

# Exploring the limits of Transfer learning with a Unified Text-to-Text Transformer

## 1. Introduction

## 2. Setup

### 2.1 Model

### 2.2. The Colossal Clean Crawled Corpus

### 2.3 Downstream Tasks

### 2.4 Input and Output Format

## 3. Experiments

### 3.1 Baseline

#### 3.1.1. Model

#### 3.1.2 Training

#### 3.1.3 Vocabulary

#### 3.1.4 Unsupervised Objective

#### 3.1.5 Baseline Performance

### 3.2 Architectures

#### 3.2.1 Model Structures

#### 3.2.2 Comparing Different Model Structures

#### 3.2.3 Objectives

#### 3.2.4 Results

### 3.3 Unsupervised Objectives

#### 3.3.1 Disparate High-level Approaches

#### 3.3.2 Simplifying the BERT Objective

#### 3.3.3 Varying the Corruption Rate

#### 3.3.4 Corruption Span

#### 3.3.5 Discussion

### 3.4 Pre-training Data set

#### 3.4.1 Unlabeled Data Sets

#### 3.4.2 Pre-training Data Set Size

### 3.5 Training Strategy

#### 3.5.1 Fine-tuning Methods

#### 3.5.2 Multi-task Learning

#### 3.5.3 Combining Multi-Task Learning with Fine-Tuning

### 3.6 Scaling

### 3.7 Putting it All Together

## 4. Reflection

### 4.1 Takeaways

### 4.2 Outlook

## 2.1. Model

- roughly equivalent to the original Transformer.
- removing the Layer Norm bias
- placing the layer normalization outside the residual path
- using different position embedding scheme.

## 2.4. Input & Output Format

- "text-to-text" format
- provides a consistent training objective both for pre-training & fine-tuning

## 3.1. Baseline

- to reflect modern, typical practice
- pre-train a standard Transformer using a simple denoising objective
- separately fine-tune on each of downstream task.

### 3.1.2 Training

- "inverse square root" learning rate schedule

$$1 / \sqrt{\max(n, k)}$$

current training iteration      number of warm-up steps (set to  $10^4$  in all experiments)

- constant learning rate of 0.01 for the first  $10^4$  steps
- then exponentially decays the learning rate until pre-training is over.

### 3.1.4. Unsupervised Objective

- mask consecutive spans of tokens & only predict dropped-out tokens  
→ reduce computational cost of pre-training

### 3.1.5. Baseline Performance

- pre-training provides significant gains across almost all benchmarks
- inter-run variance
  - GLUE & SuperGLUE : average of scores of each benchmark
  - CoLA, CB, COPA : ↑ inter-run variance...
  - harder to compare models using GLUE & SuperGLUE scores alone.