hierarchical attention networks for sentence ordering. In *Proc. AAAI Conference on Artificial Intelligence (AAAI)*, pages 7184–7191.

Yongjing Yin, Linfeng Song, Jinsong Su, Jiali Zeng, Chulun Zhou, and Jiebo Luo. 2019. Graph-based neural sentence ordering. *arXiv preprint arXiv:1912.07225*.

Zheng Yuan and Ted Briscoe. 2016. Grammatical error correction using neural machine translation. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 380–386.