# Corruption Is Not All Bad: Incorporating Discourse Structure Into Pre-Training via Corruption for Essay Scoring

Farjana Sultana Mim, Naoya Inoue, Paul Reisert, Hiroki Ouchi, and Kentaro Inui

*Abstract*—**Existing approaches for automated essay scoring and document representation learning typically rely on discourse parsers to incorporate discourse structure into text representation. However, the performance of parsers is not always adequate, especially when they are used on noisy texts, such as student essays. In this paper, we propose an unsupervised pre-training approach to capture discourse structure of essays in terms of coherence and cohesion that does not require any discourse parser or annotation. We introduce several types of token, sentence and paragraph-level corruption techniques for our proposed pre-training approach and augment masked language modeling pre-training with our pre-training method to leverage both contextualized and discourse information. Our proposed unsupervised approach achieves a new state-of-the-art result on the task of essay Organization scoring.**

*Index Terms*—**Natural Language Processing, Automated Essay Scoring, Unsupervised Learning, Pre-training, Discourse, Cohesion, Coherence, Corruption.**

## I. Introduction

AUTOMATED Essay Scoring (AES), the task of grading and evaluating written essays using machine learning techniques, is an important educational application of natural language processing (NLP). Since manual grading of student essays is extremely time consuming and requires a lot of human effort, AES systems are widely adopted for many large-scale writing assessments such as the Graduate Record Examination (GRE) [1]. Recent studies in AES not only focus on scoring overall quality (i.e., holistic scoring) of essays but also scoring a particular dimension of essay quality (e.g., Organization, Argument Strength, Style) in order to provide constructive feedback to learners [2]–[9].

In general, an essay is a discourse where sentences and paragraphs are logically connected to each other to provide comprehensive meaning. Conventionally, two types of connections have been discussed in the literature: *coherence* and *cohesion* [10]. Coherence refers to the semantic relatedness among sentences and logical order of concepts and meanings in a text. For example, *"I saw Jill on the street. She was going home."* is coherent, whereas *"I saw Jill on the street. She has two sisters."* is incoherent. Two types of coherence are well known in the literature: *local coherence* and *global coherence*. Local coherence generally refers to how well-connected adjacent sentences are [11] whereas global coherence represents the discourse relation among remote sentences to present the main idea of the text [12], [13]. Cohesion refers to how well sentences and paragraphs in a text are linked by means of linguistic devices. Examples of these linguistic devices include conjunctions such as discourse indicators (DIs) (e.g., *"because" and "for example"*), coreference (*e.g., "he" and "they"*), substitution, ellipsis, etc.

For the precise assessment of overall essay quality or some dimensions of an essay, it is crucial to encode such discourse structure (i.e., coherence and cohesion) into an essay representation. One such dimension of an essay is *Organization*, which refers to how good an essay structure is [2]. Essays with high Organization score have a structure where writers introduce a topic first, state their position regarding the topic, support their position by providing reasons, and finally conclude by repeating their position.

An example of the relation between coherence, cohesion, and an essay's Organization is shown in Fig. 1. The high-scored essay (i.e., Essay (a) with an Organization score of 4) first states its position regarding the prompt and then provides several reasons to strengthen the claim. The essay is considered coherent because it follows a logical order that makes the writer's position and arguments very clear. However, Essay (b) is not clear on its position and what it is arguing about. The third paragraph gives a vibe that the writer is supporting the prompt, but then the fourth paragraph provides a clear statement that the writer is opposing

Manuscript received October 21, 2020; revised May 3, 2021; accepted May 31, 2021. Date of publication June 10, 2021; date of current version July 2, 2021. This work was supported in part by JSPS KAKENHI under Grant 19K20332 and in part by JST CREST under Grant JPMJCR20D2. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Zoraida Callejas. *(Corresponding author: Farjana Sultana Mim.)*

Farjana Sultana Mim is with the Graduate School of Information Sciences, Tohoku University, Sendai, Miyagi 980-8579, Japan (e-mail: farjana.mim59@gmail.com).

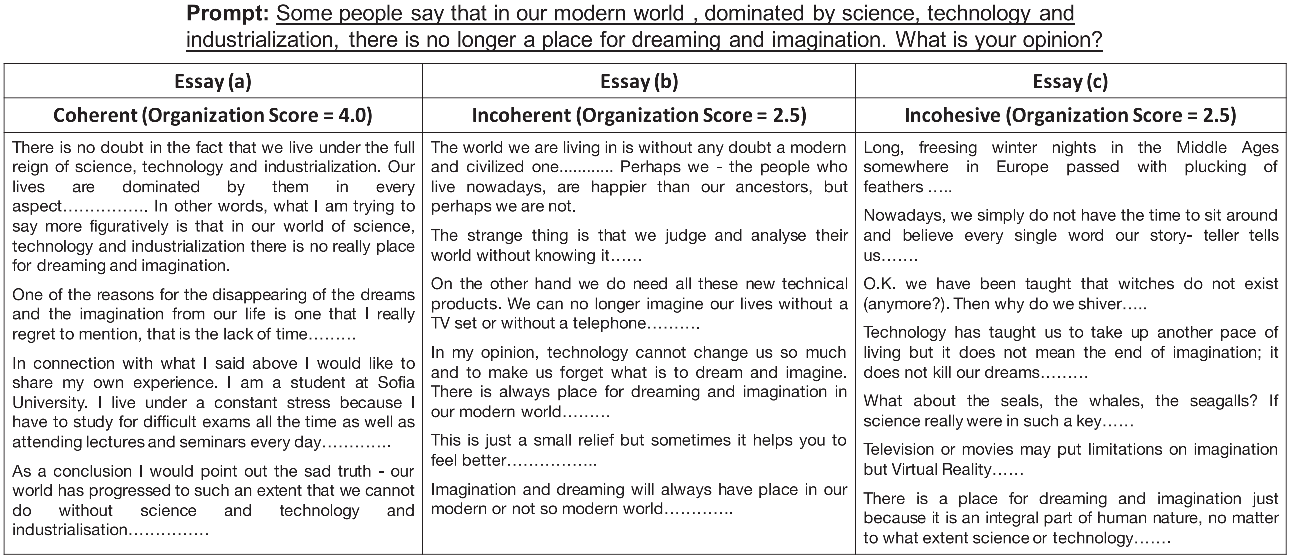
Naoya Inoue was with the Graduate School of Information Sciences, Tohoku University, Sendai, Miyagi 980-8579, Japan, and also with the Riken Center for Advanced Intelligence Project, Sendai, Miyagi 980-8579, Japan. He is now with the Department of Computer Science, Stony Brook University, NY 11794-2424 USA (e-mail: naoya.inoue.lab@gmail.com).

Paul Reisert is with the Riken Center for Advanced Intelligence Project, Sendai, Miyagi 980-8579, Japan, and also with the Graduate School of Information Sciences, Tohoku University, Sendai, Miyagi 980-8579, Japan (e-mail: paul.reisert@riken.jp).

Hiroki Ouchi is with the Riken Center for Advanced Intelligence Project, Sendai, Miyagi 980-8579, Japan (e-mail: hiroki.ouchi@riken.jp).

Kentaro Inui is with the Graduate School of Information Sciences, Tohoku University, Sendai, Miyagi 980-8579, Japan, and also with the Riken Center for Advanced Intelligence Project, Sendai, Miyagi 980-8579, Japan (e-mail: inui@tohoku.ac.jp).

Digital Object Identifier 10.1109/TASLP.2021.3088223



**Prompt:** Some people say that in our modern world , dominated by science, technology and industrialization, there is no longer a place for dreaming and imagination. What is your opinion?

| **Essay (a)** | **Essay (b)** | **Essay (c)** |
|---|---|---|
| **Coherent (Organization Score = 4.0)** | **Incoherent (Organization Score = 2.5)** | **Incohesive (Organization Score = 2.5)** |
| There is no doubt in the fact that we live under the full reign of science, technology and industrialization. Our lives are dominated by them in every aspect……………. In other words, what I am trying to say more figuratively is that in our world of science, technology and industrialization there is no really place for dreaming and imagination.<br>One of the reasons for the disappearing of the dreams and the imagination from our life is one that I really regret to mention, that is the lack of time………<br>In connection with what I said above I would like to share my own experience. I am a student at Sofia University. I live under a constant stress because I have to study for difficult exams all the time as well as attending lectures and seminars every day…………<br>As a conclusion I would point out the sad truth - our world has progressed to such an extent that we cannot do without science and technology and industrialisation…………… | The world we are living in is without any doubt a modern and civilized one........... Perhaps we - the people who live nowadays, are happier than our ancestors, but perhaps we are not.<br>The strange thing is that we judge and analyse their world without knowing it……<br>On the other hand we do need all these new technical products. We can no longer imagine our lives without a TV set or without a telephone……….<br>In my opinion, technology cannot change us so much and to make us forget what is to dream and imagine. There is always place for dreaming and imagination in our modern world………<br>This is just a small relief but sometimes it helps you to feel better……………..<br>Imagination and dreaming will always have place in our modern or not so modern world………… | Long, freesing winter nights in the Middle Ages somewhere in Europe passed with plucking of feathers …..<br>Nowadays, we simply do not have the time to sit around and believe every single word our story- teller tells us…….<br>O.K. we have been taught that witches do not exist (anymore?). Then why do we shiver…..<br>Technology has taught us to take up another pace of living but it does not mean the end of imagination; it does not kill our dreams………<br>What about the seals, the whales, the seagalls? If science really were in such a key……<br>Television or movies may put limitations on imagination but Virtual Reality……<br>There is a place for dreaming and imagination just because it is an integral part of human nature, no matter to what extent science or technology……. |

Fig. 1. Example of coherent/cohesive and incoherent/incohesive essays with their respective Organization score. The essays have been shortened for the example, indicated by ellipses.

the prompt. Therefore, it can be considered incoherent since it lacks logical sequencing.

Furthermore, Essay (a) has cohesive markers (e.g., "in connection with," "as a conclusion") at the beginning of paragraphs which helps the reader understand the flow of ideas throughout the essay. Thus, it is considered as a cohesive essay. However, Essay (c) should have some cohesive markers at the beginning of fifth paragraph (e.g., "moreover," "besides") and sixth paragraph (e.g., "therefore," "hence") to connect the ideas between paragraphs, but it doesn't have such cohesive markers. In addition, there is no cohesive marker at the beginning of the last paragraph (e.g., "in conclusion") to indicate that the author is summing up their opinions which makes the last paragraph slightly disconnected from former paragraphs. Due to the absence of these cohesive markers, it is difficult to understand the arguments of the essay and connections between them. Therefore, Essay (c) is considered as an incohesive essay.

Although discourse is one of the most important aspects of documents, less attention has been given to capturing discourse structure in an unsupervised manner for document representation. Most of the works that encapsulate discourse structure into document representation are dependent on argument or Rhetorical Structure Theory (RST) based parser and annotations [14]–[16]. However, such annotations are costly, and parsers generally considers that the text is well-written which is not always true, especially in case of student essays that comprise different types of flaws (e.g., grammatical, spelling, discourse etc.). To sum up, using parsers for document representation has its own limitations [17], especially when used on poorly written text, and it has not yet been explored how long-range discourse dependencies can be included in text embeddings in an unsupervised way without any expensive parser or annotation.

Recent advances in language model (LM) pre-training has inspired researchers to use contextualized language representations for different document-level downstream tasks of NLP, including essay scoring. Several document-level tasks such as document classification, summarization [18]–[20] as well as essay scoring [21]–[23] achieved state-of-the-art performance by leveraging pre-trained language models. Note that many of these tasks obtained only the sentence or text block representations from pre-trained language models instead of a whole document representation and subsequently joined them using some complex architecture, because Transformer-based [24] pre-trained models (e.g., BERT, RoBERTa [25], [26]) are unable to process long document due to token constraints (i.e., they accept up to 512 tokens). Furthermore, due to the self-attention operation of Transformer, processing long documents is very expensive. The recent work of Beltagy *et al.* [27] addressed these limitations and introduced Transformer-based model *Longformer* which is suitable for processing long documents. However, long-range discourse dependencies are not well captured by the pre-trained language models [20] because of the token and sentence level pre-training (not document level).

In this paper, we propose an unsupervised method that enhances a document encoder to capture discourse structure of essay Organization in terms of cohesion and coherence (Section III,IV). We name our unsupervised technique *Discourse Corruption (DC)* pre-training. We introduce several types of token, sentence, and paragraph level corruption strategies to artificially produce "badly-organized" (incoherent/incohesive) essays. We then pre-train a document encoder which learns to discriminate between original (coherent/cohesive) and corrupted (incoherent/incohesive) essays.

We augment Longformer [27], a strong document encoder pre-trained with Masked Language Modeling (MLM) objective, with our proposed DC pre-training in order to utilize both contextual and discourse information of essays. We assume that the MLM objective will capture the transition of ideas at the local level (e.g., word or sentence level) while our DC pre-training will capture the transition of ideas at global level (e.g., paragraph), and the combination of these two strategies will successfully capture the overall Organization structure of an essay. To the best of our knowledge, we are the first to attach discourse-aware pre-training on top of MLM pre-training. The advantage of our approach is that it is unsupervised and does not require any expensive parser or annotation. Our proposed strategy outperforms two baseline models by a significant margin, and we achieve new state-of-the-art results for essay Organization scoring (Section V,VI). We make our implementation publicly available.[1]

## II. Related Work

The focus of this study is the unsupervised encapsulation of discourse structure into document representation for essay Organization scoring. In this section, we briefly review the previous works on automated essay scoring, unsupervised document representation learning, and document representation learning using pre-trained language models.

### A. Automated Essay Scoring

AES research generally follows two lines of approaches: feature-engineering approach and deep neural network (DNN) based approach. Traditional AES research utilizes handcrafted features in a supervised regression or classification setting to predict the score of essays [1], [2], [5], [7], [28]–[30]. Recent studies of AES adopt DNN-based approaches which have shown promising results [31]–[39].

A major shortcoming of many of the AES systems is that they use holistic score of essays [30]–[32], [34], [38]. Holistic scoring schemes limit the scope of providing constructive feedback to learners since it is not clear how different dimensions of essay quality (e.g., Organization, content, etc.) are summarized into a single score or whether the score refers to only one dimension. In order to address this problem, recent studies have focused on scoring specific dimensions of essay such as Organization, Argument strength [2], [5], [7], Thesis clarity [3], Relevance to prompt [4], [40], Stance [6], Style [8], etc.

Many aspects of essay quality have been exploited for the assessment of essays, and among them, the one that is used often is discourse coherence. Mesgar *et al.* [41] used an end-to-end local coherence model for the assessment of essays that encodes semantic relations of two adjacent sentences and their pattern of changes throughout the text. Farag *et al.* [36] evaluated the robustness of a neural AES model and showed that neural AES models are not well-suited for capturing adversarial input of grammatically correct but incoherent sequences of sentences. Therefore, they developed a neural local coherence model and jointly trained it with a state-of-the-art AES model to build an adversarially robust AES system. However, these works utilized the particular essay quality "coherence" for the assessment of overall essay quality (holistic scoring). In contrast to these previous works, we capture discourse cohesion and coherence in an unsupervised way to assess a specific dimension of essays i.e., Organization.

Recently, pre-trained deep language representation models have fascinated the NLP community by achieving state-of-the-art results on various downstream tasks of NLP, including essay scoring. One of the widely used masked language models is *BERT* [25], which was trained with MLM objective i.e., predicting the masked tokens in the text. In addition to the MLM objective, BERT is also trained with "next sentence prediction" task i.e., predicting if the second sentence of a sentence-pair is the actual next sentence or not. Several essay scoring tasks achieved state-of-the-art performance by leveraging BERT. Steimel *et al.* [21] fine-tuned BERT and achieved a state-of-the-art result for content scoring of essays. Liu *et al.* [22] proposed a two-stage learning framework (TSLF) that integrates both end-to-end neural AES model as well as feature-engineered model and achieved state-of-the-art performance on holistic scoring of essays. In their framework, sentence embeddings are obtained using the pre-trained BERT model. They also incorporated a Grammar Error Correction (GEC) system into their AES model and added adversarial samples to the original dataset which led to a performance gain. Nadeem *et al.* [23] used existing discourse-aware models and tasks from literature to pre-train AES models for holistic scoring of essays. They utilized contextualized BERT embeddings for the AES task, hypothesizing that the next sentence prediction task of BERT would capture discourse coherence. They also pre-trained their models with other objectives i.e., natural language inference and discourse marker prediction tasks. Their results showed that contextualized embeddings from BERT performs better than other two pre-training tasks. However, all these studies consider holistic scores where it is unclear which criteria of the essay the score considers. We are the first to show how Transformer-based [24] architecture with MLM pre-training performs on the assessment of a specific dimension of essays, i.e. essay Organization scoring.

Persing *et al.* [2] annotated essays with Organization scores and established a baseline model for this scoring. They employed heuristic rules utilizing various DIs, words, and phrases to capture the discourse function labels of sentences and paragraphs of an essay. Those function labels were then exploited by various techniques, such as sequence alignment, alignment kernels, and string kernels, for the prediction of Organization score. Later, Wachsmuth *et al.* [7] achieved state-of-the-art performance on Organization scoring by utilizing argumentative features such as sequence of argumentative discourse units (ADU) (e.g., *(conclusion, premise, conclusion)*, *(None, Thesis)*), frequencies of ADU types, etc. In addition to the argumentative features, they also used sequences of paragraph discourse functions of Persing *et al.* [2] as well as sentiment flows, relation flows, POS n-grams, frequency of tokens in training essays, etc. A simple, supervised regression model is then applied for scoring. However, their work used an argument parser to obtain ADUs, and in this work, we focus on overcoming that parser bottleneck for capturing discourse.

It should be noted that our proposed unsupervised DC pre-training was first introduced in our previous works [9], [42]. The document representation obtained from DC pre-training was used for essay Organization and Argument Strength scoring.

[1] Our implementation is publicly available at https://github.com/FarjanaSultanaMim/DisCorrupto

However, in this study, we only focus on essay Organization scoring. In this work, we present several new corruption techniques in addition to our previous corruption strategies [9] to capture the Organization structure of essays. Besides, in contrast to our previous research, in this study we use a Transformer-based model pre-trained with MLM objective as our document encoder and augment our DC pre-training on top of it. To elaborate, in this paper, we extend our previous research by introducing new corruption techniques and by enhancing a pre-trained document encoder with our DC pre-training to capture discourse structure of essay Organization.

### *B. Unsupervised Document Representation Learning*

Several unsupervised methods for document representation learning have been introduced in recent years [43]–[46]. However, less studies have been conducted on unsupervised learning of discourse-aware text representations. One of the studies that illustrated the role of discourse structure for document representation is the study by Ji and Smith [17] who implemented a discourse structure (defined by RST) [16] aware model and showed that their model improves text categorization performance (e.g., sentiment classification of movies and Yelp reviews, and prediction of news article frames). The authors utilized an RST-parser to obtain the discourse dependency tree of a document and then built a recursive neural network on top of it. The issue with their approach is that texts need to be parsed by an RST parser and the parsing performance of RST is not always adequate, especially when used on noisy text. Furthermore, the performance of RST parsing is dependent on the genre of documents [17].

### *C. Pre-Trained Language Models and Document Representation Learning*

Lately, Tansformer-based pre-trained models have achieved significant performance gain in different document-level downstream tasks of NLP. Adhikari *et al.* [18] first investigated the effect of pre-trained deep contextualized models on document representation learning. They fine-tuned BERT [25] for several document classification tasks and demonstrated that knowledge can be distilled from BERT to small bidirectional LSTMs which provides competitive results at a low computational expense.

Chang *et al.* [47] proposed methods for pre-training hierarchical document representations that generalize and extend the pre-training method of ELMo [48] and BERT [25], respectively. In their approach, LSTM-based architecture consider a document as sequences of text blocks, each block comprising a sequence of tokens, where the text blocks are basically sentences or paragraphs. Zhang *et al.* [19] presented a strategy to pre-train hierarchical bidirectional transformer encoders for document representation. They randomly masked sentences of documents and predicted those masked sentences with their proposed architecture, a hierarchical fusion of Transformer-based [24] sentence and document encoders.

A recent work by Beltagy *et al.* [27] indicated the attention mechanism and token constraints of Transformer-based [24] masked language models for long document representation. To mitigate these problems, they introduced a Transformer-based model *Longformer*, which has an attention mechanism that scales linearly with the sequence length, hence being suitable for processing long documents. They pre-trained Longformer with the MLM objective, continuing from the RoBERTa [26] released checkpoint and added extra position embeddings to support long sequence of tokens. The pre-trained Longformer outperformed renowned RoBERTa on various long document tasks.

One recent study by Xu *et al.* [20] utilized a pre-trained language model for capturing the discourse structure of documents. They constructed a discourse-aware neural extractive summarization model *DISCOBERT*. DISCOBERT encodes RST-based discourse unit (a sub-sentence phrase) instead of sentence using BERT. A Graph Convolutional Network is then used to create discourse graphs based on RST trees and coreference mentions. However, this work is dependent on the RST discourse parser, and as mentioned a priori, we would like to overcome that parser bottleneck.

## III. Model Architecture

### *A. Overview*

Our model consists of (i) a base document encoder, (ii) an auxiliary encoder, and (iii) a scoring function. The base document encoder produces a vector representation $\mathbf{h}^{\text{base}}$ by capturing a sequence of words in each essay. The auxiliary encoder captures additional essay-related information and produces a vector representation $\mathbf{h}^{\text{aux}}$.

Then, these representations are concatenated into one vector, which is mapped to a feature vector $\mathbf{z}$.

$$\mathbf{z} = \tanh(\mathbf{W} \cdot [\mathbf{h}^{\text{base}}; \mathbf{h}^{\text{aux}}]) \ , \tag{1}$$

where $\mathbf{W}$ is a weight matrix. Finally, we use the following scoring function to map $\mathbf{z}$ to a scalar value by the sigmoid function.

$$y = \text{sigmoid}(\mathbf{w} \cdot \mathbf{z} + b) \ ,$$

where $\mathbf{w}$ is a weight vector, $b$ is a bias value, and $y$ is a score in the range of $[0, 1]$. In the following subsections, we describe the details of each encoder.

### *B. Base Document Encoder*

The base document encoder produces a document representation $\mathbf{h}^{\text{base}}$ in Equation 1. For the base document encoder, we use the pre-trained Longformer model [27].

Longformer is a Transformer-based [24] model with a modified attention mechanism. Longformer's attention mechanism scales linearly with the input sequence length, making it easy for processing long documents. The attention mechanism of Longformer combines a sliding windowed self-attention for capturing local-context and a task specific global attention. In this attention operation, if the sliding window size is $w$, then each token will attend to $\frac{1}{2}w$ token on each side, and a token with a global attention will attend to all the tokens across the sequence and all the tokens in the sequence will attend to it as well. Longformer is pre-trained with the MLM objective, continued from the RoBERTa released checkpoint. During pretraining, Longformer's attention mechanism is used as a drop-in replacement for the self-attention mechanism of Transformer-based RoBERTa [26]. Specifically, RoBERTa's self-attention is replaced by Longformer's attention. Longformer can process much longer documents by accepting up to 4096 tokens, whereas other pre-trained models like BERT [25] or RoBERTa [26] only accept up to 512 tokens. Since the Transformer architecture [24] is well-known and widely used in NLP, we will omit the detailed information. Instead, we present a brief overview of how Longformer is used in our essay scoring model.

Given an input essay of $N$ tokens $t_{1:N} = (t_1, t_2, \ldots, t_N)$, special tokens are inserted at the beginning and the end of the essay, with the input essay of $N$ tokens as $t_{0:N+1} = ([\text{CLS}], t_1, t_2, \ldots, t_N, [\text{EOS}])$. Next, taking $t_{0:N+1}$ as input, the Longformer model produces a sequence of contextual representations $\mathbf{h}_{0:N+1} = (\mathbf{h}_0, \mathbf{h}_1, \ldots, \mathbf{h}_{N+1})$. Note that, we obtain the representation from the second-to-last layer of Longformer.

$$\mathbf{h}_{0:N+1} = \text{Longformer}(t_{0:N+1}) \ ,$$

Next, we use a mean-over-time layer $\mathbf{h}_{0:N+1}$ as input, which produces a vector averaged over the sequence.

$$h^{\text{mean}} = \frac{1}{N+2} \sum_{n=0}^{N+1} \mathbf{h}_n \ . \tag{2}$$

We use this resulting vector as the base document representation, i.e. $\mathbf{h}^{\text{base}} = \mathbf{h}^{\text{mean}}$.

### C. Auxiliary Encoder (AE)

The auxiliary encoder produces a representation of a sequence of paragraph function labels $\mathbf{h}^{\text{aux}}$ in Equation 1.

Each paragraph in an essay plays a different role. For instance, the first paragraph tends to introduce the topic of the essay, and the last paragraph tends to sum up the whole content and make some conclusions. Here, we capture such paragraph functions.

Specifically, we obtain paragraph function labels of essays using Persing *et al.*'s [2] heuristic rules.[2] Persing *et al.* [2] specified four paragraph function labels: Introduction (**I**), Body (**B**), Rebuttal (**R**) and Conclusion (**C**). We represent these labels as vectors and incorporate them into our model. Our auxiliary encoder which encodes paragraph function labels consists of two modules, an embedding layer and a Bi-directional Long Short-Term Memory (BiLSTM) [49] layer.

We assume that an essay consists of $M$ paragraphs, and the $i$-th paragraph has already been assigned a function label $p_i$. Given the sequence of paragraph function labels of an essay $p_{1:M} = (p_1, p_2, \ldots, p_M)$, the embedding layer (Emb$^{\text{para}}$) produces a sequence of label embeddings $\mathbf{p}_{1:M} = (\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_M)$.

$$\boldsymbol{p}_{1:M} = \text{Emb}^{\text{para}}(p_{1:M}),$$

where each embedding $\boldsymbol{p}_i$ is $\mathbb{R}^{d^{\text{para}}}$. Note that each embedding is randomly initialized and learned during training.

Then, taking $\boldsymbol{p}_{1:M}$ as input, the BiLSTM layer produces a sequence of vector representations $\mathbf{h}_{1:M} = (\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_M)$.

$$\mathbf{h}_{1:M} = \text{BiLSTM}(\boldsymbol{p}_{1:M}),$$

where $\mathbf{h}_i$ is $\mathbb{R}^{d^{\text{aux}}}$.

We use the last hidden state $\mathbf{h}_M$ as the paragraph function label sequence representation, i.e. $\mathbf{h}^{\text{aux}} = \mathbf{h}_M$.

## IV. Proposed Pre-Training Method

### A. Overview

Fig. 2 summarizes our proposed DC pre-training method. First, we pre-train the base document encoder (Section III-B) to distinguish between original and their artificially corrupted documents. This pre-training is motivated by the following hypotheses: (i) artificially corrupted incoherent/incohesive documents lack logical sequencing, (ii) moderately corrupted documents have better logical sequencing compared to highly corrupted documents and (iii) training a base document encoder to differentiate between original documents and their different types of artificially corrupted documents makes the encoder logical sequence-aware, in other words, discourse-aware. Based on these hypotheses, we train a base document encoder on the original documents and their artificially corrupted documents.

The pre-training is done in two steps. First, the document encoder is pre-trained with large-scale, unlabeled essays from various corpora. Second, the encoder is fine-tuned on the unlabeled essays of the target corpus (essay Organization scoring corpus). We expect that this fine-tuning alleviates the domain mismatch between the large-scale essays and target essays (e.g., essay length). Finally, the pre-trained encoder is then re-trained on the annotations for the essay scoring task in a supervised manner.

Note that our base document encoder (i.e., Longformer) is already pre-trained with the MLM objective, where the aim is to predict randomly masked tokens in a sequence. Previous work (e.g., [23]) have shown that the next sentence prediction task of BERT i.e., predicting whether the subsequent sentence of a sentence-pair is the actual next sentence or not, is able to capture discourse coherence. Hence, we also pre-train our model with the binary *next sentence prediction* (N-SentP) task, similar to BERT's. The sentence-pairs are generated from our pre-training corpora and we follow BERT's strategy for the generation of these sentence-pairs. More specifically, when we choose the sentences A and B for each sentence-pair, 50% of the time B is the actual next sentence that follows A and 50% of the time B is a random sentence[3]. We hypothesize that the MLM and N-SentP pre-training would capture local-context while our DC pre-training would capture the long-range dependencies effective for essay Organization scoring.

### B. Corruption Strategies

We would like to produce "badly organized" essays with our corruption techniques so that the encoder can learn the

[2] See http://www.hlt.utdallas.edu/~persingq/ICLE/orgDataset.html for further details.

[3] See Appendix VII for more details

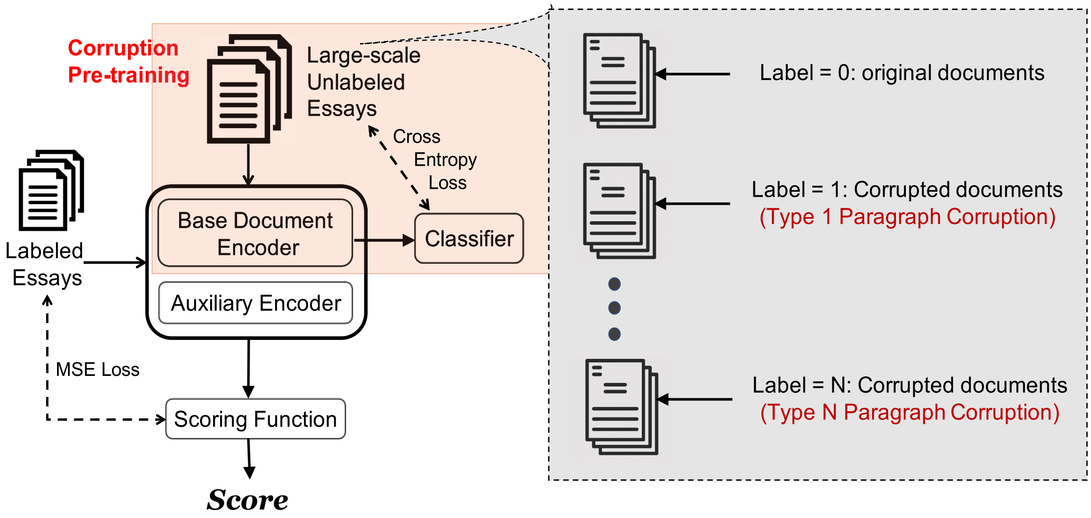
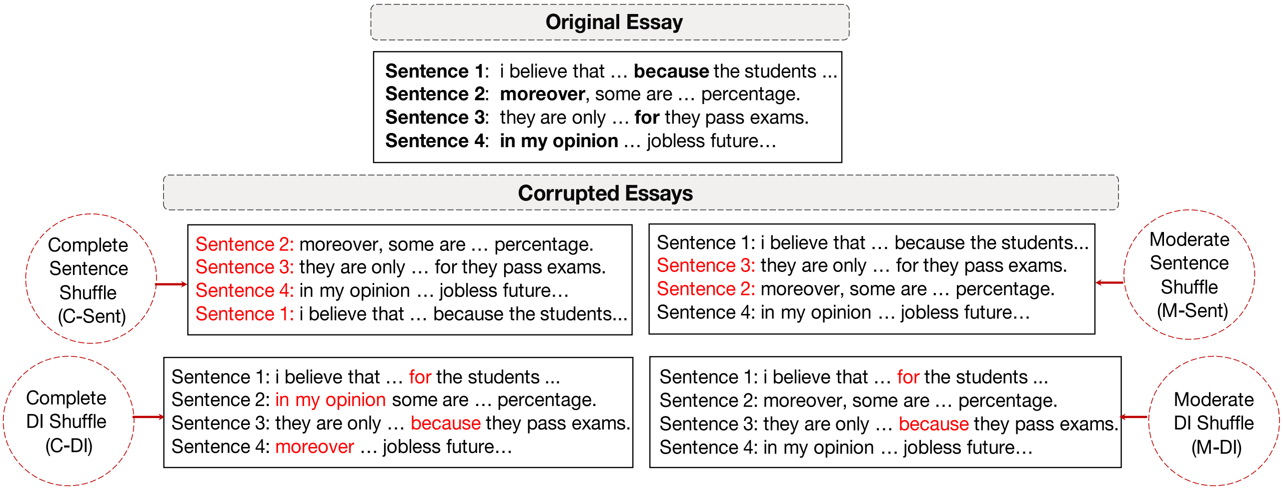

Corruption Pre-training
Large-scale Unlabeled Essays
Cross Entropy Loss
Labeled Essays
Base Document Encoder
Classifier
Auxiliary Encoder
MSE Loss
Scoring Function
Score
Label = 0: original documents
Label = 1: Corrupted documents
(Type 1 Paragraph Corruption)
Label = N: Corrupted documents
(Type N Paragraph Corruption)

Fig. 2. Proposed DC pre-training for unsupervised learning of discourse-aware text representation utilizing original and artificially corrupted documents and the use of the discourse-aware pre-trained model for essay scoring.


Original Essay
Corrupted Essays
Complete Sentence Shuffle (C-Sent)
Moderate Sentence Shuffle (M-Sent)
Complete DI Shuffle (C-DI)
Moderate DI Shuffle (M-DI)

**Sentence 1**: i believe that … **because** the students …
**Sentence 2**: **moreover**, some are … percentage.
**Sentence 3**: they are only … **for** they pass exams.
**Sentence 4**: **in my opinion** … jobless future…

Sentence 2: moreover, some are … percentage.
Sentence 3: they are only … for they pass exams.
Sentence 4: in my opinion … jobless future…
Sentence 1: i believe that … because the students…

Sentence 1: i believe that … because the students…
Sentence 3: they are only … for they pass exams.
Sentence 2: moreover, some are … percentage.
Sentence 4: in my opinion … jobless future…

Sentence 1: i believe that … for the students …
Sentence 2: in my opinion some are … percentage.
Sentence 3: they are only … because they pass exams.
Sentence 4: moreover … jobless future…

Sentence 1: i believe that … for the students …
Sentence 2: moreover, some are … percentage.
Sentence 3: they are only … because they pass exams.
Sentence 4: in my opinion … jobless future…

Fig. 3. Example of different types of Sentence and Discourse Indicator Corruption methods.

difference between good and bad discourse. Note that essays are not only scored as high or low but throughout a range of scores which means that there is Organization structure which is moderately good/bad. Therefore, in addition to the high corruption techniques, we introduce several types of moderate corruption techniques in order to produce "moderately bad" Organization of essays.

We categorize our corruption strategies into 3 groups: (1) *sentence*, (2) *discourse indicator (DI)* and (3) *paragraph* corruption. Each group has several types of corruption schemes. We discuss the details of each corruption strategy in the following subsections.

*1) Sentence Corruption (SC):* This group has 2 different types of corruption. In *Complete Sentence Shuffle* (C-Sent), all the sentences of a document are shuffled. In *Moderate Sentence Shuffle* (M-Sent), only a subset of the sentences of a document are shuffled. Specifically, we randomly select two sentences from a document and shuffle all the sentences between them, including those two sentences as well. Fig. 3 shows an example of C-Sent and M-Sent.

*2) Discourse Indicator Corruption (DIC):* We corrupt DIs since they represent the logical connection between sentences. For example, *"Mary did well although she was ill"* is logically connected, but *"Mary did well but she was ill."* and *"Mary did well. She was ill."* lack logical sequencing because of improper and lack of DI usage, respectively.

We perform two types of DI corruption. In *Complete Discourse Indicator Shuffle* (C-DI), we shuffle all the discourse indicators of a document. In *Moderate Discourse Indicator Shuffle* (M-DI), we first select 50% of unique DIs in a document and randomly shuffle each of their instances in a document. Fig. 3 shows an example of C-DI and M-DI.

*3) Paragraph Corruption (PC):* How ideas are transmitted throughout the paragraphs of an essay determines how good its Organization structure is. For example, coherent essays have paragraph sequences like *Introduction-Body-Conclusion*

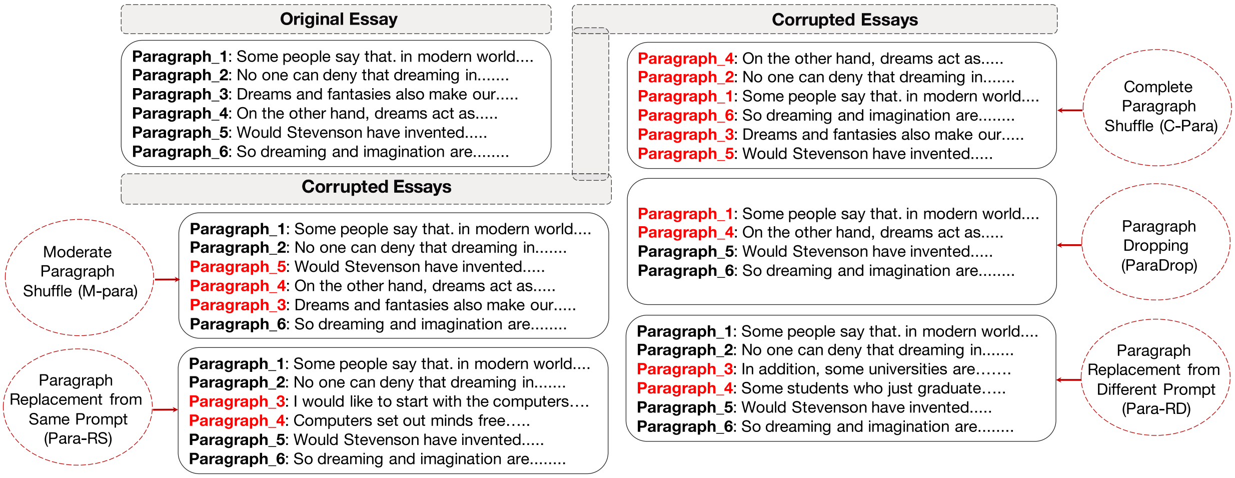

Original Essay
Paragraph_1: Some people say that. in modern world....
Paragraph_2: No one can deny that dreaming in.......
Paragraph_3: Dreams and fantasies also make our.....
Paragraph_4: On the other hand, dreams act as.....
Paragraph_5: Would Stevenson have invented.....
Paragraph_6: So dreaming and imagination are........
Corrupted Essays
Paragraph_4: On the other hand, dreams act as.....
Paragraph_2: No one can deny that dreaming in.......
Paragraph_1: Some people say that. in modern world....
Paragraph_6: So dreaming and imagination are........
Paragraph_3: Dreams and fantasies also make our.....
Paragraph_5: Would Stevenson have invented.....
Complete Paragraph Shuffle (C-Para)
Corrupted Essays
Moderate Paragraph Shuffle (M-para)
Paragraph_1: Some people say that. in modern world....
Paragraph_2: No one can deny that dreaming in.......
Paragraph_5: Would Stevenson have invented.....
Paragraph_4: On the other hand, dreams act as.....
Paragraph_3: Dreams and fantasies also make our.....
Paragraph_6: So dreaming and imagination are........
Paragraph_1: Some people say that. in modern world....
Paragraph_4: On the other hand, dreams act as.....
Paragraph_5: Would Stevenson have invented.....
Paragraph_6: So dreaming and imagination are........
Paragraph Dropping (ParaDrop)
Paragraph Replacement from Same Prompt (Para-RS)
Paragraph_1: Some people say that. in modern world....
Paragraph_2: No one can deny that dreaming in.......
Paragraph_3: I would like to start with the computers....
Paragraph_4: Computers set out minds free.....
Paragraph_5: Would Stevenson have invented.....
Paragraph_6: So dreaming and imagination are........
Paragraph_1: Some people say that. in modern world....
Paragraph_2: No one can deny that dreaming in.......
Paragraph_3: In addition, some universities are.......
Paragraph_4: Some students who just graduate.....
Paragraph_5: Would Stevenson have invented.....
Paragraph_6: So dreaming and imagination are........
Paragraph Replacement from Different Prompt (Para-RD)

Fig. 4. Example of different types of Paragraph Corruption.

to provide a logically consistent meaning of the text. Therefore, we conduct five types of paragraph corruption, as illustrated in Fig. 4.

In *Complete Paragraph Shuffle* (C-Para), we randomly shuffle all the paragraphs of a document. In *Moderate Paragraph Shuffle* (M-Para), we shuffle a subset of the paragraphs of a document. Precisely, we randomly pick two paragraphs from a document and shuffle all the paragraphs between them including those two paragraphs as well. For example, in the M-Para of Fig. 4, only paragraph 3,4 and 5 are shuffled.

In *Paragraph Drop* (ParaDrop), we drop 30% of randomly selected paragraphs of a document. Fig. 4 shows an example of ParaDrop where paragraph 2 and 3 are dropped.

In *Paragraph Replacement from Same Prompt* (Para-RS), we randomly choose two paragraphs from a document and replace all the paragraphs between them (including those two as well) with the paragraphs of another document of the same prompt. Hence, the main theme of the replaced document is still intact but the logical sequencing would be slightly distorted. Note that, during replacement of the paragraphs, the positions of the chosen paragraphs of another document are the same as the positions of the to be replaced paragraphs of the current document. For example, if we want to replace paragraph number 3 and 4 of a document, then we choose paragraph number 3 and 4 of another document of the same prompt for replacement. In the Para-RS example of Fig. 4, paragraph number 3 and 4 are replaced from paragraphs of another essay of the same prompt. Lastly, we perform a corruption called *Paragraph Replacement from Different Prompt* (Para-RD) which is same as the Para-RS but this time the paragraphs are replaced from another document of different prompt. Therefore, this corruption techniques produce incoherent documents where both main idea as well as logical sequencing are distorted. It is to be noted that, we hope to capture paragraph-level long range dependencies with these corruption strategies.

## C. *Discourse Corruption (DC) Pre-Training*

We treat DC pre-training as a multi-class (or binary) classification task where the encoder assigns a label to each document. In our experiments, we consider many combinations of corruption types (see Table I). For example, for 6-way DC pre-training, the encoder tries to predict which class the document belongs to among the 6 classes (original essays, C-Para, M-Para, ParaDrop, Para-RS, Para-RD corrupted essays). For implementation, we add a classification layer on top of the base document encoder (Section III-B). The classification layer consists of (i) a linear layer that takes $\mathbf{h}^{\text{base}}$ as input and (ii) a softmax layer. To train the model parameters, we minimize the cross-entropy loss function.

## D. *Extension of Existing Pre-Training Idea*

We also propose an extension of the idea of next sentence prediction (N-SentP) task, i.e., *next paragraph prediction* (N-ParaP) pre-training. Same as N-SentP, the objective of N-ParaP pre-training is to predict if the second paragraph of a paragraph-pair is the actual next paragraph or not. We follow the same strategy as N-SentP for the generation of paragraph-pairs i.e., when we choose paragraphs A and B for each paragraph-pair, 50% of the time B is the actual next paragraph that follows A and 50% of the time B is a random paragraph[4]. We hope to capture paragraph level dependencies to some extent with this pre-training. We treat the N-ParaP as a binary classification task that pre-trains paragraph-pair representations and we follow our two-step DC pre-training method for implementation.

# V. Experimental Setup

## A. *Data*

*1) Essay Organization Scoring:* We use the International Corpus of Learner English (ICLE) [50] for essay scoring which contains 6085 essays and 3.7 million words. Most essays (91%) are argumentative and vary in length, having 7.6 paragraphs and 33.8 sentences on average [7]. Some essays have been annotated with scores along multiple dimension among which 1003 essays are annotated with Organization scores. The scores range from 1.0 (worst score) to 4.0 (best score) at half-point increments. The

[4] The random paragraph is either chosen from the same document or from a random document in the corpora. See Appendix VII for more details

TABLE I
PERFORMANCE OF CLASSIFICATION TASKS IN THE FIRST STEP (USING LARGE-SCALE UNLABELED ESSAYS) AND SECOND STEP OF CORRUPTION PRE-TRAINING (USING UNLABELED ESSAYS OF TARGET ESSAY SCORING CORPUS)

| Pretraining Phase | Classification Task | Objective/Corruption Type Used | Validation Accuracy |
|---|---|---|---|
| 1st Step (All pre-training data) | Binary | N-SentP | 0.747 |
| | Binary | N-ParaP | 0.764 |
| | Binary | C-Sent | 0.955 |
| | Binary | M-Sent | 0.800 |
| | Binary | C-DI | 0.984 |
| | Binary | M-DI | 0.971 |
| | Binary | C-Para | 0.919 |
| | 3-way | C-Para, M-para | 0.786 |
| | 4-way | C-Para, M-para, ParaDrop | 0.770 |
| | 5-way | C-Para, M-Para, ParaDrop, Para-RS | 0.707 |
| | 6-way | C-Para, M-Para, ParaDrop, Para-RS, Para-RD | 0.734 |
| 2nd Step (Finetuned on ICLE pre-training data) | Binary | N-SentP | 0.728 |
| | Binary | N-ParaP | 0.773 |
| | Binary | C-Sent | 0.985 |
| | Binary | M-Sent | 0.781 |
| | Binary | C-DI | 1.000 |
| | Binary | M-DI | 0.998 |
| | Binary | C-Para | 0.890 |
| | 3-way | C-Para, M-Para | 0.717 |
| | 4-way | C-Para, M-Para, ParaDrop | 0.656 |
| | 5-way | C-Para, M-Para, ParaDrop, Para-RS | 0.606 |
| | 6-way | C-Para, M-Para, ParaDrop, Para-RS, Para-RD | 0.666 |

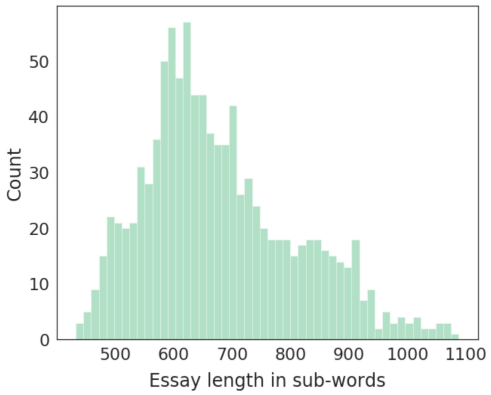


Fig. 5. Histogram of lengths of ICLE essays used in scoring

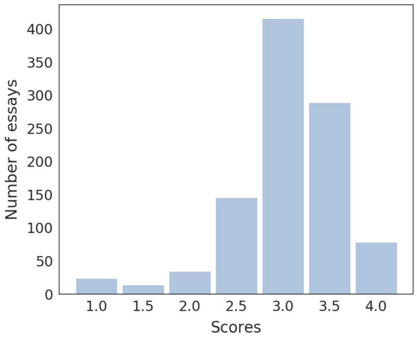


Fig. 6. Distribution of Organization scores

distribution of Organization scores is demonstrated in Fig. 6. For our scoring task, we utilize these 1003 essays. The average number of tokens per esssay is 679 (in sub-words) and the longest essay has 1090 tokens. The histogram of the essay lengths is shown in Fig. 5.

*2) DC Pre-Training:* To pre-train the document encoder, we use four datasets, (i) Kaggle's Automated Student Assessment Prize (ASAP) dataset[5] (12 976 essays) (ii) TOEFL11 [51] dataset (12 100 essays), (iii) The International Corpus Network of Asian Learners of English (ICNALE) [52] dataset (5600 essays), and (iv) ICLE essays not used for Organization scoring (4546 essays). In total, we acquire 35 222 essays from the four datasets which are used during pre-training with N-SentP, SC, and DIC. However, for pre-training with all types of PC and N-ParaP, we use only 16 646 essays (TOEFL11 and ICLE essays) since ASAP and ICNALE essays are limited to single paragraphs.

### *B. Evaluation Procedure*

We use five-fold cross-validation for evaluating our models with the same split as Persing *et al.* [2] and Wachsmuth *et al.* [7]. However, our results are not directly comparable since our training data is smaller, as we reserve a validation set (100 essays) for model selection while they do not. We use mean squared error (MSE) as an evaluation measure. The reported results are averaged over five folds.

We evaluate two learning strategies of the encoder in the essay scoring task: *fine-tuning* and *fixed*. In the fine-tuning setting, both the pre-trained base document encoder and auxiliary encoder are fine-tuned on the essay scoring task. In the fixed setting, only the parameters of the auxiliary encoder are fine-tuned.

Our first baseline model is the *Base+AE* model. In our preliminary experiments, we first experimented with different settings such as fine-tune Base (pre-trained Longformer) model then merge AE, fine-tune both Base and AE and then merge, etc. However, we found that merging both models simultaneously

[5] [Online]. Available: https://www.kaggle.com/c/asap-aes

(either in fine-tuning or fixed encoder setting) results in the best performance. Therefore, even for all the proposed systems, we merge the DC pre-trained Base model and AE at the same time in both fine-tuning and fixed-encoder settings. Our second baseline model is the *Base+AE* model pre-trained with the N-SentP task.

### *C. Preprocessing*

We use the same preprocessing steps for both pre-training and essay scoring. We lowercase the tokens and specify an essay's paragraph boundaries with special tokens. Special tokens [CLS] and [EOS] are inserted at the beginning and end of each essay respectively. We normalize the gold-standard scores to the range of [0, 1]. During pre-training with SC and DIC, paragraph boundaries are not used.

For DIC, we collect 847 DIs from the Web.[6] We exclude the DI "and" since it is not always used for initiating logic (e.g., milk, banana *and* tea). In essay scoring dataset, we found 176 DIs and around 24 DIs per essay. In the pre-training data, the total number of DIs is 204 and the average number of DIs per essay is around 13. We identified DIs by simple string-pattern matching.

### *D. Implementation Choices*

From the two sizes of pre-trained Longformer models, we use Longformer-base model. The global attention of Longformer is set on the [CLS] token. For the auxiliary encoder, we use a BiLSTM with hidden units of 200 in each layer ($d^{\text{AUX}} = 200$).

We use Adam optimizer, batch sizes of 4 on the first-step of pre-training and batch sizes of 2 on the second-step of pre-training as well as on the essay scoring. The learning rate is set to $1e - 5$ for pre-training and fine-tuning setting of essay scoring while it is set to 0.001 for fixed encoder setting of essay scoring. We use early stopping with patience 12 (5 for pre-training), and train the network for 100 epochs. In the pre-training phase, 80% of the data is used for training and 20% of the data is used for validation. We perform hyperparameter tuning for the scoring task and choose the best model. We tuned dropout rates (0.5, 0.7, 0.9) for all models on the validation set. To select hyper-parameters, we monitor performance on the validation set and choose the model that yields the lowest MSE. We choose the best model for each particular fold. In the testing phase, we re-scale the predicted normalized scores to the original range of scores and then measure the performance.

## VI. Results and Discussion

### *A. Results of DC Pre-Training*

Table I shows the classification accuracy of both steps of DC pre-training on the validation data. We observe that the document encoder learns to distinguish not only between coherent/cohesive and incoherent/incohesive documents (binary classification) but also between different types of incoherent (3,4,5 and 6 way classification) documents.

Pre-training with C-DI provides the best classification accuracy. We anticipate that since we do not change the position of the DIs during shuffling, the encoder may only learn the sequence of DIs within each essay and try to distinguish between the DI sequence of original and corrupted essays. Therefore, the task becomes easier for the encoder.

The visualization of document vectors obtained from the first and second step of DC pre-training (5-way classification task) is shown in Fig. 7. To visualize the high-dimensional document vectors into a 2-dimensional space, we use dimensionality reduction algorithm T-Distributed Stochastic Neighbouring Entities (t-SNE). Fig. 7 shows that the encoder is able to perfectly separate C-Para essays from other essays since the transition of ideas between paragraphs is fully distorted in these essays, hence easy to distinguish. We also observe that the encoder separates M-Para and ParaDrop essays better compared to Para-RS essays. Para-RS essays lie close to the original coherent essays and frequently overlap. We speculate that since we replace the paragraphs of the same positions, the sequencing of ideas of Para-RS essays is the least distorted compared to M-Para, ParaDrop or C-Para essays, hence these essays are similar to the original essays.

### *B. Results of Essay Scoring*

Table II lists MSE (averaged over five folds) of baseline models and our proposed systems (N-ParaP and DC pre-trained) for Organization scoring task.[7] It shows that the proposed unsupervised DC pre-training improves the performance of essay Organization scoring (statistically significant by Wilcoxon's signed rank test, $p < 0.05$) and we obtain significant performance gain over the baseline models. Also, we achieve new state-of-the-art result with our proposed method.

The best performance is obtained with the 5-way DC Pre-training. These results support our hypothesis that training with corrupted documents helps a document encoder learn logical sequence-aware text representations. In most of the cases, fine-tuning the encoder for scoring task provides better performance.

From Table II we observe that next paragraph prediction or paragraph corruption based DC pre-training is effective for Organization scoring while sentence and DI corruption based pre-training is not. This could be attributed to the fact that the paragraph level transition of ideas (global coherence) is not captured by sentence and DI level corruption. Besides, a manual inspection of DIs identified by the system shows that the identification of DIs is not always reliable. Almost half of DIs identified by our simple pattern matching algorithm (see Section V-C) were not actually DIs (e.g., *we have survived* **so***far only external difficulties*). We also found that some DI-shuffled documents are often cohesive. This happens when original document counterparts have two or more DIs with more or less same meaning (e.g., *since* and *because*).

It can be seen that as the classification task of Corruption Pre-training becomes more complicated by adding more corruption

[6] [Online]. Available: http://www.studygs.net/wrtstr6.htm, http://home.ku.edu.tr/~doregan/Writing/Cohesion.html etc.

[7] Our model is Base+AE model (Section III-B, III-C). The performance of the Base (pre-traned Longformer) encoder without AE and without any DC pre-training when finetuned on essay Organization scoring is: MSE = 0.246

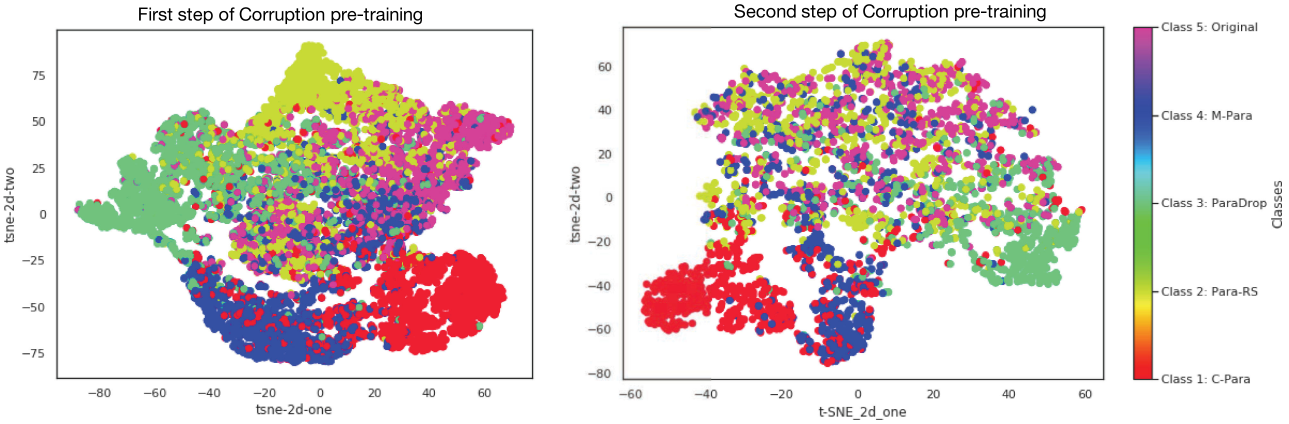


Fig. 7. Visualization of document representations obtained from DC pre-trained (5-way classification scheme) encoder.

TABLE II

PERFORMANCE OF ESSAY SCORING. NUMBERS IN **BOLD** AND **UNDERLINE** DENOTE IMPROVEMENT OVER BASELINE AND PREVIOUS STATE-OF-THE-ART RESPECTIVELY. '*' INDICATES A STATISTICAL SIGNIFICANCE (WILCOXON SIGNED-RANK TEST, $p < 0.05$) AGAINST THE BASELINES

| Model | Classification Task | Objective/ Corruption Type | Fine-tuning | Mean Squared Error<br>Organization |
|---|---|---|---|---|
| Baseline 1 | - | - | - | 0.175 |
| | - | - | ✓ | 0.181 |
| Baseline 2 | Binary | N-SentP | - | 0.185 |
| | Binary | N-SentP | ✓ | 0.196 |
| Proposed | Binary | N-ParaP | - | 0.177 |
| | Binary | N-ParaP | ✓ | **0.172** |
| | Binary | C-Sent | - | 0.184 |
| | Binary | C-Sent | ✓ | 0.198 |
| | Binary | M-Sent | - | 0.175 |
| | Binary | M-Sent | ✓ | 0.193 |
| | Binary | C-DI | - | 0.189 |
| | Binary | C-DI | ✓ | 0.185 |
| | Binary | M-DI | - | 0.183 |
| | Binary | M-DI | ✓ | 0.198 |
| | Binary | C-Para | - | **0.172** |
| | Binary | C-Para | ✓ | **0.167**$^*$ |
| | 3-way | C-Para, M-Para | - | **0.173** |
| | 3-way | C-Para, M-Para | ✓ | **0.162**$^*$ |
| | 4-way | C-Para, M-Para, ParaDrop | - | **0.169** |
| | 4-way | C-Para, M-Para, ParaDrop | ✓ | **0.157**$^*$ |
| | 5-way | C-Para, M-Para, ParaDrop, Para-RS | - | **0.166**$^*$ |
| | 5-way | C-Para, M-Para, ParaDrop, Para-RS | ✓ | **0.155**$^*$ |
| | 6-way | C-Para, M-Para, ParaDrop, Para-RS, Para-RD | - | 0.179 |
| | 6-way | C-Para, M-Para, ParaDrop, Para-RS, Para-RD | ✓ | **0.162**$^*$ |
| Persing et al. (2010) | | | | 0.175 |
| Wachsmuth et al. (2016) | | | | 0.164 |

types, the essay scoring performance improves (except for 6-way classification). We obtain the best performance with 5-way classification task. We speculate that this is because with more corruption types, the model learns more styles of transition of ideas among paragraphs as well as differences between them. Finally, the model connects those differences to scores at the essay scoring phase by figuring out which flow of concepts is better than the other.

It should be noted that 6-way classification task could not outperform 5-way classification task. This might be because of adding Para-RD corruption in 6-way classification task. Since in Para-RD, we replace the paragraphs of document with paragraphs of a document of different prompt, instead of learning the flow of the ideas throughout the text the encoder might also be learning something else (e.g, topic difference). We speculate that this confuses the document encoder at the essay scoring phase.

### *C. Analysis*

*1) Importance of Fine-Grained Corruption Types:* To investigate how important it is for the model to learn the difference between fine-grained corruption types, we collapsed four

TABLE III
ESSAY SCORING RESULTS WHEN A 5-WAY DC PRE-TRAINING IS REDUCED TO A BINARY AND 3-WAY DC PRE-TRAINING

| Model | Classification Task | Corruption Type | Fine-tuning | Mean Squared Error<br>Organization |
|---|---|---|---|---|
| Baseline 1 | - | - | - | 0.175 |
| | - | - | ✓ | 0.181 |
| Baseline 2 | - | N-SentP | - | 0.185 |
| | - | N-SentP | ✓ | 0.196 |
| Proposed | 5-way | C-Para, M-Para, ParaDrop, Para-RS | - | **0.166**$^*$ |
| | 5-way | C-Para, M-Para, ParaDrop, Para-RS | ✓ | **0.155**$^*$ |
| | 5-way to Binary | C-Para, M-Para, ParaDrop, Para-RS | - | 0.179 |
| | 5-way to Binary | C-Para, M-Para, ParaDrop, Para-RS | ✓ | 0.185 |
| | 5-way to 3-way | C-Para, M-Para, ParaDrop, Para-RS | - | 0.181 |
| | 5-way to 3-way | C-Para, M-Para, ParaDrop, Para-RS | ✓ | **0.162**$^*$ |

corruption types into one or two classes in DC pre-training. Specifically, we reduced the best performing 5-way DC pre-training into (i) binary DC pre-training with original v.s. corrupted essays ({C-Para, M-Para, ParaDrop, Para-RS}), and to (ii) 3-way DC pre-training with original v.s. fully corrupted (C-Para) v.s. partially corrupted essays ({M-Para, ParaDrop, Para-RS}).

Table III demonstrates the results. It shows that transforming 5-way classification to binary classification performs worse than the baseline. We attribute this to combining fully corrupted (CPS) essays with partially corrupted (MPS, PD, PRSP) essays, so the model cannot distinguish between extremely bad and relatively bad essays. This hypothesis is solved when we transform it to a 3-way classification task. We obtain much better performance during finetuning, but the performance is not as good as the original 5-way classification task. Overall, these experiments indicate that differentiating between fine-grained corruption types is essential.

*2) Effectiveness of Corruption Pre-Training in Low Resource Setting:* To investigate how beneficial our DC pre-training is when labeled data is less available, we reduce the training data at the essay scoring phase. We examine the two best performing DC pre-trained models (4-way and 5-way classification) and compare them with the baseline model (model without DC pre-training). We select Baseline 1 for comparison since it has the best result among 2 baselines.

Fig. 8 shows a plot of number of training essays vs. MSE. MSE is obtained with all training data (703 essays) as well as with training data being reduced to $\frac{1}{2}$ (352 essays), $\frac{1}{4}$ (176 essays) and $\frac{1}{8}$ (88 essays). We observe that our proposed models constantly outperform the baseline model when we reduce the training data. This indicates both the strength and effectiveness of our DC pre-training with less information from labeled data and that the model understands which Organization structure is better than the others.

Our 4-way DC model (indicated via orange line) does not perform better than the 5-way DC model (green line). This result indicates that having more fine-grained corruption types in DC pre-training helps the model to be less dependent on the annotated information of which essay Organization is better.

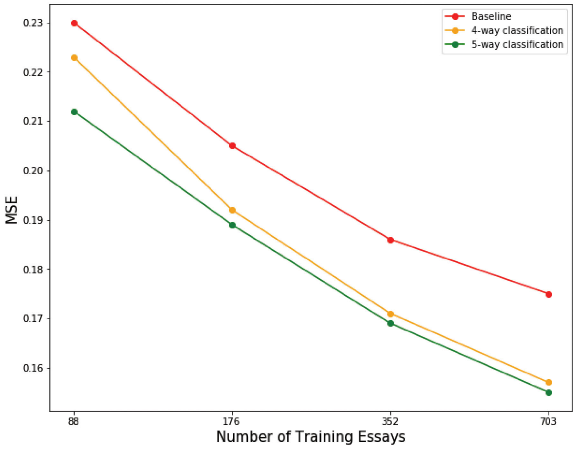


Fig. 8. Plot of training data vs MSE at essay scoring phase.

*3) Essay Embeddings:* In order to identify which scores are better distinguished by our models than the baseline model, we visualized essay embeddings (i.e. $\mathbf{h}^{\text{base}}$) obtained from the fine-tuned baseline model[8] and our proposed DC pre-trained (5-way classification) model.

The results are shown in Fig. 9. In the baseline model essay embeddings, the essays are scattered, and the low-scored essays (scored 1, red dots) are sometimes close to the high-scored essays (scored 4, blue dots) (upper-left of the figure). In contrast, the essay representations of our DC pre-training (5-way classification) shows that our model is good at separating essays of different scores and more cluster of scores appear compared to the baseline model. The highest scored (scored 4, blue dots) and the lowest scored (scored 1, red dots) essays are at the complete opposite position and furthest from each other in the embedding space. This means our model knows the difference between high scored and low scored Organization. We see that the lowest scored essays (red dots) are clustered and fully separated from other essays. Besides, other low scored essays (scored 1.5 and 2.0, lime and brown dots respectively) as well as highest scored essays (scored 4, blue dots) are also well

[8] We select Baseline 1 for the visualization of essay embeddings since it has the best result among 2 baselines

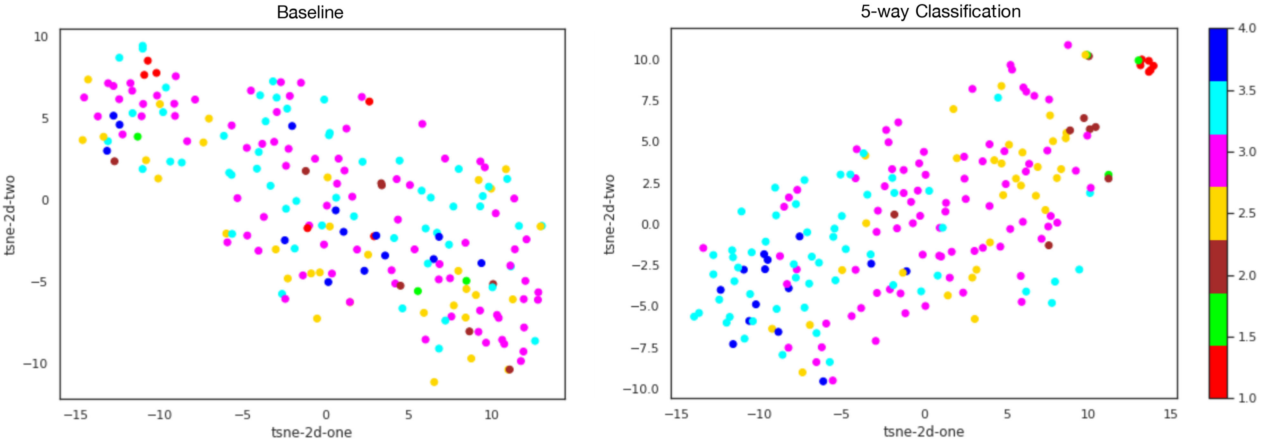


Fig. 9. Visualization of essay representations.

TABLE IV
SCORE PREDICTION OF TEST INSTANCES BY BASELINE AND OUR BEST DC PRE-TRAINED MODEL

| Gold Score | Baseline Predicted | 5-way Predicted | MSE (gold&baseP) | MSE (gold&5-wayP) |
|---|---|---|---|---|
| 1.0 | 2.3 | 1.1 | 1.69 | 0.01 |
| 2.5 | 1.2 | 2.1 | 1.69 | 0.16 |
| 2.5 | 3.7 | 2.5 | 1.44 | 0.00 |
| 2.5 | 1.4 | 2.5 | 1.21 | 0.00 |
| 1.0 | 2.3 | 1.7 | 1.69 | 0.49 |
| 2.0 | 3.3 | 2.9 | 1.69 | 0.81 |
| 2.0 | 2.9 | 2.4 | 0.81 | 0.09 |
| 4.0 | 3.1 | 3.6 | 0.81 | 0.16 |
| 4.0 | 3.2 | 3.7 | 0.64 | 0.09 |
| 1.5 | 2.5 | 2.2 | 1.00 | 0.49 |

distinguished. This represents that our model is not only good at separating bad Organization from good ones, but our model is also good at distinguishing different levels of "goodness" of essay Organization.

Table IV presents 10 test instances for which the prediction of our DC pre-trained model is better (i.e., lower MSE between gold and predicted score) than the baseline model. Column 1 shows the gold essay score, columns 2 and 3 show the scores predicted by the baseline model and our best DC pre-trained model (5-way classification) respectively.[9] Column 4 shows the MSE between the gold score and baseline predicted score, whereas column 5 presents the MSE between the gold score and the score predicted by DC pre-trained model. Table IV shows that our DC pre-trained model predicts low-to-medium and high essay scores well in comparison to the baseline. Observing the MSE difference between columns 4 and 5, one can see how better DC pre-trained model's prediction is in comparison to the baseline.

*4) Combining Different Pre-Training:* We have observed that (from Table II) N-ParaP pre-training improves the Organization scoring performance a bit although not as much as DC pre-training. In order to further analyse the effect of different pre-training, we have combined our DC pre-training with N-ParaP pre-training (e.g., first pre-train the model with the next paragraph prediction task and then pre-train it again with DC corruption strategies). For this combined pre-training task, we choose our best DC pre-trained model (5-way classification). However, The results in Table V show that combining paragraph level pre-training with document level DC pre-training doesn't perform very well, i.e., the proposed DC pre-training performs better without any additional local pre-training.

## VII. CONCLUSION

In this paper, we proposed an unsupervised pre-training strategy to capture discourse structure (i.e., coherence and cohesion) of essay Organization. We have presented various token, sentence, and paragraph level corruption techniques that produce several types of fully corrupted (totally incoherent/incohesive) or partially corrupted (partially incoherent/incohesive) essays. Then, we train a document encoder to discriminate between original essays and their artificially corrupted essays in order to make the encoder logical-sequence aware. Afterwards, the logical-sequence aware encoder is used to obtain feature vectors of essays for the task of essay Organization scoring. Our proposed pre-training strategy does not require any expensive parser or annotation. The experimental results show that the proposed method successfully captures the discourse structures of essay Organization, and we obtain a new state-of-the art result for essay Organization scoring. Our results also show that the combination of MLM pre-trained document encoder and paragraph level discourse corruption pre-training is effective for capturing the discourse of essay Organization. The combination of these two can handle both global and local coherence.

One possible future direction of this work is to determine how to exploit other unannotated argumentative texts (except student essays) for the proposed pre-training method. Since student essays are not perfect (i.e., can contain grammatical and/or spelling errors), it would be interesting to see how the proposed method behaves when pre-trained with perfectly written or error-less texts. We hope that our work inspires the exploration of new ways of unsupervised encapsulation of discourse structure in text representation.

[9] The predicted scores are shown to one decimal place.

TABLE V
ESSAY SCORING RESULTS OF 5-WAY DC PRE-TRAINING COMBINED WITH NEXT PARAGRAPH PREDICTION (N-PARAP) PRE-TRAINING

| Model | Classification Task | Corruption Type | Fine-tuning | Mean Squared Error Organization |
|---|---|---|---|---|
| Baseline 1 | - | - | - | 0.175 |
| | - | - | ✓ | 0.181 |
| Baseline 2 | Binary | N-SentP | - | 0.185 |
| | Binary | N-SentP | ✓ | 0.196 |
| Proposed | Binary | N-ParaP | - | 0.177 |
| | Binary | N-ParaP | ✓ | **0.172** |
| | 5-way | C-Para, M-Para, ParaDrop, Para-RS | - | **0.166**$^*$ |
| | 5-way | C-Para, M-Para, ParaDrop, Para-RS | ✓ | **0.155**$^*$ |
| | 5-way + Binary | (C-Para, M-Para, ParaDrop, Para-RS) + N-ParaP | - | 0.178 |
| | 5-way + Binary | (C-Para, M-Para, ParaDrop, Para-RS) + N-ParaP | ✓ | **0.173** |
| | Binary + 5-way | N-ParaP + (C-Para, M-Para, ParaDrop, Para-RS) | - | 0.181 |
| | Binary + 5-way | N-ParaP + (C-Para, M-Para, ParaDrop, Para-RS) | ✓ | **0.162**$^*$ |

TABLE VI
PRE-TRAINING RESULTS OF N-SENTP AND N-PARAP FOR THE TASK SETTING *SELECT-RANDOM-NEXT-FROM-RANDOM-DOC*

| Pre-training Objective | Classification Task | Validation Accuracy (first-step) | Validation Accuracy (second-step) |
|---|---|---|---|
| N-SentP | Binary | 0.914 | 0.878 |
| N-ParaP | Binary | 0.934 | 0.958 |

TABLE VII
ESSAY SCORING PERFORMANCE OF N-SENTP AND N-PARAP PRE-TRAINED MODELS IN THE TASK SETTING *SELECT-RANDOM-NEXT-FROM-RANDOM-DOC*

| Pre-training | Classification Task | Fine-tuning | Mean Squared Error Organization |
|---|---|---|---|
| N-SentP | Binary | - | 0.184 |
| N-SentP | Binary | ✓ | 0.196 |
| N-ParaP | Binary | - | 0.172 |
| N-ParaP | Binary | ✓ | 0.183 |

## APPENDIX
## ADDITIONAL DETAILS FOR NEXT SENTENCE AND PARAGRAPH PREDICTION PRE-TRAINING

For the next sentence/paragraph prediction (N-SentP/N-ParaP) tasks, when we select sentences/paragraphs A and B for each sentence/paragraph pair, 50% of the time B is the actual next sentence/paragraph that follows A and 50% of the time B is a random sentence/paragraph either chosen from the same document or from a random document in the corpora. If B is a random sentence/paragraph chosen from the same document, it means that the topic of the sentences/paragraphs A and B are the same. However, if B is a random sentence chosen from a random document in the corpora, the topic of the sentences A and B most likely would be different. The reported results in the Table I, Table II, Table III and Table V for N-SentP/N-ParaP are the results from the task setting where 50% of the time B is a random sentence/paragraph chosen from the same document. The results of N-SentP/N-ParaP tasks for the setting where 50% of the time B is a random sentence/paragraph chosen from a random document in the corpora (*select-random-next-from-random-doc*) is given in the Table VI and Table VII.

## ACKNOWLEDGMENT

We thank all the reviewers for their valuable suggestions and feedback.

## REFERENCES

[1] Y. Attali and J. Burstein, “Automated essay scoring with e-rater v. 2,” *J. Technol., Learn. Assessment*, vol. 4, no. 3, 2006.
[2] I. Persing, A. Davis, and V. Ng, “Modeling organization in student essays,” in *Proc. Conf. EMNLP*, 2010, pp. 229–239.
[3] I. Persing and V. Ng, “Modeling thesis clarity in student essays,” in *Proc. 51st Annu. Meeting Assoc. Comput. Linguistics*, 2013, pp. 260–269.
[4] I. Persing and V. Ng, “Modeling prompt adherence in student essays,” in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, 2014, pp. 1534–1543.
[5] I. Persing and V. Ng, “Modeling argument strength in student essays,” in *Proc. 53rd Annu. Meeting ACL 7th Int. Joint Conf. Natural Lang. Process.*, 2015, pp. 543–552.
[6] I. Persing and V. Ng, “Modeling stance in student essays,” in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics* (Volume 1: Long Papers), 2016, pp. 2174–2184.
[7] H. Wachsmuth, K. Al Khatib, and B. Stein, “Using argument mining to assess the argumentation quality of essays,” in *Proc. COLING, 26th Int. Conf. Comput. Linguistics: Tech. Papers*, 2016, pp. 1680–1691.
[8] S. Mathias and P. Bhattacharyya, “Thank “goodness”! a way to measure style in student essays,” in *Proc. 5th Workshop Natural Lang. Process. Techn. Educ. Appl.*, 2018, pp. 35–41.
[9] F. S. Mim, N. Inoue, P. Reisert, H. Ouchi, and K. Inui, “Unsupervised learning of discourse-aware text representation for essay scoring,” in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics: Student Res. Workshop*, 2019, pp. 378–385.
[10] M. Halliday, “*An Introduction to Functional Grammar:*” Hodder Arnold, 1994. [Online]. Available: https://books.google.co.jp/books?id=a88lnQEACAAJ
[11] R. Barzilay and M. Lapata, “Modeling local coherence: An entity-based approach,” *Comput. Linguistics*, vol. 34, no. 1, pp. 1–34, 2008.
[12] C. Unger, Genre, *Relevance and Global Coherence: The Pragmatics of Discourse Type*, ser. Palgrave Studies in Pragmatics, Language and Cognition. Palgrave Macmillan UK, 2006. [Online]. Available: https://books.google.co.jp/books?id=lkZ9DAAAQBAJ
[13] R. Zhang, “Sentence ordering driven by local and global coherence for summary generation,” in *Proc. ACL Student Session*, 2011, pp. 6–11.
[14] C. Stab and I. Gurevych, “Annotating argument components and relations in persuasive essays,” in *Proc. COLING, 25th Int. Conf. Comput. Linguistics: Tech. papers*, 2014, pp. 1501–1510.
[15] C. Stab and I. Gurevych, “Identifying argumentative discourse structures in persuasive essays,” in *Proc. 2014 Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 46–56.
[16] W. C. Mann and S. A. Thompson, “Rhetorical structure theory: Toward a functional theory of text organization,” Text, vol. 8, no. 3, pp. 243–281, 1988.
[17] Y. Ji and N. Smith, “Neural discourse structure for text categorization,” in *Proc. 55th Annu. Meeting Assoc. Comput. Linguist. (Volume 1: Long Papers)*, 2017, pp. 996–1005.
[18] A. Adhikari, A. Ram, R. Tang, and J. Lin, “Docbert: Bert for document classification,” 2019, *arXiv:1904.08398*.
[19] X. Zhang, F. Wei, and M. Zhou, “Hibert: Document level pre-training of hierarchical bidirectional transformers for document summarization,” in *Proc. 57th Annu. Meeting Assoc. Comput. Linguist.*, 2019, pp. 5059–5069.

[20] J. Xu, Z. Gan, Y. Cheng, and J. Liu, “Discourse-aware neural extractive text summarization,” in *Proc. 57th Annu. Meeting Assoc. Comput. Linguist.*, 2019, pp. 5059–5069.

[21] K. Steimel and B. Riordan, “Towards instance-based content scoring with pre-trained transformer models.”2020.

[22] J. Liu, Y. Xu, and Y. Zhu, “Automated essay scoring based on two-stage learning,” 2019, *arXiv:1901.07744*.

[23] F. Nadeem, H. Nguyen, Y. Liu, and M. Ostendorf, “Automated essay scoring with discourse-aware neural models,” in *Proc. 14th Workshop Innov. Use NLP Building Educ. Appl.*, 2019, pp. 484–493.

[24] A. Vaswani *et al.*, “Attention is all you need,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. 2019 Conf. North Amer. Chapter Assoc. Comput. Linguist.: Human Lang. Technol. Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[26] Y. Liu *et al.*, “A. robustly optimized bert pretraining approach,” 2019, *arXiv:1907.11692*.

[27] I. Beltagy, M. E. Peters, and A. Cohan, “Longformer: The long-document transformer,” 2020, *arXiv:2004.05150*.

[28] L. S. Larkey, “Automatic essay grading using text categorization techniques,” in *Proc. 21st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 1998, pp. 90–95.

[29] H. Chen and B. He, “Automated essay scoring by maximizing human-machine agreement,” in *Proc. 2013 Conf. Empirical Methods Natural Lang. Process.*, 2013, pp. 1741–1752.

[30] P. Phandi, K. M. A. Chai, and H. T. Ng, “Flexible domain adaptation for automated essay scoring using correlated linear regression,” in *Proc. 2015 Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 431–439.

[31] K. Taghipour and H. T. Ng, “A neural approach to automated essay scoring,” in *Proc. 2016 Conf. EMNLP*, 2016, pp. 1882–1891.

[32] D. Alikaniotis, H. Yannakoudakis, and M. Rei, “Automatic text scoring using neural networks,” in *Proc. 54th Annu. Meeting Assoc. Comput. Linguist. (Volume 1: Long Papers)*, 2016, pp. 715–725.

[33] F. Dong and Y. Zhang, “Automatic features for essay scoring-an empirical study,” in *Proc. 2016 Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 1072–1077.

[34] F. Dong, Y. Zhang, and J. Yang, “Attention-based recurrent convolutional neural network for automatic essay scoring,” in *Proc. 21st Conf. Comput. Natural Lang. Learn.*, 2017, pp. 153–162.

[35] B. Riordan, A. Horbach, A. Cahill, T. Zesch, and C. Lee, “Investigating neural architectures for short answer scoring,” in *Proc. 12th Workshop Innov. Use NLP Building Educ. Appl.*, 2017, pp. 159–168.

[36] Y. Farag, H. Yannakoudakis, and T. Briscoe, “Neural automated essay scoring and coherence modeling for adversarially crafted input,” in *Proc. 2018 Conf. North Amer. Chapter Assoc. Comput. Linguist.: Human Lang. Technol. Volume 1 (Long Papers)*, 2018, pp. 263–271.

[37] H. Zhang and D. Litman, “Co-attention based neural network for source-dependent essay scoring,” in *Proc. 13th Workshop Innov. Use NLP Building Educ. Appl.*, 2018, pp. 399–409.

[38] Y. Wang, Z. Wei, Y. Zhou, and X.-J. Huang, “Automatic essay scoring incorporating rating schema via reinforcement learning,” in *Proc. 2018 Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 791–797.

[39] R. Cummins and M. Rei, “Neural multi-task learning in automated assessment,” 2018, *arXiv:1801.06830*.

[40] D. Higgins, J. Burstein, D. Marcu, and C. Gentile, “Evaluating multiple aspects of coherence in student essays,” in *Proc. Human Lang. Technol. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2004, pp. 185–192.

[41] M. Mesgar and M. Strube, “A neural local coherence model for text quality assessment,” in *Proc. 2018 Conf. EMNLP*, 2018, pp. 4328–4339.

[42] F. S. Mim, N. Inoue, P. Reisert, H. Ouchi, and K. Inui, “Unsupervised learning of discourse-aware text representation,” 2019.

[43] Q. Le and T. Mikolov, “Distributed representations of sentences and documents,” in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 1188–1196.

[44] L. Wu *et al.*, “Word mover’s embedding: From Word2vec to document embedding,” in *Proc. 2018 Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 4524–4534.

[45] R. T. Ionescu and A. M. Butnaru, “Vector of locally-aggregated word embeddings (vlawe): A novel document-level representation,” in *Proc. NAACL-HLT*, 2019, pp. 363–369.

[46] V. Gupta, A. Saw, P. Nokhiz, P. Netrapalli, P. Rai, and P. Talukdar, “P-sif: Document embeddings using partition averaging,” in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 7863–7870.

[47] M.-W. Chang, K. Toutanova, K. Lee, and J. Devlin, “Language model pre-training for hierarchical document representations,” 2019, *arXiv:1901.09128*.

[48] M. E. Peters *et al.*, “Deep contextualized word representations,” in *Proc. 2018 Conf. North Amer. Chapter Assoc. Comput. Linguist.: Human Lang. Technol. Volume 1 (Long Papers)*, 2018, pp. 2227–2237.

[49] M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks,” *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[50] S. Granger, E. Dagneaux, F. Meunier, and M. Paquot, “International corpus of learner english,” 2009.

[51] D. Blanchard, J. Tetreault, D. Higgins, A. Cahill, and M. Chodorow, “TOEFL11: A corpus of non-native english,” *ETS Res. Rep. Ser.*, vol. 2013, no. 2, pp. i- 15, 2013.

[52] S. Ishikawa, “The icnale and sophisticated contrastive interlanguage analysis of asian learners of english,” *Learner corpus studies in Asia and the world*, vol. 1, , pp. 91–118, 2013.

**Farjana Sultana Mim** received the B.Sc. degree in computer science and engineering from Patuakhali Science and Technology University, Patuakhali, Bangladesh, in 2016 and the M.S. degree from Tohoku University, Sendai, Japan, in 2019. She is currently working toward the Ph.D. degree with Tohoku University, Sendai, Japan. Her research interests include essay scoring, unsupervised learning, discourse analysis, argumentation, and commonsense reasoning.

**Naoya Inoue** received the M.S. degree in engineering from the Nara Institute of Science and Technology, Ikoma, Japan, in 2010 and the Ph.D. degree in information science from Tohoku University, Sendai, Japan, in 2013. In 2013, he joined DENSO Corporation as a Researcher and since 2015, has been an Assistant Professor with Tohoku University. Since 2020, he has been a Postdoctoral Associate with Stony Brook University, Stony Brook, NY, USA. His research interests include explainable QA systems and neuro-symbolic reasoning.

**Paul Reisert** received the B.S. degree in computer science from Purdue University, West Lafayette, IN, USA, in 2010, and the M.S. and Ph.D. degrees in system information sciences from Tohoku University, Sendai, Japan, in 2017. He is currently a Postdoctoral Researcher with RIKEN, Tokyo, Japan. His research interests include argumentation mining, discourse analysis, and natural language processing.

**Hiroki Ouchi** received the B.A. degree from the Miyagi University of Education, Sendai, Japan, in 2011, the M.A. degree from Ritsumeikan University, Kyoto, Japan, in 2013, and the M.S. and Ph.D. degrees in engineering from the Nara Institute of Science and Technology, Ikoma, Japan, in 2015 and 2018, respectively. He is currently a Postdoctoral Researcher with RIKEN, Tokyo, Japan. His research interests include intersection of natural language processing and machine learning, in particular, structured prediction, and instance-based learning.

**Kentaro Inui** is currently a Professor with the Graduate School of Information Sciences, Tohoku University, Sendai, Japan, where he is also the Head of the Natural Language Processing Lab. He also leads the Natural Language Understanding Team with RIKEN Center for the Advanced Intelligence Project. His research interests include natural language processing and artificial intelligence. He is currently the Vice-chairperson of Association for Natural Language Processing, Member of Science Council of Japan, and Director of NPO FactCheck Initiative Japan.